

## **WORKSHEET – 4**

**Name: Shubham Shrivastava**  
**Section/Group: 618 'B'**  
**Branch: BE CSE (5<sup>th</sup> Semester)**

**UID: 20BCS5472**  
**Subject: ML LAB**

### **Aim:**

Implement Support Vector Machine on any data set and analyze the accuracy with Logistic regression.

### **Implement Classification and Regression Analysis:**

Support Vector Machine (SVM) is one of the most important algorithms to be learned in python to master machine learning and data science and it is basically a primary classifier method that performs classification tasks by constructing hyperplanes in the multidimensional space that separates cases of different class labels.

### **Apparatus / Simulator Used:**

Anaconda (Jupyter)

## CODE:

jupyter 1.4 WORKSHEET SHUBHAM Last Checkpoint: 2 minutes ago (autosaved)



Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted

Python 3 (ipykernel) O

Run Code

```
In [1]: import pandas as pd
```

```
In [2]: df = pd.read_csv("D:\\ML\\appleorange.csv")
x = df.iloc[:, :2]
y = df.iloc[:, 2]
```

```
In [3]: # Import train_test_split function
from sklearn.model_selection import train_test_split
# Split dataset into training set and test set
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=42)
from sklearn import svm
# Create a svm Classifier
clf = svm.SVC(kernel='linear') # Linear Kernel
# Train the model using the training sets
clf.fit(x_train, y_train)
# Predict the response for test dataset
y_pred = clf.predict(x_test)
```

```
In [4]: y_pred
```

```
Out[4]: array(['apple', 'orange', 'orange', 'orange', 'apple'], dtype=object)
```

```
In [5]: y_test
```

```
Out[5]: 9    apple
11   orange
0    orange
13   orange
5    apple
Name: class, dtype: object
```

Jupyter 1.4 WORKSHEET SHUBHAM Last Checkpoint: 2 minutes ago (autosaved)



Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted

Python 3 (ipykernel) O

Run Stop Restart Code

```
In [2]: df = pd.read_csv("D:\\ML\\appleorange.csv")
x = df.iloc[:, :2]
y = df.iloc[:, 2]
```

```
In [3]: # Import train_test_split function
from sklearn.model_selection import train_test_split
# Split dataset into training set and test set
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=42)
from sklearn import svm
# Create a svm Classifier
clf = svm.SVC(kernel='linear') # Linear Kernel
# Train the model using the training sets
clf.fit(x_train, y_train)
# Predict the response for test dataset
y_pred = clf.predict(x_test)
```

```
In [4]: y_pred
```

```
Out[4]: array(['apple', 'orange', 'orange', 'orange', 'apple'], dtype=object)
```

```
In [5]: y_test
```

```
Out[5]: 9    apple
11   orange
0    orange
13   orange
5    apple
Name: class, dtype: object
```

```
In [6]: clf.score(x_test, y_test)
```

```
Out[6]: 1.0
```