BME777: Emerging Topics in Biomedical Engineering (Fall 2017)
(Machine Learning for Health Analytics)

Lab 1: Bayesian Decision Theory

Objective

- To understand and implement Bayes decision rule for performing simple classifications.

Background

Bayesian decision theory is a fundamental statistical approach to the problem of classification. It makes the assumption that the decision problem is posed in probabilistic terms, and that all the relevant probability values are known. The Bayes formula states that the posterior probability of a *state of nature* or *class* (ω_j) given the *observation* or *feature* (x) can be computed using the prior probability of the *class* (ω_j) and the class-conditional probability of *feature* (x) given the *class* (ω_j). According to the Bayes decision rule, the *feature* (x) is then assigned to the *class*(ω_j) with the highest posterior probability. The formal definition of the Bayes formula is given by the Equ.1 as follows:

Bayes Formula

$$P(\omega_j|x) = \frac{p(x|\omega_j)\ P(\omega_j)}{p(x)} \tag{1}$$

where in case of c classes

$$p(x) = \sum_{j=1}^{c} p(x|\omega_j)\ P(\omega_j) \tag{2}$$

$P(\omega_j|x)$ = Posterior probability of *class* (ω_j) given the feature (x).
$p(x|\omega_j)$ = Class-conditional probability density of *feature* (x) given the *class* (ω_j).
$P(\omega_j)$ = Prior probability of *class* (ω_j).
$p(x)$ = Probability density function of the *feature* (x).

In general for a multi-dimensional feature input $\mathbf{x} = [x_1, x_2, ..., x_k]$, the Bayes formula could be written as

$$P(\omega_j|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_j)\ P(\omega_j)}{p(\mathbf{x})}. \tag{3}$$

Laboratory Exercises

Part I

The *lab1.zip* file contains this handout, *Diabetes.mat*, and a skeleton MATLAB program *lab1.m*. Load the "Diabetes.mat". This loads a matrix variable "Diabetes" to the MATLAB workspace. The dataset contains two features [x_1 - plasma glucose concentration and x_2 - diastolic blood pressure] from two different groups (ω_1 - positive for Diabetes and ω_2 - negative for Diabetes). The size of the matrix variable "Diabetes" is 536 X 3 (i.e. 536 rows and 3 columns). Each row corresponds to one case, the first two columns contains the features x_1 and x_2, and the third column contains the class labels ω_1 and ω_2. Explore commands *hist* and *boxplot* to study the distribution of each of the features for each of the classes. The class-conditional probability densities of each of x_i are assumed to be Gaussian. The general from of a univariate Gaussian density is given by

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma}\, exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right] \tag{4}$$

where μ and σ are mean and standard deviation of the density function.

- Using a single discriminant function $g(x_1)$, design a 2 class minimum-error-rate classifier(dichotomizer) from the given data, to classify the cases into either those who are positive or negative to Diabetes, according to x_1.
- Write a program that will take the feature value as the input and will return the posterior probabilities and the value of $g(x_1)$.
- Arrive at a optimal threshold ($Th1$) that separates class ω_1 and ω_2.
- Arrive at a threshold ($Th2$) if a high penalty is associated for classifying the class ω_1 as class ω_2.
- Identify the class labels for the below feature values using your program.

  $x_1 = [180,\ 130,\ 50,\ 80,\ 100]$
- Adjust your program and redesign your classifier with x_2 as the feature. Suggest which of the two features might be a better choice for separating the two classes ω_1 and ω_2. Justify.

Part II

This is a bonus part and will not be used for evaluation. However students are encouraged to do this part to get a better understanding of multivariate data. For this part of the lab we will use the previously loaded matrix variable "Diabetes" and design a classifier that will use both features i.e., $\mathbf{x}= [x_1\ x_2]$. The class-conditional probability

densities of $\mathbf{x}$ are assumed to be multivariate Gaussian. The general form of a multivariate Gaussian density is given by

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}}|\sum|^{0.5}} \, exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^t \sum^{-1}(\mathbf{x}-\boldsymbol{\mu})\right] \tag{5}$$

where $\mathbf{x}$ is a d component vector, $\boldsymbol{\mu}$ is the d component mean vector, $\sum$ is the d by d covariance matrix, and $|\sum|$ and $\sum^{-1}$ are its determinant and inverse.

- Using both the features [x_1, x_2] and a single discriminant function $g(\mathbf{x})$, design a 2 class minimum-error-rate classifier(dichotomizer) from the given data.
- Write a program that will take the feature vector as input and will return the posterior probabilities and the value of $g(\mathbf{x})$.
- Plot the feature space using the two features.

Lab Evaluation Information

- Submit via email the documented MATLAB code for Part I of the lab exercise before the demo. In the lab, the programs should be demonstrated to the TA/Instructor and each student will be evaluated on the components covered in the lab material. Marks will be provided based on the (i) demo and (ii) student′s response to the questions [3 + 4 marks].
- Submit via email a report including the answers to the questions asked in Part I, and your observations and conclusions. (3 marks)

Due date

- Due date for the report will be before the start of the next lab. Reports can be submitted via email till 5 pm on the due date, however demo/ evaluation in lab must be completed during scheduled lab sessions for this lab (i.e., Week 2-4).

Acknowledgment

We thankfully acknowledge UCI Machine Learning Repository for the dataset used in this lab exercise.
Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.