

Perspective

Neural Modelling from the Perspective of Selected Statistical Methods on Examples of Agricultural Applications

Piotr Boniecki ¹, Agnieszka Sujak ^{1,*}, Gniewko Niedbała ¹, Hanna Piekarska-Boniecka ²,
Agnieszka Wawrzyniak ¹ and Andrzej Przybylak ¹

¹ Department of Biosystems Engineering, Poznań University of Life Sciences, 50 Wojska Polskiego Str., 60-637 Poznań, Poland

² Department of Entomology and Environmental Protection, Poznań University of Life Sciences, 159 Dąbrowskiego Str., 60-594 Poznań, Poland

* Correspondence: agnieszka.sujak@up.poznan.pl; Tel.: +48-61-846-6074

Abstract: Modelling plays an important role in identifying and solving problems that arise in a number of scientific issues including agriculture. Research in the natural environment is often costly, labour demanding, and, in some cases, impossible to carry out. Hence, there is a need to create and use specific “substitutes” for originals, known in a broad sense as models. Owing to the dynamic development of computer techniques, simulation models, in the form of information technology (IT) systems that support cognitive processes (of various types), are acquiring significant importance. Models primarily serve to provide a better understanding of studied empirical systems, and for efficient design of new systems as well as their rapid (and also inexpensive) improvement. Empirical mathematical models that are based on artificial neural networks and mathematical statistical methods have many similarities. In practice, scientific methodologies all use different terminology, which is mainly due to historical factors. Unfortunately, this distorts an overview of their mutual correlations, and therefore, fundamentally hinders an adequate comparative analysis of the methods. Using neural modelling terminology, statisticians are primarily concerned with the process of generalisation that involves analysing previously acquired noisy empirical data. Indeed, the objects of analyses, whether statistical or neural, are generally the results of experiments that, by their nature, are subject to various types of errors, including measurement errors. In this overview, we identify and highlight areas of correlation and interfacing between several selected neural network models and relevant, commonly used statistical methods that are frequently applied in agriculture. Examples are provided on the assessment of the quality of plant and animal production, pest risks, and the quality of agricultural environments.

Keywords: artificial neural networks; empirical data analysis; statistical methods; agriculture



Citation: Boniecki, P.; Sujak, A.; Niedbała, G.; Piekarska-Boniecka, H.; Wawrzyniak, A.; Przybylak, A. Neural Modelling from the Perspective of Selected Statistical Methods on Examples of Agricultural Applications. *Agriculture* **2023**, *13*, 762. <https://doi.org/10.3390/agriculture13040762>

Academic Editor: Dengpan Xiao

Received: 17 February 2023

Revised: 22 March 2023

Accepted: 23 March 2023

Published: 25 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In scientific research, a model represents the studied object that may differ from that concept which exists in reality. In the above context, the original is always a primary concept in relation to the substitute that is its model. Thus, anything that is similar in some respect to something else, which is referred to as the original, can be regarded as a model. A model can be a static or dynamic structure that represents the original in a satisfactory (in some respects) way. Thus, a model creates the possibility of indirectly examining existing originals to better understand them or to improve them.

Researchers’ aspirations to understand and to explain the laws governing nature more fully and scientifically have led to the importance of searching for new methods that most effectively support cognitive processes [1–3]. These methods undoubtedly include artificial intelligence techniques, among which neural network models occupy a special place. Neural network models are created deductively on sets of assumptions, resulting

from current scientific knowledge concerning the structure and mechanisms of the brain. The constructed models aim to mimic, in particular, those characteristics of biological neural systems that may be of technical use, which primarily include the resilience of biological systems to damage even to a significant part of their components, the extraordinary capacity for learning, and the high speed of information processing. These models, which are essentially probabilistic structural models of the studied systems, allow for effective and fast support and enrichment of the cognitive process, both at the stage of empirical research and at the analytical stage. In agricultural sciences, often, the cause-and-effect relationship of the examined phenomena and processes is unknown due to the complex structure of the investigated problems. Therefore, an obvious consequence is that the mathematical descriptions of the investigated problems are also generally unknown. If only random empirical data or, possibly, the results of computer simulations are available, it seems important to search for alternative methods of building models of empirical systems that describe investigated dependencies in agriculture. Such a possibility is offered by methods based on modern techniques that use neural network models.

In recent years, neural network models have played a special role because they are able to solve a range of issues, including scientific problems defined as unstructured (not susceptible to algorithmising). They can also effectively solve problems for which there is insufficient scientific knowledge or a lack of representative empirical data [4–6]. Neural network models are the general name for mathematical formulas and their software (or hardware) structures that implement signal processing through a network of interconnected elements (neurons) performing elementary mathematical operations. The morphology presented above is inspired by the architecture and functioning of natural neural systems, in particular, the structure and operation of the brain [7,8].

In a mathematical context, neural network models represent weighted graphs that form network topologies in which empirical systems are analogous to processes occurring in biological neuronal structures [9–11].

Neural modelling is the process of creating, verifying, and exploiting generated artificial neural networks. Therefore, the term “neural modelling” can be interpreted either as the process of producing structural models of the brain (generation of hardware and software neural network models) or as the process of reproducing the studied reality (both material and non-material) into an existing (hardware or software) neural (cellular) structure.

Neural network models and traditional statistical methods largely overlap and are complementary. In fact, they are tools for processing (analysing) and extracting, for example, knowledge contained in empirical data, among other things, obtained as a result of conducted experimental research. Therefore, the results of using neural network models and classical statistics methods are often similar, despite the fact that their techniques are fundamentally different.

An additional complication that can overshadow the mutual similarities and interrelationships is the fact that each of these scientific methodologies uses a different set of characteristic terminology. On the one hand, the nomenclature of statistics has been developed over the years and is primarily from the theory of probability. In the terminology used in neural network models, on the other hand, investigators use statistics and are generally seen as researchers primarily concerned with the generalisation process carried out on the basis of noisy data acquired through the design and subsequent execution of an experiment. It is worth noting that, in this context, artificial neural networks creatively complement and even expand traditional statistical models [12–17].

The aim of this perspective paper was to identify and highlight areas of correlation and interfacing between several selected neural network models and relevant commonly used statistical methods. Our work is divided into subsections showing different approaches in the context of models and mathematics. At the end of each chapter, some examples are described on issues in agriculture, agronomy, and agricultural engineering.

2. Overview of Trends and Methods Used by Researchers

An alternative approach to the classical description and analysis of empirical systems is to use methods of artificial intelligence in the broadest sense, and so-called neural network techniques in particular. Artificial neural networks are an intensively evolving field of knowledge increasingly used in many scientific approaches. They are also often of practical importance, where they are used in various types of artificial intelligence applications, for example, expert systems, autonomous systems, etc. Neural networks are versatile approximation systems that represent multidimensional (e.g., empirical) datasets. They have the ability to learn and adapt to changing environmental conditions. They also support the generalisation of acquired knowledge gained through learning. Unlike traditional methods of information processing, offered by cyclic calculating machines (computers) executing a prewritten programme, artificial neural networks are based on optimisation algorithms that enable the design of an appropriate network morphology, and then the selection of parameters of this structure that are adapted to the problem being solved.

2.1. Linear Neural Networks versus Linear Regression Models

The popularity of artificial neural networks is largely due to the possibility of modelling nonlinear issues relatively easily, i.e., practically solving problems described by bias regression models. However, this does not mean that structurally simple linear models should be neglected in the process of analysing empirical data. The general rule used in scientific research is that when there is a choice between a simple and a more complex model, the simpler (less complex) model should always be preferred, unless of course the complex model fits the data significantly better [3,18]. It is worth noting that the simplest model approximating the observed empirical relationship (described mathematically) is the linear model. Its intuitive simplicity is undoubtedly the reason for its great popularity (also in the empirical sciences), in particular, because of the computational possibilities offered by discrete mathematics and, especially, matrix calculus [14,19,20].

A linear model is represented by an artificial neural network with no hidden layers which means that it is one-layer structure (except for the special case of three-layer auto-associative networks implementing dimension reduction of the data vector by means of a linear principal components analysis (PCA) transformation). The neurons in the output layer are fully linear [21,22], i.e., they are neurons in which the total excitation is determined as a linear combination of the input values and which have a linear activation function. Of course, only the second layer processes the information, while the role of the first layer is to introduce the information (signal) into the network. It is important that it has both a linear postsynaptic function and a linear activation function. The training of linear neural networks is performed using supervised methods (with a teacher). During the training process of linear networks, the pseudo-inversion method works best. This technique optimises the output layer of the network, that is, it directly provides optimal coefficient values for all neurons of the entire linear layer of the network.

The statistical equivalents for standard linear neural network models are logistic regression for classification issues and linear (least squares) regression for regression analysis methods [5,22].

These methods can be equivalent to some simple forms of selected artificial neural network topologies. On the one hand, it is well known that logistic regression is based on the assumption of normality of the class distribution and homogeneity of the covariance matrix. Linear regression, on the other hand, requires a linear relationship between output and input variables and assumes a Gaussian perturbation of the output variables. When these assumptions are not met, which is unfortunately a common case, neural networks give better predictions compared to classical regression analysis [2,23,24].

This fact makes it possible to consider artificial neural networks as a more excellent and complementary stochastic tool with a wide range of applications. It is worth emphasising that they are useful, among other things, for analysing empirical data, both numerical

(numerical) and linguistic (nominal). This also means that artificial neural networks are a versatile instrument, characterised by a large spectrum of potential applications of a utilitarian nature and significant ease during their practical operation. Due to their low-complexity structure and ease of learning, as well as their large range of applications, networks with a linear transition function (activation function) are among those frequently used in engineering practice.

However, it is important to note that the aforementioned neural network models can only solve tasks that involve finding a linear (i.e., representable by a non-personal transformation matrix) reproduction of a set of input signals to a set of output signals.

An exemplary structure of a linear neural network is shown in Figure 1.

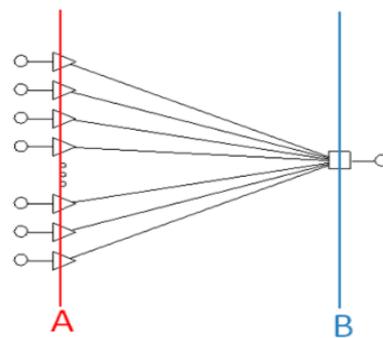


Figure 1. Structure of a linear network with single output. A—input layer and B—output layer.

Boniecki et al., 2009 [25] conducted a neural network analysis of digital images to identify the degree of maturity of compost produced from sewage sludge. In [21], the authors proposed a methodology for creating the learning sets necessary to generate artificial neural networks (ANN) for analysing graphically encoded information on the quality of pork half carcasses. Publication [22] deals with the determination of a selected family of butterflies presented in digital images by means of a generated ANN classifier, and an additional study [24] describes the application of neural network image analysis methods in quality assessment of greenhouse tomatoes using linear ANNs.

2.2. Nonlinear Layered Networks Trained with Supervision as a Subclass of Nonlinear Regression Models

The concept of regression in mathematical statistics is primarily associated with the process of empirically determining a stochastic relationship between correlated variables, where one most often attempts to model the problem using linear regression. However, often, the enormous complexity of the empirical systems under study does not allow such a significant simplification, and then it is necessary to build a nonlinear regression model [26].

In general, to solve a regression problem using a neural network model, the primary objective is to estimate the unknown value of the continuous output variable when the values of the input signals are known. Regression problems can be solved using different types of neural networks. The most commonly used network topologies are: multilayer perceptron (MLP); radial basis function (RBF), i.e., radial neural networks; generalised regression neural network (GRNN), i.e., networks implementing generalised regression; and, in special cases, linear neural networks.

In regression problems, the purpose of an artificial neural network is to acquire (during the learning process) the ability to translate data representing input variables into continuous output variables, which are then the response of the network in its exploitation phase (after the learning process).

Unidirectional multilayer perceptron-type networks, i.e., MLP networks, are among the best studied and most widely used network topologies in practice. A multilayer perceptron network represents the so-called parametric class of neural network models (the number of neurons constituting its structure is significantly smaller than the size of the learning set).

The basic properties of MLP networks include the following:

- MLPs are one-way networks;
- MLPs are trained using a supervised technique (i.e., algorithms modify weights and threshold values using learning sets containing both input and set output values);
- An MLP network has a multilayer architecture, i.e., there is an input layer, a hidden layer and an output layer;
- Connections only allow communication between neurons in adjacent layers;
- Neurons in the network aggregate the input data by determining the weighted sum of the inputs (using a linear aggregation formula);
- The activation function of the input neurons is linear, that of the hidden neurons is nonlinear, and that of the output neurons, in general, is nonlinear;
- Due to the saturation level in sigmoidal activation functions, the data processed by the network require appropriate rescaling (so-called pre- and post-processing of the data).

MLP-type networks can be trained by using a supervised technique (based on learner, validation, and test sets) following several efficient iterative methods. Training techniques have their roots in both classical optimisation methods and heuristic methods. The choice of algorithm depends on the type and nature of the investigated problem.

An illustrative structure of a nonlinear neural network using MLP as an example is shown in Figure 2.

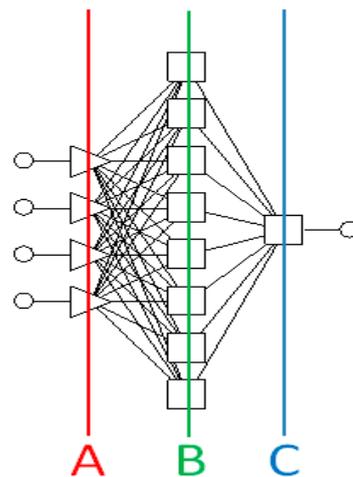


Figure 2. Structure of a multilayer perceptron (MLP) network with 1 input layer, 1 output layer, and 1 hidden layer: A—input layer; B—hidden layer; C—output layer.

Specialised neural networks of the GRNN type provide a neural representation of a statistical function approximation method. These networks have exactly four layers [24,27]. They consist of an input layer, a radial layer (storing the so-called centres), a regression layer, and an output layer. The neurons of the radial layer represent the cluster centres present in the learning data that are generally derived from an experiment. An appropriate clustering procedure such as the resampling algorithms k-means or Kohonen's is used to teach this layer. This allows the number of neurons in the radial layer to be smaller than the number of learning instances. The regression layer must have exactly one more neuron than the output layer and it is made of linear neurons. This layer contains two types of neurons. Type A neurons calculate the conditional regression for each output variable and a single type B neuron is used to calculate the probability density. It is known from practice that regression networks of the GRNN type learn in a very short time, but unfortunately tend to reach large sizes. This fact makes their performance during normal operation rather slow.

Hinton and Osindero, 2006 [27] presented a novel fast-learning algorithm dedicated to deep neural networks for the identification of graphical information used mostly in agriculture. Moody and Darkin, 1988 [26], in turn, dealt with fast learning implemented

in adaptive ANNs built from locally tuned processing units. Wawrzyniak et al., 2023 [28] described a neural network model that was used to study the potential of biomethane from cattle manure. Boniecki et al., 2021 [29] compared the affectivity of MLP and RBF neural network models on graphical classification and qualitative identification of compost. Pentoś et al., 2022 [30] used an ANN and MLR to investigate the relationship between electrical and selected soil mechanical properties. Construction of MLP ANN and multiple regression (MLR) models for tuber yield prediction of three very early potato varieties Arielle, Riviera, and Viviana was the aim of the work by Piekutowska et al., 2021 [31]. Sabzi-Nojadeh et al., 2021 [32] conducted a study to compare the performance of an ANN and MLR in predicting essential oil yield (EOY%) and trans-anethole yield (TAY%) of fennel populations. In contrast, Gorzelany et al., 2022 [33] studied the selected mechanical properties of fresh and stored large fruit cranberry fruit using an ANN and MLR. An example of a multilayer perceptron neural network (MLP-NN) designed and applied to model the internal temperature and relative humidity in a greenhouse was included in work by [34]. For the specific NN backpropagation as a training algorithm, the input variables were the external temperature and relative humidity, wind speed, solar irradiance, as well as the internal temperature and relative humidity, up to three timesteps before the modelled time step. The maximum errors of the modelled temperature and relative humidity were 0.877 K and 2.838%, respectively, whereas the coefficients of determination were 0.999 for both parameters. The intention of the authors was to provide the right parameters for the greenhouse control system.

Crop diseases are critical factors that hamper agricultural development, severely affecting yields and crop security. The existing crop disease identification models make it difficult to focus on a disease area. In addition, crops with similar disease traits are easily misidentified [35–37]. To address the above problems, accurate and efficient disease identification models have been proposed that incorporate local and global image features into the trait analysis and improve the separation between similar diseases. Frequently, the traditional methods of agricultural pest detection cannot satisfy the needs of agricultural production because of low efficiency and accuracy. In their work, Yao et al., 2017 [38] proposed a method for classification of high-resolution agricultural remote sensing images based on convolution neural networks (CNNs). By training and testing the CNNs (with a large number of high-resolution images), the crop classification achieved a rate of 99.66% after optimising the network parameters. Li et al., 2022 [39] proposed a model that applied convolutional neural networks for pest identification based on transfer learning. A dataset of agricultural pest images was adopted as an experimental dataset. The effect of data augmentation on the classification performance of different samples was compared. The results showed that the classification effect of the model based on transfer learning was generally better than that based on new learning. Compared to new learning, transfer learning significantly improved the recognition ability of the model and significantly reduced the training time to achieve the same classification accuracy. In [40], an anchor-free region convolutional neural network (AF-RCNN) for precision recognition and classification of 24 classes of pests was proposed. Interestingly, a running time of 0.07 s per image was achieved to perform real-time detection. In addition, a transformer encoder was introduced in the model as a convolution operation, and therefore, the model could establish the relationship between long-distance traits and extract global disease image traits, and centre loss was applied as a penalty term to expand the inter-class difference of crop disease traits and reduce their intra-class gap. An identification accuracy of more than 96% was achieved, even with a complex background [41].

2.3. Discriminant Analysis with Kernel versus Normalised Neural Networks with Radial Symmetry

A discriminant function analysis is most commonly used to obtain information on deciding which variables best divide an empirical dataset into naturally occurring groups. The main idea behind a discriminant function analysis is to decide whether the groups

differ due to the mean of a certain variable, and then use this variable to predict group membership (e.g., new cases).

Networks with radial basis functions are examples of neural networks characterised by circular symmetry. In 1988, the radial basis function (RBF) network was proposed by Broomhead and Lowe, 1988 [42] and, independently, in 1989 by Moody and Darkin [26]. RBF-type radial networks implement a distinctive way of transforming an input set into an output set. Such a representation involves fitting an approximating function of multiple variables to the desired values. Thus, it involves spanning over the learning set and fitting a multidimensional hypersurface to the desired values.

RBF networks usually require more neurons than perceptron-type networks, but they are much faster to learn. They typically consist of three layers: an input layer (usually linear), a hidden layer with neurons with radial activation functions, and an output layer containing linear neurons. The basic properties of RBF-type networks include the following:

- RBFs are unidirectional networks;
- RBFs are trained using a hybrid technique, i.e., the hidden radial layer is trained using a nonsupervised method (weights and threshold values are modified using learning sets containing only input values, for these algorithms, output values in the dataset are not required and are ignored if they occur), while the linear output layer is trained using a supervised technique, for example, using a pseudo-inversion method;
- RBFs have a three-layer architecture, i.e., there is an input layer, a hidden (radial) layer, and an output layer;
- Connections only allow communication between neurons in adjacent layers;
- The activation function of the input neurons is linear and that of the hidden neurons is nonlinear (radial), and that of the output neurons is fully linear.

RBF networks learn relatively quickly and have the advantage of never extrapolating functions over too large a distance from known data. These networks can have definite advantages, especially when modelling (with them) physical phenomena known to have circular symmetry. However, they are generally much larger than MLP networks solving the same tasks, making them more time-consuming to run on a computer simulating the network. However, these networks can have definite advantages when the network is implemented as a dedicated electronic circuit.

Despite the similarity, there are important differences between the two types of networks:

- A radial RBF-type network has a determined structure (with one radial hidden layer and a linear output layer), while an MLP-type network can have a different number of hidden layers and the output neurons can be either linear or nonlinear;
- When radial functions are used, there is more variation in the choice of their shape.

The most common equivalent of the sigmoid function is the basis function.

The structure of a nonlinear neural network is shown in Figure 3 using the RBF model as an example.

Boniecki et al., 2012 [11] described a neural network image analysis process for estimating the aerobic and anaerobic decomposition of organic matter using the example of straw degradation. Piekarska-Boniecka et al. (2008) [43] identified parasitic wasps of the subfamily pimpliniae (Hymenoptera, ichneumonidae) using RBF-type ANNs. The authors of [44] addressed the problem of dimension reduction of digital image descriptors in the process of neural network identification of damaged grains of malting barley. In [45], RBF-type neural networks were analysed as a dedicated tool for the study of selected empirical systems derived from agricultural engineering. In the study by [46], the effectiveness of multiple regression techniques combined with different types of artificial neural networks (ANNs) including RBF were investigated using regression results as input variables to estimate sunflower grain yield under normal and salinity conditions. A very interesting group of studies are those that predict the possible quality of water. In the study by Hong et al., 2020 [47], a radial basis function artificial neural network (RBF-ANN) as well as a hybrid method of RBF-ANN and grey relational analysis (GRA)

were proposed to predict trihalomethane (THM) levels in drinking water. The results indicated the high capability of RBF-ANN to learn the complex nonlinear relationships involved in THM formation (regression coefficients ($r_p = 0.760\text{--}0.925$) and prediction accuracy ($N_{25} = 92\text{--}98\%$). It was found that RBF-ANN using fewer water quality parameters based on GRA achieved excellent performance in THM prediction ($r_p = 0.760\text{--}0.946$ and $N_{25} = 92\text{--}98\%$). In the work by Deng et al., 2021 [48], occurrence of halo ketones (including dichloropropanone, trichloropropanone, and total HKs) was studied with means of linear, log-linear regression models, back propagation (BP), as well as radial basis function (RBF) artificial neural networks (ANNs) in real water supply systems. The results showed that the overall prediction ability of RBF and BP ANNs was better than linear/log-linear models. Though the BP ANN showed excellent prediction performance in internal validation ($N_{25} = 98\text{--}100\%$, $R^2 = 0.99\text{--}1.00$), it could not effectively predict HK occurrence in external validation ($N_{25} = 62\text{--}69\%$, $R^2 = 0.202\text{--}0.848$). The prediction ability of RBF-ANN in external validation ($N_{25} = 85\%$, $R^2 = 0.692\text{--}0.909$) was quite good, which was comparable to that in internal validation ($N_{25} = 74\text{--}88\%$, $R^2 = 0.799\text{--}0.870$).

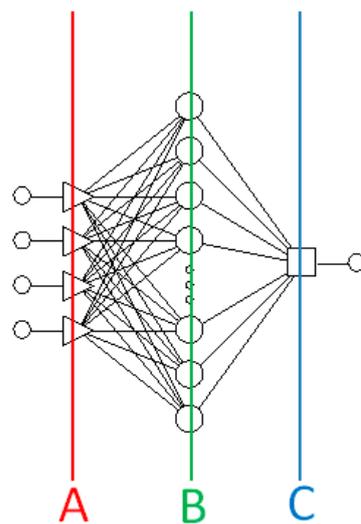


Figure 3. Structure of a radial basis function (RBF) with 1 input layer, 1 output layer, and 1 radial hidden layer: A—input layer, B—hidden layer, C—output layer.

2.4. Principal Component Analysis Method vs. Unsupervised Neural Networks

Principal components analysis (PCA) is a linear dimension reduction technique to identify mutually orthogonal axes (dimensions) with the greatest variance within the original empirical data. This technique realises a projection of the data into a space of fewer dimensions, formed from a subset of the components with the greatest variance. Therefore, principal component analysis allows the orthogonal directions of the new coordinate system to be determined in such a way that they correspond to the directions of maximum variance of the original data. Dimension reduction is achieved by projecting the data from the original signal space into the aforementioned new space characterised by a lower dimension and formed by the subset of components with the greatest variance.

The PCA-based learning algorithm implements a linear transformation by rotating the data included in the learning set to a new coordinate system, formed by the eigenvectors of the autocorrelation matrix determined for the learning data. The eigenvalues, corresponding to each eigenvector, determine how much of the “variability” present in the empirical data is represented by the corresponding eigenvectors. The PCA method often makes it possible to solve problems that would otherwise simply have too many variables to build an effective neural network. Principal component analysis can be easily implemented in an artificial neural network optimised in nonsupervised mode (trained using Hebb’s rule) [8]. It transforms the input cases of the network (which are points in a “larger” space to a “smaller” space) in such a way that the variability in the resulting

dataset is maximised. A special group of neural networks with Hebbian learning are auto-associative networks, which use the aforementioned algorithm in the learning process. This means that the auto-associative network, during the learning process, acquires the ability to reduce the dimension of the input data (thus performing a type of data compression). By definition, auto-associative neural networks are those that reproduce input values at their outputs. Such an operation is justified because the considered auto-associative network has a significantly smaller number of neurons in the middle (hidden) layer than in the input (and output) layer. During the process of feeding the learning vector into the ANN, the data must pass through a bottleneck between the input and output. In order to fulfil its task, which in this case is to reproduce the input information in the output (as defined by an auto-associative network), the network must first learn to represent the input data with a smaller number of signals generated by the neurons of the hidden layer.

The neural network can only master the ability to reconstruct the full input data from the previously compressed information encoded inside (in the hidden layer) of the network in the next stage. This means that the auto-associative neural network, during the learning process, acquires the ability to reduce the dimension of the input data packed into the hidden (middle) layer.

In general, an auto-associative neural network designed for data compression should consist of a minimum of three layers:

- An input layer (with the number of neurons corresponding to the input number of data);
- An output layer (of the same size);
- A hidden layer (with a much smaller number of neurons).

An auto-associative neural network should be trained in such a way that it accurately reproduces the input data on its outputs. This is why it has exactly the same number of inputs as outputs, and the variables used to teach it are assigned a particular character of so-called input/output variables. As indicated above, the idea behind the operation of such a network is that the number of hidden neurons is much smaller than the number of inputs or outputs, effectively forcing the information to be "squeezed" through a representation of a smaller (compressed) dimension. An auto-associative (three-layer) network constructed as above performs a transformation of the input data to the hidden layer (of reduced dimension), and then another transformation back to the output layer. It can be shown that when the neurons used in the hidden layer and in the output layer have linear (or quasi-linear) characteristics, such a network actually learns to approximate the standard PCA algorithm. Thus, in the present case, it is a substitute for the linear PCA method presented above [49].

One substantial problem with the PCA method is its linear nature. For this reason, it cannot be used to reduce the dimension of the input data in any case. It can only be used to identify exclusively linear transformations that optimise the condensation of the information contained in the considered variables, based on the directional search for the maximum variance.

An alternative approach, free of the limitation indicated above, is to use a particular auto-associative neural network topology that realises a nonlinear version of PCA. The idea of realising a nonlinear dimension reduction is, in fact, to use an analogous neural network topology, but built with nonlinear neurons, i.e., having in their structure a nonlinear membrane potential of the postsynaptic function. To fully exploit the possibilities offered by a nonlinear network, however, more than one layer of neurons is needed for each of the two transformations implemented (for both compression and decompression). For this reason, when creating a network for nonlinear data compression, a neural network topology with five layers should be generated [17,50–52].

The hidden middle layer is the layer that reduces the dimension of the input signal, while the layer between it and the input layer performs precisely the required nonlinear compression of the input data. The corresponding two-layer network structure, located between the hidden layer and the output layer, performs the inverse transformation by

decompressing the previously compressed signal. Figure 4 illustrates the structure of a nonlinear neural network based on the example of a PCA model.

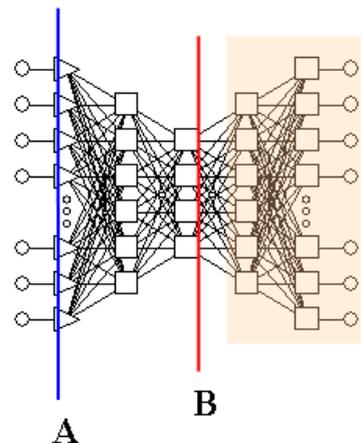


Figure 4. A neural auto-associative network implementing a nonlinear PCA transformation: A—input layer of the 5-layer auto-associative network; B—hidden layer of the 5-layer auto-associative network (compressing in a non-linear manner).

Studies by [44,52] presented a method using neural network reduction of graphical descriptors to determine the quality and extent of mechanical damage in malting barley acquired from digital images. A novel approach for analysing a graphical image was developed to map disturbances and to evaluate the ecological status of mangroves based on multispectral WorldView-2 and Sentinel-2 and remote sensing techniques [53]. The spatial disturbance index (SDI) was calculated based on landscape metrics and principal component analysis (PCA) in order to detect degraded mangroves. The first three components of the PCA explained in total 87% (47%, 27%, and 13%, respectively) of the variance of information distribution, demonstrating PCA effectiveness.

Bourland and Kamp, 1988 [49] demonstrated an example of an auto-associative process using MLP-type ANNs and matrix singular value decomposition techniques. The authors of [54] dealt with nonlinear principal component analysis using selected auto-associative neural network topologies.

Krevh et al., 2023 [55] examined long-term data on soil water regime and nitrate dynamics to better understand the implications of N management in relation to groundwater pollution. The principal component analysis (PCA) method was performed to identify the relationships among all soil properties and environmental characteristics. The results showed the complex interaction of soil hydraulic properties, precipitation patterns, plant uptake, and N application.

A vegetation ecosystem model was constructed to provide a simulation method for improving water use efficiency capacity as well as for predicting the future response of water use efficiency (WUE) to climate change [56]. The structural relationship and degree of influence among factors was determined by using a structural equation model (SEM) that was transformed into ANN topology, where the PCA method was employed to reduce spatial dimensionality. The results showed that different influencing factors on WUE presented a diversity with different levels. The ANN structure optimised by using SEM fit better, and the PCA-SEM-ANN model had a very high explanatory and precision for environmental control of the ecosystem as well as WUE simulation.

2.5. Cluster Analysis by K-Means vs. Topological Kohonen Maps

A k-means cluster analysis is based on a statistical procedure that has been known for a relatively long time. It is an algorithm for determining k centers that optimally represent the structure of N points ($k < N$). In the initial phase of the algorithm, randomly selected cases from the empirical dataset are taken as centres. Each case is then assigned to the

cluster represented by the centre nearest the case. Once all cases have been assigned, new centres are determined as the focus of each cluster. The process of assigning cases to classes is repeated until equilibrium is reached.

With reference to neural network models, the k-means algorithm determines radial centres, which are stored in radial neurons located in the first hidden layer of the network. The k-means method assigns each learning event to one of the k clusters (where k is the number of radial neurons). Each of the assigned clusters is represented by the centre of gravity of the cases belonging to it, and the distance of each case from the centre of gravity of the cluster assigned to it is smaller than the distance between that case and the centre of gravity of any other cluster (this is the basic criterion for including a particular case in a particular cluster). The centres of gravity of the clusters are copied to the radial neurons.

An important group of unsupervised networks trained with methods based on the k-means algorithm are artificial neural networks also known as Kohonen maps. They have a two-layer structure (without hidden layers), i.e., they only have an input layer and an output layer made of radial neurons. The learning algorithm proposed by TuevoKohonen is, in fact, an adapted and modified (for artificial neural networks) method of the well-known k-means method. This algorithm determines cluster centres for the output radial layer, which is usually a two-dimensional topological map. During its operation, the input pattern is repeatedly presented and the weights of the winner neuron and the bespoke neurons immediately adjacent to the winner are modified. The correction carried out is intended to make the "tuned" weights similar to the input pattern. Thus, the aim of learning with the Kohonen method is to identify a set of cluster centres that best reflect the actual distribution of learning cases. As a result of learning, the neural network acquires the ability to activate neighbouring neurons (when given similar input data) which results in a set of output signals from the network that can be interpreted as a map representing the topological relationships between the input data in the space of weights and inputs. The neighbourhood plays a key role in the learning of Kohonen networks. By modifying the neurons surrounding the winning neuron, rather than just the neuron itself, similar data will be close in the topological map. The neighbourhood size is reduced as the learning process progresses, and the learning rate is similarly reduced. Thus, at the beginning of learning, a coarse mapping is obtained, with large clusters of neurons corresponding to similar cases. Later, smaller details of the map are obtained, as individual neurons in the clusters begin to respond to more subtle differences between similar cases [21,57].

Kohonen-type neural networks are typically used in the classification process. They are also sometimes used to generate signals useful for vector coding of input data. The "output" of the network is then a vector of weights of the winning neuron of the output layer of the Kohonen network, representing a vector code, suitable, for example, for signal compression using a so-called codebook. Figure 5 shows an example of a Kohonen-type neural network structure.

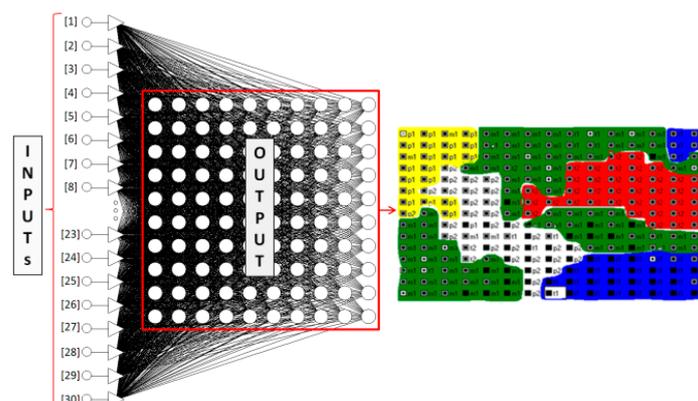


Figure 5. An example of the structure of a Kohonen-type neural network applied as a colour (classifier) separator.

The Kohonen self-organising map neural network (KSOM-NN) has attracted significant attention from researchers because of the quality of data visualisation and interpretation [18]. In [58], a comprehensive review is presented of the usefulness of the Kohonen neural network model for solving selected agricultural classification problems. Some applications of the cluster analysis by using k-means or Kohonen maps are described below. In [59], it was found that classical machine learning methods such as K-nearest neighbours could achieve good classification of agricultural images collected using Sentinel-1 radar. The accuracy was as high as 86%.

In [13], the application of a Kohonen-type neural network was demonstrated in the process of nonparametric qualitative classification of tomatoes. In turn, [18,60] performed the classification of selected orchard pests using a neural network of the self-organising feature maps (SOFM) type. The same neural network topology was used in [61] for neural network classification of compost maturity levels. A Kohonen's neural network map was used to construct a multi-objective genetic local search algorithm. In the new algorithm, for each generation, a population of neurons was trained using elite solutions of the genetic algorithms in the current population. The training rule of neurons was developed using the concepts of the learning rule of self-organising map and variable neighborhood search algorithm to improve the local and global search [62].

2.6. Bayesian Statistics versus Probabilistic Neural Networks

Techniques for estimating probability density functions from extracted data have a long history in statistics. Bayesian estimation is a well-known method of statistical analysis based on Bayes theorem, which is the foundation of probability theory [50,51,63]. This theorem states that the a posteriori probability of a parameter "p" is proportional to the a priori probability "p" multiplied by the probability "p" determined from the empirical data. Bayesian statistics can be applied for the probability density function estimation of model parameters using available data, and as a result of this analysis, the model is selected whose parameters maximise the aforementioned probability density function, thus reducing the global error.

In recent years, an approach to probability density function estimation based on so-called kernel approximation has become important. In kernel approximation, simple functions (e.g., a Gaussian function) are located at the point of occurrence of each available case, and then these are added to obtain an estimator of the joint probability density function. The operation of the aforementioned method can be described in that the presence of a case at a point in the input space implies a high probability density at that point. Therefore, the clustering of cases that are close to each other indicates an area of high density. If enough empirical data are available, a relatively good approximation of the true probability density function is obtained [8,64].

The kernel approach to approximating probability density functions is very similar to the use of neural networks with radial basis functions. This inspired the creation of a new category of neural network models called probabilistic neural networks (PNNs). Probabilistic neural networks, which are a subset of Bayesian networks, are essentially implementation of the so-called kernel function approximation method and are used exclusively for classification. They typically have three or four layers. These distinguish between an input layer, a layer of radial neurons, and a layer of linear classifier neurons. Optionally, a fourth layer, which is also linear and contains a squared cost matrix, can be included. Since the attached cost matrix is squared, the third and fourth layers must have the same number of neurons. The basic properties of PNN-type networks include the following:

- PNNs are unidirectional networks;
- PNNs are trained using a nonsupervised technique, i.e., weights and threshold values are modified using learning sets containing only input values (for these algorithms, output values in the dataset are not required and are ignored if they occur);

- PNNs have a three or four-layer architecture, distinguishing between an input layer, a hidden (radial) layer, and a linear output classifier layer (optionally, a fourth layer may be included, which is also linear and contains a quadratic cost matrix and has the same number of neurons as the third layer),
- Connections only allow communication between neurons located in neighbouring layers;
- The activation function of the input neurons is linear, that of the hidden neurons is nonlinear (radial), and that of the output neurons is fully linear.

Among the greatest advantages of PNNs is the generation of probability values (or, more precisely, their estimates) at the outputs, rather than just the “raw decisions” themselves, which greatly facilitates the evaluation and interpretation of the results. Another advantage of PNN-type networks is the high speed of their learning. The biggest disadvantage of PNN-type networks is their size, due to the fact that the structure of networks of this type must contain neurons corresponding to the individual examples considered, which means that the entire learning set is mapped in the structure of a PNN-type neural network. This causes, among other things, very high memory requirements for the creation and operation of such networks and this is the reason why the time required to run such a network is rather long. PNN-type neural networks are particularly useful during experiments aimed at defining network prototypes (e.g., when decisions are made regarding the choice of input variables), since the short learning time of these networks makes it possible to perform a large number of tests (e.g., with different input datasets) in a short period of time. The structure of a nonlinear neural network using the PNN model as an example is shown in Figure 6.

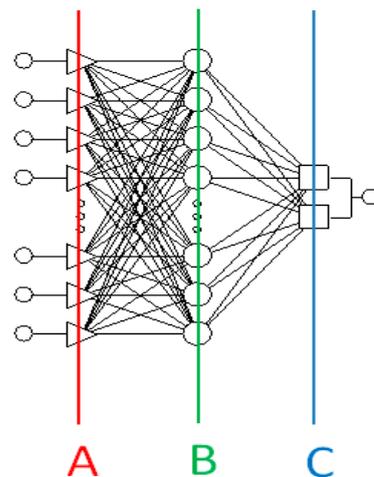


Figure 6. Probabilistic neural network (PNN) with 1 input layer, 1 output layer, and 1 radial hidden layer: A—input layer, number of neurons equal to the number of features; B—hidden layer built of radial neurons also called the pattern layer, the number of neurons is equal to the number of learning instances; C—output layer built of linear classifying neurons (summation layer), the number of neurons is equal to the number of classes.

Examples of application of Bayesian statistics and probabilistic neural networks are described below. First, the newest tutorial by Krause and Bokinala demonstrated how Bayesian networks can be built from a wide range of sources of expert knowledge and data [65]. Drury et al., 2017 [66] applied Bayesian networks in agricultural examples. Studies by [51] and [67] provided a comprehensive compendium of knowledge on probabilistic neural networks. In [68], Sujak et al., 2017 conducted an environmental bioindication study using a Bayesian network implementation with the grey heron as a model species. The work was aimed at identifying the quality of the agricultural environment. Interesting applications of Bayesian networks include their use to model agricultural water demand, groundwater intake, surface water supply, and climate which show complex, nonlinear relationships with groundwater storage in agricultural regions. A novel machine

learning-based approach to model a groundwater ensemble has been used in combination with a Bayesian model averaging approach to predict groundwater storage. The results showed that the machine learning model had remarkable predictive ability without loss of accuracy, but with higher computational efficiency [69]. An exemplary application was the development of a Bayesian network (BN) model to predict nitrogen loading to surface water at the farm level, using specific site characteristics such as landscape, soil, cropping system, and to compare different conventional and conservation agricultural practices. A sensitivity analysis showed the significance of weather factors on surface runoff and topographic information for soil erosion, while agricultural treatments were found to be less important. Bayesian networks have been shown to be a useful and flexible tool for data and knowledge assimilation and a practical approach for testing and comparing the effects of different agricultural interventions on agricultural system emissions [70].

Wu et al., 2023 [71] proposed a novel agricultural drought prediction model for long lead times by integrating vine copulas with Bayesian model averaging (hereafter, the BVC model) which improved agricultural drought management, food security assessment, and early drought warning. An interesting application of Bayesian networks was the project by Lachaud et al., 2022 [72] in which Bayesian networks were applied to study food security. There are many other applications of Bayesian networks, and it is impossible to mention it all in one single publication.

2.7. Analysis of Time Series and Their Neural Network Representation

A time series is used to refer to a set of data values (most often empirical) that is in a determined order over time. A time series data analysis implies the study of sequences of measurement data that are characterised by a non-random (determined) ordering. Unlike classical analyses performed on random samples, a time series analysis is based on the assumption that successive values in a dataset represent successive measurements, taken at equal time intervals [8].

A time series analysis aims to detect the nature of a phenomenon represented by a sequence of realised observations and to forecast the future values of an identified time series. The above-mentioned priorities require identifying and then describing the characteristic elements of the time series in question. In the course of a time series analysis, it is most often the case that subsequent values of a series are projected on the basis of previous values of the same variable or on the basis of values of other variables. Regardless of the accuracy of the theoretical justification of the form of the generated model, it is always possible to predict future values of a time series based on extrapolation methods.

Artificial neural networks are well suited to analysis time series predictions. Usually, this is because a continuous variable is being predicted; therefore, time series prediction is essentially a specialised form of regression. Typically, the next value of a time series is forecast based on a certain number of preceding values. In this case, the aim is to determine a value that is “one step ahead” in time. However, there is nothing to prevent the process of long-term forecasting from being considered in a similar way. It should be noted that any type of artificial neural network can be used for time series forecasting. However, the choice of network topology (MLP, RBF, PNN, GRNN, Kohonen network, etc.) must be adequate to realise the appropriate form of input signal processing. This form is determined by the type and nature of the problem studied. Indeed, first, it must be decided whether it is a regression or classification problem. Then, the selected neural network is adapted for time series forecasting by setting values for characteristic parameters such as “row” and “forecast horizon”. The “row” parameter specifies how many instances (observations of previous values of the variable under consideration) must be introduced into the inputs of the network so that, based on them, the network can determine the output value (the projected next value of this variable). The “forecast horizon” parameter, in turn, tells how far away the value of the variable to be predicted (forecasted) is from the end of the series of retrospective observations given at the input.

Time series analyses have broad applications. Examples of using time series with neural network representation applications are described below. Bishop (1995) [8] presented an overview of applications of time series interpreted in the form of artificial neural networks. Boniecki and Weres, 2001 [45] demonstrated selected applications of neural network time series for empirical modelling of processes occurring in agricultural systems. In turn, [73] described techniques for forecasting yields of selected agricultural crops using a neural network time series model.

An analysis of the literature has shown that, compared with traditional models, deep neural networks can enhance data structure mining and overall information simulation capabilities through innovative and efficient structures. This means that it is also possible to extend the range of environmental parameter selection for agricultural facilities and to achieve environmental prediction end-to-end optimisation through an intelligent time series model based on deep neural networks [74].

One of the exemplary studies aimed at analyzing time-series trends and variability analysis of observed rainfall and temperature records is that of Tofu et al. (2023) [75]. The study focused on smallholder farmers' perceptions, including the analysis of their response strategies to both observed and perceived climate variability and its determinants. The impacts of abnormal trends or shifts in observed rainfall and temperature patterns, along with socioeconomic, technological, and behavioural factors, call for policymakers to set strategies that will enable smallholder farmers to improve their livelihoods, ensure food security, and build resilience in the face of climate extremes.

3. Conclusions and Recommendations

The growing interest in artificial neural networks is not the result of chance but due to the need to improve data analysis methods by making them faster and more automated. The primary distinguishing factor of neural network models, compared to symbolic machine learning methods (e.g., rules or decision trees), is the completely different manner in which knowledge is acquired by the system during the learning process. This post-symbolic representation is a direct consequence of the applied method of computation and the adopted topology of the neural network model.

Neural network models have simply proven to be convenient as well as effective instruments that are useful for a wide variety of practical tasks. In fact, they have been successfully applied to solve all sorts of problems, not only in agriculture but also in fields as diverse as finance, medicine, engineering in the broadest sense, geology, and physics. In fact, there may be many more applications, since artificial neural networks can be used wherever problems arise in data processing and analysis (also in the form of multidimensional random variables), prediction, classification, or control. They should also include so-called unstructured types of problems, i.e., problems that are difficult or even impossible to algorithmise, or for which there is insufficient scientific knowledge or limited available empirical data.

Neural networks are a sophisticated modelling technique capable of representing extremely complex functions. In particular, ANNs are nonlinear in nature, which significantly enriches their application possibilities, and therefore gives them an advantage over commonly used statistical methods. Neural networks can also control the complex problem of multidimensionality, which, when using other methods (e.g., statistical methods), significantly hinders the process of modelling nonlinear functions with a large number of independent variables. It is worth highlighting that calculations based on neural network analyses can be carried out on computers with a neuroprocessor central unit that performs its tasks in a significantly different way from currently used microprocessors, in which processing is carried out according to the concept proposed in the 1940s (previous century) by John von Neuman (cyclic machine). The neuroprocessor was inspired by the results of many years of neuroscientific research on the brain. For technical reasons, numerical simulators are currently most commonly used instead of hardware neuroprocessors, occurring in the form of the artificial neural networks discussed above.

These applications have emerged and have been developed as a result of work carried out in the field of distributed processing techniques (in particular parallel information processing) and research in the area of artificial intelligence. Scientific work that has concerned the construction of models of the basic neural structures found in the brain has been of fundamental importance.

The most important and desirable characteristics of neural network models include the following:

- High processing power;
- Reliability and resistance to interruptions;
- Ease of use;
- Simple structure;
- Ability to generalise the acquired knowledge;
- Biological inspiration.

The creation of neural network models (as well as stochastic models) requires an adequate set of empirical data, which is needed to construct the training, validation, and test sets that are the basis for creating the weights matrix, which is the learning set structure necessary for generating ANNs. For numerical reasons, it is important that the number of learning vectors (experimental cases) is as large as possible. It is standard practice to have 10 learning cases per variable. The structure of the learning set is generally determined by the procedures implemented in the individual ANN software simulators such as Statistica and MatLab. This also applies to the methods of validation and verification of the generated neural network models.

The studies in the literature described in this paper indicate the usefulness of the developed neural network models, among others, as important supports for decision-making processes occurring during agricultural production (broadly defined). In particular, the generated ANNs can be implemented in modern information systems to support a variety of processes in agricultural systems.

Author Contributions: Conceptualization, P.B., A.S. and G.N.; Investigation P.B., A.S., G.N., H.P.-B., A.P. and A.W.; Resources, P.B., A.S., A.W. and G.N.; Writing—original draft preparation, P.B., A.S. and G.N.; Writing—review and editing, P.B., A.S. and G.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: This study includes no data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Méndez, L.J.; Lira, A.; Lasa, R.; Cerdeira, S. Delineation of site-specific management zones for pest control purposes: Exploring precision agriculture and species distribution modeling approaches. *Comput. Electron. Agric.* **2019**, *167*, 105101. [[CrossRef](#)]
2. Haan, N.L.; Zhang, Y.; Landis, D.A. Predicting Landscape Configuration Effects on Agricultural Pest Suppression. *Trends Ecol. Evol.* **2020**, *35*, 175–186. [[CrossRef](#)] [[PubMed](#)]
3. Castañeda, A.; Castaño, V.M. Smart frost measurement for anti-disaster intelligent control in greenhouses via embedding IoT and hybrid AI methods. *Measurement* **2020**, *164*, 108043. [[CrossRef](#)]
4. Nowakowski, K.; Raba, B.; Tomczak, R.J.; Boniecki, P.; Kujawa, S.; Nowak, P.J.; Matz, R. Identification of Physical Parameters of Cereal Grain using Computer image Analysis and Neural Models. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2013), Beijing, China, 21–22 April 2013. [[CrossRef](#)]
5. Badgujar, C.; Das, S.; Figueroa, D.M.; Flippo, D. Application of Computational Intelligence Methods in Agricultural Soil–Machine Interaction: A Review. *Agriculture* **2023**, *13*, 357. [[CrossRef](#)]
6. Kujawa, S.; Niedbała, G. Artificial Neural Networks in Agriculture. *Agriculture* **2021**, *11*, 497. [[CrossRef](#)]
7. Fausett, L. *Fundamentals of Neural Networks*; Prentice Hall: New York, NY, USA, 1994.
8. Bishop, C. *Neural Networks for Pattern Recognition*; University Press: Oxford, UK, 1995.

9. Boniecki, P.; Nowakowski, K.; Tomczak, R.Ł. Neural networks type MLP in the process of identification chosen varieties of maize. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2011), Chengdu, China, 15–17 April 2011. [[CrossRef](#)]
10. Boniecki, P.; Koszela, K.; Piekarska-Boniecka, H.; Nowakowski, K.; Przybył, J.; Zaborowicz, M.; Raba, B.; Dach, J. Identification of selected apple pests, based on selected graphical parameters. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2013), Beijing, China, 21–22 April 2013. [[CrossRef](#)]
11. Boniecki, P.; Nowakowski, K.; Ślósarz, P.; Dach, J.; Pilarski, K. Neural image analysis for estimating aerobic and anaerobic decomposition of organic matter based on the example of straw decomposition. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2012), Kuala Lumpur, Malaysia, 8–9 June 2012. [[CrossRef](#)]
12. Nowakowski, K.; Boniecki, P.; Dach, J. The identification of mechanical damages of kernels basis on neural image analysis. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2009), Bangkok, Thailand, 7–9 March 2009.
13. Boniecki, P.; Nowakowski, K.; Tomczak, R.J.; Kujawa, S.; Piekarska-Boniecka, H. The application of the Kohonen neural network in the non-parametric quality-based classification of tomatoes. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2012), Kuala Lumpur, Malaysia, 8–9 June 2012. [[CrossRef](#)]
14. Boniecki, P.; Piekarska-Boniecka, H.; Świerczyński, K.; Koszela, K.; Zaborowicz, M.; Przybył, J. Detection of the granary weevil based on X-ray images of damaged wheat kernels. *J. Stored Prod. Res.* **2014**, *56*, 38–42. [[CrossRef](#)]
15. Nowakowski, K.; Boniecki, P.; Tomczak, R.J. Identification process of corn and barley kernels damages using neural image analysis. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2011), Chengdu, China, 15–17 April 2011. [[CrossRef](#)]
16. Sujak, A.; Jakubas, D.; Kitowski, I.; Boniecki, P. Identification of Factors Affecting Environmental Contamination Represented by Post-Hatching Eggshells of a Common Colonial Waterbird with Usage of Artificial Neural Networks. *Sensors* **2022**, *22*, 3723. [[CrossRef](#)]
17. Świetlicka, I.; Sujak, A.; Muszyński, S.; Świetlicki, M. The application of artificial neural networks to the problem of reservoir classification and land use determination on the basis of water sediment composition. *Ecol. Indic.* **2017**, *72*, 759–765. [[CrossRef](#)]
18. Pilarski, K.; Boniecki, P.; Ślósarz, P.; Dach, J.; Piekarska-Boniecka, H.; Koszela, K. Classification of chosen orchard pests with using the SOFM neural network. *AJAR* **2012**, *7*, 6357–6362.
19. Deepa, S.; Alli, A.; Sheetac; Gokila, S. Machine learning regression model for material synthesis prices prediction in agriculture. *Mater. Today Proc.* **2021**, *in press*. [[CrossRef](#)]
20. Maes, W.H.; Steppe, K. Perspectives for Remote Sensing with Unmanned Aerial Vehicles in Precision Agriculture. *Trends Plant Sci.* **2019**, *24*, 152–164. [[CrossRef](#)] [[PubMed](#)]
21. Zaborowicz, M.; Foujd, A.; Boniecki, P.; Przybył, K.; Gierz, Ł.; Koszela, K.; Ślósarz, P.; Lisiak, D.; Przybył, J. Methodology of data processing in the process of neural image analysis of pork half carcasses. In Proceedings of the Tenth International Conference on Digital Image Processing (ICDIP 2018), Shanghai, China, 11–14 May 2018. [[CrossRef](#)]
22. Zaborowicz, M.; Boniecki, P.; Piekarska-Boniecka, H.; Koszela, K.; Mueller, W.; Górna, K.; Okoń, P. Neural classification of the selected family of butterflies. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2017), Hong Kong, China, 19–22 May 2017. [[CrossRef](#)]
23. Elbeltagi, A.; Nagy, A.; Mohammed, S.; Pande, C.B.; Kumar, M.; Bhat, S.A.; Zsembeli, J.; Huzsvai, L.; Tamás, J.; Kovács, E.; et al. Combination of Limited Meteorological Data for Predicting Reference Crop Evapotranspiration Using Artificial Neural Network Method. *Agronomy* **2022**, *12*, 516. [[CrossRef](#)]
24. Zaborowicz, M.; Boniecki, P.; Koszela, K.; Przybył, J. Application of neural image analysis in evaluating the quality of greenhouse tomatoes. *Sci. Hortic.* **2017**, *218*, 222–229. [[CrossRef](#)]
25. Boniecki, P.; Dach, J.; Nowakowski, K.; Jakubek, A. Neural image analysis of maturity stage during composting of sewage sludge. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2009), Bangkok, Thailand, 7–9 March 2009. [[CrossRef](#)]
26. Moody, J.; Darkin, C.J. Fast learning in networks of locally-tuned processing units. *Neural Comput.* **1989**, *1*, 281–294. [[CrossRef](#)]
27. Hinton, G.; Osindero, S.; Tech, Y.-W. A Fast Learning Algorithm for Deep Belief Nets. *Neural Comput.* **2006**, *18*, 1527–1554. [[CrossRef](#)]
28. Wawrzyniak, A.; Przybylak, A.; Sujak, A.; Boniecki, P. Neural Modelling in the Exploration of the Biomethane Potential from Cattle Manure: A Case Study on Herds Structure from Wielkopolskie, Podlaskie, and Mazowieckie Voivodeships in Poland. *Sensors* **2023**, *23*, 164. [[CrossRef](#)]
29. Boniecki, P.; Zaborowicz, M.; Sujak, A. Comparison of MLP and RBF neural models on the example graphical classification. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2021), Singapore, 20–21 June 2021. [[CrossRef](#)]
30. Pentoś, K.; Mbah, J.T.; Pieczarka, K.; Niedbała, G.; Wojciechowski, T. Evaluation of Multiple Linear Regression and Machine Learning Approaches to Predict Soil Compaction and Shear Stress Based on Electrical Parameters. *Appl. Sci.* **2022**, *12*, 8791. [[CrossRef](#)]
31. Piekutowska, M.; Niedbała, G.; Piskier, T.; Lenartowicz, T.; Pilarski, K.; Wojciechowski, T.; Pilarska, A.A.; Czechowska-Kosacka, A. The Application of Multiple Linear Regression and Artificial Neural Network Models for Yield Prediction of Very Early Potato Cultivars before Harvest. *Agronomy* **2021**, *11*, 885. [[CrossRef](#)]

32. Sabzi-Nojadeh, M.; Niedbała, G.; Younessi-Hamzekhanlu, M.; Aharizad, S.; Esmaeilpour, M.; Abdipour, M.; Kujawa, S.; Niazian, M. Modeling the Essential Oil and Trans-Anethole Yield of Fennel (*Foeniculum vulgare* Mill. var. *vulgare*) by Application Artificial Neural Network and Multiple Linear Regression Methods. *Agriculture* **2021**, *11*, 1191. [[CrossRef](#)]
33. Gorzelany, J.; Belcar, J.; Kuźniar, P.; Niedbała, G.; Pentoś, K. Modelling of Mechanical Properties of Fresh and Stored Fruit of Large Cranberry Using Multiple Linear Regression and Machine Learning. *Agriculture* **2022**, *12*, 200. [[CrossRef](#)]
34. Petrakis, T.; Kavga, A.; Thomopoulos, V.; Argiriou, A.A. Neural Network Model for Greenhouse Microclimate Predictions. *Agriculture* **2022**, *12*, 780. [[CrossRef](#)]
35. Altalak, M.; Ammaduddin, M.; Alajmi, A.; Rizg, A. Smart Agriculture Applications Using Deep Learning Technologies: A Survey. *Appl. Sci.* **2022**, *12*, 5919. [[CrossRef](#)]
36. Abdullahi, H.S.; Sheriff, R.E.; Mahieddine, F. Convolution neural network in precision agriculture for plant image recognition and classification. In Proceedings of the 2017 Seventh International Conference on Innovative Computing Technology (INTECH), Luton, UK, 16–18 August 2017; pp. 1–3. [[CrossRef](#)]
37. Escamilla-García, A.; Soto-Zarazúa, G.M.; Toledano-Ayala, M.; Rivas-Araiza, E.; Gastélum-Barrios, A. Applications of Artificial Neural Networks in Greenhouse Technology and Overview for Smart Agriculture Development. *Appl. Sci.* **2020**, *10*, 3835. [[CrossRef](#)]
38. Yao, C.; Zhang, Y.; Zhang, Y.; Liu, H. Application of Convolutional Neural Network in Classification of High Resolution Agricultural Remote Sensing Images. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *42*, 989–992. [[CrossRef](#)]
39. Li, C.; Zhen, T.; Li, Z. Image Classification of Pests with Residual Neural Network Based on Transfer Learning. *Appl. Sci.* **2022**, *12*, 4356. [[CrossRef](#)]
40. Jiao, L.; Dong, S.; Zhang, S.; Xie, C.; Wang, H. AF-RCNN: An anchor-free convolutional neural network for multi-categories agricultural pest detection. *Comput. Electron. Agric.* **2020**, *174*, 105522. [[CrossRef](#)]
41. Zhu, W.; Sun, J.; Wang, S.; Shen, J.; Yang, K.; Zhou, X. Identifying Field Crop Diseases Using Transformer-Embedded Convolutional Neural Network. *Agriculture* **2022**, *12*, 1083. [[CrossRef](#)]
42. Broomhead, D.S.; Lowe, D. Multivariable functional interpolation and adaptive networks. *Complex Syst.* **1988**, *2*, 321–355.
43. Piekarska-Boniecka, H.; Wilkaniec, B.; Dolańska-Niedbała, E. Parasitic wasps of the pimlinae subfamily (hymenoptera, ichneumonidae) of agricultural landscape refugium habitats in central Wielkopolska. *Acta Sci. Pol. Hortorum Cultus* **2008**, *7*, 23–29.
44. Boniecki, P.; Sujak, A.; Pilarska, A.A.; Piekarska-Boniecka, H.; Wawrzyniak, A.; Raba, B. Dimension Reduction of Digital Image Descriptors in Neural Identification of Damaged Malting Barley Grains. *Sensors* **2022**, *22*, 6578. [[CrossRef](#)]
45. Boniecki, P.; Weres, J. Neural Networks as a Tool in the Analysis of Agricultural Engineering Empirical Systems. *J. Res. Appl. Agric. Eng.* **2001**, *46*, 73–76.
46. Khalifani, S.; Darvishzadeh, R.; Azad, N.; SeyedRahmani, R. Prediction of sunflower grain yield under normal and salinity stress by RBF, MLP and, CNN Models. *Ind. Crops Prod.* **2022**, *189*, 115762. [[CrossRef](#)]
47. Hong, H.; Zhang, Z.; Guo, A.; Shen, L.; Sun, H.; Liang, Y.; Wu, F.; Lin, H. Radial Basis Function Artificial Neural Network (RBF Ann) as well as the hybrid method of RBF Ann and Grey Relational Analysis able to well predict trihalomethanes levels in tap water. *J. Hydrol.* **2020**, *591*, 125574. [[CrossRef](#)]
48. Deng, Y.; Zhou, X.; Shen, J.; Xiao, G.; Hong, H.; Lin, H.; Wu, F.; Liao, B.-Q. New methods based on back propagation (BP) and radial basis function (RBF) Artificial Neural Networks (Anns) for predicting the occurrence of haloketones in tap water. *Sci. Total Environ.* **2021**, *772*, 145534. [[CrossRef](#)]
49. Bourland, H.; Kamp, Y. Auto-association by multilayer perceptrons and singular value decomposition. *Biol. Cybern.* **1988**, *59*, 204–291. [[CrossRef](#)]
50. Dach, J.; Czekala, W.; Boniecki, P.; Lewicki, A.; Piechota, T. Specialised internet tool for biogas plant modelling and marked analysing. In Proceedings of the International Conference on Digital Image Processing (ICDIP 2014), Singapore, 9–10 February 2014. [[CrossRef](#)]
51. Speckt, D.F. Probabilistic Neural Networks. *Neural Netw.* **1990**, *3*, 109–118. [[CrossRef](#)]
52. Boniecki, P.; Raba, B.A.; Pilarska, A.; Sujak, A.; Zaborowicz, M.; Pilarski, K.; Wojcieszak, D. Neural Reduction of Image Data in Order to Determine the Quality of Malting Barley. *Sensors* **2021**, *21*, 5696. [[CrossRef](#)]
53. Toosi, N.B.; Soffianian, A.R.; Fakheran, S.; Waser, L.T. Mapping Disturbance in Mangrove Ecosystems: Incorporating Landscape Metrics and PCA-Based Spatial Analysis. *Ecol. Indic.* **2022**, *136*, 108718. [[CrossRef](#)]
54. Kramer, M.A. Nonlinear principal components analysis using autoassociative neural networks. *AIChE J.* **1991**, *37*, 233–243. [[CrossRef](#)]
55. Krevh, V.; Filipović, L.; Petošić, D.; Mustać, I.; Bogunović, I.; Butorac, J.; Kisić, I.; Defterdarović, J.; Nakić, Z.; Kovač, Z.; et al. Long-Term Analysis of Soil Water Regime and Nitrate Dynamics at Agricultural Experimental Site: Field-Scale Monitoring and Numerical Modeling Using HYDRUS-1D. *Agric Water Manag.* **2023**, *275*, 108039. [[CrossRef](#)]
56. Lu, N.; Niu, J.; Kang, S.; Singh, S.K.; Du, T. A hybrid PCA-Sem-Ann Model for the prediction of water use efficiency. *Ecol. Model.* **2021**, *460*, 109754. [[CrossRef](#)]
57. Kebonye, N.M.; Eze, P.N.; Agyeman, P.C.; John, K.; Ahado, S.K. Efficiency of the T-Distribution Stochastic Neighbor Embedding Technique for Detailed Visualization and Modeling Interactions between Agricultural Soil Quality Indicators. *Biosyst Eng.* **2021**, *210*, 282–298. [[CrossRef](#)]

58. Boniecki, P. The Kohonen Neural Network in Solving Classification Problems in Agricultural Engineering. *J. Res. Appl. Agric Eng.* **2005**, *50*, 37–41.
59. Ndikumana, E.; Ho Tong Minh, D.; Baghdadi, N.; Courault, D.; Hossard, L. Deep Recurrent Neural Network for Agricultural Classification using multitemporal SAR Sentinel-1 for Camargue, France. *Remote Sens.* **2018**, *10*, 1217. [[CrossRef](#)]
60. Boniecki, P.; Piekarska-Boniecka, H. The SOFM Neural network in the process of identification of selected orchard pests. *J. Res. Appl. Agric. Eng.* **2004**, *49*, 5–10.
61. Boniecki, P.; Idzior-Haufa, M.; Pilarska, A.; Pilarski, K.; Kolasa-Wiecek, A. Neural classification of compost maturity using artificial neural network type Self-Organizing Feature Map and algorithm. *IJERPH* **2019**, *16*, 3294. [[CrossRef](#)]
62. Hakimi-Asiabar, M.; Ghodsypour, S.H.; Kerachian, R. Multi-Objective Genetic Local Search Algorithm Using Kohonen's Neural Map. *Comput. Ind. Eng.* **2009**, *56*, 1566–1576. [[CrossRef](#)]
63. Van de Schoot, R.; Depaoli, S.; King, R.; Kramer, B.; Märtens, K.; Tadesse, M.G.; Vannucci, M.; Gelman, A.; Veen, D.; Willemsen, J.; et al. Bayesian statistics and modelling. *Nat. Rev. Methods Prim.* **2021**, *1*, 1. [[CrossRef](#)]
64. Imholz, M.; Vandepitte, D.; Moens, D. Bayesian estimation of interval bounds based on limited data. In Proceedings of the ISMA 2018-USD 2018, Leuven, Belgium, 17–19 September 2018; KU Leuven, Department of Mechanical Engineering: Leuven, Belgium, 2019; pp. 5207–5214.
65. Krause, P.J.; Bokinala, V. A Tutorial on Data Mining for Bayesian Networks, with a Specific Focus on IoT for Digital Agriculture. *Internet Things* **2023**, *22*, 100738. [[CrossRef](#)]
66. Drury, B.; Valverde-Rebaza, J.; Moura, M.-F.; de Andrade Lopes, A. A survey of the applications of Bayesian networks in agriculture. *Eng. Appl. Artif. Intell.* **2017**, *65*, 29–42. [[CrossRef](#)]
67. Patterson, D. *Artificial Neural Networks*; Prentice Hall: Singapore, 1996.
68. Sujak, A.; Kusz, A.; Rymarz, M.; Kitowski, I. Environmental Bioindication Studies by Bayesian Network with Use of Grey Heron as Model Species. *Environ. Model. Assess.* **2017**, *22*, 103–113. [[CrossRef](#)]
69. Yin, J.; Medellín-Azuara, J.; Escrivá-Bou, A.; Liu, Z. Bayesian machine learning ensemble approach to quantify model uncertainty in predicting groundwater storage change. *Sci. Total Environ.* **2021**, *769*, 144715. [[CrossRef](#)] [[PubMed](#)]
70. Radomyski, A.; Ashauer, R. A Site-Specific Indicator of Nitrogen Loads into Surface Waters from Conventional and Conservation Agriculture Practices: Bayesian Network Model. *Ecol. Indic.* **2022**, *145*, 109641. [[CrossRef](#)]
71. Wu, H.; Su, X.; Singh, V.P.; AghaKouchak, A.; Liu, Z. Bayesian vine copulas improve agricultural drought prediction for long lead times. *Agric. For. Meteorol.* **2023**, *331*, 109326. [[CrossRef](#)]
72. Lachaud, M.A.; Bravo-Ureta, B.E. A bayesian statistical analysis of return to agricultural R&D investment in Latin America: Implications for food security. *Technol. Soc.* **2022**, *70*, 102054.
73. Boniecki, P.; Mueller, W. Expectation crops of chosen agricultural fetuses with the help of neural model by time series. *J. Res. Appl. Agric Eng.* **2006**, *51*, 40–44.
74. Liu, G.; Zhong, K.; Li, H.; Chen, T.; Wang, Y. A State of Art Review on Time Series Forecasting with Machine Learning for Environmental Parameters in Agricultural Greenhouses. *Inf. Process. Agric.* **2022**, *in press*. [[CrossRef](#)]
75. Tofu, D.A.; Mengistu, M. Observed time series trend analysis of climate variability and smallholder adoption of new agricultural technologies in west Shewa, Ethiopia. *Sci. Afr.* **2023**, *19*, e01448. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.