

In [29]: `import pandas as pd`

In [30]: `df = pd.read_csv('House_Rent_main4.csv')`
`df1=df`
`df1.sample(5)`

Out[30]:

	Unnamed: 0.1	Unnamed: 0	Posted On	BHK	Rent	Size	Floor	Area Type	Area Locality	City	Furn
1334	1334	1346	2022-05-27	1	18000	300	3 out of 10	1	Kurla West	2	
168	168	169	2022-04-30	4	40000	1300	1 out of 2	1	Ballygunge	1	
1848	1848	1862	2022-07-04	2	17000	900	4 out of 5	1	Bhuvaneswari Nagar	3	
2053	2053	2067	2022-05-19	1	12500	500	Ground out of 3	1	Seshadripuram	3	
3237	3237	3252	2022-07-08	3	25000	2064	11 out of 19	1	Old Mahabalipuram Road	5	

In [34]: `#define x and y`
`x = df1[['Rent', 'Size', 'Furnishing Status', 'Bathroom']]`
`y = df1['BHK']`

In [35]: `#import classification algos and cross validation`
`from sklearn.svm import SVC`
`from sklearn.tree import DecisionTreeClassifier`
`from sklearn.linear_model import LogisticRegression`
`from sklearn.ensemble import RandomForestClassifier`
`from sklearn.neighbors import KNeighborsClassifier`
`from sklearn.metrics import accuracy_score`
`from sklearn.model_selection import cross_val_score`

In [36]: `#Cross validation for SVC`
`cross_val_score(SVC(),x, y, cv = 5)`

Out[36]: `array([0.46080508, 0.46186441, 0.54025424, 0.51059322, 0.49416755])`

In [37]: `#Cross validation for Decision Tree`
`cross_val_score(DecisionTreeClassifier(), x, y, cv = 5)`

Out[37]: `array([0.6684322 , 0.77224576, 0.70021186, 0.74152542, 0.73064687])`

```
In [38]: #Cross validation for Logistic regression
cross_val_score(LogisticRegression(), x, y, cv = 5)
```

C:\Users\student\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\linear_model_logistic.py:458: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
<https://scikit-learn.org/stable/modules/preprocessing.html> (<https://scikit-learn.org/stable/modules/preprocessing.html>)
Please also refer to the documentation for alternative solver options:
https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression (https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)

```
n_iter_i = _check_optimize_result(
```

```
Out[38]: array([0.44385593, 0.54449153, 0.4375      , 0.48834746, 0.45917285])
```

```
In [39]: #Cross validation for Random forest
cross_val_score(RandomForestClassifier(n_estimators=50), x, y, cv = 5)
```

```
Out[39]: array([0.69809322, 0.82097458, 0.70233051, 0.76694915, 0.77306469])
```

```
In [40]: #Cross validation for KNN
cross_val_score(KNeighborsClassifier(),x, y ,cv = 5)
```

```
Out[40]: array([0.65889831, 0.71398305, 0.61334746, 0.72775424, 0.70307529])
```

```
In [41]: #Train test
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size = 0.2)
```

```
In [42]: #Model design
model = RandomForestClassifier()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
```

```
In [43]: #Confusion matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
cm
```

```
Out[43]: array([[176,  48,   3,   0],
 [ 57, 350,  43,   1],
 [   1,  56, 161,   8],
 [   0,   1,  14,  25]], dtype=int64)
```

```
In [44]: #Accuracy
print(accuracy_score(y_test, model.predict(X_test)))
```

```
0.7542372881355932
```

In []: