```
!pip install pycaret
```

```
Collecting pycaret
  Downloading pycaret-2.3.10-py3-none-any.whl (320 kB)
     |████████████████████████████████| 320 kB 27.5 MB/s
Collecting Boruta
  Downloading Boruta-0.3-py3-none-any.whl (56 kB)
     |████████████████████████████████| 56 kB 4.4 MB/s
Requirement already satisfied: yellowbrick>=1.0.1 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: wordcloud in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: ipywidgets in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: numba<0.55 in /usr/local/lib/python3.7/dist-packages (fr
Collecting pyLDAvis
  Downloading pyLDAvis-3.3.1.tar.gz (1.7 MB)
     |████████████████████████████████| 1.7 MB 50.6 MB/s
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Installing backend dependencies ... done
    Preparing wheel metadata ... done
Collecting pandas-profiling>=2.8.0
  Downloading pandas_profiling-3.1.0-py2.py3-none-any.whl (261 kB)
     |████████████████████████████████| 261 kB 44.3 MB/s
Collecting scikit-plot
  Downloading scikit_plot-0.3.7-py3-none-any.whl (33 kB)
Collecting kmodes>=0.10.1
  Downloading kmodes-0.12.1-py2.py3-none-any.whl (20 kB)
Requirement already satisfied: cufflinks>=0.17.0 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: textblob in /usr/local/lib/python3.7/dist-packages (from
Collecting scikit-learn==0.23.2
  Downloading scikit_learn-0.23.2-cp37-cp37m-manylinux1_x86_64.whl (6.8 MB)
     |████████████████████████████████| 6.8 MB 41.8 MB/s
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (from py
Requirement already satisfied: plotly>=4.4.1 in /usr/local/lib/python3.7/dist-packages (
Collecting lightgbm>=2.3.1
  Downloading lightgbm-3.3.2-py3-none-manylinux1_x86_64.whl (2.0 MB)
     |████████████████████████████████| 2.0 MB 56.0 MB/s
Collecting umap-learn
  Downloading umap-learn-0.5.3.tar.gz (88 kB)
     |████████████████████████████████| 88 kB 5.0 MB/s
Collecting mlxtend>=0.17.0
  Downloading mlxtend-0.19.0-py2.py3-none-any.whl (1.3 MB)
     |████████████████████████████████| 1.3 MB 50.6 MB/s
Requirement already satisfied: nltk in /usr/local/lib/python3.7/dist-packages (from pyca
Requirement already satisfied: joblib in /usr/local/lib/python3.7/dist-packages (from py
Requirement already satisfied: matplotlib in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: pyyaml<6.0.0 in /usr/local/lib/python3.7/dist-packages (f
Collecting imbalanced-learn==0.7.0
  Downloading imbalanced_learn-0.7.0-py3-none-any.whl (167 kB)
     |████████████████████████████████| 167 kB 57.0 MB/s
Requirement already satisfied: gensim<4.0.0 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: spacy<2.4.0 in /usr/local/lib/python3.7/dist-packages (fr
Collecting pyod
  Downloading pyod-0.9.9.tar.gz (116 kB)
     |████████████████████████████████| 116 kB 56.6 MB/s
Requirement already satisfied: IPython in /usr/local/lib/python3.7/dist-packages (from p
Requirement already satisfied: scipy<=1.5.4 in /usr/local/lib/python3.7/dist-packages (f
Collecting mlflow
  Downloading mlflow-1.25.1-py3-none-any.whl (16.8 MB)
```

```
     |████████████████████████████████| 16.8 MB 42.3 MB/s
Requirement already satisfied: seaborn in /usr/local/lib/python3.7/dist-packages (from p
Requirement already satisfied: numpy>=1.13.3 in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: setuptools>=34.4.1 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: colorlover>=0.2.1 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: six>=1.9.0 in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: smart-open>=1.2.1 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: pexpect in /usr/local/lib/python3.7/dist-packages (from I
Requirement already satisfied: traitlets>=4.2 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: prompt-toolkit<2.0.0,>=1.0.4 in /usr/local/lib/python3.7/
Requirement already satisfied: decorator in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: pickleshare in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: pygments in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: simplegeneric>0.8 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: widgetsnbextension~=3.6.0 in /usr/local/lib/python3.7/dis
Requirement already satisfied: ipython-genutils~=0.2.0 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: jupyterlab-widgets>=1.0.0 in /usr/local/lib/python3.7/dis
Requirement already satisfied: nbformat>=4.2.0 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: ipykernel>=4.5.1 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: jupyter-client in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: tornado>=4.0 in /usr/local/lib/python3.7/dist-packages (t
Requirement already satisfied: wheel in /usr/local/lib/python3.7/dist-packages (from lig
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/li
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.7/dist-packages (t
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: jsonschema>=2.6 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: jupyter-core in /usr/local/lib/python3.7/dist-packages (t
Requirement already satisfied: fastjsonschema in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: importlib-metadata in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: pyrsistent!=0.17.0,!=0.17.1,!=0.17.2,>=0.14.0 in /usr/loc
Requirement already satisfied: attrs>=17.4.0 in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: importlib-resources>=1.4.0 in /usr/local/lib/python3.7/di
Requirement already satisfied: zipp>=3.1.0 in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: llvmlite<0.35,>=0.34.0.dev0 in /usr/local/lib/python3.7/d
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (t
Collecting pyyaml<6.0.0
  Downloading PyYAML-5.4.1-cp37-cp37m-manylinux1_x86_64.whl (636 kB)
     |████████████████████████████████| 636 kB 50.9 MB/s
Requirement already satisfied: missingno>=0.4.2 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: tqdm>=4.48.2 in /usr/local/lib/python3.7/dist-packages (t
Collecting pydantic>=1.8.1
  Downloading pydantic-1.9.0-cp37-cp37m-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (
     |████████████████████████████████| 10.9 MB 42.3 MB/s
Collecting phik>=0.11.1
  Downloading phik-0.12.2-cp37-cp37m-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (690
     |████████████████████████████████| 690 kB 44.8 MB/s
Collecting requests>=2.24.0
  Downloading requests-2.27.1-py2.py3-none-any.whl (63 kB)
     |████████████████████████████████| 63 kB 1.4 MB/s
Collecting multimethod>=1.4
  Downloading multimethod-1.8-py3-none-any.whl (9.8 kB)
Collecting joblib
  Downloading joblib-1.0.1-py3-none-any.whl (303 kB)
     |████████████████████████████████| 303 kB 47.4 MB/s
```

```
     |████████████████████████████████| 303 kB 47.4 MB/s
Requirement already satisfied: jinja2>=2.11.1 in /usr/local/lib/python3.7/dist-packages
Collecting tangled-up-in-unicode==0.1.0
  Downloading tangled_up_in_unicode-0.1.0-py3-none-any.whl (3.1 MB)
     |████████████████████████████████| 3.1 MB 55.3 MB/s
Collecting htmlmin>=0.1.12
  Downloading htmlmin-0.1.12.tar.gz (19 kB)
Collecting visions[type_image_path]==0.7.4
  Downloading visions-0.7.4-py3-none-any.whl (102 kB)
     |████████████████████████████████| 102 kB 8.1 MB/s
Requirement already satisfied: markupsafe~=2.0.1 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: networkx>=2.4 in /usr/local/lib/python3.7/dist-packages (
Collecting imagehash
  Downloading ImageHash-4.2.1.tar.gz (812 kB)
     |████████████████████████████████| 812 kB 49.5 MB/s
Requirement already satisfied: Pillow in /usr/local/lib/python3.7/dist-packages (from vi
Collecting scipy<=1.5.4
  Downloading scipy-1.5.4-cp37-cp37m-manylinux1_x86_64.whl (25.9 MB)
     |████████████████████████████████| 25.9 MB 1.4 MB/s
Requirement already satisfied: tenacity>=6.2.0 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: wcwidth in /usr/local/lib/python3.7/dist-packages (from p
Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/lib/python3.7/dis
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: wasabi<1.1.0,>=0.4.0 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: srsly<1.1.0,>=1.0.2 in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python3.7/dist-pa
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/python3.7/dis
Requirement already satisfied: catalogue<1.1.0,>=0.0.7 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: plac<1.2.0,>=0.9.6 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: thinc==7.4.0 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: blis<0.5.0,>=0.4.0 in /usr/local/lib/python3.7/dist-packa
Requirement already satisfied: notebook>=4.4.1 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: terminado>=0.8.1 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: Send2Trash in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: nbconvert in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: pyzmq>=13 in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: ptyprocess in /usr/local/lib/python3.7/dist-packages (fro
Collecting yellowbrick>=1.0.1
  Downloading yellowbrick-1.3.post1-py3-none-any.whl (271 kB)
     |████████████████████████████████| 271 kB 55.5 MB/s
Collecting numpy>=1.13.3
  Downloading numpy-1.19.5-cp37-cp37m-manylinux2010_x86_64.whl (14.8 MB)
     |████████████████████████████████| 14.8 MB 41.6 MB/s
Requirement already satisfied: PyWavelets in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: Flask in /usr/local/lib/python3.7/dist-packages (from mlf
Collecting alembic
  Downloading alembic-1.7.7-py3-none-any.whl (210 kB)
     |████████████████████████████████| 210 kB 52.8 MB/s
Collecting docker>=4.0.0
  Downloading docker-5.0.3-py2.py3-none-any.whl (146 kB)
     |████████████████████████████████| 146 kB 56.8 MB/s
Requirement already satisfied: click>=7.0 in /usr/local/lib/python3.7/dist-packages (fro
Collecting gitpython>=2.1.0
  Downloading GitPython-3.1.27-py3-none-any.whl (181 kB)
```

```
     |████████████████████████████████| 181 kB 48.8 MB/s
Requirement already satisfied: sqlparse>=0.3.1 in /usr/local/lib/python3.7/dist-packages
Collecting querystring-parser
  Downloading querystring_parser-1.2.4-py2.py3-none-any.whl (7.9 kB)
Requirement already satisfied: cloudpickle in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: packaging in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: entrypoints in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: sqlalchemy in /usr/local/lib/python3.7/dist-packages (fro
Collecting gunicorn
  Downloading gunicorn-20.1.0-py3-none-any.whl (79 kB)
     |████████████████████████████████| 79 kB 7.0 MB/s
Collecting databricks-cli>=0.8.7
  Downloading databricks-cli-0.16.6.tar.gz (62 kB)
     |████████████████████████████████| 62 kB 734 kB/s
Requirement already satisfied: protobuf>=3.7.0 in /usr/local/lib/python3.7/dist-packages
Collecting prometheus-flask-exporter
  Downloading prometheus_flask_exporter-0.20.1-py3-none-any.whl (18 kB)
Collecting pyjwt>=1.7.0
  Downloading PyJWT-2.3.0-py3-none-any.whl (16 kB)
Requirement already satisfied: oauthlib>=3.1.0 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: tabulate>=0.7.7 in /usr/local/lib/python3.7/dist-packages
Collecting websocket-client>=0.32.0
  Downloading websocket_client-1.3.2-py3-none-any.whl (54 kB)
     |████████████████████████████████| 54 kB 2.4 MB/s
Collecting gitdb<5,>=4.0.1
  Downloading gitdb-4.0.9-py3-none-any.whl (63 kB)
     |████████████████████████████████| 63 kB 1.5 MB/s
Collecting smmap<6,>=3.0.1
  Downloading smmap-5.0.0-py3-none-any.whl (24 kB)
Collecting Mako
  Downloading Mako-1.2.0-py3-none-any.whl (78 kB)
     |████████████████████████████████| 78 kB 6.0 MB/s
Requirement already satisfied: greenlet!=0.4.17 in /usr/local/lib/python3.7/dist-package
Requirement already satisfied: Werkzeug<2.0,>=0.15 in /usr/local/lib/python3.7/dist-pack
Requirement already satisfied: itsdangerous<2.0,>=0.24 in /usr/local/lib/python3.7/dist-
Requirement already satisfied: defusedxml in /usr/local/lib/python3.7/dist-packages (fro
Requirement already satisfied: bleach in /usr/local/lib/python3.7/dist-packages (from nb
Requirement already satisfied: testpath in /usr/local/lib/python3.7/dist-packages (from
Requirement already satisfied: mistune<2,>=0.8.1 in /usr/local/lib/python3.7/dist-packag
Requirement already satisfied: pandocfilters>=1.4.1 in /usr/local/lib/python3.7/dist-pac
Requirement already satisfied: webencodings in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: prometheus-client in /usr/local/lib/python3.7/dist-packag
Collecting pyLDAvis
  Downloading pyLDAvis-3.3.0.tar.gz (1.7 MB)
     |████████████████████████████████| 1.7 MB 62.9 MB/s
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Installing backend dependencies ... done
    Preparing wheel metadata ... done
  Downloading pyLDAvis-3.2.2.tar.gz (1.7 MB)
     |████████████████████████████████| 1.7 MB 65.3 MB/s
Requirement already satisfied: numexpr in /usr/local/lib/python3.7/dist-packages (from p
Requirement already satisfied: future in /usr/local/lib/python3.7/dist-packages (from py
Collecting funcy
  Downloading funcy-1.17-py2.py3-none-any.whl (33 kB)
Requirement already satisfied: statsmodels in /usr/local/lib/python3.7/dist-packages (fr
Requirement already satisfied: patsy>=0.4.0 in /usr/local/lib/python3.7/dist-packages (f
```

```
Requirement already satisfied: patsy>=0.4.0 in /usr/local/lib/python3.7/dist-packages (f
Collecting pynndescent>=0.5
  Downloading pynndescent-0.5.6.tar.gz (1.1 MB)
     |████████████████████████████████| 1.1 MB 65.4 MB/s
Building wheels for collected packages: htmlmin, imagehash, databricks-cli, pyLDAvis, py
  Building wheel for htmlmin (setup.py) ... done
  Created wheel for htmlmin: filename=htmlmin-0.1.12-py3-none-any.whl size=27098 sha256=
  Stored in directory: /root/.cache/pip/wheels/70/e1/52/5b14d250ba868768823940c3229e9950
  Building wheel for imagehash (setup.py) ... done
  Created wheel for imagehash: filename=ImageHash-4.2.1-py2.py3-none-any.whl size=295206
  Stored in directory: /root/.cache/pip/wheels/4c/d5/59/5e3e297533ddb09407769762985d1341
  Building wheel for databricks-cli (setup.py) ... done
  Created wheel for databricks-cli: filename=databricks_cli-0.16.6-py3-none-any.whl size
  Stored in directory: /root/.cache/pip/wheels/96/c1/f8/d75a22e789ab6a4dff11f18338c3af43
  Building wheel for pyLDAvis (setup.py) ... done
  Created wheel for pyLDAvis: filename=pyLDAvis-3.2.2-py2.py3-none-any.whl size=135617 s
  Stored in directory: /root/.cache/pip/wheels/f8/b1/9b/560ac1931796b7303f7b517b949d2d31
  Building wheel for pyod (setup.py) ... done
  Created wheel for pyod: filename=pyod-0.9.9-py3-none-any.whl size=139325 sha256=6f018a
  Stored in directory: /root/.cache/pip/wheels/68/32/f0/0dc3050775e77b6661a116b70817b02b
  Building wheel for umap-learn (setup.py) ... done
  Created wheel for umap-learn: filename=umap_learn-0.5.3-py3-none-any.whl size=82829 sh
  Stored in directory: /root/.cache/pip/wheels/b3/52/a5/1fd9e3e76a7ab34f134c07469cd6f16e
  Building wheel for pynndescent (setup.py) ... done
  Created wheel for pynndescent: filename=pynndescent-0.5.6-py3-none-any.whl size=53943
  Stored in directory: /root/.cache/pip/wheels/03/f1/56/f80d72741e400345b5a5b50ec3d929ac
Successfully built htmlmin imagehash databricks-cli pyLDAvis pyod umap-learn pynndescent
Installing collected packages: numpy, tangled-up-in-unicode, smmap, scipy, multimethod,
  Attempting uninstall: numpy
    Found existing installation: numpy 1.21.6
    Uninstalling numpy-1.21.6:
      Successfully uninstalled numpy-1.21.6
  Attempting uninstall: scipy
```

```
# importing all the libraries
import pandas as pd
pd.set_option('max_columns',None)
pd.set_option('max_rows',81)
import numpy as np
import random as rnd
import seaborn as sns
sns.set_style('darkgrid')
import matplotlib.pyplot as plt
%matplotlib inline
from sklearn.metrics import f1_score
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split,KFold,cross_val_score
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC, LinearSVC , LinearSVR
from sklearn.ensemble import RandomForestClassifier,RandomForestRegressor,AdaBoostRegressor,A
from sklearn.neighbors import KNeighborsClassifier,KNeighborsRegressor
from sklearn.naive_bayes import GaussianNB
from sklearn.linear_model import Perceptron
from sklearn.linear_model import SGDClassifier,SGDRegressor
from sklearn.tree import DecisionTreeClassifier,DecisionTreeRegressor
```

```
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import OrdinalEncoder,StandardScaler,LabelEncoder,OneHotEncoder
from sklearn.impute import SimpleImputer
from sklearn.metrics import classification_report
from sklearn.preprocessing import MinMaxScaler,StandardScaler
import xgboost as xgb
from sklearn.metrics import mean_squared_error
import scipy.stats

from pycaret.regression import setup,compare_models
#from catboost import CatBoostRegressor

from sklearn.linear_model import BayesianRidge,HuberRegressor,Ridge,OrthogonalMatchingPursuit
from lightgbm import LGBMRegressor

from sklearn.ensemble import GradientBoostingRegressor,ExtraTreesRegressor

from xgboost import XGBRegressor
```

```
albumentations 0.1.12 requires imgaug<0.2.7,>=0.2.5, but you have imgaug 0.2.9 which is
```

```
df = pd.read_csv('/content/House Price Prediction in Bangaluru/Train.csv')
df_test = pd.read_csv('/content/House Price Prediction in Bangaluru/Train.csv')
```

```
You must restart the runtime in order to use newly installed versions.
```

```
df.head(50)
```

	area_type	availability	location	size	society	total_sqft	bath	bal
0	Super built-up Area	19-Dec	Electronic City Phase II	2 BHK	Coomee	1056	2.0	
1	Plot Area	Ready To Move	Chikka Tirupathi	4 Bedroom	Theanmp	2600	5.0	
2	Built-up Area	Ready To Move	Uttarahalli	3 BHK	NaN	1440	2.0	
3	Super built-up Area	Ready To Move	Lingadheeranahalli	3 BHK	Soiewre	1521	3.0	
4	Super built-up Area	Ready To Move	Kothanur	2 BHK	NaN	1200	2.0	
5	Super built-up Area	Ready To Move	Whitefield	2 BHK	DuenaTa	1170	2.0	
6	Super built-up Area	18-May	Old Airport Road	4 BHK	Jaades	2732	4.0	
7	Super built-up Area	Ready To Move	Rajaji Nagar	4 BHK	Brway G	3300	4.0	
8	Super built-up Area	Ready To Move	Marathahalli	3 BHK	NaN	1310	3.0	
9	Plot Area	Ready To Move	Gandhi Bazar	6 Bedroom	NaN	1020	6.0	
10	Super built-up Area	18-Feb	Whitefield	3 BHK	NaN	1800	2.0	
11	Plot Area	Ready To Move	Whitefield	4 Bedroom	Prrry M	2785	5.0	
12	Super built-up Area	Ready To Move	7th Phase JP Nagar	2 BHK	Shncyes	1000	2.0	
13	Built-up Area	Ready To Move	Gottigere	2 BHK	NaN	1100	2.0	
14	Plot Area	Ready To Move	Sarjapur	3 Bedroom	Skityer	2250	3.0	
15	Super built-up Area	Ready To Move	Mysore Road	2 BHK	PrntaEn	1175	2.0	

16	Super built-up Area	Ready To Move	Bisuvanahalli	3 BHK	Prityel	1180	3.0	
17	Super built-up Area	Ready To Move	Raja Rajeshwari Nagar	3 BHK	GrrvaGr	1540	3.0	

```
df.shape
```

```
(13320, 9)
```

```
df_test
```

	area_type	availability	location	size	society	total_sqft	bath	bal
0	Super built-up Area	Ready To Move	Brookefield	2 BHK	Roeekbl	1225	2.0	
1	Plot Area	Ready To Move	Akshaya Nagar	9 Bedroom	NaN	2400	9.0	
2	Plot Area	18-Apr	Hennur Road	4 Bedroom	Saandtt	1650	5.0	
3	Super built-up Area	Ready To Move	Kodichikkanahalli	3 BHK	Winerri	1322	3.0	
4	Super built-up Area	Ready To Move	Konanakunte	2 BHK	AmageSa	1161	2.0	
...	...	...	...	...	...	...	...	
1475	Super built-up	Ready To	Vittasandra	2 BHK	Prlla C	1246	2.0	

```
target = df['price']

test_columns = df_test[['area_type','availability','location','size','society','total_sqft','

df1 = df.drop(['price'],axis=1)
df_test1 = df_test.drop(['price'],axis=1)

data = pd.concat([df1,df_test1],axis=0).reset_index(drop=True)
```

```
data
```

	area_type	availability	location	size	society	total_sqft	bath	b
0	Super built-up Area	19-Dec	Electronic City Phase II	2 BHK	Coomee	1056	2.0	
1	Plot Area	Ready To Move	Chikka Tirupathi	4 Bedroom	Theanmp	2600	5.0	
2	Built-up Area	Ready To Move	Uttarahalli	3 BHK	NaN	1440	2.0	
3	Super built-up Area	Ready To Move	Lingadheeranahalli	3 BHK	Soiewre	1521	3.0	
4	Super built-up Area	Ready To Move	Kothanur	2 BHK	NaN	1200	2.0	
...	...	...	...	...	...	...	...	...
	Super	Ready To						

```
data.shape
```

```
(14800, 8)
```

```
# Check null values
data.isnull().sum()
```

```
area_type          0
availability       0
location           1
size              18
society         6128
total_sqft         0
bath              80
balcony          678
dtype: int64
```

EDA(Explanatory Data Analysis)

total_sqft

```
len(data['total_sqft'].unique())
```

```
2221
```

```
data['total_sqft'].dtype
```

```
dtype('O')
```

```
data['total_sqft'].unique()
```

```
array(['1056', '2600', '1440', ..., '3383', '5149', '6750'], dtype=object)
```

```
range_type = (data['total_sqft'].str.len()>4)
```

```
array = data[range_type]['total_sqft'].unique()
```

```
# 34.46Sq. Meter
# 1330.74
# 2100 - 2850
# 4125Perch
# 5.31Acres
# 547.34 - 827.31
# 1574Sq. Yards
# 3Cents
# 2.09Acres
#38Guntha
```

```
array
```

```
'596 - 804', '1776.42', '11338', '30000', '1255 - 1863',
'1300 - 1405', '1500 - 2400', '117Sq. Yards', '934 - 1437',
'980 - 1030', '1564 - 1850', '1452.19', '1446 - 1506', '1419.59',
'461.82', '1070 - 1315', '3040Sq. Meter', '500Sq. Yards',
'1020.07', '2806 - 3019', '613 - 648', '1430 - 1630', '1451.5',
'704 - 730', '1584.01', '1205.47', '1482 - 1846', '1689.28',
'2805 - 3565', '10000', '1819.18', '3293 - 5314', '1210 - 1477',
'3369 - 3464', '1125 - 1500', '167Sq. Meter', '1076 - 1199',
'381 - 535', '1627.86', '2215 - 2475', '524 - 894', '1369.1',
'1053.4', '540 - 670', '315Sq. Yards', '1650 - 2538',
'2725 - 3250', '1732.46', '910.2', '1974 - 2171', '2006.8',
'888 - 1290', '1360 - 1890', '2466 - 2856', '660 - 700',
'385 - 440', '2100 - 5405', '770 - 841', '42000', '3Cents',
'1310 - 1615', '36000', '1113.12', '1782 - 2000', '1548.3',
'1525.84', '188.89Sq. Yards', '1469 - 1766', '204Sq. Meter',
'1255 - 1350', '870 - 1080', '45Sq. Yards', '2777.29',
'133.3Sq. Yards', '1554.3', '3405.1', '2580 - 2591', '1652.5',
'2563 - 2733', '605 - 624', '4260 - 4408', '1349 - 3324',
'78.03Sq. Meter', '1208.51', '3300 - 3335', '1180 - 1630',
'1660.4', '1365 - 1700', '3500 - 3600', '122Sq. Yards'
```

```
       '1000.4', '1505 - 1760', '3980 - 5000', '1225Sq. Yards',
       '84.53Sq. Meter', '2.09Acres', '1482 - 1684', '981 - 1249',
       '1100 - 1225', '1565 - 1595', '24Guntha', '1270 - 1275',
       '840 - 1010', '697Sq. Meter', '655 - 742', '800 - 2660', '1107.83',
       '1408 - 1455', '4050 - 4075', '1266.67', '942 - 1117', '1777.26',
       '1750 - 2640', '1390 - 1600', '598 - 958', '1500Cents', '1567.2',
       '26136', '132Sq. Yards', '1691.2', '1010 - 1300', '11000',
       '2Acres', '3103 - 3890', '1450 - 1950', '2274.24', '1100Sq. Meter',
       '947.55', '15Acres', '3301.8', '1450 - 1595', '763 - 805',
       '3307 - 3464', '1.26Acres', '620 - 934', '1542.14', '2144.6',
       '2462 - 2467', '540 - 740', '10200', '1618 - 1929', '20000',
       '3508 - 4201', '4900 - 4940', '1610 - 1880', '755 - 770', '1181.7',
       '664 - 722', '151.11Sq. Yards', '700 - 900', '596 - 861',
       '1925 - 2680', '615 - 985', '540 - 565', '2087.01', '750 - 800',
       '1660 - 1805', '1719.3', '1410 - 1710', '1079 - 1183',
       '2800 - 2870', '1230 - 1290', '943 - 1220', '2041 - 2090',
       '1234.6', '1763.25', '527 - 639', '2249.81', '1Grounds',
       '1160 - 1315', '706 - 716', '2940Sq. Yards', '1791 - 4000',
       '45.06Sq. Meter', '799 - 803', '2470 - 2790', '783 - 943',
       '4500 - 5540', '10030', '2801.25', '1688.12', '1255 - 1375',
       '1733.5', '10624', '610 - 615', '854 - 960', '2650 - 2990',
       '1.25Acres', '86.72Sq. Meter', '1230 - 1490', '896.9', '660 - 780',
       '1626.6', '1150 - 1194', '684 - 810', '866.28', '1200 - 1800',
       '1510 - 1670', '1248.52', '1370.07', '1550 - 1590', '777.4',
       '1235 - 1410', '3484 - 3550', '1139.7', '38Guntha', '929 - 1078',
       '2150 - 2225', '1520 - 1759', '629 - 1026', '1215 - 1495',
       '6Acres', '1140 - 1250', '2400 - 2600', '1052 - 1322',
       '5666 - 5669', '2162.03', '2999.97', '1950.2', '712 - 938',
       '2171.66', '1783 - 1878', '120Sq. Yards', '24Sq. Meter',
       '2528 - 3188', '1623.29', '650 - 760', '1400 - 1421', '16335',
       '4000 - 4450', '142.84Sq. Meter', '300Sq. Yards', '1932.47',
       '2204 - 2362', '1437 - 1629', '850 - 1060', '30400', '1200 - 1470',
       '5665.84', '1331.95', '1020 - 1130', '1133 - 1384', '1902.55',
       '1691 - 2170', '1200 - 2000', '1594.98', '1892 - 2798',
       '534 - 763', '1498 - 1523', '5Acres', '660 - 690', '1165 - 1225',
       '1975 - 2289', '1522.5', '1094.21', '1740.91', '700 - 800',
       '100Sq. Meter', '530 - 575', '1160 - 1260', '1741 - 2074',
       '1152 - 1197', '600 - 4000', '2563 - 3914', '475 - 1058.27',
       '1149.91'], dtype=object)
```

```
data1 = data.copy()
```

```
index = 0
for feature in data1['total_sqft']:

    if '-' in feature:
       # print(feature.index)
        res = feature.split(' - ')
        first = float(res[0])
        last = float(res[1])
        avg = (first+last)/2.0
        data1.at[index,'total_sqft_new'] = avg
        index+=1
        #print(res)
```

```
        #print(avg)

    elif 'Sq. Meter' in feature:
            sq = feature.split('Sq. Meter')
            sq_area = 10.76391042*float(sq[0])
            data1.at[index,'total_sqft_new'] = sq_area
            index+=1

            #print(sq_area)
            #print(feature)
    elif 'Perch' in feature:
        per = feature.split('Perch')
        per_area = 272.25 * float(per[0])
        data1.at[index,'total_sqft_new'] = per_area
        index+=1
        #print(per_area)
        #print(feature)
    elif 'Acres' in feature:
        acr = feature.split('Acres')
        acr_area = 43560*float(acr[0])
        data1.at[index,'total_sqft_new'] = acr_area
        index+=1
         #print(feature)
    elif 'Sq. Yards' in feature:
        sq_yeard = feature.split('Sq. Yards')

        sq_yeard_area = 9*float(sq_yeard[0])
        data1.at[index,'total_sqft_new'] = sq_yeard_area
        index+=1

        #print(feature)
    elif 'Cents' in feature:
        cent = feature.split('Cents')
        cent_area = 435.56 * float(cent[0])
        data1.at[index,'total_sqft_new'] = cent_area
        index+=1

        #print(feature)
    elif 'Guntha' in feature:
        guntha = feature.split('Guntha')
        guntha_area = 1089 * float(guntha[0])
        data1.at[index,'total_sqft_new'] = guntha_area
        index+=1

    elif 'Grounds' in feature:
        ground = feature.split('Grounds')
        ground_area = 2400 * float(ground[0])
        data1.at[index,'total_sqft_new'] = ground_area
        index+=1

         #print(feature)
```

```
    else:
        data1.at[index,'total_sqft_new'] = float(feature)
        index+=1
```

```
# 34.46Sq. Meter
# 1330.74
# 2100 - 2850
# 4125Perch
# 5.31Acres
# 547.34 - 827.31
# 1574Sq. Yards
# 3Cents
# 2.09Acres
#38Guntha
```

```
data1['total_sqft_new'].value_counts()
```

```
1200.00     940
1100.00     240
1500.00     239
2400.00     225
600.00      200
            ...
462.00        1
1306.68       1
1462.50       1
36000.00      1
6750.00       1
Name: total_sqft_new, Length: 2126, dtype: int64
```

```
data1['total_sqft_new'].head(50)
```

```
0     1056.00
1     2600.00
2     1440.00
3     1521.00
4     1200.00
5     1170.00
6     2732.00
7     3300.00
8     1310.00
9     1020.00
10    1800.00
11    2785.00
12    1000.00
13    1100.00
14    2250.00
15    1175.00
16    1180.00
17    1540.00
18    2770.00
```

```
19    1100.00
20     600.00
21    1755.00
22    2800.00
23    1767.00
24     510.00
25    1250.00
26     660.00
27    1610.00
28    1151.00
29    1025.00
30    2475.00
31    1075.00
32    1760.00
33    1693.00
34    1925.00
35     700.00
36    1070.00
37    1724.00
38    1290.00
39    1143.00
40    1296.00
41    1254.00
42     600.00
43     660.00
44    1330.74
45     600.00
46     970.00
47    1459.00
48     800.00
49     869.00
Name: total_sqft_new, dtype: float64
```

```
data2 = data1.copy()
```

Check Nan Value or Not

```
data2.isnull().sum()
```

```
area_type           0
availability        0
location            1
size               18
society          6128
total_sqft          0
bath               80
balcony           678
total_sqft_new      0
dtype: int64
```

```
data2.dtypes
```

```
area_type          object
availability       object
location           object
size               object
society            object
total_sqft         object
bath              float64
balcony           float64
total_sqft_new    float64
dtype: object
```

```
# There is nan value in

# location
# size
# socity
# bath
# balcony
# Which will be take care in Feature Engineering Part!
```

area_type

```
data2['area_type'].value_counts()
```

```
Super built-up  Area    9736
Built-up  Area          2688
Plot  Area              2279
Carpet  Area              97
Name: area_type, dtype: int64
```

```
data2['area_type'].hist()
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f6a4022b3d0>
```

availability

```
data2['availability'].value_counts()
```

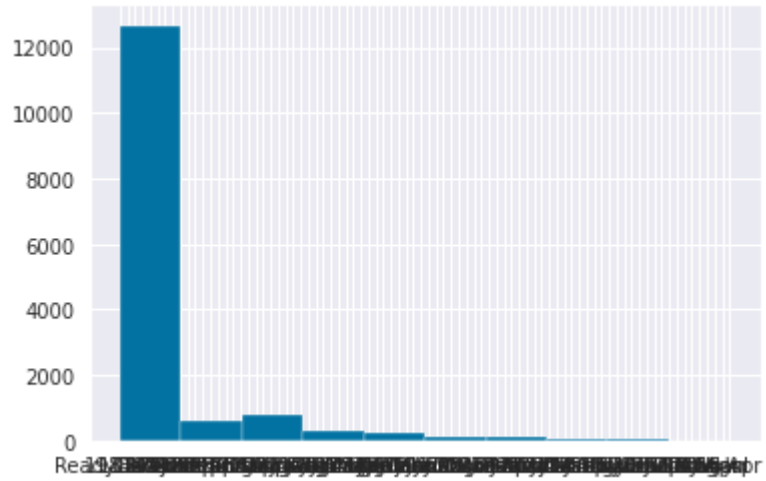
```
Ready To Move    11743
18-Dec             351
18-May             324
18-Apr             296
18-Aug             225
                  ...
15-Aug               1
16-Nov               1
16-Jan               1
14-Jul               1
16-Apr               1
Name: availability, Length: 82, dtype: int64
```

```
# Ready To Move
# Immediate Possession
# Jan,Feb,Mar,Apr,May,Jun,Jul,Aug,Sep,Oct,Nov,Dec
```

```
# One Hot Encoding Use
```

```
data2['availability'].hist( bins = 10)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f6a40175d50>
```



location

```
data2['location'].isnull().sum()
```

```
1
```

```
top_253_values = data2['location'].value_counts().sort_values(ascending=False).head(253).inde
```

```
top_253_values
```

```
Index(['Whitefield', 'Sarjapur  Road', 'Electronic City', 'Kanakpura Road',
       'Thanisandra', 'Yelahanka', 'Uttarahalli', 'Hebbal',
       'Raja Rajeshwari Nagar', 'Marathahalli',
       ...
       'KUDLU MAIN ROAD', '4th Block Koramangala', 'Pattandur Agrahara',
       'Nehru Nagar', 'Kaverappa Layout', 'Ganga Nagar', 'MS Pallya',
       'Chandra Layout', 'Sarjapur Road, ', 'HAL 2nd Stage'],
      dtype='object', length=253)
```

```
data2['location']= data2['location'].fillna(data2['location'].mode()[0])
```

```
data2['location'].isnull().sum()
```

```
0
```

```
data3 = data2.copy()
```

size

```
data3['size'].isnull().sum()
```

```
18
```

```
data3['size'].dtype
```

```
dtype('O')
```

```
total_unique_values = data3['size'].unique()
```

```
total_unique_values
```

```
array(['2 BHK', '4 Bedroom', '3 BHK', '4 BHK', '6 Bedroom', '3 Bedroom',
       '1 BHK', '1 RK', '1 Bedroom', '8 Bedroom', '2 Bedroom',
       '7 Bedroom', '5 BHK', '7 BHK', '6 BHK', '5 Bedroom', '11 BHK',
       '9 BHK', nan, '9 Bedroom', '27 BHK', '10 Bedroom', '11 Bedroom',
```

```
       '10 BHK', '19 BHK', '16 BHK', '43 Bedroom', '14 BHK', '8 BHK',
       '12 Bedroom', '13 BHK', '18 Bedroom', '16 Bedroom'], dtype=object)
```

```
# BHK
# Bedroom
# RK
# nan
```

```
data3['size'].value_counts()
```

```
2 BHK         5739
3 BHK         4788
4 Bedroom      918
4 BHK          673
3 Bedroom      613
1 BHK          592
2 Bedroom      368
5 Bedroom      338
6 Bedroom      208
1 Bedroom      122
7 Bedroom       95
8 Bedroom       94
5 BHK           61
9 Bedroom       54
6 BHK           35
7 BHK           18
1 RK            15
10 Bedroom      15
9 BHK           13
8 BHK            8
10 BHK           2
11 BHK           2
11 Bedroom       2
19 BHK           1
16 BHK           1
43 Bedroom       1
14 BHK           1
27 BHK           1
12 Bedroom       1
13 BHK           1
18 Bedroom       1
16 Bedroom       1
Name: size, dtype: int64
```

```
#!pip install nums_from_string
```

```
Collecting nums_from_string
  Downloading nums_from_string-0.1.2-py3-none-any.whl (5.0 kB)
Installing collected packages: nums-from-string
Successfully installed nums-from-string-0.1.2
```

```
# new feature size_sqft
import nums_from_string
# Bangalaure 1BHK is different from others
# Also RK and Badroom

index = 0

for feature in data3['size']:


    if str('BHK') in str(feature) or str('RK') in str(feature):
        kos = nums_from_string.get_nums(feature)
        #print(kos)
        #print(type(kos))
        ans = 450*int(kos[0])
        data3.at[index,'size_sqft'] = ans
        index += 1
    elif str('Bedroom') in str(feature):
        kos = nums_from_string.get_nums(feature)
        ans = 160*int(kos[0])
        data3.at[index,'size_sqft'] = ans
        index += 1
    else:
        data3.at[index,'size_sqft'] =  data3.at[index,'size']
        index += 1
```

```
data3['size_sqft'].unique()
```

```
array([  900.,   640.,  1350.,  1800.,   960.,   480.,   450.,   160.,
        1280.,   320.,  1120.,  2250.,  3150.,  2700.,   800.,  4950.,
        4050.,    nan,  1440., 12150.,  1600.,  1760.,  4500.,  8550.,
        7200.,  6880.,  6300.,  3600.,  1920.,  5850.,  2880.,  2560.])
```

```
# Nan value replaced
data3['size_sqft'] = data3['size_sqft'].fillna(data3['size_sqft'].mean())
```

```
data3['size_sqft'].isnull().sum()
```

```
0
```

```
data4 = data3.copy()
```

location

```
data4['location'].isnull().sum()
```

```
0
```

```
data4['location'].value_counts()
```

```
Whitefield                          592
Sarjapur  Road                      437
Electronic City                     337
Kanakpura Road                      298
Thanisandra                         277
                                   ...
K R C kothanur                        1
1Channasandra                         1
Vijayabank bank layout                1
Saptagiri Layout                      1
Vishweswaraiah Layout 4th Block       1
Name: location, Length: 1358, dtype: int64
```

```
label_encoder = LabelEncoder()

data4['location_encoded'] = label_encoder.fit_transform(data4['location'])
```

```
data4
```

	area_type	availability	location	size	society	total_sqft	bath	b
0	Super built-up Area	19-Dec	Electronic City Phase II	2 BHK	Coomee	1056	2.0	
1	Plot Area	Ready To Move	Chikka Tirupathi	4 Bedroom	Theanmp	2600	5.0	
2	Built-up Area	Ready To Move	Uttarahalli	3 BHK	NaN	1440	2.0	
3	Super built-up Area	Ready To Move	Lingadheeranahalli	3 BHK	Soiewre	1521	3.0	

```
data5 = data4.copy()
```

Area

Socity

14795 built-up Move Vittasandra 2 BHK Prila C 1246 2.0

```
data5['society'].isnull().sum()
```

```
6128
```

Area

```
data5['society'].value_counts().sort_values(ascending=False).head(50)
```

```
GrrvaGr    92
PrarePa    78
Prtates    64
Sryalan    63
Bhmesy     63
GMown E    60
Prityel    57
Prarkun    57
PrityTr    55
Prtanha    54
Dhalsh     52
IBityin    48
Soitya     45
Soresea    42
SNity S    40
Bhe 2ko    39
Rosha I    39
Adeatlm    39
SunceEs    38
SNnia E    38
Prncyrn    34
Prlla C    34
JRrnauv    33
PhestOn    31
DLhtsnd    31
```

```
Puachal    31
MenueNo    30
PuandHi    29
Raard B    29
GoAirej    29
Dieldli    28
Aklia R    28
ViistLa    28
RothaVa    27
Soazak     26
PualeSk    26
Brnia G    25
Sodgere    25
Shitt S    25
Soiewre    25
AsastDe    24
Itelaa     24
Brway G    24
DLhtsan    24
Prrtht     23
Eladero    23
Jaades     23
SoechHa    22
RInceeg    22
Klark L    21
Name: society, dtype: int64
```

```
data5['society'] = data5['society'].fillna(data5['society'].mode()[0])
```

```
data5['society'].isnull().sum()
```

```
0
```

```
data5['society'].dtype
```

```
dtype('O')
```

```
label_encoder = LabelEncoder()
data5['society_encoded'] = label_encoder.fit_transform(data5['society']).astype('float64')
```

```
data5['society_encoded']
```

```
0         487.0
1        2568.0
2         852.0
3        2302.0
4         852.0
          ...
14795    1675.0
14796     852.0
14797    2626.0
```

```
14798    2071.0
14799    1284.0
Name: society_encoded, Length: 14800, dtype: float64
```

```
data6 = data5.copy()
```

```
data6
```

	area_type	availability	location	size	society	total_sqft	bath	b
0	Super built-up Area	19-Dec	Electronic City Phase II	2 BHK	Coomee	1056	2.0	
1	Plot Area	Ready To Move	Chikka Tirupathi	4 Bedroom	Theanmp	2600	5.0	
2	Built-up Area	Ready To Move	Uttarahalli	3 BHK	GrrvaGr	1440	2.0	
3	Super built-up Area	Ready To Move	Lingadheeranahalli	3 BHK	Soiewre	1521	3.0	
4	Super built-up Area	Ready To Move	Kothanur	2 BHK	GrrvaGr	1200	2.0	
...	...	...	...	...	...	...	...	...
14795	Super built-up Area	Ready To Move	Vittasandra	2 BHK	Prlla C	1246	2.0	
14796	Super built-up Area	Ready To Move	Gottigere	3 BHK	GrrvaGr	1660	3.0	
14797	Super built-up Area	Ready To Move	Channasandra	2 BHK	Unm 2El	1216	2.0	
14798	Built-up Area	18-Feb	Tumkur Road	2 BHK	Sahtsva	996	2.0	
14799	Built-up Area	Ready To Move	7th Phase JP Nagar	2 BHK	MaicaRS	1150	2.0	

14800 rows × 12 columns

```
data6['area_type'].value_counts()
```

```
Super built-up  Area    9736
Built-up  Area          2688
Plot  Area              2279
Carpet  Area              97
Name: area_type, dtype: int64
```

```
# One Hot Encoding in area_type

data6 = pd.get_dummies(data6,columns = ['area_type'])

#data6['area_type_encoded'] = label_encoder.fit_transform(data6['area_type']).astype('float64
```

```
data6
```

	availability	location	size	society	total_sqft	bath	balcony	tot

```
data7 = data6.copy()
```

```
# extract the values

data7.drop(columns=data7.columns[:5],axis=1, inplace=True)
```

```
data7
```

	bath	balcony	total_sqft_new	size_sqft	location_encoded	society_encoded	are
0	2.0	1.0	1056.0	900.0	430	487.0	
1	5.0	3.0	2600.0	640.0	325	2568.0	
2	2.0	3.0	1440.0	1350.0	1219	852.0	
3	3.0	1.0	1521.0	1350.0	778	2302.0	
4	2.0	1.0	1200.0	900.0	736	852.0	
...	...	...	...	...	...	...	
14795	2.0	1.0	1246.0	900.0	1284	1675.0	
14796	3.0	2.0	1660.0	1350.0	477	852.0	
14797	2.0	2.0	1216.0	900.0	314	2626.0	
14798	2.0	1.0	996.0	900.0	1208	2071.0	
14799	2.0	2.0	1150.0	900.0	76	1284.0	

14800 rows × 10 columns

```
data7.isnull().sum()
```

```
bath                                80
balcony                            678
total_sqft_new                       0
size_sqft                            0
location_encoded                     0
society_encoded                      0
area_type_Built-up  Area             0
area_type_Carpet  Area               0
area_type_Plot  Area                 0
area_type_Super built-up  Area       0
dtype: int64
```

```
# Bath

data7['bath'] = data7['bath'].fillna(data7['bath'].mean()+1)
```

```
data7['bath'].isnull().sum()
```

```
0
```

```
# Balcony

data7['balcony'].unique()
```

```
array([ 1.,  3., nan,  2.,  0.])
```

```
data7['balcony'].value_counts()
```

```
2.0    5658
1.0    5444
3.0    1861
0.0    1159
Name: balcony, dtype: int64
```

```
data7['balcony'].mean()
```

```
1.582141339753576
```

```
data7['balcony'] = data7['balcony'].fillna(data7['balcony'].mean()+1)
```

```
data7['balcony'].isnull().sum()
```

```
0
```

```
data8 = data7.copy()
```

```
data8
```

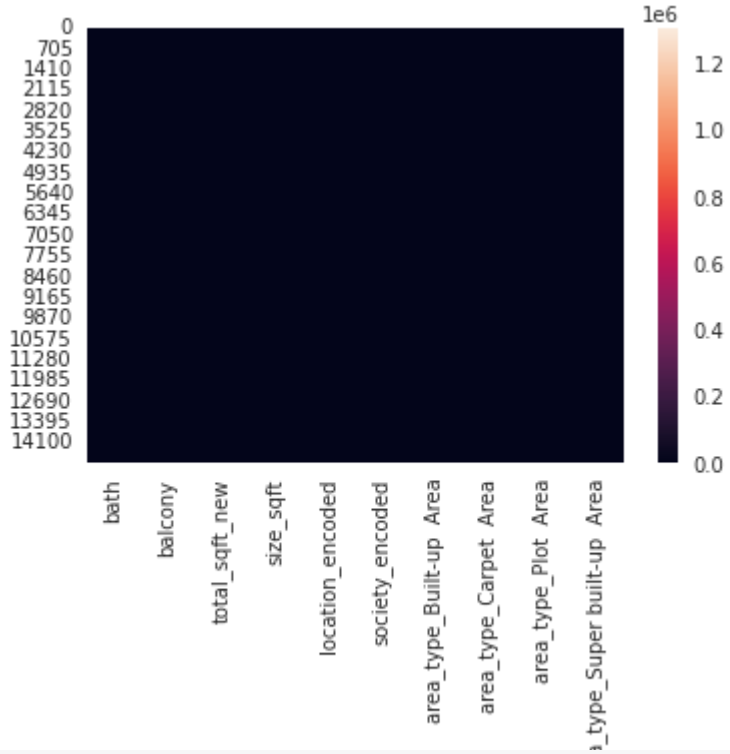
	bath	balcony	total_sqft_new	size_sqft	location_encoded	society_encoded	are
0	2.0	1.0	1056.0	900.0	430	487.0	
1	5.0	3.0	2600.0	640.0	325	2568.0	
2	2.0	3.0	1440.0	1350.0	1219	852.0	
3	3.0	1.0	1521.0	1350.0	778	2302.0	
4	2.0	1.0	1200.0	900.0	736	852.0	
...	...	...	...	...	...	...	
14795	2.0	1.0	1246.0	900.0	1284	1675.0	
14796	3.0	2.0	1660.0	1350.0	477	852.0	
14797	2.0	2.0	1216.0	900.0	314	2626.0	

```
data8.corr()
```

	bath	balcony	total_sqft_new	size_sqft	location_encoded	soci
bath	1.000000	0.247079	0.051391	0.445434	0.003084	
balcony	0.247079	1.000000	0.011836	0.315329	0.000016	
total_sqft_new	0.051391	0.011836	1.000000	0.005273	-0.000109	
size_sqft	0.445434	0.315329	0.005273	1.000000	-0.005195	
location_encoded	0.003084	0.000016	-0.000109	-0.005195	1.000000	
society_encoded	-0.075444	-0.006290	-0.007538	0.070231	-0.003154	
area_type_Built-up Area	-0.017806	-0.062356	0.007257	-0.026570	-0.028987	
area_type_Carpet Area	-0.008297	-0.015189	-0.002724	0.019955	-0.022839	
area_type_Plot Area	0.377525	-0.064027	0.037627	-0.390004	0.001181	
area_type_Super built-up Area	-0.271330	0.101965	-0.034059	0.314901	0.026541	

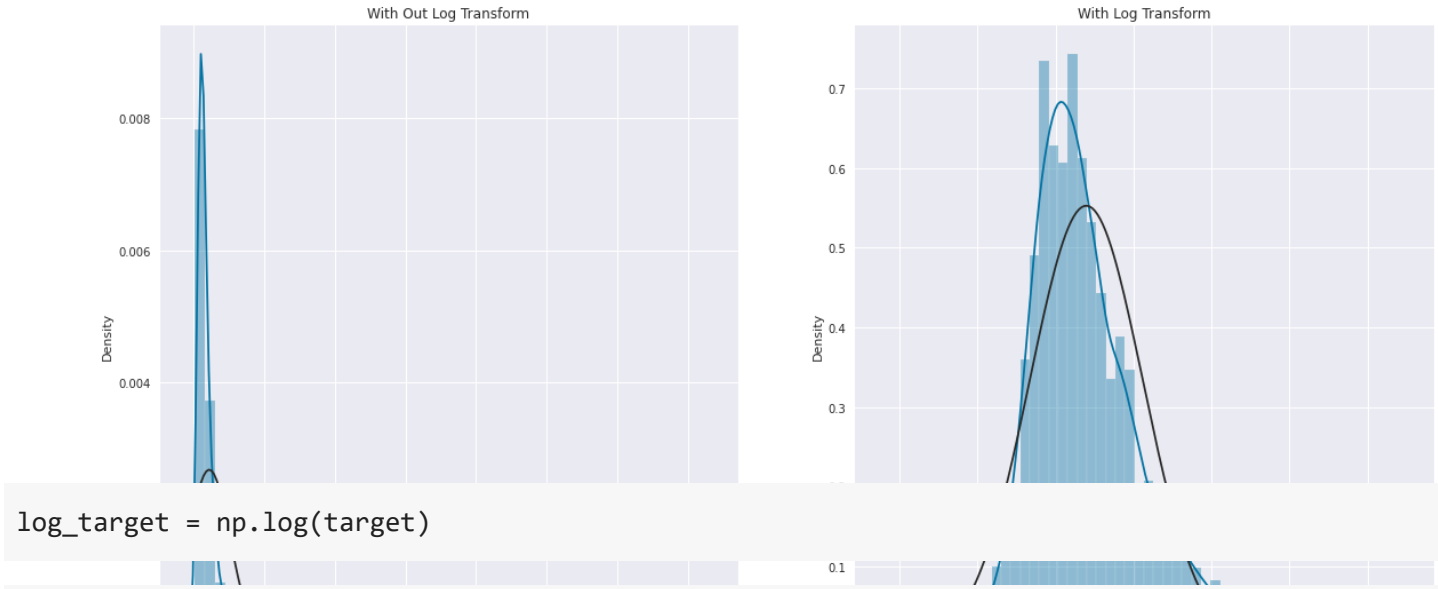
```
sns.heatmap(data8)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f6a3df1aa10>
```




```
import scipy.stats
plt.figure(figsize=(20,10))
plt.subplot(1,2,1)
sns.distplot(target,kde=True,fit = scipy.stats.norm)
plt.title("With Out Log Transform")
plt.subplot(1,2,2)
sns.distplot(np.log(target),kde=True,fit = scipy.stats.norm)
plt.xlabel('Log Sale Price')
plt.title("With Log Transform")

plt.show()
```




```
log_target = np.log(target)
```

```
log_target
```

```
0        3.665355
1        4.787492
2        4.127134
3        4.553877
4        3.931826
           ...
13315    5.442418
13316    5.991465
13317    4.094345
13318    6.190315
13319    2.833213
Name: price, Length: 13320, dtype: float64
```

```
train_final = data8.loc[:df.index.max(), :].copy()
test_final = data8.loc[df.index.max() + 1:, :].reset_index(drop=True).copy()
```

```
train_final.shape
```

```
(13320, 10)
```

```
train_final
```

