České vysoké učení technické v Praze

Fakulta elektrotechnická

Katedra kybernetiky
Katedra počítačů

Intelligent Data Analysis
RESEARCH GROUP

COMPUTATIONAL
INTELLIGENCE
GROUP

Vytěžování dat – přednáška 9

Umělé neuronové sítě v data miningu

Osnova přednášky

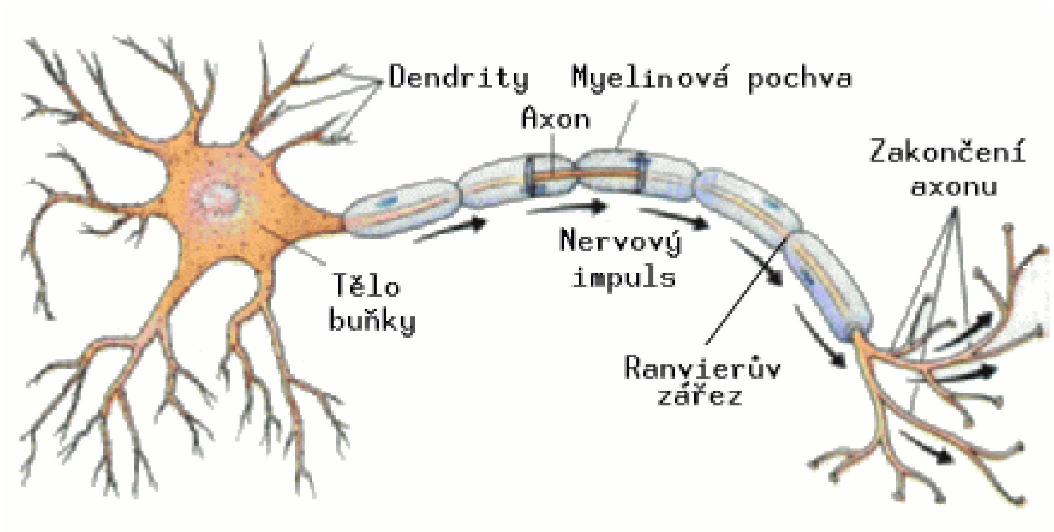
Úvod do neuronových sítí

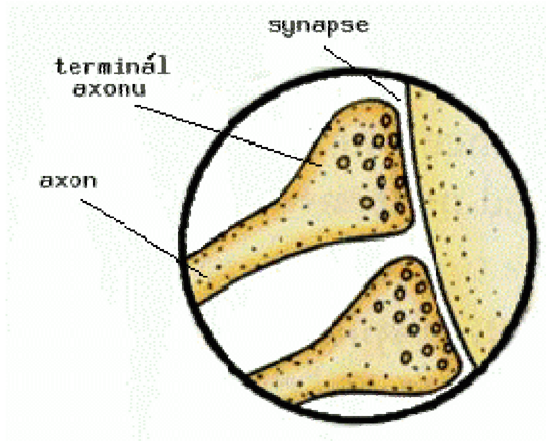
- Biologická inspirace
- Historie
- Perceptron
- Minsky-Pappert omyl
- Vícevrstvý perceptron
- Příklady neuronových sítí – MLP, GMDH, RBFN, SOM

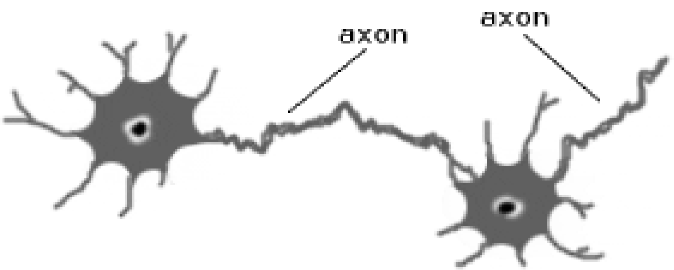
- Neuronové sítě a data mining
  - Použití neuronových sítí k řešení reálných problémů
  - Konkrétní příklady vytěžování dat pomocí NS

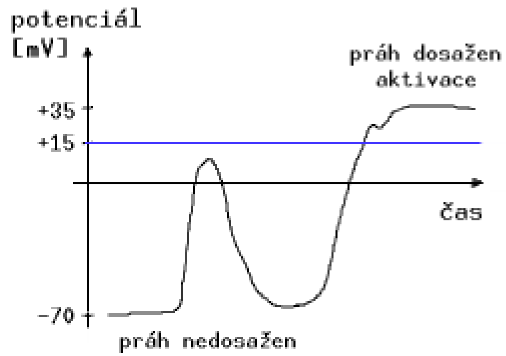
Úvod do neuronových sítí

Biologický neuron









http://neuron.felk.cvut.cz/courseware/data/chapter/36nan026/s01index.html

Co jsou to ty neuronové sítě?

- Umělé informační systémy, které jsou principiálně schopny napodobovat funkce nervových soustav a mozků živých organizmů podstatně dokonalejším způsobem, než to činí dosavadní, tzv. konvenční výpočetní technika.
- Pozor! Od umělých neuronových sítí k mozku je ale ještě strašně daleko...

Umělá neuronová síť

- Distribuované, paralelní zpracování dat
- Vzájemně propojené výkonné prvky (neurony)
- Každý výkonný prvek transformuje vstupní data na výstupní podle jisté přenosové funkce.
- Přitom se též může uplatnit obsah jeho lokální paměti.
- Signál se šíří sítí tak, že výstupy neuronů jsou přivedeny na vstupy dalších neuronů.

Pracovní fáze umělé neuronové sítě

- Umělá neuronová síť pracuje v zásadě ve dvou fázích - adaptivní a aktivní.
- V adaptivní fázi se síť **učí** (learning phase),
- v aktivní vykonává naučenou činnost, **vybavuje** (evaluation phase).
- Paměť sítě reprezentují hodnoty synaptických vah na jednotlivých vstupech neuronů.

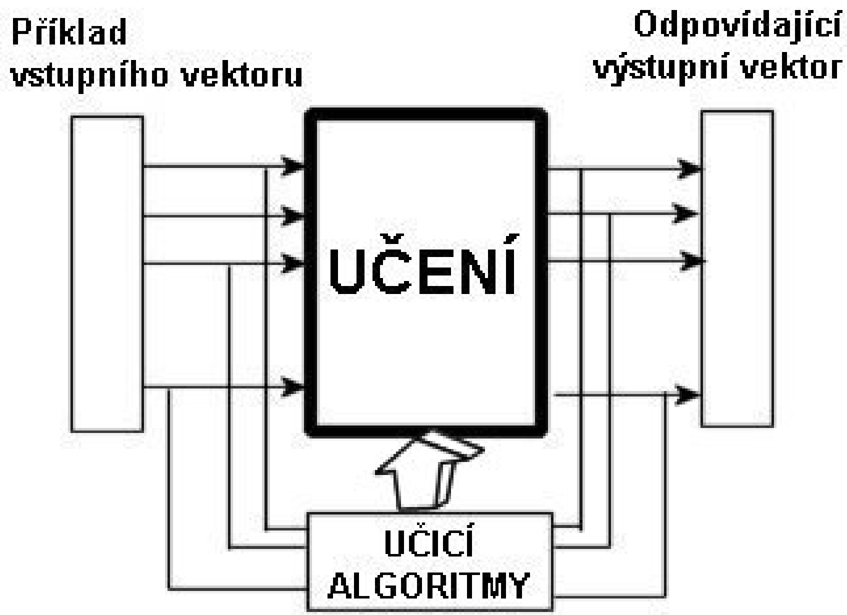
Učení a jeho typy

- Při **učení s učitelem** se umělá neuronová síť učí srovnáváním aktuálního výstupu s výstupem požadovaným a přestavováním synaptických vah tak, aby se propříště snížil rozdíl mezi skutečným a žádaným výstupem. Metodika snižování rozdílu je určena učicím algoritmem.
- Do **učení bez učitele** není zapojen žádný vnější arbitr a celé učení je založeno pouze na informacích, které samotná síť během celého procesu učení získala.

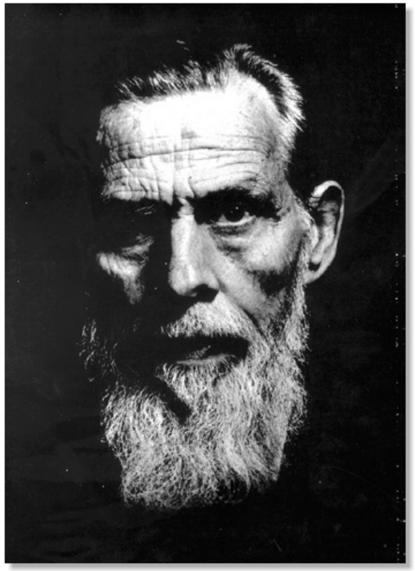
Možná se dostane i na Hebbovské učení, posilované učení, atd.

Učení umělé neuronové sítě

- Minule – kompetiční učení (bez učitele)
- Dnes – učení s učitelem:



Historie

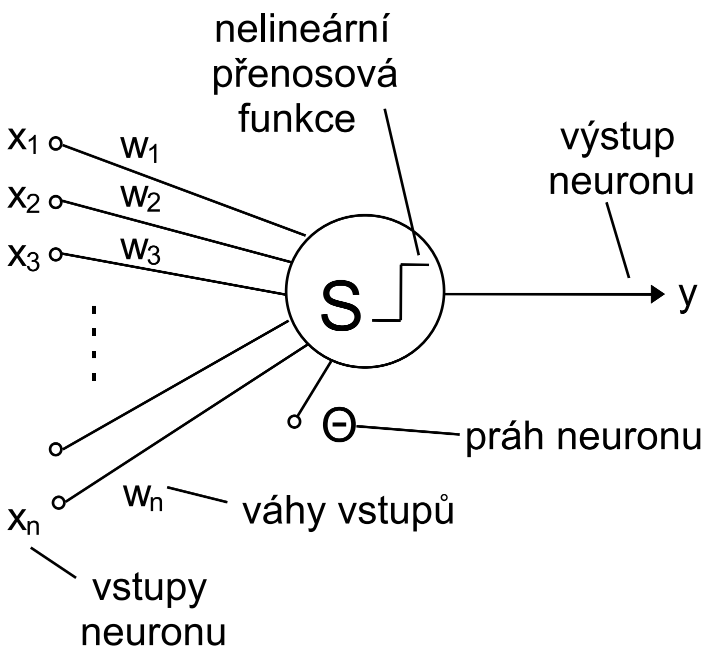


1899 - 1969

1923 - 1969

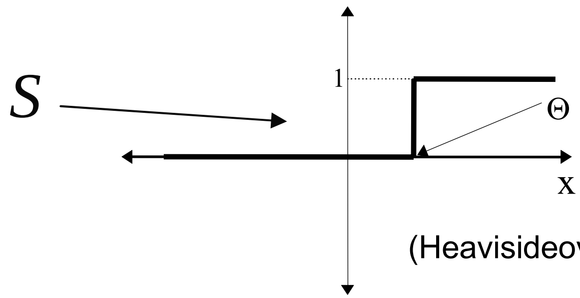
Warren McCulloch, Walter Pitts

McCulloch-Pittsův neuron (1943)



- y je výstup neboli aktivita neuronu,
- x_i je i-tý vstup neuronu, vstupů je celkem N,
- w_i představuje hodnotu i-té synaptické váhy,
- S je (nelineární) přenosovou funkcí neuronu a
- Θ představuje prahovou hodnotu (vlastně posunutí).
- Výraz v závorce je vnitřní potenciál.

$$y = S\left(\sum_{i=1}^{N} w_i x_i + \Theta\right)$$



(Heavisideova funkce)

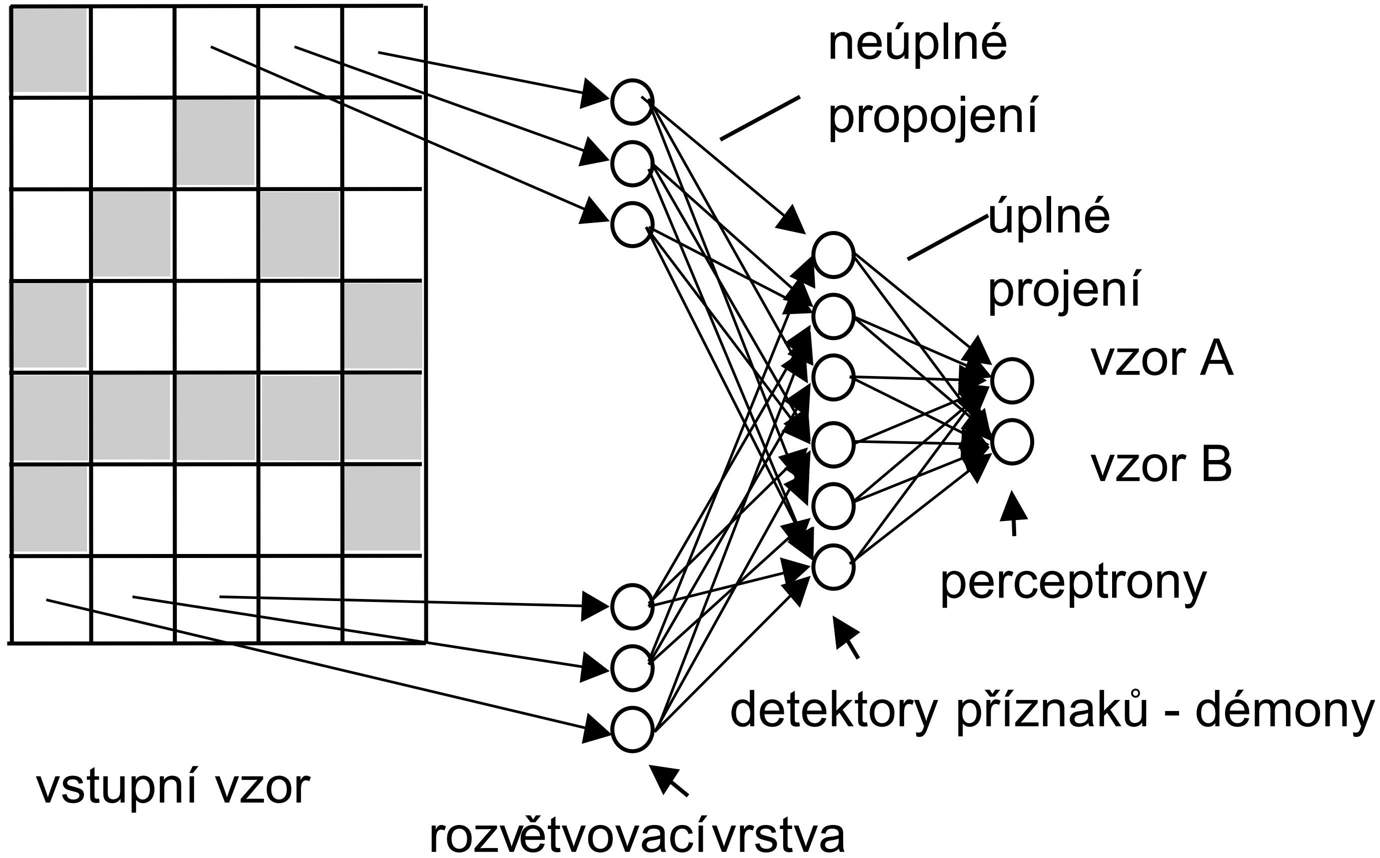
Rosenblattův perceptron (1957)

Frank Rosenblatt -

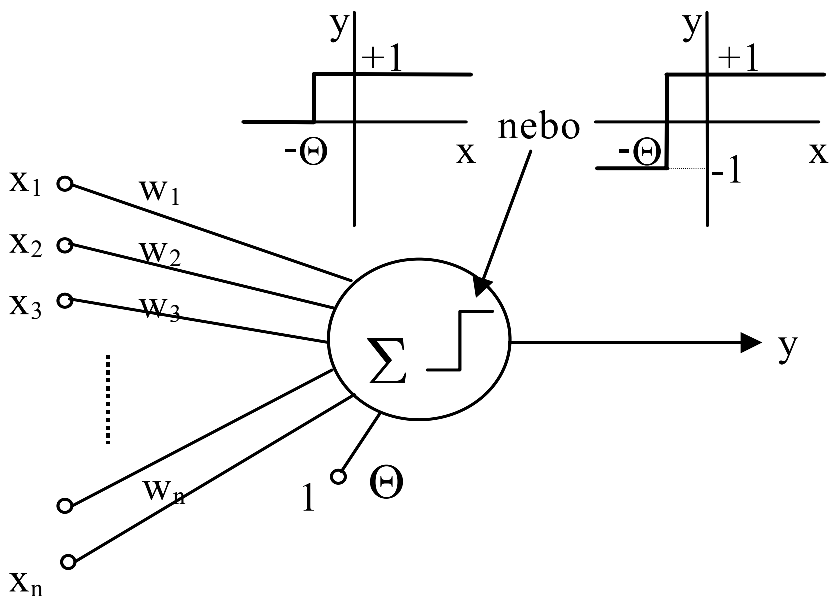
tvůrce prvního neuropočítače Mark I, 1960

- Vymyslel pro MP-neuron algoritmus učení
- Nazývá neuron perceptronem (pattern recognizer)
- Použil ho pro klasifikaci písmen

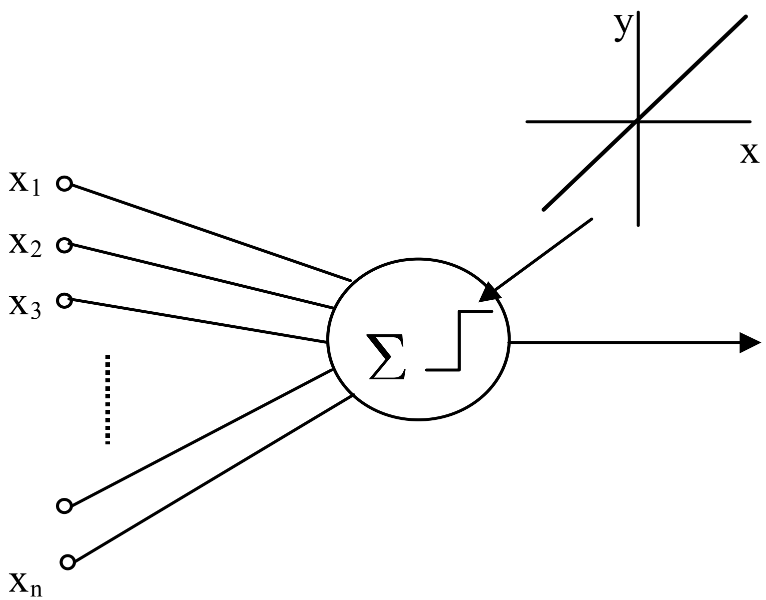
Rosenblattova perceptronová síť



Neurony v Rosenblattově síti



neuron v perceptronové vrstvě



neuron v démonické vrstvě

Rosenblatt - učicí algoritmus

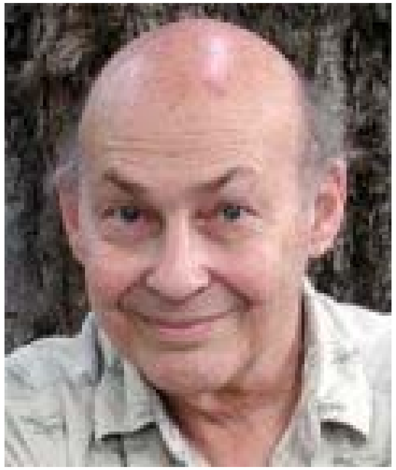
- Váhy se nastaví náhodně.
- Je-li výstup správný, váhy se nemění.
- Má-li být výstup roven 1, ale je 0/-1, inkrementuj váhy na aktivních vstupech.
- Má-li být výstup roven 0/-1, ale je 1, dekrementuj váhy na aktivních vstupech.
- Vstupy jsou přitom aktivní tehdy, když je jejich hodnota nad prahem, tedy nenulová. Velikost změny vah (inkrementace nebo dekrementace) závisí na konkrétně zvolené ze tří možných variant:
  - Při inkrementaci i dekrementaci se aplikují pevné přírůstky.
  - Přírůstky se mění v závislosti na velikosti chyby. Je výhodné, jsou-li při větší chybě větší a naopak. Takto dosažené zrychlení konvergence však může mít za následek nestabilitu učení.
  - Proměnné a pevné přírůstky se kombinují v závislosti na velikosti chyby.

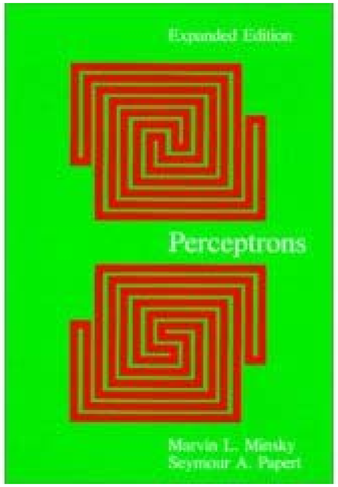
Rosenblatt a spol.

- Velmi dobré výsledky
- Přitažlivý výzkum slibující velkou perspektivu.
- S dostatečně velkou sítí budu schopen rozpoznat cokoli!
- Na to grantové agentury slyší.
- Mnoho vědců se snaží výsledky napodobit a rozvinout.

MP-Perceptron

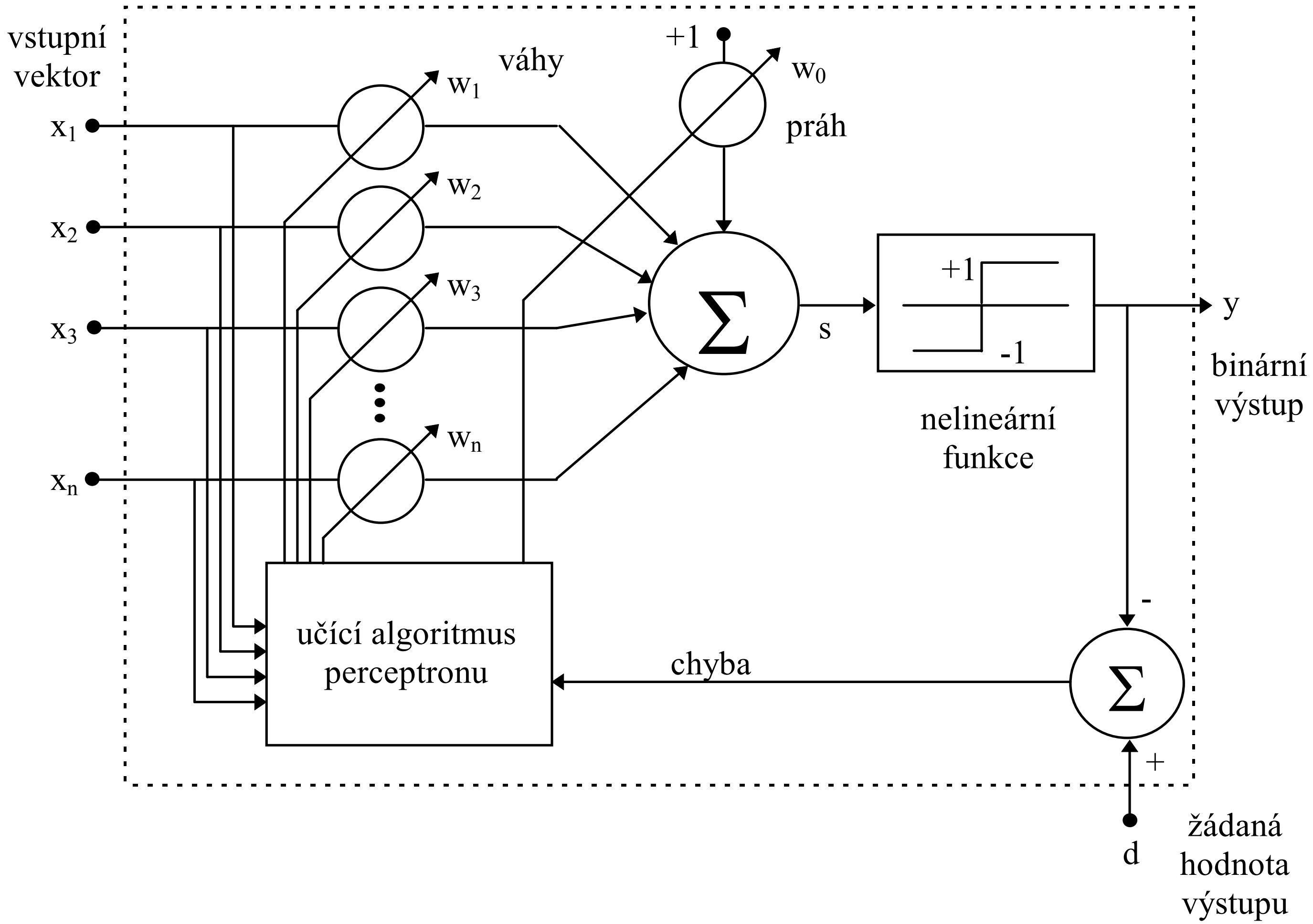
- MP = Marvin Minsky a Seymour Papert,
  - MIT Research Laboratory of Electronics,
  - V roce 1969 publikovali knihu:





Perceptrons

MP-perceptron



MP – algoritmus učení

Pro každý učicí vzor uprav váhy perceptronu:

$$w_i(t+1) = w_i(t) + \eta e(t) x_i(t),$$

$$e(t) = [d(t) - y(t)] \quad ,$$

Je to chyba? Jakých hodnot nabývají *d* a *y*?

kde

w_i je váha itého vstupu

x_i je hodnota itého vstupu

η je koeficient učení

y je výstup neuronu

d je požadovaný výstup

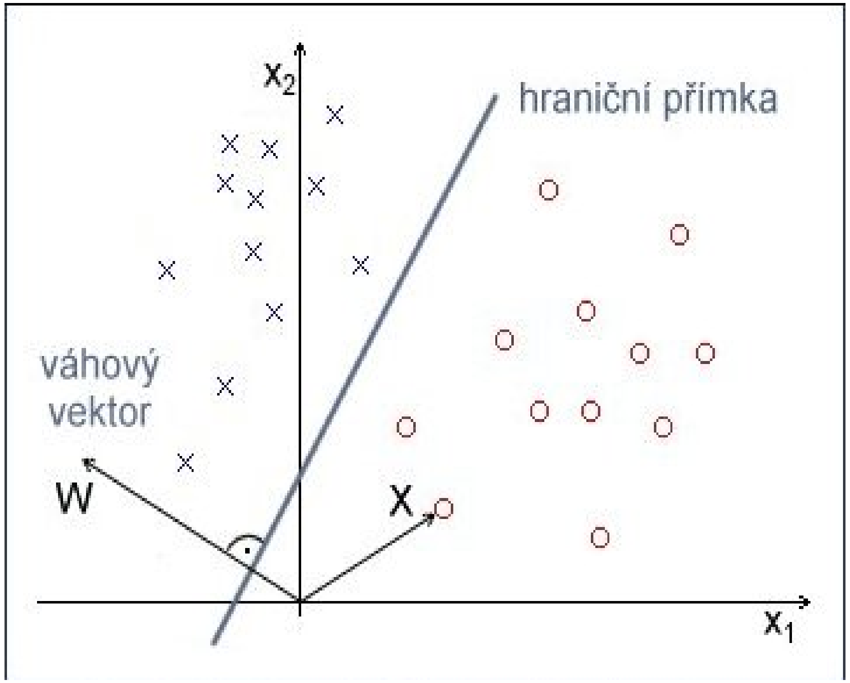
Geometrická interpretace neuronu

$$f(\boldsymbol{X}) = \boldsymbol{W} \cdot \boldsymbol{X} - \theta = \sum_{i=1}^{N} w_i \cdot x_i - \theta$$ je vnitřní potenciál neuronu

Položíme nule (ve 2D):

$$w_1 * x_1 + w_2 * x_2 - \theta = 0$$

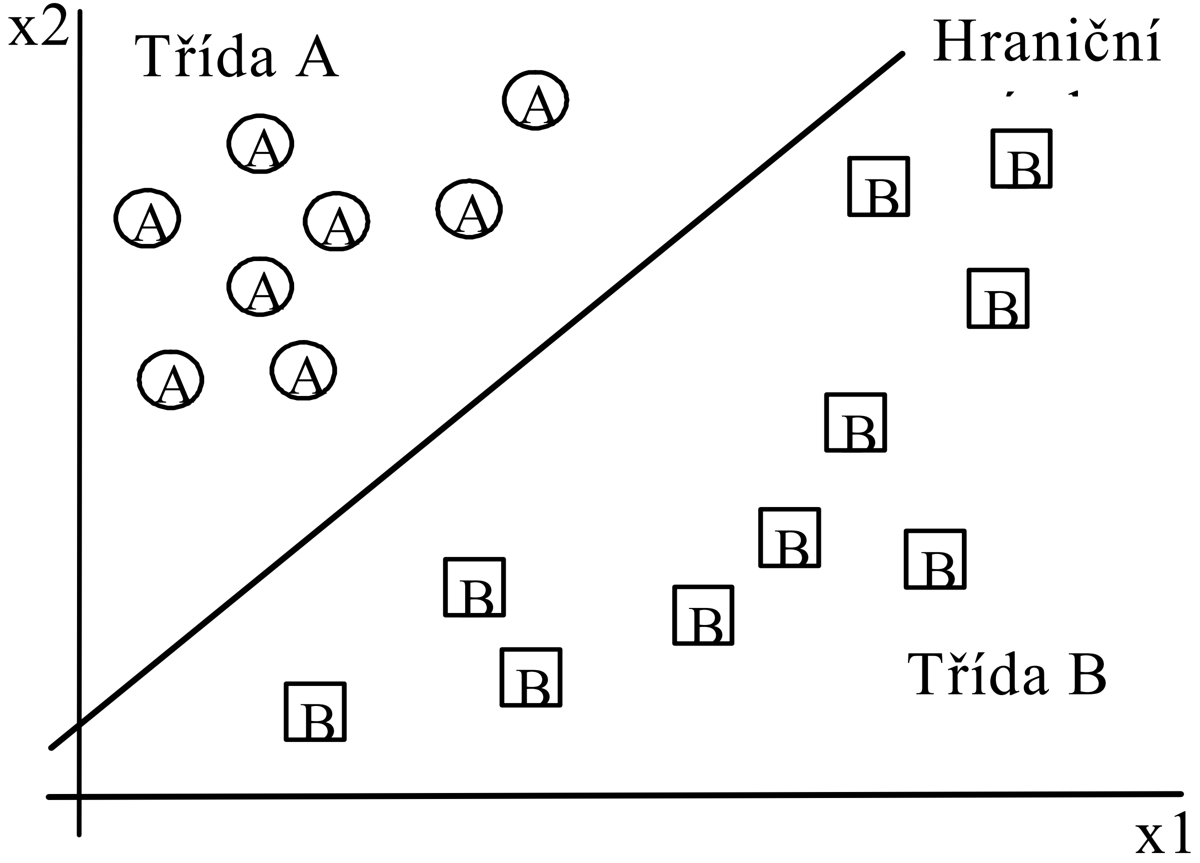
$$x_2 = -\frac{w_1}{w_2} * x_1 + \frac{\theta}{w_2}$$



Co s tím udělá nelinearita na výstupu?

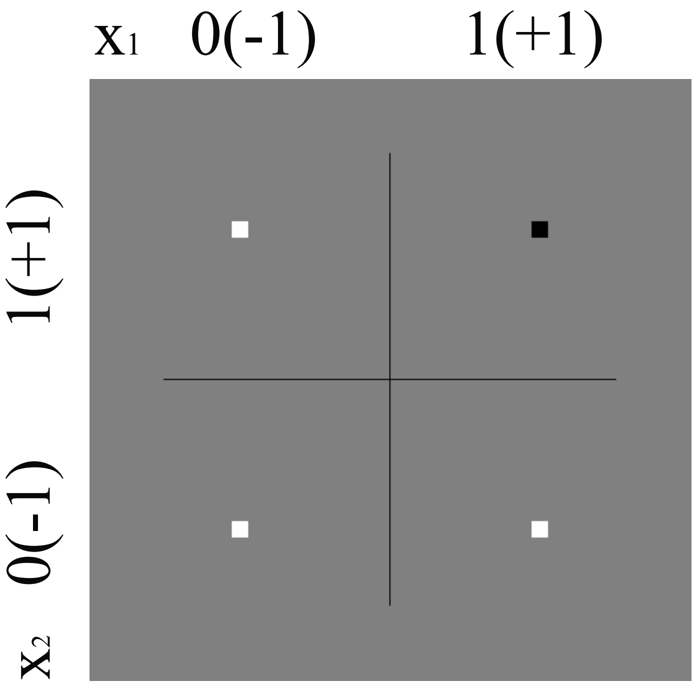
Rovnice přímky se směrnicí $-w_1/w_2$ a posuvem na ose x_2 rovným θ/w_2

Jak tedy klasifikuje Rosenblattův perceptron?



Příklad: AND

- Podívejme se na reprezentaci funkce AND
- Bílý čtvereček znamená 0, černý 1
- 0 je kódovaná jako -1, jednička jako +1



-1 AND -1 = false

-1 AND +1 = false

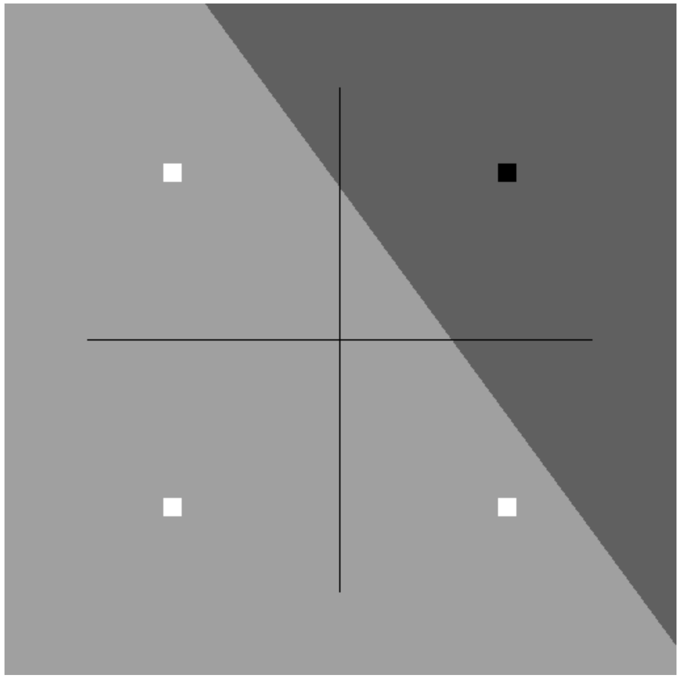
+1 AND -1 = false

+1 AND +1 = true

Použity slidy: Daniel A. Jiménez, Department of Computer Science, Rutgers University

Příklad: AND, pokračování

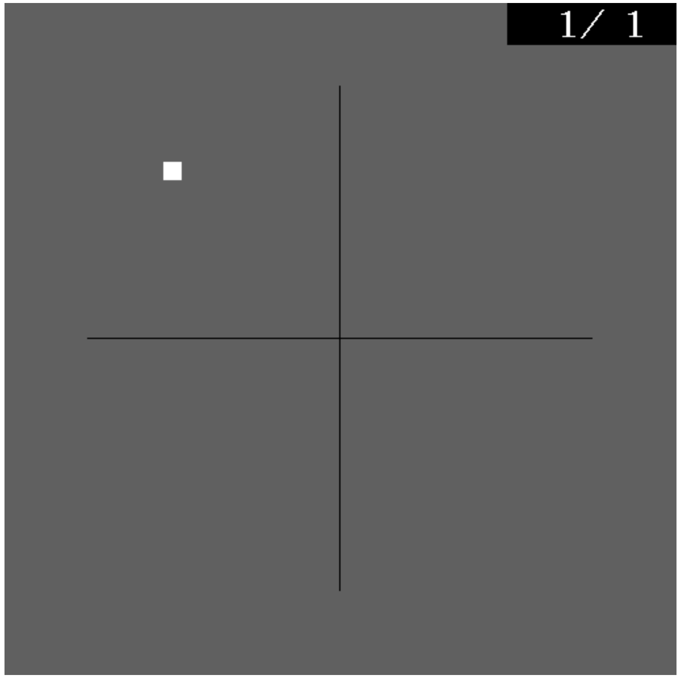
- Lineární řezná rovina - pro všechny kombinace vstupů je výstup buď +1, nebo -1. Hranice, kde se výstup mění se nazývá *decision boundary*



Použity slidy: Daniel A. Jiménez, Department of Computer Science, Rutgers University

Příklad: AND, pokračování

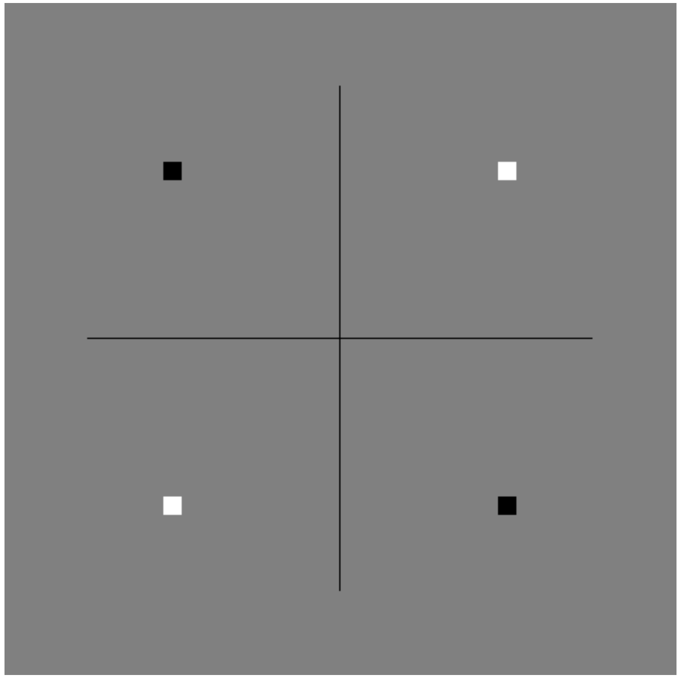
- Dívejte se, jak se perceptron naučí funkci AND:



Použity slidy: Daniel A. Jiménez, Department of Computer Science, Rutgers University

Příklad: XOR

- Funkce XOR je reprezentována takto:



-1 XOR -1 = *false*

-1 XOR +1 = *true*

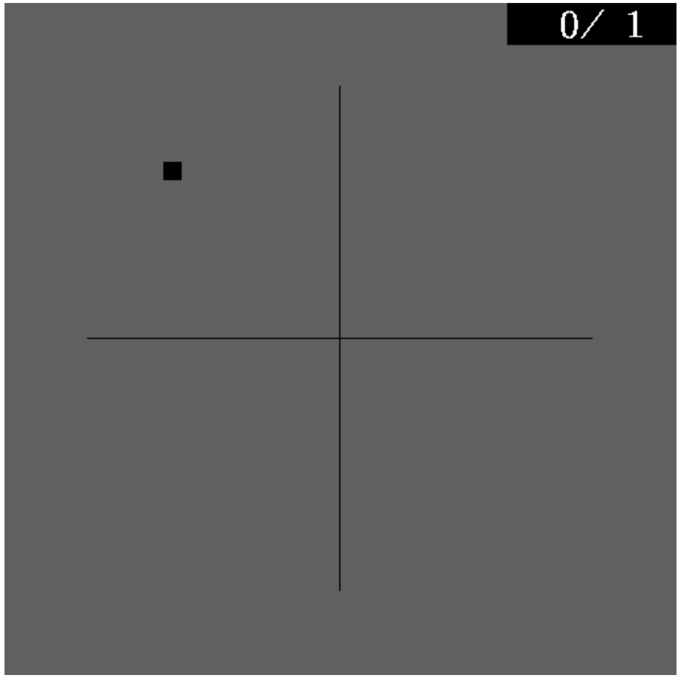
+1 XOR -1 = *true*

+1 XOR +1 = *false*

Použity slidy: Daniel A. Jiménez, Department of Computer Science, Rutgers University

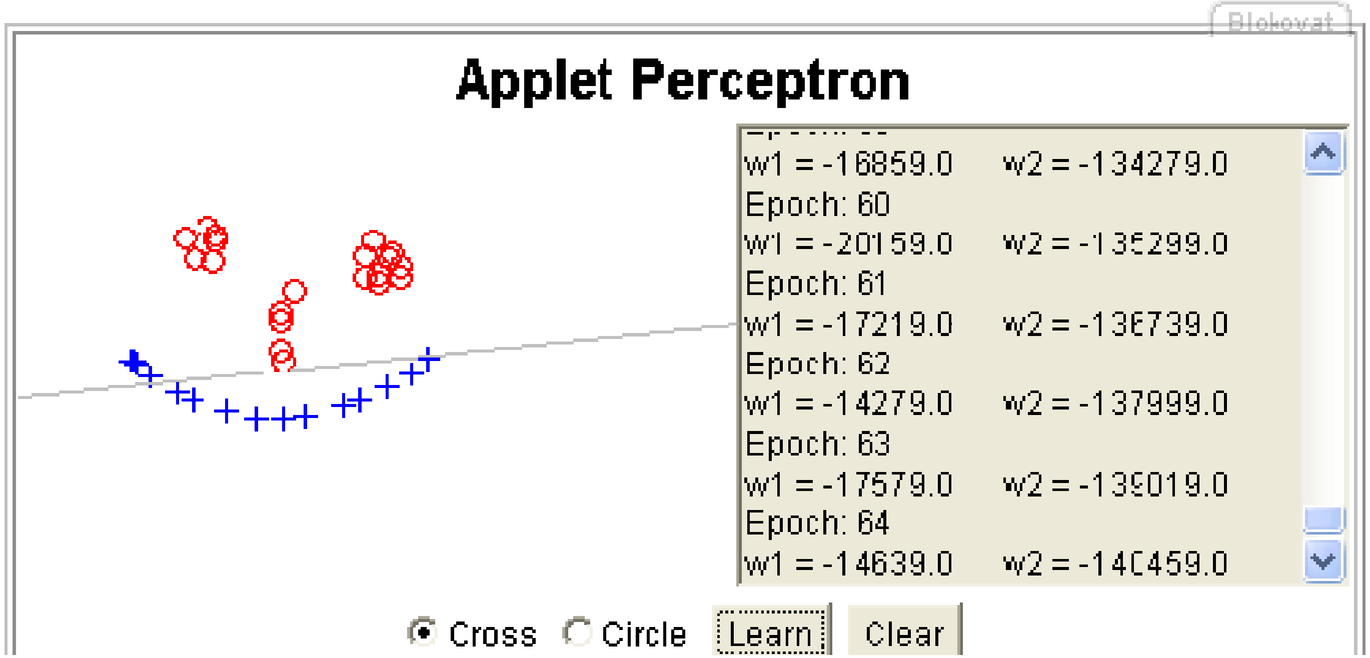
Příklad: XOR, pokračování

- Sledujte, jak se perceptron snaží naučit XOR:

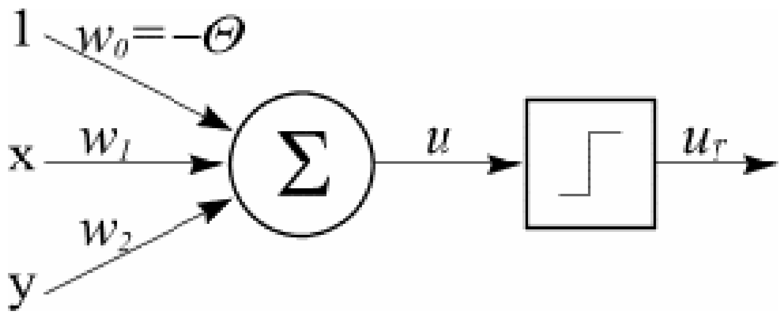


Použity slidy: Daniel A. Jiménez, Department of Computer Science, Rutgers University

Applet perceptron



http://neuron.felk.cvut.cz/courseware/data/chapter/36nan028/s04.html



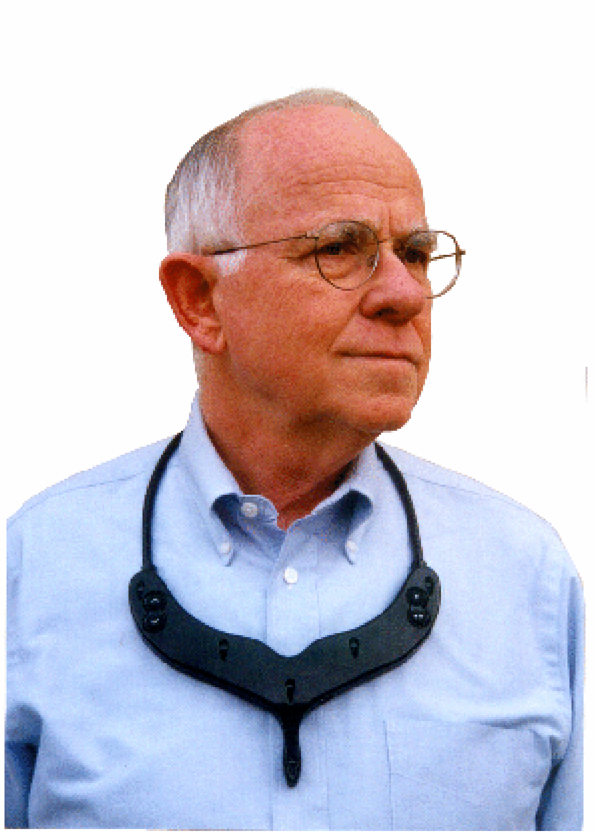
Kdy je perceptron naučen?

- Tehdy, když je pro všechny vzory **učicí** množiny (podmnožina množiny vstupních vzorů) splněno zadané kritérium naučenosti formulované např. tak, že
  - chyba perceptronu je menší než …
- Jak vypočítám jeho chybu?
- Kdy mám ukončit učení?
  - chyba na validační množině začíná růst
  - stagnace po určitý počet epoch

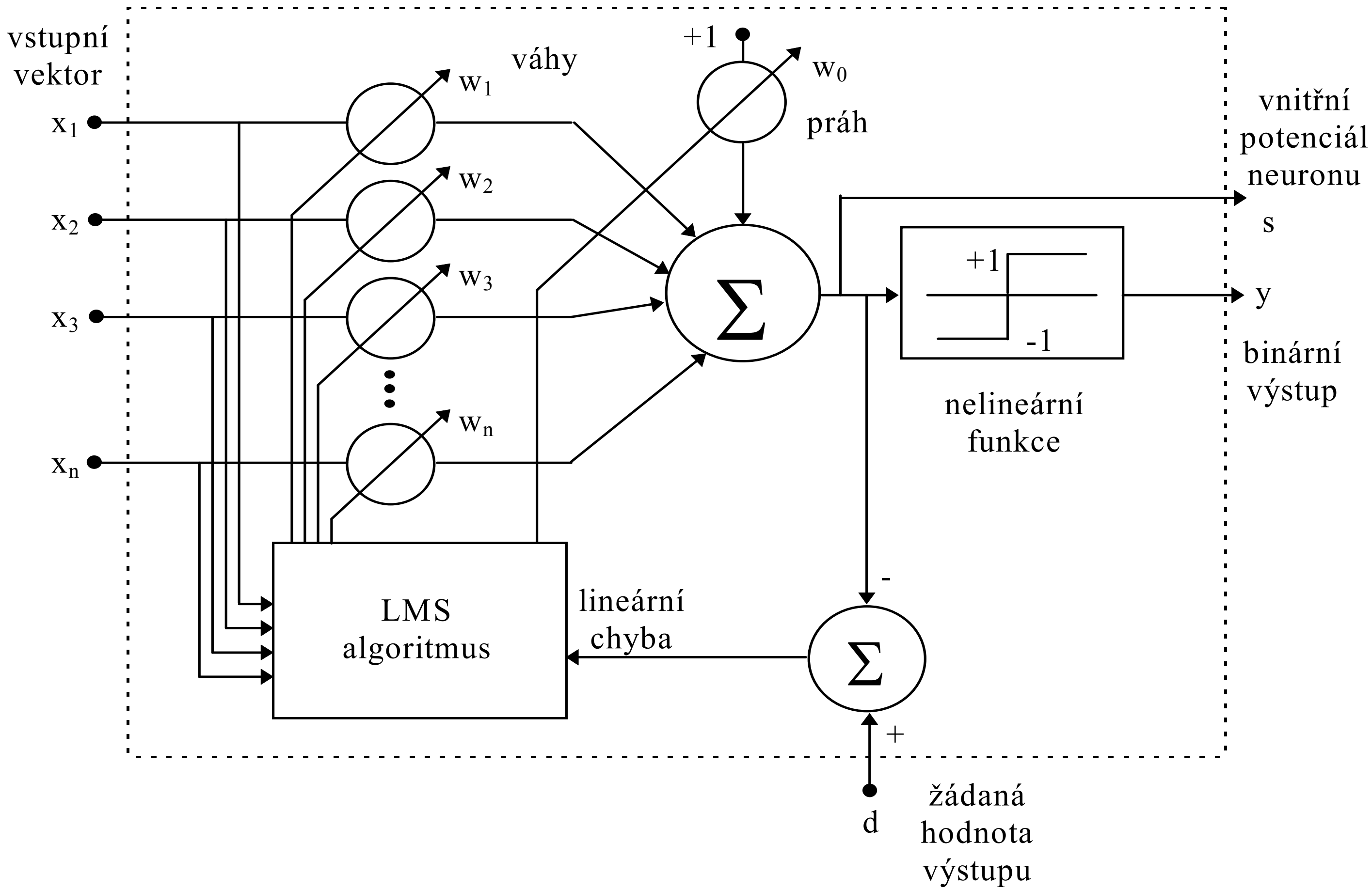
Epocha = předložení všech vzorů trénovací množiny na vstup

ADALINE

B.Widrow, Stanfordská universita, 1960



ADALINE - **AdA**ptive **Li**near **Ne**uron



Učicí algoritmus ADALINE (tzv. delta rule)

$$w_i(t+1) = w_i(t) + \eta e(t) x_i(t),$$

$$e(t) = d(t) - s(t).$$

kde

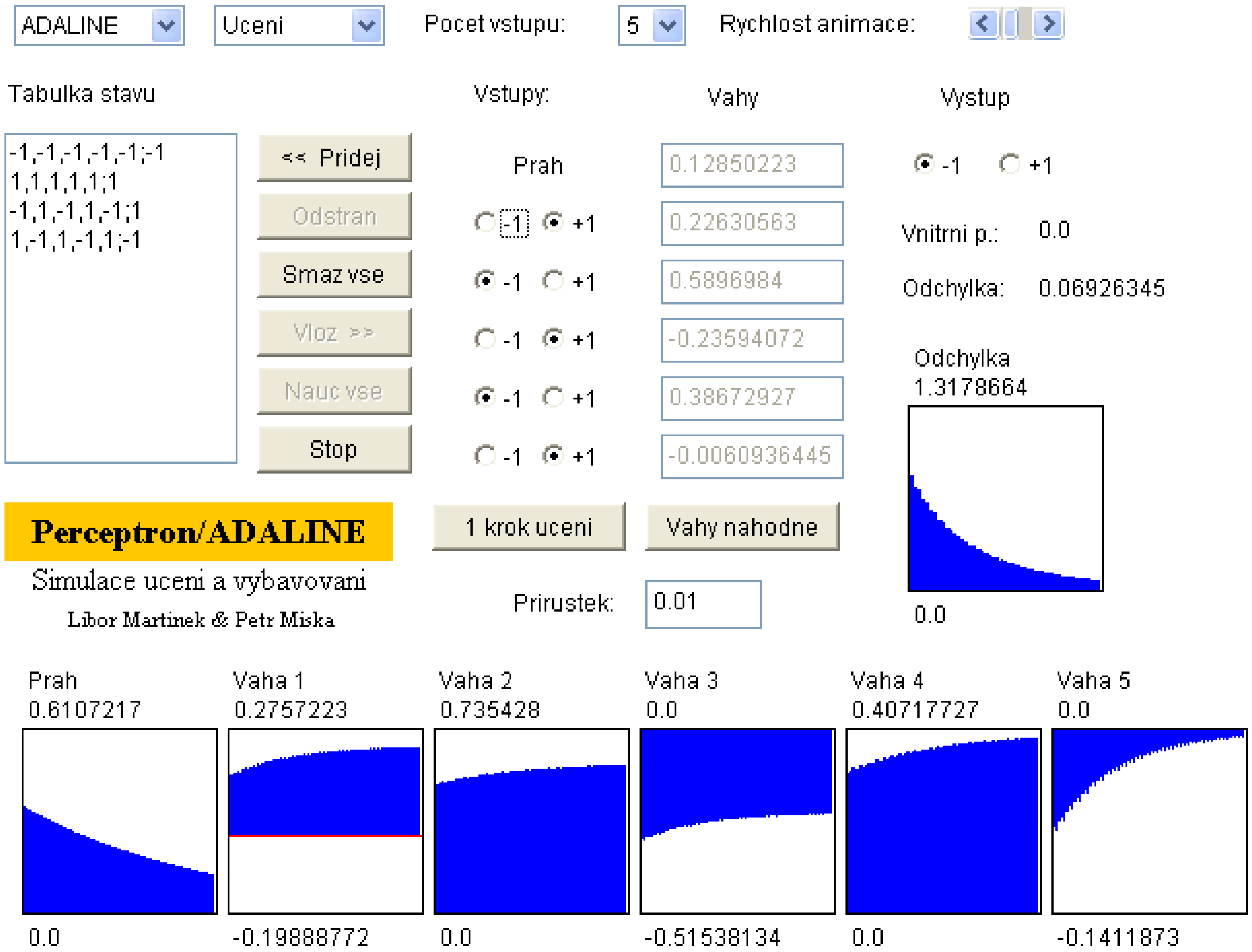
w_i je váha itého vstupu

x_i je hodnota itého vstupu

η je koeficient učení

s je vnitřní potenciál neuronu

d je požadovaný výstup



http://neuron.felk.cvut.cz/courseware/data/chapter/36nan009/example/s03Applet.html

Poznáte,v čem se liší MP-perceptron a ADALINE?

- Perceptron: $\Delta w_{ij} = n(d_i - y_i)x_j$
- Delta rule: $\Delta w_{ij} = n(d_i - s_i)x_j$

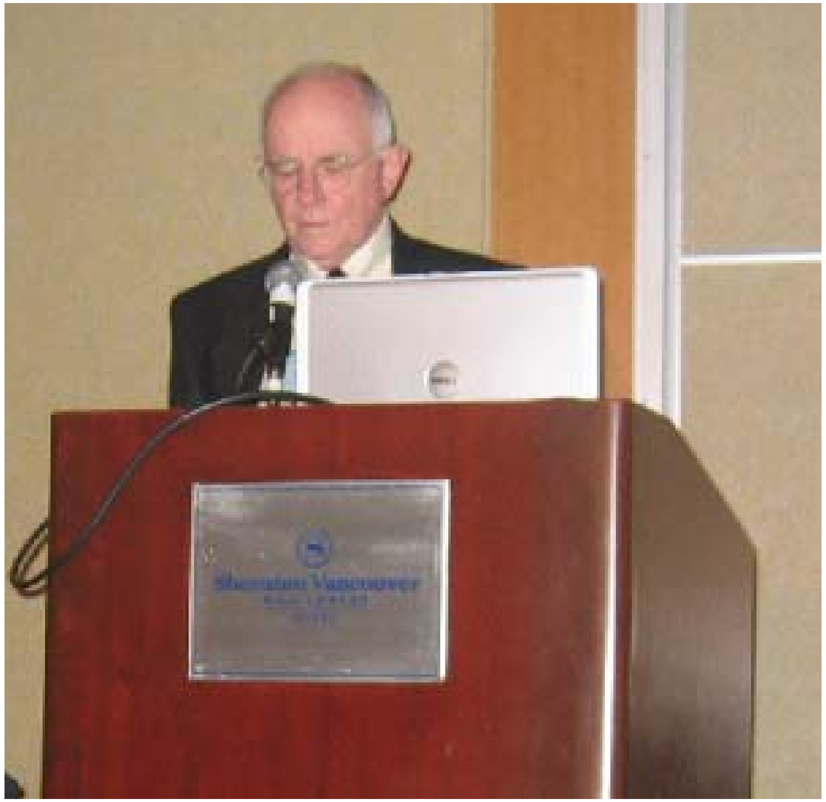
y_i -vs- s_i (výstup nebo vnitřní potenciál):

- Delta rule se může asymptoticky přiblížit k minimu chyby pro lineárně **ne**separabilní problémy; perceptron NE.
- Perceptron vždy konverguje k bezchybnému klasifikátoru lineárně separabilních problémů; delta rule se to podařit nemusí!

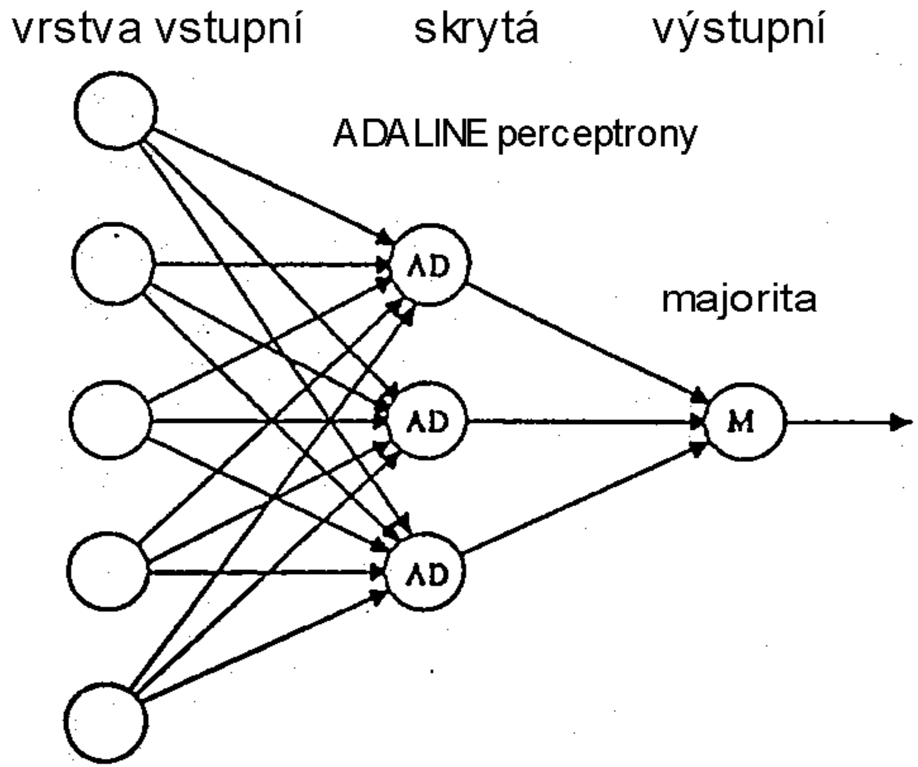
Kdy by percepton zkonvergovat nemusel?

MADALINE (**M**ultiple **Ada**ptive **Lin**ear **E**lement)

- Zpracovává binární signály,
- pracuje s bipolárním kódováním,
- jedna skrytá, jedna výstupní vrstva,
- učí se s učitelem,
- autorem je opět B. Widrow.



MADALINE - architektura sítě



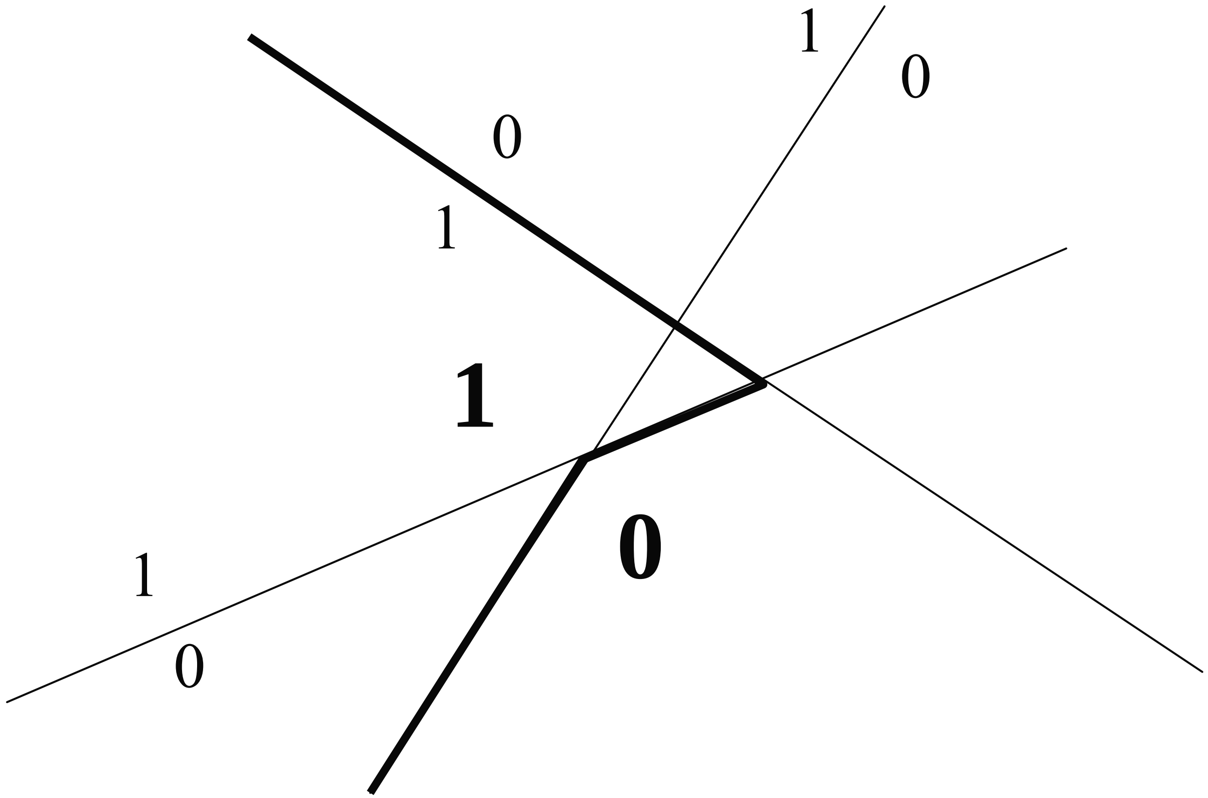
Pozn.: učí se pouze synapse na vstupech ADALINE perceptronů.

MADALINE možnosti použití

- První neuronová síť reálně nasazená pro řešení problému z praxe.
- Použita jako adaptivní filtr pro potlačení ozvěn na telefonních linkách (dodnes se někde používá).

- Zvládne vyřešit nelineárně separabilní problém s nulovou chybou?

MADALINE



MADALINE omezení

- MADALINE není použitelná na složitější problémy.

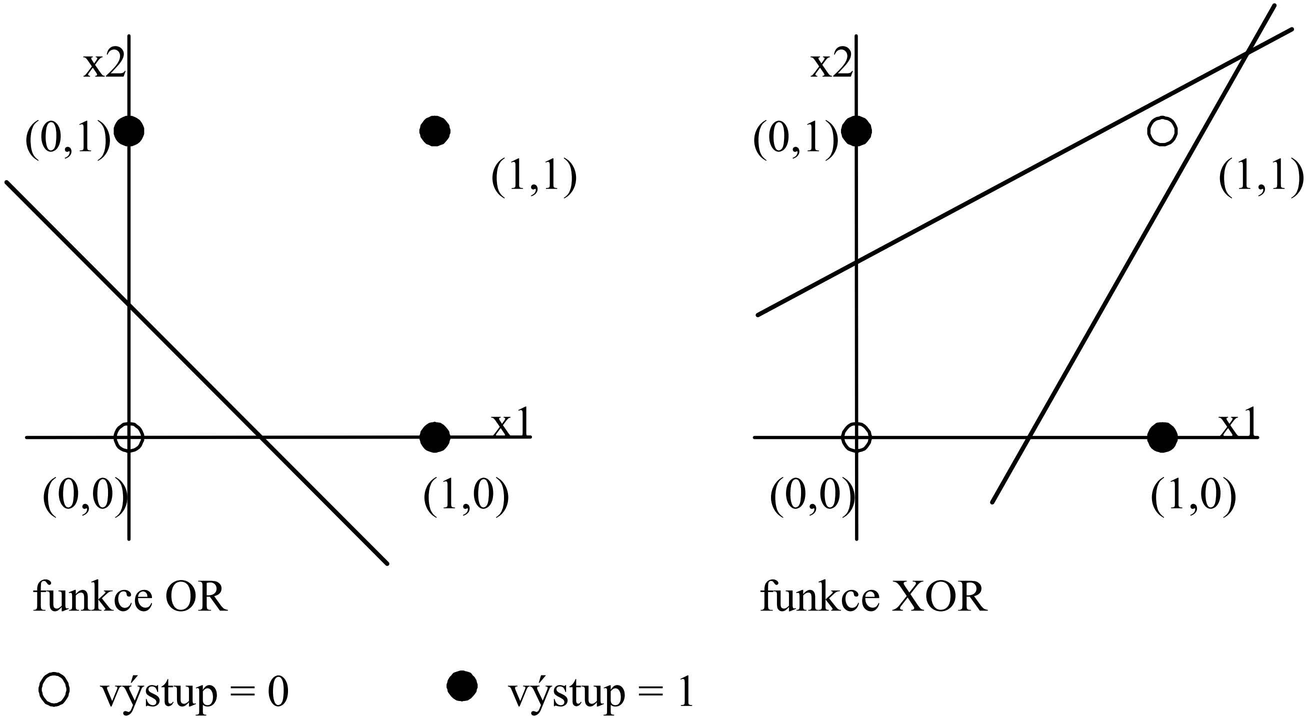
- Proč?
- Kvůli algoritmu učení – ADALINE se učí nezávisle, nejsou schopny si nějakým způsobem rozdělit vstupní prostor aby společně vytvořily složitou rozhodovací hranici …

Útlum výzkumu neuronových sítí

- Respektovaná kniha Minskyho-Paperta „Perceptron“ (1969) uvádí, že neuronové sítě nezvládnou uspokojivě řešit lineárně neseparabilní problémy.

- Grantové agentury usoudili, že financování výzkumu neuronových sítí není perspektivní – 20 let živoření

Lineární neseparabilita - Minsky-Papertův omyl



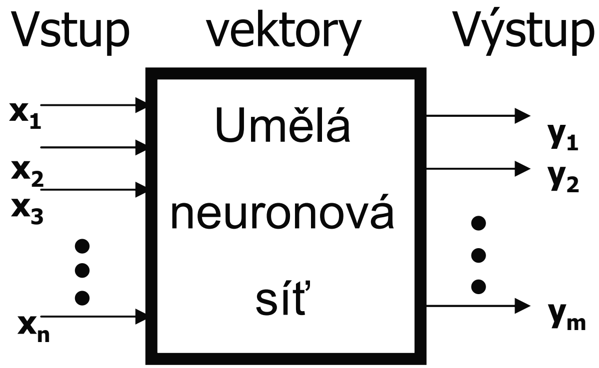
Neuronová síť to zvládne (viz. MADALINE), ale je třeba vymyslet, jak ji to naučit!

Zajímavé Intermezzo

Funkci ANN můžeme chápat jako transformaci *T* vstupního vektoru ***X***

na výstupní vektor ***Y***

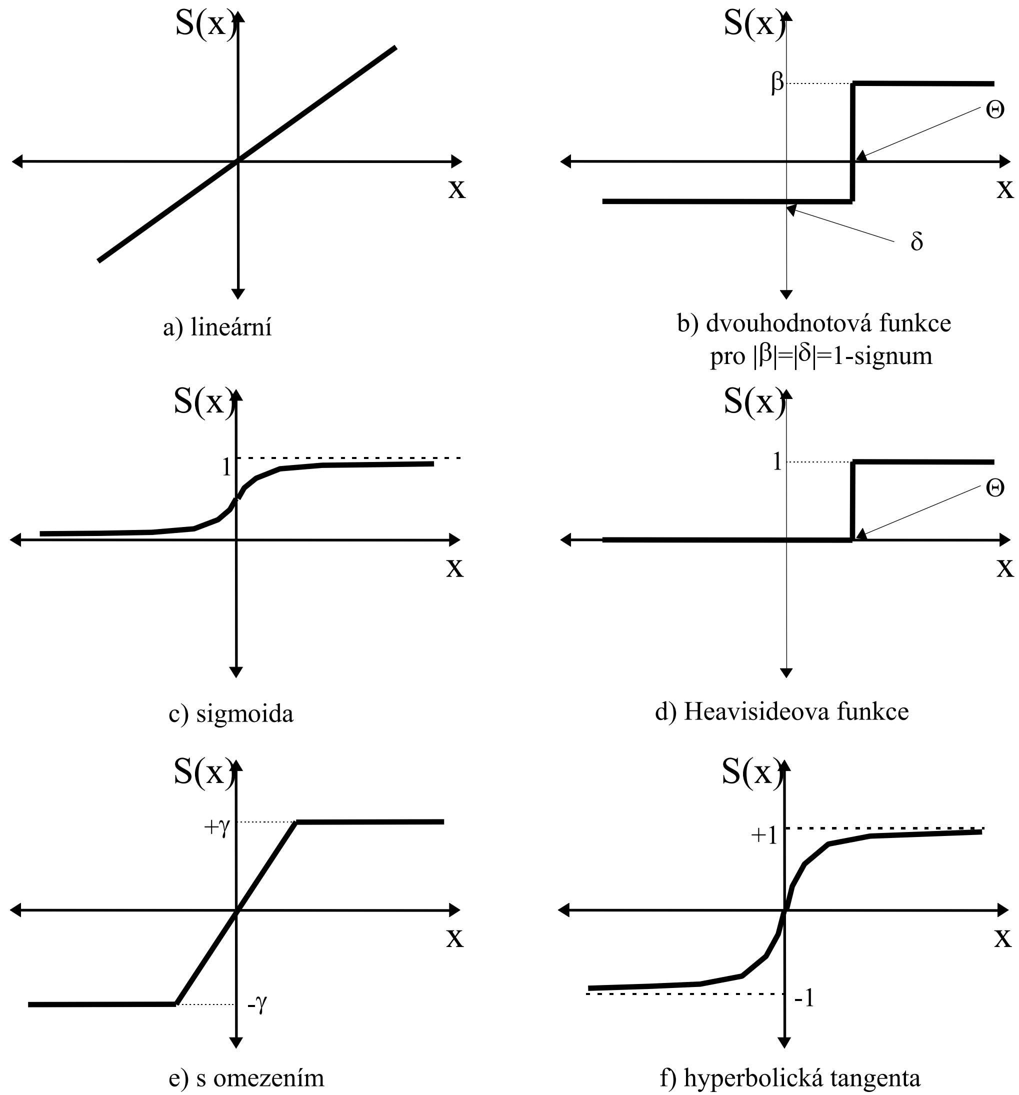
$$\boldsymbol{Y} = T(\boldsymbol{X})$$



Jaké všechny transformace *T* může ANN realizovat?

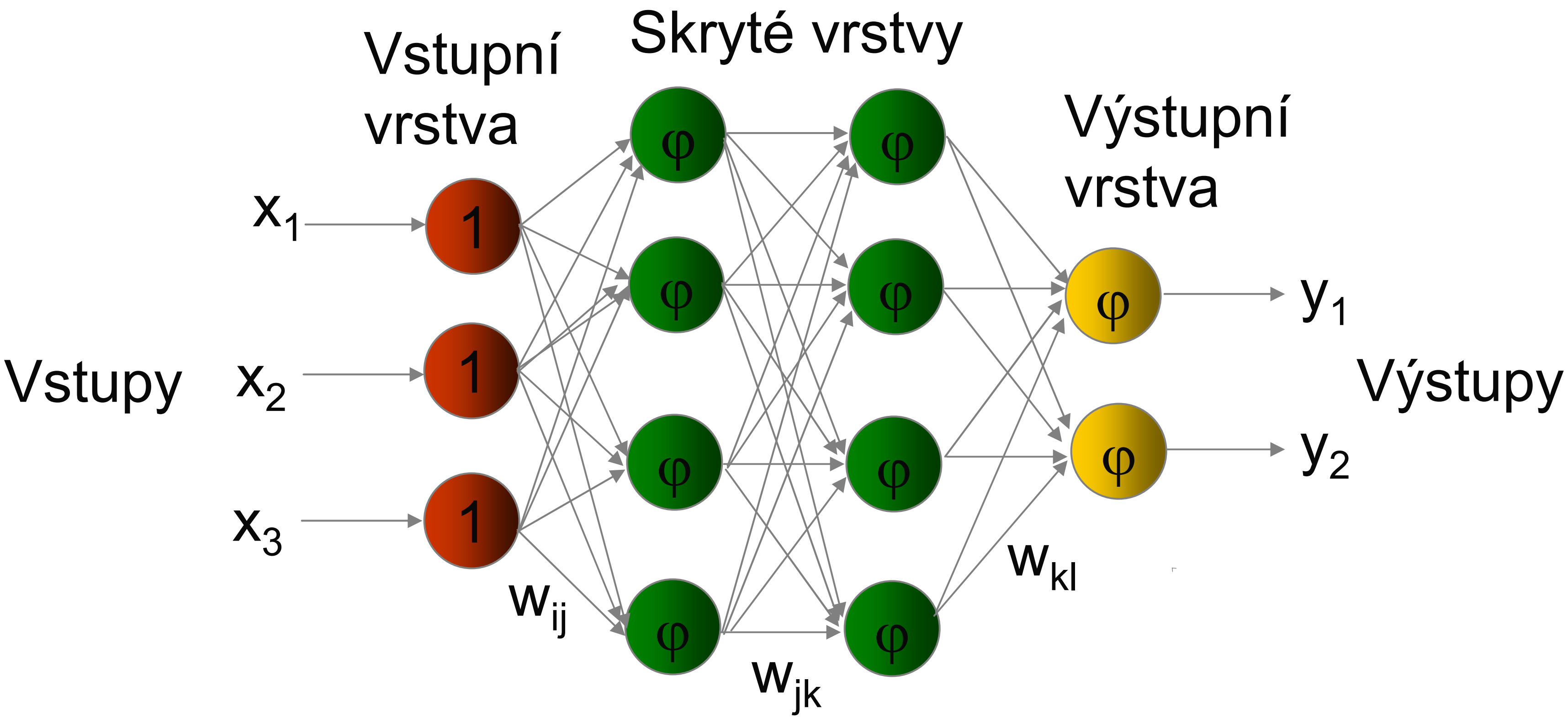
Toto byla vědecká výzva od počátku existence discipliny.

Průlom I: diferencovatelné aktivační funkce

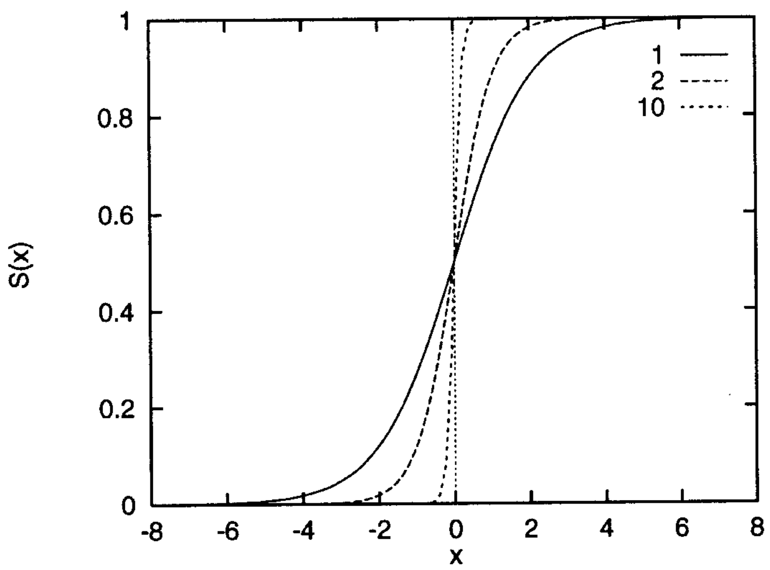


Průlom II: vícevrstvé sítě s dopředným šířením

- MLP (MultiLayered Perceptron)



Nelineární přenosová funkce MultiLayered Perceptronu



$$S(\varphi) = \frac{1}{1 + e^{-\gamma\varphi}}$$

Co je tam to φ?

Sigmoida zobrazená pro různé argumenty γ

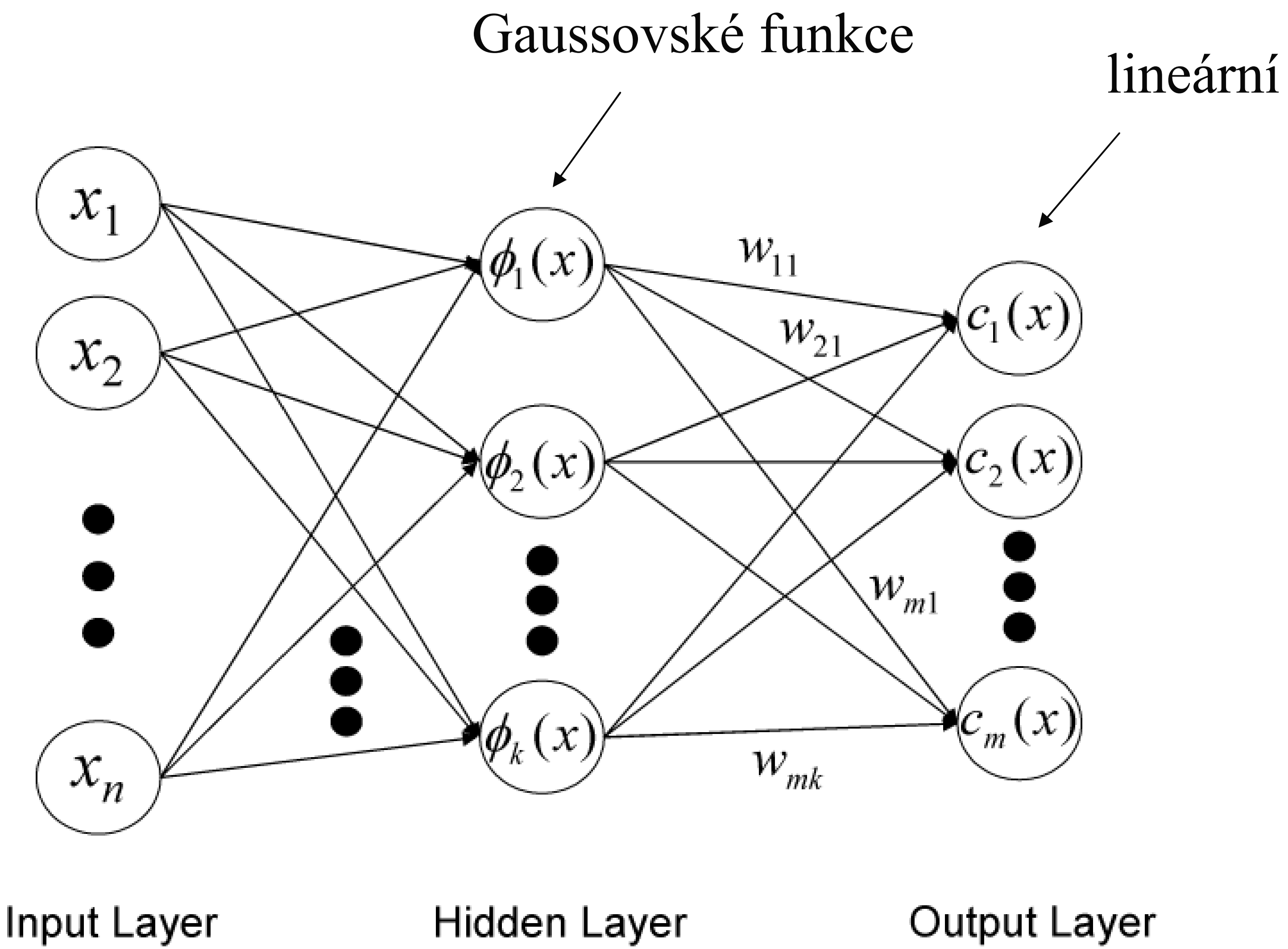
Průlom III: Algoritmus učení

- K natrénování MLP sítě lze použít algoritmus zpětného šíření chyby – Back propagation of error (zkráceně Backpropagation).
- MLP síť trénovaná Backpropagation algoritmem je dodnes nejpoužívanější neuronová síť.
- Více příští přednášku …

Další dopředné neuronové sítě

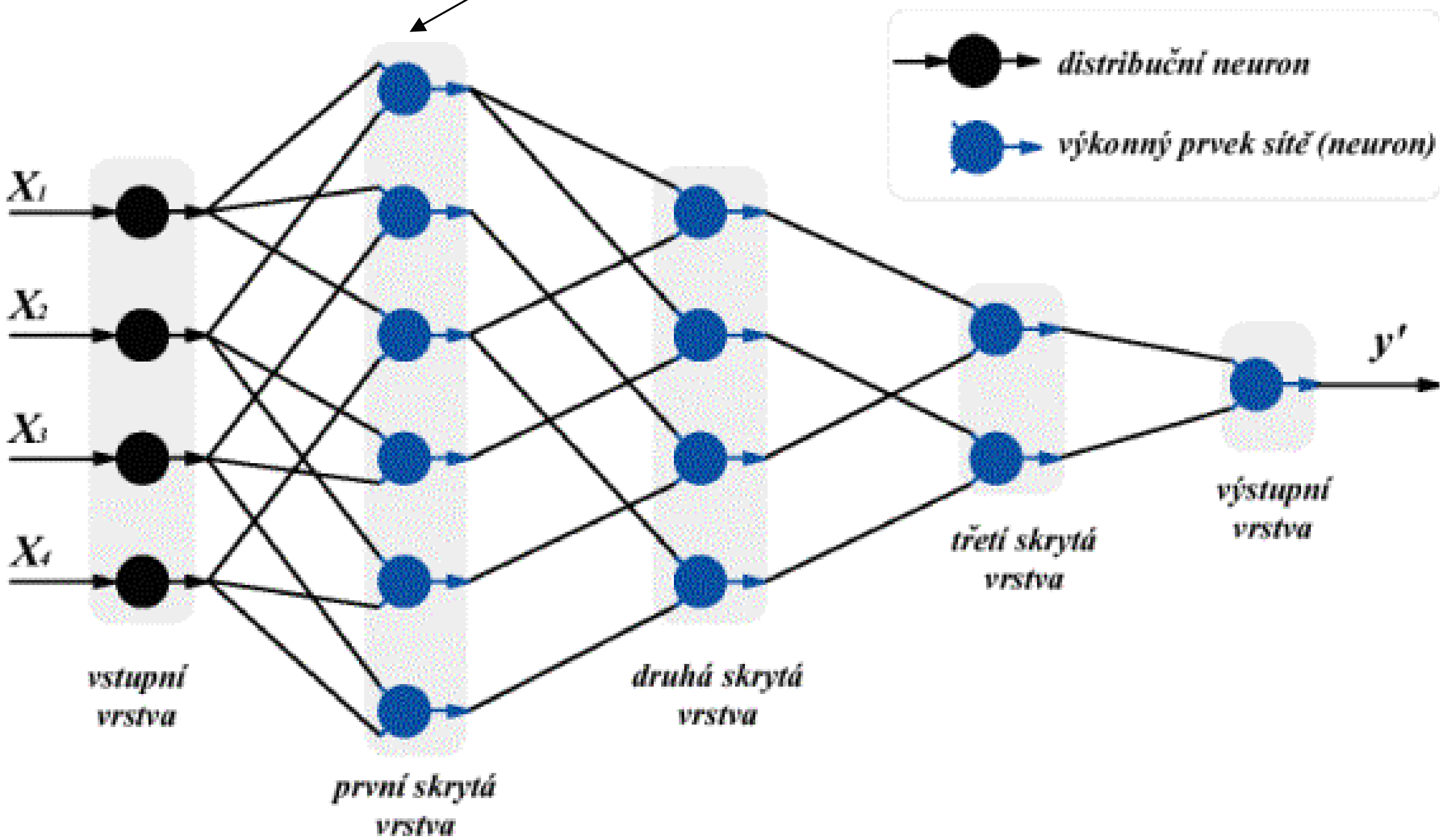
- Radial Basis Function Network
- MIA GMDH Network
- Cascade Correlation Network
- Neocognitron
- …

RBFN


Gaussovské funkce
lineární
x_1
x_2
x_n
$\phi_1(x)$
$\phi_2(x)$
$\phi_k(x)$
w_{11}
w_{21}
w_{m1}
w_{mk}
$c_1(x)$
$c_2(x)$
$c_m(x)$
Input Layer
Hidden Layer
Output Layer

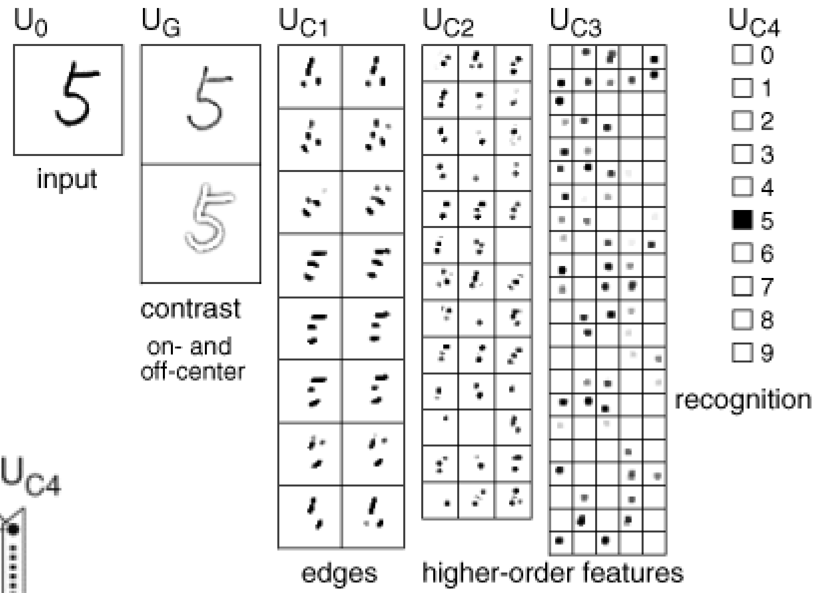
MIA GMDH

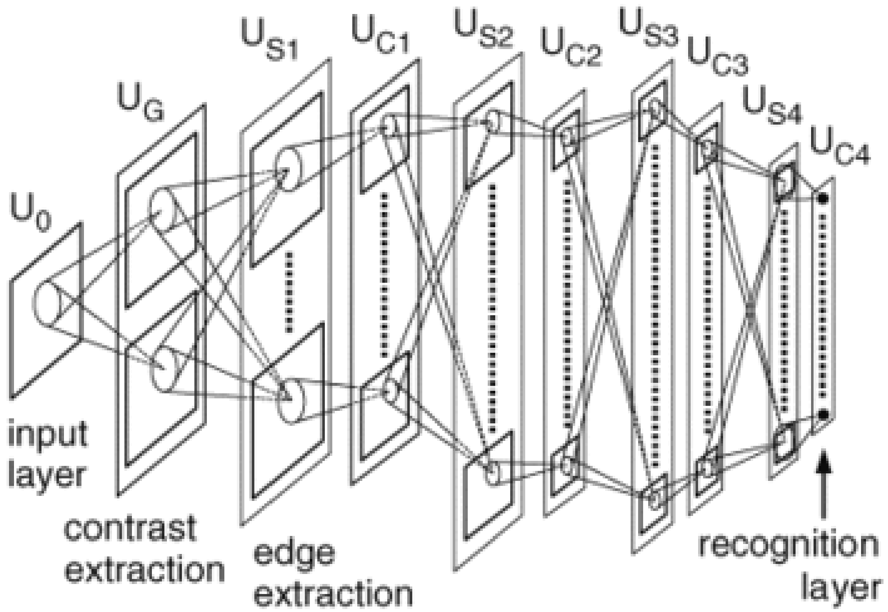
Neurony s polynomiální přenosovou funkcí



Neocognitron

Prof. Fukushima (1980)





Feedforward NN for rapid vision

- Serre, Thomas (2007)
- Hodně podobné

- U nás GOLOKO
(Brunner 1987)

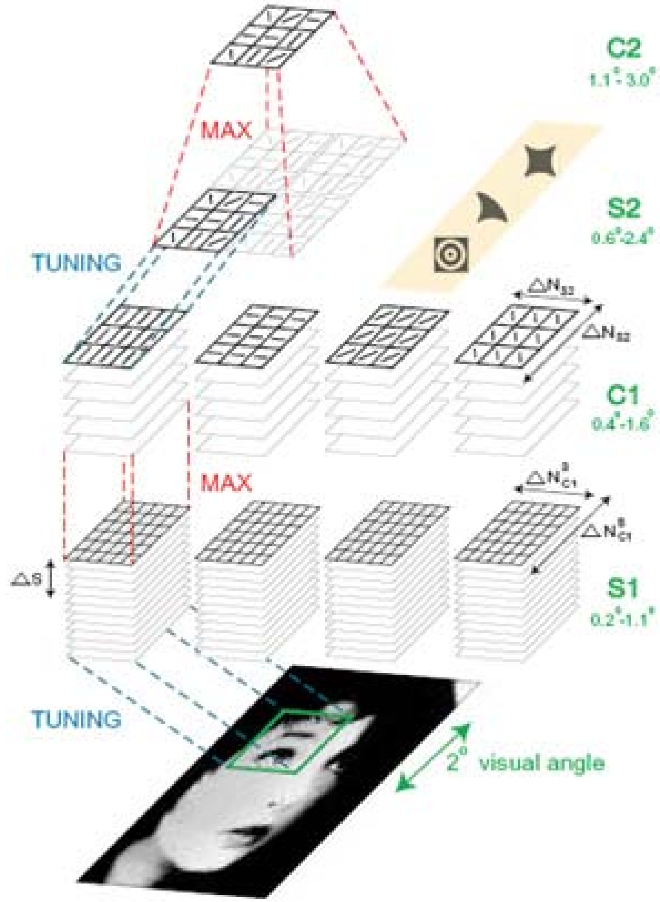
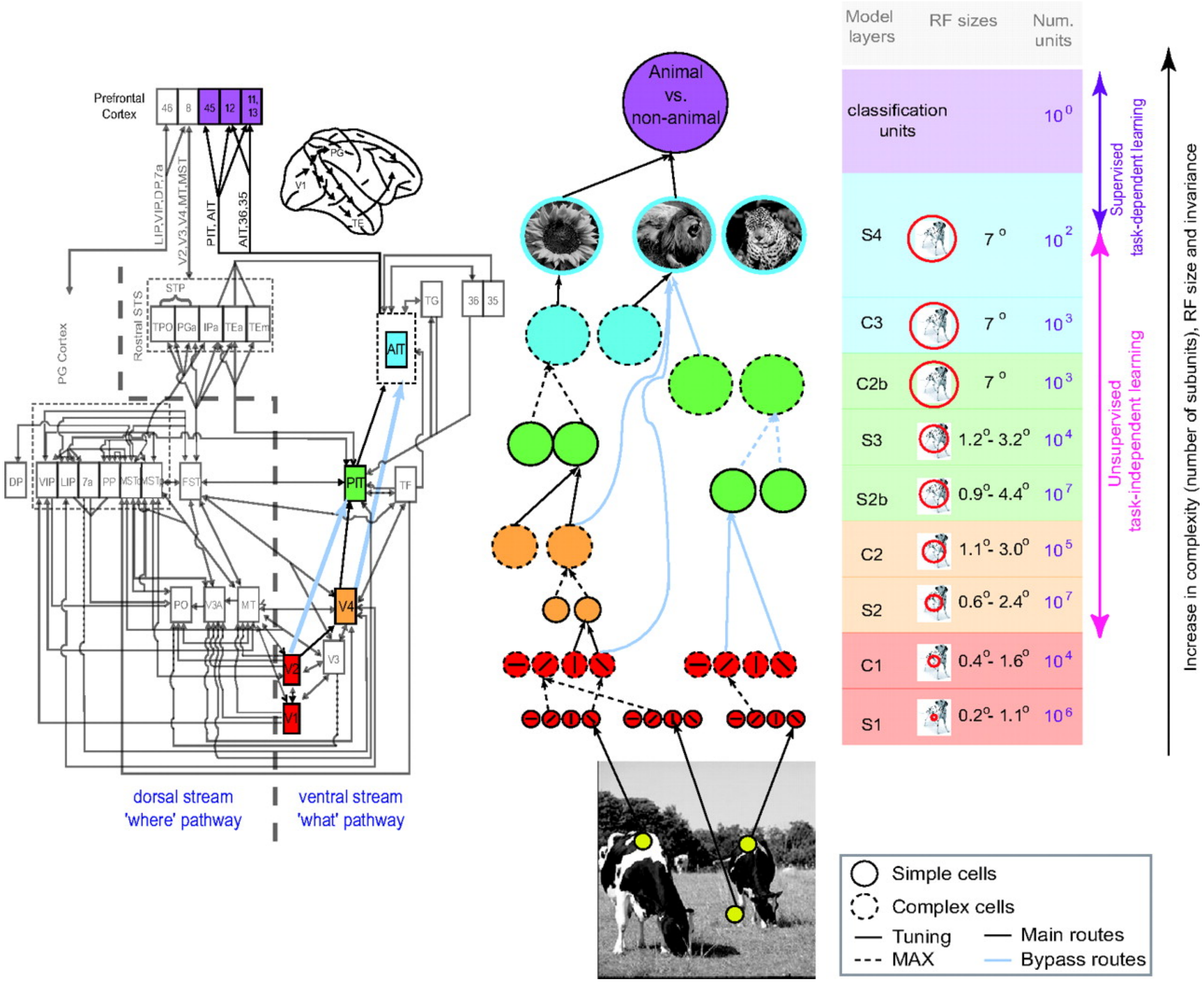


Fig. 1. Sketch of the model



Model layers	RF sizes	Num. units
classification units		10^0
S4	7°	10^2
C3	7°	10^3
C2b	7°	10^3
S3	1.2°- 3.2°	10^4
S2b	0.9°- 4.4°	10^7
C2	1.1°- 3.0°	10^5
S2	0.6°- 2.4°	10^7
C1	0.4°- 1.6°	10^4
S1	0.2°- 1.1°	10^6

Serre, Thomas et al. (2007) Proc. Natl. Acad. Sci. USA 104, 6424-6429

Neuronové sítě v data miningu

- Viz např. Industrial Conference on Data Mining 2007

http://www.informatik.uni-trier.de/~ley/db/conf/incdm/incdm2007.html

nebo

www.cs.uml.edu/~ckrieger/user/Neural_Networks.pdf

- Klasifikace
- Predikce
- Shlukování
- Identifikace
- Filtrování
- Asociace
- …