(I) Datasets Get data from sklearn.datasets

Cancer

In [33]:

```
from sklearn.datasets import load_breast_cancer
import pandas as pd

cancerData = load_breast_cancer()
cancerDF = pd.DataFrame(cancerData.data, columns=cancerData.feature_names)

cancerDF['target']=cancerData.target
cancer_target_name = cancerData.target_names

cancer_x=cancerDF[cancerData.feature_names]
cancer_y=cancerDF['target']

cancerDF.head()
```

Out[33]:

	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	d
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	

5 rows × 31 columns

Iris

In [59]:

```
from sklearn.datasets import load_iris
import pandas as pd

irisData = load_iris()

irisDF = pd.DataFrame(irisData.data, columns=irisData.feature_names)
irisDF['target']=irisData.target

iris_x = irisDF[irisData.feature_names]
iris_y = irisDF['target']
iris_target_name =irisData.target_names
irisDF.head()
```

Out[59]:

	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

Boston

In [51]:

```
from sklearn.datasets import load_boston
import pandas as pd

bostonData = load_boston()

bostonDF = pd.DataFrame(bostonData.data, columns=bostonData.feature_names)
bostonDF['target']=bostonData.target
boston_x = bostonDF[bostonData.feature_names]
boston_y = bostonDF['target']

bostonDF.head()
```

Out[51]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33

USE PCA

In [57]:

```
from sklearn.decomposition import PCA

pca = PCA(n_components=1)
cancer_xx = pca.fit_transform(cancer_x)
cancer_xxx=pd.DataFrame(data=cancer_xx, columns=['features'])
iris_xx = pca.fit_transform(iris_x)
iris_xxx=pd.DataFrame(data=iris_xx, columns=['features'])
boston_xx = pca.fit_transform(boston_x)
boston_xxx=pd.DataFrame(data=boston_xx, columns=['features'])
```

Split dataset

Cancer

In [58]:

```
from sklearn.model_selection import train_test_split
cancer_x_train,cancer_x_test,cancer_y_train,cancer_y_test = train_test_split(can
```

IRIS

In [60]:

```
iris_x_train,iris_x_test,iris_y_train,iris_y_test = train_test_split(iris_xxx,ir
```

Boston

In [61]:

```
boston_x_train,boston_x_test,boston_y_train,boston_y_test = train_test_split(bos
```

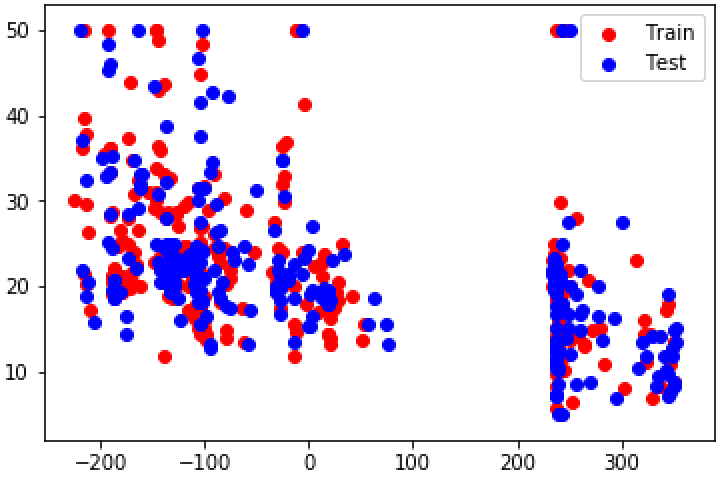
Charts

Boston

In [67]:
```
import matplotlib.pyplot as plt

plt.scatter(boston_x_train, boston_y_train,color="red", label="Train")
plt.scatter(boston_x_test, boston_y_test,color="blue", label="Test")
plt.legend()
plt.show
```

Out[67]:
```
<function matplotlib.pyplot.show(*args, **kw)>
```

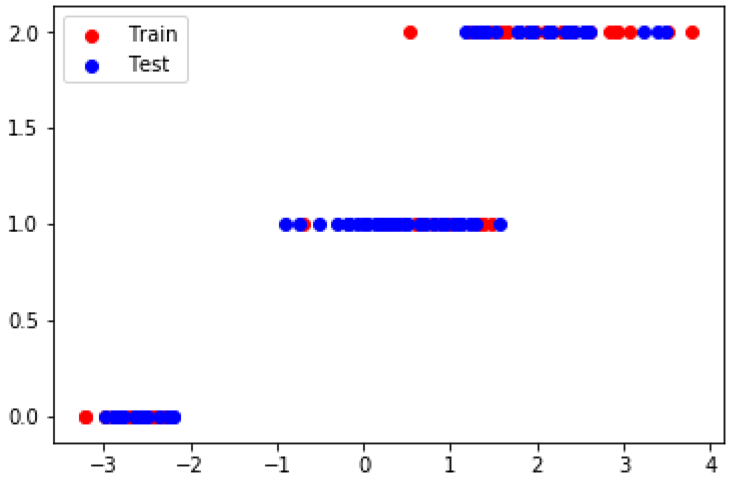


IRIS

In [68]:
```
import matplotlib.pyplot as plt

plt.scatter(iris_x_train, iris_y_train,color="red", label="Train")
plt.scatter(iris_x_test, iris_y_test,color="blue", label="Test")
plt.legend()
plt.show
```

Out[68]:
```
<function matplotlib.pyplot.show(*args, **kw)>
```

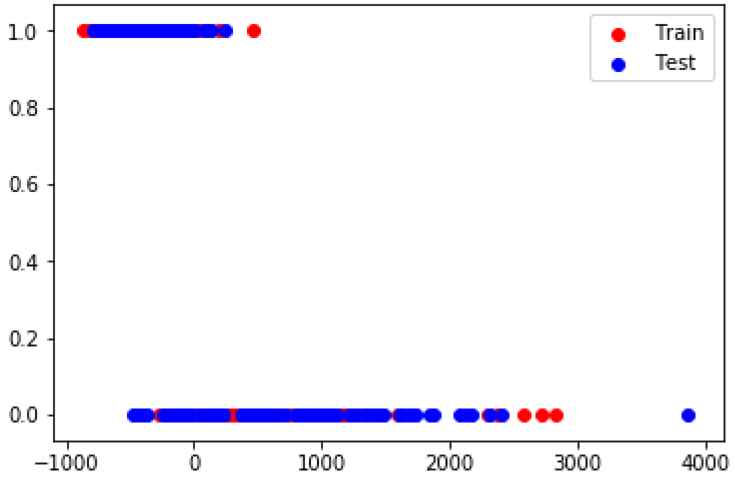


Cancer

In [69]:
```
import matplotlib.pyplot as plt

plt.scatter(cancer_x_train, cancer_y_train,color="red", label="Train")
plt.scatter(cancer_x_test, cancer_y_test,color="blue", label="Test")
plt.legend()
plt.show
```

Out[69]:
```
<function matplotlib.pyplot.show(*args, **kw)>
```



(II) Get Data from files

In [3]:
```
import pandas as pd

bankdata = pd.read_csv("bill_authentication.csv")
print(bankdata.columns)
bankdata.head(5)
```

```
Index(['Variance', 'Skewness', 'Curtosis', 'Entropy', 'Class'], dtype='object')
```

Out[3]:

	Variance	Skewness	Curtosis	Entropy	Class
0	3.62160	8.6661	-2.8073	-0.44699	0
1	4.54590	8.1674	-2.4586	-1.46210	0
2	3.86600	-2.6383	1.9242	0.10645	0
3	3.45660	9.5228	-4.0112	-3.59440	0
4	0.32924	-4.4552	4.5718	-0.98880	0

Get data without header

```
df = pandas.read_csv("file_name.csv", header=None, names=(['city']),sep=",")
```

In [ ]:
```
```