

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



Projekt SFC

Zadání: 27. Demonstrace učení sítě CPN

Zpracoval: Vladislav Halva
Datum: 8. prosince 2019

1 Counterpropagation neuronové sítě (CPN)

CPN (Counterpropagation network) jsou dvouvrstvé neuronové sítě. Hlavním využitím těchto sítí je komprese a aproximace funkcí. [3]

První, skrytá (Kohonenova) vrstva provádí přiřazování vstupních vektorů do shluků. Základem je soutěživé učení bez učitele podobně jako u SCL (Simple Competitive Learning) algoritmu. Neurony této vrstvy mají radiální bázové funkce. Vítězný neuron r^* má vždy výstupní hodnotu $y_{r^*} = 1$, všechny ostatní neurony mají výstupní hodnoty $y_r = 0, r = 1 \dots q, r \neq r^*$, kde q je počet neuronů skryté vrstvy. [1]

Druhá, výstupní (Grossbergova) vrstva je plně propojena se skrytou vrstvou. Neurony této vrstvy mají lineární bázové i aktivační funkce a jejich učení probíhá s učitelem. Platí tedy : [1]

$$o_j = y_j = \sum_{r=1}^q v_{jr} \cdot y_r, j = 1 \dots m$$

kde m je počet neuronů výstupní vrstvy, q počet neuronů skryté vrstvy a v_{jr} váha mezi výstupním neuronem j a skrytým neuronem r .

Výstupem této vrstvy (odezva \vec{o}) je poté jistý průměr vektorů trénovací množiny, které byly zařazeny do shluku reprezentovaného vítězným neuronem skryté vrstvy.

Učení CPN neuronových sítí probíhá s vektorovými páry (\vec{i}, \vec{d}) a je rozděleno do dvou fází. V první fázi jsou vstupní vektory shlukovány, probíhá tedy nastavování vah neuronů skryté vrstvy. V druhé fázi jsou upravovány váhy neuronů výstupní vrstvy tak, aby bylo dosaženo požadovaného výstupu pro každý shluk reprezentovaný neurony skryté vrstvy. [1]

CPN sítě se dělí do dvou typů: **Full counterpropagation** a **Forward-only counterpropagation**.

1.1 Forward-only counterpropagation

Je zjednodušená varianta sítě Full counterpropagation. Sít' používá v první fázi učení, tedy ke shlukování ve skryté vrstvě, pouze vstupní vektory \vec{i} . Vstupní vektory \vec{d} jsou následně použity ve druhé fázi učení, při trénování (nastavování) vah mezi neurony skryté a výstupní vrstvy. Výstupem této sítě je vektor \vec{y} , který je průměrem všech vektorů \vec{d} trénovací množiny, které byly zařazeny do daného shluku. [4]

1.1.1 Algoritmus učení neuronové sítě [1]

1. fáze - učení neuronů skryté vrstvy

1. Nastavení kroku učení $k = 0$, inicializace koeficientu učení μ , inicializace všech vah $w_{ri}(0), r = 1 \dots q, i = 1 \dots n$ neuronů skryté vrstvy náhodnými hodnoty z rozsahu $< 0, 1 >$.

2. Postupná iterace přes všechny prvky trénovací množiny \vec{i} :

- Inkrementace kroku učení k.
- Nalezení vítězného neuronu ve skryté vrstvě r^* , tj. jehož váhový vektor $\vec{w}(k-1)$ je nejbližší vstupnímu vektoru \vec{i} , hledáme tedy neuron s minimální hodnotou:

$$\delta = \sqrt{\sum_{i=1}^n (w_{ri} + i_i)^2}$$

- Upravení vah vítězného neuronu dle vztahu:

$$\vec{w}_{r^*}(k) = \vec{w}_{r^*}(k-1) + \mu \cdot (\vec{i} - \vec{w}_{r^*}(k-1))$$

3. Pokud nejsou splněny ukončovací podmínky (např. počet cyklů učení), pokračuj krokem 2.

2. fáze - učení neuronů výstupní vrstvy

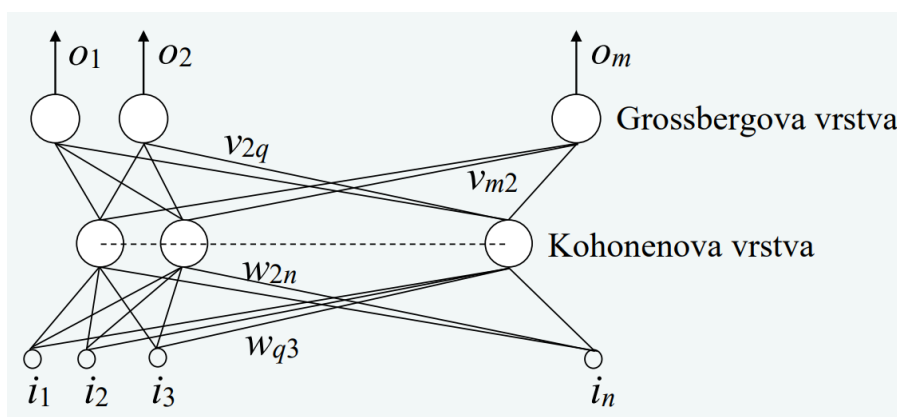
1. Nastavení kroku učení $k = 0$, inicializace koeficientu učení μ , inicializace všech vah $v_{jr}(0), j = 1 \dots m, r = 1 \dots q$ neuronů výstupní vrstvy náhodnými hodnoty z rozsahu $< 0, 1 >$

2. Postupná iterace přes všechny prvky trénovací množiny \vec{i} :

- Inkrementace kroku učení k.
- Nalezení vítězného neuronu ve skryté vrstvě r^* , tj. jehož váhový vektor $\vec{w}(k-1)$ je nejbližší vstupnímu vektoru \vec{i} .
- Upravení vah vedoucích od vítězného neuronu k výstupním neuronům dle vztahu:

$$v_{jr^*}(k) = v_{jr^*}(k-1) + \mu \cdot (d_{pj} - v_{jr^*}(k-1)), j = 1 \dots m$$

3. Pokud nejsou splněny ukončovací podmínky (např. počet cyklů učení), pokračuj krokem 2.



Obrázek 1: Forward-only counterpropagation síť. [1]

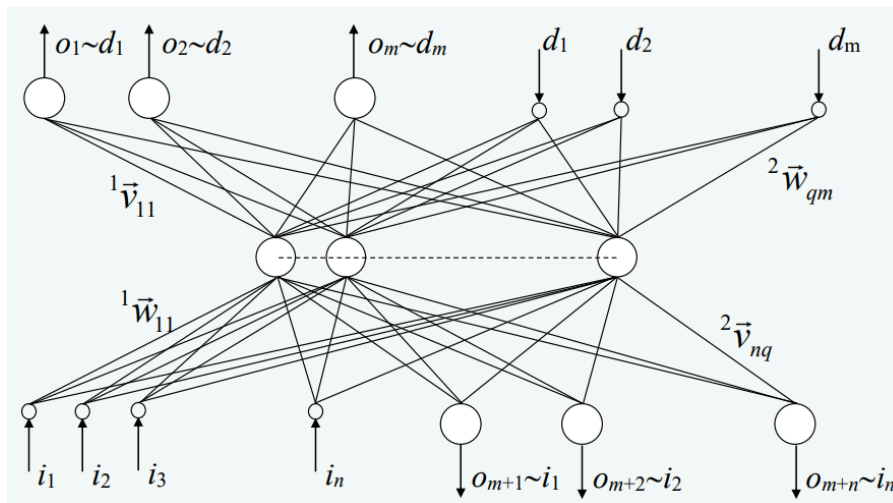
1.2 Full counterpropagation

Full counterpropagation síť je velice podobná síti Forward-only counter propagation. Na rozdíl od Forward-only varianty je však schopna obousměrného mapování. Síť dále předpokládá normalizované vstupní vektory a váhové vektory neuronů skryté vrstvy. [1]

1.2.1 Algoritmus učení neuronové sítě

Algoritmus učení je opět velice podobný jako u sítě u Forward-only varianty. Rozdíl spočívá v několika následujících bodech: [1] [2]

- Při učení sítě se na vstup přikládají vektorové páry (\vec{i}, \vec{d}) .
- Ve skryté vrstvě vítězí neuron r^* , jehož dvojice váhových vektorů $(\vec{w}_r^1, \vec{w}_r^2)$ je nejbližší vstupnímu vektorovému páru.
- Sít' má tedy zdvojenou vstupní a výstupní vrstvu. Učení výstupní vrstvy probíhá obdobně jako u Forward-only sítě.



Obrázek 2: Full counterpropagation sít'. [1]

2 Implementace

Program je implementován v jazyce Java s využitím knihovny JavaFX pro tvorbu grafického uživatelského rozhraní. Aplikace demonstruje učení Full a Forward-only counterpropagation neuronových sítí.

Trénovacími daty jsou vektorové páry (třírozměrných vektorů), které reprezentují barvy v RGB a HSV modelu, takto zvolená trénovací data umožňují snadnější vizualizaci samotného učení. Trénovací množina, jejíž rozsah volí uživatel, je vždy náhodně generována třídou **DatasetGenerator** při inicializaci učení.

Aplikace kromě demonstrace a vizualizace učení umožňuje také vyhodnotit odezvu neuronové sítě na zvolené vstupní vektory, po jejím naučení. K řízení stavu aplikace slouží třída **AppStateCarrier**.

Řízení samotného procesu učení je implementováno v metodách třídy **CPModel**. Tuto třídu dále rozšiřují třídy **ForwardCPModel** a **FullCPModel**, které reprezentují samotné neuronové sítě a jsou v nich implementovány metody pro výpočet a kroky specifické pro jednotlivé typy sítí. Tyto třídy k výpočtu a reprezentaci sítě využívají zejména následující třídy: *HiddenLayerNeuron*, *OutputLayerNeuron* a *InputPoint*.

Za zobrazení menu, vizualizace neuronové sítě a stavového panelu jsou zodpovědné třídy *LeftMenuController*, *CanvasPaneController* a *BottomBarController*.

3 Manuál programu

3.1 Spuštěná aplikace

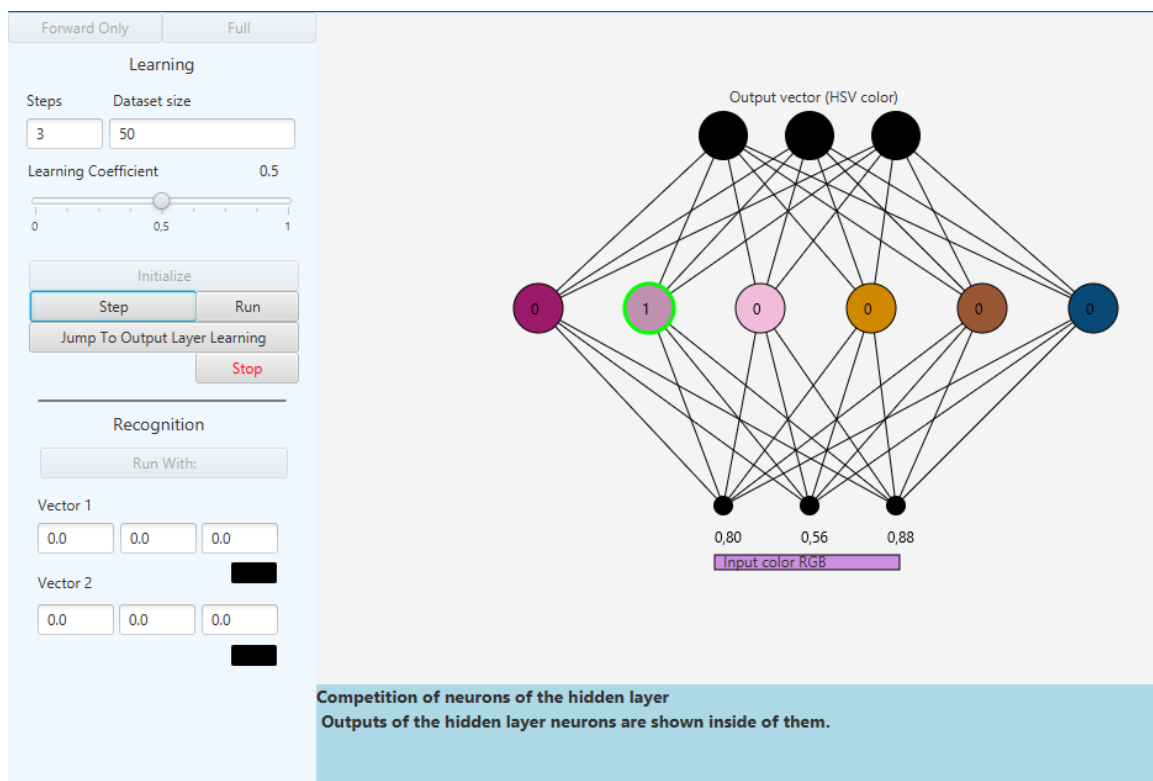
Aplikaci je napsána v Javě verze 1.8.0_231 a lze ji spustit zadáním následujícího příkazu v adresáři *dist*.

```
java -jar CPNDemonstration.jar
```

3.2 Obsah adresáře

- Dokumentace programu *xhalva04.pdf* (tento soubor).
- Zdrojové soubory v adresáři *src*.
- Spustitelná aplikace *CPNDemonstration.jar* v adresáři *dist*.

3.3 Ukázka aplikace



Obrázek 3: Ukázka aplikace.

3.4 Popis ovládání

Aplikace je rozdělena na tři základní části: levý ovládací panel, zobrazovací plocha a stavový panel viz 3.

Po spuštění je nejprve nutné vybrat mezi demonstrací Full a Forward-only counterpropagation sítě tlačítka *Full* a *Forward-only*. Následně je třeba nastavit *Steps* (počet učicích cyklů), *Dataset size* (velikost trénovací množiny) a *Learning Coefficient* (učicí koeficient), který je společný pro obě fáze učení. Následně lze tlačítkem *Initialize* spustit učení sítě.

Tlačítkem *Step* lze vždy učinit jeden krok učení sítě. Tlačítko *Jump to Output Layer Learning* dokončí fázi učení neuronů skryté vrstvy a lze dále tlačítkem *Step* pokračovat v procházení kroků učení neuronů výstupní vrstvy. Tlačítko *Run* dokončí učení neuronové sítě.

Po dokončení učení lze v dolní části levého ovládacího panelu zadat vstupní vektor/y, a tlačítkem *Run With:* spustit výpočet odezvy na tento/tyto vstupní vektor/y.

Tlačítkem *Stop* lze kdykoliv přerušit fázi učení, nebo výpočet odezvy sítě a následně lze inicializovat novou demonstraci učení sítě. V dolním stavovém panelu jsou vždy zobrazeny informace o aktuálním stavu učení/výpočtu odezvy, případně pokyny pro další práci s aplikací.

Reference

- [1] *Slidy k předmětu SFC (Softcomputing)*, FIT VUT v Brně,
URL: https://www.fit.vutbr.cz/study/courses/SFC/private/19sfc_3.pdf, [Online Accessed: 2019-12-7].
- [2] Freeman, James A. and Skapura, David M.: *Neural Networks: Algorithms, Applications, and Programming Techniques*. Addison Wesley Longman Publishing Co., Inc., 1991, ISBN: 0-201-51376-5
- [3] *Our Education Blog*: CPN(Counterpropagation Network),
URL: <https://blog.oureducation.in/cpn-counterpropagation-network/>, [Online Accessed: 2019-12-7].
- [4] *KopyKitab*: Counter Propagation Network Notes,
URL: <https://www.kopykitab.com/blog/counter-propagation-networks-notes/>,
[Online Accessed: 2019-12-7].