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

```
In [2]: from sklearn.datasets import load_digits
```

```
In [3]: digit=load_digits()
```

```
In [13]: df=pd.DataFrame(digit.data,digit.target)
```

```
In [15]: df['number']=digit.target
```

```
In [16]: df
```

Out[16]:

	0	1	2	3	4	5	6	7	8	9	...	55	56	57	58	59	60	61	62
0	0.0	0.0	5.0	13.0	9.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	6.0	13.0	10.0	0.0	0.0
1	0.0	0.0	0.0	12.0	13.0	5.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	11.0	16.0	10.0	0.0
2	0.0	0.0	0.0	4.0	15.0	12.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	3.0	11.0	16.0	9.0
3	0.0	0.0	7.0	15.0	13.0	1.0	0.0	0.0	0.0	8.0	...	0.0	0.0	0.0	7.0	13.0	13.0	9.0	0.0
4	0.0	0.0	0.0	1.0	11.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	2.0	16.0	4.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9	0.0	0.0	4.0	10.0	13.0	6.0	0.0	0.0	0.0	1.0	...	0.0	0.0	0.0	2.0	14.0	15.0	9.0	0.0
0	0.0	0.0	6.0	16.0	13.0	11.0	1.0	0.0	0.0	0.0	...	0.0	0.0	0.0	6.0	16.0	14.0	6.0	0.0
8	0.0	0.0	1.0	11.0	15.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	2.0	9.0	13.0	6.0	0.0
9	0.0	0.0	2.0	10.0	7.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	5.0	12.0	16.0	12.0	0.0
8	0.0	0.0	10.0	14.0	8.0	1.0	0.0	0.0	0.0	2.0	...	0.0	0.0	1.0	8.0	12.0	14.0	12.0	1.0

1797 rows × 65 columns

In [19]:
```python
x=df.drop(['number'],axis=1)
x
```

Out[19]:

	0	1	2	3	4	5	6	7	8	9	...	54	55	56	57	58	59	60	61
0	0.0	0.0	5.0	13.0	9.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	6.0	13.0	10.0	0.0
1	0.0	0.0	0.0	12.0	13.0	5.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	11.0	16.0	10.0
2	0.0	0.0	0.0	4.0	15.0	12.0	0.0	0.0	0.0	0.0	...	5.0	0.0	0.0	0.0	0.0	3.0	11.0	16.0
3	0.0	0.0	7.0	15.0	13.0	1.0	0.0	0.0	0.0	8.0	...	9.0	0.0	0.0	0.0	7.0	13.0	13.0	9.0
4	0.0	0.0	0.0	1.0	11.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	2.0	16.0	4.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
9	0.0	0.0	4.0	10.0	13.0	6.0	0.0	0.0	0.0	1.0	...	4.0	0.0	0.0	0.0	2.0	14.0	15.0	9.0
0	0.0	0.0	6.0	16.0	13.0	11.0	1.0	0.0	0.0	0.0	...	1.0	0.0	0.0	0.0	6.0	16.0	14.0	6.0
8	0.0	0.0	1.0	11.0	15.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	2.0	9.0	13.0	6.0
9	0.0	0.0	2.0	10.0	7.0	0.0	0.0	0.0	0.0	0.0	...	2.0	0.0	0.0	0.0	5.0	12.0	16.0	12.0
8	0.0	0.0	10.0	14.0	8.0	1.0	0.0	0.0	0.0	2.0	...	8.0	0.0	0.0	1.0	8.0	12.0	14.0	12.0

1797 rows × 64 columns

In [21]:
```python
y=pd.DataFrame(digit.target)
y
```

Out[21]:

	0
0	0
1	1
2	2
3	3
4	4
...	...
1792	9
1793	0
1794	8
1795	9
1796	8

1797 rows × 1 columns

In [22]:
```python
from sklearn.model_selection import train_test_split
```

In [23]:
```python
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2)
```

In [24]:
```python
from sklearn.svm import SVC
```

In [25]:
```python
model=SVC()
```

In [26]:
```python
model.fit(x_train,y_train)
```

```
c:\users\sovit\appdata\local\programs\python\python39\lib\site-packages\sklearn
\utils\validation.py:63: DataConversionWarning: A column-vector y was passed wh
en a 1d array was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  return f(*args, **kwargs)
```

Out[26]: SVC()

In [27]:
```python
model.score(x_test,y_test)
```

Out[27]: 0.9888888888888889

In [32]:
```python
model_k=SVC(kernel='rbf')
```

In [33]:
```python
model_k.fit(x_train,y_train)
```

```
c:\users\sovit\appdata\local\programs\python\python39\lib\site-packages\sklearn
\utils\validation.py:63: DataConversionWarning: A column-vector y was passed wh
en a 1d array was expected. Please change the shape of y to (n_samples, ), for
example using ravel().
  return f(*args, **kwargs)
```

Out[33]: SVC()

In [34]:
```python
model_k.score(x_test,y_test)
```

Out[34]: 0.9888888888888889

In [ ]: