ML Assignment - 2
Nitesh Jaiswal
2018400
CSB

Q-1

(a) Read about PCA and write a note on how it works.

Ans - PCA (Principal Component Analysis) is a dimensionality reducing method which can reduce dimensions of large datasets while storing most of the information of large dataset. PCA used SVD (Singular Value Decomposition) of the data to project it into lower dimensions. The input data is centered but not scaled for each feature before applying the SVD.

Working-

1)Standardization

2)Covariance Matrix computation

3)Compute the eigenvectors and eigenvalues of the covariance matrix to identify the principal components

4)constructs the Principal Components

5) Creating feature vector

6)Recast the data along the principal components axes

(b) Read about SVD and write a note on how it works.

Ans - SVD (Singular Value Decomposition) is a matrix factorization technique where a matrix is decomposed into a product of a square matrix, a diagonal (possible rectangular) matrix, and another square matrix. It can be used for dimension reduction.

$A=U*S*V^T$

Where:

A is an $m \times n$ matrix

U is an $m \times n$ *orthogonal* matrix

S is an $n \times n$ *diagonal matrix*

V is an $n \times n$ orthogonal matrix

Unlike PCA It does not do any centralization of data.

(c) Read about t-SNE and write a note on how it works.
Ans - t-SNE(t- Distributed Stochastic Neighbour Embedding) is a non linear method that is used for data exploration and visualization of high dimensional data.
Working:

1) Measure the similarity between two points in high dimensional space using Gaussian Distribution
2) Plot the matrix of similarity scores.
3) We randomly plot all the points on less dimensional space and then calculate similarity score using t-distribution and it return set of low dimensional space.
4) And then we will try to convert low dimensional space set to the high dimensional space set.

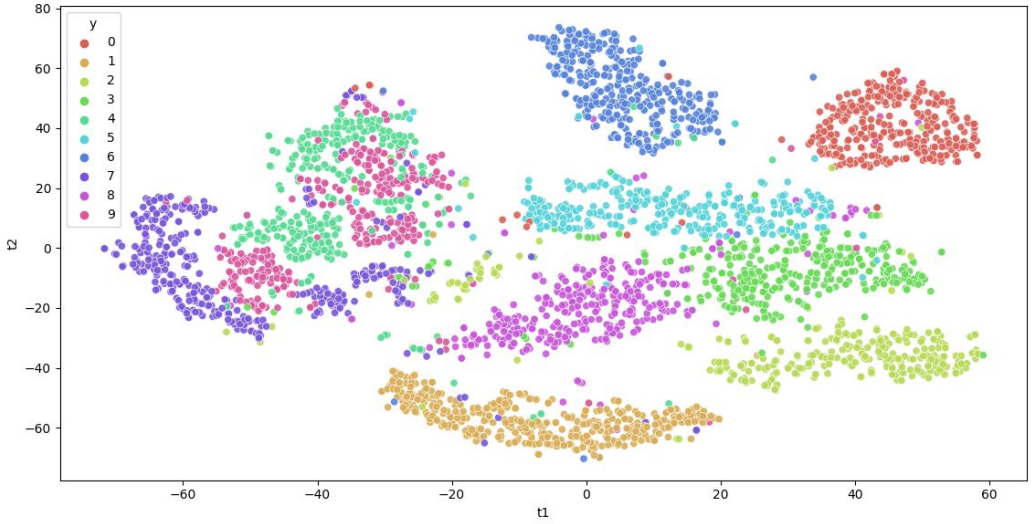
(d) Read about stratified sampling and perform an 80:20 stratified train-test split on the dataset. Comment on the class frequency of training and testing samples.
Ans - In Stratified sampling, firstly we divide the data on the basis of classes and then we create samples by drawn data from each separate class groups.

Frequency distribution:

```
PS D:\ML_A2> python -u "d:\ML_A2\q1.py"
Frequency of class: 0 1 2 3 4 5 6 7 8 9
Training Set: 320 395 314 339 333 319 353 345 328 314
Testing Set: 80 99 79 85 83 79 88 86 82 79
```

Frequency of training and testing samples is not biased towards one or two classes and whole data is equally distributed to all classes.

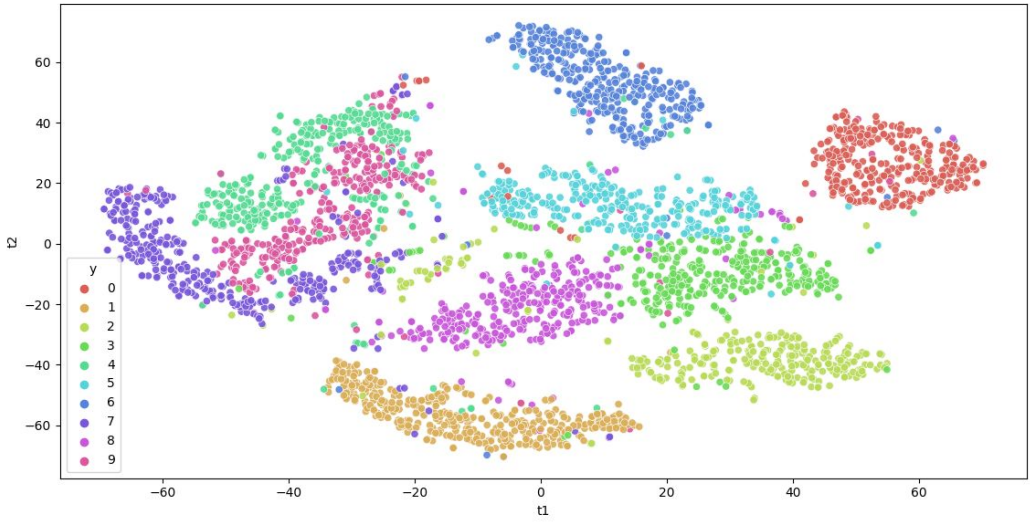
(e) Use PCA on the dataset provided. Train a Logistic Regression model on the training set and report the test accuracy. Further, use t-SNE to analyze the training data.
Ans - Cluster formed for different classes. Some cluster are significantly maintain distance with others like 6,0 etc. and some clusters have merged with others like 4,9,etc



(f) Use SVD on the dataset provided. Train a Logistic Regression model on the training set and report the test accuracy. Further, use t-SNE to analyze the training data.

Ans -

Cluster formed for different classes. Some cluster are significantly maintain distance with others like 6,7,0 etc. and some clusters have merged with others like 4,9,etc



(g) Compare the accuracy obtained while using PCA & SVD and write a note on the results obtained.

Ans -

```
Accuracy after applying PCA: 0.8738095238095238
Accuracy after applying SVD: 0.8702380952380953
PS D:\ML_A2>
```

PCA and SVD mostly gives same accuracy (approximately) but sometime PCA perform better because PCA centralise the date before applying SVD.

Q2-

(a) Use dataset C and train a Linear Regression model to predict weight based on the height of the person. Using bootstrapping, measure the bias & variance of the model and report them.

Ans -

```
PS D:\ML_A2> python -u "d:\ML_A2\q2.py"
Mse: 145.04050084992997
Bias: -0.310011528772663
Variance: 0.046537651785195575
(Noise)^2: 144.8978560501728
PS D:\ML_A2>
```

(b) Assuming noise in the data to be zero, report the value of: MSE − Bias^2 − Variance and give a comment on the value obtained.

Ans - (Noise)^2 = MSE − Bias^2 − Variance = 144.8978560501728

If we assume that the value of noise in data is zero than MSE-Bias^2-Variance should also be equal to zero but it is greater than zero so there is noise present in data.

Q-3

(a) Find optimal depth as a parameter in-case of DT using Grid Search and use K-Fold cross validation to validate it. Implement your own K-Fold cross validation function from scratch and use for both GNB and DT. Make these functions in such a way so that these can be used in future assignments.

Ans -

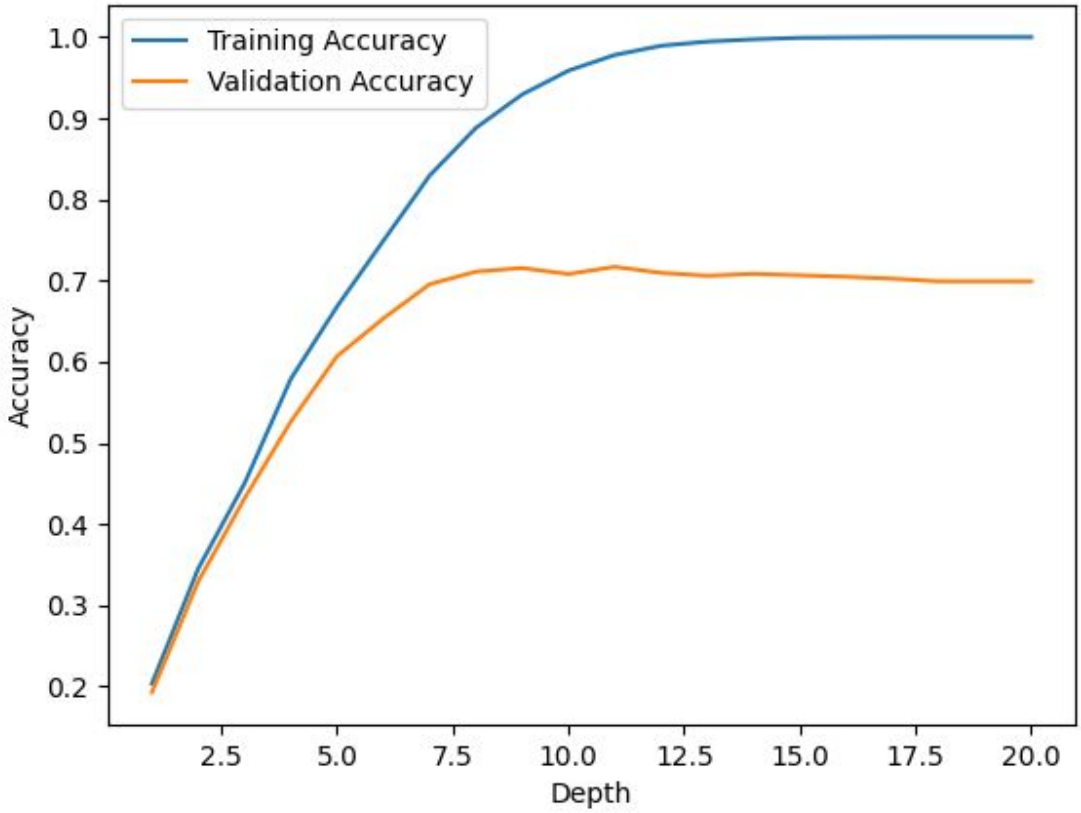
```
Dataset-1 Results
depth: 11
accuracy: 0.7169642857142857
Decision Tree Accuracy: 0.7169642857142857
GaussianNB Accuracy: 0.6122023809523809
```

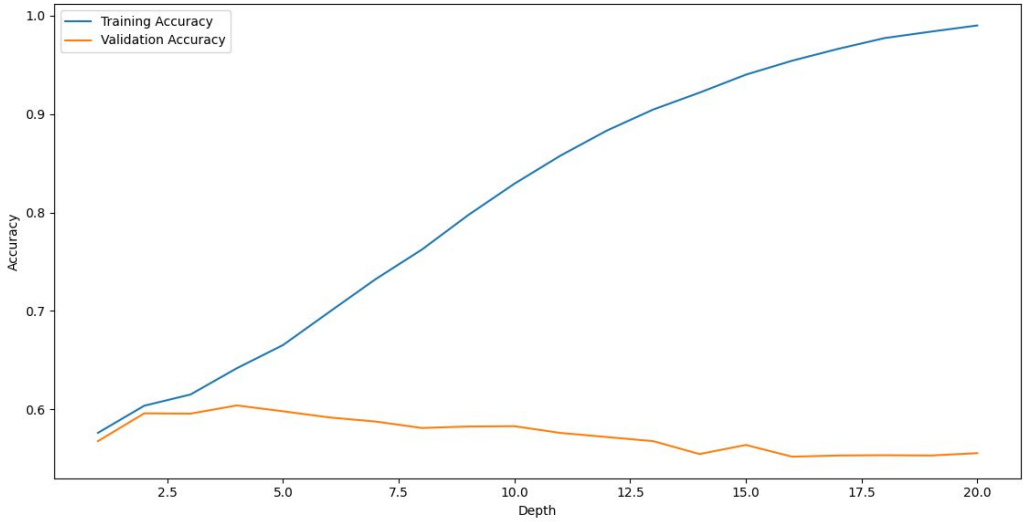
```
Dataset-2 Results
depth: 4
accuracy: 0.6023809523809524
Decision Tree Accuracy: 0.6023809523809524
GaussianNB Accuracy: 0.575595238095238
```

(b) For DT plot training and validation accuracy plot with respect to tree depth and write your analysis.

Ans- After a certain depth value accuracy does not change drastically it becomes stable for further depth and testing set accuracy decreasing if we increasing depth due to overfitting.

Dataset-1 and Dataset-2 Plots





(d) Write a function evaluation metric to evaluate testing data. Function should calculate accuracy, precision, recall, F1-Score, plot ROC-curve and return the confusion matrix. In case of multi-class data it should return Macro and Micro Average values. Read Macro and Micro average values in Multi-class data.

Ans -

Dataset-1

```
Evaluation Matrix Results
Confusion Matrix: [[60, 0, 4, 2, 1, 5, 2, 2, 2, 0], [1, 86, 3, 1, 3, 1, 0, 1, 2, 1], [3, 2, 46, 1, 0, 3, 4, 3, 5, 1], [3, 0, 7, 50, 0, 8, 0, 0, 7, 5], [1, 3,
6, 2, 68, 2, 5, 2, 5, 7], [1, 1, 4, 5, 3, 61, 3, 2, 4, 1], [0, 0, 2, 3, 1, 6, 64, 0, 2, 0], [0, 0, 3, 1, 0, 0, 1, 76, 1, 4], [2, 1, 6, 6, 4, 5, 3, 5, 53, 5],
[1, 1, 3, 4, 5, 1, 2, 1, 4, 53]]
Accuracy: 73.45238095238096
Macro_precision: 0.7330091918010069
Micro_precision: 0.7345238095238096
Macro_recall: 0.7325388961404088
Micro_recall: 0.7345238095238096
Macro_F1_Score: 0.7309203238310669
Micro_F1_Score: 0.7345238095238096
```

Dataset-2

```
Evaluation Matrix Results
Confusion Matrix: [[225, 142], [200, 273]]
Accuracy: 59.285714285714285
Macro_precision: 0.5951230190505268
Micro_precision: 0.5928571428571429
Macro_recall: 0.5936215450035436
Micro_recall: 0.5928571428571429
Macro_F1_Score: 0.5915233415233414
Micro_F1_Score: 0.5928571428571429
```

ROC Plots for Dataset-1 and Dataset-2