Neuronové sítě

Doc. RNDr. Iveta Mrázová, CSc.

Katedra teoretické informatiky
Matematicko-fyzikální fakulta
Univerzity Karlovy v Praze

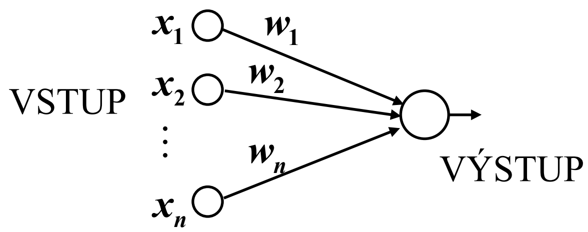
Neuronové sítě

– Perceptron a lineární separabilita –

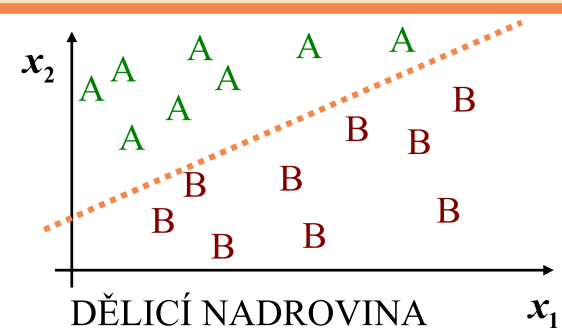
Doc. RNDr. Iveta Mrázová, CSc.

Katedra teoretické informatiky
Matematicko-fyzikální fakulta
Univerzity Karlovy v Praze

Formální neuron



$$y = f_h\left(\sum_{i=1}^{n} w_i\, x_i + \vartheta\right)$$



$$x_2 = -\frac{w_1}{w_2} x_1 - \frac{\vartheta}{w_2}$$

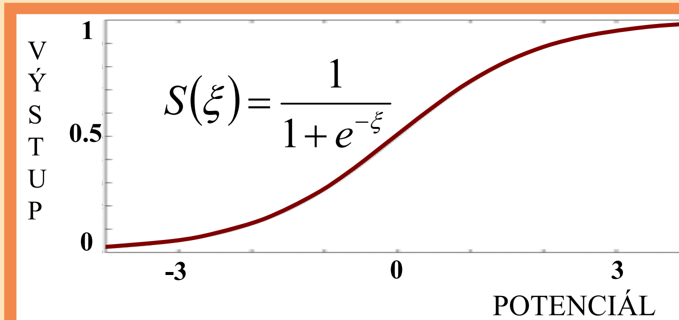
$$y = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} w_i x_i + \vartheta \geq 0: \ \text{TŘÍDA A} \\ 0 & \text{if } \sum_{i=1}^{n} w_i x_i + \vartheta < 0: \ \text{TŘÍDA B} \end{cases}$$

Typy přenosových funkcí

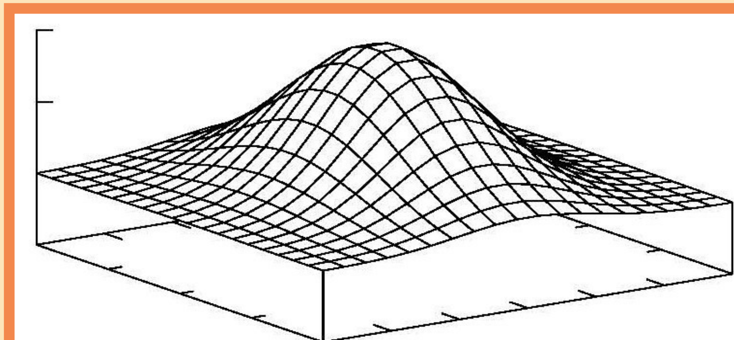
Skoková

$$y = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} w_i x_i + \vartheta \geq 0 : \ \text{TŘÍDA A} \\ 0 & \text{if } \sum_{i=1}^{n} w_i x_i + \vartheta < 0 : \ \text{TŘÍDA B} \end{cases}$$

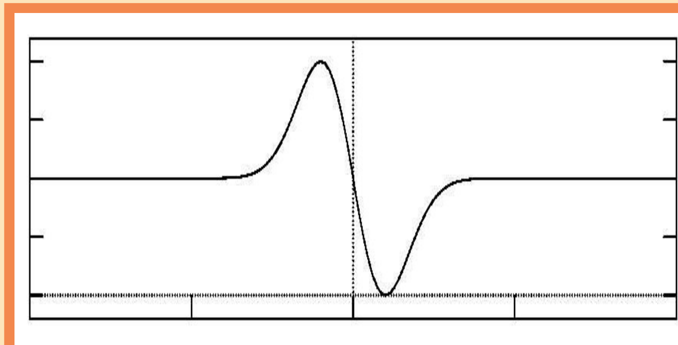
Sigmoidální



Radiální (RBF)



Waveletová



Definice formálního neuronu

Neuron s vahami $(w_1, \ldots, w_n) \in R^n$, prahem $\vartheta \in R$ a přenosovou funkcí $f : R^{n+1} \times R^n \longrightarrow R$ počítá pro libovolný vstup $\vec{z} \in R^n$ svůj výstup $y \in R$ jako hodnotu přenosové funkce v $\vec{z}$, $f[\vec{w}, \vartheta](\vec{z})$.

Nejčastěji se uvažuje tzv. sigmoidální přenosová funkce:

$$y = f[\vec{w}, \vartheta](\vec{z}) = f(\xi) = \frac{1}{1 + e^{-\xi}}$$

$\xi = \sum_{i=1}^{n} z_i w_i + \vartheta$ označuje tzv. potenciál neuronu, R množinu reálných čísel.

Definice stavů neuronu

Nechť $\vec{z}$ označuje vstup neuronu.

- Jestliže $f[\vec{w}, \vartheta](\vec{z}) = 1$, říkáme, že je neuron aktivní;
- Jestliže $f[\vec{w}, \vartheta](\vec{z}) = \frac{1}{2}$, říkáme, že je neuron tichý;

  Tato skutečnost znamená, že příslušný vstup leží v dělicí nadrovině určené tímto neuronem.
- Jestliže $f[\vec{w}, \vartheta](\vec{z}) = 0$, říkáme, že je neuron pasivní.

Učení a rozpoznávání

- **Učení:**
  - **S učitelem - trénovací množina** tvaru [vstup/požadovaný výstup]
  - **Bez učitele (samoorganizace)** - chybí požadovaný výstup
  - ⇒ **Cíl: nastavení (adaptace) synaptických vah**
    (např. minimalizací střední kvadratické odchylky)

  Cílová funkce: např. $$\sum_p \sum_j \left(y_{p,j} - d_{p,j} \right)^2$$

  y je skutečný a ***d*** je požadovaný výstup

- **Rozpoznávání:**
  - **nově předkládaných vstupních vzorů**

  => Cíl: získat odezvu (výstup) neuronové sítě

Definice trénovacích vzorů

Pro BP-síť B s n vstupními a m výstupními neurony:

- vstupní vzor označuje vstupní vektor $\vec{x} \in R^n$ zpracovávaný sítí.
- požadovaný výstup $\vec{d} = (d_1, \ldots, d_m)$ tvoří požadované výstupy neuronů výstupní vrstvy.
- Pro daný vstupní vzor představuje skutečný výstup B vektor $\vec{y} = (y_1, \ldots, y_m)$ tvořený skutečnými výstupy neuronů výstupní vrstvy.

Trénovací množina T je množina p uspořádaných dvojic tvaru vstupní vzor/požadovaný výstup:

$$T = \{[\vec{x}_1, \vec{d}_1], \ldots, [\vec{x}_p, \vec{d}_p]\}.$$

Perceptron a lineární separabilita (1)

D: **Jednoduchý perceptron** je výpočetní jednotka s prahem ϑ, která pro n reálných vstupů $x_1, x_2, \ldots, x_n$ a váhy $w_1, \ldots, w_n$ dává výstup 1, jestliže platí nerovnost

$$\sum_{i=1}^{n} w_i \, x_i \geq \vartheta$$

(tzn. pokud $\vec{w} \cdot \vec{x} \geq \vartheta$) a 0 jinak.

Pozn.: Obdobně pro tzv. **rozšířený váhový a vstupní vektor**:

$$\vec{w} = (w_1, w_2, \ldots, w_n, w_{n+1}) \quad ; \quad w_{n+1} = -\vartheta$$

$$\vec{x} = (x_1, x_2, \ldots, x_n, 1)$$

=> výstup 1, jestliže $\vec{w} \cdot \vec{x} \geq 0$

Perceptron a lineární separabilita (2)

Lineární separabilita:

D: Dvě množiny A a B se nazývají **lineárně separabilní** v n-rozměrném prostoru, pokud existuje $n+1$ reálných čísel $w_1, \dots, w_n, \vartheta$ takových, že každý bod $(x_1, x_2, \dots, x_n) \in A$ splňuje $\sum_{i=1}^{n} w_i x_i \geq \vartheta$ a každý bod $(x_1, x_2, \dots, x_n) \in B$ splňuje $\sum_{i=1}^{n} w_i x_i < \vartheta$

Perceptron a lineární separabilita (3)

Příklad:

- ***n=2*** => 14 z 16 možných Boolovských funkcí
  je „lineárně separabilních“
- ***n=3*** => 104 z 256 - ´´ -
- ***n=4*** => 1882 z 65536 - ´´ -

- Pro obecný případ zatím není znám výraz pro vyjádření odpovídajícího počtu lineárně separabilních funkcí v závislosti na ***n***

Perceptron a lineární separabilita (4)

Absolutní lineární separabilita:

D: Dvě množiny $\boldsymbol{A}$ a $\boldsymbol{B}$ se nazývají **absolutně lineárně separabilní** v $\boldsymbol{n}$-rozměrném prostoru, pokud existuje $\boldsymbol{n+1}$ reálných čísel $\boldsymbol{w_1, \ldots, w_n, \vartheta}$ takových, že každý bod $\left(x_1, x_2, \ldots, x_n\right) \in A$ splňuje $\sum_{i=1}^{n} w_i x_i > \vartheta$ a každý bod $\left(x_1, x_2, \ldots, x_n\right) \in B$ splňuje $\sum_{i=1}^{n} w_i x_i < \vartheta$

Perceptron a lineární separabilita (5)

V: Dvě konečné množiny bodů A a B, které jsou lineárně separabilní v n-rozměrném prostoru, jsou také absolutně lineárně separabilní.

Důkaz: Protože jsou množiny A a B lineárně separabilní, existují reálná čísla $w_1, \ldots, w_n, \vartheta$ taková, že platí

$\sum_{i=1}^{n} w_i x_i \geq \vartheta$ pro všechny body $(x_1, x_2, \ldots, x_n) \in A$

a $\sum_{i=1}^{n} w_i x_i < \vartheta$ pro všechny body $(x_1, x_2, \ldots, x_n) \in B$

Perceptron a lineární separabilita (6)

Dále nechť: $\varepsilon = \max \left\{ \sum_{i=1}^{n} w_i b_i - \vartheta ; (b_1, \dots, b_n) \in B \right\}$

a zřejmě $\varepsilon < \varepsilon/2 < 0$

Nechť $\vartheta' = \vartheta + \frac{\varepsilon}{2}$ (tedy: $\vartheta = \vartheta' - \frac{\varepsilon}{2}$)

=> Pro všechny body z ***A*** platí, že $\sum_{i=1}^{n} w_i a_i - \left(\vartheta' - \frac{1}{2}\varepsilon \right) \geq 0$

To znamená, že $\sum_{i=1}^{n} w_i a_i - \vartheta' \geq -\frac{1}{2}\varepsilon > 0$

$$\Rightarrow \sum_{i=1}^{n} w_i a_i > \vartheta' \qquad (\forall (a_1, \dots, a_n) \in A) \qquad (\,*\,)$$

Perceptron a lineární separabilita (6)

Podobně pro všechny body z ***B***

$$\sum_{i=1}^{n} w_i b_i - \vartheta = \sum_{i=1}^{n} w_i b_i - \left(\vartheta' - \frac{1}{2}\varepsilon \right) \le \varepsilon$$

a tedy $$\sum_{i=1}^{n} w_i b_i - \vartheta' \le \frac{1}{2}\varepsilon < 0 \qquad (**)$$

Z (*) a (**) vyplývá, že množiny A a B jsou absolutně lineárně separabilní.

QED

Dělicí nadrovina – pro rozšířený váhový, resp. příznakový prostor (1)

D: Otevřený (uzavřený) pozitivní poloprostor určený ***n*** – rozměrným váhovým vektorem $\vec{w}$ je množina všech bodů $\vec{x} \in R^n$, pro které $\vec{w} \cdot \vec{x} > 0$ $\left(\vec{w} \cdot \vec{x} \geq 0\right)$

Otevřený (uzavřený) negativní poloprostor určený ***n*** – rozměrným váhovým vektorem $\vec{w}$ je množina všech bodů $\vec{x} \in R^n$, pro které $\vec{w} \cdot \vec{x} < 0$ $\left(\vec{w} \cdot \vec{x} \leq 0\right)$

Dělicí nadrovina – pro rozšířený váhový, resp. příznakový prostor (2)

D: Dělicí nadrovina určená $\boldsymbol{n}$ – rozměrným váhovým vektorem $\vec{w}$ je množina všech bodů $\vec{x} \in R^n$, pro které $\vec{w} \cdot \vec{x} = 0$

Problém: Nalézt takové váhy, resp. práh, které by umožnily separaci (oddělení) dvou množin vzorů

=> např. **PERCEPTRONOVÝ ALGORITMUS UČENÍ**

Předpoklad:

- A … množina vstupních vektorů v $\boldsymbol{n}$ –rozměrném prostoru
- B … množina vstupních vektorů v $\boldsymbol{n}$ –rozměrném prostoru

Dělicí nadrovina – pro rozšířený váhový, resp. příznakový prostor (3)

<u>SEPARACE A a B:</u>

⇒ Perceptron by měl realizovat binární funkci $f_{\vec{w}}$ tak, aby $f_{\vec{w}}(\vec{x}) = 1 \quad \forall \vec{x} \in A$ a $f_{\vec{w}}(\vec{x}) = 0 \quad \forall \vec{x} \in B$

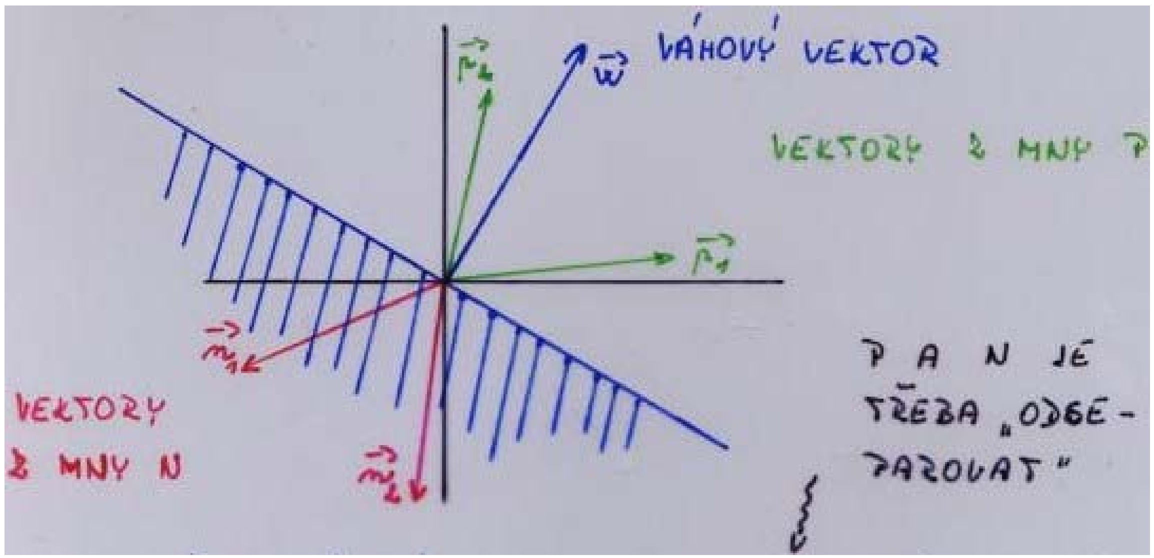
($f_{\vec{w}}$ závisí na vahách, resp. prahu)

- Chybová funkce odpovídá počtu chybně „zařazených“ vzorů: $E(\vec{w}) = \sum_{\vec{x} \in A} \left(1 - f_{\vec{w}}(\vec{x})\right) + \sum_{\vec{x} \in B} f_{\vec{w}}(\vec{x})$

<u>Cíl učení:</u> minimalizace $E(\vec{w})$ ve váhovém prostoru

(=> nejlépe $E(\vec{w}) = 0$)

Perceptronový algoritmus učení (1)



Hledáme váhový vektor $\vec{w}$ s pozitivním skalárním součinem pro všechny vektory reprezentované body v ***P*** a se záporným skalárním součinem pro všechny vektory reprezentované body v ***N***

Perceptronový algoritmus učení (2)

⇒ **OBECNĚ:** za předpokladu, že ***P*** a ***N*** jsou množiny ***n*** – rozměrných vektorů chceme nalézt takový váhový vektor $\vec{w}$, že:

$$\vec{w} \cdot \vec{x} > 0 \quad \forall \vec{x} \in P$$

$$\vec{w} \cdot \vec{x} < 0 \quad \forall \vec{x} \in N$$

- Perceptronový algoritmus učení začíná s náhodně zvoleným váhovým vektorem $\vec{w}_0$
- Pokud existuje vektor $\vec{x} \in P$ takový, že $\vec{w} \cdot \vec{x} < 0$, znamená to, že úhel mezi těmito dvěma vektory je větší než 90°
  → Váhový vektor je nutné zadaptovat (~ otočit) ve směru $\vec{x}$ (tak, aby se tento vektor dostal do „pozitivního“ poloprostoru definovaného $\vec{w}$

Perceptronový algoritmus učení (3)

→ Otočení ve směru $\vec{x}$ lze provést přičtením $\vec{x}$ k vektoru $\vec{w}$

- Pokud existuje vektor $\vec{x} \in N$ takový, že $\vec{w} \cdot \vec{x} > 0$, znamená to, že úhel mezi těmito dvěma vektory je menší než $90°$
  - → Váhový vektor je nutné zadaptovat (~ otočit) směrem od $\vec{x}$ (tak, aby se tento vektor dostal do „negativního" poloprostoru definovaného $\vec{w}$
  - → Otočení směrem od $\vec{x}$ lze provést odečtením $\vec{x}$ od vektoru $\vec{w}$
- vektory z $\boldsymbol{P}$ tedy otáčejí váhový vektor opačným směrem než vektory z $\boldsymbol{N}$
- Pokud existuje řešení, lze ho nalézt v konečném počtu kroků

Perceptronový algoritmus učení (4)

Krok 1: Inicializace vah malými náhodnými hodnotami $w_i(0)$

$w_i(0)$ … váha vstupu i v čase 0; $(1 \leq i \leq n+1)$

Krok 2: Předložení trénovacího vzoru ve tvaru

$(x_1, \ldots, x_{n+1})$ … vstupní vzor a

$d(t)$ ………….. požadovaný výstup (pro předložený vstup)

Krok 3: Výpočet skutečného výstupu (odezvy sítě)

$$y(t) = \operatorname{sgn}\left(\sum_{i=1}^{n+1} w_i(t)\, x_i(t)\right)$$

Krok 4: Adaptace vah podle:

$w_i(t+1) = w_i(t)$ výstup je správný

$w_i(t+1) = w_i(t) + x_i(t)$ výstup je 0 a měl být 1

$w_i(t+1) = w_i(t) - x_i(t)$ výstup je 1 a měl být 0

Krok 5: Pokud t nedosáhl požadované hodnoty, přejdi ke Kroku 2

Perceptronový algoritmus učení (5)

- **Heuristika pro počáteční nastavení vah:**
  Začít s průměrem „pozitivních“ vstupních vektorů minus průměr „negativních“ vektorů

- **Modifikace:** **parametr učení** η $(0 \leq \eta \leq 1)$
  (stupeň adaptivity vah ~ plasticita sítě)

  - Adaptace vah podle:

$w_i(t+1) = w_i(t)$	výstup je správný
$w_i(t+1) = w_i(t) + \eta\, x_i(t)$	výstup je 0 a měl být 1
$w_i(t+1) = w_i(t) - \eta\, x_i(t)$	výstup je 1 a měl být 0

Konvergence perceptronového algoritmu učení (Rosenblatt, 1959)

V: Nechť $\boldsymbol{P}$ a $\boldsymbol{N}$ jsou konečné a lineárně separabilní množiny. Potom provede perceptronový algoritmus učení **konečný počet aktualizací** váhového vektoru $\vec{w}_t$.

(Pokud se budou cyklicky testovat jeden po druhém vzory z $\boldsymbol{P}$ a $\boldsymbol{N}$, najde perceptronový algoritmus učení po provedení konečného počtu aktualizací váhový vektor, pomocí něhož lze navzájem separovat $\boldsymbol{P}$ a $\boldsymbol{N}$.)

Důkaz: Ukážeme, že perceptronový algoritmus učení přiblíží počáteční váhový vektor $\vec{w}_0$ dostatečně blízko „hledaného řešení“ $\vec{w}^*$.

Konvergence perceptronového algoritmu učení (2)

Tři zjednodušení – bez újmy na obecnosti:

a) Namísto $\boldsymbol{P}$ a $\boldsymbol{N}$ vytvoříme jedinou množinu
$P' = P \cup N^-$ (N^- tvoří „negované“ prvky z $\boldsymbol{N}$)

b) Vektory z $\boldsymbol{P'}$ budou normalizované
(Jestliže byl nalezen váhový vektor $\vec{w}$, pro který platí $\vec{w} \cdot \vec{x} > 0$, potom totéž platí i pro každý další vektor $\eta\ \vec{x}\ ; \eta > 0$.)

c) Váhový vektor bude také normalizovaný
(Předpokládané normalizované řešení problému lineární separace budeme označovat jako $\vec{w}^*$.)

Konvergence perceptronového algoritmu učení (3)

- Předpokládejme, že po $t+1$ aktualizacích byl vypočten váhový vektor $\vec{w}_{t+1}$

  $\rightarrow$ to znamená, že po t aktualizacích byl vektor $\vec{p}_i \in P'$ chybně klasifikován (pomocí váhového vektoru $\vec{w}_t$), a tedy $\vec{w}_{t+1} = \vec{w}_t + \vec{p}_i$

- Kosinus úhlu ρ mezi $\vec{w}_{t+1}$ a $\vec{w}^*$ je:

$$\cos \rho = \frac{\vec{w}^* \cdot \vec{w}_{t+1}}{\left\| \vec{w}_{t+1} \right\|} \qquad (*)$$

Konvergence perceptronového algoritmu učení (4)

- Pro výraz v čitateli (*) víme, že:

$$\vec{w}^* \cdot \vec{w}_{t+1} = \vec{w}^* \cdot \left(\vec{w}_t + \vec{p}_i\right) = \vec{w}^* \cdot \vec{w}_t + \vec{w}^* \cdot \vec{p}_i \geq \vec{w}^* \cdot \vec{w}_t + \delta$$

kde $\delta = \min \left\{ \vec{w}^* \cdot \vec{p} \; ; \; \forall \vec{p} \in P' \right\}$

  - Protože váhový vektor $\vec{w}^*$ definuje absolutní lineární separaci $\boldsymbol{P}$ a $\boldsymbol{N}$, víme, že $\delta > 0$

    $\rightarrow$ indukcí dostáváme:

$$\vec{w}^* \cdot \vec{w}_{t+1} \geq \vec{w}^* \cdot \vec{w}_0 + (t+1)\,\delta \qquad (**)$$

Konvergence perceptronového algoritmu učení (5)

- Zároveň víme, že pro výraz ve jmenovateli (*) platí:

$$\left\|\vec{w}_{t+1}\right\|^2 = \left(\vec{w}_t + \vec{p}_i\right)\cdot\left(\vec{w}_t + \vec{p}_i\right) = \left\|\vec{w}_t\right\|^2 + 2\,\vec{w}_t \cdot \vec{p}_i + \left\|\vec{p}_i\right\|^2$$

  - Protože $\vec{w}_t \cdot \vec{p}_i \leq 0$

    (Jinak by nebylo třeba aktualizovat $\vec{w}_t$ podle $\vec{p}_i$.)

  - Platí, že: $\left\|\vec{w}_{t+1}\right\|^2 \leq \left\|\vec{w}_t\right\|^2 + \left\|\vec{p}_i\right\|^2 \leq \left\|\vec{w}_t\right\|^2 + 1$

    (Protože všechny vektory z P′ byly normalizovány.)

    → indukcí dostáváme:

$$\left\|\vec{w}_{t+1}\right\|^2 \;\leq\; \left\|\vec{w}_0\right\|^2 + \left(\, t + 1 \,\right) \qquad (***)$$

Konvergence perceptronového algoritmu učení (6)

- Z (**) a (***) dostáváme porovnáním s (*) nerovnici:

$$\cos \rho \geq \frac{\vec{w}^{*} \cdot \vec{w}_0 + (t+1)\delta}{\sqrt{\|\vec{w}_0\|^2 + (t+1)}}$$

→ pravá strana nerovnice roste proporcionálně k $\sqrt{t}$, a protože $\delta > 0$, mohla by být libovolně velká

× protože ale $\cos \rho \leq 1$, musí existovat horní mez a počet aktualizací váhového vektoru musí být konečný.

QED

Přihrádkový algoritmus učení (1)
Gallant, 1990

(aproximuje „ideální lineární separaci“)

IDEA:

- Nejlepší váhový vektor nalezený pomocí perceptronového algoritmu učení je „uložen v přihrádce“
- Současně se pokračuje v aktualizaci váhového vektoru
- Pokud se najde „lepší“ váhový vektor, nahradí se ním vektor uložený v přihrádce

Přihrádkový algoritmus učení (2)

START:

- Náhodná inicializace váhového vektoru $\vec{w}$ a uložení váhového vektoru do přihrádky: $\vec{w}_S = \vec{w}$
- Nastavení historie uloženého váhového vektoru: $\boldsymbol{h_S = 0}$

ITERACE:

- Aktualizace $\vec{w}$ pomocí jedné iterace perceptronového algoritmu učení
- Aktualizace $\boldsymbol{h}$ podle po sobě jdoucích úspěšně testovaných vektorech
- Jestliže nastane $\boldsymbol{h > h_S}$, nahraď $\vec{w}_S$ vektorem $\vec{w}$ a $\boldsymbol{h_S}$ číslem $\boldsymbol{h}$
- Pokračuj v iteraci

Přihrádkový algoritmus učení (3)

- Protože se bere v úvahu jen informace o posledně zvolených vzorech, může dojít i k záměně „správného“ váhového vektoru za horší – pravděpodobnost tohoto jevu by však měla klesat s rostoucím počtem iterací
- Pokud je trénovací množina konečná a složky váhového a příznakových vektorů jsou racionální, lze ukázat, že přihrádkový algoritmus konverguje k optimálnímu řešení s pravděpodobností 1