www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 2 Issue 10 October, 2013 Page No. 2927-2931

.

Artificial Neural Netwok

Lovenish Saluja, Akshay Kanwar, Aditi Khazanchi

Abstract

Artificial neural networks are algorithms that can be used to perform nonlinear statistical modeling and provide a new alternative to logistic regression, the most commonly used method for developing predictive models for dichotomous outcomes in medicine. Neural networks offer a number of advantages, including requiring less formal statistical training, ability to implicitly detect complex nonlinear relationships between dependent and independent variables, ability to detect all possible interactions between predictor variables, and the availability of multiple training algorithms. Disadvantages include its "black box" nature, greater computational burden, proneness to overfitting, and the empirical nature of model development. An overview of the features of neural networks and logistic regression is presented, and the advantages and disadvantages of using this modeling technique are discussed.

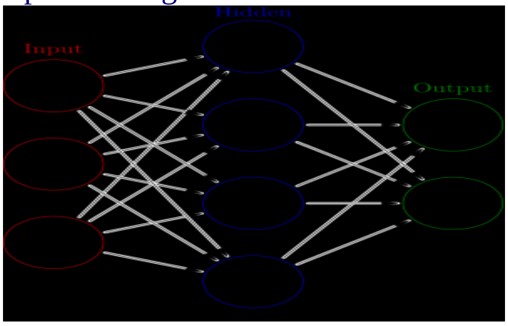
Comment in

1. Introduction

In computer science and related fields, **artificial neural networks** are models inspired by animal central nervous systems (in particular the brain) that are capable of machine learning and pattern recognition. They are usually presented as systems of interconnected "neurons" that can compute values from inputs by feeding information through the network.

For example, in a neural network for handwriting recognition, a set of input neurons may be activated by the pixels of an input image representing a letter or digit. The activations of these neurons are then passed on, weighted and transformed by some function determined by the network's designer, to other neurons, etc., until finally an output neuron is activated that determines which character was read.

Like other machine learning methods, neural networks have been used to solve a wide variety of tasks that are hard to solve using ordinary rule-based programming, including computer vision and speech recognition.



2.History

McCulloch and Pitts[1] (1943) created a computational model for neural networks based on mathematics and algorithms. They called this model threshold logic. The model paved the way for neural network research to split into two distinct approaches. One approach focused on biological processes in the brain and the other focused on the application of neural networks to artificial intelligence.

In the late 1940s psychologist Donald Hebb[2] created a hypothesis of learning based on the mechanism of neural plasticity that is now known as Hebbian learning. Hebbian learning is considered to be a 'typical' unsupervised learning rule and its later variants were early models for long term potentiation. These ideas started being applied to computational models in 1948 with Turing's B-type machines.

Farley and Clark[3] (1954) first used computational machines, then called calculators, to simulate a Hebbian network at MIT. Other neural network computational machines were created by Rochester, Holland, Habit, and Duda[4] (1956).

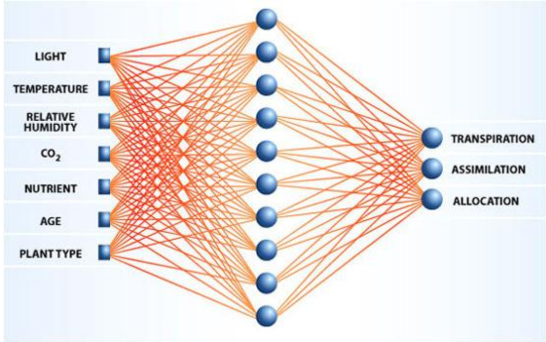
Rosenblatt[5] (1958) created the perceptron, an algorithm for pattern recognition based on a two-layer learning computer network using simple addition and subtraction. With mathematical notation, Rosenblatt also described circuitry not in the basic perceptron, such as the exclusive-or circuit, a circuit whose mathematical computation could not be processed until after thebackpropagation algorithm was created by Werbos[6] (1975).

Neural network research stagnated after the publication of machine learning research by Minsky and Papert[7] (1969). They discovered two key issues with the computational machines that processed neural networks. The first issue was that single-layer neural networks were incapable of processing the exclusive-or circuit. The second significant issue was that computers were not sophisticated enough to effectively handle the long run time required by large neural networks. Neural network research slowed until computers achieved greater processing power. Also key in later advances was the backpropagation algorithm which effectively solved the exclusive-or problem (Werbos 1975).[6]

The parallel distributed processing of the mid-1980s became popular under the name connectionism. The text by Rumelhart and McClelland[8] (1986) provided a full exposition on the use of connectionism in computers to simulate neural processes.

Neural networks, as used in artificial intelligence, have traditionally been viewed as simplified models of neural processing in the brain, even though the relation between this model and brain biological architecture is debated, as it is not clear to what degree artificial neural networks mirror brain function.[9]

In the 1990s, neural networks were overtaken in popularity in machine learning by support vector machines and other, much simpler methods such as linear classifiers. Renewed interest in neural nets was sparked in the 2000s by the advent of deep learning.



3.Real-life applications

The tasks artificial neural networks are applied to tend to fall within the following broad categories:

- Function approximation, or regression analysis, including time series prediction, fitness approximation and modeling.
- Classification, including pattern and sequence recognition, novelty detection and sequential decision making.
- Data processing, including filtering, clustering, blind source separation and compression.
- Robotics, including directing manipulators, prosthesis.
- Control, including Computer numerical control

Application areas include system identification and control (vehicle control, process control, natural resources management), quantum chemistry,[34] game-playing and decision making (backgammon, chess, poker), pattern recognition (radar systems, face identification, object recognition and more), sequence recognition (gesture, speech, handwritten text recognition), medical diagnosis, financial applications (automated trading systems), data mining (or knowledge discovery in databases, "KDD"), visualization and e-mail spam filtering.

Artificial neural networks have also been used to diagnose several cancers. An ANN based hybrid lung cancer detection system named HLND improves the accuracy of diagnosis and the speed of lung cancer radiology.[35] These networks have also been used to diagnose prostate cancer. The diagnoses can be used to make specific models taken from a large group of patients compared to information of one given patient. The models do not depend on assumptions about correlations of different variables. Colorectal cancer has also been predicted using the neural networks. Neural networks could predict the outcome for a patient with colorectal cancer with a lot more accuracy than the current clinical methods. After training, the networks could predict multiple patient outcomes from unrelated institutions.[36]

4.Why Use Neural Nets?

Artificial neural nets have a number of properties that make them an attractive alternative to traditional problem-solving techniques. The two main alternatives to using neural nets are to develop an algorithmic solution, and to use an expert system.

Algorithmic methods arise when there is sufficient information about the data and the underlying theory. By understanding the data and the theoretical relationship between the data, we can directly calculate unknown solutions from the problem space. Ordinary von Neumann computers can be used to calculate these relationships quickly and efficiently from a numerical algorithm.

Expert systems, by contrast, are used in situations where there is insufficient data and theoretical background to create any kind of a reliable problem model. In these cases, the knowledge and rationale of human experts is codified into an expert system. Expert systems emulate the deduction processes of a human expert, by collecting information and traversing the solution space in a directed manner. Expert systems are typically able to perform very well in the absence of an accurate problem model and complete data. However, where sufficient data or an algorithmic solution is available, expert systems are a less than ideal choice.

Artificial neural nets are useful for situations where there is an abundance of data, but little underlying theory. The data, which typically arises through extensive experimentation may be non-linear, non-stationary, or chaotic, and so may not be easily modeled. Input-output spaces may be so complex that a reasonable traversal with an expert system is not a satisfactory option. Importantly, neural nets do not require any a priori assumptions about the problem space, not even information about statistical distribution. Though such assumptions are not required, it has been found that the addition of such a priori information as the statistical distribution of the input space can help to speed training. Many mathematical problem models tend to assume that data lies in a standard distribution pattern, such as Gaussian or Maxwell-Boltzmann distributions. Neural networks require no such assumption. During training, the neural network performs the necessary analytical work, which would require non-trivial effort on the part of the analyst if other methods were to be used.

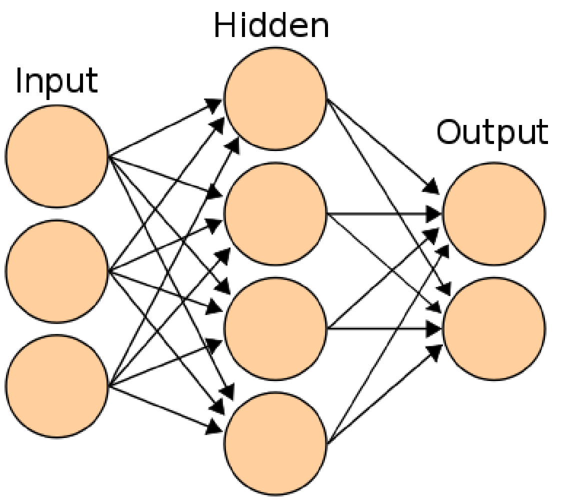
5.Network Parameters

There are a number of different parameters that must be decided upon when designing a neural network. Among these parameters are the number of layers, the number of neurons per layer, the number of training iterations, et cetera. Some of the more important parameters in terms of training and network capacity are the number of hidden neurons, the learning rate and the momentum **parameter.**

Number of neurons in the hidden layer[edit]

Hidden neurons are the neurons that are neither in the input layer nor the output layer. These neurons are essentially hidden from view, and their number and organization can typically be treated as a black box to people who are interfacing with the system. Using additional layers of hidden neurons enables greater processing power and system flexibility. This additional flexibility comes at the cost of additional complexity in the training algorithm. Having too many hidden neurons is analogous to a system of equations with more equations than there are free variables:

the system is over specified, and is incapable of generalization. Having too few hidden neurons, conversely, can prevent the system from properly fitting the input data, and reduces the robustness of the system.



6. Neural networks versus conventional computers

Neural networks take a different approach to problem solving than that of conventional computers. Conventional computers use an algorithmic approach i.e. the computer follows a set of instructions in order to solve a problem. Unless the specific steps that the computer needs to follow are known the computer cannot solve the problem. That restricts the problem solving capability of conventional computers to problems that we already understand and know how to solve. But computers would be so much more useful if they could do things that we don't exactly know how to do.

Neural networks process information in a similar way the human brain does. The network is composed of a large number of highly interconnected processing elements(neurones) working in parallel to solve a specific problem. Neural networks learn by example. They cannot be programmed to perform a specific task. The examples must be selected carefully otherwise useful time is wasted or even worse the network might be functioning incorrectly. The disadvantage is that because the network finds out how to solve the problem by itself, its operation can be unpredictable.

On the other hand, conventional computers use a cognitive approach to problem solving; the way the problem is to solved must be known and stated in small unambiguous instructions. These instructions are then converted to a high level language program and then into machine code that the computer can understand. These machines are totally predictable; if anything goes wrong is due to a software or hardware fault.

Neural networks and conventional algorithmic computers are not in competition but complement each other. There are tasks are more suited to an algorithmic approach like arithmetic operations and tasks that are more suited to neural networks. Even more, a large number of tasks, require systems that use a combination of the two approaches (normally a conventional computer is used to supervise the neural network) in order to perform at maximum efficiency.

Advantages:

- A neural network can perform tasks that a linear program can not.
- When an element of the neural network fails, it can continue without any problem by their parallel nature.
- A neural network learns and does not need to be reprogrammed.
- It can be implemented in any application.
- It can be implemented without any problem.

Disadvantages:

- The neural network needs training to operate.
- The architecture of a neural network is different from the architecture of microprocessors therefore needs to be emulated.

- Requires high processing time for large neural network

Refrences

http://www.learnartificialneuralnetworks.com/introduction-to-neural-networks.html
http://www.learnartificialneuralnetworks.com/
http://en.wikipedia.org/wiki/Artificial_neural_network
http://pages.cs.wisc.edu/~bolo/shipyard/neural/local.html