Lecture 3

Spring 2018

Faglærer: Magne Hallstein Johnsen,

Institutt for elektronikk og telekommunikasjon, NTNU

NTNU

Lecture content

- Plus and minus for linear discriminant classifiers (LDC)
- A discussion regarding LDC formulation and decision strategies
- Training functions/criterias and corresponding problem types
  - Some general guidelines
  - The perceptron criteria for linear separable problems (LSP)
  - Introducing the activation function
  - The least square error training criteria for nonseparable problems (NSP)
  - Extension to $C > 2$ classes

Plus and minus for LDC

- Matched to LSP and are only suboptimal for NSP
- Decision borders are hyperplanes
- Computationally simple both with respect to classification and training
- Trained on training data from all classes simultaneously
- Generalizes well; i.e. a LDC have few parameters to estimate by training
- Theoretically well studied and explicit solution methods exist (Fisher LDC)
- We will focus on the $C = 2$ class problem (but will show extension to $C > 2$)

A discussion regarding LDC formulation and decision strategies

- In the 2-class problem we can define a single discriminant function :

$$g(x) = g_1(x) - g_2(x) = w_1^T x + w_{01} - w_2^T x - w_{02} = w^T x + w_0 \qquad (1)$$

- We can simplify further by defining $x = [x\ 1]^T$ and $w = [w\ w_0]^T$

- This leads to the logical decision rule :

$$x \in \omega_1 \ \Leftrightarrow \ g(x) = w^T x > 0 \qquad (2)$$

- The deviation from the treshold zero will indicate how far x is from the decision border.

- In many cases it is an advantage to use discrete output targets.
  - In the 2-class problem we then would have two target values $\{t_1, t_2\}$
  - During training the outputs should to be as close as possible to the targets
  - During use/test the treshold will be the mean of the two targets
  - The activation function (to come) will facilitate this

General guidelines for training functions versus problem types

- A training function is a cost which should be optimized with respect to classifier parameters.
- Log Likelihood is an example of a statistical training function
- A variety of training functions exist; most are suited for NSP cases
- Most functions are suited for both linear and non-linear classifiers
- The perceptron function/criteria is a classical non-statistical function for LSP cases
- The (mean) square error from a chosen target is the most used for NSP cases
- Practically all functions have intrinsic form; i.e. iterative gradient techniques are needed

The perceptron criteria for a LSP

- Defining the class indicator function

$$t_k = \begin{cases} t_1 = +1 & x_k \in \omega_1 \\ t_2 = -1 & x_k \in \omega_2 \end{cases} \tag{3}$$

- Thus we want $t_k g(x_k) \geq 0$ for all x_k in the training set
- The perceptron training criteria is given by

$$J(w) = \sum_{x_j \in X_e} t_j g(x_j) < 0 \tag{4}$$

- X_e is the set of all **misclassified** training vectors $\Rightarrow$ all $t_j g(x_j) < 0$!
- The goal is then to maximize $J(w)$ with respect to w
- For a LSP we can have zero errors; i.e $J(w) \equiv 0$!
- Updating (iteration $m+1$) in the direction of the gradient :

$$w_{m+1} = w_m + \alpha \nabla_w J(w) = w_t + \alpha \sum_{x_j \in X_e} t_j x_j \tag{5}$$

NTNU

Introducing the activation function

- The activation function "squashes" the output towards target values.
- This reduces the influence of large output values on the gradient
- It also opens for Least (mean) square error (LS) criterias (next slide)
- Define $y(x) = w^T x$ and assume target class values $t_1 = 1$ and $t_2 = 0$
- The sigmoid is a popular activation function for these target values :

$$g(x) = \frac{1}{1 + e^{-y}} = \frac{1}{1 + e^{-w^T x}} \tag{6}$$

- Except for a small transition area round $y = 0$ the output $g(x)$ will be approximately equal to one of the two target values
- The sigmoid has a simple and positive derivative for all arguments :

$$\frac{\delta g(x)}{\delta y} = g(x)[1 - g(x)] \tag{7}$$

- Note that decision under use/test still can be done based on y; i.e no sigmoid is necessary
- An alternative to the sigmoid is tanh (combined with $t_2 = -1$)

The least square error training criteria for a NSP

- Given the sigmoid and the matching target class values the mean square error (MSE) training criteria is

$$J(w) = \sum_{k=1}^{N} [g(x_k) - t_k]^2 \tag{8}$$

- Using the chain rule for derivation we end up with :

$$\nabla_w J(w) = \sum_{k=1}^{N} [g(x_k) - t_k][1 - g(x_k)]g(x_k)x_k \tag{9}$$

- And since we use error as criteria we must perform a minimization :

$$w_{m+1} = w_m - \alpha \nabla_w J(w) \tag{10}$$

Extension to $C > 2$ classes

- We now need a sigmoid output for each class; i.e a C-dimensional output vector $g(x) = [g_1(x)....g_C(x)]^T$
- The corresponding target vector t has $C-1$ zero elements and a single element of value one
- The weights now have the form of a $C \times D$ matrix W; i.e $y = Wx$ (where $dim(x) = D$)
- The MSE criteria becomes

$$J(W) = \sum_{k=1}^{N} [g(x_k) - t_k]^T [g(x_k) - t_k] = \sum_{k=1}^{N} \sum_{i=1}^{C} [g_i(x_k) - t_{ik}]^2 \qquad (11)$$

- Introducing the elementwise multiplication operator $\circ$ on vectors the gradient matrix becomes

$$\nabla_W J(W) = \sum_{k=1}^{N} [[g(x_k) - t_k] \circ [1 - g(x_k)] \circ g(x_k)] x_k^T \qquad (12)$$