

Neuronové sítě pro humanitní studia

Doc. Dr. Ing. Vlasta Radová, katedra kybernetiky

1. Úvod

1.1. Motivace pro vývoj umělých neuronových sítí, významná historická data

Zájem o vývoj umělých neuronových sítí pramení z poznání, že lidský mozek pracuje jiným způsobem než běžné číslicové počítače. Počítače přesně a rychle provádějí posloupnosti instrukcí, které pro ně byly formulovány. Lidský mozek je tvořen neurony, které pracují přibližně milionkrát pomaleji než obvody číslicového počítače, přesto člověk dokáže lépe řešit řadu výpočetně náročných úkolů (zpracování vizuální informace, porozumění řeči, hraní šachů, a podobně). Proto je snaha napodobit schopnosti mozku a vytvořit umělou neuronovou síť, která by uměla simulovat chování biologické neuronové sítě živých organismů.

1943	považován za začátek vývoje umělých neuronových sítí, navržen McCullochův-Pittsův model neuronu
1949	Hebbův zákon učení (1. zákon učení neuronových sítí)
1958	navržen nejznámější model neuronu (perceptron), autorem Frank Rosenblatt
začátek 60. let 20. stol	první model neuronových sítí (lineární sítě), navržena pravidla pro jejich učení
polovina 60. let až polovina 80. let 20. stol.	pokles zájmu o neuronové sítě, někteří nadšenci přesto pokračují ve výzkumu (např. T. Kohonen navrhuje tzv. samoorganizující se sítě a S. Grossberg zákony pro jejich učení)
od poloviny 80. let 20. stol. dosud	období renesance neuronových sítí, objevují se tzv. vícevrstvé nelineární sítě a pravidla pro jejich učení

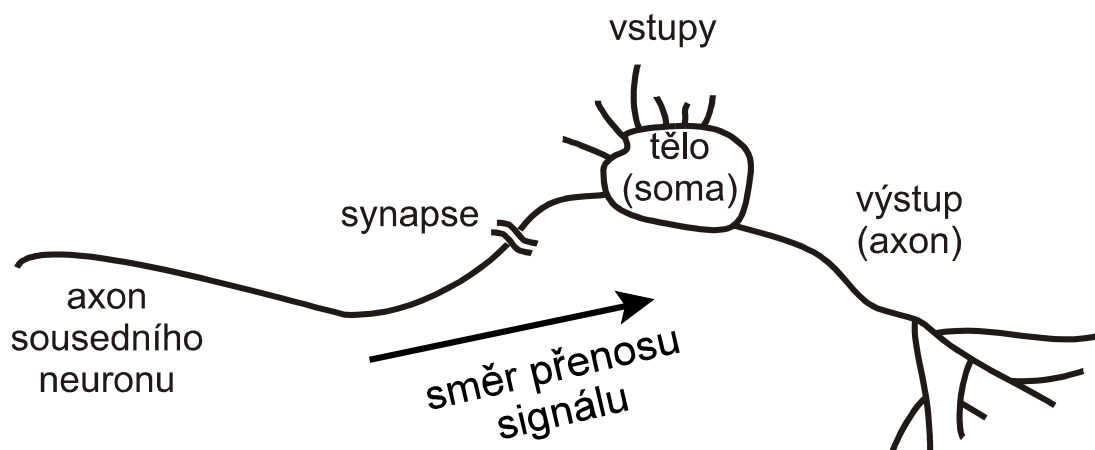
2. Základní pojmy. Model neuronu a neuronové sítě

2.1. Biologický neuron a jeho umělý model

Lidský mozek se skládá asi ze 100 miliard výpočetních elementů, které se nazývají neurony. Ty mezi sebou komunikují prostřednictvím sítě vazeb. Vstup do této sítě je umožněn prostřednictvím tzv. **receptorů**, které získávají podněty jednak z vnitřku těla, a jednak z vnějšku těla prostřednictvím smyslových orgánů (zrak, sluch, čich, chuť, hmat). Podnět je ve formě elektrických impulsů přenášen z receptorů do centrálního nervového systému, kde je zpracován. Podle výsledku zpracování jsou řízeny tzv. **efektory** (tj. výkonné orgány) a odezva člověka na daný podnět se projeví ve formě rozmanitých akcí.

2.1.1. Biologický neuron

Biologický neuron je základní buňkou biologických neuronových sítí.

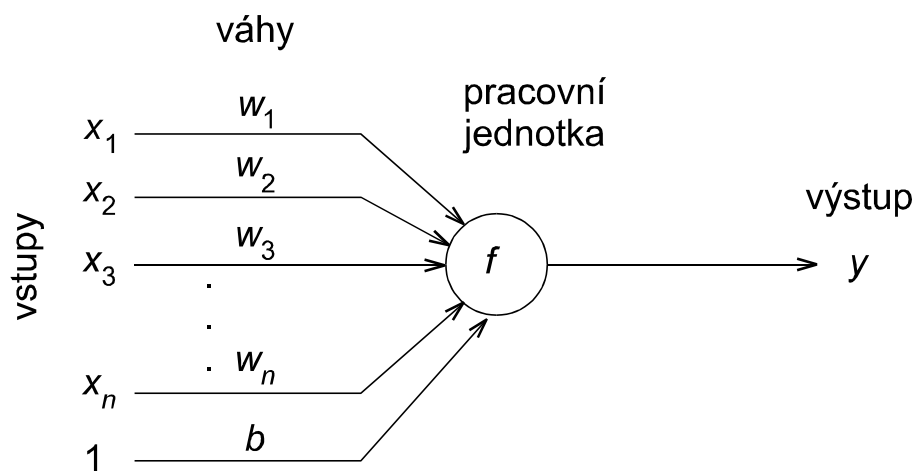


Skládá se ze 3 hlavních částí: tělo (**soma**), z něho vychází jeden poměrně dlouhý (až 60 cm) výstup (**axon**) a vstupem neuronu je velké množství (až několik tisíc) poměrně krátkých (asi do 3 mm) **dendritů**. Dendrity se s axony sousedních neuronů stýkají prostřednictvím rozhraní (tzv. **synapse**). Synapsí je v průměru 10 tisíc na každý neuron. Rozhraní fungují jako jednosměrné brány, tj. přenášejí signály pouze ve směru od axonu k dendritu, ale ne naopak. Přenášené signály jsou elektrické impulsy, jejichž přenos je ovlivněn uvolňováním chemických látek v synapsích. Tyto chemické látky působí jako tzv.

excitátory (budící signály), pokud umožňují, aby neuron generoval impuls (tzn. aktivoval se), nebo jako tzv. **inhibitory** (tlumící signály), pokud snižují schopnost následujícího neuronu generovat impuls. Aktivace neuronu nastane tehdy, překročí-li hodnota budících vstupních signálů hodnotu tlumících signálů o určitou hodnotu – tzv. **práh**.

2.1.2. Perceptron - model biologického neuronu

Tento model byl navržen v roce 1958 a dodnes se používá. Je tvořen pracovní jednotkou s několika vstupy a jediným výstupem.



Funkce perceptronu:

Hodnoty vstupních signálů $x_1, x_2, x_3, x_4, \dots$ jsou vynásobeny váhami $w_1, w_2, w_3, w_4, \dots$, v pracovní jednotce se sečtou, přičte se k nim práh b , výsledek se ztransformuje pomocí aktivační funkce f a poté je vygenerován odpovídající výstup y . Tato funkce perceptronu se v technické literatuře popisuje vzorcem

$$y = f(w_1 \cdot x_1 + w_2 \cdot x_2 + w_3 \cdot x_3 + w_4 \cdot x_4 + \dots + w_n \cdot x_n + b)$$

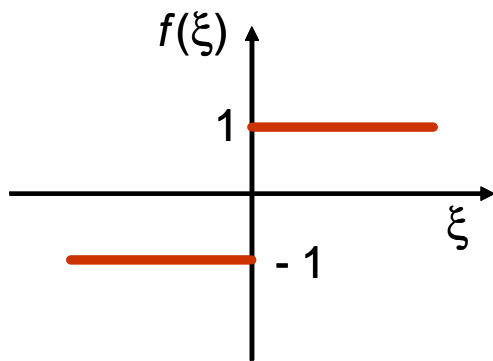
$$= f\left(\sum_{i=1}^n w_i x_i + b\right)$$

kde n určuje počet vstupů neuronu.

2.1.3. Nejčastěji používané aktivační funkce

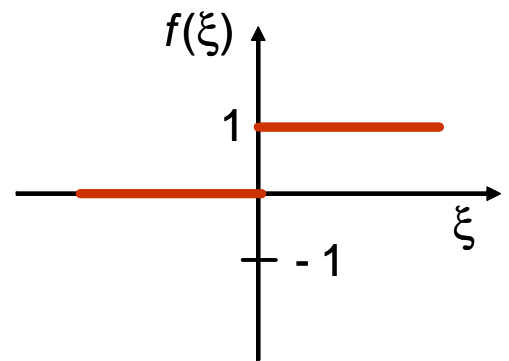
n bipolární binární

$$f(\xi) = \text{sgn}(\xi) = \begin{cases} +1 & \text{pro } \xi \geq 0 \\ -1 & \text{pro } \xi < 0 \end{cases}$$



