# Fundamentals of Neural Networks

*Seminar Data Mining*

Mathias Jackermeier
Fakultät für Informatik
Technische Universität München
Email: mathias.jackermeier@tum.de

***Abstract*— Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrumCurabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.**

***Index Terms*— test**

## I. Introduction

Artificial intelligence systems have been becoming more and more powerful over the last 10 years. We have seen outstanding advances in a variety of fields including computer vision, natural language processing and fraud detection, which power many end-user technologies such as digital assistants or self-driving cars. Much of the recent progress can be attributed to *deep learning*, a powerful set of techniques that enable computers to understand the world by decomposing complex concepts into a hierarchy of simpler abstractions.

While numerous other approaches to machine learning exist, deep learning has shown to outperform other methods in a wide variety of applications. To name a few examples, deep learning models dominate the task of object recognition in images [1], even surpassing human-level performance [2], have been successfully applied to sentiment analysis [3], and have significantly improved speech recognition systems [4]. Deep learning has also been used in problems such as style transfer between images [5], image description generation [6], and learning to play video games [7].

By learning everything required to solve a task purely from raw data, these techniques have alleviated the need for problem-specific expert knowledge. Thus, very similar models building on the same core ideas can be applied to a vast array of different tasks with outstanding success.

One such core idea that is fundamental to deep learning is the *neural network*, a computing model loosely inspired by neuroscience. While neural networks are not new, it was not until recently that enough data and computational resources became available to train them effectively and fully appreciate their power [8, Ch. 1, pp. 18-21].

Since neural networks have become so prevalent in modern machine learning applications, many libraries exist that abstract their concepts and provide simple programming interfaces. However, it does not suffice to be familiar with such libraries to use neural networks effectively; in order to understand which architectures perform well, and why, one must also know their mathematical foundations.

In this paper we thus aim to give a thorough overview of neural networks and the fundamental techniques and algorithms associated with them. We first briefly examine the motivation and history behind neural networks in Section II by introducing the *perceptron* model. Section III then shows how this model has been adjusted and extended to obtain the neural network, focusing in particular on *feedforward neural networks*. In Section IV, we then proceed to explain how these networks can be trained, introducing ideas such as *stochastic gradient descent* and *back-propagation*. Subsequently, Section V discusses the effectiveness of neural networks from a theoretical point of view. In Section VI we examine several extensions to the basic feedforward neural network that are often used in practice, before we conclude our paper in Section VII.

## II. The Perceptron

When researchers developed the first machine learning models, they often used ideas based closely on our understanding of the brain. One such model, inspired by the biological neuron, is the perceptron, which was first conceived by [9].

Like its biological counterpart, the perceptron receives information and produces an output. More specifically, it accepts $n$ input values $x_1, \ldots, x_n$ and calculates a corresponding output value $\hat{y} \in \{-1, 1\}$ by computing

$$\hat{y} = \operatorname{sign}\left(\sum_{i=1}^{n} w_i x_i\right), \tag{1}$$

where the weights $w_i$ are the parameters of the model, and $\operatorname{sign}(x)$ is defined as

$$\operatorname{sign}(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0. \end{cases} \tag{2}$$

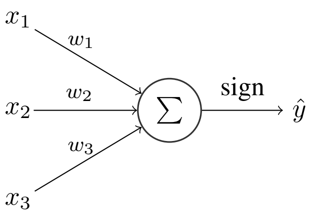


Fig. 1. An illustration of the perceptron model. In this example, the perceptron accepts three inputs $x_1, x_2, x_3$, has the parameters $w_1, w_2, w_3$, and computes $\hat{y} = \text{sign}(w_1 x_1 + w_2 x_2 + w_3 x_3)$.

By representing the input values and weights as vectors $\boldsymbol{x}$ and $\boldsymbol{w}$, we can rewrite (1) as

$$\hat{y} = \text{sign}(\boldsymbol{w}^\top \boldsymbol{x}). \tag{3}$$

For a visual representation of this model, see Fig. 1.

Perceptron models can be used to solve binary classification problems. In this scenario, we are given a set of $m$ training examples $\mathbb{X} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}\}$ and their corresponding binary labels $\mathbb{Y}$, and wish to predict the most probable label for an unseen vector $\boldsymbol{x} \notin \mathbb{X}$.

For example, the vectors $\boldsymbol{x}^{(i)}$ might describe features of an email using a *bag-of-words* representation. That is, we define a fixed vocabulary, and the $j$th entry in the vector $\boldsymbol{x}^{(i)}$ specifies how often the $j$th word of the vocabulary occurs in the particular email represented by $\boldsymbol{x}^{(i)}$. The corresponding label $y^{(i)} = 1$ then might signify that the email is a legitimate email, whereas a value of $y^{(i)} = -1$ might label the email as spam.

In the beginning, the weights are randomly initialized and the model thus makes arbitrary predictions. During the process of *training* the perceptron, we iteratively adjust the weights in order to improve the prediction accuracy on the training set.

One common method of training is the perceptron learning algorithm proposed by [10]. Essentially, the algorithm iterates through the training data $\mathbb{X}$ and makes small adjustments to the weights if a particular training example $x^{(i)}$ is misclassified. For example, if the perceptron predicts $\hat{y} = 1$ and the actual label is $y^{(i)} = -1$, the weights are corrected in the negative direction. Since the perceptron learning algorithm is not directly applicable to neural networks, we will not discuss it further; a more in depth explanation can be found in [11, Ch. 8, pp. 265-267].

A major shortcoming of the perceptron is that it can only learn to classify linearly separable data [12]. For example, the XOR function, where

$$\text{XOR}(x) = \begin{cases} 0 & \text{if } x = [0,0] \lor x = [1,1] \\ 1 & \text{if } x = [1,0] \lor x = [0,1], \end{cases} \tag{4}$$

cannot be learned with the perceptron. The discovery of these limitations has greatly reduced interest in the field of biological learning, until more sophisticated models, such as neural networks, were developed [8, Ch. 1, pp. 12-18].

## III. Feedforward Neural Networks

## IV. Training Feedforward Neural Networks

## V. Universal Approximation Capabilities

## VI. Extensions and Applications

## VII. Conclusion

## References

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015. [Online]. Available: https://doi.org/10.1007/s11263-015-0816-y

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 1026–1034. [Online]. Available: https://doi.org/10.1109/ICCV.2015.123

[3] A. Severyn and A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, R. A. Baeza-Yates, M. Lalmas, A. Moffat, and B. A. Ribeiro-Neto, Eds. ACM, 2015, pp. 959–962. [Online]. Available: http://doi.acm.org/10.1145/2766462.2767830

[4] A. Mohamed, G. E. Dahl, and G. E. Hinton, "Acoustic modeling using deep belief networks," *IEEE Trans. Audio, Speech & Language Processing*, vol. 20, no. 1, pp. 14–22, 2012. [Online]. Available: https://doi.org/10.1109/TASL.2011.2109382

[5] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 2414–2423. [Online]. Available: https://doi.org/10.1109/CVPR.2016.265

[6] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 664–676, 2017. [Online]. Available: https://doi.org/10.1109/TPAMI.2016.2598339

[7] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. A. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015. [Online]. Available: https://doi.org/10.1038/nature14236

[8] I. J. Goodfellow, Y. Bengio, and A. C. Courville, *Deep Learning*, ser. Adaptive computation and machine learning. MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org/

[9] W. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, 1943. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-51249194645&doi=10.1007%2fBF02478259&partnerID=40&md5=edb67afceee33d22eaabbf1f8c1dca90

[10] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-11144273669&doi=10.1037%2fh0042519&partnerID=40&md5=f6ad02750f121e6d5d33566003d5ac8b

[11] K. P. Murphy, *Machine learning - a probabilistic perspective*, ser. Adaptive computation and machine learning series. MIT Press, 2012.

[12] M. Minsky and S. Papert, *Perceptrons - an introduction to computational geometry*. MIT Press, 1987.