

Business Problem - Predict the Price of Bangalore House

Using Random Forest Regression - Supervised Machine Learning Algorithm*

Load Libraries

In [6]: `import pandas as pd`

Load Data

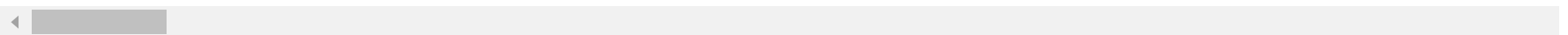
In [7]: `path = r"https://drive.google.com/uc?export=download&id=1xxDtrZKfuWQf1-6KA9XEd_eatitNPnkB"`
`df = pd.read_csv(path)`

In [8]: `df.head()`

Out[8]:

	bath	balcony	price	total_sqft_int	bhk	price_per_sqft	area_typeSuper built-up Area	area_typeBuilt- up Area	area_typePlot Area	avail
0	3.0	2.0	150.0	1672.0	3	8971.291866	1	0	0	
1	3.0	3.0	149.0	1750.0	3	8514.285714	0	1	0	
2	3.0	2.0	150.0	1750.0	3	8571.428571	1	0	0	
3	2.0	2.0	40.0	1250.0	2	3200.000000	1	0	0	
4	2.0	2.0	83.0	1200.0	2	6916.666667	0	0		1

5 rows × 108 columns



Split Data

```
In [9]: X = df.drop('price', axis=1)
        y = df['price']

        print('Shape of X = ', X.shape)
        print('Shape of y = ', y.shape)
```

```
Shape of X = (7120, 107)
Shape of y = (7120,)
```

```
In [10]: 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, random_state=51)

        print('Shape of X_train = ', X_train.shape)
        print('Shape of y_train = ', y_train.shape)
        print('Shape of X_test = ', X_test.shape)
        print('Shape of y_test = ', y_test.shape)
```

```
Shape of X_train = (5696, 107)
Shape of y_train = (5696,)
Shape of X_test = (1424, 107)
Shape of y_test = (1424,)
```

##Random Forest Regression - ML Model Training

```
In [11]: from sklearn.ensemble import RandomForestRegressor
```

```
In [12]: regressor = RandomForestRegressor(n_estimators=100, criterion='mse')  
regressor.fit(X_train, y_train)
```

```
Out[12]: RandomForestRegressor(bootstrap=True, ccp_alpha=0.0, criterion='mse',  
                                max_depth=None, max_features='auto', max_leaf_nodes=None,  
                                max_samples=None, min_impurity_decrease=0.0,  
                                min_impurity_split=None, min_samples_leaf=1,  
                                min_samples_split=2, min_weight_fraction_leaf=0.0,  
                                n_estimators=100, n_jobs=None, oob_score=False,  
                                random_state=None, verbose=0, warm_start=False)
```

```
In [13]: regressor.score(X_test, y_test)
```

```
Out[13]: 0.8971439764729167
```

```
In [14]: regressor_100 = RandomForestRegressor(n_estimators=500, criterion='mse')  
regressor_100.fit(X_train, y_train)
```

```
Out[14]: RandomForestRegressor(bootstrap=True, ccp_alpha=0.0, criterion='mse',  
                                max_depth=None, max_features='auto', max_leaf_nodes=None,  
                                max_samples=None, min_impurity_decrease=0.0,  
                                min_impurity_split=None, min_samples_leaf=1,  
                                min_samples_split=2, min_weight_fraction_leaf=0.0,  
                                n_estimators=500, n_jobs=None, oob_score=False,  
                                random_state=None, verbose=0, warm_start=False)
```

```
In [15]: regressor_100.score(X_test, y_test)
```

```
Out[15]: 0.8885044458147481
```

Predict the value of Home

```
In [18]: X_test.iloc[-1, :]
```

```
Out[18]: bath                2.000000  
balcony                    0.000000  
total_sqft_int            1566.000000  
bhk                        2.000000  
price_per_sqft            11494.252874  
...  
location_Hosur Road       0.000000  
location_Horamavu Banaswadi 0.000000  
location_Domlur           0.000000  
location_Mahadevpura       0.000000  
location_Tumkur Road      0.000000  
Name: 43, Length: 107, dtype: float64
```

```
In [19]: regressor.predict([X_test.iloc[-1, :]])
```

```
Out[19]: array([175.68])
```

```
In [20]: y_test.iloc[-1]
```

```
Out[20]: 180.0
```

```
In [21]: y_pred = regressor.predict(X_test)  
y_pred
```

```
Out[21]: array([ 79.6861,  39.9732, 119.42 , ...,  33.4522,  63.3733, 175.68  ])
```

In [22]: y_test

```
Out[22]: 2435      80.00
          3113      40.00
          426     120.00
          1124      79.00
          1161      45.00
          ...
          2078      28.34
          6855      84.00
          4381      32.00
          3862      63.00
           43     180.00
          Name: price, Length: 1424, dtype: float64
```

Ab milenge next tutorial me,Tab tak ke liye SIKHATE SIKHATE kuch IMPLEMENT karte raho, Thank You.....:-)