Department of Computer Science

TECHNISCHE UNIVERSITÄT
KAISERSLAUTERN

Machine Learning, 2
Prof. Dr. Marius Kloft and Dr. Antoine Ledent

Machine Learning 2, Winter Term 2019

Problem Sheet 2

Problem 1

Quadratic Discriminant Analysis

Consider a *Gaussian multi-variate empirical Bayes classifier*

$$f(\mathbf{x}) := \underset{y \in \{-1,+1\}}{\text{argmax}} \; p_{\mu_y,\Sigma_y}(\mathbf{x}) P(Y = y)$$

where $p_{\mu_y,\Sigma_y}(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \mu_y, \Sigma_y)$.

Task: Show analytically that the *decision boundary* of the above of classifier is a quadratic hypersurface. For simplicity, assume that $P(Y = 1) = P(Y = -1) = \frac{1}{2}$.

(a) First, assume $\mathbf{x} \in \mathbb{R}$. Then, the decision rule has the form

$$ax^2 + bx + c > 0$$

Determine the coefficients a, b, c!

(b) Now consider the d-dimensional case, i.e. $\mathbf{x} \in \mathbb{R}^d$. The decision boundary is again a *quadric*, and the decicion rule takes the form

$$x^\top Ax + b^\top x + c > 0$$

where $A \in \mathbb{R}^{d \times d}, \;\; b \in \mathbb{R}^d, \;\; c \in \mathbb{R}$. Determine $A, \;\; b, \;\; c$!

Problem 2

(Use Python only.) Consider the normal random variables X_1, X_2 in $\mathbb{R}^2$ with means $(-0.7, 0)$ and $(0.7, 0)$ respectively, and with covariance $\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$.

(a) Simulate 10 (and then 100, seperately) values of from both distributions.

(b) For both of the above cases, plot the values with different colours for each distribution.

(c) Run LDA as a classification algorithm on using the correct covariance matrix and means. Plot the decision boundary and give the equation. Calculate the accuracy on the data sets above.

(d) Run quadratic discriminant analysis assuming the covariance matrix (but not the means) is unknown, and estimated through maximum likelihood. Calculate the accuracy on the data sets above.

(e) Do the same thing assuming the mean is also unknown.

Problem 3

Stan is a probabilistic modeling tool. In this course, we will use it to implement the models learned in the class. In this exercise we implement the simple models we learned to get acquainted with the tool.

(a) Install STAN on python `https://pystan.readthedocs.io/en/latest/getting_started.html`.

(b) Implement LDA and QDA using STAN and use the data generated in Problem 2 to fit them and compare the errors with your python implementation.