```
import pandas as pd
from sklearn.datasets import load_iris
iris = load_iris()
```

```
iris.feature_names
```

```
['sepal length (cm)',
 'sepal width (cm)',
 'petal length (cm)',
 'petal width (cm)']
```

```
iris.target_names
```

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

```
df = pd.DataFrame(iris.data,columns=iris.feature_names)
df.head()
```

	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

```
df['target'] = iris.target
df.head()
```

	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
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0

```
df[df.target==1].head()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target
50	7.0	3.2	4.7	1.4	1
51	6.4	3.2	4.5	1.5	1
52	6.9	3.1	4.9	1.5	1
53	5.5	2.3	4.0	1.3	1
54	6.5	2.8	4.6	1.5	1

```
df[df.target==2].head()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target
100	6.3	3.3	6.0	2.5	2
101	5.8	2.7	5.1	1.9	2
102	7.1	3.0	5.9	2.1	2
103	6.3	2.9	5.6	1.8	2
104	6.5	3.0	5.8	2.2	2

```
df['flower_name'] =df.target.apply(lambda x: iris.target_names[x])
df.head()
```

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target	fl
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	
3	4.6	3.1	1.5	0.2	0	
4	5.0	3.6	1.4	0.2	0	

```
df[45:55]
```

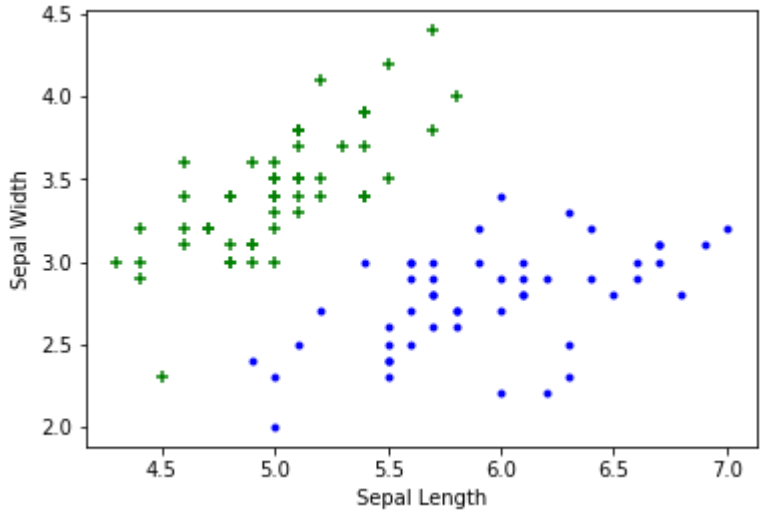
	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target	f
45	4.8	3.0	1.4	0.3	0	
46	5.1	3.8	1.6	0.2	0	
47	4.6	3.2	1.4	0.2	0	
48	5.3	3.7	1.5	0.2	0	

```
df0 = df[:50]
df1 = df[50:100]
df2 = df[100:]
```

```
import matplotlib.pyplot as plt
%matplotlib inline
```

```
plt.xlabel('Sepal Length')
plt.ylabel('Sepal Width')
plt.scatter(df0['sepal length (cm)'], df0['sepal width (cm)'],color="green",marker='+')
plt.scatter(df1['sepal length (cm)'], df1['sepal width (cm)'],color="blue",marker='.')
```

```
<matplotlib.collections.PathCollection at 0x7f9b03ed1550>
```




```
plt.xlabel('Petal Length')
plt.ylabel('Petal Width')
plt.scatter(df0['petal length (cm)'], df0['petal width (cm)'],color="green",marker='+')
plt.scatter(df1['petal length (cm)'], df1['petal width (cm)'],color="blue",marker='.')
```

```
<matplotlib.collections.PathCollection at 0x7f9b03ec5850>
```


```
from sklearn.model_selection import train_test_split
```

```
           |    + +++ +                                        |
```

```
X = df.drop(['target','flower_name'], axis='columns')
y = df.target
```

Petal Length

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
```

```
len(X_train)
```

```
120
```

```
len(X_test)
```

```
30
```

```
from sklearn.svm import SVC
model = SVC()
```

```
model.fit(X_train, y_train)
```

```
SVC(C=1.0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='rbf',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
```

```
model.score(X_test, y_test)
```

```
0.9
```

```
model.predict([[4.8,3.0,1.5,0.3]])
```

```
array([0])
```

```
model_C = SVC(C=1)
model_C.fit(X_train, y_train)
```

```
model_C.fit(X_train, y_train)
model_C.score(X_test, y_test)
```

```
0.9
```

```
model_C = SVC(C=10)
model_C.fit(X_train, y_train)
model_C.score(X_test, y_test)
```

```
0.9
```

```
model_g = SVC(gamma=10)
model_g.fit(X_train, y_train)
model_g.score(X_test, y_test)
```

```
0.9333333333333333
```

```
model_linear_kernal = SVC(kernel='linear')
model_linear_kernal.fit(X_train, y_train)
```

```
SVC(C=1.0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='linear',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
```

```
model_linear_kernal.score(X_test, y_test)
```

```
0.9666666666666667
```

✓ 0s completed at 19:04