Tulha

FNU Tulha

October 2016

1 Introduction

$$f(x) = (1/((2\pi\sigma^2)^{0.5}))exp((-1/2)[(X-\mu)/\sigma] \tag{1}$$

Since we have a continuous dataset we will first assume a Gaussian (normal) distribution for the training dataset.

$$f(x) = (1/((2\pi\sigma^2)^{0.5}))exp((-1/2)[(X-\mu)/6] \tag{2}$$

But for one specific case:

$$p(x_i|y) = 1/((2\pi\sigma_y^2)^{0.5})e^{(-(x_i-\mu_y)^2/2\sigma_y^2)} \tag{3}$$

So the features we would like to determine the posterior probability for, can be done using this. The second thing we will assume is the independence of all these features and the probabilities of these features occuring given a specific class.

Training set

$$p(c|x_i) = p(x_i|c)p(c)/p(x_i) \tag{4}$$

Where p(x_i) is disregarded. Where c is a class parameter c $\in Yes, No$ and Yes indicates 1 for Parkinson's disease and O for none. p(x_1) is disregarded because it is the same for all posterior probabilities.

$$p(y=1|x=1) = p(x=1|y=1).p(y=1) \tag{5}$$

We were trying to determine:

$$p(y=1|x_1, x_2, ..., x_{22}) = \alpha \tag{6}$$

The above is for the 22 different features for this problem. and

$$p(y=0|x_1, x_2, ..., x_{22}) = \beta \tag{7}$$

If the ratio α / β is greater than 1 then we label the datapoint as 1 (meaning that for that configuration of features (x1,...x2) Parkisnson's disease would most likely be detected. Otherwise it won't.

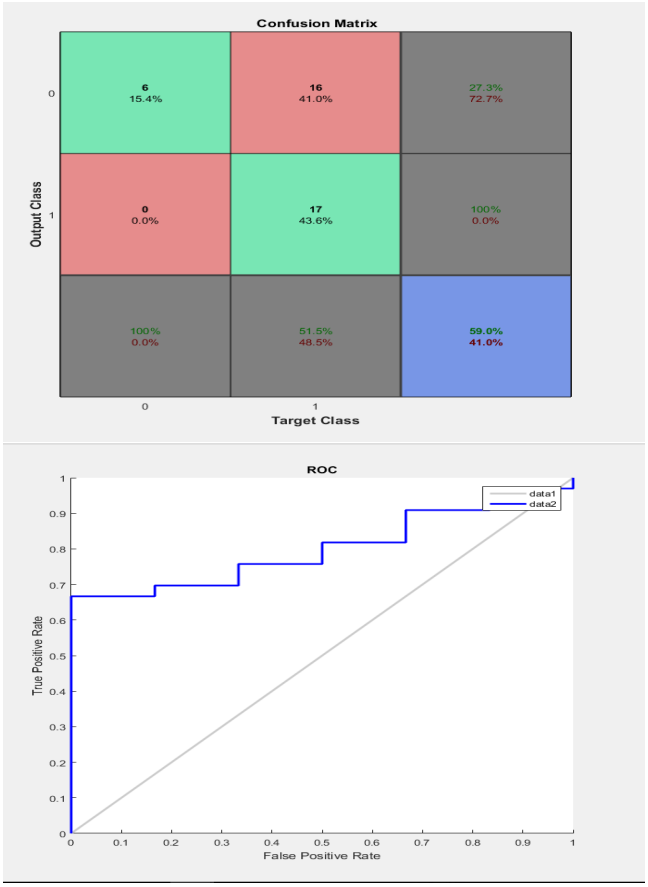

Figure 1: Results for task 1.

The above files were generated using the algorithm for confusion matrix. Regarding the quality of the data analysis, since the assumption that all the feautures are independent can prove to be troublesome in some cases this may not be a very good data set analysis algorithm(naive bayes) in this case. Since the voice measurements may be dependant on other factors as well, this assumption will lead to less accurate results. If the size of the training set is reduced, then the posterior probabilities for the test set will be least close to being accurate. The estimates for the mean and variance calculated by classifying the dataset into binary decisions(yes and no for the mean and standard deviation), it can be said that the estimates will get closer to inaccuracy as the size of the training set is reduced in this scenario.

In the second problem we have to first divide the dataset into I's and 0's according to the labels that are present (training). σ_1,σ_0 will be evaluated according to this using the CO function of matlab. After this μ_1 and μ_0 will be evaluated too for the 4 features present. The binary decision boundary equation. That was discussed in class will be used to label the test set.

$$p(w_1|x)/p(w_2|x) > 1 then chose w_1 (Both criteria have to be met) p(x|w_1)/p(x|w_2) > p(w_2)/p(w_1) \tag{8}$$

The resulting equation is:

$$x^{'}((\sigma_0^{-1})-(\sigma_1))x+2((\sigma_1^{-1})\mu_1-(\sigma_0^{-1})\mu_0)^{'}x+\mu_0^{'}\sigma_0^{-1}\mu_0-\mu_1^{'}\sum_{1}^{-1}\mu_1-log((|\sigma_1|)/(|\sigma_0|))-2log(p(w_1)/p(w_2)) > 1 \tag{9}$$

Multivariate Gaussian Distribution

p(x)= $(1/(2\pi)^{d/2}|\sigma|^{1/2})exp(-1/2(x-\mu)^{'}\sigma^{-1}(x-\mu))$

This equation was used to generate the probabilities of Yes cases and the probMatrix in the code file was given as input to the function. This generated a very accurate confusion matrix with few number of errors compared to naive bayes.