CSE5334: Data Mining
Assignment 2

Problem 1: Naïve Bayes Classifier

What is a classifier?

A classifier is a machine learning model that is used to discriminate different objects based on certain features.

Principle of Naive Bayes Classifier:

A Naive Bayes classifier is a probabilistic machine learning model that's used for classification task. The crux of the classifier is based on the Bayes theorem.

Bayes Theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Using Bayes theorem, we can find the probability of A happening, given that B has occurred. Here, B is the evidence and A is the hypothesis. The assumption made here is that the predictors/features are independent. That is presence of one particular feature does not affect the other. Hence it is called naive.

In plain English, using Bayesian probability terminology, the above equation can be written as:

$$\textbf{posterior} = \frac{\textbf{prior} \times \textbf{likelihood}}{\textbf{evidence}}$$

- P(A) is the probability of hypothesis H being true. This is known as the prior probability.
- P(B) is the probability of the evidence (regardless of the hypothesis).
- P(B|A) is the probability of the evidence given that hypothesis is true.
- P(A|B) is the probability of the hypothesis given that the evidence is there.

In practice, there is interest only in the numerator of that fraction, because the denominator does not depend on A and the values of the features B(i) are given, so that the denominator is effectively constant.

Gaussian naive Bayes:

When dealing with continuous data, a typical assumption is that the continuous values associated with each class are distributed according to a Gaussian distribution. For example, suppose the training data contains a continuous attribute, x. We first segment the data by the class, and then compute the mean and variance of x in each class. Let μ_k be the mean of the values in x associated with class C_k, and let σ_k^2 be the variance of the values in x associated with class C_k. Suppose we have collected some observation value v. Then, the probability *distribution* of v given a class C_k, $p(x = v \mid C_k)$, can be computed by plugging v into the equation for a Normal distribution parameterized by μ_k and σ_k^2. That is,

$$p(x = v \mid C_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(v-\mu_k)^2}{2\sigma_k^2}}$$

CSE5334: Data Mining
Assignment 2

Problem 1.1 Implementation:

Step 1: Generated two sets of 2-D Gaussian random data, each set containing 500 samples and merged the data into a single dataset. Generated testing dataset of 1000 samples with the same method.

Step 2: Generated training class labels of 1000 samples with equal probability of 1 and 0. Generated testing class labels of 1000 samples with the same method.

Step 3: Classified data based on the classification labels and counted mean and std (standard deviation) of each attribute for each classification labels.

Step 4: This mean and std data is used to find the conditional probability P(B|A) of each attributes of test dataset samples for each classification labels. Prior probabilities P(A) is calculated for each classification labels.

Step 5: Now as in the case of Gaussian naïve Bayes, the denominator P(B) is not required as it is constant.

Posterior Probability P(A|B) = P(B|A) of each attributes * Prior probabilities P(A)

This way, probability of Label 0 & Label 1 is found for each attribute and compared. The label with the maximum probability is chosen as predicted one.

Step 6: Calculated TP (True Positive), TN (True Negative), FP (False Positive), FN (False Negative).

Plotted the confusion matrix & Calculated other metrics as below:

		actual class: positive	actual class: negative
predicted class	positive	true positives (TP)	false positives (FP)
predicted class	negative	false negatives (FN)	true negatives (TN)

$$\text{true positive rate (recall)} = \frac{TP}{\text{actual pos}} = \frac{TP}{TP+FN}$$

$$\text{false positive rate} = \frac{FP}{\text{actual neg}} = \frac{FP}{TN+FP}$$

$$\text{accuracy} = \frac{TP+TN}{TP+FP+FN+TN}$$

$$\text{recall (TP rate)} = \frac{TP}{\text{actual pos}} = \frac{TP}{TP+FN}$$

$$\text{error} = 1-\text{accuracy} = \frac{FP+FN}{TP+FP+FN+TN}$$

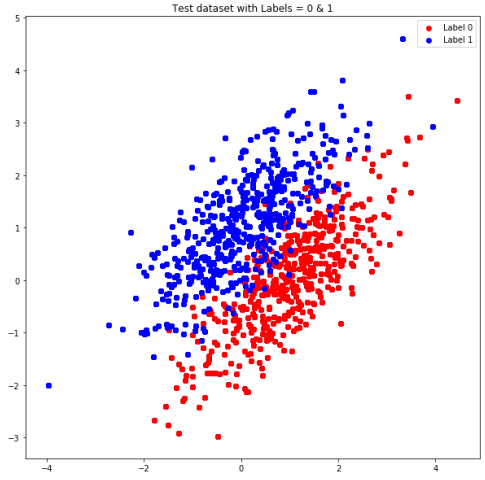
$$\text{precision (positive predictive value)} = \frac{TP}{\text{predicted pos}} = \frac{TP}{TP+FP}$$

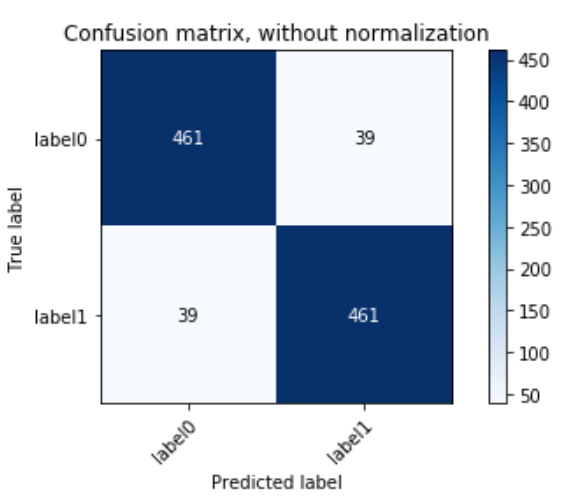
CSE5334: Data Mining
Assignment 2

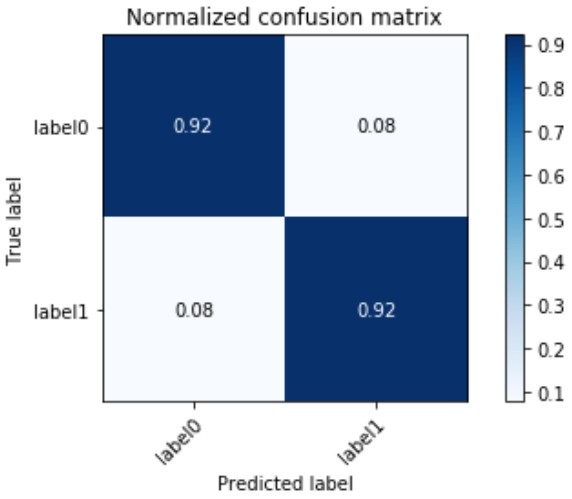
<u>Problem 1.2</u> Perform prediction on the testing data with your code. In your report, report the accuracy, precision and recall as well as a confusion matrix. Also, make sure to include a scatter plot of data points whose labels are color coded (i.e., the samples in the same class should have the same color) in the report.

Prediction of Label 0 and Label 1 for Test dataset of 1000 samples based on training of model with 1000 train dataset samples.

```
TN: 461, FP: 39, FN: 39, TP: 461

Accuracy:  0.922

Precision:  0.922

Recall:  0.922

Error Rate : 7.8%
```







CSE5334: Data Mining
Assignment 2

<u>Problem 1.3</u> In your training data, change the number of examples in each class to (10, 20, 50, 100, 300, 500) and perform prediction on the testing data with your code. In your report, show a plot of changes of accuracies w.r.t. the number of examples and write your brief observation.

```
Number of Training Data Points: 20
Number of Label 0: 10
Number of Label 1: 10
TN: 461, FP: 39, FN: 181, TP: 319
Accuracy:  0.78
```

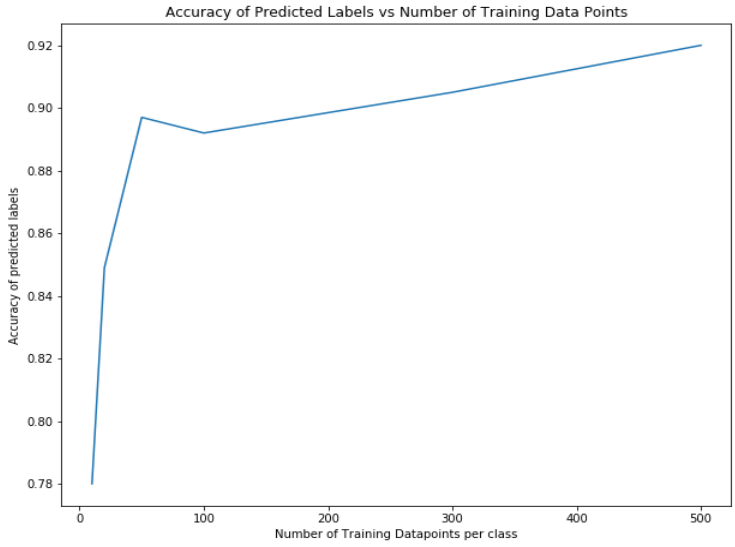
```
Number of Training Data Points: 40
Number of Label 0: 20
Number of Label 1: 20
TN: 429, FP: 71, FN: 80, TP: 420
Accuracy:  0.849
```

```
Number of Training Data Points: 100
Number of Label 0: 50
Number of Label 1: 50
TN: 465, FP: 35, FN: 68, TP: 432
Accuracy:  0.897
```

```
Number of Training Data Points: 200
Number of Label 0: 100
Number of Label 1: 100
TN: 451, FP: 49, FN: 59, TP: 441
Accuracy:  0.892
```

```
Number of Training Data Points: 600
Number of Label 0: 300
Number of Label 1: 300
TN: 440, FP: 60, FN: 35, TP: 465
Accuracy:  0.905
```

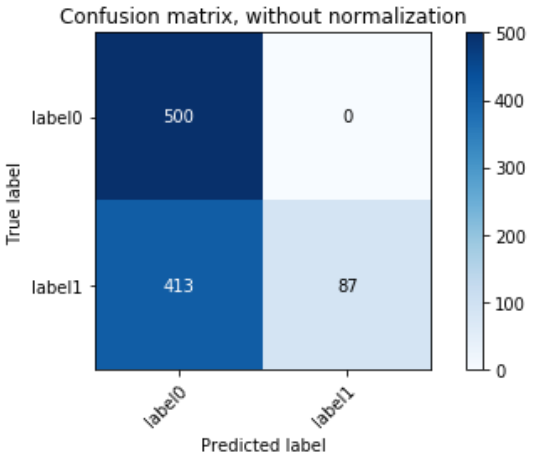
```
Number of Training Data Points: 1000
Number of Label 0: 500
Number of Label 1: 500
TN: 459, FP: 41, FN: 39, TP: 461
Accuracy:  0.92
```



CSE5334: Data Mining
Assignment 2

<u>Problem 1.4</u> Now, in your training data, change the number of examples in class 0 as 700 and the other as 300. Perform prediction on the testing dataset. How does the accuracy change? Why is it changing? Write your own observation.

```
Number of Training Data Points: 1000

Number of Label 0: 700

Number of Label 1: 300

TN: 500, FP: 0, FN: 413, TP: 87

Accuracy:  0.587
```



Observation:

Change in accuracy:

- Accuracy dropped to 58.7% from 92%

Reason for Change:

- Trained the model with almost 2.5 times label 0 than label 1.
- Because of overfitting, model is predicting label 1 as label 0 most of the times.

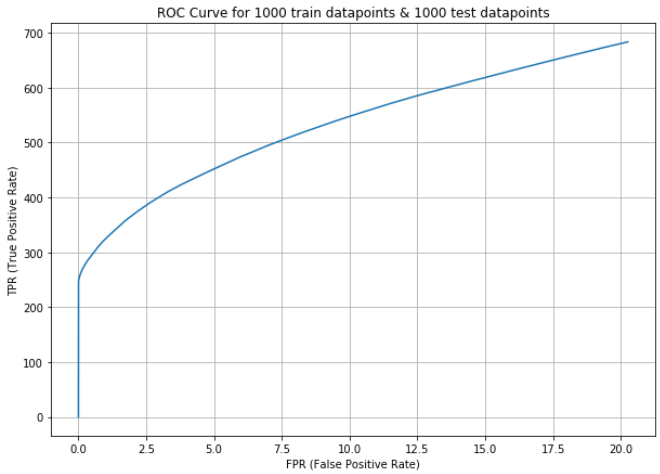
Result:

- Increase in FN (False Negatives) due to model predicting label 1 as label 0.
- Decrease in FP (False Positives) due to model predicting label 0 accurately.

CSE5334: Data Mining
Assignment 2

Problem 1.5 Write a code to plot a ROC curve and calculate Area Under the Curve (AUC) based on the posterior for class 1 (i.e., the confidence measure for class 1 is the posterior). The implementation should be done on your own without using explicit library that lets you draw the curve. Report the ROC curves from the two cases discussed in P1-2 and P1-4 above (i.e., one with equal distribution of classes and unequal distributions in the training data).

ROC Curve for equal distribution of classes:



Here, ROC Curve is not scaled to 1 for both the axis, so to scale it, the result points need to be divided by total number of datapoint (1000 in our case).

i.e.: **TP: 600 → FP: 14** || **TPR**: 600/1000: **0.6** → **FPR**: 14/1000: **0.014**

ROC Curve for equal distribution of classes: