

```
In [1]: import pandas as pd
        from sklearn.datasets import load_iris
```

```
In [2]: iris=load_iris()
```

```
In [3]: df=pd.DataFrame(iris.data,columns=iris.feature_names)
```

```
In [4]: df.head()
```

Out[4]:

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

```
In [5]: iris.target
```

```
Out[5]: array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  
               0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  
               0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  
               1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  
               1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2,  
               2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,  
               2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2])
```

```
In [6]: df['target']=iris.target
```

```
In [7]: df.head(3)
```

Out[7]:

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0

```
In [8]: iris.target_names
```

```
Out[8]: array(['setosa', 'versicolor', 'virginica'], dtype='<U10')
```

```
In [9]: df['flowers_name']=df.target.apply(lambda x:iris.target_names[x])
```

```
In [10]: df.head(4)
```

```
Out[10]:
```

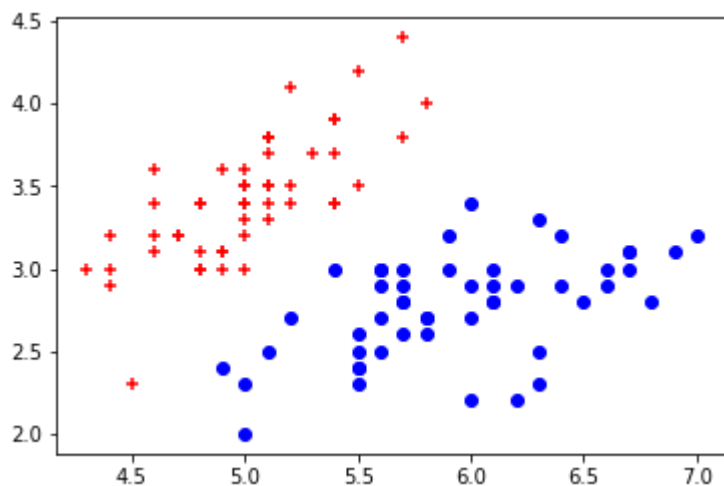
	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target	flowers_name
0	5.1	3.5	1.4	0.2	0	setosa
1	4.9	3.0	1.4	0.2	0	setosa
2	4.7	3.2	1.3	0.2	0	setosa
3	4.6	3.1	1.5	0.2	0	setosa

```
In [11]: df0=df[:50]
df1=df[50:100]
df2=df[100:150]
```

```
In [12]: import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [13]: plt.scatter(df0['sepal length (cm)'],df0['sepal width (cm)'],marker='+',color='r')
plt.scatter(df1['sepal length (cm)'],df1['sepal width (cm)'],color='blue')
```

```
Out[13]: <matplotlib.collections.PathCollection at 0x1635b6d2af0>
```



```
In [14]: x=df.drop(['target','flowers_name'],axis=1)
```

```
In [15]: y=df.target
```

```
In [16]: from sklearn.model_selection import train_test_split
```

```
In [17]: x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2)
```

```
In [18]: len(x_train)
```

```
Out[18]: 120
```

```
In [19]: len(x_test)
```

```
Out[19]: 30
```

```
In [20]: from sklearn.svm import SVC  
model=SVC()
```

```
In [21]: model.fit(x_train,y_train)
```

```
Out[21]: SVC()
```

```
In [22]: model.score(x_test,y_test)
```

```
Out[22]: 0.9666666666666667
```

```
In [23]: model.predict(x_test)
```

```
Out[23]: array([2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 2, 0, 0, 0, 0, 2, 1, 0, 2, 1, 2,  
                0, 0, 1, 2, 1, 1, 1, 0])
```

```
In [24]: model_c=SVC(C=20)
```

```
In [25]: model_c.fit(x_train,y_train)
```

```
Out[25]: SVC(C=20)
```

Best model score=100%

```
In [26]: model_c.score(x_test,y_test)
```

```
Out[26]: 1.0
```

```
In [27]: model_g=SVC(gamma=5)
```

```
In [28]: model_g.fit(x_train,y_train)
```

```
Out[28]: SVC(gamma=5)
```

```
In [29]: model_g.score(x_test,y_test)
```

```
Out[29]: 0.9666666666666667
```

```
In [30]: model_k=SVC(kernel='rbf')
```

```
In [31]: model_k.fit(x_train,y_train)
```

```
Out[31]: SVC()
```

```
In [32]: model_k.score(x_test,y_test)
```

```
Out[32]: 0.9666666666666667
```

```
In [ ]:
```