Neuronové sítě

Doc. RNDr. Iveta Mrázová, CSc.

Katedra teoretické informatiky
Matematicko-fyzikální fakulta
Univerzity Karlovy v Praze

Neuronové sítě

– Učení bez učitele –

Doc. RNDr. Iveta Mrázová, CSc.

Katedra teoretické informatiky
Matematicko-fyzikální fakulta
Univerzity Karlovy v Praze

Učení bez učitele

- **Učení bez učitele:**
  - Samoorganizace a shlukování
- **Motivace:**
  - Síť sama rozhodne, která odezva je pro daný vzor nejlepší a podle toho nastaví své váhy
- **Problém:**
  - Určit počet a rozložení shluků v příznakovém prostoru

Učení bez učitele (2)

NEVĚDĚT, ŽE JE NA NAŠÍ STRANĚ PRAVDA, TAK BYCH SE DOCELA BÁL.

Učení bez učitele (3)

DÍVČÍ VÁLKA VLASTNĚ NIKDY NEBYLA. ROZMĚLNILA SE IHNED V DROBNÉ ŠARVÁTKY.

Učení bez učitele (4)

Kompetiční učení:

- Boj o „právo reprezentovat předložený vzor“
- „Potlačování soupeřů“ → **INHIBICE**
- Pravidlo „**vítěz bere vše**“ (WTA – Winner_takes_all)

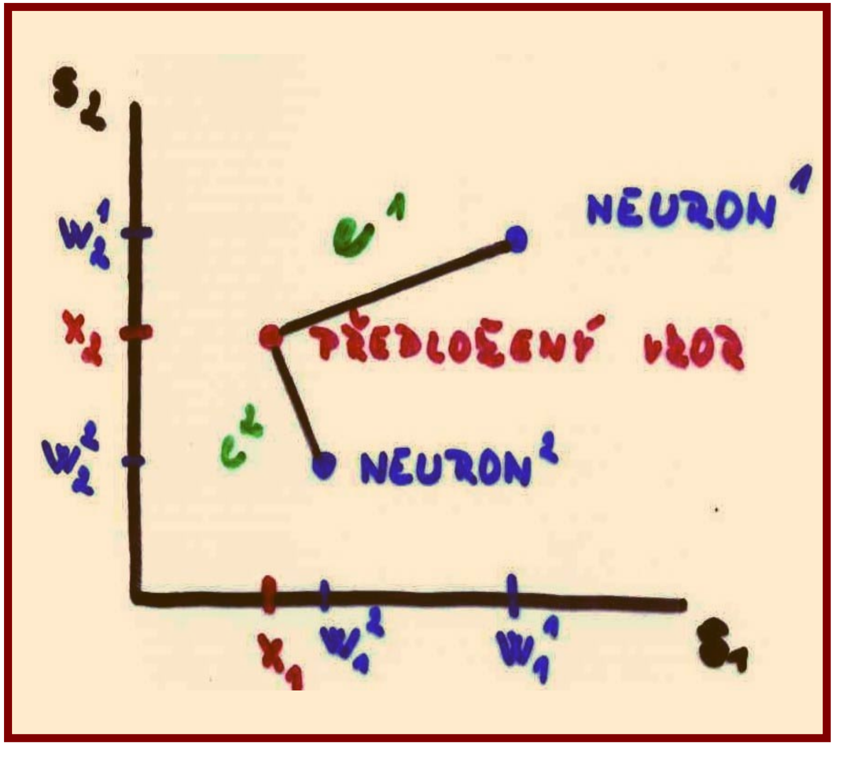
Posilované učení (reinforcement):

- Důraz na co nejlepší reprodukci vstupů

Kompetiční učení bez učitele

- ***n*** – rozměrný vstupní vzor je zpracováván pomocí takového počtu neuronů, který odpovídá (předpokládanému) počtu shluků
- Neurony v tomto případě počítají (Euklidovskou) **vzdálenost mezi předloženým vzorem a** svým **váhovým vektorem**

Kompetiční učení bez učitele (2)



- V kompetici „vítězí“ neuron, který ke nejblíže předloženému vzoru
- Vítězný neuron bude nejaktivnější a bude potlačovat – **inhibovat** – aktivitu ostatních neuronů

Kompetiční učení bez učitele (2a)

- Inhibice pomocí „laterálních spojů“

  ==> **laterální inhibice**

- Pro rozhodnutí, zda bude neuron aktivní nebo ne, je nutná globální informace o stavu všech neuronů v síti
- Aktivita neuronu signalizuje příslušnost předloženého vstupu ke shluku vektorů reprezentovaných tímto neuronem

Kompetiční učení bez učitele (3)

- Vítězný neuron zadaptuje své váhy směrem k předloženému vzoru:

$$\Delta \vec{w} = \alpha \cdot \left(\vec{x} - \vec{w} \right)$$

plasticita sítě (během učení pomalu klesá)

Cíl:

- Umístit neurony do středu shluků vzorů
- Zachovat již vytvořenou strukturu sítě

Kompetiční učení bez učitele (4)

- **Urychlení procesu učení:**
  - Vhodná inicializace vah
  - Např. podle náhodně vybraných vzorů
- **Problémy:**
  - Mrtvé (nevyužité) neurony
    - Mřížka v Kohonenově vrstvě
    - Topologické okolí neuronu
    - Řízená kompetice a mechanismus svědomí

Kompetiční učení bez učitele (5)

- Během učení by se měly váhy jednotlivých neuronů nastavit tak, aby odpovídaly „těžišti příslušného shluku“
- Energetická funkce množiny ***n***– rozměrných normovaných vstupních vzorů $X = \{\vec{x}_1, \ldots, \vec{x}_m\}; (n \geq 2)$ je pro ***1*** neuron s váhovým vektorem $\vec{w}$ dána pomocí:

$$E_X(\vec{w}) = \sum_{i=1}^{m} \left\| \vec{x}_i - \vec{w} \right\|^2 ; \quad \vec{w} \in R^n$$

Kompetiční učení bez učitele (6)

==> lze ukázat, že v optimálním případě je vektor vah umístěn v těžišti shluku vstupních vzorů

$$E_X(\vec{w}) = \sum_{i=1}^{m} \left\|\vec{x}_i - \vec{w}\right\|^2 = \sum_{i=1}^{m}\sum_{j=1}^{n} \left(x_{ij} - w_j\right)^2 =$$

$$= \sum_{i=1}^{m}\sum_{j=1}^{n} \left(x_{ij}^2 - 2x_{ij}w_j + w_j^2 \right) =$$

$$= m\left(\sum_{j=1}^{n} w_j^2 - \frac{2}{m}\sum_{j=1}^{n} w_j \left(\sum_{i=1}^{m} x_{ij}\right)\right) + \sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}^2 =$$

Kompetiční učení bez učitele (7)

$$E_X(\vec{w}) = m\sum_{j=1}^{n}\left(w_j^2 - \frac{2}{m}w_j\left(\sum_{i=1}^{m}x_{ij}\right) + \frac{1}{m^2}\left(\sum_{i=1}^{m}x_{ij}\right)\left(\sum_{i=1}^{m}x_{ij}\right)\right) -$$

$$\underbrace{-\frac{1}{m}\sum_{j=1}^{n}\left(\sum_{i=1}^{m}x_{ij}\right)\left(\sum_{i=1}^{m}x_{ij}\right) + \sum_{i=1}^{m}\sum_{j=1}^{n}x_{ij}^2}_{=\,K} =$$

$$= m\left(\sum_{j=1}^{n}\left(w_j - \frac{1}{m}\sum_{i=1}^{m}x_{ij}\right)^2\right) + K =$$

$$= m\left\|\vec{w} - \vec{x}^*\right\|^2 + K$$

Kompetiční učení bez učitele (8)

→ vektor $\vec{x}^*$ představuje centroid shluku $\left\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_m\right\}$ a $\boldsymbol{K}$ je konstanta

→ energetická funkce má globální minimum v $\vec{x}^*$

Kompetiční učení bez učitele (9)

Klastrovací metody pro empirická vícerozměrná data:

- dva základní přístupy:
  - ***k*** nejbližších sousedů
  - algoritmus ***k*** středů
- ***k*** **nejbližších sousedů** (učení s učitelem)
  - Vzory z trénovací množiny jsou uloženy a klasifikovány do jedné z ***l*** různých tříd
  - Neznámý vstupní vektor je zařazen do té třídy, ke které patří většina z ***k*** nejbližších vektorů z uložené množiny

Kompetiční učení bez učitele (10)

- **algoritmus *k* středů** (*k*-means clustering)
  - Učení bez učitele
  - Vstupní vektory jsou klasifikovány do ***k*** různých shluků (na začátku učení obsahuje každý shluk právě ***1*** vektor)
  - Nový vektor $\vec{X}$ je zařazen k tomu shluku ***k***, jehož centroid $\vec{C}_k$ je nejblíže tomuto vzoru

Kompetiční učení bez učitele (11)

- **algoritmus *k* středů** (pokračování)
  - Centroid $\vec{c}_k$ je pak aktualizován pomocí:

$$\vec{c}_k(new) = \vec{c}_k(old) + \frac{1}{n_k}\left(\vec{x} - \vec{c}_k(old) \right)$$

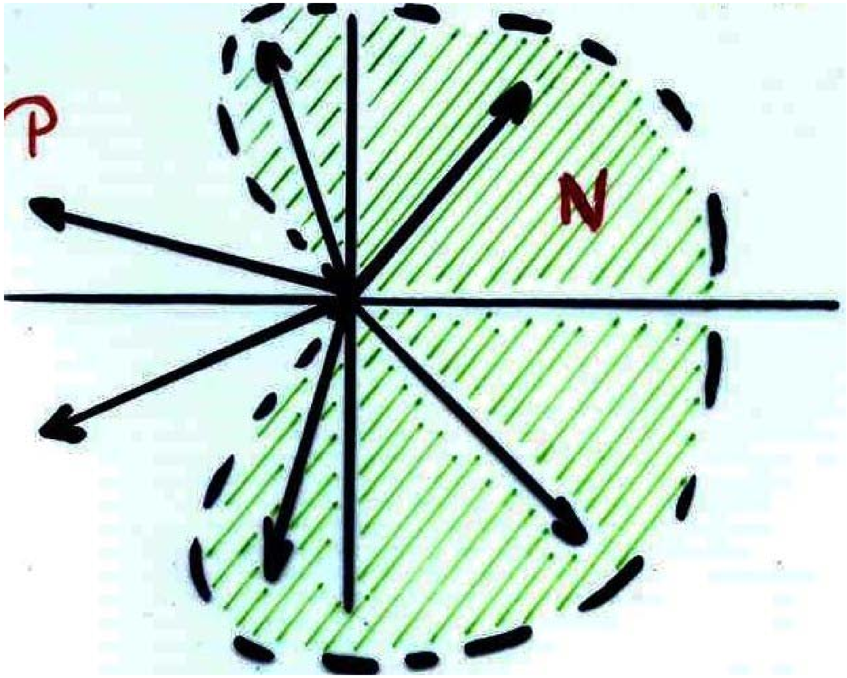
  $\boldsymbol{n_k}$ …. Počet vektorů již přiřazených shluku ***k***

  - Tato procedura se iterativně opakuje pro celou množinu dat (jejich strukturu pak vystihují „váhové vektory“ $\vec{c}_i$; ***i* = 1, …, *k***)

→ **Vektorová kvantizace**

Problém shlukování

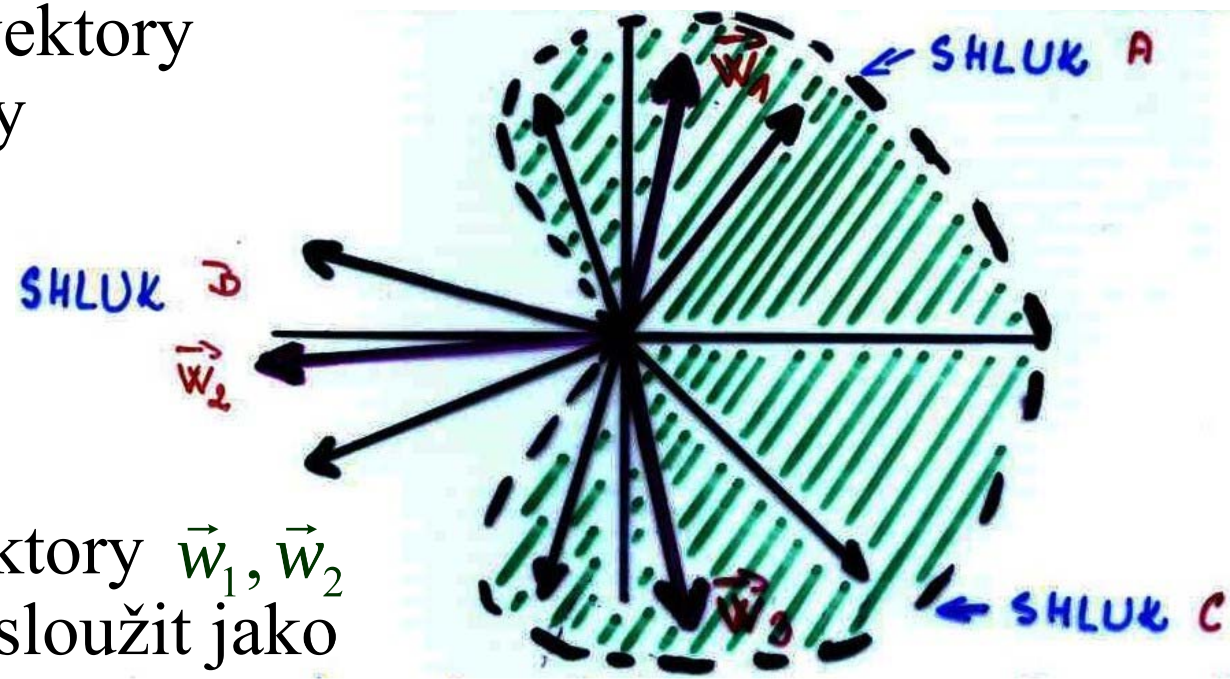
- Dvě množiny vektorů:
  - ***P*** a ***N***
- Obtížná „separace“ shluků pomocí jednoduchého perceptronu tak, aby:



$$\vec{w} \cdot \vec{p} \geq 0 \quad \forall \vec{p} \in P \quad \wedge \quad \vec{w} \cdot \vec{n} < 0 \quad \forall \vec{n} \in N$$

Problém shlukování (2)

- Tři váhové vektory pro tři shluky



- Tři různé vektory $\vec{w}_1, \vec{w}_2$ a $\vec{w}_3$ mohou sloužit jako „reprezentanti" v jednotlivých shlucích ***A***, ***B*** a ***C***

Problém shlukování (3)

- Každý z těchto tří vektorů je „relativně blízko“ ke každému vektoru z příslušného shluku
- Každý váhový vektor odpovídá jednomu bezprahovému neuronu, který je aktivní pouze v případě, že je vstupní vektor dostatečně blízko tomuto váhovému vektoru

==> Jak určit počet a rozložení shluků?

Kompetiční učení - algoritmus

- Nechť $X = \{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_l\}$ je množina normovaných vstupních vektorů v $\boldsymbol{n}$ – rozměrném prostoru, které chceme zařadit do $\boldsymbol{k}$ různých shluků
- Neuronová síť se skládá z $\boldsymbol{k}$ neuronů, z nichž každý má $\boldsymbol{n}$ vstupů a nulový práh

Inicializace:

- Normované váhové vektory $\vec{w}_1, \ldots, \vec{w}_k$ jsou generovány náhodně

Kompetiční učení – algoritmus (2)

Test:

- Náhodně vyber vektor $\vec{x}_j \in X$
- Spočítej $\vec{w}_i \cdot \vec{x}_j$ pro $\boldsymbol{i = 1, \ldots, k}$
- Vyber $\vec{w}_m$ takový, že $\vec{w}_m \cdot \vec{x}_j \geq \vec{w}_i \cdot \vec{x}_j \quad (\forall i = 1, \ldots, k)$
- Proveď Aktualizaci

Aktualizace:

- Nahraď $\vec{w}_m(new)$ vektorem $\vec{w}_m(old) + \vec{x}_j$ a znormuj
- Pokračuj v Testu

Kompetiční učení – algoritmus (3)

- Po provedení předem určeného počtu kroků lze algoritmus ukončit
- Váhové vektory ***k*** neuronů jsou „přitahovány“ směrem k centru jednotlivých shluků ve vstupním prostoru
- Algoritmus je založen na principu **„vítěz bere vše“** (~ WINNER-TAKES-ALL)

Kompetiční učení – algoritmus (4)

- Použití normovaných vektorů zabraňuje tomu, aby byl některý z váhových vektorů příliš velký, takže by „vyhrával“ kompetici příliš často
  - Ostatní neurony by se pak nikdy neaktualizovaly a zůstaly by nevyužité $\rightarrow$ **„mrtvé neurony“**
- Protože jsou vstupní i váhové vektory normované, je skalární součin $\vec{w}_i \cdot \vec{x}_j$ váhového a vstupního vektoru roven kosinu úhlu mezi těmito dvěma vektory

Kompetiční učení – algoritmus (5)

- Algoritmus učení zaručuje, že bude aktualizován ten vektor $\vec{w}_m$, který leží nejblíž k předloženému vstupnímu vzoru
- Při aktualizaci je váhový vektor $\vec{w}_m$ otočen směrem k $\vec{x}_j$

Různá pravidla učení:

- **Aktualizace pomocí parametru učení**

$$\Delta \vec{w}_m = \eta \, \vec{x}_j \qquad ; \eta \in (0,1) \text{ pomalu klesá v čase}$$

$\rightarrow$ **plasticita sítě**

Kompetiční učení – algoritmus (6)

Různá pravidla učení (pokračování):

- **Diferenční aktualizace**

$$\Delta \vec{w}_m = \eta \left(\vec{x}_j - \vec{w}_m \right)$$

  - „oprava“ odpovídá rozdílu mezi oběma vektory
- **Dávková aktualizace**
  - „korekce“ vah pro jednotlivé vzory se kumuluje
  - Váhy se změní po několika iteracích nejednou

    → stabilnější proces učení

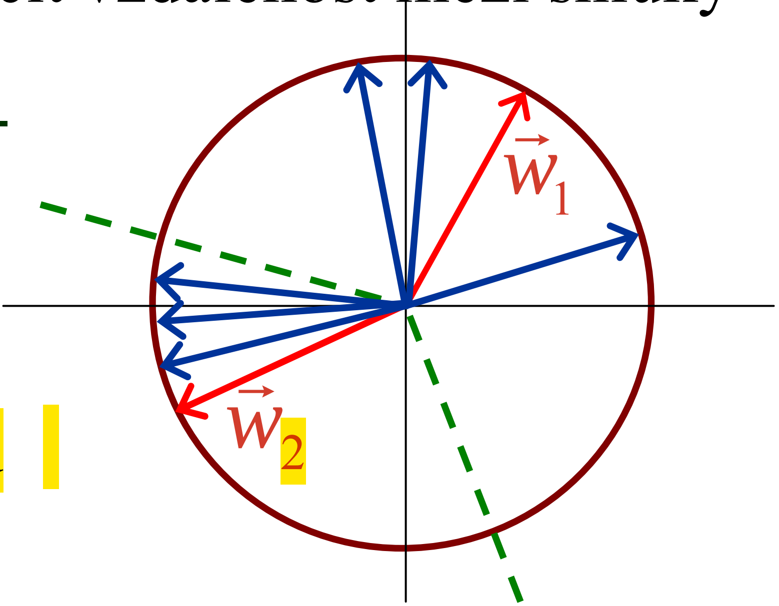
Kompetiční učení – algoritmus (7)

Stabilita řešení

- Nutnost vhodné míry pro „dobré shlukování“

  → **nejjednodušší řešení:** určit vzdálenost mezi shluky

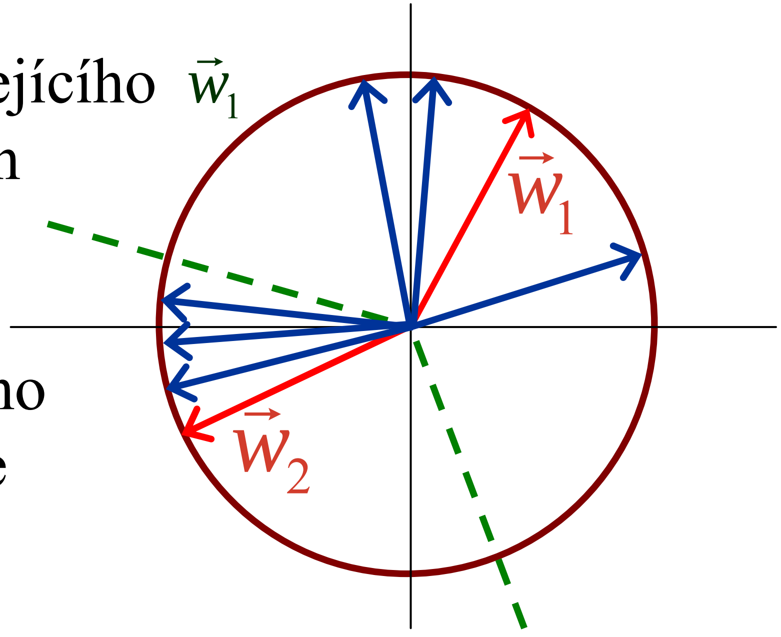
- Dva shluky vektorů a dva vektory „reprezentativních vah“:
  - Oba vektory „reprezentantů“ leží blízko vektorů, které leží v jejich odpovídajícím shluku



Kompetiční učení – algoritmus (8)

Stabilita řešení (pokračování):

- $\vec{w}_1$ leží uvnitř „svazku“ definovaného vektory svého shluku
- $\vec{w}_2$ leží vně „svazku“ příslušejícího $\vec{w}_1$
- Vektor $\vec{w}_1$ v dalších iteracích už „neopustí“ svůj „svazek“
- Vektor $\vec{w}_2$ se v některé z dalších iterací dostane do svého svazku a dál v něm už zůstane



→ takové řešení bude stabilní

Kompetiční učení – algoritmus (9)

Řešení ve stabilním rovnovážném stavu:

- **Intuitivní představa:**
  - **stabilní rovnovážný stav vyžaduje jasně ohraničené shluky**
- Pokud se jednotlivé shluky navzájem překrývají nebo jsou příliš rozsáhlé, nemusí být stabilní řešení nalezeno

==> **nestabilní rovnovážný stav**

Stabilita řešení - analýza

Definice:

Nechť $\boldsymbol{P}$ je množina $\{\vec{p}_1, \ldots, \vec{p}_m\}$ $\boldsymbol{n}$ – rozměrných ($\boldsymbol{n} \geq 2$) vektorů ležících ve stejném poloprostoru (~ formální omezení velikosti shluku).

Svazek $\boldsymbol{K}$ definovaný pomocí $\boldsymbol{P}$ je množina všech vektorů $\vec{x}$ tvaru $\vec{x} = \alpha_1 \vec{p}_1 + \ldots + \alpha_m \vec{p}_m$, kde $\boldsymbol{\alpha_1, \ldots, \alpha_m}$ jsou kladná reálná čísla.

- Svazek shluku obsahuje všechny vektory „uvnitř" shluku
- Průměr svazku definovaného normovanými vektory odpovídá největšímu možnému úhlu mezi vektory ve shluku

Stabilita řešení – analýza (2)

Definice:

(Úhlový) průměr $\boldsymbol{\varphi}$ svazku $\boldsymbol{K}$ definovaného normovanými vektory $\{\vec{p}_1, \ldots, \vec{p}_m\}$ odpovídá:

$$\varphi = \sup \left\{ \arccos\left(\vec{a} \cdot \vec{b}\right) \middle| \forall\, \vec{a}, \vec{b} \in K ; \|\vec{a}\| = \left\|\vec{b}\right\| = 1 \right\}$$

kde $0 \leq \arccos\left(\vec{a} \cdot \vec{b}\right) \leq \pi$

- Postačující podmínkou pro stabilní řešení je, aby byl úhlový průměr svazků menší než jejich vzájemná vzdálenost

Stabilita řešení – analýza (3)

Definice:

Nechť $P = \{\vec{p}_1, \dots, \vec{p}_m\}$ a $N = \{\vec{n}_1, \dots, \vec{n}_k\}$ jsou dvě neprázdné množiny normovaných vektorů v $\boldsymbol{n}$–rozměrném prostoru ($\boldsymbol{n \geq 2}$), které definují svazky $\boldsymbol{K_P}$ a $\boldsymbol{K_N}$

- Jestlliže je průnik těchto dvou svazků prázdný, je (úhlová) vzdálenost mezi $\boldsymbol{K_N}$ a $\boldsymbol{K_P}$ dána pomocí:

  $$\psi = \inf\left\{\arccos\left(\vec{p}\cdot\vec{n}\right);\ \vec{p}\in K_P,\ \vec{n}\in K_N \text{ a } \|\vec{p}\| = \|\vec{n}\| = 1\right\}$$

  kde $0 \leq \arccos\left(\vec{p}\cdot\vec{n}\right) \leq \pi$

- Jestliže se $\boldsymbol{K_P}$ a $\boldsymbol{K_N}$ prolínají, je $\boldsymbol{\psi_{P,N} = 0}$

Stabilita řešení – analýza (4)

- Jestliže je úhlová vzdálenost mezi shluky větší než úhlový průměr svazků, potom existuje stabilní řešení
  - Váhové vektory budou ležet ve svazku jim příslušejícího shluku
  - Jestliže se váhový vektor už „dostal“ dovnitř příslušného svazku, zůstane tam i nadále

Kontrola kvality shlukování:

- Výhodnější je menší počet „kompaktnějších“ shluků
- Ohodnocovací funkce pro penalizaci příliš vysokého počtu shluků

PCA – Principal Component Analysis

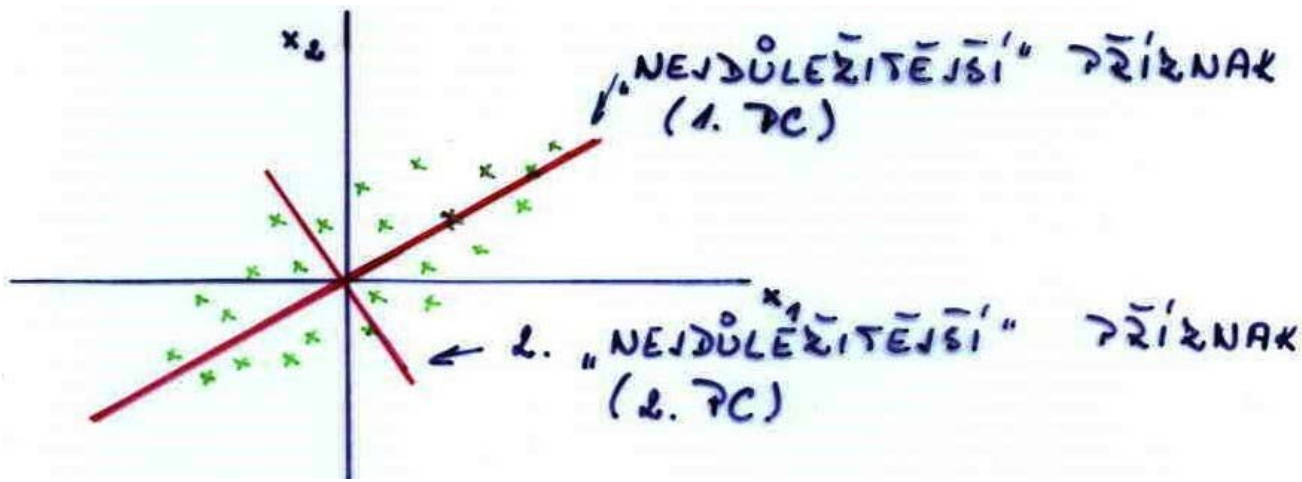
- redukce dimenzionality vstupních dat
  - → použít méně příznaků bez ztráty podstatné informace
  - → výběr nejdůležitějších příznaků
- dána množina ***m n*** – rozměrných vektorů:

$$\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_m\}$$

- prvním „nejdůležitějším“ příznakem pro tuto množinu vektorů je vektor $\vec{w}$, který maximalizuje výraz

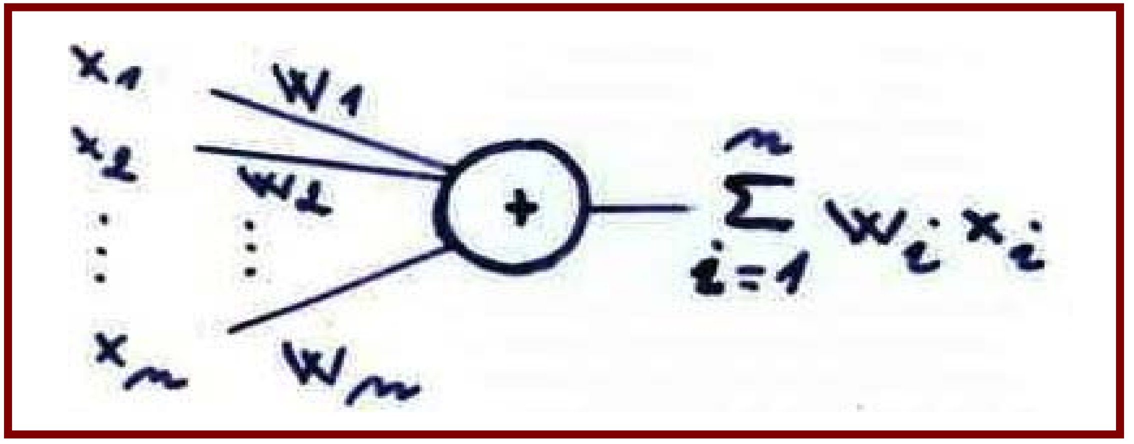
$$\frac{1}{m} \sum_{i=1}^{m} \| \vec{w} \cdot \vec{x}_i \|^2$$

PCA – Principal Component Analysis (2)



- Rozložení vstupních dat:
  - 1. PC: největší rozptyl
  - 2. PC: kolmá k 1. PC a největší rozptyl
    (~ odečíst od $\vec{x}$ jeho ortogonální projekci na 1. PC)

PCA – Principal Component Analysis (3)



- **Použitý model:**
  - Lineární asociátor
    – jako výsledek počítá pouze vážený součet vstupů
  - Posilované učení – Ojův algoritmus učení

Výpočet nejdůležitějších příznaků pomocí umělých neuronových sítí

Ojův algoritmus učení (E. Oja, 1982)
(pro výpočet prvního nejdůležitějšího příznaku)

Předpoklad:

- Těžiště vstupních dat je vycentrováno v počátku

Start:

- Nechť X je množina n – rozměrných vektorů
- Vektor $\vec{w}$ je inicializován náhodně ($\vec{w} \neq 0$)
- Je zvolen parametr učení γ ($0 < \gamma \leq 1$)

Výpočet nejdůležitějších příznaků pomocí umělých neuronových sítí (2)

Ojův algoritmus učení (pokračování):

Adaptace:

- z množiny X je náhodně vybrán vektor $\vec{x}$
- Spočítá se skalární součin $\Phi = \vec{x} \cdot \vec{w}$
- Nový váhový vektor je $\vec{w} + \gamma \Phi \left(\vec{x} - \Phi \vec{w} \right)$
- Zmenši γ a přejdi k „Adaptaci“

Podmínka pro ukončení adaptace

- např. počet provedených iterací

Výpočet nejdůležitějších příznaků pomocí umělých neuronových sítí (3)

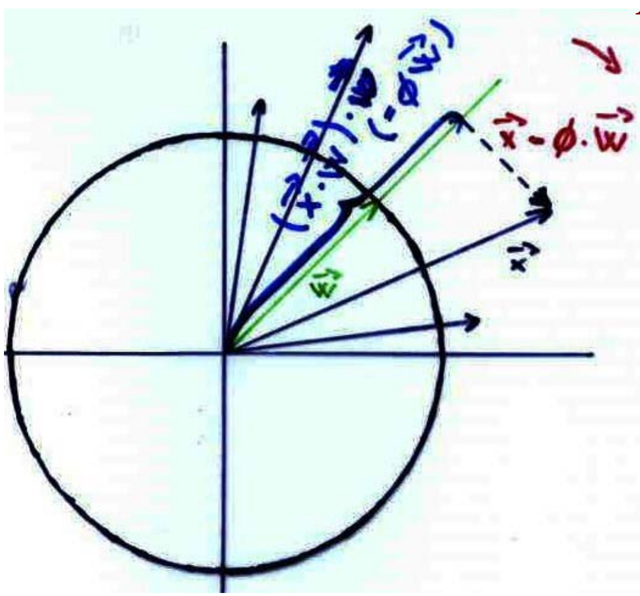
- **Parametr učení γ**
  - Parametr učení je třeba zvolit dostatečně malý, aby byla $\vec{w}$ zaručena adekvátní adaptace vah (omezení velkých oscilací)
- **„Automatická normalizace“ váhového vektoru**
  - Odpadá nutnost znát globální informaci o všech vzorech
  - Postačí lokální informace o aktualizované váze, vstupu a skalárním součinu asociátoru
- **První nejdůležitější příznak odpovídá směru nejdelšího vlastního vektoru korelační matice uvažovaných vstupních vektorů**

Konvergence Ojova algoritmu učení

Pokud existuje jednoznačné řešení, bude Ojův algoritmus učení konvergovat:

Idea důkazu:

Adaptace $\vec{w}$ směrem k $\vec{x}$



-) začne-li Ojův algoritmus učení s váhovým vektorem uvnitř svazku, bude v něm oscilovat, ale neopustí ho

-) pro $\|\vec{w}\| = 1$ odpovídá skalární součin $\Phi = \vec{x} \cdot \vec{w}$ délce projekce $\vec{x}$ na $\vec{w}$

Konvergence Ojova algoritmu učení (2)

Idea důkazu (pokračování):

-) vektor $\vec{x} - \Phi \vec{w}$ je kolmý na $\vec{w}$
-) iterace Ojova algoritmu učení přitahují $\vec{w}$ k vektorům ze shluku X
-) pokud zůstane délka $\vec{w}$ rovna 1 (anebo blízko 1), umístí se $\vec{w}$ v procesu učení v centru shluku
-) dále je ještě třeba ukázat, že vektor $\vec{w}$ je Ojovým algoritmem automaticky normován:

  Idea: a) délka vektoru $\vec{w}$ je větší než 1
  b) délka vektoru $\vec{w}$ je menší než 1

Konvergence Ojova algoritmu učení (3)

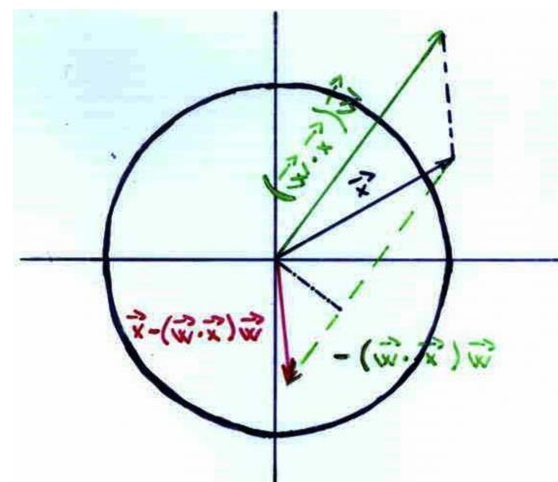
a) délka vektoru $\vec{w}$ je větší než **1**

- Délka vektoru $(\vec{x} \cdot \vec{w})\vec{w}$ je větší než délka ortogonální projekce $\vec{x}$ na $\vec{w}$
- Dále předpokládejme, že $\vec{x} \cdot \vec{w} > 0$, tj. že vektory $\vec{x}$ a $\vec{w}$ nejsou navzájem příliš vzdálené
- Vektor $\vec{x} - (\vec{x} \cdot \vec{w})\vec{w}$ má zápornou projekci na $\vec{w}$,

protože

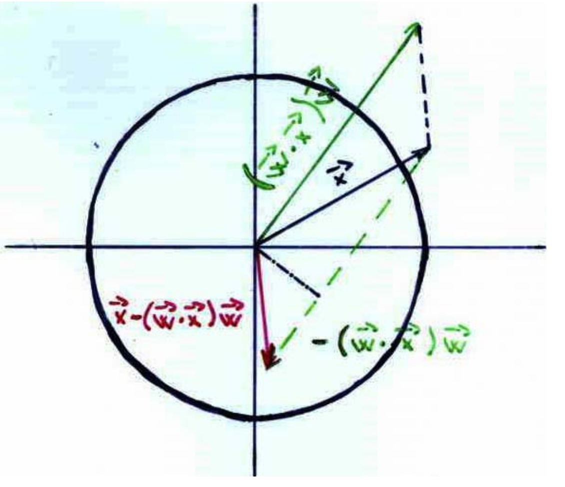
$$(\vec{x} - (\vec{x} \cdot \vec{w})\vec{w}) \cdot \vec{w} =$$

$$= \vec{x} \cdot \vec{w} - \|\vec{w}\|^2 \vec{x} \cdot \vec{w} < 0$$



Konvergence Ojova algoritmu učení (4)

- Výsledek po větším počtu iterací:
  - (vektor $\vec{x} - (\vec{x} \cdot \vec{w})\vec{w}$ má jednu složku kolmou na $\vec{w}$ a druhou složku, která směruje opačným směrem než $\vec{w}$)
  - $\vec{w}$ přejde do centra shluku vektorů (kolmá složka v průměru vymizí)
  - $\vec{w}$ se bude s rostoucím počtem iterací tohoto typu zmenšovat (!pozor na nebezpečí změny orientace $\vec{w}$ v jedné iteraci)



Konvergence Ojova algoritmu učení (5)

- **Vhodná volba parametru učení γ a normalizace vektorů z trénovací množiny před začátkem učení:**
  - jestliže má vektor $\vec{x}$ pozitivní skalární součin Φ s $\vec{w}$, měl by mít i nový váhový vektor pozitivní skalární součin s $\vec{x}$:
  - mělo by tedy platit: $\vec{x} \cdot \left(\vec{w} + \gamma\, \Phi\, \left(\vec{x} - \Phi\, \vec{w}\right)\right) > 0$

$$\Phi + \gamma\, \Phi\, \left\|\vec{x}\right\|^2 - \gamma\, \Phi\, \Phi^2 > 0$$

$$\underbrace{\Phi}_{>0}\ \underbrace{\left(1 + \gamma\left(\left\|\vec{x}\right\|^2 - \Phi^2\right)\right)}_{\Rightarrow\ \gamma\left(\left\|x^2\right\| - \Phi^2\right) > -1} > 0$$

  - Pro dostatečně malé kladné γ platí vždy

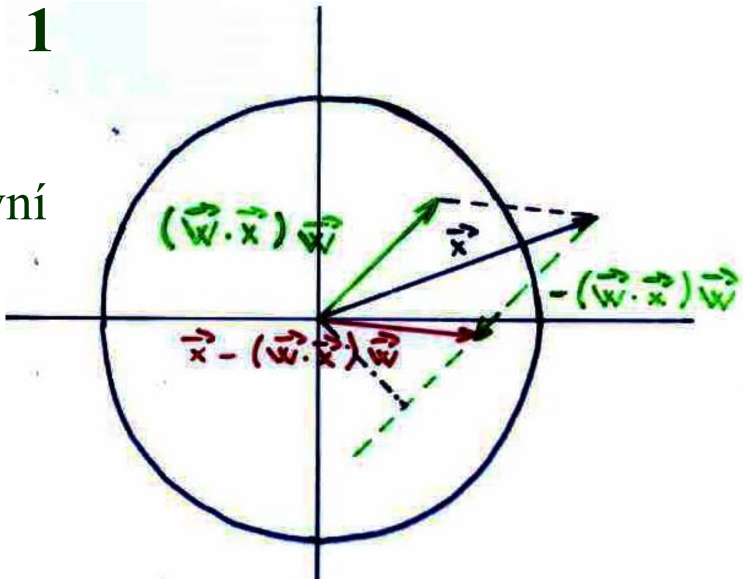
Konvergence Ojova algoritmu učení (6)

a) délka vektoru $\vec{w}$ je menší než **1**

(analogicky k a))

→ vektor $\vec{x} - (\vec{x} \cdot \vec{w})\vec{w}$ má pozitivní projekci na $\vec{w}$

→ zvětšování $\vec{w}$



- Kombinace a) a b) $\Rightarrow$ $\vec{w}$ se dostane do centra shluku a délka bude oscilovat kolem 1 (pro dostatečně malé γ)
- problémy:
  „řídké“ shluky

Ojův algoritmus učení: problémy a zobecnění

<u>Problémy:</u> -) „řídké“ shluky

-) příliš velké rozdíly v délce vstupních vektorů

Výpočet většího počtu nejdůležitejších příznaků: