IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 1, 2019, accepted July 23, 2019, date of publication July 30, 2019, date of current version September 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2932055

Analysis of Artificial Neural Network Architectures for Modeling Smart Lighting Systems for Energy Savings

ALBERTO GARCES-JIMENEZ[1], JOSÉ LUIS CASTILLO-SEQUERA[2], ANTONIO DEL CORTE-VALIENTE[3], JOSÉ MANUEL GÓMEZ-PULIDO[2], AND ESTEBAN PATRICIO DOMÍNGUEZ GONZÁLEZ-SECO[4]

[1]Center for Research and Innovation in Knowledge Management, University of Francisco de Vitoria, 28223 Pozuelo de Alarcón, Spain
[2]Department of Computer Sciences, Polytechnic School, University of Alcala, 28871 Alcala de Henares, Spain
[3]Department of Computer Engineering, Polytechnic School, University of Alcala, 28871 Alcala de Henares, Spain
[4]Department of Construction, School of Architecture, Polytechnic University of Madrid, 28040 Madrid, Spain

Corresponding author: Antonio del Corte-Valiente (antonio.delcorte@uah.es)

ABSTRACT Currently, population growth is global and tends to concentrate in large cities, which increases the demand for illuminating public spaces for safety, visual orientation, aesthetic considerations, and quality of life. The undesirable side effects are increase in energy consumption and light pollution. The current tools used for designing public lighting systems are not suitable for optimizing multiple objectives in addition to energy savings, and these solutions could provide for a more sustainable environment. The application of evolutionary optimization techniques seems to be growing rapidly because of the nonlinearity of the model behavior and the nonproprietary nature of the algorithms, which are considered as *black box systems*. This paper develops a data model for these types of optimizers, analyzing the ability of different artificial neural network (ANN) architectures to simulate a simple public lighting design by measuring the performance with respect to the fitness function, training speed, and goodness of fit with a dataset generated with different conditions. The architectures selected in this paper are those with multilayer perceptrons (MLPs) with different hidden layer configurations using different numbers of neurons in each layer, which have been analyzed to determine the configuration that best fits the purpose of this work. The data for training the ANNs were generated with a recognized open-software platform, DIALux. The experiments were repeated and analyzed to determine the variance of the results obtained. In this way, it was possible to identify the most appropriate number of iterations required. The results show that better precision is obtained when using the Levenberg–Marquardt training algorithm, especially when the ANN architecture has fewer neurons in the hidden layer.

INDEX TERMS Public lighting design, artificial neural networks, multilayer perceptron, data modeling, energy efficiency, uniformity ratio of luminance, sustainable cities.

I. INTRODUCTION

The urban population maintains a growth rate that evolves and is expected to reach two-thirds of the overall population by 2050 [1]. This requires wider illumination of public areas, and the undesired effects are an increase in energy consumption and light pollution. Greenhouse gas emissions and global warming, partially caused by the production of energy, put pressure on the private and public sectors to find more sustainable solutions. Public lighting consumes 19% of the global electricity production [2]. Security and aesthetic concerns [3] are unavoidable constraints for saving energy, leading researchers to consider multiple objectives in the optimization approaches. Design optimization is expected to achieve significant energy savings, approximately 35% with adaptive methods [4] or 45% by using optimal elements [5].

The associate editor coordinating the review of this manuscript and approving it for publication was Akhilesh Thyagaturu.

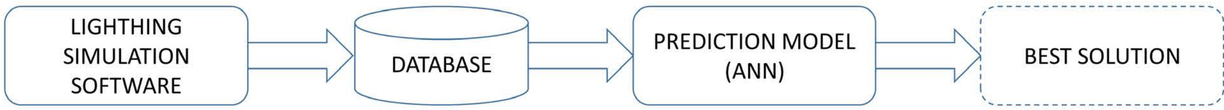

FIGURE 1. Process to build the best data-driven model that will simulate the behavior of a lighting system. The selected architecture will support an evolutionary optimizer with energy consumption as the cost function.

The European Union, under the European Standardization Framework, is regulating both the functional requirements and indicators of energy performance within Standard EN13201.

This article follows the Guidelines from Part 2 of the Standard –EN13201-2:2016, regarding the selection of more appropriate lighting types, according to given situations and their required performance. It also follows Part 3 –EN13201-3:2016-, using the accepted mathematical procedures and conventions, such as the photometric performance in road lighting with its specific parameters. Moreover, it uses Part 5 –EN13201-5:2016- for the performance indicators for compliance. It is necessary to note that the final decision to illuminate a road is left to each country and its cities, according to the standard from the International Commission on Illumination governing the lighting of roads for motor and pedestrian traffic, CIE115:2010, which is taken as a reference.

These guidelines aim to maximize the visualization, orientation and security levels for pedestrians and vehicles [6]–[9]. An appropriate illumination of streets, roads and parks helps to reduce the crime rates and vandalism that transform cities into unsafe places to live [10], [11]. The lamp types, pole features, street dimensions and surrounding requirements are the inputs for designing any lighting project. The design tools compute them and indicate which results are compliant, such as the minimum overall uniformity ratio of luminance, the road surface illuminance in dry conditions, disturbing brightness (discomfort glare) or the surroundings conditions [12]–[16].

However, most common design tools lack the option for optimizing the energy consumption together with other cost objectives. The AGi32 [17], DIALux [18], DL-Light [19], FocusTrack [20], TracePro [21], LD Assistant [22], Vectorworks Spotlight [23], systems compute the energy usage according to the normative standard but do not suggest alternative designs for saving energy.

The fundamental contribution of this article is to determine the most efficient multilayer ANN architecture with conventional training functions and demonstrate its capabilities in a set of simplified public lighting scenarios [24]–[28]. The justification of this research about ANNs over other algorithms is the potential brought for engineers who are designing new public light installations or re-designing existing ones in the field, being able to optimize multiple objectives others than the standard ones, such as energy, cost, maintenance, aesthetics or durability. The machine learning capability in the ANN provides, not only those discrete values set by the Normative, but also continuous values obtained from standard features and from new unforeseen variables, like observed nearby reflective materials, unavoidable shapes, unexpected shadows, nonstandard climate conditions, singular spacing or color combinations. Other simpler algorithms, like white box or formulae-based ones, possibly give more accurate results, but are rigid and cannot adapt to these new scenarios because required to be previously obtained. In addition, the Mean Squared Error (MSE) analysis, the number of epochs and the time consumed obtained from the tried algorithms show that the learning process speed is affordable. Computational speed issues will be anyhow matter of future discussions.

The simulator generates two separated datasets with the lighting simulation software: one for training, the Training Dataset (TDS), and another for proofing the obtained models, the Proof Dataset (PDS), prepared with different lamps and design parameters than those chosen for the TDS. The neural learning starts randomly dividing the TDS into 3 sets for training, testing and validating. PDS does not train any ANN, but proves the models obtained with the TDS. The best performance in terms of MSE and coefficient of determination (R^2) allows the comparison of the ANN architectures. Other parameters have been recorded to observe the learning process, but they are not required in the comparison. The PDS is then tried in the models and its performance is expected to confirm the comparison (Fig. 1).

The rest of the paper is organized as follows: In Section 2, the concept of an ANN-based model, the ANN architectures and the training algorithms are presented. Section 3 describes the experimental results and introduces the discussions. Then, Section 4 presents the conclusions and suggests future work.

II. MATERIALS AND METHODS

This research selects the best ANN architecture to produce fast lighting simulations [29] that will provide the data for an optimizer with the objective function of minimizing the energy consumption while maintaining the uniformity ratio of luminance (U_0) within the normative standard [30]. The first step is to obtain reliable data for training and testing the potential models. The data are generated with DIALux Public Lighting Design Tool [31], [32] and configured with two predefined solution spaces, giving two datasets, TDS and PDS. Then, ten multilayer feed forward ANNs are built and trained with the TDS. The ANN architectures are then tested with the second PDS.

The process is repeated several times until the selected performance parameters suffer less variance. The maximum number of iterations is predetermined with a simple

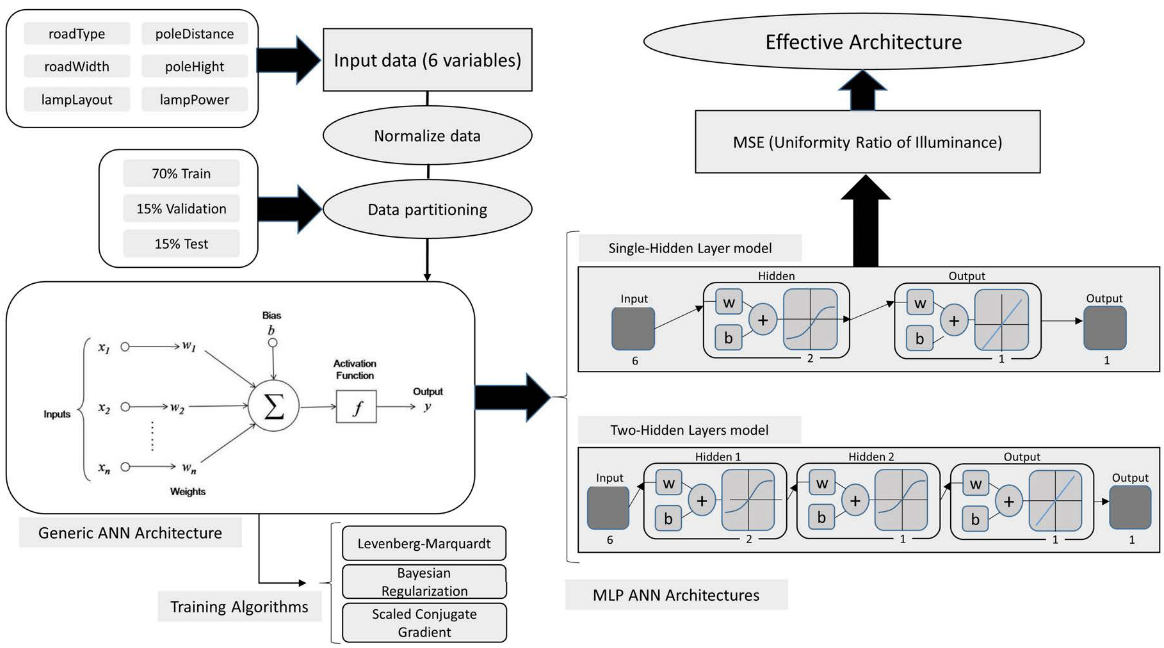

FIGURE 2. The ANN analysis models the dataset by training the ten ANN architectures with the three back-propagation algorithms.

graphical approach. The best architecture is the fastest and most accurate lighting simulator.

A. PUBLIC LIGHT DESIGN

The design of public road lighting follows the above mentioned normative approach. The key objective is to appropriately light roads for car traffic security, visibility and orientation. The goals of road lighting design are as follows:

1) obtain a sufficient average luminance (L_m) [33],
2) minimize the uniformity ratios of luminance (U_0 and U_L),
3) limit glare to avoid blinding,
4) consider surrounding lighting (edge factor), and
5) ensure optical orientation.

To simplify the analysis with the aim to include the remaining parameters in subsequent research, several simplifications have been applied to the road conditions. The datasets are generated for the ME1 and ME2 road lighting recommendations, since the generalization to other types is simple. The difference between ME1 and ME2 is that the average luminance is 2cd/m^2 for the former and 1.5cd/m^2 for the latter. The threshold increment of 10% and surrounding rate of 0.50 are the same for both. The model only considers the overall uniformity ratio of luminance [34], being extendable to the global uniformity or surround ratio afterwards.

The model requires the selection of lamps and ballasts, but the large number of brands, models and types makes its implementation time consuming. The common types of lamps are made of sodium-vapor, mercury-vapor, metal-halide or LED [9]. This research only considers LED-type lamps because they are becoming the predominant type used among designers due to their low energy consumption compared with other types of lamps.

B. MULTILAYER PERCEPTRON ANN

The ANNs are capable of incorporating nonlinear effects and interactions among the variables of the data model as a black box. The intrinsic machine learning ability automatically extracts the hidden patterns from the data and detects trends, offering an alternative way to evaluate complex relationships. However, the ANN parameters are difficult to interpret and explain due to the wide empirical process of construction and training. For this reason, our research shows the best ANN model, taking as a reference the comparison of the results obtained (Fig. 2).

Data-driven optimization methods work with data generated by system simulators, which holds true for energy consumption simulation.

For some problems, the ANN is basically designed by trial and error, selecting the best configuration by analyzing the results. This is the case in this research, which obtains the best MLP architecture by varying the number of hidden layers and neurons and testing various back-propagation algorithms [35].

The number of hidden layers and neurons in each layer affect the capacity of the model for generalization, i.e., the accuracy in computing new examples. Some authors demonstrate that the number of hidden layers is normally between the size of the input layer and the output layer, or slightly higher [36]. The generally accepted convention is the *universal approximation theorem* [37], which simplifies the problem by stating that two-hidden layers or in most cases, one-hidden layer, are sufficient to achieve the best results. Lippmann adds that single hidden layer ANNs are able to solve arbitrarily complex problems, given that the hidden layer includes at least three times the number of input nodes [38].

Hecht-Nielsen extends the Kolmogorov theorem [39] to demonstrate that single hidden layer ANNs with 2N+1 neurons and continuous, nonlinear monotonically increasing transfer functions are sufficient to compute any continuous function of N input variables [40]. However, the estimation of the number of hidden neurons in each layer is done with empirical rules and is therefore difficult to justify. The *rule of the geometric pyramid* assumes that the number of neurons in the hidden layer must be less than the total number of input variables but higher than the number of output variables [41]. In addition, the number of neurons in each layer follows a geometrically decreasing progression from the input to the output. The number of intermediate neurons must be close to $\sqrt{M.N}$ where N is the number of input variables and M is the number of output neurons. On the other hand, according to the *rule of the hidden layer*, the number of hidden neurons is proportional to the number of input neurons [42]. Typically, the number of hidden neurons should not be more than twice the number of input variables.

C. BACKPROPAGATION TRAINING

The supervised training of an MLP normally obtains its configuration by comparing the outputs after using the dataset to model the expected outputs in an iterated trial-and-error process, seeking to minimize the error with the backpropagation algorithms that conveniently tune the weights and biases. These algorithms require that the MLP structure or topology, i.e., the MLP architecture, and the number of neurons and hidden layers, are set before the training starts, making the selection of the architecture a guessing procedure. The number of forward and backward operations is large, requiring nonnegligible computational resources, although once the MLP is trained and the system is modeled, the computation is fast.

Any training algorithm modifies the weights according to the following expression:

$$W_{ji}(n+1) = W_{ji}(n) + \Delta W_{ji}(n) \tag{1}$$

where W_{ji} is obtained via the training algorithm rule and n is n^{th} iteration. If $E(n)$ is the output error after the *nth* training iteration, there are two possible ways to test the performance:

$$E(n) = \frac{1}{2N} \sum\nolimits_z \sum\nolimits_j (e_j^z(n))^2 \tag{2}$$

$$E(n) = \frac{1}{2} \sum_{j=1}^{M} e_j(n)^2 \tag{3}$$

where N is the total number of input/output pairs, z is the *zth* input/output pair of the TDS, j is the *jth* output layer neuron, and M is number of output neurons.

The first expression defines the ANN global error, while the second is the immediate MSE that approximates the former with less computational effort. In both cases, e_j is the error between the prediction and real value.

MLPs can be compared to logistic regression classifiers where the inputs are first transformed with a nonlinear transformation, θ, in the hidden layer that makes the data linearly separable:

$$h(x) = \theta(x) = S(b_{(1)} + W_{(1)} \tag{4}$$

Vector $h(x)$ is the hidden layer. The sub index in parenthesis represents the particular layer. The machine learning process seeks to minimize the error function and adjusts the weight matrix. Specifically, for a one-hidden layer, the function F(x) is:

$$F(x) = G(b_{(2)} + W_{(2)} S\left(b_{(1)} + W_{(1)}\right)) \tag{5}$$

where $b_{(1)}$ and $b_{(2)}$ are the bias vectors; $W_{(1)}$, $W_{(2)}$ are the weight matrices, and G and S are the activation functions. The $W_{(1)}$ columns are the weights from the input units to the i^{th}-hidden layer unit.

For computation, we use the multipurpose numeral-computing tool MATLAB, developed by Mathworks, which has three default choices for the backpropagation algorithm, represented by S(we have taken the default Matlab values):

1) Levenberg-Marquardt [43]–[46]: This algorithm iteratively locates the minimum of a multivariate function, expressed as the sum of squares of nonlinear real-valued functions. MATLAB recommends it for most problems as it trains the MLP faster but requires more memory.
2) Bayesian Regularization: This algorithm updates weights and biases in the same way as the Levenberg-Marquardt algorithm but minimizes a combination of squared errors and weights. The resulting model also generalizes well and obtains better solutions for many practical problems but is slower.
3) Scaled Conjugate Gradient: This is the simplest algorithm and is used for stable training. It updates the weights and biases towards the negative gradient of the performance function [47]. It is recommended for large problems as it works with first-order gradients and not with the second-order Jacobian, being more memory efficient. The weights are initialized with small values around the origin so that the activation function can operate in its linear zone, where the gradients are larger.

The Levenberg-Marquardt training algorithm computes the expression defined by:

$$(J^t * J + \lambda * I) * \delta = J^t * E \tag{6}$$

where J is the Jacobian matrix for the system, λ is the damping factor, I is the identity matrix, δ is the weight update vector that we want to obtain, and E is the error vector containing the output errors for each input vector used in training the network. Figure 3 shows a diagram of the Levenberg-Marquardt

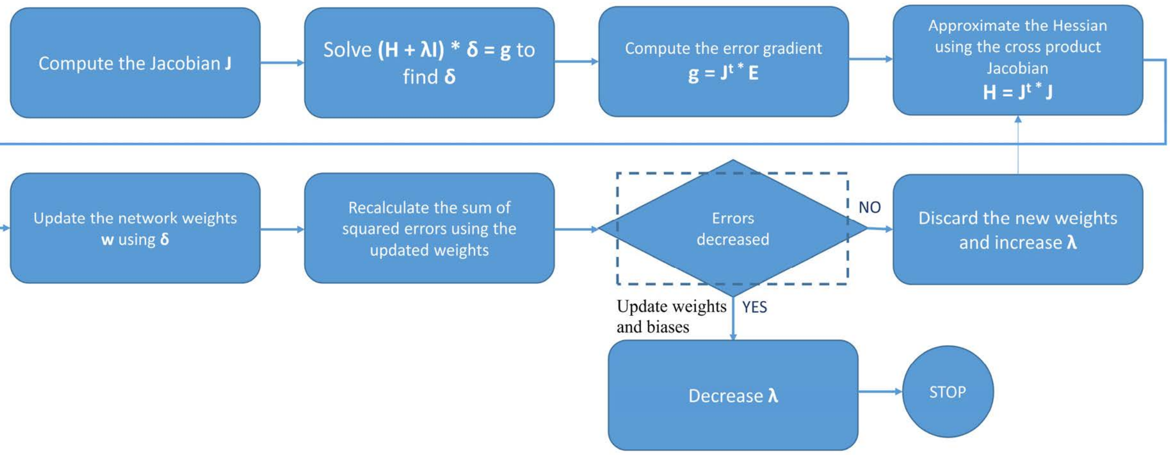

FIGURE 3. Flow diagram of the Levenberg-Marquardt training algorithm.

training algorithm. Only the method of updating the weights and biases differs in each algorithm.

The MLP architecture has some drawbacks [48], [49]. The main limitation is that it cannot guarantee a global minimum during training and the network may converge to a local minimum. Training the network several times by using a different random starting positions each time and then obtaining the model that results in the best RMS error will mitigate this issue. In any case, finding the global minima in deep networks appears to be unnecessary because local minima are approximately as good as global minima [50].

Another limitation is that the number of hidden neurons must be set manually. Setting this value too low may result in underfitting, while setting this value too high may result in overfitting. By training a regular multilayer neural network in classification tasks using a training dataset and starting from randomly initialized weights, the stochastic gradient descent algorithm can attain 100% accuracy [51].

D. TRANSFER FUNCTION

The MLP architecture also defines the transfer/output functions for the network topology. To ensure that the learning process obtains a result, these functions must be continuous and differentiable at all points. The most common functions are the sigmoid function for the hidden layers and the linear function for the output layer. The sigmoid function is defined as

$$Sc\left(x\right) = \frac{1}{1 + e^{-cx}} \tag{7}$$

where c is the steepness of the curve.

E. PROPOSED MULTILAYER FEED FORWARD ARCHITECTURES

Designing an ANN is mostly an empirical process that balances the accuracy and the ANN generalization capability. Adding more hidden layers to the architecture normally worsens its performance and generalization capability.

The *rule of the single layer* [52] suggests building the first hidden layer of this experiment with up to six neurons, one for each input variable from the input layer. The experiment will also try one and two-hidden layer architectures, anticipating possible discontinuities in the data behavior. For the studied models with two-hidden layers, the *rule of the geometric pyramid* is applied [53]. The rule says that the size of every hidden layer decreases in geometric order related to the previous layer from the input to the output.

The activation function for hidden layers is the sigmoid and the linear function is for the output layer. Following these considerations, the experiment builds the following ten architectures (Fig. 4):

1) One-hidden layer with two neurons: "2"
2) One-hidden layer with three neurons: "3"
3) One-hidden layer with four neurons: "4"
4) One-hidden layer with five neurons: "5"
5) One-hidden layer with six neurons: "6"
6) Two-hidden layers with two and one neurons: "2-1"
7) Two -hidden layers with three and one neurons: "3-1"
8) Two -hidden layers with four and two neurons: "4-2"
9) Two -hidden layers with six and two neurons: "6-2"
10) Two-hidden layers with six and three neurons: "6-3"

Only the number of neurons, number of layers, and learning algorithms are changed. The *learning rate* and other parameters remain unchanged for all the architectures. Once selected the ANN topologies and the appropriate activation functions, the model can be trained with the training algorithms to obtain a simulation of the results. Each of these analyzed architectures represents different configurations, with one or two hidden layers, and with different numbers of hidden neurons to determine the configuration that best suits the purpose of the work.

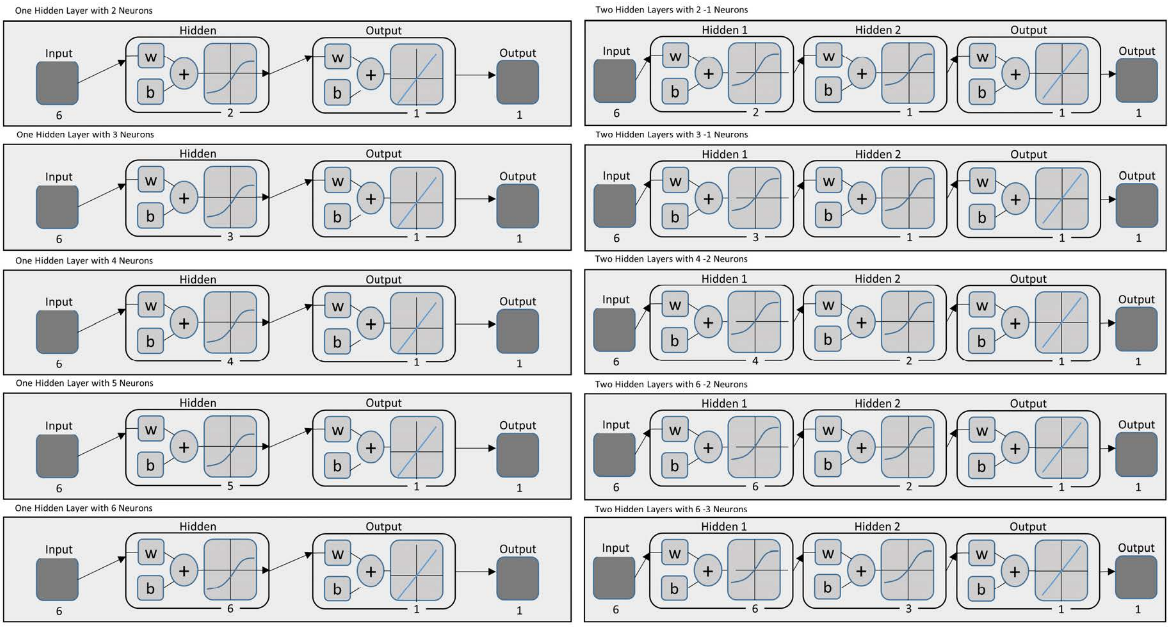

FIGURE 4. **Proposed MLP architectures for the analysis: (left) One HL; (right) Two HLs.**

F. MEASURED PERFORMANCE STABILITY

The performance measured at the end of the process is variable, as the standard process takes three random datasets for modeling: training, testing and validating. In this case, 70% of the TDS is used to train and 15% each to test and validate. The results are biased and prevent a reliable comparison. The algorithm takes examples randomly.

The solution proposed in this research is the repetition of the experiment, assuming some natural trend that is verifiable at the end of the repetitions, until the performance values become stable. The results of every repetition are grouped into 15 sets containing 5 to 75 elements in increments of 5. These groups are then parsed through a boxplot graphical analysis to identify the group with fewer repetitions that first reaches the stable parameters for every performance indicator. The minimum number of tries allows for the reliable performance measurement of the ten architectures for comparison.

G. TRAINING DATASET (TDS)

The models corresponding to each MLP architecture are trained with the TDS generated with the DIALux software tool and are made up of 648 different design conditions given by these variables (Table 1):

1) Road lighting classes, according to EN13201: ME1 or ME2 restrictions for heavy traffic.
2) Road width: 7 m, 9 m or 12.5 m.
3) Lamp layout: only one-side of the road, two-sides of the road or alternating-one-right-next-left.
4) Separation between poles: 10 m, 25 m or 50 m
5) Pole height (3): 4 m, 8 m or 12 m

TABLE 1. **Training data set.**

Variables	Values	Description
RoadType	ME1, ME2	Road lighting classes
RoadWidth	7, 9, 12.5	Width of the road [m]
LampLayout	One side, two sides, alternating	Position of lamps in the road
PoleDistance	10, 25, 50	Separation between poles [m]
PoleHeight	4, 8, 12	Height of the pole [m]
LampPower	30, 91, 174, 276	Power of the LED lamp [W]

6) LED lamp power (4): 30 W, 91 W, 174 W or 276 W

The software tool simulates these configurations and yields the average luminance (L_m), global and longitudinal uniformity (U_0 and U_L), threshold increment (TI) and surround ratio (SR). For this research, the single output is U_0, the dependent variable of the fitness function, for simplicity purposes [54], leaving the other variables for coming research.

To obtain a more representative dataset, we used the actual nominal value of the lamp power, which was checked with the manufacturer.

H. PROOF DATASET (PDS)

To test that the selected model better approaches any other design condition, the DIALux software generates a second dataset, the PDS, with 64 examples interpolating and extrapolating values different to those of the TDS (Table 2):

1) Road lighting requirements according to EN13201: ME1 or ME2
2) Road width: 8 m or 15 m

TABLE 2. Proof data set.

Variable	Values	Description
RoadType	ME1, ME2	Road lighting requirements
RoadWidth	8, 15	Width of the road [m]
LampLayout	One side, two sides	Position of lamps in the road
PoleDistance	15, 35	Separation between poles [m]
Pole height	10, 15	Height of the pole [m]
LampPower	105, 182	Power of the LED lamp [W]

3) Lamp layout: only one-side of the road or two-sides of the road
4) Separation between poles: 15 m or 35 m
5) Pole height: 10 m or 15 m
6) LED Lamps Power: 105 W or 182 W

III. EXPERIMENTAL RESULTS AND DISCUSSION

The ANN training process randomly splits the TDS into three subgroups: training (70%), validation (15%) and testing (15%). The training subset of the TDS is used to adjust the weights and biases of the network. The validation subset is used to measure the network generalization. The test subset analyzes the network performance after the training. At the end of this stage, the trained topologies are trialed with different design conditions from the PDS to measure their accuracy.

The discussion addresses the MSE performance with the TDS, total time consumed, required number of epochs and MSE performance with the PDS. The process is repeated to obtain reliable values.

Taking the 1-sample mean or median of the TDS or PDS MSE, it is observable that the experiment requires more than one repetition to obtain stable values; otherwise, it would be impossible to evaluate the behaviors.

To quantify the required number of samples, we have assumed as a first approach the study of [55]. We assume that the population is large as it is derived from combining 648 examples using sets of 75, which are the times to train every architecture with every single learning algorithm.

We use the formula of the mentioned study to estimate the number of samples necessary to obtain enough confidence to ensure the MSE achieved by the TDS and PDS:

$$n_o = \frac{Z^2 * s^2}{e^2} \quad (8)$$

where n_o is the sample size, Z is the abscissa of the normal curve that cuts off an area of α at the tails (1 - α equals the desired confidence level), e is the desired level of precision (in the same unit of measure as the variance) and s^2 is the variance of an attribute in the population.

Assuming a confidence level of 99%, an acceptable error of the MSE value of approximately 6% and the obtained standard deviation with the 75 repetitions, we observe that it is necessary to repeat the experiment at least 39 times for the TDS. In addition, we can visualize these figures in boxplots to depict the variations of the average performance (MSE) for the TDS (Fig. 5) and PDS (Fig. 6) with the sample size used for training.

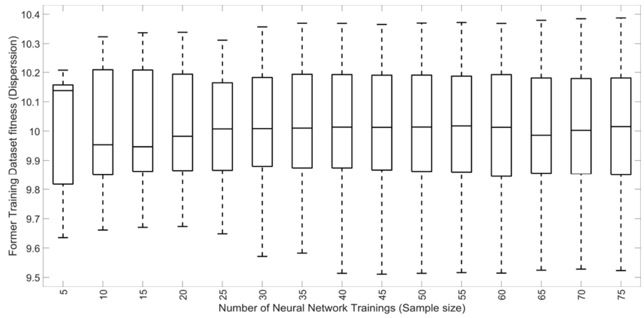

FIGURE 5. Variation of the average performance (MSE) for the TDS with the sample size used for training.

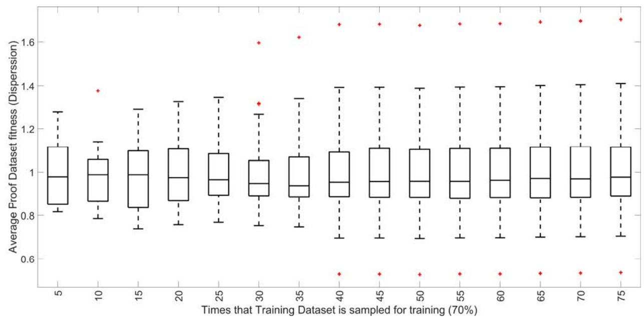

FIGURE 6. Variation of the average performance (MSE) of the PDS with the sample size used for training.

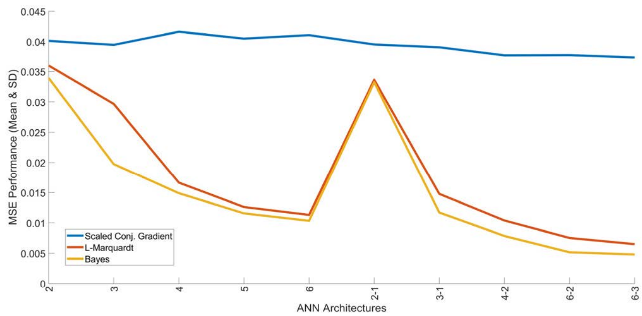

FIGURE 7. Performance (MSE) versus ANN architectures for different training algorithms.

Figure 7 shows the ANN topology performance, with the median MSE and Standard Deviation (SD) for each of the ten topologies applying the three proposed back propagation algorithms: Levenberg-Marquardt, Bayesian regulation and scaled conjugate gradient.

Figure 8 shows the MSE performance of the PDS for each MLP architecture.

Figures 9 and 10 depict similar results for the number of epochs and the time consumed by each algorithm.

The Levenberg-Marquardt achieves results very similar to the best value but at the expense of a shorter time, so comparing it with the other algorithms, it is the chosen algorithm in this experiment. However, the Bayesian algorithm shows the worst characteristics due to the additional calculation needed to refine the optimization. The number of neurons and hidden layers increases the computational resources required.

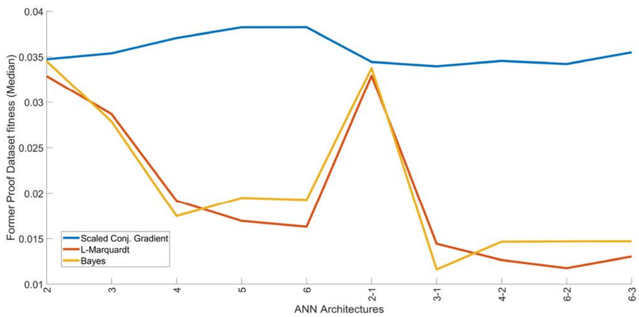

FIGURE 8. Architecture fitness for each architecture and training algorithm.

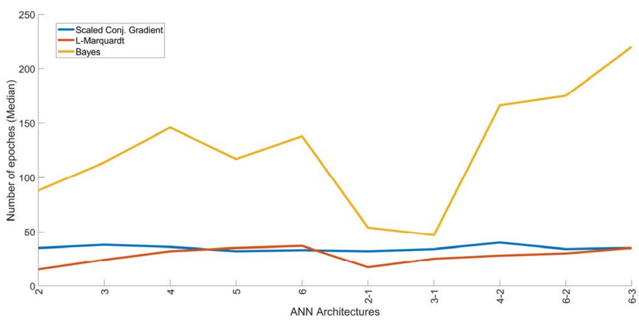

FIGURE 9. Number of Epochs required for each architecture and training algorithm.

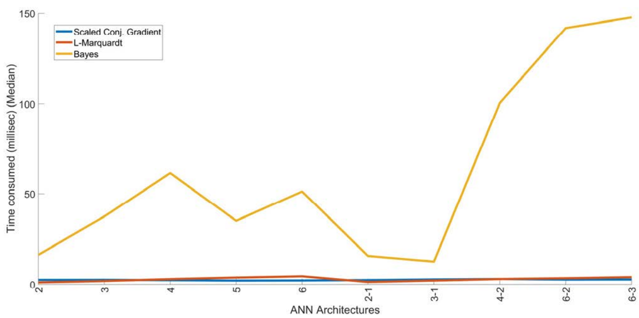

FIGURE 10. Time consumed for each architecture and training algorithm.

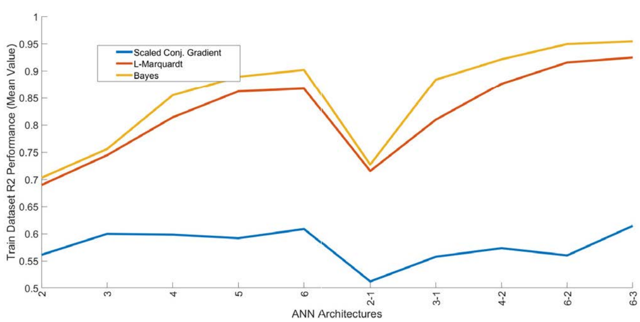

FIGURE 11. Coefficient of determination (R2) of the TDS versus ANN architectures for different training algorithms.

Additionally, Figures 11 and 12 plots the coefficient of determination (R^2) as performance indicators since it considers the intrinsic variance of the training/test data.

The R^2 analysis confirms our initial conclusions, improving the visibility of their variance. Negative values obtained

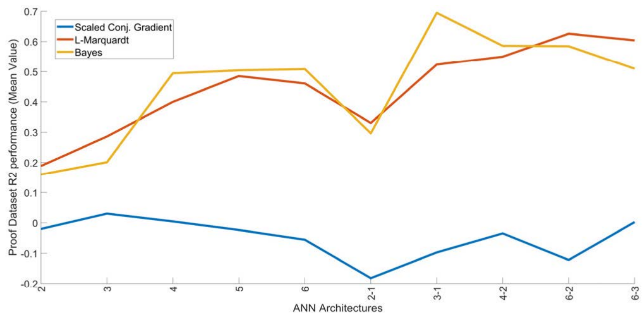

FIGURE 12. Coefficient of determination (R2) of the PDS versus ANN architectures for different training algorithms.

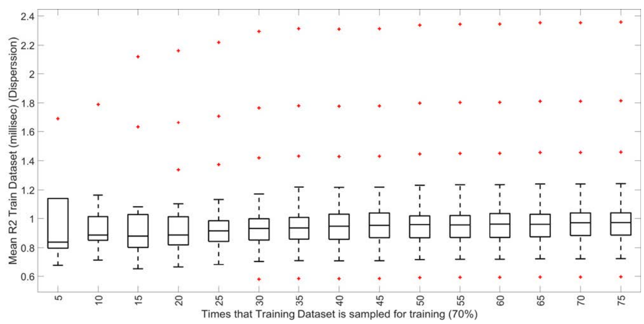

FIGURE 13. Variation of the coefficient of determination (R2) for the TDS with the sample size used for training.

for certain ANN architectures are possible and mean that the goodness of these models are even worse than the horizontal mean [56]. The parametrical contrast of the normalized values of TDS and PDS for R^2 performance concludes that both datasets have the same probability distribution with SCG and LM learning algorithms.

Figure 13 visualize a boxplot to depict the variations of the R^2 for the TDS with the sample size used for training, observing similar stabilization with the sample size to that of the MSE.

We observed that the different architectures showed very different performances with the proof dataset when compared to the TDS. The median and mean TDS MSE performance improve with ANN complexity. The median and mean PDS MSE performance also improved with the same ANN complexity. Both TDS and PDS have similar MSE performance levels even when the PDS is different from the TDS.

Now, the best performance in terms of the MSE is achieved by the Bayesian learning algorithm, and the worst is achieved by the scaled conjugate gradient. However, the worst time consumer is the Bayesian algorithm, and we observe that the Levenberg-Marquardt algorithm performance is slightly worse than that of the Bayesian algorithm, but it is faster; as it is a good trade-off between the two the Levenberg-Marquardt algorithm is the favorite in this experiment.

IV. CONCLUSION AND FUTURE WORK

As the number of neurons increases, the MLP architecture performance improves. The performance with a two-layer architecture also increases with the number of neurons.

The best performance value is for the MLP with six neurons in the input layer and two or three neurons in the output layer.

It is important to note that with only 648 samples of lamps in the dataset with certain conditions, it is possible to train an MLP that provides a good model for another dataset of different lamps in other environmental conditions. This promising result will be extended by research in the near future. The best performance of the MLPs used in the experiment with the second dataset is again a 6-2 architecture, giving a similar order of magnitude compared with the original dataset.

The Bayesian training algorithm achieved the best performance, while the scaled conjugate gradient method achieved the worst. However, the time consumed by the Bayesian training algorithm is nearly 10 times higher for the 6-2 architecture. The time consumed seems to be proportional to the number of epochs that each algorithm repeats in the simulation. Bayesian training takes 270 epochs to model the 6-2 architecture, while the SCG and Levenberg Marquardt algorithms only require approximately 50 epochs. It is noticeable that for the two latter methods, the time consumption and the epochs are quite steady.

The Levenberg-Marquardt performance is similar to that of the Bayesian algorithm, and the resource consumption, which is similar to that of the scaled conjugate gradient method, also works, making it the best option as a training algorithm.

The simulation process is repeated to obtain more reliable values, as it is impossible to establish a model with only one sample. The sample size has been roughly calculated for a given confidence level and average MSE. Both the median and mean values are similar, and their boxplot figures confirm the analysis

This research proposes a new methodology to analyze the best ANN architecture to simulate the behavior of a lighting system. ANNs have the advantage of learning and modeling complex relationships in nonlinear systems. After the learning process, ANNs can generalize and infer unseen relationships on unseen data, making the model generalizable and predictive. Because ANNs do not impose any restrictions on the input variables, they can better model highly volatile data with nonconstant variance. Unlike many other prediction techniques, ANNs have parallel processing capability and high-speed response [57].

The empirical nature of this paradigm requires methodical studies on different architectures to widen the initial scope. Future research aims to extend the analysis as follows:

1) To include multiple outputs for designer work.
2) The study of the influence of each input variable explaining the outputs.
3) To include the rest of variables normally considered by road lighting simulators, analyzing their predominance with a previous feature analysis.
4) To study the online learning mechanisms for unexpected variables, like climate, human behavior, aesthetics considerations or nearby obstacles.
5) The use of signal processing to transform the lamp variables into ANN readable values (new types of lamps, such as solar-powered lamps, could be easily included in the model [58], [59]).
6) Analysis of the ANN architectures behavior with different activation functions.
7) Analysis of new ANN architectures that could bring more efficiency in the optimization and adaptation to context information, such as weather variability [60], [61].
8) Analysis of evolutionary optimization algorithms to for energy savings and other requirements from public lighting designers and site surveys.

REFERENCES

[1] M. R. Montgomery, "The urban transformation of the developing world," *Science*, vol. 319, no. 5864, pp. 761–764, 2008.

[2] W. Paul and T. Satoshi, "Light's labour's lost: Policies for energy-efficient lighting," Int. Energy Agency, Paris, France, Tech. Rep., 2006. doi: 10.1787/19900694.

[3] C. Atıcı, T. Özçelebi, and J. J. Lukkien, "Exploring user-centered intelligent road lighting design: A road map and future research directions," *IEEE Trans. Consum. Electron.*, vol. 57, no. 2, pp. 788–793, May 2011.

[4] S. Escolar, J. Carretero, M. C. Marinescu, and S. Chessa, "Estimating energy savings in smart street lighting by using an adaptive control system," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 5, 2014, Art. no. 971587.

[5] A. Ozadowicz and J. Grela, "Energy saving in the street lighting control system—A new approach based on the EN-15232 standard," *Energy Efficiency*, vol. 10, pp. 563–576, Jun. 2017. Accessed: Jul. 6, 2018. doi: 10.1007/s12053-016-9476-1.

[6] R. C. Coetzer and G. P. Hancke, "Eye detection for a real-time vehicle driver fatigue monitoring system," in *Proc. Intell. Vehicles Symp. (IV)*, Baden-Baden, Germany, Jun. 2011, pp. 66–71.

[7] Ö. Güler and S. Onaygil, "A new criterion for road lighting: Average visibility level uniformity," *J. Light Vis. Environ.*, vol. 27, no. 1, pp. 39–46, 2003.

[8] N. Matout, "Estimation of the influence of artificial roadway lighting on road collision frequency," M.S. thesis, Gina Cody School Eng. Comput. Sci., Building, Civil Environ. Eng., Concordia Univ., Montreal, QC, Canada, 2013.

[9] M. Dully, "Traffic safety evaluation of future road lighting systems," M.S. thesis, Dept. Sci. Technol., Linköping Univ., Linköping, Sweden, 2013.

[10] T. Lorenc, M. Petticrew, M. Whitehead, D. Neary, S. Clayton, K. Wright, H. Thomson, S. Cummins, A. Sowden, and A. Renton, "Environmental interventions to reduce fear of crime: Systematic review of effectiveness," *Syst. Rev.*, vol. 2, 2013, Art. no. 30. doi: 10.1186/2046-4053-2-30.

[11] C. R. Jeffery, *Crime Prevention Through Environmental Design*. New York, NY, USA: MAC, 1972.

[12] A. Castillo, A. Gutiérrez, J. M. Gutiérrez, J. M. Gómez, and E. García-López, "Blackout on outdoor light," in *Proc. Int. Conf. Water, Energy Environ. Manage. (ICWEEM)*, Madrid, Spain, no. 75, 2013, pp. 786–789.

[13] A. Gutierrez-Escolar, A. Castillo-Martinez, J. M. Gomez-Pulido, J.-M. Gutierrez-Martinez, Z. Stapic, and J.-A. Medina-Merodio, "A study to improve the quality of street lighting in Spain," *Energies*, vol. 8, no. 2, pp. 976–994, 2015.

[14] F. Fournier, W. J. Cassarly, and J. P. Rolland, "Method to improve spatial uniformity with lightpipes," *Opt. Lett.*, vol. 33, no. 11, pp. 1165–1167, 2008.

[15] H. Yang, J. W. M. Bergmans, T. C. W. Schenk, J. P. M. G. Linnartz, and R. Rietman, "Uniform illumination rendering using an array of LEDs: A signal processing perspective," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1044–1057, Mar. 2009.

[16] A. Gutierrez-Escolar, A. Castillo-Martinez, J. M. Gomez-Pulido, J.-M. Gutierrez-Martinez, E. P. D. González-Seco, and Z. Stapic, "A review of energy efficiency label of street lighting systems," *Energy Efficiency*, vol. 10, no. 2, pp. 265–282, 2017.

[17] AGi32. *Illumination Engineering Software*. Accessed: May 29, 2018. [Online]. Available: https://lightinganalysts.com/

[18] DIALux. *Professional Lighting Design*. Accessed: May 29, 2018. [Online]. Available: https://www.dial.de/

[19] DL-Light. *Consultancy and Research for Natural and Artificial Lighting*. Accessed: May 29, 2018. [Online]. Available: https://deluminaelab.com/en/what-is-dl-light/
[20] FocusTrack. *Production Lighting*. Accessed: May 29, 2018. [Online]. Available: http://www.focustrack.co.uk/
[21] TracePro. *Illumination and Non-Imaging Optical Design and Analysis Program*. Accessed: May 29, 2018. [Online]. Available: https://www.lambdares.com/tracepro/
[22] LD Assistant. *Design and Drafting Development Software*. Accessed: May 29, 2018. [Online]. Available: https://www.design-drafting.com/
[23] Vectorworks Spotlight. *Industry-Leading Design Software*. Accessed: May 29, 2018. [Online]. Available: http://www.vectorworks.net/spotlight
[24] F. Viani, A. Polo, F. Robol, E. Giarola, and A. Ferro, "Experimental validation of a wireless distributed system for smart public lighting management," in *Proc. IEEE 2nd Int. Smart Cities Conf. (ISC2)*, Sep. 2016, pp. 1–6. doi: 10.1109/ISC2.2016.7580852.
[25] Z. Perko, D. Topic, and J. Perko, "Standardized system for monitoring and control of public lighting networks," in *Proc. Int. Conf. Smart Syst. Technol.*, Oct. 2017, pp. 45–50. doi: 10.1109/SST.2017.8188668.
[26] R. Matos, P. S. Paulo, R. Ribeiro, J. O. Nunes, and P. Valverde, "Smart led lighting systems implementation in Lisbon metropolitan area," in *Proc. CIRED Workshop*, 2016. doi: 10.1049/cp.2016.0623.
[27] E. R. Sanseverino, G. Scaccianoce, V. Vaccaro, G. Zizzo, and S. Pennisi, "Smart city and public lighting," in *Proc. IEEE 15th Int. Conf. Environ. Elect. Eng. (EEEIC)*, Jun. 2015, pp. 665–670.
[28] H.-J. Suk, "Designing user-centered lighting scenarios," in *Proc. Light, Energy Environ. Solid-State Organic Lighting*, 2015, Paper DW2C.1. doi: 10.1364/SOLED.2015.DW2C.1.
[29] A. Del Corte-Valiente, J. L. Castillo-Sequera, A. Castillo-Martinez, J. M. Gómez-Pulido, and J.-M. Gutierrez-Martinez, "An artificial neural network for analyzing overall uniformity in outdoor lighting systems," *Energies*, vol. 10, no. 2, p. 175, 2017.
[30] S. H. Shikder, M. Mourshed, and A. D. Price, "Luminaire position optimisation using radiance based simulation: A test case of a senior living room," in *Proc. Int. Conf. Comput. Civil Building Eng. (ICCCBE)*, Nottingham, U.K., Jun./Jul. 2010, pp. 379–387.
[31] N. Zafar, P. K. Soori, and M. Vishwas, "The use of simulation tools in teaching lighting system design," in *Proc. IEEE 7th Int. Power Eng. Optim. Conf.*, Jun. 2013, pp. 459–463.
[32] L. E. Mavromatidis, X. Marsault, and H. Lequay, "Daylight factor estimation at an early design stage to reduce buildings' energy consumption due to artificial lighting: A numerical approach based on Doehlert and Box–Behnken designs," *Energy*, vol. 65, pp. 488–502, Feb. 2014.
[33] B. van Wout, *Road lighting: Fundamentals, Technology and Application*. Amsterdam, The Netherlands: Springer, 2014.
[34] A. Pellegrino, "Assessment of artificial lighting parameters in a visual comfort perspective," *Int. J. Lighting Res. Technol.*, vol. 31, no. 3, pp. 107–115, 1999.
[35] A. Mosa, K. Kyamakya, M. Ali, and F. Al Machot, "Online self-adaptive cellular neural network architecture for robust time-series forecast," 2016.
[36] T. Rashid, *Make Your Own Neural Network*. Seattle, WA, USA: CreateSpace, 2016.
[37] L. Breiman, "Comment-neural networks: A review from a statistical perspective," *Stat. Sci.*, vol. 9, no. 1, pp. 38–41, 1994.
[38] R. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Mag.*, vol. 4, no. 2, pp. 4–22, Apr. 1987. doi: 10.1109/MASSP.1987.1165576.
[39] G. AurÈlien, *Hands-On Machine Learning with Scikit-Learn and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems*, 1st ed. Newton, MA, USA: O'Reilly Media, 2017.
[40] J. A. Hertz, "Neurocomputing," *Nature*, vol. 359, p. 463, Oct. 1992. 10.1038/359463a0.
[41] N. Pat, *Neural Networks and Deep Learning: Deep Learning Explained to Your Granny—A Visual Introduction for Beginners Who Want to Make Their Own Deep Learning Neural Network*. Seattle, WA, USA: CreateSpace, 2017.
[42] M. T. Hagan, H. B. Demuth, and M. H. Beale, *Neural Network Design*. USA: Martin Hagan, 2014.
[43] N. Michael. (2018). *Neural Networks and Deep Learning*. Accessed: Jun. 15, 2018. [Online]. Available: http://neuralnetworksanddeeplearning.com/index.html
[44] K. Madsen, H. B. Nielsen, and O. Tingleff, "Methods for non-linear least squares problems," 2nd ed., Inform. Math. Model., Tech. Univ. Denmark, DTU Lecture Note, 2004, p. 60. Accessed: Jan. 25, 2017. [Online]. Available: http://www2.imm.dtu.dk/pubdb/views/edoc_download.php/3215/pdf/imm3215.pdf
[45] M. L. A. Lourakis and A. A. Argyros, "Is Levenberg–Marquardt the most efficient optimization algorithm for implementing bundle adjustment?" in *Proc. 10th IEEE Int. Conf. Comput. Vis.*, Marseille, France, Oct. 2005, pp. 12–18.
[46] B. M. Wilamowski, S. Iplikci, O. Kaynak, and M. O. Efe, "An algorithm for fast convergence in training neural networks," in *Proc. Int. Joint Conf. Neural Netw.*, Washington, DC, USA, Jul. 2001, pp. 15–19.
[47] O. Makarynskyy, "Improving wave predictions with artificial neural networks," *Ocean Eng.*, vol. 31, pp. 709–724, Apr. 2004.
[48] M. Seyedali, "Evolutionary multi-layer perceptron," in *Evolutionary Algorithms and Neural Networks* (Studies in Computational Intelligence). New York, NY, USA: Springer, 2018. doi: 10.1007/978-3-319-93025-1_7.
[49] R. Kruse, C. Borgelt, F. Klawonn, C. Moewes, M. Steinbrecher, and P. Held, "Multi-layer perceptrons," in *Computational Intelligence*. London, U.K.: Springer, 2013. doi: 10.1007/978-1-4471-5013-8_5.
[50] A. Choromanska, M. Henaff, M. Mathieu, A. G. Ben, and Y. LeCun, "The loss surfaces of multilayer networks," *J. Mach. Learn. Res.*, vol. 38, pp. 192–204, Feb. 2015.
[51] S. Arora, N. Cohen, N. Golowich, and W. Hu, "A convergence analysis of gradient descent for deep linear neural networks," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–35.
[52] J. T. Heaton, *Introduction to Neural Networks With Java*, 2nd ed. Heaton Research, 2008.
[53] T. Masters, *Practical Neural Network Recipies in C++*. Burlington, MA, USA: Morgan Kaufmann, 1993.
[54] A. Bayram, A. Bilici, and H. Kutucu, "A fast and simple computer aided lighting simulator," in *Proc. Int. Conf. Adv. Technol., Comput. Eng. Sci.*, Safranbolu, Turkey, 2018.
[55] A. Bielecki, "Foundations of artificial neural networks," in *Models of Neurons and Perceptrons: Selected Problems and Challenges* (Studies in Computational Intelligence). Newbury Park, CA, USA: Sage, 2019.
[56] R. Taylor, "Interpretation of the correlation coefficient: A basic review," *J. Diagnostic Med. Sonography*, vol. 6, no. 1, pp. 35–39, 1990.
[57] J. R. Galvão, L. M. Moreira, R. M. T. Ascenso, and S. A. Leitão, "Energy systems models for efficiency towards Smart Cities," in *Proc. Int. Conf. Comput. Tool (EUROCON)*, Sep. 2015, pp. 1–6.
[58] S. J. Myer, P. H. Cooperrider, and D. Gonzalez, "Energy-efficient solar-powered outdoor lighting," U.S. Patent 2015 0 021 990 A1, Jan. 22, 2015.
[59] S. Katoch, G. Muniraju, S. Rao, A. Spanias, P. Turaga, C. Tepedelenlioglu, M. Banavar, and D. Srinivasan, "Shading prediction, fault detection, and consensus estimation for solar array control," in *Proc. IEEE Ind. Cyber-Phys. Syst. (ICPS)*, May 2018, pp. 217–222.
[60] S. Katoch, P. Turaga, A. Spanias, and C. Tepedelenlioglu, "Fast non-linear methods for dynamic texture prediction," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 2107–2111. doi: 10.1109/ICIP.2018.8451479.
[61] R. S. José, J. L. Pérez, L. Péreza, and R. M. G. Barras, "Effects of climate change on the health of citizens modelling urban weather and air pollution," *Energy*, vol. 165, pp. 53–62, Dec. 2018. doi: 10.1016/j.energy.2018.09.088.

ALBERTO GARCES-JIMENEZ was born in Madrid, Spain, in 1964. He received the B.S. and M.Sc. degrees in telecommunication engineering from the Polytechnic School of Madrid, Spain, in 1991. He is currently pursuing the Ph.D. degree from the Health Computing and Information Systems Center, University of Alcala, Spain.

He has also developed an extensive career in telecommunications and computing industries, from 1992 to 2016, where he has performed business engineering support and managerial activities in large corporations and SMEs. He is currently a Researcher with the Biomedical Research Foundation, Hospital Principe Asturias, Alcala de Henares, Spain, and the Center for Research and Innovation in Knowledge Management, University of Francisco de Vitoria, Pozuelo de Alarcón, Spain. He is also a Professor with the Polytechnic School of University Francisco de Vitoria, where he has been teaching computing engineering degree and industrial systems engineering degree, since 2016.

JOSÉ LUIS CASTILLO-SEQUERA received the degree in computing from the National University of San Marcos, Peru, the master's degree in computer project management and the master's degree in university teaching in Spain, and the Ph.D. degree in information systems, documentation, and knowledge from the University of Alcala.

He is currently a Full Professor with the Department of Computer Science, University of Alcala, Spain. He has multiple publications at the level of high-level indexed JCR journals and is the author of books and book chapters related to the field of artificial intelligence. His research interests include information retrieval, knowledge extraction based on evolutionary learning and neural network, data mining and big data, documentation, learning, and teaching innovation with Web 2.0. He also leads projects of technological innovation oriented to the use of ICTs in university research groups. Their areas of interests focus on business technologies and intelligent technologies for knowledge management with interdisciplinary technologies.

ANTONIO DEL CORTE-VALIENTE was born in Barcelona, Spain, in 1969. He received the B.S. and M.S. degrees in telecommunications engineering and the Ph.D. degree in computer science from the University of Alcala, Madrid, Spain, in 2005, 2011, and 2011, respectively, where he has been a Research Assistant Professor with the Department of Computer Engineering, since 2005.

He is also a Senior Consultant in a state-owned consulting and engineering Company, in-house technical services provider of the Spanish General Administration and serving national and international public organizations. He has extensive experience in defense and aerospace sectors. His areas of expertise are communications systems, navigation, surveillance, and air traffic management. He has authored more than 30 peer-reviewed publications and conferences related with electromagnetic radiation, radio-propagation, sensor networks, localization algorithms, and ray-tracing techniques for mobile communications. His research interests include computer science, artificial intelligence, and telecommunications technology (UNESCO Codes: 1203.04 and 3325.05).

JOSÉ MANUEL GÓMEZ-PULIDO was born in Cáceres, Spain. He received the B.S. and M.S. degrees in telecommunications engineering from the University Polytechnic of Madrid, Spain, in 1988 and 1994, respectively, and the Ph.D. degree in telecommunication from the University of Alcala, Madrid, in 2004. In 1992, he was an Assistant Researcher and a Professor with the University Polytechnic of Madrid. From 1993 to 2004, he was with the Signal Theory and Communications Department, University of Alcala, where he is currently an Assistant Professor with the Computational Science Department. He has participated in more than 24 research projects several research projects with Spanish and European companies, related with radio propagation in mobile communications, signal processing in radar, ultrasound and laser applications, EMI–EMC design in on-board satellite instrumentation, radio propagation materials proprieties, numerical technics applied in telecommunication, and so on. His research interests include the areas of wireless sensors networks, smart grids, and new technologies in the data mining and artificial intelligence world applied to the ICT.

ESTEBAN PATRICIO DOMÍNGUEZ GONZÁLEZ-SECO received the Ph.D. degree (Hons.) in industrial engineering from the Polytechnic University of Madrid, Spain.

He is currently an Industrial Engineer in the specialty of Energy. His dissertation was titled Analysis of the Influence of the Neutral Regime on Electromagnetic Compatibility in Hospital Buildings and Proposal of a New Corrective Techniques. His professional career combines the engineering activities of installations (electrical, mechanical, and special) in singular projects (hospitals, building in height, and so on) with his teaching and research facet. In the university, he is a Professor in more than five specialization masters (climate control and sustainable rehabilitation of buildings, efficient building, energy and environmental rehabilitation, home automation and digital home, and pathologies of bioclimatic architecture building). He currently directs three doctoral theses, having collaborated in 20 technical scientific publications, participated in four technical books, as well as a Speaker at specialized conferences. He is also a member of the Technical Committee of American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE).