Seminar in Deep Learning

Lecture 0: Introduction

Alexander Tkachenko
University of Tartu

9 September, 2014

Today’s topics

- Machine Learning
- Neural Networks
- Deep Learning

Why machine learning?

- It is very hard to write programs that solve problems like recognizing a three-dimensional object from a novel viewpoint in new lighting conditions in a cluttered scene.

- It is hard to write a program to compute the probability that a credit card transaction is fraudulent.

The machine learning approach

- **Definition** Machine Learning is a field of study that gives computers the ability to learn without being explicitly programmed [Arthur Samuel,1959]

- Instead of writing a program by hand for each specific task, we collect lots of examples that specify the correct output for a given input.
- A machine learning algorithm then takes these examples and produces a program that does the job.
- Massive amounts of computation are now cheaper than paying someone to write a task-specific program.

Some examples of tasks best solved by learning

- Recognizing patterns:
  - Objects in real scenes
  - Facial identities or facial expressions
  - Spoken words

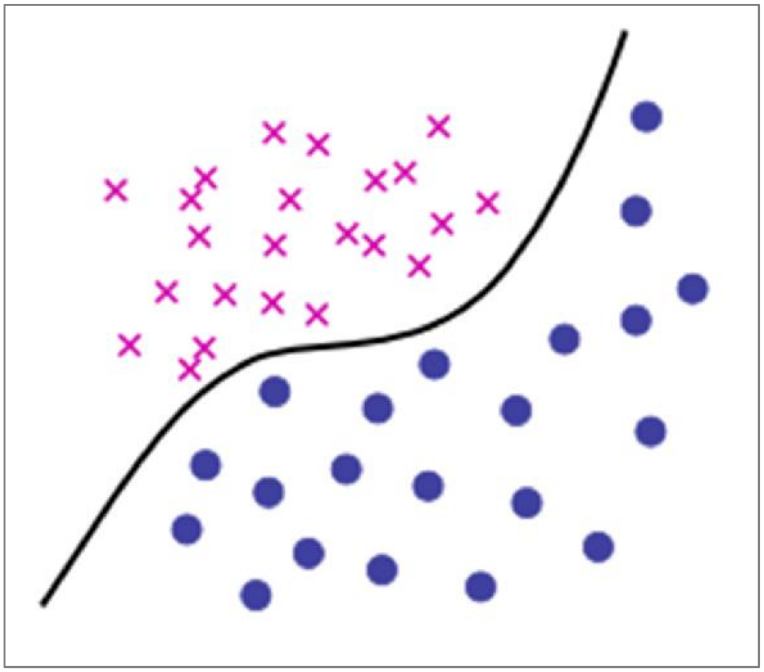
- Recognizing anomalies:
  - Unusual sequences of credit card transactions
  - Unusual patterns of sensor readings in a nuclear power plant

- Prediction:
  - Future stock prices or currency exchange rates
  - Which movies will a person like?

Types of learning tasks

- Supervised learning
  - Learn to predict an output when given an input vector.
  - Each training example consists of an input vector *x* and a target output *t*.
- Unsupervised learning
  - Discover a good internal representation of the input
- Others:
  - Reinforcement learning, recommender systems

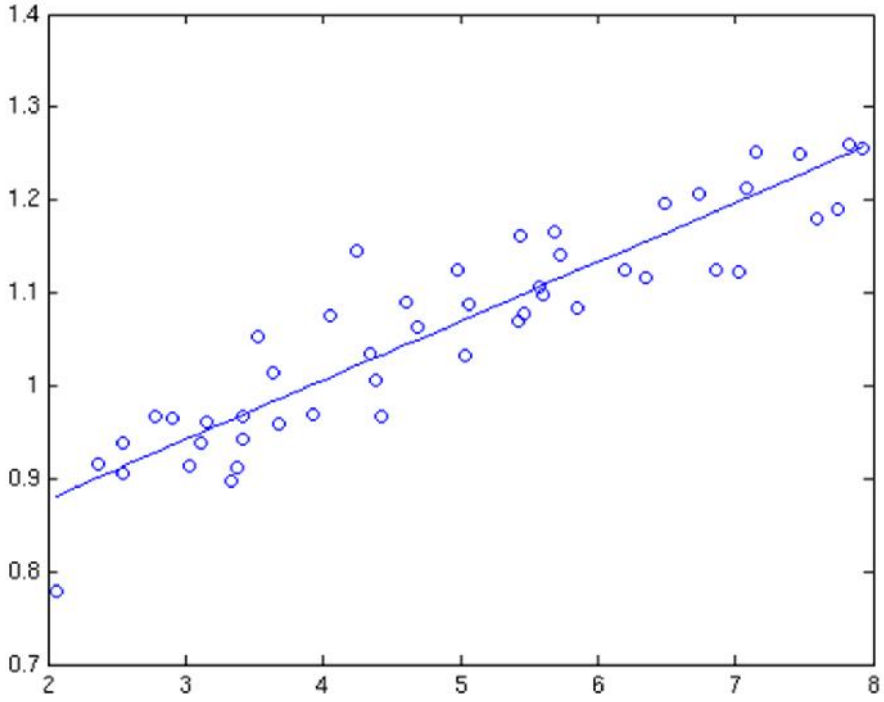
Supervised learning: Classification



Predict a discrete class label

- The simplest case is a choice between 1 and 0.
- We can also have multiple alternative labels

Supervised learning: Regression



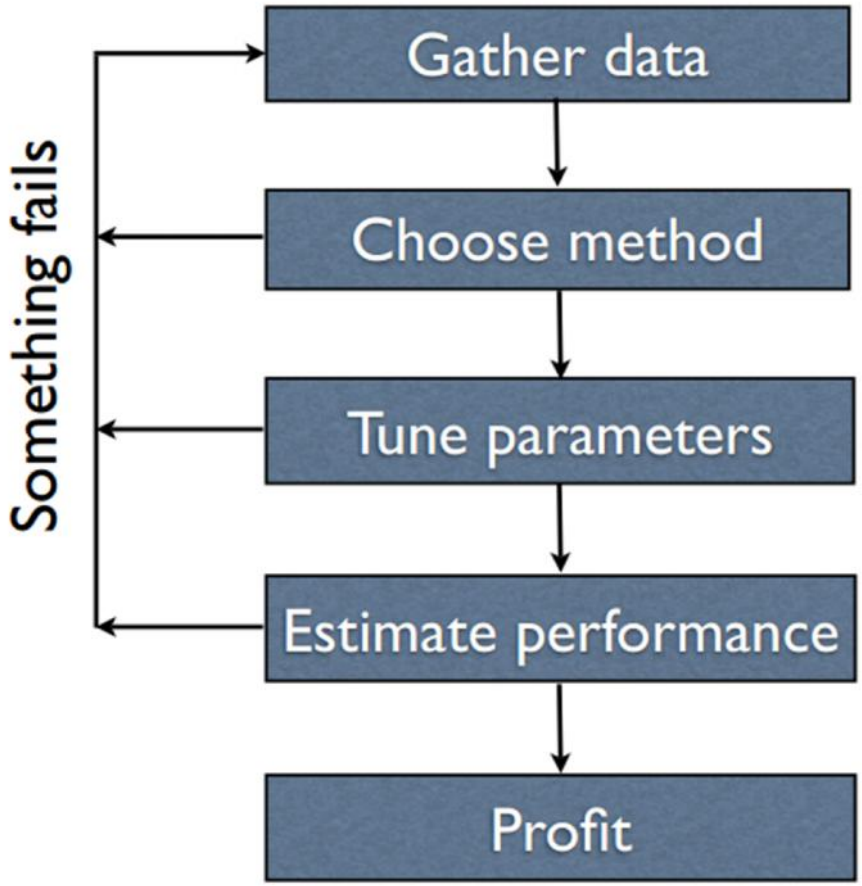
Predict continuous valued output

- The price of a stock in 6 months time
- The temperature at noon tomorrow

How supervised learning typically works

- We start by choosing a model-class:
  - A model-class, f, is a way of using some numerical parameters W, to map each input vector, x, into a predicted output y.
- Learning usually means adjusting the parameters to reduce the discrepancy between the target output, t, on each training case and the actual output, y, produced by the model.
  - For regression, is often a sensible measure of the discrepancy.
  - For classification there are other measures that are generally more sensible (they also work better).

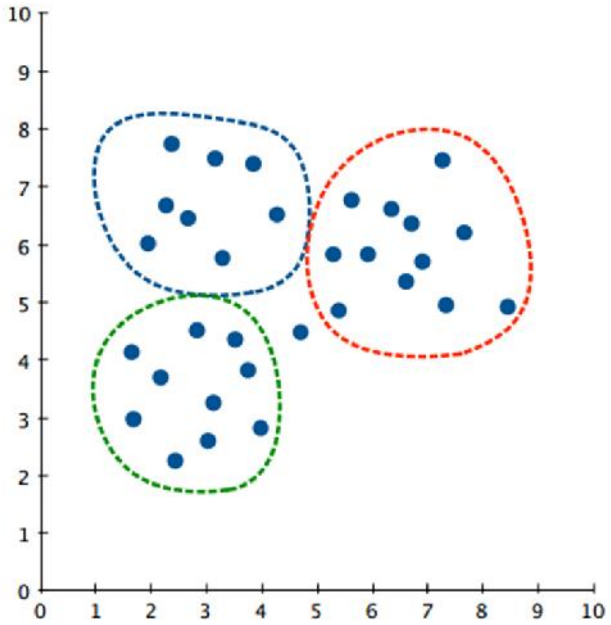
How supervised learning typically works



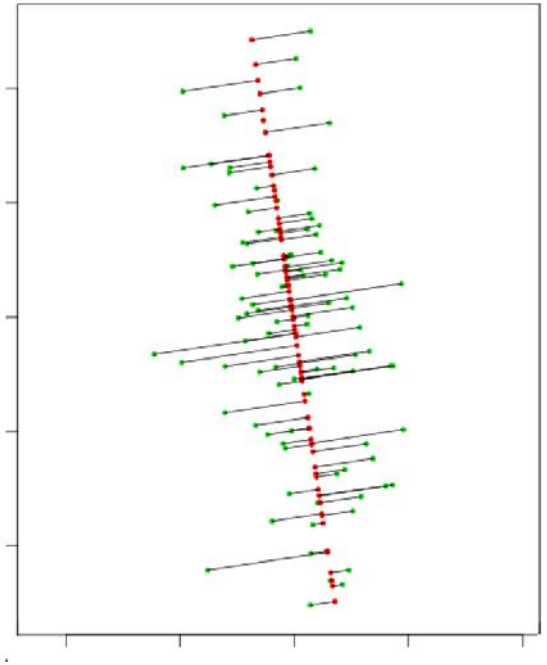
Unsupervised learning

- For about 40 years, unsupervised learning was largely ignored by the machine learning community
  - Some widely used definitions of machine learning actually excluded it.
  - Many researchers thought that clustering was the only form of unsupervised learning.
- It is hard to say what the aim of unsupervised learning is.
  - One major aim is to create an internal representation of the input that is useful for subsequent supervised learning.
  - You can compute the distance to a surface by using the disparity between two images. But you don't want to learn to compute disparities by stubbing your toe thousands of times.

Unsupervised learning



Clustering



Dimensionality reduction

In the context of deep learning, the aim is to create an internal representation of the input that is useful for subsequent supervised learning.

Other goals for unsupervised learning

- It provides a compact, low-dimensional representation of the input.
  - High-dimensional inputs typically live on or near a low-dimensional manifold (or several such manifolds).
  - Principal Component Analysis is a widely used linear method for finding a low-dimensional representation.
- It provides an economical high-dimensional representation of the input in terms of learned features.
  - Binary features are economical.
  - So are real-valued features that are nearly all zero.
- It finds sensible clusters in the input.
  - This is an example of a very sparse code in which only one of the features is non-zero

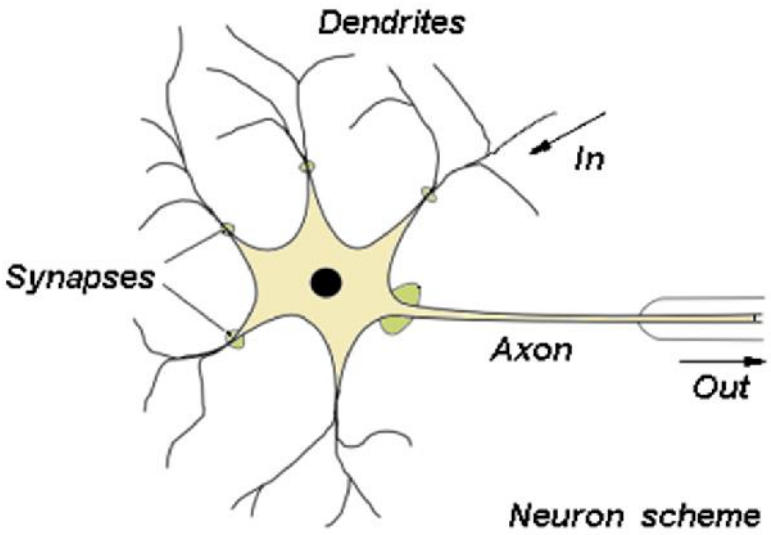
Neural Networks

- Inspired by our understanding of how the brain learns
- Powerful tool for addressing typical machine learning tasks such as regression and classification
- Perform exceptionally well in speech recognition and object detection in images

Reasons to study neural computation

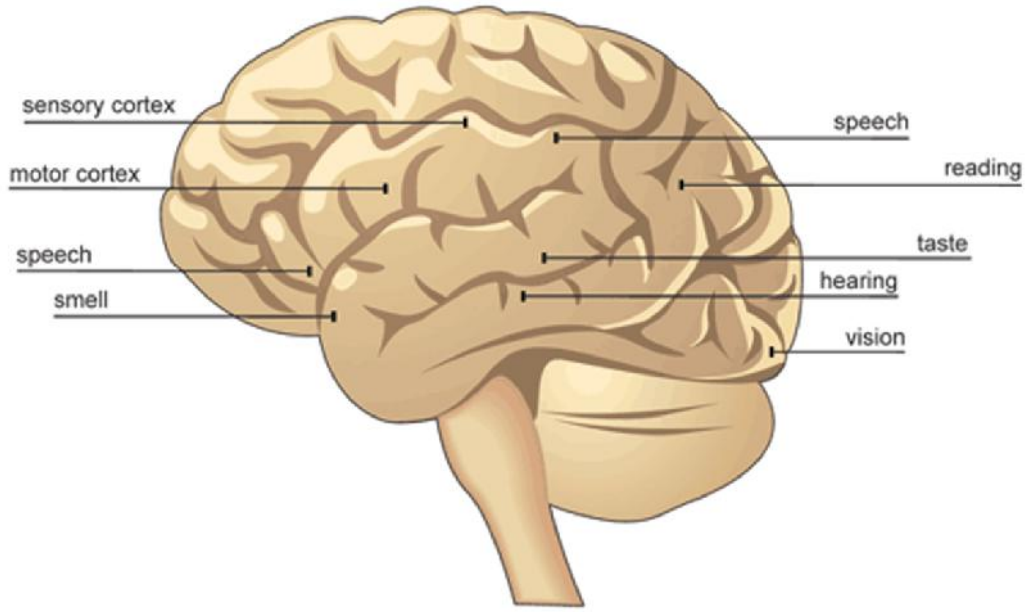
- To understand how the brain actually works.
  - Its very big and very complicated and made of stuff that dies when you poke it around. So we need to use computer simulations.
- To understand a style of parallel computation inspired by neurons and their adaptive connections.
  - Very different style from sequential computation.
    - should be good for things that brains are good at(e.g. vision)
    - Should be bad for things that brains are bad at (e.g. 23 x 71)
- To solve practical problems by using novel learning algorithms inspired by the brain (this course)
  - Learning algorithms can be very useful even if they are not how the brain actually works.

How the brain works



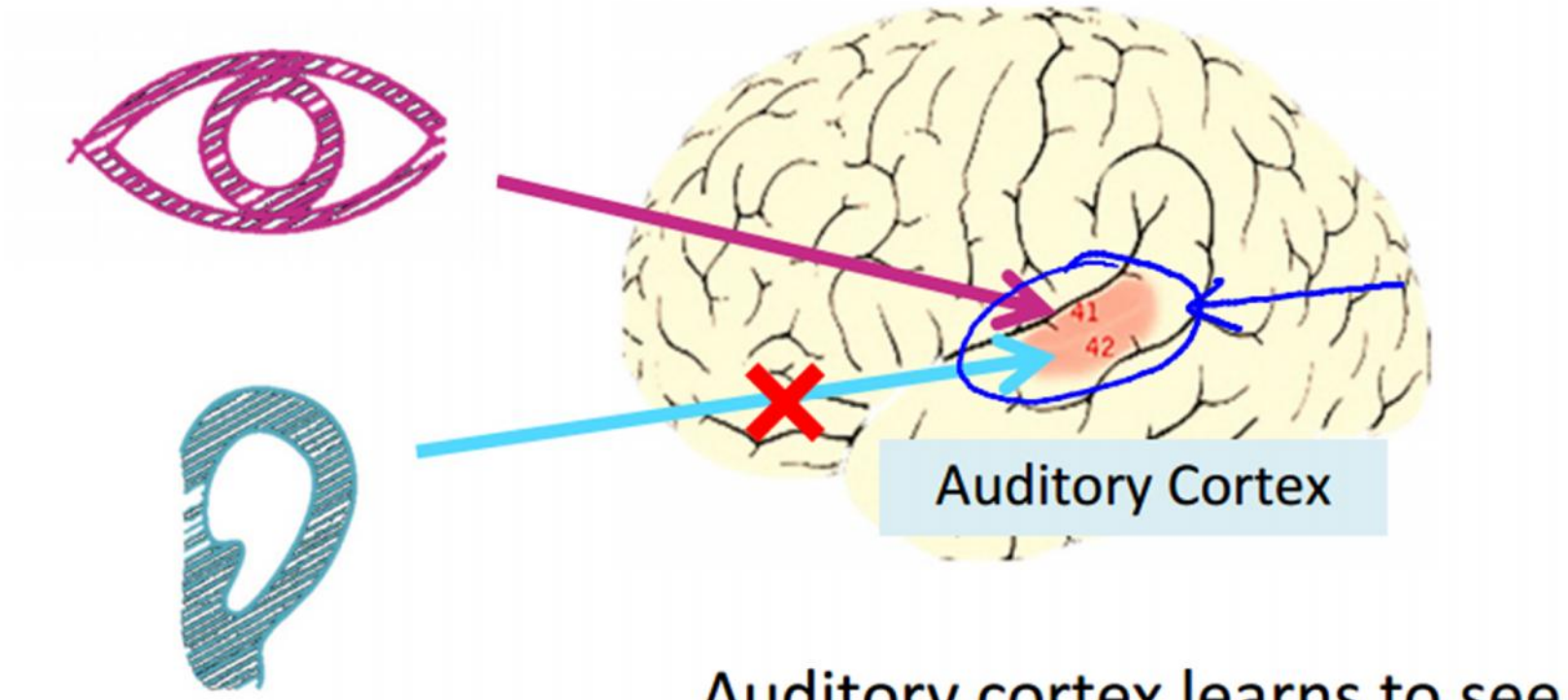
- Each neuron receives inputs from other neurons
- The effect of each input line on the neuron is controlled by a synaptic weight
- The synaptic weights adapt so that the whole network learns to perform useful computations
- There are about 10^11 neurons each with about 10^4 weights.

How the brain works



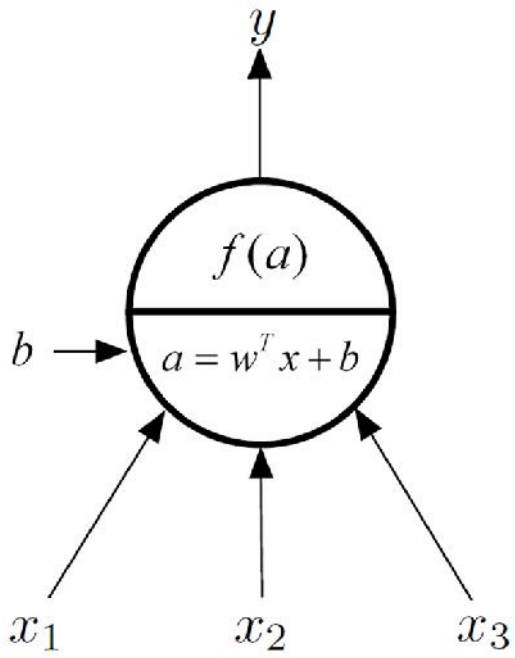
- Different bits of the cortex do different things.
- But cortex looks pretty much the same all over.

The “one learning algorithm” hypothesis



Auditory cortex learns to see

Neuron model



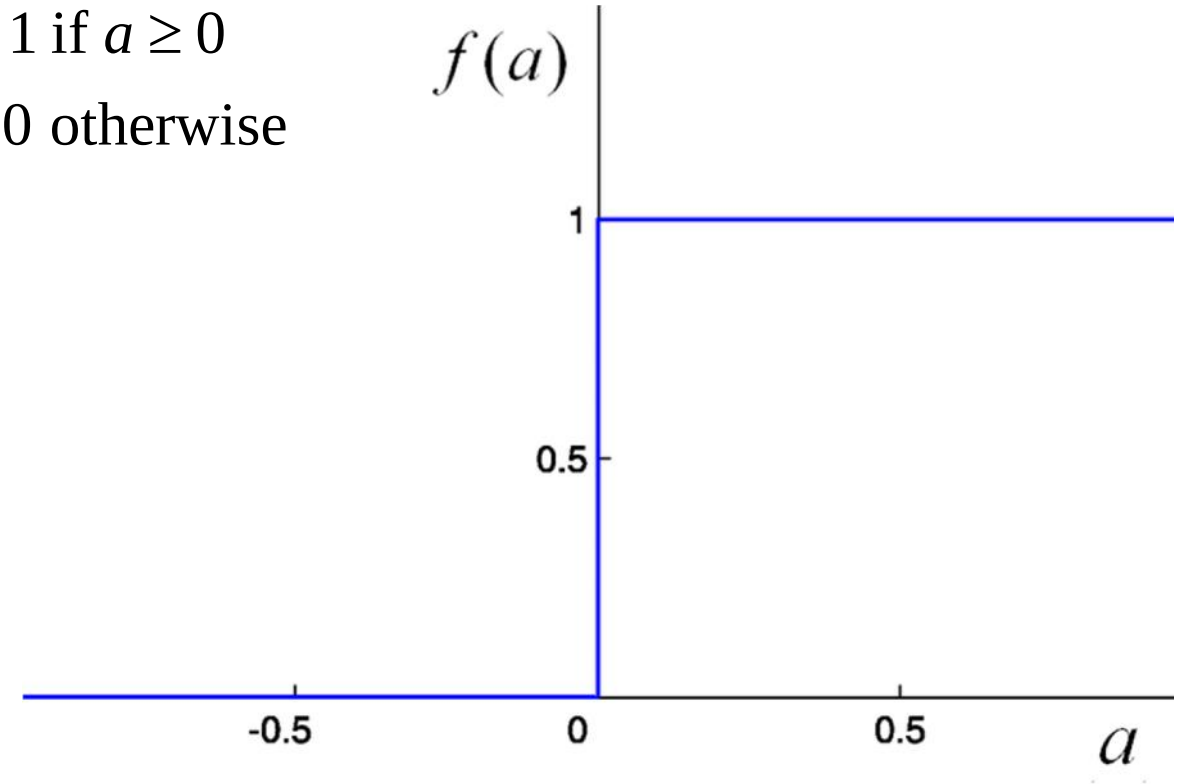
Let $x = (x_1, x_2, x_3)$ be input vector, $w = (w_1, w_2, w_3)$ be weights vector, and b - bias term.

First inputs are linearly aggregated: $a = x_1 w_1 + x_2 w_2 + x_3 w_3 + b$.

Then the output y is obtained as: $y = f(a)$.

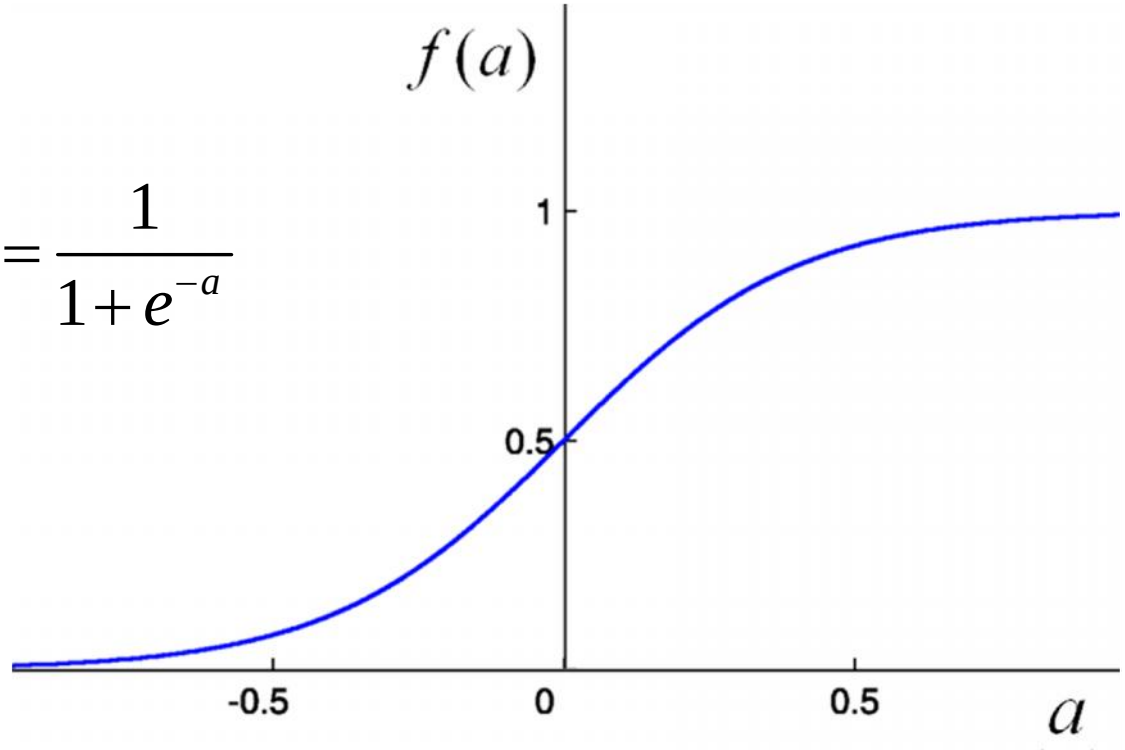
Classification: Binary threshold neuron

$$f(a) = \begin{cases} 1 \text{ if } a \geq 0 \\ 0 \text{ otherwise} \end{cases}$$



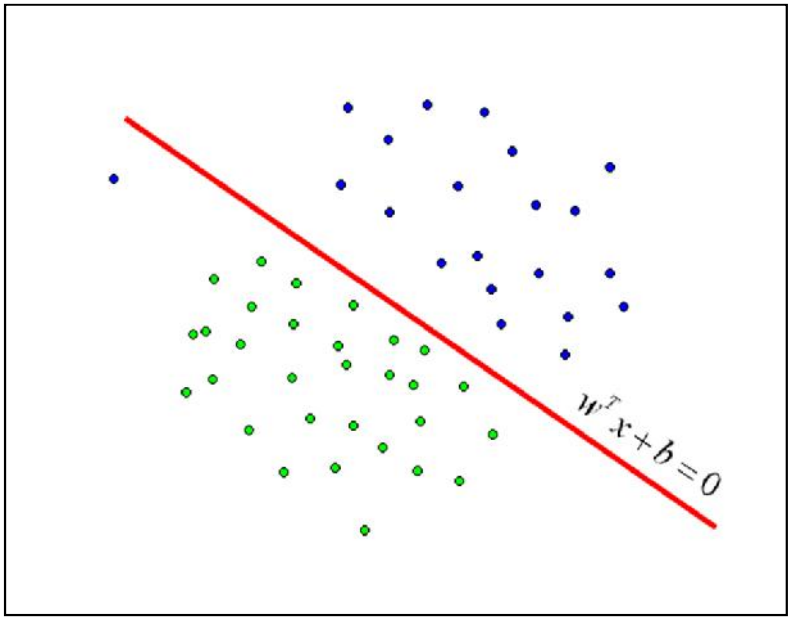
Classification: Sigmoid neurons

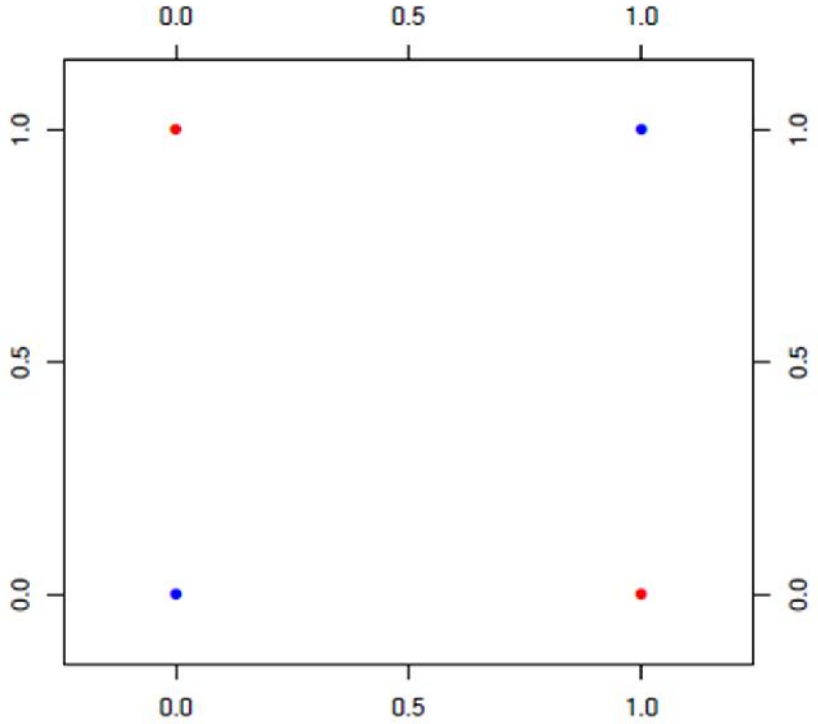
$$f(a) = \frac{1}{1+e^{-a}}$$



- These give a real-valued output that is a smooth and bounded function.
- They have nice derivatives which make learning easy.

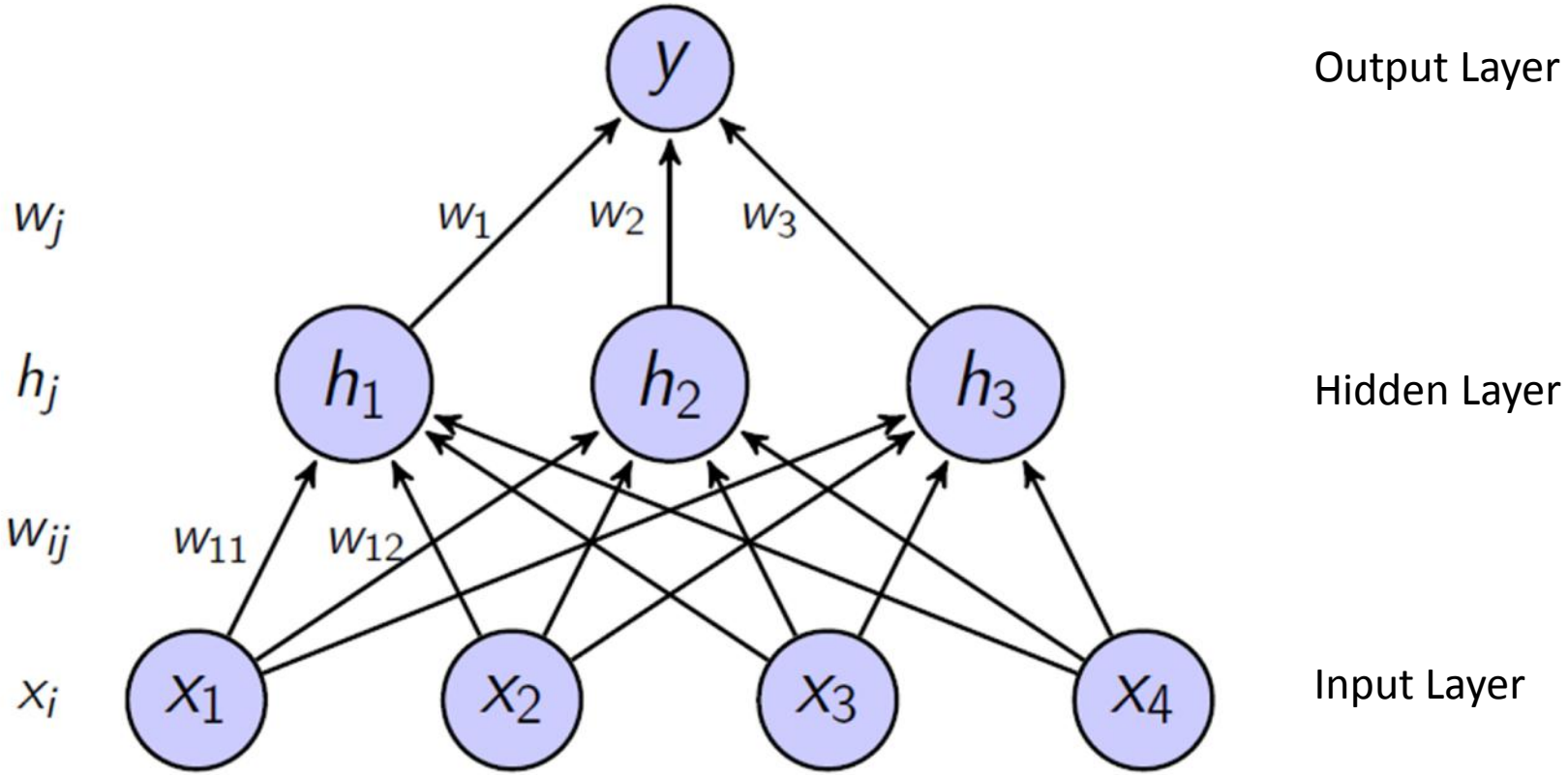
Limitations of a single neuron network





- A decision border of a single sigmoid neuron is a straight line.
- Sigmoid neuron cannot learn XOR.

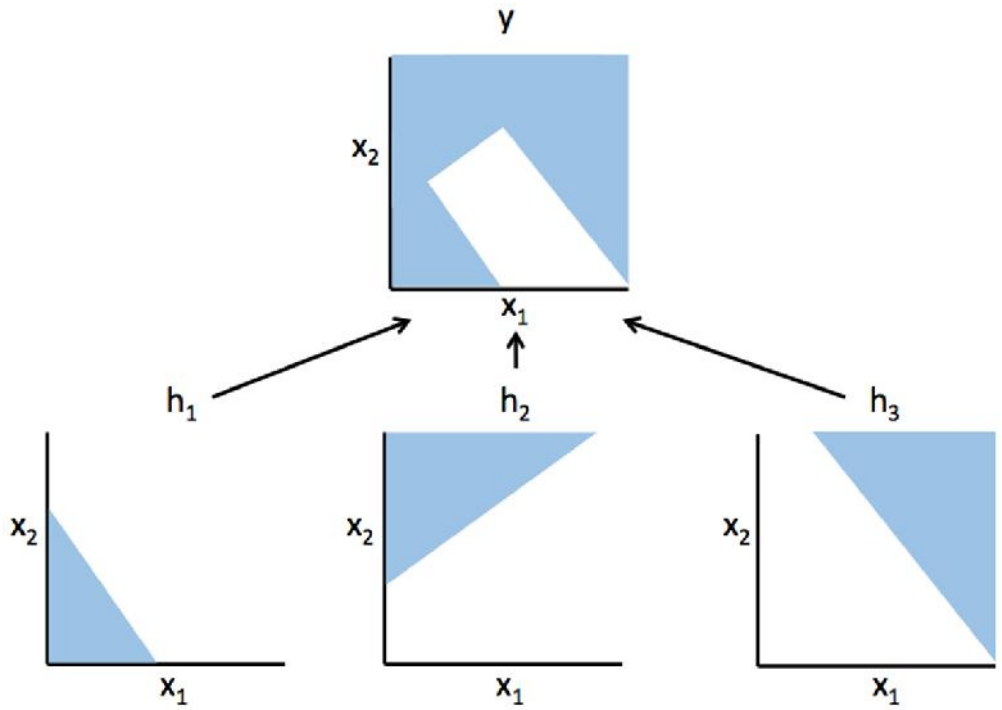
Multilayer Neural Networks



$$y = f(\sum_j w_j h_j) = f(\sum_j w_j f(\sum_i x_j w_{ij}))$$

Multilayer Neural Networks

- Deeper architecture is more expressive than a shallow one
  - 1-layer nets only model linear hyperplanes
  - 2-layer nets are universal function approximators: given infinite hidden nodes, it can express any continuous function.



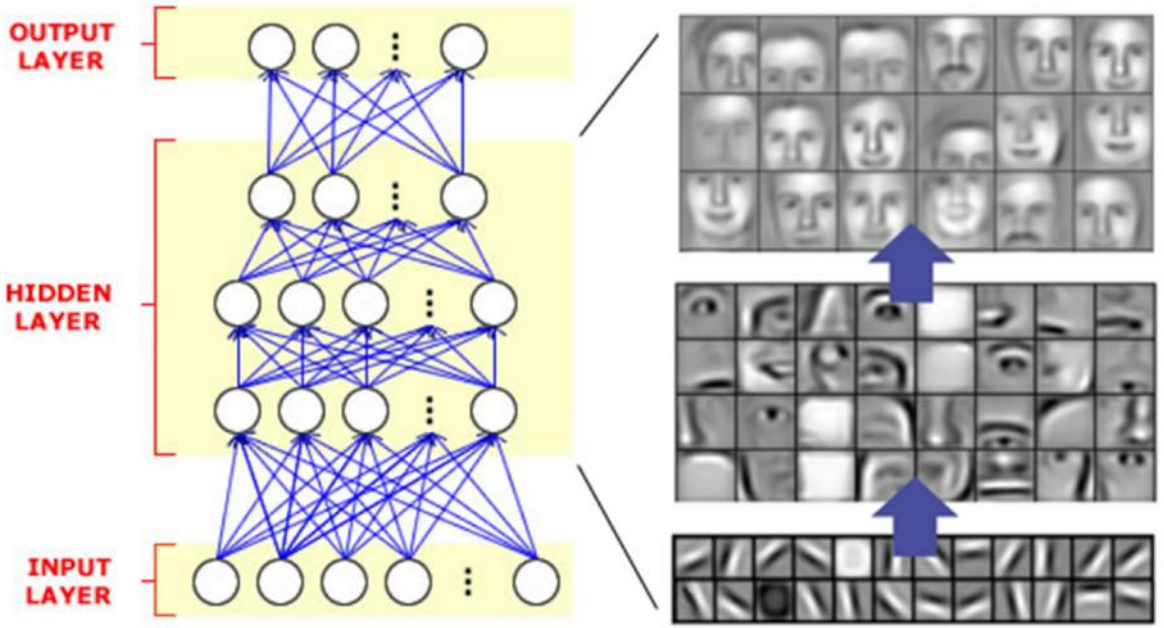
Layer 2

Layer 1

What is deep learning?

A family of methods that uses **deep architectures** to learn **high-level feature representations**

and using these representations to perform typical machine learning tasks such as classification and regression.



Deep learning approach

- Deep architectures are often based on neural networks
  - Contructed by stacking layers of neurons to achieve more abstract feature representations.

- Commonly use unsupervised layer-wise pre-training
  - Restricted Boltzmann Machines
  - Autoencoders

History

- **Early days of AI**. Invention of artificial neuron [McCulloch and Pitts, 1943] & perceptron [Rosenblatt, 1958]
- **AI Winter**. [Minsky and Papert, 1969] showed perceptron only learns linearly separable concepts
- **Revival in 1980s:** Multi-layer Perceptrons (MLP) and Back-propagation [Rumelhart et al., 1986]
- **Other directions** (1990s - present): SVMs, Bayesian Networks
- **Revival in 2006**: Deep learning [Hinton et al., 2006]
- **Successes in applications**: Speech at IBM/Toronto [Sainath et al., 2011], Microsoft [Dahl et al., 2012]. Vision at Google/Stanford [Le et al., 2012]

Results

Currently deep learning systems are state of the art in fields:

- Automatic speech recognition
  - Currently used in android
- Image classification
- Natural Language processing
  - Language modeling

Other inspiring applications

- Playing atari games (DeepMind)
- Recognising cats on Youtube (Google)
- Speech Recognition for the Spoken, Translated Word (MicrosoftResearch)

References

- McCulloch, W. S. and Pitts, W. H. (1943). A logical calculus of the ideas immanent in nervous activity. In Bulletin of Mathematical Biophysics, volume 5, pages 115–137.
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 65:386–408.
- Minsky, M. and Papert, S. (1969). Perceptrons: an introduction to computational geometry. MIT Press.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. Nature, 323:533–536.
- Hinton, G., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. Neural Computation, 18:1527–1554.

Materials

- Andrew Ng's machine learning course
  https://www.coursera.org/course/ml
- Feed-forward neural networks for prediction tasks, Machine Learningm 2012, Sven Laur
  https://courses.cs.ut.ee/MTAT.03.227/2013_spring/uploads/Main/lecture-6.pdf
- Deep Learning and Neural Networks, Kevin Duh, January 2014
  http://cl.naist.jp/~kevinduh/a/deep2014/