THE DEEP NEURAL NETWORK-A REVIEW

Eman Jawad*

**Assistant. Lecher/ AL –Furat Al-Awsat Technical University/Iraq.*

Corresponding Author:
eman.naji@atu.edu.iq

Abstract:

Deep neural networks are considered the backbone of artificial intelligence, we will present a review of an article about the importance of neural networks and their role in other sciences, their characteristic, networks architecture, types, mathematical definition of deep neural networks, as well as their applications.

Keywords: *neural networks, deep neural networks, ReLU function, optimization*

INTRODUCTION

Neural networks have seen an explosion of interest over the last few years and are being successfully applied across an extraordinary range of problem domains, in areas as diverse as finance, medicine, engineering, geology, physics and biology. The excitement stems from the fact that these networks are attempts to model the capabilities of the human brain. From a statistical perspective neural networks are interesting because of their potential use in prediction and classification problems**.**

Artificial neural networks (ANNs) are non-linear data driven self-adaptive approach as opposed to the traditional model based methods. They are powerful tools for modeling, especially when the underlying data relationship is unknown. ANNs can identify and learn correlated patterns between input data sets and corresponding target values. After training, ANNs can be used to predict the outcome of new independent input data. ANNs imitate the learning process of the human brain and can process problems involving non-linear and complex data even if the data are imprecise and noisy.

These networks are “neural” in the sense that they may have been inspired by neuroscience but not necessarily because they are faithful models of biological neural or cognitive phenomena. In fact majority of the network are more closely related to traditional mathematical and/or statistical models such as non-parametric pattern classifiers, clustering algorithms, nonlinear filters, and statistical regression models than they are to neurobiology models**.**

Neural networks (NNs) have been used for a wide variety of applications where statistical methods are traditionally employed. They have been used in classification problems, such as identifying underwater sonar currents, recognizing speech, and predicting the secondary structure of globular proteins. In time-series applications, NNs have been used in predicting stock market performance. As statisticians or users of statistics, these problems are normally solved through classical statistical methods, such as discriminant analysis, logistic regression, Bayes analysis, multiple regression, and ARIMA time-series models. It is, therefore, time to recognize neural networks as a powerful tool for data analysis**.**

Characteristics of neural networks

- The NNs exhibit mapping capabilities, that is, they can map input patterns to their associated output patterns.
- The NNs learn by examples. Thus, NN architectures can be ‘trained’ with known examples of a problem before they are tested for their ‘inference’ capability on unknown instances of the problem. They can, therefore, identify new objects previously untrained.
- The NNs possess the capability to generalize. Thus, they can predict new outcomes from past trends.
- The NNs are robust systems and are fault tolerant. They can, therefore, recall full patterns from incomplete, partial or noisy patterns.
- The NNs can process information in parallel, at high speed, and in a distributed manner**.**

Basics of artificial neural networks

The terminology of artificial neural networks has developed from a biological model of the brain. A neural network consists of a set of connected cells: The neurons. The neurons receive impulses from either input cells or other neurons and perform some kind of transformation of the input and transmit the outcome to other neurons or to output cells. The neural networks are built from layers of neurons connected so that one layer receives input from the preceding layer of neurons and passes the output on to the subsequent layer. Mathematically a Multi-Layer Perceptron network is a function consisting of compositions of weighted sums of the functions corresponding to the neurons**.**

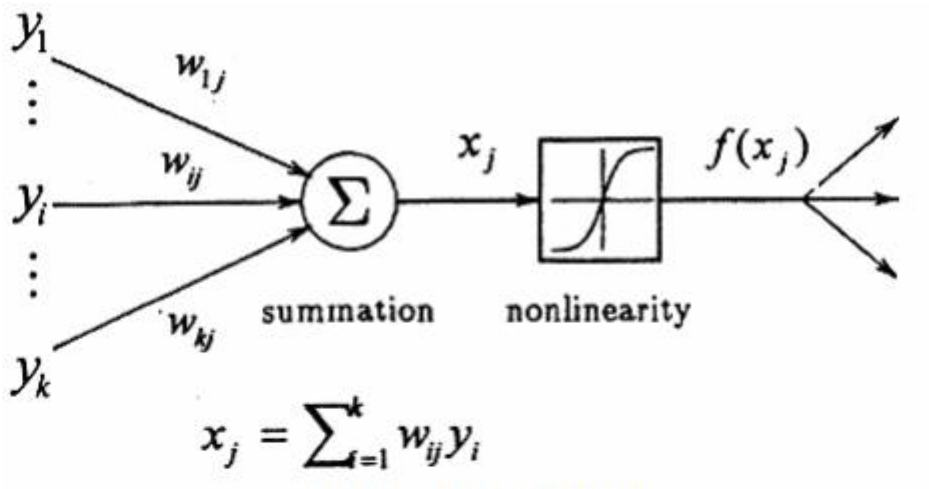

Figure: A single neuron

Neural networks architectures:

Feed forward networks In a feed forward network, information flows in one direction along connecting pathways, from the input layer via the hidden layers to the final output layer. There is no feedback (loops) i.e., the output of any layer does not affect that same or preceding layer.

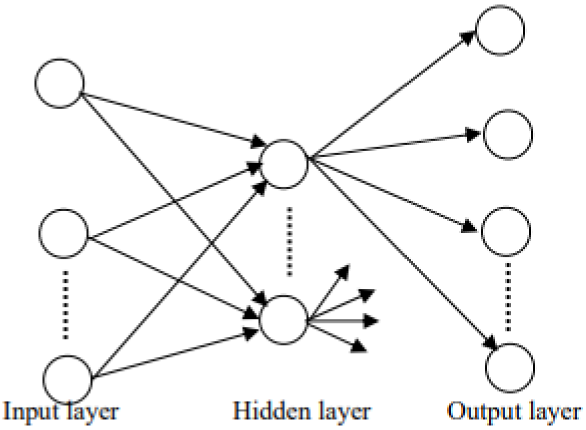

Figure: A multi-layer feed forward neural network

Recurrent networks These networks differ from feed forward network architectures in the sense that there is at least one feedback loop. Thus, in these networks, for example, there could exist one layer with feedback connections as shown in figure below. There could also be neurons with self -feedback links, i.e. the output of a neuron is fed back into itself as input.

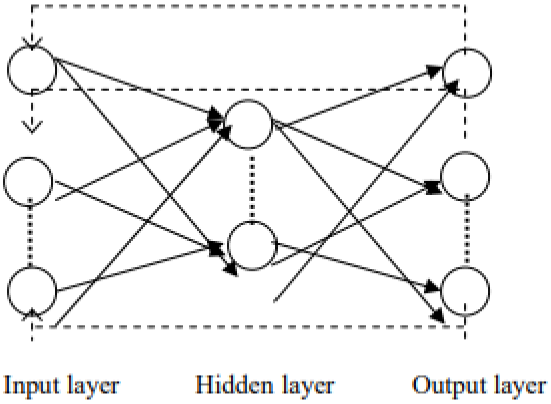

Figure: A recurrent neural network

Types of neural networks

The most important class of neural networks for real world problems solving includes:

- Multilayer Perceptron
- Radial Basis Function Networks
- Kohonen Self Organizing Feature Maps

Definition of Deep Neural Networks (DNN)

The core building blocks are, as said, artificial neurons. For their definition, let us recall the structure and functionality of a neuron in the human brain**.** This forms the basis for a mathematical definition of an artificial neuron.

Definition : An artificial neuron with weights $\omega_1, \dots, \omega_n \in R$, bias $b \in R$, and activation function $\rho : R \to R$ is defined as the function $f: R \to R$ given by

$$f(x_1, \dots \dots, x_n) = \rho\left(\sum_{i=1}^{n} x_i\omega_i - b\right) = \rho(\langle x, \omega\rangle - b)$$

By now, there exists a zoo of activation functions with the most well-known ones being as follows:

1- Heaviside function $\rho(x) = \begin{cases} 1\,, \mathrm{x} > 0 \\ 0\,, \mathrm{x} \le 0 \end{cases}$

2- Sigmoid function $\rho(x) = \frac{1}{1+e^{-x}}$

3- Rectifiable Linear Unit (ReLU) $\rho(x) = max\{0, x\}$.

Definition: Let $d \in \mathbb{N}$ be the dimension of the input layer, L the number of layers, $N_0 = d, N_\ell, \ell = 1, \dots\dots, L$ the dimensions of the hidden and last layer, $\rho : R \to R$ a (non-linear) activation function, and, for $\ell = 1, \dots., N$ let T_ℓ be the affine-linear functions

$$\mathrm{T}_{\boldsymbol{\ell}}: \mathbb{R}^{\boldsymbol{N_{\ell-1}}} \to \mathbb{R}^{\boldsymbol{N_\ell}}, \qquad \mathrm{T}_\ell \mathrm{x} = \mathrm{W}^{(\ell)} + \mathrm{b}^{(\ell)}$$

With $W^{(\ell)} \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$ being the weight matrices and $b^\ell \in \mathbb{R}^{N_\ell}$ the bias vectors of the ℓth layer. Then $\Phi : \mathbb{R}^d \to \mathbb{R}^{N_L}$, given by

$$\Phi(x) = T_L \rho \left(T_{L-1} \rho \left(\dots \dots \rho\left(T_1(x)\right)\right)\right)$$

is called (deep) neural network of depth L.

An illustration of the multilayered structure of a deep neural network can be found in Figure 1.

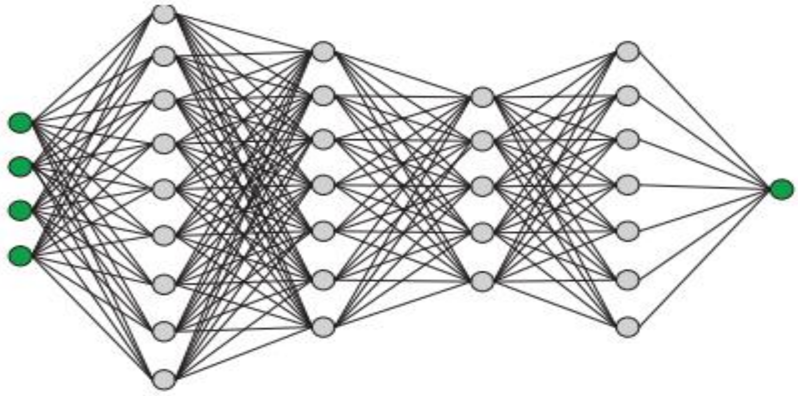

Figure 1: Deep neural network $\Phi: \mathbb{R}^4 \to \mathbb{R}$ with depth 5

Application of a Deep Neural Network

1- **(Train-test split of the dataset)**: We assume that we are given samples $(x^{(i)}, y^{(i)})_{i=1}^{\widetilde{m}}$ of inputs and outputs. The task of the deep neural network is then to identify the relation between those. For instance, in a classification problem, each output $y^{(i)}$ is considered to be the label of the respective class to which the input $x^{(i)}$belongs. One can also take the viewpoint that $(x^{(i)}, y^{(i)})_{i=1}^{\widetilde{m}}$ arises as samples from a function such as $g : M \to \{1, 2, \dots, K\}$, where M might be a lower-dimensional manifold of $\mathbb{R}^d$, in the sense of $y^{(i)} = g(x^{(i)})$ for all $i = 1, \dots, \widetilde{m}$. The set $(x^{(i)}, y^{(i)})_{i=1}^{\widetilde{m}}$ is then split into a training data set $(x^{(i)}, y^{(i)})\,{}_{i=1}^{\widetilde{m}}$and a test data set $(x^{(i)}, y^{(i)})\,{}_{i=m+1}^{\widetilde{m}}$. The training data set is—as the name indicates—used for training, whereas the test data set will later on solely be exploited for testing the performance of the trained network. We emphasize that the neural network is not exposed to the test data set during the entire training process.

2- **(Choice of architecture)**: For preparation of the learning algorithm, the architecture of the neural network needs to be decided upon, which means the number of layers L , the number of neurons in each layer $(N_\ell)_{\ell=1}^{L}$, and the activation function ρ have to be selected. It is known that a fully connected neural network is often difficult to train, hence, in addition, one typically preselects certain entries of the weight matrices $\left(W^\ell\right)_{\ell=1}^{N}$ to already be set to zero at this point. For later purposes, we define the selected class of deep neural networks by $\mathcal{NN}_\theta$ With θ encoding this chosen architecture.

3- **(Training)**: The next step is the actual training process, which consists of learning the affine-linear functions $(T_\ell)_{\ell=1}^{N} = \left((W)^\ell + (b)^\ell\right)_{\ell=1}^{N}$ This is accomplished by minimizing the empirical risk**.**

$$\hat{R}\left(\Phi\left(W^\ell, b^\ell\right)_\ell\right) = \frac{1}{m} \sum_{i=1}^{m} \left(\Phi\left(W^\ell, b^\ell\right)_\ell \left(x^{(i)}\right) - y^{(i)}\right)^2$$

more general form of the optimization problem is min

$$\min_{\left(W^{(\ell)}, b^{(\ell)}\right)_\ell} \sum_{i=1}^{m} \mathcal{L}\left(\Phi\left(W^\ell, b^\ell\right)_\ell (x_i) - y^{(i)}\right) + \lambda \mathcal{P}\left(\left(W^\ell + b^\ell\right)_\ell\right)$$

Where $\mathcal{L}$ is a loss function to determine a measure of closeness between the network evaluated in the training samples and the (known) values $y^{(i)}$ and where $\mathcal{P}$ is a penalty/regularization term to impose additional constraints on the weight matrices and bias vectors. One common algorithmic approach is gradient descent. Since, however, m is typically very large, this is computationally not feasible. This problem is circumvented by randomly selecting only a few gradients in each iteration, assuming that they constitute a reasonable average, which is coined stochastic gradient descent. Solving the optimization problem then yields a network $\Phi\left(W^\ell, b^\ell\right)_\ell : \mathbb{R}^d \to \mathbb{R}^{N_L}$, where

$$\Phi\left(W^{(\ell)}, b^{(\ell)}\right)_\ell (x) = T_L \rho \left(T_{L-1} \rho \left(\dots \dots \rho\left(T_1(x)\right)\right)\right)$$

4- **(Testing)**: Finally, the performance (often also called generalization ability) of the trained neural network is tested using the test data set $(x\ (i)\ , y(i)\ )\ _{i=1}^{\widetilde{m}}$ by analyzing whether**.**
$\Phi\left(W^{(\ell)}, b^{(\ell)}\right)_{\ell}\left(x^{(i)}\right) \approx y^{(i)}\ \ , i = m + 1 \ldots \ldots \widetilde{m}$.

References

[1]. Adler.j and Oktem.o, (2017), Solving ill-posed inverse problems using iterative deep neural networks. Inverse ¨ Probl. 33, 124007

[2]. Anderson, J. A. (2003). An Introduction to neural networks. Prentice Hall.

[3]. Andrade-Loarca.H, Kutyniok.G,Oktem .O, and Petersen.P, (2019), Extraction of digital wavefront sets using ¨ applied harmonic analysis and deep neural networks. SIAM J. Imaging Sci. 12, 1936–1966.

[4]. Andrade-Loarca.H, Kutyniok.G, Oktem.O, and Petersen.P , (2021), Deep Microlocal Reconstruction for Limited- ¨ Angle Tomography, arXiv:2108.05732.

[5]. Bach.S, Binder.A, Montavon.G, Klauschen.F, M¨uller.K.R, and Samek.W, (2015), On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PLoS ONE 10 ,e0130140.

[6]. Belkin.M, Hsu.D, Ma.S, and Mandal.S, (2019), Reconciling modern machine-learning practice and the classical bias–variance trade-off. Proc. Natl. Acad. Sci. USA 116 ,15849–15854.

[7]. Berner.J, Grohs.P, Kutyniok.G, and Petersen.P,(2021), The Modern Mathematics of Deep Learning. In: Mathematical Aspects of Deep Learning, Cambridge University Press, to appear.

[8]. ¨olcskei H.B, Grohs.P, Kutyniok .G, and Petersen.P, (2019), Optimal Approximation with Sparsely Connected Deep Neural Networks. SIAM J. Math. Data Sci. 1 ,8–45.

[9]. Cheng, B. and Titterington, D. M. (1994). Neural networks: A review from a statistical perspective. Statistical Science, 9, 2-54

[10]. Cybenko.G (1989), Approximation by superpositions of a sigmoidal function. Math. Control Signal 2 (, 303– 314.

[11]. Dewolf, E.D., and Francl, L.J., (1997). Neural networks that distinguish in period of wheat tan spot in an outdoor environment Phytopathalogy, 87, 83-87 .

[12]. Dewolf, E.D. and Francl, L.J. (2000) Neural network classification of tan spot and stagonespore blotch infection period in wheat field environment. Phytopathalogy, 20-,108-113.

[13]. D Donoho.D, (2001), Sparse components of images and optimal atomic decompositions. Constr. Approx. 17 ,353–382.

[14]. E.W and B. Yu. (2018), The deep ritz method: a deep learning-based numerical algorithm for solving variational problems. Commun. Math. Stat. 6 ,1–12.

[15]. Gaudart, J. Giusiano, B. and Huiart, L. (2004). Comparison of the performance of multi-layer perceptron and linear regression for epidemiological data. Comput. Statist. & Data Anal., 44, 547-70 .

[16]. Hassoun, M. H. (1995). Fundamentals of Artificial Neural Networks. Cambridge: MIT Press .

[17]. Hopfield, J.J. (1982). Neural network and physical system with emergent collective computational capabilities. In proceeding of the National Academy of Science(USA79) ,2554-2558

[18]. Kaastra, I. and Boyd, M.(1996). Designing a neural network for forecasting financial and economic time series. Neurocomputing, 10, 215-236 .

[19]. Kohzadi, N., Boyd, S.M., Kermanshahi, B. and Kaastra, I. (1996). A comparision of artificial neural network and time series models for forecasting commodity prices Neurocomputing, 10, 169-181 .

[20]. Kumar, M., Raghuwanshi, N. S., Singh, R,. Wallender, W. W. and Pruitt, W. O. (2002)Estimating Evapotranspiration using Artificial Neural Network. Journal of Irrigation and Drainage Engineering, 128, 224-233

[21]. Masters, T. (1993). Practical neural network recipes in C++, Academic press, NewYork .

[22]. Mcculloch, W.S. and Pitts, W. (1943) A logical calculus of the ideas immanent in nervous activity. Bull. Math. Biophy., 5, 115-133

[23]. Pal, S. Das, J. Sengupta, P. and Banerjee, S. K. (2002). Short term prediction of atmospheric temperature using neural networks. Mausam, 53, 471-80

[24]. Patterson, D. (1996). Artificial Neural Networks. Singapore: Prentice Hall .

[25]. Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage ang organization in the brain. Psychological review, 65, 386-408 .

[26]. Rumelhart, D.E., Hinton, G.E and Williams, R.J. (1986). "Learning internal representation by error propagation", in Parallel distributed processing: Exploration in microstructure of cognition, Vol. (1) (D.E. Rumelhart, J.L. McClelland and the PDP research gropus ,edn.) Cambridge, MA: MIT Press, 318-362.

[27]. Saanzogni, Louis and Kerr, Don (2001) Milk production estimate using feed forward artificial neural networks. Computer and Electronics in Agriculture, 32, 21-30 .

[28]. Warner, B. and Misra, M. (1996). Understanding neural networks as statistical tools American Statistician, 50, 284-93 .

[29]. Yegnanarayana, B. (1999). Artificial Neural Networks. Prentice Hall .

[30]. Zhang, G., Patuwo, B. E. and Hu, M. Y. (1998). Forecasting with artificial neural networks :The state of the art. International Journal of Forecasting, 14, 35-62.