Machine Learning Algorithms

What is Machine Learning?

Machine Learning is the science of making computers learn and act like humans by feeding data and information without being explicitly programmed

TYPES OF ML ALGORITHMS:

- SUPERVISED LEARNING ALGORITHMS
- SEMI SUPERVISED LEARNING ALGORITHMS
- UNSUPERVISED LEARNING ALGORITHMS
- REINFORCEMENT LEARNING ALGORITHMS

SUPERVISED LEARNED ALGORITHMS

CLASSICATION:

- NAÏVE BAYES
- KNN
- DECISION TREE
- SVM
- KERNAL SVM
- RANDOM FOREST
- LOGISTIC REGRESSION

REGRESSION:

- LINEAR REGRESSION
- NEURAL NETWORK REGRESSION
- SUPPORT VECTOR REGRESSION

SEMI SUPERVISED LEARNING ALGORITHMS:

- KNN
- LOGISTIC REGRESSION
- LINEAR REGRESSION
- DECISION TREE

UNSUPERVISED LEARNING ALGORITHMS:

- K-MEANS CLUSTERING
- AGGLOMERATIVE HIERARCHICAL CLUSTERING
- PRINCIPAL COMPONENT ANALYSIS
- MEAN-SHIFT CLUSTERING
- GAUSSIAN MIXTURE

REINFORCEMENT LEARNING ALGORITHMS:

- Q-LEARNING
- R-LEARNING
- TD-LEARNING

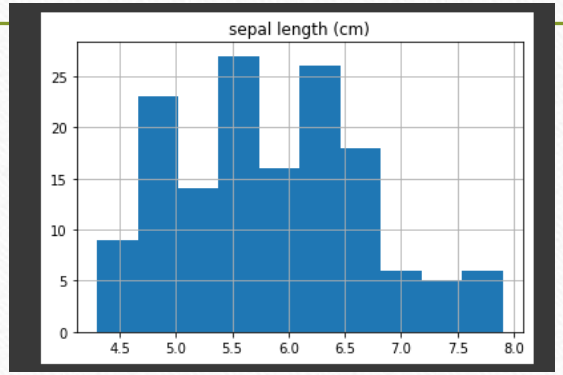
DATASET USED- IRIS

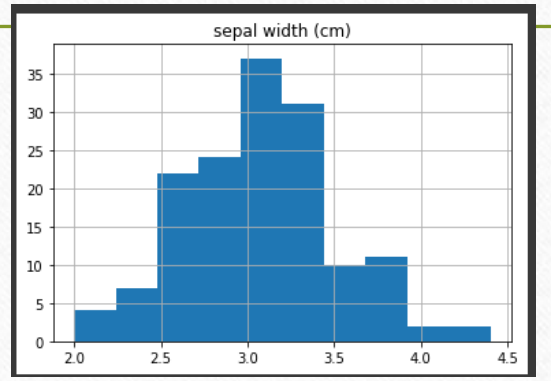
4 NUMERIC ATTRIBUTES

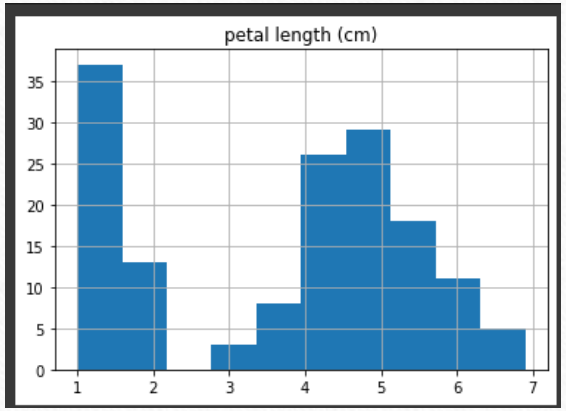
- Sepal length in cm
- Sepal width in cm
- Petal length in cm
- Petal width in cm

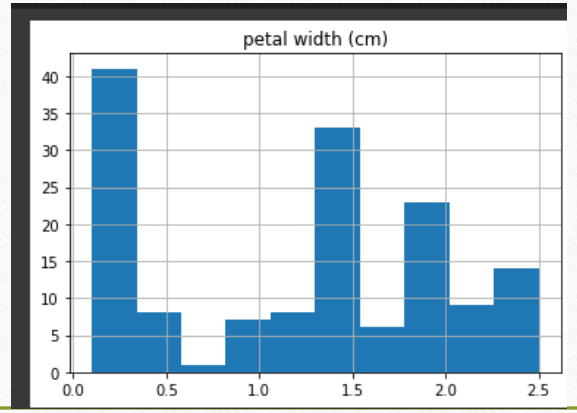
3 SPECIES CLASSIFIED

- Iris-virginica
- Iris-versicolor
- Iris-setosa







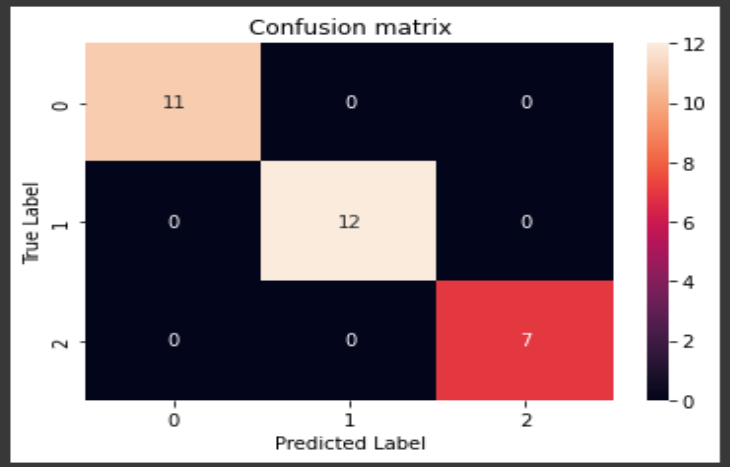


KNN(classification)

- K Nearest Neighbours
- It classifies based on how its neighbours are classified
- K refers to number of major nearest neighbors
- We use KNN when data is small

RESULT AND ANALYSIS OF KNN:

[24]



Finding the Classifier Accuracy

```
[25] from sklearn.metrics import accuracy_score
     accuracy_score(y_test, y_pred)
```

```
1.0
```

TP	FN
12	0
FP	TN
0	11

METRICS:

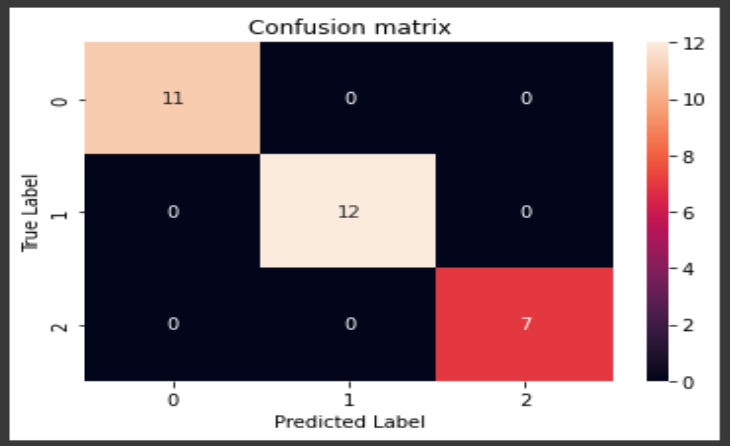
	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	1.00	1.00	1.00	12
2	1.00	1.00	1.00	7
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

NAÏVE BAYES:

- Classification technique based on Baye's Theorem
- Easy to build and stable to data changes
- Paticularly useful for large datasets
- It allows to classify based on a given set of features using probability
- As an example it can mark email as **spam** or **not spam**

RESULT AND ANALYSIS OF NAIVE BAYES:

[24]



Finding the Classifier Accuracy

```
[25] from sklearn.metrics import accuracy_score
     accuracy_score(y_test, y_pred)
```

```
1.0
```

TP	FN
12	0
FP	TN
0	11

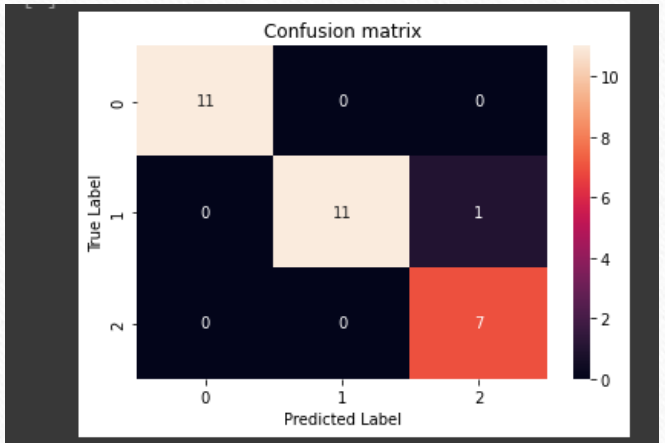
METRICS:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	1.00	1.00	1.00	12
2	1.00	1.00	1.00	7
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

DECISION TREE:

- A decision tree is a decision support tool that uses a tree like graph of decisions and their possible consequences
- It is one way to display an algorithm that only contains conditional control statements
- It is a binary tree of nodes

RESULT AND ANALYSIS OF DECISION TREE:



Finding the Classifier Accuracy

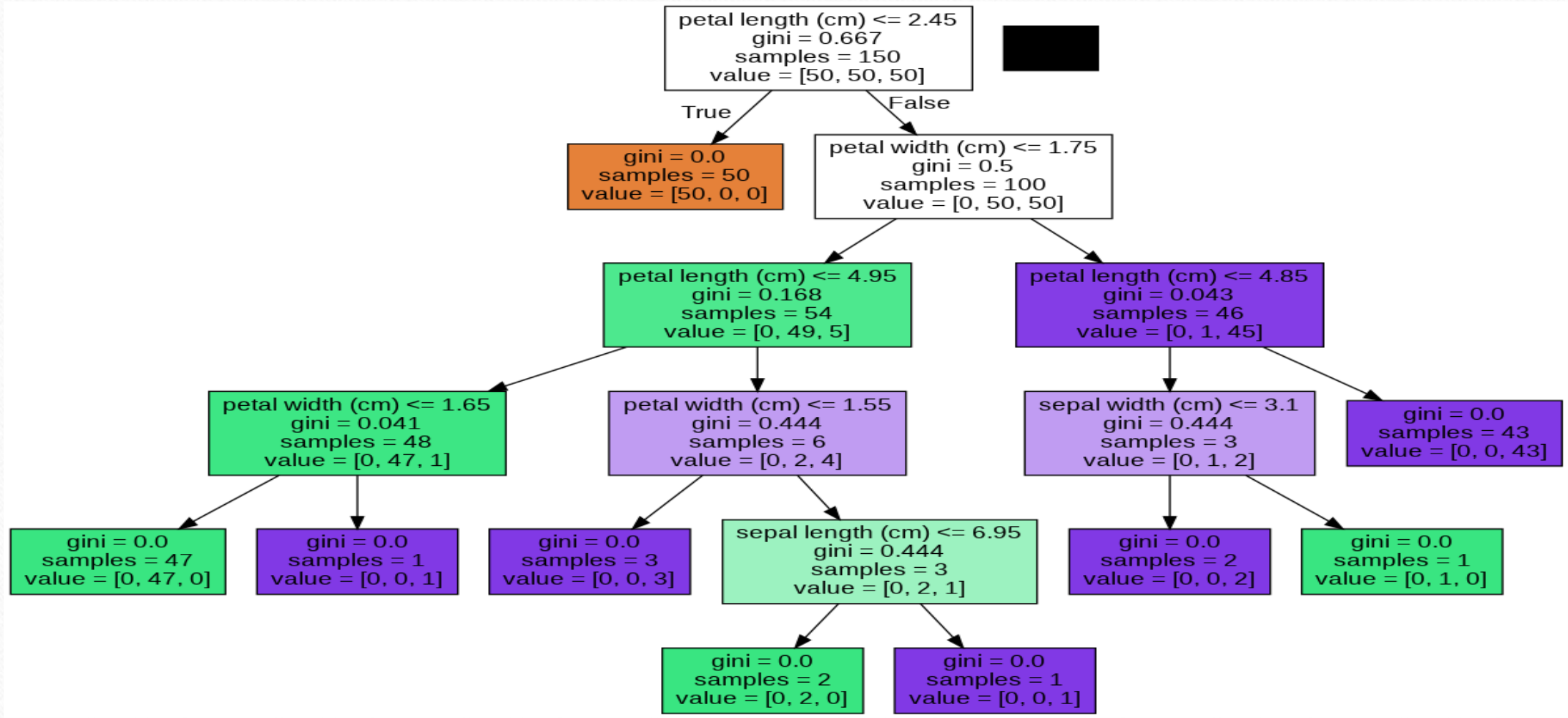
```
[ ] from sklearn.metrics import accuracy_score
    accuracy_score(y_test, y_pred)
```

```
0.9666666666666667
```

TP	FN
12	0
FP	TN
1	11

METRICS:

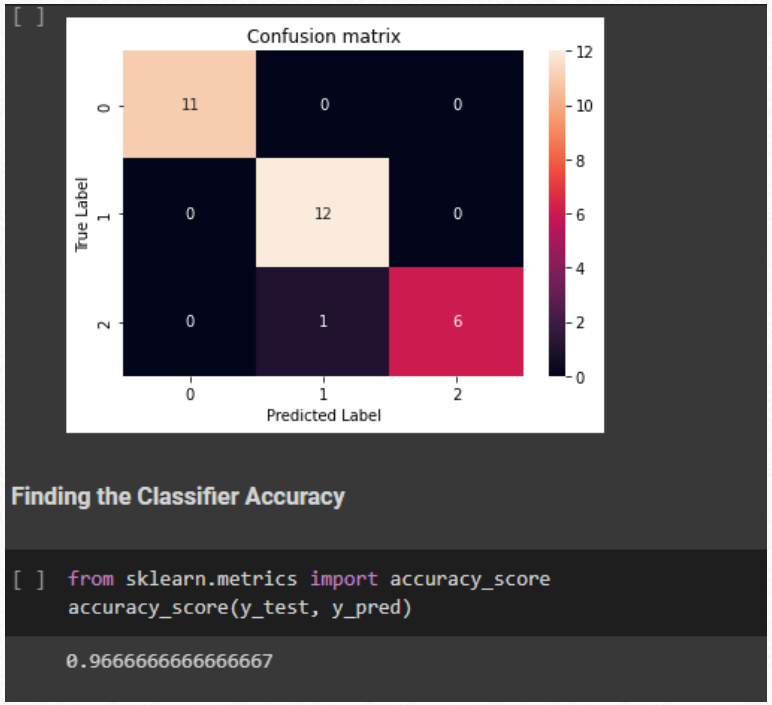
	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	0.92	1.00	0.96	12
2	1.00	0.86	0.92	7
accuracy			0.97	30
macro avg	0.97	0.95	0.96	30
weighted avg	0.97	0.97	0.97	30


petal length (cm) <= 2.45
gini = 0.667
samples = 150
value = [50, 50, 50]
True
False
gini = 0.0
samples = 50
value = [50, 0, 0]
petal width (cm) <= 1.75
gini = 0.5
samples = 100
value = [0, 50, 50]
petal length (cm) <= 4.95
gini = 0.168
samples = 54
value = [0, 49, 5]
petal length (cm) <= 4.85
gini = 0.043
samples = 46
value = [0, 1, 45]
petal width (cm) <= 1.65
gini = 0.041
samples = 48
value = [0, 47, 1]
petal width (cm) <= 1.55
gini = 0.444
samples = 6
value = [0, 2, 4]
sepal width (cm) <= 3.1
gini = 0.444
samples = 3
value = [0, 1, 2]
gini = 0.0
samples = 43
value = [0, 0, 43]
gini = 0.0
samples = 47
value = [0, 47, 0]
gini = 0.0
samples = 1
value = [0, 0, 1]
gini = 0.0
samples = 3
value = [0, 0, 3]
sepal length (cm) <= 6.95
gini = 0.444
samples = 3
value = [0, 2, 1]
gini = 0.0
samples = 2
value = [0, 0, 2]
gini = 0.0
samples = 1
value = [0, 1, 0]
gini = 0.0
samples = 2
value = [0, 2, 0]
gini = 0.0
samples = 1
value = [0, 0, 1]

SVM:

- Support Vector Machine algorithm
- It uses a technique called the kernel trick to transform data
- Then based on that it finds optimal boundary between possible outputs
- It works by classifying the data into different classes by finding a line
- Can be used to reduce redundant information
- Mainly used for face or hand writing recognition,bioinformatics

RESULT AND ANALYSIS OF SVM:

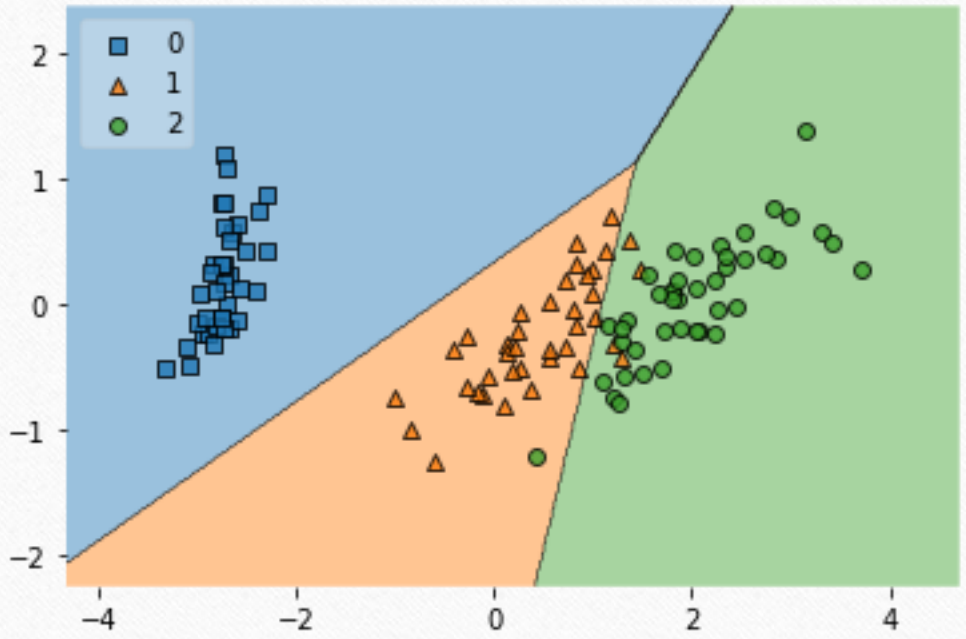


TP	FN
11	0
FP	TN
0	11

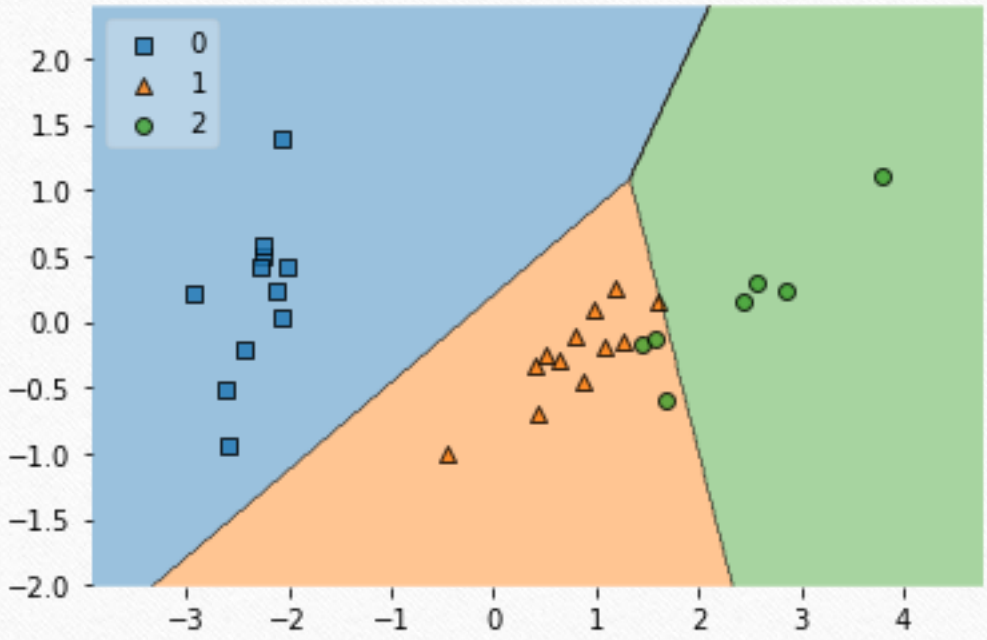
METRICS:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	1.00	0.92	0.96	12
2	0.88	1.00	0.93	7
accuracy			0.97	30
macro avg	0.96	0.97	0.96	30
weighted avg	0.97	0.97	0.97	30

TRAIN SET

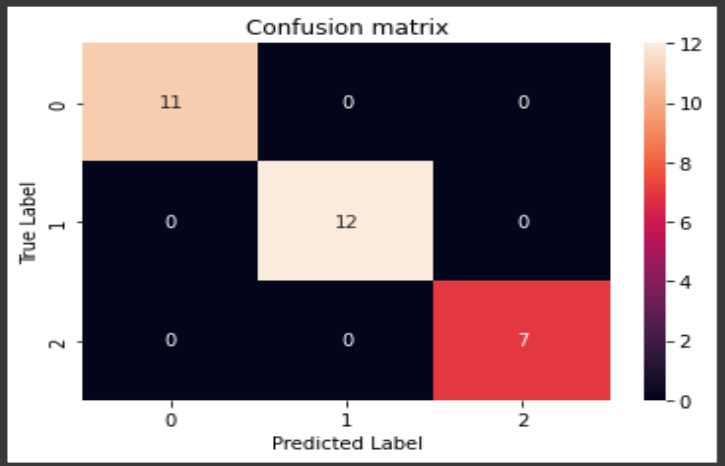


TEST SET



RESULT AND ANALYSIS OF KERNAL SVM:

[24]



Finding the Classifier Accuracy

```
[25] from sklearn.metrics import accuracy_score
     accuracy_score(y_test, y_pred)
```

```
1.0
```

TP	FN
12	0
FP	TN
0	11

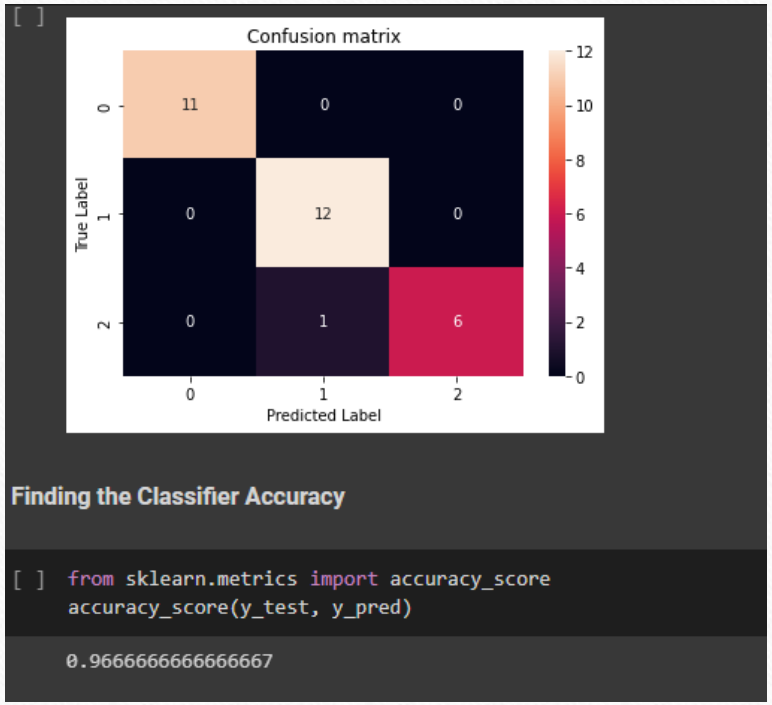
METRICS:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	1.00	1.00	1.00	12
2	1.00	1.00	1.00	7
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

RANDOM FOREST:

- A learning algorithm combining multiple algorithms to generate better results
- Each individual classifier is weak but when combined with others,can produce excellent results
- It gives estimation of what variables are important in classification
- Maintains high accuracy even large data is missing

RESULT AND ANALYSIS OF RANDOM FOREST:

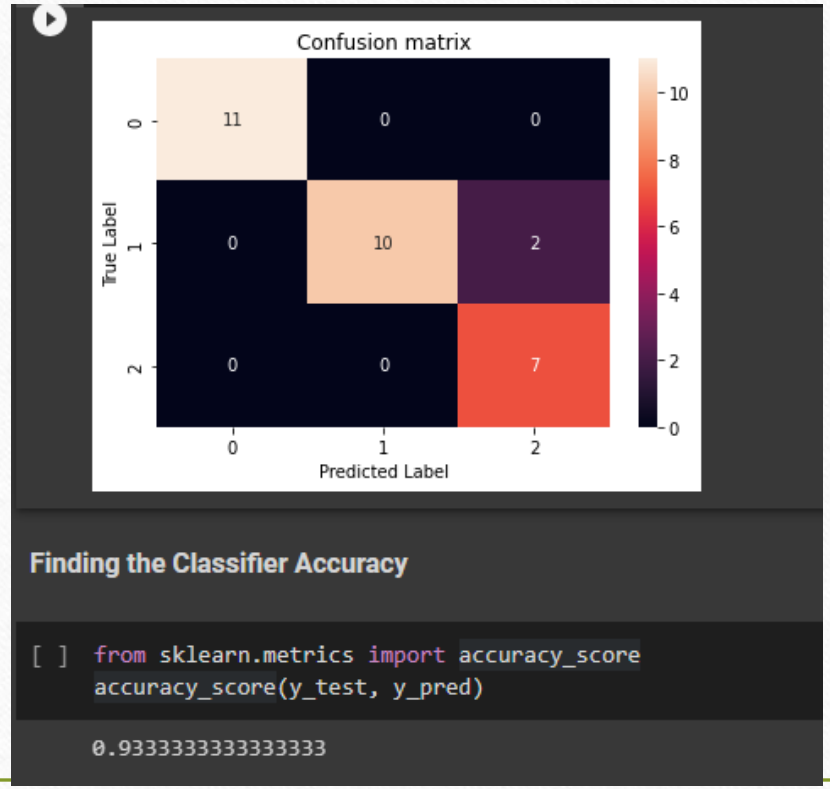


TP	FN
11	0
FP	TN
0	11

METRICS:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	1.00	0.92	0.96	12
2	0.88	1.00	0.93	7
accuracy			0.97	30
macro avg	0.96	0.97	0.96	30
weighted avg	0.97	0.97	0.97	30

RESULT AND ANALYSIS OF RANDOM FOREST(implemented with principal component analysis):



TP	FN
10	0
FP	TN
0	11

METRICS:

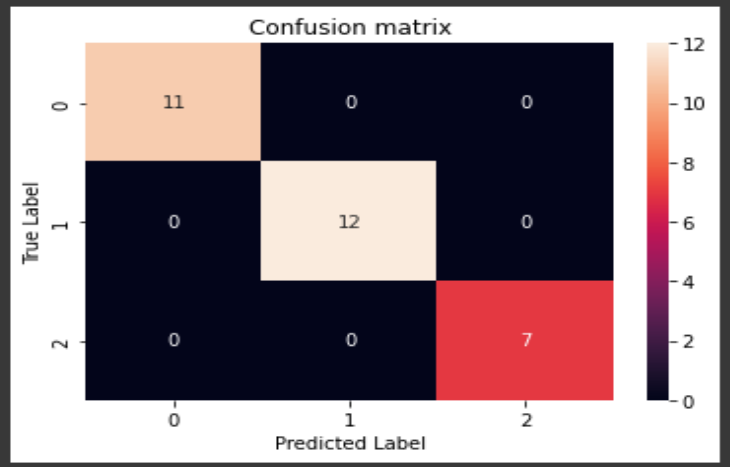
	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	1.00	0.83	0.91	12
2	0.78	1.00	0.88	7
accuracy			0.93	30
macro avg	0.93	0.94	0.93	30
weighted avg	0.95	0.93	0.93	30

LOGISTIC REGRESSION CLASSIFICATION:

- Simple but performs well
- Estimates probability of an event occurring based on the previous data provided
- It is used to cover a binary dependent variable i.e 0 and 1
- Mainly used for sentimental analysis like classifying good reviews from bad ones and depression prediction

RESULT AND ANALYSIS OF LRC:

[24]



Finding the Classifier Accuracy

```
[25] from sklearn.metrics import accuracy_score
     accuracy_score(y_test, y_pred)
```

```
1.0
```

TP	FN
12	0
FP	TN
0	11

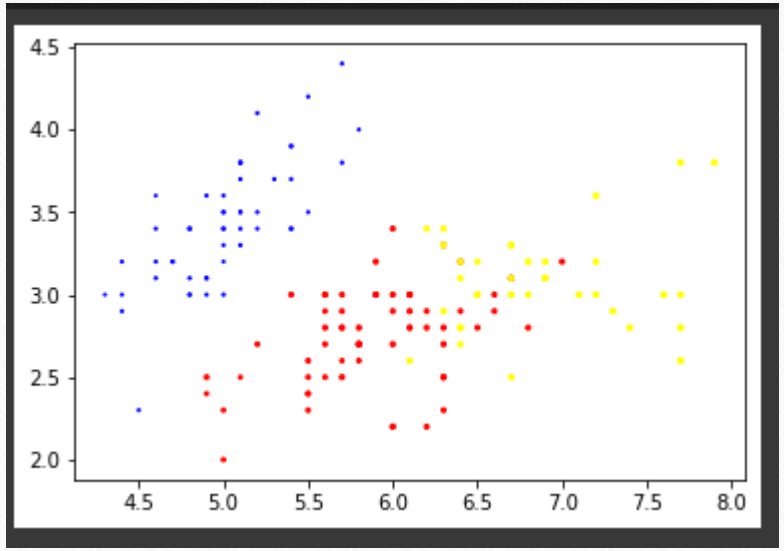
METRICS:

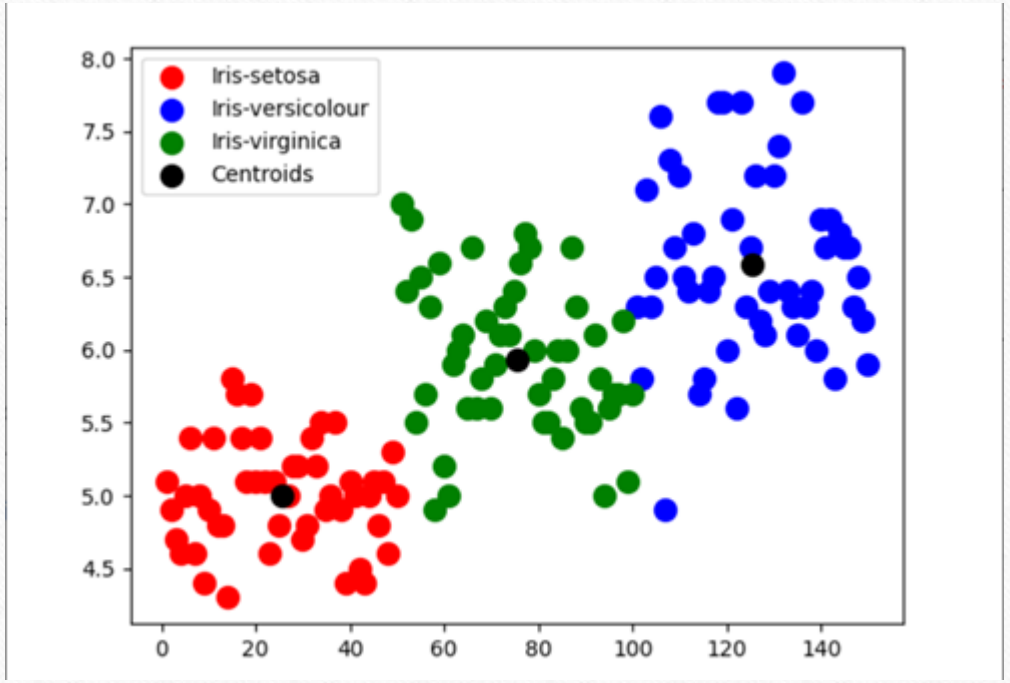
	precision	recall	f1-score	support
0	1.00	1.00	1.00	11
1	1.00	1.00	1.00	12
2	1.00	1.00	1.00	7
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

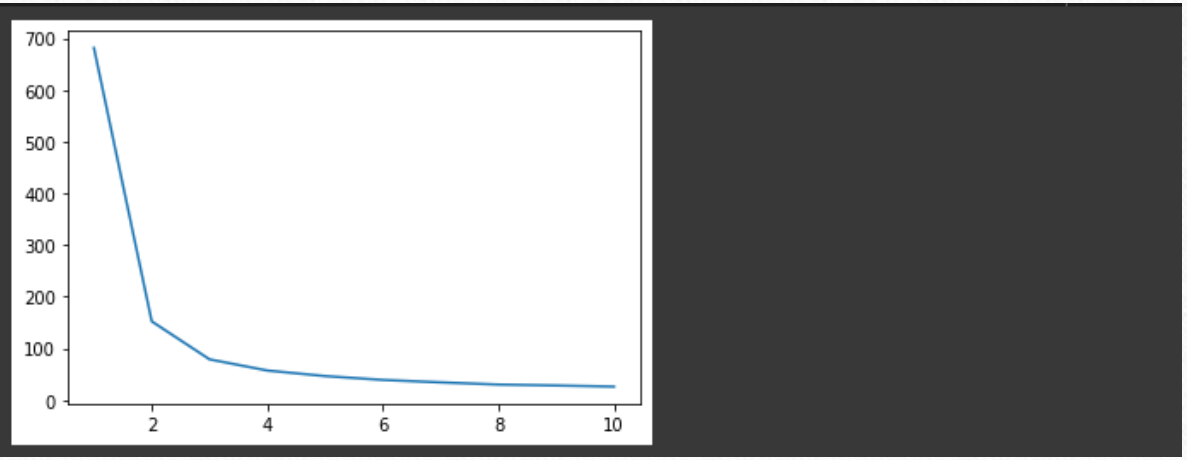
K-MEANS CLUSTERING:

- Used to categorize unlabeled data
- It works by finding groups within the data with the number of groups represented
- The results of the K-means clustering algorithm are the centroids of the k clusters which are used to label new data
- Mainly used for sorting sensor measurments and detecting bots

RESULT AND ANALYSIS OF KMeans:





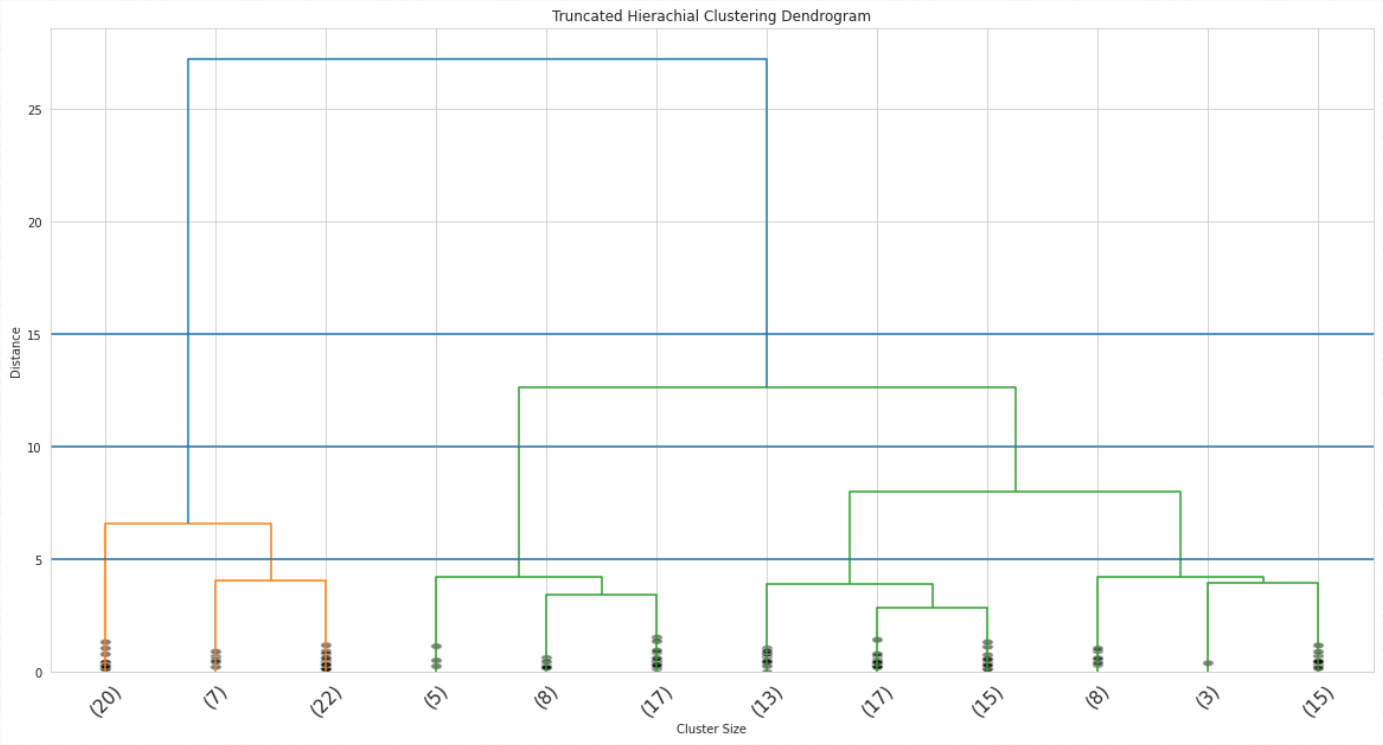



```
[ ]  accuracy_score(iris.target,km.labels_)

     0.24

[ ]  silhouette_score(x,km.labels_,metric ="euclidean")

     0.5528190123564095
```

AGGLOMERATIVE HIERARCHICAL CLUSTERING:

- Agglomerative(bottom-up)
- Start with each document being a single cluster
- Eventually all documents belong same cluster
- Does not require number clusters in advance
- Key operation is the computation of the distance between two clusters

RESULT AND ANALYSIS OF HIERARCHICAL CLUSTERING:



```
sm.accuracy_score(target,HClustering.labels_)
```

```
0.013333333333333334
```

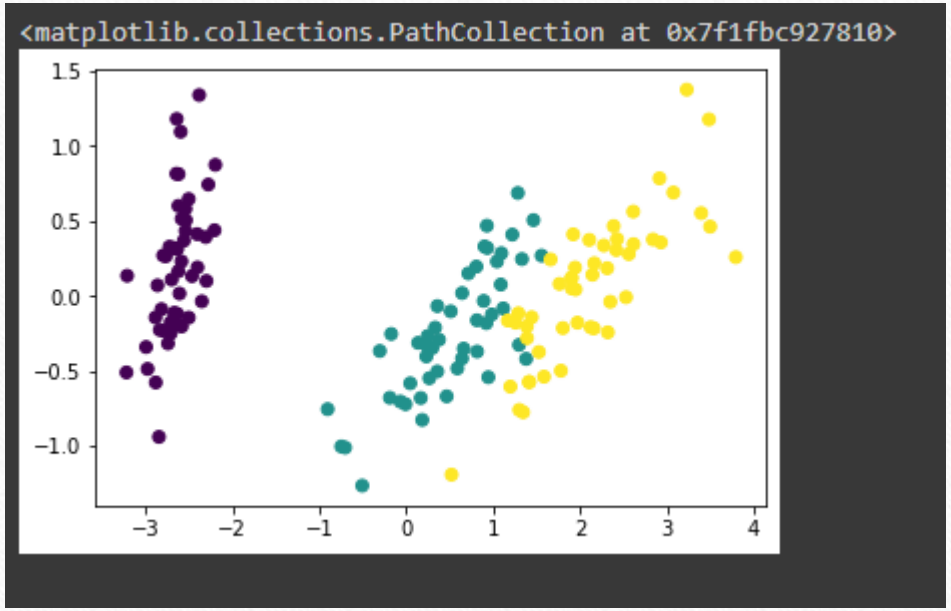
PRINCIPAL COMPONENT ANALYSIS:

- An exploratory technique used to reduce the dimentionality of the data to 2D or 3D
- Find patterns in high dimensionality
- Visualize data of high dimensionality
- Examples :

  - Face recognition

  -Image compression

RESULT AND ANALYSIS OF PCA:



COMPARISION OF ACCURACIES OF ALGORITHMS