CONFUSION MATRIX

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

from sklearn.tree import DecisionTreeClassifier

from sklearn import tree

%matplotlib inline

df = pd.read\_csv('/content/Iris.csv')

df.head()

X = df.drop(['Species'], axis = 1)

y = df['Species']

from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()

X = scaler.fit\_transform(X)

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

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.2, stratify = y, random\_state = 0)

classifier = DecisionTreeClassifier(criterion="entropy")

#Applying classifier on training data

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

#make prediction on test data

pred = classifier.predict(X\_test)

from sklearn.metrics import accuracy\_score

print(accuracy\_score(y\_test, pred))

#confusion matrix for Decision Tree

from sklearn.metrics import confusion\_matrix

cm = confusion\_matrix(y\_test, pred)

print(cm)

df = df.drop(['Species'], axis = 1)

from six import StringIO

from IPython.display import Image

from sklearn.tree import export\_graphviz

import pydotplus

from six import StringIO

dot\_data = StringIO()

export\_graphviz(classifier, out\_file=dot\_data, feature\_names=df.columns, filled=True, rounded=True,special\_characters=True)

graph = pydotplus.graph\_from\_dot\_data(dot\_data.getvalue())

Image(graph.create\_png())

