

# Referát z předmětu 36SPA Neuročipy

Oldřich Nič

## Neuronová síť

### Co je neuronová síť?

Struktura pro distribuované paralelní zpracování dat, skládající se obvykle z vysokého počtu vzájemně propojených výkonných prvků. Každý z nich přijímá konečný počet vstupních dat a transformuje je podle jisté přenosové funkce na jediný výstupní údaj, přičemž se může uplatnit i jeho vnitřní paměť.

- realizace funkce  $\vec{Y} = T(\vec{X})$ , převod vstupních dat na výstupní, popis T nemusí být dán analyticky...

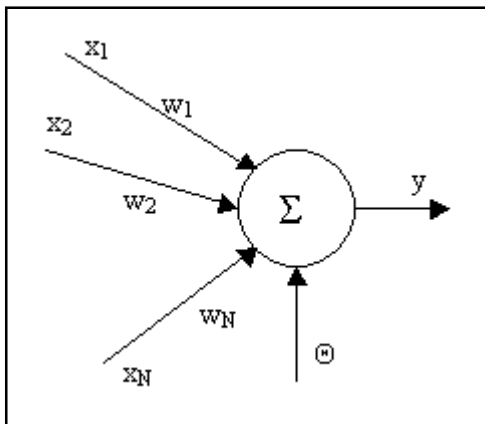
### Na jakém principu pracuje?

- chování sítě není formováno analytickým popisem, je určeno ve fázi učení
  - Nenatréované síti, která nemá tušení o skutečné podobě T, se předkládají vzory – vstupy X...
  - ...síť reaguje, svůj výsledek srovná s výsledkem správným, který jsme přiložili – výstupy Y...
  - ...opraví T podle stanoveného algoritmu...
  - ...proces se mnohokrát opakuje.
  - naučená síť vybavuje: na vstup vydává asociovaný výstup, vybavuje si naučené vzory



### Struktura sítě

#### Výkonný prvek – neuron

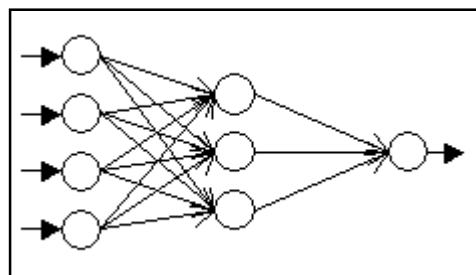


- převod vstupních dat na výstupní podle vztahu:

$$y = S\left(\sum_{i=1}^N w_i x_i + \Theta\right)$$

- pro různé síť různé modifikace takového neuronu (toto konkrétně perceptron)

#### Vlastní síť



- spojení neuronů ve směru šíření signálu: orientovaný graf
- na propojení závisí vlastnosti sítě

## Typy sítí

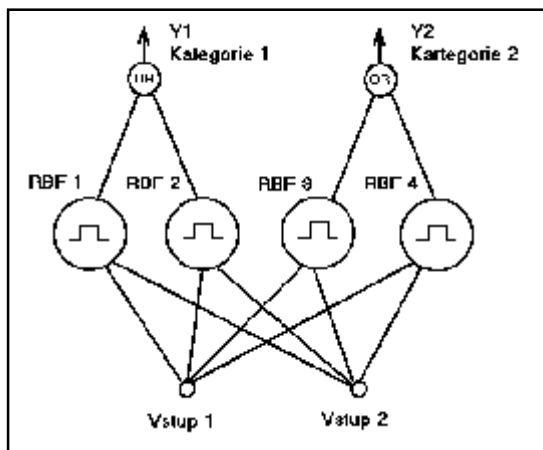
- odzkoušené typy sítí vhodné pro určité úlohy
- specifikace charakteristiky neuronu, spojení neuronů, učící algoritmus

## Aplikace

- predikce – přepovídání hodnot veličin na základě jejich průběhu v minulosti
- aproximace – přibližné určení hodnoty
- rozpoznávání – kategorizace vstupů
- filtrace – vyhlazení průběhu vstupů
- optimalizace – optimální hodnota proměnné

## Sítě RBF

### Struktura



- dvě vrstvy neuronů (+vstupní), vrstva vstupní je se skrytou úplně propojena, skrytá s výstupní může být propojena jen částečně
- ve výstupní vrstvě neurony perceptronové
- ve skryté vrstvě neurony RBF

### RBF neurony

$$y = S\left(\sqrt{\sum_{i=1}^N (x_i - c_i)^2}\right)$$

- údaj v závorce (vnitřní potenciál) je vzdáleností vstupního vektoru X a vektoru C v Eukleidovské metrice – vektor C označíme jako prototyp, je vlastní každému neuronu
- funkce S (aktivační funkce) je zpravidla Gaussova – při shodě nabývá maxima – v hardwarové implementaci nespojitá funkce – když je potenciál menší než sféra vlivu, neuron aktivní, jinak ne - obecně označuje podobnost X, C

### Aplikace

- aproximace
- klasifikace – zařazení kategorie podle vzoru – rozpoznávání shluků

### Práce sítě

- každopádně s překládáním párů vstup, správný výstup
- dvě fáze
  - určení prototypu pro neuron – náhodně vybraný vzor ze vstupu nebo sofistikovaně
  - určení vah výstupů – gradientní metody
- vybavování – klasifikace do tříd

## Číslicová implementace

- letem světem

- programově – Neural Works II, rychlost na výkonu počítače
  - akcelerátory – specializace hardware pro simulaci neuronových sítí
  - DSP procesory pro realizaci sčítání, násobení
- neuročipy – architektura speciálně pro neuronové sítě
- univerzity, komerčně IBM, Intel, specializované

## Neuročip ZISC

- zkratka Zero Instruction Set Computer – lze konfigurovat, ale ne programovat

### Jak to vypadá?

- konkrétně typ ZISC036 (36 neuronů)
- firma IBM, Essonnes Lab poblíž Paříže v roce 1994
- prodává se jako ISA karta (16 čipů, tedy 576 neuronů), PCI karta s jedním čipem a třemi sloty pro rozšiřující moduly, až 684 neuronů
- obrázek!

### Implementace neuronů

- každý neuron reprezentuje jednu sféru vlivu, určenou středem, velikostí a kategorií
- zjednodušení oproti definici: čip implementuje metriky  $\sum_{i=1}^N |x_i - c_i|$  (ve 2D čtverec postavený na vrcholu) a  $\text{Max}_{i=1}^N |x_i - c_i|$  (čtverec na hraně)
- výpočet metriky: hodnoty  $x_i$  vstupují do obvodu postupně, odčítačka realizuje  $x_i - c_i$ , poté absolutní hodnota, následující registr jako akumulátor (v druhém případě hledání maxima)
- zjednodušení oproti definici: neurony výstupní jsou logické obvody počítající funkci OR, zaměřené na klasifikaci

### Ovládání

- řízení pomocí registrů – jednotlivých neuronů nebo celé sítě
- neurony:
  - Neuron Weight Register – pro zápis středu sféry prototypu
  - Neuron Actual Influence Field – velikost sféry vlivu
  - Category – kategorie
- síť:
  - Control and Status Register – řídící a stavový
  - Vector Component Register – postupné zapisování vzoru po složkách, vstup pro všechny neurony
  - Minimum Influence Field – minimální hodnota sféry vlivu
  - Maximum Influence Field – maximální hodnota sféry vlivu

### Rozpoznávání vzorů

- postupné zapsání 64 složek vzoru do registru VCR
- postupné vyhodnocení vzdálenosti od prototypu
- porovnání se sférou vlivu – spadá vzor do sféry vlivu – ano/ne?
- pokud ano, je vrácena hodnota jeho kategorie na sběrnici, stavový registr:
  - žádný neuron není aktivní
  - aktivní neurony patří do stejné kategorie
  - alespoň dva aktivní neurony nepatří do této kategorie

## **Učení**

- dva typy algoritmů: implementovaný a s účastí vnější aplikace (dovoluje číst vzdálenost vzor – prototyp)
- vyhodnocení, neuron, do jehož sféry vlivu padne vzor, zapíše kategorii:
  - žádný neuron není aktivní: založí se nový neuron s prototypem jako vzor, sféra vlivu jako nejkratší vzdálenost vzor a středy již založených neuronů
  - aktivní neuron nespadá do požadované kategorie: redukce sféry vlivu na vzdálenost mezi středem sféry a vzorem: Redukce nepřekračuje hodnotu v MIF
  - aktivní neuron náleží ke kategorii: žádná akce
- odlišnosti od teorie: aktivní zakládání neuronů, u prototypů úprava jen sféry, ne středu, víc neuronů, než by bylo nutné – úprava pomocí druhého typu učícího algoritmu

## **Akcelerátor a počítač**

- vnitřní sběrnice přes FPGA připojena na ISA/PCI sběrnici, možnost propojit více karet mezi sebou
- max. dimenze prototypu i vstupu je 64, každá složka na 8 bitech
- frekvence 20 Mhz, klasifikace vzoru 64 za 4  $\mu$ s
- rozdělení neuronů do skupin – vytváření podsítí
- funkce uložit/načít – přenesení naučených dat z čipu na čip
- registry pro ovládání mapovány do V/V prostoru – jednoduché ovládání buďto přímo nebo pomocí dodávaných DLL knihoven
- realizované příklady: počítání cestujících v pařížském metru, rozpoznávání bílých krvinek, klasifikace hran, potlačování šumu