DryBeans Data Set Description :

Question 1.1

What the data is about?

The Dry Beans Data Set is a collection of data based on 13,611 images of beans taken with a high resolution camera of seven different types of dry beans. The dataset contains various physical characteristics of the beans such as size, shape, and color. The dataset includes 16 different attributes, such as area, perimeter, compactness, length of kernel, width of kernel, asymmetry coefficient, etc. and one class attribute.

The bean images obtained by computer vision were subjected to feature extraction to obtain a total of 17 attributes.

The format of the data attributes is as follows:

1.) Area (A): The area of a bean zone and the number of pixels within its boundaries.
2.) Perimeter (P): Bean circumference is defined as the length of its border.
3.) Major axis length (L): The distance between the ends of the longest line that can be drawn from a bean.
4.) Minor axis length (l): The longest line that can be drawn from the bean while standing perpendicular to the main axis.
5.) Aspect ratio (K): Defines the relationship between L and l.
6.) Eccentricity (Ec): Eccentricity of the ellipse having the same moments as the region.
7.) Convex area (C): Number of pixels in the smallest convex polygon that can contain the area of a bean seed.
8.) Equivalent diameter (Ed): The diameter of a circle having the same area as a bean seed area.
9.) Extent (Ex): The ratio of the pixels in the bounding box to the bean area.
10.)Solidity (S): Also known as convexity. The ratio of the pixels in the convex shell to those found in beans.
11.)Roundness (R): Calculated with the following formula: (4piA)/(P^2)
12.)Compactness (CO): Measures the roundness of an object: Ed/L
13.)ShapeFactor1 (SF1)
14.)ShapeFactor2 (SF2)
15.)ShapeFactor3 (SF3)
16.)ShapeFactor4 (SF4)
17.)Class (Seker, Barbunya, Bombay, Cali, Dermosan, Horoz and Sira)

2. What type of benefit you might hope to get from data mining?

Mining the Dry Beans Data Set can offer several benefits, including:

Improved Understanding of Dry Beans: The data set can help researchers gain a better understanding of the properties and characteristics of dry beans, such as their physical and chemical composition and nutritional value

There is also a wide range of genetic diversity in dry beans which is the most produced one among the edible legume crops in the world. Seed quality is an influential factor in crop production. Therefore, seed classification can help in both marketing and production to provide the principles of sustainable agricultural systems. Using data mining we can obtain a method for obtaining uniform seed varieties from crop produce, so the seeds are not certified as a sole variety.

3. Discuss data quality issues : For each attribute

Are there problems with the data?

Unstructured Data : If data is not entered correctly into the system or it is present in different formats for different samples, it can make data mining difficult. However in our data set all samples are structured and present in the same format.

Duplicate Data : duplicate copies of the same data samples can increase data processing time and produce skewed results. The wifi dataset does not contain duplicate samples.

Missing Data : The geometrical data present in the dataset carry no information about the bean colour. Practically, this is a missing insight, since different dry bean species tend to have different colours. However, since we are only using the dataset as a part of an exercise to create machine learning models the absence of certain attributes makes little difference since we are using the classification exercise as a computer training model.

Outliers : Outliers are samples that have attributes that are significantly different from the attributes of other samples in terms of deviation. The wifi data set contains outliers.

Appropriate response to quality issues :

Dealing with outliers : We are not going to remove any outliers since outliers with geometric attributes that have a wide deviation from usual attribute values might represent a class of zero.

We will normalize the geometric attributes using Min-Max scaling so that the values lie in the range between 0 and 1 for efficient performance and accuracy of the various model implementations.

Since the class attribute is the seven different types of beans, (Seker, Barbunya, Bombay, Cali, Dermosan, Horoz and Sira) we are providing a dictionary mapping to convert them into numerical attributes.

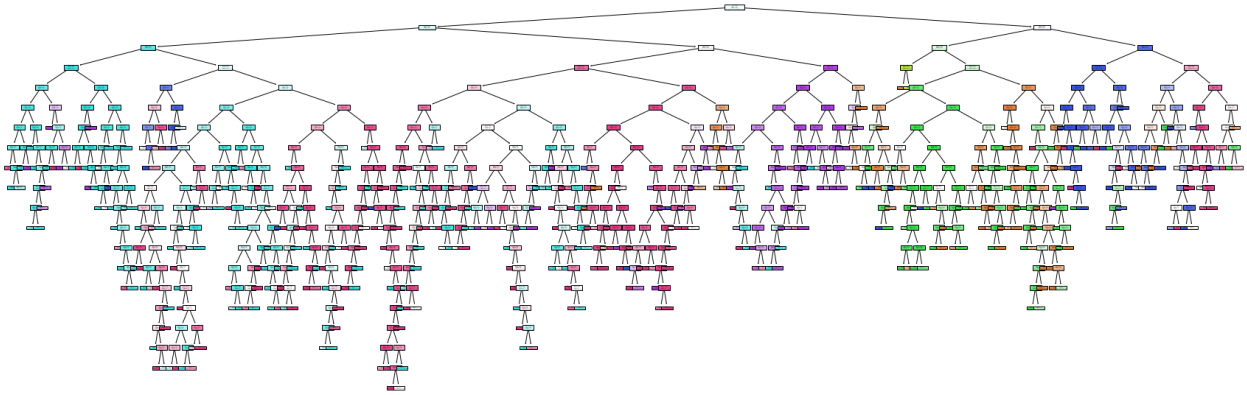
Accuracy and Precision of Different Models

