Perform linear discriminant analysis using the Scikit-learn library on the Iris dataset.

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

from sklearn.datasets import load\_iris

from sklearn.preprocessing import StandardScaler, LabelEncoder

from sklearn.model\_selection import train\_test\_split

from sklearn.discriminant\_analysis import LinearDiscriminantAnalysis

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score, confusion\_matrix

iris = load\_iris()

dataset = pd.DataFrame(columns=iris.feature\_names,

data=iris.data)

dataset['target'] = iris.target

X = dataset.iloc[:, 0:4].values

y = dataset.iloc[:, 4].values

sc = StandardScaler()

X = sc.fit\_transform(X)

le = LabelEncoder()

y = le.fit\_transform(y)

X\_train, X\_test,\

y\_train, y\_test = train\_test\_split(X, y,

test\_size=0.2)

lda = LinearDiscriminantAnalysis(n\_components=2)

X\_train = lda.fit\_transform(X\_train, y\_train)

X\_test = lda.transform(X\_test)

plt.scatter(

X\_train[:, 0], X\_train[:, 1],

c=y\_train,

cmap='rainbow',

alpha=0.7, edgecolors='b'

)

classifier = RandomForestClassifier(max\_depth=2,

random\_state=0)

classifier.fit(X\_train, y\_train)

y\_pred = classifier.predict(X\_test)

print('Accuracy : ' + str(accuracy\_score(y\_test, y\_pred)))

conf\_m = confusion\_matrix(y\_test, y\_pred)

print(conf\_m)