

Samoorganizující se neuronové sítě (*Self – Organizing Maps*)

SOM

Neuronové sítě s učení bez učitele

(*unsupervised learning*)

Inspirace: organizace mozku do oblastí – map,
podmnožiny podobných vektorů jsou navzájem propojeny

Praktické aplikace: sítě tvořené asi 1 000 neurony, učení časově náročné,
výkonné počítače

Použití: zejména pro analýzu velkého počtu vstupních parametrů,
např. při technologických procesech (VLSI),
při rozpoznávání řeči – jen dílčí problémy,
pro identifikaci řečníka,
v robotice (řízení ramene robota v nepřístupném prostředí)

pohyb robota v labyrintu),
problém obchodního cestujícího,
při detekci poruch řeči,
při zpracování biologických signálů (EEG, EKG, MEG)



magnetoencefalograf

Princip: učení je založeno na schopnosti NN rozeznat ve vstupních vzorech stejné nebo blízké vlastnosti a třídít přicházející vektory podle nich. Podobné vektory sdružuje do shluků (*clusters*)

mění se vektory vah **nejsou porovnávány s požadovanými hodnotami** (*target*)

Použití: tam, kde neznáme správné řešení, pro shlukování, vizualizaci, pro abstrakci

Nelze použít: pro rozpoznání statistických veličin

Pozn: *při nesprávné volbě úlohy jsou výsledky trénování špatné*

Kritérium: výpočet vzdálenosti mezi vzory a aktuálními hodnotami,
hledání extrémů $\begin{cases} \rightarrow \text{minimální vzdálenost} \\ \rightarrow \text{maximální výstupní hodnota} \end{cases}$

Kompetitivní učení (**Competitive learning**)

zákon kompetice, soutěžní učení

sekvence vzorků : $x = x(t) \in R^n$ $t \dots$ čas

množina referenčních vektorů:

$$\left\{ w_i(t) : w_i \in R^n, i = 1, 2, \dots, k \right\}$$

$w_i(0)$ inicializace

*Pozn.: referenční vektory
bývají značeny také
jako m_i*

lokální propojení, signál je šířen také k sousedním neuronům,
každý vstupní vektor vzorů je v každém časovém okamžiku porovnáván
s každým vektorem W_j

Míra porovnání : vzdálenost $d(X, W_j)$

Index $j = c$ je index nejbližšího referenčního vektoru.

Pak vzdálenost $d(X, W_c)$ je minimum všech vzdáleností.

Vektor W_c se nazývá **vítěz v soutěži** (kompetici) (*winner, centroid*)

Míru naučenosti určujeme pomocí vzdálenosti resp. blízkosti reprezentace
vzorů.

Nejčastěji používané vzdálenosti:

- **Euklidovská** - pro pravoúhlý souřadnicový systém

$$\mathbf{d}_E(x, y) = \sqrt{(x_1 - y_1)^2 + \dots + (x_n - y_n)^2}$$

$$\mathbf{d}_E(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

- **Hammingova** – pro binární vektory (0, 1)

Určuje jaké množství elementů dvou vektorů je různé.

Lze ji použít na porovnání jakýchkoliv množin z diskrétních hodnot

- **Minkowskiho** – zobecnění Euklidovské vzdálenosti

$$\mathbf{d}_M(x, y) = \left(\sum_{i=1}^n |x_i - y_i|^\lambda \right)^{1/\lambda}$$

Elementy vektorů jsou buď binární hodnoty (0, 1) nebo písmena abecedy.

$$\mathbf{x} = (1, \mathbf{0}, \mathbf{1}, 1, \mathbf{1}, \mathbf{0})$$

$$\mathbf{y} = (1, \mathbf{1}, \mathbf{0}, 1, \mathbf{0}, \mathbf{1})$$

$$\mathbf{u} = (\mathbf{p}, \mathbf{a}, \mathbf{t}, \mathbf{t}, \mathbf{e}, \mathbf{r}, \mathbf{n})$$

$$\mathbf{w} = (\mathbf{w}, \mathbf{e}, \mathbf{s}, \mathbf{t}, \mathbf{e}, \mathbf{r}, \mathbf{n})$$

$$d_H(\mathbf{x}, \mathbf{y}) = 4$$

$$d_H(\mathbf{x}, \mathbf{y}) = 3$$

spatially

temporally

Hodnoty signálu sousedící v **prostoru** nebo **v čase** jsou představovány vzorky (**patterns**) uspořádaná množina reálných čísel - vektor

Elementy vektoru $X = (x_1, x_2, \dots, x_n)$ představují souřadnice v prostoru n -té dimenze

Koncepce deterministická nebo stochastická (přirozenější)

Hodnoty vzorků jsou pravděpodobností (*samples*), mohou nabývat diskrétní nebo spojité hodnoty.

Jde o funkci rozložení pravděpodobnosti

Kde je možné najít literaturu?

Na Internetu :

- Kaski, S., Kangas, J., Kohonen, T.: Bibliography of self-organizing map
- (SOM) papers: 1981-1997. Neural Computing Surveys, 1(3&4):1-176, 1998

Knihy: Kohonen, T.: Self-Organizing Maps. Berlin Heidelberg, **3-rd ed.** Springer Series in Information Sciences, Springer-Verlag, 2001, ISBN 3-540-67921-9

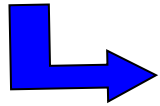
Software: <http://www.cis.hut.fi/research/software.shtml>

SOM_PAK (UNIX) <http://www.cis.hut.fi/software.shtml>

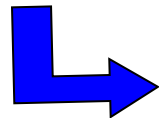
SOM Toolbox (Win NT) <http://www.mbnet.fi/~phodju/nenet/Nenet/General.html>

Kohonenovo učení

základní myšlenka



prostorová reprezentace dat



kompresa dat

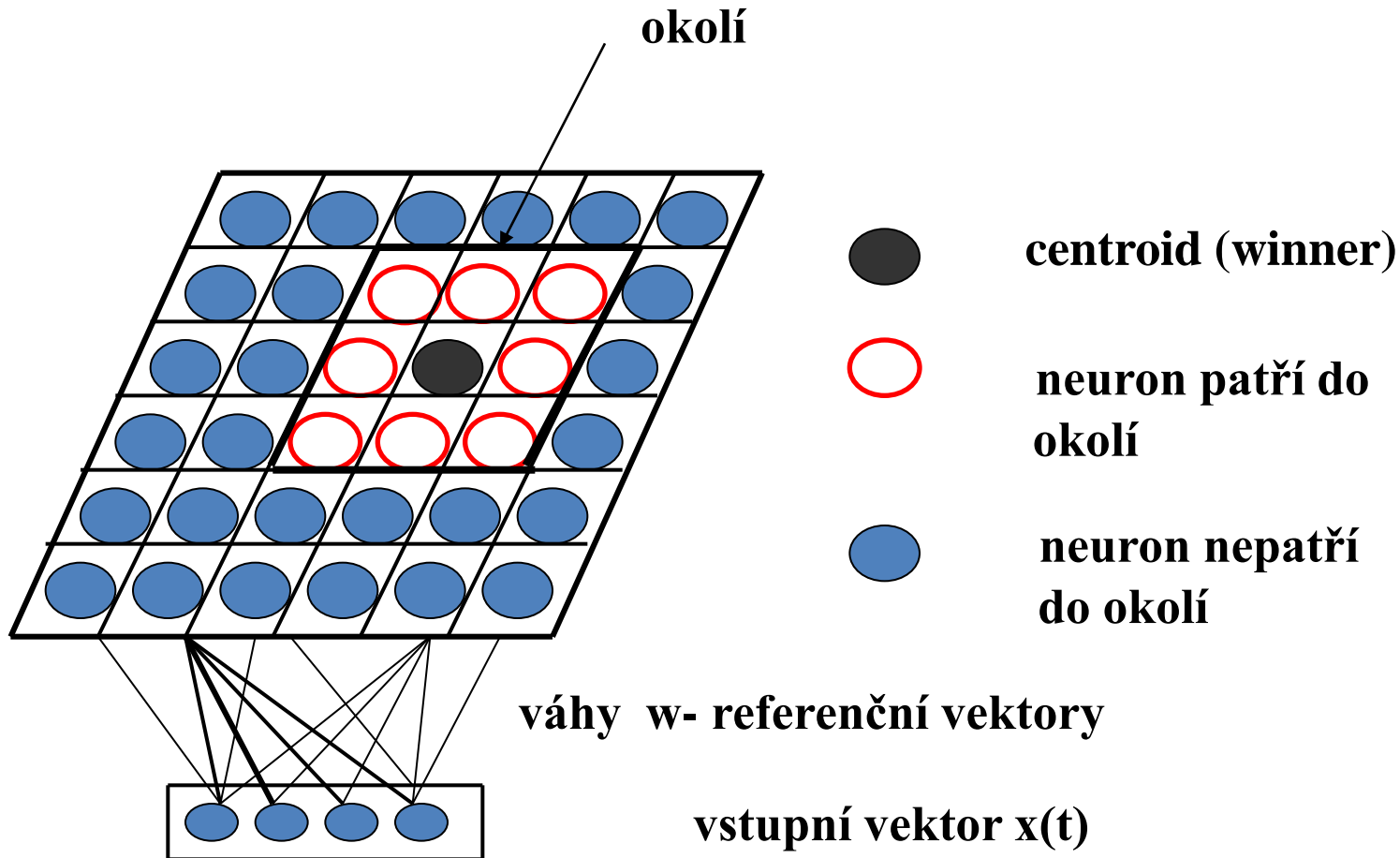
transformace z n - rozměrného na m - rozměrný prostor, $n > m$

