Guided Assignment On Feature Engineering

Name: **Shreya Shrivastava**

Course: **Artificial Intelligence And Machine Learning**

Batch Four

Duration: **12 Months**

Problem statement: In order to implement LDA, first generate a dummy dataset **(say IRIS dataset having 4 features)** and the use LDA to decrease the number of **features to one/two**. Now using this modified dataset, try to learn a classifier to test the performance of LDA for dimensionality reduction.

Prerequisites:

The libraries as well as things required in order for the program to work:

I. **Python 3.6** : The following url https://www.python.org/downloads/ can be referred to download python. Once you have python downloaded and installed, you will need to setup PATH variables (if you want to run python program directly, detail instructions are below in how to run software section). To do that check this: https://www.pythoncentral.io/add-python-to-path-python-is-not-recognized-as-an-internal-or-external-command/. Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic. Second option is to download anaconda and use its anaconda prompt to run the commands. To install anaconda check this url : https://www.anaconda.com/download/

II. **ADDITIONAL PACKAGES :** You will also need to download and install below 3 packages- numpy, scikit learn and matplotlib after you install either python or anaconda from the steps above. If you have chosen to install python 3.6,then run the following commands in command prompt/terminal to install these packages :

NUMPY: pip install -U numpy

MATPLOTLIB: pip install -U matplotlib

SCIKIT LEARN: pip install -U scikit-learn

If using Anaconda then run the following commands in anaconda prompt to install these packages:

NUMPY: conda install -c anaconda numpy

MATPLOTLIB: conda install -c anaconda matplotlib

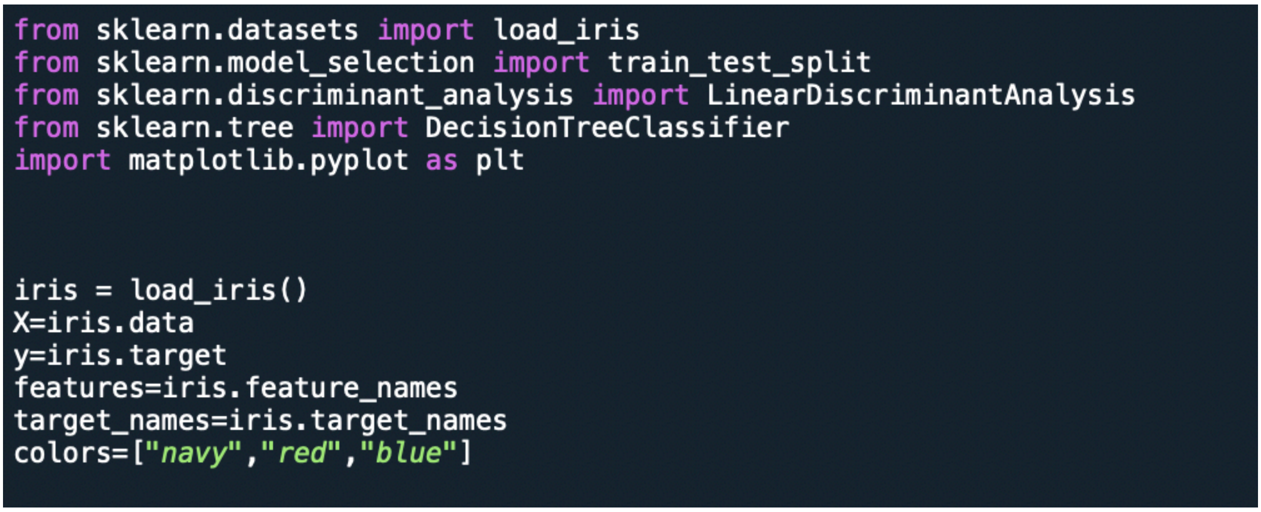
SCIKIT LEARN: conda install -c scikit-learn

III. METHOD USED:

A. LINEAR DISCRIMINANT ANALYSIS

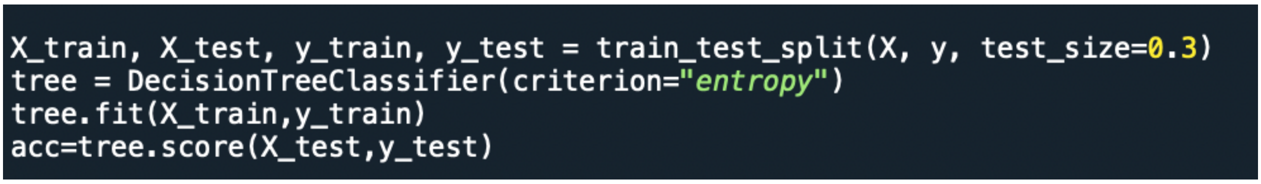
THE PROJECT :

1. Importing the libraries and loading the iris dataset. We then proceed to load the data as well as the labels and store them in X and y respectively. Defining a colour list for the plotting of the graphs further down the course.



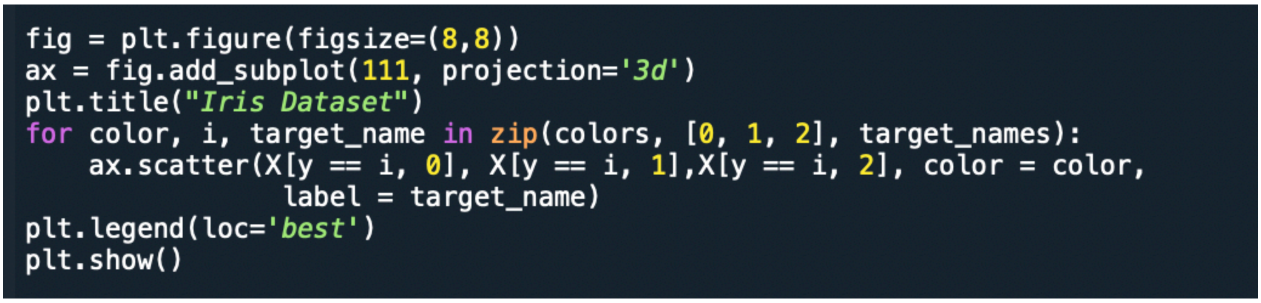
```
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.tree import DecisionTreeClassifier
import matplotlib.pyplot as plt



iris = load_iris()
X=iris.data
y=iris.target
features=iris.feature_names
target_names=iris.target_names
colors=["navy","red","blue"]
```

2. Splitting the data into training and testing respectively. Then we fit a decision tree classifier on the training set and calculate the accuracy on the test dataset.



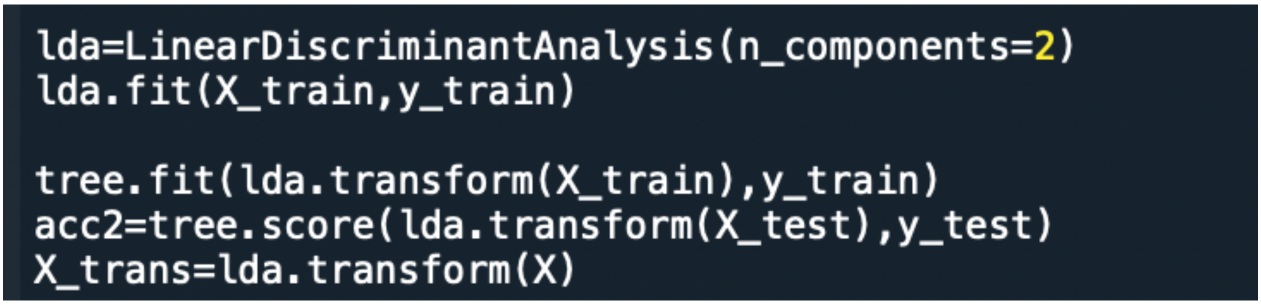
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
tree = DecisionTreeClassifier(criterion="entropy")
tree.fit(X_train,y_train)
acc=tree.score(X_test,y_test)
```

3. Plotting the iris dataset.



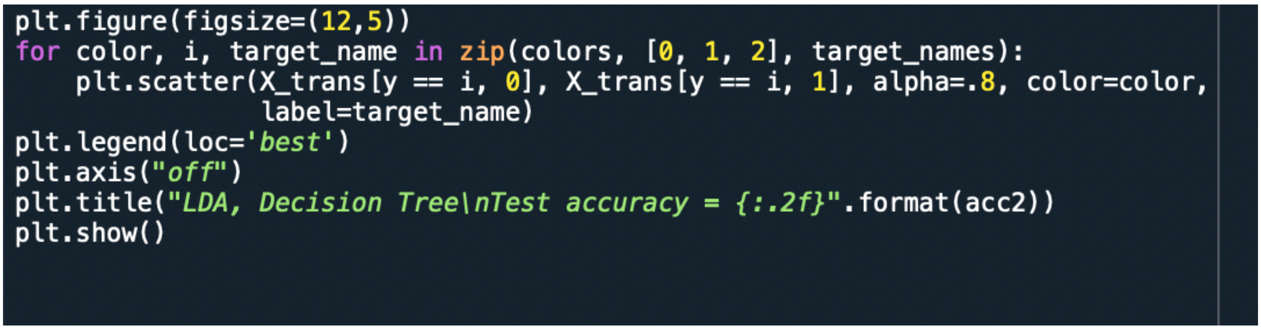
```
fig = plt.figure(figsize=(8,8))
ax = fig.add_subplot(111, projection='3d')
plt.title("Iris Dataset")
for color, i, target_name in zip(colors, [0, 1, 2], target_names):
    ax.scatter(X[y == i, 0], X[y == i, 1],X[y == i, 2], color = color,
               label = target_name)
plt.legend(loc='best')
plt.show()
```

4. Performing LDA and calculating the accuracy:

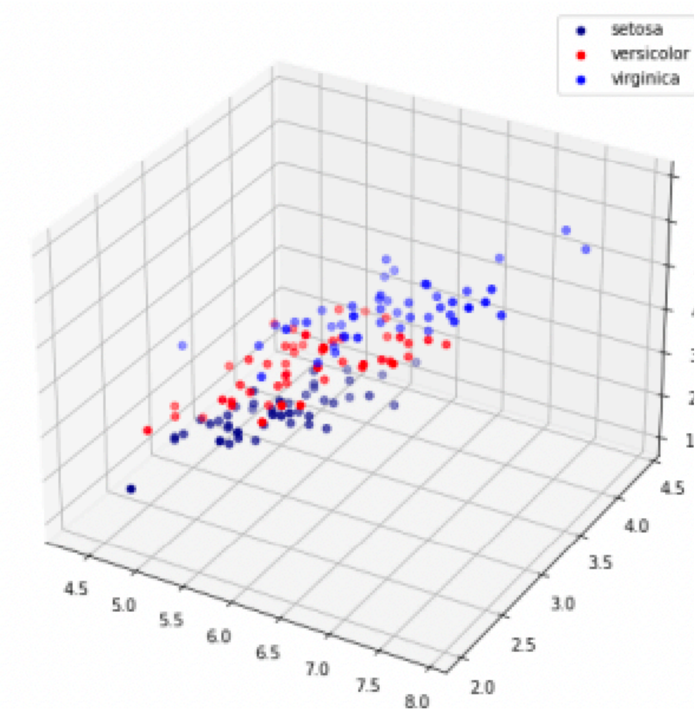


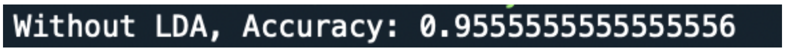
```
lda=LinearDiscriminantAnalysis(n_components=2)
lda.fit(X_train,y_train)

tree.fit(lda.transform(X_train),y_train)
acc2=tree.score(lda.transform(X_test),y_test)
X_trans=lda.transform(X)
```

5. Plotting the final results



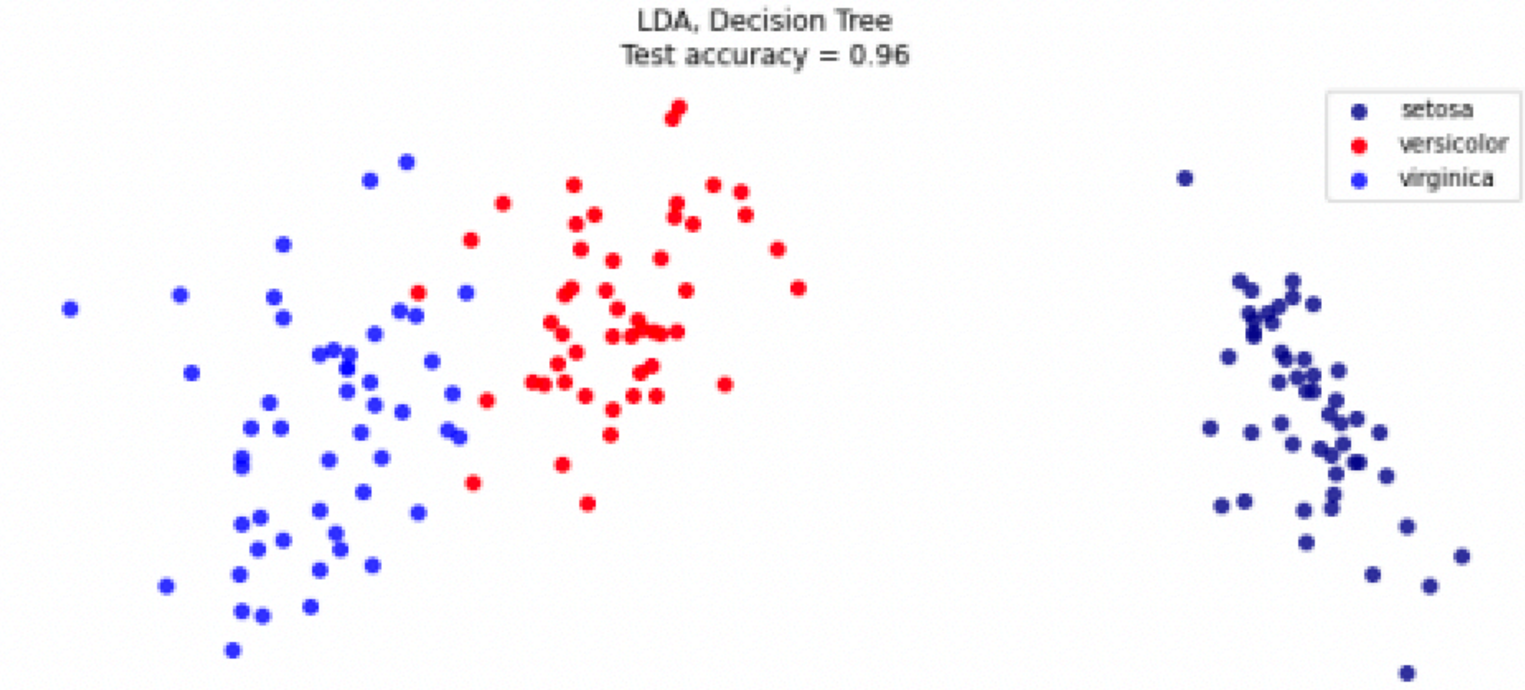
```
plt.figure(figsize=(12,5))
for color, i, target_name in zip(colors, [0, 1, 2], target_names):
    plt.scatter(X_trans[y == i, 0], X_trans[y == i, 1], alpha=.8, color=color,
                label=target_name)
plt.legend(loc='best')
plt.axis("off")
plt.title("LDA, Decision Tree\nTest accuracy = {:.2f}".format(acc2))
plt.show()
```





```
Without LDA, Accuracy: 0.9555555555555556
```

THE IRIS DATASET



ACCURACY IMPROVED AFTER PERFORMING LDA