1. Introduction

1.1. About the content

At first, some background ideas are given and what the origins of neurocomputing and artificial neural networks were. Then we start from single neurons or computing units that are arranged and put together to form networks to be used as nodes in special directed graph data structures in order to express some computational tasks depending on the types of neural networks.

After introducing neural network basics, their different main types are considered, such as feedforward or multilayer perceptron neural networks and self-organizing maps. Learning or training algorithms are also considered, e.g., the "traditional" Backpropagation Training. Training and evaluation are considered in general since these are always an essential part while constructing neural networks in order to model something on the basis of data given.

Few necessary (particularly in this context) preprocessing techniques are explained in brief, for instance, One-of-N encoding or binarization of categorical or nominal variables, features or attributes and normalization of variable ranges.

To follow most modern development in the field of neurocomputing we make acquaintance with convolutional neural networks in association with deep learning.

As frequently in machine learning, the purpose is, ultimately, to construct computational models for some interesting data. Relevant themes of it are presented.

1.2 On the origin and development of neurocomputing

An idea to model single biological neurons by means based on simple biophysical formulas was presented by W. McCulloch and W. Pitts in 1943. At first, the objective was to model biological systems, an animal or human brain. Then it was invented that algorithms that were begun to outline and develop can be used for computation independent of whether they modelled any biological or biochemical functions.

Minsky and Papert (1969) showed that a (one-layer) perceptron, i.e. simple network, could only do linearly separable problems. Nevertheless, when Rumelhart and McClelland proposed their improved version (1986), called the multilayer perceptron this can be seen as a beginning of efficient artificial neural networks. The most recent area is deep learning for neural networks (Hinton, 2006), but its roots were already developed in the 1990s.

F. Rosenblatt (1958) designed the first version of backpropagation algorithm which computes the weight matrix for a neural network.
1.3 Background on the nerves and brain

Why do we talk about artificial neural networks? Some of early and even later thoughts and methods, e.g., self-organising maps by T. Kohonen in the 1980s are based on the ideas how actual nerves and brains function and even endeavour to model these. Since a human brain is a really complicated organ, such modelling is naturally an extremely challenging task. The human brain is one of the most complicated things that have been studied in detail and obviously, on the whole, still poorly understood.

We do not have satisfactory answers to the most fundamental of questions such as "what is one's mind" and "how one thinks".

Nevertheless, we have a basic understanding of the operation of the brain at a low level. It is believed that the brain contains around ten thousand million \(10^{10}\) basic units, neurons. Each of these neurons is connected to about ten thousand \(10^4\) others.

Although the connection to real biology may be quite loose, the current area of artificial intelligence, neurocomputing has yielded extensively versatile neural network algorithms since the 1980s.

The neuron is the basic unit of the brain, and is a stand-alone analogue-logical processing unit.

The neurons form two main types, local processing interneuron cells that have their input and output connections over about 100 \(\mu\)s \(10^{-6}\) s, and output cells connecting different regions of the brain to each other, the brain to muscle or from sensory organs into the brain.

It is known that the neuron accepts many inputs, which are all added up in some way. If enough active inputs are received at once, then the neuron will be activated and "fire"; if not, the neuron will remain in its inactive, quiet state.

Let us look at a simplified description of a biological neuron in Fig. 1.1.

Figure 1.1 The basic features of a biological neuron.

The soma is the body of the neuron. Attached to the soma there are long, irregularly shaped filaments, dendrites which have complex branching shapes.
Through dendrites all inputs arrive in the neuron. These cells are able
to perform more complicated functions than simple addition on the
inputs received, but considering a simple summation is a reasonable
approximation.

Axon is also connected to the neuron, is electrically active, unlike the
dendrite, and serves as the output channel of the neuron. Axons
always appear on output cells, but are often absent from interneurons,
which have dendrites for both inputs and outputs. The axon is a non-
linear threshold device generating a voltage pulse, called an action
potential, that lasts at least 0.2 ms ($10^{-3}$ s) when the resting potential
within the soma rises above a certain critical threshold. In fact, such
action potential is a series of rapid voltage spikes. See Fig. 1.2.

The axon terminates in a special contact called a synapse that couples
the axon with the dendrite of another cell. The junction is a temporary
chemical one. The synapse releases chemicals called neurotransmitters
after its potential is raised sufficiently by the action potential. It may
take the arrival of more than one action potential before the synapse is
triggered. Some synapses excite the dendrite they affect, whilst others
serve to inhibit it. This corresponds to altering the local potential of the
dendrite in a positive or negative direction. A single neuron will have
many synaptic inputs, and may have also many synaptic outputs.

---

Fig. 1.2(a) (a) An action potential; (b) Schematically seen, the input to
the biological neuron body has to exceed a threshold before the cell
will fire. This is called all-or-nothing principle.

Fig. 1.3 Neurotransmitters released from the synaptic vesicles diffuse
across the synaptic cleft or gap and trigger the receivers on the
dendrite.

Learning in biological systems is thought to occur when modifications
are made to the effective coupling between one cell and another, at
the synaptic junctions. Fig. 1.3 shows this.
Releasing more transmitters has the effect of opening more gates on the dendrite on the post-synaptic side of the junction and so increasing the coupling effect of the two cells. The adjustment of coupling so as to favourably reinforce good connections is essential for artificial neural net models, as is the effective coupling or weighting that appears on connections into a neuronal cell.

The brain is organised into different regions, each responsible for different functions and in humans this organisation is very marked. The largest parts of the brain are the cerebral hemispheres, which occupy most of the interior of the skull. They are layered structures, the most complex being the outer layer, known as the cerebral cortex.

18

In the visual part of the cortex, electrical stimulation of the cells can produce the sensation of light and specific layers of neurons are sensitive to particular orientations of input stimuli, so that one layer responds maximally to horizontal lines, while another responds maximally to vertical ones. Although much of this structure is genetically pre-determined, the orientation-specific layout of the cells appears to be learnt at an early stage. This self-organization of the visual cortex, so called since there is no external teacher to guide the development of these structures, was a seed for ideas such as self-organising maps of T. Kohonen.

19

1.4 Machines vs. actual brains

Both biological brains and artificial neural networks utilise distributed processing. However, the latter contains massively parallel systems, but the latter are moderate in this sense even among the new supercomputers. For both the ability to learn (from data) and to adapt is essential even if the capabilities of a human and a computer are finally very different.

A machine can easily "remember" a long sequence of numbers. A human can make complicated deductions. On the other hand, machine learning methods and huge data set available, e.g., in Internet, also make machines and programs more efficient to make decisions within a limited scope.

20

Both are more or less fault-tolerant. A computer might still do even if some of its "neurons", e.g., memory or if its parallel units were inoperative. A human may still survive after stroke and even then, suffering from aphasia, again to learn to talk if one is lucky enough to receive speech therapy.
1.5 More background information for neurocomputing

Pattern recognition is a task that neural networks are frequently good to accomplish. At the highest level, a typical neural network can perform only this function although patterns need not to be any visual or visually represented data, but a pure abstraction or an item of some reasonable data set. Fig. 1.4 illustrates a neural network at this level. In principle, this neural network accepts a pattern and returns a pattern. For instance, it may recognize a dog from a figure or a cat. A human might recognize a person known to the spectator or find the person to be unknown (or perhaps to be uncertain, something between 'yes' and 'no').

Neural networks consist of layers of similar neurons or nodes. Most have an input layer, one or more hidden layers, but not always mandatory, and an output layer. Input variable values are given to the input layer. Then the output pattern is returned from the output layer. What happen between the input and output layer is seemingly a "black box". Black box means that we do not know exactly why a neural network outputs what it does. The properties (mainly weights, parameters, used in the network) are computed by a learning or training algorithm for a suitable structure and construction of the network with evaluated parameter values. Finally this network is run to generate outputs.

At this point we are not yet concerned with the internal structure of the neural network or the black box. Different architectures define the interaction between the input and output layer. Later, these architectures will be examined.

1.6 Learning or training: supervised and unsupervised

If there are "correct" outputs known at our disposal, we can use them as examples to train a neural network. This is called supervised learning. For instance, if there are data of patients and normals or controls, subjects who are assumed to be healthy, we may be able to construct a neural network to separate these two groups (classes) of subjects. Naturally, the data, i.e. variables, features or attributes, have to include such information that measure important indicators useful for the separation or classification between the two classes. Experts of a certain field, say, physicians are able to define them. Alternatively, we may sometimes be able to do it computationally. This depends on the sort of data.
For example, if the data concern digital documents like web pages, our program can calculate (relative) frequencies of semantically meaningful words, such as nouns and verbs, and then compute various measures for the classification of documents on the basis of their topics like politics, sports etc.

Using known examples, i.e., their correct classes are known in advance, we can use a set of these known examples or samples, called learning or training set, and construct neural networks by means of some training algorithm. This is of supervised learning.

In unsupervised learning, no information about correct classes or corresponding is at our disposal. A training algorithm has to be capable of taking care how to form appropriate subsets of samples or cases of a training set. Usually these are called clusters. A cluster then includes such cases that are more similar between each other than compared to those cases of another cluster. A network built is finally tested with a test set different from the training set.

Similarity or dissimilarity (generally proximity) measures are used for the computation of these properties. Metrics, for instance Euclidean metric, are suitable, as specific distance measures, for certain situations depending on the types of variables applied (see the material of Data Mining course 2016, Ch. 3).

Most neural networks are purposed to supervised learning.

1 http://www.uta.fi/sis/tie/ti/index.html

1.7 Problem domains

The following presents a short list of common problem domains.

- Clustering – Unsupervised clustering or grouping problems.
- Regression – The network must output a number based on input.
- Classification - The network must classify data points or cases.
- Prediction - The network must predict events in time, such as signals for finance applications.
- Vision – Computer vision problems require the computer to analyse images.
- Optimization - These problems require that the network finds the best ordering or set of values to achieve an objective.

After all, in the mathematical sense we may reduce virtually most neural network computation methods to the concept of optimization. To the point, some error, cost or loss function is then attempted to be minimized.
1.8 Simple examples

The following are more or less "toy examples", but useful to get started. Sir Ronald Fisher (1936) collected the data as an example of discriminant analysis. The data set has become very popular in machine learning. The following URL contains the iris data set with measurements and species information for 150 iris flowers and five features: sepal length, sepal width, petal length, petal width and species. The last one is the class or "label" variable. Petals refer to innermost petals of the iris plant, and sepals to outermost leaves. Here the four input features presented usually as a vector include some real (floating point) values. The class variable is encoded with integers \{1,\ldots,C\}, now \(C=3\). The data is used to test classifiers.

https://archive.ics.uci.edu/ml/datasets/Iris

Sunspots are temporary phenomena on the surface of the sun that appear visibly as dark spots compared to surrounding regions. Intense magnetic activity causes sunspots. Although they occur at temperatures of roughly 2727-4227 °C, the contrast with the surrounding material at about 5500 °C leaves them visible as dark spots. Sunspots appear and disappear with regularity, making them a good data set for time series prediction. The data set contains four variables: the year, month, sunspot count and standard deviation of sunspots observed. See Fig. 1.5.

https://solarscience.msfc.nasa.gov/greenwch/spot_num.txt

The exclusive OR (XOR) operator is a Boolean operator. This sounds extremely simple, but requires a "clever enough" network to be solved correctly. In the following truth table the third column includes the output values:

\[
\begin{array}{c|c|c}
0 & 0 & 0 \\
1 & 0 & 1 \\
0 & 1 & 1 \\
1 & 1 & 0 \\
\end{array}
\]