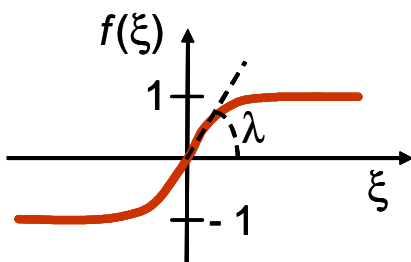
n unipolární binární

$$f(\xi) = \begin{cases} +1 & \text{pro } \xi \geq 0 \\ 0 & \text{pro } \xi < 0 \end{cases}$$



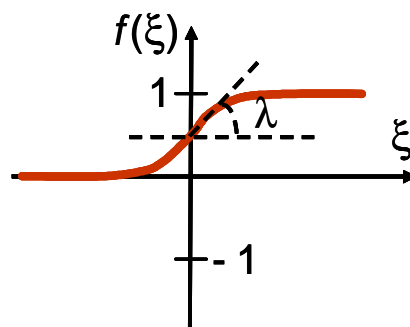
n bipolární spojitá

$$f(\xi) = \frac{2}{1 + \exp(-\lambda \xi)} - 1$$



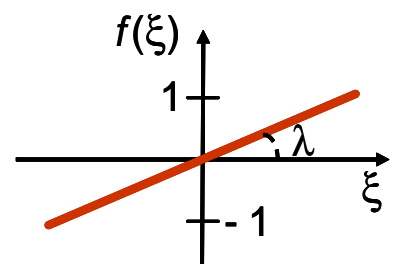
n unipolární spojitá

$$f(\xi) = \frac{1}{1 + \exp(-\lambda \xi)}$$



n lineární

$$f(\xi) = \lambda \xi$$

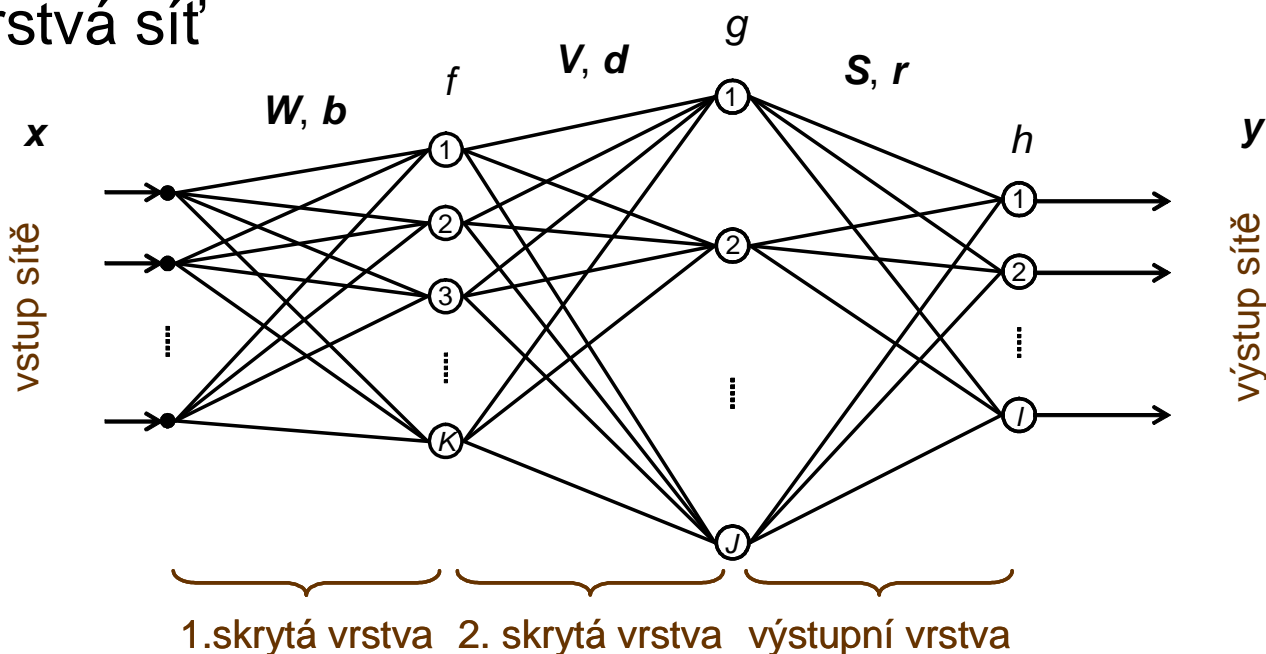


2.2. Modely umělých neuronových sítí

Umělá neuronová síť vznikne spojením jednotlivých modelů neuronů. Výsledná funkce sítě je určena způsobem propojení jednotlivých neuronů, váhami těchto spojení a způsobem činnosti jednotlivých neuronů. Nejpoužívanější jsou tzv. **vícevrstvé dopředné neuronové sítě**, kdy výstup jedné vrstvy je připojen na vstup následující vrstvy a signál se šíří pouze ze vstupu sítě na její výstup, a **sítě se zpětnou vazbou**, kdy se signál na rozdíl od vícevrstevných sítí šíří také z výstupu sítě zpět na její vstup.

2.2.1. Vícevrstvé neuronové sítě

n třívrstvá síť



$$\text{výstup } i\text{-tého neuronu: } y_i = h \left(\sum_{j=1}^J s_{ij} g \left(\sum_{k=1}^K v_{jk} f \left(\sum_{l=1}^L w_{kl} x_l + b_k \right) + d_j \right) + r_i \right)$$

x ... vstup sítě

y ... výstup sítě

W ... váhy 1.vrstvy

V ... váhy 2.vrstvy

S ... váhy 3.vrstvy

b ... prahy 1.vrstvy

d ... prahy 2.vrstvy

r ... prahy 3.vrstvy

L ... počet vstupů sítě

K ... počet neuronů v 1.vrstvě

J ... počet neuronů ve 2.vrstvě

l ... počet neuronů ve 3.vrstvě, tj. počet výstupů sítě

f ... aktivační funkce neuronů 1.vrstvy

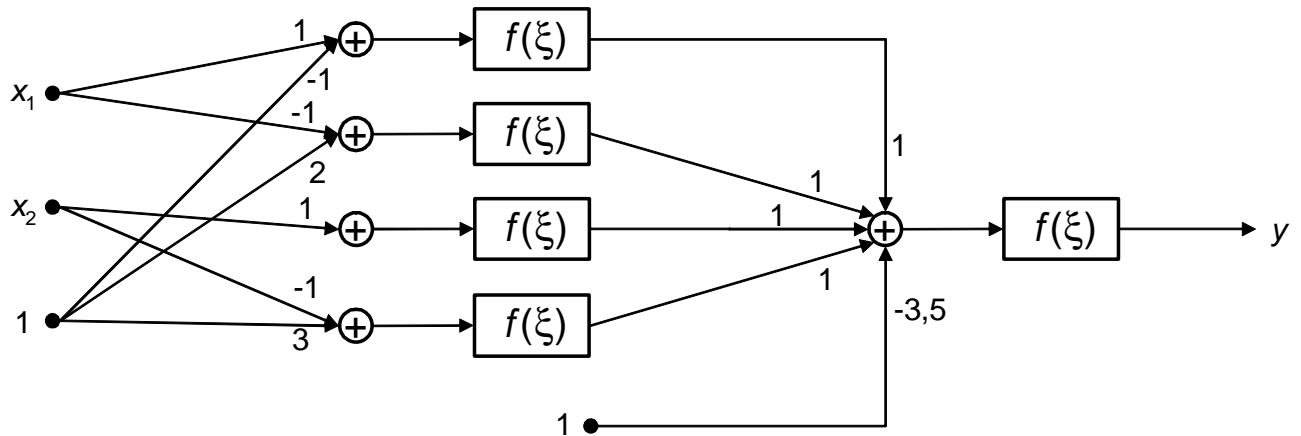
g ... aktivační funkce neuronů 2.vrstvy

h ... aktivační funkce neuronů 3.vrstvy

2.2.1.1. Činnost vícevrstevných neuronových sítí

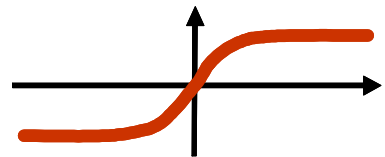
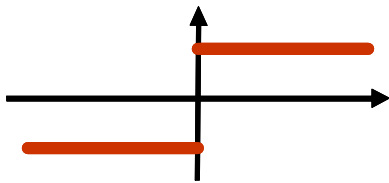
n Po předložení vstupu je vygenerován odpovídající výstup.

n Příklad: Analyzujte chování dvouvrstvé neuronové sítě znázorněné na obrázku (úkolem je zjistit závislost y na x_1 a x_2).



a) $f(\xi) = \text{sgn}(\xi) = \begin{cases} +1 & \text{pro } \xi \geq 0 \\ -1 & \text{pro } \xi < 0 \end{cases}$

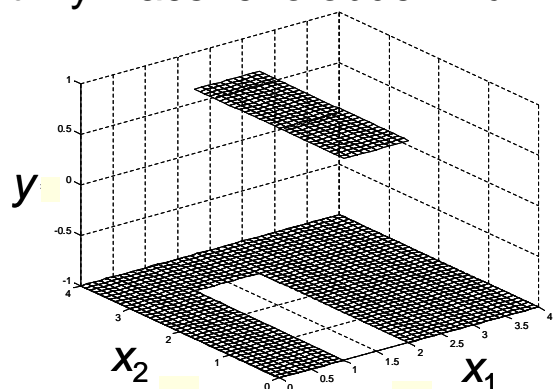
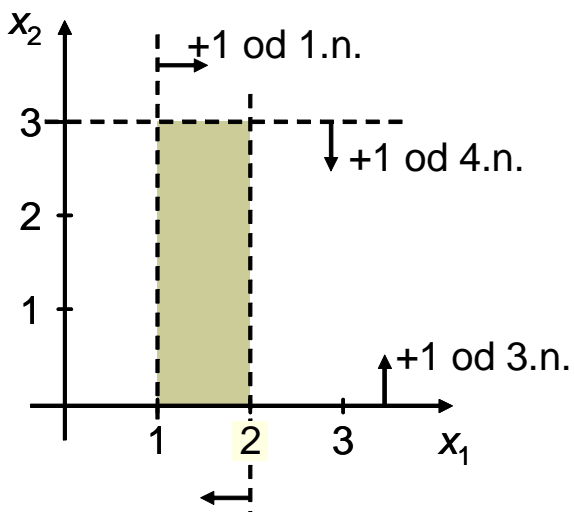
b) $f(\xi) = \frac{2}{1 + \exp(-\lambda \xi)} - 1$



n Řešení: ad a)

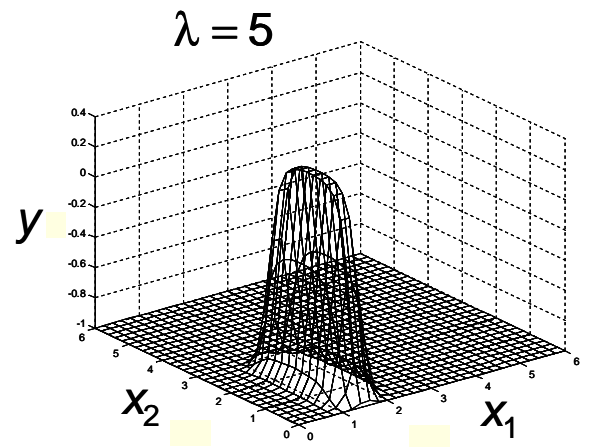
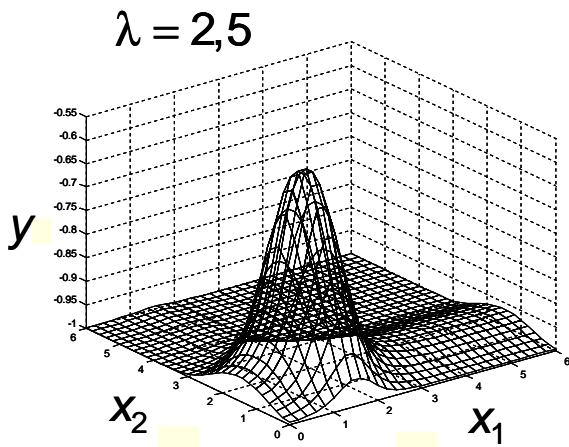
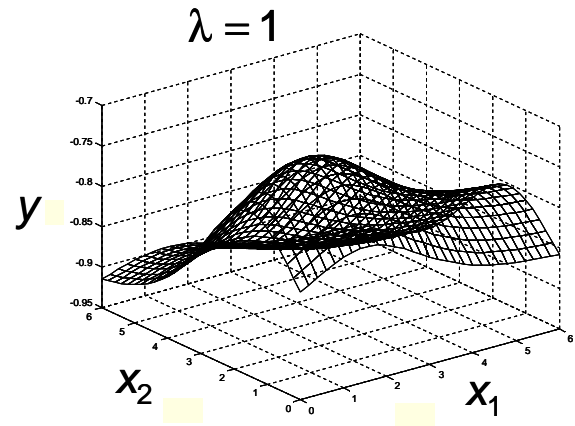
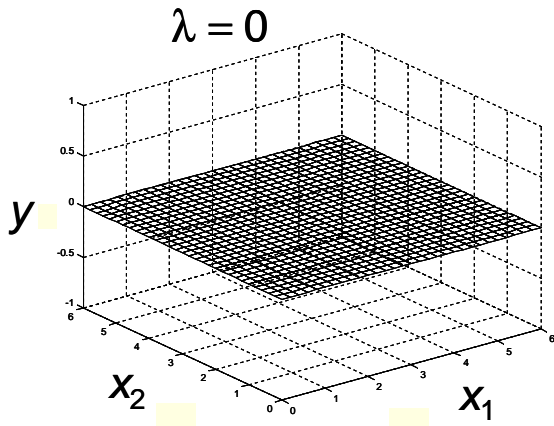
n výstup každého neuronu v 1.vrstvě může nabývat pouze hodnot +1 nebo -1, tzn. každý neuron rozdělí vstupní rovinu $x_1 x_2$ na 2 poloroviny. Jedné přiřadí +1 a druhé -1.

n na výstupu neuronu 2.vrstvy může být +1 pouze v případě, že na výstupech všech neuronů 1.vrstvy je +1, tzn. uvnitř vyznačeného obdélníku



n Řešení: ad b)

- n Každý neuron 1.vrstvy opět rozdělí vstupní rovinu na 2 části. Jedna část odpovídá kladným hodnotám na výstupu příslušného neuronu a druhá záporným. Přechod mezi polorovinami však není skokový, nýbrž plynulý. Neuron 2.vrstvy opět zkombinuje výstupy 1.vrstvy.



2.2.1.2. Fáze činnosti vícevrstvé neuronové sítě

- 1) fáze nastavování (učení , trénování)
- 2) fáze pracovní

1) Fáze učení

Cílem učení je nastavit váhy a prahy jednotlivých neuronů tak, aby síť prováděla požadovanou činnost. Existují 2 způsoby učení:

a) učení s učitelem (supervised learning)

Síť jsou postupně předkládány dvojice [vstup, požadovaný výstup], jinými slovy vstup a informace učitele o tom, jak má síť na tento vstup zareagovat. Po předložení vstupu je vypočten skutečný výstup, který se porovná s požadovaným výstupem, a trénovací algoritmus automaticky upraví hodnoty vah tak, aby rozdíl mezi skutečným a požadovaným výstupem byl minimální.

Poznámka: Množina dvojic [vstup, požadovaný výstup] tvoří tzv. **trénovací množinu**.

Trénovací množina:

[vstup 1, požadovaný výstup 1]

[vstup 2, požadovaný výstup 2]

[vstup 3, požadovaný výstup 3]

[vstup 4, požadovaný výstup 4]

.

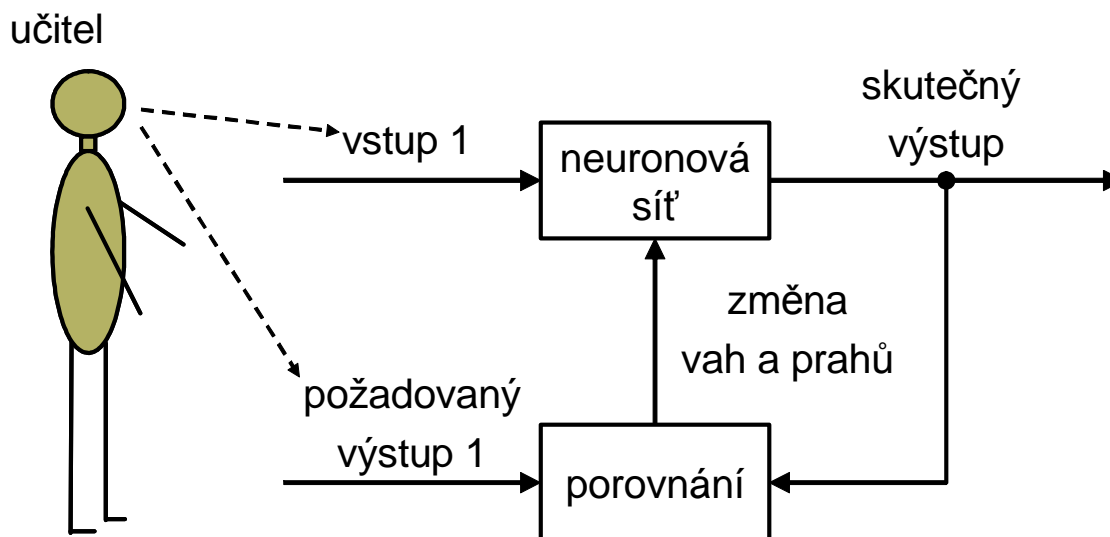
.

.

[vstup P , požadovaný výstup P]

Ilustrace učení s učitelem

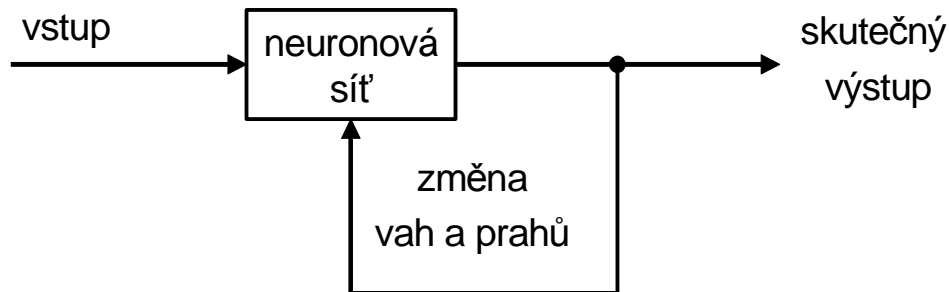
[vstup 1, požadovaný výstup 1]



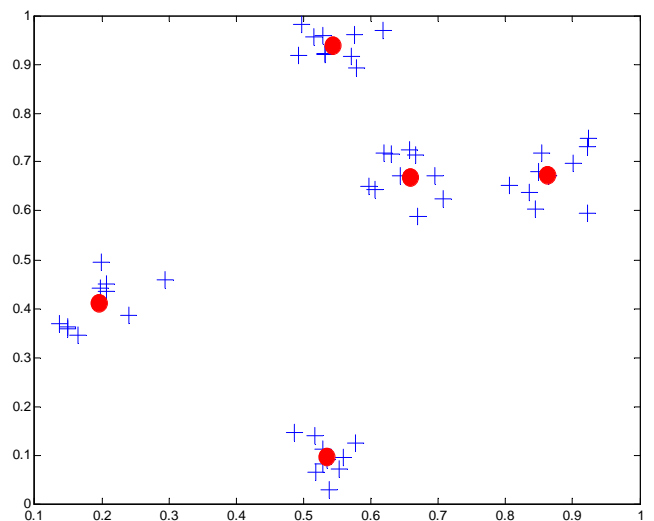
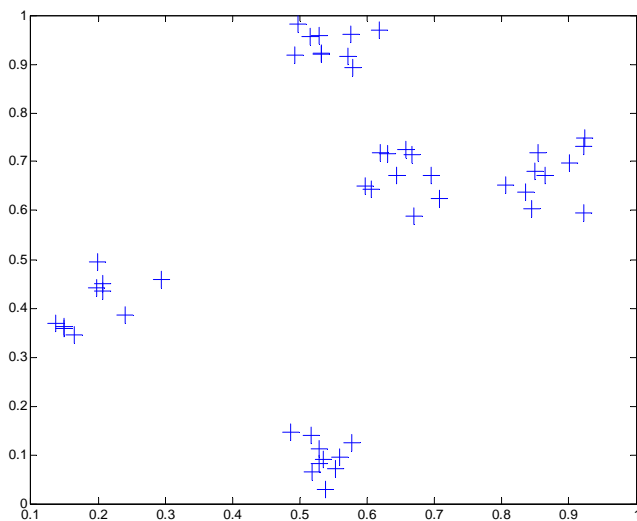
- Dvojice z trénovací množiny se přivádějí na síť opakovaně (opakování matka moudrosti)
- Podle chyby na výstupu se nastavují váhy a prahy předchozích vrstev, proto se algoritmus učení s učitelem se nazývá **algoritmus backpropagation** (algoritmus zpětného šíření chyby)

b) učení bez učitele (unsupervised learning)

Síť jsou předkládány pouze vstupy, informace učitele (tj. požadovaný výstup) chybí. Síť se sama snaží najít zákonitosti ve vstupních datech a nastavit své váhy a prahy tak, aby na podobné vstupy reagovala podobnými výstupy.



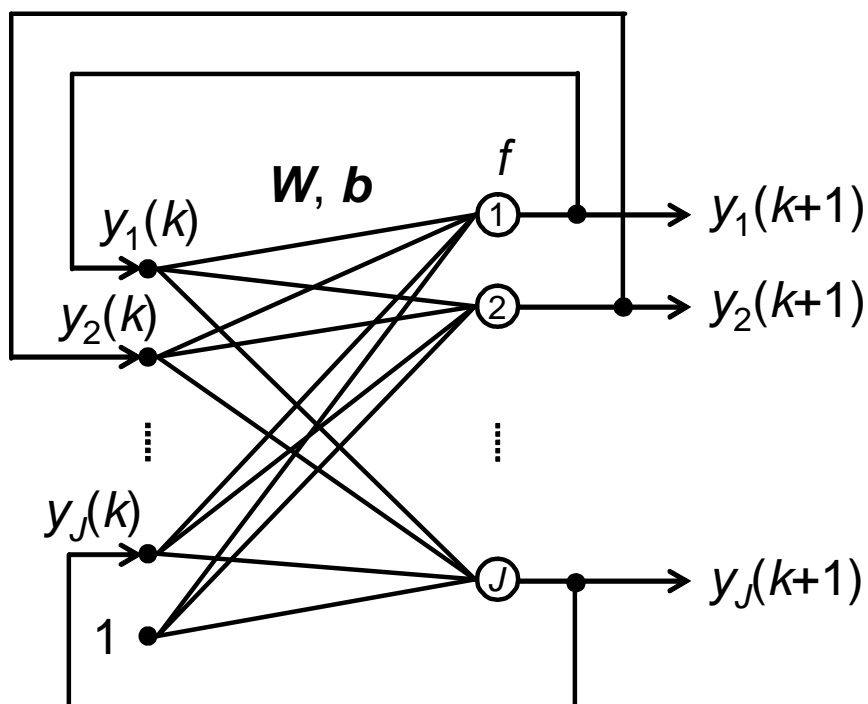
Podobnost je obvykle definována Eukleidovskou vzdáleností → dochází k tzv. **shlukování** vstupních dat.



2) Fáze pracovní

Neuronová síť reaguje na předložené vstupy podle nastavení, které se "naučila" ve fázi učení.

3. Neuronové sítě se zpětnou vazbou



W ... váhy sítě

b ... prahy sítě

J ... počet vstupů a výstupů sítě

f ... aktivační funkce

y ... výstup (stav) sítě

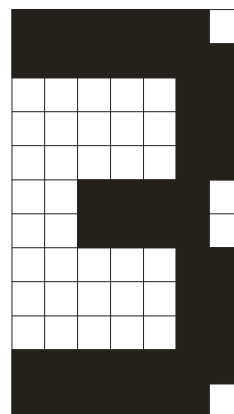
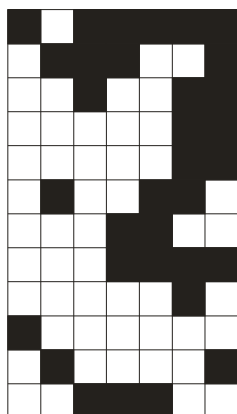
k ... čas

$y(0)$... inicializační stav sítě

$$\text{výstup } j\text{-tého neuronu: } y_j(k+1) = f\left(\sum_{i=1}^J w_{ji} y_i(k) + b_j\right)$$

3.1. Činnost neuronových sítí se zpětnou vazbou

- po inicializaci v čase $k=0$ se samovolně mění výstup sítě – síť přechází z jednoho stavu do jiného
- proces samovolné změny stavu končí v okamžiku, kdy síť dosáhne **rovnovážného stavu**, kdy se výstup sítě již nemění, anebo **rovnovážného cyklu**, který je tvořen periodicky se opakujícími stavy



3.2. Určení vah a prahů sítí se zpětnou vazbou

- k nastavení vah a prahů dochází pomocí tzv. **záznamového algoritmu**, kdy jsou do sítě zaznamenávány požadované rovnovážné stavy
- pro Hopfieldovu síť například platí

$$\mathbf{W} = \sum_{p=1}^P \mathbf{u}_p \mathbf{u}_p^T - P \mathbf{I}$$

\mathbf{u}_p ...tzv. prototypy, tj. rovnovážné stavy, které mají být do sítě zaznamenány

P ... počet prototypů

\mathbf{I} ... identická matice

4. Použití neuronových sítí

- při řešení problémů, kde není znám jednoznačný algoritmus řešení, ale kde existuje dost velká množina příkladů, jejichž řešení známe
 - rozpoznávání složitých signálů a obrazců
 - modelování funkčních závislostí
 - optimalizační úlohy
 - rekonstrukce zašuměných dat
 - ...
- neuronové sítě **nejsou vhodné** pro přesné výpočty, pro zpracování numerických úloh apod.

Příklady použití neuronových sítí

Klasifikátory	-	úkolem je zařadit vstupní data do skupin (tříd) podle vzájemné podobnosti
Aproximátory funkcí	-	z několika naměřených hodnot je třeba sestavit funkční závislost (např. predikce počtu slunečních skvrn, vývoj kursu koruny apod.)
Asociativní paměti	-	na základě předloženého vstupu je síť schopna “vybavit si” odpovídající výstup