In [1]:
```
#Ayush Sharma 209303312
# 8.2 Program to demonstrate SVM for classification in python
import pandas as pd
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
```

In [2]:
```
digits = load_digits()
df = pd.DataFrame(digits.data,digits.target)
df.head()
```

Out[2]:

	0	1	2	3	4	5	6	7	8	9	...	54	55	56	57	58	59	60	61	6
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	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	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	9
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	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	0

5 rows × 64 columns

In [3]:
```
df['target'] = digits.target
df.head(20)
```

Out[3]:

	0	1	2	3	4	5	6	7	8	9	...	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	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	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	...	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	...	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	2.0	16.0	4.0	
5	0.0	0.0	12.0	10.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	9.0	16.0	16.0	10.0	
6	0.0	0.0	0.0	12.0	13.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	1.0	9.0	15.0	11.0	
7	0.0	0.0	7.0	8.0	13.0	16.0	15.0	1.0	0.0	0.0	...	0.0	0.0	0.0	13.0	5.0	0.0	0.0	
8	0.0	0.0	9.0	14.0	8.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	11.0	16.0	15.0	11.0	
9	0.0	0.0	11.0	12.0	0.0	0.0	0.0	0.0	0.0	2.0	...	0.0	0.0	0.0	9.0	12.0	13.0	3.0	
0	0.0	0.0	1.0	9.0	15.0	11.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	1.0	10.0	13.0	3.0	
1	0.0	0.0	0.0	0.0	14.0	13.0	1.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	1.0	13.0	16.0	
2	0.0	0.0	5.0	12.0	1.0	0.0	0.0	0.0	0.0	0.0	...	2.0	0.0	0.0	3.0	11.0	8.0	13.0	1
3	0.0	2.0	9.0	15.0	14.0	9.0	3.0	0.0	0.0	4.0	...	0.0	0.0	2.0	12.0	12.0	13.0	11.0	
4	0.0	0.0	0.0	8.0	15.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	10.0	15.0	4.0	
5	0.0	5.0	12.0	13.0	16.0	16.0	2.0	0.0	0.0	11.0	...	0.0	0.0	4.0	15.0	16.0	2.0	0.0	
6	0.0	0.0	0.0	8.0	15.0	1.0	0.0	0.0	0.0	0.0	...	2.0	0.0	0.0	0.0	7.0	15.0	16.0	1
7	0.0	0.0	1.0	8.0	15.0	10.0	0.0	0.0	0.0	3.0	...	0.0	0.0	0.0	0.0	11.0	9.0	0.0	
8	0.0	0.0	10.0	7.0	13.0	9.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	11.0	14.0	5.0	0.0	
9	0.0	0.0	6.0	14.0	4.0	0.0	0.0	0.0	0.0	0.0	...	2.0	0.0	0.0	7.0	16.0	16.0	13.0	1

20 rows × 65 columns

In [4]:
```
df.columns[df.isna().any()]
df
```

Out[4]:

	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 [5]:
```
X_train, X_test, y_train, y_test = train_test_split(df.drop('target',axis='columns'
df.target, test_size=0.3)
model = SVC(kernel='rbf')
model.fit(X_train, y_train)
```

Out[5]:
```
▾ SVC
SVC()
```

In [6]:
```
model.score(X_test,y_test)
```

Out[6]:
```
0.987037037037037
```

In [ ]: