19

Optimising Neural Networks for Water Stress Prediction in Europe: A Sustainable Approach

Laura Sanz-Martín[1], Manal Jammal[2], and Javier Parra-Domínguez[1]

[1]University of Salamanca, Spain
[2]IoT Digital Innovation Hub, Spain

Abstract

Sustainability in water resource management is critical, given the necessity to monitor and predict key indicators such as SDG 6.4.2: Water Stress. This indicator, which measures the ratio of water resources abstracted relative to their availability, is vital for assessing the pressure on water resources and ensuring their long-term sustainability. The objective of this study is to compare various neural network architectures utilizing different optimisers to predict water stress. The analysis was based on a dataset sourced from AQUASTAT, encompassing data from 28 European countries. Several neural network architectures, with configurations ranging from two to four layers, were implemented, and evaluated using optimisers including SGD, Adam, RMSprop, and Adagrad. The findings revealed that a three-layer architecture combined with the Adam optimiser delivered the best performance, achieving an MSE of 0.02187 and a R^2 of 0.9745, indicating high predictive accuracy. Nevertheless, a two-layer architecture with the SGD optimiser also exhibited strong performance, highlighting its simplicity and effectiveness. These results underscore the importance of meticulous selection of both architecture and optimiser when predicting critical indicators such as water stress. This study not only enhances the accuracy of water-related risk predictions but also supports informed decision-making for sustainable water resource management in Europe.

Keywords: sustainability, water stress, artificial neural networks (ANN), artificial intelligence (AI), optimizers.

19.1 Introduction and Background

For several decades, climate change has affected the hydrological cycle, modifying rainfall patterns, raising water temperature, and intensifying extreme phenomena such as floods and droughts. These changes put pressure on water resources, with potential adverse impacts on ecosystems and human health. In addition, sea level rise may intensify the salinization of groundwater and estuaries, decreasing the presence of freshwater in coastal areas. Furthermore, climate variations in water volume and quality are expected to impact food availability and access to water, particularly in arid areas, and impact water infrastructure such as irrigation systems and hydropower [1]. In Europe, climate change will accentuate the differences in water resources between north and south. In the north, precipitation is expected to increase, improving water availability, but with risks of flooding and increased greenhouse gas emissions from the decomposition of carbon in the soil. In the south, especially in the Mediterranean regions, there will be a decrease in water supply, increasing irrigation demand and agricultural vulnerability, as well as problems of erosion, salinization and soil degradation. Changes in rivers and more frequent droughts and floods will affect the carbon cycle, complicating mitigation efforts [2].

Water stress occurs when the demand for water exceeds the quantity available during a given period or when poor quality limits its use, resulting in the deterioration of freshwater resources in terms of quantity and quality (https://www.eea.europa.eu/). Water stress, intensified by climate change, population growth and economic development, affects both human consumption and key sectors such as agriculture, energy and industry. This phenomenon, aggravated by the degradation of ecosystems and the depletion of water sources, calls for adaptive water management to ensure sustainability and economic stability. Global warming is expected to significantly alter water availability, with increased risks of droughts and floods. By 2030, global water consumption is projected to exceed 160% of available supply, exacerbating global water challenges [3].

Water stress is closely related to Sustainable Development Goal (SDG) 6. Modifications in the hydrological cycle, such as changes in precipitation and water salinization, directly affect access to clean and safe water, especially in coastal areas and arid regions. This underscores the need to implement solutions that address not only the quantity of water available, but also its quality, in line with SDG 6 targets. To address these global challenges, it is crucial to improve the capacity to predict and manage water stress and other indicators related to water availability. In this context, technologies such as Artificial Intelligence (AI) and machine learning models play an increasingly relevant role. These models provide advanced tools for the analysis and prediction of complex water-related phenomena, such as water stress, by processing large amounts of climatic, hydrological, and agricultural data efficiently.

Early prediction of water stress is vital to mitigate impacts on agriculture, energy and other sectors that are highly dependent on water. AI models enable more accurate, real-time analysis of key indicators such as the water stress index and other relevant parameters. These predictions help optimize decision-making on water use, facilitating more effective adaptation strategies in response to climate variations. Despite the growing interest in the use of machine learning models for water resources management, there is a lack of studies that systematically compare various neural network architectures and optimizers specifically in the context of water stress prediction in Europe. Many previous works have focused on general applications of neural networks in water data modelling but have not performed a comprehensive assessment of how different configurations affect predictive accuracy in a specific regional context. Furthermore, the integration of AQUASTAT data, which spans multiple European countries, provides a unique opportunity to address this gap by facilitating the identification of patterns and trends that may be critical for sustainable water management. This study seeks to close this gap by providing a rigorous and contextualized comparison of machine learning models applied to European water data, thus contributing to the development of more effective decision-making tools for water resources management.

The objective of this study is to compare different artificial neural network architectures using different optimizers to predict water stress in 28 European countries, using an AQUASTAT dataset. Through this comparison, we seek to evaluate the performance of each architecture and optimizer in terms of predictive accuracy. Identifying the most effective configuration will not only contribute to improve informed decision making for sustainable water resources management in Europe but will also allow policy makers and water managers to select analytical tools that optimize water use.

This article is structured as follows: first, we will present a state-of-the-art review of the main papers on water stress prediction, analyzing the methodologies and approaches used in previous research. Subsequently, the use of different neural network architectures and optimizers in this context will be discussed, highlighting their advantages and disadvantages. Next, the methodology employed in this study will be described, including data selection. Finally, the results obtained from the comparison between different neural network architectures and optimizers will be presented and discussed, as well as the implications of the findings for sustainable water resources management in Europe, together with recommendations for future research.

19.2 State of the Art

Concern for water availability has gained relevance in a global context where climate change and the growing demand for water resources are intertwined with sustainability objectives. In this regard, the energy sector faces significant challenges, particularly in thermoelectric generation, which requires large volumes of water for cooling. Recent studies have assessed how periods of extreme heat impact electricity generation, highlighting the reduction in production at 1,326 thermoelectric plants in the European Union. Despite efforts to reduce water withdrawals, the number of watersheds experiencing water stress is expected to increase, emphasizing the need to implement integrated approaches to water and energy resource management [4].

On the other hand, the sensitivity of river basins to climate change has been analyzed using advanced methodologies that combine climate modelling and multi-model simulations. This approach has made it possible to classify different basins according to their vulnerability, revealing that, although the Nordic basins show high sensitivity, those of southern and central Europe face greater challenges in overcoming low flow and water stress thresholds [5]. This research underlines the importance of proactive and adaptive management to cope with the adverse effects of climate change on water availability.

In addition, the use of artificial intelligence techniques to characterize and predict water stress has gained attention in several regions. A study conducted in Hyderabad, India, demonstrates how models such as support vector regression (SVR) outperform traditional approaches such as ARIMA in drought prediction. The results highlight the ability of artificial intelligence to capture complex patterns in data, which can significantly contribute to more efficient water resource management [6].

Finally, the application of thermography in agriculture has been shown to be a valuable resource for detecting water stress in crops such as maize. A study in Thailand suggests that continuous monitoring of the Crop Water Stress Index (CWSI) can effectively predict yield losses under drought conditions. This approach highlights the importance of integrating technology into agricultural management to optimize water use [7]. Taken together, these studies evidence the need to adopt holistic strategies that address water stress from multiple perspectives, combining climate research, artificial intelligence, and innovation in agricultural practices.

19.3 Material and Methods

19.3.1 Data

Our database includes 4 variables and 337 records, with information from 28 European countries related to SDG 6.4 indicators: Water Use Efficiency and Water Stress, covering the period from 2010 to 2021. The Water Use Efficiency indicator measures the economic value added generated by each unit of water used (expressed in dollars per cubic meter). Its objective is to evaluate how much economic value is produced with each cubic meter of water withdrawn and used in the economy, considering the following sectors:

1. Agriculture, the largest consumer of water.
2. Industry, where water is essential for production processes.
3. Services, where its use can be more efficient.

The Water Stress Level indicator measures the percentage of renewable freshwater withdrawal in a region or country, comparing water demand with total availability. It is expressed as the percentage of renewable water resources withdrawn in a year. A high value indicates higher water stress, while a low value reflects more sustainable water use.

Indicator 6.4.2 is defined as the ratio of total freshwater withdrawal (TFWW) in all major sectors to the difference between total renewable freshwater resources (TRWR) and environmental flow requirements (EFR). It is calculated using the following formula [12]:

$$\textit{Water Stress}\ (\%) = \frac{TFWW}{TRWR - EFR} \times 100. \tag{19.1}$$

Table 19.1 presents descriptive statistics on Water Use Efficiency and Water Stress. Water use efficiency averages 134.95 $/m^3, with a standard deviation of 215.60 $/m^3, indicating considerable variability among European countries. The minimum value observed is $6.29/m^3 and the maximum reaches $1,294.91/m^3, suggesting large differences in water management. The median is 80.41 $/m^3, indicating that half of the observations are below this value.

For water stress, the average is 20.74%, with a standard deviation of 19.13%, reflecting significant variations. The minimum value is 0.99%, showing that some areas do not face water stress, while the maximum is 91.29%, indicating a risk to sustainability in certain regions. The median of 17.24% indicates that half of the areas analyzed have relatively low water stress.

Table 19.2 shows the descriptive statistics of water stress in European countries from 2010 to 2021. In general, there is a large variability in water stress levels between countries. Malta has the highest average level at 83.19%, indicating chronic pressure on its water resources. Other countries with high stress levels are Belgium (56.43%) and Bulgaria (41.19%), while Croatia (1.45%) and Austria (9.08%) show considerably lower levels.

Comparing the average data with that of 2021, some countries, such as Austria, Belgium and Czechia, have reduced their water stress in 2021 relative to the averages of the 2010-2021 period. For example, Belgium decreases from 56.43% to 51.88%. However, other countries, such as Cyprus and Denmark, have seen an increase in their water stress levels in 2021, which may reflect higher demand or lower water availability.

Although water stress in Malta decreased slightly in 2021 (78.28%), it is still very high. This highlights the continued pressure on its water resources, attributable to limited freshwater availability. In summary, the data highlight both improvements in water management in certain countries and increases in water stress in others, evidencing the various challenges Europe faces in managing its water resources.

Table 19.1 Descriptive statistics for the two variables under study

	Water Use Efficiency($/m^3)	**Water Stress (%)**
Mean	135,95	20,74
Std	215,60	19,13
Min	6,29	0,99
25%	34,99	6,01
50%	80,41	17,24
75%	143,09	29,8
Max	1294,91	91,29

Table 19.2 Descriptive statistics for the level of water stress of the countries under study

Country	Mean	Std	Min	50%	Max	2021
Austria	9,079396	0,349331	8,67643	9,052531	9,643548	8,67643
Belgium	56,42826	7,228664	49,06634	52,91819	73,13268	51,87961
Bulgaria	41,19484	2,675824	37,5194	41,01338	47,19492	37,5194
Croatia	1,44622	0,065285	1,289266	1,469994	1,518462	1,478435
Cyprus	30,09158	2,179796	27,46036	29,72936	34,89612	32,12138
Czechia	24,90686	2,826334	20,51399	24,8236	29,66849	20,51399
Denmark	24,78854	2,818227	19,69984	24,89713	29,90311	26,40427
Estonia	16,62548	3,929511	9,231085	18,13312	20,28139	10,8198
Finland	7,736582	1,909514	5,454114	7,114062	11,51814	7,114062
France	23,72531	1,359882	21,59814	23,68161	26,39062	21,59814
Germany	39,52151	5,377602	33,50192	37,46542	49,9216	35,35166
Greece	20,28583	0,252374	19,93815	20,28121	20,6838	20,6838
Hungary	8,098914	0,880236	6,775475	8,12532	9,274611	8,070121
Ireland	10,63118	7,986741	4,001176	5,971554	22,20506	22,20506
Italy	29,79749	0,111825	29,64578	29,80592	29,94407	29,64578
Latvia	1,259613	0,351461	0,992929	1,074131	2,175015	1,067831
Lithuania	2,986097	1,668811	1,834102	2,399063	7,467916	1,834102
Luxembourg	3,866086	0,206621	3,573798	3,8267	4,328358	3,963516
Malta	83,18589	5,022172	75,44555	82,18479	91,28713	78,28007
Netherlands	17,59585	2,071677	15,02981	16,90208	20,75185	16,07566
Northern Ireland	13,86417	0,665184	12,42114	14,22251	14,35465	14,35465
Poland	35,61036	3,509251	30	35,88958	41,22534	32,07684
Portugal	15,3874	2,708669	12,31571	17,2372	17,97442	12,31571
Romania	6,26299	0,465228	5,822489	6,060762	7,362606	7,362606
Slovakia	2,501927	0,15682	2,388124	2,421629	2,862737	2,436631
Slovenia	6,420104	0,518243	5,749155	6,296146	7,813387	6,294794
Spain	43,9606	3,191685	39,8289	43,25404	50,10225	43,25404
Sweden	3,592352	0,135445	3,427128	3,567388	3,880231	3,582973

A threshold of 25% has been established to evaluate water stress, where values below are considered safe and values above represent an increasing threat. This stress is classified into five categories: no stress (<25%), low (25-50%), medium (50-75%), high (75-100%) and critical (>100%). Figure 19.1 illustrates the frequency of water stress values in different ranges, using a colour palette ranging from green (low stress) to red (high stress). Most of the observations are in the safe range, indicating that many regions do not face significant pressure on their water resources, as reflected in the high frequency of values between 0 and 25%. However, there are a notable number of areas in the medium stress category (50-75%), with fewer areas in high stress (75-100%). Although there are not many observations that exceed

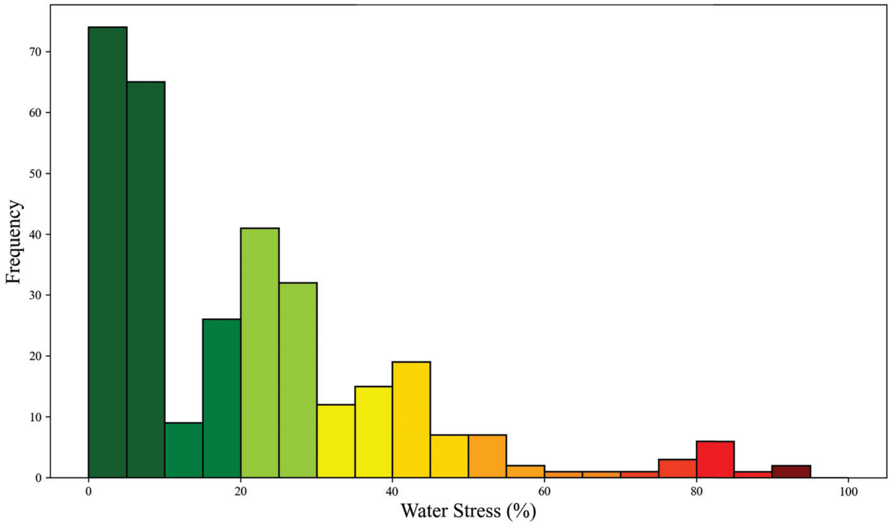

Figure 19.1 Frecuency of water stress (%).

100% (critical category), the presence of some indicates that these regions face serious water sustainability concerns.

19.3.2 Methodology

19.3.2.1 Data

The data were processed to select only the relevant column and then normalized using a MinMax scaler, which ensures that all values were within the range [0, 1]. Normalization is crucial to improve the stability and efficiency of neural network training. Subsequently, temporal sequences were created to capture the dependencies between the data over time. For each output (target) value, the values of the previous 28 observations were used as input. This process of creating sequences turns the problem into one of sequential prediction.

The normalized data were divided into three subsets: training set (70%), used to adjust the model weights, validation set (15%) to monitor model performance during training and avoid overfitting, and test set (15%), reserved for evaluating the final model performance on unseen data.

19.3.2.2 Neural network architecture

Different dense layer architectures with ReLU activations were evaluated for each hidden layer. The following layer configurations were used:

- Architecture (64, 32): This architecture, consisting of two layers with 64 and 32 neurons respectively, was used in the initial training phases. Its purpose was to evaluate the behaviour of the model in the prediction problem with a relatively simple structure. This allowed to obtain a benchmark to compare more complex architectures.
- Architecture (128, 64, 32): By adding an intermediate layer and increasing the number of neurons, this architecture increases the model's ability to learn more complex patterns in the data. This configuration is expected to improve the generalization of the model by capturing deeper relationships between input variables.
- Architecture (256, 128, 64, 32): This deeper architecture with larger number of neurons is designed to address highly complex prediction problems. The addition of more layers and neurons provides the model with the ability to learn high-level feature representations, which can be crucial for improving accuracy in water stress prediction.

In all architectures, optimization techniques, such as Adam, RMSProp and SGD, were applied in order to minimize the loss function during training. Each layer was regularized using the L2 penalty (regularization term) to avoid overfitting. In addition, a Dropout layer with a rate of 30% was included after each hidden layer to improve the robustness of the model and avoid overdependence of some neurons.

19.3.2.3 Model optimization

SGD (Stochastic Gradient Descent) with a learning rate of 0.01 and a momentum de 0.9

The Stochastic Gradient Descent (SGD) methodology, as presented by LeCun et al. (2002), can be summarized in the following key steps in the context of neural network training:

1. Definition of the optimization problem: The objective is to minimize a cost function E(W), which measures the difference between the expected value and the output predicted by the model. The most commonly used cost function is the mean square error (MSE), which is defined as:

$$E(W) = \frac{1}{P}\sum_{p=1}^{P} E_P. \tag{19.2}$$

   Where E_P is the error associated with pattern p, W is the vector of model parameters, and P is the size of the training set.

2. Gradient calculation: At each iteration, the gradient of the cost function with respect to the model parameters is calculated:

$$\nabla_W E(W). \tag{19.3}$$

   This gradient indicates the direction in which the W parameters should be adjusted to reduce the error.
3. Parameter update: The model parameters are updated using the gradient of the cost function computed in the previous step. In SGD, the update is performed for each p training pattern stochastically, i.e., individual samples are used instead of the entire data set:

$$w_{t+1} = w_t - \eta \nabla_w E_p(w). \tag{19.4}$$

   Where η is the learning rate, a hyperparameter that controls the step size taken at each iteration.
4. Repetition: The process of updating the parameters is repeated over several epochs, going through all the samples in the training set several times until the model converges or until a predefined number of iterations is reached.
5. Improving generalization: In addition to minimizing the cost function, the paper also addresses the importance of improving the generalization capability of the model, i.e., the ability of the model to make correct predictions on unseen data. The SGD method is combined with techniques such as regularization to avoid overfitting and improve model generalization.

Adam with a learning rate of 0.001

The Adam (Adaptive Moment Estimation) optimizer is used to update the model parameters during training. Adam combines the advantages of the AdaGrad and RMSProp algorithms, providing an adaptive learning rate adjustment for each model parameter. Its implementation is straightforward, requiring only first-order gradients, making it computationally efficient and suitable for problems with nonstationary targets and noisy or sparse gradients.

The algorithm employs adaptive estimates of the first and second moments of the gradients, automatically adjusting the step size during the optimization process. The standard hyperparameters used were initial learning rate $\alpha = 0.001$, $\beta = 0.999$, and $\varepsilon = 10\text{-}8$, which did not require additional adjustments. During the training process, the initial biases of the moments were corrected by a correction mechanism to ensure proper convergence.

This setup ensures that the model parameters converge efficiently, maintaining a balance between speed of convergence and numerical stability of the training [9].

Key formulas of the Adam algorithm:

1. Moving average of the gradient (first moment):

$$m_t = \beta_1 - m_{t-1} + (1 - \beta_1) \cdot g_t. \tag{19.5}$$

2. Moving average of the squares of the gradient (second moment):

$$v_t = \beta_2 - v_{t-1} + (1 - \beta_2) \cdot g_t^2. \tag{19.6}$$

3. Updating of parameters:

$$\theta_t = \theta_{t-1} - \frac{\alpha \cdot m_t}{\sqrt{v_t} + \varepsilon}. \tag{19.7}$$

RMSprop with a learning rate of 0.0001

RMSProp (Root Mean Square Propagation) is an adaptive optimization algorithm that adjusts the learning rate by dividing the gradient by a moving average of the magnitudes of recent gradients. This technique is useful for handling the oscillations problem in optimization and for improving convergence in deep learning problems, especially in neural networks trained with mini-batches [10].

1. Adaptive update: Unlike AdaGrad, which accumulates gradients indefinitely, RMSProp maintains an exponentially decreasing average of the squares of past gradients. The update of the weights is performed as follows:

   $$x_{t+1} = x_t - \frac{\eta}{\sqrt{E\left[g^2\right]_t + \epsilon}} g_t. \tag{19.8}$$

   Where E[g2]t is the exponential moving average of the squared gradients at time t, and ϵ is a small value to avoid divisions by zero.
2. Exponential moving average: The key to RMSProp is that it calculates a moving average of the squares of the gradients with a decay factor. This allows the algorithm to remain effective even in problems where the gradients vary significantly over time. The calculation of this moving average is defined as:

   $$E\left[g^2\right]_t = \gamma E\left[g^2\right]_{t-1} + (1 - \gamma) g_t^2. \tag{19.9}$$

   Where γ is the decay factor, usually close to 0.9.

Advantages of RMSProp: RMSProp solves the problem of excessive learning rate decay in AdaGrad by limiting the influence of old gradients through the use of a moving average. This allows maintaining an adequate learning rate throughout the training, especially in deep neural networks where gradients can vary considerably between layers. This method has proven to be particularly effective in optimization tasks with redundant data or features of very different magnitudes.

Adagrad with a learning rate of 0.01

AdaGrad is an adaptive optimization algorithm that adjusts the learning rate of each parameter based on the cumulative magnitude of gradients in previous iterations. This adjustment allows the algorithm to be effective in scenarios where features are sparse or gradients vary considerably between dimensions [11].

1. Adaptive update: At each iteration t, the parameters x_t are updated using the gradient g_t, adjusted by a factor that depends on the cumulative sum of squared gradients:

$$x_{t+1} = x_t - \eta \frac{g_t}{\sqrt{G_t + \varepsilon}}. \tag{19.10}$$

   Where Gt is the sum of the squared gradients up to t, and ε is a small value to avoid divisions by zero.
2. Adaptive advantages: AdaGrad automatically adjusts the learning rate for each parameter. Parameters associated with large gradients are updated more slowly, while those with small gradients are adjusted more quickly, improving learning efficiency on sparse data.
3. Mahalanobis norm: AdaGrad uses the Mahalanobis norm to project the updates, which ensures that the update steps are adjusted to the behaviour of the accumulated gradients, resulting in more stable and efficient updates.

This adaptive approach allows AdaGrad to achieve better convergence, especially in problems with sparse data or where features have different scales of importance. To improve the convergence of the model, the following callbacks were implemented during EarlyStopping and ReduceLROnPlateau training.

19.3.2.4 Evaluation and metrics

The model was evaluated using the following metrics:

- RMSE (Root Mean Square Error): measures the standard deviation of the predictions with respect to the actual values.

- MAE (Mean Absolute Error): Measures the average of the absolute errors.
- R^2 Score: Evaluates how well the predictions fit the actual values (coefficient of determination).

These metrics were calculated for both the training and test sets.

19.4 Results

Table 19.3 presents an analysis of various neural network architectures and optimizers, evaluated using the mean square error (MSE), the mean absolute error (MAE) and the coefficient of determination (R^2). These indicators are key to measure the accuracy and generalization capability of the model. In the two-layer architecture (64 and 32 neurons), the Stochastic Gradient Descent (SGD) optimizer stands out with an MSE of 0.0320 in training and 0.0254 in testing, showing good generalization. In contrast, Adam underperforms, while RMSprop and Adagrad offer intermediate results. For the three-layer architecture, Adam achieves the best performance with an MSE of 0.0284 in training and 0.0218 in test, indicating excellent generalization ability. SGD and RMSprop are not as effective in this more complex configuration. In the four-layer architecture, SGD maintains competitive performance with an MSE of 0.0445 in training and 0.0375 in test. Adam fails to match its previous performance, and both Adagrad and RMSprop show higher errors. In summary, SGD is effective on simple architectures, while Adam excels

Table 19.3 Results of the different architectures and optimizers

	MSE train	**MAE train**	**R^2 train**	**MSE test**	**MAE test**	**R^2 test**
2L(64,32) SGD	0.032	0.1149	0.9689	0.0255	0.1066	0.9703
2L(64,32) Adam	0.077	0.2019	0.9253	0.0634	0.1695	0.9261
2L(64,32) RMS	0.0398	0.1512	0.9614	0.0306	0.1228	0.9644
2L(64,32) Ada	0.0388	0.1245	0.9624	0.0344	0.1191	0.9599
3L(128,64,32) SGD	0.1223	0.1827	0.8814	0.0959	0.1646	0.8883
3L(128,64,32) Adam	0.0284	0.1174	0.9724	0.0219	0.0982	0.9745
3L(128,64,32) RMS	0.1205	0.2486	0.8831	0.081	0.1945	0.9057
3L(128,64,32) Ada	0.0544	0.1416	0.9472	0.0446	0.1301	0.948
4L(256,128,64,32) SGD	0.0446	0.1444	0.9568	0.0375	0.1361	0.9563
4L(256,128,64,32) Adam	0.0615	0.1935	0.9404	0.0515	0.1741	0.94
4L(256,128,64,32) RMS	0.0772	0.2415	0.9252	0.0858	0.2574	0.9001
4L(256,128,64,32) Ada	0.0547	0.1764	0.9469	0.0429	0.1636	0.95

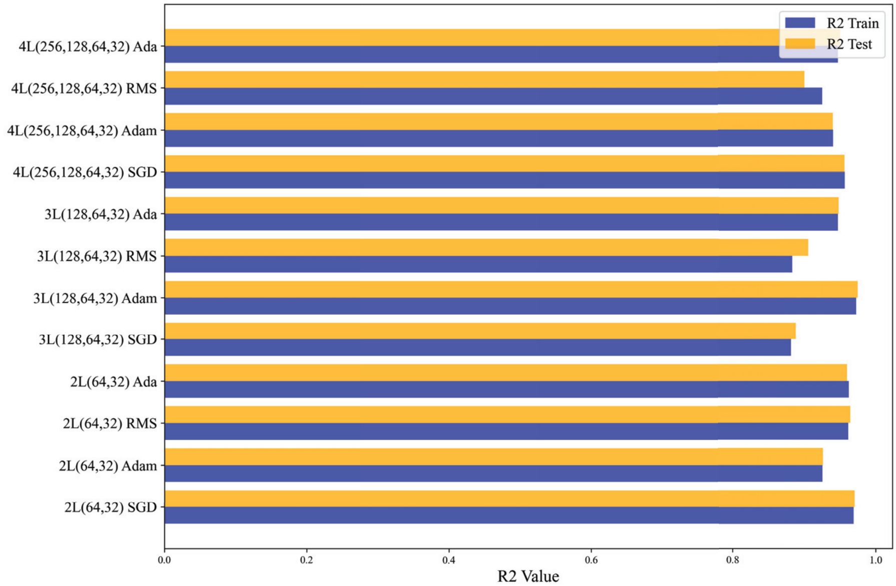

Figure 19.2 Comparison of R^2.

on more complex structures. The choice of optimizer should consider the complexity of the neural network to optimize performance.

Figure 19.3 presents the R^2 values for different combinations of model architectures and optimizers, evaluating their performance on the training and test sets. An R^2 close to 1 indicates a good ability to explain the variability of the data, and the results suggest that there is no overfitting, as the values are similar in both sets. The combination of 2L(64,32) with the SGD optimizer stands out with an R^2 of about 0.970 in both ensembles, showing excellent generalization. Likewise, the 3L(128,64,32) architecture with Adam achieves an R^2 of 0.975 on the test set, indicating optimal performance. However, more complex architectures, such as 4L(256,128,64,32), show slightly lower results, especially with RMSprop, which has an R^2 of 0.900 on test. This suggests that, despite their ability to capture complexities, they do not always improve performance. An important finding is that Adam offers superior performance on complex architectures, while SGD is very competitive on simpler configurations. In summary, simpler models with well-tuned optimizers achieve good results in R^2, and the generalization capability is robust, indicating stability in predicting unseen data.

Figure 4 compares the prediction errors of different neural network models using two metrics: MSE (Mean Squared Error) and MAE (Mean

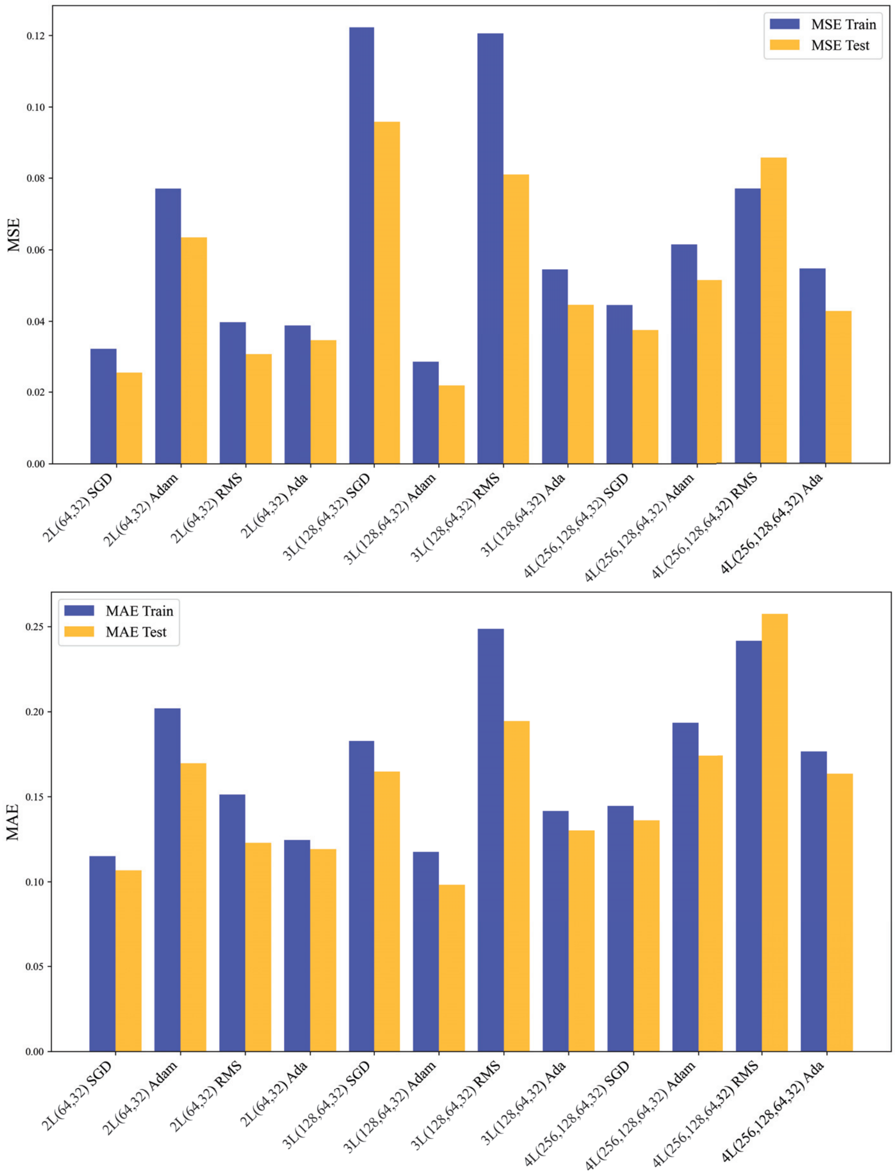

Figure 19.3 Comparison of MSE and MAE.

Absolute Error), for the training and test sets. In the MSE plot, models with optimizers such as Adam and SGD show better results, with the 2L(64,32) model with SGD achieving the lowest MSE, indicating a good fit and balanced prediction ability in both sets. However, more complex models such as 3L(128,64,32) with SGD and 4L(256,128,64,32) with RMS present higher MSE, suggesting difficulties in generalizing. The MAE plot reflects similar patterns, where 2L(64,32) with Adam shows lower absolute error, indicating good accuracy. In contrast, models such as 4L(256,128,64,64,32) with RMS and 3L(128,64,32) with SGD show higher errors, implying optimization problems. In general, simpler architectures, especially 2L(64,32) with optimizers such as SGD or Adam, tend to perform better on both metrics. More complex architectures, such as 4L(256,128,64,32), show worse results, with larger discrepancies between training and test performance, which may indicate overfitting or suboptimal optimization.

19.5 Conclusions

The conclusions of this study highlight key implications for policy adoption and future research design in the field of deep learning. The findings reinforce the need for clear guidelines on the selection of architectures and optimizers in neural networks, not only from a technical perspective, but also to facilitate efficient implementation in practical applications. Simplicity, as observed in the performance of less complex architectures such as 2L(64,32), is a principle that can guide strategic decisions in the development of predictive models, suggesting that in many contexts it is preferable to opt for less complex solutions that balance accuracy and resource efficiency. This approach could be crucial for industries seeking to integrate artificial intelligence into their processes, as it minimizes the risks associated with computational overhead and overfitting.

As for optimizers, the results reinforce the adoption of approaches such as SGD and Adam, which proved to be the most effective depending on the complexity of the architecture. The superior performance of SGD on simple architectures and Adam on more complex models suggests that model development policies should be adaptive, adjusting both the architecture and the optimizer according to the context and specific data needs.

For future research, these results open multiple avenues of exploration. It is essential to continue to evaluate how different datasets, particularly those with more complex nonlinear relationships or larger volumes, affect the performance of more complex architectures. In addition, the impact of hyperparameters should be further investigated and alternative optimizers that have not yet been widely studied in this context, such as AdamW or Nadam, should be tested to confirm the validity of these patterns in different scenarios.

Finally, the transfer of this knowledge to the applied domain, whether in AI policies for industrial sectors or to improve technology adoption in the public sector, requires an evidence-based approach such as the one presented here. The promotion of policies that incentivize simplicity and efficient optimization in model development could have a significant impact, encouraging a more sustainable and effective use of artificial intelligence. The implementation of policies that promote simplicity and optimization in the development of AI models can be key to accelerating their adoption in both industrial and public sectors. The study highlights that simple architectures and efficient optimizers such as SGD or Adam not only improve performance, but also facilitate the scalability and maintainability of technology solutions. In the industrial domain, this could reduce costs and barriers to entry, enabling a democratization of AI. In the public sector, the use of accessible models can contribute to scalable and sustainable solutions, such as predictive health systems and urban resource management. In addition, promoting simplicity in AI design is also aligned with green objectives, reducing energy consumption associated with complex models.

Acknowledgements

This work has been partially funded by the EU Project dAIEDGE (GA Nr 101120726).

References

[1] P. Quevauviller, "Adapting to climate change: reducing water-related risks in Europe–EU policy and research considerations," Environmental Science & Policy, vol. 14, no. 7, pp. 722-729, 2011. https://doi.org/10.1016/j.envsci.2011.02.008

[2] P. Falloon and R. Betts, "Climate impacts on European agriculture and water management in the context of adaptation and mitigation—the importance of an integrated approach," Science of the Total Environment, vol. 408, no. 23, pp. 5667-5687, 2010. https://doi.org/10.1016/j.scitotenv.2009.05.002

[3] S. Lavrnić, M. Zapater-Pereyra, and M. L. Mancini, "Water scarcity and wastewater reuse standards in Southern Europe: focus on agriculture," *Water, Air, & Soil Pollution*, vol. 228, pp. 1-12, 2017. https://doi.org/10.1007/s11270-017-3425-2

[4] P. Behrens, M. T. Van Vliet, T. Nanninga, B. Walsh, and J. F. Rodrigues, "Climate change and the vulnerability of electricity generation to water stress in the European Union," *Nature Energy*, vol. 2, no. 8, pp. 1-7, Aug. 2017. https://doi.org/10.1038/nenergy.2017.114

[5] M. Weiß and J. Alcamo, "A systematic approach to assessing the sensitivity and vulnerability of water availability to climate change in Europe," *Water Resour. Res.*, vol. 47, W02549, 2011, doi:10.1029/2009WR008516.

[6] A. Gorlapalli et al., "Characterization and Prediction of Water Stress Using Time Series and Artificial Intelligence Models," *Sustainability*, vol. 14, no. 11, pp. 6690, 2022. https://doi.org/10.3390/su14116690

[7] C. Pradawet et al., "Thermal imaging for assessment of maize water stress and yield prediction under drought conditions," *Journal of Agronomy and Crop Science*, vol. 209, pp. 56–70, 2023. https://doi.org/10.3390/su14116690

[8] Y. LeCun, L. Bottou, G. B. Orr, and K. R. Müller, "Efficient backprop," in *Neural Networks: Tricks of the Trade*, 2nd ed., Berlin, Germany: Springer, 2002, pp. 9-50. https://doi.org/10.1007/3-540-49430-8_2

[9] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," presented at the *Int. Conf. on Learning Representations (ICLR)*, 2014.

[10] G. Hinton, N. Srivastava, and K. Swersky, "Neural networks for machine learning: Overview of mini-batch gradient descent," presented at the *Univ. of Toronto*, 2012.

[11] J. K. Fang et al., "Adagrad gradient descent method for AI image management," in *Proc. 2020 IEEE Int. Conf. on Consumer Electronics-Taiwan (ICCE-Taiwan)*, 2020, pp. 1-2, https://doi.org/10.1109/ICCE-Taiwan49838.2020.9258085

[12] Food and Agriculture Organization of the United Nations FAO Terminology Portal. https://www.fao.org/faoterm/viewentry/en/?entryId=172346.