DECISION TREE

Accuracy without tuning = 0.8933294

Tuned Model Accuracy on training set after applying Grid Search = 97.58

Accuracy of Tuned Model on Test Set = 90.68

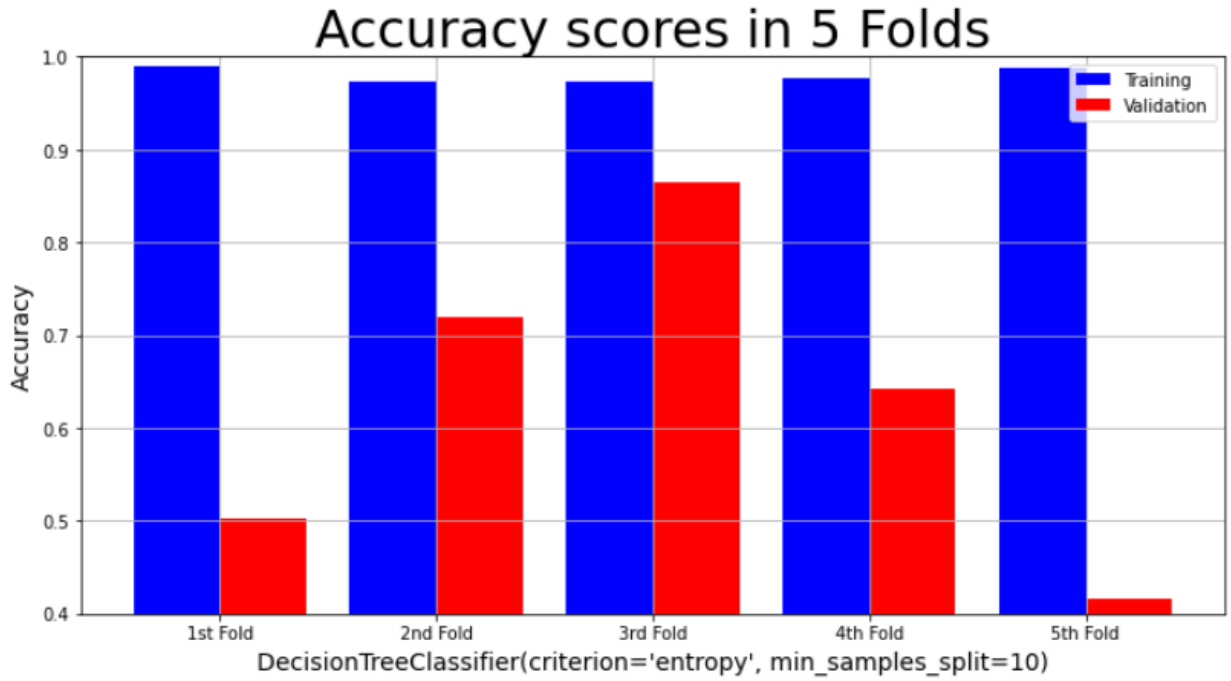


Splits created by Decision Tree

Randomized Search for Hyperparameter Tuning :

Best parameters found : {'criterion': 'entropy', 'max_depth': 5, 'max_features': 8, 'min_samples_split': 2}

Plot of Accuracy of tuned model while doing 5 fold validation on each fold



Confusion Matrix for Decision Tree

```
for Decision Tree, Confusion Matrix:
 [[283   0  20   0   3   9   4]
 [  1 110   0   0   0   0   0]
 [ 26   0 381   0   8   1   3]
 [  0   0   0 826   2  15  62]
 [  1   0  14   4 454   0  18]
 [  2   0   0   8   0 446  14]
 [  2   0   2  81  13   5 585]]
for Decision Tree, Classification Report
```

RANDOM FOREST

Randomized Search for Hyperparameter Tuning :

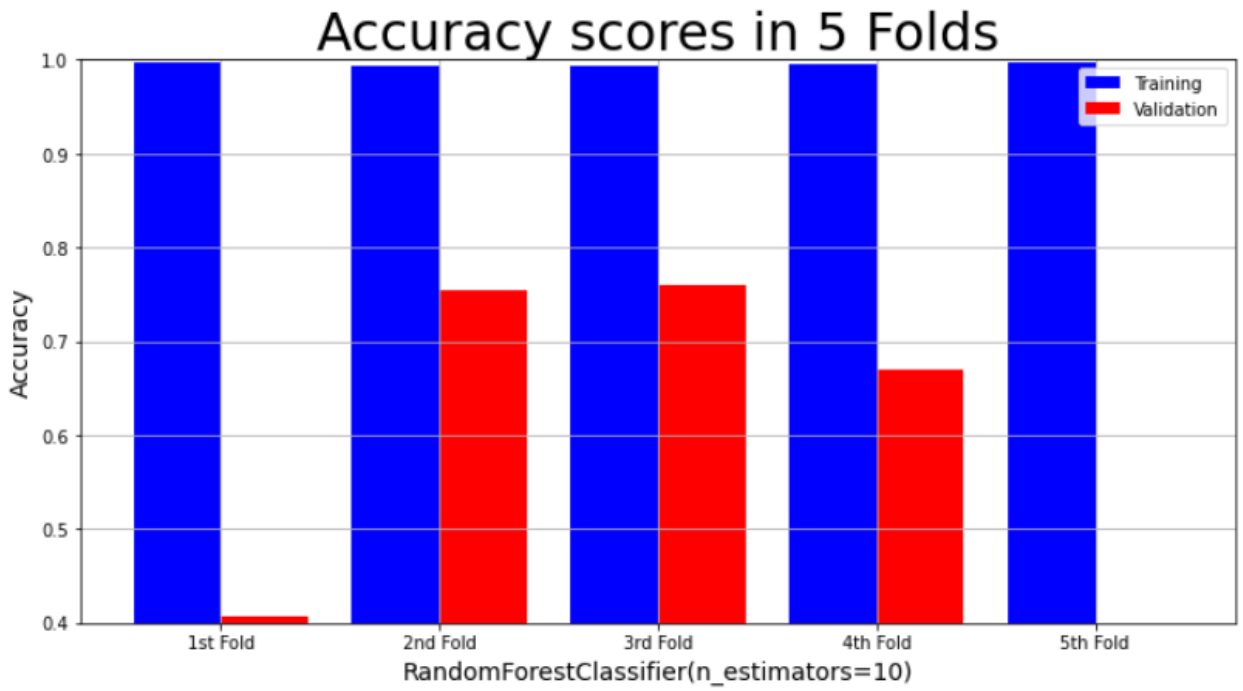
Best parameters found : {'bootstrap': True, 'criterion': 'gini', 'max_depth': 5, 'max_features': 1, 'min_samples_split': 6}

Grid Search for Hyperparameter Tuning :

Tuned model Accuracy on Training Set: 0.920160934109197

Plot of Accuracy of tuned model while doing 5 fold validation on each fold

Plot of Accuracy of tuned model while doing 5 fold validation on each fold

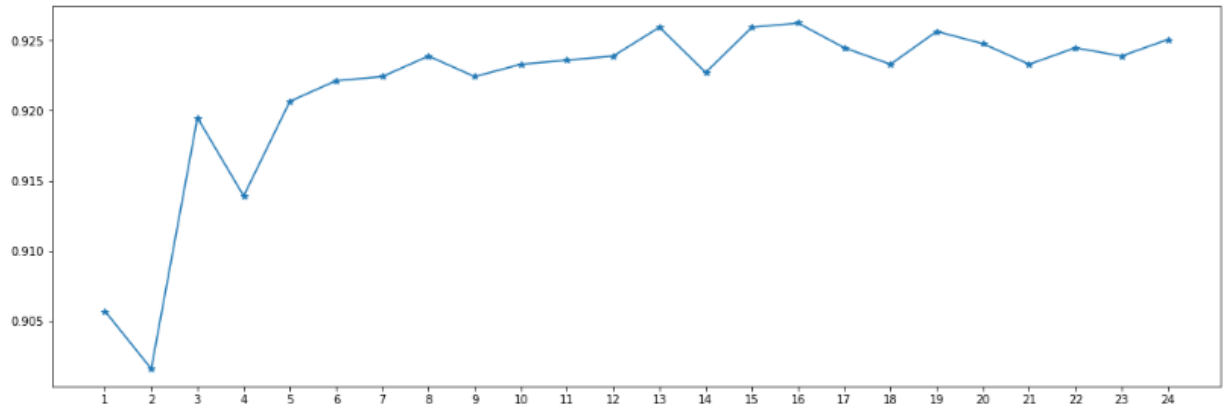


Confusion Matrix for Decision Tree

```
for Random Forest, Confusion Matrix:
 [[283   0  20   0   3   9   4]
 [  1 110   0   0   0   0   0]
 [ 26   0 381   0   8   1   3]
 [  0   0   0 826   2  15  62]
 [  1   0  14   4 454   0  18]
 [  2   0   0   8   0 446  14]
 [  2   0   2  81  13   5 585]]
for Random Forest, Classification Report
```

KNN

Plot of accuracy on different values of k :