propojení mezi sousedními neurony

model neuronu McCulloch-Pitts

nepracuje se s aktivační funkcí, ale s funkcí okolí (*neighborhood function*)

výpočet vzdáleností a jejich minimalizace



předem zvolená topologie mapy a okolí – *často čtvercová nebo hexagonální*
 způsob modifikace velikosti okolí

Kohonenův algoritmus učení (Delta pravidlo)

pp. Euklideovská metrika

Obecně: mapa nejčastěji dvojdimenzionální (princiálně neexistuje

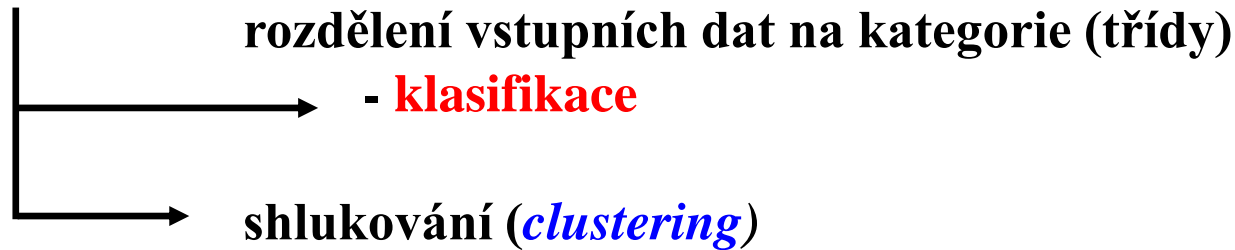
konvergence učení - závisí na mnoha parametrech – jsou to:

- *inicializační hodnoty vstupních parametrů a vah*
(*hodnoty referenčních vektorů*),
- *počet neuronů a způsob jejich organizace v mapě,*
- *tvar a velikost sousedního okolí i-tého neuronu,*
- *rychlost učení*

při učení nejsou jednotlivé neurony na sobě navzájem závislé,
sdružují vektory s podobnými vlastnostmi

je hledán *vítězný neuron* (*winner*)

cíl:



Shluk (**cluster**) je tvořen určitým počtem vstupních vektorů, které se sobě nejvíce podobají (mají nejvíce společných nebo velmi blízkých vlastností).

Je to oblast v N-dimenzionálním prostoru (data jsou popsána N vlastnostmi – *features*) s relativně vysokou hustotou bodů a je oddělena od ostatních oblastí s relativně nízkou hustotou bodů.

- Důležité:**
- rozmístění v mapě
 - počet dominantních vlastností v rámci jednoho tréninkového procesu
 - posun ve vstupních datech a „přeřazení“ některé vlastnosti do jiné skupiny při opakování procesu.

Problém: počet shluků

Možný počet rozdělení P vzorků do K tříd: $K^P / K!$ jen pro malý
Určení míry

Určení míry podobnosti (*similarity measure*) resp. míry neshody
počet vzorů

Nejčastěji vzdálenost

vítězný neuron:

$$d(X, W_c) = \min_i [d(X, W_i)]$$

$$d_j(X, W) = \sqrt{\sum_{i=1}^n (X_j - W_i)^2}, \quad j=1, \dots, N$$

$$X = [x_1, x_2, \dots, x_N],$$

$$W_i = [w_{i1}, w_{i2}, \dots, w_{iN}]$$

Poloměr okolí N_c je funkcí času, s rostoucím časem se zmenšuje, proces končí pro $N_c = c$

Okolí je totožné s vítězným neuronem 
Kohonenův algoritmus přejde na jednoduché kompetiční učení.

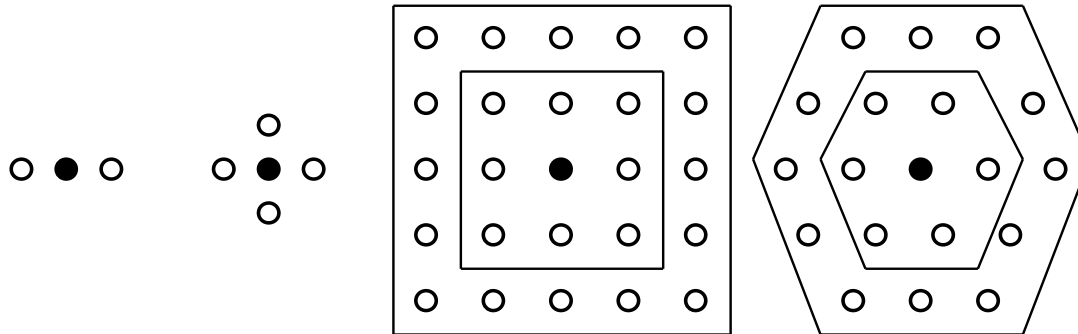
Vektory blízké neuronu „c“ patří do jeho okolí, jejich váhy jsou adaptovány v soulase s algoritmem učení. Tvoří jednotlivé shluky. Váhy neuronů ležících vně okolí „ N_c “ se nemění.

Do okolí (*neighbourhood*) se klasifikují všechny vstupní vektory nejvíce se podobající vítězi.

Vždy klasifikační úloha.

Topologie mapy (kompetitivní vrstvy) i okolí budoucího vítěze jsou předem zvolené.

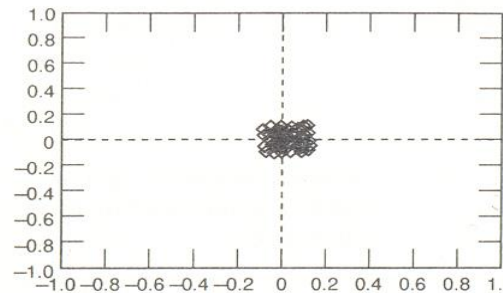
Příklady topologií Kohonenovy mapy a okolí vítězného neuronu.



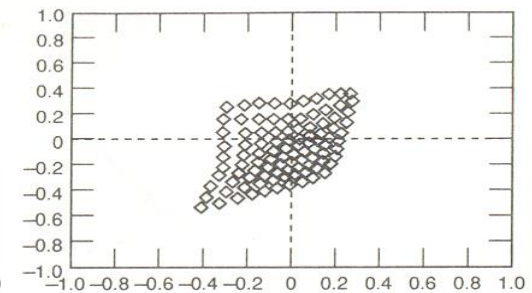
Váňové vektory v
KSOM:

- a) rozložení při inicializaci
- b) po 20-ti iteracích
- c) po 40-ti iteracích
- d) po 100 iteracích

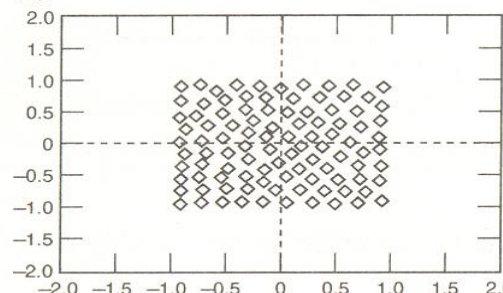
20 Mathematical background for neural computing



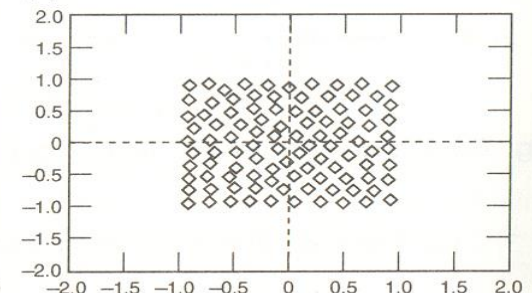
(a)



(b)



(c)



(d)

Vektory vah během trénování, 1-D mapa

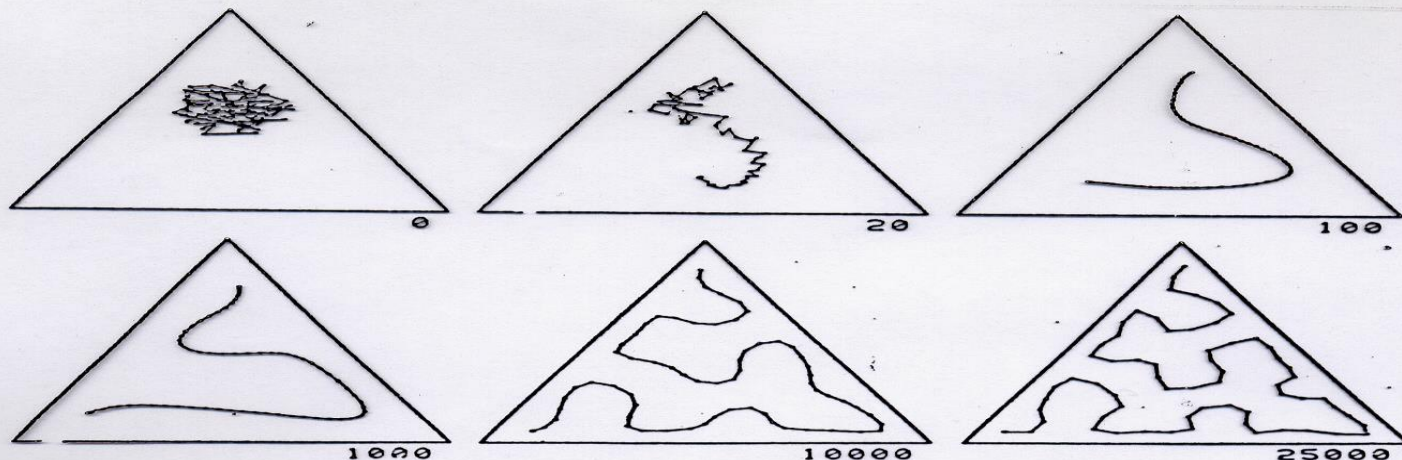


Fig. 4. Weight vectors during the ordering process, one-dimensional array.

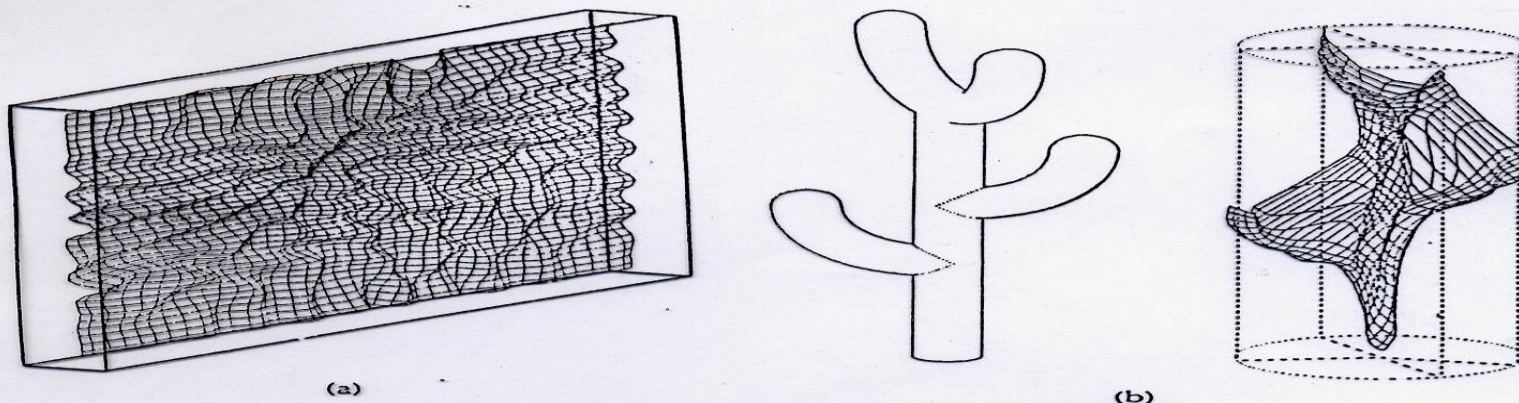


Fig. 5. Representation of three-dimensional (uniform) density functions by two-dimensional maps.

Reprezentace 3-D funkce rozložení pomocí 2-D mapy