PRACTICAL-09

```
# SVC Classification
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.image as maping
import pandas as pd
dataset = pd.read_csv('/content/iris.csv')
dataset
dataset.head()
%matplotlib inline
img = maping.imread('/content/iris_types.jpg')
plt.figure(figsize=(5,15))
plt.axis('off')
plt.imshow(img)
X = dataset.iloc[:,:4].values
y = dataset['species'].values
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=82)
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
print('X_train', X_train)
print('X_test', X_test)
from sklearn.svm import SVC
svcclassifier = SVC(kernel = 'linear', random_state=0)
svcclassifier.fit(X_train, y_train)
y_pred = svcclassifier.predict(X_test)
print(y_pred)
y_compare = np.vstack((y_test,y_pred)).T
print(y_compare[:5,:])
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
print(cm)
a = cm.shape
corrPred = 0
falsePred = 0
for row in range(a[0]):
    for c in range(a[1]):
        if row == c:
            corrPred +=cm[row,c]
        else:
            falsePred += cm[row,c]
print('Correct predictions: ', corrPred)
print('False predictions', falsePred)
kernelLinearAccuracy = corrPred/(cm.sum())
print ('Accuracy of the SVC Clasification is: ', corrPred/(cm.sum()))
```

PRACTICAL-09

Output:

```
['virginica' 'virginica' 'setosa' 'setosa' 'setosa' 'virginica'
 'versicolor' 'versicolor' 'virginica' 'versicolor' 'versicolor'
 'virginica' 'setosa' 'setosa' 'setosa' 'setosa' 'virginica'
'versicolor'
 'setosa' 'versicolor' 'setosa' 'virginica' 'setosa' 'virginica'
 'virginica' 'versicolor' 'virginica' 'setosa' 'virginica'
'versicolor']
[['virginica' 'virginica']
 ['virginica' 'virginica']
 ['setosa' 'setosa']
 ['setosa' 'setosa']
 ['setosa' 'setosa']]
[[11  0  0]
 [ 0  8  1]
 [ 0  0 10]]
Correct predictions:  29
False predictions 1
Accuracy of the SVC Clasification is:  0.9666666666666667
```