**Naïve Bayes classifier**

Introduction

I have build a naïve bayes classifier to predict whether a person has diabetes based on the diagnostic measurements in the database. I am not using scikit learn for this project.

Dataset

This dataset is originally from the National Institute of Diabetes and Digestive and Kidney Diseases. The objective of the dataset is to diagnostically predict whether a patient has diabetes, based on certain diagnostic measurements included in the dataset. Several constraints were placed on the selection of these instances from a larger database. In particular, all patients here are females at least 21 years old of Pima Indian heritage.

The list of the fields in order of columns in the data file is:

1. PregnanciesNumber of times pregnant
2. GlucosePlasma glucose concentration a 2 hours in an oral glucose tolerance test
3. BloodPressureDiastolic blood pressure (mm Hg)
4. SkinThicknessTriceps skin fold thickness (mm)
5. Insulin2-Hour serum insulin (mu U/ml)
6. BMIBody mass index (weight in kg/(height in m)^2)
7. DiabetesPedigreeFunctionDiabetes pedigree function
8. Age (years)
9. OutcomeClass variable (0 No Diabetes or 1 Diabetes) 268 of 768 are 1, the others are 0.

Key features of the classifier

A function that uses Pandas API to load files (training and test) as a Panda’s data frame

b. A function that calculates 𝜇 and σ for each column

c. A function for calculating normal distribution likelihood

d. A classifier function

e. An accuracy function to count/estimate accuracy (and hence error)

The generated output

1. An evaluation of accuracy by counting classified and misclassified points
2. An evaluation of accuracy by using confusion matrix as follows:
   1. Classification Error: Overall, how often is the classifier correct? Accuracy = (TP + TN) / (TP + FP + TN +FN)
   2. Classifier Error: Overall, how often is the classifier correct? Error = (FP + FN) / (TP + FP + TN +FN)
   3. Classifier Sensitivity: When the actual value is positive, how often is the prediction correct? Sensitivity = TP / (FN + TP)
   4. Classifier Specificity: When actual value is negative, how often is the prediction correct? Specificity = TN / (TN/FP)

Results

An accuracy of 75 % was obtained by this classifier on the given set of data.