Perceptrons

PARASTOU KOHVAEI

01.02.2012

ALBERT-LUDWIGS-UNIVERSITÄT FREIBURG
DEPARTMENT OF COMPUTER SCIENCE
CHAIR FOR ALGORITHMS AND DATA STRUCTURES

Outline

Introduction

Perceptron

Learning Perceptrons

Limitations and Enhancements

Applications

Summary

Intro

- Machine learning is a branch of artificial intelligence which is about building computer systems that automatically improve with experience.
- In machine learning, Pattern Recognition is the assignment of a label to a given input value. (i.e., part of speech tagging)
- Classification is one example of Pattern Recognition. (i.e., spam filtering)
- Classification is the algorithmic process of classifying input data into categories according to shared qualities or characteristics.
- Classifier : A system that performs classification by means of a function inferred from a set of training data

Perceptron

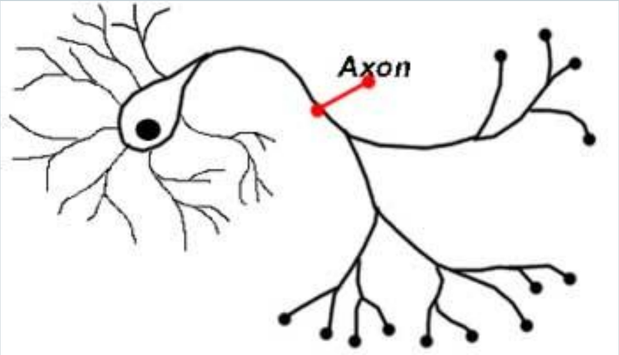
- The simplest type of Artificial Neural Network

- Artificial Neural Networks are mathematical models of Biological Neural Networks.

- Perceptron is a Binary Classifier. It classifies input data into two groups: “True” or “False”.

History and Origins

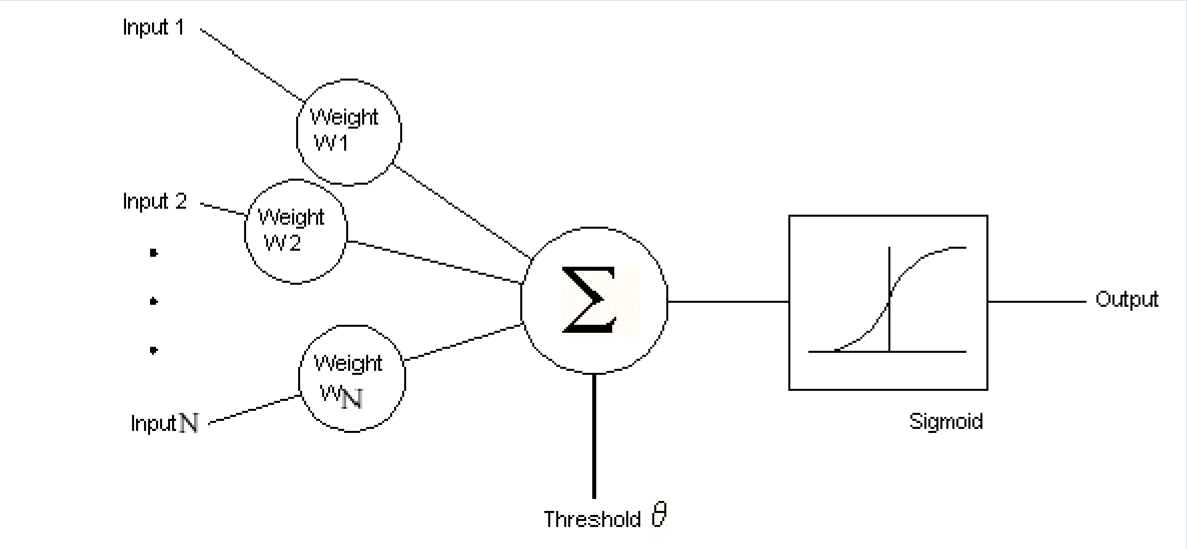
- Rooted from works in Neurological Science field (early 20th century) and a simple model of a nerve cell



- First introduced in a paper in 1943
- Completed and presented in 1957 at the Cornell Aeronautical Laboratory by Frank Rosenblatt

Structure

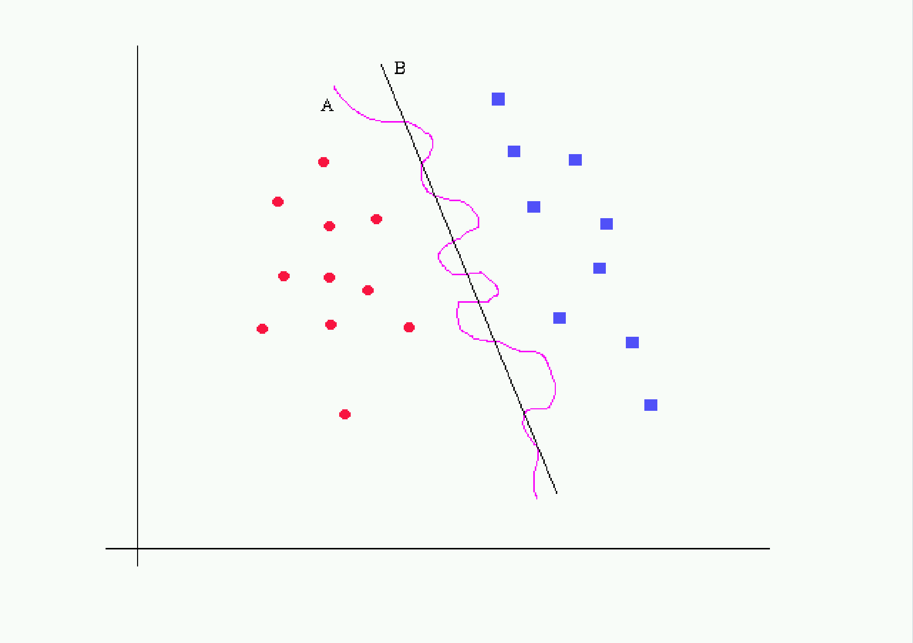
- Multiple inputs, each input line having it’s own weight



- An activation function, which activates the perceptron and forms the output; usually step function or a threshold function like Sigmoid function.

What Can Perceptrons Do?

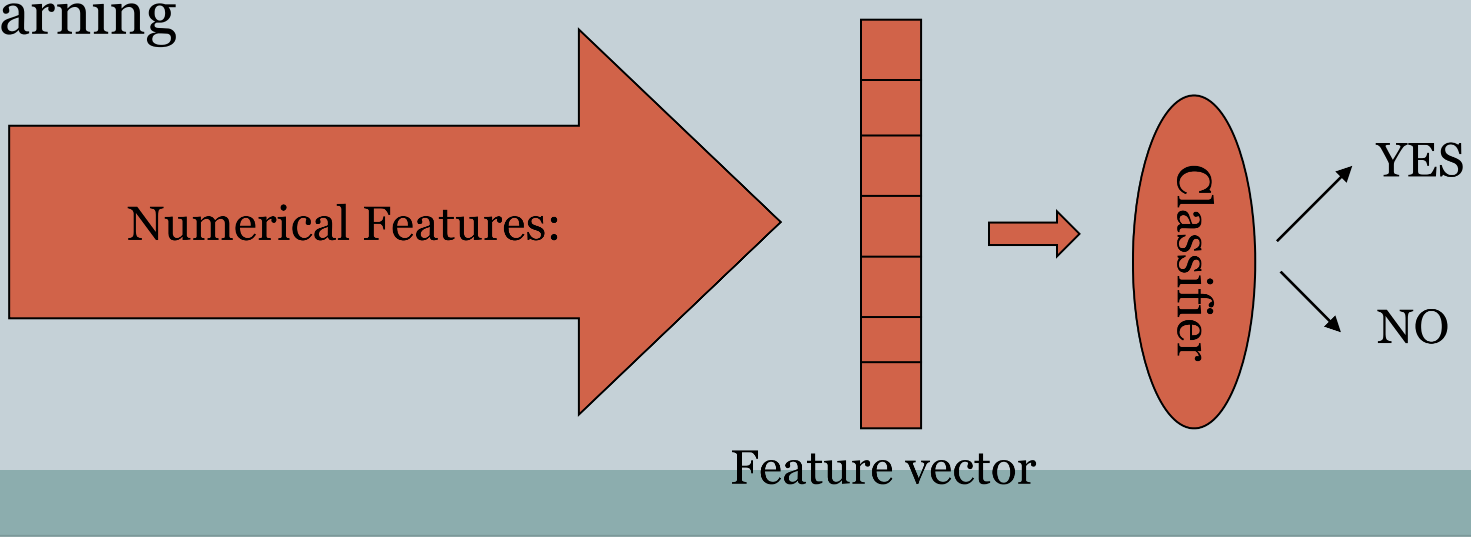
- Perceptrons are linear classifiers.
- Linearly separable sets can be separated by straight lines.



Feature Vectors

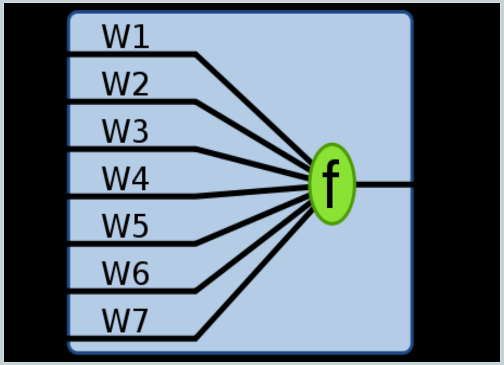
- Feature : Any distinctive aspect, quality, or characteristic of an entity

  Example : Symbolic (i.e. color) or numeric (i.e. height)
- Feature Vector : The combination of numerical features in an n-dimensional vector
- Used in many algorithms of pattern recognition and machine learning



Learning Perceptrons

- N-dimensional feature vector as input
- Having w as weights vector and x as feature vector and $w.x$ as the dot-product of the two :



$$f(x) = \begin{cases} 1 & if\ w.x + b > 0 \\ 0 & otherwise \end{cases}$$

- Original Perceptrons use step function as activation function.

Learning Algorithm

- A 3-phase algorithm: Initialization, Iteration, Termination
- Different learning algorithms according to the way the output of the Perceptron is used for modifying the weights
- As the simplest method, learning algorithm uses the output of the step function.

Initialization

- Randomly chosen weights ($w(i)$)

- Training set consisting of feature vectors ($x(i)$)

- A constant bias value, this value can change the position of the separating line (b)

- Relatively low learning rate, it is the modification rate for weights. (α)

Iteration

- Weights and bias are modified for a given vector over iteration

$$w(i) = w(i) + \alpha[actual\ output - desired\ output] * x(i)$$

$$b = b + \alpha[actual\ output - desired\ output]$$

- This repeats for every member in the training set

Termination

- Weights do not change.
- The perceptron classifies correctly.

Note : Convergence guaranteed for linearly separable sets

Example: NAND Gate

- A perceptron learning the binary NAND
  - x_0 and x_1 and x_2 as input, with x_0 held constant at 1
  - Bias = 0
  - Threshold = 0.5
  - Learning Rate = 0.1
  - Training set {((0,0),1),((0,1),1),((1,0),1),((1,1),0)}

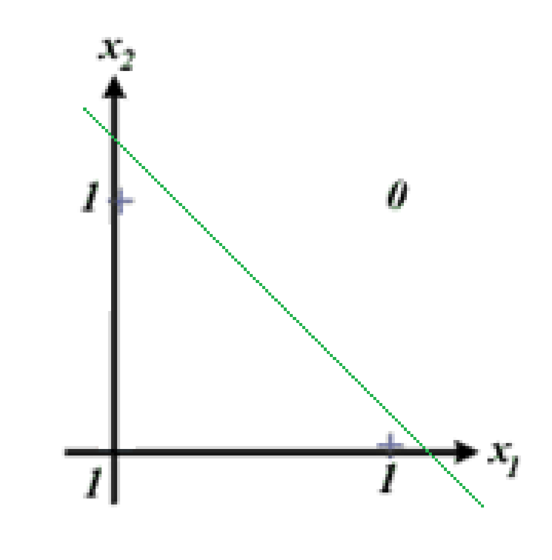
Learning Table

Input				Initial Weights			Output	Error	Final Weights		
x_0	x_1	x_3	Desired Output (z)	w_0	w_1	w_2	Network (n)	z - n	w_0	w_1	w_2
1	1	0	1	0	0	0	0	1	0.1	0	0
1	0	1	1	0.1	0	0	0	1	0.2	0	0.1
1	1	0	1	0.2	0	0.1	0	1	0.3	0.1	0.1
1	1	1	0	0.3	0.1	0.1	0	0	0.3	0.1	0.1
… after 9 iterations …											
1	**1**	**0**	**1**	**0.8**	**-0.2**	**-0.1**	**1**	**0**	**0.8**	**-0.2**	**-0.1**
1	**1**	**1**	**0**	**0.8**	**-0.2**	**-0.1**	**0**	**0**	**0.8**	**-0.2**	**-0.1**
1	**0**	**0**	**1**	**0.8**	**-0.2**	**-0.1**	**1**	**0**	**0.8**	**-0.2**	**-0.1**
1	**0**	**1**	**1**	**0.8**	**-0.2**	**-0.1**	**1**	**0**	**0.8**	**-0.2**	**-0.1**

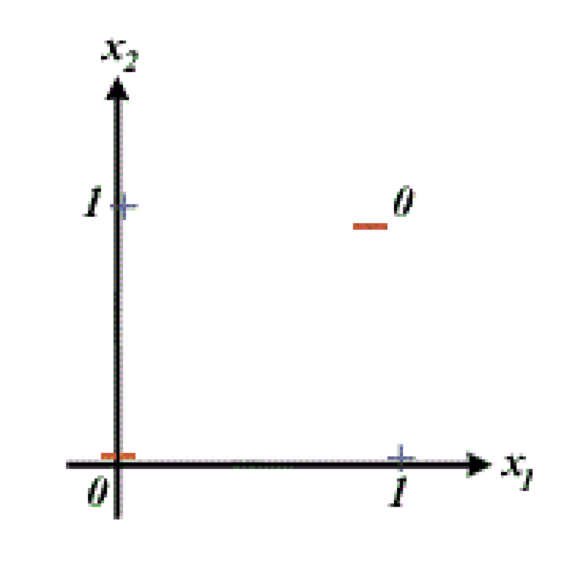
Limitations And Enhancements

- Limited output values (True or False)
- Not applicable to non-linearly separable entities
- Most famous example: boolean XOR

Graph Comparison of NAND and XOR



NAND on two inputs



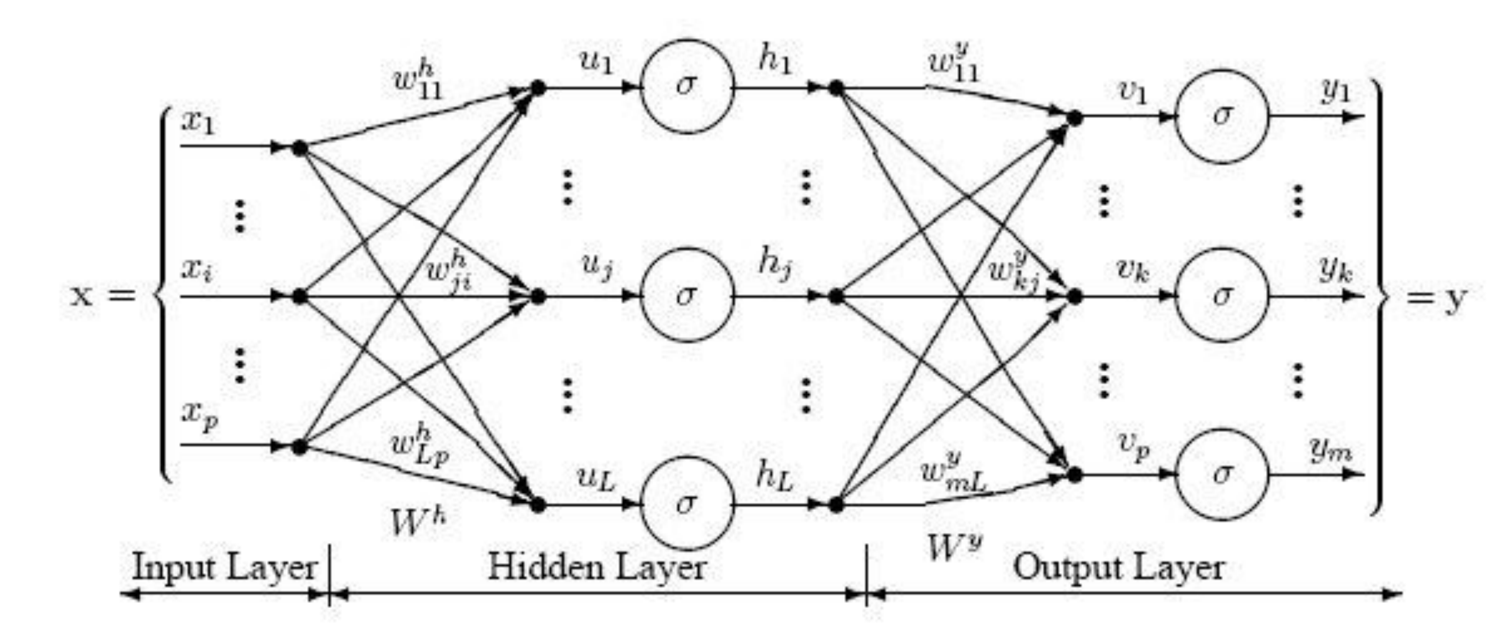
XOR on two inputs

Evolution to MLPs

- Introduced in 1980’s as a development of the perceptron
- They contain middle hidden layers of processing units.
- Hidden layers detect specific features in the input.
- Capable of solving more challenging problems in AI

Multi-layer Perceptron

- Each layer is fully connected to the next one.



- Nodes have their own weight vector and activation function.

MLP Construction

- Multiple layers and layer units based on the classification function complexity

- Problem-specific number of input and output units

- None-linear and differentiable activation functions

Learning MLPs

- Backpropagation mostly used as the learning algorithm
- Two phases: Propagation and Weight Update
    - Forward and backward propagation of training values and output respectively
    - Calculating and adjusting weight gradients, e.g. the direction in which the weight is growing, to reduce error
- Iteration over two phases

Applications

- MLPs are used in Natural Language Processing, Image and Speech Recognition and Cyber Security.
- Currently they are used in research areas which involve complex approximations.
- They are themselves subject of ongoing research in Computational Neuroscience and Parallel Distributed Processing.

Summary

- Perceptrons are simple algorithms widely used in classification problems.
- Multilayer perceptrons are extensions which can recognize more complex patterns.
- Perceptrons have opened the way for a myriad of applications in artificial intelligence.
- Their impact has been remarkable.

THANK YOU!

Main References

- Estebon, M. (Spring 1997).Perceptrons: An Associative Learning Network. Virginia Tech CS 3604. Retrieved from

http://www.ccs.fau.edu/~bressler/EDU/CompNeuro/Resources/Estebon_Perceptrons.pdf

- History of the Perceptron.(2012, Jan 31). Retrieved from

http://www.csulb.edu/~cwallis/artificialn/History.htm

- Wikipedia.(2012, Jan 27).Multi Layer Perceptron. Retrieved from

http://en.wikipedia.org/wiki/Multilayer_perceptron

- Wikipedia.(2012, Jan 25).Perceptron. Retrieved from

http://en.wikipedia.org/wiki/Perceptron

- Weisman, O. & Pollak, Z. (1995, Aug 13). Retrieved from

http://www.cs.bgu.ac.il/~omri/Perceptron/

- Kuro5hin.(2003, Nov 12).Perceptrons: Intro to machine Learning. Retrieved from

http://www.kuro5hin.org/story/2003/11/11/17383/475

Appendix A

- The perceptron applet:

http://www.eee.metu.edu.tr/~alatan/Courses/Demo/AppletPerceptron.html