Pattern Recognition and Machine Learning

Lab - 4 Assignment Report

Aryan Himmatlal Prajapati (B21EE012)

Question 1.

Pre-Processing of Dataset:

1. Checked for NULL values: No null values found.
2. Separated Features and Labels
3. Encoded Categorical Label Data:
    a. Iris-setosa = 0
    b. Iris-versicolor = 1
    c. Iris-virginica = 2
4. Splitted dataset into training dataset and testing dataset into ratio 70:30(Train:Test).

Part 1

Implemented Gaussian Bayes Classifier from Scratch. The classifier class have 3 variants defined using its constructor:

1. Case $\Sigma_i = \sigma^2.I$ (I stands for the identity matrix)
2. Case $\Sigma_i = \Sigma$ (covariance of all classes are identical but arbitrary!)
3. Case Σi = actual covariance

Bayes Theorem:

Bayes Theorem can be used to calculate conditional probability.

The Formula For Bayes' Theorem Is

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

where:

$P(A) =$ The probability of A occurring

$P(B) =$ The probability of B occurring

$P(A|B) =$ The probability of A given B

$P(B|A) =$ The probability of B given A

$P\left(A \bigcap B\right)) =$ The probability of both A and B occurring

Gaussian Naive Bayes:

When working with continuous data, an assumption often taken is that the continuous values associated with each class are distributed according to a normal (or Gaussian) distribution. The likelihood of the features is assumed to be-

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

Sometimes assume variance

- is independent of Y (i.e., σi),
- or independent of Xi (i.e., σk)
- or both (i.e., σ)

Case 1:

In this case we assume that the $Cov(X, X) = \sigma^2$ and $Cov(X, Y) = 0$. so we can take $\sigma_y^2 = \sigma^2$.

Case 2:

In this case we assume that the $Cov(X, X) = Cov(X, Y) = \sigma^2$. so we can take $\sigma_y^2 = \sigma^2$.

Case 3:

In this case $Cov(X, X) \neq Cov(Y, Y) \neq Cov(X, Y) \neq Cov(Y, X)$. So $\sigma_y^2 = Covariance\ Matrix$.

Part 2

Implemented the Following Functions in Gaussian Bayes Classifier:

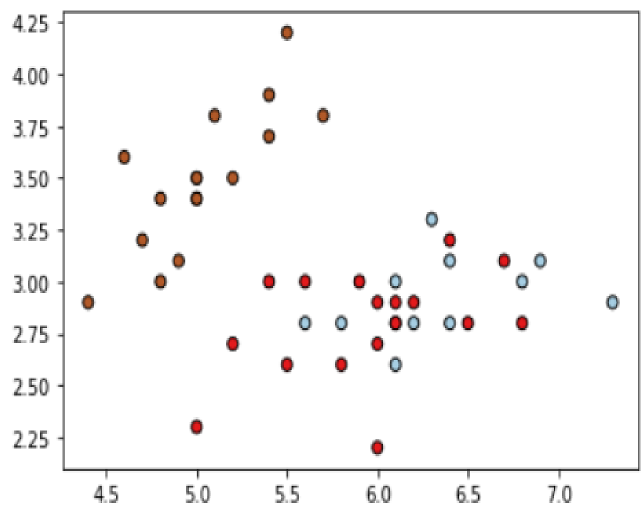
a. Train: Takes x,y (training data) as input and trains the model.
b. Test: Takes testing data, testing labels as input, and outputs the predictions for every instance in the testing data, and also the accuracy.
c. Predict: Takes a single data point as input, and outputs the predicted class.
*d. Plotdecision boundary: Takes input the training data points, and their labels, and plots the decision boundary of the model with the data points superimposed on it.

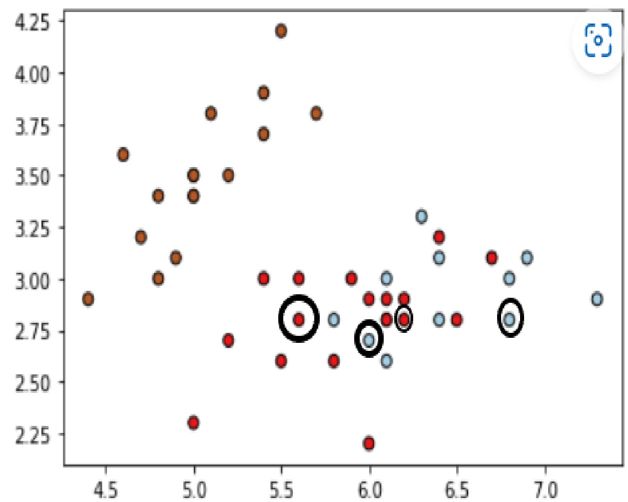
Part 3

Case 1:

Accuracy Score: 0.9111111111111111

Original Distribution vs Predicted Distribution(black circles are wrongly classified points in prediction):





Case 2:

Same as Case 1.

Case 3:

Accuracy Score: 1.0

In case 3 Actual and Predicted distribution is fully same as Accuracy Score is 1(100% Accurate).

Part 4

5 Fold Cross Validation

Case 1:

Reported Accuracies: 1.0 , 0.93 , 0.97 , 0.93 , 1.0

Average Accuracy: 0.966

Case 2:

Reported Accuracies: 0.97 , 0.97 , 1.0 , 0.93 , 0.97

Average Accuracy: 0.97

Case 3:

Reported Accuracies: 1.0 , 0.93 , 0.97 , 1.0 , 1.0

Average Accuracy: 0.98

Generalizability:

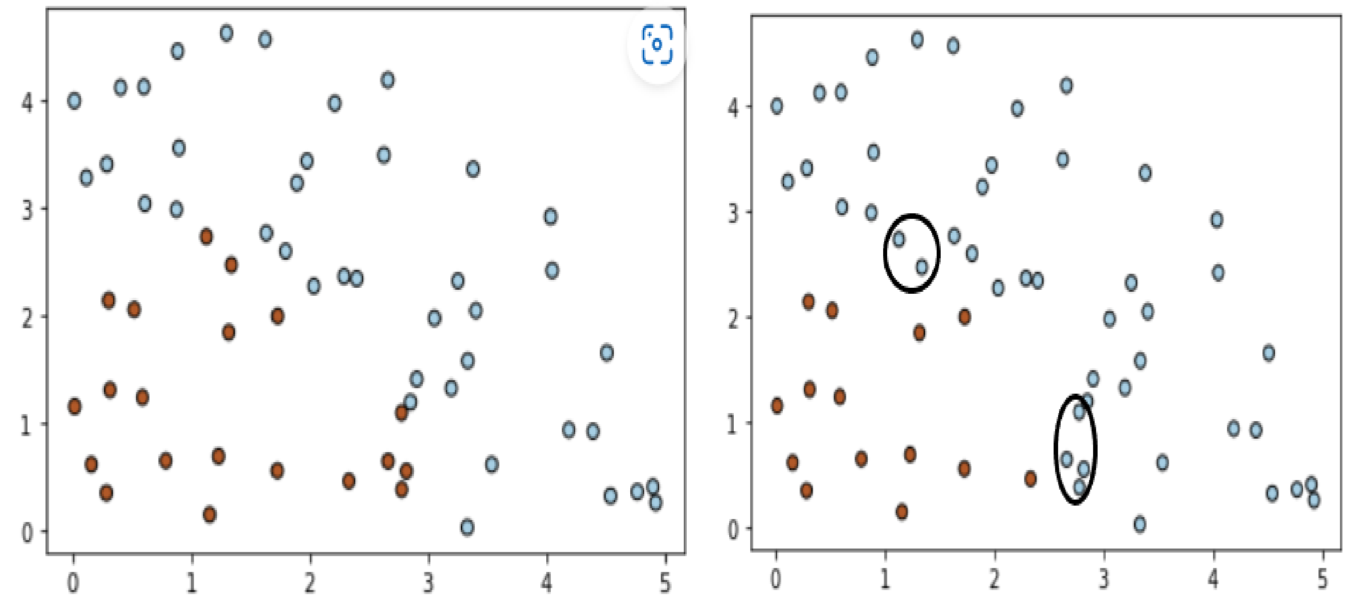
As we can see from obtained reported accuracies in different cases. We observe that variation across reported accuracies is least for case 3 and also Average Accuracy is highest in case 3. So case 3 is more generalizable than case 1 and case 2 and It is quite obvious because we aren't taking any assumption in Case 3 apart from the assumption that features are independent. So due to fewer assumptions the model is trained better while in other cases there are more assumptions.

Part 5

Gaussian Bayes Classifier (Case 3):

Accuracy Score : 0.9

Plot for Original Distribution vs Predicted Distribution(Black Circle are wrongly predicted points):



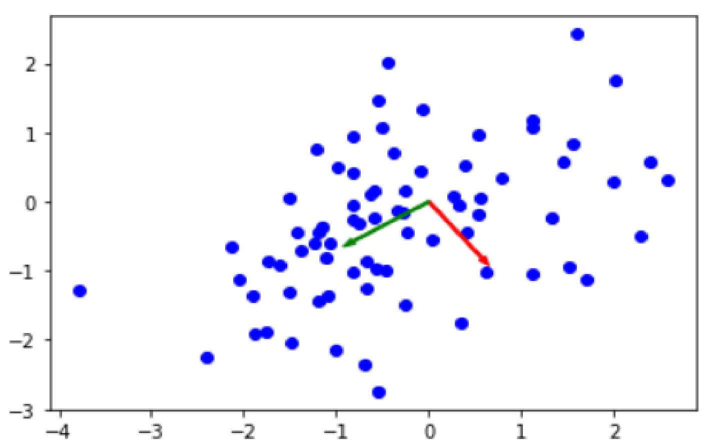
Question 2.

Part 1

Calculate the covariance matrix of the sample X(Σ_s). Found the eigenvectors and eigenvalues of Σ_s and plotted it superimposed on the data points X.

Covariance Matrix : [[1.56134944 0.60381577]
[0.60381577 1.12455011]]

Plot:



Part 2

Performed Transformation $Y = \Sigma_s^{-1/2} X$ on data points X. Calculate the covariance matrix of transformed data points Y.

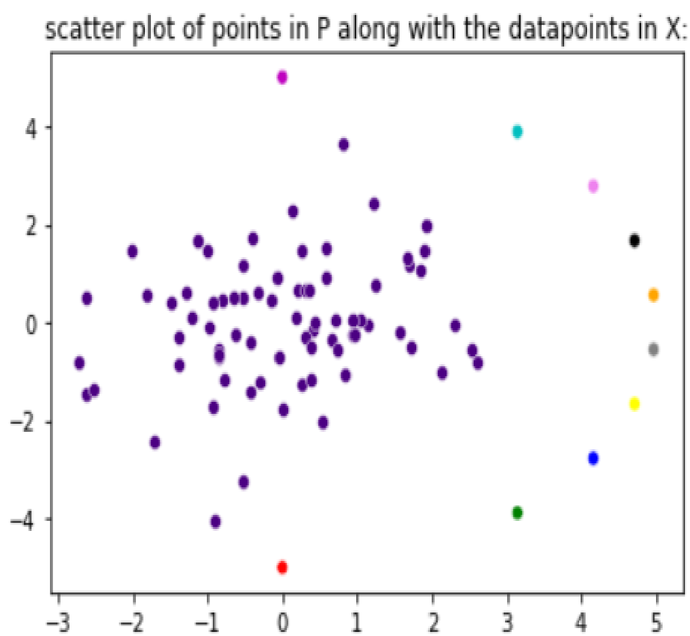
Obtained Covariance Matrix: [[1.00000000e+00 9.90198914e-17]
[9.90198914e-17 1.00000000e+00]]

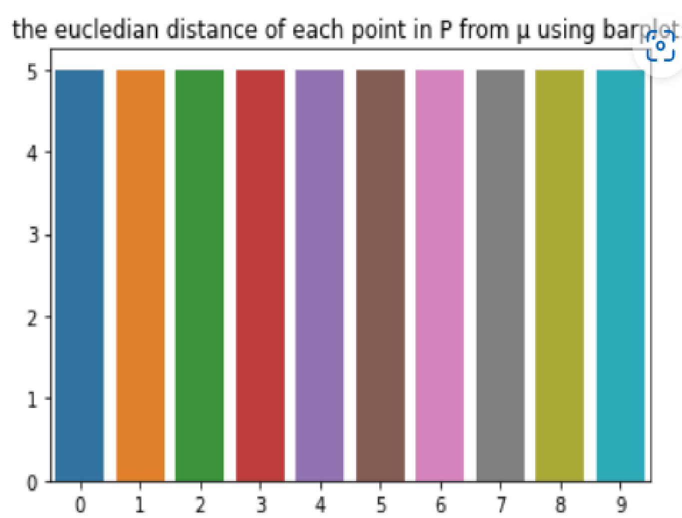
The obtained covariance matrix is close to the identity matrix, indicating that the transformed data points are uncorrelated. The purpose of the transformation Y = Σ^(-1/2) X is to decorrelate the variables and make them have unit variance. This is useful in various statistical and machine learning applications where it is beneficial to work with uncorrelated and normalized variables.

Part 3

Uniformly sampled 10 points on the curve x^2 + y^2 = 25 . Let these sets of points be called P . Plotted points in P along with the data points in X. Report the euclidean distance of each point from μ using barplot.

Plots:

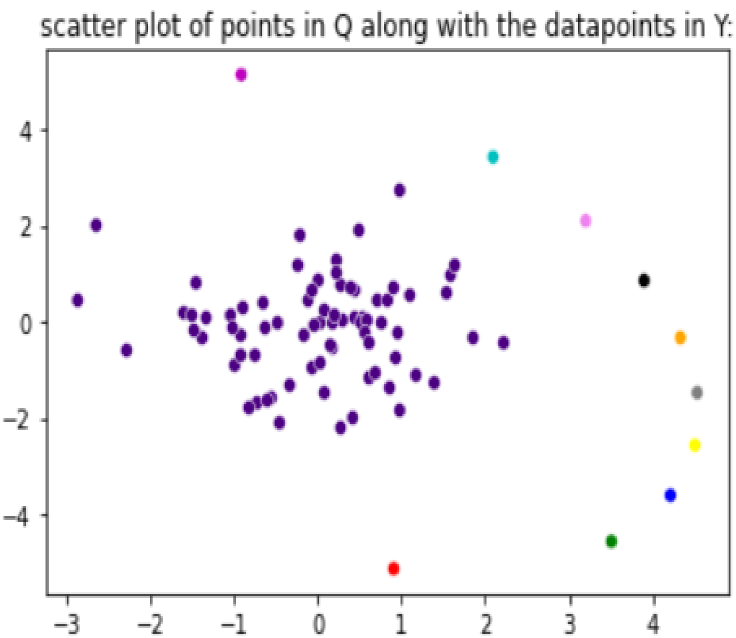


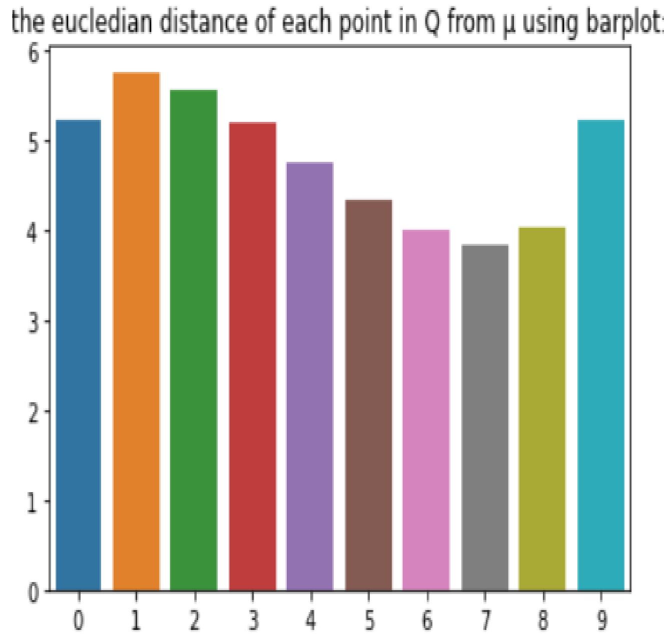


Part 4

Performed the transformation $Q = \Sigma_s^{-1/2} P$ on the data points P. Calculated the euclidean distance of transformed data points Q from μ and reported it using barplot . Plotted points in Q along with data points in Y.

Obtained Plots:





The Euclidean before Transformation was 5 for each point. But After Transformation it is changed for each point and data points are more spreaded like an oval shape which were earlier in circular shape because after transformation The obtained covariance matrix is close to the identity matrix, indicating that the transformed data points are uncorrelated. The purpose of transformation was to decorrelate the variables and make them have unit variance.