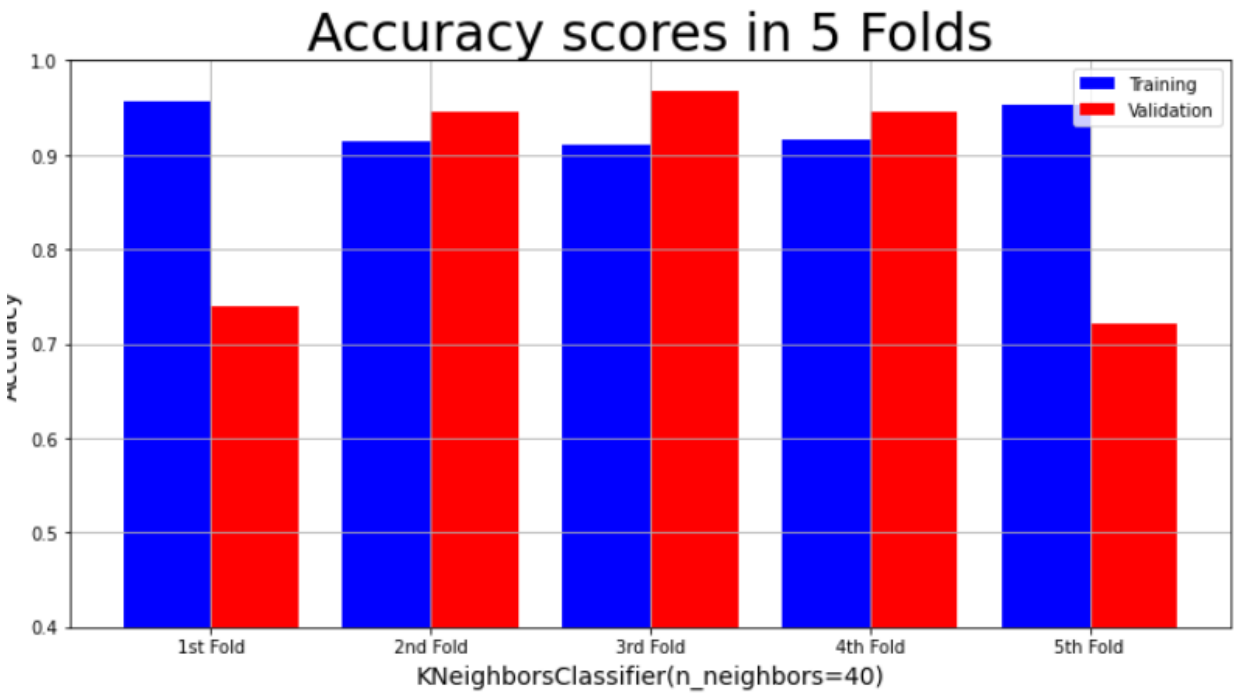


Best value of k obtained through randomized search :

Accuracy = 0.9218259683893144

Value of k = 20

Plot of Accuracy of tuned model while doing 5 fold validation on each fold



Confusion Matrix for KNN Model

```
for KNN, Confusion Matrix:
 [[274   0  23   0   1   4  17]
 [  0 111   0   0   0   0   0]
 [  8   0 401   0   4   1   5]
 [  0   0   0 849   1  15  40]
 [  0   0  12   4 464   0  11]
 [  0   0   0   2   0 445  23]
 [  1   0   1  80   4   6 596]]
for KNN, Classification Report
```

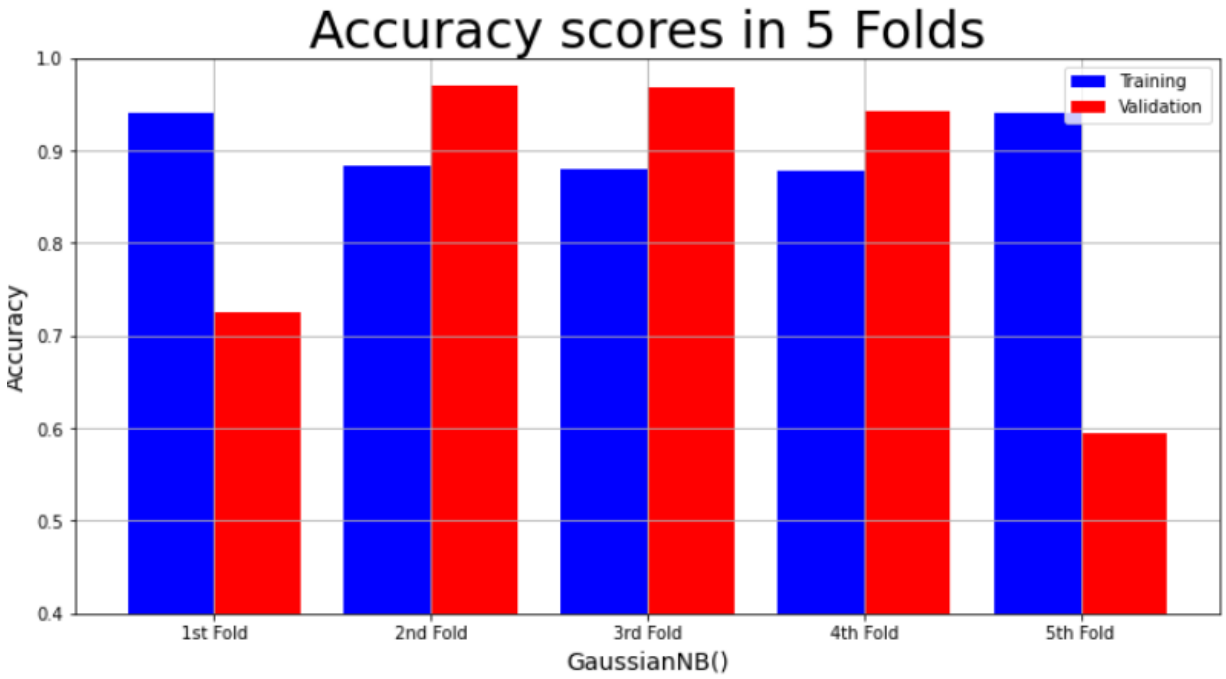
NAIVE BAYES

Grid Search for Hyperparameter Tuning : {'var_smoothing': 1e-09}

Tuned model Accuracy: 90.04

Accuracy obtained after variable smoothing : 0.8583

Plot of Accuracy of tuned model while doing 5 fold validation on each fold



Confusion Matrix for Naive Bayes

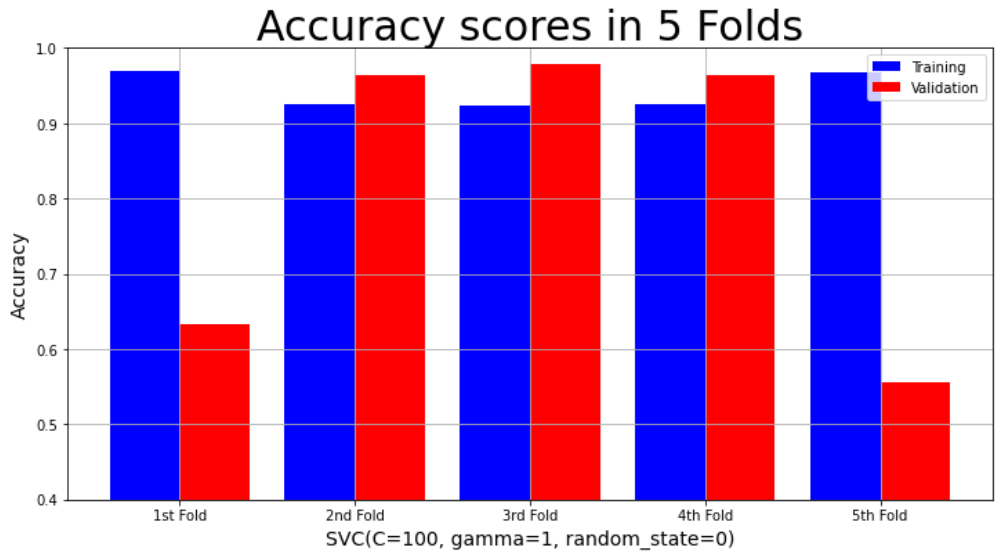
```
for Naive Bayes, Confusion Matrix:
 [[259   0  40   0   2   3  15]
 [  0 111   0   0   0   0   0]
 [ 35   0 377   0   5   1   1]
 [  0   0   0 808   2  19  76]
 [  0   0  10   4 469   0   8]
 [  2   0   0   3   0 442  23]
 [  4   0   1  57  18  10 598]]
for Naive Bayes, Classification Report
```

SUPPORT VECTOR MACHINE (SVM)

Grid Search for Hyperparameter Tuning :
Best parameters : {'C': 100, 'gamma': 1, 'kernel': 'rbf'}
Best estimator : SVC(C=100, gamma=1)

Tuned model Accuracy on Training Set: 0.9343

Plot of Accuracy of tuned model while doing 5 fold validation on each fold



Confusion Matrix for SVM Model

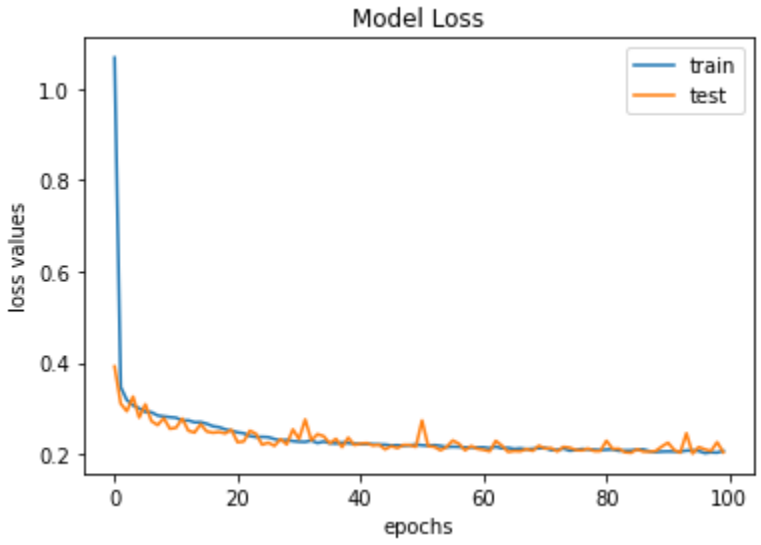
```
[[293   0  16   0   2   3   5]
 [  0 111   0   0   0   0   0]
 [ 10   0 400   0   5   1   3]
 [  1   0   0 849   1  10  44]
 [  1   0  11   3 467   0   9]
 [  1   0   0   5   0 448  16]
 [  1   0   1  77   5   3 601]]
```

Artificial Neural Network (ANN)

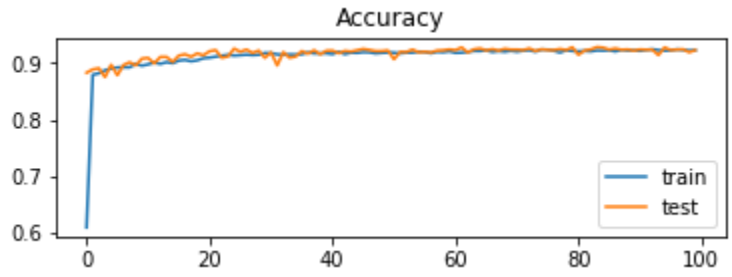
Grid search for hyperparameter tuning :

Best parameters : {'batch_size': 16, 'epochs': 100, 'optimizer': 'adam'}

Plotting Loss Function During Training of ANN



Accuracy during training of model



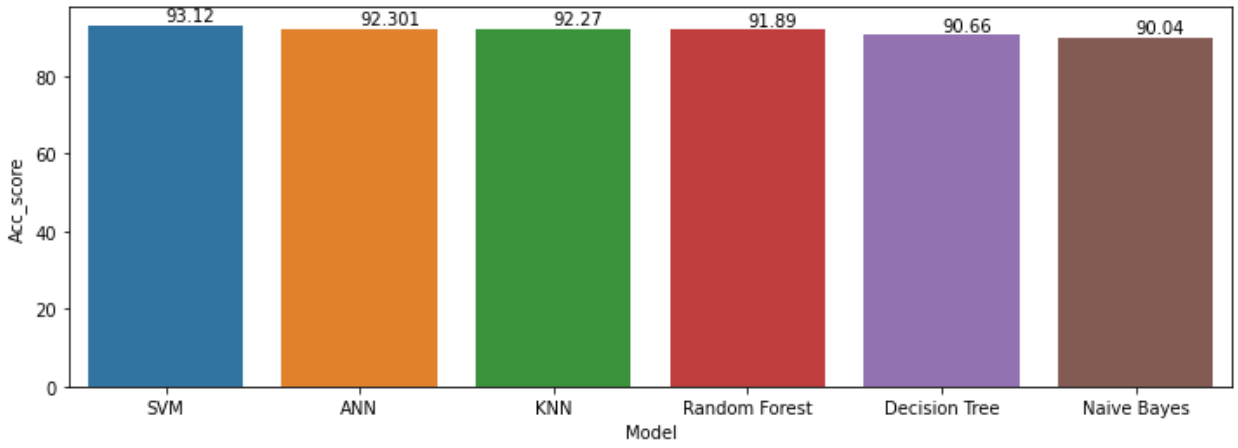
\

Confusion Matrix

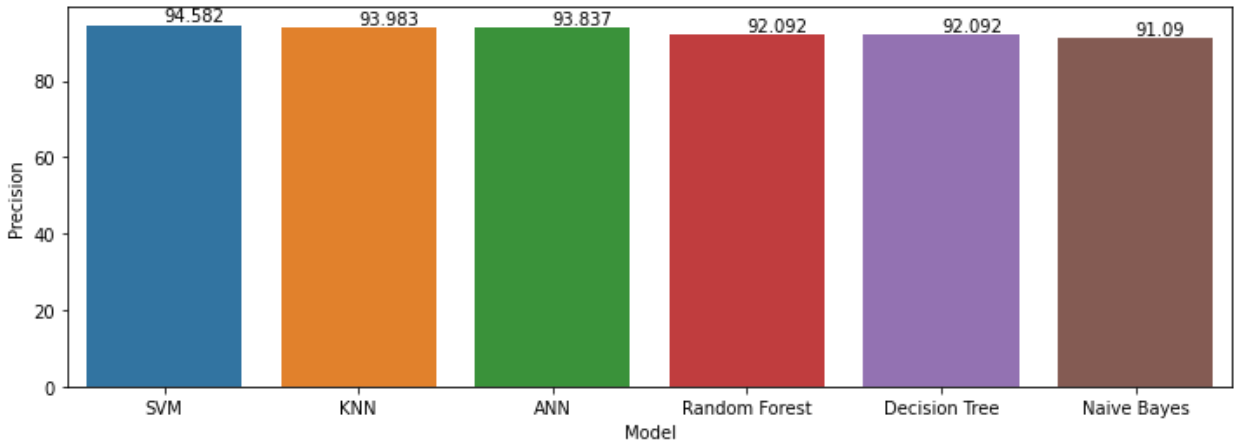
```
[[281   0  24   0   2   5   7]
 [  0 110   1   0   0   0   0]
 [  6   0 403   0   4   1   5]
 [  0   0   0 838   1  16  50]
 [  0   0  12   2 464   0  13]
 [  2   0   0   3   0 446  19]
 [  1   0   2  70   9   7 599]]
```

Comparison of Models

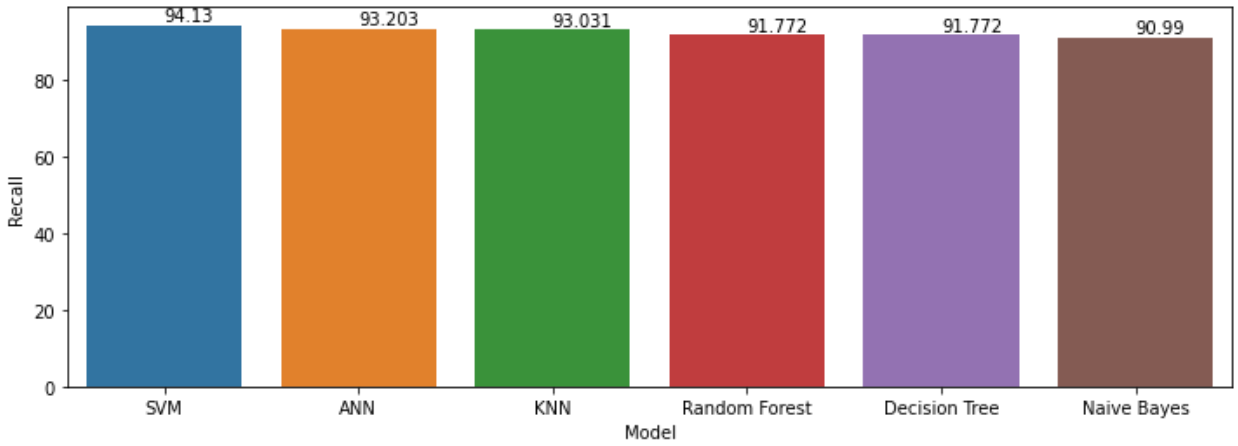
Accuracy



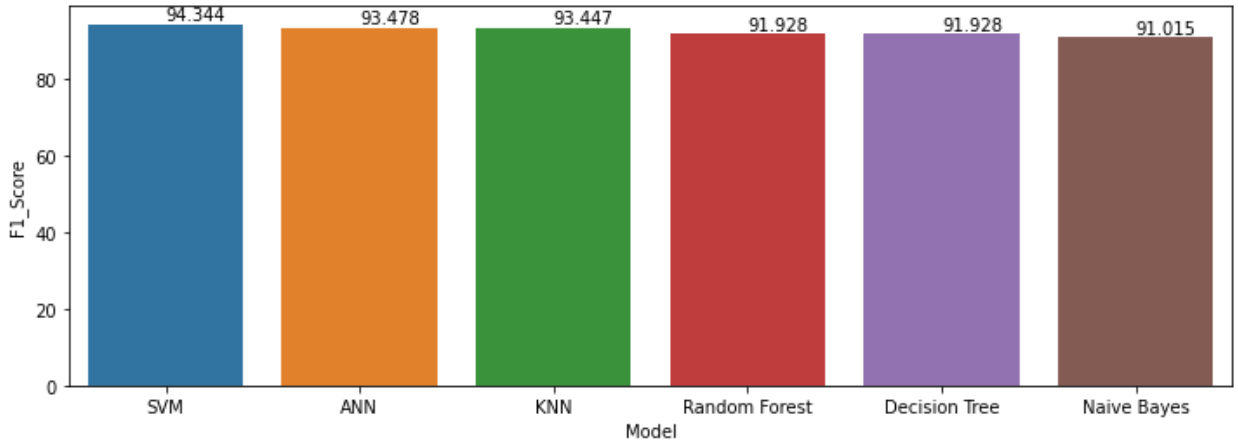
Precision



Recall



F1 Score


94.344
93.478
93.447
91.928
91.928
91.015
F1_Score
0
20
40
60
80
SVM
ANN
KNN
Random Forest
Decision Tree
Naive Bayes
Model

MNIST DATASET (Question 2)

Decision Tree:

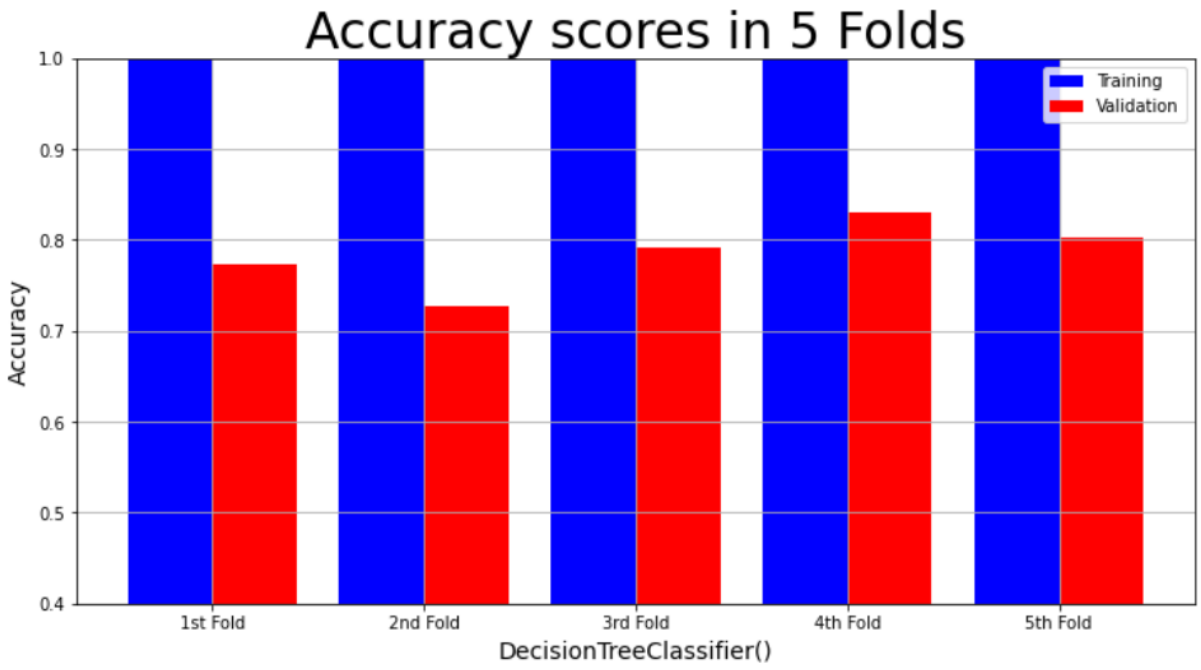
Randomized Search for Hyperparameter Tuning :

Best parameters found : {'criterion': 'entropy', 'max_depth': 5, 'max_features': 10, 'min_samples_split': 3}

Grid Search for Hyperparameter Tuning :

Tuned model Accuracy on Training Set: 0.8593822843822844

Plot of Accuracy of tuned model while doing 5 fold validation on each fold



Confusion Matrix for Decision Tree

```
for Decision Tree, Confusion Matrix:
 [[24  0  0  0  0  1  0  0  0  2]
 [ 0 30  0  3  1  0  0  0  0  1]
 [ 1  0 31  2  0  0  0  0  1  1]
 [ 0  0  0 26  0  0  0  1  0  2]
 [ 1  0  0  0 26  0  2  1  0  0]
 [ 0  0  0  1  1 32  0  1  1  4]
 [ 0  0  0  2  2  0 40  0  0  0]
 [ 0  0  0  0  1  1  0 37  0  0]
 [ 1  3  4  2  0  0  0  2 27  0]
 [ 0  0  0  3  0  3  0  0  2 33]]
for Decision Tree, Classification Report
```

RANDOM FOREST

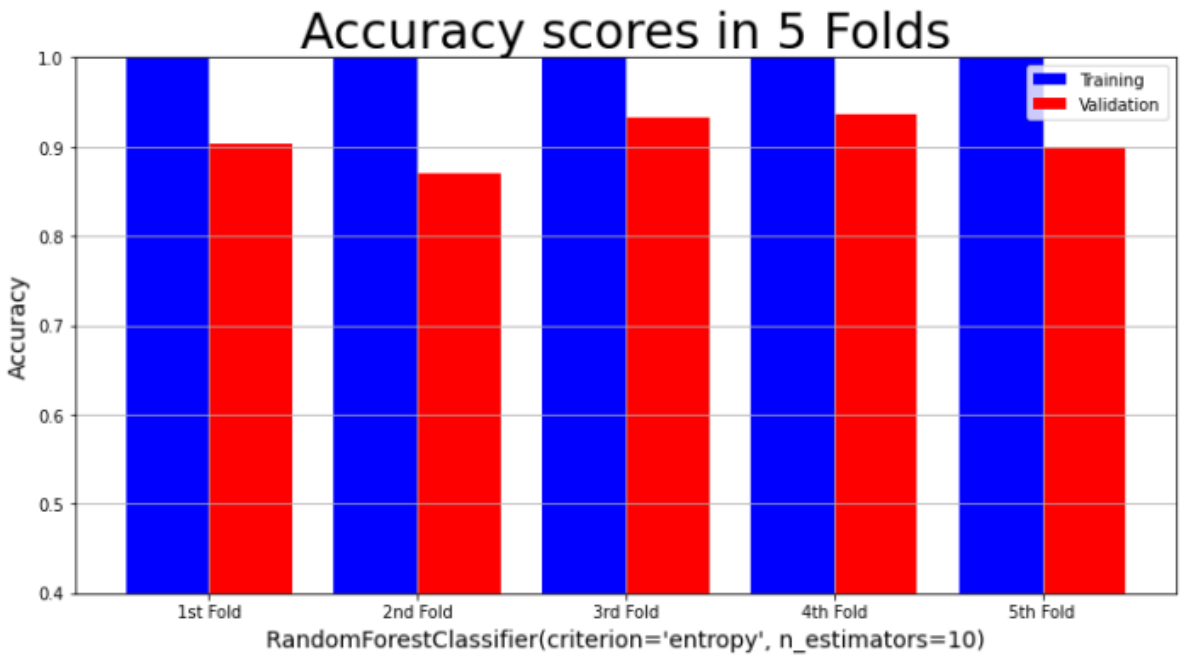
Randomized search for hyperparameter tuning :

Best parameters found : {'criterion': 'entropy', 'max_depth': 5, 'max_features': 10, 'min_samples_split': 3}

Grid Search for Hyperparameter Tuning :

Tuned model Accuracy on Training Set: 97.5

Plot of Accuracy of tuned model while doing 5 fold validation on each fold

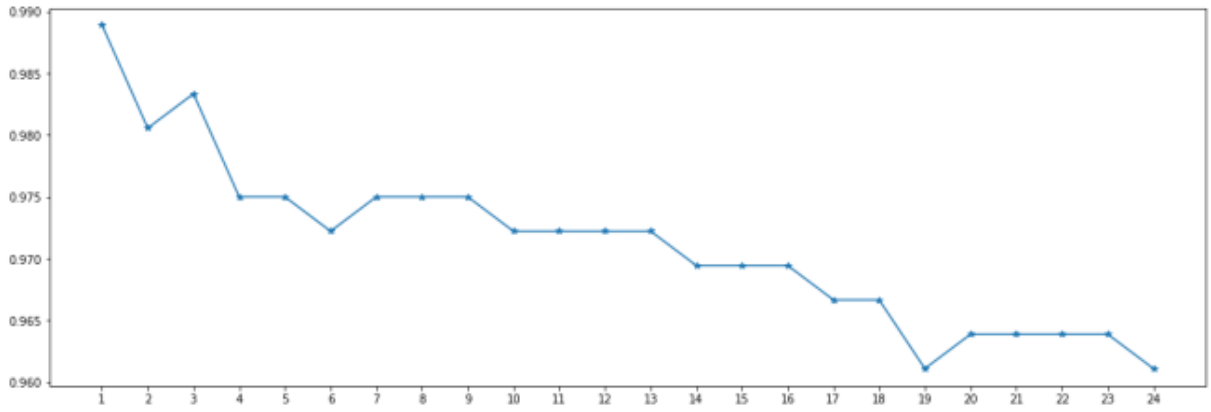


Confusion Matrix for Random Forest Model :

```
for Random Forest, Confusion Matrix:
 [[27  0  0  0  0  0  0  0  0  0]
 [ 0 34  0  0  0  1  0  0  0  0]
 [ 1  1 34  0  0  0  0  0  0  0]
 [ 0  0  0 29  0  0  0  0  0  0]
 [ 0  0  0  0 29  0  0  1  0  0]
 [ 0  0  0  0  0 39  0  0  0  1]
 [ 0  0  0  0  0  0 44  0  0  0]
 [ 0  0  0  0  0  0  0 39  0  0]
 [ 0  1  0  0  0  0  0  1 37  0]
 [ 0  0  0  1  0  1  0  0  0 39]]
for Random Forest, Classification Report
```

K-Nearest Neighbors

Plot of accuracy on different values of k :

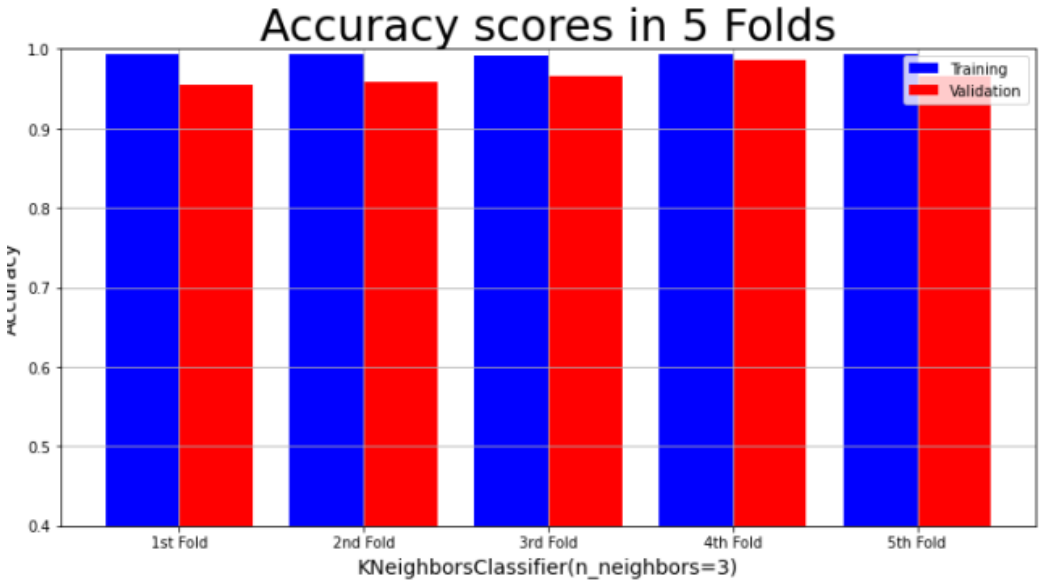


Best value of k obtained through randomized search :

Accuracy = 0.9766

Value of k = 3

Plot of Accuracy of tuned model while doing 5 fold validation on each fold



Confusion Matrix for KNN Model

```
for KNN, Confusion Matrix:
 [[27  0  0  0  0  0  0  0  0  0]
 [ 0 34  0  0  0  1  0  0  0  0]
 [ 0  0 36  0  0  0  0  0  0  0]
 [ 0  0  1 28  0  0  0  0  0  0]
 [ 0  0  0  0 29  0  0  1  0  0]
 [ 0  0  0  0  0 39  0  0  0  1]
 [ 0  0  0  0  0  0 44  0  0  0]
 [ 0  0  0  0  0  0  0 39  0  0]
 [ 0  0  0  2  0  0  0  0 37  0]
 [ 0  0  0  0  0  0  0  0  0 41]]
for KNN, Classification Report
```

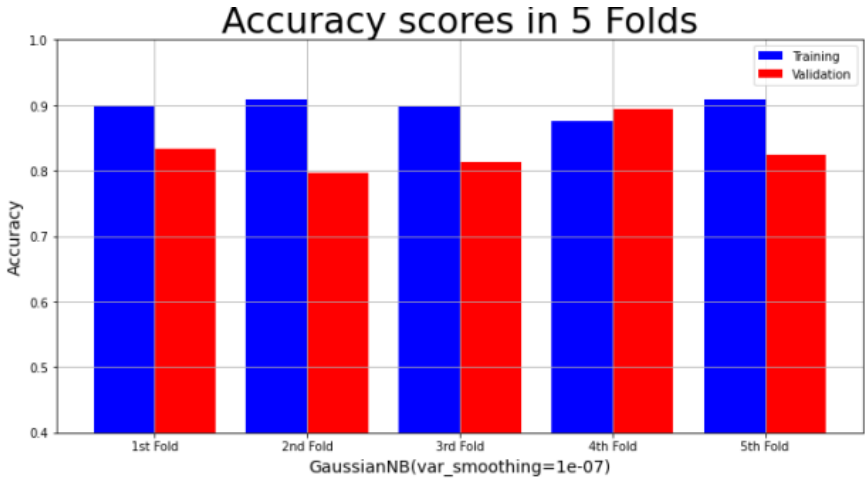
NAIVE BAYES

Grid Search for Hyperparameter Tuning : {'var_smoothing': 1e-07}

Tuned model Accuracy: 0.8204631242740998

Accuracy obtained after variable smoothing : 0.8583

Plot of Accuracy of tuned model while doing 5 fold validation on each fold



Confusion Matrix for Naive Bayes Model

```
for Naive Bayes, Confusion Matrix:
 [[27  0  0  0  0  0  0  0  0  0]
 [ 0 29  0  0  0  0  0  0  5  1]
 [ 0  6 21  2  0  0  0  0  7  0]
 [ 0  0  1 24  0  0  0  0  4  0]
 [ 0  1  0  0 27  0  0  2  0  0]
 [ 0  1  0  0  0 36  0  2  0  1]
 [ 0  0  0  0  0  0 44  0  0  0]
 [ 0  0  0  0  0  0  0 39  0  0]
 [ 0  3  0  0  0  1  0  1 34  0]
 [ 0  1  0  3  0  0  0  3  6 28]]
for Naive Bayes, Classification Report
```

SUPPORT VECTOR MACHINE

Randomized search for hyperparameter tuning :

0.9902534965034965 {'kernel': 'rbf', 'gamma': 0.001, 'C': 100}

Grid Search for Hyperparameter Tuning :

Best parameters found : {'C': 1, 'gamma': 0.001, 'kernel': 'rbf'} SVC(C=1, gamma=0.001)

Tuned model Accuracy on Training Set: 97.5

Confusion Matrix for SVM :

```
for SVM, Confusion Matrix:
 [[27  0  0  0  0  0  0  0  0  0]
 [ 0 35  0  0  0  0  0  0  0  0]
 [ 0  0 36  0  0  0  0  0  0  0]
 [ 0  0  0 29  0  0  0  0  0  0]
 [ 0  0  0  0 30  0  0  0  0  0]
 [ 0  0  0  0  0 39  0  0  0  1]
 [ 0  0  0  0  0  0 44  0  0  0]
 [ 0  0  0  0  0  0  0 39  0  0]
 [ 0  1  0  0  0  0  0  0 38  0]
 [ 0  0  0  0  0  1  0  0  0 40]]
for SVM, Classification Report
```

ARTIFICIAL NEURAL NETWORK

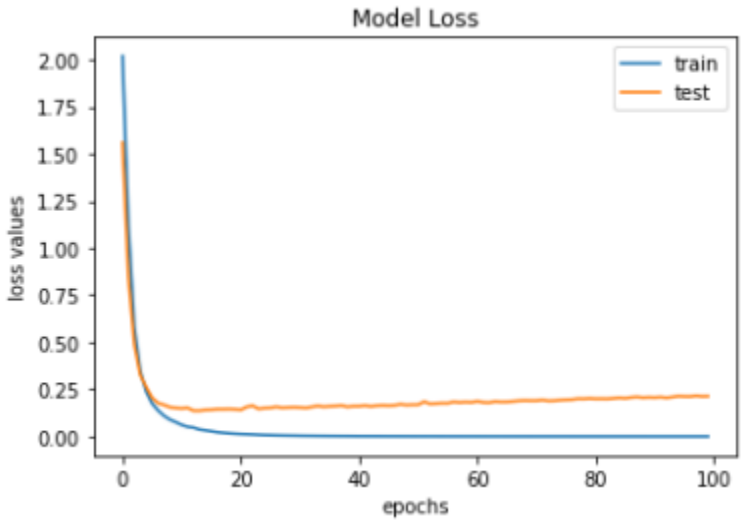
Applying randomized search CV :

0.9742523228803716 {'optimizer': 'adam', 'epochs': 150, 'batch_size': 16}

Applying GridSearchCV for hyperparameter tuning :

Best Parameters: {'batch_size': 16, 'epochs': 100, 'optimizer': 'adam'}

PLOTTING THE LOSS FUNCTION DURING TRAINING OF THE ANN MODEL



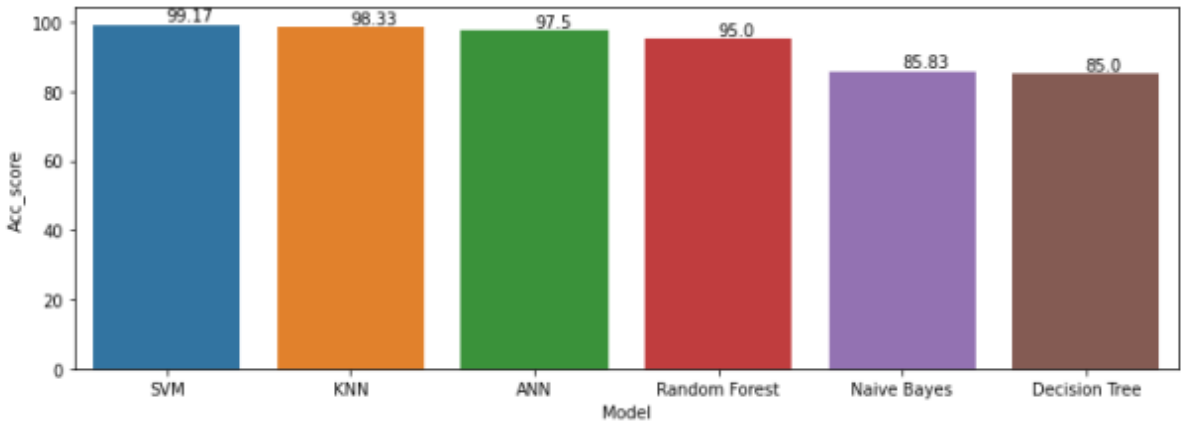
Confusion Matrix

```
for ANN, Confusion Matrix:
 [[27  0  0  0  0  0  0  0  0  0]
 [ 1 33  0  0  0  0  0  0  0  1]
 [ 0  0 36  0  0  0  0  0  0  0]
 [ 0  0  0 28  0  1  0  0  0  0]
 [ 0  0  0  0 30  0  0  0  0  0]
 [ 0  0  0  0  0 39  0  0  0  1]
 [ 0  1  0  0  0  0 43  0  0  0]
 [ 0  0  0  0  1  0  0 38  0  0]
 [ 0  1  0  0  0  0  0  0 38  0]
 [ 0  0  0  0  0  1  0  1  0 39]]
for ANN, Classification Report
```

Comparison of Models

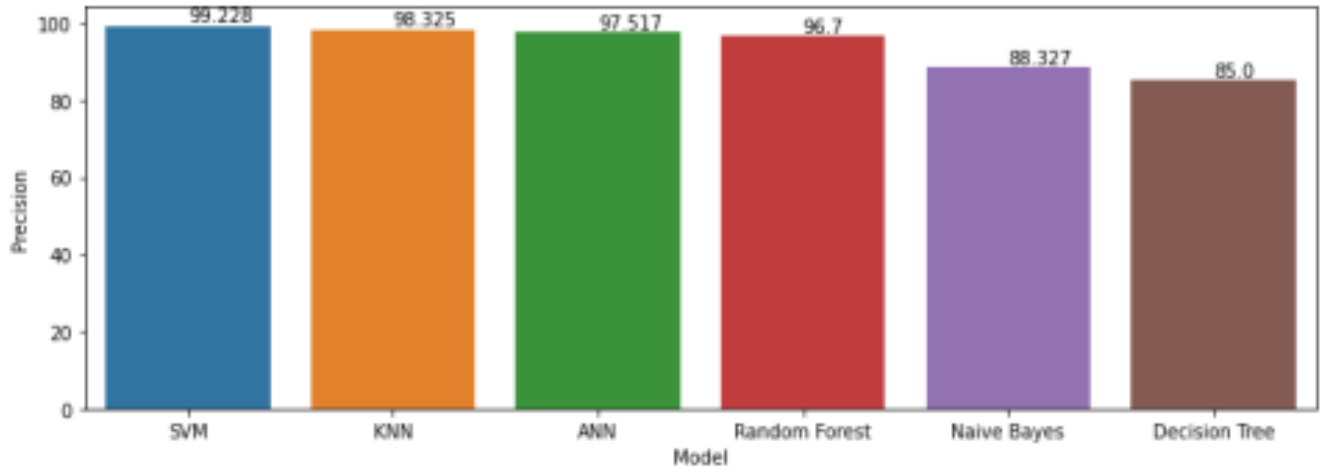
1. ACCURACY

	Model	Score	Acc_score
0	SVM	99.93	99.17
1	KNN	99.10	98.33
2	ANN	100.00	97.50
3	Random Forest	100.00	95.00
4	Naive Bayes	88.94	85.83
5	Decision Tree	100.00	85.00



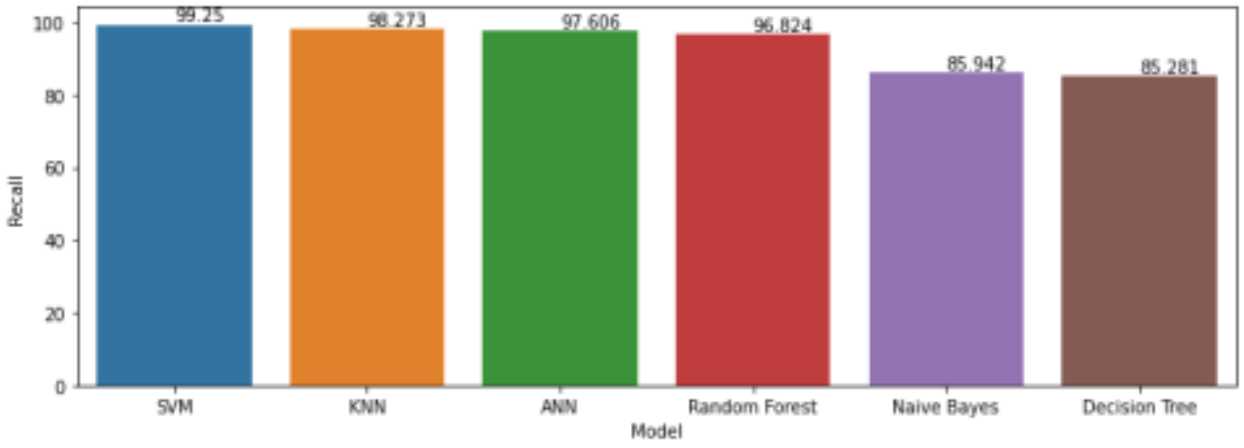
Precision

	Model	Precision
0	SVM	99.228
1	KNN	98.325
2	ANN	97.517
3	Random Forest	96.700
4	Naive Bayes	88.327
5	Decision Tree	85.000



RECALL

	Model	Recall
0	SVM	99.250
1	KNN	98.273
2	ANN	97.606
3	Random Forest	96.824
4	Naive Bayes	85.942
5	Decision Tree	85.281



F1 Score

	Model	F1_Score
0	SVM	99.235
1	KNN	98.280
2	ANN	97.548
3	Random Forest	96.701