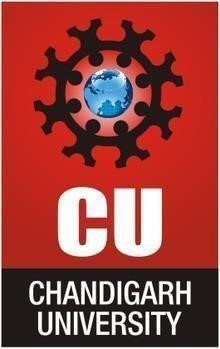
**CHANDIGARH UNIVERSITY**

**UNIVERSITY INSTITUTE OF ENGINEERING**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**



|  |  |
| --- | --- |
| **Submitted by:**  **Yash Gupta 20BCS5009** | **Submitted To:**  Ajay Kumar (E13141) |
| **Subject Name:** | Machine Learning Lab |
| **Subject Code:** | 20CSP-317 |
| **Branch:** | CSE |
| **Semester:** | 5th |

LAB INDEX

**Experiment 2.1**

1. **Aim/Overview of the practical:**

**SVM**

1. **Source Code:**

import pandas as pd

from sklearn.datasets import load\_iris

from ast import Lambda

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

from sklearn.svm import SVC # Support Vector Classifier

import matplotlib.pyplot as plt

%matplotlib inline

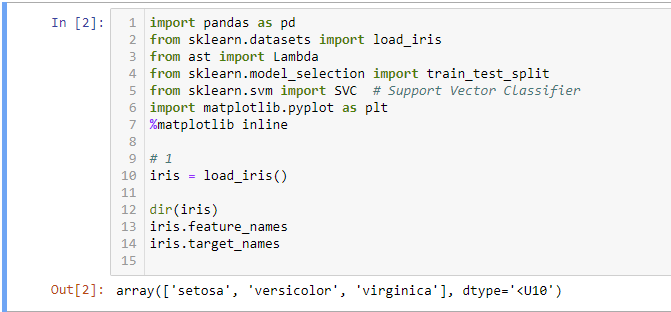
# 1

iris = load\_iris()

dir(iris)

iris.feature\_names

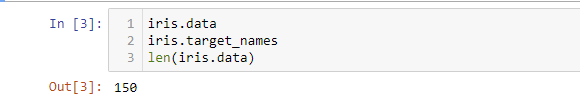
iris.target\_names



iris.data

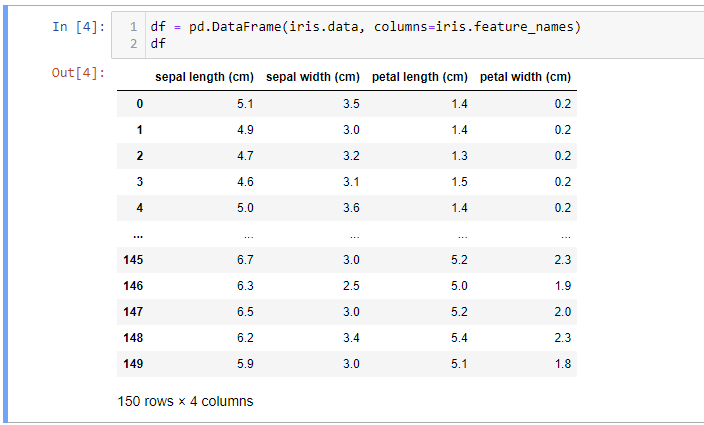
iris.target\_names

len(iris.data)



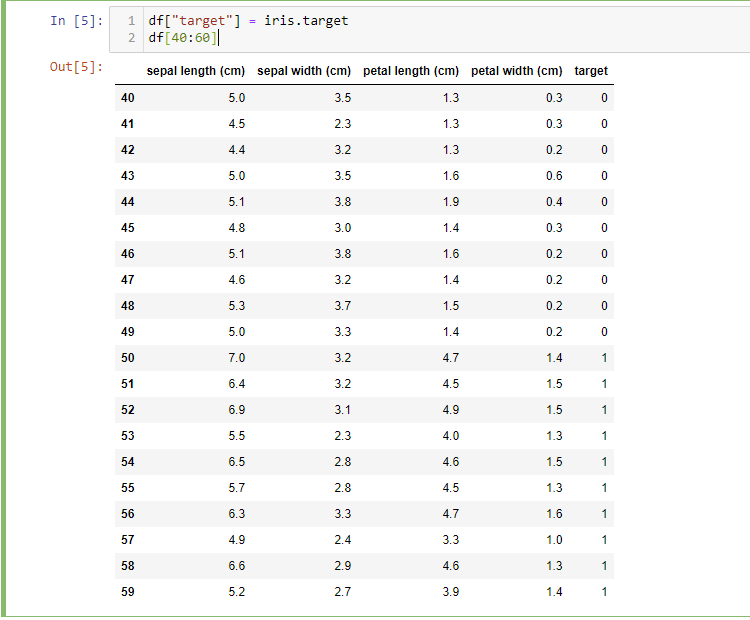
df = pd.DataFrame(iris.data, columns=iris.feature\_names)

df

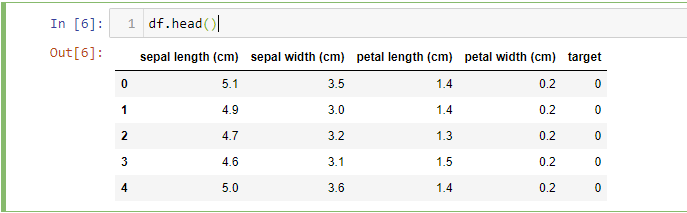


df["target"] = iris.target

df[40:60]

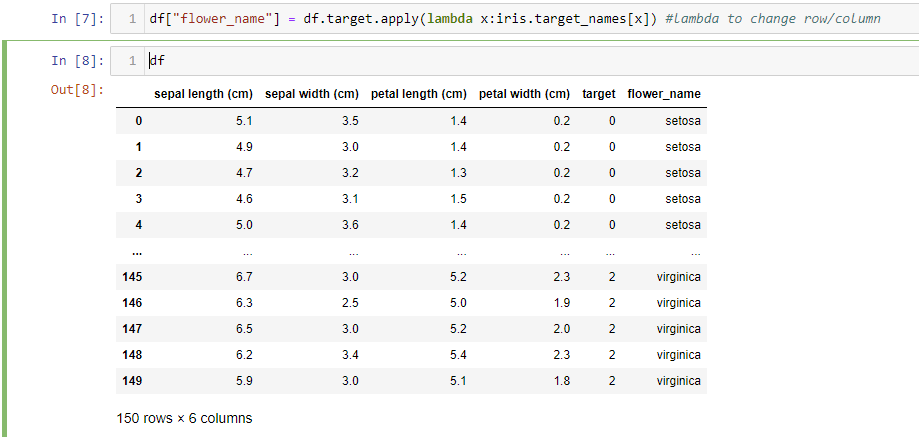


df.head()

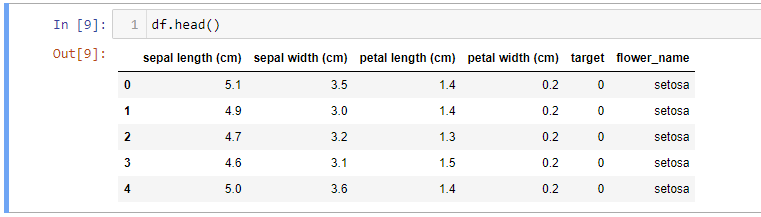


df["flower\_name"] = df.target.apply(lambda x:iris.target\_names[x]) #lambda to change row/column

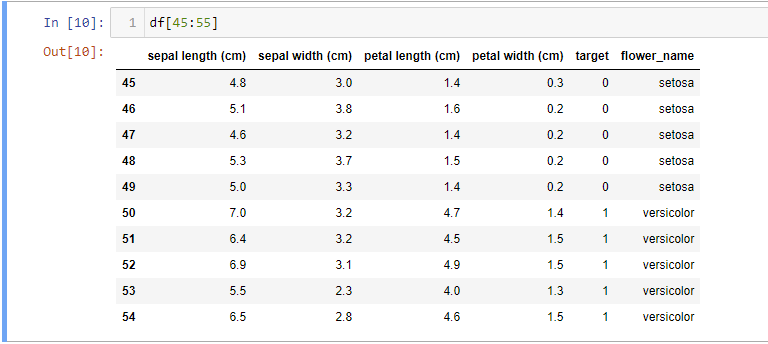
df



df.head()



df[45:55]



df0=df[:50]

df1=df[50:100]

df2=df[100:]

plt.xlabel("sepal length")

plt.ylabel("sepal breadth")

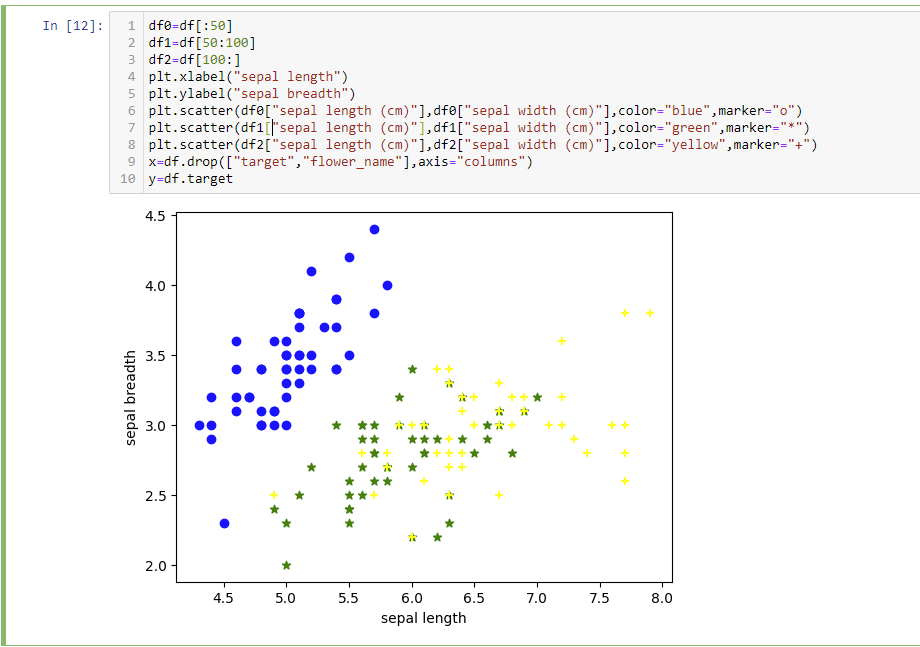
plt.scatter(df0["sepal length (cm)"],df0["sepal width (cm)"],color="blue",marker="o")

plt.scatter(df1["sepal length (cm)"],df1["sepal width (cm)"],color="green",marker="\*")

plt.scatter(df2["sepal length (cm)"],df2["sepal width (cm)"],color="yellow",marker="+")

x=df.drop(["target","flower\_name"],axis="columns")

y=df.target



df0=df[:50]

df1=df[50:100]

df2=df[100:]

plt.xlabel("sepal length")

plt.ylabel("sepal breadth")

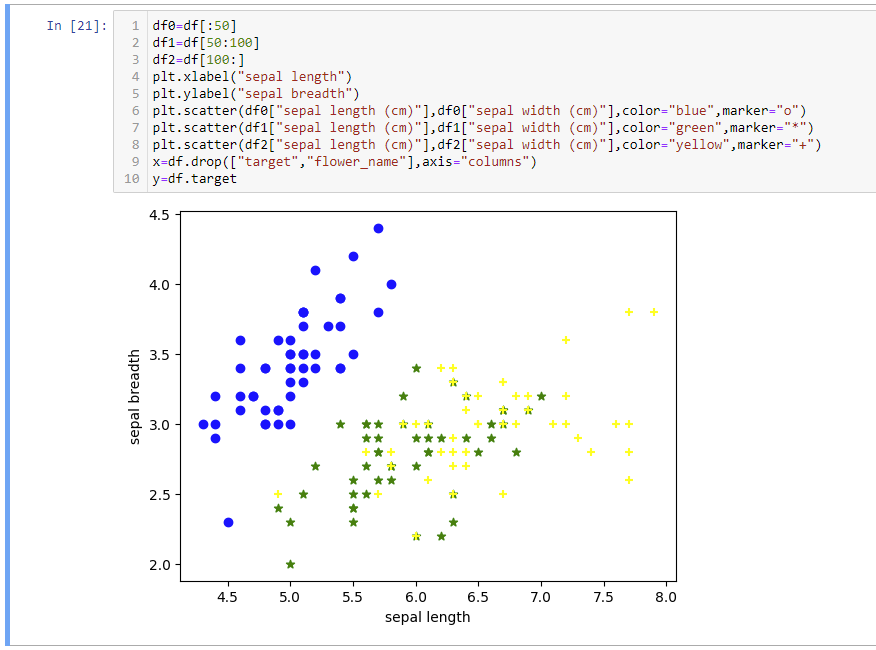
plt.scatter(df0["sepal length (cm)"],df0["sepal width (cm)"],color="blue",marker="o")

plt.scatter(df1["sepal length (cm)"],df1["sepal width (cm)"],color="green",marker="\*")

plt.scatter(df2["sepal length (cm)"],df2["sepal width (cm)"],color="yellow",marker="+")

x=df.drop(["target","flower\_name"],axis="columns")

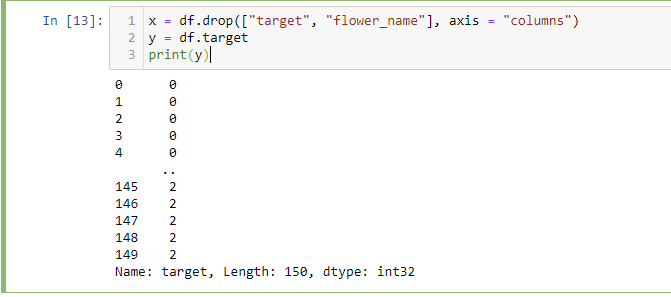
y=df.target



x = df.drop(["target", "flower\_name"], axis = "columns")

y = df.target

print(y)



x\_train, x\_test, y\_train, y\_test = train\_test\_split(x,y,test\_size=0.2)

len(y\_test)

x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.2)

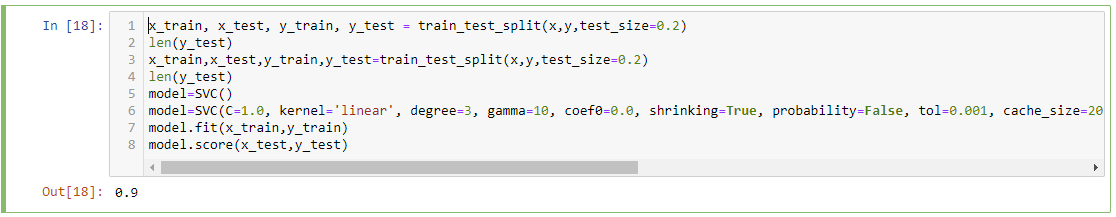
len(y\_test)

model=SVC()

model=SVC(C=1.0, kernel='linear', degree=3, gamma=10, coef0=0.0, shrinking=True, probability=False, tol=0.001, cache\_size=200, class\_weight=None, verbose=False, max\_iter=-1, decision\_function\_shape='ovr', break\_ties=False, random\_state=None)

model.fit(x\_train,y\_train)

model.score(x\_test,y\_test)



**Evaluation Grid :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No.** | **Parameters** | **Marks Obtained** | **Maximum Marks** |
| **1.** | **Student Performance  (Conduct of experiment) objectives/Outcomes.** |  | **12** |
| **2.** | **Viva Voce** |  | **10** |
| **3.** | **Submission of Work Sheet (Record)** |  | **8** |
|  | **Total** |  | **30** |