

Kohonen Self-organizing Feature Maps



Kohonenove neurónové siete

Samorganizácia

- **Systém** – skupina navzájom sa ovplyvňujúcich prvkov, ktoré pôsobia ako jeden celok, ktorého hranice sú rozpoznateľné od okolia.
- **Vlastnosti systému** – nie je to len súhrn vlastností častí, má čosi viac.
- **Organizácia** -
externá organizácia – ovládaná externými faktormi,
samoorganizácia - evolúcia systému do organizovaného tvaru bez vonkajších vplyvov.
- Čo je atraktor systému?
Preferovaný stav systému, taký, že ak systém odštartoval z iného stavu, vyvíja sa tak, aby sa dostal do atraktora.

Ako sa mozog učí spontánne bez vplyvu učiteľa?

Prvý filozofický prístup:

- Postulation of homunculus – malý mužík žijúci niekde v mozgu, ktorý pôsobí ako rozhodca/tútor pri učení sa
- **Kanadský príspevok: Donald Hebbov prístup:**
 - explicitne určené podmienky, ktoré umožňujú zmenu na synaptickej úrovni reflektovať aj pri učení sa a v pamäti (1949):

“Keď axon bunky A dosť blízky k vybudeniu bunky B a opakovane alebo persistentne sa zúčastňuje pri jej vybudení,
určitý rastový proces alebo metabolické zmeny sa udejú v jednej alebo v oboch bunkách tak dôjde k tomu, že potenciál A je zvýšený, keď A je jednou z buniek, ktoré vybudili bunku B”

Ako sú tieto zmeny dosiahnuté?

- 1) Zvýšením # prenášaných impulzov cez synaptickú väzbu
- 2) Zvýšením veľkosti synaptickej väzby
- 3) Tvorbou nových synapsíí

Záver:

Mozog je samoorganizujúci sa systém, ktorý sa dokáže učiť sám zmenou (pridaním, odstránením, zosilnením) prepojení medzi neurónmi.

Mapy základných rysov

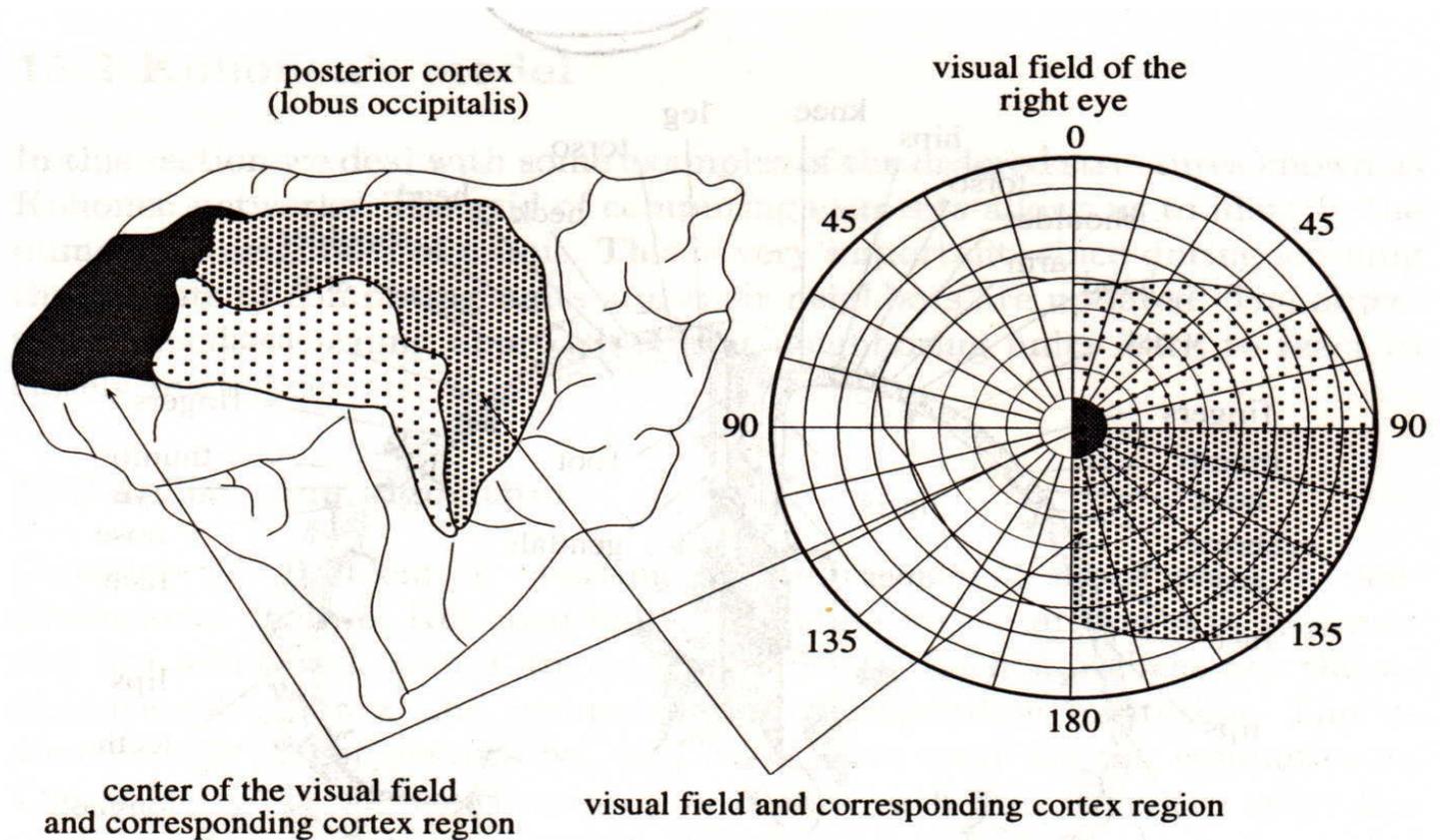
Aký je výsledok samoorganizácie mozgu?

- vytvorenie máp základných rysov v mozgu, ktoré majú lineárnu alebo planárnu topológiu (teda sú 1- alebo 2- dimenzionálne)

Príklady:

- **tonotopické mapy** – zvukové frekvencie sú priestorovo mapované do oblasti kortexu a sú usporiadane od nízkych k vysokým frekvenciám.
- **retinotopické mapy** – vizuálne pole je mapované vo vizuálnej oblasti kortexu (okcipitálny lalok) s vyššou kvalitou pre centrum tohto poľa
- **somatosensorické mapy** – mapuje dotyky

Príklad

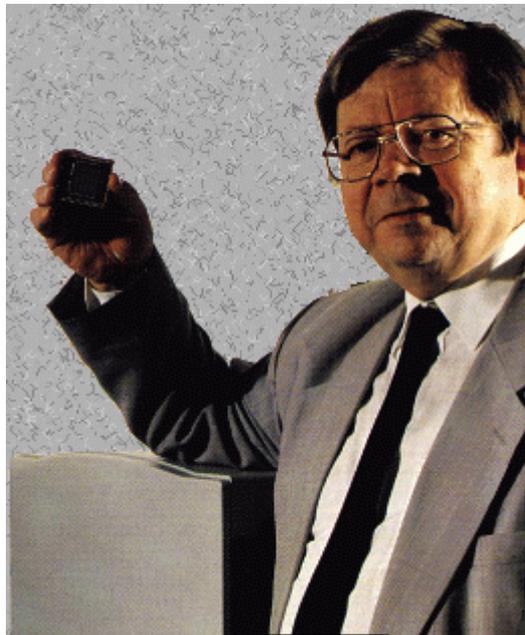


Mapping of the visual field on the cortex

Model Kohonovej neurónovej siete

Kohonenove siete, súťažný filter asociatívnych pamätí

- reprezentuje naše predchádzajúce idey
- Pomenované po Dr. Eng. Teuvo Kohonenovi



Model Kohonovej neurónovej siete

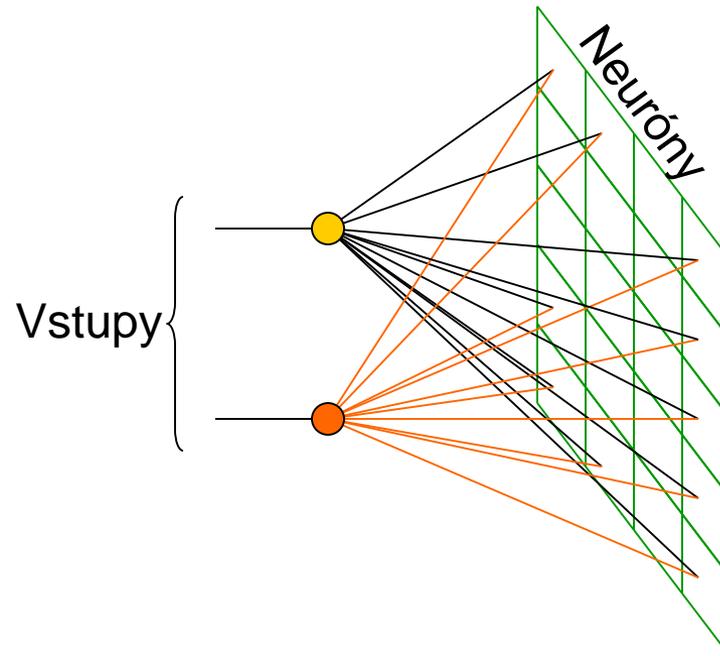
Špecifickou črtou SOM je to, že (pri splnení istých podmienok) umožňuje **realizovať zobrazenie zachovávajúce topológiu** a zobrazí tak charakteristické príznaky (črty) trénovacej množiny dát.

Za týmto účelom sú neuróny zoradené v pravidelnej, zväčša dvojrozmernej alebo jednorozmernej štruktúre (mriežka alebo reťaz).

Model Kohonenovej neurónovej siete

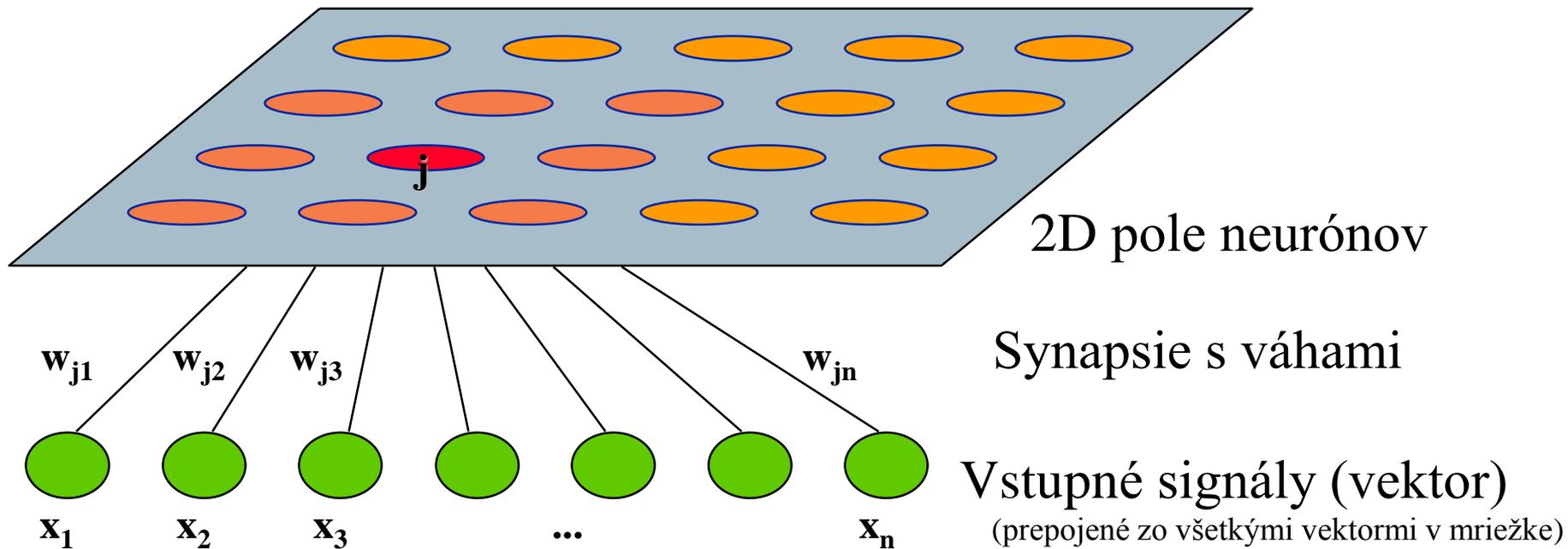
- Uvažované usporiadanie neurónov predstavuje výstupný priestor, v ktorom vzdialenosť dvoch neurónov je obyčajne daná ich **euklidovskou vzdialenosťou vektorov** v uvažovanej štruktúre.
- Zobrazenie zachovávajúce topológiu, ktoré vznikne po natrénovaní SOM, má dôležitú vlastnosť: **ľubovoľné dva vzory blízke vo vstupnom priestore evokujú v sieti odozvy na neurónoch, ktoré sú tiež fyzicky blízke** (vo výstupnom priestore).

Model Kohonovej neurónovej siete



Architektúra SOM

Výpočtová vrstva je zložená z mriežky neurónov.
Tým je daná topológia.



Základné vlastnosti siete

- Vstupné jednotky sú spojené s každým neurónom na mriežke
- Víťazný neurón je jeden, a to ten, ktorého váhy sú najbližšie k vstupu
- Neuróny participujú v hre “víťaz berie všetko”:
 - Výstup víťazného neurónu je nastavený na 1 a všetky ostatné sú nastavené na 0
 - Len váhy k víťazovi a jeho susedom sú adaptované

Susedné neuróny

- Symetrické okolie víťaza, maximálna zmena hodnôt váh je aplikovaná blízko centra (víťaza)
- Excitátor blízko centra, obklopený hranicou inhibície, vplyv excitátora rýchlo klesá na nulu

Siet'ové rovnice: podobnosť

- Podobnosť je meraná pomocou Euclidovej vzdialenosti od vstupného vzoru - vektor **input_p**
- Vektor váh **w[i]** reprezentuje všetky váhy do neurónu *i*; teda, **w[i][j]** je váha spájajúca vstup *j* a neurón *i*

$$s_p[i] = \| \mathbf{w}[i] - \mathbf{input}_p \| = \sqrt{\sum_j (\mathbf{w}[i][j] - \mathbf{input}_p[j])^2}$$

- Víťaz má najmenšiu hodnotu *s*

Siet'ové rovnice: adaptácia váh

- Predpokladajme že víťazný neurón má index α
- Pre vstup j a neurón i , je zmena váhy vyjadrená $\Delta \mathbf{w}[i][j]$ a nová hodnota váhy je $\mathbf{w}[i][j]$:
 - $\Delta \mathbf{w}[i][j] = \eta (\mathbf{input}[j] - \mathbf{w}[i][j]) \text{NbdWt}(\alpha, i)$
 - $\mathbf{w}[i][j] = \mathbf{w}[i][j] + \Delta \mathbf{w}[i][j]$

kde

- η je učiaci pomer a
- $\text{NbdWt}(\alpha, i)$ je susedská váhová funkcia

Siet'ové rovnice: susedská váhová funkcia

- ▣ Susedská váhová funkcia môže mať tvar:

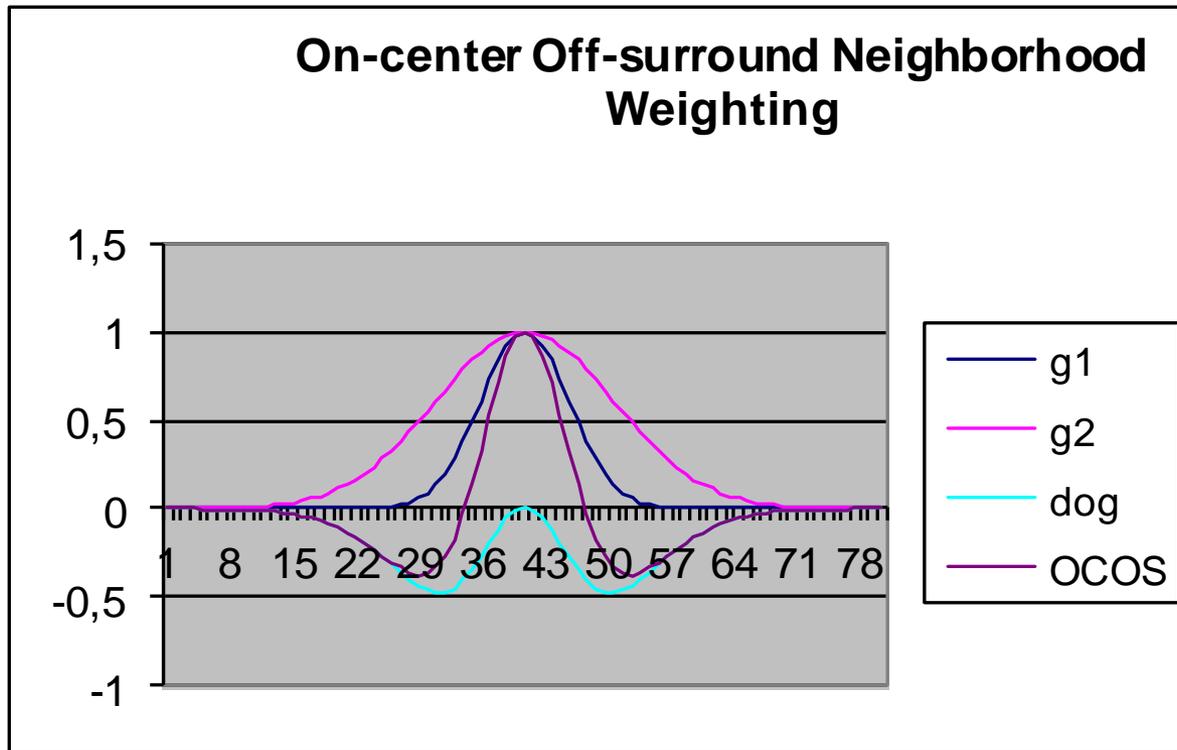
$$NbdWt(\alpha, i) = \frac{1}{1 + \left(\frac{d(\alpha, i)}{s}\right)^2} \quad NbdWt(\alpha, i) = e^{-\left(\frac{d(\alpha, i)}{s}\right)^2}$$

- ▣ kde s je skalár, ktorý určí dilatáciu váhovej funkcie

Centrum a okolie - susedstvo

- Oblasť, ktorá:
 - Vylepší stimul “blízko” centra (centrum)
 - Zhorší efekt vplyvu v oblasti okolo centra
- Prispieva k riadeniu a lokalizuje reprezentáciu

Centrum a okolie - susedstvo



Mexický klobúk pre susedstvo

Dve Gaussove funkcie

$$g_1(x) = e^{-\left(\frac{x}{s_1}\right)^2} \quad \text{a} \quad g_2(x) = e^{-\left(\frac{x}{s_2}\right)^2}$$

kde $s_2 > s_1 > 0$.

Nech $h(x) = g_1(x) - g_2(x)$, je ich rozdiel

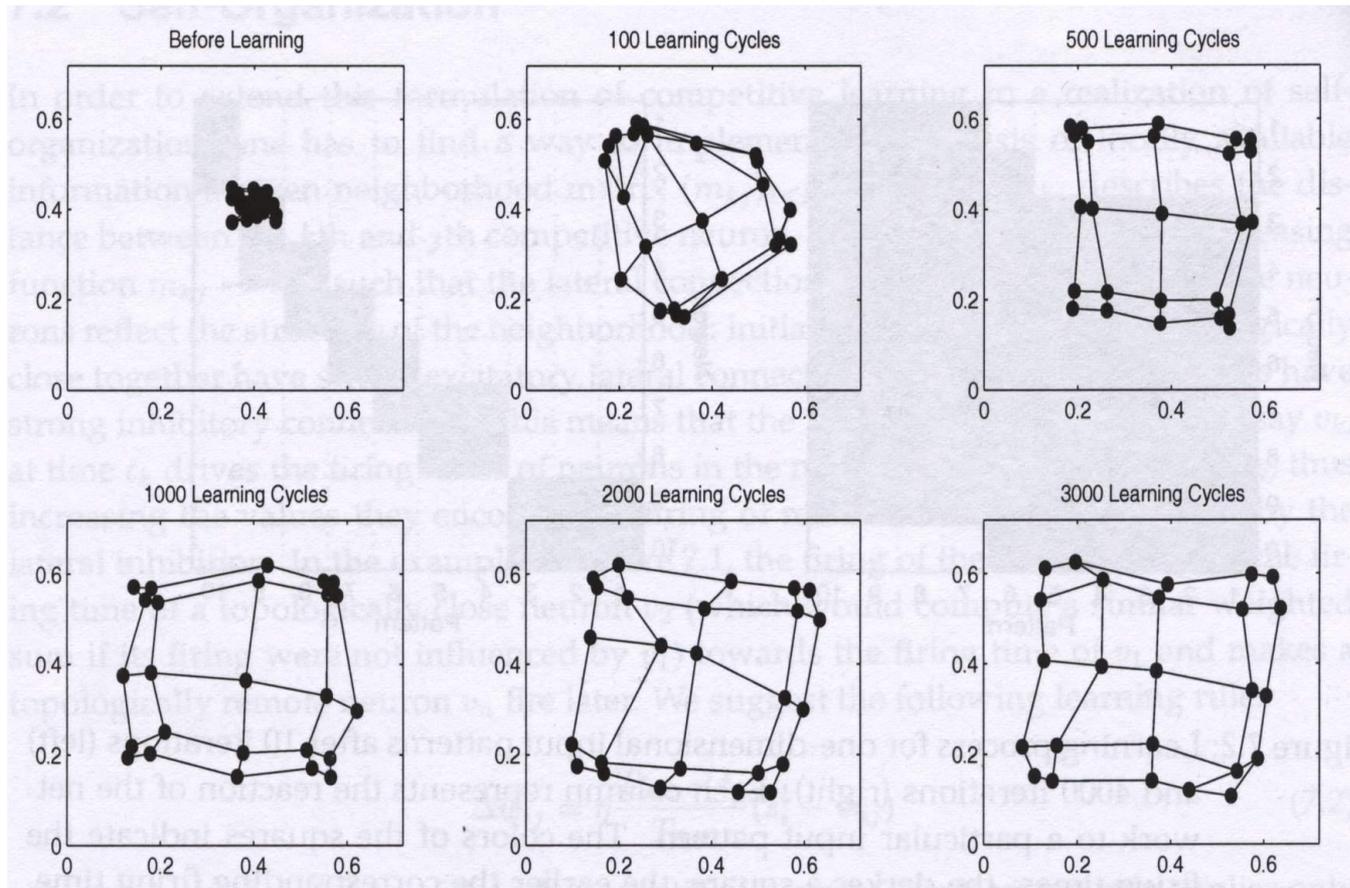
Funkcia Mexický Klobuk je :

$$\text{MexHat}(x) = g_i(x) + \alpha \cdot h(x) \quad \text{kde } \alpha > 0.$$

Základný trénovací algoritmus

- Inicializovať váhy od vstupných jednotiek k neurónom
- Vybrať funkciu susednosti
- Pokiaľ (vstupné vzory sa nezhodujú s váhami)
 - Nájsť víťaza
 - Adaptovať váhy v závislosti od vplyvu víťaza
- Navrhnuť interpretáciu pre kódovanie — identifikovať, ktorý neurón ktoré vzory kóduje

Trénovanie Kohonovej siete



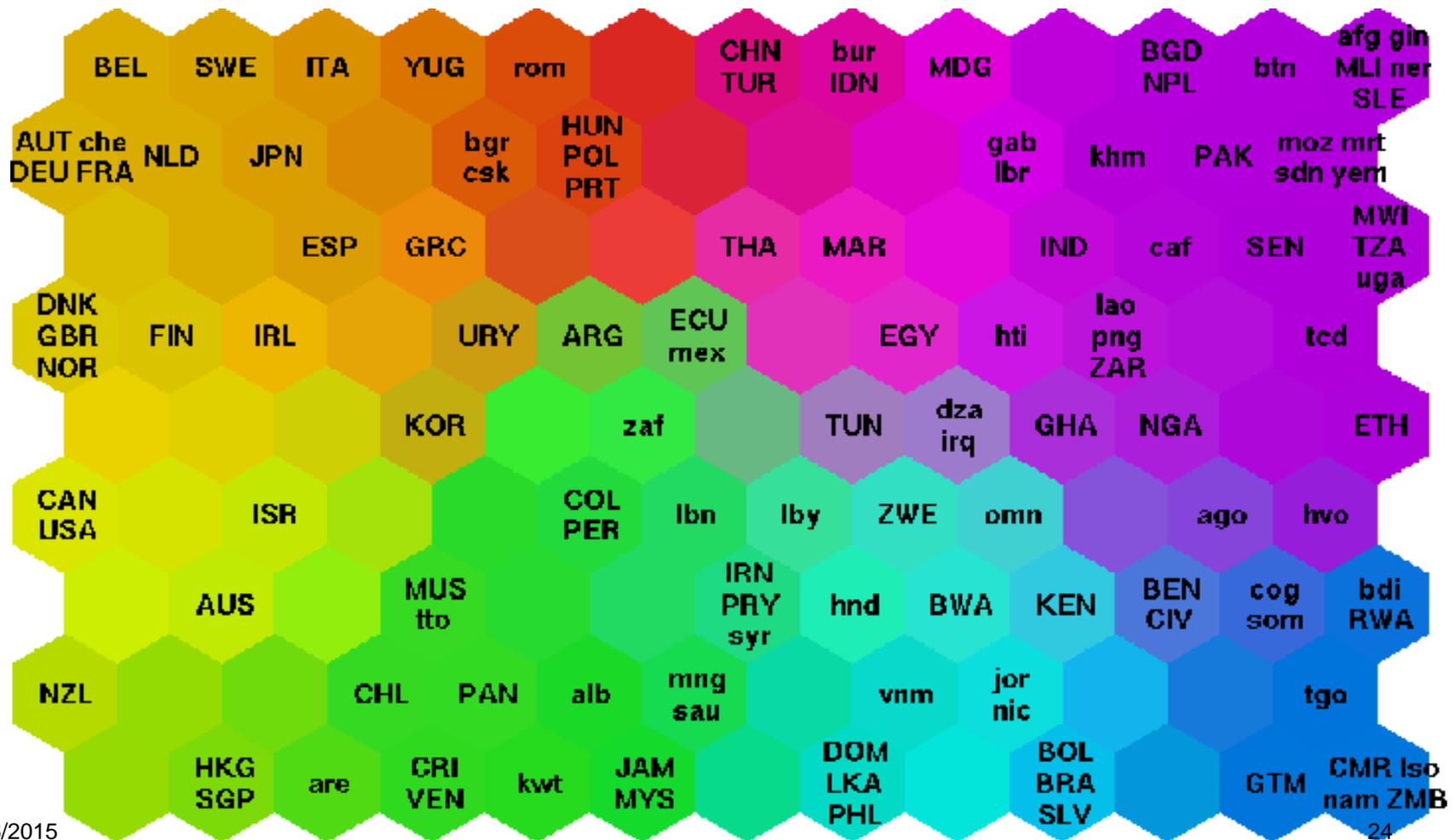
Aplikácie

- Kódovanie reči pre fonetický typewriter
- Kódovanie geometrických vzorov

SOM a zhlukovanie k -priemerov

- Kompetitívne samoučenie je možné chápať ako on-line verziu k -priemerov.
- SOM navyše pridáva mriežku a interakcie len v okolí.
- Metóda k -priemerov sa hodí pre zhlukovanie. SOM sa predovšetkým hodí na učenie variety menšej dimenzie než je dimenzia dát (napr. 2D alebo 3D).

Aplikácia: miera chudoby vo svete



Teoretická analýza algoritmu SOM

- Konverguje tento algoritmus vždy?
- Koľko iterácií je potrebných?
- Nevystupuje tu problém lokálnych miním?
- Ako voliť parametre?

Siete typu counterpropagation (CPN)

□ **Základná idea CPN**

- **Ciel'**: rýchla a hrubá aproximácia vektorového mapovania $y = \phi(x)$

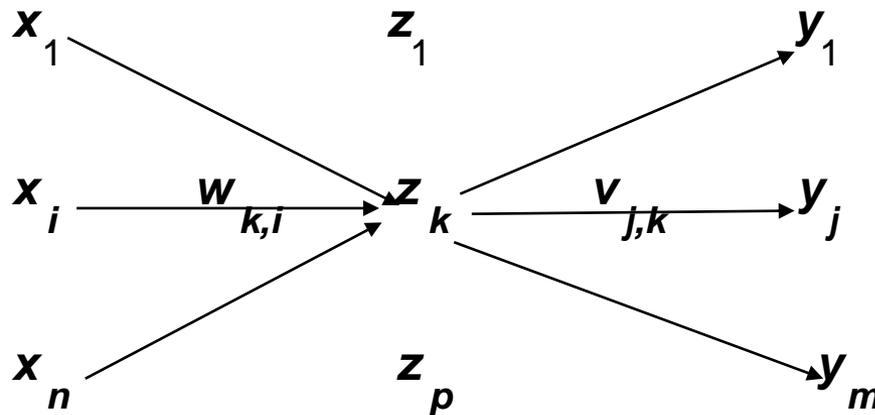
- Nemapovať žiadne dané x na $\phi(x)$ s vopred danou presnosťou,
- Vstupné vektory x sú rozdelené do klastrov/tried.
- Každý klaster pre x má jeden výstup y , ktorý je (dúfame) priemerom $\phi(x)$ pre všetky x v tejto triede.

Siete typu counterpropagation (CPN)

- Hecht – Nielsen, 1986
- CPN pracuje ako vyhľadávacia tabuľka, ktorá k danému vstupu nájde najbližšieho reprezentanta a odpovie výstupnou hodnotou, ktorá je s týmto reprezentantom spojená

CPN

- **Architektúra:**
- Jednoduchý prípad: **DOPREDNÁ CPN,**



Od vstupu ku
skrytej (triede)

Od skrytej (triedy) k
výstupu

CPN

- CPN pozostáva z troch vrstiev
- Vstupnú vrstvu tvorí n vstupných neurónov, ktoré distribuujú do ďalšej vrstvy vstupné signály x_1, x_2, \dots, x_n
- Druhá vrstva je tvorená p samoorganizačnými jednotkami, ktorej váhy označíme \mathbf{w}_j a výstupy y_j
- Tretiu vrstvu tvorí m Grossbergových jednotiek instar s výstupnými váhami \mathbf{v}_r ($r=1, \dots, m$). Výstupy jednotiek y_r tvoria výstup celej siete.

Učenie CPN v dvoch fázach:

- Trénovacia vzorka (\mathbf{x}, \mathbf{d}) kde $d = \phi(x)$ je predpokladané presné mapovanie.
- **Fáza 1:** Váhy w_k vstupujúce do skrytého neurónu z_k sú trénované použitím **súťažného učenia** (*competitive learning*), aby sa stali reprezentatívnymi vektormi pre klaster vstupných vektorov x : (použiť len x zo vstupu (x, d))
 1. Pre vybrané \mathbf{x} , dopredným šírením určiť víťaza z_{k^*}
 2. $w_{k^*,i}(new) = w_{k^*,i}(old) + \eta (x_i - w_{k^*,i}(old))$
 3. Upraviť η , opakovať kroky 1 a 2 do splnenia podmienky zastavenia

η je učiaci pomer

Učenie CPN v dvoch fázach:

- **Fáza 2:** váhy v_k idúce od skrytých neurónov z_k na výstup sú trénované pomocou **pravidla delta** (*delta rule*), aby sa stali priemerným výstupom $\phi(x)$ kde x je vstupný vektor, ktorý určí z_{k^*} za víťaza (treba použiť oba x a d).

1. Pre vybrané x , dopredným šírením určiť z_{k^*}
2. $w_{k^*,i}(new) = w_{k^*,i}(old) + \eta (x_i - w_{k^*,i}(old))$ (voliteľné)
3. $v_{j,k^*}(new) = v_{j,k^*}(old) + \eta(d_j - v_{j,k^*}(old))$
4. Opakovať kroky 1-3 pokiaľ nie je splnená podmienka zastavenia

CPN – štatistické vlastnosti

- Jednotky v druhej vrstve majú rovnakú pravdepodobnosť víťazstva za predpokladu, že vyberáme vstupy náhodne s rozložením zodpovedajúcim tréningovej množine.
- Váhy výstupných jednotiek sú adaptované tak, aby aproximovali priemernú výstupnú hodnotu patriacu tým vstupom, ktoré aktivovali odpovedajúce jednotky v druhej vrstve.

Príklad

- $n = 3, m = 2, p = 4$
- $((1, 2, 4), (1, 1))$
- $((-1, -2, 2), (-1, -2))$
- $((3, 3, 3), (1, 2))$

- $W1 = (2, 1, 2), W2 = (3, 4, 2)$
- $W3 = (-1, -2, 4), W4 = (3, 2, 1)$
- $V1 = (1, 2), V2 = (1, 3)$
- $V3 = (2, 4), V4 = (-3, 1)$