

# Sieci neuronowe, cd wykład 9

Joanna Jędrzejowicz

Instytut Informatyki

# Uczenie sieci metodą wstecznej propagacji (Back Propagation)

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Uczenie metodą BP ma dwie fazy:

- wzorzec treningowy aplikuje się do warstwy we, przechodzi przez kolejne warstwy ukryte i wylicza się sygnały warstwy wy, porównuje z wynikami i wylicza błąd, (poza warstwą wyjściową błędu nie można wyliczyć bezpośrednio, tylko przybliżyć)
- błąd propaguje się poczynając od warstwy wyjściowej przez kolejne warstwy ukryte (od ostatniej do przedostatniej itd)
  - modyfikując kolejne wagi połączeń

# Funkcja aktywacji

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Neurony używają funkcji aktywacji sigmoidalnej

$$Y^{sigmoid}(x) = \frac{1}{1+e^{(-x)}}.$$

Funkcja przyjmuje wartości z przedziału  $(0, 1]$  i ma łatwą pochodną:

$$Y' = \frac{-1 \cdot e^{-x} \cdot (-1)}{(1 + e^{-x})^2} = \frac{1}{1 + e^{-x}} \left(1 - \frac{1}{1 + e^{-x}}\right) = Y(1 - Y)$$

# Uczenie metodą BP - oznaczenia

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Zakładamy, że jest **jedna** warstwa ukryta. Oznaczenia:

$n$  liczba neuronów w warstwie we,

$m$  liczba neuronów w warstwie ukrytej,

$l$  - liczba neuronów w warstwie wyjściowej

$v_{ij}$  - waga połączenia neuronu  $j$  w warstwie we oraz  $i$  w warstwie ukrytej,

$w_{ij}$  - waga połączenia neuronu  $j$  w warstwie ukrytej oraz  $i$  w warstwie wy,

$\alpha$  - współczynnik wzrostu

# Uczenie metodą BP - korekta wag od warstwy ukrytej do wy

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

$p$  - numer iteracji,  
 $y_i^{(f)}(p)$  - wyliczone  $i$ -te wyjście  
 $y_{d,i}$  -  $i$ -te wyjście dokładne

$$\text{błąd } \epsilon_i(p) = y_{d,i} - y_i^{(f)}(p)$$

$$w_{ij}(p+1) = w_{ij}(p) + \Delta w_{ij}$$

$$\Delta w_{ij}(p) = \alpha \cdot y_j(p) \cdot \delta_i^{(2)}(p)$$

$y_j$  wartość wyjścia  $j$ -tego neuronu w warstwie ukrytej  
 $\delta_i^{(2)}(p)$  gradient, wyliczany jako wartość pochodnej funkcji sigmoidalnej pomnożony przez błąd;

# Uczenie metodą BP - korekta wag od warstwy ukrytej do wy

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

$$\delta_i^{(2)}(p) = \epsilon_i(p) \cdot z_i(p) \cdot (1 - z_i(p))$$

$$z_i(p) = \frac{1}{1 + e^{-s_i(p)}}$$

wartość wejścia do  $i$ -tego neuronu:

$$s_i(p) = \sum_{j=0}^m w_{ij}(p) \cdot y_j(p), \quad y_0 = 1$$

# Uczenie metodą BP - korekta wag od warstwy we do ukrytej

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Po dokonaniu korekty wag od warstwy ukrytej do wyjściowej i wyliczeniu gradientu  $\delta_i^{(2)}(p)$  dokonuje się korekty wag od warstwy wejściowej do ukrytej:

$$\Delta v_{ij}(p) = \alpha \cdot x_j(p) \cdot \delta_i^{(1)}(p)$$

$$\delta_i^{(1)}(p) = \epsilon_i^{(2)}(p) \cdot z_i(p) \cdot (1 - z_i(p))$$

$$\epsilon_i^{(2)}(p) = \sum_{r=1}^m w_{ri}(p) \cdot \delta_r^{(2)}(p)$$

# Uczenie metodą BP - korekta wag od warstwy we do ukrytej

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

$$z_i(p) = \frac{1}{1 + e^{-s_i(p)}}$$

$$s_i(p) = \sum_{j=0}^n v_{ij}(p) \cdot x_j(p), \quad x_0 = 1$$



# Uwagi

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Jeżeli uczenie sieci (modyfikacje wag) przeprowadza się każdorazowo po podaniu na wejście wektora uczącego, to mówimy o **przyrostowym uaktualnianiu wag**.

Możliwe jest **kumulacyjne uaktualnianie wag**:

- na wejście sieci podaje się kolejno wektory uczące,
- wyznacza się odpowiadające im sygnały wyjściowe,
- po porównaniu ich z wartościami wzorcowymi sumuje się błędy
- po zakończeniu podawania próbek z całej epoki dokonuje się korekty wszystkich wag z wykorzystaniem skumulowanej wartości błędu

# Sieci radialne - RBFN - Radial Basis Functions Networks

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Sieci radialne składają się z neuronów, których funkcje aktywacji realizują odwzorowanie:

$$\mathbf{x} \rightarrow \phi(\|\mathbf{x} - \mathbf{c}\|)$$

$\|\cdot\|$  jest normą euklidesową,  
funkcje  $\phi$  są **radialnymi funkcjami bazowymi**, ich wartości zmieniają się radialnie wokół środka  $\mathbf{c}$ .

Sygnały wejściowe  $x_1, \dots, x_n$  tworzące wektor  $\mathbf{x}$  są podawane jako wejścia do neuronów w warstwie ukrytej realizujących funkcje radialne, zaś neuron w warstwie wyjściowej realizuje sumę ważoną:

$$y = \sum_{i=1}^K w_i \phi(\|\mathbf{x} - \mathbf{c}_i\|)$$

$K$  jest liczbą funkcji bazowych

# Przykłady radialnych funkcji bazowych

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

- $\phi(||\mathbf{x} - \mathbf{c}_i||) = \exp\left(\frac{-||\mathbf{x} - \mathbf{c}_i||^2}{r^2}\right)$
- $\phi(||\mathbf{x} - \mathbf{c}_i||) = \frac{||\mathbf{x} - \mathbf{c}_i||^2}{r^2}$
- $\phi(||\mathbf{x} - \mathbf{c}_i||) = (r^2 + ||\mathbf{x} - \mathbf{c}_i||^2)^{-\alpha}$
- $\phi(||\mathbf{x} - \mathbf{c}_i||) = (r||\mathbf{x} - \mathbf{c}_i||^2) \ln(r||\mathbf{x} - \mathbf{c}_i||)$

dla  $r > 0$

# Sieci radialne - metody aktualizacji wag

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Dane: wektory wejściowe  $x_1, \dots, x_M$  oraz wartości rzeczywiste  $d_1, \dots, d_M$

Szukamy funkcji  $F$  takiej że  $F(x_i) = d_i$ ,

Przyjmujemy

$$F(x) = \sum_{i=1}^K w_i \phi(\|x - c_i\|)$$

gdzie zwykle  $K < M$ .

Minimalizuje się funkcję

$$E = \sum_{i=1}^M \left( \sum_{j=1}^K w_j \phi(\|x_i - c_j\|) - d_i \right)^2$$

# Sieci radialne - metody aktualizacji wag, cd

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Macierz Greena:

$$G = \begin{pmatrix} \phi(\|x_1 - c_1\|) & \cdots & \phi(\|x_1 - c_K\|) \\ \vdots & \ddots & \vdots \\ \phi(\|x_M - c_1\|) & \cdots & \phi(\|x_M - c_K\|) \end{pmatrix}$$

$$d = [d_1, \dots, d_M]^T$$

Szukamy  $w$

$$w = [d_1, \dots, w_K]^T$$

spełniającego

$$w = G^+ d$$

gdzie  $G^+$  jest pseudo inwersją macierzy prostokątnej  $G$

$$G^+ = (G^T G)^{-1} G^T$$

# Uczenie sieci radialnej

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Etapy:

- Dobiera się liczbę i kształt funkcji bazowych oraz srodki  $c_i$  metodami takimi jak dobór losowy, dobór metodą wstecznej propagacji,
- Wyznacza się wagi korzystając z

$$w = G^+ d$$

# Zastosowania sieci radialnych

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

Zastosowania podobne jak sieci sigmoidalnych:

- problemy klasyfikacyjne,
- zadania predykcji

Algorytm uczenia dla sieci radialnych jest uproszczony - sieci radialne mają architekturę złożoną tylko z dwóch warstw oraz w warstwie ukrytej mogą stosować różne funkcje bazowe, w przeciwieństwie do sieci sigmoidalnych.

# Przykład - zastosowanie sieci radialnej do klasyfikacji

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

```
library(RSNNS)
data(iris)
#shuffle the vector
iris <- iris[sample(1:nrow(iris),length(1:nrow(iris))),
1:ncol(iris)]
# dane wjsciove
irisValues <- iris[,1:4]
#dane wyjsciowe
irisTargets <- decodeClassLabels(iris[,5])
# podzial na dane treningowe i testowe
iris <- splitForTrainingAndTest(irisValues, irisTargets,
ratio=0.20)
#normalizacja danych
iris <- normTrainingAndTestSet(iris)
# tworzenie sieci RBFN Radial Basis Function Network
model <- rbf(iris$inputsTrain, iris$targetsTrain, size=5,
maxit=20, initFuncParams=c(-4, 4, 0.0, 0.2, 0.04),
learnFuncParams=c(1e-3, 0, 1e-3, 0.1, 0.8), linOut=FALSE)
```



## cd przykladu

Sieci  
neuronowe, cd  
wykład 9

Joanna  
Jędrzejowicz

```
# wydruk wag
weightMatrix(model)
# wydruki w oknie graficznym wymiaru 2 wiersze, 2 kolumny
par(mfrow=c(2,2))
#wydruk bledu
plotIterativeError(model)
# klasyfikacja danych testowych
predictions <- predict(model,iris$inputsTest)
plotRegressionError(predictions[,2], iris$targetsTest[,2])
#macierz bledu
confusionMatrix(iris$targetsTest,predictions)
plotROC(predictions[,2], iris$targetsTest[,2])
```