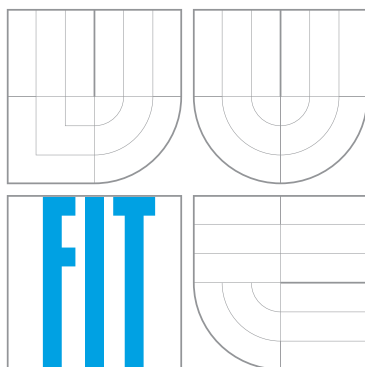


# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



Soft Computing - projekt

## Demonstrace komprese dat pomocí Back Propagation

## Obsah

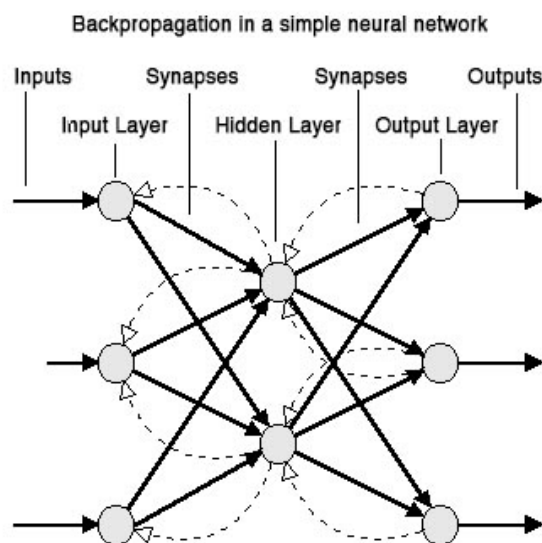
## Zoznam obrázkov

## Zoznam tabuliek

## 1 Abstrakt

Nasledujúci text sa zaoberá kompresiou *grayscale* obrázkov pomocí bipolárni a sigmodiální Back Propagation neuronovej sítí. Bipolárni BP prevzatej z [?]. Je zde uvedený algoritmus učení BP pre sigmodiální a bipolárnu aktivačnú funkciu. Na konci dokumentu sa implementované algoritmy testujú, porovnávajú a je analyzovaná chyba kompresie implementovaných algoritmov.

## 2 Teoretický úvod



Obr. 1: neuronová sít back propagation [?]

### 2.1 Algoritmus učení BP

1. do { Set GlobalError to 0
  - (a) for all input samples  $p$  {
    - i.  $X = X_p$
    - ii. for all layers  $l$  {
      - A. for all neurons on the layer  $j$  {
      - B. set  $u$  to 0

- C. for all neuron inputs  $i$  {
- D.  $u = u + {}^l w_{ji}^l x_i$
- E.  ${}^l y_j = 1/(1 + e^{-\lambda u})$  .... SIGMODIÁLNÍ AKTIVAČNÍ FUNKCE
- F.  ${}^l y_j = (1 - e^{-\lambda u})/(1 + e^{-\lambda u})$  .... BIPOLÁRNÍ AKTIVAČNÍ FUNKCE[?]
- (b) set error  $E$  to 0
- (c) for all neurons  $j$  on output layer  $L$ 
  - i.  $E = E + (d_j - {}^L y_j)^2$  to 0
  - ii.  ${}^L \delta_j = (d_j - {}^L y_j) \lambda^L y_j (1 - {}^L y_j)$
- (d)  $GlobalError = GlobalError + \mu * E$
- (e) for all layers that are not output  $l$ 
  - i. for all neurons  $j$  on current layer
    - A.  ${}^l \delta_j = (\sum_{k=1}^{n_{l+1}} {}^{l+1} \delta_k^{l+1} w_{kj}) \lambda^l y_j (1 - {}^l y_j)$
- (f) for all layers  $l$  .... vypočet nových váh
  - i. for all neurons  $j$  on current layer
    - A. for all neuron inputs  $i$
    - B.  $\Delta^l w_{ji} = \mu^l \delta_j^l x_i$
    - C.  ${}^l w_{ji} = {}^l w_{ji} + Delta^l w_{ji}$
- 2. while  $GlobalError > \epsilon$  AND not maxstep

## 2.2 Transformace obrázku na trénovací množinu vzorkov

Povedzme, že sa snažíme preniesť *grayscale* obrázok 256x256. Najprv je nutné tento obrázok rozdeliť na niekoľko rovnako veľkých blokov, pričom počet pixelov v bloku definuje počet neurónov vo vstupnej resp. výstupnej vrstve. Tieto bloky budú reprezentovať sadu vstupných vzorkov na ktorých sa BP bude učiť. Povedzme že tieto bloky sú 8x8 bodov veľké. Potom potrebujeme 64-K-64 BP neurónovú sieť v prípade ak uvažujeme 3-vrstvovú sieť. Body bloku očísľujeme z ľava do prava a zhora nadol. Aby bola komprese efektívna musí platiť  $k < 64$ .

## 2.3 Princíp komprese dat

Najprv príjemca a odosielateľ obrázku naučia hore uvedenú neuronovú sieť na rovnakú sadu vzorkov. Ako toho docielia nie je predmetom tejto práce. Uvedieme len, že učenie prebieha nad identitou vstupných vzorkov.

V nasledujúcom texte uvedieme, pre jednoduchosť, scenár prenosu len jedného bloku obrázku.

Odosielateľ zakóduje odosielaný obrázok a to tak, že vypočítá výstupy všetkých neurónov skrytej vrstvy pre odosielaný blok dát. Tieto výstupy sú reálne hodnoty, v prípade BBP sú v intervale  $< -1.0, 1.0 >$  a v prípade SBP v intervale  $< 0.0, 1.0 >$ . Preto sa tieto hodnoty musia najprv zaokrúhliť do binárnej podoby.

Príjemateľ prijme tieto zaokrúhlené dáta, konkrétne  $K * 2^n$ , kde  $n$  je rozsah zaokrúhlených čísel. Následne sa vypočítajú nové hodnoty výstupov neurónov výstupnej vrstvy a tým sa rozkódujú dáta.

V prípade, ak sa jedná o viac ako 3-vrstvú sieť, odosielateľ odosiela len hodnoty výstupov vrstvy bezprostredne za výstupnou vrstvou. V prípade prenosov viacerých obrázkov sa váhy vstupov jednotlivých neurónov už nemenia.

Kompresný pomer je potom  $\frac{K*N}{8*8*8}$  pre prípad blokov dát 8x8 a *grayscale* formát v rozasahu 8 bitov.

### **3 Implementace**

### **4 Testování**

### **5 Závěr**

## **A Príloha 1: demonstrace komprese**