Neuronale Netze (SS 2001), 8.4.

NN – where are they good for?

- **Technical point of view:**
  - Feedforward networks (FNN) for: function approximation, classification, time series prediction, image processing, …
  - Partially recurrent networks (pRNN) for: time series processing, language, control, robotics, structure processing, …
  - Fully recurrent networks (fRNN) for: optimization, associative memory, …
  - Selforganizing maps (SOM) for: data mining, information extraction, data preprocessing, …

  Pros: robust, effective, parallel, domain independent, learning instead of modelling

  Cons: requires lots of data, domain specific preprocessing and modularization, black boxes, monolithic

  $\rightarrow$ NNs are often the second best solution.

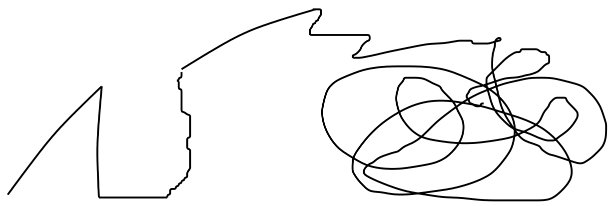
- **Biological point of view:**
  - FNN: direct information processing e.g. in the visual cortex
  - pRNN: information processing with conditioning/interior state
  - fRNN: memory
  - SOM: topology preserving maps e.g. in the neocortex

  But: how is information processed in the brain?

  - Single neuron doctrine $\rightarrow$ simple but not plausible
  - Distributed representation $\rightarrow$ robust, generalization, …, but: how to extract information, binding problem

  $\rightarrow$ temporal structure might be necessary for binding, fast processing; we can here model only some aspects.

History of NNs:



... this gives a pretty good picture ... ;-)

The Perceptron:

... one neuron with step activation ($\rightsquigarrow$ biological neuron without fine temporal structure)

Given by:

$n =$ dimensionality of the inputs,
weights $w_1, \ldots, w_n \in \mathbb{R}$, (short $\vec{w} \in \mathbb{R}^n$)
bias/threshold $\theta \in \mathbb{R}$.

Computes:

$$f : \mathbb{R}^n \to \{0, 1\}, (x_1, \ldots, x_n) \mapsto \begin{cases} 1 & \text{if } w_1x_1 + \ldots, w_nx_n \geq \theta \\ 0 & \text{otherwise} \end{cases}$$

Short notation:

$$\vec{x} \mapsto \mathrm{H}(\vec{w}^t\vec{x} - \theta)$$

Def: a neuron is given by:

$n =$ dimensionality of the inputs,
weights $w_1, \ldots, w_n \in \mathbb{R}$, (short $\vec{w} \in \mathbb{R}^n$)
bias/threshold $\theta \in \mathbb{R}$,
activation function $\sigma : \mathbb{R} \to \mathbb{R}$.

It computes

$$f : \mathbb{R}^n \to \mathbb{R}, (x_1, \ldots, x_n) \mapsto \sigma(\vec{w}^t\vec{x} - \theta)$$