UNIVERSIDAD DE ALCALÁ DE HENARES

Facultad de Ciencias Ambientales

GENERACIÓN DE MÉTODOS BASADOS EN INTELIGENCIA ARTIFICIAL PARA EL ANÁLISIS DE DATOS MEDIOAMBIENTALES. APLICACIONES PRÁCTICAS

TESIS DOCTORAL

para la obtención del grado de doctor

Beatriz Díaz Gómez

Licenciada de Grado en CC. Geológicas

Madrid, 2005

UNIVERSIDAD DE
ALCALÁ DE HENARES
Facultad de Ciencias Ambientales
Departamento de Geología

CONSEJO SUPERIOR DE
INVESTIGACIONES CIENTIFÍCAS
Instituto de Automática Industrial
Departamento de sistemas

GENERACIÓN DE MÉTODOS BASADOS EN INTELIGENCIA ARTIFICIAL PARA EL ANÁLISIS DE DATOS MEDIOAMBIENTALES. APLICACIONES PRÁCTICAS

Autora:
Beatriz Díaz Gómez
Licenciada de Grado en CC. Geológicas

Directora:
Ángela Ribeiro Seijas *Dra. en CC Físicas*
Instituto de Automática Industrial (IAI) CSIC

Ponente:
Ramón Bienes Allas *Dr. Ing. Agrónomo*
Instituto Madrileño de Investigación Agraria y Alimentaria (IMIA) CSIC

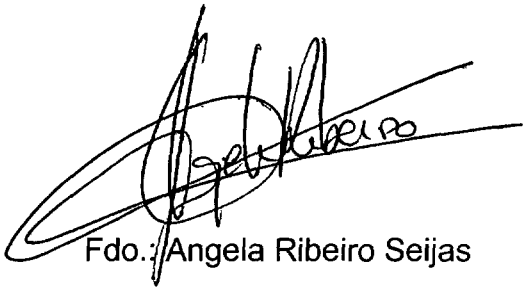
Tutor:
Antonio Sastre Merlín *Dr. en Geología*
Facultad de Ciencias (Dpto. de Geología) UAH

D. ANGELA RIBEIRO SEIJAS, Doctora en Ciencias Físicas e investigadora (CT) del Instituto de Automática Industrial (IAI) CSIC

CERTIFICA

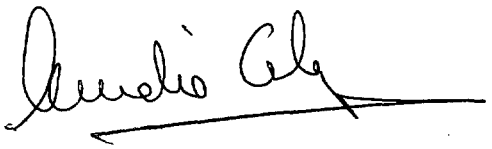
Que Beatriz Díaz Gómez, Licenciada en Ciencias Geológicas, ha realizado bajo mi dirección y asesoramiento el presente trabajo titulado "Generación de métodos basados en inteligencia artificial para el análisis de datos medioambientales. Aplicaciones prácticas", el cual considero que reúne las condiciones y la calidad científica deseada para su presentación con vista a optar al grado de Doctor en Ciencias Ambientales.

Y para que así conste, expido el presente certificaco en Alcalá de Henares, a veintiuno de julio de dos mil cinco.



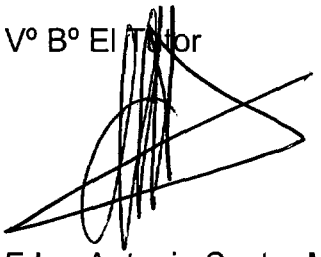
Fdo.: Angela Ribeiro Seijas

Vº Bº La Directora del Departamento



Fdo.: Amelia Calonge García

Vº Bº El Tutor



Fdo.: Antonio Sastre Merlín

Al caminante, ...

Caminante, son tus huellas
el camino, y nada más
Caminante, no hay camino,
se hace camino al andar,
al andar se hace el camino [...]
ANTONIO MACHADO:1875-1939

... a todos los que aportaron luz y energía a mi camino, ...

... y a mis padres por compartir su camino con continua devoción.

Gracias

Agradecimientos

Elaborar una tesis, como muchas otras cosas en la vida, no es un camino sencillo. Sin embargo, no podemos negar que es una experiencia apasionante y excitante. Cuantas veces habré estado en puntos que parecían no tener salida. Afortunadamente esta sensación asalta muy pocas veces, y la mayoría del tiempo se puede disfrutar mucho de la experiencia. Sin embargo, en general, el proceso requiere mucha energía, fuerza, motivación y constancia, y otros muchos elementos que hasta el momento desconoces tener. Por lo tanto, una vez finalizada alcanzas un tímido grado de satisfacción personal y orgullo al recapacitar sobre el curioso camino recorrido. ¿No es asombrosa esa capacidad de superación que tenemos los humanos?

A pesar de este exceso de soberbia, que espero comprendan, soy muy consciente de que es imposible conseguir este reto, sin el ánimo de instituciones, que fomentan y subvencionan este tipo de trabajos de investigación, y las personas que de diferentes formas nos ayudan a superar las dificultades que van surgiendo.

Es justo agradecer este apoyo, y con estas breves líneas quiero mostrar mi agradecimiento tanto a las instituciones que han financiado mi investigación, el *Ministerio de Educación y Ciencia* y el *Ministerio de Ciencia y Tecnología*, así como aquella donde además he realizado mi trabajo, el *Instituto de Automática Industrial* perteneciente al *Consejo superior de Investigaciones Científicas*. A los grupos y todas las personas de estas instituciones, que han confiado en mi capacidad para realizar el trabajo, que requería aprender y manejar nuevas disciplinas para lograr los objetivos propuestos en el marco de los proyectos que se mencionan a lo largo de la memoria. También a los equipos de trabajo que nos han facilitado la datos e información para poder desarrollar esta investigación, y que también se referencian debidamente dentro del texto.

Pero además durante todo este camino ha sido imprescindible el cariño y ánimo de mi pareja y de los amigos, de los que tenía antes y de los que han ido

apareciendo durante esta importante etapa, que aguantando días buenos, días malos o ausencias, me han ayudado y enseñado a caminar, mejor dicho a soñar por senderos desconocidos. Sin esta compañía y esa clase de amistad, no sólo esto no sería lo que es, sino yo misma no sería lo que soy hoy, por lo que ahora me resulta imposible dejar de compartir esta tímida felicidad al conseguir uno de aquellos sueños.

Igual de importante es el empuje que siempre proporciona la familia de forma incondicional, que empieza con las primeras reglas de la infancia y que nos ayuda a iniciar un camino desconocido. Reglas que de alguna forma nos modelan pero, al mismo tiempo e inevitablemente, nos permiten ser como somos. La familia, siempre inquieta, observando cada pasito, que desafortunadamente va menguando aunque también creciendo, y cuyo papel y valor he comprendido con el devenir del tiempo.

Me gustaría para acabar también dedicar unas líneas a todas esas personas que con su trabajo desinteresado han contribuido a la mejora paulatina de la situación laboral de muchos investigadores becarios, entre los que me he encontrado yo en este periodo.

Es muy fácil, por todas estas razones, daros a todos gracias por estar en estos importantes cinco últimos años de `mi camino`.

Resumen

En los últimos tiempos se ha puesto de manifiesto la gran importancia del análisis de datos con vistas a la búsqueda de modelos y a la inferencia de información nueva y relevante. En concreto, en ciencias medioambientales estas tareas de análisis son de especial importancia debido a la paulatina degradación ambiental que sufre nuestro entorno y que requiere actuaciones urgentes y de gran precisión.

La investigación que se presenta en este trabajo de tesis es el fruto de la integración de dos áreas de conocimiento bien conocidas; las áreas de *inteligencia artificial* y de *ciencias medioambientales*, con el objetivo de diseñar y desarrollar métodos de análisis o de inferencia de modelos que permitan explorar nuevos aspectos de los problemas medioambientales a partir de un conjunto de observaciones. Habitualmente estos problemas presentan una gran complejidad que limita, en muchos casos, la eficacia de las técnicas estadísticas de inferencia para la extracción de información o conocimiento. La metodología propuesta pretende ser una ayuda útil y complementaria a los estudios estadísticos. La memoria presenta todas las fases del diseño y del desarrollo de un sistema de extracción de conocimiento en bases de datos (*Knowledge Discovery Database* - KDD) que ha sido implementado teniendo en cuenta características propias de los datos y muestreos medioambientales. Entre las aportaciones principales se encuentra un sistema de inferencia de modelos que utiliza un procedimiento de aprendizaje automático, en concreto aprendizaje basado en ejemplos. El sistema genera modelos fácilmente interpretables ya que el conocimiento viene representado por un conjunto de reglas `Si-entonces`. En este sistema de inferencia de modelos se ha implementado un algoritmo genético como método de búsqueda de los mejores conjuntos de reglas que permite evitar la exploración sesgada del espacio de posibles soluciones (modelos) que presentan otros procedimientos de búsqueda. Además como parte del sistema KDD desarrollado, se ha implementado una herramienta de ayuda a la recogida georeferenciada

de datos en campo que los almacena, en tiempo real, en una base de datos relacional con un formato que permite el tratamiento posterior de la información almacenada con un Sistema de Información Geográfica.

El conjunto de herramientas desarrolladas se aplican a un problema medioambiental; el *control de malas hierbas en sistemas agrícolas*, una de las líneas centrales de la denominada *agricultura de precisión*, área que desde las perspectivas ecológica y económica busca una gestión óptima de los productos agroquímicos empleados en los tratamientos fitosanitarios. En concreto el análisis que se presenta en la memoria va encaminado a la obtención, a partir de un conjunto de datos, de modelos basados en reglas que expliquen, en función de parámetros ambientales y para un mismo campo, la existencia de una mayor cantidad de malas hierbas en unas zonas del cultivo frente a otras. El conocimiento incluido en los modelos extraídos aporta información de utilidad que puede plasmarse en un mapa de riesgo que permita asesorar en la aplicación precisa de herbicida sólo en las zonas del cultivo que lo requieran y en una dosis ajustada a cada situación de infestación. Los datos utilizados para la obtención de los modelos provienen de varias parcelas de cereal de invierno situadas en la Comunidad de Madrid y en la provincia de Barcelona y de dos tipos de mala hierba (*Avena sterilis L.* y *Lolium rigidum G.*). Asimismo, los conjuntos de reglas obtenidos con la metodología propuesta se han contrastado con los modelos generados, para el mismo conjunto de datos, con algoritmos comerciales como C&RT y C5.0, dando como resultado una mejora en la calidad de los modelos inducidos con los métodos desarrollados, es decir que nuestros modelos describen con mayor exactitud y confianza las observaciones de partida.

ABSTRACT

Recently, data analysis has made a great impact in the search for models and the inference of new and relevant information. These analytical tasks are especially important in the environmental sciences because of the slow ecological degradation of our environment, which needs to be addressed by urgent action based and high-precision analyses.

The Ph.D. work presented here is based on the combination of two well-known areas of knowledge: Artificial Inteligence and Environmental Science. The goal of this study was to design and develop methods for analyzing data and inferring models that allow people to explore new aspects of these environmental problems based on observations. Commonly, these problems are very complex, and in many cases this complexity makes it difficult to use techniques of statistical inference for knowledge discovery. The proposed methodology is intended to be a useful and complementary aid to the use of statistical studies. The text describes every stage of design and development of a system of Knowledge Discovery Database (KDD). The implementation of the system was based on the characteristic features of environmental data and samples. The main novel contribution is a system to infer models using a Machine Learning procedure, specifically, examples-based learning. The models generated by the system are easily interpretable, because the knowledge is expressed as a set of **If-Then** *rules. The Machine Learning procedure that searches for best rule sets is a genetic algorithm, which avoids the biased exploration of possible solutions (models), which is common to other search methods. In addition, as part of the developed KDD system, a tool has been implemented that aids in field sampling tasks, allowing people to gather and store georefenced data in real time in a spatial database, which can then be managed by a Geographic Information System.*

The set of developed tools was used to study a specific environmental problem: site specific weed management of agriculture systems, *which is a main line of research in Precision Agriculture (PA). From a ecological and economical perspective, PA looks for an optimal management of the chemicals used for weed and crop management. The analysis presented here is directed toward the discovery, based on sets of data, of rule-based models that explain in terms of environmental parameters the uneven distribution of weeds in winter cereal crop fields.*

The knowledge gained from the discovered models gives information that can be used to create risk maps. These maps would allow a selective and appropriate

application of herbicide only to those cultivated areas where weeds might appear.

Data for this study came from several fields of cereal located in the province of Madrid and the province of Barcelona, both of which had infestations of Avena sterilis L. In the Barcelona field Lolium rigidum G was also present.

Moreover, the discovered rule sets using the proposed methodology were compared with models generated by comercial algorithms (C&RT y C5.0) using the same data sets. This comparison demonstrated that the tool presented in this research discovered models with higher quality (accuracy and/or confidence) than did the comercial tools.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

Capítulo 1

INTRODUCCIÓN

En las últimas décadas temas como la contaminación, la conservación ambiental, el control biológico y medioambiental han puesto de manifiesto la importancia de la ecología y, como consecuencia, se ha incrementado la necesidad de conocer el medio ambiente y analizar los (sub)sistemas ecológicos que lo conforman. Este interés, fundamentalmente de la comunidad científica y en menor medida de los gobiernos presionados por la opinión pública, viene impulsado principalmente por el deterioro que está sufriendo el medio ambiente, que aconseja la monitorización y el control de las alteraciones que provocan estos cambios a diferentes escalas, incluso cambios a escala global. Para el estudio de los problemas medioambientales es necesario un análisis de los datos ecológicos que permita la inferencia del conocimiento indispensable para tomar la decisión más apropiada en cada situación, logrando con ello una gestión más eficaz [GUNTHER, 1998]. La extracción de conocimiento a partir de datos se lleva a cabo mediante de técnicas inductivas que permiten simplificar la realidad representándola por medio de *modelos* imprescindibles para comprender los fenómenos del entorno que nos rodea. Este proceso de análisis también se conoce como *prospección de datos* o *data-mining.*

Dentro del grupo de técnicas inductivas se encuentra la inferencia estadística que es una herramienta poderosa para la generación de modelos, no aplicable en todos los casos y que genera con frecuencia modelos de difícil interpretación. Esto motiva que el desarrollo de nuevas herramientas y métodos, capaces de encontrar nuevos modelos fáciles de interpretar, sea un objetivo abierto de gran

trascendencia para el conjunto de la comunidad científica [SÀNCHEZ-MARRÈ ET AL., 2004]. Así, los sistemas de descubrimiento automático de conocimiento (KDD o *Knowledge Discovery Database*) determinan una de las áreas actuales de investigación más activas dentro del análisis inteligente de datos. Un proceso KDD tiene por objetivo la extracción de nueva información, es decir conocimiento válido y comprensible a partir de grandes volúmenes de datos. Además un proceso KDD puede inducir diferentes tipos de modelos porque abarca diferentes técnicas de inferencia como el agrupamiento automático de datos (*clustering*), el descubrimiento de patrones, la detección de anomalías o el análisis de cambios y de tendencias [FAYYAD ET AL., 1996b]. Relacionado con el desarrollo de técnicas KDD se encuentra el aprendizaje automático marco en el que se encuadra la presente tesis.

Por otra parte, en los últimos años los avances tecnológicos han permitido la adopción de técnicas innovadoras en el campo de la agricultura, aumentando la rentabilidad económica y reduciendo el impacto medioambiental. Así surge la denominada *agricultura de precisión* que engloba tecnologías y prácticas encaminadas a minimizar el uso de productos agroquímicos asegurando un control efectivo de plagas, malas hierbas y enfermedades, a la vez que se suministra una cantidad de nutrientes adecuada a los cultivos [STTAFORD & MILLER, 1993, KROPFF ET AL., 1997]. Dentro de este área resulta clave el diseño de programas de tratamiento específico con herbicidas para lo que es imprescindible tener un buen conocimiento de los aspectos ecológicos que influyen en la existencia y la dinámica espacio-temporal de malas hierbas.

Esta tesis se enmarca dentro de las líneas de investigación del Instituto de Automática Industrial del Consejo Superior de Investigaciones Científicas, en concreto en la línea de *inteligencia artificial y sus aplicaciones* y en las sublíneas de *extracción de conocimiento, medio ambiente* y *agricultura de precisión.* El trabajo ha tenido como soporte dos proyectos de investigación del Plan Nacional de I+D en el área agraria: AGF99-1125-C03-03 "*Sistema KDD de apoyo a la toma de decisiones para control de malas hierbas basándose en mapas de riesgos (GEA I)*" y AGL2002-04468-C03-01 "*Visión artificial*

y razonamiento espacio-temporal para tratamientos localizados con un tractor autónomo (GEA II)". En estos proyectos además del Instituto de Automática Industrial (IAI) han participado entre otros el departamento de protección vegetal del Centro de Ciencias Medioambientales también del CSIC (CCMA) y el departamento de Biología Vegetal de la universidad de Barcelona (UB).

El objetivo de este trabajo de tesis involucra los conocimientos de diferentes disciplinas ya que aborda, por una parte, *el diseño y desarrollo de un sistema genérico basado en técnicas de aprendizaje automático para la inducción de modelos descriptivos expresados como un conjunto de reglas* y, por otra parte, la aplicación del sistema desarrollado a la generación de un modelo descriptivo que permita explicar la mayor o menos abundancia de malas hierbas en términos de factores edáficos u otras características ambientales, todo esto referido a cultivos de cereal de invierno. Este objetivo se complementa con un control total sobre cada una de las etapas que llevan a la obtención del modelo, en concreto la adquisición de datos, la preparación de los datos y la evaluación y visualización de los modelos generados. En consecuencia este objetivo primario da lugar al siguiente conjunto de subobjetivos:

- ⊙ Estudio del proceso KDD o de descubrimiento de conocimiento, haciendo especial énfasis en todos los aspectos relacionados con la aplicación de técnicas de aprendizaje automático a la extracción de conocimiento o generación de modelos.

- ⊙ Diseño y desarrollo de una herramienta de adquisición de datos georeferenciados en campo siguiendo un plan de muestreo. Este subobjetivo requerirá el diseño de una base de datos para entidades espaciales que pueda ser gestionada con un SIG (Sistema de Información Geográfica) para almacenar los datos directamente en el campo y en tiempo real, y posteriormente, también guardar otro tipo de información como por ejemplo los resultados de los análisis químicos de laboratorio.

- ⊙ Estudio del dispositivo de localización (receptor DGPS) que se utilizará en el campo para conocer la exactitud y precisión de la medida y

adaptar las características de la herramienta de adquisición de modo que se detecten y suavicen, en la medida de lo posible, fallos en la localización.

- ⊙ Diseño y desarrollo de una herramienta de preprocesamiento de los datos que ayude en la labor de selección y limpieza de los mismos, preparándolos para la etapa de análisis o generación de modelos.
- ⊙ Diseño y desarrollo de una herramienta de generación de modelos descriptivos basados en reglas a partir de un gran número de observaciones o datos.
- ⊙ Diseño y desarrollo de una herramienta de ayuda a la evaluación, visualización e interpretación de los modelos generados.
- ⊙ Estudio del problema de malas hierbas en agricultura desde una perspectiva de gestión precisa.
- ⊙ Aplicación de la metodología propuesta a un caso de estudio en el contexto del control de malas hierbas.

El trabajo llevado a cabo en esta tesis se describe en la presente memoria en 12 capítulos y 6 anexos según la siguiente estructura:

[**Parte I**] REVISIÓN DE CONOCIMIENTOS.

- **Capítulo 2** - Este capítulo introduce conceptos básicos relacionados con la construcción de modelos describiendo concisamente algunos de los métodos utilizados en la generación de modelos en medio ambiente.
- **Capítulo 3** - En este capítulo se presenta el concepto de *sistema de descubrimiento de conocimiento* (KDD) y se analizan sus diferentes etapas para posteriormente presentar una perspectiva de estos sistemas en el marco del aprendizaje automático. Se ahonda en la representación del conocimiento descubierto mediante reglas `Si-Entonces` y se analizan los aspectos del lenguaje SQL más relevantes para la transformación de reglas en consultas sobre una base de datos.
- **Capítulo 4** - El mecanismo de inducción de modelos descriptivos basados en reglas propuesto en esta tesis tiene como base un algoritmo genético. En este capítulo se presentan los fundamentos de los algoritmos genéticos y se muestra la aplicación de estos en la inducción de modelos basados en reglas.

[**Parte II**] PROPUESTA: Implementación de un sistema para descubrimiento de conocimiento

- **Capítulo 5** - En este capítulo se describe en detalle la herramienta diseñada y desarrollada para la adquisición de datos georeferenciados en el campo.
- **Capítulo 6** - Este capítulo analiza los diferentes problemas que pueden presentar los datos de entrada así como las formas de solventar los mismos. El capítulo termina con la descripción de la herramienta diseñada y desarrollada para ayudar al usuario en las tareas de preparación de los datos.

- **Capítulo 7** - En este capítulo se describe la herramienta desarrollada para de generación de modelos basados en reglas. Además se detalla el procedimiento de búsqueda del mejor modelo construido según el paradigma de un algoritmo genético y se muestra la utilización del lenguaje SQL como mecanismo de consulta a la base de datos en el proceso de inducción y verificación de las reglas.

- **Capítulo 8** - Este capítulo explica el diseño e implementación de la herramienta de evaluación que incorpora capacidades para la visualización en gráficos de la calidad de los modelos.

[**Parte III**] CASO DE ESTUDIO: El control de malas hierbas:

- **Capítulo 9** - En el capítulo se reseñan las principales líneas de actuación de la agricultura de precisión haciendo énfasis en el control de malas hierbas. En este capítulo además se analiza la importancia de los factores edáficos en el desarrollo del cultivo.

- **Capítulo 10** - El capítulo presenta los datos con los que se realizará la experimentación y pormenoriza la etapa de preprocesamiento llevada a cabo sobre los mismos.

- **Capítulo 11** - Este capítulo está dedicado a analizar los resultados obtenidos al utilizar la herramienta desarrollada para inducir un modelo descriptivo basado en reglas a partir de los datos descritos en el capítulo anterior. Asimismo sobre la misma base de los datos se compara el rendimiento de la propuesta frente a otras soluciones comerciales, en concreto los algoritmos de extracción de reglas que suministra el sistema Clementine© de SPSS.

- **Capítulo 12** - El último capítulo de la memoria resume las aportaciones más importantes de esta tesis y comenta las principales líneas de investigación futuras.

[**Anexos**] APÉNCICES.

- El apéndice A presenta los valores estadísticos de los datos del campo de Barcelona.
- El apéndice B muestra los mapas de la variación espacial de las propiedades edáficas utilizadas (pH, materia orgánica, textura, nitrógeno, potasio, etc) de los campos de Madrid y del campo de Barcelona.
- El apéndice C presenta las tablas que contienen los umbrales utilizados en la categorización de los rangos numéricos de los datos del campo de Barcelona.
- El apéndice D incluye los informes obtenidos por la herramienta de preparación de los datos tras la etapa de limpieza.
- El apéndice E presenta los archivos `fit` que se utilizan de los diferentes experimentos realizados con el sistema propuesto.
- El apéndice F muestra los resultados obtenidos por los algoritmos C5.0 y C&RT que incluye Clementine© para la obtención de reglas.
- El apéndice G presenta los resultados obtenidos por la herramienta propuesta en esta tesis, es decir el sistema compuesto (`AGlearner + SQLdeco`).

Parte I

Revisión de conocimientos

Capítulo 2

Modelos y principales áreas de investigación

2.1. Modelos en Ecología

El uso de modelos juega un papel fundamental en las ciencias medioambientales, al existir multitud de procesos complejos que deben ser formulados con aproximaciones fiables que permitan la predicción de situaciones, por las ventajas socio-económicas que puede tener conocer o predecir la alteración y modificación del medio ambiente.

Esto conlleva la medición y registro de variables así como su representación y modelado a fin de comprender el comportamiento del sistema en estudio y predecir sus cambios. Sin embargo, estos sistemas son de difícil análisis debido fundamentalmente al gran número de variables y procesos que se hayan involucrados en los sistemas naturales, lo que dificulta la capacidad de predecir el riesgo ecológico, a medio e incluso, a corto plazo. La excesiva información que se puede obtener de estos sistemas complejos exige el desarrollo de metodologías analíticas o heurísticas capaces de abordar la complejidad inherente al sistema y que permitan la extracción de conocimiento relevante del mismo a partir del análisis de grandes volúmenes de datos.

Se entiende por *modelo* cualquier representación de un sistema o proceso, utilizando tanto mecanismos verbales o simbólicos, como matemáticos o físicos, comprensibles para el humano.

Esta definición da una idea de la gran utilidad que poseen los modelos,

empleados consciente o inconscientemente en nuestra continua interacción con el entorno, para poder expresar y comprender la realidad que nos rodea de forma simplificada. Su formulación comienza con aproximaciones con un alto grado de simplificación y/o realizando divisiones del sistema o proceso, analizando las partes que lo constituyen por ser más fácilmente comprensibles. Los modelos de los sistemas que integran el mundo, constituyen una herramienta psicológica para comprender y abordar la complejidad del mismo.

Los modelos se aproximan en función del área de aplicación, y por ello un modelo puede ser una simple explicación con términos del lenguajes natural, de los aspectos fundamentales de un proceso (*modelos verbales*), o bien, un conjunto de ecuaciones matemáticas para la representación tridimensional de objetos. En ingeniería, son frecuentes los diagramas de flujo de datos y procesos matemáticos, que pueden ser muy complejos, utilizando un conjunto de figuras geométricas y flechas, que permiten seguir la evolución de los procesos y las interacciones entre los mismos con una aproximación muy intuitiva. En las ciencias naturales, los modelos pueden ser esquemas que permitan la compresión del funcionamiento de un proceso natural (*modelos gráficos*). Aunque existen tantos tipos de modelos como sistemas o procesos, en general estos se suelen clasificar atendiendo a las siguientes características generales:

- *Escala.* Según la definición de HOLLING [2001], los modelos que pueden explorar hipótesis generales se denominan *estratégicos* y los que contestan a cuestiones específicas, se denominan modelos *tácticos.*

- *Dinámica o estabilidad.* Un *modelo determinista* es aquel del que se obtiene la misma respuesta en diferentes instantes de tiempo. Por el contrario, los modelos *estocásticos* dependen del error aleatorio debido a la variabilidad natural. Cuando se quieren evaluar propiedades de estado, no hay razón para incluir información estocástica. Sin embargo, cuando el objetivo es determinar la probabilidad de un evento es frecuente recurrir a un modelo estocástico que determinen la frecuencia con la sucede dicho evento.

- *Complejidad.* Los modelos *analíticos* se construyen con un número limitado y reducido de variables, mediante funciones analíticas. Sin embargo, si se aumenta el número de variables, resulta extremadamente difícil o incluso imposible formular analíticamente el modelo y entonces se utilizan las *simulaciones* o bien los modelos *aproximados* que son más flexibles.

Los modelos son imprescindibles en cualquier área de conocimiento. En primer lugar, porque es un referente de ayuda ante sistemas complejos y permite predecir situaciones futuras con un determinado grado de aproximación. Los modelos permiten verificar hipótesis y repetir experimentos de situación con baja probabilidad de ocurrencia en el mundo real. Esto último es una característica fundamental para la construcción de modelos idóneos, ya que es deseable que incluyan la dinámica del sistema [MAXWELL, 1999]. En definitiva, un modelo constituye un resumen del funcionamiento de un aspecto del mundo, que permite extraer nuevo conocimiento del sistema y de su interacción con el entorno, y de ahí que constituya el punto central de esta tesis.

Ahora bien, en la generación de modelos hay que tener en cuenta las limitaciones, ya que un modelo no es el sustituto de la realidad. Por lo tanto, independientemente de la calidad del modelo, siempre se está a una gran distancia de poder representar la complejidad completa del proceso natural en estudio. En tareas de modelización se asume que a pesar de que el modelo perfecto de un aspecto del mundo existe, es casi imposible formularlo, por lo que la descripción del mismo siempre es una aproximación con un cierto grado de incertidumbre de la descripción ideal [STUCKENSCHMIDT, 1999].

2.2. PRINCIPALES ÁREAS IMPLICADAS EN LA OBTENCIÓN DE MODELOS EN MEDIO AMBIENTE

El modelado analítico ha sido hasta la fecha el más utilizado en las múltiples disciplinas científicas y su formulación es tan amplia como lo es el espectro de expresiones analíticas que constituyen su base. Esta variabilidad se debe a que estos modelos dependen del dominio de aplicación y de los objetivos que se persiguen. En concreto, según SMITH [1974], las motivaciones en el ámbito ecológico para la creación de modelos analíticos, obedecen tanto a consideraciones de tipo práctico como teórico. Con respecto a la primera consideración, los modelos se utilizan para la predicción del comportamiento del sistema real mediante *simulaciones*; de gran utilidad en las etapas de gestión y toma de decisiones relativas al sistema. Estas *simulaciones* son implementaciones guiadas por el objetivo de buscar descripciones generales de sistemas específicos. El gran desarrollo de los procesadores y de la ciencia de la computación ha hecho posible manejar los grandes volúmenes de datos requeridos para la generación de simulaciones realistas, mediante el uso de programas de ordenador que imitan el funcionamiento de los sistemas complejos y producen modelos de los procesos. Este tipo de modelos se emplean en numerosas aplicaciones medioambientales como son, el calentamiento global del planeta, los modelos atmosféricos y oceánicos, aplicaciones biológicas para analizar la dinámica de poblaciones, en hidrología y edafología para estimar la dispersión de contaminantes, en agricultura para predecir la evolución de plagas, rodales de malas hierbas etc. Por ello, existen ya algunas herramientas comerciales capaces de abordar algunos problemas en simulación. Entre ellas destacan: *Fluent*©, *Stella*©, *Simulink*© combinado con MATLAB (*TheMathWorks*©), *VISsim*© o *Extend*©.

En el segundo lugar, se encuentran las *descripciones teóricas* que mediante modelos analíticos o heurísticos, buscan formulaciones generales, denominadas *teorías*, para todo un sistema y que deben ser capaces de englobar los casos particulares previamente estudiados. Cada teoría consiste en un grupo de mo-

delos, donde algunos pueden ser alternativos o complementarios de otros, que contemplan un conjunto amplio de situaciones y permiten la comprensión de la naturaleza de los procesos subyacentes; de ahí la enorme importancia de estos modelos teóricos [COOPER, 1996]. Los modelos teóricos deben ser tan generales y sencillos como sea posible, sin incluir detalles relativos a situaciones específicas, al contrario que las simulaciones que son descripciones ricas en detalles. Actualmente, los modelos teóricos por razones obvias de complejidad, no pueden ser inferidos a partir de la información relativa a todo el ecosistema, por lo que el objetivo suele ser observar las diferencias que existen entre los modelos o aproximaciones obtenidas de los diferentes sub-sistemas que lo componen. En la literatura, se describen modelos teóricos diseñados para la predicción de procesos ecológicos. Por ejemplo, los desarrollados específicamente para la ecología como los incluidos en los paquetes informáticos Ecowin [FERREIRA, 1995], el sistema de modelización y simulación ECOBAS (2004) y el servicio REM (*Register of Ecological Models*) que ofrece una meta-base de datos que almacena distintos modelos matemáticos[1]. En los sistemas agrícolas, caracterizados también por multiples variables de entrada que poseen un comportamiento imprevisible, el modelado es normalmente empírico y existen muy pocas aproximaciones teóricas. A pesar de su complejidad, se han desarrollado ya algunos modelos teóricos para su aplicación en agricultura [VOHNOUT, 2003].

Finalmente, para tareas concretas de modelado de la distribución de especies se encuentran modelos teóricos relativamente sencillos, basados en correlaciones de ocurrencia de especies en función de factores ambientales, como el sistema BIOCLIM de Busby (1991) que utiliza información climática, o bien el sistema GLIMS de Austin et al. (1990) que realiza regresiones logísticas, ambos trabajos referenciados en [STOCKWELL, 1999]. Estudios de evolución de especies o dinámica de poblaciones constituyen el objetivo de ciencias como la cibernética -combinación de la biología teórica, matemáticas y sistemas artificiales de proceso- que ha experimentado un interesante avance debida a la

[1] Universidad de Kassel y el GSF National research center for Environment and Health (Alemania). *http://eco.wiz.uni-kassel.de/*

necesidad de representar el funcionamiento de los sistemas ecológicos mediante modelos analíticos [LEVINS, 1968].

Además del modelado analítico, existen otras tres grandes áreas dedicadas a la generación de modelos y a la inferencia de conocimiento relevante: *inferencia estadística*, los *sistemas de gestión de bases de datos* (DBMS) y el *descubrimiento de conocimiento* basado en técnicas de *aprendizaje artificial*. Aunque son áreas claramente diferentes, es difícil establecer los limites entre estas categorías, debido a sus interrelaciones. Por ejemplo, la estadística depende de los desarrollos matemáticos; los sistemas de extracción de conocimiento recurren también a las matemáticas o a la estadística para representar la calidad y validar los patrones [FLACH, 2001], que suelen ser probabilísticos o estadísticos; y también los sistemas de gestión de bases de datos están muy vinculadas a los métodos automáticos de descubrimiento de conocimiento, intercambiando con ellos diversos métodos de análisis.

La *inferencia estadística*, tradicionalmente utilizada para el análisis teórico y para el modelado de datos, se define como el conjunto de métodos estadísticos que permiten realizar una inferencia sobre la distribución de la población en estudio y establecer relaciones entre algunas de las variables implicadas, con el fin de obtener un modelo de probabilidad a partir de la información que proporciona una muestra representativa. En concreto, la inferencia estadística inductiva busca parámetros válidos para una población a través de la información de una muestra de la misma, y al estar basada en medidas de probabilidad, proporciona resultados probables, pero no exactos como son los derivados de la inferencia estadística deductiva. Las técnicas estadísticas fundamentales son la *estimación de parámetros* y el *contraste de hipótesis*. La estimación de parámetros se hace mediante valores aproximados y utiliza una función adecuada, llamada *estimador*, que se elige atendiendo a ciertos requisitos de consistencia, suficiencia y eficiencia, mientras que, el contraste de hipótesis permite comprobar si la información que proporciona una muestra observada concuerda (o no) con la hipótesis estadística formulada sobre un modelo de probabilidad, entendiendo por *hipótesis estadística*, cualquier conje-

tura sobre una o varias características de interés de un modelo de probabilidad [RUIZ-MAYA, 2000], y puede ser o no paramétrica. Para realizar *inferencia estadística paramétrica* se parte de la función de distribución de la variable objeto de estudio, que es aleatoria, y se estiman los parámetros que la determinan. En el caso de la *inferencia estadística no paramétrica*, no se conoce *a priori* la distribución de la variable aleatoria objeto de estudio.

Existen diferentes métodos estadísticos, cuya elección depende del objetivo que se persigue y del tipo de datos y se pueden clasificar en: a) *globales* como los índices de agregados basados en varianza o la media; b) estudios de *periodicidad* como el análisis de espectros (frecuencias), fractales o análisis de onda (*wavelet*); c) estudio de rango o *intensidad espacial* como la autocorrelación con índices como función k de *Ripley*, los índices *SADIE* o la *semivarianza*; y d) la *interpolación* realizada con *krigeado* (*kriging*) o los polígonos de *Voronoi*.

Entre todas estas técnicas destaca por su popularidad, la autocorrelación que utiliza el coeficiente de semivarianza [HEVESI ET AL., 1992], base de la *estadística espacial*, comúnmente denominada *geoestadística*. La autocorrelación se basa en los métodos matemáticos de cálculo de la varianza o de conteo. Fundamentalmente, la geoestadística ha sido utilizada para mejorar la habilidad para detectar patrones espaciales [FORTIN ET AL., 2002], así como para la creación de cartografía a partir de los datos de una muestra. Concretamente en el campo de la agricultura, la geoestadística es la técnica más repetida para la búsqueda de factores que están asociados a la aparición de ciertas especies, y existen numerosos trabajos que utilizan este método, [DONALD, 1994, CARDINA ET AL., 1995, WALTER ET AL., 2002, MARY ET AL., 2001, KERRY & OLIVER, 2001, JURADO-EXPÓSITO ET AL., 2002, MIAO ET AL., 2003, BARROSO, 2004]. Los semivariogramas -gráficos de la correlación espacial en función de la distancia- combinados con un método de interpolación como el *krijeado* permiten conocer la relación entre la distribución espacial de diferentes especies. En el mismo marco de investigación, además de los semivariogramas, se utilizan otras técnicas estadísticas. Por ejemplo, DIELEMAN ET AL. [2000a], OFFICER ET AL. [2003] presentan estudios utilizando el *análisis de correlación canóniga*

que también se basa en coeficientes de la varianza, que permite determinar el grado de asociación de dos grupos de atributos. Otra técnica es el análisis de krijeado multivariante [BOURENNANE ET AL., 2003] utilizado para estudios de corregionalización entre las propiedades del suelo y la abundancia de un determinado cultivo, así como la separación de las variables que causan la variación dependiendo de la escala espacial. Finalmente, la variabilidad espacial de las propiedades del suelo y su asociación con el rendimiento del cultivo se han analizado también mediante técnicas de regresión múltiple, quizás uno de los métodos analíticos más sencillos, combinadas con métodos de interpolación [MELCHIORI ET AL., 2001].

Numerosos trabajos han demostrado que la estadística es una herramienta potente, pero en ocasiones presenta restricciones relacionadas con el control sobre el tipo o el volumen de datos experimentales [DEGROOT, 1988], limitando su utilización. Otro factor que dificulta su uso, aunque no lo limita, es el estar sujetas a la decisión e interpretación, y que en determinados casos depende del conocimiento, la experiencia y también de la forma en la que el analista describe el problema, es decir, en muchas ocasiones la interpretación de los resultados numéricos, pobres en detalles y difícilmente comprensibles, es muy subjetivo. Incluso en etapas previas, la intuición del investigador es fundamental, al tratarse de una metodología dirigida, para determinar *a priori* las variables a considerar en el modelo.

Estas desventajas, han potenciado la evolución de las técnicas estadísticas hacia otros métodos de inferencia que sean más flexibles con la distribución de los datos o capaces de discriminar variables no relevantes durante el propio proceso de análisis, como son los métodos basados en heurísticos o en inteligencia artificial. Estas técnicas que se conocen como de *aprendizaje automático* permiten encontrar patrones de comportamiento complejo en los datos analizados, debido fundamentalmente a que emplean una noción más sofisticada de modelo y mecanismos de aprendizaje que involucran tanto la búsqueda del modelo como la estimación de parámetros. De hecho, con el fin de evitar o paliar algunas de las desventajas de los modelos estadísticos, se han incorporado

técnicas de aprendizaje incluso en herramientas estadísticas. Un ejemplo reciente es el método de *selección de modelos* [JOHNSON & OMLAND, 2004], que se basa en la generación iterativa de diferentes hipótesis o modelos candidatos, que compiten en la búsqueda de un modelo robusto, validado por las observaciones de un caso particular. Mediante esta técnica, las diferentes hipótesis pueden ser comparadas, clasificadas y combinadas en función de la bondad de ajuste de los datos. Atendiendo también a esta demanda, en las últimas décadas los investigadores de las diferentes disciplinas involucradas en el estudio medioambiental trabajan en el desarrollo de una respuesta tecnológica más adaptada a las necesidades de este tipo de problemas, mediante técnicas de inteligencia artificial, de descubrimiento de conocimiento en bases de datos, de sistemas expertos, árboles de decisión, redes neuronales, algoritmos genéticos, fractales teoría del caos y redes Bayesianas [FIELDING, 1999b, RIAÑO, 1998, WALLEY & O'CONNOR, 2001, KALOGIROU, 2002, KANEVSKI ET AL., 2004]. Estos métodos poseen mecanismos que se ajustan bien al análisis de problemas complejos, constituyendo una importante línea de investigación para la búsqueda de soluciones y modelos para los sistemas complejos analizados en ecología [RECKNAGEL, 2001]. La utilización de estas técnicas supone nuevos retos y una prometedora línea de investigación para la comprensión de los sistemas y procesos medioambientales y ecológicos [SÀNCHEZ-MARRÈ ET AL., 2004]. En esta dirección, en el modelado ecológico, se están desarrollando aplicaciones que abordan un amplio espectro de métodos y que incluyen modelos basados en lógica *fuzzy* y métodos de modelado dinámico, capaces de describir cambios, a fin de abordar la complejidad analítica asociada al modelado de problemas complejos [POCH ET AL., 2004]. De igual forma, en el entorno agrícola, empiezan a aplicarse técnicas basadas en inteligencia artificial, como los árboles de decisión, algoritmos genéticos y más frecuentemente las redes neuronales, en el análisis y modelado de sistemas y procesos relativos a operaciones agrícolas, [SCHULTZ & WIELAND, 1997, MING CHEN, 1997, HASHIMOTO, 1997, DÍAZ ET AL., 2003, MURASE, 2000, FARKAS, 2003, RIBEIRO ET AL., 2003, WEIGERT & WAGNER, 2003, DÍAZ ET AL., 2005].

Capítulo 3

DESCUBRIMIENTO DE CONOCIMIENTO EN BASES DE DATOS

3.1. EL KDD: DESCUBRIMIENTO DE CONOCIMIENTO Y MINERÍA DE DATOS

El proceso de *descubrimiento de conocimiento*, también denominado como *KDD* (siglas de la expresión inglesa *Knowlegde Discovery in Databases*) se define como *el proceso iterativo e interactivo de extracción no trivial de información potencialmente útil y comprensible a partir de un gran volumen de datos en los que la información está implícita, y es desconocida* [AGRAWAL ET AL., 1993, FAYYAD ET AL., 1996b]. El objetivo principal de un proceso KDD es la identificación de atributos relevantes, que tienen en común un conjunto de casos o registros, y el conocimiento que se obtiene del proceso es una consecuencia de que la información extraída sea de interés. FRAWLEY ET AL. [1991] exponen la siguiente definición formal: dado un conjunto de hechos (observaciones o datos) F, un lenguaje L y alguna medida de certidumbre C, se define modelo como un estamento S en L que describe la relación entre el subconjunto F_S de F con una certidumbre c, tal que S es más simple que la enumeración de todos los hechos. Por lo tanto, la salida de un programa de descubrimiento que monitoriza al conjunto F produce a un modelo que expresa el *conocimiento descubierto*. Un modelo debe cumplir tres requisitos fundamentales para que pueda ser considerado conocimiento desde la perspec-

tiva de KDD: primero, *certeza*, segundo, un elevado *interés* y por último, que sea de gran *utilidad*.

Según FAYYAD ET AL. [1996b], un proceso KDD se puede llevar a cabo con dos posibles finalidades, la *predicción* y la *descripción*. Estos dos objetivos primarios se persiguen a través de una gran variedad de métodos específicos de minería de datos como son:

- CLASIFICACIÓN o DESCRIPCIÓN: Sabiendo la existencia de ciertas clases, clasificar consiste en establecer un modelo para ubicar las observaciones en alguna de las clases existentes.
- PREDICCIÓN: Los modelos que describen un conjunto de clases, generados a partir de algún método de inducción, se utilizan para ubicar observaciones futuras en alguna de las clases existentes.
- AGRUPAMIENTO (*clustering*): A partir de una serie de observaciones se establecen la existencia de clases o grupos en los datos utilizando alguna medida de similitud.
- RESUMEN (*summaritization*): Se obtienen representaciones compactas para subconjuntos de datos de entrada. Ejemplos son la generación automática de informes, la visualización de datos, la estadística descriptiva, etc.
- MODELADO DE DEPENDENCIAS: Se expresan las relaciones existentes entre variables. Un ejemplo es descubrimiento de reglas de asociación, en la que se obtiene conocimiento interesante para los usuarios en forma de reglas que reflejan relaciones entre los atributos presentes en los datos
- ANÁLISIS DE SECUENCIAS: Se intenta modelar el cambio que provoca la evolución temporal de alguna variable.

La figura 3.1 muestra las diferentes etapas de un proceso KDD [PÉREZ & RIBEIRO, 1996, JOHN, 1997]. En primer lugar, en cualquier proceso KDD existe una etapa de **selección** de los datos más adecuados para la extracción de

conocimiento. A continuación los datos se **pre-procesan**, en concreto se limpian y **transforman** en un formato ajustado al algoritmo de extracción que se utilizará posteriormente. La siguiente fase del KDD es la etapa de **extracción** de conocimiento o inducción de modelo también conocida como *minería de datos*. Finalmente, en la etapa de **análisis** la información extraída es verificada y transformada en a un formato que facilite la labor de interpretación del usuario final.

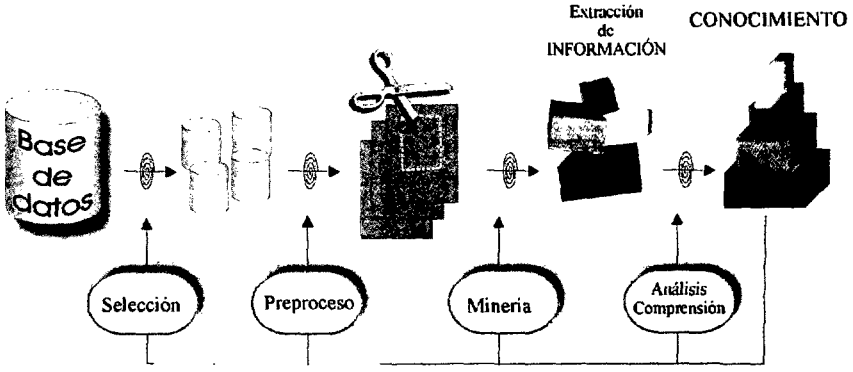


Figura 3.1: Etapas del proceso KDD

A continuación, se describen las principales características de cada una de las etapas que constituyen el proceso KDD.

(a) Etapa de adquisición/recolección y selección de los datos

Las bases de datos reales que almacenan los datos del estudio suelen presentar distintos problemas que puede ser necesario solventar antes de comenzar la extracción. En primer lugar, suelen contener atributos irrelevantes e incluso, puede que los relevantes no estén almacenados [FRAWLEY ET AL., 1991]. Además, es frecuente que en bases de datos reales los atributos presenten cierta interacción, un hecho que podría provocar un efecto no deseado en los resultados, ya que la mayoría de los algoritmos buscan precisamente esas relaciones

[FREITAS, 2001]. Esta es una de las razones principales por la que es necesario realizar una selección previa de los datos asesorado por los expertos. Durante esta etapa, se realiza una exploración de los datos con el fin de asegurar una correcta compresión del dominio en el que se sitúa el problema. Esta tarea puede proporcionar información básica que nos ayude en la toma de decisiones en etapas posteriores. Además, es una etapa esencial que permite intuir el conocimiento que se puede encontrar o no en los datos. Esta fase además facilita que los algoritmos traten simultáneamente con una gran cantidad de datos ya que la selección permite definir ventanas de trabajo, aumentado la probabilidad de encontrar conocimiento útil. En definitiva, la selección del conjunto de datos y en consecuencia, la identificación del problema es un paso fundamental en la búsqueda de conocimiento [PYLE, 1999].

(b) Etapa de preparación

Después de la selección de los datos, se realiza un proceso de limpieza y preprocesado para eliminar el ruido e inconsistencias. Se determina que hacer con la información perdida en la limpieza y, en algunos casos, posteriormente se transforman (normalizan o etiquetan) los datos si el método de extracción que se va utilizar así lo requiere.

Algunas de las tareas que se realizan en esta etapa son: buscar las características más importantes para representar los datos dependiendo de los objetivos, reducir un excesivo número de dimensiones, transformar las variables, etc. Todas estas tareas pueden requerir cierto conocimiento del dominio del problema.

(c) Etapa de minería de datos: *Data Mining*

Es en esta etapa en la que se realiza la propia extracción o búsqueda de nueva información. Durante el proceso, han de tenerse en cuenta, tanto la calidad de la información, la complejidad como el coste que requiere encontrar el conocimiento. En consecuencia, la elección del algoritmo más apropiado para extraer/obtener el conocimiento deseado depende de varios factores que incluyen el tipo del problema, el tipo de datos y el tipo de conocimiento deseado, así como la forma de ese conocimiento. Esta etapa puede verse como

un proceso de búsqueda de hipótesis usando técnicas de inteligencia artificial (aprendizaje automático) o métodos estadísticos, y en el que se produce una continua generación, verificación y modificación o eliminación de un conjunto de modelos candidatos a solución. La verificación se realiza a través del valor suministrado por una función de calidad que determina la capacidad que tiene un modelo de ajustarse a los datos, y provoca el rechazo o la aceptación de cada hipótesis o modelo.

(d) Etapa de análisis los resultados

El conocimiento extraído en la etapa de minería debe ser evaluado y validado, ya que recordemos la información es interesante en el grado en el que es exacta, nueva y útil para los objetivos del usuario final. En esta etapa de interpretación de resultados se evalúa qué información de la obtenida es de interés, es decir se decide qué debe ser presentado al usuario. En algunos sistemas la evaluación de la información la lleva a cabo el propio algoritmo de extracción. No por ello deja de ser un factor clave una etapa posterior de evaluación del conocimiento descubierto para saber si el modelo es estadísticamente significativo. Por ejemplo, un patrón que no es muy frecuente no es interesante. Por tanto, es necesario tener alguna medida de confianza de los modelos, obtenidos a partir de la información contenida en una base de datos, que nos indique su representatividad. Esto es especialmente importante en aquellos casos en los que los modelos se utilizan posteriormente en tareas de predicción. Asimismo, y dependiendo del método utilizado, en algunos casos es interesante disponer de ciertas medidas de calidad, como por ejemplo, el tanto por ciento de ejemplos de entrada cubiertos, es decir el grado de exactitud del modelo. En cualquier caso, es necesario tener información del dominio para decidir el grado de interés, ya que los factores específicos que influyen en la determinación del conocimiento extraído o impacto varían según las diferentes bases de datos, dominios del problema y objetivos del usuario.

(e) Etapa de presentación del conocimiento obtenido

Finalmente, el sistema debe presentar el conocimiento descubierto/obtenido de un modo útil y comprensible para el usuario final. Por ejemplo, pueden ge-

nerarse informes usando plantillas que incorporan el lenguaje natural, o utilizar representaciones del conocimiento extraído cercanas al lenguaje natural como por ejemplo las reglas Si-Entonces (If-Then), etc. En esta etapa se requiere conocimiento del dominio para decidir el formato más adecuado con el que mostrar el conocimiento extraído.

Desde la perspectiva del KDD, todas las etapas son complementarias en el sentido de que un analista especializado podría saltar de una a otra hasta extraer conocimiento realmente útil.

Las aplicaciones del KDD son muy numerosas, entre todas ellas se pueden mencionar como ejemplos interesantes en las ciencias de la tierra: la catalogación de objetos del espacio (estrellas, galaxias, planetas, etc.) extrayendo reglas mediante la optimización estadística de árboles de decisión, o la interpretación del genoma humano mediante el análisis las secuencias de ADN utilizando la combinación de los modelos estadísticos de *Markov* y la minería de datos [FAYYAD ET AL., 1996a].

Entre el KDD y las bases de datos existe un área de análisis de datos relativamente reciente. Se trata de el llamado *data warehouseing*, que se puede traducir como *almacenamiento de datos*, y que se refiere a una tendencia de análisis *on-line* de datos transaccionales depurados, con el fin de extraer información útil para la toma de decisiones. Este tipo de repositorio recopila información de diferentes almacenes, guardando los datos en un sólo sitio físico, y sus tareas principales son la limpieza de los datos, eliminación de ruido e inconsistencias; la transformación de los datos en formatos adecuados; la integración de las diferentes fuentes de datos; la carga y la actualización periódica de la información. Los datos se organizan en materias principales y desde una perspectiva histórica en lugar de transaccional como sucede en las bases de datos convencionales. Las técnicas de OLAP (*On-Line Analitical Processing*) propuestas en [CODD ET AL., 1993] que se realizan sobre este tipo de repositorios, son herramientas cercanas al campo de la minería de datos para el análisis avanzado, interactivo y automático de la información.

3.1.1. El KDD desde un punto de vista del aprendizaje automático

Habitualmente se han considerado dos modos básicos de razonamiento o inferencia: la deducción (inferencia desde las causas hacia los efectos, o desde lo universal hacia lo particular) y la inducción (que recorre el camino inverso). A estos dos tipos de razonamiento hay que agregarle un tercer modo propuesto recientemente llamado abducción o retroducción, que está relacionado con la génesis de hipótesis, tanto en el razonamiento científico como en el pensamiento ordinario. En concreto, la abducción es el proceso de razonamiento mediante el cual se engendran las nuevas ideas, las hipótesis explicativas y las teorías científicas.

Atendiendo a su definición, la inferencia es la clave del aprendizaje automático (*Machine Learning*), que desarrollado en el marco de la inteligencia artificial, se define como el proceso por el que un ordenador o sistema aumenta su "conocimiento" y la habilidad de solucionar un determinado problema mediante la experiencia [Mitchell, 1997]. Más concretamente, los sistemas cognitivos intentan entender su entorno usando una simplificación del mismo, es decir un modelo. La creación de un modelo del entorno se conoce como *aprendizaje inductivo*. Este modelo se entiende como una *descripción* de distintos elementos del entorno, que atendiendo a un conjunto de propiedades comunes, pueden agruparse en una determinada *clase* o concepto. Este aprendizaje inductivo, entre otras tareas, ha permitido el desarrollo de herramientas de inducción de descriptores en dominios donde existen problemas de elevada complejidad [Riaño, 1997].

De las diferentes estrategias que se plantean en aprendizaje automático para la inducción de modelos tienen especial relevancia las conocidas como aprendizaje *supervisado* y aprendizaje *no supervisado*. En **aprendizaje supervisado** se obtienen modelos o descriptores de las clases utilizando un conjunto de ejemplos preclasificados. En otras palabras, los métodos supervisados constan de una etapa de entrenamiento inicial o búsqueda de descriptores en la que se suministra al sistema el conjunto de ejemplos de entrada para una o varias clases. La segunda etapa de este tipo de métodos es la clasificación o verificación, y en

ella se emplea el modelo encontrado en la fase de entrenamiento. Según el tipo de representación seleccionada para el descriptor o modelo (redes neuronales, árboles de decisión, sistemas de reglas, etc.) el conocimiento embebido en el mismo podrá ser más o menos interpretable. Por otra parte en el **aprendizaje no supervisado** no se parte de conocimiento previo acerca de las clases, por lo que la primera tarea consiste en agrupar (*clustering*) los datos a partir de alguna medida de similitud. Como resultado se obtienen agrupaciones con alta similitud entre los elementos del grupo y gran disimilitud entre los elementos de grupos distintos. Una vez que se han formado los grupos, las descripciones de las clases se obtienen a través de un aprendizaje supervisado. La semejanza o similitud se basa fundamentalmente en medidas de distancia para valores numéricos, como por ejemplo la distancia euclídea. También, aunque en menor proporción, existen medidas de distancia para valores conceptuales o categóricos. Entre los diferentes métodos de *clustering* destacan por ser muy conocidos, el algoritmo *Chain-map* o distancia encadenada que es útil en una primera exploración de los datos y el algoritmo *K-medias* que requiere conocer el número de clases en las que se distribuirán los datos.

Un proceso de aprendizaje consta fundamentalmente de tres elementos: (1) la *estrategia de búsqueda* que se define como el modo en el que se explora el espacio de todos los modelos posibles (espacio de búsqueda), en definitiva el modo en el que se generan los modelos; (2) una *función de calidad* que permite determinar el ajuste del modelo a los datos; y finalmente, (3) un método de *representación del conocimiento* que es la forma en la que se expresa el modelo y por tanto el conocimiento embebido en el mismo. Cada uno de estos elementos se explican a continuación.

El conjunto de todas las *descripciones* que se pueden construir con la representación seleccionada y las condiciones sobre las variables del problema se conoce como **espacio de búsqueda** o espacio de soluciones. En ese espacio de búsqueda los procedimientos de generación de modelos se mueven a partir del conjunto O de *operaciones* que se pueden realizar sobre las descripciones. Finalmente, una *función* f *de calidad* evalúa lo correcto que es cada modelo

permitiendo orientar la búsqueda hacia modelos de mayor calidad. Dentro de esta formalización de la búsqueda de descriptores, se entiende por *cubierta* de un modelo el número de instancias o ejemplos descritos o explicados por dicho modelo. Los encargados de atravesar el espacio de soluciones son los **algoritmos de búsqueda**, un elemento clave del aprendizaje automático. Mediante operaciones de generalización o especificación, según la forma en la que exploren el espacio de búsqueda, los algoritmos van transformando una *descripción inicial*, durante un proceso iterativo que termina cuando la calidad del modelo es igual o superior a la deseada inicialmente. Las operaciones de *generalización* convierten un modelo dado en otro más genérico y en consecuencia se dice que debilitan la descripción aumentando la cobertura, mientras que las operaciones de especificación o *especialización* buscan refinar el modelo por lo que lo fortalecen, reduciendo su cobertura.

Así mismo, estos algoritmos pueden realizar la exploración mediante distintos **tipos de búsqueda**, que atienden a varias clasificaciones. En primer lugar, según el orden o estrategia de exploración, la búsqueda puede ser *sistemática*, si el algoritmo sigue un orden determinado, o *estocástica*, si sigue un orden aleatorio. Atendiendo a la capacidad retroactiva, la búsqueda puede ser *irrevocable* o *tentativa* (*backtracking*), permitiendo en esta última la vuelta a puntos del espacio explorados previamente para aplicar otras operaciones. Finalmente, teniendo en cuenta la optimización, la búsqueda puede ser *exhaustiva*, cuando el algoritmo examina todas la posibles combinaciones, o *heurística*, cuando utiliza conocimiento para evitar explorar completamente el espacio de búsqueda. Si la búsqueda se estructura en forma de árbol, la exploración *exhaustiva* tiene dos posibles estrategias: en *anchura*, una exploración completa según niveles, y en *profundidad*, visitando primero los nodos de mayor profundidad. Las técnicas de exploración exhaustiva son viables en espacios de búsqueda pequeños donde son capaces de encontrar la mejor solución aunque requieren con frecuencia una etapa *poda* que simplifique la complejidad de los modelos. Ahora bien, los casos que presentan una explosión combinatoria o

los problemas denominados $NP - completos$[1], no se ajustan a una búsqueda exhaustiva y en estos casos la búsqueda heurística es la única opción factible, aunque hay que alcanzar un compromiso entre la calidad del modelo y el coste computacional de encontrarlo. La búsqueda en este caso se basa en una *función expectativa* cuyos valores permiten expandir las diferentes rutas del árbol, quedándose con la ruta que marca el nodo de más valor. La *búsqueda heurística* dirige el proceso de búsqueda basándose en el conocimiento específico del dominio o en la calidad del modelo candidato como descriptor de la clase. Por lo tanto, los métodos que utilizan esta información permiten una interacción entre el proceso y los objetivos del usuario, provocando como consecuencia una reducción del tiempo de computo. Entre las búsquedas heurísticas que utilizan información de la calidad de la solución se encuentran, la búsqueda de *primero el mejor* (*best-first search*) y una simplificación de ésta, que propone un procedimiento de *escalada* (*hill-climbing*) utilizando la función expectativa A^*. También, son consideradas búsquedas de tipo heurístico, pero estocásticas y no sistemáticas, los *algoritmos genéticos* (*Genetic Algorithms*), basados el la teoría de la selección natural, y el *recocido simulado* (*Simulated Annealing*) que imita el proceso de cristalización-ordenación de los sólidos, basándose en los principios de la termodinámica y el cálculo de la entropía [SKIENA, 1998].

Otro elemento crítico en la búsqueda de modelos es la **función de calidad** o *criterio de preferencia*, que asigna un valor de calidad a cada descripción encontrada. En cualquier proceso búsqueda de descripciones la calidad es fundamental para caracterizar los modelos candidatos, por lo que es una medida que suele condicionar el proceso de búsqueda. Esta calidad puede venir dada por cuatro conceptos: la *utilidad*, la *certidumbre* del modelo, la *simplicidad* y, por último, el *grado de novedad* [HAN & KAMBER, 2001]. Excepto este último, que suele ser un concepto muy subjetivo, dependiente del usuario y difícil de implementar, el resto de los conceptos pueden ser cuantificados mediante una función, expresión matemática o una regla heurística. Por ejemplo, la *simplici-*

[1] Problemas cuya complejidad aumenta con el tamaño del problema de forma exponencial. Por ejemplo, el problema del viajante en el que se busca el camino de distancia mínima para recorrer sin repetir una serie de ciudades.

dad, que se basa en teoría de Ockham[2] que expresa que el modelo más simple es el que describe la esencia. La *certidumbre*, es el parámetro más frecuente y utiliza funciones de exactitud o fiabilidad del modelo usando técnicas de probabilidad; y finalmente la *utilidad* se mide detectando umbrales de ruido o simplemente el concepto de cobertura. En este sentido, la función debe tener en cuenta que los modelos evaluados sean correctos, utilizando los ejemplos conocidos, y también conocer como funcionan en situaciones desconocidas. Para ello se pueden fijar dos medidas: la *precisión de clasificación* (*precision*) y el *alcance* (*recall*), que respectivamente se definen como:

↪ *Precisión de clasificación* se define como la probabilidad de que un objeto pertenezca a la clase e indica el funcionamiento de la descripción en el almacén de datos. El valor de precisión es máximo, es decir igual a 1 cuando el modelo cubre solo ejemplos de la clase que describe. En este caso, hablamos de un modelo *determinista* que es condición suficiente para la clase.

↪ *Alcance* o *validez* es la probabilidad de la correcta descripción de un objeto arbitrario. El valor de alcance es máximo, es decir es igual a 1 cuando el modelo cubre al menos todos ejemplos de la clase que describe. Estos modelos se denominan *completos* y son condición necesaria para la clase.

Combinando estos dos términos, decimos que un modelo es *exacto* si satisface dos requisitos al mismo tiempo: precisión máxima y alcance máximo, en otra palabras cubre todos los ejemplos de la clase que describe y no cubre ejemplos de otras clases.

En la literatura [FIELDING, 1999a, FURNKRANZ & FLACH, 2003], existen numerosos métodos para calcular la calidad de los modelos basados en parámetros probabilísticos, estadísticos o en la teoría de la información, etc. A

[2] Ley de Parsimonia o Navaja de Ockham : *entia non sunt multiplicanda sine necessitate* (no multiplicar los entes sin necesidad). Resumen de la filosofía terminística de William de Ockham (1285-1349) que supone el rechazo de lo superfluo y la exigencia de simplicidad en la explicación de los sucesos reales.

		REAL		
		p	n	
ESTIMADO	p	V_p	F_p	E_p
	n	F_n	V_n	E_n
		R_p	R_n	m

Tabla 3.1: Tabla de clasificación para dos clases (p) de ejemplos positivos y (n) de negativos para determinar los parámetros *Vp*, *Fn*, *Vn* y *Fp*

continuación se describe una de las posibles función de calidad que se podría definir.

- *Tasa de error* (t_{err}) es la medida más habitual y más simple de exactitud de un modelo, y viene dada por $t_{err} = \frac{m_{err}}{m}$ donde m_{err} es el número de errores y m el número total de instancias, entendiendo error como la descripción errónea de una instancia. En este caso, el objetivo global de un proceso de aprendizaje es dirigir la búsqueda hacia modelos que minimicen los errores.

En un problema de dos clases, una que llamaremos de ejemplos positivos (p) y otra de ejemplos negativos (n) o contraejemplos, se pueden definir dos tipos de error. El primero es el derivado de considerar como positivo un ejemplo negativo, en este caso hablamos de *falsos positivos* (F_p). El segundo tipo de error lo provoca la consideración de un ejemplo positivo como negativo, en este caso hablamos de *falsos negativos* (F_n). Siguiendo esta nomenclatura, son *verdaderos positivos* (V_p) y *verdaderos negativos* (V_n) aquellas instancias que se clasifican correctamente en su correspondiente grupo. Estos cuatro parámetros se pueden representar dentro de una tabla denominada de clasificación (tabla 3.1) que presenta en filas los valores reales de las instancias (R) y en las columnas los valores estimados (E). En la tabla m es igual a todos los ejemplos.

- La función *aciertos*, llamada también CCR (*Correct Clasification Rate*) y la más utilizada en estudios ecológicos [FIELDING, 1999a], se formula

en términos de verdaderos y falsos, y calcula la exactitud utilizando la relación entre el número de aciertos ($V_p + V_n$) frente al total de ejemplos. Por otro lado, la función llamada *aciertos y errores* es una extensión de la función *aciertos*, que incluye la proporción de fallos para que la búsqueda sea más efectiva e incluya el término alcance. Estas fórmulas pueden expresarse de la siguiente forma:

$$f_{aciertos} = \frac{Vp + Vn}{m} \tag{1}$$

$$f_{aciertos\ y\ errores} = \left[\frac{Vp + Vn}{m}\right] \times \left[1 - \frac{Fp + Fn}{m}\right] \tag{2}$$

donde m es igual a $Vp + Fp + Vn + Fn$.

La función *aciertos* requiere únicamente realizar dos cálculos, por lo que se trata de una función que puede consumir menos tiempo de computo. Sin embargo, según FIELDING [1999a] incluir los errores, como hace la función *aciertos y errores*, puede evitar que durante la evaluación se produzca un posible fenómeno de prevalencia de alguna de las clases y que éste afecte al resultado.

- Una de las funciones más utilizada en el campo de la minería de datos para la búsqueda de descriptores es la basada en el cálculo de dos términos: la *sensibilidad y la especificidad*; propuesta en [HAND, 1997] se basa también en estos cuatro parámetros de calidad anteriormente definidos. Esta función incluye los dos tipos de errores. En primer lugar el término *sensibilidad* (S) que se define como la capacidad que tiene un modelo para describir los ejemplos positivos y formula como $S = \frac{V_p}{V_p + F_n}$. El segundo término, *especificidad* (E), se define como la capacidad que tiene ese mismo modelo de no cubrir ejemplos negativos y su expresión matemática es: $E = \frac{V_n}{V_n + F_p}$. Estos dos términos componen la función de calidad final que se expresa en la ecuación 3.

$$f_{S \times E} = S \times E \tag{3}$$

- *Soporte* y la *confianza* [AGRAWAL ET AL., 1993] son otros parámetros de calidad descritos en la literatura y muy utilizados en la extracción de reglas de asociación. *Soporte* (s) se define como el número de registros que cubre una regla y se expresa como $s(I \Rightarrow p) = \rho(I \cup p)$, donde I representa al conjunto de instancias que satisface el antecedente y p las que pertenecen a la clase, y ρ es la probabilidad de que se den ambas condiciones. Convencionalmente, en clasificación este parámetro representa al número de instancias cubiertas por el modelo a evaluar. Según la nomenclatura de verdaderos y falsos, vendría dado por la ecuación $s = \frac{V_p}{V_p+F_n}$. Una medida asociada al soporte, es la proporción de ejemplos correctos en clasificación, es decir la confianza (c). Formalmente, se define *confianza* como la proporción de ejemplos que satisface las condiciones de la regla además de pertenecer a la clase, es decir el cálculo de la probabilidad condicionada de que los elementos cumplan el modelo cuando éstos pertenecen a la clase. Matemáticamente, se expresa como $c(I \Rightarrow p) = \rho(p/I)$. De nuevo, en términos de verdaderos y falsos este parámetro se expresa del siguiente modo, $c = \frac{V_p}{V_p+F_p}$.

- Por último, otra función que utilizan muchas técnicas de inducción de modelos es la basada en el concepto *ganancia de información* de la *teoría de la información* enunciada en [SHANNON, 1948] . El elemento esencial de esta función es la *entropía* básicamente expresada como $(-log_c\ \rho)$ para un problema de c clases con probabilidad ρ, y que se define como la magnitud que mide la información contenida en un conjunto de datos, es decir la información que aporta un dato o hecho concreto, y que se expresa matemáticamente como:

$$H(A_v) = -\sum_{v=1}^{V} \rho_{v(i)}\ log_2\ \rho_{v(i)} \tag{4}$$

donde $\rho_{v(i)}$ es la probabilidad de clasificar un ejemplo i al dividir el conjunto de ejemplos según un determinado atributo A y un valor $v \in V$, que es el conjunto de todos los valores de A.

La cantidad de la información $I(p,n)$ que se necesita para decidir si una muestra cualquiera pertenece a p o a n se define como:

$$I(p,n) = -\rho_p \; log_2 \; \rho_p - \rho_n \; log_2 \; \rho_n \tag{5}$$

Finalmente, la *ganancia de la información* G que se obtiene del conjunto dividido a través el atributo A se calcula como la diferencia entre la cantidad de información $I(p,n)$ menos la *entropía*, es decir el grado de pureza de los subconjuntos generados $H(A_v)$.

$$G(A) = I(p,n) - H(A_v) \tag{6}$$

Como se deduce de la expresión, una menor entropía implica una mayor ganancia de la información, y esta ganancia es máxima cuando cada subconjunto contiene un sólo tipo de ejemplos (sólo positivos o sólo negativos), es decir no existen impurezas y la entropía es igual a cero.

En términos de verdaderos y falsos para un conjunto de m ejemplos que se dividen en dos clases (p y n), la entropía y ganancia de la información [BALDI ET AL., 2000] se expresa como:

$$\begin{aligned} E(p,n) = -H\left(\frac{Vp}{m}, \frac{Vn}{m}, \frac{Fp}{m}, \frac{Fn}{m}\right) &- \frac{Vp}{m} log \frac{Vp+Fp}{m} \frac{Vp+Fn}{m} \\ &- \frac{Fn}{m} log \frac{Vp+Fn}{m} \frac{Vn+Fn}{m} \\ &- \frac{Fp}{m} log \frac{Vp+Fp}{m} \frac{Vn+Fp}{m} \\ &- \frac{Vn}{m} log \frac{Vn+Fn}{m} \frac{Vn+Fp}{m} \end{aligned} \tag{7}$$

donde

$$H\left(\frac{Vp}{m}, \frac{Vn}{m}, \frac{Fp}{m}, \frac{Fn}{m}\right) = -\frac{Vp}{m}log\frac{Vp}{m} - \frac{Vn}{m}log\frac{Vn}{m} - \frac{Fp}{m}log\frac{Fp}{m} - \frac{Fn}{m}log\frac{Fn}{m} \quad (8)$$

3.1.2. BÚSQUEDA SUPERVISADA DE DESCRIPTORES

El aprendizaje supervisado, atendiendo a la definición expresada en la sección anterior precisa conocer ejemplos distribuidos en unas clases, previamente predefinidas, de las que se desea conocer su descripción o modelo.

La búsqueda supervisada de un modelo descriptivo (descriptor) se basa en la siguiente idea. Supongamos un conjunto de m ejemplos que poseen dos atributos H cuyos valores posibles son $\{h_1, h_2, \ldots, h_m\}$ y G con los valores $\{g_1, g_2, \ldots, g_m\}$, y una tercera variable C que se utiliza para predefinir las clases p y n. Los modelos para explicar una de las clases, se expresan en función de los dos atributos, por lo que también reciben el nombre de variables predictoras. Esta situación se representa en la figura 3.2, que muestra la distribución de todas las ejemplos en función de estos dos atributos. En la figura los ejemplos positivos, es decir los ejemplos que pertenecen a la clase p $(C = p)$ están representados por el $\oplus$ positivos p y los ejemplos negativos, los ejemplos que pertenecen a la clase n $(C = n)$ se representan con el símbolo $\ominus$.

Ahora bien, en la búsqueda de modelos descriptores de clases, se puede dividir el dominio de valores de los atributos predictores, antes o durante el proceso de búsqueda. Esta fase de división no es requerida, es frecuente en la búsqueda de modelos comprensibles e interpretables. Además, esta partición del dominio (en los atributos numéricos fundamentalmente hablamos de *discretización*) facilita el proceso de aprendizaje. Existen diferentes métodos para realizar la partición del dominio de cualquier variable, que se explican con mayor detalle en el apartado 6.1 de preparación de los datos.

En nuestro ejemplo, para el atributo H se crean las particiones a y b y de forma análoga, se divide el dominio para el atributo G en las particiones c y d, tal y como se puede ver en la figura 3.3.

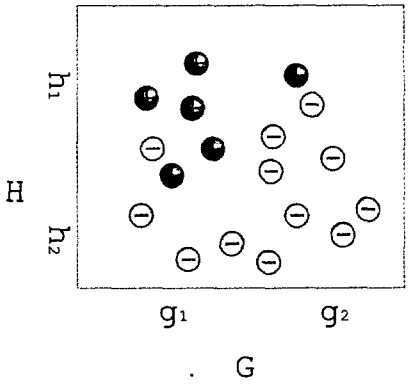


Figura 3.2: Ejemplo de instancias con dos atributos H y G que se distribuyen en dos clases p ($\oplus$) y n ($\ominus$)

En cualquier caso, utilizando los atributos predictores y sus diferentes valores se construyen los modelos candidatos para la clase objetivo. Y tras la construcción de una hipótesis, se evalúa su calidad como solución, determinando el número de aciertos y de errores de clasificación de dicha hipótesis, comparando los valores reales de los ejemplos con los valores estimados por el modelo (figura 3.4).

Supongamos en nuestro ejemplo que se construye el modelo candidato M_c de la forma $f_{(H,G)} = p$ que relaciona los atributos según la expresión: $[f_{(H,G)} = H(a) \vee G(c)]$. La aplicación de este modelo sobre los datos de partida, determina que un ejemplo pertenece a la clase p si el valor de atributos H y G cumplen la expresión. En caso contrario, la instancia pertenece a la otra clase (n), tal y como se observa en la figura 3.4b. Por lo tanto, un parámetro de calidad como la exactitud para este ejemplo se podría calcular teniendo en cuenta que 16 instancias de las 18 que existen se han descrito correctamente. Los dos fallos, que se pueden observar en la figura, son el ejemplo negativo, número 1 en la figura, descrito como positivo (F_p), y a un F_n, ejemplo positivo (2) que no sido descrito como tal.

En el ejemplo, para explicar de un modo sencillo el fundamento de las técnicas de búsqueda de descriptores se han utilizado tan sólo dos variables.

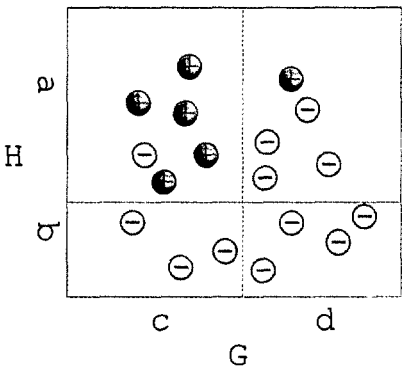


Figura 3.3: Distribución de las instancias según las particiones de los atributos G y H

No obstante, lo más frecuente es que los ejemplos posean un número elevado de atributos y por lo tanto, que el proceso de aprendizaje involucre un gran número de variables, decimos entonces que son espacios de búsqueda N-dimensionales.

El objetivo general un proceso de aprendizaje es buscar iterativamente empleando algún algoritmo de búsqueda el modelo descriptivo que maximice la calidad. La etapa inicial de búsqueda de descripciones se realiza sobre una colección de ejemplos seleccionados aleatoriamente a partir del conjunto inicial de ejemplos, que se denomina *conjunto de entrenamiento.*

Tras la etapa de entrenamiento en la que se descubre el modelo final, es necesario calcular la calidad, o lo que se conoce como *exactitud predictiva*, en una etapa denominada de *validación.* El método más simple y popular para calcular este parámetro es la utilización de un conjunto de datos que no hayan intervenido en la etapa de aprendizaje, es decir un *conjunto de validación* separado de forma aleatoria de los datos antes del aprendizaje. Este método de evaluación se denomina H del inglés *holdout*, traducible como "a perdurar". La exactitud del modelo se obtiene a partir del porcentaje de ejemplos de este segundo conjunto correctamente descritos por el modelo. Sin embargo, existen otros métodos que permiten validar y conocer la exactitud predictiva de los modelos descubiertos como el remuestreo y el método de *Jackknife*, entre otros

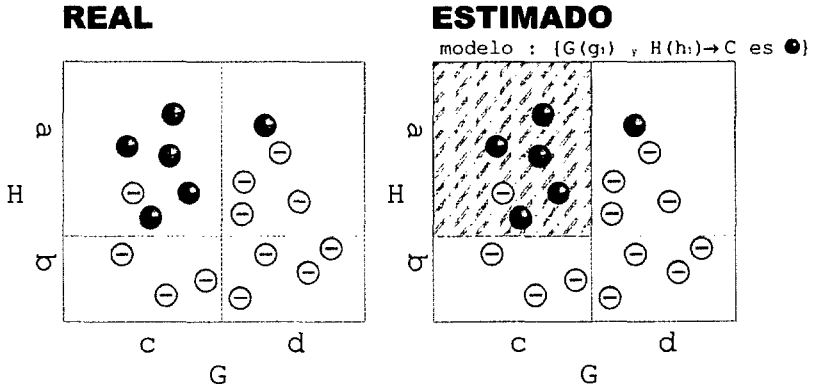


Figura 3.4: Comparación entre (a) el valor real de cada ejemplo y (b) el valor estimado por el modelo ejemplo M_c para la clase p, es decir cuando $C = \oplus$

[FIELDING, 1999a].

3.1.3. REPRESENTACIÓN DEL CONOCIMIENTO. ÁRBOLES DE DECISIÓN Y REGLAS

Definidos la estrategia de búsqueda y la función de cadidad, el último elemento importante de un proceso de aprendizaje automático es la forma en la que el conocimiento es expresado, es decir como se escribe el modelo. La **representación del conocimiento** es una forma consistente y útil de organizar la información en las máquinas y está ligada estrechamente al paradigma de aprendizaje que se utilice. Además, la representación debe ser inteligible con fin de facilitar su procesamiento. Asimismo, el conocimiento debe estar representado por un lenguaje capaz de mostrar descripciones del mundo, según un conjunto de convenciones sintácticas y semánticas, es decir un lenguaje que contenga tanto símbolos como significado. La evaluación de cualquier tipo de representación se realiza a partir de su capacidad para solucionar el problema, su expresividad, sencillez e interpretabilidad.

Existen diferentes formas de representar el conocimiento, algunas de las más importantes se describen a continuación.

1. Las *representaciones simbólicas* son expresiones lógicas con un signifi-

cado preestablecido utilizando los símbolos y conceptos que emplearía un experto. Este tipo de representación evita las ambigüedades típicas de los lenguajes naturales y simula directamente caracteres inteligentes del ser humano. Representaciones de tipo simbólico son la *representación cuasi-proposicional* y la *representación proposicional.* Los métodos *cuasi-proposicionales* usan sentencias declarativas y se puede expresar de forma *normal-conjuntiva*, como la suma de atributos, o de forma *normal-disyuntiva*, como la separación de los atributos que forman el descriptor. Los métodos de *representación proposicional* usa sentencias declarativas a las que se puede asociar un valor lógico de verdad (verdadero o falso), es decir se usan proposiciones lógicas. La asociación de proposiciones requiere la utilización de *conectivas lógicas* como negación, conjunción, disyunción, condicional y equivalencia o bicondicional, que permiten construir expresiones nuevas a partir de las existentes, obteniendo nuevos significados. El razonamiento en un lenguaje de símbolos utiliza la lógica y cálculo algebraico establecidos por las leyes de Boole [BOOLE, 1848].

Los siguientes tipos de representación simbólica se derivan de la lógica proposicional.

- La *lógica de primer orden* pretende formalizar las expresiones del conocimiento humano y está formada por frases declarativas simples.
- Los *árboles de decisión* (o *árboles de clasificación y regresión*) son gráficos de flujo con estructura de árbol.
- Las *reglas de clasificación* surgen como una alternativa a los grandes árboles, que son complejos e ininterpretables.
- Los sistemas expertos están basados *reglas de producción*, o también las *listas de decisión*, que se utilizan para automatizar el razonamiento, realizan inferencias concatenadas, por lo que el razonamiento se hace de forma progresiva.

2. La *representación sub-simbólica* simula los elementos de más bajo nivel que componen los procesos inteligentes, esperando que la combinación

surja de forma espontánea, por lo tanto, utiliza un nivel de abstracción bajo y en consecuencia no son directamente interpretables. Este tipo de representación es la que usan las en redes neuronales o clasificadores bayesianos, ambos difíciles de interpretar.

3. La *representación estructural*, que se puede considerar un caso particular de representación simbólica, dotan a la expresión de una distribución y orden de la información representada, y pueden ser:

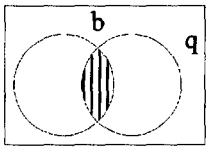
   - Las *redes semánticas* y *mapas conceptuales*, estos últimos de mayor complejidad estructural, contienen nodos que representan conceptos, dotados por tanto de significado, y que están conectados por arcos que representan relaciones o predicados entre los conceptos.
   - Los *esquemas* o *marcos* son un tipo de representación que permite describir objetos y clases de objetos incluyendo los atributos y sus valores en los llamados *slots* (ranuras).

Por su popularidad destacan los árboles de decisión y las reglas. A continuación, se exponen los elementos fundamentales de la lógica proposicional porque es la base de estos dos tipos de representación, y parte de la base teórica de la propuesta de esta tesis.

Lógica proposicional

La lógica proposicional permite generar estamentos o proposiciones lógicas, utilizando los siguientes elementos lógicos:

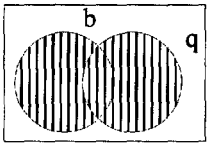
- La *conjunción* equivale a la expresión en lenguaje natural "*b y q*", que es la proposición compuesta de *b* y *q*. Puede ser denotada por AND o el símbolo $\wedge$. Los valores de la tabla de verdad de este operador (figura 3.5b) determina que la conjunción es verdadera (V), cuando ambas proposiciones son verdaderas; de lo contrario, es falsa (F).



b	*q*	*b AND q*
V	V	V
V	F	F
F	V	F
F	F	F

Figura 3.5: Representación gráfica y la tabla de verdad de la conjunción (AND) de las proposiciones b y q

- La *disyunción* se expresa en lenguaje natural como "*b o q*", "*ambos*" o "*o bien b o bien q*". El conectivo o se denota por OR (*o*) o el símbolo $\vee$. Siguiendo la información dispuesta en la tabla de verdad del OR, figura 3.6b, la proposición compuesta $b \vee q$ es verdadera si, al menos una de las proposiciones b o q son verdaderas, y es falsa sólo cuando las dos proposiciones son falsas. Este conectivo tiene dos formas diferentes de empleo. Uno, cuando ambas proposiciones pueden cumplirse, es decir al menos una de las dos es verdadera; y un segundo modo, en el que se excluye la intersección, es decir cuando ambas son verdaderas al mismo tiempo, XOR es falso. Hablamos entonces del *O excluyente* o XOR (figura 3.6c).



b	*q*	*b OR q*
V	V	V
V	F	V
F	V	V
F	F	F

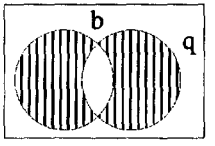
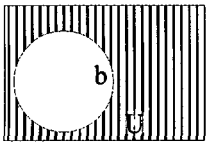


Figura 3.6: (a) Representación gráfica del operador OR, (b) la tabla de verdad del operador OR y (c) el operador XOR

- *Negación* se utiliza para negar la proposición p, es decir corresponde a la proposición *no b*, denotada por NOT b o $\neg p$[3].

  NOT no es un conectivo propiamente dicho, dado de que no une dos proposiciones, sino que es un operador que determina el significado de

[3] En algunos textos también se utiliza $\sim b$ o bien $\bar{b}$



b	NOT b
V	F
F	V

Figura 3.7: (a) Representación gráfica del conectivo NOT de la proposición p y (b) su tabla de verdad

una proposición. La tabla de verdad de NOT, en la figura 3.7b determina que el valor de $\neg b$ es el contrario de la proposición b.

Es importante mencionar que, por convención, en una proposición que está formada por los diferentes operadores lógicos, el AND (que representa el producto) se ejecuta de forma precedente al operador OR (que representa a la suma). Por consiguiente, para determinar el empleo de los paréntesis, se asume la siguiente igualdad:

$$\texttt{A} \times \texttt{X} + \texttt{Y} \times \texttt{Z} = (\texttt{A} \times \texttt{X}) + (\texttt{Y} \times \texttt{Z})$$

y por lo tanto,

```
A and X or Y and Z = (A and X) or (Y and Z)
```

Árboles de decisión y reglas

Los dos paradigmas más empleados en tareas de aprendizaje inductivo de descripción y clasificación son los árboles de decisión y los conjuntos de reglas. Los dos tipos de representación son considerados equivalentes, sin embargo los mecanismos de búsqueda y la forma de los resultados difieren, y como consecuencia su elección depende de los objetivos y lo que se espera del proceso de búsqueda.

Árboles de Decisión

QUINLAN [1986] propuso el lema *divide y vencerás* para la segmentación del espacio de búsqueda en la que se basa la construcción de los árboles de

decisión. Básicamente, un *árboles de decisión*, también llamado de clasificación y regresión, es un cuestionario formado por nodos y ramas. Los nodos representan preguntas sobre atributos y las ramas son los valores salida a cada pregunta. Todas las ramas acaban en nodos hojas, que representan la respuesta final, es decir la clase en la que un ejemplo pertenece. Los árboles se construyen de forma recursiva de arriba a abajo (*top-down*), seleccionando en cada nivel la variable que más información proporciona respecto a la clase predefinida. La información se cuantifica utilizando el criterio heurístico *ganancia de la información*, explicado en la sección anterior. El proceso de cálculo de ganancia se repite con todos atributos en cada nivel del árbol, y siempre se elige el atributo que produce la mayor reducción de la entropía. Esta técnica basada en la entropía, que, frente a otro tipo de búsqueda minimiza el esfuerzo necesario para clasificar un elemento, garantiza un árbol simple, aunque no el más simple. Cuando las soluciones son muy complejas, los árboles son excesivamente grandes, y el propio algoritmo realiza una poda que simplifica la estructura final con el fin de que sea inteligible. Los árboles funcionan bien en presencia de ruido y se suelen aplicar en el análisis de riesgos y la diagnosis médica.

La información que contienen los árboles de decisión se puede representar de forma más simple, como un conjunto de reglas sin grandes perdidas de información. Cada regla del conjunto se construye a partir de la hoja con más cobertura del árbol, previamente podado. Para ello, cada camino o rama, desde la raíz hasta un nodo hoja se convierte en una conjunción de pares *valor-atributo*, que constituye el antecedente de la regla, mientras que el consecuente es la referencia a la clase del nodo-hoja de la rama. Los ejemplos del conjunto de datos cubiertos por la regla construida se eliminan, y se repite el proceso generando nuevas reglas hasta que no queden ejemplos por cubrir. Las diferentes ramas de un árbol, que representan caminos alternativos hasta un nodo hoja se representan como disyunciones (OR). El resultado de la transformación es un conjunto de reglas, que son mutuamente excluyentes.

Los conjuntos de reglas también pueden ser obtenidos directamente sin ne-

cesidad de construir un árbol previamente.

Conjuntos de reglas SI-ENTONCES

Las *reglas* son una forma de representación del conocimiento muy utilizada tanto en minería de datos como en inteligencia artificial. Esta popularidad se debe a que, según ciertas teorías psicológicas, el comportamiento inteligente de los humanos utiliza este esquema de razonamiento y, por lo tanto, son capaces de expresar conocimiento, resolver y representar problemas humanos y cognitivos [NEWELL & SIMON, 1972]. Además, las reglas pueden ser completas y precisas, son fáciles de usar y factibles desde un punto de vista informático y computacional. En resumen, teniendo en cuenta todas consideraciones básicas y decisivas son una elección adecuada de representación del conocimiento, que presentan siguientes características:

- *Interpretatibidad.* Los humanos pueden interpretar directamente el modelo encontrado y reconocer fácilmente el conocimiento que se descubra. En definitiva, las reglas son modelos capaces de reconocer determinados conceptos o clases. En muchas ocasiones, la interpretabilidad, que depende del número de reglas, se encuentra en compromiso con la exactitud, ya que modelos muy exactos suelen ser muy complejos e ininterpretables. En estos casos, se considera primero la simplicidad [HOLTE, 1993].

- *Modularidad.* Las reglas representan el conocimiento de forma global mediante la unión de conocimientos pseudo-independientes, es decir dentro del conjunto, cada regla unidad cubre una parte del problema, correspondiendo a una parte conocimiento requerido para la solución total. Este planteamiento corresponde a teorías reduccionistas, por las que se determina que un problema es igual a la suma de sus partes más las relaciones entre esas soluciones parciales. Esta modularidad implícita dota a este tipo de modelos de una gran flexibilidad que permite abordar más fácilmente problemas muy complejos. No obstante, justamente esta perspectiva reduccionista representa la única preocupación sobre estos

métodos de representación, ya que según DIETTERICH [1998] están limitados sólo al aprendizaje de pequeñas fracciones de todas las hipótesis posibles. Sin embargo, éste es un debate abierto dentro del área de la psicología y la cognición humana.

- *Ingeniería del conocimiento.* Las reglas son muy fáciles de crear, interpretar, verificar, manejar y utilizar, y durante el proceso de aprendizaje son generadas, modificadas, representadas o gestionadas de forma automática usando procedimientos cognitivos de razonamiento estándares.

- *Utilidad.* Las reglas son bien aceptadas en sistemas artificiales como los sistemas expertos o sistemas de ayuda a la decisión y tienen una estructura sencilla y pueden ser, por un lado, fácilmente adaptables a sistemas existentes y, por otro lado, pueden ser extendidas con otros métodos como como la lógica difusa.

- *Escalabilidad.* Cada una de las reglas de un modelo único representan conocimiento complementario, este hecho las concede flexibilidad y escalabilidad. Por ejemplo, para representar un problema complejo basta con añadir diferentes reglas que cubran cada parte del problema. No obstante, existe una limitación en cuanto al tamaño del conjunto de reglas, debido a que un excesivo número de reglas puede hacer que no sean comprensibles, o pueden incluso contradictorias.

El empleo de reglas ha sido un éxito en muchas aplicaciones, y entre ellas destacan los sistemas expertos, desarrollados en la década de 1980, utilizan y describen reglas de producción para la toma de decisiones y pueden realizar determinadas acciones cuando se producen ciertos eventos.

La estructura de las reglas es una *implicación lógica*, que consta de dos partes y que en lenguaje natural se expresa como:

```
SI antecedente ENTONCES consecuente
```

La parte SI (*IF*) es el antecedente, premisa, condición o situación; y la parte ENTONCES (*THEN*) es el consecuente, conclusión, acción o respuesta. El

antecedente está formado por los atributos A_N que tienen los ejemplos, y que definen las variables predictivas. El consecuente, por otro lado, está formado por un atributo preseleccionado para la clase C, donde $C \in A_N$. Tanto el antecedente como el consecuente están formados por unidades lógicas mínimas, llamadas en la literatura *literales* $< A_N = v_N >$, y que a su vez están formados por un atributo, su valor y un operador de comparación (por ejemplo =) que los une. Estos literales, además de proposiciones, pueden ser una comparación entre atributos como sucede en la lógica de primer orden. Cuando estos pares forman parte del antecedente son normalmente llamados *precondiciones*, o simplemente *condiciones*. La formación de un conjunto de condiciones utiliza los elementos típicos de la lógica proposicional (`NOT`, `AND` Y `OR`) explicados en la sección anterior, es decir diferentes proposiciones se combinan mediante conjunciones y disyunciones para formar antecedente según:

R: `SI` $[(A_1 = v_1)$ `AND` $(A_2 = v_2)]$ `OR` $[(A_N = v_N)]$ `ENTONCES` C

Teniendo en cuenta las características del operador OR, este conjunto de precondiciones puede ser convertido en el siguiente conjunto de reglas no excluyentes, es decir que un ejemplo puede ser cubierto por una o varias reglas del conjunto.

r1: `SI` $(A_1 = v_1)$ `AND` $(A_2 = v_2)$ `ENTONCES` C
r2: `SI` $(A_N = v_N)$ `ENTONCES` C

Fundamentalmente, según MITCHELL [1997] la búsqueda de reglas y árboles se realiza por técnicas basadas en ejemplos, que se pueden realizar fundamentalmente utilizando dos tipos de técnicas: a) técnicas *sistemáticas* o recursivas de tipo voraz (*Greedy*) utilizadas por los *algoritmos clásicos de inducción de reglas*, y b) los métodos *estocásticos* que utilizan las *técnicas evolutivas*. Ambos tipos buscan en el mismo espacio de soluciones, ya que la forma de representación es la misma y por lo tanto, ambos métodos permiten descubrir los mismos modelos.

La lista de publicaciones que presentan algoritmos clásicos de inducción de reglas y/o árboles es muy extensa [FÜRNKRANZ, 1999]. Destacan dos trabajos, por un lado, el trabajo anteriormente mencionado de [QUINLAN, 1986] que presenta el algoritmo *ID3* (*Iterative Dichotomizer*) para la construcción de árboles de decisión basándose en la división recursiva del conjunto utilizando como función de calidad basada en la entropía. El segundo trabajo es CLARK & NIBLETT [1989] que describe el algoritmo *CN2*, un método de especialización de construcción secuencial de conjuntos de reglas que va eliminando los registros que se van cubriendo. Otros algoritmos importantes son los siguientes.

Algoritmos de la familia de algoritmos de $ID3$ para la construcción de árboles son por ejemplo el $C4$, que está especializado en atributos numéricos continuos y utiliza operadores como $\leq$ o $\geq$, o el algoritmo *C4.5* [QUINLAN, 1993]. Dentro de esta familia $ID3$ prefiere árboles sencillos, es decir los caminos del árbol más cortos hasta la hoja y que proporcionan mayor información. Otro algoritmo clave en la literatura para la inducción de árboles es *CART* (*Classification and Regression Trees*) [BREIMAN ET AL., 1984] que construye árboles binarios (basado en la partición binaria de las variables) y utiliza el índice de diversidad de *Gini*. Una variante de este método es *QUEST* (Quick, Unbiased and Efficient Statistical Tree) [LOH & SHIH, 1997] que usa otros métodos estadísticos para la selección de los atributos. Los algoritmos *SLIQ* (*Supervised Learning in Quest*) y *SPRINT* (*Scalable PaRallelizable INduction of decision Trees*) construyen árboles en anchura en vez de en profundidad como los explicados y emplean un criterio de poda basado en el principio de longitud de descripción mínima (*MDL*).

Los algoritmos para la inducción directa de reglas también usan el principio de Quinlan, aunque ha sido paulatinamente modificado hacia lemas como *separa-y-vencerás* o *reconsidera-y-vencerás* [BOSTRÖM & ASKER, 1999], para reducir el efecto de la replicación por el que diferentes reglas definen al mismo grupo de registros. La metodología *START* de Michalski, utilizada por el mencionado *CN2*, destaca entre los métodos más populares de inducción directa de reglas. Los primeros algoritmos desarrollados en este entorno son *INDUCE*

[HOFF ET AL., 1983] y *AQ* [HONG ET AL., 1986]. Esta metodología constituye un conjunto de técnicas de aprendizaje, fundamentalmente incremental, que utiliza expresiones lógicas disyuntivas para incluir las distintas reglas en un único modelo. Ambos algoritmos repiten el proceso mientras que existan ejemplo positivos sin clasificar. Métodos más recientes combinan *CN2* y el teorema de Bayes para presentar un algoritmo para la clasificación, también busca una alternativa al solapamiento de las reglas resultantes [LINDGREN & BOSTRÖM, 2002, 2003]. Sin embargo estas técnicas presentan una mayor probabilidad de aumentar el número de reglas, es decir la complejidad. Otro grupo de técnicas de inducción es la familia de algoritmos *Ripper*, basados en la reducción del error durante la etapa posterior de poda [COHEN, 1995]. Además de algoritmos de búsqueda, existen sistemas que realizar la inducción en diferentes fases, como por ejemplo el trabajo propuesto en [WIDMER, 2003] que propone una primera fase de búsqueda de reglas parciales simples y robustas con un algoritmo tipo *Ripper*, que posteriormente sufren *clustering*, generalización y la selección según un heurístico basado en *Laplace*. Otros sistemas inductivos muy populares en el campo del aprendizaje automático y desarrollados concretamente en el marco de la programación lógica inductiva o ILP (*Inductive Logic Programming*) son *FOIL* [QUINLAN, 1990] o *ALLiS* [DÉJEAN, 2002]. Este último se basa en la lingüística con el fin de extraer las reglas más importantes que incluyan también excepciones y basándose en exactitud de los modelos. En esta misma línea de investigación, el sistema PROGOL realiza la búsqueda *general-a-específica* utilizando un gráfico de refinamiento [MUGGLETON, 1990]. Otro interesante sistema es *GAR* [RIAÑO & CORTÉS, 1997, RIAÑO, 1997], que también propone una proceso de aprendizaje multietapa basándose en selectores estadísticos y combinando técnicas de agrupamiento. Finalmente, mencionar los trabajos de MARON & LOZANO-PÉREZ [1998] y CHEVALEYRE & ZUCKER [2001] que proponen métodos para aprender más de una clase en un mismo proceso a partir de un sólo conjunto.

A pesar de buscar en el mismo espacio de búsqueda, los *algoritmos evolu-*

tivos, que se explican detalladamente en el siguiente capítulo, son diferentes a los algoritmos de inducción mencionados, debido fundamentalmente en la componente estocástica del proceso y consecuentemente, en el modo de construir las hipótesis. Mientras que los algoritmos de inducción hacen una búsqueda local modificando candidatos parcialmente mediante operadores determinísticos de especialización o generalización, (quitando o insertando precondiciones a la regla), los algoritmos evolutivos evalúan y construyen modelos globales y esto tiene la gran ventaja de que los modelos cubren mejor diferentes elementos del problema. A pesar de que este grupo de técnicas sufre un vacío de conocimiento y que realiza una búsqueda basada en la probabilidad, son unos métodos eficientes y robustos para la inducción de reglas. Estas características han provocado que los algoritmos evolutivos sean los paradigmas más utilizados para el descubrimiento de reglas [FREITAS, 2002b].

3.2. Descubrimiento de conocimiento en bases de datos

3.2.1. Sistemas de gestión de bases de datos

El descubrimiento de conocimiento en bases de datos (KDD) explicado en la sección anterior está estrechamente ligado a los sistemas de gestión de bases de datos. Básicamente proporcionan el almacén para los datos que se van a explotar, aunque comienzan a incluir procedimientos procedentes de la minería para el análisis de estos datos.

Los *sistemas de gestión de bases de datos* (DBMS), consisten en una colección de datos interrelacionados, conocidos como **bases de datos**, y un conjunto de programas de gestión y acceso a esos datos. Esos programas involucran mecanismos para definir la estructura de la base de datos, gestionar el almacenamiento y la proteger los datos y otras tareas relativas a la seguridad de la información.

Una base de datos relacional, que es el tipo de almacenamiento propuesto en esta tesis, es una colección de **tablas**, cada una de ellas con nombre único dentro del mismo sistema. A su vez, cada tabla consta de un conjunto de **atributos**, también llamados *columnas* o *campos*, que almacenan los datos como **instancias**, registros o tuplas. Cada uno de estos registros en una tabla relacional representa a un objeto, que se identifica por una **clave** única o identificador y se describe por un conjunto de valores correspondientes a cada atributo de la tabla. Este modelo, que es el más popular, recibe el nombre de modelo *entidad-relación*, es un modelo semántico que representa a una clase o conjunto de entidades y a las relaciones que existen entre ellas. Las bases de datos, además de datos, incorporan operaciones, funciones y métodos, entre otras herramientas, para extraer información específica y más concreta a partir de todo el almacén. Además de la capacidad de almacenar datos en la base de datos, acceder a ellos y actualizarlos, un sistema de gestión de bases de datos proporciona un catálogo en el que se almacenan las descripciones y características de los datos, que que se denomina *diccionario de datos* o

metadatos.

Por otra parte, un tipo específico de bases de datos que tiene especial interés en la actualidad son las espaciales, ya que son la base de los *Sistemas de Información Geográfica* (SIG). Según la definición tradicional, un SIG es entendido como un *conjunto de hardware, software, datos geográficos, personas y procedimientos, organizados para capturar, almacenar, actualizar, manejar, analizar y desplegar eficientemente rasgos de la información geográfica.* Una definición más actual establece que *un SIG es un sistema que mediante herramientas informáticas y datos geográficos ayuda a entender el mundo y resolver problemas complejos del mundo real* [BOSQUE-SENDRÁ, 1998]. Las bases de datos espaciales se diferencian de los otros tipos de bases de datos, fundamentalmente en las características espaciales de los objetos almacenados contienen y, como consecuencia, también en los métodos necesarios para gestionarlos. En una base de datos espacial las entidades u objetos almacenados como tuplas, incluyen los atributos normales de una base de datos y la información que los diferencia espacialmente, mediante dos atributos fundamentales la *localización* y la *geometría.* La localización se expresa en coordenadas geográficas, que pueden ser bien absolutas utilizando algún sistema de proyección cartográfico, o bien relativas cuando se utiliza un sistema de referencia propio. La geometría, por otro lado, atiende a la forma que presentan los objetos, que puede básicamente ser un punto, una línea o un polígono.

3.2.2. LENGUAJE SQL PARA LA EXPLOTACIÓN DE LAS BASES DE DATOS

En general, la información almacenada en las bases de datos puede ser recuperada mediante consultas que se escriben en un lenguaje de específico, como es el lenguaje estructurado SQL o mediante interfaces gráficas. Una consulta se transforma en un conjunto de operaciones relacionales, como puede ser, entre otras, la unión o la selección que permiten la recuperación de subconjuntos específicos de datos. Para realizar estas consultas, el usuario debe tener conocimiento de la estructura de la base de datos, lo que le permitirá extraer los subconjuntos de instancias que cumplen ciertas condiciones. El lenguaje estándar

ISO *SQL* (*Structure Query Languaje*), que surge a partir de la investigación del modelo de datos relacional de E. F. Codd de IBM como un lenguaje para la especificación de características de las bases de datos relacionales, se trata de un lenguaje específico de bases de datos y no de programación general, por lo que se usa embebido en otros lenguajes de programación como Cobol, C o Visual Basic. Su origen está en el lenguaje SEQUEL (*Structured English QUEry Language*) desarrollado, también, en IBM en los años 1974-75. Posteriormente, Oracle, el primer fabricante de sistemas de bases de datos, comercializa una implementación de SQL en 1979. Finalmente, IBM lanzó el producto SQL/DS (*Structured Query Language/Data System (IBM)*) en 1981. Los elementos que componen el lenguaje SQL son *comandos*, *cláusulas*, *operadores* y *funciones de agregado*, que se combinan para crear, actualizar y manipular las bases de datos mediante las denominadas consultas lógicas (boleanas). Se distinguen dos tipos de comandos, los comandos propios del lenguaje de definición de datos (DDL *Data Definition Language*) y los comandos del lenguaje de manipulación de datos (DML *Data Manipulation Language*). Los tipo DLL que permiten crear y definir nuevas bases de datos, campos e índices; y los DML, los interesantes desde el punto de vista de esta tesis, que permiten generar consultas para ordenar, filtrar y extraer datos de la base de datos. Los principales comandos DML son INSERT, DELETE, SELECT y UPDATE. El comando INSERT se usa en tareas de carga de datos en la base de datos y el comando DELETE que sirve para eliminar registros. De este grupo de comandos destacamos, porque son utilizados por la propuesta del presente trabajo, **SELECT** que se utiliza para realizar consultas de selección de grupos o recuperación de registros y **UPDATE** que sirve para la modificación de la información de los registros almacenados en los campos.

Otros elementos importantes son las cláusulas, que son condiciones para la modificación de datos concretos. Las principales cláusulas son FROM y WHERE para especificar la tabla y/o los registros que se quieren manipular. En concreto **WHERE**, que también es fundamental en nuestra propuesta, permite seleccionar registros cuyos atributos satisfacen determinadas condiciones.

Otras cláusulas son GROUP BY que permite separar los registros seleccionados, la cláusula HAVING que sirve para expresar una condición que debe satisfacer un grupo de registros y finalmente, ORDER BY cuya función es ordenar los registros según un orden determinado. El SQL usa los operadores lógicos AND, OR y NOT, igual que la lógica proposicional cuyas elementos principales han sido explicadas anteriormente. Igualmente, existen los operadores de comparación que se utilizan en las condiciones para recuperar los registros que cumplan las funciones del operador utilizado. Entre los principales operadores de comparación están *menor que* (<), *mayor que* (>), *menor o igual que* (<=), *mayor o igual que* (>=) para datos numéricos, e indistintamente de tipo de dato, se emplea *igual que* (=) y *distinto de* (<>). Finalmente, las funciones de agregado que se utilizan en una cláusula para calcular diferentes parámetros estadísticos a partir de un grupo de registros, y son AVG, para calcular el valor promedio, COUNT que devuelve el número de registros que tiene el grupo, SUM para calcular la suma de todos los valores de un campo determinado, y MAX y MIN que proporcionan el máximo y el mínimo de los valores del campo seleccionado. Finalmente, cabe destacar que SQL es un lenguaje que puede trabajar con los diferentes tipos de datos que se pueden almacenar en una base de datos, desde categorías hasta números pasando por fechas o valores lógicos.

La utilización de bases de datos para el descubrimiento de conocimiento presenta múltiples ventajas. En primer lugar, los datos suelen estar almacenados en este tipo de almacenes, y la mayor parte de los sistemas comerciales de análisis (SPSS©, Clementine©, etc.) permiten trabajar directamente con este tipo de archivos sin necesidad de transformar el formato de los datos. En segundo lugar, en determinado dominios como el ecológico los datos pueden ser recopilados en el campo mediante un ordenador portátil pudiendo ser almacenados directamente en una base de datos, por medio de aplicaciones como la que se describirá en una de las secciones posteriores (sección 5). Tercero, el lenguaje SQL, al cumplir las leyes del álgebra de Boole y lógica proposicional, hace que la expresividad de este lenguaje sea muy alta, permitiendo utilizar

palabras propias del lenguaje natural de los humanos. De hecho, una sentencia SQL es directamente interpretable, incluso por personas que no están familiarizados con él. Además, utilizar consultas permite conocer en un sólo paso el número de instancias que cumplen un conjunto de condiciones, es decir que una sola consulta es capaz de calcular cuantos registros satisfacen determinadas proposiciones, sin necesidad de recorrer cada uno de los registros. Y por último, pero quizás la más interesante de las ventajas es, se pueden transformar directamente a reglas. Algunas de estas ventajas, ya han sido descubiertas por investigadores que principalmente trabajan en el campo de la *recuperación de información* (*Information Retrieval*) presentando sistemas basados en bases de datos y en lenguajes de consulta como el SQL [CHEN & SHE, 1994, THOMAS & SARAWAGI, 1998]. La importancia de los lenguajes de consulta sobre bases de datos es tal, que muchas investigaciones se han centrado en el desarrollo de un lenguaje propio para tareas de minería para bases de datos inductivas, como por ejemplo DMQL (*Data Mining Query Languaje*) de [HAN ET AL., 1996] donde se describe la especificación de un lenguaje para tareas de generación de reglas de asociación, reglas discriminantes, reglas de clasificación y reglas características, que además tiene en cuenta que las expresiones de consulta preprocesen los datos. *Dmajor* es propuesto en [IMIELINSKI ET AL., 1999] y permite la generación de reglas selectivamente, extrayendo categorías de reglas a partir de colecciones de reglas. En [MEO ET AL., 1996], se propone un nuevo operador del lenguaje SQL, llamado LIKE, cuya sintaxis permite únicamente generar reglas de asociación. En [MEO, 2003] se presentan las implicaciones prácticas de un conjunto de algoritmos diseñados basados en ese operador. Otro lenguaje de consulta para un lenguaje específico de minería, propuesto por primera vez en [IMIELINSKI & MANNILA, 1996], se utiliza para seleccionar reglas en postproceso, lenguaje implementado en sistemas como *SMART*SKIP [HIPP ET AL., 2002].

Convencionalmente, los lenguajes de consulta, como el SQL, se emplean fundamentalmente para para recuperación, manipulación y selección de registros de una o varias tablas. Básicamente, estas tareas tratan de obtener

resultados a partir de una consulta, proceso que muestra la figura 3.8. Para realizar este tipo de tareas, es necesario conocer el contenido de los datos a la hora de ejecutar una consulta, y conocer exactamente las características de las instancias que se quieren recuperar.

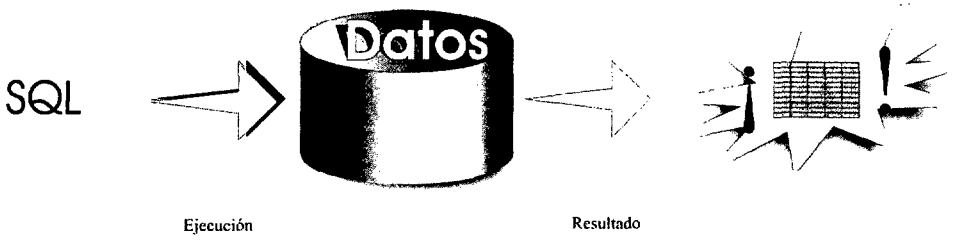


Figura 3.8: Uso convencional del lengunaje SQL para recuperar un conjunto de instancias de una base de datos

Desde el punto de vista del descubrimiento de datos, interesa la dirección contraria de este proceso, es decir se conoce un conjunto determinado de elementos, y sería deseable conocer características que comparten, y por lo tanto, encontrar la consulta que incluye esas características. A este tarea, se le ha denominado *derivación de consultas a partir de resultados* [RYU & EICK, 1996a], y requiere un proceso iterativo de construcción, ejecución y evaluación de cada consulta generada, contrastándola con la base de datos, como muestra la figura 3.9. La propuesta de esta tesis está directamente relacionada con esta última idea, la construcción de la consulta que recupere los registros que pertenecen a una clase, sin recuperar aquellos que no pertenecen.

Igual que los conjuntos de reglas, la búsqueda de estas consultas puede realizarse con técnicas evolutivas. Por ejemplo, en [PETRY ET AL., 1997] se presenta un sistema sencillo basado en algoritmos evolutivos para descubrir una consulta de suficiente calidad. El sistema utiliza un heurísticos basado en la calidad de la información recuperada. SATTLER & DUNEMANN [2001] presentan un algoritmo evolutivo de creación de arboles de decisión, que utiliza e implementa los primitivos y operaciones de SQL útiles para tareas de cla-

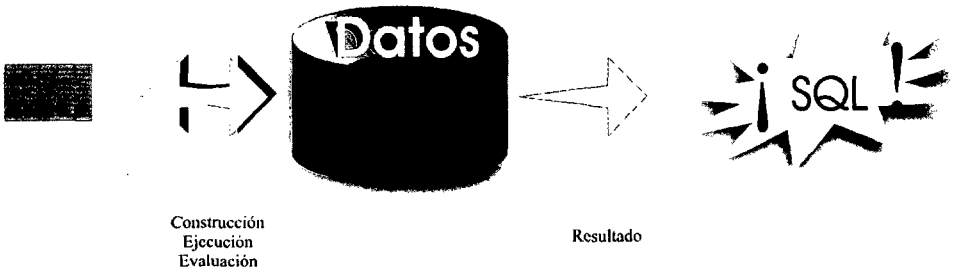


Figura 3.9: Uso en minería de datos para descubrir una consulta a partir de un predeterminado conjunto de instancias

sificación. Otro sistema denominado MASSON, presentado en [RYU & EICK, 1996b], es una herramienta que emplea programación genética para encontrar automáticamente consultas sintácticamente correctas, con el fin último de descubrir las características comunes entre un grupo de registros. Los cromosomas también codifican funciones, que ellos denominan de navegación, como SELECT de SQL y otros operadores como UNION, INTERSECTION, etc. En [SALIM & YAO, 2002] también se emplean algoritmos evolutivos, pero existen algunas diferencias entre ambos sistemas como, por ejemplo, el modo en el que se representan los cromosomas, incluyendo en las consultas, de forma aleatoria, los conectores lógicos AND y OR. Estos últimos autores, además, no utilizan operadores convencionales de cruce y mutación, sino que los incluyen como genes dentro de los cromosomas. Finalmente, el lenguaje de consulta puede utilizarse simplemente como herramienta adicional de manipulación de registros y para contar registros y no como base de representación de los modelos [FREITAS & LAVINGTON, 1996, KRIEGEL ET AL., 2000].

Capítulo 4

INDUCCIÓN DE REGLAS USANDO TÉCNICAS GENÉTICAS

4.1. INTRODUCCIÓN A LA COMPUTACIÓN EVOLUTIVA Y LA BÚSQUEDA GENÉTICA

Los algoritmos evolutivos proponen la búsqueda estocástica para el aprendizaje de modelos o la solución a un determinado problema, y utilizan determinados mecanismos aleatorios y heurísticos para evitar la búsqueda exhaustiva. La búsqueda se hace dentro de un espacio N-dimensional en el que se combinan todas las variables (N) que definen dicho problema.

El origen de la computación evolutiva está relacionada con la investigación del científico J. von Neumann (1903-1957) en *autómatas celulares*, que basándose, incluso antes del descubrimiento del ADN, en la idea que los seres vivos y los mecanismos de evolución debían regirse por un código que los describiera y que se transmitiera a descendientes, concibió estas máquinas autoreplicativas. Posteriormente y utilizando todas estas ideas, John Holland comenzó a aplicar la teoría de autómatas celulares a problemas de adaptación y optimización, en la creación de un simulador genérico y en el desarrollo del *algoritmo genético básico* [HOLLAND, 1975]. Esta línea de investigación (la *computación evolutiva*) es un ejemplo de como la informática se ha nutrido de las ciencias biologías para su desarrollo, al igual que otros campos como la visión artificial, redes neuronales, la robótica multiagentes o vida artificial,

que utilizan ideas basadas en modelos biológicos presentes en la naturaleza y que funcionan. También la biología también ha sacado partido de los avances informáticos en el desarrollo de áreas como la genética para la construcción del genoma de diferentes especies (bioinformática), en la medicina para la diagnosis [BOJARCZUK ET AL., 2001] o entre otras muchas aplicaciones, el modelado de la distribución de especies (como ejemplo ver las investigaciones de David R.B. Stockwell (*Advances computational to enviromental and biodiversity information (BIODI)* [1])). La computación y adaptación digital de procesos biológicos determina diferentes niveles de complejidad; a) el nivel inferior, simplemente la *computación*, donde están ubicados los algoritmos genéticos, b) niveles de *computación inteligente* que engloba a los sistemas de alto nivel de abstracción en la representación como los sistemas borrosos o redes neuronales artificiales, y c) finalmente, niveles de máxima complejidad de la *computación natural* de la geometría fractal y la vida artificial.

La computación evolutiva utiliza como inspiración la teoría de la selección natural de las especies de Charles R. Darwin (1809-1882), en concreto el funcionamiento molecular de los organismos, la genética y los principios de la adaptación, constituyendo una línea sólida de investigación que ha permitido el desarrollo de técnicas robustas para multitud de aplicaciones que buscan optimizar y resolver problemas representados digitalmente. Por ejemplo, se han empleado en informática para tareas de aprendizaje con el objetivo de conocer el funcionamiento de los sistemas adaptativos; en ingeniería para problemas de optimización de procesos industriales cuyo objetivo puede ser minimizar costes, material defectuoso o el tiempo de un proceso; en la teoría de juegos donde se busca maximizar la probabilidad de ganar; en robótica para la navegación en entornos inciertos; en el desarrollo de la ciencia cognitivas descubriendo modelos de pensamiento; en el dominio de la física para modelar el mundo real; en la biología en el estudio de la genética; y finalmente, en el aprendizaje de conceptos donde se intenta minimizar el error de clasificación de modelos basándose en ejemplos conocidos, este último estrechamente relacionado con

[1] *http://biodi.sdsc.edu/index.html*

la inducción de reglas, tema que aborda esta tesis.

Estas técnicas evolutivas se explican a través de la genética. Desde el punto de vista de la biología, la información genética de los organismos, denominada *genoma*, esta en los *cromosomas*, que consisten en un conjunto de genes que controlan la herencia y las funciones celulares. En el proceso de evolución es fundamental que las células reciban la colección completa de cromosomas que las caracterizan como especie para garantizar su supervivencia. La evolución, por lo tanto, es consecuencia de la *supervivencia* de los individuos más adaptados que son capaces de reproducirse, así como de la propia *reproducción*, que asegura la recombinación del material genético entre los mejores individuos. Ambos conceptos determinan la selección natural. Estas variaciones genéticas suponen que unos individuos estén mejor adaptados que otros para reproducirse y sobrevivir a determinados ambientes, entonces, según esta teoría, los caracteres hereditarios de los individuos mejor adaptados estarán más representados en las generaciones siguientes. Sin embargo, además de la recombinación en la evolución son importantes fenómenos de mutación, que provocan la alteración del material genético y que produce cambios de los individuos que son transmitibles por herencia. En definitiva, la reproducción y la mutación parecen ser las claves de la evolución [BARREIRO, 1996].

Todos estos elementos de la genética son la base de los algoritmos evolutivos, e implementan el concepto de adaptación para realizar tareas de aprendizaje. En el marco de la computación evolutiva, esta implementación se realiza según cuatro ramas principales: la *Programación Evolutiva*, las *estrategias evolutivas*, los *algoritmos genéticos* y la *programación genética*. Todas ellas tienen características comunes básicas basadas en las ideas genéticas anteriormente expuestas, pero presentan algunas diferencias relacionadas básicamente con la representación de los individuos y consecuentemente, de los operadores que los modifican durante el proceso. Por ejemplo, las *estrategias evolutivas* utilizan la mutación como el principal operador para la explotación de todo el espacio de soluciones, y codifican también en un individuo los parámetros que controlan la probabilidad de mutación. Por otro lado, la *programación evolutiva* tampo-

co utiliza el cruce, y la tasa de mutación varía durante el proceso, siguiendo una distribución concreta. Los *algoritmos genéticos* dan más importancia a la operación de cruce, aunque también utiliza la mutación. Estos últimos buscan la verdadera simulación de un mecanismo genético natural. Finalmente, la *programación genética* es una variación de los algoritmos genéticos, donde los individuos codifican programas además de los datos, es decir, se representan tanto los datos como los operadores y funciones que modifican la información. Estas cuatro ramas se pueden reagrupar en dos grupos más generales según la forma que tienen de trabajar. Mientras que las *estrategias evolutivas* y *programación evolutiva* se preocupan de la relaciones progenitor-descendiente y tienen en cuenta elementos que contienen información del dominio y del proceso como por ejemplo, observar tasa de aprendizaje, los *algoritmos genéticos* y la *programación genética* se dedican a simular de la mejor forma posible los procesos evolutivos naturales, y no tener en cuenta que sucede entre generaciones, son ajenos a cualquier información del dominio.

4.2. Fundamentos de un Algoritmo Genético

La propuesta de esta tesis se basa en los *algoritmos genéticos clásicos* de Holland [1975], que proporcionan una herramienta flexible y eficaz, que permite solucionar problemas de naturaleza muy diferente, adaptando únicamente la representación y la función de calidad [Freitas, 2002a].

Este tipo de algoritmos mantienen una población de individuos, de los que cada uno es una posible solución a un determinado problema, es decir una hipótesis. En este proceso, la probabilidad de que un determinado material genético pase a la siguiente generación viene determinada por la calidad del individuo, es decir, a la bondad como solución, que en este ámbito recibe el nombre de *fitness*. Durante el proceso de aprendizaje, los individuos son modificadas utilizando operadores genéticos como el cruce o la mutación que son procedimientos estocásticos. Progresivamente, los algoritmos evolutivos van construyendo hipótesis, que tienden a una misma representación, es decir el mejor modelo, este fenómeno se conoce como *convergencia*.

En cualquier paradigma de computación evolutiva destacan los siguientes componentes básicos:

(1) La *representación genética* de las soluciones.

(2) Un método para *generar la población inicial.*

(3) Una *función de evaluación* que representa el entorno del problema.

(4) *Operadores* que alteren la composición genética de los individuos durante el proceso de reproducción.

(5) Un *mecanismo de selección* de los individuos supervivientes de una generación.

(6) Determinar el *mecanismo de selección* de los padres y los valores para determinados parámetros de entrada como son el tamaño de la población o los umbrales de probabilidad de mutación y cruce.

La heurística es también un elemento básico de un proceso evolutivo. De hecho, cada elemento de un algoritmo genético, como los operadores genéticos, la elección de probabilidades y otros parámetros, la representación y estructura cromosómica para conseguir individuos viables o la generación de la población inicial, suelen utilizar criterios heurísticos que dirigen el proceso hacia la convergencia [MICHALEWICZ, 1996b]. Asimismo, la función de evaluación que guia el proceso, en si misma, se puede considerar un heurístico.

De entre los elementos enumerados, uno de los elementos clave en el diseño de estos algoritmos, es la **representación de un individuo** y del espacio de búsqueda en general, y como consecuencia la determinación de los operadores que mejor se ajustan al problema. La representación es la conexión entre el mundo real y el proceso genético [EIBEN & SMITH, 2003]. Las posibles soluciones del problema en la realidad, normalmente referidos como *fenotipos*, se representan dentro de un genético como *genotipos*. Entonces, se denomina, *espacio de los fenotipos* al conjunto de todas las posibles soluciones del problema real, mientras que al conjunto de puntos donde el algoritmo realiza la búsqueda se le llama *espacio de los genotipos*. Ambos espacios pueden ser muy diferentes, y puede que el mejor genotipo encontrado sea un buen fenotipo, una buena solución, pero no la mejor. Para la representación de cada genotipo, de aquí en adelante *individuo*, se utiliza el *cromosoma* tradicionalmente construido como una cadena de bits[2]. Estos cromosomas utilizan generalmente una representación binaria y son de longitud fija, no obstante la representación de algunas soluciones mucho más complejas, puede requerir cromosomas de longitud variable y/o utilizar números o letras.

La terminología dentro del campo de la computación evolutiva emplea numerosos sinónimos para designar a los mismos elementos de un sistema genético, que es conveniente diferenciar. Se destacan a continuación los más comunes. Por ejemplo, una **solución candidata**, un fenotipo o un **individuo** representan cada posible solución. Por otro lado, genotipo, **cromosoma** y de nuevo

[2] La palabra bit, procedente de *BInary digiT*, incorporada al diccionario de la real academia de la lengua española, es utilizada para designar a un carácter o dígito cuyo valor es binario, es decir, 1 ó 0

individuo, se utiliza para denominar cada hipótesis del espacio de búsqueda del algoritmo. Finalmente, para designar a las variables de un problema se emplean los términos posición, variable, **locus** (**loci** en plural) o *gen*, y los valores que puede adoptar son denominados **alelo**.

El segundo asunto fundamental en el diseño de una aplicación basada en genéticos, es la **función de evaluación** que en computación evolutiva comúnmente llamada *fitness*. Esta función garantiza la optimización porque formula matemáticamente los requisitos que deben poseer los individuos, y determina por lo tanto su calidad como solución. El diseño de esta función depende de diferentes factores como, la capacidad de procesado de datos, el conocimiento que se tenga del problema y los requisitos marcados por el problema o por el propio usuario.

Finalmente, los **operadores genéticos** (cruce, mutación, selección, etc.) son también importantes en el proceso, porque son los responsables de la transformación de los individuos durante el proceso, y de la exploración en diferentes áreas del espacio de búsqueda. La tasa de transformación debe ser adecuada al proceso, siendo importante que exista un equilibrio entre la explotación de los mejores candidatos y la exploración del espacio de búsqueda [EIBEN & SMITH, 2003]. La explotación y exploración son mecanismos dirigidos por el proceso selección y por los operadores de cruce y mutación respectivamente [SPEARS & DE JONG, 1992].

Básicamente, durante un proceso genético se construyen aleatoriamente y verifican continuamente diferentes hipótesis, tal como muestra la figura 4.1. La verificación de la calidad de cada hipótesis se realiza mediante el contraste con el conjunto de ejemplos. En cada iteración, los individuos son modificados con los operadores genéticos.

La secuencia genética comienza con la generación aleatoria de un conjunto de hipótesis, que constituye la *población inicial*. La nueva población es evaluada utilizando la función *fitness*, por la que a cada individuo se le asigna un valor numérico de calidad. El siguiente paso es la selección de individuos según un determinado criterio heurístico, por ejemplo los individuos de mayor

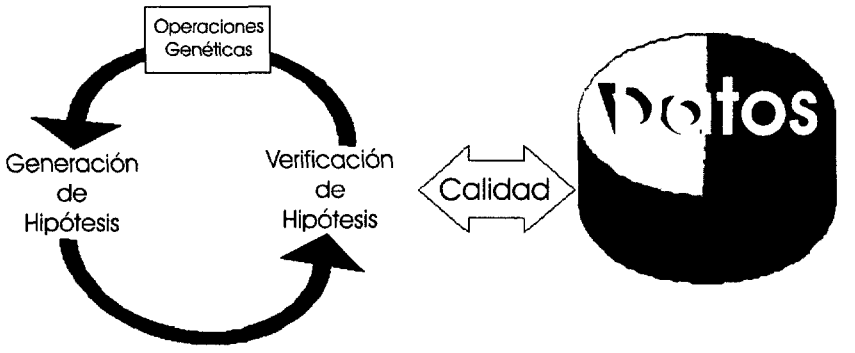


Figura 4.1: Funcionamiento básico de un algoritmo genético

calidad tienen mas probabilidad de formar parte de la siguiente generación. A continuación, el cruce por el que algunos de estos individuos seleccionados, siempre que superen una determinada probabilidad de cruce, son recombinados, es decir, el material genético de estos individuos se mezcla. El siguiente proceso es la mutación, en este caso de los individuos seleccionados y cruzados que superen una probabilidad de mutar modifican algunos bits elegidos aleatoriamente. Tras estos procedimientos, los nuevos individuos son evaluados, y se les asigna la nueva calidad. El siguiente paso, el elitismo se utiliza para forzar la convergencia del proceso, si éste está activado el mejor individuo de la generación anterior reemplaza al peor de la nueva generación. En este punto, se ha generado la nueva población y el proceso termina si se cumple una determinada condición de parada -como puede ser un número de iteraciones o se alcanza un cierta calidad- en caso contrario, el proceso continúa hasta que la condición se cumple.

Las siguientes líneas muestran el *pseudo-código* de programación que implementa un algoritmo genético.

```
Sub genetico (f,l,p,r,m)
        t=1
        Crea pob(1); l,p
        Evalua pob(1); f
        Do while condicion = true
             t=t+1
             Seleccion pob(t)
             Cruce pob(t); c
             Mutacion pob(t); m
             Evalua pob(t); f
             If elitismo
        Loop
End Sub
```

donde `f` es la función de calidad para evaluar las hipótesis, `p` es el tamaño de la población, `l` es la longitud de los individuos, `c` y `m` son tasas de cruce y mutación, `t` el número de generaciones y `pob` representa una población, y que es la población inicial cuando `t=1`. Los parámetros `f`, `p`, `c` y `m` son datos de entrada del sistema.

A continuación, se describe con más detalle cada procedimiento que compone un algoritmo genético básico.

- `Crea pob(1); l,p` es un procedimiento aleatorio de creación de la primera población de individuos (`pob(1)`). Esta parte del proceso requiere como datos de entrada `l` la longitud de los individuos que componen la población y `p` el número de individuos de cada población. Este proceso genera los cromosomas de todos los individuos de esa primera población, en los que cada bit de la cadena puede adoptar aleatoriamente el valor 0 ó 1. Otra vía para la generación de la población inicial es utilizar una *semilla*, que es un individuo que no ha sido construido aleatoriamente y cuya representación y calidad es conocida. El empleo de la semilla resulta interesante en ciertas situaciones, por ejemplo cuando debido a la complejidad del problema, todos los individuos de la primera generación tiene

un valor de calidad (fitness) igual a cero, hecho que impide la continuación del proceso de evolución, o cuando se desea comenzar este proceso de aprendizaje a partir de un punto concreto del espacio porque se tiene esa información.

- `Evalua pob(1); f` es el paso en el que se determina la calidad de los individuos utilizando una función establecida por el usuario, que evalúa la capacidad que tiene de solucionar el problema planteado y asigna un valor numérico a su fitness.

- `Seleccion pob(t)` (*selección*) es un procedimiento que elige a los individuos de una generación con mayor probabilidad de éxito, es decir, a los individuos considerados mejores según su valor de fitness. Existen principalmente tres métodos para realizar la selección de los individuos de una población: el método de la ruleta, la ordenación y el torneo. Los tres tienen como idea utilizar el valor de calidad para realizar una selección proporcional, dando mayor probabilidad a los que tienen mayor fitness.

  –*Ruleta:* se adjudica una probabilidad de selección a los individuos usando porciones de una ruleta cuyo tamaño es función de su fitness, tal que la suma acumulada de todas es el tamaño total de la ruleta. Las áreas mayores corresponden a los individuos de mayor calidad (figura 4.2). Posteriormente, se eligen los individuos cuya probabilidad de selección (área en la ruleta) supere determinado valor de probabilidad ρ elegido aleatoriamente. Por lo tanto, los individuos que presenten una mayor superficie de la ruleta tendrán una mayor probabilidad de ser seleccionados para posteriores operaciones genéticas, y consecuentemente, tienen más posibilidades de trasmitir sus cromosomas a las siguientes generaciones. Sólo es válido para procesos en los que se busca la maximización de la función, y no de minimización, ya que requiere que al menos un individuo de la primera generación tenga una fitness diferente a cero.

  –*Ordenación:* Es quizás el procedimiento más sencillo. Se ordena todos los individuos según su calidad, y de nuevo, los individuos son seleccionados

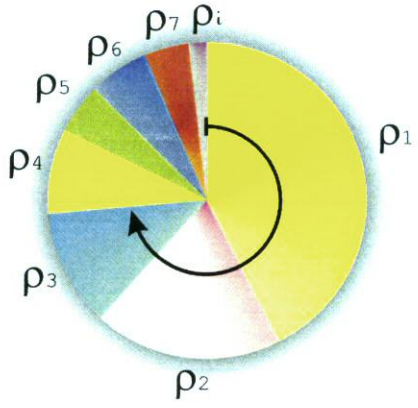


Figura 4.2: Método de selección de la ruleta

según su probabilidad.

–*Torneo:* Se elige un subconjunto k de individuos aleatoriamente, y se elige el que mayor calidad tenga, y se proclama vencedor del torneo. Este método es una variación del anterior, sin embargo el torneo puede ser implementado cuando se realiza aprendizaje en paralelo, es decir, distintos ordenadores buscan la solución de un determinado y único problema, ya que realiza un muestreo de tamaño k del total de la población. Tanto el torneo como el *raking* pueden trabajar en procesos de minimización y además, pueden operar sobre fitness con valor negativo.

- `Cruce pob(t); c` representa al *Cruce* o recombinación, y es el procedimiento por el que dos individuos progenitores comparten sus genes para generar dos nuevos individuos hijos. Este proceso es explorativo y realiza, a través del intercambio de material genético ya evaluado, saltos entre el área del espacio que existe entre los progenitores. El cruce representa a la reproducción sexual de los organismos, y procura una nueva generación utilizando el material genético de la generación anterior, pero recombinado, es decir, que los descendientes heredan de ambos progenitores, y son diferentes entre si pero también diferentes a los padres. Existen diferen-

tes tipos de cruce, que podemos clasificar para su explicación como en un punto (tipo I en la figura 4.3), en dos puntos (tipo II) y cruce uniforme (tipo III). En los tipo I y II, los puntos de cruce, determinan aleatoriamente el punto o los puntos que dividen a cada cromosoma progenitor en dos o tres partes respectivamente, que posteriormente intercambian para generar los dos nuevos individuos, tal y como muestra la figura. En el tipo de cruce III, ambos progenitores recombinan la mitad de los genes alternando cada posición (figura 4.4).

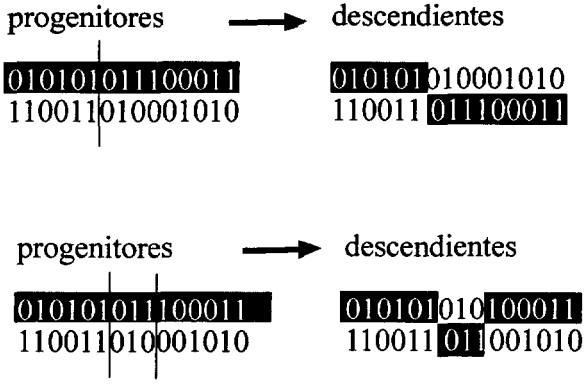


Figura 4.3: Esquema de la operación genética de cruce entre individuos progenitores (a) tipo I y (b) tipo II

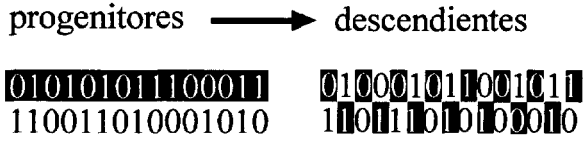


Figura 4.4: Esquema de la operación genética de cruce tipo III

- `Mutacion pob(t);` `m` representa la etapa de *mutación*, y se encarga cambiar el valor de un bit elegido aleatoriamente de un progenitor por el valor contrario en el nuevo individuo. Por ejemplo, si un bit, elegido aleatoria-

mente que además supera la probabilidad de mutar, tiene un valor 1, la mutación lo convierte en un 0 (figura 4.5). Existen distintas formas de mutar un individuo. La forma más básica es la *puntual* en la que se invierte un sólo bit de la cadena. El procedimiento de mutación más drástico permite que todos los bit de la cadena muten. Mientras que el cruce interviene en la exploración del espacio de búsqueda, la mutación se caracteriza por ser proceso de explotación de las regiones no exploradas del espacio, ya que puede incluir información genética nueva (no examinada), provocando saltos cercanos pero fuera del área entre los progenitores.

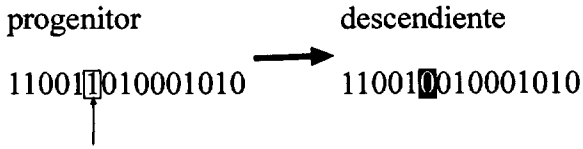


Figura 4.5: Esquema de la operación genética de mutación entre individuos progenitores

- La fase de `elitismo` es un proceso que asegura el paso del mejor individuo de una generación a la siguiente. Este operador puede implementarse de distintas maneras, por ejemplo mediante el denominado elitismo unitario por el que el peor individuo de una generación es sustituido por el mejor de la población anterior.

- Finalmente, la condición de parada (`condicion = true`) es un criterio que determina cuando se concluye un proceso. Teóricamente, el proceso de aprendizaje genético finaliza cuando la fitness alcanza el máximo global, es decir se produce la convergencia, que puede requerir un tiempo considerable. Es frecuente, por lo tanto que se utilice algún otro criterio que evite un proceso se alargue demasiado tiempo, como: 1) parar el proceso cuando se han realizado un número específico de iteraciones (el más utilizado) y 2) cuando algún individuo alcance un determinado valor de fitness que el usuario considera suficiente.

Tras exponer las fases de un algoritmo genético es probable que se desprenda la idea de que este mecanismo algorítmico depende del azar para conseguir la convergencia y hallar la solución. Sin embargo, lo cierto es que los algoritmos genéticos y otros procesos estocásticos utilizados tienen una importante formulación matemática. Estudios como los presentados en [GOLDBERG, 1989, SCHMITT, 2000, 2004] describen los fundamentos matemáticos y probabilísticos del todo proceso y de los diferentes operadores. La explicación más básica de por qué funcionan los algoritmos genéticos se encuentra en *el teorema de esquemas* enunciado por el propio Holland. Los *esquemas* son conjuntos de cadenas compuestas por los valores {0, 1 (cuando la codificación de cromosoma es binaria) ó *}, donde cada valor * actúa de comodín del resto de valores. Según esta teoría los individuos que presentan un mayor valor de calidad convergen hacia un esquema común. Supongamos un conjunto de individuos entre los que destacan por su calidad dos individuos {101,100}. Como se puede observar estas cadenas tienen los dos primeros bits en común, y se puede definir el esquema 10*. Progresivamente, el algoritmo hace que los individuos de las sucesivas generaciones presenten en mayor proporción individuos con este esquema y la población converge hacia esta representación. Por otro lado, el número de realizaciones posibles que contiene un esquema responde a la expresión 2^k donde k es igual al número de comodines (*). Por lo tanto, cada cadena puede verse como un representante de cada uno de los distintos esquemas que satisface. Es decir, la cadena 101 representa a 2^4 esquemas, como el 10*, el 1*1, el ***, etc. Como consecuencia, estos algoritmos funcionan por un fenómeno denominado comúnmente como *paralelismo implícito*, por el cual el algoritmo en una sola población procesa una enorme cantidad de información, que además es continuamente seleccionada y modificada.

4.3. APRENDIZAJE DE REGLAS CON ALGORITMOS GENÉTICOS

Una vez explicado el funcionamiento básico de los algoritmos genéticos, prestamos atención a la representación de los modelos y su codificación de los

cromosomas, una de las claves de nuestra propuesta. La forma de codificar los individuos juega un papel importante dentro de este proceso de aprendizaje, ya que la elección de una u otra representación puede hacer que el algoritmo trabaje de forma más o menos eficiente y que además represente de la mejor forma la solución real de problema estudiado. Los cromosomas son capaces de codificar cualquier problema.

La codificación de los conjuntos de reglas (apartado 3.1.3) que queremos en nuestra propuesta se puede realizar de dos modos [MICHALEWICZ, 1996a]: a) la representación de *Michigan* en la que la solución total del problema es representada por todo el conjunto de individuos de la población, y por lo tanto, cada individuo es una solución parcial del problema y todos son complementarios; y b) la representación de *Pittsburg* (Pitt) para la que cada individuo es una posible solución total y compite contra otros individuos de su población. A pesar de que ambas representaciones presentan ventajas y desventajas, la representación de Pitt es la aproximación más popular porque la comparación entre individuos resulta menos compleja al realizarse en una única población, y facilitando así la fase de selección de los mejores candidatos.

El aprendizaje de reglas con técnicas evolutivas está directamente relacionado con los sistemas clasificadores. De acuerdo con la definición de GOLDBERG [1989], un sistema clasificador es un sistema de aprendizaje automático que aprende reglas sintácticamente simples para guiar su actuación en un entorno variable. El primer sistema clasificador fue el *CS-1* de Holland y Reiter (1978) citado en [GOLDBERG, 1989], que aprendía reglas de producción usando algoritmos genéticos y basándose en la recompensa de las acciones exitosas. Este sistema utiliza la representación de Michigan. La representación de Pitt, más adecuada para ambientes estáticos, es utilizada por los sistemas descritos en [BASSETT, 2002] donde se pueden encontrar las siguientes referencias: LS-1 de Smith (1980) aprende conjuntos de reglas simbólicas y utiliza un mecanismo que penaliza soluciones complejas; GABIL desarrollado por De Jong et al.(1993), similar al anterior, usa una representación binaria para codificar

la disyunción del conjunto de reglas (OR); el sistema GIL de Janikow (1993) aprende reglas simbólicas y la principal diferencia con los anteriores es que éste incluye operadores especializados para el aprendizaje inductivo de Michalski basado en ejemplos; también el sistema REGAL de Giordana y Saita (1994) y Giordana y Neri (1996) aprende descripciones de conceptos simbólicos utilizando la representación de Michigan y Pitt conjuntamente y además utiliza, aunque de forma limitada, el operador NOT; Sandip Sen y Leslie Knight (1995) presentan un prototipo, PLEASE, para el cual una regla define un único punto del espacio y la clasificación se hace mediante una aproximación a los vecinos más cercanos; Carse y Fogarty (1994) desarrollaron un sistema que aprende reglas fuzzy (borrosas) y para concluir, el sistema DELVAUX de Eick et al. (1996) que aprende conjuntos de reglas bayesianas a partir de ejemplos. Antes de concluir el estado del arte relativo al aprendizaje de reglas mediante computación evolutiva, cabe destacar trabajos en el marco ecológico, como la investigación de David R.B. Stockwell y su equipo (véase por ejemplo [STOCKWELL, 1999]), cuya principal aportación es el sistema GARP usa reglas bayesianas para el modelado de datos geográficos almacenados en imágenes raster. Este sistema permite diferentes tipos de reglas (rangos numéricos, regresiones, etc.), pero que pueden ser difíciles de interpretar debido a que dan como resultado reglas numéricas. Otros sistemas que según JEFFERS [1999] son destacables son: el BEAGLE de (forsyth, 1981) que descubre árboles de decisión a partir de variables numéricas que requiere la binarización (presentar sólo dos categorías) y la monitorización y guia de un experto en el dominio durante la búsqueda; y el sistema GAFFER (South, 1994) que permite descubrir reglas para problemas de predicción/clasificación también a partir de variables numéricas, según el mismo autor, dada su flexibilidad resulta complejo de utilizar.

Como se puede desprender de esta extensa enumeración, la última década ha sido una etapa importante para la investigación de este tipo de métodos de inducción de reglas mediante técnicas evolutivas y los sistemas análogos, área que en la actualidad continua creciendo. Los trabajos más recientes de este campo son los realizados por el equipo de investigación dirigido por A.A.

Freitas dedicados al desarrollo de las técnicas evolutivas aplicadas a la minería de datos en bases de datos relacionales y la inducción de reglas discretas y también borrosas (*lógica fuzzy*) [FREITAS, 2002b,a].

En general, en todos sistemas genéticos de inducción de reglas, el proceso de aprendizaje es similar, fundamentalmente basado en ejemplos y por lo tanto, supervisado. La evolución y convergencia del genético sucede al encontrar paulatinamente individuos con mayor capacidad de clasificar correctamente el grupo de instancias de partida. Las diferencias que tienen los métodos enumerados no son muy significantes y están relacionadas con la forma de representar y codificar las reglas. No obstante, la representación suele seguir también unas pautas generales. Por ejemplo, la codificación de los cromosomas sigue un esquema que distingue el consecuente del antecedente de las reglas. En primer lugar, la parte del consecuente, la clave del aprendizaje supervisado, codifica a la variable objetivo. A pesar de que existen trabajos de clasificación [NODA ET AL., 1999] que realizan una codificación "al-vuelo (*on-the-fly*)[3] del consecuente. Normalmente, la parte de la cadena que contiene al consecuente es la misma para todos los individuos, y por lo tanto, también para todas las poblaciones. Por otro lado, la parte del cromosoma que representa al antecedente codifica las N variables según el esquema que muestra la tabla 4.1 donde $< Act_N, v_N, OpC_N >$ representa cada precondición ($Cond_N$), mientras que $< OpL_{N-1} >$ codifica a los operadores lógicos. La parte Act_N es un parámetro de activación que determina si la variable es o no incluida en un modelo candidato, v_N codifica el valor o etiqueta del atributo representando el par $< gen, alelo >$ (atributo-valor), y finalmente, OpC_N que codifica el operador de comparación ($=, \neq, >, <, etc$). El parámetro de activación ofrece la posibilidad de descubrir reglas que no contengan todos los atributos del estudio, construyendo sólo las asociaciones entre condiciones que determinan la mejor solución. Así mismo la desactivación de todos las precondiciones de una regla,

[3] Codificar al vuelo el consecuente es entendido como el mecanismo por que se incluye el consecuente en la cadena, provocando que el proceso de aprendizaje también se encargue de encontrar una de las clases (la que mejor se adapta a los ejemplos) de la base de datos. O que sea capaz de encontrar e incluir reglas de diferentes clases en el mismo modelo

permite eliminar dicha regla del modelo.

$< Act_1, v_1, OpC_1 >$	$< OpL_1 >$	$< Act_2, v_2, OpC_2 > \ldots$	$< OpL_{N-1} >$	$< Act_N, v_N, OpC_N >$
Cond_1	OpL	$\text{Cond}_2 \ldots$	OpL	Cond_N

Tabla 4.1: Esquema general para codificar el antecedente de una regla

En definitiva, la codificación de un regla en un cromosoma utilizando una representación simbólica se basa en la siguiente idea. Supongamos los dos atributos predictores H y G, con dos posibles valores cada uno {a, b} y {c, d}, respectivamente, y por último, la variable de clase C, que divide el conjunto de datos en dos grupos; el de ejemplos positivos p si $C = p$ y el de ejemplos negativos n si $C = n$. Los cromosomas, en primer lugar, incluyen el parámetro de activación del primer atributo H, se codifica con un bit *si* este atributo se activa (1) o *no* en el modelo (0). En segundo bit del cromosoma representa a uno de los valores del primer atributo a o b, y el siguiente bit codifica el operador de comparación OpC que puede ser $\{\neq, =\}$. Hasta aquí, la primera condición de la regla estaría codificada. A continuación, la estructura necesita un bit para unir esta precondición con la siguiente mediante el operador lógico OpL que puede valer {`AND`, `OR`}. El siguiente paso para la codificación del antecedente es incluir la segunda precondición para el atributo G, que utilizaría el mecanismo explicado. En este ejemplo, cada precondición usa un total 3 bits, y un antecedente con las dos precondiciones utiliza 7 bits. Siguiendo el procedimiento explicado, por ejemplo la cadena cromosómica {0101100} representaría el antecedente ($H \neq a$ `AND` $G = d$) y el modelo completo para la clase P sería:

0101100 $\Leftrightarrow$ (Si H $\neq$ a AND G = d entonces C=p)

Este mecanismo de codificación es empleado principalmente para variables nominales, que codifican las distintas categorías de los atributos [DE JONG ET AL., 1993]. Para la codificación de variables numéricas, se puede, o bien, realizar una etapa previa de categorización y usar la misma técnica, o bien, diseñar un código para los valores numéricos, decimales o enteros. Uno de los

códigos estándar más populares es el código Gray [GRAY, 1953], que codifica números enteros adyacentes en valores binarios, resultando muy útil para la etapa de mutación, ya que pequeños cambios en la cadena binaria representan importantes cambios en la decodificación. A pesar de la existencia de estas técnicas para codificar números, el esquema de codificación categórica es más popular, por ser más simple y genérico, porque en esta representación simbólica, cada bit de un cromosoma representa un elemento individual de información, y son elementos indivisibles. Además, esta característica hace que el diseño de los operadores genéticos que no sea muy complejo. Asimismo, los atributos numéricos siempre pueden transformarse en categorías, hecho que, sin embargo, no sucede en sentido contrario. La aproximación simbólica descrita, asegura por lo tanto, una metodología escalable y flexible.

Por otro lado, la codificación de modelos muy complejos podría requerir cromosomas longitud variable en un mismo proceso, o bien, caracteres alfanuméricos en vez de ceros y unos, ambas técnicas involucran un enorme esfuerzo de codificación y un mayor control de cada parte del cromosoma durante las operaciones genéticas. Estas y otras ideas forman parte de la línea de investigación que implementa las llamadas técnicas de auto-adaptación (*self-adaptation*), dentro de la programación genética, en la que destacan los trabajos de Thomas Bäck, por ejemplo [BÄCK ET AL., 2000]. Sin embargo, cualquier método utilizado para codificar los cromosomas depende de cada problema, de los datos y del objetivo del experto.

Parte II

Implementación de un sistema para descubrimiento de conocimiento (KDD)

Sistema para descubrimiento de conocimiento (KDD)

En esta parte de la memoria se describe la implementación de un sistema compuesto por modulos independientes, desarrollado para realizar tareas de descubrimiento de conocimiento en bases de datos explicadas en el capítulo 3.

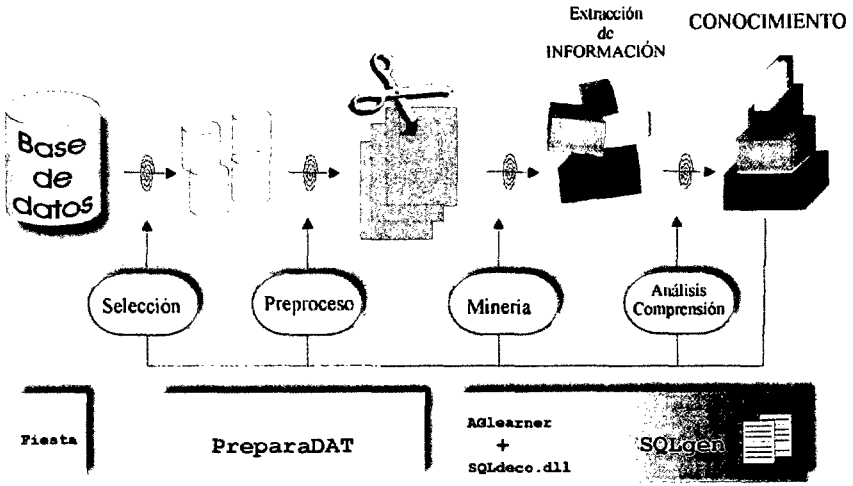


Figura 4.6: Diferentes herramientas para cada etapa del sistema KDD propuesto

Cada una de las fases de un proceso KDD se ha llevado a cabo con las aplicaciones desarrolladas que se muestran en la figura 4.6 y que son:

- `Fiesta` - Aplicación para la recogida y almacenamiento de datos georeferenciados en campo en una base de datos Microsoft Access©.
- `PreparaDAT` - Aplicación para la preparación de los datos almacenados en una base de datos relacional.

- La combinación `AGlearner + SQLdeco.dll` - Aplicación para la etapa de minería de datos. Se basa, por un lado, en la implementación de un algoritmo genético como algoritmo de búsqueda de la mejor hipótesis, y por otro lado, una librería DLL basada en el lenguaje SQL como lenguaje de consulta a la base de datos.

- `SQLgen` - Aplicación de evaluación y presentación, mediante una visualización adecuada, de los modelos obtenidos al usuario final en la etapa de minería.

Los siguientes capítulos, de esta segunda parte, describen cada una de las aplicaciones enumeradas.

Capítulo 5

ADQUISICIÓN DE DATOS GEOREFERENCIADOS EN CAMPO: *El muestreo*

5.1. INTRODUCCIÓN

El muestreo es una técnica de recogida de información, comúnmente utilizada en aquellas ciencias en las que es de interés estudiar las variaciones espaciales de determinadas características, es decir la heterogeneidad espacial.

El principal problema que presentan los muestreos es que son tareas duras y tediosas debido a la gran cantidad de información y material que hay que recoger en campos, que habitualmente son grandes (por ejemplo de varias hectáreas en el caso agrícola). Con el fin de facilitar las tareas de muestreo existen en el mercado herramientas de ayuda que guían al usuario en la recogida de información. Ahora bien, estos sistemas presentan ciertos inconvenientes como por ejemplo tener un formato propio que implica la adquisición de diferentes paquetes informáticos para las distintas etapas de adquisición de datos y posterior análisis de los mismos. En otras palabras las aplicaciones utilizan diferentes tipos de ficheros para los datos, consecuentemente fuerzan a conocer y aprender el funcionamiento de esos formatos, que frecuentemente no son genéricos. Además, estas aplicaciones son cerradas, lo que imposibilita el desarrollo de nuevas herramientas en lenguajes estándares de programación. Desde el punto de vista de los dispositivos físicos o *hardware*, estos sistemas se venden de forma integral, es decir que es necesario comprar los receptores de la señal de localización (habitualmente receptores GPS - *Global Positioning System*)

junto a terminales compactos para la gestión del sistema completo. Por tanto, se tratan de sistemas no modulares cuyas actualizaciones obligan, en muchos casos, a desechar totalmente el equipo para comprar uno nuevo con alguna mejora.

Para evitar las desventajas mencionadas se ha diseñado e implementado el sistema de recogida de datos georeferenciados que se presenta en este capítulo.

5.2. SISTEMA FIESTA

El sistema desarrollado de adquisición de datos georeferenciados, bautizado con el nombre de FIESTA (*FIEld Sampling TAsks System*) está implementado en el lenguaje de programación Visual Basic© 6 y se basa en las siguientes bibliotecas de funciones:

- ⋆ GeoMedia Object© y *GDO Server*© (*Geographic Data Object*) para el acceso, automatización y modificación de datos geográficos.
- ⋆ Microsoft DAO© (*Data Access Object*) para la generación y modificación general de la base de datos en formato Microsoft Access. El formato de la base de datos de Microsoft Access© (*.mdb) presenta varias ventajas como son la facilidad de edición, la posibilidad de exportar/importar datos de otras aplicaciones y la interoperabilidad entre diferentes almacenes geográficos.
- ⋆ Lenguaje SQL© (*Structure Query Language*) para la gestión de la base de datos

La figura 5.1 muestra el esquema general del sistema FIESTA, que comienza en el laboratorio con el paquete COPLAS (*COnfiguration PLAn Sampling*), con el que se planifican todas las estrategias de muestreo que se van a practicar y además se crea la estructura de la base de datos que contendrá la información espacial recogida. Ya en el campo, se utiliza la aplicación FIEL (*FIELd data collection subsystem*) que usa la información y los metadatos definidos en el

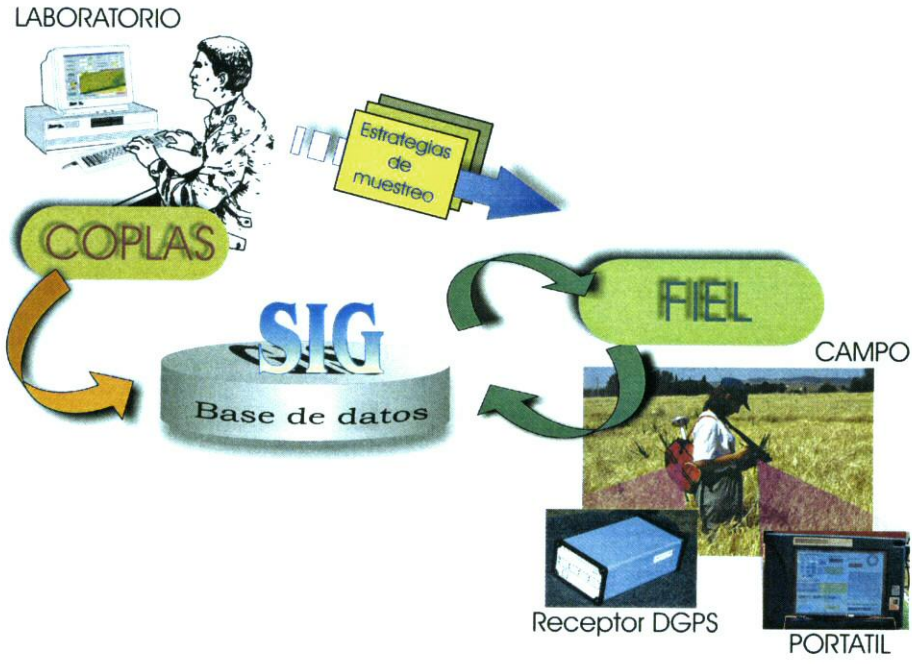


Figura 5.1: Arquitectura software del sistema FIESTA

laboratorio con COPLAS para guiar y dirigir al usuario a los puntos de muestreo. Cada una de las etapas, de definición de muestreo y de recogida de información en campo, se pueden repetir tantas veces como sea necesario hasta obtener la campaña completa de muestreos.

5.3. Sub-sistema de *Configuración de muestreos*

COPLAS es una aplicación que permite personalizar fácilmente cualquier proyecto de muestreo y recopilación de información, definiendo estrategias propias. El sistema ayuda al usuario a crear la estructura de una base de datos geográfica SIG, apropiada para el almacenamiento de la información de campo, sin la necesidad de poseer conocimientos en administración de bases de datos [Díaz & Ribeiro, 2002].

COPLAS comienza con una sencilla pantalla (figura 5.2) que presenta cuatro posibilidades, que son:

1. *Crear*: Aparece una secuencia de pantallas para la creación de un nuevo proyecto.

2. *Abrir*: La misma secuencia se presenta para editar un proyecto previamente generada.

3. *Grid*: Esta opción permite incluir o eliminar estrategias de muestreo.

4. *Ensayos*: Mediante esta opción se accede a una pantalla que permite añadir sesiones y ensayos para recoger los datos deseados.

Figura 5.2: Menu principal del sistema COPLAS

A continuación se explican brevemente todas las opciones partiendo de la creación de un nuevo proyecto. Cuando se pulsa la opción *Crear*, aparece la pantalla que se muestra en la figura 5.3 y que permite la configuración del proyecto. A través de este formulario se incluyen datos como el nombre, la localización, la duración de la campaña, características generales físicas y biológicas de la zona de estudio, así como datos de las malas hierbas y del cultivo. Esta es la información más genéricas del proyecto.

Cuando el usuario ha cumplimentado el formulario y presiona el botón *Guardar*, la aplicación COPLAS activa la secuencia automática de procedimientos que se muestra en la figura 5.4. De forma sucinta se crea una base de datos y se añaden las tablas denominadas *Geometry*, encargadas de almacenar la

información geométrica, es decir, la información de *localización* y *forma* de los objetos. En el caso concreto de las tareas de muestreo en el campo los objetos son los puntos de muestreo definidos únicamente por sus coordenadas x e y. Asimismo, en este mismo paso se generan todas las tablas que contienen los metadatos geográficos.

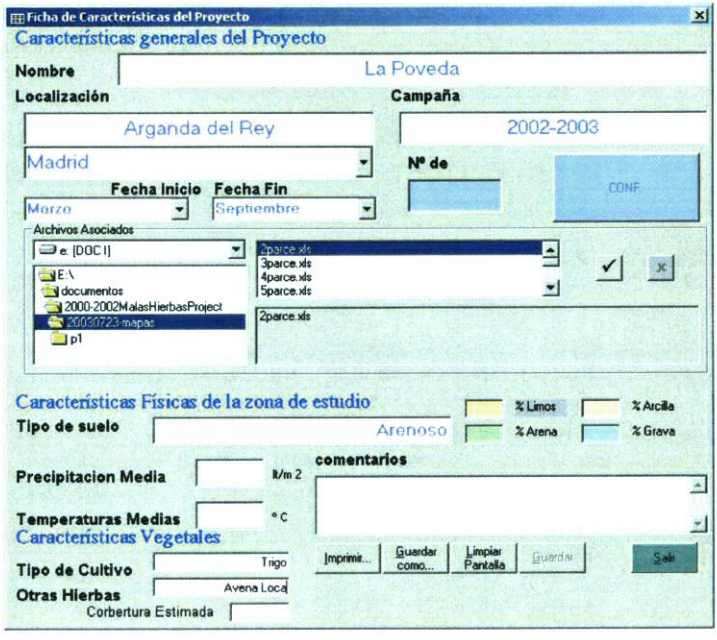


Figura 5.3: Pantalla de configuración de un proyecto

A continuación la aplicación genera las otras tablas de metadatos de los datos no geográficos, y que contienen la información general suministrada por el usuario en ese primer formulario. Finalmente, se generan todas las relaciones necesarias entre las tablas. En este instante la aplicación COPLAS ha creado *la base de datos espacial* de forma automática.

Una vez guardada la información del primer formulario de la figura 5.3, COPLAS presenta al usuario una segunda pantalla (figura 5.5), que corresponde a la opción *Ensayos*. Mediante esta pantalla, se pueden definir todas las

Figura 5.4: Secuencia de eventos y procedimientos para la creación automática de un SIG

estrategias de muestreo, personalizando de esta forma el proyecto. El usuario dispone de una lista que presenta las mallas o retículas definidas hasta el momento (en el cuadro superior derecho de la pantalla). En el caso de que la lista esté vacía, se debe crear al menos una malla. Para ello, el usuario seleccionará el botón de la opción *Mallas* lo que dará lugar a la aparición en la pantalla del formulario de la figura 5.6.

Esta nueva pantalla ofrece dos posibilidades: a) definir la malla indicando el número de puntos en los ejes x e y, la distancia entre puntos y la orientación de cada eje, introduciendo el azimut en grados; y b) definir la malla a través de una foto o croquis de la zona de muestreo. En este caso la imagen se presenta en la parte inferior de la pantalla que se sirve de base al usuario para dibujar los puntos de cada eje. La malla queda definida al guardar esta información.

En esta versión inicial de `COPLAS` los muestreos son únicamente de tipo malla regular por ser los más comunes. Esto último no significa que el segunda herramienta implementada `FIEL`, que presentaremos a continuación, no per-

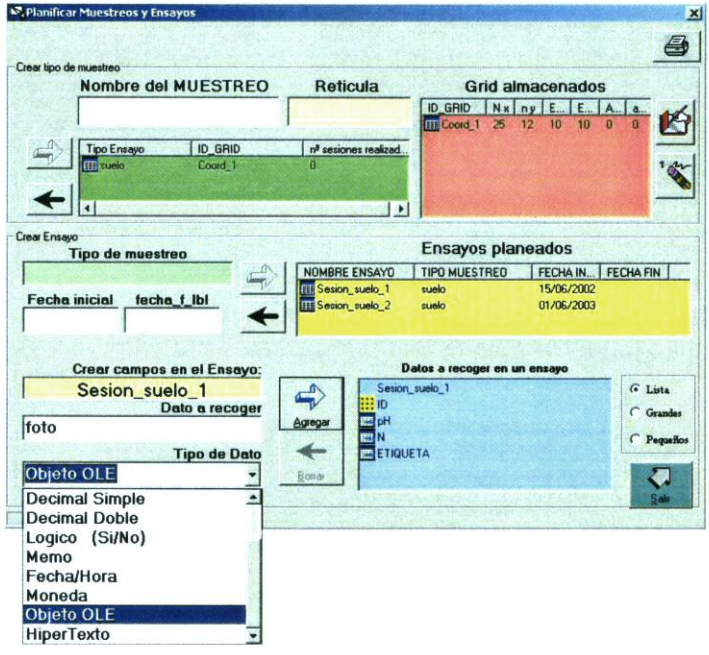


Figura 5.5: Pantalla de creación y edición de sesiones y ensayos

mita el movimiento libre por el campo y la recogida de información en puntos que no estén incluidos en la malla, evitando de este modo situaciones en las que los puntos de muestreo pueden ser de difícil o imposible acceso, como por ejemplo cuando coinciden con zanjas, hoyos, rocas, etc.

Una vez diseñada y almacenada la malla se pueden crear sesiones de muestreo asociando la malla generada a los tipos de muestreo y recogida de datos que se desea llevar a cabo. Esto último permite realizar un experimento en una fecha concreta con una malla determinada (*sesion1*) y repetir el experimento en otra temporada (*sesion2*). Los datos que se recogen y se almacenan pueden ser de muchos tipos por ejemplo una estimación de la cantidad de mala hierba en un punto que se almacenaría como un número o una etiqueta, una muestra recogida del suelo que se almacenaría como la etiqueta de la bolsa que contiene la muestra, un comentario sonoro o escrito, o incluso, una foto de la zona mues-

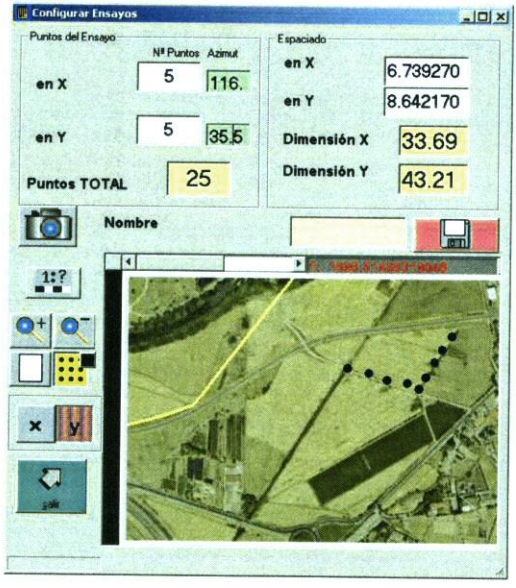


Figura 5.6: Pantalla de creación y edición de retículas

treada que se guardarían, o bien, como un archivo o como un enlace al archivo que contiene la información. De forma más precisa, COPLAS permite utilizar para las variables los mismos tipos de datos que ofrecen el formato Access©, es decir *cadenas* (text), *entero* (interger), *sencillo* (single), *largo* (long), *decimal* (double), *byte*, *lógico* (boolean), *texto largo* (memo), *Fecha* (date/time), *Moneda* (currency), *documentos* y *objetos OLE* (*Object Linking and Embedding*), es decir, incluir cualquier tipo de archivo informático.

En este punto, el proyecto ha sido definido por completo y se ha creado la estructura de la base de datos que se muestra en la figura 5.7. Esta base de datos que sigue el modelo entidad relación, del que se habló en el apartado 3.2, que incluye tablas de metadatos, tablas específicas del SIG Geomedia© a para la información espacial y el esqueleto de las tablas que contendrán los datos de campo.

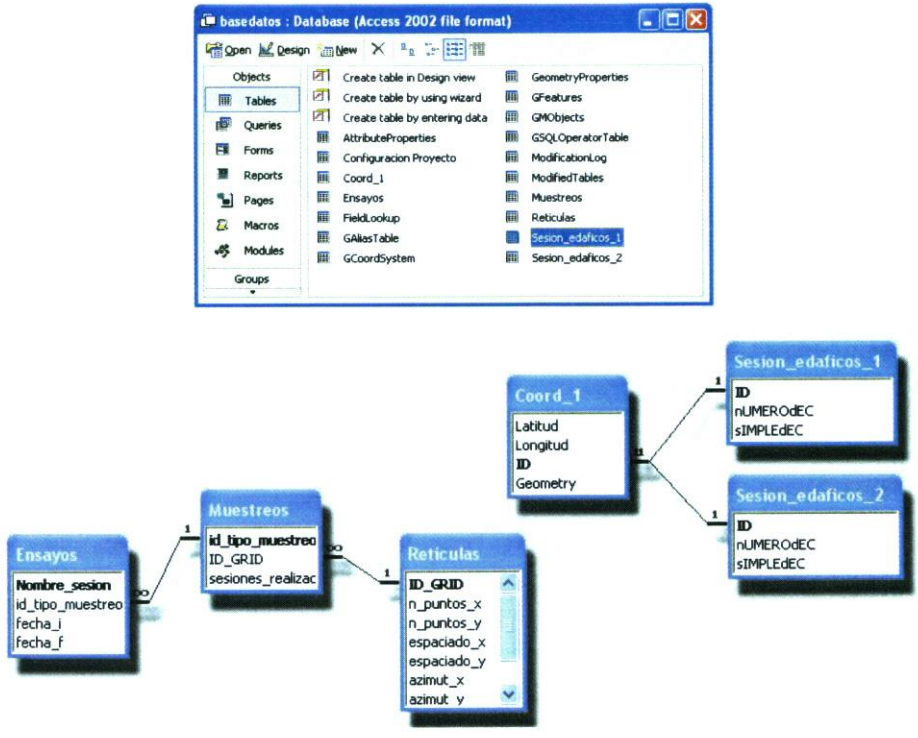


Figura 5.7: Las tablas de la base de datos espacial y relacional resultantes pueden ser editadas y visualizadas desde la aplicación Microsoft Access©

5.4. Sub-sistema de *Adquisición*

Iniciamos este apartado comentando brevemente el equipo necesario para llevar a cabo la tarea de muestreo en campo tal como se presenta en la figura 5.8. El equipo externo utilizado consta de un receptor GPS y un ordenador portátil (figura 5.10). Se trata de un receptor *Ashtech*© de doce canales con corrección diferencial servida en código en tiempo real por *OmniSTAR*©[1] que trabaja con un protocolo estándar de transferencia de datos (NMEA). El receptor suministra la localización corregida con una precisión submétrica. El

[1] http://www.omnistar.com/

ordenador portátil es una máquina convencional con una pantalla especial para exteriores, y la comunicación entre ambos dispositivos es a través de un puerto COM serie RS232.

Figura 5.8: Fotografías que muestran la colocación de los dispositivos en un muestreo a pie

El sistema GPS esta diseñado para calcular la localización geográfica de cualquier punto del globo. No obstante, diversos elementos como la desincronización entre relojes, entre otros, hacen necesaria la eliminación de los errores del cálculo utilizado técnicas de corrección diferencial (DGPS). El sistema de corrección diferencial por vía satélite que proporciona tecnología *OmniSTAR*©, desarrollada y comercializada por el grupo *Fugro*, permite obtener valores de localización corregidos para tener calidad submétrica en el 90 % de la superficie terrestre a tiempo real. Este mecanismo de corrección evita la utilización de una estación base local que sirva dichas correcciones, que supone, en primer lugar duplicar el equipamiento del usuario; segundo, la necesidad de localizar un punto lo suficientemente alto capaz de transmitir la corrección a todos los puntos del campo a muestrear, y además conocer su posición geográfica exacta para trabajar con coordenadas absolutas. El sistema RASANT© permite igualmente la corrección en tiempo real de las medidas de localización GPS, en este caso a través de una frecuencia de radio FM, sin embargo, también es necesario un dispositivo adicional que las reciba, además existe un error relacionado con la distancia entre el receptor del usuario y la

antena de referencia que calcula la corrección.

Para conocer las limitaciones en tiempo real del receptor de partida se realizaron varios estudios de caracterización de la calidad de la señal. En el primer análisis se almaceno la localización proporcionada por el DGPS para un punto fijo y de posición conocida [DÍAZ & RIBEIRO, 2000b], y se desprende que la exactitud en la localización varía a lo largo de un periodo de veinticuatro horas. En el área de la información geográfica, la calidad de los dispositivos de localización se puede definir mediante dos conceptos: *exactitud*, que se entiende como el número de veces que el valor de localización obtenido es correcto, y *precisión*, que es el grado de resolución de la medición, por ejemplo, métrico o centrimétrico. Atendiendo a estas dos medidas de calidad, y a partir del estudio de calidad mencionado, se observa que el valor de localización de dicho punto esta por debajo de los dos metros de error según una exactitud del 83 %, por debajo del metro con un 60 %. Este último valor permite caracterizar la precisión del receptor GPS como submétrica. Eventualmente, se dan valores que están por encima de los cinco metros, correspondiendo al 2 % de las medidas que fueron tomadas. Esta variación de la medida a lo largo del tiempo, que parece responder a fenómenos periodicos, y que puede relacionarse con la existencia de aparatos antiguos en la constelación de satélites GPS, y que provocan, cuando es utilizada su señal, una degradación el cálculo global de la posición. Lógicamente, esta variación repercute a tareas de localización en campo que se realizan en movimiento. Un segundo estudio de calidad se realizó mediante la medición de los puntos de una malla predefinida en diferentes momentos de tiempo [DÍAZ & RIBEIRO, 2000a], que demuestra que el error no es aleatorio, ya que el error es análogo en los diferentes puntos de la malla para una misma sesión, pero es diferente entre sesiones tal y como muestra la figura 5.9, donde se representan con diferentes símbolos y colores las coordenadas GPS obtenidas en diferentes sesiones (horas) realizadas en tres días diferentes. Este error de exactitud en la localización con apariencia sistemática, tal y como se propone en el estudio mencionado, puede subsanarse empleando determinados puntos o ubicaciones, permantentes, del campo durante el muestreo, que

sirvan de referencia para el calibrado de ese error, y convertir las diferentes mediciones de un mismo punto en un único valor. Otro método alternativo es realizar diferentes mediciones y calcular un valor de centralización, como puede ser la media.

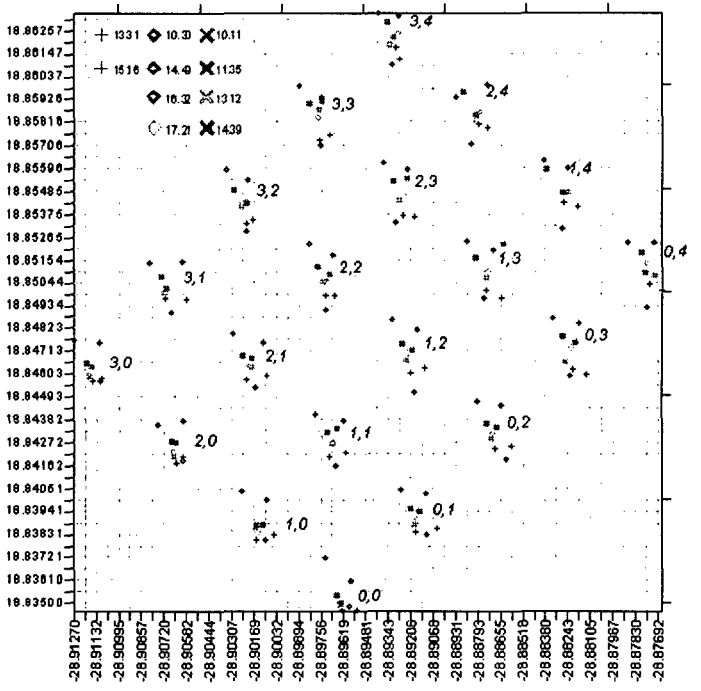


Figura 5.9: Distribución de los valores de localización GPS para los diferentes puntos de una malla predefinida tomados en distintos momentos de tiempo

Como vemos la disminución de la calidad pueden ser debida al propio sistema GPS y sus elementos, pero también a otros fenómenos como son los apantallamientos provocados, por ejemplo, por árboles. Sin embargo, esta tecnología de posicionamiento global está en continuo avance y los receptores actuales llegan a precisar la posición en escala centimétrica, y cada vez a menor precio. Cabe recordar que el sistema desarrollado de muestreo para la investigación aquí descrita, es capaz de trabajar con cualquier receptor que emita una de las

tramas estándar NMEA, la trama GGA, a través del puerto de comunicaciones.

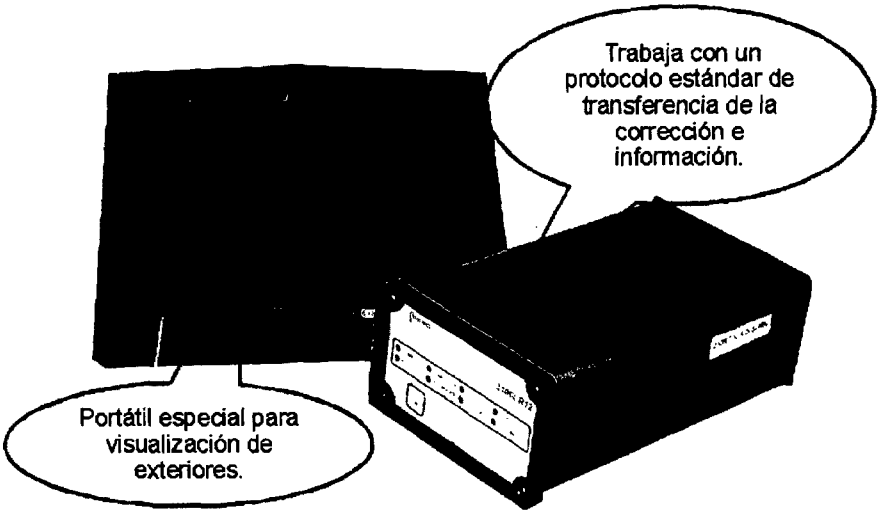


Figura 5.10: Requisitos externos al sistema (*Hardware*)

Como ya se mencionó al comienzo del capítulo, FIEL es una aplicación basada en DGPS/SIG de ayuda y guia en el campo durante el proceso de muestreo [Díaz & Ribeiro, 2005]. Ofrece un software intuitivo y amigable que puede ser utilizado por usuarios no muy expertos en el que se personalizan las vistas del mapa que representa el campo en el que se realiza el muestreo. También ofrece un sistema de coordenadas fácil de entender con unidades adecuadas como son la proyección UTM o las coordenadas geográficas. Utiliza también una visualización adecuada a exteriores que tiene en cuenta las limitaciones en el tamaño de la pantalla y las condiciones extremas de iluminación. Verifica e informa de los errores que se puedan producir en el muestreo, como por ejemplo la recepción de una señal GPS incorrecta o degradada, lo que evita que el usuario pierda la concentración en las tareas de muestreo. Por último, reiterar que los datos son recogidos y almacenados georeferenciados en el SIG en tiempo real lo que evita la etapa de volcado de datos posterior que presenta

otras herramientas.

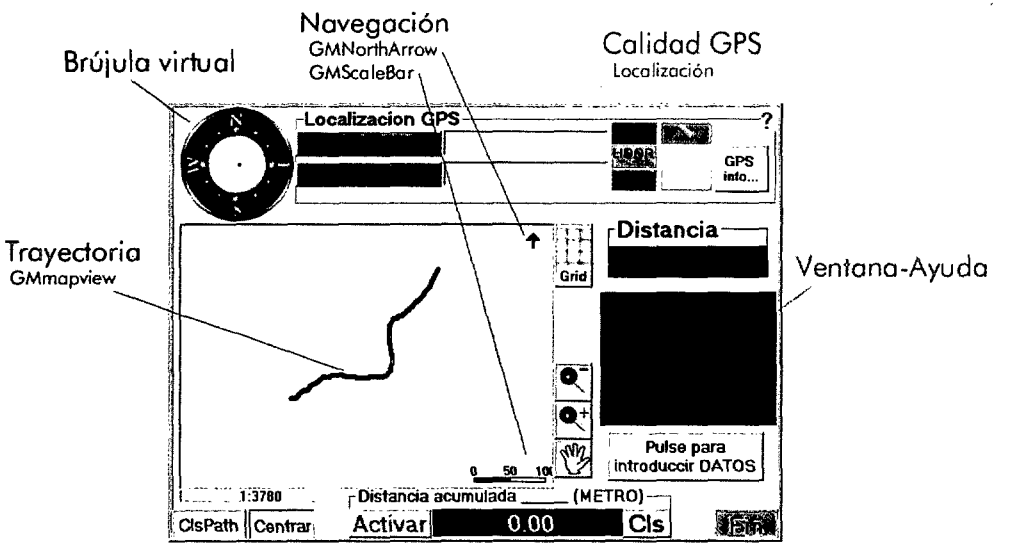


Figura 5.11: Herramientas principales del navegador

Cuando se ejecuta la aplicación FIEL, se selecciona el proyecto que se desea acometer. Al abrirse el proyecto se abre también la base de datos y se visualizan todas las sesiones disponibles y preprogramadas en la etapa anterior con COPLAS. La selección de una sesión muestra los datos que se deben recoger en ese muestreo. Por ejemplo, recoger los identificadores de las bolsas con las muestras y dejar la información química para rellenarla en laboratorio.

A continuación FIEL despliega la pantalla de navegación en *modo Localizar*. La figura 5.11 muestra todos los útiles que esta pantalla de navegación proporciona al usuario para ayudar a éste en la tarea de navegación y recogida de información. Entre los más importantes se encuentran:

- Para la *navegación* la pantalla dispone de diferentes elementos como una "brújula" virtual, una barra que indica la escala de la vista del mapa

(*GMSscaleBar*), también se expresa de forma numérica y, además, una flecha que indica el norte (*GMNorthArrow*).

- En cada instante se visualiza la *trayectoria* recorrida y punto donde se encuentra el usuario.

- El usuario puede conocer las coordenadas UTM o las obtenidas del receptor GPS, coordenadas geográficas, de cada punto de la ventana de navegación sin más que señalar el puntero del ratón.

- La *calidad* de la información del GPS aparece en cajas de texto con fondo de color amarillo mientras la señal se recibe correctamente y de color rojo cuando la señal se degradada.

- Existe una *ventana de información* que indica al usuario en todo momento los pasos que debe seguir, por ejemplo, en para ubicar definitivamente la malla empleada en el campo utilizando coordenadas geográficas o el estado del proceso de almacenamiento de los datos.

Una vez desplegada esta pantalla, el usuario puede cargar la malla seleccionada (cuadrícula rosa en la figura 5.12) para la sesión seleccionada, tan sólo presionando el botón *Grid*.

A continuación comienza el procedimiento de recogida. Cuando el usuario decide que ha alcanzado un punto de muestreo, esté o no en la malla, puede adquirir la información presionando el botón *OK*. Automáticamente, aparecerá una batería de pantallas como las que se muestran en la figura 5.13 que sirven para incluir los diferentes datos planteados para esa sesión, y definido en la fase de diseño realizada con el modulo anteriormente explicado `COPLAS`. Estos datos son, además de las coordenadas, la etiqueta que recibe la bolsa de la muestra, un comentario o por ejemplo, una foto de punto.

La aplicación *FIEL* también puede funcionar en *modo Relocalizar* que permite visualizar los puntos de las sesiones anteriores, es decir, de las tablas contienen datos. En este modo de funcionamiento, la pantalla de navegación

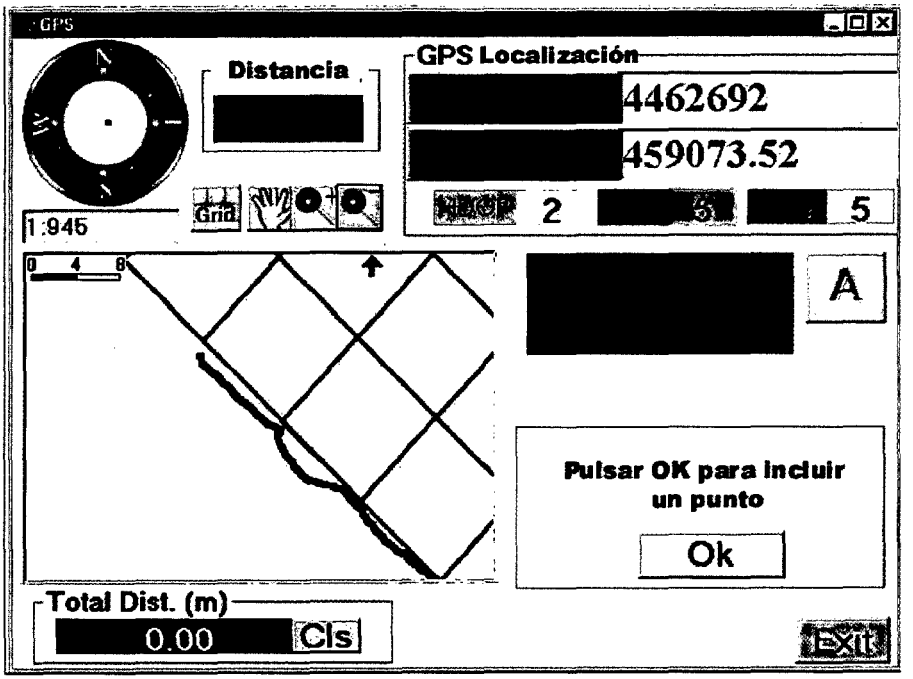


Figura 5.12: Pantalla del navegador en funcionamiento en modo Localizar

presenta unos círculos azules, como puede verse en la figura 5.14 y que corresponden a los puntos ya muestreados.

Finalmente, para los introducir datos que no es posible o no se desea incluir directamente en el campo, como podrían ser las propiedades químicas analizadas de las muestras de suelo, se pueden emplear aplicaciones como Microsoft Access© o la herramienta SIG de Intergraph Geomedia© (figura 5.15).

Para terminar y a modo de resumen decir que se ha desarrollo una herramienta de ayuda a la recolección de datos georeferenciados en campo que presenta el siguiente conjunto de características [RIBEIRO ET AL., 2001, DÍAZ

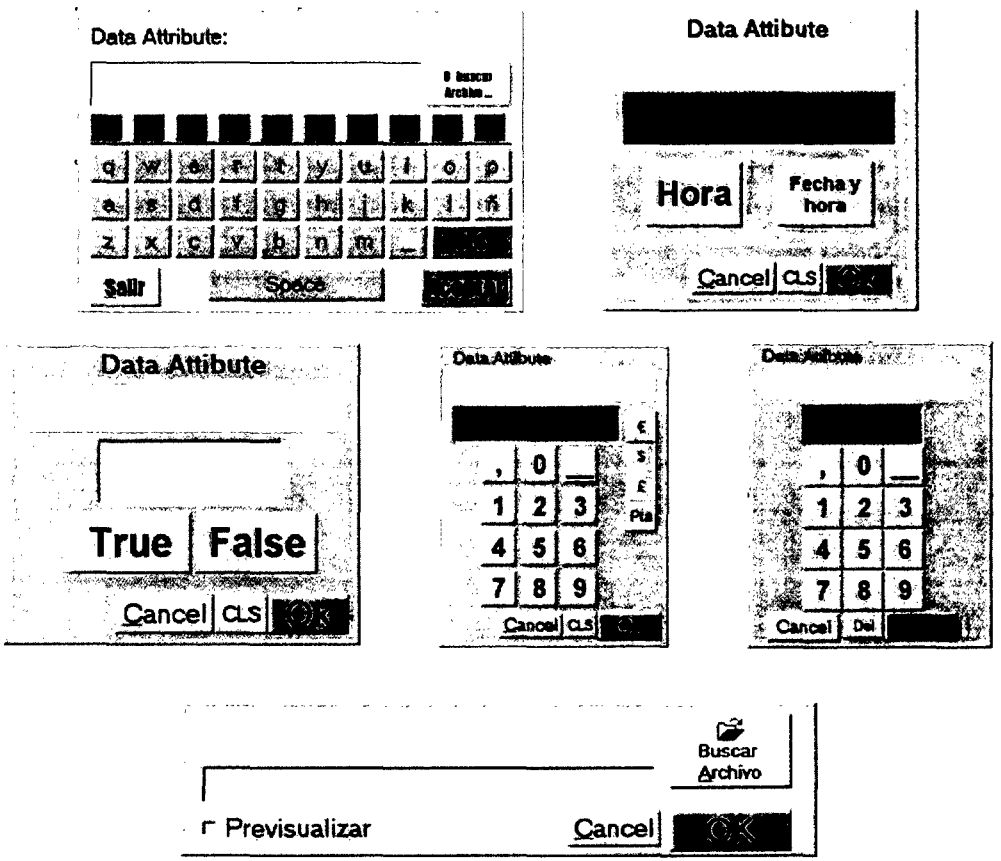


Figura 5.13: Pantallas para la introducción de datos en la base de datos

ET AL., 2002]:

a) Tiene una interfaz con el usuario intuitiva y amigable; b) Se pueden definir con facilidad estrategias de muestreo; c) El sistema no precisa que se realice el volcado de datos a otros dispositivos y los datos se almacenan directamente en el dispositivo final en una estructura SIG, que facilitará la etapa de análisis posterior; d) Aunque inicialmente se han elegido los formatos *Microsoft Access*© y *Geomedia Objects*© como SIG, para la creación de los almacenes de datos, el sistema diseñado es extrapolable a otros formatos; e) La inter-

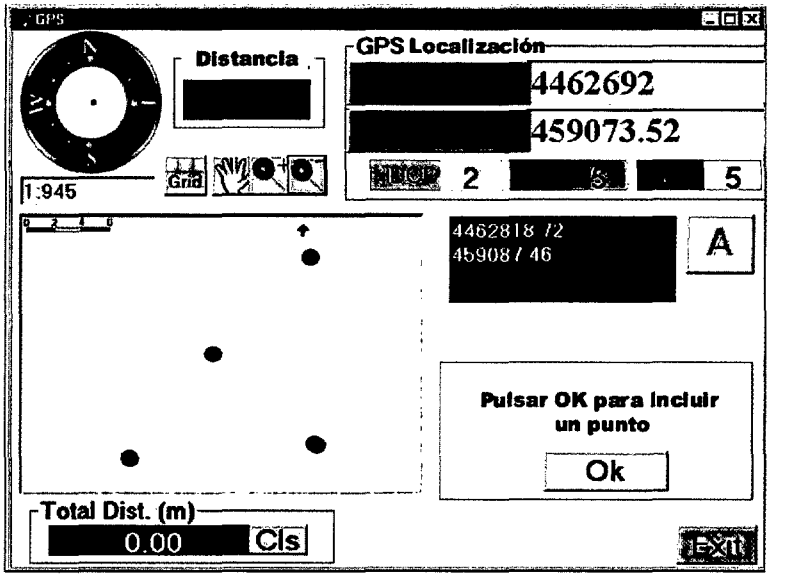


Figura 5.14: Pantalla del navegador en funcionamiento en modo Relocalizar

operabilidad que poseen la mayoría de los SIG que cumplen las disposiciones del consorcio encargado del desarrollo de la normativa y estandarización de la información geográfica (*OpenGIS Consorcium*©),como es el caso de Geomedia Objects©, hace que la herramienta generada sea compatible con otras herramientas informáticas; f) Una importante cualidad es la capacidad de almacenar cualquier tipo de información en la base de datos, entre las que destacan imágenes o sonidos que un operario en el campo considere oportuno incluir; g) El sistema es flexible por lo que puede trabajar en distintos dispositivos (ordenadores portátiles) y, debido a que trabaja con una secuencia estándar del protocolo GPS, puede emplear diferentes receptores GPS; h) La herramienta

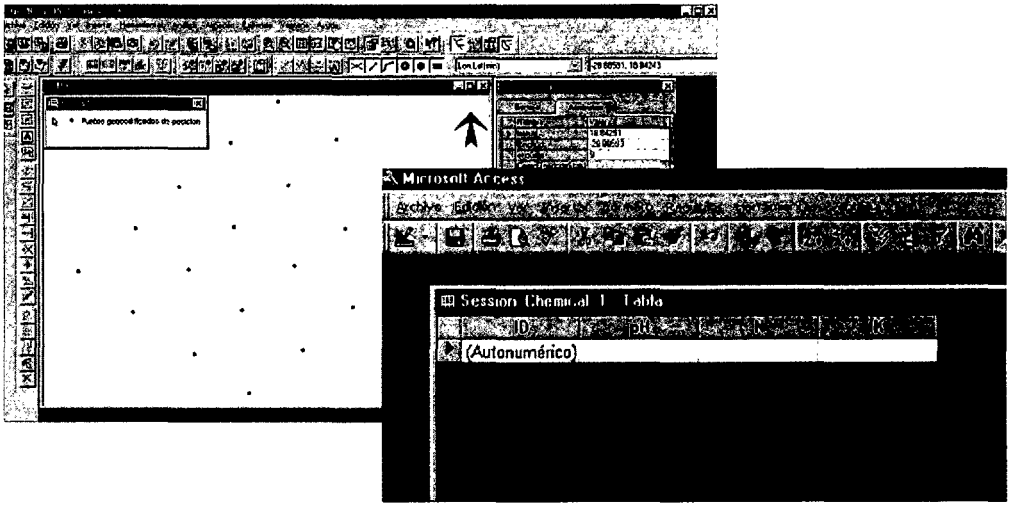


Figura 5.15: Añadiendo datos de laboratorio en el proyecto

se adecúa a las tareas de muestreo; localización y relocalización de puntos en el campo, pero es fácilmente ampliable debido a su diseño modular.

Capítulo 6

PREPROCESAMIENTO DE LOS DATOS

6.1. ¿POR QUÉ ES IMPORTANTE EL PREPROCESAMIENTO?

Como ya se trató en los capítulos iniciales, debido a las diferentes manipulaciones que sufren los datos en una base de datos, éstos suelen presentar alguno de los siguientes problemas:

- *Estar incompletos*: Atributos realmente interesantes pueden no haber sido considerados desde el principio como tales y por lo tanto, puede que no siempre hayan sido almacenados y faltar en ciertos registros.

- *Ser inconsistentes y contener redundancias*: La *integración* de los datos que provienen de diferentes fuentes en un único almacén de datos puede provocar la aparición de variables con la misma información (**redundancia**), lo que podría convertirse en un problema en la etapa de análisis o minería, y por lo que es conveniente eliminar las variables con información repetida. Asimismo, la integración acarrea otros problemas como los relacionados con la identificación cuando la misma información en cada almacén utiliza una forma diferente de representación, un ejemplo típico es el de la variable *sexo* que puede tomar valores como varón, hembra, hombre,mujer, V,H, etc.

- *Presentar valores perdidos*: Entre los datos de un mismo atributo pueden darse perdidas puntuales de información.

- *Ruido*: Los datos pueden estar sometidos al ruido, provocado por errores o valores incorrectos debido a aparatos de medición o de almacenamiento. Algunos algoritmos no trabajan bien con el ruido y puede ser necesario eliminarlo o identificarlo, ya que puede crear confusión en las siguientes etapas de extracción y alterar los resultados. Lo ideal es que el algoritmo o mecanismo de extracción sea capaz de trabajar con datos que contengan ruido, por ejemplo utilizando técnicas probabilísticas o estadísticas. Sin embargo, puede que esos métodos no sean robustos ya que se concentran en evitar el sobreajuste de los patrones e impedir la convergencia.

Para solventar estos problemas, existen diferentes técnicas de preprocesamiento de los datos cuyo objetivo es limpiar los datos, rellenando los agujeros de información y suavizando el ruido, con el fin de mejorar la eficacia de la etapa de análisis. En las siguientes líneas se describen las tareas más características de la etapa de preprocesamiento [FAYYAD ET AL., 1996b]:

- *Selección* - El primer paso, cuando se abordan problemas de extracción de conocimiento o construcción de modelos, es identificar y seleccionar los atributos relevantes, es decir los que deben ser tenidos en cuenta en la etapa de análisis posterior.

- *Limpieza* - En el caso de existir **ruido** o **valores perdidos** se pueden seguir distintas estrategias. Si hay datos perdidos puede o ignorarse la tupla que tiene el valor perdido, o puede completarse el valor que falta por ejemplo, con una constante, con el valor medio de los valores del atributo o con el valor más probable de ese atributo. En el caso del ruido, la tarea es más compleja ya que el atributo tiene valor pero éste no es el adecuado y además la distribución del ruido en el almacén es al azar. Existen diferentes técnicas para determinar el ruido en los datos y posteriormente eliminarlo, la más elemental, es la *inspección directa de los datos*, que puede desembocar en una tarea tediosa cuando el almacén de datos es grande. Otra técnica muy usada es la *regresión*, en la que se analizan, mediante funciones lineales simples o múltiples, aquellos valores

que se alejan excesivamente de la función que representa al resto. Este método sólo es válido para sistemas simples en los que los datos pueden representarse mediante una función lineal. Otra forma de distinguir datos con ruido consiste en agrupar los datos (*Clustering*) y considerar como ruido todos aquellos valores que no pertenecen a ninguno de los grupos. Una vez detectado el ruido se puede eliminar utilizando métodos como es el *Binning*, que se traduce como *cajonera*, que provoca un suavizado de los valores anómalos al ser sustituidos por valores como la media, mediana o de acuerdo a los bordes del grupo de datos vecinos.

Esta tarea de limpieza también permite la detección de información redundante y su eliminación. Por ejemplo, para la identificación de redundancias en datos con valor numérico continuo normalmente se puede realizar un análisis de correlación excluyendo los datos que presenten coeficientes cercanos a 1 o a -1. Sin embargo, en sistemas naturales es frecuente la correlación entre variables, y la decisión de eliminarlas es difícil, ya que en ocasiones son estas variables precisamente las que contienen la información que se busca.

- *Reducción y proyección de los datos* - Además de la selección, puede ser útil una etapa de *reducción* del número de dimensiones efectivas del vector de características que representa a los datos, para encontrar durante el proceso de extracción las características más interesantes. La reducción de la dimensión de este vector, en el caso de una base de datos del número de campos de un registro, es una técnica que permite que algunos algoritmos de extracción realicen una búsqueda más eficiente. Esta tarea puede llevarse a cabo de diferentes formas, como agregar determinadas dimensiones en una única variable o también ir seleccionando atributos utilizando las técnicas *hacia delante* (*forward*) que añade el atributo más importante en cada paso; técnicas *hacia atrás* (*backward*) que elimina el atributo menos relevante en cada paso o las técnicas *mixtas* (*hacia delante + hacia detrás*). Finalmente, los *árboles decisión* también se suelen utilizar para reducir y elegir las variables más importantes y se desechar

las irrelevantes.

La reducción también puede realizarse sobre el volumen de datos, y en este caso se utilizan métodos como, *cambios de representación* mediante cadenas de menor longitud; la *generalización* o métodos de envoltura mediante la pre-clasificación de los ejemplos en un número determinado y menor de categorías; la *compresión* mediante la codificación, sin o con ligeras perdidas de información; y finalmente, el empleo de *filtros* mediante la eliminación de todas las instancias con información perdida. Los filtros son las soluciones más precisas pero frecuentemente costosas computacionalmente, sobre todo filtros complejos como por ejemplo la transformada de Fourier que utiliza en el sistema WEKA[1][WITTEN & FRANK, 1999].

La *transformación* de los datos es otro método que mejora la respuesta de algunos algoritmos de extracción de modelos. Un tipo frecuente de transformación es la *normalización*. Existen diferentes formas de normalizar conjuntos de datos, muchas de ellas descritas en [HAN & KAMBER, 2001] como son, el *reescalado* que tiene en cuenta la distribución original de los datos o el método llamado *Z-core*, basado en la media y la desviación estándar cuando se desconocen los valores mínimo y máximo del conjunto de datos. Otro tipo de normalización la suministra el procedimiento más simple, denominado *escalado decimal* (*decimal scaling*), que, tan sólo desplazando la coma hasta conseguir un dominio decimal y que todos los valores estén comprendidos entre cero y uno, manteniendo al mismo tiempo el carácter original de los datos.

En muchos casos, la transformación implica las reducción de los datos, por esa razón las técnicas para las dos tareas suelen y pueden ser las mismas. Por ejemplo, los datos también se pueden transformar a partir de la combinación de información, a partir de grupos de atributos la construcción de otros nuevos o también, la creación de agregados, mediante parámetros estadísticos como la media, la suma, etc., calculados a partir

[1] http://davis.wpi.edu/ xmdv/weka/

del conjunto de datos. Desde este mismo enfoque, también son de interés las técnicas de generalización que mediante el empleo de jerarquías transforman los datos obteniendo mayor o menor resolución en el espacio de muestral[2]. La *discretización* y la **categorización** son transformaciones muy frecuentes, para hacer que datos inicialmente numéricos puedan utilizase como nominales o simbólicos. Se trata de una técnica que puede ser un buen medio de tratar con la frecuente incertidumbre asociada a los datos numéricos, y que en muchos casos obstaculiza la obtención de modelos precisos. Además, la discretización del universo de discurso de los datos de entrada y la posterior asignación de etiquetas lingüísticas con contenido semántico permite la obtención de modelos directamente transferibles al usuario final. Esta transformación puede realizarse en esta etapa previa de preprocesado de los datos, pero también durante la misma etapa de extracción en la que el algoritmo de búsqueda es el encargado de generar las particiones del conjunto de datos según diferentes criterios heurísticos incluidos en el proceso. El método QUEST [LOH & SHIH, 1997], por ejemplo, es un algoritmo que realiza un análisis discriminante cuadrático que estima la función de densidad de los datos y que permite seleccionar un umbral para dividir el conjunto. Otros algoritmos, como SLIQ [MEHTA ET AL., 1996] menos eficaces ordenan los valores y van probando iterativamente que valor mejor funciona como umbral para la división. Posteriormente, la elección del mejor umbral se basa en medidas que cuantifican la impureza de los grupos, como son el criterio de *Gini* y el cálculo de la entropía o el estadístico χ^2. No obstante, lo más frecuente es realizar la discretización durante la etapa de preprocesamiento, y de los métodos que se pueden emplear durante esta fase, el más deseable es utilizar umbrales determinados por *usuario experto* [VALLS, 1999]. Sin embargo, no siempre esta información existe o no se

[2] El espacio muestral es un concepto estadístico que se puede considerar como el conjunto de los diferentes resultados posibles, y donde cada resultado puede representarse como un punto o elemento de dicho espacio [DEGROOT, 1988]. En el dominio de la minería de datos puede entenderse como *espacio de búsqueda*, definido en secciones anteriores

considera oportuno usarla, por lo que se utilizan métodos alternativos [CHMIELEWSKI & GRZYMALA-BUSSE, 1996]. En [DOUGHERTY ET AL., 1995] se presenta una clasificación, atendiendo a diferentes propiedades, según la cual los métodos para la discretización de los datos continuos pueden ser: (1) *Locales o globales* (respectivamente, las particiones se realizan sobre regiones localizadas definidas por el conjunto de instancias o por el contrario, la discretización se realiza teniendo en cuenta el espacio completo); (2) *Supervisados o no supervisados* (la discretización utiliza o no categorías predefinidas); (3) *Estáticos o dinámicos* (los métodos pueden utilizar un número diferente y preestablecido de intervalos para cada atributo o en contraposición, pueden utilizar un número determinado de intervalos para todas las características procesadas simultáneamente).

La técnica de discretizado más simple y también la más popular, es la definición de intervalos regulares que se obtienen con métodos como el mencionado *Binning*, que permite generar grupos de instancias que tendrán una única etiqueta, dividiendo en este caso el dominio en un número determinado de rangos iguales teniendo en cuenta el valor máximo y mínimo del conjunto. A pesar de ser una técnica sensible a la presencia de ruido, permite que la etiquetado de los intervalos admita semántica de carácter relativo, ya que únicamente se tiene el cuenta la información proporciona por el valor de los datos, y no su frecuencia estadística. No obstante, las técnicas basadas en la frecuencia y en *histogramas*, y que por lo tanto tienen en cuenta la distribución de las valores eliminan el posible ruido durante este proceso, pero requieren más esfuerzo computacional y las etiquetas no tienen un significado relativo (semántico) tan claro. Otra técnica cercana a estos métodos es la *segmentación natural*, que es método por el cual los umbrales resultantes que pudieran ser poco intuitivos o útiles se transforman (por expertos en el dominio) a valores más comprensibles. Así, por ejemplo, en un conjunto de datos que contiene la información sobre la edad de las personas, mientras que con otros métodos se podrían obtener umbrales como *46,25* o *53,02*, mediante la

segmentación se obtendría umbrales más intuitivos como *40* y *50*. Finalmente, en esta tarea de discretización también son frecuentes técnicas no supervisadas de agrupamiento (*clustering*) o supervisadas utilizando heurísticos como los ya mencionados de entropía, *Gini*, χ^2 o *MDL* (*Minimum Description Length*) entre otros [DOUGHERTY ET AL., 1995, GRZYMALA-BUSSE & STEFANOWSKI, 2001]. Incluso, existen métodos más complejos para de discretización utilizando heurísticos sofisticados como *Zeta* [HO & SCOTT, 1997] o *Khiops* [BOULLE, 2004]. Este último, en concreto, para encontrar los umbrales óptimos de discretización propone un proceso que busca la maximización de criterios estadísticos como χ^2 o MDL.

Para concluir, es importante remarcar que en procedimientos de extracción o búsqueda de conocimiento requieren que los intervalos producidos en esta etapa de discretización tengan un significado útil para los expertos, por lo que, es necesario que el método elegido de discretización procure intervalos y etiquetas (categorías) que sobre todo sean interpretables o comprensibles. Este compromiso hace que se utilicen métodos más simples y que creen intervalos sencillos de interpretar, frente a los más complicados y refinados que pueden provocar un efecto no deseado, como por ejemplo, que los modelos resultantes no sean comprensibles y por lo tanto poco útiles.

6.2. HERRAMIENTA DE PREPROCESO: PREPARADAT

Cualquiera de las técnicas involucradas en las etapas anteriores requieren la manipulación de los datos, normalmente contenidos en una base de datos. Los datos que se utilizan en la presente memoria están contenidos en las tablas de una base de datos relacional en formato Microsoft Access© (*.mdb).

Teniendo en cuenta alguna de las técnicas anteriores y para facilitar la preparación de los datos se ha desarrollado una herramienta auxiliar `PreparaDAT` basada en lenguaje SQL (explicado en el apartado 3.2.2) que facilita las tareas de la etapa de preprocesamiento con los datos recogidos en en una campaña de muestreo. A continuación, se explican los útiles que ofrece esta herramienta para el pre-procesamiento.

- Un primer paso en el preprocesamiento de los datos en una base de datos relacional puede ser la **integración** de datos que están contenidos inicialmente en diferentes tablas. En esta operación se requiere un campo o atributo que relacione los datos, mantener la integridad de los mismos y almacenar correctamente todas las características de cada objeto (muestra en nuestro caso) en un sólo registro.

- Otro paso común, y muy adecuado en nuestro caso, es la **normalización** de los valores de las variables de entrada cuando el rango de éstas es numérico. Los datos originalmente se distribuyen en un dominio de valores entre un mínimo y un máximo que se convierte en un dominio continuo entre cero y uno, mediante un reescalado lineal cuya expresión se presenta en la ecuación 9 [PYLE, 1999]. Recordemos que esta técnica no distorsiona la distribución de la variable.

$$V(_{Normal})_i = \frac{V_i - Min(V_1 \dots V_m)}{Max(V_1 \dots V_m) - Min(V_1 \dots V_m)} \tag{9}$$

  En esta ecuación $V(_{Normal})_i$ representa el valor resultante normalizado y V_i es el valor i de la variable V antes de la normalización. Un valor o instancia $i \in m$, siendo m el número total de valores o instancias de

la variable que se está normalizando y que están presentes en la tabla resultante del proceso de integración anterior.

El formulario mostrado en la figura 6.1 permite al usuario realizar la normalización de los atributos numéricos seleccionados, de acuerdo con esta expresión. Esta herramienta permite incluir un valor de referencia para tomarlo como máximo, y eliminar el posible ruido en ese rango del dominio provocado por valores muy altos pero con poca frecuencia. Este nuevo valor sobre el que se basará la normalización se introduce en la casilla "*Valor del Máximo de Referencia*". Si se deja a "0" `PreparaDat` utiliza como valor de referencia el encontrado en la tabla original.

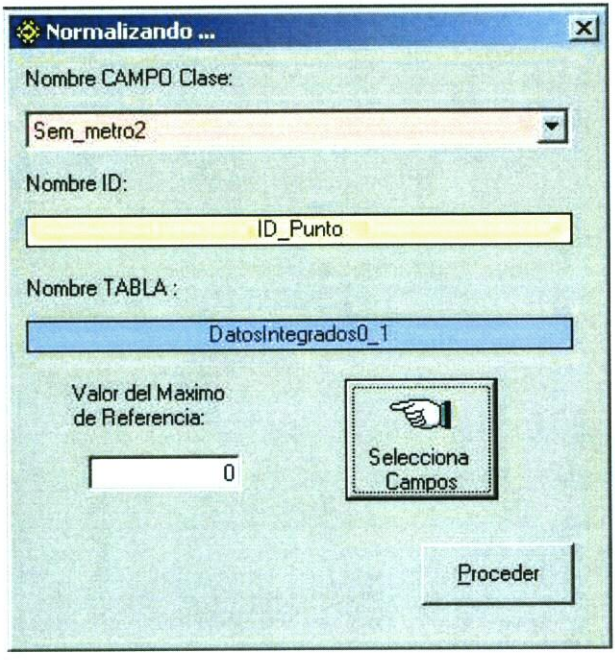


Figura 6.1: Pantalla que permite la normalización y reescalado de distintos campos de una misma tabla

- La **discretización** con `PreparaDat` se puede hacer de dos posibles for-

mas: a) calculando y utilizando umbrales automáticos para obtener *intervalos regulares* y b) usando *umbrales arbitrarios* para conseguir intervalos de diferente tamaño.

El procedimiento interno de la aplicación para obtener k intervalos regulares de forma automática, comienza con el cálculo del rango r, entendido éste como la longitud constante de los intervalos, mediante la formula:

$$r = \frac{Max - Min}{k} \tag{10}$$

Posteriormente, cada intervalo i_c se obtiene a partir de los umbrales inferior (U_i) y superior (U_s) utilizando el rango r calculado, según:

$$i_c = (U_{i(c)}, U_{s(c)}] \;=\; \left(\sum_{j=0}^{c-1} r_j, \; \sum_{j=0}^{c} r_j \right] \tag{11}$$

donde $dom(c) = [0, 1, ..., k-1, k]$.

Este procedimiento se realiza sucintamente a través de la pantalla que muestra la figura 6.2a, donde el usuario sólo necesita incluir el número de intervalos en los que desea discretizar la variable. Mediante este método, las etiquetas de las variables, que son números de 1 a K, se crean automáticamente, tal y como se puede observar en la figura. El valor de las etiquetas utilizadas representan a los diferentes intervalos tal que los valores menores son los intervalos de menor valor, y al contrario, los números mayores son los intervalos más cercanos al valor máximo del dominio. Así, por ejemplo para un dominio [1, 7], cuyo rango para tres intervalos ($K = 3$) es $r = 2$, mediante este procedimiento se crearían los siguientes tres intervalos con las correspondientes etiquetas, el primer intervalo [1,3] tendría la etiqueta 1, el segundo (3,5] la etiqueta 2 y por último el (5,7] sería el 3.

El segundo método mencionado, el más abierto, permite el empleo de valores arbitrarios para los umbrales de los intervalos. En este caso, los

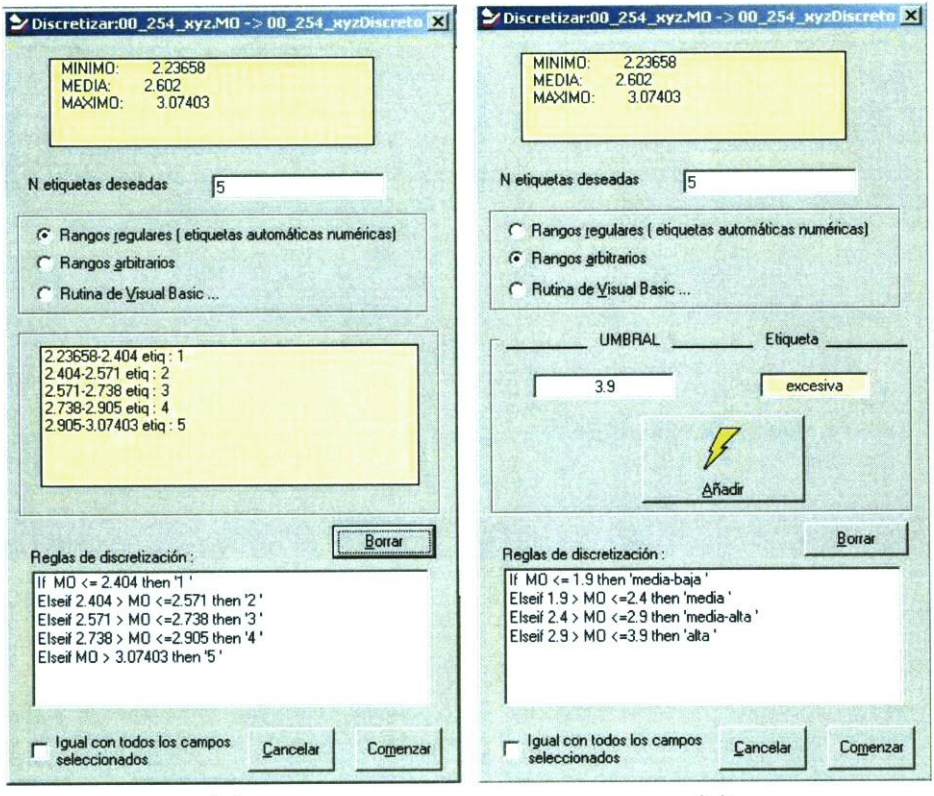


Figura 6.2: Proceso de etiquetado basada en (a) umbrales regulares y (b) arbitrarios

valores pueden ser los determinados, o bien por un experto en el dominio de aplicación o bien, con cualquier otro método diferente al de intervalos regulares. La pantalla que utiliza el usuario en el proceso de discretización y categorización de umbrales arbitrarios es mostrada en la figura 6.2b. Este método requiere por parte del usuario el número de intervalos, la etiqueta de cada intervalo y el valor de cada umbral, sólo el superior ya que el inferior es calculado a partir del intervalo anterior o del valor mínimo. Este mismo formulario permite utilizar procedimientos específicos para una discretización más sofisticada. Para el caso concreto de estudio, se ha

incorporado un procedimiento para etiquetar una de las variables edáficas, el tipo de suelo, que proporciona el nombre de la categoría a la que pertenecen las muestras del suelo en función de las proporciones (tomadas de *www.insuelos.org.ar/servicios/estudiosuelos/lasorpresaAnexo.htm*) de cada uno de los componentes texturales: arena, limo y arcilla. El procedimiento de discretización incorporado, figura 6.3, se basa en el triángulo de texturas desarrollado por el *US Department of Agriculture* (*USDA*) y utilizado con frecuencia por los ingenieros agrónomos [MATEOS, 1967].

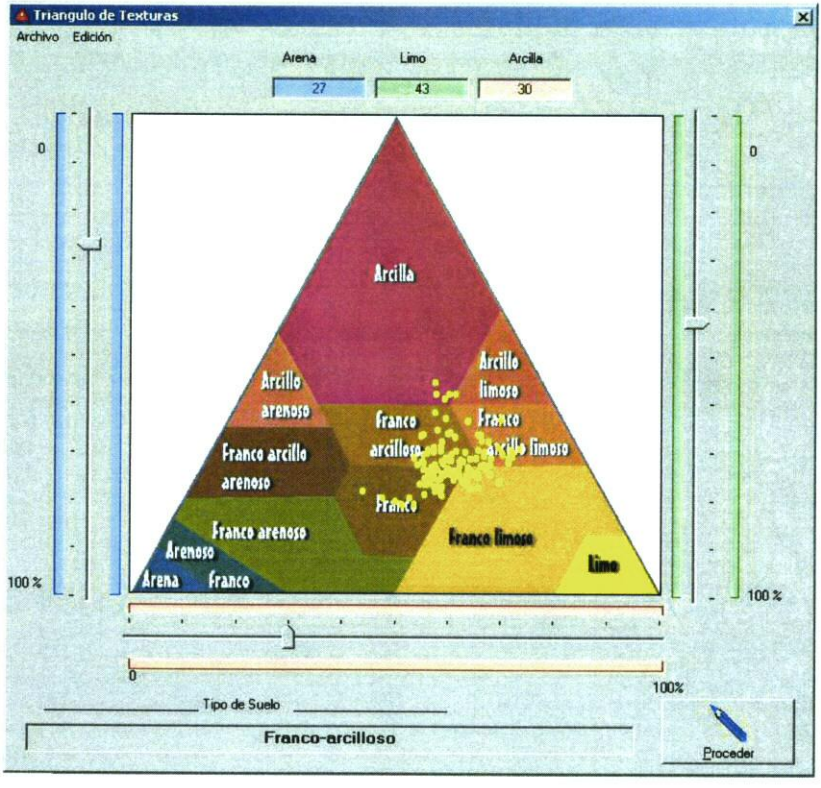
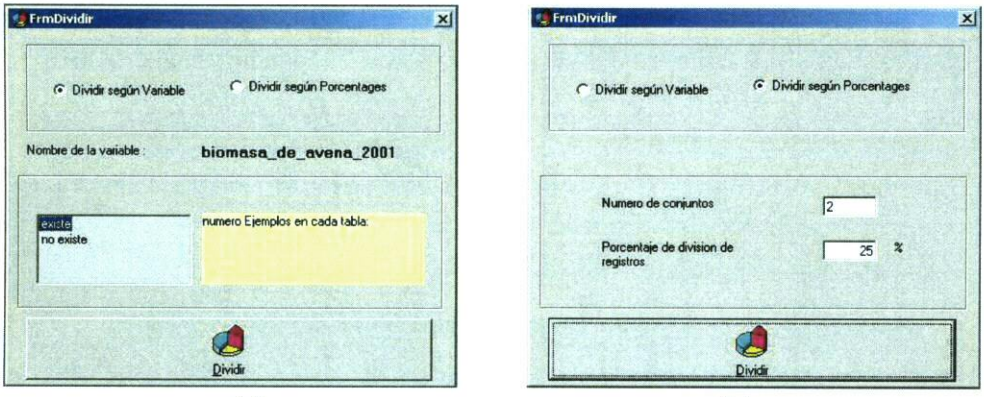


Figura 6.3: Subaplicación para una categorización automática de la variable tipo de suelo basada en porcentajes de granulometría

- En algunas aplicaciones es necesario **dividir** la tablas, por ejemplo, en

función del valor de un atributo, como cuando se desea definir clases.



(a) (b)

Figura 6.4: Separación de (a) ejemplos positivos y ejemplos positivos en diferentes tablas y (b) conjuntos de entrenamiento y validación

La aplicación desarrollada `PreparaDAT` permite realizar dos tipos de divisiones: a) según los valores de un atributo categórico (figura 6.4a) y las tablas resultantes contienen una los registros pertenecientes a la clase que se desea modelar y la otra los contraejemplos de la clase; y b) según un porcentaje que separe el conjunto de datos en un conjunto de entrenamiento y el conjunto de los registros preseleccionados aleatoriamente para la operación de validación de los modelos descubiertos (figura 6.4b).

- A partir de las tablas que contienen los ejemplos para la clase que se desea modelar y de la tabla que contiene los contraejemplos (esta última mejora los límites del modelo haciéndolo más preciso, pero no es estrictamente necesaria), se puede hallar un modelo para la clase aunque es conveniente realizar una limpieza de los datos de entrada.

  La tarea de limpieza es clave a la hora de obtener un buen modelo ya que después de la categorización y de la partición es posible que la intersección entre el conjunto de ejemplos de clase y el conjunto de contrajemplos no sea vacía. La figura 6.5 muestra la herramienta de ayuda para la etapa de

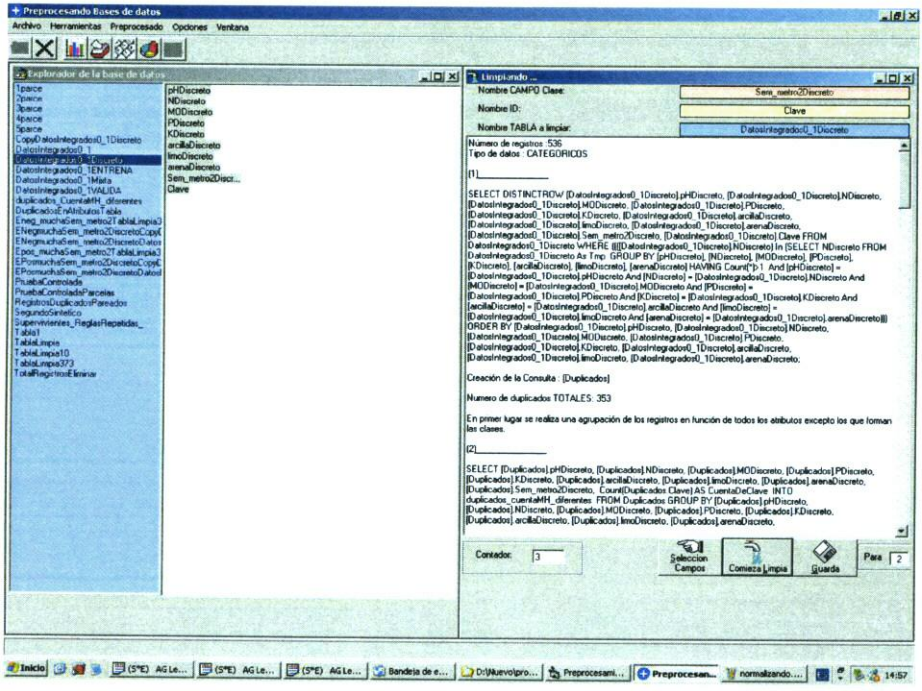


Figura 6.5: Formulario de limpieza de `PreparaDAT` para conseguir que las dos clases predefinidas estén compuestas por conjuntos disjuntos

limpieza de datos que suministra la aplicación `PreparaDAT`. El proceso de limpieza se basa en una secuencia de consultas SQL de selección, creación y borrado de registros (véase el apéndice D) realizada sobre la tabla que se quiere limpiar, con el fin de eliminar todos aquellos registros que contienen la misma información, es decir, que todos los atributos presentan los mismo valores, pero pertenecen a las dos clases a la vez, provocando una inconsistencia.

Capítulo 7

EXTRACCIÓN AUTOMÁTICA DE REGLAS

7.1. BASES TEÓRICAS DE LA PROPUESTA

Esta sección describe los fundamentos del sistema diseñado y desarrollado para la extracción de reglas a partir de un conjunto de datos de entrada, contenidos en tablas relacionales, y donde los ejemplos se almacenan como registros. Concretamente, los campos de estos registros guardan las características de estos ejemplos. A estos campos les llamaremos también atributos. Los elementos fundamentales del sistema desarrollado son, en primer lugar, un procedimiento de búsqueda del mejor modelo en formato de reglas basado en un algoritmo genético y, en segundo lugar, una función de evaluación de la exactitud de los modelos. En este capítulo se describe en detalle estos dos elementos.

7.1.1. EL ESPACIO DE BÚSQUEDA. LA CODIFICACIÓN DE UNA HIPÓTESIS

En la búsqueda genética que se propone en esta tesis las soluciones candidatas o hipótesis codificadas en los cromosomas son un conjunto R de reglas donde $cardinal(R) \geq 1$. Las reglas que se codifican son `Si-Entonces` y por tanto de la forma:

$$\texttt{SI } Cond_1 \texttt{ AND } Cond_2 \texttt{ AND } \ldots \texttt{ AND } Cond_N \texttt{ ENTONCES } C$$

donde la conjunción formada por las precondiciones $Cond_1$, $Cond_2$,..., $Cond_N$ sobre N atributos es el antecedente de la regla y C es el consecuente que representa la clase preestablecida en la que se distribuyen los ejemplos de entrada.

Atendiendo a la codificación genética tal y como se introdujo en la sección 4.3, la construcción de los cromosomas se basan en una unidad mínima capaz de codificar en bits todos y cada uno de los atributos y sus correspondientes valores definidos en la tabla de entrada. A esta cadena básica la hemos llamado *nucleosoma* (η) por la similitud que presenta con este concepto de la biología genética[1]. Este elemento básico, como se desprende de todo lo que se expone más adelante, determina la complejidad de los modelos, ya que el número de nucleosomas establece el número de veces que se repite un atributo en una hipótesis.

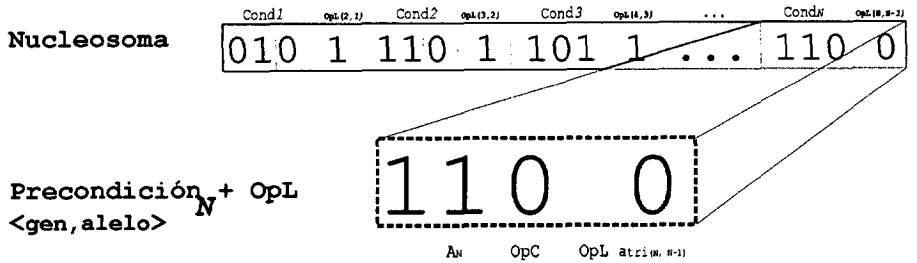


Figura 7.1: Un nucleosoma es la unidad mínima para la construcción de un cromosoma

La figura 7.1 muestra gráficamente un esquema de esta cadena básica y mínima (nucleosoma) para codificar en un cromosoma una regla formada por las condiciones sobre cada uno de los N atributos. Recordemos que una precodición $Cond_N$ de un atributo N se representa con el par $< gen, alelo >$.

En nuestra propuesta, un cromosoma puede estar constituido por más de un nucleosoma lo que permite la construcción de modelos más complejos, es decir modelos compuestos por más de una regla. Teóricamente, el número de nucleosomas η, que forman un cromosoma, es un número entero que puede variar entre 1 e ∞ según las necesidades del problema. Consecuentemente, este número η se puede relacionar con la complejidad del modelo, ya que cromosomas con un número pequeño de nucleosomas darán lugar a conjuntos pequeños

[1] Se denominan *nucleosomas* a ciertos empaquetamientos de ADN creados por las histonas dentro del cromosoma.

de reglas mientras que cromosomas con un número alto de nucleosomas darán lugar, con alta probabilidad, a modelos formados por un gran número de reglas.

Dentro de la estructura total que integra un cromosoma y dentro de los nucleosomas se pueden distinguir, además de los pares $< gen, alelo >$, los operadores de comparación (OpC), englobados por ese par, y los operadores lógicos (OpL). En la figura 7.1, se muestra que los operadores lógicos OpL ocupan dos posiciones diferentes dentro del cromosoma; la primera entre las condiciones dentro de cada nucleosoma, en este caso se denotan por OpL_{atri}, y la segunda, entre los distintos nucleosomas, y la notación en este caso es OpL_{η}.

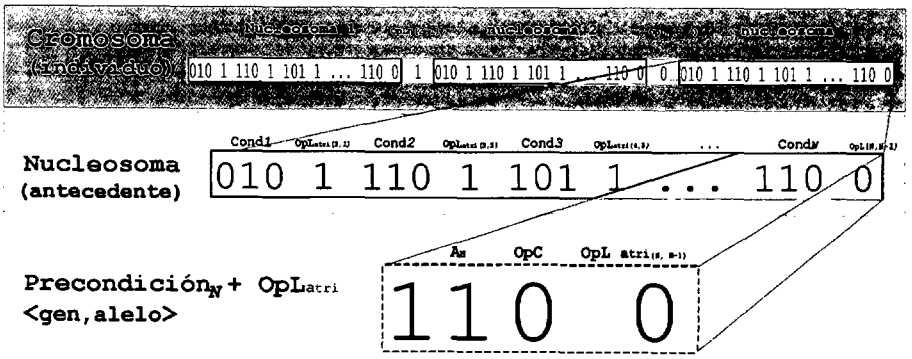


Figura 7.2: Niveles de codificación de un cromosoma para representar cada una de las partes que compone un sistema de r reglas

Una vez que se construye un cromosoma, con un determinado número de nucleosomas se puede establecer el número de reglas que configurarán el conjunto final. Los conjuntos de reglas se forman a partir de una disyunción, es decir, siempre que exista un operador lógico $OpL =$ `OR` conectando dichas reglas. Por otro lado, cada regla viene representada en el cromosoma únicamente por su parte antecedente, es decir por una conjunción de precondiciones que se unen mediante el operador lógico $OpL =$ `AND`. Lo definido hasta aquí corresponde al esquema convencional de un conjunto de reglas codificado en un cromosoma. Sin embargo, la codificación de cromosoma que nosotros proponemos permite que los operadores Lógicos (OpL_{η} y OpL_{atri}) puedan tomar indistintamente

bits Gen-Alelo	`aspecto-cielo`	`viento`
01	soleado	suave
10	nublado	fuerte
11	lluvioso	muy fuerte
00	no se representa	no se representa

Tabla 7.1: Código para representar en un algoritmo genético el problema del *partido de tenis*

los valores `OR` o `AND`, si previamente se configura esta opción en el experimento. En consecuencia, el número de reglas r es mayor o igual que el número de nucleosomas η, cuando se configura al algoritmo para que pueda interpretar los bits dentro del cromosoma asignados a los conectores lógicos (OpL). En contraposición, se puede configurar la búsqueda de modo que, independientemente del valor que presente el bit asignado a estos operadores lógicos, éstos siempre se interpreten como `AND` para la conexión entre precondiciones (OpL_{atri}) y como `OR` para los operadores entre nucleosomas (OpL_{η}). En este último caso, el número de reglas que formará el modelo es fijo y queda definido inicialmente. Todas estas consideraciones irán siendo más claras a medida que se avance en la lectura de la presente memoria.

Para ilustrar lo expuesto hasta ahora en cuanto a la codificación de un cromosoma y sus nucleosomas, explicamos a continuación un problema ejemplo, que utiliza variables climáticas para determinar si se puede jugar o no un partido de tenis, en el cual la variable objetivo es `Jugar-tenis` y puede adoptar los valores `Sí` o `No`. El objetivo en este ejemplo es encontrar las reglas que definen la clase `Jugar-Tenis=Sí`. En este caso, las variables predictivas (genes) que componen los antecedentes de las hipótesis son, por un lado, `aspecto-cielo` que presenta tres etiquetas (alelo) {`soleado, nublado, lluvioso`} y por otro lado, `viento` que puede ser {`muy fuerte, fuerte, suave`}. La codificación de cada precondición, es decir el par gen-alelo requiere dos bits cuya combinación permite representar todas las posibilidades, tal y como se muestra en la tabla 7.1.

Además, cada precondición necesita un bit para codificar el operador de comparación OpC como un igual ($=$) o como un desigual ($\neq$), por lo que

en total, el número de bits para cada precondición es tres en el ejemplo. Las precondiciones unen entre si con el operador lógico (OpL_{atri}) que puede ser `AND` o `OR` según el valor de el siguiente bit, 0 ó 1. El consecuente de las hipótesis es siempre `Jugar-Tenis=Sí`, razón por la que se utiliza para su codificación un bit con valor 1. Teniendo en cuenta este código, un cromosoma compuesto de un único nucleosoma ($\eta = 1$) sería por ejemplo la siguiente cadena:

$$\overbrace{\underbrace{(10)1}_{condicion}\ 1\ (00)1\ 1\ 1}^{nucleosoma} \tag{12}$$

donde los bits que representan el par gen-alelo están escritos entre paréntesis para diferenciarlos del otro grupo de bits que codifican a los conectores lógicos. Los tres primeros bits `(10)1` representan a la primera precondición, y según la tabla es `aspecto-cielo=nublado`. El cuarto bit de la cadena, representa al operador lógico `AND`, que unirá esta primera precondición con la siguiente que exista. En este caso, la segunda condición es la codificada en el siguiente grupo de bits `(00)1`, y que es `viento = nulo`, es decir que no se activa y no aparece en la regla. Finalmente, a partir de el resto de la cadena (1 1) se decodifica el consecuente, `Jugar-Tenis=Sí`, por lo que la regla candidata sería :

```
SI (aspecto-cielo=nublado) ENTONCES (Jugar-Tenis=Sí)
```

La longitud de los cromosoma con un nucleosoma, que debe ser el número de bits necesarios para codificar una vez cada variable y los necesarios operadores, en este ejemplo es 9 bits.

Siguiendo este mismo esquema de decodificación, los cromosomas construidos con más de un nucleosoma codifican varias reglas, pero además incluyen bits adicionales para su mutua conexión y formar el conjunto final de reglas. Este bit añadido representa un operador lógico OpL_{η}, que igualmente puede valer `AND` o `OR`. Por ejemplo, en la decodificación de un cromosoma con dos nucleosomas ($\eta = 2$) como el presentado en la línea 13. En esta cadena se puede

observar que el primer nucleosoma corresponde al analizado previamente.

$$\overbrace{(10)1\ 1\ (00)1\ 1\ 1}^{nucleosoma1}\underbrace{\mathbf{0}}_{OpL_\eta}(11)0\ 0\ (01)1\ 1\ 1 \tag{13}$$

Este cromosoma se transforma, siguiendo el procedimiento explicado para el caso anterior, en el siguiente conjunto de dos reglas.

```
r1: SI (aspecto-cielo=nublado) ENTONCES (Jugar-Tenis=Sí)
r2: SI (aspecto-cielo≠lluvioso) OR (viento=suave) ENTONCES
(Jugar-Tenis=Sí)
```

Para concluir, el conjunto final queda finalmente en las tres siguientes reglas debido a la propiedad disyuntiva del OR:

```
r1: SI (aspecto-cielo=nublado) ENTONCES (Jugar-Tenis=Sí)
r2: SI (aspecto-cielo≠lluvioso) ENTONCES (Jugar-Tenis=Sí)
r3: SI (viento=suave) ENTONCES (Jugar-Tenis=Sí)
```

Una vez decodificado el cromosoma por un interprete que funciona según el procedimiento explicado, el algoritmo lo somete a su evaluación, que en esta implementación se realiza utilizando el lenguaje de consulta SQL que se explica a continuación.

7.1.2. La función de calidad. La evaluación de las hipótesis

Como se recordará, partimos de que los datos que se desea modelar se encuentran almacenados en una base de datos relacional. Como ya se vio en los capítulos iniciales (sección 3.2.1) la consulta en este tipo de almacenes de datos se realiza a partir de instrucciones en lenguaje SQL. Por otra parte, en la evaluación de una hipótesis lo que se quiere es asignar una calidad a la hipótesis en función de lo bien que ésta describe los datos de entrada. En este sentido el cromosoma se transforma en una proposición compuesta de disyunciones y conjunciones que pueden utilizarse directamente en la cláusula WHERE. La

consulta decodificada permite recuperar los registros que cumplen la proposición, y que puede ser considerada como el conjunto de antecedentes de un conjunto de reglas. En definitiva, el resultado de la consulta con la cláusula WHERE ϑ_p es el mismo que el obtenido con la operación de equiparación que realizan el conjunto de reglas R_p formado por las reglas $r_{(j)p}$ donde j es el número de reglas del conjunto, debido a que ambos modos de operación se basan en álgebra Boole. Este hecho se puede ilustrar con siguiente ejemplo.

Supongamos que tenemos el conjunto de seis instancias I que se muestra en la tabla 7.II y un modelo M_p para los registros que pertenecen a la clase p.

id	$atributo_1$	$atributo_2$	*Clase*
1	a	c	p
2	a	c	n
3	a	c	p
4	b	c	p
5	a	d	n
6	b	d	n

Tabla 7.II: Ejemplo para comprobar la equiparación de las cláusulas WHERE de una sentencia SQL con un conjunto de reglas

El modelo define la clase p a partir de las precondiciones formadas por el $atributo1 = a$ o $atributo_1 = b$ y el $atributo2 = c$. Si hablamos en términos de consultas SQL a una base de datos, el cálculo de la cobertura para este modelo se obtendrá mediante la consulta:

ϑ_p =SELECT * FROM I
WHERE ($atributo_1$=a *AND* $atributo_2$=c) OR ($atributo_1$=b *AND* $atributo_2$=c);

que coincide con el mismo grupo de registros que cubrirán las precondiciones del conjunto de reglas de la forma[2]:

$r1_p$: SI $atributo_1$= a y $atributo_2$ =c ENTONCES clase= p

[2] Este conjunto R_p se puede expresar también como: "SI ($atributo_1$= a o b) y $atributo_2$ =c ENTONCES clase= p"

$r2_p$: SI $atributo_1$= b y $atributo_2$ =c ENTONCES clase= p

En otras palabras $cobertura_\vartheta(E)$=($cobertura_R(E)$) es igual a los registros 1,2,3,4

Como consecuencia de lo dicho hasta el momento los cromosomas de la búsqueda genética representan proposiciones lógicas que pueden decodificarse manteniendo el significado bien como antecedentes de un conjuntos de reglas, o bien como las condiciones de la cláusula WHERE de una consulta a una base de datos.

Una vez descrito los principales elementos de consulta necesarios para implementar el cálculo de la función de calidad, y con ello evaluar los cromosomas, pasamos a ver cómo se desarrollarían las funciones descritas en el apartado 3.1.3 en términos de sentencias SQL. Recordemos que todas las funciones propuestas utilizan los parámetros *Verdaderos positivos* (V_p), *Verdaderos negativos*(V_p), *Falsos negativos* (F_n) y *Falsos positivos* (F_p). El cálculo de estos parámetros se presenta a partir de dos aproximaciones distintas que utilizan los comandos UPDATE y SELECT además de la cláusula WHERE.

– Aproximación con SELECT

Con esta aproximación por el cromosoma de cada individuo *ind*(i) se generan cuatro consultas con la siguiente sintaxis:

(1.1) $Vp_{ind(i)}$ = `Card(SELECT * FROM <tabla> WHERE [cromosoma(i)] AND Clase=P)`

(1.2) $Vn_{ind(i)}$ = `Card(SELECT * FROM <tabla> WHERE [NOT cromosoma(i)] AND NOT Clase=P)`

(1.3) $Fp_{ind(i)}$ = `Card(SELECT * FROM <tabla> WHERE [cromosoma(i)] AND NOT Clase=P)`

(1.4) $Fn_{ind(i)}$ = `Card(SELECT * FROM <tabla> WHERE [NOT cromosoma(i)] AND Clase=P)`

donde `cromosoma(i)` representa al conjunto de condiciones codificadas en la cadena binaria de un individuo *ind(i)* y `Card()` simboliza el cardinal de cada

conjunto.

De modo que $\texttt{Vp}_{ind(i)}$ (linea 1.1) se calcula como el cardinal del conjunto resultante de la intersección entre el conjunto de registros recuperados por el comando SELECT con la cláusula WHERE, formado a partir de las condiciones que define el cromosoma `cromosoma(i)`, y el conjunto formado por todos los ejemplos de la clase p. Así mismo, $\texttt{Vn}_{ind(i)}$ (linea 1.2) se calcula como el cardinal del conjunto resultante de la intersección entre el conjunto de registros seleccionados con la cláusula WHERE, en este caso formado sobre la negación de las condiciones definidas por el cromosoma `cromosoma(i)`, y el conjunto formado por los elementos que no pertenecen a la clase p, es decir, elementos de la clase n. $\texttt{Fp}_{ind(i)}$ (linea 1.3) se calcula como el cardinal del conjunto resultante de la intersección entre el conjunto de registros recuperados por la cláusula WHERE con las condiciones que define el cromosoma `cromosoma(i)` y el conjunto formado por todos los ejemplos de la clase n. Y finalmente, $\texttt{Fn}_{ind(i)}$ (linea 1.4) se calcula como el cardinal del conjunto resultante de la intersección entre el conjunto de registros recuperados por la cláusula WHERE sobre la negación de las condiciones definidas por el cromosoma `cromosoma(i)` y el conjunto formado por los elementos de la clase p.

En definitiva lo que tenemos es que:

- Vp es el cardinal el conjunto de instancias de la clase p que cumplen la cláusula WHERE.
- Vn representa el cardinal del conjunto de instancias que no pertenecen a la clase p y que no es seleccionado, o lo que es lo mismo, que no satisfacen la consulta y además no son ejemplos de la clase n.
- Fp es el cardinal del conjunto de registros seleccionados por la consulta que no pertenecen a la clase p.
- Fn es el cardinal del conjunto de registros que no satisfacen la consulta pero si son elementos de la clase p.

La implementación que proponemos para el cálculo de los valores Vp, Fp, Vn y Fn asume que la unión de las clases p y n representa el universo de discurso, que es el contenido de la tabla de entrenamiento de la base de datos. Además se cumple que:

$$\mathrm{Vp} + \mathrm{Vn} + \mathrm{Fp} + \mathrm{Fn} = \mathrm{N}$$

donde N es el número total de registros en la tabla o conjunto de entrenamiento. Ya que Vp, Fp, Vn y Fn se han construido de forma que los respectivos conjuntos recuperados CR de la tabla de entrenamiento, a partir de estas expresiones, no comparten elementos. En otras palabras:

$$CR_{Vp} \cap CR_{Vn} \cap CR_{Fp} \cap CR_{Fn} = \emptyset$$

– Aproximación con UPDATE

La segunda aproximación utiliza UPDATE que permite un procedimiento más simple. La idea en la que se inspira este procedimiento se basa en la propia tabla de clasificación (explicada previamente en la sección 3.1.3 página 32), donde se enfrentan los valores reales y los estimados por un modelo candidato. Para la implementación de esta idea es necesario añadir dos nuevos campos a la tabla sobre la que se realiza el aprendizaje. A estos nuevos campos se llaman `REAL` y `ESTIMADO`. Lógicamente, el campo `REAL` será el encargado de almacenar la información real de pertenencia a la clase, mientras que el campo `ESTIMADO` incluirá la información de cada registro sobre la pertenencia o no a la clase estimada a partir del modelo que se está evaluando. Por lo tanto, los dos campos anexionados almacenan los valores lógicos *verdadero (TRUE)* o *falso (FALSE)*, pudiendo valer uno o cero. Para el campo `REAL`, el valor uno (1) indica que un registro pertenece a la clase objetivo p mientras que el valor cero (0) indica que no pertenece a esa clase. Para el campo `ESTIMADO`, el valor uno (1) se utiliza para representar que según el modelo candidato evaluado un registro pertenecería a la clase objetivo, y cero para lo contrario.

Mientras que la información del campo `REAL`, lógicamente, permanece invariable, y por lo tanto, el campo `REAL` es automáticamente rellenado, con los

valores 1 y 0 correspondientes, por primera y única vez al principio del proceso. Por el contrario, el campo ESTIMADO es modificado cada vez que un nuevo modelo candidato se evalúa utilizando la siguiente consulta auxiliar (ϑ_{Aux}):

$\vartheta_{Aux(ind(i))}$ = UPDATE <tabla> SET [ESTIMADO]=1
WHERE (cromosoma(i));

donde cromosoma(i), igual que en la aproximación explicada previamente, representa al conjunto de condiciones codificadas en la cadena binaria de un individuo *ind(i)*. Trás ejecutar la consulta creada ϑ_{Aux}, el campo estimado presenta el valor 1 en aquellos registros que el modelo estima que pertenecen a la clase y 0 en caso contrario. La evaluación finaliza con el cálculo de los parámetros V_p, V_n, F_p y F_n, ejecutando sobre la base de datos las siguientes consultas:

(2.1) Vp = Card(SELECT * FROM <tabla> WHERE [REAL] = 1 AND [ESTIMADO] = 1)

(2.2) Vn = Card(SELECT * FROM <tabla> WHERE [REAL] = 0 AND [ESTIMADO] = 0)

(2.3) Fp = Card(SELECT * FROM <tabla> WHERE [REAL] = 0 AND [ESTIMADO] = 1)

(2.4) Fn = Card(SELECT * FROM <tabla> WHERE [REAL] = 1 AND [ESTIMADO] = 0)

Ambos métodos, que producen los mismos resultados según la experimentación durante el desarrollo de la implementación, tienen interesantes ventajas. Por ejemplo, el procedimiento SELECT a pesar de presentar una construcción algo más compleja, deja un espacio abierto e interesante para la ampliación del sistema al aprendizaje con más de dos clases, ya que la consulta SELECT incluye la condición de la clase. Sin embargo, UPDATE que es el procedimiento óptimo para un problema de dos clases, como la clase de ejemplos positivos junto a la clase de ejemplos negativos, y por lo tanto, el recomendado en esta primera versión de la implementación.

Funciones de calidad en términos de verdaderos y falsos

Numerosas funciones pueden definirse e implementarse en función de verdaderos y falsos, tal y como se detallo en los capítulos de revisión de conocimiento (sección 3.1.1). La implementación, incorpora algunas (*aciertos*, *aciertos y errores* y *SE*), que han sido utilizadas para la experimentación. Sin embargo, incluir nuevas funciones no es una tarea compleja.

7.2. Descripción del desarrollo para la extracción de modelos basados en reglas

En el sistema desarrollado para la extracción de modelos basados en reglas, cuyas bases teóricas han sido explicadas en la sección anterior, se diferencian dos aplicaciones. La primera y principal basada en un algoritmo genético es la encargada de realizar la búsqueda de la hipótesis que mejor explica los datos de entrada (`AGLearner`). La segunda, tiene que ver con el tipo de representación de las hipótesis, en nuestro caso conjuntos de reglas, y que es la encargada de codificar y decodificar la información de los individuos generados con algoritmo genético en forma de reglas. Con esta finalidad ha sido implementada como una librería DLL en lenguaje Visual Basic© llamada `SQLdeco.dll` que se incrusta en el código de `AGLearner` para calcular la cobertura de las hipótesis.

A continuación se describen las principales características de `AGLearner` para continuar con la descripción de `SQLdeco.dll`.

7.2.1. Aprendizaje con AGLearner V1.0

La aplicación `AGlearner`(figura 7.3) ha sido desarrollada en Visual Basic 6© [Alonso et al., 2002].

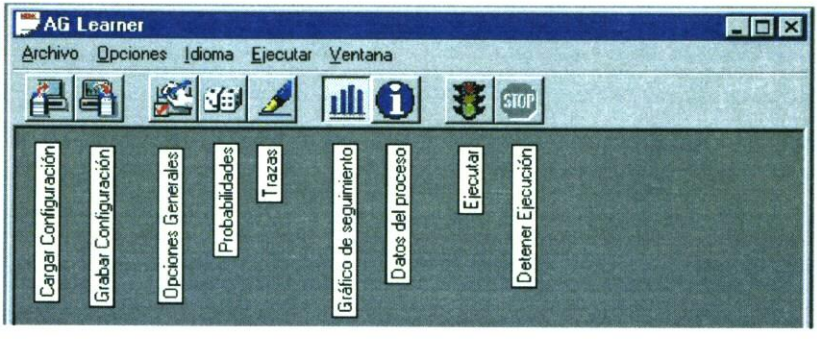


Figura 7.3: Menu y botonera de la aplicación AGLearner

La estructura de `AGLearner` está compuesta por los ocho formularios (ex-

tensión **.frm*) y los cinco módulos (archivos con extensión **.bas*), que aparecen el la figura 7.4. Los formularios implementan los elementos de interfaz de la aplicación con los usuarios. A partir de ellos se pueden incluir valores para parámetros de ejecución del algoritmo genético, como por ejemplo las probabilidades de mutación o cruce (*Fprob.frm*) o parámetros más generales como la longitud de los cromosomas, tipo de mutación, etc. (*FrmGeneral.frm*). Otros formularios sirven para visualizar de forma gráfica (FrmGraficas.frm) y numérica la evolución del proceso de búsqueda (*FrmDatosProceso.frm*), para seleccionar elementos para el almacenamiento de la información de las gráficas (*FrmTrazas.frm*) o para confirmar en cada uno de los anteriores formularios la selección realizada una vez pulsada la opción aceptar (*FrmConfirmacion.frm*). De todos estos formularios merece especial atención *FrmGraficas* formularios dedicado a la visualización de la evolución del proceso de aprendizaje o búsqueda por lo que se dedicarán más adelante unas líneas a explicar su funcionamiento. En cuanto a los módulos, éstos representan cinco subsistemas diferentes dedicados a: 1)iniciación de la herramienta (*móduloIniciacion.bas*), 2)representación gráfica de la evolución del proceso de búsqueda (*GraficoSetting.bas*), 3) bucle principal del algoritmo genético (*Main.bas*), 4) creación, sustitución y otras operaciones genéticas que se realizan durante el proceso de búsqueda sobre las diferentes poblaciones (*Genetico.bas*) y, finalmente, 5) incorporación de la función de fitness o función de calidad (*ModDescripcionProblema.bas*). Además, existen dos módulos de clase, *individuo.cls* y *poblacion.cls*, que permiten implementar, el algoritmo genético bajo el paradigma de orientado a objetos.

En lo que sigue se explica los componentes más interesantes y novedosos de la implementación de `AGLearner` dejando a un lado aspectos menos relevantes de la implementación por ser comunes a otras aplicaciones informáticas.

Módulo de clase: `Individuo` (individuo.cls)

La clase `Individuo` es un objeto que representa la unidad básica de un algoritmo genético. Por ello esta clase contiene propiedades como `cromosoma`

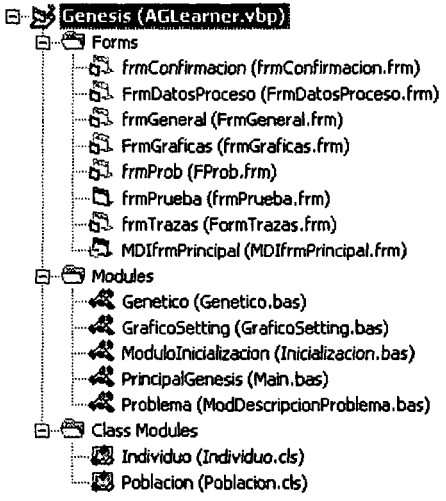


Figura 7.4: Archivos que componen la aplicación AGLearner

y `fitness`. La propiedad `cromosoma`, de tipo cadena (string), almacena la secuencia de caracteres binarios que representa al individuo. La propiedad `fitness`, de tipo numérica de doble precisión decimal (double), guarda el valor numérico de la función de calidad elegida.

Módulo de clase: `poblacion` (poblacion.cls)

La clase `Poblacion` representa a la colección de objetos `Individuos` y al grupo de posibles soluciones de cada generación. Este objeto contiene la propiedad `Count`, que cuenta el número de elementos de cada población, y los métodos `add` y `remove`, para añadir o eliminar individuos a una población. Durante el proceso de búsqueda, tan sólo se genera tres tipos de objetos `poblacion`: la población inicial, la población vieja o intermedia y la nueva población. Estas tres poblaciones cambian sus individuos a través de los operadores genéticos elitismo, mutación y cruce.

A continuación se describen brevemente los módulos y procedimientos más importantes en el proceso de búsqueda genética implementado.

Módulo: `PrincipalGenesis` (Main.bas)

`PrincipalGenesis` es el módulo donde se encuentra el bucle principal de ejecución del algoritmo. Las siguientes líneas muestran el pseudo-código de este proceso iterativo, desde el cual se hacen llamadas a los procedimientos que realizan las diferentes operaciones sobre las nuevas poblaciones. Este código se encuentra en el módulo `Genetico`.

```
Sub Public PrincipalGenesis()
          CrearPoblacionGeneracionInicial (PoblacionVieja)
      Do
           EvaluaFitness (PoblacionVieja)
           Seleccion (PoblacionVieja, PoblacionNueva)
           Cruzar (PoblacionNueva)
           Mutar (PoblacionNueva)
           If Elitismo Then IncorporaElitismo (PoblacionVieja, PoblacionNueva)
           PoblacionVieja = PoblacionNueva
Loop Until (CondicionParada)
End Sub
```

El primer paso del algoritmo es generar una población, que se denomina *población inicial*, con un número de individuos determinado por el usuario. Posteriormente comienza el proceso de búsqueda. Como se puede ver en el pseu-código se trata de un proceso iterativo realizado por el bucle `Do` y `Loop`, y que repite las mismas tareas en cada iteración. La población se evalúa, y después, se seleccionan los individuos en función de su calidad para posteriormente ser cruzados y/o mutados con lo que se generan nuevos individuos. El proceso almacena el mejor individuo en el caso de tener activo alguno de los mecanismos de elitismo. La etapa final del bucle consiste en reemplazar la población de partida por los nuevos individuos generados. Este proceso es repetido iterativamente hasta que se satisface una condición de parada, que puede ser o bien porque se alcanza la solución que sería el final ideal del proceso, bien se llega a un número determinado de generaciones (iteraciones) o bien, el usuario decide finalizar manualmente el proceso.

Módulo: Genético (genetico.bas)

El módulo genético (*Genetico.bas*) contiene el código de las funciones más importantes en el proceso de búsqueda genética, todos explicados en el apartado 4.2, como son: a) la generación de la población inicial aleatoria o generación de población inicial mediante semilla, b) la selección de los individuos más adaptados mediante el método de ruleta, c) los operadores genéticos de cruce en un punto, en dos puntos o de cruce uniforme, d) los operadores genéticos de mutación en un punto o en todos de la cadena cromosómica y e) los operadores genéticos de elitismo tanto *unitario*, se conserva el mejor individuo, como el llamado *modelo elitista* en el que se ordenan los individuos según su calidad y se seleccionan para formar la nueva población con los que presentan mayor valor de fitness.

Formulario: FrmGraficas (FrmGraficas.frm)

La visualización de la evolución del proceso de búsqueda o aprendizaje es determinante para evaluar si el proceso de búsqueda del modelo está progresando adecuadamente. Además puede permitir al usuario realizar operaciones encaminadas a ajustar mejor el procedimiento así como parar el proceso de búsqueda cuando se alcanza una solución exacta o de calidad suficiente para el usuario final.

La figura 7.5a presenta la pantalla de visualización gráfica del proceso, donde aparecen dos gráficas. La principal muestra la evolución de la búsqueda en las últimas generaciones mientras que la secundaria muestra un histórico de la evolución de generaciones anteriores. En la gráfica principal en el eje X se representa el número de generaciones y en el eje Y se muestra la fitness de cada individuo, el valor medio, máximo, mínimo o la desviación estándar de la fitness de la población, si están activadas las casillas correspondientes. Es importante destacar que un proceso de búsqueda que evoluciona de forma adecuada presenta gráficas para el valor medio, el mínimo y máximo con crestas y depresiones pero con una tendencia general hacia un valor máximo de fitness (convergencia). Los diferentes valores obtenidos durante el proceso se

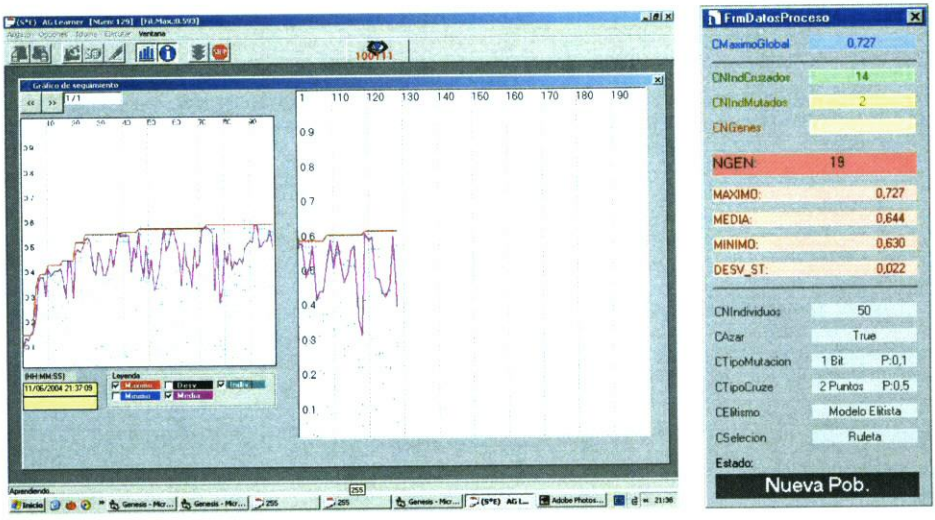


Figura 7.5: Pantallas de visualización (a) gráfica de proceso de evolución genética y (b) numérica de proceso de evolución genética

visualizan en la pantalla que muestra la figura 7.5b.

Módulo: ModDescripcionProblema (Problema.bas)

Este módulo es otra de las partes importantes de la implementación, ya que es donde se conecta el problema que se quiere resolver con el proceso genético, es decir donde se incluye un interprete que decodifique los cromosomas y/o calcular el correspondiente valor de fitness. Se trata por lo tanto del modulo que debe incorporar las funciones necesarias para estas dos tareas, y que dependerán del problema. El único requisito para su desarrollo es incluir un procedimiento llamado `Public Sub CalculaFitness(INDIVIDUO As INDIVIDUO)` que debe incluir las rutinas y funciones necesarias para calcular y asignar el valor de fitness del individuo examinado. Este modulo permite que `AGLearner` pueda analizar cualquier problema representado por medio de cadenas binarias.

7.2.2. Descubrimiento de Reglas con `SQLdeco.dll`

Con `AGLearner` se puede aplicar la búsqueda genética a la resolución de un amplio conjunto de problemas ya que los procedimientos asociados a la búsqueda genética, selección, cruce, mutación, etc son genéricos. En nuestro caso la búsqueda genética tiene cómo finalidad encontrar el conjunto de reglas (modelo) que mejor explica un conjunto de datos de entrada almacenados en una base de datos. Para la evaluación es necesario decodificar los individuos de la población y estimar la bondad de cada individuo actuando como solución del problema planteado. En nuestro caso esto se plasma en el desarrollo de una función específica que evalúa modelos basados en reglas mediante sentencias SQL. Esta función es parte de un archivo tipo DLL (*Dynamic Link Library*) que se ensambla en la aplicación `AGLearner`.

La DLL implementada está formada por cuatro archivos tal como se muestra en la figura 7.6: dos formularios (*SQLdecoFrm.frm* y *SQLdecoFitFrm.frm*), un modulo (*SQLdecoFrmMdl.bas*) y un modulo de clase (*SQLdeco.cls*).

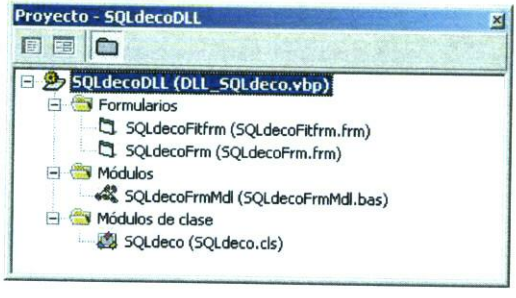


Figura 7.6: Archivos de la implementación `SQLDeco.DLL`

Tanto el formulario principal *SQLdecoFrm.frm* como *SQLdecoFitFrm.frm*, de nuevo, representan la parte de interfaz entre los procedimientos informáticos y el usuario, que se utilizan para introducir los parámetros del experimento que se desea realizar como el nombre y ubicación de los archivos contenedores de datos y conocimiento, o el tipo de aprendizaje a realizar (figura 7.7). Los procedimientos generales de la aplicación se incluyen en el modulo (*SQLdeco-*

FrmMdl.bas). Y finalmente, un modulo de clase, *SQLdeco.cls* que representa a objetos donde se incorporan las propiedades y métodos, que son visibles desde la aplicación principal (`AGlearner`).

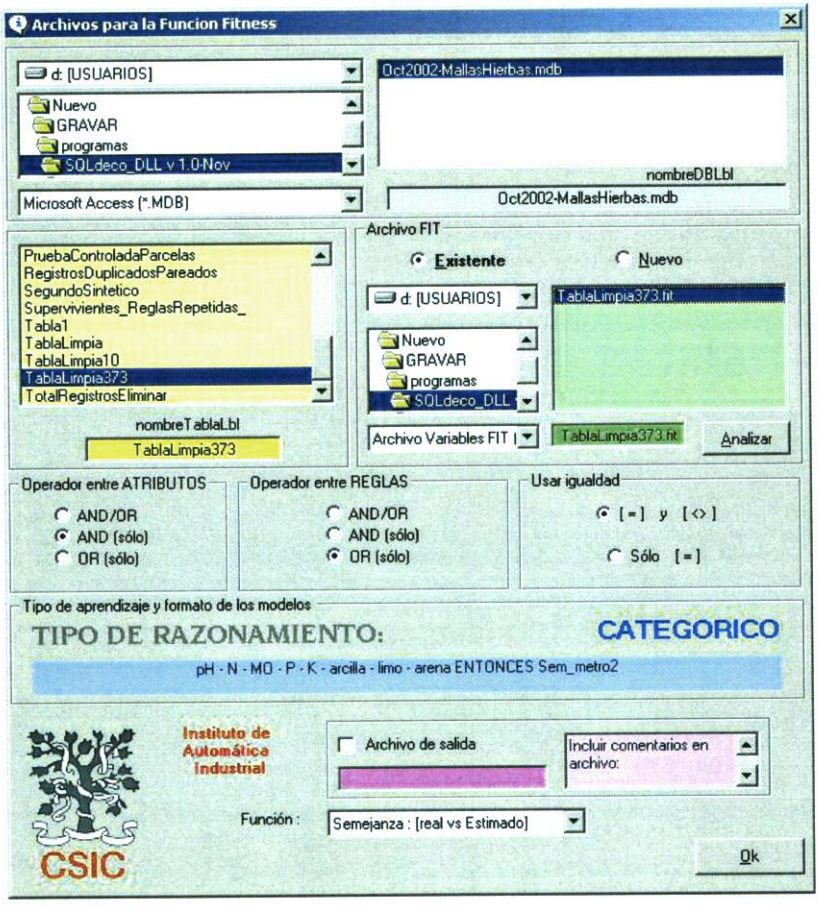


Figura 7.7: Formulario interfaz de la DLL donde se incluyen la información relativa al problema

`SQLDeco.DLL` presenta dos modos de funcionamiento. Cuando esta DLL es llamada por primera vez desde la aplicación `AGlearner` aparece el formulario de la figura 7.7 en el que el usuario especifica la base de datos Access© que

contiene los datos. A continuación aparecen todas las tablas contenidas en la base de datos, lo que permite al usuario detallar el conjunto de entrenamiento. Además, el usuario debe definir la información y parámetros necesarios para decodificar y evaluar a los individuos. En primer lugar, seleccionar el archivo de texto que contiene el nombre de las variables, las etiquetas y el número de bits que requiere cada una de ellas. Este archivo es denominado `fit`, y presenta los siguientes datos y estructura:

$$
\begin{array}{l}
\texttt{a,Atri}_1\texttt{,bits}_1\texttt{,etiqueta}_1(a),\ldots,\texttt{etiqueta}_1(z) \\
\texttt{a,Atri}_2\texttt{,bits}_2\texttt{,etiqueta}_2(a),\ldots,\texttt{etiqueta}_2(z) \\
\vdots \\
\texttt{a,Atri}_N\texttt{,bits}_N\texttt{,etiqueta}_N(a),\ldots,\texttt{etiqueta}_N(z) \\
\texttt{c,Atri}_{clase}\texttt{,bits}_{clase}\texttt{,etiqueta}_{(clase)},\ldots,\texttt{etiqueta}_{(no\ clase)}
\end{array}
$$

Cada línea escrita en el archivo representa la información de cada variable utilizada en la búsqueda, y que forman parte del cuerpo, tanto antecedentes como el consecuente, de las reglas. Cada línea contienen cuatro partes que se separan por comas. La primera parte, hasta la primera coma, incluye un letra, que puede ser `a` o `c`, para representar si el atributo que menciona esa línea formará parte del antecedente o en el consecuente, respectivamente. La parte representada por $\texttt{Atri}_N$ corresponde al nombre que tomará la variable N en el futuro modelo, y que coincide con el nombre de cada uno de los campos de la tabla a utilizar en el entrenamiento. A continuación, $\texttt{bits}_N$ determina el número de bits necesarios para codificar todas las posibilidades de una variable. El número de bits, que debe ser entero, depende del número o cardinal de valores o etiquetas que cada atributo puede adoptar $\texttt{Card}\ (\texttt{etiqueta}_{N(a)},\ldots,\texttt{etiqueta}_{N(z)})$, pero siempre es potencia de dos debido al carácter binario de cada bit. para conocer este número entero se utiliza el valor entero inmediatamente superior al que se obtiene de la expresión 14.

$$bits_N = log_2\ Card\ (etiqueta_{N(a)},\ldots,etiqueta_{N(z)}) \tag{14}$$

Cabe mencionar, que el número de etiquetas reales puede ser igual o menor a las posibilidades que permita un determinado número de bits, es decir, que tres etiquetas requieran dos bits, pero éstos permiten codificar cuatro. Por lo tanto, en estas ocasiones la posibilidad extra se decodifica como "`nulo`", es decir, no se incluye en la regla. El final de cada línea incluye todas las z etiquetas que presenta la variable N ($\texttt{etiqueta}_{N(z)}$), es decir, todo el dominio de valores simbólicos o nominales.

Este archivo al ser ASCII puede ser creado fácilmente de forma manual y con cualquier editor de texto, siguiendo las pautas descritas, o bien de forma automática con una herramienta a la que se accede a través del botón *Crear* que incluye el formulario de la DLL (figura 7.7). Presionando este botón aparece el formulario *SQLdecoFitFrm.frm* (Figura 7.8a), y la única acción que requiere del usuario es determinar qué variables formarán los antecedentes y cuales el consecuente, ya que los otros parámetros como etiquetas y número de bits son extraídas automáticamente de la tabla elegida previamente en el primer formulario (figura 7.7). Esta pantalla ofrece también la posibilidad de crearlo de forma semi-automática, donde el usuario va incluyendo variable por variable cada atributo, los bits y las etiquetas y le sirve de ayuda en este paso la información de la tabla que va presentando el formulario. Para conocer si un archivo es correcto, si los modelos incluyen toda la información deseada, o datos importantes como la longitud del cromosoma, entre otros parámetros interesantes, el botón *Analizar* del mismo formulario (Figura 7.8b) permite ver su contenido.

Una vez seleccionado el archivo donde se almacena la información de la codificación de las variables que compondrán las reglas, se puede determinar el valor de los operadores que van a componer las precondiciones y los conectores entre las reglas que formarán el modelo completo. Tal y como se vio en el apartado de las bases teóricas, el valor los dos operadores lógicos puede ser AND u OR, y el operador de comparación puede ser igual o desigual, según las necesidades y objetivos de la experimentación.

Es también en este punto en el que el usuario puede activar una operación

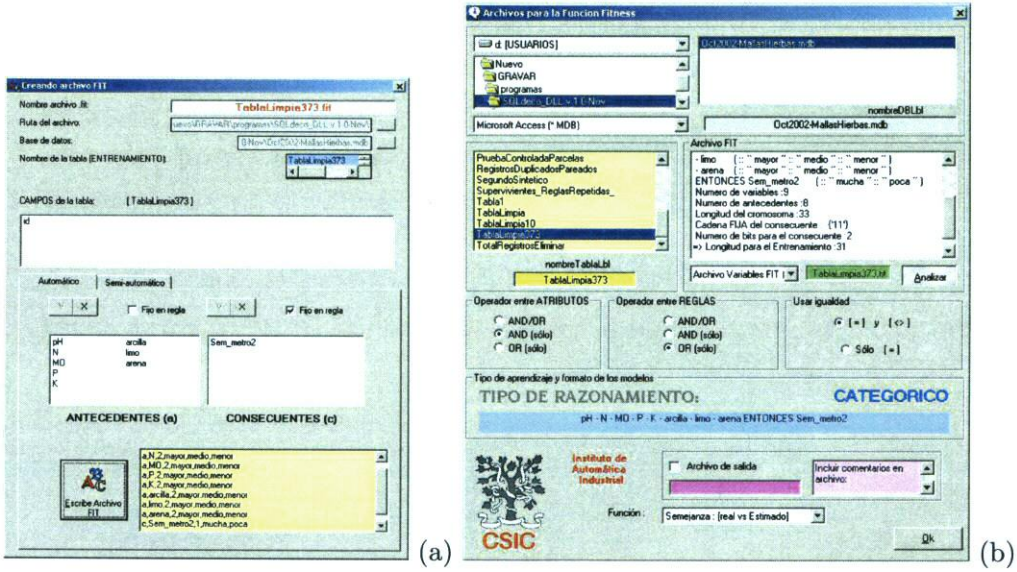


Figura 7.8: (a) Formulario que permiten la creación de archivos que contiene la información para la decodificación (`archivo fit`). (b) Pantalla del formulario principal de `SQLdeco.DLL` cuando se analiza algún archivo `fit` existente

que permite almacenar en cada generación todas las soluciones evaluadas que superen el último máximo alcanzado. Esta opción puede ralentizar el proceso, pero asegura que se guardan todas las soluciones de mayor calidad encontradas durante el proceso de búsqueda.

Para finalizar, un menú desplegable permite seleccionar la función de fitness que se desea aplicar de las descritas en la sección 3.1.3.

Una vez definida toda esta información la llamada a `SQLDeco.DLL` desde `AGLearner` desencadenan el cálculo de calidad de una hipótesis de la forma que se muestra en la figura 7.9.

En pocas palabras, el cromosoma, es decir las cadenas de ceros y unos que representa a un modelo candidato se transforma en una sentencia SQL utilizando la información contenida en el archivo `fit`, explicado anteriormente. Una rutina `intérprete` decodifica las subcadenas del cromosomas, formando las precondiciones con el nombre de una variable unido por un operador de

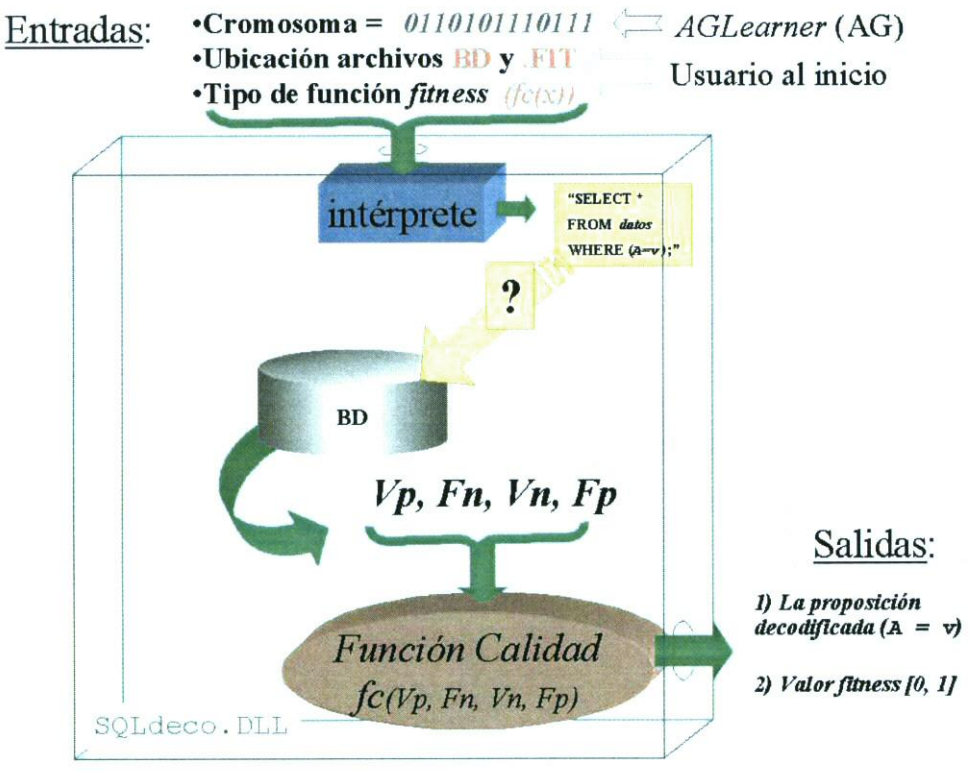


Figura 7.9: Flujo de tareas realizadas en `SQLDeco.DLL` para calcular la calidad de un cromosoma o individuo

comparación a un valor del dominio de la variable, e interpreta los operadores lógicos entre las precondiciones. Una vez se ha obtenido el conjunto de precondiciones se forma la sentencia SQL que denominaremos ϑ, que representa al posible modelo y que se incluye en una cláusula WHERE. Efectuada las consultas, se obtienen los valores para los parámetros Verdaderos positivos (V_p), Falsos negativos (F_n), Verdaderos negativos (V_n) y Falsos positivos (F_p). Con estos cuatro parámetros se computa la fitness utilizando la función que ha sido elegida por el usuario al comienzo de la ejecución.

Para finalizar decir que la utilización de esta DLL con `AGlearner` ofrece la posibilidad de visualizar los cromosomas y su representación en formato reglas,

tal como se muestra en la figura 7.10.

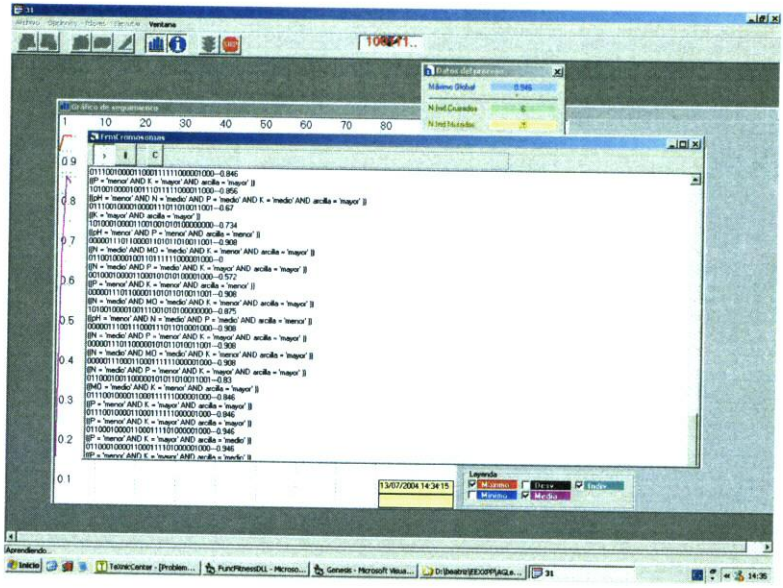


Figura 7.10: Visor de los cromosomas: la cadena binaria, la fitness y su representación (reglas)

Capítulo 8

EVALUACIÓN CON SQLGEN

El resultado final de la aplicación `AGlearner` es una secuencia binaria que se almacena en un archivo de texto una vez ha concluido el proceso de búsqueda. El cromosoma almacenado codifica al mejor individuo obtenido y se transforma en el conjunto de reglas que se presenta al usuario a través del siguiente modulo del sistema llamada `SQLgen`. Esta herramienta se basa en la librería explicada `SQLDeco.DLL`. La figura 8.1 muestra la pantalla de decodificación "cromosoma/conjunto de reglas". En este formulario el usuario debe introducir el cromosoma y, utilizando la información contenida en el archivo `fit` explicado en la sección 7.2.2, aparece el conjunto de reglas codificado y su valor de fitness, tal y como se muestra en la figura. Adicionalmente, para modelos de más de una regla en el formulario se indica la calidad por separado de cada una de las reglas.

Una vez que se conoce el modelo generado por `AGLearner` como un conjunto de reglas a partir de un conjunto de datos de entrenamiento, otro módulo de la herramienta (figura 8.2) permite validar el modelo mediante un nuevo conjunto el de validación, que recordemos puede ser un conjunto con nuevos registros previamente separado de resto, el conjunto inicial de datos, o cualquier conjunto, que tenga el mismo tipo de datos (variables y etiquetas) dependiendo de la técnica de validación seleccionada. Como se explicó en la sección 3.1.2.

Seleccionados el archivo fit y el conjunto que servirá para la validación, la herramienta presenta los siguientes parámetros de calidad del modelo descubierto: los parámetros V_p, F_n, V_n y F_p, el valor de fitness calculada mediante

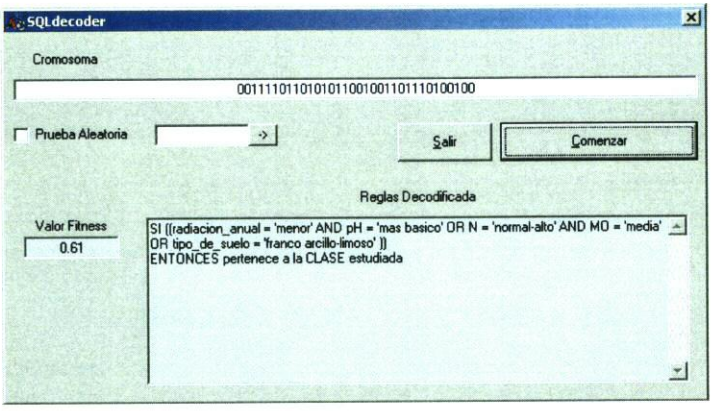


Figura 8.1: La decodificación en el conjunto de reglas final se realiza con el formulario `SQLDecoder`

la función seleccionada, la proporción de elementos bien clasificados, es decir la *exactitud* (Ex) y la certidumbre a través del parámetro *confianza* (Co), que determinaría el porcentaje de casos en los que el modelo es verdadero frente al total de casos en los que se dan las condiciones del modelo. El cálculo de la exactitud y de la confianza se realiza a partir de las ecuaciones 15 y 16.

$$Ex(\%) = \frac{Vp + Vn}{Vp + Fn + Vn + Fp} \times 100 \tag{15}$$

$$Co(\%) = \frac{Vp}{Vp + Fp} \times 100 \tag{16}$$

Esta herramienta además incorpora la posibilidad de presentar visualmente los resultados de la clasificación, a través de dos imágenes que representan la distribución espacial de los valores reales y los estimados por el modelo para la clase, tal como se puede ver en la figura 8b. La razón de incorporar una visualización en forma de mapa es que la herramienta de extracción de conocimiento (KDD) desarrollada se ha orientado principalmente a la obtención de modelos a partir de datos de entrada que tienen una componente espacial y que son habitualmente el resultado de una tarea de muestreo, por lo que la visualización

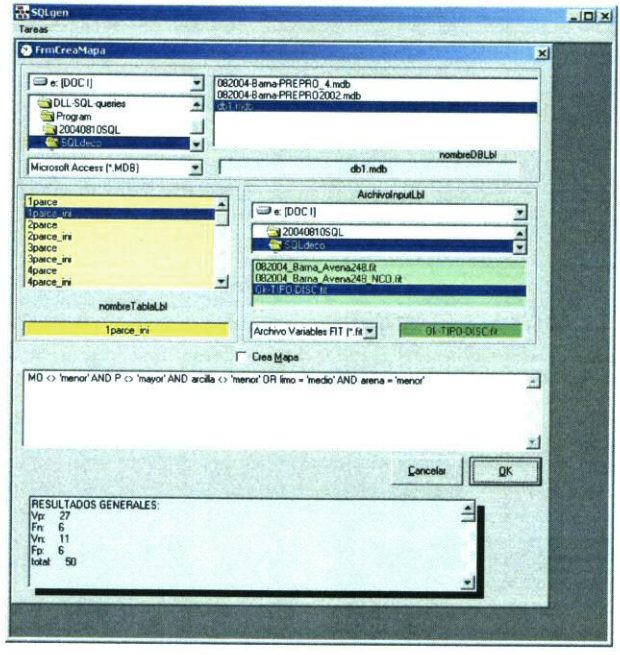
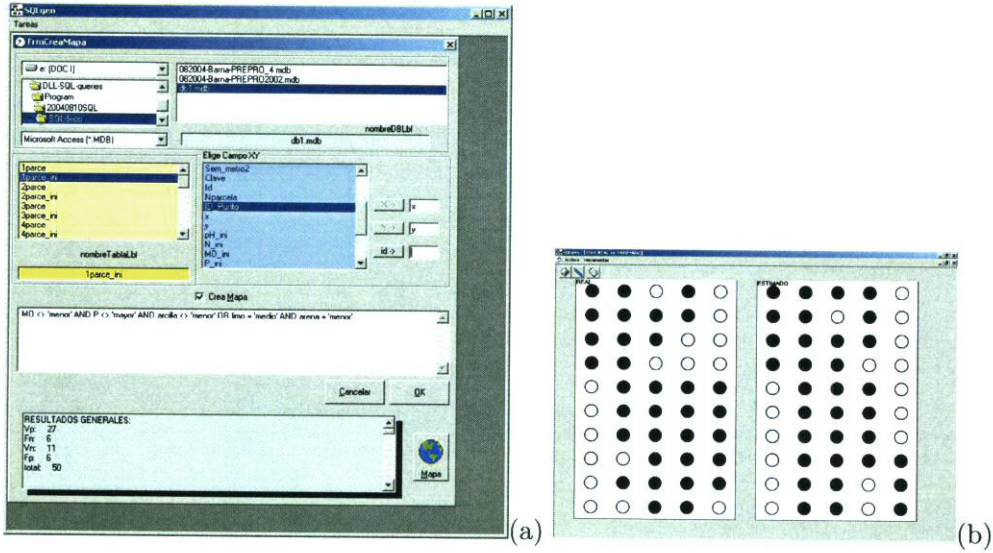


Figura 8.2: Pantalla para la evaluación y determinación de la calidad del conjunto de reglas decodificado a partir de un cromosoma

de la distribución espacial real frente a la estimada de la variable que se desea predecir es de gran utilidad para el usuario final. Por otra parte, recuérdese que la estrategia para la obtención de modelos, que se propone en este trabajo, parte de desdoblar el conjunto de datos de entrada en dos subconjuntos que representan ejemplos que pertenecen a la clase que se desea modelar y ejemplos que no pertenecen a esa clase. Tanto en el mapa que muestra la distribución espacial real de los ejemplos (figura 8a) como en el que muestra la estimada (figura 8b), los círculos negros representan la pertenencia a la clase que se esta modelando mientras que los círculos blancos representan la no pertenencia a la clase. Esta forma de representación de resultados permite el análisis visual de los mismos y la comparación de los valores estimados con los reales, lo que deriva en una potente herramienta para apreciar dónde el modelo funciona

mejor o funciona peor.



(a) (b)

Figura 8.3: Pantallas del evaluador **SQLgen** para elegir las variables para representar los resultados visualmente

Parte III

Caso de Estudio: *El control de malas hierbas*

Capítulo 9

Búsqueda de Modelos para Agricultura de Precisión

Las prácticas de cultivo tradicionalmente se han orientado hacia una gestión uniforme del campo ignorando la variabilidad espacial inherente constatada por la mayor parte de los agricultores. Es más, el aumento del tamaño de los campos debido a una creciente mecanización ha contribuido a incrementar este novedoso tipo de gestión. Esto contrasta con la situación que se suscita en los sectores más ecologistas de la sociedad que demandan el desarrollo de técnicas que permitan una agricultura sostenible, siguiendo las directrices de la FAO, lo que frecuentemente se conoce como técnicas de agricultura ecológica. A pesar de esta fuerte demanda social y por muy diversas razones, entre ellas, culturales, económicas, técnicas o la propia dificultad de transformar los sistemas agrícolas [Maxwell, 1999], hoy en día todavía se realiza una gestión uniforme agroquímicos lo que se traduce en un fuerte impacto medioambiental. En consecuencia generar prácticas agrícolas orientadas una gestión óptima (la mejor acción, en el mejor lugar y en el mejor momento) de agroquímicos es sumamente importante desde un punto de vista medioambiental [Bongiovanni & Lowenberg-Deboer, 2004].

El término *agricultura de precisión* nace en 1986, según D. Fairchild en su trabajo presentado en el *II Congreso Internacional de Gestión Precisa para Sistemas Agrícolas* de 1994 (*Site-Specific Management for Agricultural System*), para describir una área de investigación que tiene por objetivo introducir nuevos principios y tecnologías para alcanzar una actuación más precisa (espacial y

temporalmente), minimizando las cantidades de agroquímicos utilizadas, y por lo tanto, gestionando óptimamente los campos de cultivo [PIERCE & NOWAK, 1999]. La idea de adecuar las prácticas agrícolas a las necesidades de cada lugar del campo se remonta a la época de los primeros colonos norteamericanos que, imitando a los indios nativos, enterraban pescado en ciertos sitios donde el cultivo crecía peor, lo que tenía como efecto una mejora en los nutrientes de la zona [SEIM, 2000]. Análogamente, las técnicas de precisión actuales se basan en el tratamiento de los cultivos de forma heterogénea, e incluso proponen utilizar los agroquímicos a dosis variables, teniendo en cuenta las características observadas en cada campo y siempre que exista una variación significativa en la abundancia de malas hierbas o del cultivo a lo largo de un mismo campo. De forma sintética, la tarea principal de la agricultura de precisión es transformar los datos, recogidos en los campos, en decisiones necesarias en las etapas de tratamiento. Ahora bien, la Agricultura de Precisión no es simplemente la habilidad de aplicar localmente tratamientos variables, debe ser vista como la habilidad de supervisar y evaluar de modo preciso la actividad agrícola a escala local o de finca y entender con suficiente profundidad los procesos involucrados de modo que se puedan obtener los objetivos deseados con modificaciones sobre las prácticas agrícolas. No se trata necesariamente de maximizar la producción si no más bien de obtener el máximo beneficio con el menor impacto medioambiental [COUSENS ET AL., 1987, BLACKMORE, 1994, EARL ET AL., 1996, KROPFF ET AL., 1997, ROBERT, 1999].

Para entender los beneficios medioambientales de incorporar una gestión agrícola de precisión hagamos un pequeño examen de la contaminación que pueden provocar las diferentes sustancias empleadas habitualmente como tratamiento en los campos de cultivo. Los agroquímicos más peligrosos son los pesticidas, término que engloba insecticidas, fungicidas y herbicidas. La mayoría de ellos son sustancias químicas orgánicas -COPs (*contaminantes orgánicos persistentes*)- que pueden deteriorar el medioambiente porque son bioacumulativos y persisten mucho tiempo en el suelo, en el agua y en la biota, y además, son más móviles que las sustancias inórgánicas. Algunos de estos pesticidas son

de baja solubilidad lo que aumenta su potencia de contaminación de las aguas [AUGE & NAGI, 1999]. El DDT (*dicloro difenil tricloroetano*) pertenece a este grupo y fue prohibido por la legislación europea para uso agrícola, además de por su elevada toxicidad, por su baja solubilidad, y sin embargo, se han detectado recientemente exposiciones a estas sustancias [PRESS, 2003]. con estas características, la utilización en exceso en general de agroquímicos pueden provocar la salinización y contaminación de los suelos fértiles y de acuíferos de forma irreversible. Uno de los perjuicios más conocidos del exceso de químicos es la eutrofización de las aguas, provocada por los nitratos en concentraciones superiores a 50mg/litro (corroborado por la Organización Mundial de la Salud), fenómeno que llega a producir la muerte de la fauna acuícola. Pero más impactante resulta conocer que el uso excesivo de estos fertilizantes es peligroso también para la salud humana, ya que puede llegar a producir una alteración de la hemoglobina[1] provocada por envenenamiento y que puede ser mortal. En los últimos tiempos, además se han hecho muchos estudios que asocian los agroquímicos con determinadas enfermedades, como el Parkinson o la infertilidad (Veáse la recopilación realizada por el centro de Minnesota para la defensa medioambiental [WWW.MNCERTER.ORG, 2004]). Incluso, la EPA (*Environmental Protection Agency*) tiene catalogados algunos pesticidas como cancerígenos. Destacamos que el contacto con los pesticidas puede ser por los alimentos tratados así como por la respiración, a través de su exposición a la piel y los ojos. Otro ejemplo de los efectos producidos por los herbicidas se observó en 1998 cuando el gobierno colombiano, para erradicar los cultivos de coca, utilizó productos altamente tóxicos capaces de aniquilar cualquier vegetal con hojas. Incluso ligeras exposiciones eran suficientes para matar árboles maduros lo que provocó la deforestación de millares de hectáreas en áreas cercanas a los campos de cultivo de coca [MCGRAW-HILL, 1998]. Además, de la inutilización de las tierras de cultivo, el fenómeno de la persistencia de

[1] La Metahemoglobinemia o síndrome del bebe azul es una patología provocada por el alto porcentaje de metahemoglobina en la sangre, que puede provocar cianosis (10 a 25 %), alteraciones respiratorias (35 a 40 %) e incluso la muerte (>70 %). De hecho, este tipo de globulos mal formados son un indicador del grado de exposición a determinados agentes químicos. Enciclopedia Medline Plus (*www.nlm.nih.gov*)

agroquímicos puede llegar a provocar, en ambientes relativamente húmedos, que los residuos de la fumigación, también letales, pasen al ciclo del agua infiltrándose a través de lluvias y del riego y dispersándose rápidamente una vez que entran en arroyos o corrientes subterráneas. En las últimas décacas se ha hecho patente que elementos tóxicos como el selenio, el mobdileo y el arsénico, procedentes del drenaje agrícola e incluidos en el ciclo del agua de forma no intencionada, pueden provocar problemas de contaminación hídrica que paradógicamente son una amenaza para los cultivos (Letey et al., citado en [RHOADES, 1993]). La fumigación realizada para la gestión de un cultivo juega también un papel importante en los flujos de gases de invernadero (CO_2, CH_4 y N_2O) [ROBERTSON ET AL., 2000]. El peligro es claro si se combina este fenómeno de contaminación de las aguas con el hecho de que estas sustancias son tóxicas para los seres vivos, incluidos los humanos. Así pues es un hecho que la protección frente a las anteriores sustancias requiere un esfuerzo en áreas científicas como la ecotoxicología y en general en las ciencias relacionadas con estos problemas ecológicos.

En resumen, determinar una gestión adecuada del campo que siga, entre otros, principios como el enunciado en la *Ley del Mínimo de Liebig*[2] es la tarea fundamental de las técnicas desarrolladas en agricultura de precisión. Este nuevo estilo de hacer agricultura ha sido implementado en países de tradición agrícola, como los Estados Unidos o Australia, mostrando un porcentaje de éxito del 60%. La incorporación en Europa ha sido más reciente, principalmente en Dinamarca donde se han obtenido también buenos resultados [FERNÁNDEZ-QUINTANILLA, 2003].

Poco a poco este procedimiento de gestión del campo va siendo más popular, pero su incorporación de forma generalizada está aún lejos y depende básicamente de varios factores entre los cuales son fundamentales los siguientes:

1) El desarrollo de nuevas técnicas que abaraten los costes de las tareas

[2] Ley formulada para los sistemas agrícolas por *Justus von Liebig* (1803-1873) que establece que el crecimiento no está controlado por el total de las recursos disponibles sino por la escasez y que pone de manifiesto el hecho de que el aumento de nutrientes no incrementa el crecimiento de las plantas [THEFREEDICTIONARY.COM, 2004]

de evaluación del cultivo y de la generación de actuaciones adecuadas sobre el cultivo.

2) La implementación de herramientas de fácil uso e implantación que no requieran unos conocimientos técnicos complejos.

3) Un endurecimiento de las leyes (por ejemplo las directrices de la CEE) que impida la degradación de los recursos.

4) Una mayor concienciación social sobre la importancia de la conservación de nuestro entorno.

9.1. Control preciso de malas hierbas

La agricultura de precisión tiene dos líneas principales de actuación. Por un lado, la fertilización de aquellas zonas de las parcelas donde el cultivo crece peor, y por otro, el control de aquellas plantas que reducen la producción y compiten por los recursos disponibles en el suelo, las *malas hierbas*. La segunda línea es de gran importancia debido a que la mayor parte de los cultivos sufren en mayor o menor grado la presencia de mala hierba y aunque se trata de un problema que presenta la agricultura desde sus comienzos todavía no se ha encontrado una solución que siendo eficaz sea inocua con el medio.

Todas aquellas plantas que aparecen en cantidad suficiente dentro del cultivo y que no son deseadas se catalogan como *malas hierbas*. Al contrario que otras plagas no atacan al cultivo, sin embargo afectan a su crecimiento reduciendo el tamaño y la calidad de la cosecha bien debido a la competencia entre especies o simplemente por contaminación del grano con otro tipo de semillas. En algunos casos se ha constatado que las malas hierbas pueden beneficiar a un cultivo[3] sin embargo no existe el conocimiento suficiente sobre cómo incorporar las malas hierbas en el sistema de gestión del cultivo cara a obtener beneficios. Lo cierto es que las malas hierbas deben ser controladas por sus

[3] La mala hierba puede ser huésped de determinados insectos que eliminan parásitos del cultivo que causan de daños peores, además ayudan a aumentar la diversidad evitando el carácter monocultural, hecho que en ocasiones favorece el crecimiento del cultivo. También pueden prevenir la erosión y la consecuente perdida de nutrientes depositados en el suelo.

efectos negativos [FRÖHLING, 1980] que se traducen en perjuicios económicos [LINDQUIST ET AL., 1998] provocados principalmente por:

1) la *reducción de rendimientos* por competencia de la luz, agua o nutrientes,

2) la *interferencia en la recolección* que provoca dificultades en la etapa de cosecha,

3) la *reducción del valor de los productos* por impurezas que incorporan olor, sabor o humedad

y 4) el *incremento de los costes* de producción por etapas adicionales de limpieza o secado, además del coste del herbicida, el laboreo y tener que hacer rotación de cultivos menos rentables.

Existen diferentes estrategias para el manejo de las malas hierbas, a) la *prevención* controlando las semillas, limpiando la maquinaria y, mediante la aplicación de sustancias, manteniendo el campo libre de infestación e impidiendo, de este modo, que aparezcan las especies problemáticas; b) la contención que anualmente calcula un *umbral económico* para tratar sólo si los daños superan los beneficios; c) la *reducción* de la mala hierba hasta un determinado nivel para alcanzar la máxima rentabilidad económica y finalmente d) la *erradicación* total de especies sólo cuando son especialmente nocivas. Por otra parte, en malherbología se distinguen tres filosofías para el manejo de las malas hierbas, en función de las estrategias que se adopten: 1) la forma de enfrentarse a las malas hierbas más antigua, el control *a cualquier precio* reduciendo al más bajo nivel posible la infestación para minimizar las perdidas del rendimiento y prevenir mayores infestaciones en el futuro; 2) el control a partir del *umbral económico* por el que se controlan las malas hierbas únicamente cuando el tratamiento no supone un coste superior al beneficio; y finalmente, 3) el control que tiene en cuenta el *umbral ecológico-económico* e incorpora la filosofía de mantener las poblaciones de malas hierbas considerándolas cruciales para la biodiversidad y el equilibrio del sistema. Aunque en la actualidad, la mayor parte de los agricultores utilizan técnicas de umbral económico, existe un gran esfuerzo por parte de la comunidad científica por dirigir la agricultura a estra-

tegias menos invasivas. En cualquiera de estas prácticas resultaría interesante tener en cuenta que las malas hierbas se distribuyen irregularmente a lo largo de los campos cultivados por lo que, para obtener un control óptimo, cualquier estrategia requiere conocer cómo y dónde aparece la infestación en el cultivo con una exactitud apropiada al tipo de acción de control que posteriormente pueda llevarse a cabo.

Determinadas especies no sólo se presentan irregularmente, sino que además aparecen de forma agregada formando *rodales*, es decir áreas donde se concentra la infestación mientras que el resto del campo permanece limpio [CARDINA ET AL., 1997, KROHMANN ET AL., 2003]. Las fotografías de la figura 9.1(a y b) de [MAXWELL ET AL., 1998] muestran rodales de *Avena sterilis* en un campo de cereal de Montana (EE UU). La fotografía (9.1 c), tomada por el equipo de protección vegetal del CCMA[4] del CSIC en campos de cereales de Madrid, muestra un rodal de color grisáceo de *Ansinkia*; una especie americana introducida recientemente en España. Por último, en la fotografía (figura 9.1 d) se puede distinguir claramente varios rodales que infestan de forma muy localizada un cultivo. Una de las características destacables de los rodales de algunas especies es la persistencia espacial [WILSON & BRAIN, 1991, GERHARDS ET AL., 1997], detectada incluso después del empleo de herbicidas [BARROSO ET AL., 2001][BARROSO ET AL., 2005], lo que induce a pensar que no existe aleatoriedad en la distribución de los rodales en un campo sino que responden a determinadas condiciones. Así, muchos estudios han llegado a la conclusión de que muchas poblaciones de malas hierbas son dinámicamente estables -espacial y temporalmente- y descartan un comportamiento caótico, por lo que sería posible predecir su abundancia [FRECKLETON & WATKINSON, 2002].

Utilizar la heterogeneidad en la distribución espacial y persistencia en zonas de algunas malas hierbas para generar mapas de riesgo que guíen la aplicación de los tratamientos selectivos, puede ser un interesante modo de optimizar el control de las infestaciones. De hecho, numerosos estudios se han orientado

[4] Equipo formado por el Dr. Cesar Fernández-Quintanilla, Dra. Judit Barroso y Ing. David Ruiz

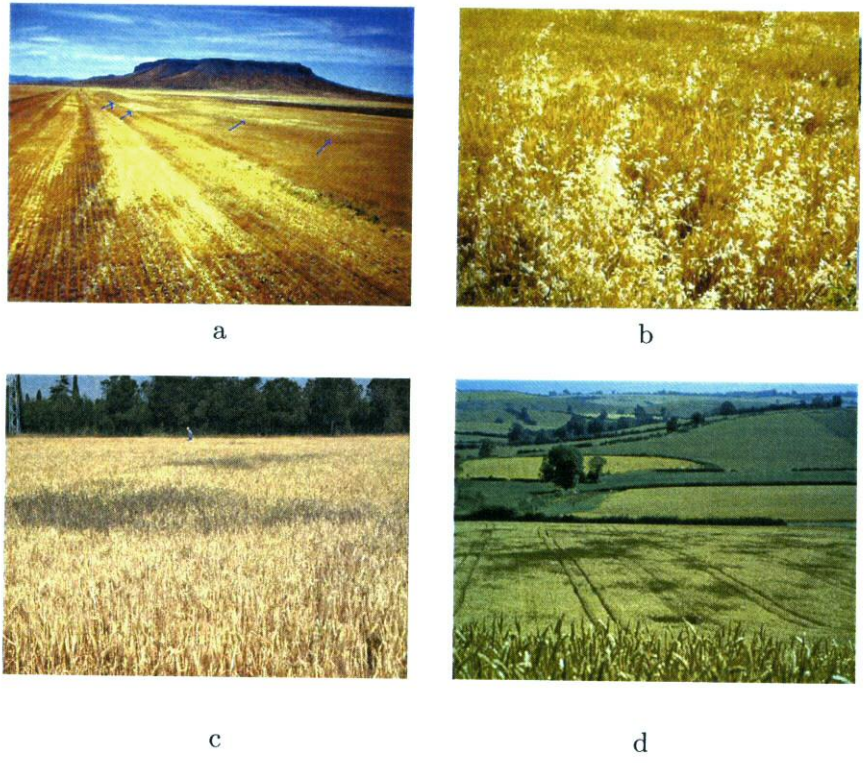


Figura 9.1: Fotografías de rodales de malas hierbas en diferentes cultivos

al desarrollo de métodos que permitan tratamientos precisos (*site-specific control*) [CHRISTENSEN ET AL., 1999, HEISEL ET AL., 1999, BARROSO, 2004]. Los tratamientos localizados siempre implican una reducción del uso de herbicida, tanto en cantidad como en número de aplicaciones. Efectivamente, desde un punto de vista económico, se ha demostrado que los tratamientos heterogéneos en campos de cereal invadidos por avena loca, podrían ahorrar aproximadamente del 31 % al 61 % de herbicida [WALTER ET AL., 2001] e incluso un 78 % [BARROSO ET AL., 2001], lo que supondría, indudablemente, un notable beneficio económico y, aún sin cuantificar, un lógico beneficio medioambiental.

Una aplicación localizada de tratamientos óptima se sustenta principalmente en una detección correcta de las zonas de mayor proliferación de las malas

hierbas [CLAY ET AL., 1999], porque la eficacia de los tratamientos a dosis variable depende de precisar la situación y de la distribución de los rodales dentro del campo de cultivo [BARROSO ET AL., 2004].

La determinación de las zonas del campo a tratar se puede abordar desde dos perspectivas diferentes [CHRISTENSEN & HEISEL, 1998]. En la primera se usan métodos de visión artificial, aún en desarrollo, que a partir del análisis de la señal suministrada por diferentes sensores pretenden discriminar el cultivo y la mala hierba en tiempo real (*localización real de la infestación*). El segundo tipo de métodos aboga por la utilización de mapas de riesgo generados a partir de la información histórica de los campos (*localización estimada de la infestación*).

Dentro de la primera perspectiva podemos considerar la supervisión automática de cultivos mediante cámaras de color basadas en sensores CCD[5] usadas desde tierra, que se pueden emplear fundamentalmente según dos metodologías claramente diferenciadas para la detección de vegetación [THOMPSON ET AL., 1990]: 1) Análisis de diferencias geométricas entre suelo, malas hierbas y cultivo (perfil, estructura de la planta, localización); por ejemplo, perímetro y área de las hojas en imágenes o análisis morfológico en imágenes en color [ANDREASEN ET AL., 1997], tamaño y número de *blobs* - "gotas", formas de aspecto orgánico - [GARCÍA-PÉREZ ET AL., 2001], o imágenes en visible e IR próximo [BRIVOT & MARCHANT, 1996, GARCÍA-PÉREZ ET AL., 2000, 2001]. 2) Estudio de las diferencias en la reflectancia espectral [VRINDTS ET AL., 2002]. Existen algunos trabajos en los que se combinan ambos métodos [BENLLOCH & RODAS, 1997, MARTIN-CHEFSON ET AL., 1999] y otros en los que se empieza a utilizar la información de contexto a fin de reducir los tiempos de cómputo para aplicaciones específicas en tiempo real [FREDRIKSSON & TURUJLIJA, 1998, GARCÍA-PÉREZ ET AL., 2000]. La evidente ventaja que supone la integración de múltiples características de los objetos de la imagen contrasta sin embargo con el coste computacional que conlleva una detección

[5] Un *Charge Couple Device* o *Dispositivo de Carga Acoplada* es un sensor en el que se basan las cámaras digitales con estructura reticular y cuya capacidad se mide en puntos o píxeles que es capaz de adquirir, y que almacenan un voltaje en proporción a la iluminación que recogen.

y clasificación multiparamétrica.

El segundo grupo de técnicas para determinar el área a tratar, que se conocen comúnmente como *históricas*, se desarrollan en dos etapas y se basan en la recogida de muestras y otros datos en los campos. En una primera campaña, se construyen los mapas de infestación basado en datos muestrales [HEISEL ET AL., 1996, COLLIVER ET AL., 1996], y posteriormente, en una segunda etapa, esa información transformada en un mapa de tratamiento se utiliza para la fumigación selectiva de aquellas zonas del campo que, según el primer tipo de mapa, tienen más probabilidad (riesgo) de ser infestadas [KRUEGER ET AL., 2000, FERNÁNDEZ-QUINTANILLA ET AL., 2001]. La información generada para la construcción de estos mapas de riesgo podría aportar además conocimiento del problema, y ser útiles en la caracterización del comportamiento de las malas hierbas.

La construcción de mapas de los rodales es especialmente útil si la especie de mala hierba es estable, y la infestación se puede extrapolar, con cierta certidumbre, a años sucesivos. La principal ventaja es que se ahorra el tiempo y los costes asociados al proceso de construcción del mapa de infestación. Existen diferentes especies de malas hierbas que se consideran estables, como pueden ser a) especies perennes clónicas, b) las que se dispersan antes de la cosecha como la especie *Avena* en trigo) c) especies de vida corta que son removidas y exponiéndolas a la superficie por la maquinaria como la especie *Veronica* y d) aquellas que están en suelos donde no hay operaciones de arado que muevan sus semillas [ZANIN ET AL., 1998].

Una vez establecida la localización más o menos exacta de la infestación, puede ser interesante conocer los factores que provocan la heterogeneidad en la dimensión y densidad de los rodales. Por este motivo es frecuente que, además de recoger datos sobre la infestación, se recopile en los mismos puntos información relacionada con el entorno ambiental, como puede ser las propiedades físico-químicas del suelo, características topográficas, biológicas, etc. El descubrimiento de relaciones entre esta información y la abundancia de malas hierbas, a través de distintas técnicas, permitiría detectar posibles pautas de

aparición y evolución de los rodales, es decir patrones de comportamiento de las infestaciones. Los modelos de densidad de mala hierba en términos de información relacionada con el entorno podrían ser útiles para detectar posibles causas, o en la elaboración de mapas de infestación predictivos, es decir que establezcan las zonas de mayor probabilidad de aparición. En algunos casos, incluso se podría informar sobre la dosis recomendable según la infestación estimada. Es más, la construcción de estos modelos ayuda en el proceso de obtener conocimiento sobre el problema, permitiendo a los expertos conocer mejor el funcionamiento de estos sistemas biológicos, y fijando qué factores son claves en la proliferación de malas hierbas [MAXWELL, 1999].

Un debate, todavía abierto, es el establecimiento de las variables relevantes en la construcción de estos modelos generales. No cabe duda que la aparición de rodales y la persistencia de estos en algunas especies es causa directa de propiedades intrínsecas de la semilla, como son el peso, la forma, el tamaño. Incluso puede estar relacionada, entre otros, con factores externos como la competencia entre especies [COUSENS & CROFT, 2000]. Ahora bien, se puede considerar que todos estos factores tienen una incidencia más o menos igual en todo el cultivo y por lo tanto, no serían suficientes para justificar la heterogeneidad espacial de la infestación. En consecuencia, además de todos estos factores, deben existir entre las variables del entorno agrícola otras que puedan ser importantes o incluso determinantes, como por ejemplo las propiedades edáficas, la climatología, la historia de cada campo, o incluso actividades antrópicas relacionadas con la siembra o la cosecha. Un sistema complejo como el agrícola, tiene un elevado número de variables que puedan tener alguna influencia. Sin embargo, son destacables las variaciones físico-químicas del suelo como la calidad del suelo, variaciones hídricas o el contenido en nutrientes entre otras [WHELAN, 1998]. Del mismo modo que zonas de campo pueden resultar más productivas por el nivel de retención de nutrientes o la textura del suelo, las malas hierbas pueden crecer allí donde las condiciones edáficas o/y ambientales favorezcan su aparición. Para plantear esta hipótesis, es importante explicar que los suelos de un campo de cultivo no son uniformes en cuanto a sus carac-

terísticas, hecho que se puede observar en la fotografía 9.2a. Esta imagen de *La Higueruela*, finca experimental del CSIC[6] en Toledo, muestra claramente la variación de la composición del suelo y probablemente, también la textura. La fotografía de la figura 9.2b muestra otro ejemplo de variación de composición, en este caso debido a fenómenos químicos de intercambio de sodio[7]

a b

Figura 9.2: Fotografías que muestran variaciones de propiedades físicas del suelo en campos de cultivo

9.2. IMPORTANCIA DE LOS FACTORES EDÁFICOS

Encontrar las condiciones que influyen en la heterogeneidad espacial de la abundancia de mala hierba, no es una labor fácil ya que aparentemente existen multitud de factores que podrían inicialmente ser responsables. Ahora bien, el suelo y su heterogeneidad influye directamente en la producción vegetal, ya que contribuye a la variación de nutrientes, retención y transporte de agua y otros fluidos. La principales características del suelo y cómo pueden afectar a la cubierta vegetal (capítulo 1 de [WHELAN, 1998]) se pormenorizan a continuación.

▷ **Tipo de suelo/textura**. La textura se refiere al tamaño de las partículas

[6] http://www.ccma.csic.es/fincas/fincas.htm
[7] http://asalag.tripod.com.mx/AS_terrestre/cultivos_con_problemas.htm

minerales (arcillas, limos y arenas) que tiene el suelo. Esta composición de partículas confiere al suelo importantes propiedades que pueden condicionar el crecimiento vegetal, porque determina el tipo de suelo, y a la vez la proporción de elementos fundamentales como las arcillas, y otros elementos orgánicos e inorgánicos. La textura diferencia tres principales clases de suelos: arenosos, cuando la arena supera el 70 %; arcillosos con más de 40 % de arcillas, que pueden ser arcillo-arenosos si tienen menos de 45 % de arena o arcillo-limosos si disponen de más del 40 % de limo; y finalmente, los suelos margosos, que constan de diversos grupos de partículas de arena, limo y arcilla, y van desde los margo-arenosos hasta los margo-arcillosos. Las texturas existentes según el triángulo de clasificación de el *US Department of Agriculture* (*USDA*) se muestra en la figura 9.3.

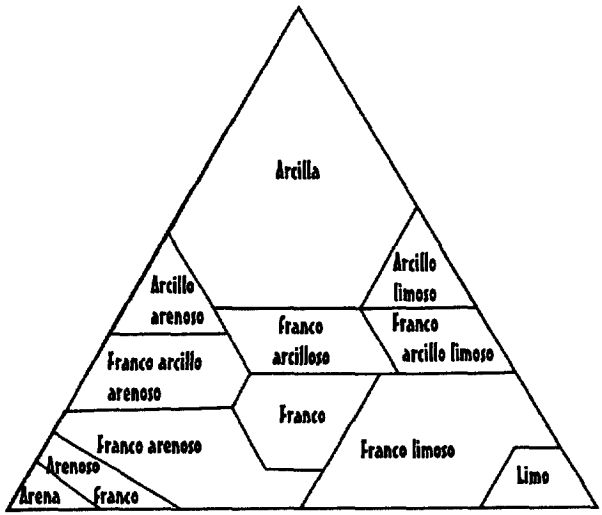


Figura 9.3: Triángulo de clasificación de suelos basado en los parámetros texturales

▷ **Estructura**. Se define como estructura de un suelo la disposición que toman las partículas que lo forman, y relacionada con la textura. La estructura del suelo condiciona la penetración física de las raíces y además

regula la relación humedad-aire necesaria en crecimiento de plantas y en la actividad microbiana. Por ejemplo, una reducción de la cantidad de oxígeno necesario en los procesos metabólicos, indirectamente provoca una disminución de los nutrientes disponibles y causa perturbaciones en el pH y en el potencial de oxidación-reducción, lo que puede provocar la desaparición de plantas. El drenaje y la retención de agua se regula también a través de la consistencia que determinan las partículas del suelo, fijando el grado potencial de erosión. Por último, la estructura es una propiedad inestable y se ve afectada por la saturación del agua de lluvia o riego o el peso por ejemplo el provocado la maquinaria. Esta compactación, además de impedir el desarrollo de las raíces o el transporte de sustancias, incrementa el esfuerzo que requiere la labranza.

- **Materia orgánica**. La cantidad de materia orgánica o *humus*, que no suele superar un 5 % del contenido del suelo, indica la fertilidad del suelo y juega un papel primordial en la conservación de las propiedades físicas y químicas, ya que almacena humedad y nutrientes, influyendo en la actividad biológica microscópica. El *humus* proporciona la mineralización de nitrógeno, de fósforo y de sulfuro, y como consecuencia puede ser el autor directo del aumento de la humedad y de los nutrientes disponibles en el suelo. Además de intervenir en el ciclo de varios nutrientes, como el nitrogeno o el azufre, aumenta la *Capacidad de Intercambio Catiónico* (CIC), también relacionada con la fertilidad.

- **Humedad**. El contenido en agua del suelo, es crucial. La humedad afecta a la fertilización determinando como se distribuyen los nutrientes. Ahora bien, depende de las propiedades arriba comentadas y de las aportaciones por precipitación, lo que hace que este factor tenga un componente estocástico importante, que dificulta la estimación del valor de humedad para cada zona del suelo y que estén en auge trabajos y desarrollos orientados a la medida directa de la humedad del suelo[8].

[8] Destacan el sistema de control *AquaPro-Sensors* capaz de estimar el porcentaje de agua en el suelo ± 2 % de error, midiendo la constante dieléctrica del suelo, a cualquier profundidad, utilizando

- **Nutrientes**. La disponibilidad y la adición de nutrientes son los pilares fundamentales de la agronomía moderna. La variación de nutrientes constituye una pieza clave en el crecimiento de las plantas ya que sólo el 10 % de las raíces son capaces de absorber los nutrientes, por lo que cualquier pequeña disminución significa una gran reducción de la cantidad que finalmente alimenta a la planta. Entre los nutrientes de primer orden se encuentran los macroelementos que suelen escasear en los suelos y provocar limitación en el crecimiento. Macroelementos son: 1) el *nitrógeno* (1-6 % de tejido vegetal seco) en estado mineral, necesario en la formación de las proteinas, clorofila y ácidos nucléicos; 2) el *fósforo* (1 % de tejido vegetal seco) que interviene activamente en la mayor parte de las reacciones bioquímicas de la planta, la síntesis de los ácidos para la síntesis de energía, proteinas y de las reservas energéticas que contienen las semillas; y 3) el *potasio* (1 % de tejido vegetal seco) que regula la permeabilidad de las membranas celulares, el equilibrio ácido-básico necesario en la formación de sustancias de reserva y resistencia a daños. Los elementos secundarios, son el azufre, el magnesio, el calcio y el boro que habitualmente no escasean en las tierras de cultivo.

- **pH**. El pH es un valor que se usa para indicar la acidez o alcalinidad de una sustancia y se define como el potencial de hidrógeno calculado como el logaritmo de concentración molar de los iones hidrógeno (H+ ó hidronio H3O+); es decir pH = -log[H+]. Por tanto, el pH describe el estado de carga eléctrica de las partículas orgánicas e inorgánicas del suelo. Su variación indudablemente afecta a la disponibilidad de nutrientes, incluso si estos se han aplicado uniformemente. Por ejemplo, con valores de pH bajos y en presencia de aluminio o manganeso, los nutrientes llegan a ser altamente disponibles pero también tóxicos para las plantas.

Muchos son los trabajos que, con técnicas fundamentalmente estadísticas, han estudiado la posible relación entre propiedades físicas y la aparición de un

técnicas de radiofrecuencia. *www.aquapro-sensors.com/Independant-Tests.htm*, 2000

determinada especie de hierba [ADREASEN ET AL., 1991, DALE ET AL., 1992, CARDINA ET AL., 1995, DIELEMAN & MORTENSEN, 1999, DIELEMAN ET AL., 2000a,b, WALTER ET AL., 2002]. Ejemplos de correlaciones encontradas entre algunas propiedades del suelo y ciertas especies de determinados campos de cultivo, pueden encontrarse en el trabajo de WALTER ET AL. [2002]. Una de las conclusiones de este artículo, es que la dependencia espacial encontrada en un campo puede ser diferente a otros campos y además, de un año a otro para el mismo campo. Asimismo, los autores indican que la variación de las propiedades edáficas es uno de los posibles factores principales que pueden determinar la presencia de rodales. Por lo tanto, la relación entre la infestación y las propiedades físicas es un hecho estudiado y corroborado para ciertas especies vegetales, pero además es frecuente que estas propiedades también estén correlacionadas entre sí [CAMBARDELLA ET AL., 1994].

Uno de los principales inconvenientes de utilizar los factores edáficos en los modelos de comportamiento de las malas hierbas en cultivos es el coste asociado al análisis (en laboratorio) y muestreo de un número significativo de puntos en cada parcela. Un modelo basado en términos edáficos puede ser muy exacto y bueno, pero en algunas ocasiones difícil de llevar a la práctica por lo costosa que resulta la realización de muestreos anuales. De aquí que exista un indudable interés en el desarrollo de herramientas y sensores que abaraten los costes asociados a la adquisición de este tipo de información. En [ADAMCHUK ET AL., 2004] se presenta una completa lista de sensores y conceptos relacionados para la toma de información "en tiempo real" (*on-the-go*) de propiedades químicas, físicas y mecánicas. Los sensores pueden ser eléctricos, electromagnéticos, ópticos, radiométricos, mecánicos, acústicos, neumáticos y membranas electroquímicas, es decir hay un amplio abanico de sensores que prácticamente abarcan la adquisición de las propiedades de interés del suelo y del cultivo.'

9.3. Problemas en cultivos de cereales de invierno

Uno de los objetivos del presente trabajo es la caracterización de las zonas de los campos de cereal de invierno (cebada y trigo en régimen de secano) en las que la *Avena sterilis L.* (también conocida como avena loca) y el *Lolium rigidum Gaud.* (cizalla o vallico) son un problema para el rendimiento del cultivo, siendo necesaria la aplicación de herbicidas. A continuación se describen brevemente las características más importantes de estas dos especies.

Avena sterilis L.

La avena (figura 9.4) se encuentra entre las 10 especies más invasivas del mundo. Se trata de una mala hierba típica de zonas mediterráneas, y pertenece a familia de las gramíneas, como el resto de los cereales. La altura de las panículas varía desde los 50 a los 150 centímetros. Las hojas tienen una longitud de 60 centímetros y su ancho no llega a 12 milímetros. Las plantas presentan ramificaciones desde la base y la espiga, y presenta inflorescencia con forma piramidal cuyas ramificaciones se expanden con la distancia. Su periodo de vida esta comprendido entre los meses de octubre y abril, y suele presentar un periodo de emergencia primario en otoño y uno secundario en primavera. Además, todas las especies de avena tienen un periodo de latencia relativamente largo en cantidades altas, sobre todo en el momento de la cosecha. Llegan a persistir en el suelo de las tierras de cultivo de tres a cuatro años. La avena se autopoliniza, por lo que plantas aisladas también pueden generar semillas, además su capacidad reproductiva es muy alta, produciendo de 400 a 800 semillas por planta e incluso cantidades superiores que se incrementa cuando no existe competencia. El efecto de proliferación rápida se intensifica al tener una baja mortalidad.

El patrón de estas especies puede estar controlado por las condiciones ambientales así como por la existencia de otras especies. Por ejemplo, la germinación generalmente se ve favorecida con temperaturas suaves (10°C), y su cantidad disminuye considerablemente si los valores de temperaturas superan los 18°C o están por debajo de los 5°C. Sin embargo, también actividades como la labranza del campo pueden determinar su existencia porque puede elevar el

a b

Figura 9.4: (a) Variedades de avena (*http://www.fao.org/docrep/T1147S/t1147s01.jpg*) y (a) detalle de *Avena sterilis* ssp. *Ludoviciana* tomada en los campos de Madrid por el equipo del Centro de Ciencias Medioambientales del CSIC

número de brotes, al remover los 10 centímetros más superficiales de cubierta edáfica donde se encuentran las semillas.

Se trata de una especie muy competitiva con el cultivo sobre todo con los cereales como el trigo o la cebada con los que comparte muchas características. A ambos les afecta la sequía, pero el cultivo resiste un poco mejor. Por otra parte, el trigo compite mejor por la luz esto quiere decir que temas como la altura del cultivo, el número de vástagos, la producción temprana de biomasa y el tamaño de las hojas del cereal pueden provocar una disminución de la avena. Así mismo algunos estudios han constatado que una mayor nitrogenización proporciona una mejor respuesta en el crecimiento del cultivo reduciendo la infestación. Ciertos experimentos sugieren la obtención de cultivos fuertemente competitivos fertilizando en determinadas condiciones ambientales y climáticas para, de este modo, disminuir el grado de infestación de avena [GONZÁLEZ-PONCE & SANTÍN, 2001].

La aparición de avena loca provoca una disminución, en primer lugar del propio cultivo por competencia y como consecuencia una reducción muy importante del beneficio, que se agrava por el bajo valor en el mercado. Entonces, el pequeño margen para las ganancias hace que el cultivo sea muy sensible a la contaminación con este otro tipo de grano [SCURSONI ET AL., 1991]. Por lo que

cantidades muy pequeñas de infestación, como 15 plantas por metro cuadrado, puede provocar perdidas importantes de cultivo, según CUSSANS [1980], quien además propone una escala de infestación de avena loca, asociando a cada nivel el grado de perjuicio económico. Además de la perdida de cultivo hay que restar al rendimiento el coste que supone la limpieza del grano. Un intervalo económico de perdida aceptado por los malherbólogos para la avena es el propuesto por [WILLE ET AL., 1998], en este trabajo se propone un tratamiento a media dosis para una situación entre 20 y 190 plántulas por metro cuadrado. Con el tratamiento recomendado se obtendrían entre 140 y 235 semillas metro cuadrado lo que, según estos autores, supondría una perdida aceptable para el cultivo. Es importante plantear estrategias apropiadas de control para reducir la cantidad de infestación, ya que realizar eficazmente el tratamiento puede hacer que no haya prácticamente avena loca en un periodo de cuatro a cinco años [FERNÁNDEZ-QUINTANILLA ET AL., 1997] citado en [BARROSO, 2004]. El último trabajo concluye también que, incluso utilizando dosis medias, la localización de la infestación se mantiene de tres a cinco años en sistemas monocultivo de cebada. En resumen, se conocen algunos factores que pueden determinar la aparición y controlar el crecimiento de la avena loca, sin embargo, el sistema en el que se incorporan estas malas hierbas es tan complejo y variable que no se han determinados factores que puedan ser considerados generales y utilizados para la gestión de cualquier campo de cultivo de cereal. La complejidad estriba en que las condiciones que determinan el tratamiento más adecuado parecen variar dependiendo del tipo de suelos, del agua disponible en el suelo, de los cultivos y del clima.

Lolium Rigium Gaud.

El *Lolium rigidum Gaudin* [TABERNER, 1996] o comúnmente conocida como Cizalla o Vallico, al igual que las avenas pertenece a la familia herbácea *Poaceae*. Es una especie nativa del área mediterránea donde se adapta muy bien a sistemas agrícolas, fundamentalmente a cultivos de cereal. También prolifera en Australia, Europa, norte de África y algunas zonas de Asia, en general en áreas caracterizadas por un clima templado y cálido. Además de

afectar a cereal de invierno, también aqueja al olivo, el almendro y la viña.

Botánicamente, el lolium (figura 9.5) es una gramínea anual que presenta gran variedad genética. Este género se distingue bien en fase de plántula del resto de gramíneas arvenses, como la avena, alopecurus, bromus, Poa y gramíneas cultivadas, por los tonos rojizos de la base de su tallo, y del trigo o la cebada, además, por el brillo y la estrechez de sus hojas. Requiere mucha cantidad de agua para su emergencia.

Figura 9.5: Detalles fisiológicos de *Lolium rigidum*

El lolium ridigum es problemático en cultivos de cereal de invierno, preferentemente de cebada, porque se adapta perfectamente a los sistemas agrícolas de estos cereales que se cultivan durante los meses de octubre y noviembre y se recogen en junio y julio.

Los daños que produce en cultivos de cebada y trigo son importantes por-

que disminuye la producción, merma la calidad del grano y, en consecuencia, obliga a una etapa posterior de limpieza mecánica de las semillas. Por ejemplo, parcelas infestadas con 700 plantas por metro cuadrado, pueden provocar perdidas de hasta el 40 % de la cosecha. Sin embargo, el fenómeno que provoca tales perdidas, es muy complejo y en consecuencia la evaluación del problema es difícil. El conocimiento en los periodos críticos de competencia con el cultivo es fundamental para comprender los daños sobre el cultivo, datos como la localización concreta de la infestación, la relación entre la densidad de la mala hierba y el cultivo, la fertilidad del suelo, la cantidad de agua disponible pueden influir en este sistema agrícola.

Al igual que en la infestación por avena, la baja rentabilidad económica y los precios de mercado obligan a reducir los costes añadidos debidos al laboreo, la utilización de fertilizante o el coste del empleo de semillas certificadas. El control del *Lolium rigidum Gaud.* se realiza tradicionalmente mediante reducción, para mantenerlas por debajo de su nivel de infestación habitual, sin necesidad de erradicarlas.

Capítulo 10

Descripción y Preparación de los Datos

Los datos utilizados para el desarrollo de este trabajo provienen de dos localidades diferentes de la Península Iberíca en las que se han realizaron los experimentos incluidos en los proyectos en el que se encuadra esta tesis. En lo que resta de capítulo se describirán los conjuntos de datos de partida así como el proceso llevado a cabo para preparar estos datos (preprocesamiento) con el fin de extraer conocimiento (modelos) en forma de conjuntos de reglas. Es importante resaltar que los datos fueron tomados en campos con diferentes versiones de la herramienta de adquisición de datos georeferenciados desarrollada, `Fiesta`, descrita en el capítulo 5. En esta etapa de preprocesamiento se ha utilizado la aplicación descrita en el capítulo 6 y que es también una aportación del presente trabajo, `preparaDAT`.

10.1. Datos de los campos de Madrid

10.1.1. Descripción

El primer grupo de muestras estudiadas procede de dos áreas del sureste de la provincia de Madrid, enclavadas en los municipios de Arganda del Rey y Nuevo Baztán (figura 10.1). Las cuatro parcelas seleccionadas eran de cultivo de cereal de invierno y presentaban, de modo natural, rodales de avena loca.

La recogida de datos, realizada por el mencionado anteriormente equipo de protección vegetal CCMA (CSIC), se efectuó en cinco muestreos reticulares con puntos distribuidos regularmente en cuatro de los cinco casos. Los datos

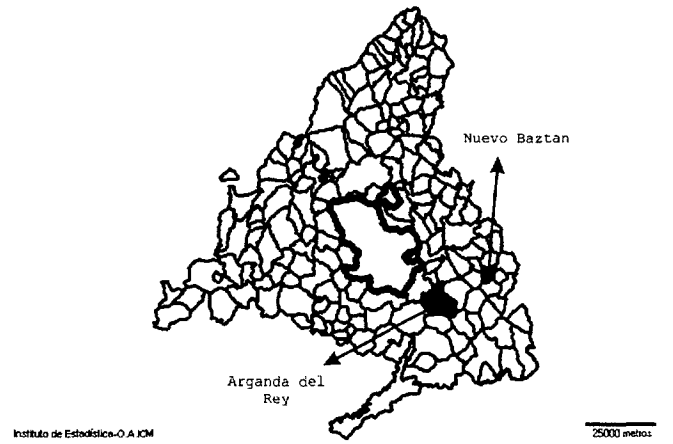


Figura 10.1: Localización de los campos situados en la provincia de Madrid en Arganda del Rey y Nuevo Baztán

recogidos se agrupan en los siguientes tipos:

(a) Datos relacionados con la densidad de infestación de avena en cada punto de muestreo. Se utilizaron dos procedimientos de medida; conteo en el campo del número de plántulas en el punto (emergencia temprana) y conteo en laboratorio del banco de semillas existente en el suelo (lluvia de semillas[1]). Como se vio en la sección 9.3, se puede establecer un número de semillas fértiles por planta de avena. A pesar de que la cantidad de semillas por planta es variable, los expertos establecen que una planta puede generar 10 semillas viables (*Ing. Agr.* D. Ruiz, *comunicación personal, 21 de enero de 2002*). En el caso de recogida de semillas, para cada punto, se recogió una muestra de tierra de dos kilos, aproximadamente una capa de 15 centímetros de profundidad de la zona incluida en un marco metálico cuadrado de 0,33 × 0,33 m^2 (figura 10.2). (b) Además, se recogieron dos kilos adicionales de muestra, localizados a los lados del cuadrado en los quince centímetros más superficiales, para analizar determinadas propiedades edáficas, como el grado de acidez (pH), la granulometría y la composición química. Con la granulometría se determinaron los

[1] *Lluvia de semillas* es el aporte de semillas al suelo por parte de las plantas adultas producidas al final del ciclo de vida

porcentajes de arena, limo y arcilla. Con el análisis de la composición química se obtuvo el nitrógeno extraíble (N), fósforo (P), el potasio (K) y la materia orgánica (MO). Según los expertos todos estos parámetros pueden considerarse estables temporalmente, hecho que es importante resaltar. Las propiedades físicas del suelo analizadas (arcilla, limo y arena y pH) son prácticamente invariables, y se siguieron practicas de laboreo superficial que en teoría no deben de alterar nada la textura del suelo. El contenido de materia orgánica del suelo puede evolucionar con el tiempo, pero habitualmente es de una forma tan lenta que se necesitan periodos de muchos años para observar diferencias. El fósforo y el potasio son más variables. Se añaden todos los años al fertilizar y se remueven también todos los años con las extracciones que hace el cultivo. Aunque este sistema de aportes y las extracciones, debería estar en equilibrio, existe una acumulación, debida al abonado en exceso, pero tan lenta que permite considerarlos estables en el suelo. El único elemento que varia rápidamente en el suelo es el nitrógeno, que puede proceder de los abonos pero además a través de la descomposición de la materia orgánica o, también por los aportes de agua, de lluvia o de riego. El ciclo del nitrógeno en el suelo es muy complejo (puede diluirse en el agua, descender a la zona de raíces, ascender por capilaridad en periodos de sequía, etc). Sin embargo, es frecuente que los agrónomos introduzcan esta variable en el estudio de variabilidad de las especies vegetales dentro de un mismo campo de cultivo por su importancia en la caracterización y gestión de tierras de cultivo. (*Dr.* C. Fernández-Quintanilla, *comunicación personal, 14 de octubre de 2003*). (c) Finalmente la muestras recogidas fueron georeferenciadas con alguna versión de la herramienta de adquisición implementada (apartado 5) y un receptor GPS. La información recopilada de cada punto permite conocer la posición absoluta de cada punto y permite la creación de mapas. En total, en las 5 campañas de muestreo se recogieron 536 puntos.

Los datos recogidos en las cuatro parcelas se han denominado *ArgandaPoveda* (muestreo n°1), *ArgandaPovedaFondo* (muestreos n°2 y 5), *Baztan* (muestreo n°3), y *BaztanLadoEntero* (muestreo n°4). Las figuras 10.3, 10.4 y 10.5 indican la distribución espacial de las muestras en cada campaña mientras

Figura 10.2: Marco metálico para determinar el área de muestreo

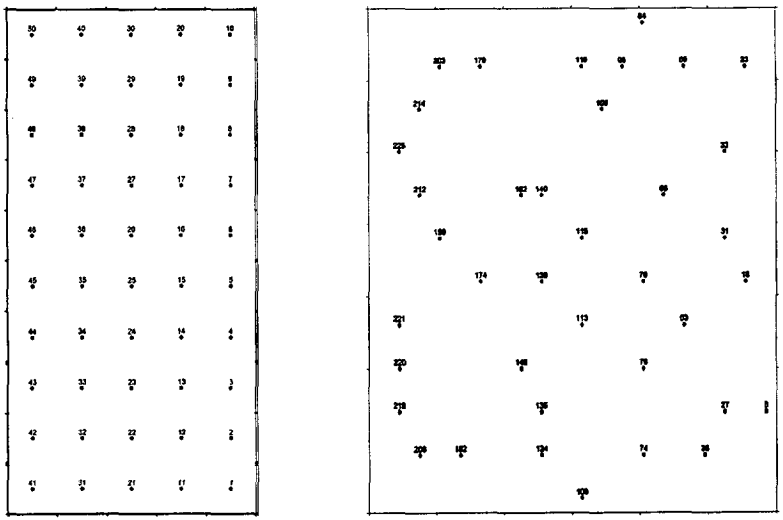
que algunas características de las parcelas y de los muestreos relacionados con la infestación de avena loca se resumen en la tabla 10.1. Asimismo, el apéndice B muestra los mapas de las variables estudiadas generados con la aplicación SURFER v.6 utilizando como método de interpolación el krijeado.

Muestreo nº	Tamaño (ha)	Topografía	Malla (m)	puntos	Conteo avena loca	Fecha avena	Fecha P.Suelo
1	0,5	plano	10×10	50	semillas	Jul-2000	Jul-2000
2	1,6	plano	12×6	228(38)	semillas	Jul-2000	Jul-2000
3	0,9	irregular	10×10	96	semillas	Jul-2000	Jul-2000
4	1,2	irregular	10×10	124	plántulas	Feb-2001	Jul-2000
5	1,6	plano	12×6	228	semillas	Feb-2001	Feb-2001

Tabla 10.1: Algunas características de los 5 muestreos de densidad de avena loca, realizados en los 4 campos de cereal de la Comunidad de Madrid

A continuación, se realiza una breve descripción de los datos de partida y la información general a tener en cuenta para el diseño y desarrollo de las posteriores etapas del proceso de extracción de modelos o proceso KDD.

ArgandaPoveda (muestreo 1). Se trata de una parcela de 0,5 hectáreas en la que se ha cultivado cebada en régimen de secano en los últimos tres años. La finca está situada en una vaguada arcillosa. En este terreno el grupo de

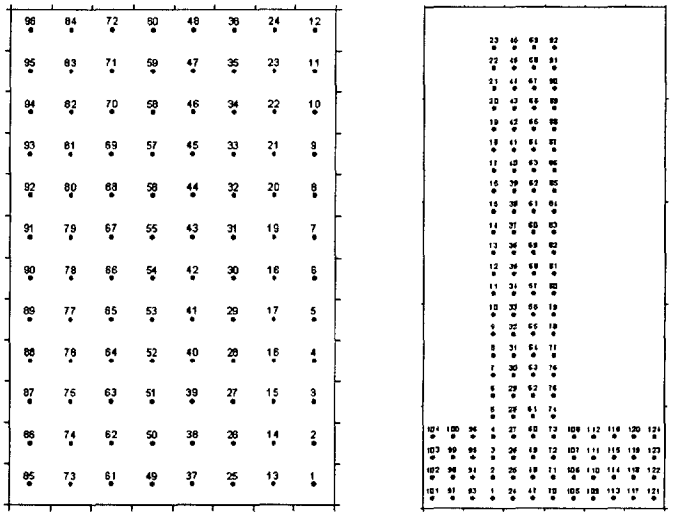


1 2

Figura 10.3: Croquis de distribución de muestras en las campañas 1 y 2, en las parcelas *ArgandaPoveda* y *ArgandaPovedaFondo*

CCMA lleva a cabo un estudio sobre la movilidad de rodales y la disminución de densidades de mala hierba con la aplicación de herbicida, y en el momento del muestreo la cantidad de herbicida fue de media dosis [BARROSO, 2004]. Se recogieron datos relativos a la densidad de avena y a los factores edáficos en una malla de 5 y 10 puntos, con un espaciado regular de 10 metros. La densidad de avena loca se calculó a partir datos de semillas (lluvia de semillas) por metro cuadrado en la campaña del 2000, obteniéndose un rango de valores para la densidad de avena en esta parcela de [*0-7 413*] en semillas por metro cuadrado. Los datos edáficos de materia orgánica (MO), nitrógeno(N), potasio (K), fósforo (P) y granulometría se recogieron también en la campaña de julio 2000, después de la cosecha. En esta parcela la granulometría media viene determinada por 21 % de arcilla, 48 % de limo y 31 % de arena, resultando un suelo tipo franco (figura 10.6).

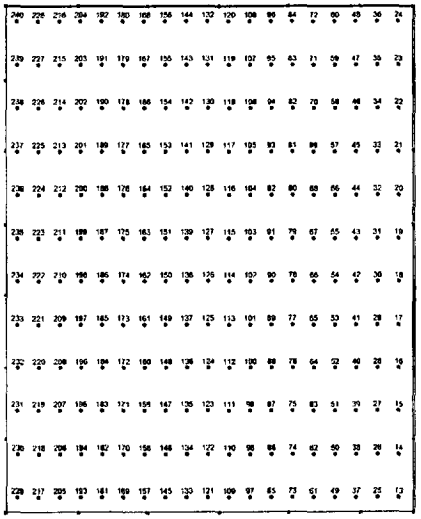
ArgandaPovedaFondo (muestreos 2 y 5). La parcela es de 1,5 hectáreas y se tienen datos de infestación de avena loca de dos campañas, 1999-00 y



3 4

Figura 10.4: Croquis de las muestras de las campañas 3 y 4 de las parcelas *Baztán*, y *BaztanLadoEntero*

2000-01. La malla está compuesta por 228 puntos distribuidos en 12 filas por 19 columnas. En la primera campaña se recogieron las 228 datos sobre avena, sin embargo solamente se analizó las propiedades edáficas de 38 muestras, que finalmente son utilizadas en el estudio. En la segunda campaña, en febrero de 2001, se miden las propiedades edáficas y el número de semillas de los 228 puntos. En ambos casos el cálculo de densidad de avena se realizó a partir del número de semillas en los 228 puntos muestreados, que se encuentran en el intervalo [*0-7 660*]. En cuanto a la composición del terrenos en valores medios tenemos un 34 % de arena, un 46 % de limo y un 20 % de arcilla. El suelo es, por tanto, de tipo franco aunque existen zonas de tipos franco limosos y franco arcillo-limosos. Durante el primer año (2000) no se empleo herbicida y el segundo año se realizó un tratamiento preciso no uniforme descrito en [RUIZ ET AL., 2002] ya que el objetivo del equipo de CCMA es conocer el comportamiento de las malas hierbas, en términos de densidad de población, frente al empleo de diferentes dosis de herbicida.



5

Figura 10.5: Croquis de las muestras recogidas en la parcela *ArgandaPovedaFondo* durante la campaña 5

Baztan (muestreo 3). Esta parcela se sitúa en una vaguada arcillosa con agricultura de secano, tiene un tamaño de 80 por 120 metros, y las muestras recogidas se distribuyen en forma de retícula regular con un espaciado de 10 metros, lo que da un total de 96 puntos de recogida. Durante el año 1998-99 se cultivó leguminosas y en 1999-2000 se sembró cebada. No se ha empleado ninguna dosis de herbicida. Datos con los que podemos trabajar son el número de semillas recogidas en julio del 2000. También fueron analizadas las propiedades edáficas durante la misma campaña. El suelo de la parcela es de tipo franco, aunque presenta áreas de tipo franco arcilloso. La composición en valores medios es de un 38 % de arena, un 37 % de limo y un 25 % de arcilla. El rango de semillas por metro cuadrado, la densidad de infestación, de las dos campañas está comprendida entre [*0-902*].

BaztanLadoEntero (muestreo 4). Se trata de una parcela de 1,5 hectáreas, se tienen un total de 124 muestras de suelo recogidas en una retícula que se muestra en la figura 10.4[4], y con un espaciado de 10 metros. En esta última

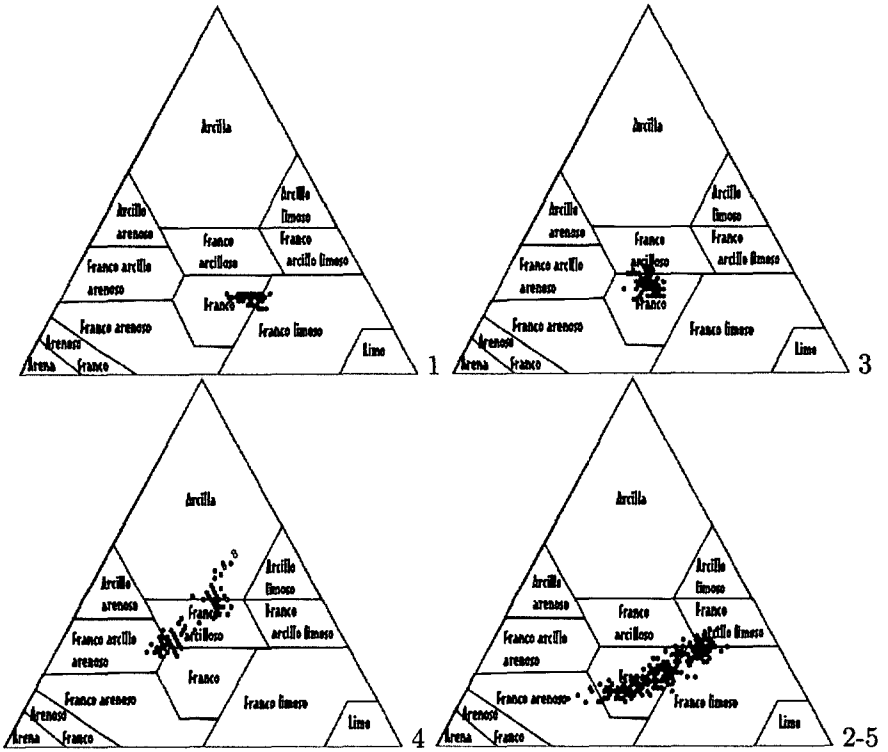


Figura 10.6: Triángulos de texturas de cada parcela muestran mayoritariamente suelos con estructura franca

parcela, los valores medios para la composición del terreno son de un 36 % de arena, un 30 % de limo y un 34 % de arcilla, por lo que el suelo es del tipo franco-arcilloso presentando zonas franco-arcilloso-arenoso. El rango de densidad en número de semillas, en este caso estimado a partir del número de plantulas, está comprendida entre $[0\text{-}7\ 100]$.

Conclusiones de los primeros análisis de caracterización

En una primera aproximación resulta útil realizar análisis estadísticos descriptivos de los datos. Un resumen de estos análisis se muestra en la tabla 10.II. El análisis revela la heterogeneidad en los datos de las parcelas, por ejemplo en el caso de la variable fósforo, las parcelas situadas en Arganda presentan

	C.	n^o	MO %	N %	pH -	P mg/100g	K mg/100g	Arena %	Limo %	Arcilla %
máximo	**1**	50	2,63	0,154	8,14	374	700	37	53	22
mínimo			1,09	0,116	7,69	158	280	26	41	18
media			2,01	0,136	7,87	243,1	520	31,5	47,7	20,8
D,S,			0,29	0,011	0,10	56,1	92,9	2,7	3,0	1,3
máximo	**2**	38	2,60	0,178	7,95	347	490	60	58	29
mínimo			1,33	0,077	7,43	214	240	16	27	13
media			1,85	0,121	7,70	281,2	370,5	33,6	46,0	20,4
D,S,			0,34	0,028	0,13	36,0	62,4	11,2	7,4	4,6
máximo	**3**	96	2,07	0,149	8,31	127	360	45	43	30
mínimo			1,12	0,064	7,10	22	140	34	31	20
media			1,41	0,092	7,83	65,0	224,1	38,3	37,1	24,5
D,S,			0,21	0,015	0,23	22,4	52,3	2,1	2,7	2,4
máximo	**4**	124	2,20	0,144	7,95	186,6	620	51	38	53
mínimo			0,76	0,054	6,09	17,8	215	16	23	25
media			1,36	0,088	7,14	67,9	350,6	35,7	29,8	34,2
D,S,			0,32	0,019	0,61	41,8	75,6	9,6	3,6	7,2
máximo	**5**	228	2,69	0,181	7,99	940	590	60	60	32
mínimo			0,82	0,063	7,25	450	260	13	27	12
media			1,77	0,117	7,69	648,6	398,9	34,2	45,8	20,0
D,S,			0,30	0,025	0,14	90,1	63,6	10,8	6,5	5,0
Max,T,			2,69	0,181	8,31	940	700	60	60	53
Min,T,			0,76	0,054	6,09	17,8	140	13	23	12
TOTAL:		536								

*Tabla 10.*II: Valores estadísticos sobre los datos edáficos de los campos de Madrid

rangos de valores que claramente no solapan con los dominios de las parcelas ubicadas en Nuevo Baztán. Otro ejemplo de tal heterogeneidad, es la existente en el contenido de mala hierbas comparado en la misma unidad, es decir en semillas por metro cuadrado. Por otra parte las parcelas que han sido sometidas a procesos de tratamiento tienen menores cantidades de mala hierbas, al menos de un orden de magnitud por debajo del resto. También se observan diferencias en pH que deben estar relacionadas con diferencias químicas en la composición. Así mismo, los diagramas texturales de cada parcela (figura 10.6) muestran que los campos de Nuevos Baztán tienen texturas ligeramente más arcillosas que las de Arganda.

Un análisis estadístico más extenso de los datos de avena loca fue realizado por el equipo de producción vegetal CCMA con los mismos datos, para estudiar la eficacia de las técnicas de agricultura de precisión y determinar la viabilidad económica en campos de cereal infestados con avena [BARROSO, 2004]. Una parte del estudio trata la determinación de la estabilidad espacial de poblaciones de *Avena sterilis L.*, mediante experimentos a largo plazo, utilizando dosis bajas de herbicida. Este estudio realizado en un periodo de cinco años concluye, a partir de los resultados del *test* estadístico de *Syrjala* [RUIZ-MAYA, 2000], que los rodales de avena son estables en cuanto a localización y densidad relativa, lo que aboga por la creación de mapas de infestación precisos en un determinado año como un modo adecuado de predecir la distribución futura de avena loca. Esta conclusión experimental, por otra parte, induce a pensar en una posible relación entre la abundancia de avena loca y las propiedades edáficas del terreno. Para establecer las posibles relaciones se realizaron diferentes estudios estadísticos descritos en [RUIZ ET AL., 2001] y [RUIZ ET AL., 2002]. Este estudio, un análisis estadístico avanzado no paramétrico, se realizó en dos etapas, del que se obtuvieron dos grupos -llamados en este contexto *conglomerados*- a partir de una técnica sencilla de *clustering*, el algoritmo *k-medias*, que agrupó las muestras utilizando una medida de similitud (la distancia euclidea) para definir el grado de semezanja en función de todas las variables. En la segunda parte del análisis, se determinó mediante el test estadístico de *Kruskal-Wallis*, la dependencia entre la avena y las categorías obtenidas en el paso anterior. En [RUIZ ET AL., 2001] se estudiaron las parcelas 1 y 3 con los datos recogidos en el muestreo del año 2001, es decir un total de 146 muestras. El trabajo concluye que ambos campos presentan mayor densidad de mala hierba allí donde la fertilidad disminuye y existe mayor cantidad de arena y menor de limo. Para los expertos, estos resultados llevan a pensar que en esos campos el cultivo llega a ahogar a la avena, aunque establecen la necesidad de ampliar el estudio a un mayor número de parcelas que permitan examinar el patrón de forma más general.

En el segundo estudio presentado en [RUIZ ET AL., 2002] la fase de *cluste-*

ring construye dos categorías para los muestreos 1 y 3 y tres para los muestreos 2 y 5. En este caso los autores concluyen que contenidos altos de materia orgánica y nitrógeno, están asociados con bajos contenidos en arena y altos contenidos de limo y arcilla. Además, excepto para el muestreo 5, los valores de materia orgánica, nitrógeno, fósforo y el potasio tienden a estar relacionados directamente entre sí, es decir, cuando una variable presenta valores bajos el resto de las variables también tiene valores bajos, y viceversa. Ahora bien, los autores encuentran algunas contradicciones en los resultados entre las categorías de los cuatro campos que impiden explicar completamente la distribución de la avena. Por ejemplo, establecen que los campos 1, 3 y 4 presentan altas densidades de avena en los grupos de muestras donde la materia orgánica, el nitrógeno, el fósforo y el potasio son bajos y además existe un alto contenido en arena. Por el contrario, el campo 5 presenta mayor abundancia en avena en aquellas áreas con niveles altos de materia orgánica, nitrógeno y potasio pero bajos de arena.

En conclusión, la baja significanción estadística de los resultados y la heterogeneidad presente en los datos, tanto edáficos como de avena de las parcelas, provocada por diferentes características estructurales del suelo además de por la diferente historia agrícola y antropogénica, impide establecer pautas generales a partir de estos estudios estadísticos. Con el fin de evitar que esta heterogeneidad en los datos provoque la ausencia de resultados comprensibles de los que se pueda establecer conclusiones concretas, tomamos la decisión de trabajar en términos relativos con estos mismos datos, en vez de absolutos, como se explicará posteriormente en la sección de preprocesamiento. De hecho, las parcelas que pueden presentar datos comprendidos en un único rango de valores en términos absolutos también presentan rodales, aunque sean de diferente densidad. La previa normalización de los datos permitirá, como veremos más adelante, determinar zonas de valores altos y bajos dentro de una misma parcela, y facilitará el estudio de las relaciones entre los rodales y las propiedades físicas y químicas de los suelos. En definitiva, tal y como se vio en la introducción, el uso de reglas combinado con la utilización de una semántica

relativa, provocada por esta etapa de normalización, permite la construcción de modelos complejos aproximados, formados por reglas individuales que cubren diferentes conjuntos de muestras y cuya unión permite describir todos los datos, es decir, todo el problema.

10.1.2. PREPARACIÓN

El problema de la avena en términos edáficos requiere una fase de preparación, compuesta por las tareas que se describen en este apartado.

Selección - El primer paso en la fase de preprocesamiento es la limpieza y selección de variables comunes entre las recogidas en todas las parcelas madrileñas. Se han seleccionado las siguientes variables: densidad de avena (determinada a partir del número de semillas o a partir del número de plantas por metro cuadrado), `materia orgánica (MO)`, `Nitrógeno (N)`, `Fósforo (P)`, `grado de acidez (pH)`, `Potasio (K)`, tanto por ciento de `arena`, tanto por ciento de `arcilla` y tanto por ciento de `limo`.

Existen dos maneras de incluir la textura en el estudio, bien utilizando el tipo de textura que según el triángulo de texturas determina cada muestra, o bien, utilizar cada una de las tres variables que la caracterizan -arena, arcilla y limo- por separado. En el caso de Madrid, se decide utilizar las variables independientemente, en primer lugar, por semejanza al procedimiento estadístico realizado por el equipo de CCMA, y segundo, el tipo de textura de las parcelas es diferente, y por lo tanto, es de esperar que no surjan relaciones entre las diferentes categorías producidas por el triángulo. Utilizar porcentajes, por el contrario, permite en primer lugar la normalización, facilita la comparación y en consecuencia, encontrar posibles relaciones texturales para todos los campos, es decir relaciones de carácter más general.

Normalización - El siguiente paso es la normalización para cada una de las parcelas. La información estadística resultante del análisis de los datos de Madrid (ver tabla 10.II) muestra que, aunque las propiedades de los suelos no tienen el mismo rango de valores, todas las parcelas presentan variaciones de mala hierba. Para reducir el efecto de factores como la historia del campo o

las características del entorno en la evolución de las malas hierbas en cada campo, los datos se homogeneizan utilizando una técnica de escalado-lineal (ecuación (9) de la página 111) que no introduce distorsión en la distribución de la variable al producir una relación lineal entre los valores de partida y los valores normalizados [PYLE, 1999]. El proceso de normalización seleccionado permite comparar los valores de las variables en términos relativos. Es decir podemos plantear el problema inicial como la búsqueda de una relación más o menos compleja entre los factores edáficos y lo proclive de ciertas zonas del campo a padecer o no plagas de avena loca. Así partimos de la hipótesis de que una parcela que ha sido tratada y que presenta algún rodal de avena loca tendrá en términos relativos, y en las zonas de los rodales, altos contenidos de mala hierba frente a los bajos o nulos contenidos del resto de la parcela. Esta transformación en los datos nos permitirá comparar parcelas entre sí.

Integración - Posteriormente, después de la normalización de los datos de cada parcela, se realiza la integración de las diferentes tablas en una única tabla de una misma base de datos. Es decir, con los datos de todos los muestreos se ha generado una tabla en la que sólo están almacenados los datos normalizados correspondientes a las 9 variables seleccionadas.

Categorización - Por último, los datos se someten a una fase conjunta de partición y etiquetado, es decir, de categorización. La categorización de 7 de las 9 variables se realiza utilizando dos umbrales de modo que los tres intervalos regulares. Para las otras dos variables se siguen filosofías diferentes. En el caso del pH la división es sólo en dos intervalos del mismo tamaño debido a que el dominio que presenta esta variable es pequeño y producir más de dos categorías, supondría trabajar con grupos con diferencias insignificantes. En el caso de la variable densidad de mala hierba, la clase objeto del aprendizaje, se ha optado por hacer una división que represente dos tipos de situaciones: a) la que no es de riesgo y que por tanto no requeriría aplicación de tratamiento y b) la que se considera como situación de riesgo y necesita la aplicación de tratamiento. Para esta variable se ha elegido un umbral que hace que los dos intervalos estén representados por el mismo número de datos. En el siguiente

punto trataremos con mayor profundidad la categorización llevada a cabo para esta variable. Por último, y dentro de esta etapa, a cada intervalo de los generados se le asigna una etiqueta con contenido semántico, es decir que aporta un significado al grupo de datos. En tabla 10.III aparecen, además de los umbrales utilizados para la creación de los diferentes intervalos, las etiquetas asignadas a cada intervalo.

nombre	intervalos normalizados	etiquetas	variable
pH	≤0,5	bajo	*grado de acidez*
	>0,5	alto	
MO	≤0,333	bajo	*materia orgánica*
	(0,333-0,666)	medio	
	≥0,666	alto	
N	≤0,333	bajo	nitrógeno
	(0,333-0,666)	medio	
	≥0,666	alto	
P	≤0,333	bajo	*fósforo*
	(0,333-0,666)	medio	
	≥0,666	alto	
K	≤0,333	bajo	*potasio*
	(0,333-0,666)	medio	
	≥0,666	alto	
Arena	≤0,333	bajo	*porcentaje de arena*
	(0,333-0,666)	medio	
	≥0,666	alto	
limo	≤0,333	bajo	porcentaje de limo
	(0,333-0,666)	medio	
	≥0,666	alto	
arcilla	≤0,333	bajo	*porcentaje de arcilla*
	(0,333-0,666)	medio	
	≥0,666	alto	
avena	≤0,2	Poca	*densidad de mala hierba*
	>0,2	Mucha	

Tabla 10.III: Umbrales para la categorización de cada variable de los datos de avena

Intervalos para la densidad de avena loca. Definición de clases - En cada campaña de muestreo se ha recogido bien información de semillas o bien información de plantas, pero en la mayoría de los casos no se han recogido ambos datos. Ahora bien la relación lineal entre la información generada por ambos métodos de medida hace que sea prácticamente indiferente, según los expertos, la elección que se realice para representar la densidad de mala hierba

en cada punto. Se elige el número de semillas en un punto, por que esta variable marca el número máximo de plantas de avena loca que podría haber en ese punto, es decir la situación de germinación total. En otras palabra, el número de plantas no podrá ser nunca mayor al número de semillas existentes.

El establecimiento de los umbrales para determinar si hay que considerar la infestación avena, a pesar de los trabajos referenciados que han sido explicados en la sección 9.3, no es un tema de fácil solución, ya que no siempre es el mismo para todos los campos, dependiendo de factores como el tipo de cultivo, la densidad de siembra, etc. En nuestro caso, los expertos recomiendan para la infestación de avena loca en estos campos, umbrales a partir de 20 plántulas por metro cuadrado, que como anteriormente se ha explicado, equivale a 200 semillas por metro cuadrado (D. Ruiz, *comunicación personal, 21 de enero de 2002*). Estos umbrales se establecen de forma orientativa teniendo en cuenta que estos valores podrían ser problemáticos.

Para aplicar el algoritmo de aprendizaje, es además importante que los dos conjuntos de ejemplos, positivos y negativos, estén equilibrados. Atendiendo a esta necesidad un umbral de 0,2 consigue una división del conjunto en dos clases con un número aproximado de ejemplos. Por otra parte, 0,2 supone que existe un 20 % de densidad total de mala hierba, lo que puede ser considerada una situación potencialmente de riesgo, ya que en términos relativos significa una elevada proporción de infestación frente al resto del campo. Por lo tanto, y además teniendo en cuenta la heterogeneidad existente entre las cuatro parcelas y tras el estudio de diferentes porcentajes, cuyos resultados en la división del conjunto se muestran en la tabla 10.IV, se decide emplear dicho valor (0,2) como umbral para obtener el conjuntos de ejemplos positivos y ejemplos negativos.

Este umbral determina que una densidad de avena loca superior al 20 % se considerá como densidad alta de mala hierba y por el contrario valores inferiores al 20 % serán considerados como densidades bajas.

Resumiendo, la variable de clase, densidad de avena y denominada de forma simplificada como **avena**, tendrá dos etiquetas **mucha** y **poca**, que representan los puntos muestreados con alta y baja densidad relativa de avena, respectiva-

Umbral	n positivos	(%)	n negativos	(%)
0,1	380	*70,90*	156	*29,10*
0,2	271	*50,56*	265	*49,44*
0,3	207	*38,62*	329	*61,38*
0,4	158	*29,48*	378	*70,52*
0,5	111	*20,71*	425	*79,29*
0,6	80	*14,93*	456	*85,07*
0,7	62	*11,57*	474	*88,43*
0,8	27	*5,04*	509	*94,96*
0,9	10	*1,87*	526	*98,13*

*Tabla 10.*IV*:* Umbrales posible para la partición de la tabla en ejemplos positivos y negativos

Figura 10.7: Gráfico que representa la división en ejemplos positivos y negativos de los datos de avena

mente.

Limpieza - La etapa de categorización junto con la etapa de segmentación de los registros de la tabla en dos clases diferentes, lleva a que aparezcan registros con el mismo contenido en ambos conjuntos de datos o clases. Para conseguir que la intersección de ambas clases sea vacía se lleva a cabo esta etapa de limpieza. Durante esta etapa, se procede a la determinación y eliminación de los registros duplicados que pertenecen a las dos clases simultáneamente. Tal y como se explicó en la descripción de la herramienta de preprocesamiento (sección 6.2), esta etapa se realiza a través en una secuencia de consultas de selección y borrado a la base de datos. Tras este proceso (véase el apéndice D), la tabla de datos resultante después de la limpieza pasa de tener 536 registros a tener 414, es decir se descarta un 33 % de los datos iniciales.

Conjunto de aprendizaje - Para la etapa de validación, se utiliza uno de los métodos más populares y descrito en el apartado 3.1.2, el método H, por el que se separan un 10 %, de los 414 registros iniciales, es decir 41 registros, que se eligen de modo aleatorio y que no se utilizaran en la etapa de extracción del modelo. Se determina utilizar este porcentaje de datos para la validación, fundamentalmente por dos razones. En primer lugar, en esta aplicación cuya finalidad es generar conocimiento, es prioritario contar con el mayor número de datos para el entrenamiento, y por lo tanto, reservar más registros para la validación reduciría significantemente el conjunto principal sobre el que se realiza el aprendizaje. Y en segundo lugar, este porcentaje o incluso menor es suficiente para determinar la capacidad predictiva de los modelos descubiertos. Por ejemplo, en el trabajo [Díaz-Diez & Morillas, 2004], los autores reservan un 5 % de los datos iniciales para la etapa de validación. La tabla del aprendizaje contiene el resto de registros, que son 373 y contiene los datos preparados para comenzar la etapa de aprendizaje o extracción del modelo.

Creación del archivo `fit` - El último paso, es crear el el archivo `fit` (explicado en la sección 7.2.2 en la página 137), a partir de la tabla obtenida en la fase anterior. El archivo para los datos etiquetados de Madrid se presenta entre los apéndices. Este archivo especifica que los antecedentes de las futuras reglas se componen de un máximo de 8 condiciones. El consecuente, sólo contiene la variable `avena` que define la clase. Todas las variables predictoras, incluida el pH, se representan mediante dos bits en el cromosoma para codificar todas las etiquetas relativas así como la posibilidad de que no tengan etiqueta, es decir, que no se activen y no aparezcan en el antecedente. Por otro lado, la clase se representa con un único bit, ya que siempre se activa, es decir, siempre aparece en el consecuente.

Las etiquetas de las variables puede ser, `bajo`, `alto` y `medio`, excepto el caso del `pH` y la clase, en las que sólo hay dos etiquetas posibles `bajo` o `alto` para el pH y `mucha` o `poca` para la clase.

10.2. Datos del campo de Barcelona

10.2.1. Descripción

El segundo grupo de muestras analizadas proceden del centro de Cataluña, concretamente de un campo de trigo (de invierno) de la comarca de L' Anoia en el término municipal de Calonge de Segarra en la provincia de Barcelona (figura 10.8)[2], cerca de Calaf.

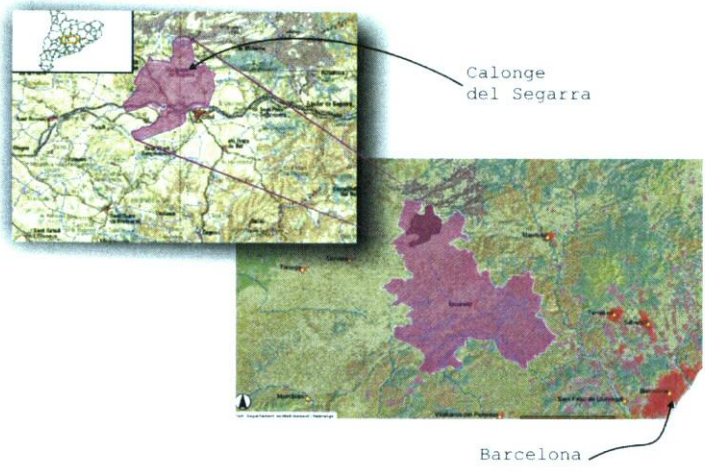


Figura 10.8: Localización del campo situado en la comarca de *L' Anoia* en el término municipal de Calonge de Segarra en la provincia de Barcelona

El conjunto de datos recogidos en el campo, que tiene un total de 2,25 hectáreas, está compuesto por 127 muestras de suelo y la información de 254 puntos relativa a las características biológicas, como biomasas de diferentes especies, tanto de cultivo como de malas hierbas. Las 127 muestras se localizan dentro de la malla de 254 puntos, tal y como se observa en la figura 10.9. Los datos relativos a la información biológica fueron recolectados durante las campañas 2001, 2002 y 2003, mientras que las muestras de suelo fueron recogidas en la campaña del 2001. El campo no fue tratado durante estos dos años

[2] Figura tomada de *http://sima.gencat.net/website/sima/viewer.htm*

por lo que se apreció una alta y fuerte infestación, natural y dominante, de *Lolium rigium Gaud.*, también conocido como *vallico* o *cizalla.* Además se observaron infestaciones de otras malas hierbas como la avena, cuya invasión fue importante, y diversas variedades de flores poco comunes en campos de cereal [ARDIACA, 2002]. Para el estudio que se presenta en esta memoria la existencia de avena loca es interesante ya que permite realizar un trabajo comparativo con los datos recogidos en Madrid.

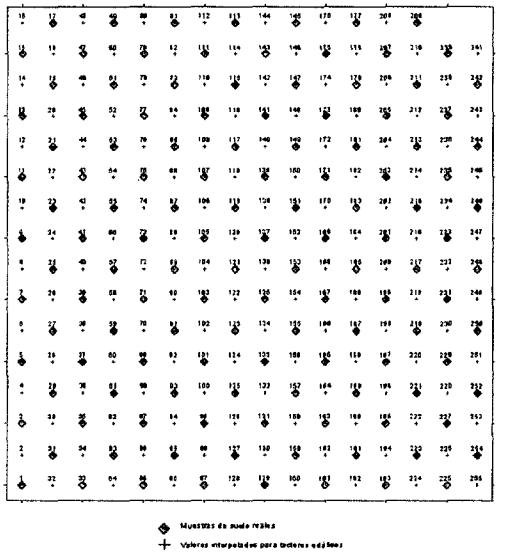


Figura 10.9: Croquis de la distribución de muestras de la parcela de Barcelona

El equipo del departamento de Biología Vegetal de la facultat de biología de la Universitat de Barcelona (UB) [3] recolectó esta información para analizar la distribución espacial y la búsqueda de posibles relaciones o patrones entre las perdidas de rendimiento de cultivo, determinados parámetros edáficos y/o ambientales, y la existencia de las principales especies infestantes, el lolium y la avena. La parcela de 150 × 150 metros cuadrados en la que se realiza el

[3] Equipo formado por el prof. Dr. Xavier Sans, prof. Dr. Jordi Carreras y Dr. Jordi Recasens

estudio, en cuanto a las características físicas, presenta una gran irregularidad topográfica (figura 10.10), llegando a tener desniveles máximos de 10,8 metros y pendientes de 20,9 %. La pendiente se orienta principalmente hacia el Norte-Noroeste. Para procesar y extraer la información sobre el desnivel topográfico de la parcela y la orientación, mediante un modelo digital del terreno (MDT) creado con la herramienta ARC-INFO© a partir de la información topográfica (en coordenadas relativas) obtenida de una estación total, aparato de elevada precisión referenciación basado en los antiguos distanciometros.

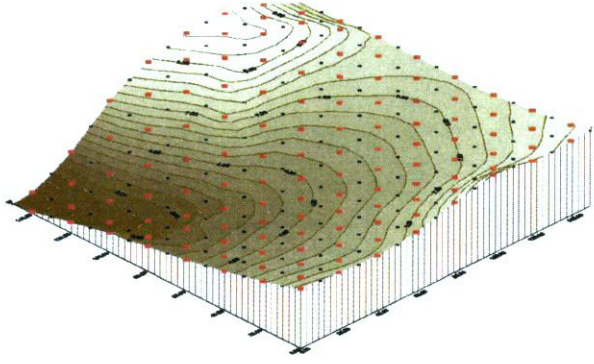


Figura 10.10: Topografía irregular de la parcela de Barcelona

La caracterización edáfica de la parcela de Barcelona, muestra un suelo calcáreo, presentando un 31 % de arena, un 16,2 % de limo grueso, un 32,1 % de limo fino y finalmente, un 20,7 % de arcillas. En consecuencia la textura es franca con tendencia a términos más arenosos como muestra la figura 10.11, donde se puede observar que la mayor parte de la nube de puntos que representan cada muestra se sitúa en el área de esta categoría. En cuanto a sus propiedades químicas, el suelo presenta unas cantidades medias-altas de materia orgánica (2,5 %), valores superiores al resto de los campos de la zona. Además, contenidos normales de nitrógeno (15 mgN-NO_3 Kg^{-1}), altos de fósforo (34 ppm) y entre bajos y medios de potasio (175 ppm). Finalmente,

un pH de 8.16 y una conductibidad eléctrica de 0,28 dS m^{-1}, determinando un suelo normal en cuanto a la salinidad según la escala de la *United States Salinity Laboratory de Riverside (EE UU)* [RHOADES ET AL., 1992].

En el apéndice B se encuentran los mapas generados de la interpolación con krijeado de las variables estudiadas, utilizando la aplicación SURFER v.6.

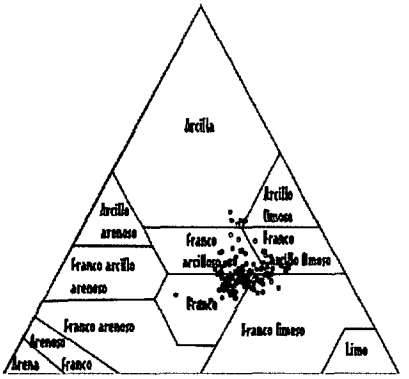


Figura 10.11: Distribución de todas las muestras de suelo en el triángulo de texturas

En este campo, se desarrollaron prácticas agrícolas normales y no se aplicaron tratamientos con herbicida, ya que el objetivo principal del grupo de la UB era estudiar el comportamiento de la infestación natural.

Para la recogida de muestras, datos como topografía o biomasas y otras variables que se resumen en la tabla 10.V de la parcela se plantearon dos distribuciones de puntos, es decir se utilizan dos mallas diferentes de muestreo, tal y como se observa en la figura 10.9. Una de las mallas consta de 254 puntos distribuidos en 16 filas de 15 puntos más una fila de 14, formando un enrejado de de 10 × 10 metros. La segunda malla se basa en la primera, es decir los puntos coinciden con la malla anterior, pero presenta un espaciado de 20 metros, lo que significa que se recogen sólo la mitad de los puntos, 127 puntos. El método elegido para tener garantía de un exacto y correcto remuestreo de los puntos seleccionados en sucesivas campañas de recogida fue el estaquillado. Además, al igual que para los datos de Madrid, fue utilizado

un receptor de localización geográfica GPS con alguna de las versiones de la herramienta de adquisición `Fiesta` para georeferenciar, en este caso con coordenadas absolutas, las muestras adquiridas. Por lo tanto, además de la información recopilada de cada punto, se tiene conocimiento de su localización espacial, información muy importante para la realización y creación de mapas.

En el apéndice A se expone una tabla con los principales valores de los parámetros estadísticos descriptivos de las características físico-químicas.

Factores biológicos	**Factores edáficos**	**Factores físicos**
banco de semillas lolium 2001	humedad estimada (kriging)	elevación
banco de semillas lolium 2002	capacidad retención agua	orientación
banco de semillas lolium 2003	conductividad eléctrica	pendiente
biomasa potencial de trigo 2001	Nitrógeno	radiación anual acum.
biomasa grano de 2001	Fósforo	
biomasa espigas de 2002	Potasio	
densidad de avena 2001	humedad	
densidad suelo	materia orgánica	
densidad de lolium 2001	textura (arena, limo y arcilla)	
densidad de lolium 2002	suelos tipificados	
densidad de lolium 2003		
biomasa de lolium 2001		
evaluación visual del lolium		
biomasa de avena 2001		
biomasa de avena 2002		
peso mil granos 2001		
peso mil granos 2002		
pies de trigo 2001		**Variables auxiliares**
presencia de paja 2001		coordenadas relativas
presencia de paja 2002		Id. puntos

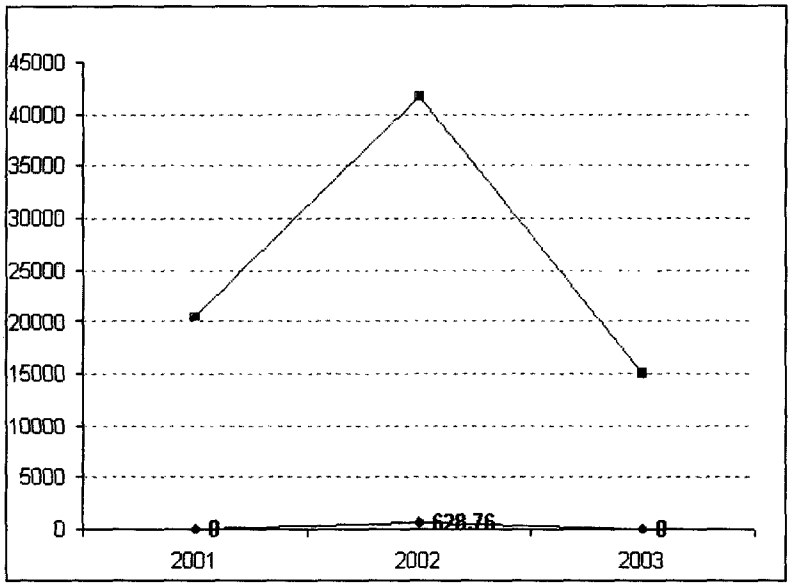
Tabla 10.v: Variables disponibles del campo de Barcenola

Conclusiones de los primeros análisis de caracterización

A partir de los datos suministrados, pueden analizarse la existencia de dos tipos de malas hierbas. Por un lado el lolium, la mala hierba dominante, y por otro lado la avena loca, la segunda especie más importante en grado de infestación.

Con el fin de conocer que objetivos se pueden alcanzar con los datos de partida, se realizaron estudios estadísticos descriptivos. Como resultado se vio

que el lolium presentaba una baja estabilidad espacio-temporal. Así mismo, tal como se ve en la gráfica 10.12, la densidad del lolium creció entre el año 2001 y 2002 de un intervalo de valores de (*0-20 592* semillas metro cuadrado) a un intervalo de valores de (*629-41 026* semillas metro cuadrado), mientras que la densidad descendió del segundo al tercer año, 2003, en el que el intervalo de valores para la densidad de lolium fue de (*0-14 933* semillas metro cuadrado).



	2001	2002	2003
min.	0	628.76	0
max	20591.89	41026.59	14933.05

Figura 10.12: (a) Gráfica y (b) valores numéricos que muestran la variación de la abundancia del lolium en tres años consecutivos 2001-2003

Por otra parte, los resultados de la correlación para las tres campañas son muy bajos e indican que los valores no varían conjuntamente, ni positiva ni negativamente, lo que lleva a suponer que no existe una permanencia temporal de los rodales al contrario de lo que pasaba con la avena loca.

correlación	2002	2003
2001	0.146	-0.092
2002	-	0.008

En la figura 10.13 se puede observar este mismo efecto de variación, en este caso espacialmente, donde es patente el desplazamiento o expansión de las áreas que presentan mayor densidad de infestación, en términos relativos. Es pues complejo buscar modelos que expresen la existencia de lolium en términos edáficos ya que en el caso del lolium y para los datos recogidos no existe evidencia de estabilidad espacial.

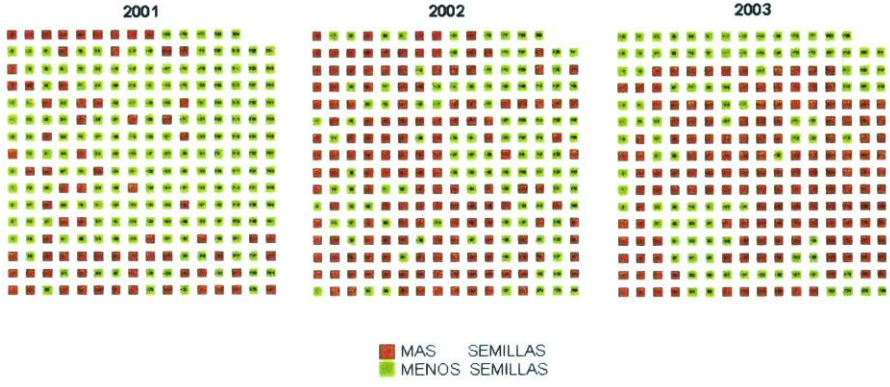


Figura 10.13: Movilidad de las áreas de mayor densidad de lolium para los tres años de campaña

Teniendo en cuenta los resultados de este análisis de correlación y si partimos de la premisa de que la existencia de la mala hierba podría estar condicionada por factores edáficos, se puede pensar que la presencia del lolium no puede determinarse a partir de la única campaña de edáficos que existe en la base de datos. Pero si de forma análoga a lo que sucede en muchos campos, los malherbólogos consideran a estos factores invariables desde un punto de vista práctico, entonces se podría llegar a deducir que la características ambientales asociadas a la existencia de esta mala hierba deben ser también variables. Entre

las características ambientales de un campo de cultivo, se pueden considerar principalmente la dinámica de otras especies, o la humedad que se acumule en todo el campo o sólo una parte en función de la climatología. A pesar de la situación que presenta los datos de distribución del lolium, estos se sometieron a la herramienta desarrollada para la extracción de modelos con el fin de buscar modelos más específicos y evaluar la metodología propuesta en aquellas muestras que no mostraron variación en la cantidad de lolium durante los tres años, es decir aquellas muestras que pertenecieron a una única categoría -siempre tuvieron mucha o siempre tuvieron poca mala hierba- en este periodo, y también usando los datos del cultivo (trigo) para conocer si esta variable puede describir la mencionada variación temporal.

En cuanto a la avena loca, el índice de correlación entre los datos de las campañas 2001 y 2002 es 0.67 lo que determina una suficiente asociación, permitiendo la búsqueda de asociaciones entre la abundancia de avena y la campaña de muestreo de edáficos del 2001 y, finalmente, lo que permite utilizar la metodología propuesta. En la figura 10.14 se puede ver la estabilidad espacio-temporal de la avena loca en el campo de Barcelona.

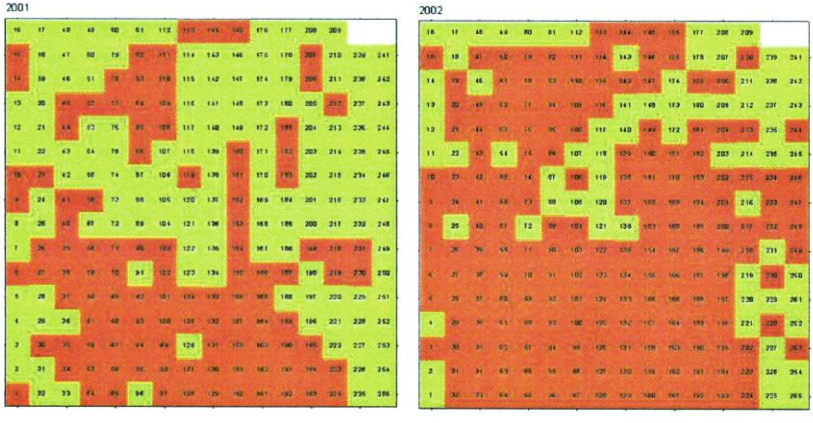


Figura 10.14: Dinámica de avena entre las campañas 2001-2002

En los datos de partida, existe información de la biomasa de avena para los

años 2001 y 2002 mientras que la densidad es únicamente del 2001. Lógicamente, ambas variables tienen una alta correlación (el coeficiente de correlación de Pearson es de un 80 %), tal y como se puede observar en la figura 10.15, ya que cuantifican información sobre la misma entidad, la densidad de avena. En consecuencia hemos decidido utilizar la biomasa como la variable para determina la clase a describir y en la validación de los modelos que se obtengan, porque esta información existe para los dos años.

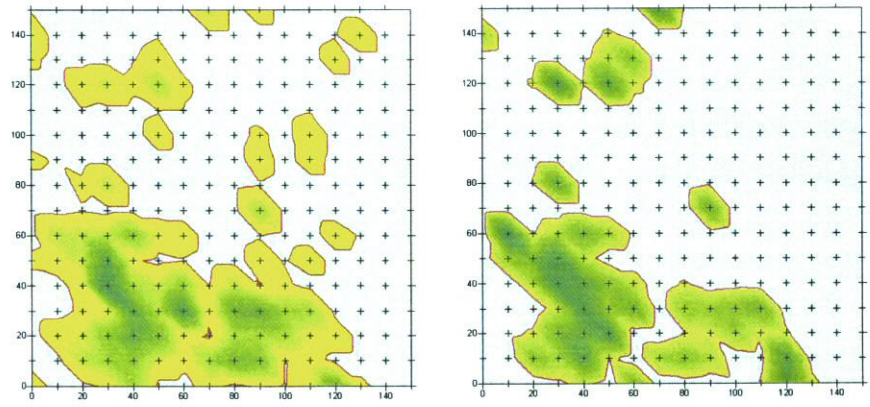


Figura 10.15: Mapas interpolados de la densidad y biomasa de avena para el año 2001 en la parcela de Barcelona

10.2.2. Preparación

Al igual que el preprocesamiento realizado sobre los datos de Madrid, el primer paso para los datos de Barcelona es la selección de los atributos que se usarán en la etapa de aprendizaje o extracción del modelo. Antes de comenzar a exponer la etapa de preparación conviene destacar que la estrategia utilizada con los datos de Barcelona es diferente porque la base de datos de partida es más rica, permitiendo considerar un mayor número variables en el modelo.

Otra diferencia, está relacionada con la transformación de los datos numéricos, ya que los datos proceden de un único campo, y por lo tanto, existe un

único rango de valores para cada variable. Consecuentemente, no existen problemas para la comparación de los datos, y entonces se evita la etapa de homogeneización generalizada de todas las variables, realizada mediante la normalización. Partiendo, entonces de valores absolutos es posible emplear umbrales determinados por expertos para la discretización, obteniendo categorías siempre comparables y, por supuesto, comprensibles para el usuario.

⋆ **Problema con datos de avena**: *Factores edáficos + factores biológicos*

El primer estudio con este conjunto de datos es la verificación de los resultados de Madrid, es decir que, como se explicará en el apartado siguiente de resultados, los modelos extraídos de los datos de avena de Madrid son validado con estos nuevos de datos de la misma mala hierba.

En el segundo estudio, el problema del avena del campo de Barcelona se estudia utilizando la información edáfica y biológica convenientemente preprocesados como veremos en lo que sigue.

Selección - El problema de la avena en el campo de Barcelona, se estudia utilizando la primera campaña de datos 2001, para predecir la existencia de avena en la siguiente campaña 2002. En otras palabras, los modelos obtenidos con los 254 registros del año 2001 se contrastan y validan como predictores con los datos recogidos de densidad de avena en la campaña 2002. Recordemos que las muestras reales de suelo son 127, pero para contar con un número superior de datos se toma la determinación de utilizar 254 registros. Teniendo en cuenta que la información de las variables biológicas fue recogida en estos 254 puntos, entonces para el aprendizaje se decide usar una tabla generada por el propio equipo de la UB que almacena esta misma información edáfica interpolada a partir de los 127 puntos. En definitiva, el conjunto de datos para el aprendizaje en este experimento consta, por un lado, de la información que proporcionan las variables físicas y edáficas para la campaña 2001 enumeradas en el apartado anterior, que son la humedad, la elevación, capacidad retención agua, orientación cardinal, conductividad eléctrica, humedad, pendiente, radiación anual acumulada, nitrógeno (N), fósforo (F), potasio (P), materia orgánica, textura del suelo a partir de arena, limo y arcilla, y

por otro lado, de la información biológica recolectada en cada campaña. En concreto las variables utilizadas y que se integran en una única tabla, son los siguientes: `biomasa avena 2001`, `biomasa trigo 2001`, `biomasa de paja del trigo 2001`, `biomasa de espigas del trigo 2001`, `biomasa de grano de trigo 2001`, `biomasa del lolium 2001`, `elevación`, `orientación`, `presencia de paja`, `pendiente`, `radiación`, `conductividad`), `pH`, `fósforo (P)`, `nitrógeno (N)`, `potasio (K)`, `materia orgánica (MO)`, `humedad`, `arcilla`, `limo`, `arena` y `tipo de suelo`.

Normalización - En este caso, la etapa de normalización se ha realizado atendiendo a diferentes objetivos de los establecido para los datos de Madrid. Recordemos que este nuevo conjunto de datos de aprendizaje pertenecen a una sola finca y a un único año, el 2001. Este hecho hace que en general todas las variables para el 2001 *a priori* sean comparables, sin necesidad de realizar una transformación, al menos en esta etapa de aprendizaje. Más adelante, explicaremos que durante la etapa de validación, variables como las biomasas no requieren una etapa de normalización *strictu sensu* pero que deben ser comparables para validar los resultados, hecho que se consigue directamente con la discretización.

Categorización - En el caso de los datos de Barcelona se han utilizado umbrales determinados por expertos, en concreto información del equipo y personal del laboratorio que analizó las muestras de este campo, para la definición de las categorías en todas las variables, con excepción de aquellas variables de las que no se conocían valores documentados en los que basar la creación de dichas categorías, entre otras, las biomasas, la humedad, la conductividad eléctrica o la radiación solar. Las tablas del apéndice C muestran los valores y correspondientes etiquetas de las categorías generadas en esta fase. Como se observa en estas tablas, las categorías del primer grupo de variables con umbrales conocidos, en concreto nitrógeno, fósforo, materia orgánica, potasio, pendiente y tipo de suelo, presentan las etiquetas que los expertos han establecido a partir de estos valores. Por ejemplo, para la variable `nitrógeno` un valor comprendido en el intervalo (15-30) tendría una etiqueta `normal-alto`, para (30-45) `alto`, (45-60) `muy alto` y finalmente, para $\geq$60 sería `excesivo`. El res-

to de las variables se discretizan utilizando intervalos regulares construyendo categorías con las etiquetas: `bajo`, `medio` y `alto`. Cualquiera de los métodos utilizados hace que todas las variables sean comparables, y empleados de la misma forma para la categorización del conjunto de validación permitirá una correcta contrastación de los resultados.

División de las clases - En el caso de la variable objetivo `biomasa avena`, la categorización que se ejecuta es más simple, ya que el umbral elegido es el valor 0, y se divide el dominio en dos grupos que determinan si existió biomasa de avena (`biomasa de avena` $= 0$) o si no existió (`biomasa de avena` $\neq 0$) en el año 2001. Durante el aprendizaje se describe la existencia de avena, y no la abundancia de avena como en el caso de Madrid. Este umbral es válido también para el grupo de validación por lo que esta variable será perfectamente comparable. Las etiquetas para cada grupo son `existe` y `no existe`, respectivamente.

Limpieza - Como ya se ha dicho repetidamente, para el estudio de los datos de avena de Barcelona partimos de 254 registros de datos recogidos en la campaña del año 2001. Tras las etapas descritas anteriormente de discretización e integración, en la fase de limpieza se eliminaron 12 registros con atributos idénticos y que aparecían en ambas clases simultáneamente. Esto supone que sólo se eliminó un 4,7 % del total. Este porcentaje es muy pequeño, sobre todo si lo comparamos con el 33 % eliminado en los datos de Madrid, y es debido principalmente a que existe un mayor número de variables, reduciendo la probabilidad de que exista intersección entre ambas clases.

Conjunto de aprendizaje - En el caso de la avena en los datos de Barcelona, tal y como se ha comentado, para realizar la experimentación utilizando una perspectiva ligeramente distinta y poder establecer otro tipo de conclusiones con la misma metodología, se toman los datos de la campaña del año siguiente 2002 como conjunto de validación para el modelo que se obtiene a partir de los datos de la campaña del año 2001. En esta experimentación, se utiliza información de un determinado año para intentar conocer la capacidad

predictiva de los modelos que se encuentren. Es decir, la principal diferencia con la experimentación con los datos de Madrid es el método de validación. En el caso de Madrid, se eligió el método H para el que se selecciona aleatoriamente un porcentaje de los datos, y en el caso de Barcelona, se eligió una variante del método de remuestreo, empleando nuevos datos.

El conjunto de entrenamiento, que corresponde al año 2001, está formado por 242 registros divididos en dos clases, una que contiene 103 registros etiquetada como `existe` o clase positiva y otra con 139 ejemplos etiquetada como `no existe` o clase negativa.

⋆ **Problema con datos de lolium**: *Factores Edáficos*

La etapa de preprocesamiento de datos con la especie *Lolium rigidum G.* se ha llevado a cabo en los pasos que se describen a continuación:

Selección - En la primera aproximación a la obtención de un modelo basado en reglas que describa la abundancia de *Lolium rigidum G.* en términos edáficos, se han incluido todos los datos de semillas relativos a todos los años de muestreo (2001, 2002 y 2003) y la única campaña de edáficos del año 2001, que como ya comentamos en secciones anteriores (apartado 10.1.1), se pueden asumir constantes. En cuanto las variables radiación solar acumulada y la pendiente también pueden ser consideradas estables, al menos en la práctica, porque fueron obtenidas a partir del MDT y éste no varía sustancialmente en esta escala de tiempo. Así pues, inicialmente se parte de tres tablas, una por campaña, de 127 registros cada una. Cada registro contiene, además de la identificación del punto muestreado y la información sobre la cantidad de semillas de lolium, la siguiente información:

`pH`, `fósforo`, `nitrógeno`, `potasio`, `materia orgánica`, `radiación solar acumulada`, `pendiente`.

Normalización - Se reescalaron y normalizaron los valores asociados al número de semillas según la ecuación 9 (página 111). En este experimento en concreto, el objetivo de la transformación de esta variable es poder trabajar en términos relativos y por tanto de un modo más adecuado al tipo de situación que tenemos, ya que los rangos en los que se mueven los niveles de infestación

son diferentes de un año a otro. Además, hablar en términos relativos significa que estamos interesados en conocer en que lugares del campo hay más mala hierba y en cuales menos, siendo los términos más o menos función de la cantidad total de mala hierba apreciada ese año para ese campo.

Integración - Las tres tablas de 127 registros se integraron en una única tabla de un total de 381 registros.

Categorización - Igual que en el experimento anterior, el resto de las variables, es decir la predictoras, fueron discretizadas y categorizadas utilizando los mismos umbrales establecidos por expertos, excepto la variable radiación solar de la que, como ya se ha comentado, no se conocían estos valores. De nuevo, para incluir esta variable se optó por utilizar tres intervalos regulares, generados a partir del mínimo y el máximo. Recordemos que las tablas A y B el apéndice C muestran estos umbrales y las etiquetas correspondientes a cada rango de estas variables.

División de las clases - Se establece a partir del valor del campo `lolium` los dos grupos de registros, ejemplos positivos y ejemplos negativos, necesarios para llevar a cabo la etapa de aprendizaje supervisado o generación de modelo basado en reglas.

A pesar de la normalización de los datos de lolium, la disparidad de la infestación en los distintos años, que son muy diferentes, requiere una especial atención en la búsqueda de un umbral para esta variable. Recordemos que el criterio para determinar el umbral es encontrar un valor que permita obtener dos clases proporcionales. Como se puede ver en la tabla 10.VI, el dominio de valores de cada año determina un umbral específico y distinto del resto: 0,046 para el 2001, 0,130 para el 2001 y 0,168 para el 2003.

Conjunto	2001	2002	2003
Umbral	0,046	0,130	0,168
(%)Ej.p	56 %	57 %	53 %
(%)Ej.n	44 %	43 %	47 %

*Tabla 10.*VI: Umbrales para la división de las clases positivos y negativos de los datos del lolium para los tres años

Partiendo de un conjunto integrado en una única tabla, la elección de un valor unitario como por ejemplo la media (0,105), provoca que las clases tenga la siguiente proporción: [positivos 70 %- negativos 30 %], un efecto no deseado, y plantea la necesidad de un análisis estadístico de este conjunto para conocer la distribución.

Este análisis revela la existencia probable de ruido, ya que en el extremo superior del dominio aparecen valores muy alejados del resto y con muy poca frecuencia. La figura 10.16 muestra un gráfico de cajas para los datos de cada año, donde se pueden apreciar los valores atípicos y extremos de la distribución que coinciden con los valores máximos y que pueden ser considerados como ruido y como tal son eliminados de la muestra.

La eliminación del ruido en este caso utiliza la técnica del reemplazamiento que se explicó en la sección de preprocesamiento de la sección 6.1, es decir aquellos valores extremos son reemplazados por un valor más adecuado, en este caso, los nuevos valores máximos. Sustituir esos valores máximos provoca la variación de la muestra, lo que obliga a normalizar e integrar otra vez los datos en un nuevo conjunto, obteniéndose un nuevo umbral para la división de las clases. Lógicamente, al disminuir los valores máximos de la muestra el umbral aumenta y el valor que mejor divide al conjunto, en ejemplos positivos y ejemplos negativos, es 0,247.

Limpieza- Después de la categorización es necesario conseguir que las clases sean disjuntas para lo cual se eliminan como en casos anteriores los ejemplos que son comunes. Este paso se ha realizado automáticamente, como en otros caso, con la herramienta de limpieza de `preparaDAT`, y el informe del proceso se presenta en el apéndice D. La fase de limpieza redujo en un 61 % la tabla quedando 163 ejemplos de los 381 registros iniciales. Recordemos que la eliminación del ruido por reemplazamiento, en nuestro caso, no ha significado la reducción del tamaño de la muestra.

Esta drástica disminución de ejemplos es coherente con lo descrito en la sección anterior referente a la variabilidad temporal observada de la infestación del lolium, así como la disponibilidad de una única campaña de muestreo de las

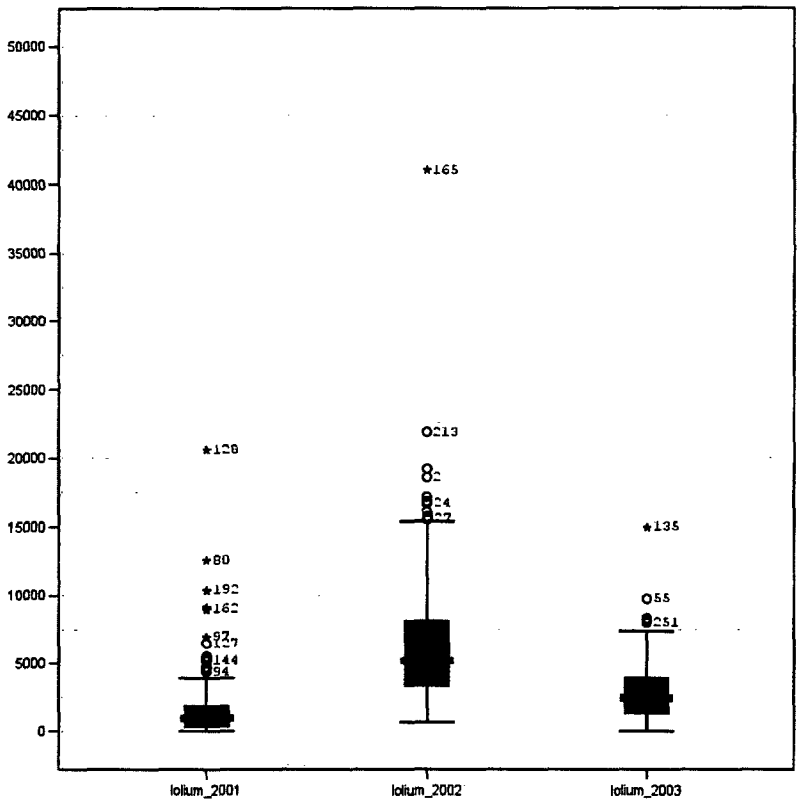


Figura 10.16: Gráfico de cajas de la variable lolium para cada año

propiedades edáficas. Estos dos hechos hacen que determinados puntos presenten en un año mucha cantidad de mala hierba y poca al año siguiente, mientras que las propiedades edáficas se mantienen. Estos registros son precisamente los que se eliminan en esta etapa, porque pertenecen a los dos conjuntos, positivos y negativos, presentando las mismas características. El 40 % de las instancias restantes, por lo tanto, corresponden a los puntos que han pertenecido a una misma clase durante los tres años. Sólo en estos puntos es posible presumir una relación entre la cantidad o existencia de mala hierba y los factores edáficos, tal y como plantean los expertos malherbólogos (sección 10.1.1).

Conjunto de aprendizaje - El conjunto de entrenamiento está formado

163 ejemplos, 76 positivos y 87 negativos. La proporción entre las respectivas clases, **mucha** y **poca**, se muestra gráficamente en la figura 10.17.

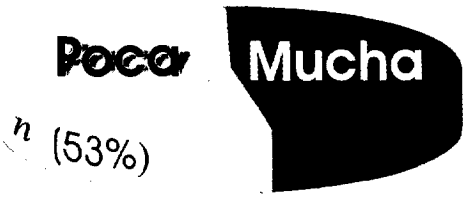


Figura 10.17: Gráfico que representa la división en ejemplos positivos y negativos de los datos de lolium utilizada en el primer estudio

⋆ **Problema con datos de lolium**: *Factores edáficos + factores biológicos*

En la segunda aproximación al estudio del lolium, se tienen en cuenta también los factores edáficos pero se añaden los factores biológicos, con el objetivo de introducir en el estudio nuevas características que puedan describir la variabilidad temporal, no incluida en el estudio anterior. En general, el preprocesamiento sigue los pasos que en el caso anterior, salvo pequeñas diferencias. Por esta razón, a continuación, sólo se describen las etapas que presentan alguna disparidad.

Selección - Se han seleccionado los datos de los 127 registros del año 2001 y los 127 del 2002. Como variables biológicas se han incorporado las variables relacionadas con el cultivo de trigo que miden la biomasa del cultivo a partir de diferentes procedimientos; espigas, grano o plantas, y un parámetro que indica la presencia o ausencia de paja. Además, se han agregado los atributos conductividad eléctrica y humedad, ambos relacionados con la salinidad y el contenido disponible el agua.

En resumen el estudio cuenta con las siguientes variables:

`Semillas lolium (2001 y 2002)`, `TRIGO`, `ESPIGAS`, `GRANO`, `AVENA`, `PAJA`,

conductividad, humedad, arena, limo, arcilla, pendiente, categoria* NESW (orientación), radiación* , pH* , N* , P* , K* , MO* y tipo* de* suelo*

Se han indicado con el símbolo (*) las variables que son comunes con el caso anterior.

Limpieza - Tras la categorización de los valores numéricos, algunos mediante umbrales regulares y otros siguiendo el criterio de los laboratorios, se realiza la limpieza de los datos, para obtener conjuntos de datos asociados a clases con intersección vacía. Esta etapa elimina un total de 16 registros, que suponen el 6,3 % de los 254 registros iniciales. Como ya se comentó, la existencia de un número mayor de variables puede hacer que la intersección entre las dos clases sea menor, y por lo tanto, la limpieza no provoca una reducción importante, como sucede en este caso.

División de las clases - Finalmente, se muestra la distribución de los registros de las dos clases mucha y poca lolium, utilizando el mismo umbral que en el caso anterior. Obsérvese que la clase mucha, o ejemplos positivos, consta de 99 instancias (42 % del total) mientras que la clase poca, o de ejemplos negativos, tiene 139 registros, lo que representa un 58 %.

Se puede ver que la proporción de elementos en ambos conjuntos es ligeramente diferente. Se ha optado por mantener el umbral en el 0,247 de modo que los modelos resultantes, a partir del conjunto de datos de cada una de las aproximaciones, puedan de alguna forma ser comparados.

Capítulo 11

EVALUACIÓN DE LA METODOLOGÍA PROPUESTA

11.1. EXPERIMENTACIÓN CON LA PROPUESTA

Una vez explicada la metodología y descritos tanto los datos recopilados como el preprocesamiento de los mismos, nos centramos en los resultados y la experimentación con la herramienta desarrollada. Recordemos que para la extracción de reglas utilizando lógica proposicional, descrita en el apartado 7, proponíamos la implementación de un algoritmo genético que realizase la búsqueda del mejor conjuntos de reglas. En la terminología de algoritmos genéticos cada regla está codificada en una parte del cromosoma que hemos denominado nucleosoma (sección 7.1.1). El sistema de extracción de reglas está diseñado de tal forma que permite configurar diferentes tipos de codificación del conjunto de reglas. Básicamente, la diferencia en las codificaciones radica en la forma en la que se utilizan los operadores lógicos y de comparación. Recordemos que los operadores lógicos OpL_{η} se usaban para conectar las reglas entre sí y los operadores lógicos OpL_{atri} para conectar las precondiciones de los antecedentes de las reglas. Además,los operadores de comparación unían cada variable a su valor para formar una precondición. La herramienta desarrollada permite una gran variedad de codificaciones, de hecho los operadores lógicos entre reglas pueden ser `OR`, `AND` o `OR/AND`; los operadores lógicos entre precondiciones pueden corresponderse con `OR`, `AND` o `OR/AND` y los operadores de comparación puedan valer (=) o (=/$\neq$). La combinación de estas posibilidades ofrece un total de 36 configuraciones diferentes, que se muestran en la tabla 11.1, donde se

destacan los casos que se analizan en el presente trabajo.

OpL_{atri}	OpL_{η}	OpL_{atri}	OpL_{η}
OpC (=)		OpC (=/≠)	
AND	AND	AND	AND
AND	OR	AND	OR
AND	OR/AND	AND	OR/AND
OR	AND	OR	AND
OR	OR	OR	OR
OR	OR/AND	OR	OR/AND
OR/AND	AND	OR/AND	AND
OR/AND	OR	OR/AND	OR
OR/AND	OR/AND	OR/AND	OR/AND

Tabla 11.1: Posibilidades de configuración del sistema

La herramienta permite también establecer, además de los conectivos que forman las reglas, la función de *fitness*, para evaluar la calidad del conjunto de reglas, y la complejidad del modelo, que se ajusta fijando el número de nucleosomas (η) que determina la longitud del cromosoma, que a su vez está relacionado con el número de reglas que tendrá el modelo.

Para el estudio de las malas hierbas se han seleccionado los tres casos de codificación más interesante, que se han denominado *caso A*, *caso B* y *caso C*. Para simplificar la descripción de cada uno de los casos, supongamos un individuo cuyo cromosoma consta de dos nucleosomas, es decir $\eta = 2$ con la siguiente estructura:

SI $[\eta_1]$ OpL_{η} $[\eta_2]$ ENTONCES C

- *Caso A (1 a η) - OpL_{atri}* OR/AND ; *OpL_{η}* OR/AND; *OpC =/≠*

  Este es el caso con un mayor número de grados de libertad. Los operadores lógicos (OpL_{atri} y OpL_{η}) pueden ser decodificados como OR o como AND y el operador de comparación OpC puede ser = o ≠. Según esto una regla para $\eta = 2$ podría tener la siguiente apariencia:

SI [A OpC a_1 OpL_{atri} B OpC b_1] OpL_{η} [A OpC a_2 OpL_{atri} B OpC b_2] ENTONCES C

Los operadores OR pueden aparecer entre atributos y entre nucleosomas al decodificar los cromosomas. Recordemos que un operador OR conecta en el mismo cromosoma antecedentes de diferentes reglas. Por lo que, con una codificación como la propuesta, es muy probable que los modelos que se generen estén formados por un gran número de reglas, fundamentalmente cuando los cromosomas son muy largos. El número máximo de reglas es igual al número de atributos que codifica todo el cromosoma. En este caso, se tendrán modelos complejos y difíciles de interpretar, siendo la etapa de aprendizaje muy lenta. Ahora bien, esta configuración puede ser eficaz para cromosomas de longitud pequeña y en algunos problemas es posible obtener soluciones sencillas con una calidad razonable por lo que conviene explorar esta posibilidad.

- *Caso B (1 a η)* - OpL_{atri} AND; OpL_{η} OR; $OpC = / \neq$

  Este segundo caso emplea el operador AND entre atributos y OR entre nucleosomas. Como ejemplo una regla tendría ahora la siguiente estructura:

  SI [A *OpC* a_1 AND B *OpC* b_1] OR [A *OpC* a_2 AND B *OpC* b_2] ENTONCES C

  donde OpL_{atri} = AND y OpL_{η} = OR, que se mantienen a lo largo de todo el aprendizaje, es decir en este caso tenemos un menor número de grado de libertad ya que el algoritmo siempre codifica un AND entre atributos y un OR entre nucleosomas, independientemente del bit que aparezca en el cromosoma. El número máximo de reglas es igual al número de nucleosomas. Por otra parte, se utilizan los operadores lógicos de comparación OpC = y $\neq$ lo que permite obtener relaciones más generales.

- *Caso C (1 a η)*- OpL_{atri} AND; OpL_{η} OR; OpC =

  Es el caso en el que no hay grados de libertad para los operadores, todos ellos se fijan de la siguiente forma: OpL_{atri}: AND, OpL_{η}: OR y OpC: =. Un ejemplo de regla en este caso sería la siguiente:

  SI [A = a_1 AND B = b_1] OR [A = a_2 AND B = b_2] ENTONCES C

Como puede observarse en el ejemplo, en este caso el OpC siempre toma el valor =. El resto de los parámetros siguen la misma configuración que el caso B, es decir OpL_{atri} = `AND` y OpL_{η} = `OR`, que se mantienen durante todo el proceso. En este caso el número máximo de reglas es igual al número de nucleosomas.

Finalmente, la siguiente tabla sintetiza las características de la configuración en cada caso.

Caso	OpL_{atri}	OpL_{η}	OpC
A	OR/AND	OR/AND	$=/\neq$
B	AND	OR	$=/\neq$
C	AND	OR	=

Descripción general de los experimentos

Para realizar los diferentes experimentos y verificar la bondad de la herramienta desarrollada así como de la metodología propuesta para la obtención de modelos descriptivos basados en reglas, se utilizaron los datos preprocesados con las técnicas explicadas en el capítulo 6.

Los experimentos de cada serie utilizan cromosomas de distinta longitud, aumentando el número de nucleosomas de uno en uno. La nomenclatura para los resultados que se presentan en la siguiente sección utiliza la letra de cada tipo de configuración {A, B o C}, seguida por un subíndice numérico que representa el número de nucleosomas que componen los cromosomas. Por ejemplo B_5 indicaría un experimento que utiliza la configuración tipo C y cromosomas con 5 nucleosomas.

Además en todos los experimentos se establecen del siguiente modo los parámetros básicos de la búsqueda genética: (1) para la selección se utiliza el método de la ruleta, (2) la tasa de cruce es de 0,5 y la tasa de mutación de está comprendida entre 0,004 y 0,01, valor que depende de la longitud de los cromosomas. Cromosomas más largos demandan una mayor tasa de mutación. Por otro lado, en aquellos experimentos en los que la longitud de los cromosomas ralentiza la etapa de aprendizaje, llegando a necesitar varios

días para producir alguna mejora del aprendizaje, se ha utilizado una semilla como situación de partida para la búsqueda y no una población generada aleatoriamente. Las semillas se usan para acelerar la búsqueda partiendo de una solución inicial más o menos buena, por ejemplo conjuntos de reglas generados previamente.

Finalmente, como se explicó en el apartado de algoritmos genéticos (sección 4.2) en todo proceso iterativo es necesario determinar el criterio de parada. La mayoría de los experimentos se han dado por concluidos cuando el valor de la *fitness* o función de calidad se estabilizaba, es decir se mantenía sin cambios durante un número considerable de generaciones. Este número depende de lo compleja que sea la solución que se busca así como la longitud, y en nuestro caso, el número de generaciones ha variado desde 100, para los cromosomas más cortos cuando buscamos las soluciones más sencillas, hasta la 30.000, en los cromosomas más largos para las soluciones más complejas. Aunque esta decisión puede ser complicada, sobre todo trabajando con cromosomas largos, cuya evolución es muy lenta y pueden ser necesario un número grande de iteraciones (generaciones) para obtener un pequeño aumento en la función de calidad. En resumen para cromosomas largos, situaciones mejorables pueden ser confundidas con situaciones de parada.

11.2. ANÁLISIS DE LOS RESULTADOS OBTENIDOS

En esta sección, se presentan los resultados de la experimentación, para los diferentes casos A, B y C propuestos en el apartado anterior, utilizando los datos de Madrid (`Serie I`) y de Barcelona (`Serie II` y `Serie III`), preparados según las etapas de preprocesamiento vistas en la sección 10.

- `Serie I` - Con los datos recogidos en las parcelas de Madrid se busca, además de evaluar el sistema desarrollado y la metodología de análisis de datos propuesta, obtener un modelo basado en reglas que proporcione conocimientos nuevos sobre el comportamiento de la *Avena sterilis L.*, de aquí en adelante avena, en cultivos de cereal de invierno. Algunos de los resultados que se expondrán en las siguientes apartados han sido publicados en [DÍAZ ET AL., 2003] y [DÍAZ ET AL., 2005].

- `Serie II` - Con los datos recogidos en Barcelona en el caso de la avena, se verifica los modelos obtenidos para las parcelas de Madrid con este nuevo conjunto de datos. A continuación, se realiza un proceso de aprendizaje para obtener un modelo basado en reglas que explique apropiadamente los datos de Barcelona.

- `Serie III` - En el caso de la infestación por *Lolium rigidum Gaud.*, para simplificar lolium, el objetivo es obtener el modelo más genérico del nivel de infestación en función de los parámetros edáficos, de modo análogo a la avena, y posteriormente buscar nuevos modelos más ricos teniendo en cuenta más información como por ejemplo la biomasa del cultivo de trigo (factores biológicos).

Por último, el capítulo presentará los resultados obtenidos con diversos algoritmos de una herramienta comercial llamada Clementine© 6.0.2, del grupo SPSS Inc[1] que usan la misma filosofía de nuestra propuesta. El objetivo de esta última fase es contrastar el funcionamiento del sistema desarrollado con

[1] http://www.spss.com/

un sistema comercial muy probado y conocido, lo que permitirá presentar los aspectos más novedosos y útiles de la herramienta desarrollada.

11.2.1. Experimentación con datos de Madrid

En la primera fase (`Serie I`) se realizaron un total de 30 experimentos diferentes sobre los datos de Madrid con el objetivo de verificar el sistema de extracción de conocimiento desarrollado. Los resultados de cada uno de los experimentos se evaluaron a partir de la calidad de los modelos generados expresada en los términos de número de reglas (r), *fitness* ($f_{(S\times E)}$), *exactitud* de clasificación ($Ex_{(\%)} = \frac{Vp+Vn}{Vp+Fn+Vn+Fp} \times 100$) y *confianza* ($Co_{(\%)} = \frac{Vp}{Vp+Fp} \times 100$) explicados en los capítulos 3 y 8, y cuyos valores se presentan en la tabla 11.II.

Para el caso A se realizaron seis experimentos diferentes $A_1, \ldots, A_6$, que indican que se utilizaron cromosomas con $\eta = 1, \ldots, \eta = 6$. Como se observa en la tabla, este caso presenta un número mayor de reglas que los experimentos tipo B y C comparando cromosomas de la misma longitud, es decir el mismo número de nucleosomas. Este funcionamiento es muy interesante, principalmente porque la posibilidad de encontrar un número mayor de reglas es debida a la capacidad de que los operadores entre atributos también puedan valer `OR` además de su valor convencional `AND`. De hecho, pueden descubrirse reglas con tantos antecedentes como atributos existan.

Observando los resultados en la tabla para cada uno de los casos, podemos ver el efecto que tiene la longitud del cromosoma en la calidad de la solución. Por ejemplo cuando el cromosoma es corto se obtienen modelos simples y de calidad, en algunos de los experimentos, aceptable. Cuando los cromosomas son largos las soluciones son más complejas y la calidad es mayor, aunque el aumento no es proporcional al número de reglas o complejidad del modelo. De hecho existe en cada uno de los casos un punto a partir del cual no se obtiene mayor calidad en la solución aumentando el tamaño del cromosoma. Por esta razón se ha llegado hasta $\eta = 6$ para el caso A, consiguiendo una exactitud del 86,60 % y un valor de confianza del 90,53 %. En los casos B y C se ha

$caso_\eta$	r	fitness S×E	V_p	F_n	V_n	F_p	Exactitud (%)	Confianza (%)
A_1	2	0,598	155	49	133	36	77,21	81,15
A_2	3	0,622	149	55	144	25	78,55	85,63
A_3	4	0,724	162	42	154	15	84,72	91,53
A_4	5	0,738	172	32	148	21	85,79	89,12
A_5	5	0,745	169	35	152	17	86,06	90,86
A_6	7	0,780	172	32	151	18	86,60	90,53
B_1	1	0,546	153	51	123	46	73,99	76,88
B_2	2	0,644	151	53	147	22	79,89	87,28
B_3	3	0,685	170	34	139	30	82,84	85,00
B_4	4	0,742	162	42	158	11	85,79	93,64
B_5	5	0,757	161	43	162	7	86,60	95,83
B_6	6	0,778	172	32	156	13	87,94	92,97
B_7	7	0,782	175	29	154	15	88,20	92,11
B_8	8	0,830	181	23	158	11	90,88	94,27
B_9	9	0,839	182	22	159	10	91,42	94,79
B_{10}	10	0,862	188	16	158	11	92,76	94,47
B_{11}	11	0,866	189	15	158	11	93,03	94,50
B_{12}	11	0,873	193	11	156	13	93,57	93,69
C_1	1	0,390	133	71	101	68	62,73	66,17
C_2	2	0,575	156	48	127	42	75,87	78,79
C_3	3	0,626	153	51	141	28	78,82	84,53
C_4	4	0,670	164	40	141	28	81,77	85,42
C_5	5	0,691	169	35	141	28	83,11	85,79
C_6	6	0,715	176	28	140	29	84,72	85,85
C_7	7	0,722	166	38	150	19	84,72	89,73
C_8	8	0,743	178	26	144	25	86,33	87,68
C_9	9	0,756	175	29	149	20	86,86	89,74
C_{10}	10	0,771	176	28	151	18	87,67	90,72
C_{11}	11	0,772	175	29	152	17	87,67	91,15
C_{12}	12	0,799	180	24	153	16	89,28	91,84

Tabla 11.II: Resultados de la serie I del conjunto de entrenamiento de Madrid

experimentado hasta $\eta = 12$ consiguiendo modelos con una exactitud de 93,57 y de 89,28 %.

Los valores de la tabla se representan gráficamente para facilitar un análisis más detallado. En primer lugar, el gráfico de la figura 11.1 muestra el número de reglas en función del número de nucleosomas utilizado para cada experimento. En todos los casos, la tendencia es creciente, es decir el número de reglas va aumentando al aumentar el número de nucleosomas. Además se observan algunas diferencias entre el caso A y los casos B y C, por ejemplo, los experimentos del caso A siempre presentan un número mayor de reglas, normalmente una más, que los casos B y C.

El caso B y C presentan gran similitud, que se explica por tener la misma configuración de los operadores lógicos, dando lugar a una gráfica con un tendencia creciente y lineal. Esta gráfica permite observar dos situaciones interesantes. La primera es en la curva del caso B, donde el número de reglas deja de crecer en el experimento con $\eta = 12$ ya que el modelo con mayor calidad que se consigue para este experimento tiene 11 reglas. Esto es debido a la desactivación completa de una de las reglas que sucede cuando se desactivan todas sus precondiciones (explicado en la sección 4.3). La segunda situación se da en el caso A, para el ensayo $\eta = 4$ que descubre un modelo de 5 reglas. En este caso, el número de reglas del modelo es consecuencia, como ya hemos indicado anteriormente, de la mayor probabilidad de encontrar operadores lógicos `OR` para el caso A.

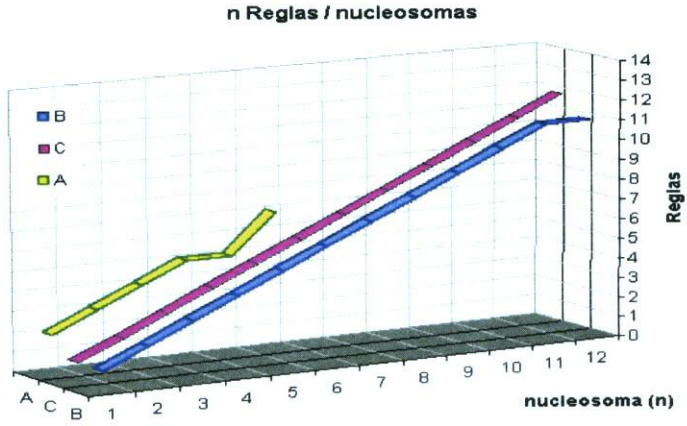


Figura 11.1: Gráficas del número de reglas en función del número de nucleosomas, es decir de la longitud del cromosoma

Además del número de reglas, en cada uno de los experimentos realizados varía también la cobertura de datos del modelo generado. Las figuras 11.2, 11.3 y 11.4 muestran esta variación con las proporciones de aciertos de clasificación (V_p y V_n) y de errores de clasificación (F_n y F_p) para cada caso.

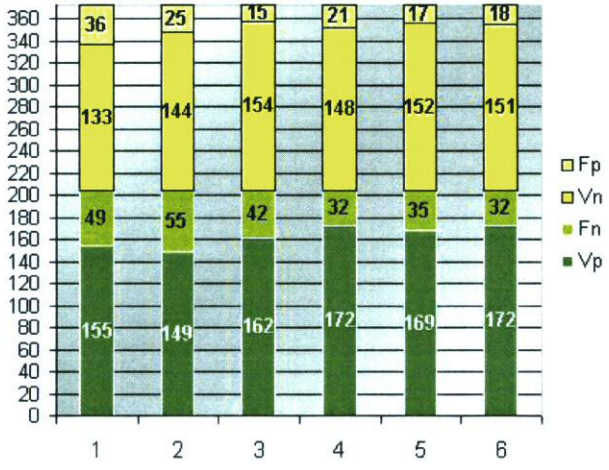


Figura 11.2: Evolución de la clasificación del caso A

En estas gráficas de nuevo se puede apreciar la tendencia a cubrir correctamente más ejemplos cuanto mayor es el número de nucleosomas, haciendo que el número de errores tienda a disminuir. Ahora bien, entre un experimento y el siguiente cada vez es menor la proporción de errores que se cubren lo que significa que cada vez es más difícil encontrar reglas que cubran grupos de registros no cubiertos en el experimento anterior. Aún con todo, el descenso continuado del número de errores hace que los modelos aumenten la calidad hacia la cobertura total, llegando a porcentajes de exactitud entorno al 93 %.

Las gráficas de la figura 11.5 muestran una comparativa de los casos A, B y C en cuanto a la evolución de la cobertura con el número de nucleosomas. En la figura 11.5a se muestra la evolución para los aciertos negativos (V_n) y en la figura 11.5b la evolución para los aciertos positivos (V_p). Las dos gráficas muestran que el caso B tiene mejor comportamiento en casi todos los experimentos con un mayor número de aciertos, positivos y negativos. Recordemos que el caso B permite la utilización del símbolo $\neq$, por lo que se puede deducir que su empleo permite una configuración menos restrictiva que la igualdad. La de-

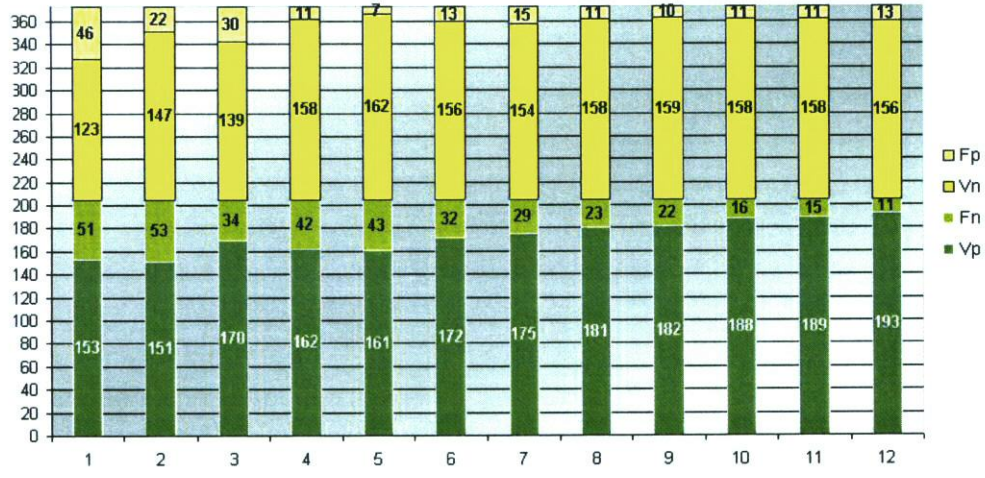


Figura 11.3: Evolución de la clasificación del caso B

sigualdad permite incluir ejemplos de varios grupos, mientras que la igualdad sólo describe a un único grupo.

Concretamente, la primera gráfica, para los verdaderos negativos (V_n), muestra que B siempre es mejor en su clasificación, excepto cuando $\eta = 3$, donde el valor desciende. Para los verdaderos negativos la curva del caso A presenta una tendencia casi constante comparada con los otros dos casos, empezando con valores de V_n altos que rápidamente se estabilizan. De hecho, pasa de ser la configuración con mejor clasificación de los V_n en $\eta = 1$ a generar siempre modelos de calidad intermedia para $\eta > 1$, si no se considera la situación excepcional que se produce en $\eta = 3$. En el extremo final de la gráfica puede observarse que el valor de V_n tienden al mismo valor en los casos B y C.

Por otro lado, la gráfica para verdaderos positivos (V_p) muestra menos diferencias entre los casos. En general el caso B obtiene modelos que clasifican mejor los ejemplos positivos, aunque existen excepciones para $\eta = \{2, 4, 5, 6\}$, donde la cobertura de la configuración B disminuye no presentando el mejor comportamiento frente a los casos A y C. Esta caída de los aciertos positivos

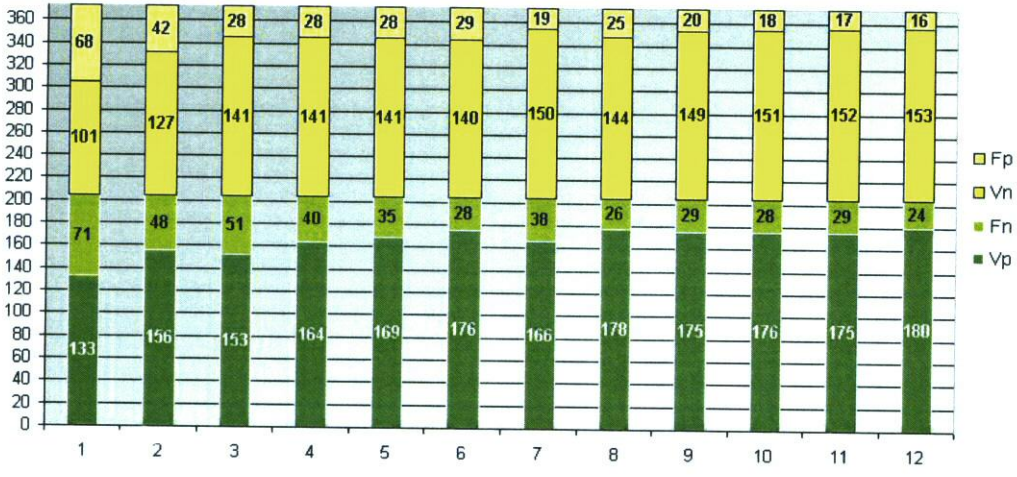
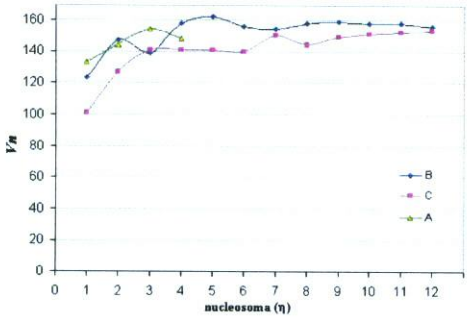
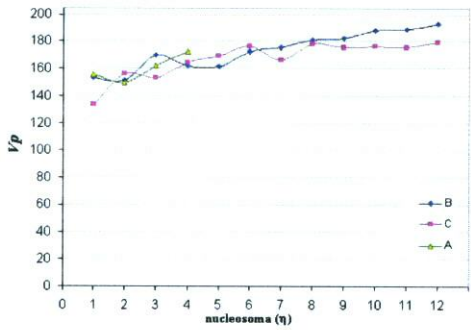


Figura 11.4: Evolución de la clasificación del caso C

de B coincide con un importante aumento de V_n en estos puntos de la otra gráfica. En cuanto al caso A, su curva de V_p presenta una tendencia similar en la gráfica de los V_p cubiertos que la que se observaba en la gráfica de los V_n, es decir para $\eta = 1$ se tiene la configuración con mayor cobertura y cuando $\eta > 1$ la cobertura de ejemplos positivos pasa a estar prácticamente entre los valores de los otros otros dos casos, aunque en este caso las variaciones no son tan acusadas. De estas dos gráficas, podríamos deducir que el uso del símbolo desigual ($\neq$) por ser menos restrictivo en su descripción en los casos A y B, repercute en la mejora de la clasificación de los elementos negativos, y también en la separación de ambas clases, frente al funcionamiento del caso C.

Por último, en estas gráficas también se observa lo visto en figuras anteriores, un aumento muy rápido de la correcta clasificación en los primeros experimentos que paulatinamente disminuye según aumenta el número de nucleosomas, sin perder la tendencia creciente en la calidad de los modelos. Esto lleva a pensar que la mayor parte de los registros se clasifican correctamente con modelos simples formados por pocos nucleosomas, y consecuentemente,



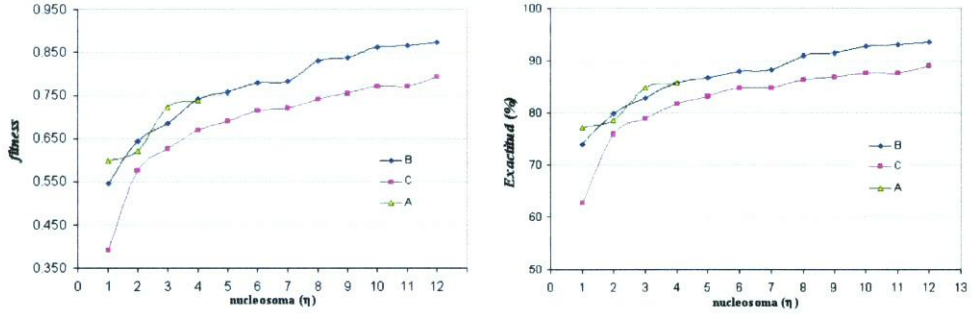


V_n V_p

Figura 11.5: Evolución de V_n y V_p para la comparación entre los casos A, B y C

con menos reglas. Los primeros conjuntos de reglas son bastante generales y poco a poco incorporan reglas de menor cobertura que por tanto clasifican grupos de registros más pequeños. En resumen, pasamos de clasificar pocos grupos de gran tamaño a cubrir un número mayor de grupos pequeños.

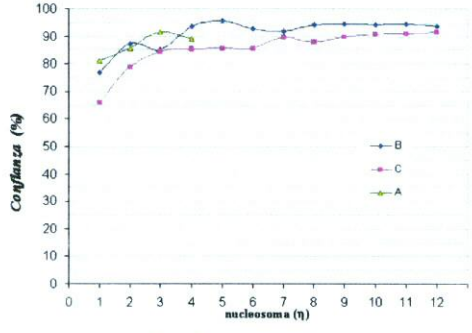
Las siguientes gráficas (figura 11.6) muestran la evolución de la calidad a través de los valores de la fitness y de la exactitud para modelos construidos a partir de diferentes números de nucleosomas. Puede observarse que ambas curvas son muy similares ya que las dos funciones están estrechamente relacionadas. La fitness fuerza la búsqueda de modelos capaces de explicar ejemplos positivos sin cubrir los ejemplos negativos, dando como resultado conjuntos de reglas que que clasifican con exactitud. La curva de evolución asociada a ambos estimadores, exactitud y fitness, presentan valores al comienzo relativamente altos que poco a poco aumentan con el número de nucleosomas presentando una curva con tendencia creciente que se va estabilizando para valores altos en el número de nucleosomas. Al igual que en casos anteriores, podemos decir que los modelos pasan de ser generales a más específicos y que el mayor aumento en la calidad se da en los primeros experimentos, es decir, que de 1 a 2 nucleosoma existe un incremento en la fitness y en la exactitud mayor que el conseguido con el paso de 11 a 12 nucleosomas.



Fitness Exactitud(%)

Figura 11.6: Fitness y la exactitud de la serie experimental

Por último, la gráfica de la confianza (figura 11.7), parámetro que representa la proporción de ejemplos que cumplen el antecedente del modelo y además pertenecen a la clase, por lo tanto al depender del número de F_p muestra una curva cuya forma es similar a la curva de V_n, ya que la confianza aumenta cuando disminuye F_p, es decir cuando aumenta V_n. La tendencia de las curvas es ligeramente creciente y parece estabilizarse en valores cercanos al 95 % para el caso B y al 90 % para C.



Confianza (%)

Figura 11.7: Gráfica de la evolución de la confianza

Como conclusión se puede establecer que según aumenta la longitud de los cromosomas los modelos son más complejos, debido a que aumenta el número de reglas. Además, los modelos son cada vez más exactos y presentan menor incertidumbre.

Además, del estudio de la calidad de los modelos resultantes con la metodología propuesta, es interesante realizar un análisis detallado de los conjuntos de reglas obtenidos y conocer el funcionamiento de cada regla dentro de cada grupo, de este modo se puede observar como se complementan las diferentes reglas para formar una solución global. En lo que sigue se muestran las reglas de los modelos más significativos y conjuntos de reglas asociados a modelos generados durante el proceso de búsqueda (modelos intermedios), lo que permite apreciar la evolución del procedimiento de búsqueda genética.

Para cada modelo se presentarán los valores para los parámetros de calidad: fitness (F), exactitud (Ex) y confianza (Co). Asimismo, para cada una de las reglas del modelo se darán los valores de la cobertura de ejemplos positivos (V_p) y la confianza (Co). Por último, los casos analizados se identificarán como hasta ahora. Es importante tener presente que el consecuente de las reglas siempre es el mismo "ENTONCES avena = mucha" y es el antecedente lo que diferencia unas reglas de otras; razón por la cual en el texto sólo aparece la parte de condiciones de las reglas.

Los modelos más sencillos A_1, B_1 y C_1 o generados a partir de cromosomas de mínima longitud (nucleosoma (η)= 1) se presentan a continuación. Puede observarse que los valores para los parámetros de calidad son relativamente altos si se tiene presente la simplicidad de los modelos, más aun si los comparamos con el resto de la serie.

- A_1 (F = 0,598 ; Ex = 77,2% ; Co = 81,15%)
  r1: MO $\neq$ menor AND P $\neq$ mayor AND arcilla $\neq$ menor (V_p 133; Co 83,7%)
  r2: limo = medio AND arena = menor (V_p 22; Co 61,1%)

- B_1 (F = 0,546 ; Ex = 74,0% ; Co = 76,9%)
  r1: MO $\neq$ menor AND limo $\neq$ mayor (V_p 153; Co 76,9%)

- C_1 (F = 0,390 ; Ex = 62,7% ; Co = 66,2%)
  r1: MO = medio (V_p 133; Co 66,2%)

Comparando estos modelos podemos destacar lo siguiente: (1) Los diferentes conjuntos de reglas no son contradictorios y representan en esencia la misma información. De hecho puede comprobarse que algunas precondiciones incluyen a otras, como por ejemplo (`MO` $\neq$ `menor`) engloba a (`MO = medio`) por ser una condición más relajada, o la condición (`limo` $\neq$ `mayor`) que abarca a la precondición (`limo = medio`). (2) Con una única regla del modelo se alcanza una cobertura de ejemplos positivos y una confianza relativamente altas, por ejemplo A_1:r_1 cubre 133 ejemplos positivos de 204 con una confianza del 83,7 % y B_1:r_1 cubre 153 de los 204 ejemplos con una confianza del 76,9 %. A la vista de estos resultados es razonable suponer que estas reglas son una descripción bastante general de los datos de entrada.

A continuación se muestran los modelos de mayor calidad generados a partir de una codificación del cromosoma con dos nucleosomas (A_2, B_2 y C_2).

- A_2 (F = 0,622 ; Ex = 78,6% ; Co = 85,6%)
  r1: P = menor AND arcilla = menor (V_p 26; *Co* 63,4%)
  r2: limo $\neq$ mayor AND arena = medio AND MO $\neq$ menor AND P $\neq$ mayor (V_p 88; *Co* 89,8%)
  r3: arcilla $\neq$ menor AND arena = mayor (V_p 36; *Co* 87,8%)

- B_2 (F = 0,644 ; Ex = 79,9% ; Co = 87,3%)
  r1: MO $\neq$ mayor AND P $\neq$ medio AND K $\neq$ medio AND arcilla $\neq$ mayor AND limo $\neq$ mayor (V_p 76; *Co* 86,4%)
  r2: MO $\neq$ menor AND P $\neq$ mayor AND arcilla $\neq$ menor AND limo $\neq$ mayor (V_p 118; *Co* 91,5%)

- C_2 (F = 0,575 ; Ex = 75,9% ; Co = 78,8%)
  r1: MO = medio AND limo = medio (V_p 71; *Co* 82,6%)
  r2: limo = menor (V_p 85; *Co* 75,9%)

Si comparamos estos modelos con los anteriores podemos observar que la cobertura de cada regla por separado ha disminuido. Estos nuevos modelos mantienen las precondiciones de las reglas anteriores y añaden nuevas condiciones haciendo que las reglas sean más específicas. Aunque la cobertura de cada regla considerada individualmente es menor los modelos son más exactos en la clasificación de los datos de entrada; obsérvese que hay porcentajes que rondan el 80 %.

Para el caso de tres nucleosomas se obtienen los modelos que se describen a continuación (A_3, B_3 y C_3):

- A_3 (F = 0,685 ; Ex = 84,7% ; Co = 91,5%)
  r1: pH $\neq$ mayor AND MO = mayor AND P $\neq$ mayor AND K $\neq$ medio (V_p 16; *Co* 94,1%)
  r2: arcilla = menor AND limo $\neq$ mayor AND arena = menor (V_p 21; *Co* 87,5%)
  r3: P $\neq$ mayor AND limo $\neq$ mayor AND arena $\neq$ menor AND pH = menor (V_p 57; *Co* 87,7%)
  r4: MO $\neq$ menor AND P $\neq$ mayor AND arcilla $\neq$ menor AND limo $\neq$ mayor AND arena $\neq$ menor (V_p 118; *Co* 94,4%)

- B_3 (F = 0,685 ; Ex = 82,8% ; Co = 85,0%)
  r1: MO = medio AND limo $\neq$ mayor (V_p 121; *Co* 81,8%)
  r2: N $\neq$ menor AND P = mayor AND arcilla = medio AND arena $\neq$ mayor (V_p18; *Co* 85,7%)
  r3: N $\neq$ menor AND MO = mayor AND P $\neq$ mayor AND arcilla $\neq$ menor (V_p38; *Co* 95,0%)

- C_3 (F = 0,626 ; Ex = 78,8% ; Co = 84,5%)
  r1: P = menor AND limo = menor (V_p 58; *Co* 86,6%)
  r2: MO = medio AND limo = medio (V_p 71; *Co* 82,6%)
  r3: P = medio AND arcilla = mayor (V_p 30; *Co* 88,2%)

Ahora las reglas de cada modelo cubren menos registros y tienen una mayor confianza, porque son más específicas. A cambio los modelos son más exactos tiene valores de confianza mayores que en los casos anteriores pero son más complejos.

Esta tendencia ligeramente creciente de la exactitud, confianza y especificidad de las reglas parciales se mantiene con el aumento de la longitud de los cromosomas. Para ilustrar de forma efectiva este comportamiento, a continuación se muestran y analizan los modelos obtenidos para cromosomas construidos con 8 nucleosomas.

- B_8 (F = 0,830 ; Ex = 90,9% ; Co = 85,0%)
  r1: N = menor AND P $\neq$ mayor AND K $\neq$ medio AND arcilla $\neq$ medio AND arena = mayor(V_p 15; *Co* 83,3%)
  r2: MO $\neq$ menor AND P = medio AND K $\neq$ menor AND limo $\neq$ mayor AND arena $\neq$ mayor (V_p 37; *Co* 94,9%)
  r3: arcilla = mayor AND limo $\neq$ mayor AND arena = mayor (V_p 33; *Co* 94,3%)
  r4: K $\neq$ mayor AND arcilla $\neq$ mayor AND limo = menor AND arena $\neq$ mayor (V_p 46; *Co* 97,9%)
  r5: pH = menor AND P $\neq$ mayor AND K $\neq$ mayor AND limo $\neq$ mayor AND arena $\neq$ menor (V_p 52; *Co* 89,7%)
  r6: MO $\neq$ medio AND P = medio AND K $\neq$ medio AND arcilla $\neq$ menor AND limo $\neq$ menor AND arena = menor (V_p 7; *Co* 100,0%)
  r7: MO = medio AND P $\neq$ medio AND K $\neq$ medio AND arcilla $\neq$ mayor AND limo = medio(V_p 37;

Co 92,5%)
r8: pH = menor AND N = mayor AND P ≠ mayor AND arcilla ≠ menor (V_p 25; *Co* 100,0%)

- C_8(F = 0,743) ; Ex = 86,1% ; Co = 89,2%)
  r1:MO = medio AND arcilla = menor AND limo = medio(V_p 23; *Co* 85,2%)
  r2: P = menor AND limo = menor(V_p 58; *Co* 86,6%)
  r3: MO = medio AND P = medio AND K = medio AND arcilla = medio (V_p 21; *Co* 91,3%)
  r4: MO = medio AND P = menor AND K = menor AND limo = medio (V_p 31; *Co* 93,9%)
  r5: P = medio AND arcilla = mayor (V_p 30; *Co* 88,2%)
  r6: MO = mayor AND arcilla = medio (V_p 17; *Co* 94,4%)
  r7: N = medio AND P = mayor AND K = mayor AND arena = medio (V_p 3; *Co* 75,0%)
  r8: pH = menor AND N = mayor AND P = menor AND arcilla = mayor (V_p 8; *Co* 100,0%)

Por último, se muestran los dos modelos más exactos generados con la metodología desarrollada. En este caso el cromosoma consta de 12 nucleosomas.

- B_{12} (F = 0,873 ; Ex = 93,6% ; Co = 93,7%)
  r1: N = menor AND MO = medio AND K = menor AND arcilla ≠ mayor AND limo ≠ medio AND arena = mayor (V_p 5; *Co* 100%)
  r2: N ≠ mayor AND MO = medio AND P ≠ menor AND arcilla = menor AND limo = medio (V_p 16; *Co* 100%)
  r3: N = mayor AND MO = mayor AND P ≠ mayor AND K ≠ medio AND arcilla ≠ menor AND limo ≠ medio AND arena ≠ medio (V_p 11; *Co* 100%)
  r4: pH ≠ mayor AND MO ≠ medio AND P = menor AND limo ≠ mayor (V_p 24; *Co* 88,9%)
  r5: N ≠ menor AND K ≠ mayor AND arcilla ≠ mayor AND limo ≠ mayor AND arena = menor (V_p 13; *Co* 100%)
  r6: N ≠ menor AND MO ≠ mayor AND P = mayor AND K = mayor AND limo ≠ mayor (V_p 4; *Co* 100%)
  r7: pH ≠ menor AND MO = menor AND P ≠ menor AND K ≠ medio AND limo = medio AND arena = menor (V_p 3; *Co* 100%)
  r8: N = menor AND MO = menor AND P ≠ mayor AND K = menor AND arcilla = menor AND arena ≠ medio (V_p 8; *Co* 80%)
  r9: MO ≠ menor AND P ≠ mayor AND arcilla ≠ menor AND limo ≠ mayor AND arena ≠ menor(V_p 118; *Co* 94,4%)
  r10: pH ≠ mayor AND N ≠ medio AND MO = medio AND P ≠ mayor AND arcilla ≠ medio AND arena = menor (V_p 6; *Co* 85,7%)
  r11: N = medio AND MO = medio AND K ≠ mayor AND arcilla ≠ menor AND limo ≠ medio AND arena ≠ mayor (V_p 16; *Co* 88,9%)

- C_{12} (F = 0,799 ; Ex = 89,3% ; Co = 91,8%)
  r1: arcilla = medio AND limo = menor AND arena = medio (V_p 41; *Co* 100%)
  r2: MO = medio AND arcilla = menor AND limo = medio AND arena = menor (V_p 15; *Co* 100%)
  r3: N = mayor AND MO = mayor AND arcilla = medio (V_p 13; *Co* 100%)
  r4: N = menor AND P = menor AND arena = mayor (V_p 14; *Co* 82,4%)
  r5: pH = mayor AND N = menor AND P = medio AND arena = mayor (V_p 6; *Co* 100%)

r6: pH = menor AND MO = menor AND P = menor AND limo = medio (V_p 8; Co 72,7%)
r7: MO = medio AND limo = medio AND arena = medio (V_p 46; Co 86,8%)
r8: arcilla = mayor AND arena = mayor (V_p 33; Co 89,2%)
r9: pH = menor AND P = medio AND arcilla = mayor (V_p 16; Co 94,1%)
r10: MO = medio AND arcilla = menor AND limo = menor AND arena = menor (V_p 5; Co 100%)
r11:N = medio AND MO = medio AND K = medio AND arcilla = medio (V_p 15; Co 83,3%)
r12: N = medio AND MO = medio AND P = menor AND K = medio AND arcilla = mayor AND limo = mayor (V_p 2; Co 100%)

De forma general podemos concluir que la cobertura individual de cada regla ha ido descendiendo a medida que el modelo presentaba más reglas. Incluso, aparecen reglas cuya cobertura se limita a tres o dos registros, como es el caso de C_8:r_7 o el de C_{12}:r_{12}. Las reglas más genéricas se obtienen por lo general en los modelos más sencillos o, lo que es lo mismo, generados a partir de cromosomas más cortos.

Por consiguiente, la decisión sobre que tipo de modelo generar a partir de unos datos de entrada es dependiente del objetivo que se persiga en cada caso. Así, si lo que se desea es inducir conocimiento sobre las características esenciales del grupo de datos, entonces siguiendo el principio de Ockham, expuesto en el apartado 3.1.1, los modelos más sencillos y en concreto las reglas con mayor cobertura son la respuesta. Si por el contrario el análisis tiene por objetivo construir un clasificador de alta exactitud de los datos de entrada, entonces posiblemente los modelos más complejos y con reglas más específicas serán los que mejor se adecuen a ese caso.

Una vez obtenidos los modelos la siguiente etapa es la de validación de los resultados. La tabla 11.III muestra los valores de los parámetros de calidad de todos los modelos obtenidos. La evaluación se realiza con la clasificación del conjunto de datos de validación, que recordemos son el 10 % porcentaje de registros seleccionados aleatoriamente y no utilizados en la fase de aprendizaje.

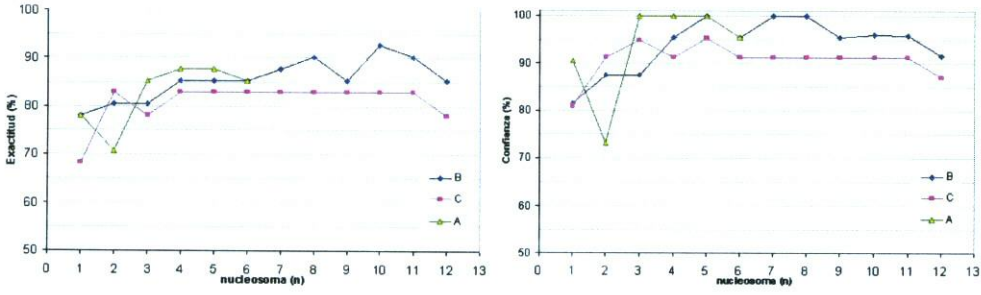
Los parámetros exactitud y confianza resultantes de la validación se muestran en las gráficas de la figura 11.8. Aunque en general los valores son altos y similares a los obtenidos con el grupo de datos de entrenamiento, se pueden observar algunas diferencias. Las funciones no muestran una tendencia clara de crecimiento con el número de nucleosomas e incluso en algunos casos la

$Caso_\eta$	V_p	F_n	V_n	F_p	Exactitud (%)	Confianza (%)
A_1	19	7	13	2	78,05	90,48
A_2	22	4	7	8	70,73	73,33
A_3	20	6	15	0	85,37	100,00
A_4	21	5	15	0	87,80	100,00
A_5	21	5	15	0	87,80	100,00
A_6	21	5	14	1	85,37	95,45
B_1	22	4	10	5	78,05	81,48
B_2	21	5	12	3	80,49	87,50
B_3	21	5	12	3	80,49	87,50
B_4	21	5	14	1	85,37	95,45
B_5	20	6	15	0	85,37	100,00
B_6	21	5	14	1	85,37	95,45
B_7	21	5	15	0	87,80	100,00
B_8	22	4	15	0	90,24	100,00
B_9	21	2	14	1	85,37	95,45
B_{10}	24	2	14	1	92,68	96,00
B_{11}	23	3	14	1	90,24	95,83
B_{12}	22	4	13	2	85,37	91,67
C_1	17	9	11	4	68,29	80,95
C_2	21	5	13	2	82,93	91,30
C_3	18	8	14	1	78,05	94,74
C_4	21	5	13	2	82,93	91,30
C_5	20	6	14	1	82,93	95,24
C_6	21	5	13	2	82,93	91,30
C_7	21	5	13	2	82,93	91,30
C_8	21	5	13	2	82,93	91,30
C_9	21	5	13	2	82,93	91,30
C_{10}	21	5	13	2	82,93	91,30
C_{11}	21	5	13	2	82,93	91,30
C_{12}	20	6	12	3	78,05	86,96

Tabla 11.III: Valores de los parámetros de calidad en la etapa de validación para los datos de avena de Madrid

exactitud y la confianza disminuye al aumentar la complejidad del modelo. Este hecho tiene su explicación en la alta especificidad de las reglas de los modelos más complejos, en otras palabras cada regla cubre un pequeño número de datos de entrenamiento. Por consiguiente, parece razonable que los modelos más específicos presenten peor calidad cuando se les somete a la clasificación de un conjunto nuevo de datos. En contraposición, los modelos más simples presentan exactitudes similares en la clasificación de nuevos ejemplos y datos de entrenamiento, característica asociada al carácter más genérico de estos modelos. En definitiva podemos concluir que si el conjunto de datos representa bien al problema, es posible que un modelo complejo y exacto sea también

buen clasificador de elementos desconocidos, en caso contrario modelos más simples y menos exactos pero con reglas de gran cobertura pueden resultar mejores clasificadores de elementos desconocidos.



Exactitud(%) Confianza(%)

Figura 11.8: Evolución del valor de la exactitud y la confianza en la validación de los modelos resultantes para las parcelas de Madrid

En resumen la elección del mejor modelo se realiza en función de los objetivos de cada problema. Para el caso de la avena y del lolium el objetivo que se persigue con el análisis es extraer conocimiento nuevo sobre el comportamiento de estas especies en cultivos de cereal de invierno por lo que se preferirán modelos genéricos frente a modelos más precisos construidos con reglas más específicas.

Atendiendo a los objetivos de esta investigación los mejores modelos, aquellos que permitan extraer conocimiento, serán los más genéricos y simples, como son A_1, B_1, C_1, B_2 o C_2 que presentan valores altos de calidad y como máximo dos reglas. De este grupo, destacamos A_1 porque tiene mayores valores de exactitud y confianza que B_1 o C_1, un valor de confianza mayor en la clasificación del conjunto de validación y además un cromosoma de menor longitud, ya que está formado por un sólo nucleosoma frente a B_2 o C_2 que están formados por dos nucleosomas.

Visualización y evaluación espacial de resultados

En este apartado se presentan los mapas de infestación estimada por el modelo más genérico A_1 y por el modelo más exacto B_{12} y se contrastan con

$Caso_\eta$	Entrenamiento		Validación	
	Ex (%)	Co(%)	Ex (%)	Co(%)
A_1	77,21	81,15	78,05	90,48
A_2	78,55	85,63	70,73	73,33
A_3	84,72	91,53	85,37	100,00
A_4	85,79	89,12	87,80	100,00
A_5	86,06	90,86	87,80	100,00
A_6	86,60	90,53	85,37	95,45
B_1	73,99	76,88	78,05	81,48
B_2	79,89	87,28	80,49	87,50
B_3	82,84	85,00	80,49	87,50
B_4	85,79	93,64	85,37	95,45
B_5	86,60	95,83	85,37	100,00
B_6	87,94	92,97	85,37	95,45
B_7	88,20	92,11	87,80	100,00
B_8	90,88	94,27	90,24	100,00
B_9	91,42	94,79	85,37	95,45
B_{10}	92,76	94,47	92,68	96,00
B_{11}	93,03	94,50	90,24	95,83
B_{12}	93,57	93,69	85,37	91,67
C_1	62,73	66,17	68,29	80,95
C_2	75,87	78,79	82,93	91,30
C_3	78,82	84,53	78,05	94,74
C_4	81,77	85,42	82,93	91,30
C_5	83,11	85,79	82,93	95,24
C_6	84,72	85,85	82,93	91,30
C_7	84,72	89,73	82,93	91,30
C_8	86,33	87,68	82,93	91,30
C_9	86,86	89,74	82,93	91,30
C_{10}	87,67	90,72	82,93	91,3
C_{11}	87,67	91,15	82,93	91,3
C_{12}	89,28	91,84	78,05	86,96

Tabla 11.IV: Comparativa de los valores de exactitud (Ex%) y confianza (Co%) en la etapa de entrenamiento y validación para los datos de avena de Madrid

los mapas de distribución real de avena, y se utilizan todos los datos (los datos del conjunto de entrenamiento, los del conjunto de validación y los que fueron eliminados en la etapa de limpieza), para representar todas las muestras recogidas y observar el comportamiento de los modelos. En todas las figuras se representan los puntos que pertenecen a la clase `mucha` mala hierba (ejemplos positivos) con un círculo negro y los que pertenecen a la clase `poca` mala hierba (ejemplos negativos) con un círculo en blanco.

En la figura 11.9 se muestran las distribuciones estimadas del modelo A_1 en contraposición con las distribuciones reales para todas las campañas realizadas

en las parcelas de Madrid explicadas en la sección 10.1.

Los valores de calidad de este modelo para cada una de las parcelas fueron:

- n^0 1: Exactitud 76,0 % Confianza 81,8 %
- n^0 2: Exactitud 63,2 % Confianza 41,2 %
- n^0 3: Exactitud 64,6 % Confianza 23,8 %
- n^0 4: Exactitud 60,5 % Confianza 47,8 %
- n^0 5: Exactitud 89,6 % Confianza 83,5 %

En estos valores se observa que el modelo más genérico describe un porcentaje alto de exactitud y de confianza los datos recogidos en las parcelas 1 y 5, mientras que para el resto de las parcelas estos valores de calidad no son tan altos.

En las figuras se puede observar que el modelo A_1 se ajusta mejor en las parcelas de Arganda y esto es debido a que este modelo cubre mejor los datos de la campaña 5 que representan el 42,54 % de los datos, casi la mitad.

Como complemento a la distribución espacial estimada por el modelo para cada parcela, en la tabla 11.v se muestran los valores de calidad distribuidos por parcelas para cada regla. La tabla indica el número de ejemplos positivos cubiertos (V_p), la confianza (Co) y la proporción de ejemplos positivos cubiertos sobre el total de ejemplos positivos ($\%+ = \frac{V_p}{p}$). Los valores de la tabla sugieren que la regla $A_1{:}r_1$ describe mejor los datos de las campañas 1, 2 y 5 -realizadas en Arganda del Rey- mientras que las campañas 3 y 4 -realizadas en Nuevo Baztán- se describen mejor con la regla $A_1{:}r_2$, aunque esta última no funciona del todo mal para la campaña 5. El hecho de que cada regla se ajuste mejor a diferentes parcelas coincide con las alguna de las ideas que, como se expuso en el apartado 10.1.1 de descripción de los datos de madrid, el equipo de CCMA intuían como conclusiones de sus estudios estadísticos sobre las distintas características edáficas de las parcelas y su relación con la aparición de la avena. Entonces, de nuestros resultados se puede concluir que las diferentes parcelas tienen modelos diferentes para la avena.

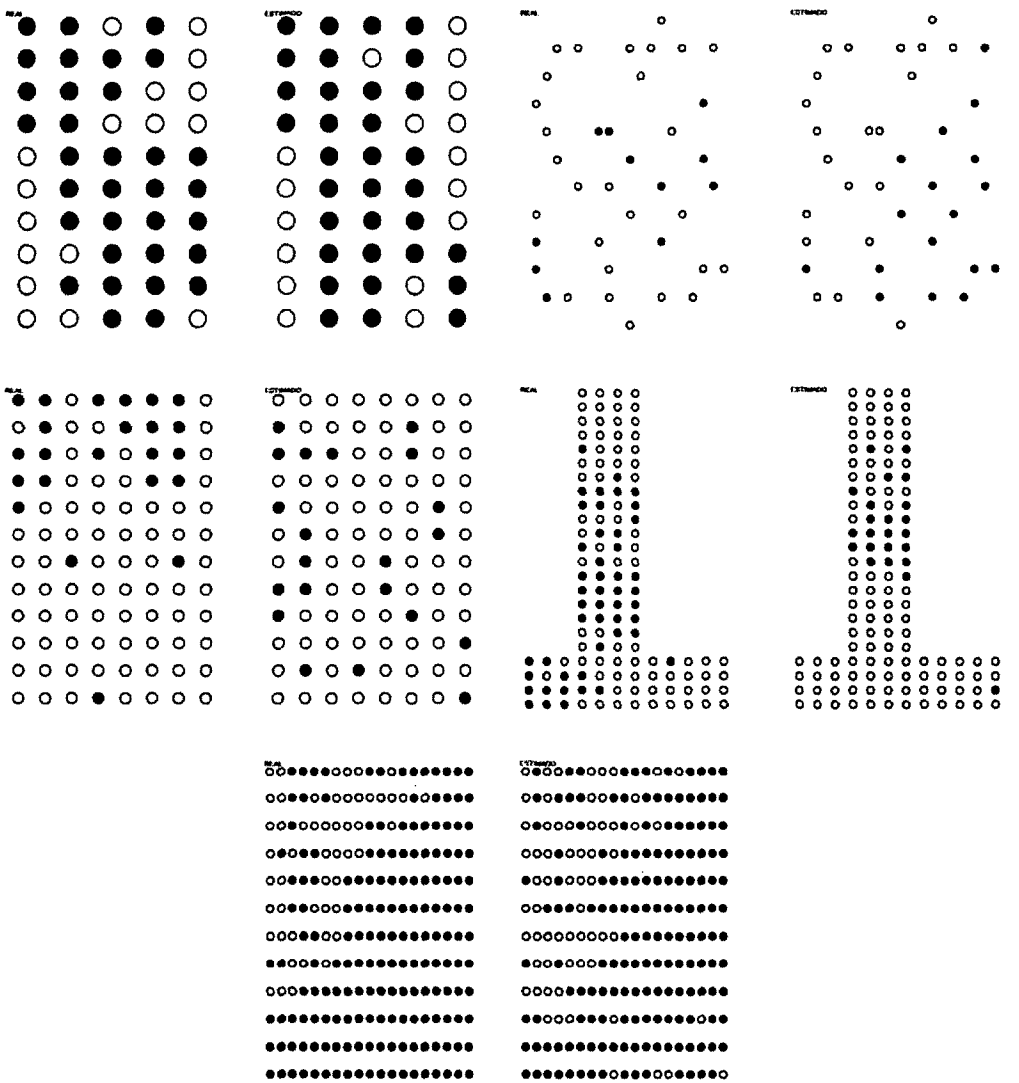


Figura 11.9: Distribución real de la infestación de avena y distribución estimada por el modelo A_1

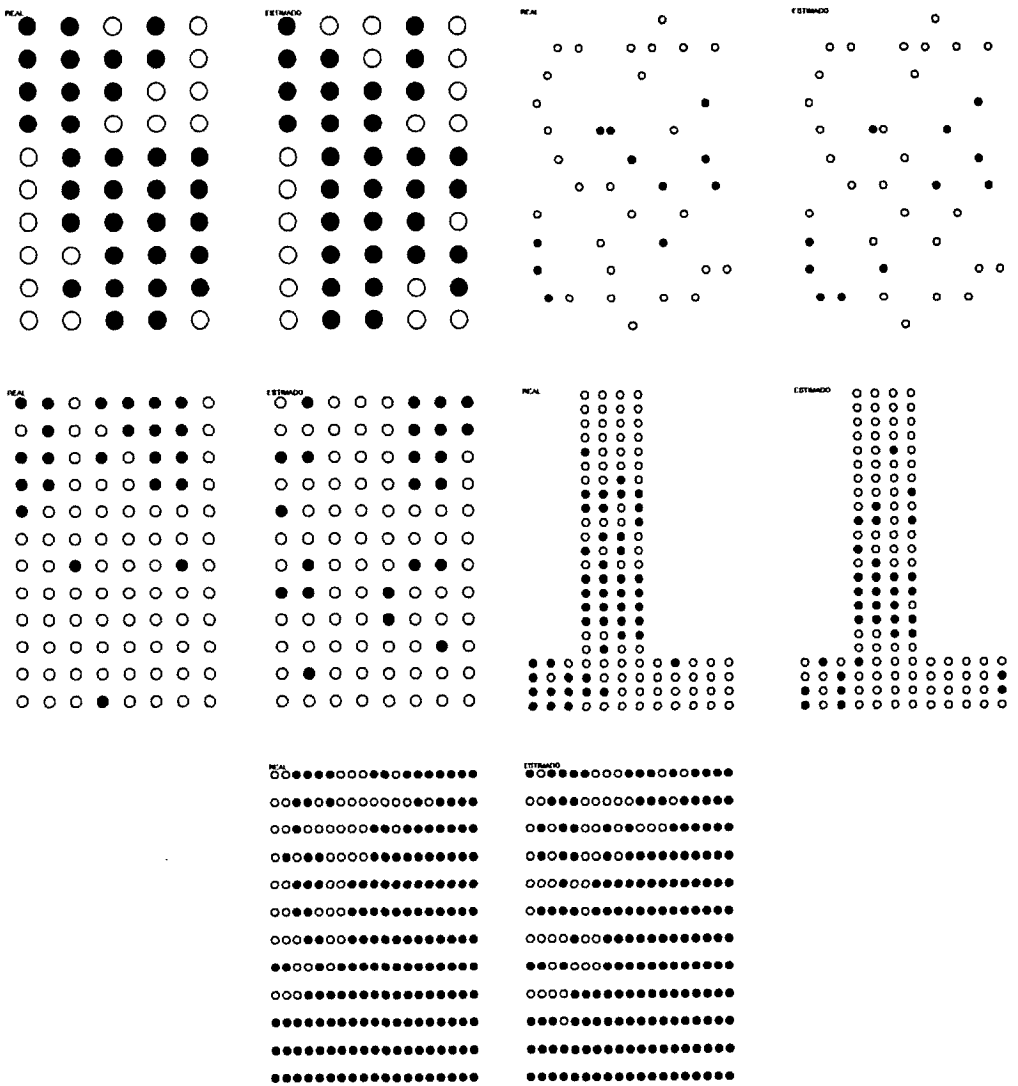


Figura 11.10: Distribución real de la infestación de avena y distribución estimada por el modelo B_{12}

A_1	r_1			r_2		
	V_p	Co	% +	V_p	Co	% +
1	27	81,8	*81,8*	0	0	*0*
2	7	41,2	*63,6*	0	0	*0*
3	2	22,2	*8,7*	3	23,1	*13*
4	8	40	*16,7*	7	50	*14,6*
5	124	93,9	*70,5*	23	71,9	*13,1*

*Tabla 11.*V: Valores de calidad de cada una de las reglas del modelo A_1 para cada parcela

La calidad del modelo B_{12} para cada una de las parcelas, utilizando también todos los datos recopilados, presentan los siguientes valores:

- n^0 1: Exactitud 82,0 % Confianza 87,5 %
- n^0 2: Exactitud 84,2 % Confianza 72,7 %
- n^0 3: Exactitud 79,2 % Confianza 56,5 %
- n^0 4: Exactitud 79,0 % Confianza 82,4 %
- n^0 5: Exactitud 86,4 % Confianza 90,5 %

En este caso, los valores de calidad para cada campo muestran exactitudes y confianzas similares, ambas muy altas, hecho que coincide con la especificidad de modelos complejos como es el caso de B_{12}.

En la figura 11.10 se puede observar que el modelo B_{12} tiende a describir bien todas las parcelas, algo que cabía esperar al ser un modelo complejo que incluye reglas muy específicas. En este caso el modelo incorpora reglas que describen grupos minoritarios en la muestra, como son los datos correspondientes a las parcelas de Nuevo Baztán. Esto último se aprecia bien si se contrastan los mapas estimados por el modelo A_1 para las campañas 3 y 4, (figura 11.9) con los estimados por el modelo B_{12} (figura 11.10) para las mismas campañas.

En la tabla 11.VI se muestran los valores de los parámetros de calidad para el modelo B_{12} en cada campaña y para cada regla. Los valores presentados en la tabla ponen de manifiesto la misma diferencia observada entre los mapas de Nuevo Baztán y de Arganda. En esta tabla además destacan: a) la regla B_{12}:r_9

describe mejor las muestras de las campañas 5 y 1, y ligeramente las otras; b) la regla B_{12}:r_4 es un buen clasificador de las campañas 3 y 4 y es capaz de explicar algunos de los datos de la campaña 5; y c) la regla r_3 describe bien la campaña 2 y algo las campañas 1 y 5. Además se puede observar un comportamiento mayoritariamente especializado de las reglas, ya que muchas reglas cubren pocos ejemplos y ninguna regla, a excepción de B_{12}:r_9 y B_{12}:r_4, funciona bien en todos los campos.

B_{12}	r_1			r_2			r_3			r_4		
	V_p	Co	% +	V_p	Co	% +	V_p	Co	% +	V_p	Co	% +
1	1	100	*3*	2	100	*6,1*	2	100	*6,1*	4	100	*12,1*
2	1	100	*9,1*	0	0	*0*	4	100	*36,4*	1	100	*9,1*
3	0	0	*0*	1	100	*4,3*	0	0	*0*	6	66,7	*26,1*
4	4	100	*8,3*	4	80	*8,3*	0	0	*0*	10	83,3	*20,8*
5	0	0	*0*	13	86,7	*7,4*	6	100	*3,4*	10	90,6	*5,7*
	r_5			r_6			r_7			r_8		
	V_p	Co	% +	V_p	Co	% +	V_p	Co	% +	V_p	Co	% +
1	0	0	*0*	0	0	*0*	0	0	*0*	0	0	*0*
2	0	0	*0*	0	0	*0*	0	0	*0*	0	0	*0*
3	1	100	*4,3*	0	0	*0*	3	100	*13*	0	0	*0*
4	0	0	*0*	0	0	*0*	0	0	*0*	10	90,9	*20,8*
5	12	100	*6,8*	7	77,8	*4*	0	0	*0*	2	40	*1,1*
	r_9			r_{10}			r_{11}					
	V_p	Co	% +	V_p	Co	% +	V_p	Co	% +			
1	17	81	*51,5*	3	100	*9,1*	4	100	*12,1*			
2	1	100	*9,1*	0	0	*0*	1	25	*9,1*			
3	2	33,3	*8,7*	1	50	*4,3*	0	0	*0*			
4	2	50	*4,2*	0	0	*0*	0	0	*0*			
5	122	96,1	*69,3*	6	54,5	*3,4*	17	94,4	*9,7*			

*Tabla 11.*VI: Valores de calidad de cada una de las reglas del modelo B_{12} para cada parcela

Resumiendo, las campañas 1, 2 y 5 (Arganda del Rey) se describen mejor con las reglas:

(B_{12}:r_9)-(MO ≠ menor AND P ≠ mayor AND arcilla ≠ menor AND limo ≠ mayor AND arena ≠ menor)

(B_{12}:r_3)-(N = mayor AND MO = mayor AND P ≠ mayor AND K ≠ medio AND arcilla ≠ menor AND limo ≠ medio AND arena ≠ medio)

(A_1:r_1)-(MO ≠ menor AND P ≠ mayor AND arcilla ≠ menor)

En estas parcelas la avena aparece fundamentalmente relacionada con cantidades bajas o medias de fósforo, altas cantidades de materia orgánica y tex-

turas de tipo grueso.

Por otra parte, las campañas 3 y 4 (Nuevo Baztán) se explican más adecuadamente con las reglas:

$(B_{12}{:}r_4)$-(pH $\neq$ mayor AND MO $\neq$ medio AND P = menor AND limo $\neq$ mayor)

$(A_1{:}r_2)$-(limo = medio AND arena = menor)

Estas dos reglas indican que la aparición de la avena en estas campañas estaría relacionada con texturas poco arenosas y con un contenido medio en limo o con cantidades bajas de fósforo y medias de materia orgánica.

Interpretación general de los resultados

A modo de resumen decir que la definición de un modelo como un conjunto de reglas que utilizan condiciones sobre variables cualitativas permite una exposición directa y fácilmente interpretable del conocimiento. De hecho, el análisis de las reglas de mayor cobertura y confianza deja conocer relaciones entre variables bien refrendadas que existen en los datos, a la vez que la interpretación de estas relaciones lleva a la extracción de conocimiento potencialmente nuevo y útil. Bajo la anterior perspectiva destaca la información suministrada por: a) la regla $B_1{:}r_1$ (MO $\neq$ menor AND limo $\neq$ mayor) que cubre el 75 % de los ejemplos positivos (153) y con una confianza (76,9 %) relativamente alta; b) la regla $A_3{:}r_4$ MO $\neq$ menor AND P $\neq$ mayor AND arcilla $\neq$ menor AND limo $\neq$ mayor AND arena $\neq$ menor que describe correctamente 118 ejemplos de un total de 204 con una confianza del 94,4 %; c) las reglas $B_2{:}r_2$ (MO $\neq$ menor AND P $\neq$ mayor AND arcilla $\neq$ menor AND limo $\neq$ mayor) y $A_2{:}r_3$ (arcilla $\neq$ menor AND arena = mayor), cubren 118 ejemplos positivos con una confianza del 91,5 %, que establece un error en la cobertura de ejemplos positivos menor del 10 %; d) la regla $A_1{:}r_1$ (MO $\neq$ menor AND P $\neq$ mayor AND arcilla $\neq$ menor) que explica 133 ejemplos positivos con una confianza de aproximadamente el 84 %. Todas estas reglas pueden ser de interés para inferir conocimiento nuevo potencialmente útil.

Todas las reglas anteriores indican mayoritariamente que las características de textura del suelo están muy relacionadas con la densidad de avena loca en las parcelas con cultivo de cereal de invierno de Madrid. Las zonas que presentan suelo con un bajo contenido en limo y contenidos medios en arcillas, por tanto

de textura más arenosa, son áreas donde hay, en términos relativos, una mayor densidad de mala hierba. Además las reglas indican que la avena se desarrolla peor cuando el contenido en materia orgánica es bajo.

Finalmente, algunas de las reglas señalan que cantidades medias o bajas de fósforo están asociadas a una mayor abundancia de avena. Sabiendo que en general este mineral es fundamental en el crecimiento y la evolución de los cereales, ya que condiciona la formación de su sistema radicular, es posible determinar que si la avena disminuye cuando la cantidad de fósforo es alta puede ser debido a que el cultivo ha crecido antes y mejor y por lo tanto es más competitivo. Ahora bien, para probar todas las hipótesis referentes a relaciones de competencia entre especies es necesario llevar a cabo análisis que incluyan variables sobre el comportamiento del cultivo como por ejemplo, el rendimiento o la cantidad de las otras especies vegetales.

11.2.2. Experimentación con datos de Barcelona

⋆ Estudio de los datos de Barcelona con modelos de avena de Madrid

En la siguiente etapa del estudio se analiza si los modelos obtenidos para la infestación de avena en los campos de Madrid, explicados en el apartado anterior, se ajustan a los datos de avena de Barcelona.

$caso_\eta$	V_p	F_n	V_n	F_p	Exactitud (%)	Confianza (%)
A_1	16	27	101	110	46,06	12,70
A_2	29	14	66	145	37,40	16,67
A_3	9	34	98	113	42,13	7,38
A_4	9	34	85	126	37,01	6,67
A_5	8	35	94	117	40,16	6,40
A_6	9	34	92	119	39,76	7,03
B_1	18	25	67	144	33,46	11,11
B_2	8	35	113	98	47,64	7,55
B_3	19	24	67	144	33,86	11,66
B_4	8	35	127	84	53,15	8,70
B_5	15	28	98	113	44,49	11,72
B_6	8	35	114	97	48,03	7,62
B_7	8	35	101	110	42,91	6,78
B_8	7	42	154	57	61,92	10,94
B_9	10	33	107	104	46,06	8,77
B_{10}	11	32	103	108	44,88	9,24
B_{11}	8	35	110	101	46,46	7,34
B_{12}	8	35	110	101	46,46	7,34
C_1	22	21	91	120	44,49	15,49
C_2	25	18	74	137	38,98	15,43
C_3	25	18	90	121	45,28	17,12
C_4	25	18	95	116	47,24	17,73
C_5	25	18	74	137	38,98	15,43
C_6	25	18	74	137	38,98	15,43
C_7	23	20	83	128	41,73	15,23
C_8	25	18	81	130	41,73	16,13
C_9	19	24	104	107	48,43	15,08
C_{10}	18	25	111	100	50,79	15,25
C_{11}	17	26	114	97	51,57	14,91
C_{12}	20	23	120	91	55,12	18,02

*Tabla 11.*VII: Contraste de todos los modelos obtenidos para Madrid con los datos de avena de Barcelona en 2001 (umbral $\geq 0{,}20$)

Para realizar este paso cada uno de los modelos ha sido analizado con los

datos de avena del año 2001, utilizando dos umbrales diferentes para determinar la pertenencia de un elemento a las clases de *mucha* o *poca* avena. En una primera etapa de contraste de los modelos se ha utilizado un umbral de 0,20, es decir se ha considerado como mucha infestación cualquier situación en la que exista al menos un 20 % de avena. Recordemos que este umbral fue el empleado en la generación de los modelos a partir de los datos de Madrid. La tabla 11.VII muestra los valores de los parámetros de calidad resultantes de la operación de contraste con los datos de Barcelona de los 30 modelos generados a partir de los datos de Madrid.

$caso_\eta$	V_p	F_n	V_n	Fp	Exactitud (%)	Confianza (%)
A_1	48	61	67	78	45,28	38,10
A_2	70	39	41	104	43,70	40,23
A_3	41	68	64	81	41,34	33,61
A_4	42	67	52	93	37,01	31,11
A_5	39	70	59	86	38,58	31,20
A_6	41	68	58	87	38,98	32,03
B_1	59	50	42	103	39,76	36,42
B_2	35	74	74	71	42,91	33,02
B_3	61	48	43	102	40,94	37,42
B_4	33	76	86	59	46,85	35,87
B_5	47	62	64	81	43,70	36,72
B_6	35	74	75	70	43,31	33,33
B_7	36	73	63	82	38,98	30,51
B_8	15	94	102	43	46,06	25,86
B_9	39	70	70	75	42,91	34,21
B_{10}	40	69	66	79	41,73	33,61
B_{11}	36	73	72	73	42,52	33,03
B_{12}	36	73	72	73	42,52	33,03
C_1	62	47	65	80	50,00	43,66
C_2	69	40	52	93	47,64	42,59
C_3	65	44	64	81	50,79	44,52
C_4	62	47	66	79	50,39	43,97
C_5	69	40	52	93	47,64	42,59
C_6	69	40	52	93	47,64	42,59
C_7	63	46	57	88	47,24	41,72
C_8	66	43	56	89	48,03	42,58
C_9	48	61	67	78	45,28	38,10
C_{10}	46	63	73	72	46,85	38,98
C_{11}	43	66	74	71	46,06	37,72
C_{12}	49	60	83	62	51,97	44,14

*Tabla 11.*VIII: Contraste de todos los modelos obtenidos para Madrid con los datos de existencia de avena de Barcelona en 2001 (umbral $\geq$ 0,00)

Como se puede observar en la tabla los modelos son capaces de describir alrededor de la mitad de los datos de Barcelona con una confianza muy baja, con valores siempre inferiores al 19 %. De la tabla se infiere que los modelos generados para Madrid no se ajustan bien a los datos de Barcelona. Hay que tener en cuenta que la gestión de cultivo y tratamiento de la parcela de Barcelona fue diferente a la realizada en las parcelas de Madrid, algo que podría parcialmente explicar un comportamiento distinto en los datos. Asimismo, es lógico pensar que existen otras variables importantes como son la historia agronómica o las características climáticas de los campos, y teniendo en cuenta que no han sido incluidas en el análisis parece razonable que los modelos obtenidos con los datos de Madrid no se ajusten bien a los datos de Barcelona.

El segundo umbral utilizado para la definición de las clases mucha y poca avena es el 0,00, es decir se considera como infestación la existencia de avena independientemente de la cantidad. Con esta última elección de umbral lo que se pretende sencillamente es ver si los modelos son capaces de describir la existencia de avena loca en función de las mismas características del suelo detectadas en el caso de Madrid. La tabla 11.VIII muestra los valores para los parámetros de calidad con este segundo umbral. Se puede observar que, al igual que en las clasificaciones anteriores, la exactitud de los modelos está entorno al 50 % aunque la confianza aumenta hasta valores cercanos al 45 %.

Así mismo, se comprobó de nuevo el comportamiento de los modelos generados a partir los datos de Madrid, pero esta vez con los datos de avena de Barcelona del año 2002, apareciendo una notable mejora en los valores de confianza. Para el umbral 0,00 el tanto por ciento de exactitud se encontraba en el intervalo (41 % - 60 %) y el tanto por ciento de confianza se distribuyó en el intervalo (65 % - 78 %). En el caso del umbral 0,20, el tanto por ciento de exactitud tomó valores en el intervalo (35 % - 59 %) y el tanto por ciento de confianza en el intervalo (9 % - 27 %).

En la figura 11.11 se muestra la distribución real de avena del año 2001 con el umbral 0,00 -que ofrece mejores resultados- en el campo de Barcelona frente a la estimada por el modelo C_3, que ha clasificado estos datos con una

exactitud del 50,79 % y una confianza del 44,52 %.

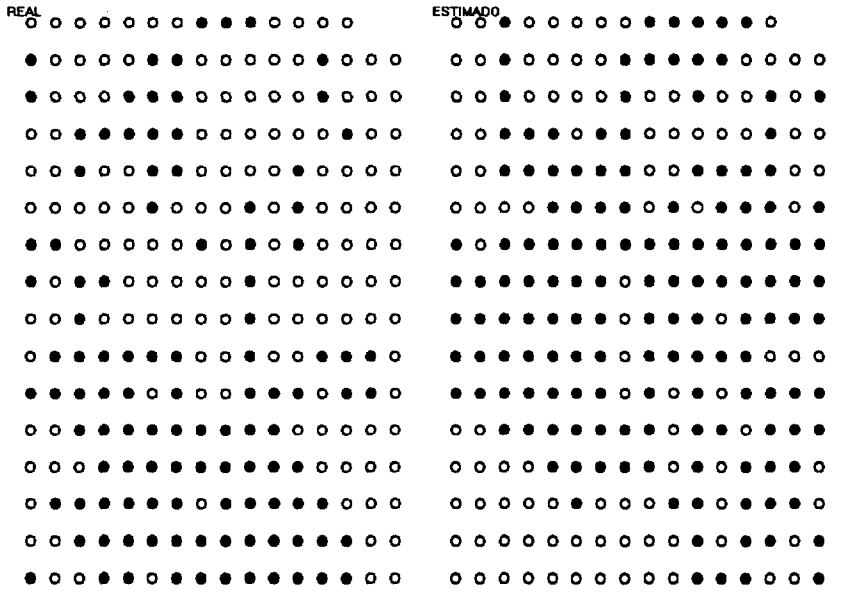


Figura 11.11: Distribución real y estimada con el modelo C_3 de avena en el campo de Barcelona para los datos del año 2001

Finalmente utilizando el mismo umbral (0,00) como corte, el modelo C_2 es el que mejor resultados presenta para la clasificación de los datos de Barcelona del año 2002, con una exactitud del 59,84 % y una confianza del 75,31 %. La figura 11.12 muestra gráficamente la distribución real de avena en el año 2002 en el campo de Barcelona frente a la estimada por el modelo. La elección de C_3 y C_2 se basa de nuevo en los parámetros de exactitud, confianza y simplicidad de los modelos.

De este estudio se desprende que la densidad de avena en el campo de Barcelona no se describe claramente con las relaciones entre propiedades del suelo encontradas para los datos de Madrid. En cualquier caso, esta falta de ajuste entre los modelos de Madrid y los datos de Barcelona era esperable si

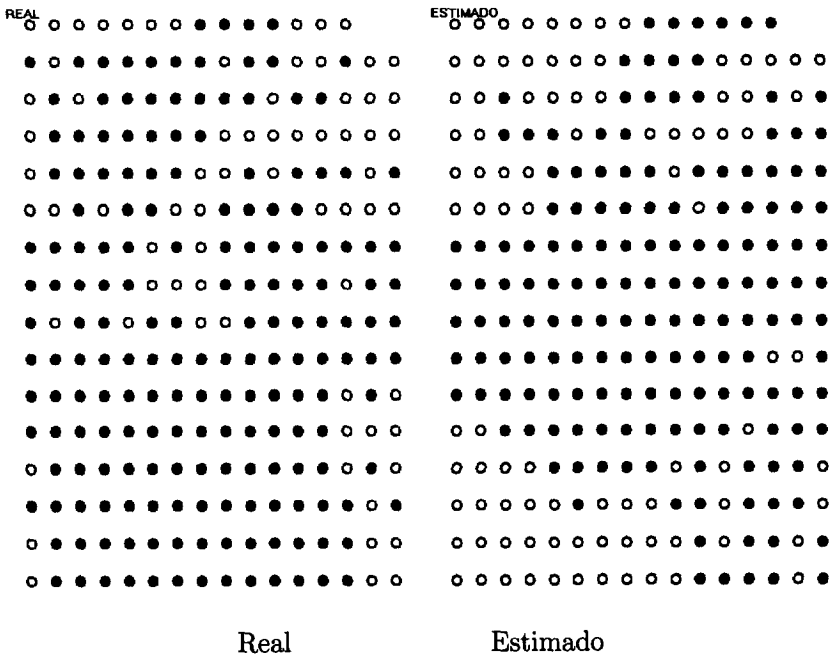


Figura 11.12: Distribución de avena real en el año 2002 y estimada por el modelo C_2 en el campo de Barcelona

tenemos en cuenta que no se han considerado en el estudio otros factores que son importantes, como por ejemplo el relieve y el clima.

⋆ Serie II - Experimentación con datos de avena

Una vez contrastados los modelos de Madrid con el conjunto de datos de Barcelona y a la vista de los resultados en la etapa de verificación explicada previamente, el siguiente paso es generar, a partir de la metodología propuesta, un conjunto de modelos que describan adecuadamente los datos de Barcelona. Partiendo de una base de datos más rica en información que la de Madrid, tal y como se expuso en el apartado 10.2, el conjunto de datos para el aprendizaje cuenta con nuevas variables como las relativas a cultivo o la radiación solar acumulada, entre otras. En este caso no es necesario que el análisis sea tan exhaustivo como el realizado con los datos de Madrid ya que inicialmente nos proponemos como objetivo generar los modelos más genéricos, por lo que utilizaremos únicamente los cromosomas cortos. Asimismo, y como ya se señaló anteriormente, el aprendizaje se realiza con los datos preprocesados de 2001 (242 ejemplos) y la verificación de los modelos con el conjunto de datos (254 ejemplos) del año siguiente (2002). Los valores de los parámetros de calidad para los mejores modelos obtenidos con los diferentes tamaños de cromosoma y las distintas codificaciones se presentan en la tabla 11.IX.

$Caso_\eta$	r	V_p	F_n	V_n	F_p	Exactitud (%)	Confianza (%)
A_1	6	68	35	116	23	76,03	74,73
A_2	7	73	30	115	24	77,69	75,26
B_1	1	72	31	107	32	73,97	69,23
B_2	2	77	26	107	32	76,03	70,64
B_3	3	72	31	121	18	79,75	80,00
C_1	1	65	38	112	27	73,14	70,65
C_2	2	76	27	111	28	77,27	73,08
C_3	3	77	26	113	26	78,51	74,76

Tabla 11.IX: Valores de los parámetros de calidad de los mejores modelos obtenidos para los diferentes tamaños de cromosoma para los distintos casos

Como ya se observó en modelos anteriores, los modelos correspondientes al tipo de codificación B, es decir que admiten el desigual como operador de comparación en la precondición, tienen mayor exactitud que los modelos de tipo C con condiciones más rígidas que sólo admiten precondiciones de

igualdad. De nuevo los modelos generados a partir de una codificación de tipo A tienen mayor calidad, pero está vez el número de reglas resultante es mucho mayor. Por ejemplo, en la tabla se observa que mientras que B_1 y C_1 tienen una sola regla A_1 presenta 6 reglas, resultando un modelo complejo de interpretar. Esta complejidad potencialmente progresiva es la razón por la que se decidió, para la codificación de tipo A, deterner la generación de modelos en $\eta = 2$, ya que se comprobó que cromosomas más grandes no mejoraban la calidad de los modelos generados. Como vimos en secciones anteriores, la codificación de tipo A permitía la aparición de operadores `OR` entre condiciones lo que daba lugar al desdoblamiento de un nucleosoma en varias reglas. En este caso el número de variables independientes (23) es mucho mayor que en el caso de los datos de Madrid (8) lo que explica porque, para el caso A, aparecen más reglas partiendo del mismo número de nucleosomas. Por otra parte, los modelos generados a partir de las codificaciones B y C presentan un rango de exactitud del 73 a 80 % con un rango de confianza del 69 a 80 %. Comparando la calidad de los modelos obtenidos para Barcelona con los generados para Madrid, se observa una diferencia de calidad de aproximadamente un 10 % para modelos del mismo nivel, es decir de igual longitud de cromosoma y codificación. Este comportamiento hace pensar que los datos de Barcelona son más heterogéneos que los de Madrid requiriendo modelos más complejos para obtener una mayor exactitud en la clasificación.

Las reglas que forman los modelos obtenidos para los datos de Barcelona se presentan a continuación, utilizando la misma forma de presentación que en el caso de Madrid, es decir sólo se muestra el antecedente ya que el consecuente es común para todas las reglas e igual a "`ENTONCES existe avena`". Los modelos obtenidos para cromosomas formados por un único nucleosoma son los siguientes:

- A_1 (F = 0,551 ; Ex = 76,0% ; Co = 74,4%)
  r1: trigo 2001 $\neq$ alta AND paja del trigo 2001 $\neq$ baja AND espigas del trigo 2001 = media AND grano trigo 2001 = baja (V_p 4; Co 66,7%)
  r2: elevacion $\neq$ baja AND orientacion = norte (V_p 58; Co 73,4%)
  r3: pendiente = media AND radiacion = alta AND pH $\neq$ alto (V_p 1; Co 100,0%)
  r4: conductividad = media AND N = normal (V_p 0; Co 00,0%)

r5: K = bajo AND Humedad ≠ alta (V_p 39; *Co* 79,6%)
r6: MO ≠ media-alta AND limo = media AND arcilla ≠ baja AND tipo suelo ≠ franco arcilloso (V_p 6; *Co* 75,0%)

- B_1 (F = 0,538 ; Ex = 74,0% ; Co = 69,2%)
  r1: grano trigo 2001 ≠ alta AND elevacion ≠ baja AND orientacion ≠ sur AND radiacion ≠ media
  AND pH ≠ bajo (V_p 72; *Co* 69,2%)

- C_1 (F = 0,508 ; Ex = 73,1% ; Co = 70,7%)
  r1: radiacion = baja AND pH = alto (V_p 65; *Co* 70,7%)

Los modelos no expresan información similar, es decir en general tienen precondiciones diferentes, pero no contradictorias. Destacan los modelos B_1 (grano trigo 2001 ≠ alta AND elevacion ≠ baja AND orientacion ≠ sur AND radiacion ≠ media AND pH ≠ bajo) que cubre 72 ejemplos positivos con una confianza de 69,2% y el modelo C_1 (radiacion = baja AND pH = alto) que cubre 65 ejemplos positivos con una confianza de 70,7%. También es interesante la regla r_2 del modelo A_1: (elevacion ≠ baja AND orientacion = norte) que cubre 58 ejemplos positivos con una confianza del 73,4%.

Es importante resaltar la existencia en el modelo A_1 de una regla r_4 cuya cobertura y confianza es igual a cero. Este tipo de situaciones se pueden dar aunque no son frecuentes. La razón es que la *fitness* que hemos elegido para el algoritmo de búsqueda puntúa los modelos en función de su calidad global no haciendo ninguna consideración sobre la calidad de las reglas que lo forman. La eliminación de este tipo de reglas del modelo debe ser el resultado de una etapa de postprocesamiento que debe llevarse a cabo una vez generado el modelo.

Con un cromosoma formado por $\eta = 2$ nucleosomas se obtuvieron los siguientes modelos.

- A_2 (F = 0,575 ; Ex = 77,7% ; Co = 75,3%)
  r1: paja trigo 2001 ≠ baja AND espigas trigo 2001 ≠ baja AND grano trigo 2001 = baja (V_p 4; *Co* 66,7%)
  r2: elevacion ≠ baja AND orientacion = norte (V_p 58; *Co* 73,4%)
  r3: pendiente = media AND radiacion = alta AND pH ≠ alto (V_p 1;100,0%)
  r4: conductividad = alta AND N = normal (V_p 0; *Co* 0,0%)
  r5: K = bajo (V_p 42; *Co* 76,4%)

r6: MO $\neq$ media-alta AND limo = 2 AND suelo $\neq$ franco arcilloso (V_p 6; *Co* 75,0%)
r7: paja trigo 2001 $\neq$ alta AND grano trigo 2001 $\neq$ alta AND pH = alto AND conductividad $\neq$ alta AND N = normal-alto AND Humedad $\neq$ media AND limo = baja AND arcilla = alta (V_p 5; *Co* 83,3%)

- B_2 (F = 0,575 ; Ex = 76,0% ; Co = 70,6%)
  r1: grano trigo 2001 = baja AND orientacion $\neq$ este AND P = alto AND Humedad $\neq$ baja AND arcilla $\neq$ media AND suelo $\neq$ franco (V_p 12; *Co* 70,6%)
  r2: radiacion = baja AND pH $\neq$ bajo AND conductividad $\neq$ alta AND K $\neq$ alto (V_p 65; *Co* 70,7%)

- C_2 (F = 0,589 ; Ex = 77,3% ; Co = 73,1%)
  r1: elevacion = media AND pH = alto AND P = alto AND MO = media-alta AND suelo = franco arcilloso (V_p 15; *Co* 83,3%)
  r2: radiacion = baja AND pH = alto AND conductividad = baja (V_p 65; *Co* 70,7%)

En este segundo conjunto de modelos se repiten algunas precondiciones en los antecedentes de las reglas como (`elevacion` $\neq$ `baja AND orientacion = norte`) y (`radiacion = alta AND pH = alto`) al que se añade la precondición (`conductividad = baja`), sin que se aprecien mejoras sustanciales en la calidad de los modelos. Este segundo conjunto de modelos tiene en general más precondiciones y son más específicos.

Los modelos obtenidos para un cromosoma con tres nucleosomas se muestran a continuación.

- B_3 (F = 0,609 ; Ex = 79,8% ; Co = 80,0%)
  r1: grano de trigo 2001 = baja AND orientacion $\neq$ este AND radiacion $\neq$ media AND N $\neq$ normal AND K $\neq$ alto AND Humedad $\neq$ baja AND arcilla $\neq$ media AND suelo $\neq$ franco limoso (V_p 13; *Co* 86,7%)
  r2: elevacion $\neq$ baja AND orientacion $\neq$ oeste AND radiacion = baja AND pH = alto AND K $\neq$ alto AND MO $\neq$ alta (V_p 57; *Co* 78,1%)
  r3: paja trigo 2001 $\neq$ media AND biomasa del lolium 2001 = baja AND elevacion = media AND pH = alto AND K $\neq$ bajo AND OM = media-alta AND arcilla $\neq$ media (V_p 7; *Co* 100,0%)

- C_3 (F = 0,608 ; Ex = 78,5% ; Co = 74,8%)
  r1: elevacion = media AND pH = alto AND P = alto AND MO = media-alta AND suelo = franco arcilloso (V_p 11; *Co* 91,7%)
  r2: radiacion = baja AND pH = alto AND MO = media-alta (V_p 63; *Co* 73,6%)
  r3: pendiente = baja AND K = bajo (V_p 21; *Co* 87,5%)

De nuevo los modelos resultantes exhiben un comportamiento más específico aunque son más parecidos entre sí que en los casos anteriores. En general las

reglas incluyen más precondiciones que reducen su cobertura parcial aumentando la confianza. Por ejemplo, la regla r_2 del modelo B_2 tiene dos precondiciones que cubren 65 ejemplos positivos con una confianza del 71 % y la regla r_2 del modelos B_3 tiene las mismas precondiciones y además cuatro precondiciones más lo que hace que su cobertura se reduzca a 57 ejemplos positivos, pero que su confianza aumente a un 78 %. Resumiendo, los modelos al igual que sucedía en el caso de Madrid tienen una tendencia a la especialización incluyendo reglas que cubren pocos ejemplos (11 ó 7 registros). No obstante, en este caso, reglas C_3:$r2$, que incluye que la precondición relacionada con la radiación mantiene una cobertura alta y continúa siendo la variable más interesante en estos modelos. En este caso, este fenómeno de mantenimiento de ciertas reglas induce a pensar que éstas son reglas robustas, ya que el algoritmo no encuentra ninguna otra combinación de precondiciones que pueda suplantar a estas reglas ofreciendo mejor calidad en los modelos, incluso permitiendo mayor complejidad en los modelos finales.

Resulta interesante resaltar que los modelos más sencillos y genéricos no incluyen, en general, la información biológica. De hecho, precondiciones como (`paja trigo` $\neq$ `baja`) y (`espigas trigo` $\neq$ `baja`) aparecen en reglas de poca cobertura (de 4 a 12 V_p). Excepción a esto último, es la precondición (`grano trigo 2001` $\neq$ `alta`) que aparece en la regla r_1 del modelo B_1 que cubre 72 ejemplos positivos y que sugeriría que las zonas donde existe infestación de avena coinciden con áreas donde la biomasa de granos de trigo relativamente no es alta o, lo que es lo mismo, donde la producción no es de las altas de la parcela.

El siguiente paso es la fase de validación de los modelos en la que se han utilizado los datos de Barcelona del año siguiente (2002). Recuérdese que en los datos de Barcelona de 2002 eran variables con respecto a 2001 la biomasa de lolium, el trigo, el grano, la paja, las espigas y la presencia de paja mientras que las características edáficas y físicas se supusieron constantes.

Los resultados de la validación se presentan en la tabla 11.x. Se puede observar que la exactitud es algo menor que la obtenida en la etapa de en-

trenamiento mientras que los valores para la confianza aumentan (80-90 %), lo que indica una disminución de los falsos positivos (F_p) e implica clasificar correctamente un mayor número de ejemplos negativos. Así mismo, los valores de exactitud, al disminuir indican que la clasificación de los ejemplos positivos no es mejor que en el caso anterior, es decir hay una mayor proporción de ejemplos positivos sin cubrir, y esto coincide con el hecho de que el número de ejemplos positivos en 2002 es mayor frente al año anterior. Y en definitiva, la tendencia de este modelo es predecir que no existiría avena en el 2002 en las áreas en las que sí apareció.

Caso$_\eta$	nucleosoma	r	V_p	F_n	V_n	F_p	Exactitud (%)	Confianza (%)
A_1	1	6	85	99	52	18	53,94	82,52
A_2	2	7	90	94	52	18	55,91	83,33
B_1	1	1	103	81	51	19	60,63	84,43
B_2	2	2	99	85	57	13	61,42	88,39
B_3	3	3	86	98	60	10	57,48	89,58
C_1	1	1	87	97	59	11	57,48	88,78
C_2	2	2	99	85	59	11	62,20	90,00
C_3	3	3	99	85	60	10	62,60	90,83

Tabla 11.X: Valores para los parámetros de calidad en la etapa de validación del modelo

Los valores de la proporción de ejemplos positivos bien clasificados (%+ $= \frac{V_p}{p}$) y negativos correctamente clasificados (%- $= \frac{V_n}{n}$), que se presenta en la tabla 11.XI, corroboran este razonamiento.

Para concluir la evaluación, se elige el mejor modelo encontrado. En este caso la selección es algo más complicada que la realizada sobre los modelos generados a partir de los datos de Madrid, porque en este caso no existen grandes diferencias de calidad. Considerando criterios basados en simplicidad y exactitud se determina que C_3 es el mejor modelo, ya que además de simple presenta buenos resultados en entrenamiento (78 % de exactitud con una confianza del 74 %) y tiene de las mayores exactitudes en la etapa de validación (exactitud del 63 % con una confianza del 91 %). La distribución de avena estimada por el modelo frente a la distribución real se muestra en los mapas de las figuras 11.13a y 11.13b.

	Entrenamiento (2001)		Validación (2002)	
Caso$_\eta$	$\frac{V_p}{p}$(%+)	$\frac{V_n}{n}$(%-)	$\frac{V_p}{p}$(%+)	$\frac{V_n}{n}$ (%-)
A_1	*66,0*	83,5	*46,2*	74,3
A_2	*70,9*	82,7	*48,9*	74,3
B_1	*69,0*	77,0	*56,0*	72,0
B_2	*74,0*	77,0	*53,8*	81,4
B_3	*69,9*	87,1	*46,7*	85,7
C_1	*63,1*	80,6	*47,3*	84,3
C_2	*73,8*	79,9	*53,8*	83,3
C_3	*74,8*	81,3	*53,8*	85,7

*Tabla 11.*XI: Proporción de ejemplos positivos y ejemplos negativos correctamente clasificados

El modelo C_3 no incluye ningún factor biológico por lo que el mapa de la distribución estimada de avena es el mismo para los dos años. Recuérdese que sólo los factores biológicos son los que varían de un año a otro. En las figuras, se observa que el modelo explica mejor los datos del año 2001, lo que coincide con los datos expuestos en las tablas 11.IX y 11.X de las etapas de entrenamiento y validación, respectivamente.

Interpretación general de los resultados

Para establecer nuevo conocimiento a partir de los modelos generados resultan de especial interés las reglas que cubren una mayor proporción de ejemplos positivos y presentan mayor exactitud y confianza. Estas son las reglas que se enumeran en la lista siguiente:

$A_1 : r2$: elevacion $\neq$ baja AND orientacion = norte (V_p 58; *Co* 73,4%)

$A_1 : r5$: K = bajo AND Humedad $\neq$ alta (V_p 39 ; *Co* 79,6%)

$B_1 : r1$: grano trigo 2001 $\neq$ alta AND elevacion $\neq$ baja AND orientacion $\neq$ sur AND radiacion $\neq$ media AND pH $\neq$ baja (V_p 72; *Co* 69,2%)

$C_1 : r1$ radiacion = baja AND pH = alto (V_p 65; *Co* 70,7%)

$A_2 : r5$ K = bajo (V_p 42; *Co* 76,4%)

B_2 : $r2$ radiacion = baja AND pH $\neq$ bajo AND conductividad $\neq$ alta AND K $\neq$ alto (V_p 65; *Co* 70,7%)

B_3 : $r2$ elevacion $\neq$ baja AND orientacion $\neq$ oeste AND radiacion = baja AND pH = alto AND K $\neq$ alto AND MO $\neq$ alta (V_p 57; *Co* 78,1%)

$C_3 : r2$ radiacion = baja AND pH = alto AND MO = media-alta (V_p 63; *Co* 73,6%)

Todos estas reglas expresan, en general, que la existencia de avena está relacionada con zonas de la parcela donde existe menor altura, orientación al norte y hay menor radiación solar acumulada. Coinciden además caracterísi-

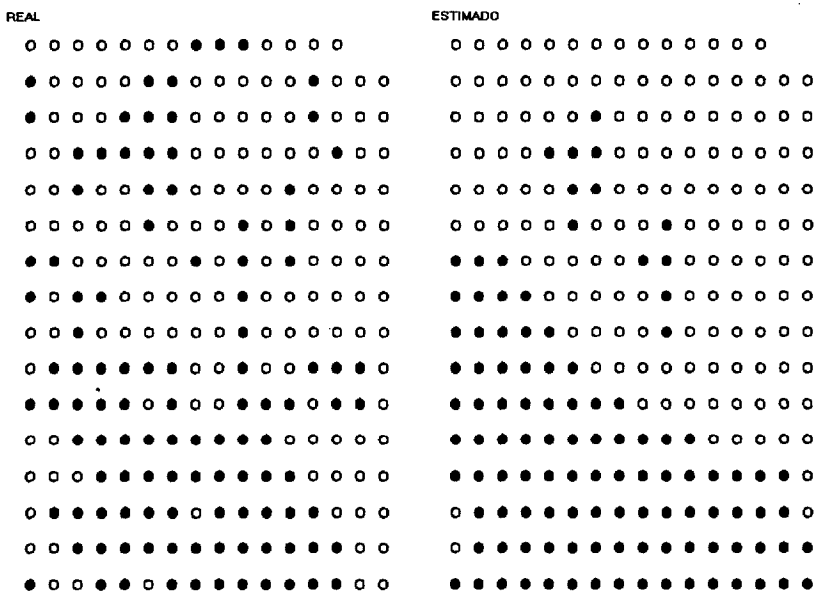


Figura 11.13: Distribución de la infestación real y estimada de avena en el año 2001 por el modelo C_3 sobre la parcela de Barcelona

cas como una menor biomasa de grano de trigo, un pH relativamente alto, potasio bajo y una conductividad distinta de alta, propia de una menor salinidad de las aguas. La información que se obtiene de las reglas puede ser el resultado de fenómenos de competencia, ya que existe avena en zonas donde la radiación es menor, el potasio es menor y existe menos biomasa de trigo, y por el contrario no existe avena cuando hay mayor biomasa de trigo, que a su vez está asociada a mayor radiación solar y mayor contenido en potasio.

⋆ `Serie III` - Experimentación con datos de lolium

Como se recordará los análisis descriptivos realizados durante la etapa de preprocesamiento (sección 10.2.2) mostraban que los datos de Barcelona, con relación a la infestación de lolium, eran muy heterogéneos. Por otra parte, los

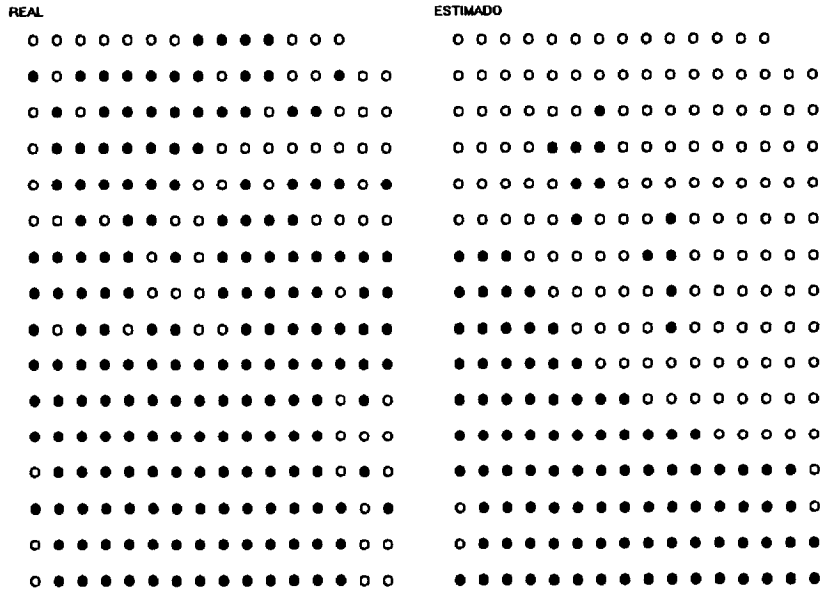


Figura 11.14: Distribución de la infestación de avena real y estimada en el año 2002 con el modelo C_3 en la parcela de Barcelona

diferentes análisis estadístico de tipo inductivo realizados por el equipo de la Universidad de Barcelona no proporcionaron resultados satisfactorios.

Para terminar la etapa de experimentación, en esta sección se exponen brevemente algunos de los experimentos realizados con el fin de generar modelos a partir de la metodología propuesta y con las aplicaciones desarrolladas, que expliquen de la forma más genérica la infestación de lolium.

- III.1. - *Factores Edáficos*

Para generar la primera colección de modelos se han tenido únicamente en cuenta los factores edáficos que es la información más básica del suelo, considerada además constante en el tiempo. Es importante recordar que la etapa de entrenamiento o aprendizaje con este primer conjunto de datos, busca la más descripción general de las muestras en las que la densidad lolium permanece

estable durante todos los años (2001, 2002 y 2003), en concreto un 40 % de los datos. Por lo tanto la experimentación se realiza únicamente con los cromosomas más cortos, porque esteconjunto representa a menos de la mitad de los datos y modelos más complejos no ayudarán a obtener más conocimiento o incluso conocimiento útil sobre comportamiento de esta especie. Se persigue, por lo tanto, conocer el modelo más general posible de este conjunto de datos.

Los valores de los parámetros de calidad para los modelos obtenidos con cromosomas de un único nucleosoma ($\eta = 1$) con el objetivo de encontrar los modelos más generales para este conjunto de datos se muestran en la tabla 11.XII, donde r representa el número de reglas.

Caso$_\eta$	r	fitness S×E	V_p	F_n	V_n	F_p	Exactitud (%)	Confianza (%)
A_1	3	0,602	60	17	68	20	77,6	75,0
B_1	1	0,517	50	27	70	18	72,7	73,5
C_1	1	0,462	54	23	58	30	67,9	64,3

Tabla 11.XII: Resultados del conjunto de entrenamiento con factores edáficos para la lolium (datos de Barcelona)

Las reglas asociadas a cada uno de los modelos son las siguientes:

- A_1 (F = 0,602 ; Ex = 77,6% ; Co = 75,0%)
  r1: pendiente $\neq$ clase 2 AND radiacion = menor AND P $\neq$ alto AND K $\neq$ medio (V_p 24 ; Co 96,0%)
  r2: MO = medio (V_p 32 ; Co 74,4%)
  r3: suelo = franco arcilloso-limoso (V_p 21 ; Co 70,0%)

- B_1 (F = 0,517 ; Ex = 72,7% ; Co = 73,5%)
  r1: radiacion $\neq$ mayor AND N $\neq$ alto AND suelo $\neq$ franco-arcilloso (V_p 50 ; Co 73,5%)

- C_1 (F = 0,462 ; Ex = 67,9% ; Co = 64,3%)
  r1: radiacion = menor (V_p 54 ; Co 64,3%)

Estos modelos determinan que las muestras con un nivel de infestación de lolium alto, mantenido durante los tres años, se sitúan en zonas de radiación baja, con medias o bajas cantidades de nitrógeno. El suelo puede ser de cualquier tipo excepto franco-arcilloso. Atendiendo a la calidad general, el modelo

más exacto y con mayor confianza es A_1, pero B_1 y C_1 presentan la ventaja de ser más simples al estar formados sólo por una regla de relativa alta calidad.

- III.2. - *Factores Edáficos + factores biológicos*

El segundo conjunto de modelos se obtiene a partir de los datos de lolium incluyendo nuevos factores, entre otros las biomasas de trigo y avena. En este caso, se han utilizado cromosomas de hasta 6 nucleosomas ($\eta = 6$), ya que para este caso concreto el conjunto de datos ofrece más posibilidades y es más representativo de los datos globales, representando un 93,7 % de los datos iniciales.

En cuanto a los valores de los parámetros de calidad del conjunto de modelos obtenidos (tabla 11.XIII), éstos son muy parecidos a los valores de calidad de los modelos anteriores. Como máximo se consigue un 78 % de exactitud y una confianza del 72 %. Los modelos en general están formados por reglas específicas y, como se observa en la tabla, es necesario un mayor número de reglas -representado como r- para obtener exactitudes y confianzas más apropiadas, comprobando una vez más la heterogeneidad que característica de los datos de Barcelona.

Caso$_\eta$	r	fitness S×E	V_p	F_n	V_n	F_p	Exactitud (%)	Confianza (%)
A_1	5	0,545	79	20	95	44	73,10	64,23
B_1	1	0,442	59	40	103	36	68,10	62,10
B_2	2	0,547	69	30	109	30	74,80	69,70
B_3	3	0,599	80	19	103	36	76,90	68,97
B_4	4	0,620	79	20	108	31	78,60	71,82
C_1	1	0,406	65	34	86	53	63,40	55,10
C_2	2	0,476	59	40	110	29	71,00	67,05
C_3	3	0,515	65	34	109	30	73,10	68,40
C_4	4	0,589	73	26	111	28	77,30	72,30

Tabla 11.XIII: Valores de los parámetros de calidad para los modelos generados a partir de factores edáficos y biológicos para el lolium 2001 + 2002 (Barcelona)

Los modelos más generales formados a partir de cromosomas con un único nucleosoma ($\eta = 1$) se muestran a continuación:

- A_1 (F = 0,545 ; Ex = 73,1% ; Co = 64,2%)
  r1: ESPIGAS $\neq$ medio AND GRANO $\neq$ medio AND AVENA = bajo (V_p 43 ; Co 62,3%)
  r2: conductividad = bajo AND humedad $\neq$ medio AND arena = bajo (V_p 5 ; Co 83,3%)
  r3: limo = alto AND PAJA = no (V_p 24 ; Co 80,0%)
  r4: pendiente = clase 3 AND radiacion = menor AND pH $\neq$ bajo (V_p 40 ; Co 64,5%)
  r5: N $\neq$ alto AND P = medio AND MO = media (V_p 4 ; Co 66,7%)

- B_1 (F = 0,442 ; Ex = 68,1% ; Co 62,1%)
  r1: orientacion = norte AND pendiente = clase 3 AND N $\neq$ alto (V_p 59 ; Co 62,1%)

- C_1 (F = 0,406 ; Ex = 63,4% ; Co = 55,08%)
  r1: conductividad = bajo AND radiacion = menor (V_p 65 ; Co 55,08%)

Finalmente, los modelos con reglas más específicas son los siguientes:

- B_4 (F = 0,589 ; Ex = 77,3% ; Co = 83,2%)
  r1: conductividad = bajo AND pendiente = clase 3 AND radiacion = menor AND pH = alto (V_p 40 ; Co 66,7%)
  r2: limo = alto AND PAJA = no (V_p 24 ; Co 80,0%)
  r3: GRANO = alto AND humedad = medio AND pendiente = clase 3 AND pH = bajo AND N = normal-alto AND P = alto (V_p 7 ; Co 100,0%)
  r4: ESPIGAS = bajo AND AVENA = bajo AND pH = bajo (V_p 13 ; Co 100,0%)

- C_4 (F = 0,620 ; Ex = 78,6% ; Co 71,8%)
  r1: AVENA $\neq$ medio AND arcilla $\neq$ alto AND orientacion = norte AND pendiente = clase 3 AND N $\neq$ alto (V_p 53 ; Co 67,1%)
  r2: ESPIGAS $\neq$ medio AND AVENA = bajo AND humedad $\neq$ alto AND arcilla $\neq$ bajo AND pendiente $\neq$ clase 1 AND radiacion $\neq$ menor AND N $\neq$ normal AND K $\neq$ bajo (V_p 15 ; Co 83,3%)
  r3: GRANO $\neq$ alto AND arena $\neq$ medio AND limo = alto AND arcilla $\neq$ bajo AND PAJA = no AND pendiente $\neq$ clase 1 AND N $\neq$ alto AND P $\neq$ medio (V_p 11 ; Co 78,6%)
  r4: AVENA = medio AND conductividad $\neq$ alto AND humedad $\neq$ medio AND limo $\neq$ alto AND radiacion = menor AND N = normal AND P $\neq$ muy alto (V_p 4 ; Co 99,3%)

La información es más difícil de interpretar que en el caso de Madrid, porque son modelos compuestos por reglas más variadas y ninguna prevalece que forma notable. No obstante, según el modelo B_1 el lolium, para los años 2001 y 2002, aparece en mayor cantidad en zonas orientadas al norte, pendientes de 3 a 10 grados y cuando el nitrógeno no es alto. Además, el modelo A_1 con 73 % de exactitud y 65 % de confianza añade que el lolium puede estar relacionado con

zonas de menor radiación, cantidades de limo alto, cuando no hay paja y el pH es más bajo (suelo relativamente más básico), o bien cuando la cantidad de espigas y grano de trigo es media y la cantidad de avena es baja.

11.3. Comparación con algoritmos comerciales que generan modelos basados en reglas

En este capítulo se compara el método propuesto y la herramienta desarrollada con otras aplicaciones para la extracción de modelos basados en reglas disponibles comercialmente e incluidas en el paquete Clementine© 6.0.2, del grupo SPSS Inc[2]. Clementine es una plataforma informática formada por herramientas de ayuda a la extracción de relaciones de interés y valiosas en un conjunto de datos. De todo el conjunto de herramientas que suministra Clementine son de especialmente interesantes, para comprobar las ventajas de la metodología propuesta, las utilidades que permiten la generación de modelos basados en reglas y de los análogos árboles de decisión, más específicamente los algoritmos C5.0 y CART.

Algoritmo C5.0

C5.0[3] (See5) es la versión comercial derivada de los algoritmos ID3 y C4.5 para la generación de árboles de decisión. El algoritmo C5.0 produce árboles de decisión a partir de un conjunto de datos y puede transformar el árbol generado en reglas, mejorando con ello la interpretación de la información contenida en el árbol. La búsqueda del mejor árbol de decisión se basa en la división recursiva del conjunto de datos a partir de condiciones sobre los diferentes atributos de modo que siempre se obtenga la máxima ganancia de información o mínima entropía, tal como se explicó en el apartado 3.1.3. Los árboles se construyen hacia delante por lo que cada hoja terminal contiene un subconjunto de los datos de entrenamiento. El proceso de generación del árbol termina cuando no existe una condición sobre un atributo que separe de nuevo los subconjuntos obtenidos con la última división. Para obtener el antecedente de la regla que describe el subconjunto se encadenan las condiciones que se encuentran en los nodos que unen la raíz del árbol con el nodo hoja o nodo terminal. Como etapa final el algoritmo C5.0 tiene una fase de poda que elimina las peores reglas, basándose en un valor de calidad, en concreto la confianza.

[2] http://www.spss.com/

[3] http://www.rulequest.com/

Cuando el algoritmo C5.0 genera reglas, los modelos resultantes son más simples y genéricos [Cle, 2000]. Cada regla obtenida por el algoritmo tiene asociados dos parámetros de calidad: (1) La cantidad de instancias que cubre la regla ($V_p + F_p$) y (2) Un parámetro similar a la confianza que evalúa la proporción de casos en los que la regla es verdadera, en otras palabras ($Co_{clem} = \frac{1+verdaderos}{2+cubierta}$) donde *verdaderos* equivale a V_p y *cubierta* es ($V_p + F_p$). Este cálculo de confianza estimada se usa para el proceso de generalización de reglas desde un árbol de decisión, que es lo que C5.0 hace cuando crea un cualquier conjunto de reglas.

Los conjuntos de reglas son derivados de árboles de decisión, y representan de forma simplificada toda la información encontrada en éstos, por lo que las reglas son modelos menos complejos. Como hemos mencionado, Clementine ofrece la posibilidad de construir "directamente" estos conjuntos de reglas, es decir que no es necesario una fase previa de construcción del árbol, y la posterior transformación. Con esta opción, el modelo resultante es directamente un conjunto de reglas. La obtención directa de reglas se puede realizar de dos formas diferentes. Por un lado, una forma precisa o específica que favorecen la exactitud (exactitud máxima de la muestra del entrenamiento) y por otro lado, una genérica que favorece la generalidad (los resultados que deben generalizar mejor a nuevos datos).

Algoritmo CART

En este caso el llamado *árbol C&R* (*Classification and Regression*) es el modelo más popularmente utilizado en el aprendizaje simbólico para la generación de árboles, por lo que comienza a ser relativamente frecuente ver estos modelos en cualquier área de investigación, también como vimos en los primeros capítulos de esta tesis, en el campo de la ecología. Este tipo de árboles se construye utilizando como criterio de división del conjunto de datos el índice o *coeficiente de Gini* que indica la diversidad de una población. El valor de este índice se utiliza para la reducción del grado de impureza obtenido a partir de la probabilidad de fallar en la clasificación, y se calcula con $Gini = 1 - \sum_i \rho_i^2$, donde ρ_i es la probabilidad de que un elemento pertenezca a la clase i en un

determinado nodo del árbol. La probabilidad de un nodo puro determina que $Gini = 0$. El índice de Gini mide el grado en que una distribución de los ejemplos se desvía con respecto a la igualdad perfecta.

Durante la construcción de un árbol, en cada posible división se calcula este índice, y posteriormente se elige la división que determina una impureza mínima. El algoritmo C&R genera una estructura de esencia binaria, es decir las divisiones producen únicamente dos grupos y por lo tanto, utiliza etiquetas lógicas de verdadero o falso para generar las condiciones en los nodos. El árbol resultante tiende a crecer hasta que no queda un sólo elemento sin describir. Por lo que en consecuencia requiere una etapa posterior de poda que está guiada por validación cruzada para valores categóricos y por desviación estándar para valores continuos. Este método presenta problemas si existe ruido en los datos o si el número de campos es enorme. Este algoritmo, a diferencia del anterior, no da la opción de construir directamente el conjunto de reglas, sino que éstas se obtienen de la conversión del árbol por el propio usuario.

Algoritmos C5.0 y CART de Clementine aplicados a los datos de Madrid y Barcelona

Para realizar la comparación se utilizaron los datos de avena de Madrid y Barcelona con los mismos conjuntos de entrenamiento y de validación definidos anteriormente (capítulo 10), pero aplicando para la extracción de conocimiento (modelos basados en árboles de decisión y en reglas) los algoritmos C5.0 y CART.

- DATOS DE AVENA EN MADRID

  C5.0 árbol

  Los modelos obtenidos con el algoritmo C5.0, tanto árboles como reglas, se presentan en el apéndice F. De estos resultados, la relación más interesante se muestra en una rama que explica los registros con `mucha` cantidad de avena y que consta de tan sólo de dos nodos (`limo=medio`) y (`materia organica = medio`). La concatenación de estas dos condiciones cubre 86 registros del conjunto de entrenamiento con una confianza

del 82,60 %, lo que significa que la mayor parte de estos 86 registros están correctamente clasificados. Este resultado coincide con una de las reglas encontradas con la metodología de la propuesta, en concreto la regla r_1 del modelo B_1. El árbol completo construido por el algoritmo C5.0 se transforma en un total de 39 reglas distribuidas de la siguiente forma: 19 reglas (17 tras una poda manual que elimina las reglas con cobertura igual a cero) para la clase mucha y 20 para la clase poca. Los valores para los parámetros de calidad de los árboles obtenidos para los conjuntos de datos de entrenamiento y de validación se muestran en las tablas 11.XIV y 11.XV. La transformación de los árboles permite obtener dos conjuntos de reglas que presentan los mismos valores para los parámetros de calidad. Además de la regla anteriormente comentada destacan para la clase "avena mucha" las siguientes reglas: la regla nº 16 (`limo = menor AND arcilla medio`) que cubre 42 registros con una confianza del 100 % y la regla nº 14 (`limo = menor AND arcilla = mayor AND arena = mayor`) que cubre 26 registros con una confianza del 92,6 %.

C5.0 reglas

Como hemos comentado anteriormente, el algoritmo C5.0 permite generar un modelo basado en reglas directamente con dos modos de funcionamiento que dan lugar a la obtención de reglas genéricas o más específicas. Al igual que sucedía con la metodología propuesta los modelos más específicos son más exactos y presentan mayor confianza. El *método específico* (*espcf.*) da como resultado 8 reglas para describir la clase "avena mucha" y 10 para describir la clase complementaria "avena poca". Con esta opción del algoritmo el modelo generado tiene un menor número de reglas que en el caso del modelo obtenido a partir del árbol de decisión y además las reglas son más simples y todo ello sin perder exactitud ni confianza e incluso aumentando algo el valor de estos parámetros tal y como se muestra en las tablas 11.XIV y 11.XV. Cuando se aplica el método de funcionamiento que genera reglas genéricas el modelo obtenido consta de 7 reglas, 4 para describir la clase "avena poca" y 3 para describir el

conjunto de los ejemplos positivos, clase "**avena mucha**". La calidad de este segundo modelo es similar ya que el paso de un tipo de modelo a otro se consigue con la eliminación de las reglas más especializadas. En general, las reglas de este modelo simplificado están compuestas por una precondición, por lo que además de genéricas son simples.

Entre el conjunto de reglas de este último modelo destacan las siguientes:

Regla nº 5 (`limo = menor`) que cubre 112 registros con una confianza del 75,4 %.

Regla nº 8 (`limo = medio`) que cubre 154 registros con una confianza del 62,2 %.

Regla nº 3 (`pH = mayor AND N = medio AND arcilla = medio`) que cubre 47 registros con una confianza del 87,8 %.

Estas reglas ofrecen información similar a la obtenida a partir de los modelos generados con la metodología propuesta. Como se muestra en la tabla el número de reglas que forman los modelos generados por los algoritmos elegidos de Clementine es muy superior al número de reglas de los modelos obtenidos por nuestra propuesta `AGlearner + SQLdeco.dll`. Por ejemplo, el algoritmo C5.0 para obtener un modelo genérico utiliza 7 reglas, mientras que en nuestro caso el modelo genérico seleccionado A_1, emplea únicamente 2, sin disminuir mucho la exactitud ni la confianza, un 5 % y un 1 % de diferencia respectivamente. Por otro lado, el modelo basado en reglas más específico de C5.0 usa 18 reglas, y en nuestro caso, B_{12} requiere 11.

C&R árbol (CART)

En cuanto al **algoritmo C&R**, este construye un modelo en forma de árbol de decisión cuyo valores para los parámetros de calidad son los que se muestran en las tablas 11.XIV y 11.XV. Puede observarse una pequeña disminución en los valores de estos parámetros aunque no es realmente significativa. Lo que sí es importante, es que el método utilizado en la construcción de este tipo de árboles es más restrictivo que el resto de

Algoritmo	r	Vp	Fn	Vn	Fp	Exactitud (%)	Confianza(%)
C5.0 árbol	39	189	15	146	23	89,81	89,15
C5.0 reglas (espcf.)	18	186	18	147	22	89,28	89,43
C5.0 reglas (gener.)	7	174	30	132	37	82,04	82,46
C&R árbol	17	190	14	137	32	87,80	85,59
$\texttt{AGLearner}_{(+espcf.)}(B_{12})$	11	193	11	156	13	93,57	93,69
$\texttt{AGLearner}_{(+gener.)}(A_1)$	2	155	49	133	36	77,21	81,15

*Tabla 11.*XIV: Resultados de Clementine© con avena en Madrid (entrenamiento)

los paradigmas explicados y por eso, menos exacto, incluido el método propuesto en esta tesis, ya que independientemente del número de etiquetas de los atributos, la división del conjunto es binaria, es decir se generan dos grupos, uno con los registros que presentan una determinada etiqueta y otro con el resto de los registros. El árbol resultante tiene 7 niveles, que dan lugar a 17 reglas para describir la clase positiva y 11 reglas para representar la clase negativa. La construcción del conjunto de reglas a partir del árbol no utiliza un proceso de poda como en el caso del algoritmo C5.0 lo que provoca provoca reglas muy específicas y por tanto más difíciles de interpretar. Los valores obtenidos para los parámetros de calidad con este modelo son los que se muestran en las tablas 11.XIV y 11.XV.

En la tablas señaladas también se han introducido los valores de los parámetros de calidad para los modelos obtenidos a partir de la metodología propuesta. A la vista de los valores se puede establecer que los modelos obtenidos con la herramienta desarrollada tienen mayor calidad en la clasificación y/o son comparativamente sencillos, ya que con un número de reglas nunca mayor a 12 ($\eta \leq 12$) se obtienen valores de exactitud y confianza superiores al 90 %.

- DATOS DE BARCELONA

  C5.0 reglas

  Tras el análisis y la comparación de métodos con los datos de Madrid y a la vista de los resultados, se ha optado, para no alargar innecesariamente

Algoritmo	Vp	Fn	Vn	Fp	Exactitud (%)	Confianza (%)
C5.0 árbol	22	4	13	2	85,37	91,67
C5.0 reglas (espcf.)	23	3	13	2	87,80	92,00
C5.0 reglas (gener.)	23	3	13	2	87,80	92,00
C&R árbol	24	2	12	3	87,80	88,89
$\texttt{AGLearner}_{(+espcf.)}(B_{12})$	22	4	13	2	85,37	91,67
$\texttt{AGLearner}_{(+gener.)}(A_1)$	19	7	13	2	78,05	90,48

*Tabla 11.*XV: Resultados de Clementine© con avena en Madrid (validación)

Algoritmo	r	Vp	Fn	Vn	Fp	Exactitud (%)	Confianza(%)
C5.0 reglas (espcf.)	9	68	35	127	12	80,6	85,0
C5.0 reglas (gener.)	5	63	40	123	16	76,9	79,7
$\texttt{AGLearner}_{(+espcf.)}(C_3)$	3	77	26	113	26	78,51	74,76
$\texttt{AGLearner}_{(+gener.)}(B_1)$	1	72	31	107	32	73,97	69,23

*Tabla 11.*XVI: Resultados de Clementine© con avena en Barcelona (entrenamiento)

el estudio comparativo, por extraer información para los datos de Barcelona únicamente con el **algoritmo C5.0** en su versión de generación de modelos basados en reglas.

El algoritmo C5.0 (reglas) se utiliza con las dos formas de funcionamiento anteriormente explicadas y que permiten obtener modelos con reglas genéricas y modelos con reglas específicas. Los resultados para los datos del conjunto de entrenamiento se presentan en la tabla 11.XVI. Se puede observar que los valores para los parámetros de calidad de los modelos obtenidos con Clementine para los datos de Barcelona son muy similares a los obtenidos con la aproximación propuesta para los mismos datos. Los modelos generados clasifican correctamente el 76,9 % de los ejemplos, el modelo genérico, y el 80,7 %, el modelo específico, con confianzas relativamente altas (80 % y 85 % respectivamente). Recordemos, que los modelos C_3, el más específico y B_1, el más genérico, ambos generados con nuestra propuesta `AGlearner + SQLdeco.dll`, ofrecen exactitudes de 78,51 y 73,97 % y confianzas del 74,76 y 69,23 %.

En cuanto a los datos de validación, los valores para los parámetros de calidad que se muestran en la tabla 11.XVII indican que los datos a partir de los cuales se generan los modelos no son representativos de la distribución de la avena para el año siguiente (2002), ya que se reduce la exactitud de la clasificación al 56 % y 59 % aunque aumentan las confianzas (95 % y 96 % respectivamente). Si comparamos la calidad de estos modelos con los obtenidos con la metodología propuesta para el mismo conjunto de datos, vemos que los valores para los parámetros de calidad son similares pero los modelos obtenidos con el algoritmo C5.0 son más complejos. Asi por ejemplo los modelos construidos con la codificación B y C tienen un máximo de 3 reglas mientras que los modelos generados con C5.0 tienen, en el caso de reglas genéricas, 5 reglas (3 para los ejemplos positivos y 2 para los negativos) y, en el caso de reglas específicas, 9 reglas (6 para los ejemplos positivos y 3 para los negativos). Además se observa que la diferencia entre el modelo específico y el genérico es de 4 reglas, pero este aumento significativo de la complejidad no repercute en un aumento sustancial de la calidad, lo que vuelve a poner de manifiesto la naturaleza compleja del problema, causada por la elevada heterogeneidad de los datos. Asimismo las reglas que componen los modelos presentan valores bajos de cobertura de ejemplos positivos y, en cuanto a la información que proporcionan las reglas,observando los valores V_p y V_n en la tabla de valores resultantes del entrenamiento, se puede destacar que los modelos describen mejor al grupo de ejemplos negativos. Por ejemplo, el modelo específico del algoritmo C5.0 reglas cubre 68 de los 103 ejemplos positivos del conjunto, es decir un 66 %, y cubre 127 de 139 ejemplos negativos, que representan al 91 % de este subconjunto. Por otra parte, de las 6 reglas del modelo específico que describen la clase "`avena mucha`" destaca por su cobertura (50 ejemplos) y confianza (78,8 %) la regla nº 4 (`P = alto AND K = bajo`). Cabe destacar que la precondición relativa al potasio aparece también en el modelo genérico, en concreto en la regla nº 3 (`K = bajo`) con una cubriendo un total de 55 ejemplos y una confian-

Algoritmo	Vp	Fn	Vn	Fp	Exactitud (%)	Confianza (%)
C5.0 reglas (espcf.)	82	102	67	3	58,7	96,5
C5.0 reglas (gener.)	77	107	66	4	56,3	95,1
$\texttt{AGLearner}_{(+espcf.)}(C_3)$	99	85	60	10	62,60	90,83
$\texttt{AGLearner}_{(+gener.)}(B_1)$	103	81	51	19	60,63	84,43

*Tabla 11.*XVII: Resultados de Clementine© con avena en Barcelona (validación)

za del 75,4 %. Esta precondición también aparece como importante en el modelo $A_2{:}r_5$ (sección 11.2.2) generado con la metodología propuesta. En general, a partir de los modelos generados con los diferentes algoritmos (comerciales y desarrollado) es fácil intuir la heterogenidad de los datos de Barcelona, sobre todo para la clase de ejemplos positivos, ya que se extraen pocas reglas de gran cobertura y muchas reglas específicas o que describen pocos ejemplos.

Conclusiones finales del estudio comparativo

Los modelos resultantes de la aplicación de los algoritmos incluidos en la herramienta Clementine©, en general, tienen peor exactitud respecto a los modelos suministrados por la herramienta desarrollada `AGlearner + SQLdeco.dll`, si comparamos los valores de los parámetros de calidad de modelos con el mismo número de reglas.

La diferencia fundamental estriba en en el método de construcción de los modelos. En el caso de la generación de árboles, se parte siempre de un nodo raíz que determina el orden de concatenación de las relaciones impidiendo que se puedan construir y evaluar todas las posibles combinaciones de relaciones sobre todos los atributos. De hecho, la forma del algoritmo C5.0 que mejor funciona realiza una poda evitando que las reglas partan siempre del nodo raíz conservando el orden entre variables. En contraste la herramienta desarrollada permite, a través de la búsqueda genética, una exploración no sesgada del espacio de búsqueda (espacio formado por todas las reglas que se pueden construir a partir de un conjunto de variables y de sus dominios).

Diferencias fundamentales del método propuesto frente al algoritmo C5.0 son además: (1) La posibilidad de obtener modelos referidos únicamente a la

clase que se desea describir, (2) la determinación del grado de complejidad del modelo y (3) la selección de la sintaxis de las reglas según los operadores lógicos `AND/OR` y los de comparación ($\neq$, $=$) que se utilicen.

Por otro lado, y a pesar de que la calidad de los modelos extraídos con la herramienta desarrollada (`AGlearner + SQLdeco`) es mayor en la mayoría de los casos estudiados, el método presenta tiempos de cómputo en la generación de modelos más altos que los algoritmos de Clementine© utilizados. Mientras que los algoritmos de Clementine generan un modelo en minutos, la herramienta desarrollada pueden requerir varios días para encontrar el máximo global cuando el modelo resulta ser muy complejo. Este "inconveniente" es admisible cuando los conjuntos de entrenamiento son relativamente grandes, cuentan con un número elevado de variables y/o se utilizan combinaciones de operadores complejas que dan más expresividad al modelo elevando la capacidad del sistema para describir clases con mayor calidad. Un espacio de búsqueda tan complejo como el descrito es el que nos podemos encontrar en los problemas de medio ambiente especialmente si se tienen en cuenta el tiempo y espacio en el modelo. Por tanto la propuesta hecha con este trabajo de tesis para el análisis de problemas medioambientales se adapta mejor que otros algoritmos de extracción de conocimiento, tanto a situaciones con espacios de búsqueda pequeños como a casos en los que el espacio de búsqueda es infinito. Además, el funcionamiento a partir de la integración de un conjunto de módulos hace que la herramienta implementada sea flexible, permitiendo por ejemplo incorporar nuevas funciones de calidad, establecer la estructura sintáctica de los modelos y/o determinar el nivel de complejidad deseado.

Capítulo 12

Conclusiones, contribuciones e investigación futura

12.1. Conclusiones

El trabajo de tesis descrito en esta memoria propone una metodología de análisis genérica que permite la generación de modelos basados en reglas a partir de un conjunto de observaciones. A través del estudio de un problema agrícola clásico, como es el control de malas hierbas, se ha diseñado y desarrollado un sistema de extracción de conocimiento en el que se abordan cada una de las etapas del proceso de inducción propuesto, desde la recogida de datos georeferenciados en campo hasta la obtención de los modelos explicativos de la densidad de los rodales de mala hierba en el cultivo en función de variables edáficas o medioambientales de esas zonas.

Cada una de las herramientas desarrolladas, así como la metodología propuesta, se puede aplicar a un amplia gama de problemas medioambientales, fundamentalmente cuando el objetivo sea el de modelar relaciones causa-efecto o asociaciones entre determinados factores y eventos.

De la investigación realizada en esta tesis se extraen dos tipos de aportaciones, las relacionadas con la metodología propuesta y aquellas relativas al estudio de un caso, verificando las ventajas del método.

La aportaciones relacionadas con la metodología propuesta son:

- La propuesta de un método de análisis de datos estructurado de acuerdo

con las pautas de un proceso de descubrimiento de conocimiento en bases de datos (*KDD-Knowledge discovery in databases*) para la extracción de conocimiento útil, mediante la generación e interpretación de modelos de relaciones o patrones existentes en un conjunto de datos. En consecuencia, para cada una de las etapas del proceso KDD se han diseñado y desarrollado el siguiente conjunto de herramientas orientadas fundamentalmente al análisis de datos y muestras medioambientales:

1. Herramienta de ayuda a la adquisición georeferenciada de datos en campo.
2. Herramienta de preprocesamiento de los datos, que permite depurar y preparar los datos para la etapa posterior de análisis o generación de modelos.
3. Herramienta de extracción de reglas (modelo heurístico) siguiendo el paradigma de aprendizaje basado en ejemplos y utilizando un algoritmo genético como método de búsqueda del modelo que mejor se ajusta o explica los datos.
4. Herramienta de evaluación de los modelos descubiertos, para tareas de clasificación y/o predicción; y visualización de resultados con gráficos bidimensionales o mapas que representan la situación real frente a la estimada por el modelo.

- En cuanto al método de generación de los modelos, la propuesta se plantea como una alternativa a la inferencia estadística y permite explorar otros aspectos de un conjunto de observaciones. De gran utilidad en aquellos casos en los que no se cumplen las hipótesis estadísticas (homocedasticidad, linealidad, normalidad, etc.) y/o que las transformaciones sobre los datos para verificar estas hipótesis hacen que los modelos obtenidos sean difíciles de interpretar.

- Se ha desarrollado una herramienta de transformación de reglas de tipo `Si-Entonces` a consultas en lenguaje SQL lo que ha permitido plantear la

operación de contraste de modelos como una consulta a una base de datos relacional gestionada a través de un Sistema de Información Geográfica (SIG), en la que se encuentran almacenados los datos a analizar (obtenidos en la etapa de muestreo en campo).

- El algoritmo genético se presenta como procedimiento de búsqueda del mejor modelo, facilitando el desarrollo de una herramienta escalable y genérica que permite una exploración adecuada de espacios de soluciones muy grandes, evitando que el proceso de búsqueda caiga en máximos(o mínimos)locales.

- La implementación es fácilmente adaptable a una gran gama de problemas cuyo objetivo sea la obtención de modelos de descripción, clasificación y/o predicción. Además, la herramienta se puede configurar y personalizar, en términos de complejidad y expresividad de las reglas, atendiendo a las necesidades del usuario.

Las aportaciones relativas al estudio del caso del control de malas hierbasse resumen en los siguientes puntos:

- El procedimiento de generación de modelos desarrollado ha sido ensayado en una serie de experimentos, destacando como resultado que los conjuntos de reglas descubiertos poseen alta fiabilidad y exactitud.

- Durante la experimentación se ha demostrado que a medida que aumenta la complejidad de los modelos, la exactitud y la confianza es cada vez mayor, siendo las reglas que componen cada conjunto de mayor especificidad. Por lo que el algoritmo genético, o proceso de búsqueda, converge hacía el máximo de la función de calidad (fitness).

- Los resultados de la comparación del procedimiento desarrollado frente a soluciones comerciales, que también extraen modelos basados en reglas (C&RT, C5.0), muestran que los modelos generados con la aproximación que aquí se propone son de mayor calidad, es decir cubren el conjunto de ejemplos de entrada con una mayor exactitud y confianza.

- En general, la densidad de avena en los campos de la Comunidad de Madrid es mayor en áreas donde la cantidad relativa de materia orgánica es media o alta y la proporción de limo es baja o media, con un confianza del 77 %, cubriendo el 75 % de los ejemplos positivos. Además, con un 94 % de confianza y cubriendo 58 % de los ejemplos positivos, se han obtenido modelos que relacionan una mayor cantidad de mala hierba con texturas más gruesas, con proporciones de arena medias o altas, cantidad de materia orgánica media o alta y cantidad de fósforo baja.

- Entre los modelos descubiertos no existe ninguna regla que por si sola explique los datos de todos los campos de la Comunidad de Madrid muestreados, lo que refuerza el hecho de que no se hayan encontrado relaciones directas entre las variables con métodos estadísticos convencionales y justifica nuevamente la importancia de desarrollar métodos alternativos que permitan descubrir modelos fácilmente interpretables capaces de describir relaciones complejas entre las variables seleccionadas.

- Los modelos descubiertos que relacionan la abundancia de avena con zonas de menor cantidad de fósforo muestran la probable existencia de fenómenos de competencia, ya que el fósforo es un componente fundamental en el desarrollo de las gramíneas, como son la avena y el trigo.

- Los modelos descubiertos con los datos de avena en el campo de Barcelona son más complejos, comparativamente, que los inducidos a partir de los datos de Madrid, es decir los modelos de avena en Barcelona están formados por un mayor número de reglas y presentan una calidad similar a los modelos de Madrid, lo que es indicativo de una parcela muy heterogénea.

- Los modelos que describen la avena en el campo de Barcelona son diferentes de los que describen esta mala hierba en las parcelas estudiadas en la Comunidad de Madrid, lo que podría llevar a pensar que el comportamiento de esta especie es distinto según la zona en la que nos encontremos, algo lógico máxime si tenemos en cuenta que hay factores que no se han

considerado y son determinantes en el desarrollo de los cultivos, como por ejemplo el clima. Ahora bien, también es importante puntualizar que en el caso de la Comunidad de Madrid se disponía de cinco fincas con una gestión de cultivo y topografía similar, mientras que en el campo de Barcelona sólo hemos tenido datos de una finca con una gestión de cultivo y topografía muy diferente a la de las parcelas de Madrid.

- El modelo obtenido para los datos de Barcelona, especifica con una confianza del 70 %, que la avena es más abundante en zonas de menor radiación solar acumulada, con una cantidad de potasio baja y donde exista menor biomasa de trigo. Esto último también se puede explicar como la consecuencia de un posible proceso de competencia entre el cultivo y la mala hierba.

- La experimentación con los datos de lolium del campo de Barcelona genera dos tipos de modelos: a) con una confianza del 55 al 60 % el modelo declara que el lolium aparece en mayor cantidad en zonas orientadas al norte, pendientes medias de 3 a 10 grados y cantidad de nitrógeno medio o bajo, y b) con 73 % de exactitud y 65 % de confianza el otro modelo, en el que destacan dos reglas, indica que las cantidades altas de lolium principalmente están relacionadas con zonas de menor radiación, pH bajo (menos básico), altas cantidades de limo y ausencia de paja; o con cantidades medias tanto de espigas como de grano de trigo y cantidades bajas de avena.

12.2. Investigación Futura

Esta tesis abre una nueva línea de investigación al análisis de datos relacionados con el entorno ecológico utilizando técnicas de aprendizaje automático con el objetivo de generar, a partir de un conjunto de observaciones, modelos heurísticos (basados en reglas). En este trabajo se ha desarrollado la base de una metodología que propone nuevos modos de procesamiento de datos,con

una gran potencialidad en futuras aplicaciones. A partir de esta propuesta inicial, son muchos los caminos que se pueden tomar como líneas futuras de investigación. Se resumen algunas en los siguientes puntos:

1. Una de las líneas de investigación más interesante, por su importancia en ecología, es la relativa a la incorporación en los modelos, de las relaciones espacio-temporales entre variables. La inclusión del espacio y del tiempo es fundamental en la detección y representación de cambios y tendencias en los procesos naturales. Por otra parte, agregar estas dimensiones aumentará considerablemente la complejidad de la búsqueda del modelo óptimo, aunque el sistema aquí desarrollado no requerirá grandes modificaciones para añadir este punto.

2. Otra línea importante de investigación está relacionada con mejorar la capacidad que tiene el mecanismo de representación de conocimiento basado en reglas de expresar adecuadamente sistemas y procesos que presenten gran incertidumbre en los datos, como sucede en la mayoría de los problemas medioambientales. En este caso la utilización de los principios de la lógica borrosa de Zadeh aportarían gran riqueza a la representación y permitirían modelar adecuadamente la incertidumbre presente; por ejemplo, en datos recogidos a partir de cualquier dispositivo de medida. En este caso estaríamos hablando de lo que se conocen como "*Genetic Fuzzy Systems*" cuyo estudio es parte del área de investigación conocida como "*softcomputing*".

3. El desarrollo de la herramienta de generación de modelos también requiere la implementación y comparación de nuevas funciones de evaluación de modelos, utilizadas por otros algoritmos y métodos, como pueden ser la ganancia de la información, el coeficiente de *Gini*, etc. Además es conveniente estudiar la incorporación en la función de calidad de un término que contemple la complejidad de los modelos, de modo que a coberturas iguales asigne un factor de calidad mayor a los modelos más simples, es decir a modelos con menor número de reglas y/o con antecedentes más

cortos (menos condiciones) en las reglas.

4. Aprovechando las ventajas del lenguaje de consulta SQL, empleado para la generación y evaluación de reglas, podría resultar interesante razonar de forma numérica, permitiendo que sea el algoritmo genético el que detecte los mejores umbrales que determinan las categorías, evitando así la etapa de categorización previa.

Bibliografía

REFERENCIAS

V. ADAMCHUK, J. HUMMEL, M. MORGAN & S. UPADHYAYA. 2004. On-the-go soil sensors for precision agriculture. *Computers and Electronics in Agriculture*, 44(1):71–91.

C. ADREASEN, J. STREIBIG & H. HASS. 1991. Soil properties affecting the distribution of 37 weed species in danish fields. *Weed Reseach*, 31:181–187.

R. AGRAWAL, T. IMIELINSKI & A. SWAMI. Mining association rules between set of items in large database. En P. Buneman & S. Jajodia (eds.), *Proceeding the 1993 ACM SIGMOD International Conference on Management of Data*, páginas 207–216, Washington DC, EE UU, 26–28 1993. URL `citeseer.nj.nec.com/agrawal93mining.html`.

S. ALONSO, B. DÍAZ, K. CANTILLO & A. RIBEIRO. Informe técnico de ag learner. Informe Técnico Dpto. Sistemas. 2002/IT/05, Instituto de Automática Industrial - CSIC, Julio 2002.

C. ANDREASEN, M. RUDEMO & S. SEVESTRE. 1997. Assessment of weed density on early stage by use of image processing. *Weed Research*, 37:5–18.

L. ARDIACA. Distribució espacial de les infestacions de margall (lolium rigidum gaud.) i de les pèrdies de collita en un camp de blat a la comarca de lánoia: relació amb els factors edàfics. Tesis (Master), Escola tecnica superior dénginyeria (ETSEA). Universitat de Lleida (UdL), marzo 2002.

M. AUGE & M.ÑAGI. Estado del agua subterránea respecto a la comtaminación con agroquímicos en la plata, provincia de buenos aires. En *II congreso argentino de hidrogeología y IV seminario hispano-argentino sobre temas actuales de la hidrología subterránea*, Santa Fe, Argentina, 1999.

T. BÄCK, A. E. EIBEN & N. A. L. VAN DER VAART. An empirical study on GAs without parameters. En M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. J. Merelo & H.-P. Schwefel (eds.), *Parallel Problem Solving from Nature (PPSN V)*, número 1917 En Lecture Notes in Computer Science, páginas 315–324. Springer, Berlin, 2000.

P. BALDI, S. BRUNAK, Y. CHAUVIN, C. ANDERSEN & H.ÑIELSEN. 2000. Assessing the accuracy of prediction algorithms for classification: An overview. *Bioinformatics*, 16(5):412–424.

J. BARREIRO. Un modelo biológico en la informática: Los algoritmos genéticos. En A. Pazos (ed.), *Redes de neuronas artificiales y algoritmos genéticos*, chapter 2, páginas 29–54. Servicio de Publicacións: Universidade de A coruña, 1996.

J. BARROSO. *Tratamientos localizados contra poblaciones de Avena Sterillis L. presentes en cultivos de cereal en la región centro de España.* Tesis Doctoral, Escuela técnica superior de ingenieros agrónomos (Universidad Politécnica de Madrid), 2004.

J. BARROSO, C. FERNÁNDEZ-QUINTANILLA, D. RUIZ, P. HERNAIZ & L. REW. 2004. Spatial stability of *Avena sterilis* ssp. ludoviciana populations under annual applications of low rates of imazamethabenz. *Weed Research*, 44(3):178.

J. BARROSO, D. RUIZ, C. FERNÁNDEZ-QUINTANILLA, P. HERNAIZ, A. RIBEIRO, B. DÍAZ, B. MAXWELL & L. REW. 2005. Comparison of sampling methodologies for site specific management of sterile oat (*Avena sterilis*). *Weed Research*, 45:165–174.

J. BARROSO, D. RUIZ, C. FERNÁNDEZ-QUINTANILLA, A. RIBEIRO & B. DÍAZ. Comparison of various sampling methodologies for site specific sterile wild oat (avena sterilis) management. En G. Grenier & S. Blackmoore (eds.), *Proceedings of the Third European Conference on Precision Agriculture,(3rdECPA)*, páginas 575–579. Agro Montpellier, Montpellier, Francia, 2001. ISBN 2-900792-13-4.

J. BASSETT. A study of generalization techniques in evolutionary rule learning. Tesis (Master), George Mason University (EE UU), 2002.

J. BENLLOCH & A. RODAS. Image processing techniques for determination of weeds in cereal. En F.Juste, G.Andreu, J.M.Valiente & J.V.Benlloch (eds.), *Proceedings of the International Workshop on Robotics and Automated Machinery for Bio-Productions*, páginas 195–200. European Society of Agricultural Engineers, 1997. ISBN 8484986594.

S. BLACKMORE. 1994. Precision farming: An introduction. *Outlook on Agriculture*, 23(4):275–280.

C. BOJARCZUK, H. LOPES & A. A. FREITAS. Data mining with constrained-syntax genetic programming: Applications in medical data sets. En *Proceeding of Intelligent Data Analysis in Medicine and Pharmacology - a workshop at MedInfo-2001*, London, September 2001.

R. BONGIOVANNI & J. LOWENBERG-DEBOER. 2004. Precision agriculture and sustainability. *Precision Agriculture*, 5(4):359–387.

G. BOOLE. 1848. The calculus of logic. *Cambridge and Dublin Mathematical Journal*, 3:183–198.

J. BOSQUE-SENDRÁ. 1998. *Sistemas de Información Geográfica.* Rialp, Madrid.

H. BOSTRÖM & L. ASKER. Combining divide-and-conquer and separate-and-conquer for efficient and effective rule induction. En *Proceeding of the 9th International workshop on Inductive Logic Programming (LNAI, 1634)*, páginas 33–43. springer, 1999.

M. BOULLE. 2004. Khiops: A statistical discretization method of continuous attributes. *Machine learning*, 55:53–69.

H. BOURENNANE, B.ÑICOULLAUD, A. COUTURIER & D. KING. Assesment of spatial correlation between wheat yields and some phisical and chemical soil propierties. En J. Stafford & A. Werner (eds.), *Proceedings of the Fourth European Conference on Precision Agriculture,(4thECPA)*, páginas 149–157, Berlin, Alemania, 2003. Wageningen Academic publishers. ISBN 9076998213.

L. BREIMAN, J. FRIEDMAN, R. OLSHEN & C. STONE. 1984. *Classification and regression trees.* Wadsworth, Inc., Monterey, EE UU.

R. BRIVOT & J. MARCHANT. Segmentation of plants and weed using infrared images. En *Acta Horticulturae. Proceedings of Workshop Mathematical and Control Applications in Agriculture and Horticulture (II IFAC/ISHS)*, volume 406, páginas 165–172. ISHS, 1996. ISBN 9066057580.

C. CAMBARDELLA, T. MOORMAN, J.ÑOVAK, T. PARKIN, D. KARLEN, R. TURCO & A. KONOPKA. 1994. Field-scale variability of soil properties in central iowa soils. *Soil Science Soci. Americam Journal*, 58(5):1501–1511.

J. CARDINA, G. A. JOHNSON & E. MCCOY. 1995. Analysis of spatial distribution of common lambsquarters in notill soybean. *Weed Science*, 43: 258–269.

J. CARDINA, G. A. JOHNSON & D. SPARROW. 1997. The nature and consequence of weed spatial distribution. *Weed Science*, 45:364–373.

H. CHEN & L. SHE. Inductive query by examples (IQBE): A machine learning approach. En J. F. Nunamaker & R. H. Sprague (eds.), *Proceedings of the 27th Annual Hawaii International Conference on System Science*, volume 3, páginas 428–437, Los Alamitos, USA, 1994. IEEE Computer Society Press. ISBN 0-8186-5070-2.

Y. CHEVALEYRE & J.-D. ZUCKER. A framework for learning rules fro multiple instance data. En *Proceeding of the 12th European conference on Machine Learning (ECML-01)*, páginas 46–60, Freiburg, Alemania, 2001. springer-Verlag.

M. CHMIELEWSKI & J. GRZYMALA-BUSSE. 1996. Global discretization of continuous attributes as preprocessing for machine learning. *International journal of approximate reasoning*, 15:319–331.

S. CHRISTENSEN & T. HEISEL. 1998. Patch spraying using historical, manual and real time monitoring of weeds in cereals. *Zeitschrift für Pflanzenkrankheiten und Pflanzenschtz, soderheft*, XVI:257–265.

S. CHRISTENSEN, E.ÑORDBO, T. HEISEL & A. WALTER. 1999. Overview of developments in precision weed management, issues of interest and future directions being considered in europe. *Precision weed management in crops and pastures*, páginas 3–13.

P. CLARK & T.ÑIBLETT. 1989. The cn2 induction algorithm. *Machine Learning*, 3:261–283. URL `citeseer.nj.nec.com/clark89cn.html`.

S. CLAY, G. LEMS, F. FORCELLA & M. ELLSBURY. 1999. sampling weed spatial variability on a fieldguide scale. *Weed Science*, 47:674–681.

Clementine Data mining Sistem Version 6.0.2. Clementine SPSS Inc., copyright 1994-2000 edition, 2000.

E. CODD, S. CODD & C. SMALLEY. 1993. Providing OLAP to user analysis. *An IT Mandate. Libro blanco.*

W. COHEN. Fast effective rule induction. En *Proceeding of the 12th International Conference on Machine Learning*, páginas 115–123, 1995.

C. COLLIVER, B. MAXWELL, D. TYLER, D. ROBERTS & D. LONG. Georeferencing wild oat infestations in small grains: accuracy and efficiency of three

weed survey techniques. En R. R. P.C. Roberts & W. Larson (eds.), *Proceedings 1996 3rd International Conference on Precision Agriculture*, páginas 453–463, Minneapolis, USA, 1996. American Society of Agronomy, Crop Science Society of America, and Soil Science Society of America.

G. COOPER. Theoretical modeling and biological laws. En *Proceedings del bienal meeting Philosophy of Science Association. Part I*, volume 6, páginas 28–35. Philosophy of Science Association, 1996.

R. COUSENS, P. BRAIN, J. O'DONOVAN & P. O'SULLIVAN. 1987. The use of biologically realistic equations to describe the effect of weed density and relative time of emergence on crop yield. *Weed Science*, 35:720–725.

R. COUSENS & A. CROFT. 2000. Weed populations and pathogens. *Weed Research*, 40(1):63–82.

G. CUSSANS. Strategic for weed control-a research's view. En *Proceedings 1980 British Crop Protection Conference - Weeds*, páginas 823–831. Sheffield Academic Press, Brighton, 1980.

M. DALE, A. THOMAS & E. JOHN. 1992. Environmental factors influencing management practices as correlates of weed community composition in spring seeded crops. *Canadian Journal of Botany*, 70:1931–1939.

K. A. DE JONG, W. M. SPEARS & D. F. GORDON. 1993. Using genetic algorithms for concept learning. *Machine Learning*, 13:161–188. Special issue on genetic algorithms.

M. H. DEGROOT. 1988. *Probabilidad y estadística.* Addison Wesley, Madrid.

H. DÉJEAN. 2002. Learning rules and their exception. *Machine learning research*, 2:669–693.

B. DÍAZ & A. RIBEIRO. Estudio del comportamiento de la señal DGPS en una retícula. problemas y soluciones para la relocalización. Informe Técnico Dpto. Sistemas. TR-11/00, Instituto de Automática Industrial - CSIC, 2000a.

B. DÍAZ & A. RIBEIRO. Estudio en estático de la señal de los DGPS: Lr3100 y lr2100. Informe Técnico Dpto. Sistemas. TR-10/00, Instituto de Automática Industrial - CSIC, 2000b.

B. DÍAZ & A. RIBEIRO. Manual de utilización COPLAS. configuration plan sampling. Informe Técnico Dpto. Sistemas. 2002/IT/06, Instituto de Automática Industrial - CSIC, Enero 2002.

B. DÍAZ & A. RIBEIRO. Manual de utilización FIEL. field data collection subsystem. Informe Técnico Dpto. Sistemas. 2005, Instituto de Automática Industrial - CSIC, Enero 2005.

B. DÍAZ, A. RIBEIRO, M. G. ALEGRE & D. GUINEA. Using geomedia objects to develop a userfriendly GIS/GPS-based system for field sampling tasks. En *GeoSpatial World 2002*. Intergraph S.A, 2002.

B. DÍAZ, A. RIBEIRO, R. BUENO, D. GUINEA, J. BARROSO, D. RUIZ & C. FERNÁNDEZ-QUINTANILLA. 2005. Modelling wild-oat density in terms of soil factors: A machine learning approach. *Precision Agriculture*, 6(2): 213–228. ISSN 1385-2256 (artículo) 1573-1618 (on-line).

B. DÍAZ, A. RIBEIRO, D. RUIZ, J. BARROSO & C. FERNÁNDEZ-QUINTANILLA. A genetic algorithm approach to discover complex associations between wild-oat density and soil properties. En J. Stafford & A. Werner (eds.), *Proceedings of the Fourth European Conference on Precision Agriculture,(4th ECPA)*, páginas 149–157. Wageningen Academic publishers, Berlín, Alemania, 2003. ISBN 9076998213. en ISI report.

B. DÍAZ-DIEZ & A. MORILLAS. 2004. Minería de datos y lógica difusa. una aplicación al estudio de la rentabilidad económica de las empresas agroalimentarias en andalucía. *Estadística española*, 46(157):409–430.

J. DIELEMAN & D. MORTENSEN. 1999. Characterizing the spatial pattern of *Abutilon theophrasti* seedling patches. *Weed Research*, 39(6):455–467.

J. DIELEMAN, D. MORTENSEN, D. BUHLER, C. CAMBARDELLA & T. MOORMAN. 2000a. Identifying associations among site properties and weed species abundance I. multivariate analysis. *Weed Science*, 48:567–575.

J. DIELEMAN, D. MORTENSEN, D. BUHLER & R. FERGUSON. 2000b. Identifying associations among site properties and weed species abundance II. hypothesis generation. *Weed Science*, 48:576–587.

T. G. DIETTERICH. Limitations of inductive learning. En *Proceedings of the Sixth International Workshop on Machine Learning*, páginas 124–128. Morgan Kaufmann, CA, EE UU, 1998. URL `http://web.engr.oregonstate.edu/~tgd/publications/ml89-limits.ps.gz`.

W. DONALD. 1994. Geostatistics for mapping weeds, with canada thistle (*Circium arvense*) patch as a case study. *Weed Science*, 42:648–657.

J. DOUGHERTY, R. KOHAVI & M. SAHAMI. Supervised and unsupervised discretization of continuous features. En *International Conference on*

Machine Learning, páginas 194–202, 1995. URL `citeseer.ist.psu.edu/dougherty95supervised.html`.

R. EARL, P. WHEELER, B. BLACKMORE & R. GODWIN. 1996. Precision farming: The management of variability. *Landwards*, 51:18–23.

A. E. EIBEN & J. E. SMITH. 2003. *Introduction to Evolutionary Computing*. Springer. ISBN 3-540-40184-9.

I. FARKAS. 2003. Special issue: artificial intelligence in agriculture. *Computers and Electronics in Agriculture*, 40:1–3.

U. FAYYAD, D. HAUSSLER & P. STOLORZ. Kdd for science data analysis: Issues and examples. En *Knowledge Discovery and Data Mining*, páginas 50–56, 1996a.

U. FAYYAD, G. PIATETSKY-SHAPIRO & P. SMYTH. 1996b. From data mining to knowledge discovery in databases. *AI Magazine*, 17(3):37–54.

C. FERNÁNDEZ-QUINTANILLA. 2003. Agricultura de precisión. factores para su implementación. *Vida Rural*.

C. FERNÁNDEZ-QUINTANILLA, J. BARROSO & D. RUIZ. Diseño de un programa de tratamientos de precisión para el control de *Avena Sterilis* en cereales. En *Congreso 2001 de la Sociedad Española de Malherbología*. Actas, 2001.

C. FERNÁNDEZ-QUINTANILLA, L.ÑAVARRETE, C. TORNER & M. S. D. ARCO. *Avena sterilis* en cultivos de cereales. En X. Sans & C. Fernández-Quintanilla (eds.), *Biología de las Malas Hierbas de España*, páginas 4–17. Phytoma, Valencia, España, 1997.

J. FERREIRA. 1995. Ecowin an object-oriented ecological model for aquiatic ecosystems. *Ecological Modelling*, 79:21–34. URL `http://www.minerva.uevora.pt/simposio/comunicacoes/Emilio_Vigo/Model-Lab2.html`.

A. H. FIELDING. *How should acuracy be measured*, chapter 8, páginas 209–225. Kluwer Academic Publishers, 1999a. ISBN 0412841908.

A. H. FIELDING. 1999b. *Machine Learning Methods for Ecological Applications*. Kluwer Academic Publishers. ISBN 0412841908.

P. FLACH. 2001. On the state of the art in machine learning: A personal review. *Artificial Intelligence*, 131:199–222.

M. FORTIN, M. DALE & J. HOEF. Spatial analysis in ecology. En A. H. El-Shaarawi & W. W. Piegorsch (eds.), *Encyclopedia of Environmetrics*, volume 4, páginas 2051–2058. John Wiley and Sons Ltd, Chichester, 2002. ISBN 0471 899976.

W. FRAWLEY, G. PIATETSKY-SHAPIRO & C. MATHEUS. Knowledge discovery in databases: An overview. En Piatetsky-Shapiro & W. J. Frawley (eds.), *Knowledge Discovery in Databases*, páginas 1–27. AAAI/MIT Press, 1991.

R. FRECKLETON & A. WATKINSON. 2002. Are weed population dynamics chaotic? *Journal of applied Ecology*, 39:699–707.

J. FREDRIKSSON & T. TURUJLIJA. Classification of sugarbeet plants based on contextual information. Proyecto de master en computer systems engineering, Halmstad University, 1998.

A. FREITAS & S. LAVINGTON. Using sql primitives and parallel db servers to speed up knowledge discovery in large relational databases. En R. Trappl (ed.), *Cybernetics and Systems '96: Proc 13th European Meeting on Cybernetics and Systems Research*, páginas 955–960, Vienna, Austria, Abril 1996. ISBN 3-85206-133-4.

A. A. FREITAS. November 2001. Understanding the crucial role of attribute interaction in data mining. *Artificial Intelligence Review*, 16(3):177–199. ISSN 0269-2821.

A. A. FREITAS. 2002a. *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Springer-Verlag, EE UU. ISBN 3-540-43331-7.

A. A. FREITAS. Evolutionary computation. En W. Klosgen & J. Zytkow (eds.), *Handbook of Data Mining and Knowledge Discovery*. Oxford University Press, August 2002b. ISBN 0195118316.

J. FRÖHLING. the necessity of weed control in winter cereals. En *Proceedings 1980 British Crop Protection Conference - Weeds*, páginas 763–769. Sheffield Academic Press, Brighton, 1980.

J. FÜRNKRANZ. 1999. Separate-and-conquer rule mining. *Artificial Intelligent review*, 13(1):3–54.

J. FURNKRANZ & P. FLACH. An analysis of rule evaluation metrics. En *Proc. 20th International Conference on Machine Learning (ICML'03)*, páginas 202–209. AAAI Press, Enero 2003. ISBN 1-57735-189-4.

L. GARCÍA-PÉREZ, M. GARCÍA-ALEGRE, J. MARCHANT & T. HAGUE. Dynamic threshold selection for image segmentation based upon a performance criterion. En G. Grenier & S. Blackmoore (eds.), *Proceedings of the Third European Conference on Precision Agriculture,(3rd ECPA)*. Agro Montpellier, Montpellier, Francia, 2001. ISBN 2-900792-13-4.

L. GARCÍA-PÉREZ, J. MARCHANT, T. HAGUE & M. GARCÍA-ALEGRE. Fuzzy decision system for threshold selection to cluster cauliflower plant blobs from field visual images. En *Proceedings of the Conference on Three-Dimensional Image Capture and Applications III (SPIE 2000)*, páginas 23–28. SPIE, Orlando, EE UU, 2000. ISBN 0-8194-3576-7.

R. GERHARDS, D. WYSE-PETER, D. MORTENSEN & G. JONHSON. 1997. Characterizing spatial stabitity of weed populations using interpolated maps. *Weed Science*, 45:108–119.

D. GOLDBERG. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison Wesley Professional publishers, USA.

R. GONZÁLEZ-PONCE & I. SANTÍN. 2001. Competitive ability of wheat cultivars with wild oats depending on nitrogen fertilization. *Agronomy journal*, 21:119–125.

F. GRAY. Pulse code communication. Patente (EE UU): 2 632 058, Marzo 1953.

. W. GRZYMALA-BUSSE & J. STEFANOWSKI. 2001. Three discretization methods for rule induction. *International journal of intelligent systems*, 16: 29–38.

O. GUNTHER. 1998. *Environmental Information System.* Springer Verlag, Europa. ISBN 3540609261.

J. HAN, Y. FU, W. WANG, K. KOPERSKI & O. ZAIANE. Dmql: A data mining query language for relational databases. En *SIGMOD'96 Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD'96)*, Montreal, Canada, 1996. URL `citeseer.ist.psu.edu/han96dmql.html`.

J. HAN & M. KAMBER. 2001. *Data Mining: Concepts and Techniques.* Morgan Kaufmann, CA, EE UU. ISBN 1-55860-489-8.

D. HAND. 1997. *Construction and Assessment of Classification Rules.* John Wiley and Sons, New-York.

Y. HASHIMOTO. 1997. Introduction: Applications of artifical neural networks and genetic algorithms to agricultural systems. *Computers and Electronics in Agriculture*, 18:71–72.

T. HEISEL, C. ANDREANSEN & A. ERSBOLL. 1996. Sampling weed spatial variability on a fieldguide scale. *Weed Research*, 36:325–337.

T. HEISEL, S. CHRISTENSEN & A. WALTER. 1999. Whole-field experiments with site-specific weed management. *Precision Agriculture*, páginas 759–768.

J. HEVESI, J. ISTOK & A. FLINT. 1992. Precipitation estimation in mountains terrain using multivariate geostatistics. part i: structural analysis. *Journal of applied meteorology*, 31(7):661–676.

J. HIPP, C. MANGOLD, U. GÜNTZER & G.ÑAKHAEIZADEH. Efficient rule retrieval and postponed restrict operations for association rule mining. En *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, páginas 52–65, 2002. URL `citeseer.ist.psu.edu/508281.html`.

K. M. HO & P. D. SCOTT. Zeta: A global method for discretization of continuous variables. En *Knowledge Discovery and Data Mining*, páginas 191–194, 1997. URL `citeseer.ist.psu.edu/ho97zeta.html`.

W. HOFF, R. MICHALSKI & R. STEPP. Induce 2: A program for learning structural descriptions from examples. Reports of the Intelligent Systems Group ISG 83-4, UIUCDCS-F-83-904, Department of Computer Science, Department of Computer Science, University of Illinois, Urbana, mayo 1983.

J. HOLLAND. 1975. *Adaptation in Natural and Artificial Systems*. University of Michigan Press.

C. S. HOLLING. 2001. Understanding the complexity of economic, ecological, and social systems. *Ecosystems*, 4:390 405.

R. HOLTE. 1993. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11(1):63–90. ISSN 0885-6125.

J. HONG, I. MOZETIC & R. MICHALSKI. Aq15: Incremental learning of attribute-based descriptions from examples, the method and user's guide. Reports of the Intelligent Systems Group ISG 86-5, UIUCDCS-F-86-949, Department of Computer Science, Department of Computer Science, University of Illinois, Urbana, mayo 1986.

T. IMIELINSKI & H. MANNILA. 1996. A database perspective on knowledge discovery. *Commun. ACM*, 39(11):58–64. ISSN 0001-0782.

T. IMIELINSKI, A. VIRMANI & A. ABDULGHANI. 1999. Dmajor - application programming interface for database mining. *Data Mining and Knowledge Discovery*, 3(4):347–372.

J. JEFFERS. *Genetic Algorithms I: Ecological application of the BEAGLE and GAFFER genetic algoritmhs*, chapter 4, páginas 107–121. Kluwer Academic Publishers, 1999. ISBN 0412841908.

G. H. JOHN. *Enhancements to the Data Mining Process.* Phd thesis, Stanford University, EE UU, 1997.

J. B. JOHNSON & K. S. OMLAND. 2004. Model selection in ecology and evolution. *Trends in Ecology and Evolution*, 19:101–108.

M. JURADO-EXPÓSITO, F. LÓPEZ-GRANADOS, L. GARCÍA-TORRES, A. GARCÍA-FERRER, M. S. DE LA ORDEN & S. ATENCIANOD. 2002. Multi-species weed spatial variability and site-specific management maps in cultivated sunflower. *Journal of Agricultural Engineering Research*, 51(3): 319–328.

S. KALOGIROU. 2002. Expert systems and gis: an application of land suitability evaluation. *Computers, environment and urban systems*, 26:89–112.

M. KANEVSKI, R. PARKIN, A. POZDNUKHOV, V. TIMONIN, M. MAIGNAN, V. DEMYANOV & S. CANU. 2004. Environmental data mining and modeling based on machine learning algorithms and geostatistics. *Environmental Sciences and Artificial Intelligence. Environmental Modelling and Software*, 19:845–855.

R. KERRY & M. OLIVER. Comparing spatial structures in soil properties and ancillary data by using variograms. En G. Grenier & S. Blackmoore (eds.), *Proceedings of the Third European Conference on Precision Agriculture,(3^{r}dECPA)*, páginas 413–418, Montpellier,Francia, 2001. Agro Montpellier. ISBN 2-900792-13-4.

H.-P. KRIEGEL, M. POTKE & T. SEIDL. Managing intervals efficiently in object-relational databases. En *Proceedings of the Twenty-Sixth International Conference on Very Large Data Bases*, páginas 407–418, 2000. URL `citeseer.ist.psu.edu/kriegel00managing.html`.

P. KROHMANN, C. TIMMERMANN, R.GERHARDS & W. KÜHBAUCH. Variation of weed populations in a crop rotation and in continous maize. implicacitions for the definition of weed patches. En J. Stafford & A. Werner (eds.), *Proceedings of the Fourth European Conference on Precision*

Agriculture,(4thECPA), páginas 587–592. Wageningen Academic publishers, Berlin, Alemania, 2003. ISBN 9076998213.

M. J. KROPFF, J. WALLINGA & L. LOTZ. Modelling for precision weed management. En J. Lake, G. Bock & J. Goode (eds.), *Precision agriculture: spatial and temporal variability of environmental quality*, páginas 182–204. John Wiley and Sons, 1997. ISBN 0471 97455 2.

D. W. KRUEGER, G. G. WILKERSON, H. D. COBLE & H. J. GOLD. 2000. An economic analysis of binomial sampling for weed scouting. *Weed Science*, 48:53–60.

R. LEVINS. 1968. *Evolution in Changing Environments: Some Theoretical Explorations.* Princeton University Press. ISBN 0-691-08062-3.

T. LINDGREN & H. BOSTRÖM. Classification with intersecting rules. En *Proceeding of the 13th conference on Algorithmic Theory*, páginas 395–402. springer, 2002.

T. LINDGREN & H. BOSTRÖM. Resolving rule conflicts with duble induction. En *Proceeding of the 3th International Symposium on Inteligent Data Analysis*, páginas 60–67. springer, 2003.

J. LINDQUIST, J. DIELEMAN, D. MORTENSEN, G. JOHNSON & D. WYSE-PESTER. 1998. Economic importance of managing spatially heterogeneous weed populations. *Weed Technology*, 12:7–13.

W. LOH & Y. SHIH. 1997. Split selection methods for classification trees. *Statistica Sinica*, 7:815–840.

O. MARON & T. LOZANO-PÉREZ. 1998. A framework for multiple-instance learning. *Neural Information processing systems*, 10:570–576.

L. MARTIN-CHEFSON, M. CHAPRON, S. PHILIPP, L. ASSEMAT & P. BOISSARD. A two dimensional method for recognising weeds from multiband image processing. En J. V. Stanfford (ed.), *Proceedings of the 2nd European Conference on Precision Agriculture*, página 473483. Sheffield Academic Press, Inglaterra, 1999.

B. MARY, N. BEAUDOIN, J. MACHET, C. BRUCHOU & F. ARIES. Characterization and analysis of soil variability within two agricultural fields: the case of water and mineral n profiles. En G. Grenier & S. Blackmoore (eds.), *Proceedings of the Third European Conference on Precision Agriculture,(3^{r}dECPA)*, páginas 431–436, Montpellier, Francia, 2001. Agro Montpellier. ISBN 2-900792-13-4.

M. MATEOS. 1967. El triángulo de texturas de suelos. *Cimbra*, 26:30.

B. MAXWELL. 1999. (extensión de) my view: A perspective on ecologically based pest management. *Weed Science*, 47:129.

B. MAXWELL, A. BUSSAN, E. LUSCHEI, L. V. WYCHEN, D. BUSCHENA & D. GOODMAN. Precision weed control in wheat and spring barley, 1998. URL `http://weedeco.msu.montana.edu/`.

MCGRAW-HILL. Colombia to spray coca cops with stronger herbicide. Informe técnico, McGraw-Hill Higher Education, Junio 1998. URL `http://www.mhhe.com/biosci/pae/es_map/index.mhtml`. Noticias medioambientales ON-LINE.

M. MEHTA, R. AGRAWAL & J. RISSANEN. SLIQ: A fast scalable classifier for data mining. En *Extending Database Technology*, páginas 18–32, 1996.

R. MELCHIORI, F. GARCÍA & H. ECHEVERRÍA. 2001. Variabilidad espacial en algunas propiedades del suelo: I asociación con las variaciones en el rendimiento del trigo. *Agricultura de precisión*. URL `http://www.agriculturadeprecision.org/mansit/VariabilidadEspacial.htm`.

R. MEO. Optimization of a language for data mining. En *Proceedings of the 2003 ACM symposium on Applied computing*, páginas 437–444. ACM Press, 2003. ISBN 1-58113-624-2.

R. MEO, G. PSAILA & S. CERI. A new sql-like operator for mining association rules. En *Proceedings of the 22th International Conference on Very Large Data Bases*, páginas 122–133. Morgan Kaufmann Publishers Inc., 1996. ISBN 1-55860-382-4.

Y. MIAO, P. ROBERT & D. MULLA. Geostatistical analysis of soil properties and corn quality. En J. Stafford & A. Werner (eds.), *Proceedings of the Third European Conference on Precision Agriculture,(3rdECPA)*, páginas 417–423, Berlin, Alemania, 2003. Wageningen Academic publishers. ISBN 9076998213.

Z. MICHALEWICZ. 1996a. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer Verlag. ISBN 3540606769.

Z. MICHALEWICZ. 1996b. Heuristic methods for evolutionary computation techniques. *Jornual of Heuristics*, 1(2):177–206.

Y. MING CHEN. 1997. Managenement of water resources using improved genetic algorithms. *Computers and electronic in agriculture*, 18:117–127.

T. MITCHELL. 1997. *Machine Learning.* McGraw-Hill Companies, Inc., USA. ISBN 0070428077.

S. MUGGLETON. 1990. Inverse entailment and progol. *New generation computing*, 13:245–286.

H. MURASE. 2000. Special issue: artificial intelligence in agriculture. *Computers and Electronics in Agriculture*, 29:1–2.

A.ÑEWELL & H. A. SIMON. 1972. *Human Problem Solving.* Englewood Cliffs (NJ) Prentice-Hall.

E.ÑODA, A. A. FREITAS & H. S. LOPES. Discovering interesting prediction rules with a genetic algorithm. En *Proceedings of the Congress on Evolutionary Computation*, volume 2, páginas 1322–1329. IEEE Press, 1999.

S. OFFICER, R. LASCANO, J. BOOKER & S. MAAS. Comparation of methods to extract correlations for canonical analysis of cotton yields. En J. Stafford & A. Werner (eds.), *Proceedings of the Fourth European Conference on Precision Agriculture,(4^{th}ECPA)*, páginas 95–101, Berlin, Alemania, 2003. Wageningen Academic publishers. ISBN 9076998213.

F. E. PETRY, B. P. BUCKLES, D. H. KRAFT, D. PRABHU & T. SADASIVAN. The use of genetic programming to build queries for information retrieval. En T. Bäck, D. Fogel & Z. Michalewicz (eds.), *Handbook of Evolutionary Computation*, páginas G2.1:1–G2.1:6. Oxford University Press, New York, 1997.

F. PIERCE & P.ÑOWAK. 1999. Aspects of precision agriculture. *Advances in Agronomy*, 67:1–85.

M. POCH, J. COMAS, I. ROGRÍGEZ-RODA, M. SÀNCHEZ-MARRÈ & U. CORTES. 2004. Designing and building real environmental decision support systems. *Environmental Modelling and software*, 19(9):857–873.

E. PRESS. 2003. A travÉs de un anÁlisis: 28 sustancias químicas en la sangre de una comisaria europea. *Diario El mundo.*

A. PÉREZ & A. RIBEIRO. An approximation to generic knowledge discovery in database systems. En *The First International Workshop on Machine Learning, Forecasting, and Optimization, MALFO96*, Madrid, 1996.

D. PYLE. 1999. *Data preparation for Data Mining.* Morgan Kaufmann, USA.

J. QUINLAN. 1990. Learning logical definitions from relations. *Machine Learning*, 5:239–266.

J. R. QUINLAN. Induction of decision trees. En J. W. Shavlik & T. G. Dietterich (eds.), *Readings in Machine Learning*. Morgan Kaufmann, 1986. Originally published in *Machine Learning* 1:81–106.

J. R. QUINLAN. C4.5: Programs for machine learning. En *Machine Learning*. Morgan Kaufmann, 1993.

F. RECKNAGEL. 2001. Applications of machine learning to ecological modelling. *Ecological Modelling*, 146:303–310.

J. RHOADES. Reducing salinization of soil and water by improving irrigation and drainage management. En *Prevention of Water Pollution by Agriculture and Related Activities*, volume 1, páginas 291–320. Actas de la Consulta de Expertos de la FAO (Water Report), 1993.

J. RHOADES, A. KANDIAH & A. MASHALI. 1992. *The use of saline waters for crop production - FAO irrigation and drainage paper 48*. FAO, Roma. ISBN 92-5-103237-8.

D. RIAÑO. Automatic construction of description rules. Tesis (Master), Department de Lleguatges i sistemes informàtcs. Universitat Politècnica de Calalunya, Noviembre 1997.

D. RIAÑO. Learning rules within the framework of environmental sciences. En *proceeding of ECAI'98 BESAI Workshop*, Universitat Rovira i Virgili, 1998.

D. RIAÑO & U. CORTÉS. 1997. Rule generation and compaction in the wwtp. *computación y sistemas*, 1(2):77–90.

A. RIBEIRO, B. DÍAZ & M. G. ALEGRE. Extracting fuzzy rules to describe weed infestation in terms of soils factors. En O.Ñasraoui & H. Frigui (eds.), *Proceedings of IEEE International conference on Fuzzy Systems*, páginas 1032–1037. IEEE, St. Louis, Missouri, EE UU, 2003. ISBN 0-7803-7811-3. en ISI report.

A. RIBEIRO, B. DÍAZ, M. G. ALEGRE, D. GUINEA, C. FERNÁNDEZ-QUINTANILLA, J. BARROSO & D. RUIZ. A GPS based system to aid in the acquisition of spatially structured field properties. En G. Grenier & S. Blackmoore (eds.), *Proceedings of the Third European Conference on Precision Agriculture,(3^{rd}ECPA)*, páginas 97–102, Montpellier, Francia, 2001. Agro Montpellier. ISBN 2-900792-13-4.

P. ROBERT. Precision agriculture: Research needs and status in usa. En J. V. Stanfford (ed.), *Proceedings of the 2nd European Conference on Precision Agriculture*, páginas 19–33. Sheffield Academic Press, Inglaterra, 1999.

G. ROBERTSON, E. PAUL & R. HARWOOD. 2000. Greenhouse gases in intensive agricultural contribution of individual gases to the radiactive forcing of the atmosphere. *Science*, 289:1922–1925.

D. RUIZ, J. BARROSO & C. FERNÁNDEZ-QUINTANILLA. Associations among soil properties and winter wild oat (avena sterilis l.) abundance in cereal fields. En 12^{th} *simposium European Weed Research Society (EWRS)*. Wageningen, 2002.

D. RUIZ, C. FERNÁNDEZ-QUINTANILLA & J. BARROSO. Dependendencia de los factores edáficos en la distribución espacial de avena loca (avena sterilis) en los campos de cultivo. En *Congreso 2001 de la Sociedad Española de Malherbología*. Actas, 2001.

L. RUIZ-MAYA. 2000. *Métodos estadísticos de investigación en las ciencias sociales: técnicas no paramétricas*. AC S. A. (ALFA CENTAURO).

T.-W. RYU & C. F. EICK. Deriving queries from examples using genetic programming. En E. Simoudis, J. W. Han & U. Fayyad (eds.), *The Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, páginas 303–306, Portland, Oregon, USA, 1996a. AAAI Press. ISBN 1-57735-004-9. URL `http://www.cs.uh.edu/~twryu/papers/kdd96.ps`.

T.-W. RYU & C. F. EICK. MASSON: discovering commonalties in collection of objects using genetic programming. En J. R. Koza, D. E. Goldberg, D. B. Fogel & R. L. Riolo (eds.), *Genetic Programming 1996: Proceedings of the First Annual Conference*, páginas 200–208, Stanford University, CA, USA, 1996b. MIT Press. URL `http://www.cs.uh.edu/~twryu/papers/gp96.ps`.

M. SALIM & X. YAO. 2002. Evolving sql queries for data mining. *Lecture Notes in Computer Science*, 2412:62–ff.

K. SATTLER & O. DUNEMANN. Sql database primitives for decision tree classifiers. En *Proceedings of the tenth international conference on Information and knowledge management*, páginas 379–386. ACM Press, 2001. ISBN 1-58113-436-3.

L. M. SCHMITT. 2000. Fundamental study: Theory of genetic algorithms. *Theoretical Computer Science*, 259:1–61. ISSN 0304-3975.

L. M. SCHMITT. 2004. Theory of genetic algorithms ii: models for genetic operators over the string-tensor representation of populations and convergence to global optima for arbitrary fitness function under scaling. *Theoretical Computer Science*, 310(1-3):181–231. ISSN 0304-3975.

A. SCHULTZ & R. WIELAND. 1997. The use of neural networks in agroecological modelling. *computers and electronics in Agriculture*, 18:73–90.

J. SCURSONI, R. BENECH-ARNOLD & H. HIRCHOREN. 1991. Demography of wild oat in barley crops: effect of crop, sowing rate, and herbicide treatment. *Agronomy Journal*, 91(3):478–485.

E. C. SEIM. Plant nutrients, soil fertility and grid sampling. (captítulo 4). Informe técnico, *Precisionag.org* desarrollada por California Polytechnic State University (San Luis Obispo) and California State University (Fresno) and the University of California (Davis). USDA and ARI, 2000. URL `http://www.precisionag.org/html/Toc.html`.

C. SHANNON. 1948. Mathematical theory of communication. *Bell System Technical Journal*, 27:379–623.

S. S. SKIENA. 1998. *The algorithm design manual.* Springer Verlag. ISBN 0-387-94860-0.

J. SMITH. 1974. *Models in ecology.* Cambrige at the university press.

M. SÀNCHEZ-MARRÈ, U. CORTÉS & J. COMAS. 2004. Environmental sciences and artificial intelligent. *Environmental Modelling & software*, 19:761–762.

W. M. SPEARS & K. A. DE JONG. 1992. Using genetic algorithms for supervised concept learning. *Artificial Intelligence Methods and Applications.* Based on previous IEEE90 AI Tools paper.

D. STOCKWELL. *Genetic Algorithms II: Species distribution modelling*, chapter 5, páginas 123–144. Kluwer Academic Publishers, 1999. ISBN 0412841908.

J. V. STTAFORD & P. MILLER. 1993. Spatially selective application of herbicides to cereal crops. *Computer and electronics in agriculture*, 9:217–229.

H. STUCKENSCHMIDT. Problem-solving methods for efficient reasoning under uncertainty. En *Workshop on Knowledge Acquisition, Modeling and Management.* Springer LNCS, Alberta Canada, 1999.

A. TABERNER. Biología de *Lolium rigidum Gaud.* como planta infestante del cultivo de cebada. aplicación al establecimiento de métodos de control. Tesis (Master), Universitat de Lleida (UdL), marzo 1996.

THEFREEDICTIONARY.COM. Encyclopedia: (liebig's law of the minimum), 2004. URL `http://encyclopedia.thefreedictionary.com/`.

S. THOMAS & S. SARAWAGI. Mining generalized association rules and sequential patterns using SQL queries. En *Knowledge Discovery and Data Mining*, páginas 344–348, 1998. URL `citeseer.ist.psu.edu/thomas98mining.html`.

J. THOMPSON, J. STAFFORD & B. AMBLER. 1990. Weed detection in cereal crops. *American Society of Agricultural Engineers*, (90-1629).

A. VALLS. 1999. On the sematics of qualitative attributes in knowledge elicitation. *International journal of intelligent systems*, 14:195–209.

K. D. VOHNOUT. 2003. *Mathematical modeling for system analysis in agricultural research.* Elsevier. ISBN 0-444-51268-3.

E. VRINDTS, D. MOSHOU, J. REUMERS, H. RAMON & J. BAERDEMAEKER. Spectral weed detection and precise spraying. En *Presentation at second workshop of the Site-Specific Weed Management (EWRS).* Working Group of the EWRS, 2002.

W. WALLEY & M. O'CONNOR. 2001. Unsupervised pattern recognition for the interpretation of ecological data. *Ecological Modelling*, 146:219–230.

A. M. WALTER, S. CHRISTENSEN & S. E. SIMMELSGAARD. 2002. Spatial correlation between weed species densities and soil properties. *Weed Research*, 42(1):26–38.

A. M. WALTER, T. HEISEL & S. CHRISTENSEN. Precision application of herbicides using injection sprayer systems. En G. Grenier & S. Blackmoore (eds.), *Proceedings of the Third European Conference on Precision Agriculture,(3rdECPA)*, páginas 611–616. Agro Montpellier, Montpellier, Francia, 2001. ISBN 2-900792-13-4.

G. WEIGERT & P. WAGNER. Development of decision rules for site-specific n fertilization by the application of data mining techniques. En J. Stafford & A. Werner (eds.), *Proceedings of the Third European Conference on Precision Agriculture,(3rdECPA)*, páginas 95–101, Berlin, Alemania, 2003. Wageningen Academic publishers. ISBN 9076998213.

B. WHELAN. *Reconciling Continuous Soil Variation and Crop Yield - A study of some implications of within-field variability for site-specific crop management.* Tesis Doctoral, Department of Agricultural chemistry and soil science. University of Sydney, Australia, noviembre 1998.

G. WIDMER. 2003. Discovering simple rules in complex data: A meta-learning algorithm and some surprising musical discoveries. *Artificial Intelligent*, 146: 129–148.

M. WILLE, D. THILL & W. PRICE. 1998. Wild oat (avena fatua) seed production in spring barley (hordeum vulgare) is affected by the interaction of wild oat density and herbicide rate. *Weed Science*, 46:336–343.

B. WILSON & P. BRAIN. 1991. Long-term stability of distribution of alopercurus myosuroides huds. within cereal fields. *Weed Research*, 31:367–373.

I. WITTEN & E. FRANK. 1999. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations.* Morgan Kaufmann, CA, EE UU. ISBN 1-55860-552-5.

WWW.MNCERTER.ORG. Recent studies regarding pesticides and health impacts. Informe técnico, Minnesota center for Environmental advocacy, Marzo 2004. URL `www.mncenter.org/p.asp?WebPage_ID=24\&Profile_ID=241`.

G. ZANIN, A. BERTI & L. RIELLO. 1998. Incorporation of weed spatial variability into the weed control decision-making process. *Weed Research*, 38(2):107–118.

Apéndices

Apéndice A

Valores estadísticos de los datos del campo de Barcelona

	semillas_lolium2001_m2	semillas_lolium2002_m2	semillas_lolium2003_m2	biomasa_maxima_2001	biomasa_trigo_2001	biomasa_de_paja_del_trigo_2001	biomasa_de_espigas_del_trigo_2001	biomasa_de_grano_de_trigo_2001	biomasa_del_loliu m_2001	biomasa_de_avena_2001
MIN	0	628.76	0	622.78	22	12	10	6	24.4	0
MEDIA	1607.79378	6405.183071	2811.47311	926.3216063	706.1858268	409.4708661	296.7149606	200.2174016	347.7333333	35.77322835
MAX	20591.89	41026.59	14933.05	1367.292	1569.2	946	672.8	478	1292.8	344

	biomasa_de_avena_2002	biomasa_de_loliu m_2002	biomasa_de_trigo_2002	biomasa_de_paja_del_trigo_2002	biomasa_de_espigas_del_trigo_2002	biomasa_de_grano_del_trigo_2002	sand	silt	clay	OM	QWP	QFC
MIN	0	54	386.4	227.4	159	115	15.36527	41.07139	23.21426	2.23658	22.24296744	38.25407821
MEDIA	99.5023622	284.023622	1335.111181	654.6771654	680.4340157	512.7574803	25.77469197	46.76873469	27.51190299	2.601588898	24.83711609	40.93704466
MAX	737.2	634.4	2686.4	1271.6	1414.8	1101	32.60139	54.5549	32.01423	3.07403	27.76893174	44.6185288

	WHC	x_m	y_m	densavena2001	densidadlolium2001	densidadlolium2002	densidadlolium2003	densidad_estimada_g_m2	z_estacion_total	z_relativa	13febrero2001	04Julio2001
MIN	14.46471846	0	0	0	11.11111111	222.2222222	244.4444444	1.19365	-12.307	0	0	0
MEDIA	16.09992857	74.4488189	74.40944882	10.70866142	470.1224847	1739.107612	2049.912511	1.262776378	-6.23185433	6.075145669	-	-
MAX	17.56912934	150	150	152	2688.888889	5077.777778	5288.888889	1.37985	-1.5	10.807	0	0

	humene	humfeb	hummar	hummay	orientacion_desde_el_N	radianes_desde_azimut_N	categoria_NESW	presencia_de_paja 2001	presencia_de_paja 2002	pendiente_en_gr	pendiente_en_rad	peso_de_mi_granos
MIN	0.10031	0.05365	0.14372	0.1481	0.8313234	0.01450933	1	0	0	0.595209	0.00595202	12.59445844
MEDIA	0.127614291	0.085517677	0.166844252	0.18632248	229.5795192	4.006918505	-	-	-	7.076922172	0.070521629	23.29095663
MAX	0.15575	0.11909	0.18126	0.21816	359.9752	6.282752466	4	1	1	20.93582	0.206377423	34.60207612

	num_granos_total	peso_mil_granos	num_de_granos_total	piestrigo_m2	radiacion_anual_acumulada	pH	cond	N	P	K	Hum	MO
MIN	91.71974522	21.36752137	912.5127162	6	74537.38	7.8	0.2	5	13	16	0.7	1.6
MEDIA	2159.764497	31.70075288	4069.070408	62.55643045	81722.55114	8.138582677	0.297244094	16.88188976	31.77165354	210.6141732	1.04015748	2.598425197
MAX	5588	39.68253968	8918.1	153	90143.38	8.4	1.05	31	80	434	1.8	4.4

	arena	limgros	limfi	argila	Phpointestimate	Phblockestimate	Condpointestimate	Condblockestimate	Nestimate	Nblockestimate	Pestimate	Pblockestimate
MIN	11.5	7.3	20.1	18.6	7.8	8.0061	0.2	0.24705	5	12.22181	13	24.34295
MEDIA	25.75826772	14.3503937	32.49133858	27.48031496	8.137283425	8.136390984	0.298072126	0.298541693	16.9618372	16.90774413	31.6497561	31.59308921
MAX	45.5	25.5	46.9	44.4	8.4	8.21099	1.05	0.426	31	20.99646	80	47.76891

	Kestimate	Kblockestimate	Humestimate	Humblockestimate	Moestimate	Moblockestimate	Sandestimate	Sandblockestimate	Siltestimate	Siltblockestimate	Clayestimate	Clayblockestimate
MIN	16	136.1928	0.7	0.81698	1.6	2.23658	11.5	15.36527	33	41.07139	18.6	23.21426
MEDIA	211.359065	211.1574358	1.040488583	1.039391339	2.599804921	2.601588898	25.75257815	25.77469197	46.79447539	46.76873469	27.52492472	27.51190299
MAX	434	322.4347	1.8	1.27695	4.4	3.07403	45.5	32.60139	57.9	54.5549	44.4	32.01423

LEYENDA

INFORMACIÓN ADICIONAL PARA CREAR LOS MAPAS TRAS EL APRENDIZAJE

VARIABLES PREDICTORAS

VARIABLE QUE DETERMINAN LA CLASE (Presencia de Avena)

Variables iniciales de la tabla de Barcelona

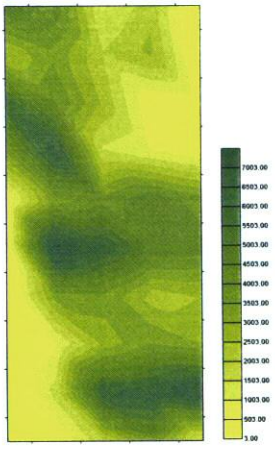
Apéndice B

Mapas de la variación espacial de las propiedades muestreadas

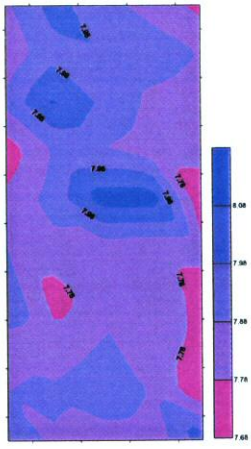
Mapas de los campos de Madrid

Parcela nº 1

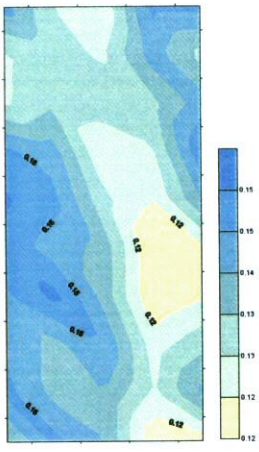
Semillas m^2



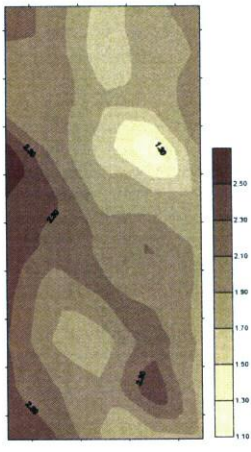
pH



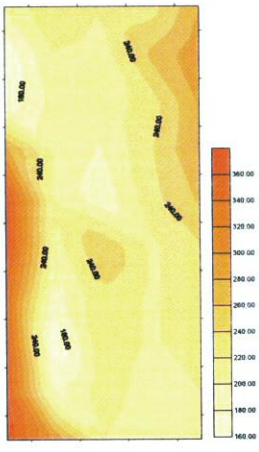
Nitrógeno (%)



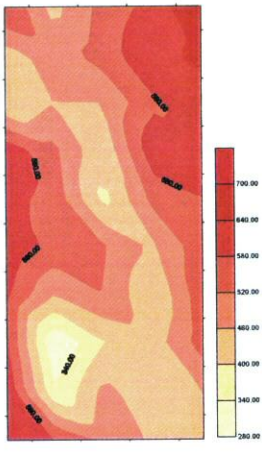
Materia Orgánica (%)



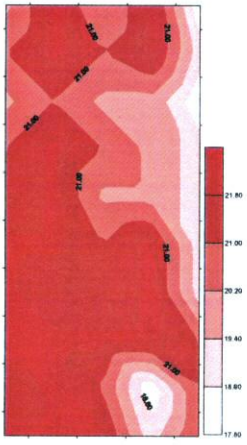
Fósforo (mg kg^{-1})



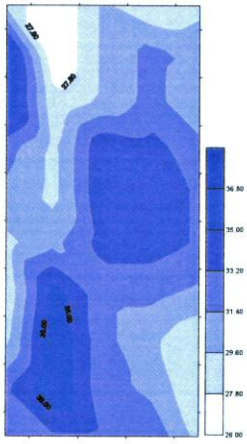
Potasio (mg kg^{-1})



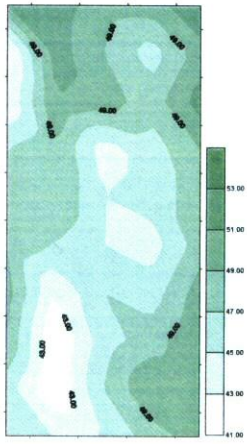
Arcilla



Arena

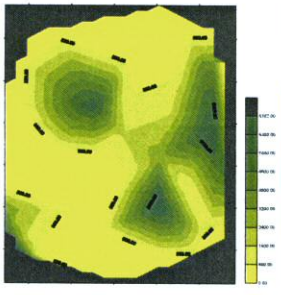


Limo (%)

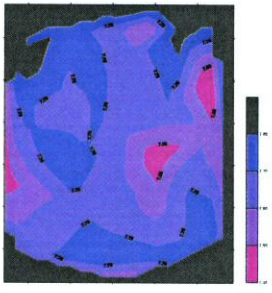


Parcela nº 2

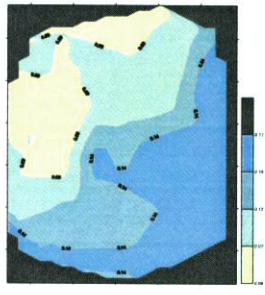
Semillas m2



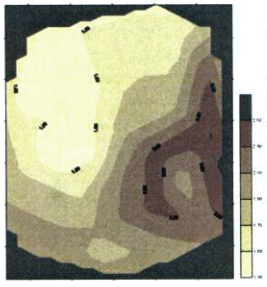
pH



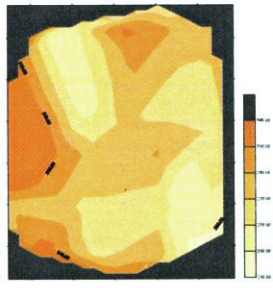
Nitrógeno (%)



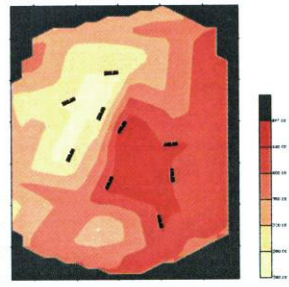
Materia Orgánica (%)



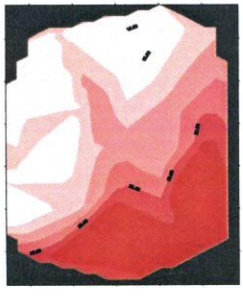
Fósforo (mg kg-1)



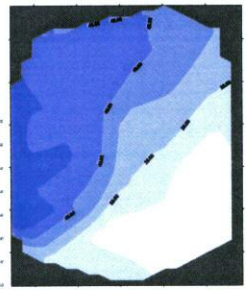
Potasio (mg kg-1)



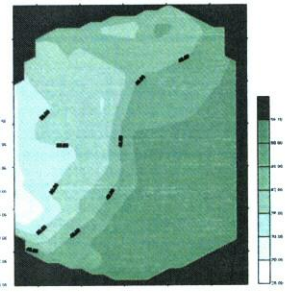
Arcilla



Arena

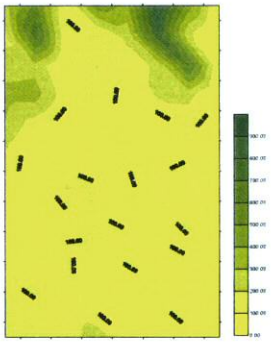


Limo (%)

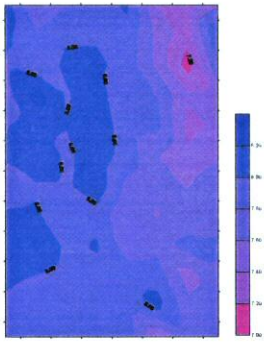


Parcela nº 3

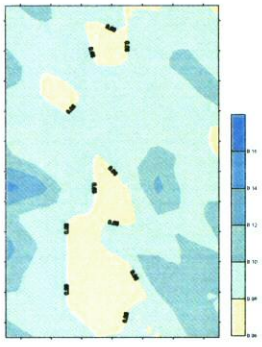
Semillas m2



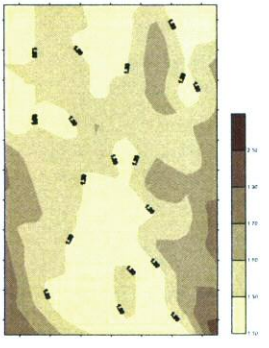
pH



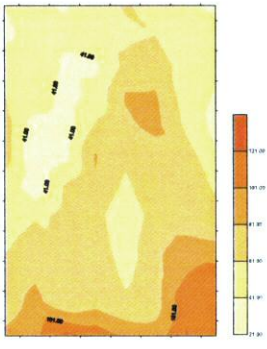
Nitrógeno (%)



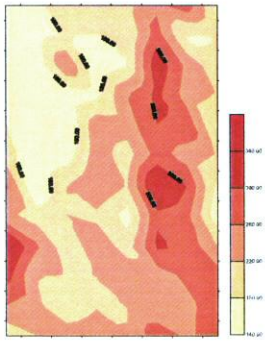
Materia Orgánica (%)



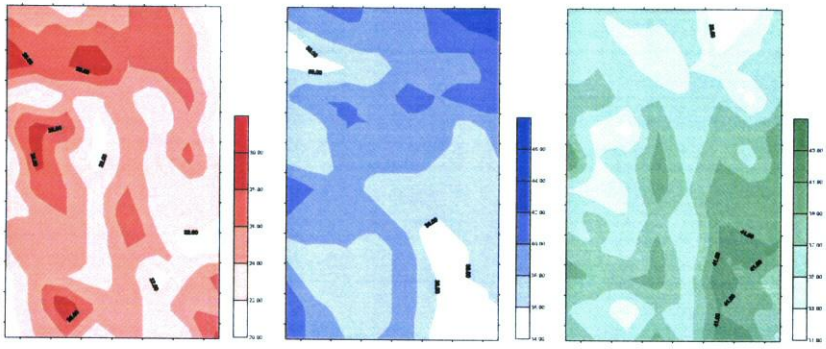
Fósforo (mg kg-1)



Potasio (mg kg-1)

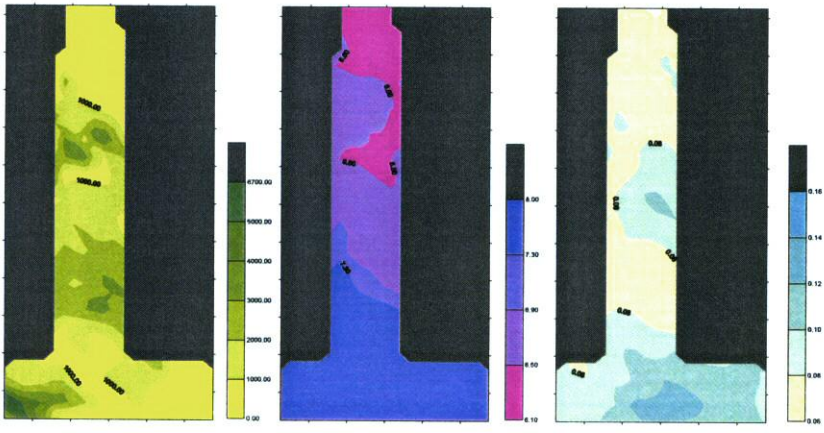


Arcilla Arena Limo (%)

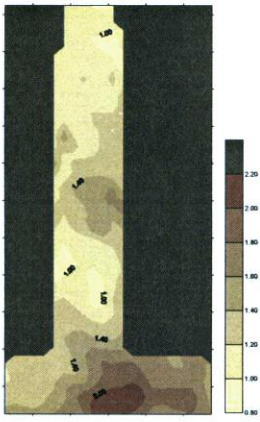


Parcela nº 4

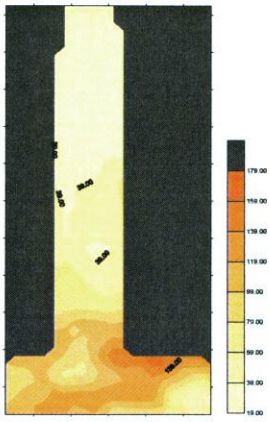
Semillas m2 pH Nitrógeno (%)



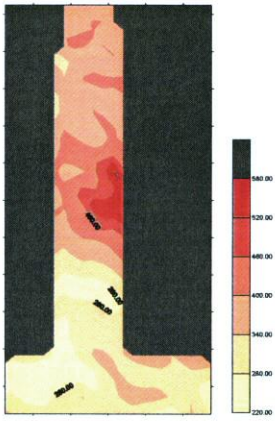
Materia Orgánica (%)



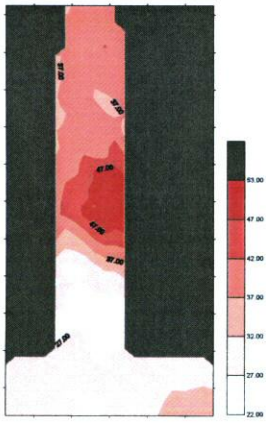
Fósforo (mg kg-1)



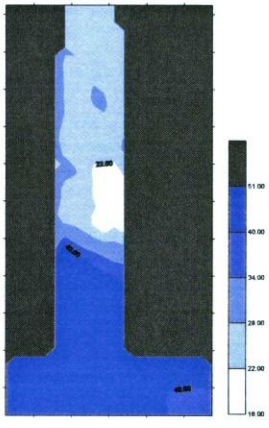
Potasio (mg kg-1)



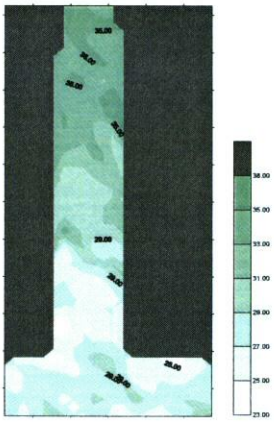
Arcilla



Arena



Limo

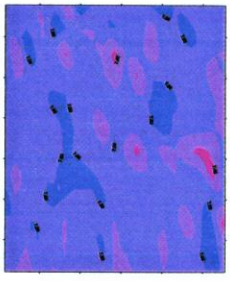


(%)

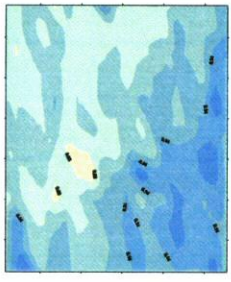
Parcela nº 5

Semillas m2

pH



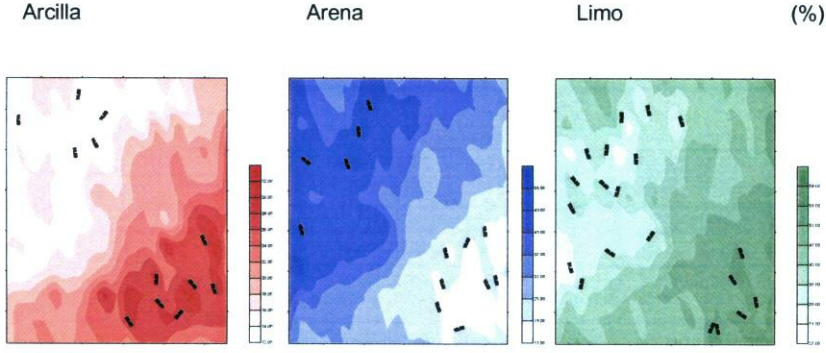
Nitrógeno (%)



Materia Orgánica (%)

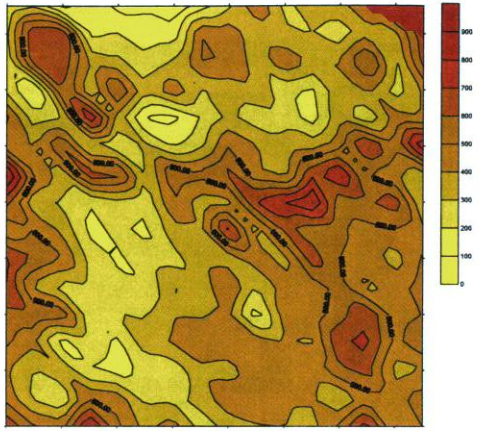
Fósforo (mg kg-1)

Potasio (mg kg-1)

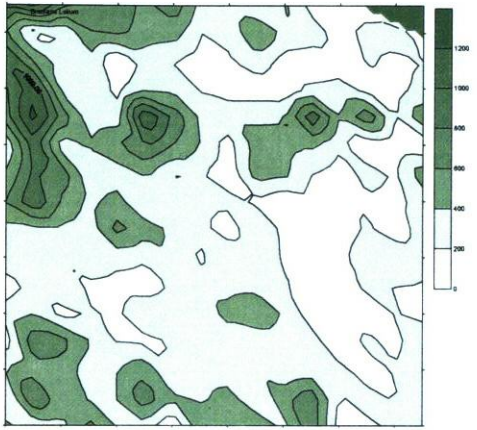

Arcilla
Arena
Limo
(%)

Mapas del Campo de Barcelona

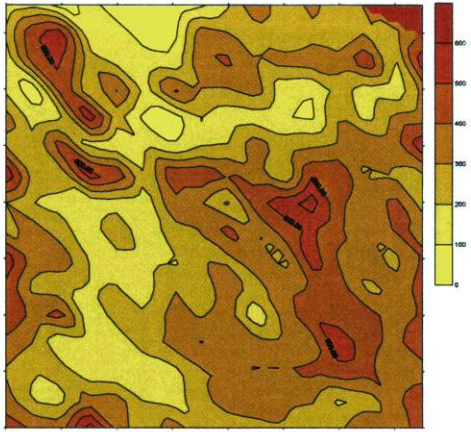
Biomasa paja de trigo (2001)



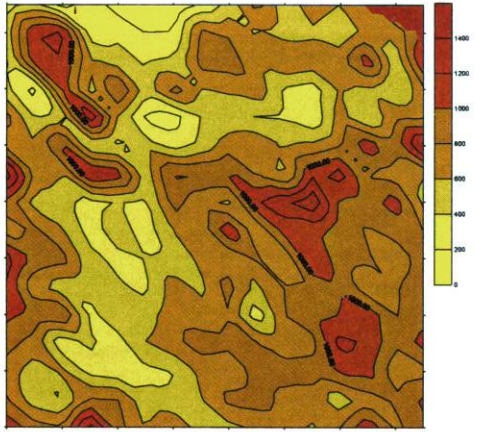
Biomasa Lolium (2001)



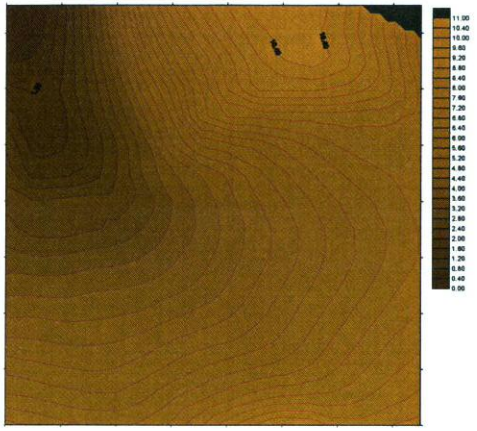
Biomasa espigas trigo (2001)



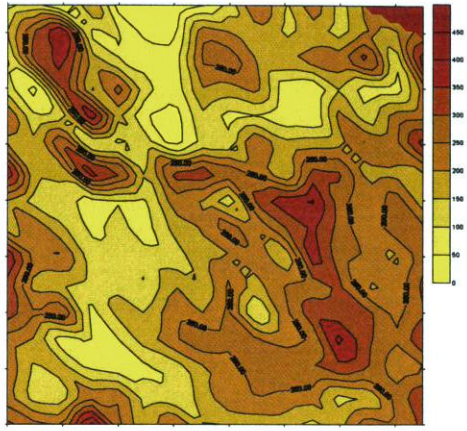
Biomasa trigo (2001)



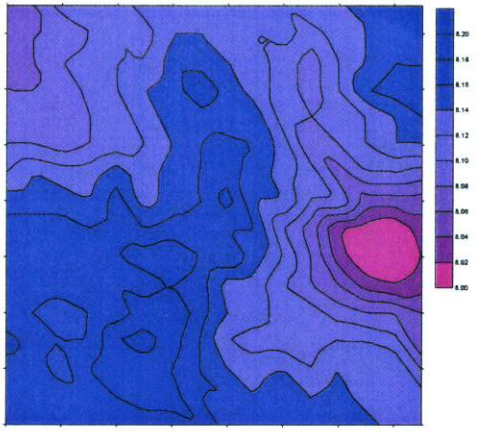
Elevación relativa



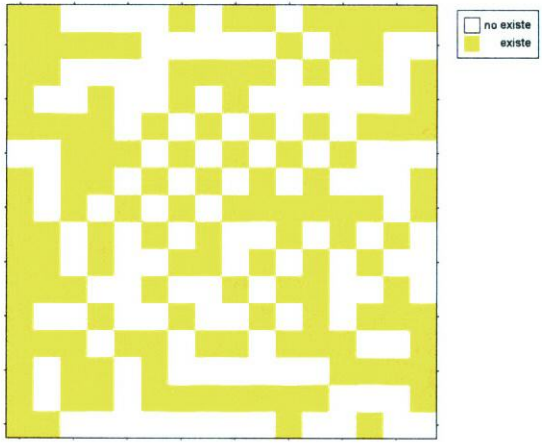
Biomasa grano de trigo (2001)



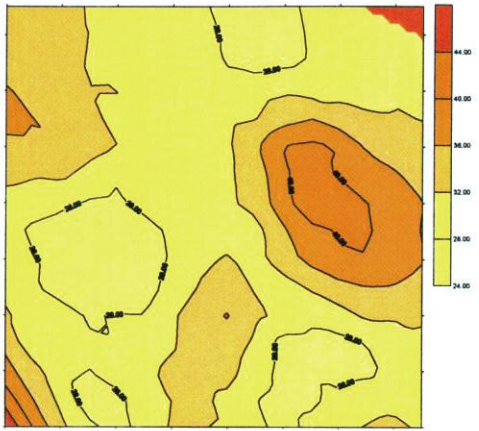
pH (2001)



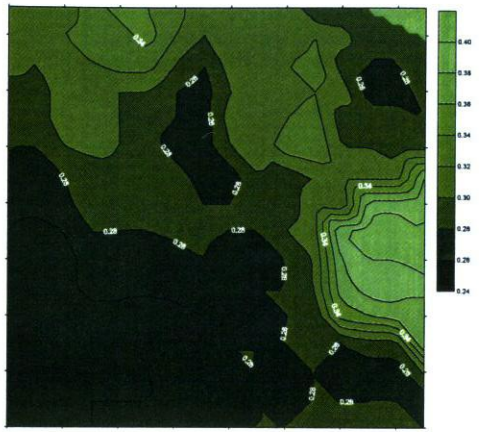
Presencia de Paja (2001)



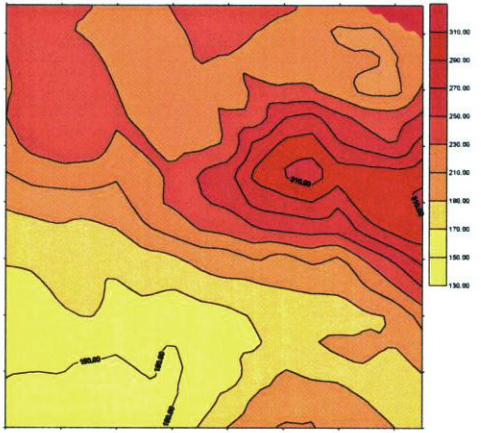
Fósforo (2001)



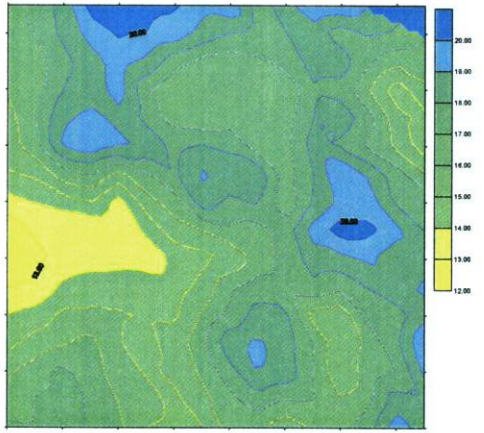
Conductividad (2001)



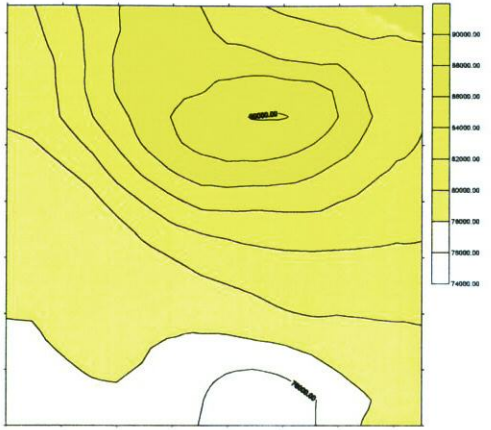
Potasio (2001)



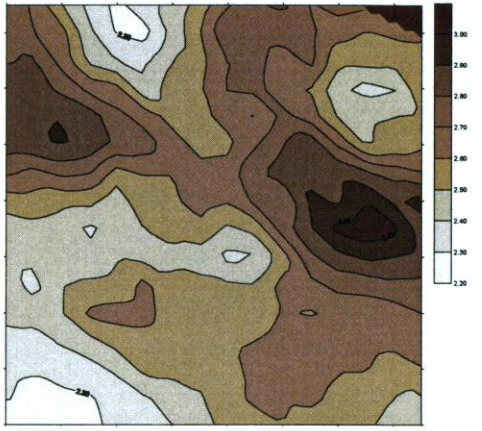
Nitrógeno (2001)



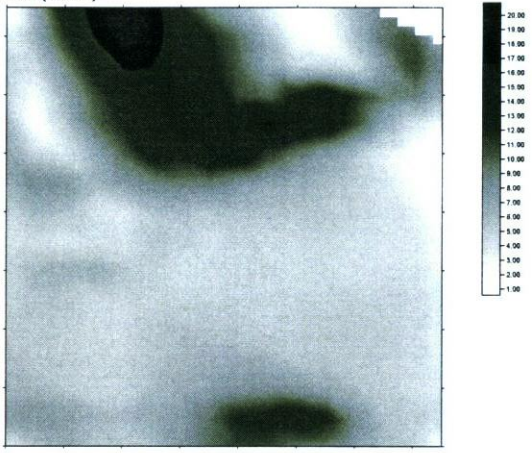
Radiación anual acumulada (2001)



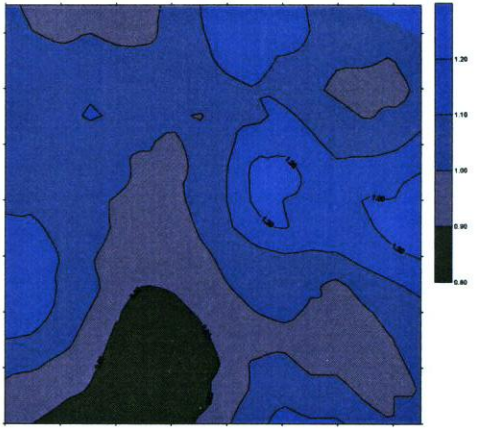
Materia Orgánica (2001)



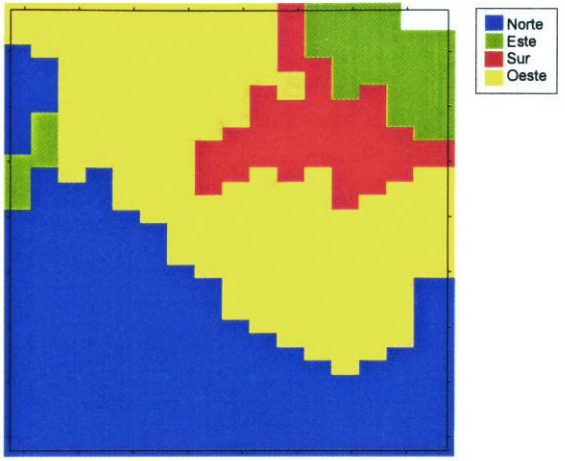
pendiente (2001)



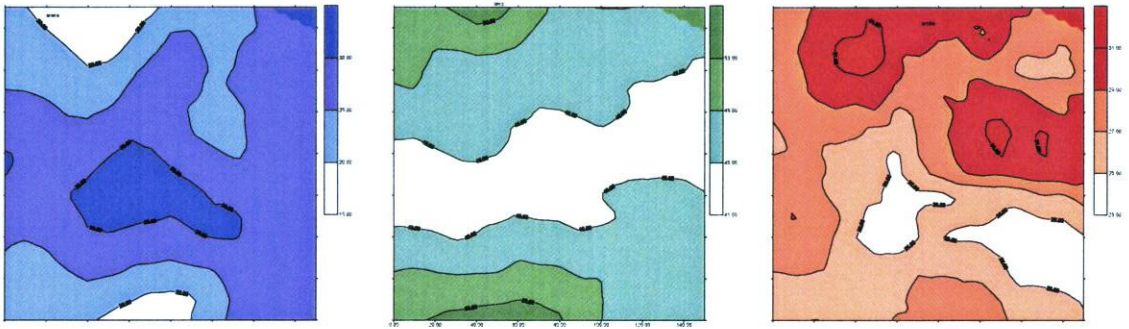
Humedad (2001)



Orientación cardinal



Textura: arena, limo y arcilla (2001)



Apéndice C

Umbrales de categorización de los datos de Barcelona

Variable	Intervalos	Etiquetas	Nombre
pH	≤8,006	bajo	*grado de acidez*
	>8,108	alto	
MO	≤1,4	baja	*materia orgánica*
	(1,4-1,9)	medio-baja	
	(1,9-2,4)	medio	
	(2,4-2,9)	medio-alta	
	(2,9-3,9)	alta	
	≥3,9	excesiva	
N	≤15	normal	*nitrógeno*
	(15-30)	normal-alto	
	(30-45)	alto	
	(45-60)	muy alto	
	≥60	excesivo	
P	≤6	muy bajo	*fósforo*
	(6-12)	bajo	
	(12-19)	medio	
	(19-36)	alto	
	(36-80)	muy alto	
	≥80	excesivo	
K	≤80	muy bajo	*potasio*
	(80-175)	bajo	
	(175-300)	medio	
	(300-425)	alto	
	≥425	muy alto	
Cond	≤0,307	baja	*conductividad*
	(0,307-0,367)	media	
	≥0,367	alta	
hum	≤0,97	baja	*humedad*
	(1,12-1,28)	media	
	≥0,367	alta	
Arcilla	≤26,1	baja	*porcentaje arcilla*
	(26,1-29,0)	media	
	≥29,0	alta	
Limo	≤45,6	baja	*porcentaje limo*
	(45,6-50,1)	media	
	≥50,1	alta	
Arena	≤21,1	baja	*porcentaje arena*
	(21,1-26,9)	media	
	≥26,9	alta	
Suelo	arcilla+limo+arena	triangulo textural	*Tipo de suelo*

(CONTINÚA)

Variable	Intervalos	Etiquetas	Nombre
pendiente	≤7,1	baja	*inclinación terreno*
	(7,1-13,6)	media	
	≥13,6	alta	
pendiente *lolium*	≤0,5	clase 0	*inclinación terreno*
	(0,5-1,0)	clase 1	
	(1,0-3,0)	clase 2	
	(3,0-10,0)	clase 3	
	(10,0-25,0)	clase 4	
elevacion	≤3,602	baja	*elevación relativa*
	(3,602-7,204)	media	
	≥7,204	alta	
Orientacion	1	norte	*orientación cardinal*
	2	este	
	3	sur	
	4	oeste	
rad	≤79 739,38	baja	*radiación anual*
	(79 739,38-84 941,38)	media	
	≥84 941,38	alta	
paja	=0	no	*presencia de paja*
	≠0	si	
Trigo	≤537,73	baja	*biomasa trigo*
	(537,73-1053,47)	media	
	≥1053,47	alta	
paja	≤323,33	baja	*biomasa*
	(323,33-634,66)	media	*paja de trigo*
	≥634,66	alta	
espiga	≤230,93	baja	*biomasa*
	(230,93-451,87)	media	*espiga de trigo*
	≥451,87	alta	
grano	≤163,30	baja	*biomasa*
	(163,30-320,66)	media	*grano de trigo*
	≥320,66	alta	
Lolium	≤0,24	poca	*semillas lolium*
	>0,24	mucha	
avena	=0	no existe	*biomasa avena*
	≠0	existe	

Apéndice D

Informes de limpieza de los datos

Para la etapa de limpieza de datos en el preprocesado de los datos, la aplicación `PreparaDAT` realiza una serie de pasos basados en consultas SQL y algunas rutinas implementadas con Visual Basic©, que se enumeran a continuación.

1. *En primer lugar se realiza la agrupación de los registros utilizando todos los atributos excepto el atributo-clase, mediante una consulta de selección.*

2. *Una segunda consulta crea una tabla que contiene los duplicados, determinando el número de registros repetidos para cada clase. Se puede observar que en esta tabla aparecen registros que están en las dos clases y otros que se repiten sólo en una.*

3. *Para filtrar esta importante información y evitar las inconsistencias mencionadas se realiza una consulta que genera una tabla nueva que contiene los registros representantes de todos aquellos que están incluidos en los dos grupos al mismo tiempo.*

4. *A continuación dos nuevas consultas concatenadas filtran de la tabla anterior cada uno de los registros que han de ser eliminados, teniendo en cuenta su clave identificadora.*

5. *La siguiente consulta contiene la tabla sin estos duplicados, la que podríamos llamar tabla limpia.*

6. *Sin embargo, del grupo de eliminados aún se puede extraer registros dominantes, es decir, hacer que sobrevivan aquellos que superen en número, y por lo tanto, estadísticamente más representativos. En esta consulta se calcula la diferencia de los registros, y se añade el número de registros de la clase que está en mayoría, tantas veces como sea la diferencia.*

7. *Una última fase crea un archivo de texto con los registros que pueden recuperarse, para que si se considera, se importen a la tabla limpia.*

Numero de duplicados POR REGISTRO: 142

(3)_____

```
SELECT [Duplicados_and_MHDif_1].pHDiscreto,
[Duplicados_and_MHDif_1].NDiscreto, [Duplicados_and_MHDif_1].MODiscreto,
[Duplicados_and_MHDif_1].PDiscreto, [Duplicados_and_MHDif_1].KDiscreto,
[Duplicados_and_MHDif_1].arcillaDiscreto,
[Duplicados_and_MHDif_1].limoDiscreto, [Duplicados_and_MHDif_1].arenaDiscreto,
 Count(Duplicados_and_MHDif_1.CuentaDeClave) AS CuentaDeCuentaDeClave INTO
DuplicadosEnAtributosTabla  FROM Duplicados_and_MHDif_1 GROUP BY
[Duplicados_and_MHDif_1].pHDiscreto, [Duplicados_and_MHDif_1].NDiscreto,
[Duplicados_and_MHDif_1].MODiscreto, [Duplicados_and_MHDif_1].PDiscreto,
[Duplicados_and_MHDif_1].KDiscreto, [Duplicados_and_MHDif_1].arcillaDiscreto,
[Duplicados_and_MHDif_1].limoDiscreto, [Duplicados_and_MHDif_1].arenaDiscreto
HAVING ((((Count(Duplicados_and_MHDif_1.CuentaDeClave)) > 1));
```

Creación de la Consulta : [nuevaDuplicadosTabla_2]

Numero de registros representados en los dos grupos, es decir, los que deben ser eliminados : 42

(4)_____

```
SELECT [Duplicados_CuentaMH_diferentes].pHDiscreto,
[Duplicados_CuentaMH_diferentes].NDiscreto,
[Duplicados_CuentaMH_diferentes].MODiscreto,
[Duplicados_CuentaMH_diferentes].PDiscreto,
[Duplicados_CuentaMH_diferentes].KDiscreto,
[Duplicados_CuentaMH_diferentes].arcillaDiscreto,
[Duplicados_CuentaMH_diferentes].limoDiscreto,
[Duplicados_CuentaMH_diferentes].arenaDiscreto,
[Duplicados_CuentaMH_diferentes].Sem_metro2Discreto,
[Duplicados_CuentaMH_diferentes].CuentaDeClave  INTO
RegistrosDuplicadosPareados  FROM Duplicados_CuentaMH_diferentes,
DuplicadosEnAtributosTabla  WHERE [Duplicados_CuentaMH_diferentes].pHDiscreto
= [DuplicadosEnAtributosTabla].pHDiscreto AND
[Duplicados_CuentaMH_diferentes].NDiscreto =
[DuplicadosEnAtributosTabla].NDiscreto AND
[Duplicados_CuentaMH_diferentes].MODiscreto =
[DuplicadosEnAtributosTabla].MODiscreto AND
[Duplicados_CuentaMH_diferentes].PDiscreto =
[DuplicadosEnAtributosTabla].PDiscreto AND
[Duplicados_CuentaMH_diferentes].KDiscreto =
[DuplicadosEnAtributosTabla].KDiscreto AND
[Duplicados_CuentaMH_diferentes].arcillaDiscreto =
[DuplicadosEnAtributosTabla].arcillaDiscreto AND
[Duplicados_CuentaMH_diferentes].limoDiscreto =
[DuplicadosEnAtributosTabla].limoDiscreto AND
[Duplicados_CuentaMH_diferentes].arenaDiscreto =
[DuplicadosEnAtributosTabla].arenaDiscreto
```

Las consultas (4) y (5) saleccionan los registros que verdaderamente han de ser eliminados

Creación de la Consulta : [RegistrosDuplicados_cuenta_a_Tbl]

Creación de la Tabla : [RegistrosDuplicadosPareados]

Numero de Registros que deben borrarse; TOTALES) : 84

(5)_____

Creación de la Consulta : [RegistrosAborrar]

```
SELECT [DatosIntegrados0_1Discreto].pHDiscreto,
[DatosIntegrados0_1Discreto].NDiscreto,
[DatosIntegrados0_1Discreto].MODiscreto,
```


Número de registros :536
Tipo de datos : CATEGORICOS

(1)_____

```
SELECT DISTINCTROW [DatosIntegrados0_1Discreto].pHDiscreto,
[DatosIntegrados0_1Discreto].NDiscreto,
[DatosIntegrados0_1Discreto].MODiscreto,
[DatosIntegrados0_1Discreto].PDiscreto,
[DatosIntegrados0_1Discreto].KDiscreto,
[DatosIntegrados0_1Discreto].arcillaDiscreto,
[DatosIntegrados0_1Discreto].limoDiscreto,
[DatosIntegrados0_1Discreto].arenaDiscreto,
[DatosIntegrados0_1Discreto].Sem_metro2Discreto,
[DatosIntegrados0_1Discreto].Clave FROM DatosIntegrados0_1Discreto WHERE
((([DatosIntegrados0_1Discreto].NDiscreto) In (SELECT NDiscreto FROM
DatosIntegrados0_1Discreto As Tmp  GROUP BY [pHDiscreto], [NDiscreto],
[MODiscreto], [PDiscreto], [KDiscreto], [arcillaDiscreto], [limoDiscreto],
[arenaDiscreto] HAVING Count(*)>1  And [pHDiscreto] =
[DatosIntegrados0_1Discreto].pHDiscreto And [NDiscreto] =
[DatosIntegrados0_1Discreto].NDiscreto And [MODiscreto] =
[DatosIntegrados0_1Discreto].MODiscreto And [PDiscreto] =
[DatosIntegrados0_1Discreto].PDiscreto And [KDiscreto] =
[DatosIntegrados0_1Discreto].KDiscreto And [arcillaDiscreto] =
[DatosIntegrados0_1Discreto].arcillaDiscreto And [limoDiscreto] =
[DatosIntegrados0_1Discreto].limoDiscreto And [arenaDiscreto] =
[DatosIntegrados0_1Discreto].arenaDiscreto))) ORDER BY
[DatosIntegrados0_1Discreto].pHDiscreto,
[DatosIntegrados0_1Discreto].NDiscreto,
[DatosIntegrados0_1Discreto].MODiscreto,
[DatosIntegrados0_1Discreto].PDiscreto,
[DatosIntegrados0_1Discreto].KDiscreto,
[DatosIntegrados0_1Discreto].arcillaDiscreto,
[DatosIntegrados0_1Discreto].limoDiscreto,
[DatosIntegrados0_1Discreto].arenaDiscreto;
```

Creación de la Consulta : [Duplicados]

Numero de duplicados TOTALES: 353

En primer lugar se realiza una agrupación de los registros en función de todos los atributos excepto los que forman las clases.

(2)_____

```
SELECT [Duplicados].pHDiscreto, [Duplicados].NDiscreto,
[Duplicados].MODiscreto, [Duplicados].PDiscreto, [Duplicados].KDiscreto,
[Duplicados].arcillaDiscreto, [Duplicados].limoDiscreto,
[Duplicados].arenaDiscreto, [Duplicados].Sem_metro2Discreto,
Count(Duplicados.Clave) AS CuentaDeClave  INTO duplicados_cuentaMH_diferentes
FROM Duplicados GROUP BY [Duplicados].pHDiscreto, [Duplicados].NDiscreto,
[Duplicados].MODiscreto, [Duplicados].PDiscreto, [Duplicados].KDiscreto,
[Duplicados].arcillaDiscreto, [Duplicados].limoDiscreto,
[Duplicados].arenaDiscreto, [Duplicados].Sem_metro2Discreto
```

Esta tabla, sin embargo, aún contiene información que nos interesa, que son los registros duplicados que solo pertenecen a una clase.

Esta segunda agrupación de los registros no crea una tabla que contiene el numero de veces que se duplica cada registro. En esta tabla se puede observar que existen registros que están en las dos clases y otros que se repiten solo en una. Para filtrar esta importante información y evitar las inconsistencias se realiza la consulta (3)

Creación de la Consulta : [Duplicados_and_MHDif_2]

Creación de la Tabla : [Duplicados_and_MHDif_1]

```
[DatosIntegrados0_1Discreto].PHDiscreto,
[DatosIntegrados0_1Discreto].NDiscreto,
[DatosIntegrados0_1Discreto].MODiscreto,
[DatosIntegrados0_1Discreto].PDiscreto,
[DatosIntegrados0_1Discreto].KDiscreto,
[DatosIntegrados0_1Discreto].arcillaDiscreto,
[DatosIntegrados0_1Discreto].limoDiscreto,
[DatosIntegrados0_1Discreto].arenaDiscreto,
[DatosIntegrados0_1Discreto].Sem_metro2Discreto, [TablaLimpia_DEF].PHDiscreto,
[TablaLimpia_DEF].NDiscreto, [TablaLimpia_DEF].MODiscreto,
[TablaLimpia_DEF].PDiscreto, [TablaLimpia_DEF].KDiscreto,
[TablaLimpia_DEF].arcillaDiscreto, [TablaLimpia_DEF].limoDiscreto,
[TablaLimpia_DEF].arenaDiscreto, [TablaLimpia_DEF].Sem_metro2Discreto;

Creación de la Consulta : [consultaRegistrosIdEliminados]

Creación de la Tabla : [TablaRegistrosEliminadosID]

Numero de Registros ELIMINADOS con su ID : 194

(7)_Fase_de_recuperación_de_registros__________

De los registros eliminados es posible extraer algunos que pueden
representarse en el grupo en el que se presenten en mayor proporcion. Entonces
se asume que ese registro tendrá mayor posibilidad de pertenecer a la clase en
la que exista más veces
0,mayor,mayor,mayor,mayor,medio,menor,medio,mayor,poca
0,mayor,mayor,mayor,mayor,medio,menor,medio,mayor,mucha
0,mayor,medio,mayor,mayor,medio,menor,menor,mayor,mucha
0,mayor,medio,mayor,mayor,medio,menor,menor,mayor,poca
0,mayor,medio,medio,mayor,mayor,medio,mayor,medio,poca
0,mayor,medio,medio,mayor,mayor,medio,mayor,medio,mucha
0,mayor,medio,medio,mayor,mayor,medio,medio,medio,poca
2,mayor,medio,medio,mayor,mayor,medio,medio,medio,mucha
0,mayor,medio,medio,mayor,menor,menor,menor,mayor,mucha
1,mayor,medio,medio,mayor,menor,menor,menor,mayor,poca
0,mayor,medio,medio,medio,mayor,medio,mayor,medio,mucha
0,mayor,medio,medio,medio,mayor,medio,mayor,medio,poca
0,mayor,medio,medio,medio,medio,medio,mayor,medio,poca
1,mayor,medio,medio,medio,medio,medio,mayor,medio,mucha
8,mayor,medio,medio,medio,medio,medio,medio,medio,mucha
0,mayor,medio,medio,medio,medio,medio,medio,medio,poca
1,mayor,medio,medio,medio,menor,menor,medio,mayor,mucha
0,mayor,medio,medio,medio,menor,menor,medio,mayor,poca
0,mayor,medio,medio,medio,menor,menor,menor,mayor,mucha
5,mayor,medio,medio,medio,menor,menor,menor,mayor,poca
0,mayor,medio,medio,menor,medio,medio,mayor,medio,poca
0,mayor,medio,medio,menor,medio,medio,mayor,medio,mucha
9,mayor,medio,medio,menor,menor,medio,menor,medio,mucha
0,mayor,medio,medio,menor,menor,medio,menor,medio,poca
0,mayor,medio,medio,menor,menor,menor,medio,medio,mucha
0,mayor,medio,medio,menor,menor,menor,medio,medio,poca
0,mayor,medio,menor,medio,medio,medio,medio,medio,mucha
1,mayor,medio,menor,medio,medio,medio,medio,medio,poca
0,mayor,medio,menor,medio,menor,menor,menor,mayor,mucha
0,mayor,medio,menor,medio,menor,menor,menor,mayor,poca
0,mayor,menor,medio,medio,medio,medio,medio,medio,poca
3,mayor,menor,medio,medio,medio,medio,medio,medio,mucha
0,mayor,menor,medio,medio,medio,menor,mayor,menor,mucha
6,mayor,menor,medio,medio,medio,menor,mayor,menor,poca
3,mayor,menor,medio,medio,medio,menor,medio,menor,mucha
0,mayor,menor,medio,medio,medio,menor,medio,menor,poca
1,mayor,menor,medio,medio,menor,menor,mayor,menor,poca
0,mayor,menor,medio,medio,menor,menor,mayor,menor,mucha
0,mayor,menor,medio,medio,menor,menor,medio,menor,mucha
0,mayor,menor,medio,medio,menor,menor,medio,menor,poca
0,mayor,menor,menor,medio,medio,medio,medio,medio,mucha
4,mayor,menor,menor,medio,medio,medio,medio,medio,poca
```



```
[DatosIntegrados0_1Discreto].PDiscreto,
[DatosIntegrados0_1Discreto].KDiscreto,
[DatosIntegrados0_1Discreto].arcillaDiscreto,
[DatosIntegrados0_1Discreto].limoDiscreto,
[DatosIntegrados0_1Discreto].arenaDiscreto,
[DatosIntegrados0_1Discreto].Sem_metro2Discreto,
[DatosIntegrados0_1Discreto].Clave INTO TotalRegistrosEliminar  FROM
DatosIntegrados0_1Discreto, RegistrosDuplicadosPareados WHERE
[DatosIntegrados0_1Discreto].PHDiscreto =
[RegistrosDuplicadosPareados].PHDiscreto AND
[DatosIntegrados0_1Discreto].NDiscreto =
[RegistrosDuplicadosPareados].NDiscreto AND
[DatosIntegrados0_1Discreto].MODiscreto =
[RegistrosDuplicadosPareados].MODiscreto AND
[DatosIntegrados0_1Discreto].PDiscreto =
[RegistrosDuplicadosPareados].PDiscreto AND
[DatosIntegrados0_1Discreto].KDiscreto =
[RegistrosDuplicadosPareados].KDiscreto AND
[DatosIntegrados0_1Discreto].arcillaDiscreto =
[RegistrosDuplicadosPareados].arcillaDiscreto AND
[DatosIntegrados0_1Discreto].limoDiscreto =
[RegistrosDuplicadosPareados].limoDiscreto AND
[DatosIntegrados0_1Discreto].arenaDiscreto =
[RegistrosDuplicadosPareados].arenaDiscreto AND
[DatosIntegrados0_1Discreto].Sem_metro2Discreto =
[RegistrosDuplicadosPareados].Sem_metro2Discreto

Creación de la Tabla : [TotalRegistrosEliminar]

Numero de Registros que deben borrarse; TOTALES con duplicados) : 194

(6.A)_Tabla_Limpia_sin_ningún_duplicado_________

SELECT DISTINCTROW [DatosIntegrados0_1Discreto].PHDiscreto,
[DatosIntegrados0_1Discreto].NDiscreto,
[DatosIntegrados0_1Discreto].MODiscreto,
[DatosIntegrados0_1Discreto].PDiscreto,
[DatosIntegrados0_1Discreto].KDiscreto,
[DatosIntegrados0_1Discreto].arcillaDiscreto,
[DatosIntegrados0_1Discreto].limoDiscreto,
[DatosIntegrados0_1Discreto].arenaDiscreto,
[DatosIntegrados0_1Discreto].Sem_metro2Discreto,
[DatosIntegrados0_1Discreto].Clave INTO TablaLimpia_DEF  FROM
DatosIntegrados0_1Discreto LEFT JOIN TotalRegistrosEliminar ON
[DatosIntegrados0_1Discreto].Clave = [TotalRegistrosEliminar].Clave WHERE
((([TotalRegistrosEliminar].Clave) Is Null));

Creación de la Consulta : [TablaDefinitivaLimpia]

Creación de la Tabla : [TablaLimpia_DEF]

Numero de Registros SOBREVIVEN : 342

(6.B)_____

SELECT DISTINCTROW [DatosIntegrados0_1Discreto].PHDiscreto,
[DatosIntegrados0_1Discreto].NDiscreto,
[DatosIntegrados0_1Discreto].MODiscreto,
[DatosIntegrados0_1Discreto].PDiscreto,
[DatosIntegrados0_1Discreto].KDiscreto,
[DatosIntegrados0_1Discreto].arcillaDiscreto,
[DatosIntegrados0_1Discreto].limoDiscreto,
[DatosIntegrados0_1Discreto].arenaDiscreto,
[DatosIntegrados0_1Discreto].Sem_metro2Discreto,
[DatosIntegrados0_1Discreto].Clave INTO TablaRegistrosEliminadosID  FROM
DatosIntegrados0_1Discreto LEFT JOIN TablaLimpia_DEF ON
[DatosIntegrados0_1Discreto].Clave = [TablaLimpia_DEF].Clave WHERE
((([TablaLimpia_DEF.Clave) Is Null))  ORDER BY
```

Número de registros :381
Tipo de datos : CATEGORICOS

(1)_____

```
SELECT DISTINCTROW [05_381].pendiente, [05_381].radiacion_anual, [05_381].pH,
[05_381].N, [05_381].P, [05_381].K, [05_381].MO, [05_381].SEMILLAS,
[05_381].tipo_de_suelo, [05_381].coverid FROM 05_381 WHERE
((([05_381].radiacion_anual) In (SELECT radiacion_anual FROM 05_381 As Tmp
GROUP BY [pendiente], [radiacion_anual], [pH], [N], [P], [K], [MO],
[tipo_de_suelo] HAVING Count(*)>1  And [pendiente] = [05_381].pendiente And
[radiacion_anual] = [05_381].radiacion_anual And [pH] = [05_381].pH And [N] =
[05_381].N And [P] = [05_381].P And [K] = [05_381].K And [MO] = [05_381].MO
And [tipo_de_suelo] = [05_381].tipo_de_suelo))) ORDER BY [05_381].pendiente,
[05_381].radiacion_anual, [05_381].pH, [05_381].N, [05_381].P, [05_381].K,
[05_381].MO, [05_381].tipo_de_suelo;
```

Creación de la Consulta : [Duplicados]

Numero de duplicados TOTALES: 381

En primer lugar se realiza una agrupación de los registros en función de todos los atributos excepto los que forman las clases.

(2)_____

```
SELECT [Duplicados].pendiente, [Duplicados].radiacion_anual, [Duplicados].pH,
[Duplicados].N, [Duplicados].P, [Duplicados].K, [Duplicados].MO,
[Duplicados].SEMILLAS, [Duplicados].tipo_de_suelo,  Count(Duplicados.coverid)
AS CuentaDeClave  INTO duplicados_cuentaNH_diferentes  FROM Duplicados GROUP
BY [Duplicados].pendiente, [Duplicados].radiacion_anual, [Duplicados].pH,
[Duplicados].N, [Duplicados].P, [Duplicados].K, [Duplicados].MO,
[Duplicados].SEMILLAS, [Duplicados].tipo_de_suelo
```

Esta tabla, sin embargo, aún contiene información que nos interesa, que son los registros duplicados que solo pertenecen a una clase.

Esta segunda agrupación de los registros no crea una tabla que contiene el numero de veces que se duplica cada registro. En esta tabla se puede observar que existen registros que están en las dos clases y otros que se repiten solo en una. Para filtrar esta importante información y evitar las inconsistencias se realiza la consulta (3)

Creación de la Consulta : [Duplicados_and_NHDif_2]

Creación de la Tabla : [Duplicados_and_NHDif_1]

Numero de duplicados POR REGISTRO: 180

(3)_____

```
SELECT [Duplicados_and_NHDif_1].pendiente,
[Duplicados_and_NHDif_1].radiacion_anual, [Duplicados_and_NHDif_1].pH,
[Duplicados_and_NHDif_1].N, [Duplicados_and_NHDif_1].P,
[Duplicados_and_NHDif_1].K, [Duplicados_and_NHDif_1].MO,
[Duplicados_and_NHDif_1].tipo_de_suelo,
Count(Duplicados_and_NHDif_1.CuentaDeClave) AS CuentaDeCuentaDeClave INTO
DuplicadosEnAtributosTabla  FROM Duplicados_and_NHDif_1 GROUP BY
[Duplicados_and_NHDif_1].pendiente, [Duplicados_and_NHDif_1].radiacion_anual,
[Duplicados_and_NHDif_1].pH, [Duplicados_and_NHDif_1].N,
[Duplicados_and_NHDif_1].P, [Duplicados_and_NHDif_1].K,
[Duplicados_and_NHDif_1].MO, [Duplicados_and_NHDif_1].tipo_de_suelo HAVING
((((Count(Duplicados_and_NHDif_1.CuentaDeClave)) > 1));
```

Creación de la Consulta : [nuevaDuplicadosTabla_2]

Numero de registros representados en los dos grupos, es decir, los que deben ser eliminados : 80


```
7,mayor,menor,menor,medio,medio,menor,mayor,menor,poca
0,mayor,menor,menor,medio,medio,menor,mayor,menor,mucha
0,mayor,menor,menor,medio,medio,menor,medio,menor,mucha
1,mayor,menor,menor,medio,medio,menor,medio,menor,poca
0,mayor,menor,menor,medio,menor,medio,medio,medio,mucha
1,mayor,menor,menor,medio,menor,medio,medio,medio,poca
0,mayor,menor,menor,medio,menor,menor,menor,mayor,mucha
0,mayor,menor,menor,medio,menor,menor,menor,mayor,poca
0,mayor,menor,menor,menor,menor,mayor,menor,menor,mucha
0,mayor,menor,menor,menor,menor,mayor,menor,menor,poca
0,mayor,menor,menor,menor,menor,medio,medio,medio,mucha
1,mayor,menor,menor,menor,menor,medio,medio,medio,poca
0,mayor,menor,menor,menor,menor,menor,medio,mayor,poca
1,mayor,menor,menor,menor,menor,menor,medio,mayor,mucha
0,mayor,menor,menor,menor,menor,menor,medio,menor,mucha
0,mayor,menor,menor,menor,menor,menor,medio,menor,poca
0,menor,mayor,mayor,medio,mayor,mayor,medio,medio,poca
0,menor,mayor,mayor,medio,mayor,mayor,medio,medio,mucha
0,menor,medio,medio,medio,medio,menor,mayor,menor,mucha
0,menor,medio,medio,medio,medio,menor,mayor,menor,poca
0,menor,medio,medio,medio,menor,mayor,mayor,medio,mucha
1,menor,medio,medio,medio,menor,mayor,mayor,medio,menor,poca
0,menor,medio,medio,medio,menor,medio,mayor,medio,menor,mucha
1,menor,medio,medio,medio,menor,medio,mayor,medio,menor,poca
0,menor,medio,menor,menor,menor,menor,medio,medio,mucha
0,menor,medio,menor,menor,menor,menor,medio,medio,poca
0,menor,menor,medio,menor,medio,medio,mayor,medio,mucha
0,menor,menor,medio,menor,medio,medio,mayor,medio,poca
0,menor,menor,medio,menor,medio,medio,mayor,menor,mucha
1,menor,menor,medio,menor,medio,medio,mayor,menor,poca
0,menor,menor,medio,menor,medio,medio,medio,medio,mucha
0,menor,menor,medio,menor,medio,medio,medio,medio,poca
0,menor,menor,menor,menor,medio,medio,mayor,medio,mucha
1,menor,menor,menor,menor,medio,medio,mayor,medio,poca
10,menor,menor,menor,menor,medio,medio,mayor,menor,poca
0,menor,menor,menor,menor,medio,medio,mayor,menor,mucha
0,menor,menor,menor,menor,medio,medio,medio,medio,mucha
1,menor,menor,menor,menor,medio,medio,medio,medio,poca
0,menor,menor,menor,menor,menor,medio,mayor,medio,mucha
1,menor,menor,menor,menor,menor,medio,mayor,medio,poca
0,menor,menor,menor,menor,menor,menor,mayor,menor,mucha
0,menor,menor,menor,menor,menor,menor,mayor,menor,poca
Numero de Registros que se recuperan : 72

Reglas sobreviven en archivo de texto
:D:\SERIE\Preprocesamiento_M04\Oct2002-MallasHierbas_00SupervivientesReglas.tx
t
```

Numero de Registros SOBREVIVEN : 66

(6.B)_____

```
SELECT DISTINCTROW [05_381].pendiente, [05_381].radiacion_anual, [05_381].pH,
[05_381].N, [05_381].P, [05_381].K, [05_381].MO, [05_381].SEMILLAS,
[05_381].tipo_de_suelo, [05_381].coverid INTO TablaRegistrosEliminadosID  FROM
05_381 LEFT JOIN TablaLimpia_DEF ON [05_381].coverid =
[TablaLimpia_DEF].coverid WHERE (((TablaLimpia_DEF.coverid) Is Null))  ORDER
BY [05_381].pendiente, [05_381].radiacion_anual, [05_381].pH, [05_381].N,
[05_381].P, [05_381].K, [05_381].MO, [05_381].SEMILLAS,
[05_381].tipo_de_suelo, [TablaLimpia_DEF].pendiente,
[TablaLimpia_DEF].radiacion_anual, [TablaLimpia_DEF].pH, [TablaLimpia_DEF].N,
[TablaLimpia_DEF].P, [TablaLimpia_DEF].K, [TablaLimpia_DEF].MO,
[TablaLimpia_DEF].SEMILLAS, [TablaLimpia_DEF].tipo_de_suelo;
```

Creación de la Consulta : [consultaRegistrosIdEliminados]

Creación de la Tabla : [TablaRegistrosEliminadosID]

Numero de Registros ELIMINADOS con su ID : 315

(7)_Fase_de_recuperación_de_registros_____

De los registros eliminados es posible extraer algunos que pueden representarse en el grupo en el que se presenten en mayor proporcion. Entonces se asume que ese registro tendrá mayor posibilidad de pertenecer a la clase en la que exista más veces

```
1       clase 1,mayor,mas basico,normal-alto,alto,alto,alta,franco
arcilloso,menos
0       clase 1,mayor,mas basico,normal-alto,alto,alto,alta,franco
arcilloso,mas
1       clase 2,mayor,mas basico,normal,alto,alto,medio-alta,franco
arcilloso,mas
0       clase 2,mayor,mas basico,normal,alto,alto,medio-alta,franco
arcilloso,menos
0       clase 2,mayor,mas basico,normal,alto,bajo,medio-baja,franco,mas
1       clase 2,mayor,mas basico,normal,alto,bajo,medio-baja,franco,menos
0       clase 2,mayor,menos basico,normal-alto,alto,alto,alta,franco
arcilloso,mas
1       clase 2,mayor,menos basico,normal-alto,alto,alto,alta,franco
arcilloso,menos
1       clase 2,mayor,menos basico,normal-alto,alto,medio,alta,franco
arcillo-limoso,mas
0       clase 2,mayor,menos basico,normal-alto,alto,medio,alta,franco
arcillo-limoso,menos
0       clase 2,menor,menos basico,normal-alto,alto,medio,medio-alta,franco
arcilloso,mas
1       clase 2,menor,menos basico,normal-alto,alto,medio,medio-alta,franco
arcilloso,menos
0       clase 3,mayor,mas basico,normal,alto,medio,alta,franco arcilloso,mas
1       clase 3,mayor,mas basico,normal,alto,medio,alta,franco arcilloso,menos
0       clase 3,mayor,mas basico,normal,medio,bajo,medio-baja,franco,mas
1       clase 3,mayor,mas basico,normal,medio,bajo,medio-baja,franco,menos
0       clase 3,mayor,mas basico,normal,muy alto,alto,alta,franco
arcillo-limoso,mas
1       clase 3,mayor,mas basico,normal,muy alto,alto,alta,franco
arcillo-limoso,menos
1       clase 3,mayor,mas basico,normal,muy alto,alto,medio-baja,franco
limoso,menos
0       clase 3,mayor,mas basico,normal,muy alto,alto,medio-baja,franco
limoso,mas
0       clase 3,mayor,mas basico,normal,muy alto,medio,media,franco,mas
1       clase 3,mayor,mas basico,normal,muy alto,medio,media,franco,menos
1       clase 3,mayor,mas basico,normal,muy alto,medio,medio-alta,franco,mas
0       clase 3,mayor,mas basico,normal,muy alto,medio,medio-alta,franco,menos
0       clase 3,mayor,mas basico,normal-alto,alto,bajo,medio-baja,franco,mas
1       clase 3,mayor,mas basico,normal-alto,alto,bajo,medio-baja,franco,menos
```


(4)_____

```
SELECT [Duplicados_CuentaMH_diferentes].pendiente,
[Duplicados_CuentaMH_diferentes].radiacion_anual,
[Duplicados_CuentaMH_diferentes].pH, [Duplicados_CuentaMH_diferentes].N,
[Duplicados_CuentaMH_diferentes].P, [Duplicados_CuentaMH_diferentes].K,
[Duplicados_CuentaMH_diferentes].MO,
[Duplicados_CuentaMH_diferentes].SEMILLAS,
[Duplicados_CuentaMH_diferentes].tipo_de_suelo,
[Duplicados_CuentaMH_diferentes].CuentaDeClave  INTO
RegistrosDuplicadosPareados  FROM Duplicados_CuentaMH_diferentes,
DuplicadosEnAtributosTabla  WHERE [Duplicados_CuentaMH_diferentes].pendiente =
[DuplicadosEnAtributosTabla].pendiente AND
[Duplicados_CuentaMH_diferentes].radiacion_anual =
[DuplicadosEnAtributosTabla].radiacion_anual AND
[Duplicados_CuentaMH_diferentes].pH = [DuplicadosEnAtributosTabla].pH AND
[Duplicados_CuentaMH_diferentes].N = [DuplicadosEnAtributosTabla].N AND
[Duplicados_CuentaMH_diferentes].P = [DuplicadosEnAtributosTabla].P AND
[Duplicados_CuentaMH_diferentes].K = [DuplicadosEnAtributosTabla].K AND
[Duplicados_CuentaMH_diferentes].MO = [DuplicadosEnAtributosTabla].MO AND
[Duplicados_CuentaMH_diferentes].tipo_de_suelo =
[DuplicadosEnAtributosTabla].tipo_de_suelo
```

Las consultas (4) y (5) seleccionan los registros que verdaderamente han de ser eliminados

Creación de la Consulta : [RegistrosDuplicados_cuenta_a_Tbl]

Creación de la Tabla : [RegistrosDuplicadosPareados]

Numero de Registros que deben borrarse; TOTALES) : 160

(5)_____

Creación de la Consulta : [RegistrosAborrar]

```
SELECT [05_381].pendiente, [05_381].radiacion_anual, [05_381].pH, [05_381].N,
[05_381].P, [05_381].K, [05_381].MO, [05_381].SEMILLAS,
[05_381].tipo_de_suelo,  [05_381].coverid INTO TotalRegistrosEliminar  FROM
05_381, RegistrosDuplicadosPareados WHERE [05_381].pendiente =
[RegistrosDuplicadosPareados].pendiente AND [05_381].radiacion_anual =
[RegistrosDuplicadosPareados].radiacion_anual AND [05_381].pH =
[RegistrosDuplicadosPareados].pH AND [05_381].N =
[RegistrosDuplicadosPareados].N AND [05_381].P =
[RegistrosDuplicadosPareados].P AND [05_381].K =
[RegistrosDuplicadosPareados].K AND [05_381].MO =
[RegistrosDuplicadosPareados].MO AND [05_381].SEMILLAS =
[RegistrosDuplicadosPareados].SEMILLAS AND [05_381].tipo_de_suelo =
[RegistrosDuplicadosPareados].tipo_de_suelo
```

Creación de la Tabla : [TotalRegistrosEliminar]

Numero de Registros que deben borrarse; TOTALES con duplicados) : 315

(6.A)_Tabla_Limpia_sin_ningún_duplicado_____

```
SELECT DISTINCTROW [05_381].pendiente, [05_381].radiacion_anual, [05_381].pH,
[05_381].N, [05_381].P, [05_381].K, [05_381].MO, [05_381].SEMILLAS,
[05_381].tipo_de_suelo, [05_381].coverid INTO TablaLimpia_DEF  FROM 05_381
LEFT JOIN TotalRegistrosEliminar ON [05_381].coverid =
[TotalRegistrosEliminar].coverid WHERE ((([TotalRegistrosEliminar].coverid) Is
Null));
```

Creación de la Consulta : [TablaDefinitivaLimpia]

Creación de la Tabla : [TablaLimpia_DEF]

0 clase 3,menor,mas basico,normal,alto,bajo,medio-baja,franco,menos
1 clase 3,menor,mas basico,normal,alto,bajo,medio-baja,franco
arcillo-limoso,mas
0 clase 3,menor,mas basico,normal,alto,bajo,medio-baja,franco
arcillo-limoso,menos
3 clase 3,menor,mas basico,normal,alto,medio,medio-alta,franco,menos
0 clase 3,menor,mas basico,normal,alto,medio,medio-alta,franco,mas
0 clase 3,menor,mas basico,normal,medio,bajo,medio-baja,franco
arcilloso,menos
1 clase 3,menor,mas basico,normal,medio,bajo,medio-baja,franco
arcilloso,mas
4 clase 3,menor,mas basico,normal,muy alto,bajo,medio-alta,franco,mas
0 clase 3,menor,mas basico,normal,muy alto,bajo,medio-alta,franco,menos
1 clase 3,menor,mas basico,normal,muy alto,bajo,medio-baja,franco,mas
0 clase 3,menor,mas basico,normal,muy alto,bajo,medio-baja,franco,menos
0 clase 3,menor,mas basico,normal-alto,alto,bajo,medio-alta,franco,menos
1 clase 3,menor,mas basico,normal-alto,alto,bajo,medio-alta,franco,mas
1 clase 3,menor,mas basico,normal-alto,alto,bajo,medio-baja,franco
limoso,menos
0 clase 3,menor,mas basico,normal-alto,alto,bajo,medio-baja,franco
limoso,mas
0 clase 3,menor,mas basico,normal-alto,alto,medio,media,franco
arcilloso,mas
1 clase 3,menor,mas basico,normal-alto,alto,medio,media,franco
arcilloso,menos
1 clase 3,menor,mas basico,normal-alto,medio,alto,alta,franco limoso,mas
0 clase 3,menor,mas basico,normal-alto,medio,alto,alta,franco
limoso,menos
1 clase 3,menor,mas basico,normal-alto,medio,bajo,media,franco
limoso,mas
0 clase 3,menor,mas basico,normal-alto,medio,bajo,media,franco
limoso,menos
0 clase 3,menor,mas basico,normal-alto,medio,bajo,medio-alta,franco,mas
1 clase 3,menor,mas
basico,normal-alto,medio,bajo,medio-alta,franco,menos
0 clase 3,menor,mas basico,normal-alto,muy alto,bajo,media,franco
limoso,menos
1 clase 3,menor,mas basico,normal-alto,muy alto,bajo,media,franco
limoso,mas
0 clase 3,menor,menos basico,normal,alto,medio,medio-alta,franco
arcilloso,mas
2 clase 3,menor,menos basico,normal,alto,medio,medio-alta,franco
arcilloso,menos
0 clase 3,menor,menos basico,normal,muy alto,medio,medio-alta,franco,mas
4 clase 3,menor,menos basico,normal,muy
alto,medio,medio-alta,franco,menos
1 clase 3,menor,menos basico,normal,muy alto,medio,medio-alta,franco
limoso,menos
0 clase 3,menor,menos basico,normal,muy alto,medio,medio-alta,franco
limoso,mas
0 clase 3,menor,menos basico,normal,muy alto,muy bajo,medio-alta,franco
limoso,menos
1 clase 3,menor,menos basico,normal,muy alto,muy bajo,medio-alta,franco
limoso,mas
1 clase 3,menor,menos basico,normal-alto,alto,bajo,media,franco,mas
0 clase 3,menor,menos basico,normal-alto,alto,bajo,media,franco,menos
1 clase 3,menor,menos basico,normal-alto,alto,bajo,media,franco
limoso,mas
0 clase 3,menor,menos basico,normal-alto,alto,bajo,media,franco
limoso,menos
0 clase 3,menor,menos basico,normal-alto,alto,bajo,medio-alta,franco,mas
1 clase 3,menor,menos
basico,normal-alto,alto,bajo,medio-alta,franco,menos
0 clase 3,menor,menos basico,normal-alto,alto,bajo,medio-alta,franco
arcillo-limoso,menos
1 clase 3,menor,menos basico,normal-alto,alto,bajo,medio-alta,franco
arcillo-limoso,mas
0 clase 3,menor,menos basico,normal-alto,alto,bajo,medio-alta,franco

0 clase 3,mayor,mas basico,normal-alto,medio,medio,media,arcilloso,mas
1 clase 3,mayor,mas basico,normal-alto,medio,medio,media,arcilloso,menos
1 clase 3,mayor,menos basico,normal,muy alto,alto,medio-alta,franco
arcilloso,menos
0 clase 3,mayor,menos basico,normal,muy alto,alto,medio-alta,franco
arcilloso,mas
1 clase 3,mayor,menos basico,normal-alto,alto,alto,alta,arcillo
limoso,menos
0 clase 3,mayor,menos basico,normal-alto,alto,alto,alta,arcillo
limoso,mas
0 clase 3,mayor,menos basico,normal-alto,alto,alto,alta,franco
arcilloso,menos
1 clase 3,mayor,menos basico,normal-alto,alto,alto,alta,franco
arcilloso,mas
1 clase 3,mayor,menos basico,normal-alto,alto,alto,medio-alta,franco,mas
0 clase 3,mayor,menos
basico,normal-alto,alto,alto,medio-alta,franco,menos
1 clase 3,mayor,menos basico,normal-alto,alto,bajo,medio-alta,franco
limoso,mas
0 clase 3,mayor,menos basico,normal-alto,alto,bajo,medio-alta,franco
limoso,menos
0 clase 3,mayor,menos basico,normal-alto,alto,medio,alta,franco,mas
1 clase 3,mayor,menos basico,normal-alto,alto,medio,alta,franco,menos
1 clase 3,mayor,menos basico,normal-alto,alto,medio,alta,franco
arcillo-limoso,menos
0 clase 3,mayor,menos basico,normal-alto,alto,medio,alta,franco
arcillo-limoso,mas
0 clase 3,mayor,menos basico,normal-alto,alto,medio,alta,franco
arcilloso,mas
5 clase 3,mayor,menos basico,normal-alto,alto,medio,alta,franco
arcilloso,menos
0 clase 3,mayor,menos
basico,normal-alto,alto,medio,medio-alta,franco,mas
2 clase 3,mayor,menos
basico,normal-alto,alto,medio,medio-alta,franco,menos
0 clase 3,mayor,menos basico,normal-alto,alto,medio,medio-alta,franco
arcilloso,mas
1 clase 3,mayor,menos basico,normal-alto,alto,medio,medio-alta,franco
arcilloso,menos
1 clase 3,mayor,menos
basico,normal-alto,alto,medio,medio-baja,franco,mas
0 clase 3,mayor,menos
basico,normal-alto,alto,medio,medio-baja,franco,menos
1 clase 3,mayor,menos basico,normal-alto,muy alto,alto,alta,franco
arcilloso,menos
0 clase 3,mayor,menos basico,normal-alto,muy alto,alto,alta,franco
arcilloso,mas
1 clase 3,mayor,menos basico,normal-alto,muy alto,medio,alta,franco
limoso,menos
0 clase 3,mayor,menos basico,normal-alto,muy alto,medio,alta,franco
limoso,mas
1 clase 3,mayor,menos basico,normal-alto,muy alto,medio,media,franco
arcillo-limoso,mas
0 clase 3,mayor,menos basico,normal-alto,muy alto,medio,media,franco
arcillo-limoso,menos
0 clase 3,mayor,menos basico,normal-alto,muy
alto,medio,medio-alta,franco arcilloso,mas
1 clase 3,mayor,menos basico,normal-alto,muy
alto,medio,medio-alta,franco arcilloso,menos
0 clase 3,menor,mas basico,normal,alto,bajo,media,franco,menos
2 clase 3,menor,mas basico,normal,alto,bajo,media,franco,mas
0 clase 3,menor,mas basico,normal,alto,bajo,media,franco arcilloso,menos
1 clase 3,menor,mas basico,normal,alto,bajo,media,franco arcilloso,mas
2 clase 3,menor,mas basico,normal,alto,bajo,media,franco limoso,mas
0 clase 3,menor,mas basico,normal,alto,bajo,media,franco limoso,menos
0 clase 3,menor,mas basico,normal,alto,bajo,medio-alta,franco,mas
1 clase 3,menor,mas basico,normal,alto,bajo,medio-alta,franco,menos
1 clase 3,menor,mas basico,normal,alto,bajo,medio-baja,franco,mas

arcillo-limoso,menos
1 clase 4,mayor,menos basico,normal-alto,alto,medio,media,franco
arcillo-limoso,mas
0 clase 4,mayor,menos basico,normal-alto,alto,medio,medio-alta,franco
arcilloso,mas
1 clase 4,mayor,menos basico,normal-alto,alto,medio,medio-alta,franco
arcilloso,menos
0 clase 4,mayor,menos basico,normal-alto,muy
alto,medio,medio-alta,franco limoso,mas
1 clase 4,mayor,menos basico,normal-alto,muy
alto,medio,medio-alta,franco limoso,menos
1 clase 4,menor,mas basico,normal,muy alto,bajo,media,franco
arcillo-limoso,mas
0 clase 4,menor,mas basico,normal,muy alto,bajo,media,franco
arcillo-limoso,menos
0 clase 4,menor,menos basico,normal-alto,alto,bajo,media,franco
arcillo-limoso,mas
1 clase 4,menor,menos basico,normal-alto,alto,bajo,media,franco
arcillo-limoso,menos
0 clase 4,menor,menos basico,normal-alto,alto,medio,medio-alta,franco
arcillo-limoso,menos
1 clase 4,menor,menos basico,normal-alto,alto,medio,medio-alta,franco
arcillo-limoso,mas
Numero de Registros que se recuperan : 97

Reglas sobreviven en archivo de texto
:D:\beatriz\Programas\Preprocesamiento_M04\barcelona_381_00supervivientesRegla
s.txt

arcilloso,menos
0 clase 3,menor,menos basico,normal-alto,alto,bajo,medio-alta,franco
arcilloso,mas
1 clase 3,menor,menos basico,normal-alto,alto,medio,alta,franco
arcilloso,menos
0 clase 3,menor,menos basico,normal-alto,alto,medio,alta,franco
arcilloso,mas
1 clase 3,menor,menos basico,normal-alto,alto,medio,media,franco,mas
0 clase 3,menor,menos basico,normal-alto,alto,medio,media,franco,menos
1 clase 3,menor,menos basico,normal-alto,alto,medio,media,franco
arcilloso,mas
0 clase 3,menor,menos basico,normal-alto,alto,medio,media,franco
arcilloso,menos
0 clase 3,menor,menos
basico,normal-alto,alto,medio,medio-alta,franco,mas
3 clase 3,menor,menos
basico,normal-alto,alto,medio,medio-alta,franco,menos
0 clase 3,menor,menos basico,normal-alto,muy
alto,bajo,medio-alta,franco,menos
1 clase 3,menor,menos basico,normal-alto,muy
alto,bajo,medio-alta,franco,mas
1 clase 3,menor,menos basico,normal-alto,muy
alto,medio,medio-alta,franco,mas
0 clase 3,menor,menos basico,normal-alto,muy
alto,medio,medio-alta,franco,menos
0 clase 4,mayor,mas basico,normal,alto,bajo,media,franco,mas
1 clase 4,mayor,mas basico,normal,alto,bajo,media,franco,menos
1 clase 4,mayor,mas basico,normal,medio,medio,media,franco
arcillo-limoso,mas
0 clase 4,mayor,mas basico,normal,medio,medio,media,franco
arcillo-limoso,menos
1 clase 4,mayor,mas basico,normal-alto,alto,bajo,medio-baja,franco,menos
0 clase 4,mayor,mas basico,normal-alto,alto,bajo,medio-baja,franco,mas
1 clase 4,mayor,mas basico,normal-alto,alto,bajo,medio-baja,franco
limoso,menos
0 clase 4,mayor,mas basico,normal-alto,alto,bajo,medio-baja,franco
limoso,mas
0 clase 4,mayor,mas
basico,normal-alto,alto,medio,medio-alta,franco,menos
1 clase 4,mayor,mas basico,normal-alto,alto,medio,medio-alta,franco,mas
0 clase 4,mayor,mas basico,normal-alto,alto,medio,medio-baja,franco
arcillo-limoso,mas
1 clase 4,mayor,mas basico,normal-alto,alto,medio,medio-baja,franco
arcillo-limoso,menos
1 clase 4,mayor,mas basico,normal-alto,muy alto,medio,alta,franco,mas
0 clase 4,mayor,mas basico,normal-alto,muy alto,medio,alta,franco,menos
1 clase 4,mayor,mas basico,normal-alto,muy alto,medio,media,franco
arcillo-limoso,mas
0 clase 4,mayor,mas basico,normal-alto,muy alto,medio,media,franco
arcillo-limoso,menos
0 clase 4,mayor,menos basico,normal,alto,alto,medio-alta,franco
arcilloso,menos
1 clase 4,mayor,menos basico,normal,alto,alto,medio-alta,franco
arcilloso,mas
1 clase 4,mayor,menos basico,normal,alto,bajo,medio-alta,franco,menos
0 clase 4,mayor,menos basico,normal,alto,bajo,medio-alta,franco,mas
1 clase 4,mayor,menos basico,normal,alto,medio,media,franco
arcilloso,mas
0 clase 4,mayor,menos basico,normal,alto,medio,media,franco
arcilloso,menos
0 clase 4,mayor,menos basico,normal,muy alto,medio,medio-alta,franco,mas
1 clase 4,mayor,menos basico,normal,muy
alto,medio,medio-alta,franco,menos
1 clase 4,mayor,menos basico,normal-alto,alto,alto,alta,franco
arcilloso,mas
0 clase 4,mayor,menos basico,normal-alto,alto,alto,alta,franco
arcilloso,menos
0 clase 4,mayor,menos basico,normal-alto,alto,medio,media,franco

```
[Duplicados].radiacion_anual_acumulada, [Duplicados].pH,
[Duplicados].Conductibilidad, [Duplicados].N, [Duplicados].P, [Duplicados].K,
[Duplicados].Humedad, [Duplicados].OM,  Count(Duplicados.id) AS CuentaDeClave
INTO duplicados_cuentaMH_diferentes  FROM Duplicados GROUP BY
[Duplicados].biomasa_trigo_2001, [Duplicados].tipo_de_suelo,
[Duplicados].biomasa_de_avena_2001,
[Duplicados].biomasa_de_paja_del_trigo_2001,
[Duplicados].biomasa_de_espigas_del_trigo_2001,
[Duplicados].biomasa_de_grano_de_trigo_2001,
[Duplicados].biomasa_del_lolium_2001, [Duplicados].sand, [Duplicados].silt,
[Duplicados].clay, [Duplicados].z_relativa, [Duplicados].categoria_NESW,
[Duplicados].presencia_de_paja2001, [Duplicados].pendiente_en_gr,
[Duplicados].radiacion_anual_acumulada, [Duplicados].pH,
[Duplicados].Conductibilidad, [Duplicados].N, [Duplicados].P, [Duplicados].K,
[Duplicados].Humedad, [Duplicados].OM
```

Esta tabla, sin embargo, aún contiene información que nos interesa, que son los registros duplicados que solo pertenecen a una clase.

Esta segunda agrupación de los registros no crea una tabla que contiene el numero de veces que se duplica cada registro. En esta tabla se puede observar que existen registros que están en las dos clases y otros que se repiten solo en una. Para filtrar esta importante información y evitar las inconsistencias se realiza la consulta (3)

Creación de la Consulta : [Duplicados_and_MHDif_2]

Creación de la Tabla : [Duplicados_and_MHDif_1]

Numero de duplicados POR REGISTRO: 16

(3)_____

```
SELECT [Duplicados_and_MHDif_1].biomasa_trigo_2001,
[Duplicados_and_MHDif_1].tipo_de_suelo,
[Duplicados_and_MHDif_1].biomasa_de_paja_del_trigo_2001,
[Duplicados_and_MHDif_1].biomasa_de_espigas_del_trigo_2001,
[Duplicados_and_MHDif_1].biomasa_de_grano_de_trigo_2001,
[Duplicados_and_MHDif_1].biomasa_del_lolium_2001,
[Duplicados_and_MHDif_1].sand, [Duplicados_and_MHDif_1].silt,
[Duplicados_and_MHDif_1].clay, [Duplicados_and_MHDif_1].z_relativa,
[Duplicados_and_MHDif_1].categoria_NESW,
[Duplicados_and_MHDif_1].presencia_de_paja2001,
[Duplicados_and_MHDif_1].pendiente_en_gr,
[Duplicados_and_MHDif_1].radiacion_anual_acumulada,
[Duplicados_and_MHDif_1].pH, [Duplicados_and_MHDif_1].Conductibilidad,
[Duplicados_and_MHDif_1].N, [Duplicados_and_MHDif_1].P,
[Duplicados_and_MHDif_1].K, [Duplicados_and_MHDif_1].Humedad,
[Duplicados_and_MHDif_1].OM, Count(Duplicados_and_MHDif_1.CuentaDeClave) AS
CuentaDeCuentaDeClave INTO DuplicadosEnAtributostabla  FROM
Duplicados_and_MHDif_1 GROUP BY [Duplicados_and_MHDif_1].biomasa_trigo_2001,
[Duplicados_and_MHDif_1].tipo_de_suelo,
[Duplicados_and_MHDif_1].biomasa_de_paja_del_trigo_2001,
[Duplicados_and_MHDif_1].biomasa_de_espigas_del_trigo_2001,
[Duplicados_and_MHDif_1].biomasa_de_grano_de_trigo_2001,
[Duplicados_and_MHDif_1].biomasa_del_lolium_2001,
[Duplicados_and_MHDif_1].sand, [Duplicados_and_MHDif_1].silt,
[Duplicados_and_MHDif_1].clay, [Duplicados_and_MHDif_1].z_relativa,
[Duplicados_and_MHDif_1].categoria_NESW,
[Duplicados_and_MHDif_1].presencia_de_paja2001,
[Duplicados_and_MHDif_1].pendiente_en_gr,
[Duplicados_and_MHDif_1].radiacion_anual_acumulada,
[Duplicados_and_MHDif_1].pH, [Duplicados_and_MHDif_1].Conductibilidad,
[Duplicados_and_MHDif_1].N, [Duplicados_and_MHDif_1].P,
[Duplicados_and_MHDif_1].K, [Duplicados_and_MHDif_1].Humedad,
[Duplicados_and_MHDif_1].OM HAVING
(((Count(Duplicados_and_MHDif_1.CuentaDeClave)) > 1));
```


Número de registros :254
Tipo de datos : NUMERICOS

(1)_____

```
SELECT DISTINCTROW [ENTRAMIENTO].biomasa_trigo_2001,
[ENTRAMIENTO].tipo_de_suelo, [ENTRAMIENTO].biomasa_de_avena_2001,
[ENTRAMIENTO].biomasa_de_paja_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_espigas_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_grano_de_trigo_2001,
[ENTRAMIENTO].biomasa_del_lolium_2001, [ENTRAMIENTO].sand, [ENTRAMIENTO].silt,
[ENTRAMIENTO].clay, [ENTRAMIENTO].z_relativa, [ENTRAMIENTO].categoria_NESW,
[ENTRAMIENTO].presencia_de_paja2001, [ENTRAMIENTO].pendiente_en_gr,
[ENTRAMIENTO].radiacion_anual_acumulada, [ENTRAMIENTO].pH,
[ENTRAMIENTO].Conductibilidad, [ENTRAMIENTO].N, [ENTRAMIENTO].P,
[ENTRAMIENTO].K, [ENTRAMIENTO].Humedad, [ENTRAMIENTO].OM, [ENTRAMIENTO].id
FROM ENTRAMIENTO WHERE ((([ENTRAMIENTO].tipo_de_suelo) In (SELECT
tipo_de_suelo FROM ENTRAMIENTO AS Tmp  GROUP BY [biomasa_trigo_2001],
[tipo_de_suelo], [biomasa_de_paja_del_trigo_2001],
[biomasa_de_espigas_del_trigo_2001], [biomasa_de_grano_de_trigo_2001],
[biomasa_del_lolium_2001], [sand], [silt], [clay], [z_relativa],
[categoria_NESW], [presencia_de_paja2001], [pendiente_en_gr],
[radiacion_anual_acumulada], [pH], [Conductibilidad], [N], [P], [K],
[Humedad], [OM] HAVING Count(*)>1  And [biomasa_trigo_2001] =
[ENTRAMIENTO].biomasa_trigo_2001 And [tipo_de_suelo] =
[ENTRAMIENTO].tipo_de_suelo And [biomasa_de_paja_del_trigo_2001] =
[ENTRAMIENTO].biomasa_de_paja_del_trigo_2001 And
[biomasa_de_espigas_del_trigo_2001] =
[ENTRAMIENTO].biomasa_de_espigas_del_trigo_2001 And
[biomasa_de_grano_de_trigo_2001] =
[ENTRAMIENTO].biomasa_de_grano_de_trigo_2001 And [biomasa_del_lolium_2001] =
[ENTRAMIENTO].biomasa_del_lolium_2001 And [sand] = [ENTRAMIENTO].sand And
[silt] = [ENTRAMIENTO].silt And [clay] = [ENTRAMIENTO].clay And [z_relativa] =
[ENTRAMIENTO].z_relativa And [categoria_NESW] = [ENTRAMIENTO].categoria_NESW
And [presencia_de_paja2001] = [ENTRAMIENTO].presencia_de_paja2001 And
[pendiente_en_gr] = [ENTRAMIENTO].pendiente_en_gr And
[radiacion_anual_acumulada] = [ENTRAMIENTO].radiacion_anual_acumulada And [pH]
= [ENTRAMIENTO].pH And [Conductibilidad] = [ENTRAMIENTO].Conductibilidad And
[N] = [ENTRAMIENTO].N And [P] = [ENTRAMIENTO].P And [K] = [ENTRAMIENTO].K And
[Humedad] = [ENTRAMIENTO].Humedad And [OM] = [ENTRAMIENTO].OM))) ORDER BY
[ENTRAMIENTO].biomasa_trigo_2001, [ENTRAMIENTO].tipo_de_suelo,
[ENTRAMIENTO].biomasa_de_paja_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_espigas_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_grano_de_trigo_2001,
[ENTRAMIENTO].biomasa_del_lolium_2001, [ENTRAMIENTO].sand, [ENTRAMIENTO].silt,
[ENTRAMIENTO].clay, [ENTRAMIENTO].z_relativa, [ENTRAMIENTO].categoria_NESW,
[ENTRAMIENTO].presencia_de_paja2001, [ENTRAMIENTO].pendiente_en_gr,
[ENTRAMIENTO].radiacion_anual_acumulada, [ENTRAMIENTO].pH,
[ENTRAMIENTO].Conductibilidad, [ENTRAMIENTO].N, [ENTRAMIENTO].P,
[ENTRAMIENTO].K, [ENTRAMIENTO].Humedad, [ENTRAMIENTO].OM;
```

Creación de la consulta : [Duplicados]

Numero de duplicados TOTALES: 23

En primer lugar se realiza una agrupación de los registros en función de todos los atributos excepto los que forman las clases.

(2)_____

```
SELECT [Duplicados].biomasa_trigo_2001, [Duplicados].tipo_de_suelo,
[Duplicados].biomasa_de_avena_2001,
[Duplicados].biomasa_de_paja_del_trigo_2001,
[Duplicados].biomasa_de_espigas_del_trigo_2001,
[Duplicados].biomasa_de_grano_de_trigo_2001,
[Duplicados].biomasa_del_lolium_2001, [Duplicados].sand, [Duplicados].silt,
[Duplicados].clay, [Duplicados].z_relativa, [Duplicados].categoria_NESW,
[Duplicados].presencia_de_paja2001, [Duplicados].pendiente_en_gr,
```

Creación de la Tabla : [RegistrosDuplicadosPareados]

Numero de Registros que deben borrarse; TOTALES) : 12

(5)_____

Creación de la Consulta : [RegistrosAborrar]

```
SELECT [ENTRAMIENTO].biomasa_trigo_2001, [ENTRAMIENTO].tipo_de_suelo,
[ENTRAMIENTO].biomasa_de_avena_2001,
[ENTRAMIENTO].biomasa_de_paja_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_espigas_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_grano_de_trigo_2001,
[ENTRAMIENTO].biomasa_del_lolium_2001, [ENTRAMIENTO].sand, [ENTRAMIENTO].silt,
[ENTRAMIENTO].clay, [ENTRAMIENTO].z_relativa, [ENTRAMIENTO].categoria_NESW,
[ENTRAMIENTO].presencia_de_paja2001, [ENTRAMIENTO].pendiente_en_gr,
[ENTRAMIENTO].radiacion_anual_acumulada, [ENTRAMIENTO].pH,
[ENTRAMIENTO].conductibilidad, [ENTRAMIENTO].N, [ENTRAMIENTO].P,
[ENTRAMIENTO].K, [ENTRAMIENTO].Humedad, [ENTRAMIENTO].OM, [ENTRAMIENTO].id
INTO TotalRegistrosEliminar  FROM ENTRAMIENTO, RegistrosDuplicadosPareados
WHERE [ENTRAMIENTO].biomasa_trigo_2001 =
[RegistrosDuplicadosPareados].biomasa_trigo_2001 AND
[ENTRAMIENTO].tipo_de_suelo = [RegistrosDuplicadosPareados].tipo_de_suelo AND
[ENTRAMIENTO].biomasa_de_avena_2001 =
[RegistrosDuplicadosPareados].biomasa_de_avena_2001 AND
[ENTRAMIENTO].biomasa_de_paja_del_trigo_2001 =
[RegistrosDuplicadosPareados].biomasa_de_paja_del_trigo_2001 AND
[ENTRAMIENTO].biomasa_de_espigas_del_trigo_2001 =
[RegistrosDuplicadosPareados].biomasa_de_espigas_del_trigo_2001 AND
[ENTRAMIENTO].biomasa_de_grano_de_trigo_2001 =
[RegistrosDuplicadosPareados].biomasa_de_grano_de_trigo_2001 AND
[ENTRAMIENTO].biomasa_del_lolium_2001 =
[RegistrosDuplicadosPareados].biomasa_del_lolium_2001 AND [ENTRAMIENTO].sand =
[RegistrosDuplicadosPareados].sand AND [ENTRAMIENTO].silt =
[RegistrosDuplicadosPareados].silt AND [ENTRAMIENTO].clay =
[RegistrosDuplicadosPareados].clay AND [ENTRAMIENTO].z_relativa =
[RegistrosDuplicadosPareados].z_relativa AND [ENTRAMIENTO].categoria_NESW =
[RegistrosDuplicadosPareados].categoria_NESW AND
[ENTRAMIENTO].presencia_de_paja2001 =
[RegistrosDuplicadosPareados].presencia_de_paja2001 AND
[ENTRAMIENTO].pendiente_en_gr = [RegistrosDuplicadosPareados].pendiente_en_gr
AND [ENTRAMIENTO].radiacion_anual_acumulada =
[RegistrosDuplicadosPareados].radiacion_anual_acumulada AND [ENTRAMIENTO].pH =
[RegistrosDuplicadosPareados].pH AND [ENTRAMIENTO].conductibilidad =
[RegistrosDuplicadosPareados].conductibilidad AND [ENTRAMIENTO].N =
[RegistrosDuplicadosPareados].N AND [ENTRAMIENTO].P =
[RegistrosDuplicadosPareados].P AND [ENTRAMIENTO].K =
[RegistrosDuplicadosPareados].K AND [ENTRAMIENTO].Humedad =
[RegistrosDuplicadosPareados].Humedad AND [ENTRAMIENTO].OM =
[RegistrosDuplicadosPareados].OM
```

Creación de la Tabla : [TotalRegistrosEliminar]

Numero de Registros que deben borrarse; TOTALES con duplicados) : 15

(6.A)_Tabla_Limpia_sin_ningún_duplicado_____

```
SELECT DISTINCTROW [ENTRAMIENTO].biomasa_trigo_2001,
[ENTRAMIENTO].tipo_de_suelo, [ENTRAMIENTO].biomasa_de_avena_2001,
[ENTRAMIENTO].biomasa_de_paja_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_espigas_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_grano_de_trigo_2001,
[ENTRAMIENTO].biomasa_del_lolium_2001, [ENTRAMIENTO].sand, [ENTRAMIENTO].silt,
[ENTRAMIENTO].clay, [ENTRAMIENTO].z_relativa, [ENTRAMIENTO].categoria_NESW,
[ENTRAMIENTO].presencia_de_paja2001, [ENTRAMIENTO].pendiente_en_gr,
[ENTRAMIENTO].radiacion_anual_acumulada, [ENTRAMIENTO].pH,
[ENTRAMIENTO].conductibilidad, [ENTRAMIENTO].N, [ENTRAMIENTO].P,
```


Creación de la Consulta : [nuevaDuplicadosTabla_2]

Numero de registros representados en los dos grupos, es decir, los que deben ser eliminados : 6

(4)_____

```
SELECT [Duplicados_CuentaMH_diferentes].biomasa_trigo_2001,
[Duplicados_CuentaMH_diferentes].tipo_de_suelo,
[Duplicados_CuentaMH_diferentes].biomasa_de_avena_2001,
[Duplicados_CuentaMH_diferentes].biomasa_de_paja_del_trigo_2001,
[Duplicados_CuentaMH_diferentes].biomasa_de_espigas_del_trigo_2001,
[Duplicados_CuentaMH_diferentes].biomasa_de_grano_de_trigo_2001,
[Duplicados_CuentaMH_diferentes].biomasa_del_lolium_2001,
[Duplicados_CuentaMH_diferentes].sand, [Duplicados_CuentaMH_diferentes].silt,
[Duplicados_CuentaMH_diferentes].clay,
[Duplicados_CuentaMH_diferentes].z_relativa,
[Duplicados_CuentaMH_diferentes].categoria_NESW,
[Duplicados_CuentaMH_diferentes].presencia_de_paja2001,
[Duplicados_CuentaMH_diferentes].pendiente_en_gr,
[Duplicados_CuentaMH_diferentes].radiacion_anual_acumulada,
[Duplicados_CuentaMH_diferentes].pH,
[Duplicados_CuentaMH_diferentes].Conductibilidad,
[Duplicados_CuentaMH_diferentes].N, [Duplicados_CuentaMH_diferentes].P,
[Duplicados_CuentaMH_diferentes].K, [Duplicados_CuentaMH_diferentes].Humedad,
[Duplicados_CuentaMH_diferentes].OM,
[Duplicados_CuentaMH_diferentes].CuentaDeClave  INTO
RegistrosDuplicadosPareados  FROM Duplicados_CuentaMH_diferentes,
DuplicadosEnAtributosTabla   WHERE
[Duplicados_CuentaMH_diferentes].biomasa_trigo_2001 =
[DuplicadosEnAtributosTabla].biomasa_trigo_2001 AND
[Duplicados_CuentaMH_diferentes].tipo_de_suelo =
[DuplicadosEnAtributosTabla].tipo_de_suelo AND
[Duplicados_CuentaMH_diferentes].biomasa_de_paja_del_trigo_2001 =
[DuplicadosEnAtributosTabla].biomasa_de_paja_del_trigo_2001 AND
[Duplicados_CuentaMH_diferentes].biomasa_de_espigas_del_trigo_2001 =
[DuplicadosEnAtributosTabla].biomasa_de_espigas_del_trigo_2001 AND
[Duplicados_CuentaMH_diferentes].biomasa_de_grano_de_trigo_2001 =
[DuplicadosEnAtributosTabla].biomasa_de_grano_de_trigo_2001 AND
[Duplicados_CuentaMH_diferentes].biomasa_del_lolium_2001 =
[DuplicadosEnAtributosTabla].biomasa_del_lolium_2001 AND
[Duplicados_CuentaMH_diferentes].sand = [DuplicadosEnAtributosTabla].sand AND
[Duplicados_CuentaMH_diferentes].silt = [DuplicadosEnAtributosTabla].silt AND
[Duplicados_CuentaMH_diferentes].clay = [DuplicadosEnAtributosTabla].clay AND
[Duplicados_CuentaMH_diferentes].z_relativa =
[DuplicadosEnAtributosTabla].z_relativa AND
[Duplicados_CuentaMH_diferentes].categoria_NESW =
[DuplicadosEnAtributosTabla].categoria_NESW AND
[Duplicados_CuentaMH_diferentes].presencia_de_paja2001 =
[DuplicadosEnAtributosTabla].presencia_de_paja2001 AND
[Duplicados_CuentaMH_diferentes].pendiente_en_gr =
[DuplicadosEnAtributosTabla].pendiente_en_gr AND
[Duplicados_CuentaMH_diferentes].radiacion_anual_acumulada =
[DuplicadosEnAtributosTabla].radiacion_anual_acumulada AND
[Duplicados_CuentaMH_diferentes].pH = [DuplicadosEnAtributosTabla].pH AND
[Duplicados_CuentaMH_diferentes].Conductibilidad =
[DuplicadosEnAtributosTabla].Conductibilidad AND
[Duplicados_CuentaMH_diferentes].N = [DuplicadosEnAtributosTabla].N AND
[Duplicados_CuentaMH_diferentes].P = [DuplicadosEnAtributosTabla].P AND
[Duplicados_CuentaMH_diferentes].K = [DuplicadosEnAtributosTabla].K AND
[Duplicados_CuentaMH_diferentes].Humedad =
[DuplicadosEnAtributosTabla].Humedad AND [Duplicados_CuentaMH_diferentes].OM =
[DuplicadosEnAtributosTabla].OM
```

Las consultas (4) y (5) seleccionan los registros que verdaderamente han de ser eliminados

Creación de la Consulta : [RegistrosDuplicados_cuenta_a_Tbl]

```
0,2,franco,2,2,2,1,3,2,1,2,4,0,1,2,1,3,normal-alto,alto,medio,2,media-alta,no
existe
0,2,franco,2,2,2,1,3,2,1,2,4,0,1,2,1,3,normal-alto,alto,medio,2,media-alta,exi
ste
0,2,franco,2,2,2,1,3,2,1,3,1,0,1,1,2,1,normal-alto,alto,medio,2,media-alta,exi
ste
3,2,franco,2,2,2,1,3,2,1,3,1,0,1,1,2,1,normal-alto,alto,medio,2,media-alta,no
existe
0,2,franco,2,2,2,1,3,2,1,3,4,1,1,2,1,3,normal-alto,alto,medio,2,media-alta,exi
ste
0,2,franco,2,2,2,1,3,2,1,3,4,1,1,2,1,3,normal-alto,alto,medio,2,media-alta,no
existe
0,2,franco,2,3,3,1,3,2,1,3,1,1,1,1,2,1,normal-alto,alto,medio,2,media-alta,no
existe
0,2,franco,2,3,3,1,3,2,1,3,1,1,1,1,2,1,normal-alto,alto,medio,2,media-alta,exi
ste
0,2,franco limoso,2,2,2,1,2,3,1,2,1,0,1,1,2,1,normal-alto,alto,bajo,1,media,no
existe
0,2,franco
limoso,2,2,2,1,2,3,1,2,1,0,1,1,2,1,normal-alto,alto,bajo,1,media,existe
Numero de Registros que se recuperan : 3

Reglas sobreviven en archivo de texto
:D:\Nuevo\programas\2004OB\Preprocesamiento\03PREPROCESADA-discreta_00Supervivi
entesReglas.txt
```



```
[ENTRAMIENTO].K, [ENTRAMIENTO].Humedad, [ENTRAMIENTO].ON, [ENTRAMIENTO].id
INTO TablaLimpia_DEF  FROM ENTRAMIENTO LEFT JOIN TotalRegistrosEliminar ON
[ENTRAMIENTO].id = [TotalRegistrosEliminar].id WHERE
((([TotalRegistrosEliminar].id) Is Null));

Creación de la Consulta : [TablaDefinitivaLimpia]

Creación de la Tabla : [TablaLimpia_DEF]

Numero de Registros SOBREVIVEN : 239

(6.B)_____

SELECT DISTINCTROW [ENTRAMIENTO].biomasa_trigo_2001,
[ENTRAMIENTO].tipo_de_suelo, [ENTRAMIENTO].biomasa_de_avena_2001,
[ENTRAMIENTO].biomasa_de_paja_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_espigas_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_grano_de_trigo_2001,
[ENTRAMIENTO].biomasa_del_lolium_2001, [ENTRAMIENTO].sand, [ENTRAMIENTO].silt,
[ENTRAMIENTO].clay, [ENTRAMIENTO].z_relativa, [ENTRAMIENTO].categoria_NESW,
[ENTRAMIENTO].presencia_de_paja2001, [ENTRAMIENTO].pendiente_en_gr,
[ENTRAMIENTO].radiacion_anual_acumulada, [ENTRAMIENTO].PH,
[ENTRAMIENTO].conductibilidad, [ENTRAMIENTO].N, [ENTRAMIENTO].P,
[ENTRAMIENTO].K, [ENTRAMIENTO].Humedad, [ENTRAMIENTO].ON, [ENTRAMIENTO].id
INTO TablaRegistrosEliminadosID  FROM ENTRAMIENTO LEFT JOIN TablaLimpia_DEF ON
[ENTRAMIENTO].id = [TablaLimpia_DEF].id WHERE (((TablaLimpia_DEF.id) Is Null))
ORDER BY [ENTRAMIENTO].biomasa_trigo_2001, [ENTRAMIENTO].tipo_de_suelo,
[ENTRAMIENTO].biomasa_de_avena_2001,
[ENTRAMIENTO].biomasa_de_paja_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_espigas_del_trigo_2001,
[ENTRAMIENTO].biomasa_de_grano_de_trigo_2001,
[ENTRAMIENTO].biomasa_del_lolium_2001, [ENTRAMIENTO].sand, [ENTRAMIENTO].silt,
[ENTRAMIENTO].clay, [ENTRAMIENTO].z_relativa, [ENTRAMIENTO].categoria_NESW,
[ENTRAMIENTO].presencia_de_paja2001, [ENTRAMIENTO].pendiente_en_gr,
[ENTRAMIENTO].radiacion_anual_acumulada, [ENTRAMIENTO].PH,
[ENTRAMIENTO].conductibilidad, [ENTRAMIENTO].N, [ENTRAMIENTO].P,
[ENTRAMIENTO].K, [ENTRAMIENTO].Humedad, [ENTRAMIENTO].ON,
[TablaLimpia_DEF].biomasa_trigo_2001, [TablaLimpia_DEF].tipo_de_suelo,
[TablaLimpia_DEF].biomasa_de_avena_2001,
[TablaLimpia_DEF].biomasa_de_paja_del_trigo_2001,
[TablaLimpia_DEF].biomasa_de_espigas_del_trigo_2001,
[TablaLimpia_DEF].biomasa_de_grano_de_trigo_2001,
[TablaLimpia_DEF].biomasa_del_lolium_2001, [TablaLimpia_DEF].sand,
[TablaLimpia_DEF].silt, [TablaLimpia_DEF].clay, [TablaLimpia_DEF].z_relativa,
[TablaLimpia_DEF].categoria_NESW, [TablaLimpia_DEF].presencia_de_paja2001,
[TablaLimpia_DEF].pendiente_en_gr,
[TablaLimpia_DEF].radiacion_anual_acumulada, [TablaLimpia_DEF].PH,
[TablaLimpia_DEF].conductibilidad, [TablaLimpia_DEF].N, [TablaLimpia_DEF].P,
[TablaLimpia_DEF].K, [TablaLimpia_DEF].Humedad, [TablaLimpia_DEF].ON;

Creación de la Consulta : [consultaRegistrosIdEliminados]

Creación de la Tabla : [TablaRegistrosEliminadosID]

Numero de Registros ELIMINADOS con su ID : 15

(7)_Fase_de_recuperación_de_registros_____

De los registros eliminados es posible extraer algunos que pueden
representarse en el grupo en el que se presenten en mayor proporcion. Entonces
se asume que ese registro tendrá mayor posibilidad de pertenecer a la clase en
la que exista más veces
0,1,franco
arcillo-limoso,1,1,1,2,1,3,3,2,4,0,3,3,2,2,normal-alto,alto,medio,2,media,no
existe
0,1,franco
arcillo-limoso,1,1,1,2,1,3,3,2,4,0,3,3,2,2,normal-alto,alto,medio,2,media,exis
te
```

Apéndice E

ARCHIVOS FIT RELATIVOS A LOS PROBLEMAS ESTUDIADOS

a,pH,2,mayor,menor
a,N,2,mayor,menor,medio
a,MO,2,mayor,menor,medio
a,P,2,mayor,menor,medio
a,K,2,mayor,menor,medio
a,arcilla,2,mayor,menor,medio
a,limo,2,mayor,menor,medio
a,arena,2,mayor,menor,medio
c,sem_metro2,1,mucha,poca

Estructura de I-Avena.fit

a,biomasa_trigo 2001,2,alta,baja,media
a,biomasa_de_paja_del_trigo 2001,2,alta,baja,media
a,biomasa_de_espigas_del_trigo 2001,2,alta,baja,media
a,biomasa_de_grano_de_trigo 2001,2,alta,baja,media
a,biomasa_del_lolium 2001,2,alta,baja,media
a,elevacion_relativa,2,alta,baja,media
a,orientacion_cardinal,3,este,norte,oeste,sur
a,presencia_de_paja,2,no,si
a,pendiente_en_gr,3,clase 1,clase 2,clase 3,clase 4
a,radiacion_anual_acumulada,2,alta,baja,media
a,Ph,2,alto,bajo
a,Conductibilidad,2,alta,baja,media
a,N,2,normal,normal-alto
a,P,2,alto,muy alto
a,K,2,alto,bajo,medio
a,Humedad,2,alta,baja,media
a,MO,2,alta,media,media-alta
a,arena,2,alta,baja,media
a,limo,2,alta,baja,media
a,arcilla,2,alta,baja,media
a,tipo_de_suelo,3,franco,franco arcillo-limoso,franco arcilloso,franco limoso
c,biomasa_de_avena,1,existe,no existe

Estructura de IIa-Avena.fit

```
a,pendiente,2,clase 1,clase 2,clase 3,clase 4
a,radiacion_anual,2,menor,mayor
a,pH,2,menos basico,mas basico
a,N,2,normal,normal-alto,alto
a,P,2,medio,alto,muy alto
a,K,3,muy bajo,bajo,medio,alto,muy alto
a,MO,3,medio-baja,media,medio-alta,alta,excesiva
a,tipo_de_suelo,3,arcillo
limoso,arcilloso,franco,franco arcillo-limoso,franco arcilloso,franco
limoso
c,SEMILLAS,1,mas,menos
```

Estructura de IIb-lolium.fit

```
a,TRIGO,2,alto,bajo,medio
a,ESPIGAS,2,alto,bajo,medio
a,GRANO,2,alto,bajo,medio
a,AVENA,2,alto,bajo,medio
a,Cond,2,alto,bajo,medio
a,Hum,2,alto,bajo,medio
a,arena,2,alto,bajo,medio
a,limo,2,alto,bajo,medio
a,arcilla,2,alto,bajo,medio
a,categoria_NESW,3,este,norte,oeste,sur
a,PAJA,2,no,si
a,pendiente_,2,alto,bajo,medio
a,radiacion_,2,mayor,menor
a,pH_,2,mas basico,menos basico
a,N_,2,alto,normal,normal-alto
a,P_,2,alto,medio,muy alto
a,K_,3,alto,bajo,medio,muy alto
a,MO_,4,alta,excesiva,media,medio-alta,medio-baja
```

```
a,tipo_de_suelo_,5,arcillo limoso,arcilloso,franco,franco
arcillo-limoso,franco arcilloso,franco limoso
c,SEMILLAS_,1,mas,menos
```

Estructura de IIc-lolium.fit

Apéndice F

Resultados de Clementine (algoritmos C5.0 y CART)

RESULTADOS CON ALGORITMOS DE CLEMENTINE 6.0.2.

De datos de Avena en el campo de Madrid

ARBOL C5.0

```
limo mayor [Moda: poca] (107)
    arcilla mayor [Moda: poca] (26)
        pH menor [Moda: mucha] (10)
            P mayor [Moda: poca] (2, 1.0) -> poca
            P medio [Moda: mucha] (3, 1.0) -> mucha
            P menor [Moda: mucha] (5, 0.8) -> mucha
        pH mayor [Moda: poca] (16, 0.688) -> poca
    arcilla medio [Moda: poca] (35)
        N mayor [Moda: mucha] (5)
            MO mayor [Moda: mucha] (3, 1.0) -> mucha
            MO medio [Moda: poca] (2, 1.0) -> poca
            MO menor [Moda: mucha] (0.0) -> mucha
        N medio [Moda: poca] (8)
            pH menor [Moda: poca] (5, 1.0) -> poca
            pH mayor [Moda: mucha] (3, 0.667) -> mucha
        N menor [Moda: poca] (22, 1.0) -> poca
    arcilla menor [Moda: poca] (46, 0.87) -> poca
limo medio [Moda: mucha] (154)
    MO mayor [Moda: poca] (20)
        arcilla mayor [Moda: mucha] (10)
            P mayor [Moda: poca] (3, 1.0) -> poca
            P medio [Moda: mucha] (5, 1.0) -> mucha
            P menor [Moda: mucha] (2, 1.0) -> mucha
        arcilla medio [Moda: mucha] (2, 1.0) -> mucha
        arcilla menor [Moda: poca] (8, 1.0) -> poca
    MO medio [Moda: mucha] (86, 0.826) -> mucha
    MO menor [Moda: poca] (48)
        pH menor [Moda: mucha] (19)
            P mayor [Moda: poca] (3, 1.0) -> poca
            P medio [Moda: poca] (5, 0.6) -> poca
            P menor [Moda: mucha] (11, 0.727) -> mucha
        pH mayor [Moda: poca] (29)
            K mayor [Moda: poca] (2, 0.5) -> poca
            K medio [Moda: poca] (16, 1.0) -> poca
            K menor [Moda: poca] (11)
                arena mayor [Moda: mucha] (2, 1.0) -> mucha
                arena medio [Moda: poca] (4, 1.0) -> poca
                arena menor [Moda: mucha] (5)
                    P mayor [Moda: mucha] (0.0) -> mucha
                    P medio [Moda: mucha] (3, 1.0) -> mucha
                    P menor [Moda: poca] (2, 1.0) -> poca
limo menor [Moda: mucha] (112)
    arcilla mayor [Moda: mucha] (34)
        arena mayor [Moda: mucha] (26, 0.923) -> mucha
        arena medio [Moda: poca] (7)
            pH menor [Moda: mucha] (3, 1.0) -> mucha
            pH mayor [Moda: poca] (4, 1.0) -> poca
        arena menor [Moda: poca] (1, 1.0) -> poca
    arcilla medio [Moda: mucha] (42, 1.0) -> mucha
    arcilla menor [Moda: poca] (36)
        P mayor [Moda: poca] (8, 1.0) -> poca
        P medio [Moda: poca] (12)
            N mayor [Moda: poca] (2, 1.0) -> poca
            N medio [Moda: poca] (8, 0.875) -> poca
            N menor [Moda: mucha] (2, 1.0) -> mucha
        P menor [Moda: mucha] (16)
            MO mayor [Moda: poca] (2, 1.0) -> poca
            MO medio [Moda: mucha] (7, 0.857) -> mucha
            MO menor [Moda: mucha] (7, 1.0) -> mucha
```

REGLAS DESDE EL ARBOL C5.0

```
Reglas para mucha:
    Regla n°1 para mucha:
        si  limo == mayor
        y  arcilla == mayor
        y  pH == menor
        y  P == medio
        entonces -> mucha (3, 1.0)

    Regla n°2 para mucha:
        si  limo == mayor
        y  arcilla == mayor
        y  pH == menor
        y  P == menor
        entonces -> mucha (5, 0.8)

    Regla n°3 para mucha:
        si  limo == mayor
        y  arcilla == medio
        y  N == mayor
        y  MO == mayor
        entonces -> mucha (3, 1.0)

    Regla n°4 para mucha:
        si  limo == mayor
        y  arcilla == medio
        y  N == mayor
        y  MO == menor
        entonces -> mucha (0.0)

    Regla n°5 para mucha:
        si  limo == mayor
        y  arcilla == medio
        y  N == medio
        y  pH == mayor
        entonces -> mucha (3, 0.667)

    Regla n°6 para mucha:
        si  limo == medio
        y  MO == mayor
        y  arcilla == mayor
        y  P == medio
        entonces -> mucha (5, 1.0)

    Regla n°7 para mucha:
        si  limo == medio
        y  MO == mayor
        y  arcilla == mayor
        y  P == menor
        entonces -> mucha (2, 1.0)

    Regla n°8 para mucha:
        si  limo == medio
        y  MO == mayor
        y  arcilla == medio
        entonces -> mucha (2, 1.0)

    Regla n°9 para mucha:
        si  limo == medio
        y  MO == medio
        entonces -> mucha (86, 0.826)

    Regla n°10 para mucha:
        si  limo == medio
        y  MO == menor
        y  pH == menor
        y  P == menor
        entonces -> mucha (11, 0.727)
```

```
    Regla n°11 para mucha:
        si  limo == medio
        y  MO == menor
        y  pH == mayor
        y  K == menor
        y  arena == mayor
        entonces -> mucha (2, 1.0)

    Regla n°12 para mucha:
        si  limo == medio
        y  MO == menor
        y  pH == mayor
        y  K == menor
        y  arena == menor
        y  P == mayor
        entonces -> mucha (0.0)

    Regla n°13 para mucha:
        si  limo == medio
        y  MO == menor
        y  pH == mayor
        y  K == menor
        y  arena == menor
        y  P == medio
        entonces -> mucha (3, 1.0)

    Regla n°14 para mucha:
        si  limo == menor
        y  arcilla == mayor
        y  arena == mayor
        entonces -> mucha (26, 0.923)

    Regla n°15 para mucha:
        si  limo == menor
        y  arcilla == mayor
        y  arena == medio
        y  pH == menor
        entonces -> mucha (3, 1.0)

    Regla n°16 para mucha:
        si  limo == menor
        y  arcilla == medio
        entonces -> mucha (42, 1.0)

    Regla n°17 para mucha:
        si  limo == menor
        y  arcilla == menor
        y  P == medio
        y  N == menor
        entonces -> mucha (2, 1.0)

    Regla n°18 para mucha:
        si  limo == menor
        y  arcilla == menor
        y  P == menor
        y  MO == medio
        entonces -> mucha (7, 0.857)

    Regla n°19 para mucha:
        si  limo == menor
        y  arcilla == menor
        y  P == menor
        y  MO == menor
        entonces -> mucha (7, 1.0)

Reglas para poca:
    Regla n°1 para poca:
        si  limo == mayor
        y  arcilla == mayor
        y  pH == menor
        y  P == mayor
```

```
    entonces -> poca (2, 1.0)

Regla n°2 para poca:
    si  limo == mayor
    y  arcilla == mayor
    y  pH == mayor
    entonces -> poca (16, 0.688)

Regla n°3 para poca:
    si  limo == mayor
    y  arcilla == medio
    y  N == mayor
    y  MO == medio
    entonces -> poca (2, 1.0)

Regla n°4 para poca:
    si  limo == mayor
    y  arcilla == medio
    y  N == medio
    y  pH == menor
    entonces -> poca (5, 1.0)

Regla n°5 para poca:
    si  limo == mayor
    y  arcilla == medio
    y  N == menor
    entonces -> poca (22, 1.0)

Regla n°6 para poca:
    si  limo == mayor
    y  arcilla == menor
    entonces -> poca (46, 0.87)

Regla n°7 para poca:
    si  limo == medio
    y  MO == mayor
    y  arcilla == mayor
    y  P == mayor
    entonces -> poca (3, 1.0)

Regla n°8 para poca:
    si  limo == medio
    y  MO == mayor
    y  arcilla == menor
    entonces -> poca (8, 1.0)

Regla n°9 para poca:
    si  limo == medio
    y  MO == menor
    y  pH == menor
    y  P == mayor
    entonces -> poca (3, 1.0)

Regla n°10 para poca:
    si  limo == medio
    y  MO == menor
    y  pH == menor
    y  P == medio
    entonces -> poca (5, 0.6)

Regla n°11 para poca:
    si  limo == medio
    y  MO == menor
    y  pH == mayor
    y  K == mayor
    entonces -> poca (2, 0.5)

Regla n°12 para poca:
    si  limo == medio
    y  MO == menor
    y  pH == mayor
```

```
        y  K == medio
        entonces -> poca (16, 1.0)

    Regla n°13 para poca:
        si  limo == medio
        y  MO == menor
        y  pH == mayor
        y  K == menor
        y  arena == medio
        entonces -> poca (4, 1.0)

    Regla n°14 para poca:
        si  limo == medio
        y  MO == menor
        y  pH == mayor
        y  K == menor
        y  arena == menor
        y  P == menor
        entonces -> poca (2, 1.0)

    Regla n°15 para poca:
        si  limo == menor
        y  arcilla == mayor
        y  arena == medio
        y  pH == mayor
        entonces -> poca (4, 1.0)

    Regla n°16 para poca:
        si  limo == menor
        y  arcilla == mayor
        y  arena == menor
        entonces -> poca (1, 1.0)

    Regla n°17 para poca:
        si  limo == menor
        y  arcilla == menor
        y  P == mayor
        entonces -> poca (8, 1.0)

    Regla n°18 para poca:
        si  limo == menor
        y  arcilla == menor
        y  P == medio
        y  N == mayor
        entonces -> poca (2, 1.0)

    Regla n°19 para poca:
        si  limo == menor
        y  arcilla == menor
        y  P == medio
        y  N == medio
        entonces -> poca (8, 0.875)

    Regla n°20 para poca:
        si  limo == menor
        y  arcilla == menor
        y  P == menor
        y  MO == mayor
        entonces -> poca (2, 1.0)

Por defecto: -> mucha
```

REGLAS (MÉTODO DIRECTO CON C5.0) - MODO ESPECÍFICO

```
Reglas para poca:
    Regla n°1 para poca:
        si  pH == mayor
        y  MO == menor
        y  K == medio
        y  limo == medio
        entonces -> poca (16, 0.944)

    Regla n°2 para poca:
        si  MO == mayor
        y  P == mayor
        entonces -> poca (12, 0.929)

    Regla n°3 para poca:
        si  MO == mayor
        y  arcilla == menor
        entonces -> poca (20, 0.909)

    Regla n°4 para poca:
        si  N == mayor
        y  P == medio
        y  arcilla == menor
        entonces -> poca (9, 0.909)

    Regla n°5 para poca:
        si  P == mayor
        y  arcilla == menor
        y  limo == menor
        entonces -> poca (8, 0.9)

    Regla n°6 para poca:
        si  pH == mayor
        y  arcilla == mayor
        y  limo == menor
        y  arena == medio
        entonces -> poca (4, 0.833)

    Regla n°7 para poca:
        si  N == medio
        y  P == medio
        y  arcilla == menor
        y  limo == menor
        entonces -> poca (8, 0.8)

    Regla n°8 para poca:
        si  limo == mayor
        entonces -> poca (107, 0.78)

    Regla n°9 para poca:
        si  MO == menor
        entonces -> poca (104, 0.698)

    Regla n°10 para poca:
        si  arcilla == mayor
        y  limo == menor
        y  arena == menor
        entonces -> poca (1, 0.667)

Reglas para mucha:
    Regla n°1 para mucha:
        si  MO == mayor
        y  arcilla == medio
        entonces -> mucha (18, 0.9)

    Regla n°2 para mucha:
        si  pH == menor
        y  P == medio
        y  arcilla == mayor
        entonces -> mucha (17, 0.895)
```

```
    Regla n°3 para mucha:
        si  pH == mayor
        y  N == medio
        y  arcilla == medio
        entonces -> mucha (47, 0.878)

    Regla n°4 para mucha:
        si  P == medio
        y  K == menor
        y  limo == medio
        y  arena == menor
        entonces -> mucha (3, 0.8)

    Regla n°5 para mucha:
        si  limo == menor
        entonces -> mucha (112, 0.754)

    Regla n°6 para mucha:
        si  pH == mayor
        y  MO == menor
        y  limo == medio
        y  arena == mayor
        entonces -> mucha (2, 0.75)

    Regla n°7 para mucha:
        si  pH == menor
        y  MO == menor
        y  P == menor
        y  limo == medio
        entonces -> mucha (11, 0.692)

    Regla n°8 para mucha:
        si  limo == medio
        entonces -> mucha (154, 0.622)

Por defecto: -> mucha
```

REGLAS (MÉTODO DIRECTO CON C5.0) - MODO GENÉRICO

```
Reglas para poca:
    Regla n°1 para poca:
        si  MO == mayor
        y  arcilla == menor
        entonces -> poca (20, 0.909)

    Regla n°2 para poca:
        si  P == mayor
        y  arcilla == menor
        y  limo == menor
        entonces -> poca (8, 0.9)

    Regla n°3 para poca:
        si  limo == mayor
        entonces -> poca (107, 0.78)

    Regla n°4 para poca:
        si  N == menor
        y  MO == menor
        y  limo == medio
        entonces -> poca (38, 0.725)

Reglas para mucha:
    Regla n°1 para mucha:
        si  MO == mayor
        y  arcilla == medio
        entonces -> mucha (18, 0.9)

    Regla n°2 para mucha:
        si  limo == menor
        entonces -> mucha (112, 0.754)

    Regla n°3 para mucha:
        si  limo == medio
        entonces -> mucha (154, 0.622)

Por defecto: -> mucha
```

ALGORITMO CART

```
limo [menor medio] [Moda: mucha] (266)
    MO [medio] [Moda: mucha] (148)
        arena [mayor] [Moda: mucha] (41)
            arcilla [menor] [Moda: poca] (20)
                N [menor] (8, 0.75) -> mucha
                N [mayor medio] (12, 0.833) -> poca
            arcilla [mayor medio] [Moda: mucha] (21)
                K [menor] (7, 0.714) -> mucha
                K [medio mayor] (14, 1.0) -> mucha
        arena [menor medio] [Moda: mucha] (107)
            K [menor] (54, 0.963) -> mucha
            K [medio mayor] [Moda: mucha] (53)
                P [medio mayor] [Moda: mucha] (44)
                    P [mayor] (11, 0.727) -> mucha
                    P [medio] [Moda: mucha] (33)
                        pH [mayor] (22, 1.0) -> mucha
                        pH [menor] (11, 0.818) -> mucha
                P [menor] (9, 0.667) -> poca
    MO [mayor menor] [Moda: mucha] (118)
        P [medio mayor] [Moda: poca] (62)
            arcilla [menor] (27, 0.926) -> poca
            arcilla [mayor medio] [Moda: mucha] (35)
                MO [menor] [Moda: poca] (19)
                    K [menor mayor] (7, 0.571) -> mucha
                    K [medio] (12, 0.917) -> poca
                MO [mayor] (16, 0.813) -> mucha
        P [menor] [Moda: mucha] (56)
            pH [mayor] [Moda: mucha] (29)
                arena [mayor] (11, 0.909) -> mucha
                arena [menor medio] [Moda: poca] (18)
                    arcilla [medio] (7, 0.857) -> mucha
                    arcilla [mayor menor] (11, 1.0) -> poca
            pH [menor] [Moda: mucha] (27)
                limo [medio] (13, 0.769) -> mucha
                limo [menor] (14, 1.0) -> mucha
limo [mayor] [Moda: poca] (107)
    arcilla [menor medio] [Moda: poca] (81)
        MO [menor medio] [Moda: poca] (73)
            P [medio mayor] [Moda: poca] (51)
                K [menor] (4, 0.5) -> mucha
                K [medio mayor] [Moda: poca] (47)
                    MO [medio] [Moda: poca] (29)
                        N [menor medio] (24, 0.792) -> poca
                        N [mayor] (5, 1.0) -> poca
                    MO [menor] (18, 1.0) -> poca
            P [menor] (22, 1.0) -> poca
        MO [mayor] (8, 0.5) -> mucha
    arcilla [mayor] [Moda: poca] (26)
        P [mayor] (6, 1.0) -> poca
        P [menor medio] [Moda: mucha] (20)
            pH [mayor] [Moda: poca] (12)
                N [menor medio] (7, 0.571) -> mucha
                N [mayor] (5, 0.8) -> poca
            pH [menor] (8, 0.875) -> mucha
```

RESULTADOS CON ALGORITMOS DE CLEMENTINE 6.0.2.

De datos de Avena en el campo de Barcelona

REGLAS (MÉTODO DIRECTO CON C5.0) – MODO ESPECÍFICO

```
Reglas para no existe:
    Regla nº1 para no existe:
        si  radiacion == media
        y  K == alto
        entonces -> no existe (5, 0.857)

    Regla nº2 para no existe:
        si  biomasa_del_lolium_2001 == baja
        y  P == muy alto
        y  K == bajo
        entonces -> no existe (3, 0.8)

    Regla nº3 para no existe:
        si  K == medio
        entonces -> no existe (179, 0.669)

Reglas para existe:
    Regla nº1 para existe:
        si  arena == media
        y  elevacion == media
        y  P == alto
        y  K == medio
        entonces -> existe (9, 0.909)

    Regla nº2 para existe:
        si  pendiente == media
        y  radiacion == baja
        y  K == medio
        entonces -> existe (13, 0.867)

    Regla nº3 para existe:
        si  arena == baja
        y  limo == media
        y  radiacion == alta
        entonces -> existe (5, 0.857)

    Regla nº4 para existe:
        si  P == alto
        y  K == bajo
        entonces -> existe (50, 0.788)

    Regla nº5 para existe:
        si  biomasa_del_lolium_2001 == media
        y  P == muy alto
        y  K == bajo
        entonces -> existe (2, 0.75)

    Regla nº6 para existe:
        si  radiacion == alta
        y  K == alto
        entonces -> existe (3, 0.6)

Por defecto: -> no existe
```

REGLAS (MÉTODO DIRECTO CON C5.0) – MODO GENÉRICO

```
Reglas para no existe:
    Regla n°1 para no existe:
        si  K == alto
        entonces -> no existe (8, 0.7)

    Regla n°2 para no existe:
        si  K == medio
        entonces -> no existe (179, 0.669)

Reglas para existe:
    Regla n°1 para existe:
        si  pendiente == media
        y  radiacion == baja
        y  K == medio
        entonces -> existe (13, 0.867)

    Regla n°2 para existe:
        si  limo == media
        y  elevacion == media
        y  radiacion == alta
        entonces -> existe (11, 0.769)

    Regla n°3 para existe:
        si  K == bajo
        entonces -> existe (55, 0.754)

Por defecto: -> no existe
```

Apéndice G

RESULTADOS DE AGLEARNER + SQLDECO

```
Número de generaciones: --
Tamaño de Población: --
Ini. Random-Pop: Verdadero
Probabilidad de Cruce: 0.5
Probabilidad de Mutación:---
Tipo de Elitismo: Elitismo Unitario
Tipo de Cruce: 1 Punto
Tipo de Mutación: Todos
Tipo de Selección: Ruleta
-----------------------------------
Máximo Global: 0.598]
[DD/MM/AA  hh:mm:ss]:   I) 29/05/2003 15:21:46  F)29/05/2003  16:12:21
Máximo: [0.546]
01010010100100100011000011000O


Fitness  :0.598


IF ((

r1: MO <> 'menor' AND P <> 'mayor' AND arcilla <> 'menor'

OR

r2: limo = 'medio' AND arena = 'menor'

)) THEN Sem_metro2 => mucha

TotalCasosEval  : 373
/Vp:    155
/Fn:     49
/Vn:    133
/Fp:     36



RESULTADOS GENERALES:
Vp:     155
Fn:     49
Vn:     133
Fp:     36
total:     373

Parámetros de calidad:

Fitness      :        0.598
Exactitud    :        0.772
Confianza    :        0.8115
Cobertura (+):        0.760
Cobertura (-):        0.787

---r1: (MO <> 'menor' AND P <> 'mayor' AND arcilla <> 'menor' )----------
Vp:     133
Fn:     71
Vn:     143
Fp:     26
total:     373

Parámetros de calidad:

Fitness      :        0.552
Exactitud    :        0.740
Confianza    :        0.8365
Cobertura (+):        0.652
Cobertura (-):        0.846
```

```
---r2: ( limo = 'medio' AND arena = 'menor')----------
Vp:      22
Fn:      182
Vn:      155
Fp:      14
total:      373

Parámetros de calidad:

Fitness       :         0.099
Exactitud     :         0.475
Confianza     :         0.6111
Cobertura (+):          0.108
Cobertura (-):          0.917
```

Número de generaciones: 102
Tamaño de Población: 75
Ini. Random-Pop: Verdadero
Probabilidad de Cruce: 0.5
Probabilidad de Mutación:0.005
Tipo de Elitismo: Elitismo Unitario
Tipo de Cruce: 1 Punto
Tipo de Mutación: Todos
Tipo de Selección: Ruleta

Máximo Global: 0.546]
[DD/MM/AA hh:mm:ss]: I) 29/05/2003 15:21:46 F)29/05/2003 16:12:21
Máximo: [0.546]
01010010100100100011000011000

Fitness : 0.546

IF ((

r1: MO <> 'menor' AND limo <> 'mayor'

)) THEN Sem_metro2 => mucha

TotalCasosEval : 373
/Vp:153
/Fn:51
/Vn:123
/Fp:46

RESULTADOS GENERALES:

Vp:	153
Fn:	51
Vn:	123
Fp:	46
total:	373

Parámetros de calidad:

Fitness :	0.546
Exactitud :	0.740
Confianza :	0.769
Cobertura (+):	0.750
Cobertura (-):	0.728

---r1: (MO <> 'menor' AND limo <> 'mayor')----------

Vp:	153
Fn:	51
Vn:	123
Fp:	46
total:	373

Parámetros de calidad:

Fitness :	0.546
Exactitud :	0.740
Confianza :	0.769
Cobertura (+):	0.750
Cobertura (-):	0.728

```
Número de generaciones: 119
Tamaño de Población: 100
Ini. Random-Pop: Falso; (Indv.Ini)--> 11000000000000000000000000000000
Probabilidad de Cruce: 0.5
Probabilidad de Mutación:0.009
Tipo de Elitismo: Elitismo Unitario
Tipo de Cruce: 1 Punto
Tipo de Mutación: Todos
Tipo de Selección: Ruleta
-----------------------------------
Máximo Global: 0.39]00000011010000000001000010010000
[DD/MM/AA  hh:mm:ss]:   I) 19/05/2004 11:54:26  F)19/05/2004  13:13:32
Máximo: [0.39]00000011010100110000000100001001


01110011010000000001000000010000
 Fitness  : 0.39

IF ((

r1:     MO = 'medio'

)) THEN Sem_metro2 => mucha
 TotalCasosEval  : 373
/Vp:133
/Fn:71
/Vn:101
/Fp:68

RESULTADOS GENERALES:
Vp:     133
Fn:     71
Vn:     101
Fp:     68
total:     373
Parámetros de calidad
Fitness    :            0.390
Exactitud:              0.627
confianza :             0.662
Cobertura (+):          0.652
Cobertura (-):          0.598


---r1: (MO = 'medio')----------
Vp:     133
Fn:     71
Vn:     101
Fp:     68
total:     373
Parámetros de calidad
Fitness    :            0.390
Exactitud:              0.627
confianza :             0.662
Cobertura (+):          0.652
Cobertura (-):          0.598
```

UNIVERSIDAD DE ALCALÁ
SERVICIO DE POSTGRADO

DILIGENCIA PARA HACER CONSTAR QUE EL PRESENTE EJEMPLAR DE LA TESIS PRESENTADA POR D.ª Beatriz Díaz Gómez
CONSTA DE 384 PAGINAS Y HA SIDO ENTREGADA CON FECHA 28 de julio de 2005
A EFECTOS DE DEPOSITO DE TESIS.

EL FUNCIONARIO.

Reunido el Tribunal que suscribe en el día de la fecha acordó otorgar a la presente Tesis Doctoral la calificación de Sobresaliente cum laude

Alcalá de Henares, 16 de Diciembre de 2005

EL PRESIDENTE

Fdo.:

EL SECRETARIO

Fdo.: Sra. A. Malpica

EL VOCAL

Fdo.: Mª CARMEN GARCÍA NIEVES SÁNCHEZ

EL VOCAL

Fdo.: AURORA PÉREZ

EL VOCAL

Fdo.: F. Javier Sans