Cognitive Algorithms
Lecture 2

Neurons - Computational Units of Cognition

Klaus-Robert Müller, Wojciech Samek
Stephanie Brandl

Berlin Institute of Technology
Dept. Machine Learning

Summary Lecture 1

- Cognitive processes:
  Perception, recognition of and inference on semantic concepts
- Cybernetics
  - simple models of cognition
  - inspired by biological organisms
  - modeled motion detection, navigation, ...
  - $\rightarrow$ but what about higher cognitive functions?
- Artificial intelligence
  - took over ideas from Cybernetics
  - focused on (biologically inspired) models of higher cognition
  - Old AI: rule based systems (e.g. *Eliza*)
  - New AI (machine learning): learns rules from data

Overview

Cognitive functions have to be investigated on 3 levels [Marr, 1982]

- Computational Level

  What does a cognitive function do?

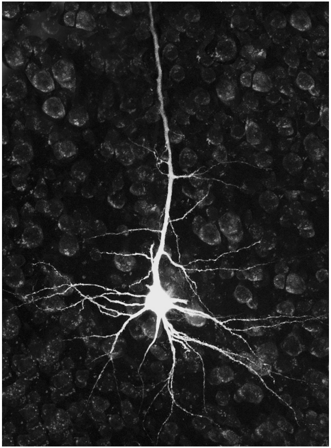
- Algorithmic Level

  What is the functional organization within a cognitive module?

- Implementational Level

  What is the physical/physiological realization of this algorithm?

Biological Neurons



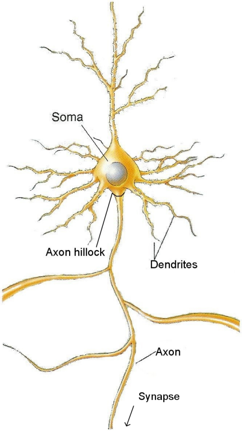
Cortical Neuron

Neurons are electrically charged cells (-50 to -70mV)

They process information by changes in membrane potential

Human brain has 10^{11} neurons and 10^{14} synapses

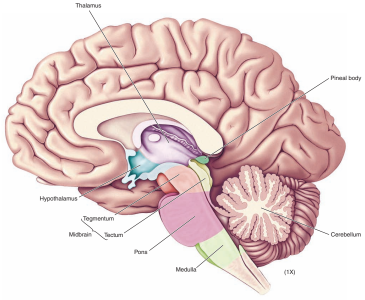
The prototypical neuron

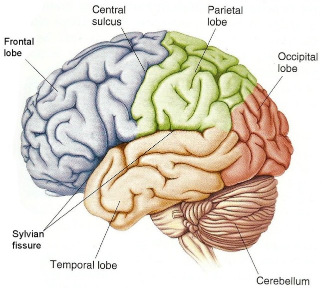


Parts of a neuron

Part	Function
Dendrites	Receive incoming 'messages' from other neurons.
Soma	Combines all incoming 'messages'
Axon hillock	If the membrane potential at the axon hillock reaches a threshold value, the axon 'fires' an action potential.
Axon	Carries the action potential over short (<1mm) or long (>1m) distances.

The structure of the human brain

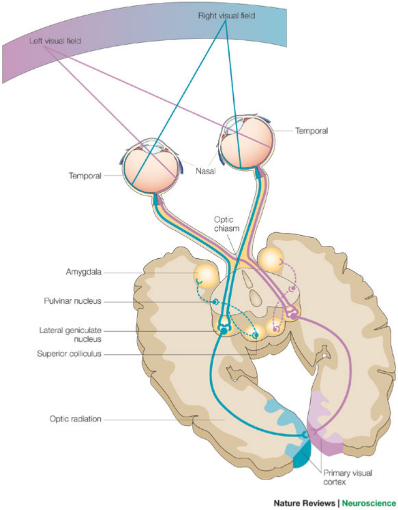




Rough cortilcal division

Frontal lobe:	executive functions, motor areas
Parietal lobe:	secondary visual perception, sensory areas
Temporal lobe:	memory (hippocampus), emotion (amygdala)
Occipital lobe:	primary visual perception

Biological Neurons in the early visual system

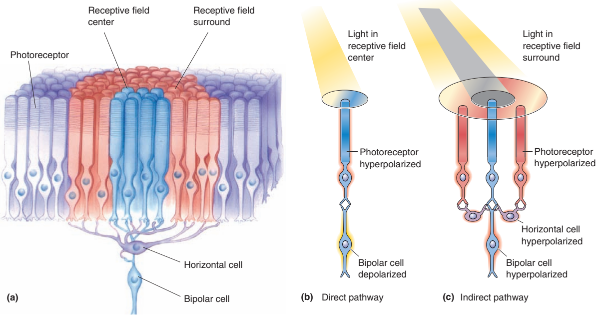


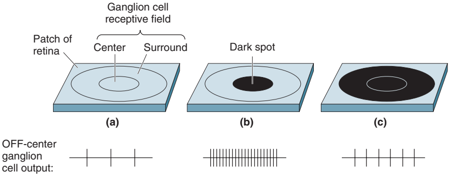
Information processing pathway in the early visual system

Functional properties of neurons are defined by stimulus-response characteristics

Much of what we know about our cognitive functions is based on neuroscientific studies

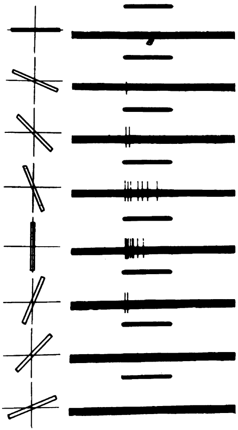
Visual System: Retina





Visual System: Retina

Visual System: Primary Visual Cortex

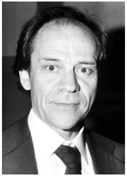


Neuron response in primary visual cortex
$\rightarrow$ Oriented edge detector

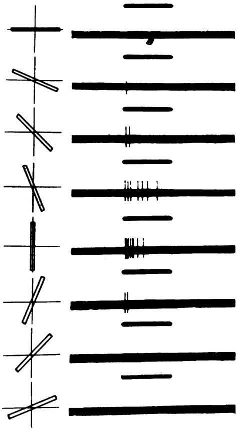
Oriented bar of light runs over visual field while neural activity is recorded from cells in *primary visual cortex*

$\rightarrow$ Neurons respond preferentially to oriented bars [Hubel and Wiesel, 1959]

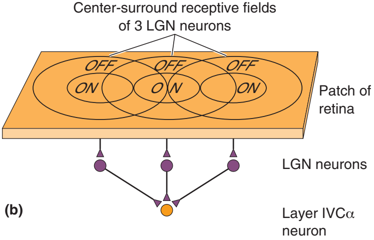
In 1981 Hubel and Wiesel received the Nobel Prize.



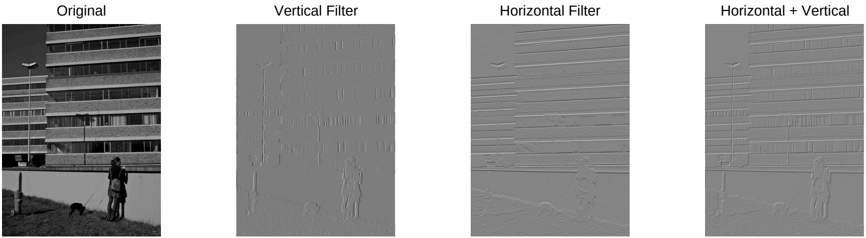
Visual System: Primary Visual Cortex



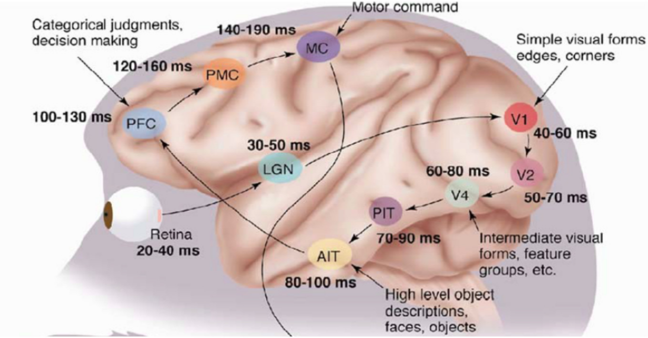
Neuron response in primary visual cortex
→ Oriented edge detector



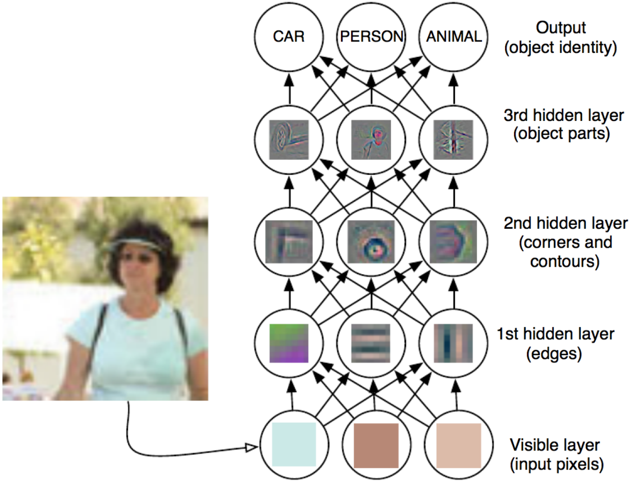
What Primary Visual Cortex Sees
Spatial Frequency Decomposition of Images



Visual System



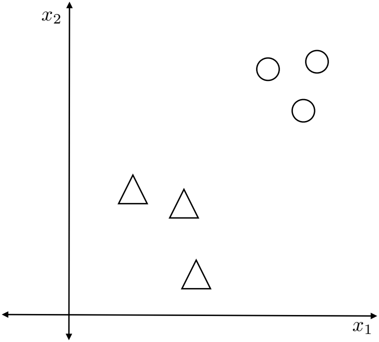
Visual System = Deep Learning ?



Prototypes: Psychological Models of Abstract Ideas

- Neurons receive (non-linearly) filtered sensory input
- How can abstract concepts be learned from this information?
  → Subject to neuroscientific research
  Sparse vs. Distributed Coding [Quiroga et al., 2005]?

Prototypes: Psychological Models of Abstract Ideas



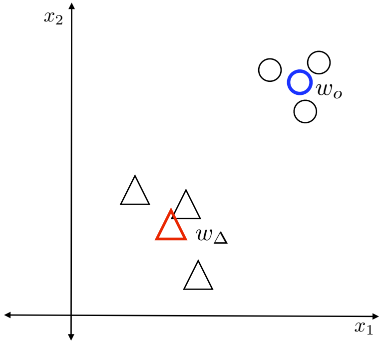
Psychologists postulated that we learn **prototypes** [Jäkel, 2007; Posner and Keele, 1968]

Toy data example:

Neuron receives two dimensional input $x \in \mathbb{R}^2$

Two *classes* of data, Δ and $\circ$

Prototypes: Psychological Models of Abstract Ideas



Prototypes $\mathbf{w}_\Delta$ and $\mathbf{w}_o$ can be the class means

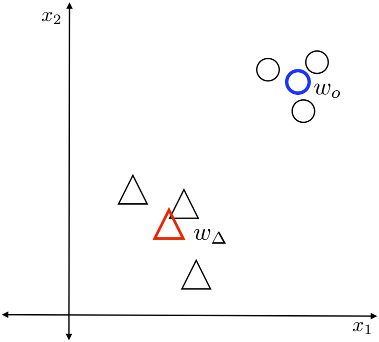
$$\mathbf{w}_\Delta = \frac{1}{N_\Delta} \sum_{n=1}^{N_\Delta} \mathbf{x}_{\Delta,n}$$

$$\mathbf{w}_o = \frac{1}{N_o} \sum_{n}^{N_o} \mathbf{x}_{o,n}$$

Distance from $\mathbf{w}_\Delta$ to new data $\mathbf{x}$

$$\|\mathbf{w}_\Delta - \mathbf{x}\| = \sqrt{\sum_{j=1}^{2} (w_{\Delta j} - x_j)^2}$$

Prototypes: Psychological Models of Abstract Ideas



For new data x check:
Is x more similar to $\mathbf{w}_o$?

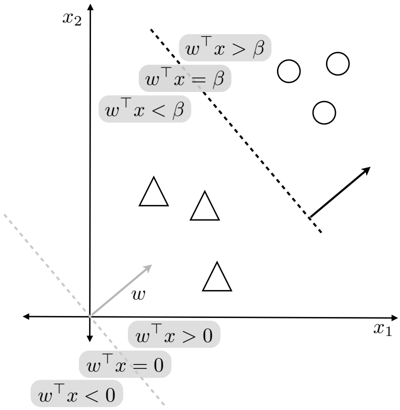
$$\|\mathbf{w}_\Delta - \mathbf{x}\| > \|\mathbf{w}_o - \mathbf{x}\|$$

yes? $\rightarrow$ **x** belongs to o

no? $\rightarrow$ **x** belongs to Δ

How does the classification boundary look like?

Linear Classification

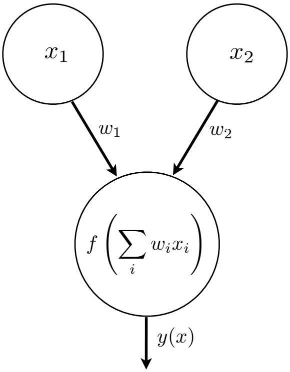


$$\mathbf{w}^\top \mathbf{x} - \beta = \begin{cases} > 0 & \text{if } \mathbf{x} \text{ belongs to } o \\ < 0 & \text{if } \mathbf{x} \text{ belongs to } \triangle \end{cases}$$

From Prototypes to Linear Classification - The Nearest Centroid Classifier

$$
\begin{aligned}
\text{distance}(\mathbf{x}, \mathbf{w}_\Delta) &> \text{distance}(\mathbf{x}, \mathbf{w}_o) \\
\|\mathbf{x} - \mathbf{w}_\Delta\| &> \|\mathbf{x} - \mathbf{w}_o\| \\
\Leftrightarrow \|\mathbf{x} - \mathbf{w}_\Delta\|^2 &> \|\mathbf{x} - \mathbf{w}_o\|^2 \\
\Leftrightarrow (\mathbf{x} - \mathbf{w}_\Delta)^\top (\mathbf{x} - \mathbf{w}_\Delta) &> (\mathbf{x} - \mathbf{w}_o)^\top (\mathbf{x} - \mathbf{w}_o) \\
\Leftrightarrow \mathbf{x}^\top \mathbf{x} - \mathbf{x}^\top \mathbf{w}_\Delta - \mathbf{w}_\Delta^\top \mathbf{x} + \mathbf{w}_\Delta^T \mathbf{w}_\Delta &> \mathbf{x}^\top \mathbf{x} - \mathbf{x}^\top \mathbf{w}_o - \mathbf{w}_o^\top \mathbf{x} + \mathbf{w}_o^T \mathbf{w}_o \\
\Leftrightarrow -2\mathbf{w}_\Delta^\top \mathbf{x} + \mathbf{w}_\Delta^T \mathbf{w}_\Delta &> -2\mathbf{w}_o^\top \mathbf{x} + \mathbf{w}_o^\top \mathbf{w}_o \\
\Leftrightarrow 0 &< \underbrace{(\mathbf{w}_o - \mathbf{w}_\Delta)}_{\mathbf{w}}{}^\top \mathbf{x} - \underbrace{\frac{1}{2}(\mathbf{w}_o^\top \mathbf{w}_o - \mathbf{w}_\Delta^\top \mathbf{w}_\Delta)}_{\beta}
\end{aligned}
$$

Artificial Neural Networks



- Input nodes x_i receive information
- Inputs are multiplied with a weighting factor w_i and summed up
- Integrated input is mapped through some (non-linear) function $f(\cdot)$

$$f(\mathbf{x}) = \begin{cases} +1 & \text{if } \mathbf{x} \text{ is preferred stimulus} \\ -1 & \text{if } \mathbf{x} \text{ is any other stimulus} \end{cases}$$

Rosenblatt's Perceptron

Frank Rosenblatt
(1928-1969)

Rosenblatt proposed the **perceptron**, an artificial neural network for pattern recognition [Rosenblatt, 1958]

Perceptrons gave rise to the field of artificial neural networks

The Perceptron Learning Algorithm

Goal Binary classification of multivariate data $\mathbf{x} \in \mathbb{R}^D$

Input Learning rate η and N tupels $(\mathbf{x}_n, y_n)$ where

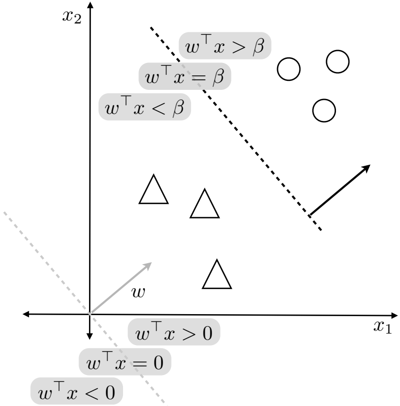
$\mathbf{x}_n \in \mathbb{R}^D$ is the D-dimensional data

$\mathbf{y}_n \in \{-1, +1\}$ is the corresponding label

Output Weight vector $\mathbf{w} \in \mathbb{R}^D$ such that

$$\mathbf{w}^\top \mathbf{x}_n = \begin{cases} \geq 0 & \text{if } y_n = +1 \\ < 0 & \text{if } y_n = -1 \end{cases}$$

Linear Classification and the Perceptron



$$\mathbf{w}^\top \mathbf{x} - \beta = \begin{cases} > 0 & \text{if } \mathbf{x} \text{ belongs to } o \\ < 0 & \text{if } \mathbf{x} \text{ belongs to } \Delta \end{cases}$$

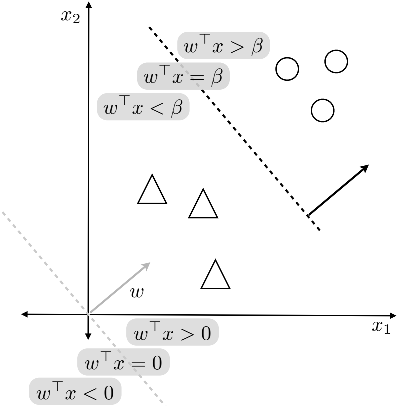
The *offset* β can be included in $\mathbf{w}$

$$\tilde{\mathbf{x}} \leftarrow \begin{bmatrix} 1 \\ \mathbf{x} \end{bmatrix} \qquad \tilde{\mathbf{w}} \leftarrow \begin{bmatrix} -\beta \\ \mathbf{w} \end{bmatrix}$$

such that

$$\tilde{\mathbf{w}}^\top \tilde{\mathbf{x}} = \mathbf{w}^\top \mathbf{x} - \beta.$$

Linear Classification and the Perceptron



What is a good **w**?

$\rightarrow$ We need an **error function** that tells us how good **w** is.

Then we chose **w** such that the error function is minimized.

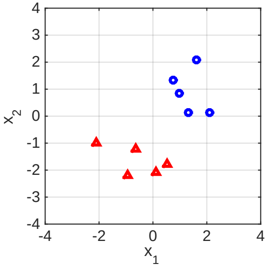
The Perceptron Error Function

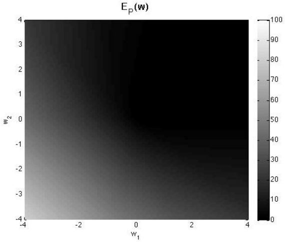
Perceptron error $\mathcal{E}_\mathcal{P}$ is a function of the weights $\mathbf{w}$

$$\underset{\mathbf{w}}{\operatorname{argmin}} \quad \left(\mathcal{E}_\mathcal{P}(\mathbf{w}) = - \sum_{m \in \mathcal{M}} \mathbf{w}^\top \mathbf{x}_m y_m \right) \tag{1}$$

where $\mathcal{M}$ denotes the index set of all *misclassified* data $\mathbf{x}_m$

Data $\mathbf{x} \in \mathbb{R}^2$





The Perceptron Learning Algorithm

Perceptron error $\mathcal{E}_{\mathcal{P}}(\mathbf{w}) = -\sum_{m \in \mathcal{M}} \mathbf{w}^\top \mathbf{x}_m y_m$
can be minimized *iteratively* using **stochastic gradient descent**
[Bottou, 2010; Robbins and Monro, 1951]

1. Initialize $\mathbf{w}^{\text{old}}$ (randomly, $1/n$, ...)
2. While there are misclassified data points

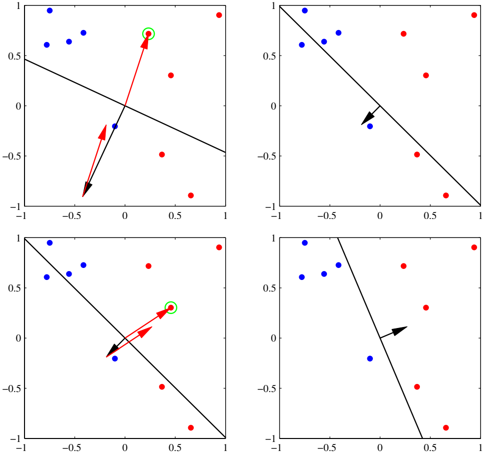
   Pick a random misclassified data point $\mathbf{x}_m$
   Descent in direction of the gradient at single data point $\mathbf{x}_m$

$$\begin{aligned} \mathcal{E}_m(\mathbf{w}) &= -\mathbf{w}^\top \mathbf{x}_m y_m \\ \nabla \mathcal{E}_m(\mathbf{w}) &= -\mathbf{x}_m y_m \end{aligned}$$

$$\mathbf{w}^{\text{new}} \leftarrow \mathbf{w}^{\text{old}} - \eta \nabla \mathcal{E}_m(\mathbf{w}^{\text{old}}) = \mathbf{w}^{\text{old}} + \eta \mathbf{x}_m y_m$$

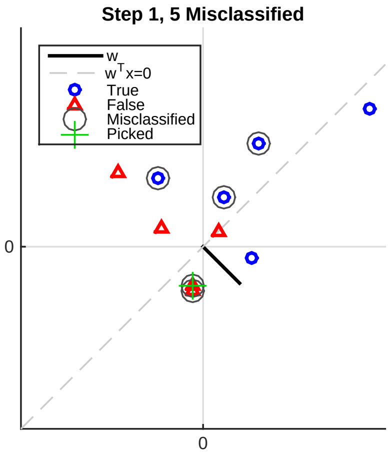
The Perceptron Learning Algorithm in Action 1

$$\mathbf{w}^{\text{new}} \leftarrow \mathbf{w}^{\text{old}} + \eta \mathbf{x}_m y_m$$

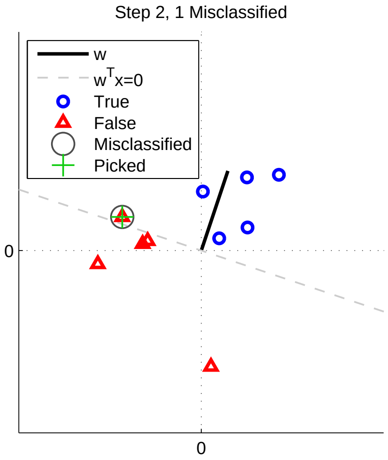


from [Bishop, 2007]

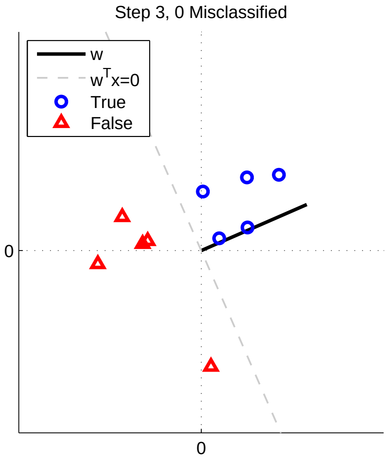
The Perceptron Learning Algorithm in Action 2



The Perceptron Learning Algorithm in Action 2



The Perceptron Learning Algorithm in Action 2



The Perceptron Learning Algorithm

$$\mathbf{w}^{\text{new}} \leftarrow \mathbf{w}^{\text{old}} + \eta \mathbf{x}_m y_m$$

After a single update the error of that data point is reduced:

$$\begin{aligned} -\mathbf{w}^{(\text{new})\top}\mathbf{x}_m y_m &= -\,\mathbf{w}^{(\text{old})\top}\mathbf{x}_m y_m - \eta(\mathbf{x}_m y_m)^\top \mathbf{x}_m y_m \\ &< -\,\mathbf{w}^{(\text{old})\top}\mathbf{x}_m y_m \end{aligned} \tag{2}$$

because $(\mathbf{x}_m y_m)^\top \mathbf{x}_m y_m > 0$

[Novikoff, 1962; Rosenblatt, 1962]

If there is a solution, the perceptron algorithm will find it in a finite number of steps

The learning rate η

[Novikoff, 1962; Rosenblatt, 1962]:

If there is a solution, the perceptron algorithm will find it in a finite number of steps

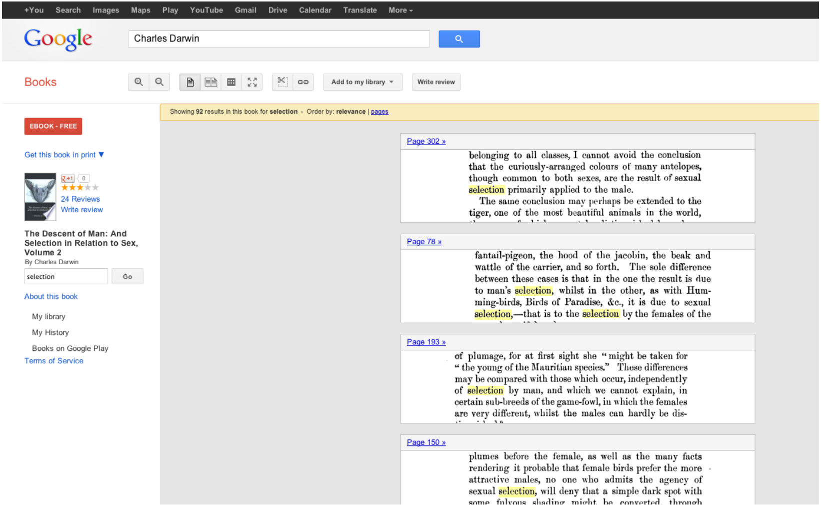
Convergence on non-linearably-separable sets:

$$\mathbf{w}^{\text{new}} \leftarrow \mathbf{w}^{\text{old}} + \eta \mathbf{x}_m y_m$$

- Proven for variable learning rate $\eta(t)$, with $\eta(t) \stackrel{t\to\infty}{\to} 0$
- Best convergence speed is achieved for $\eta(t) \sim \frac{1}{t}$

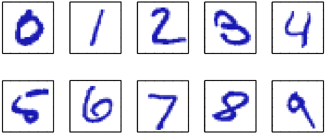
reviewed in [Bottou, 2010]

Application example: Automatic character recognition



Application example: Handwritten Digit Recognition

Handwritten digits
from USPS data set



Each digit represented as 16×16 pixel image

$\rightarrow$ $\mathbf{x} \in \mathbb{R}^{256}$ input nodes

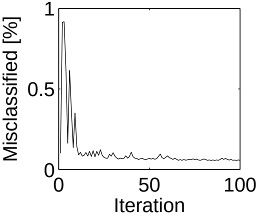
Each image is associated with a label $y \in \{0, 1, \ldots, 9\}$

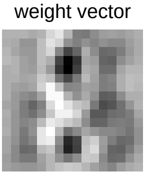
Goal Artificial neural network that recognizes the digit 8

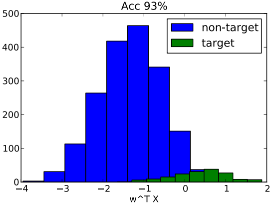
$\Rightarrow$ We need a function $f(\cdot)$ such that

$$f(\mathbf{x}) = \begin{cases} -1 & \text{if } y \in \{0, 1, \ldots, 7, 9\} \\ +1 & \text{if } y = 8 \end{cases}$$

Application example: Handwritten Digit Recognition







Summary

Biological Neural Networks

- Cascade of (non-linear) filters of sensory features
- Abstract ideas are based on integration of these features
- How integration is done is subject of neuroscientific research [Gross, 2002; Quiroga et al., 2005]

Psychologists postulated we learn **Prototypes**

- Prototypes can be the class means
- Prototype theory is closely related to linear classification

Artificial Neural Networks

- Model biological neural networks
- Can learn abstract concepts from data

References

C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. 2007.

L. Bottou. Large-scale machine learning with stochastic gradient descent. In Y. Lechevallier and G. Saporta, editors, *Proceedings of the 19th International Conference on Computational Statistics (COMPSTAT'2010)*, pages 177–187, Paris, France, 2010. Springer.

C. G. Gross. Genealogy of the "grandmother cell". *Neuroscientist*, 8(5):512–8, 2002.

D. Hubel and T. Wiesel. Receptive fields of single neurones in the cat's striate cortex. *The Journal of Physiology*, 148:574–591, 1959. doi: 10.1113/jphysiol.2009.174151.

F. Jäkel. *Some Theoretical Aspects of Human Categorization Behaviour: Similarity and Generalization*. PhD thesis, 2007.

D. Marr. *Vision. A Computational Investigation into the Human Representation and Processing of Visual Information*. W. H. Freeman and Company, 1982.

A. B. Novikoff. On convergence proofs on perceptrons. In *Proceedings of the Symposium on the Mathematical Theory of Automata*, volume 12, pages 615–622, New York, NY, USA, 1962. Polytechnic Institute of Brooklyn.

M. I. Posner and S. W. Keele. On the genesis of abstract ideas. *Journal of Experimental Psychology*, 77(3):353–363, 1968.

R. Q. Quiroga, L. Reddy, G. Kreiman, C. Koch, and I. Fried. Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102–1107, 2005. doi: 10.1038/nature03687.

H. Robbins and S. Monro. A stochastic approximation method. *Annals of Mathematical Statistics*, 22(3):400—407, 1951.

F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6): 386–408, Nov. 1958.

F. Rosenblatt. *Principles of neurodynamics; perceptrons and the theory of brain mechanisms*. Spartan Books, 1962.