	bath	balcony	total_sqft_new	size_sqft	location_encoded	society_encoded	are
0	2.0	1.000000	1056.0	900.0	430	487.0	
1	5.0	3.000000	2600.0	640.0	325	2568.0	
2	2.0	3.000000	1440.0	1350.0	1219	852.0	
3	3.0	1.000000	1521.0	1350.0	778	2302.0	
4	2.0	1.000000	1200.0	900.0	736	852.0	
...	...	...	...	...	...	...	
13315	4.0	0.000000	3453.0	800.0	1296	220.0	
13316	5.0	2.582141	3600.0	1800.0	1039	852.0	

```
test_final.shape
```

```
(1480, 10)
```

```
print(train_final.isnull().sum())
print(test_final.isnull().sum())
```

```
bath                                0
balcony                             0
total_sqft_new                      0
size_sqft                           0
location_encoded                    0
society_encoded                     0
area_type_Built-up  Area            0
area_type_Carpet  Area              0
area_type_Plot  Area                0
area_type_Super built-up  Area      0
dtype: int64
bath                                0
balcony                             0
total_sqft_new                      0
size_sqft                           0
location_encoded                    0
society_encoded                     0
area_type_Built-up  Area            0
area_type_Carpet  Area              0
area_type_Plot  Area                0
area_type_Super built-up  Area      0
dtype: int64
```

```
log_target.dtype
```

```
dtype('float64')
```

```
train_final.dtypes
```

```
bath                                float64
balcony                             float64
total_sqft_new                      float64
size_sqft                           float64
location_encoded                      int64
society_encoded                     float64
area_type_Built-up  Area              uint8
area_type_Carpet  Area                uint8
area_type_Plot  Area                  uint8
area_type_Super built-up  Area        uint8
dtype: object
```

```
test_final.dtypes
```

```
bath                                float64
balcony                             float64
total_sqft_new                      float64
size_sqft                           float64
location_encoded                      int64
society_encoded                     float64
area_type_Built-up  Area              uint8
area_type_Carpet  Area                uint8
area_type_Plot  Area                  uint8
area_type_Super built-up  Area        uint8
dtype: object
```

Model Selection

```
data10 = pd.concat([train_final , log_target],axis = 1)
```

```
data10
```

	bath	balcony	total_sqft_new	size_sqft	location_encoded	society_encoded	are
0	2.0	1.000000	1056.0	900.0	430	487.0	
1	5.0	3.000000	2600.0	640.0	325	2568.0	
2	2.0	3.000000	1440.0	1350.0	1219	852.0	
3	3.0	1.000000	1521.0	1350.0	778	2302.0	
4	2.0	1.000000	1200.0	900.0	736	852.0	
...	...	...	...	...	...	...	
13315	4.0	0.000000	3453.0	800.0	1296	220.0	

```
data10.columns
```

```
Index(['bath', 'balcony', 'total_sqft_new', 'size_sqft', 'location_encoded',
       'society_encoded', 'area_type_Built-up  Area', 'area_type_Carpet  Area',
       'area_type_Plot  Area', 'area_type_Super built-up  Area', 'price'],
      dtype='object')
```

13320 rows x 11 columns

```
data10.dtypes
```

```
bath                              float64
balcony                           float64
total_sqft_new                    float64
size_sqft                         float64
location_encoded                    int64
society_encoded                   float64
area_type_Built-up  Area            uint8
area_type_Carpet  Area              uint8
area_type_Plot  Area                uint8
area_type_Super built-up  Area      uint8
price                             float64
dtype: object
```

```
data10
```

	bath	balcony	total_sqft_new	size_sqft	location_encoded	society_encoded	are
0	2.0	1.000000	1056.0	900.0	430	487.0	
1	5.0	3.000000	2600.0	640.0	325	2568.0	
2	2.0	3.000000	1440.0	1350.0	1219	852.0	
3	3.0	1.000000	1521.0	1350.0	778	2302.0	
4	2.0	1.000000	1200.0	900.0	736	852.0	
...	...	...	...	...	...	...	
13315	4.0	0.000000	3453.0	800.0	1296	220.0	

```
data10.dtypes
```

```
bath                                 float64
balcony                              float64
total_sqft_new                       float64
size_sqft                            float64
location_encoded                       int64
society_encoded                      float64
area_type_Built-up  Area               uint8
area_type_Carpet  Area                 uint8
area_type_Plot  Area                   uint8
area_type_Super built-up  Area         uint8
price                                float64
dtype: object
```

```
#data10['location_encoded'] = data10['location_encoded'].astype(int).astype(float)
```

```
data10.dtypes
```

```
bath                                 float64
balcony                              float64
total_sqft_new                       float64
size_sqft                            float64
location_encoded                       int64
society_encoded                      float64
area_type_Built-up  Area               uint8
area_type_Carpet  Area                 uint8
area_type_Plot  Area                   uint8
area_type_Super built-up  Area         uint8
price                                float64
dtype: object
```

```
from pycaret.regression import *
```

```
model_select = setup(data10 , target ='price')
```

	Description	Value
0	session_id	8582
1	Target	price
2	Original Data	(13320, 11)
3	Missing Values	False
4	Numeric Features	10
5	Categorical Features	0
6	Ordinal Features	False
7	High Cardinality Features	False
8	High Cardinality Method	None
9	Transformed Train Set	(9323, 10)
10	Transformed Test Set	(3997, 10)
11	Shuffle Train-Test	True
12	Stratify Train-Test	False
13	Fold Generator	KFold
14	Fold Number	10
15	CPU Jobs	-1
16	Use GPU	False
17	Log Experiment	False
18	Experiment Name	reg-default-name
19	USI	927e
20	Imputation Type	simple
21	Iterative Imputation Iteration	None
22	Numeric Imputer	mean
23	Iterative Imputation Numeric Model	None
24	Categorical Imputer	constant
25	Iterative Imputation Categorical Model	None
26	Unknown Categoricals Handling	least_frequent
27	Normalize	False
28	Normalize Method	None

```
compare_models()
```

1 to 18 of 18 entries Filter

index	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
lightgbm	Light Gradient Boosting Machine	0.226	0.1025	0.3198	0.8013	0.0576	0.0513	0.147
rf	Random Forest Regressor	0.2263	0.1137	0.3368	0.7796	0.0603	0.0511	2.479
gbr	Gradient Boosting Regressor	0.2565	0.1224	0.3496	0.7626	0.0633	0.0585	0.626
et	Extra Trees Regressor	0.2383	0.1269	0.3559	0.7539	0.0638	0.0539	1.705
knn	K Neighbors Regressor	0.2664	0.1547	0.393	0.6998	0.0707	0.0602	0.077
ada	AdaBoost Regressor	0.3342	0.1907	0.4364	0.6296	0.0812	0.0788	0.262
dt	Decision Tree Regressor	0.2891	0.1992	0.4458	0.6122	0.0793	0.065	0.051
br	Bayesian Ridge	0.4042	0.3044	0.5512	0.409	0.0968	0.0915	0.018
ridge	Ridge Regression	0.4042	0.3044	0.5512	0.4089	0.0968	0.0915	0.014
lar	Least Angle Regression	0.4042	0.3044	0.5512	0.4089	0.0968	0.0915	0.016
lr	Linear Regression	0.4042	0.3044	0.5512	0.4089	0.0968	0.0915	0.317
omp	Orthogonal Matching Pursuit	0.4126	0.3204	0.5656	0.3781	0.0995	0.0935	0.016
lasso	Lasso Regression	0.5293	0.485	0.6959	0.0613	0.1248	0.1205	0.016
en	Elastic Net	0.5293	0.485	0.6959	0.0613	0.1247	0.1204	0.016
llar	Lasso Least Angle Regression	0.5533	0.5171	0.7187	-0.0009	0.1296	0.1267	0.015
dummy	Dummy Regressor	0.5533	0.5171	0.7187	-0.0009	0.1296	0.1267	0.015
huber	Huber Regressor	0.5632	10.4483	1.9529	-18.6531	0.1384	0.1272	0.163
par	Passive Aggressive Regressor	1.9384	942.8275	15.8377	-1745.7476	0.2961	0.4333	0.021

Show 100 per page

Like what you see? Visit the data table notebook to learn more about interactive tables.

```
LGBMRegressor(boosting_type='gbdt', class_weight=None, colsample_bytree=1.0,
```

```
lightgbm_model = create_model('lightgbm')
```

Fold	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	0.2257	0.1162	0.3409	0.7849	0.0619	0.0522
1	0.2329	0.1068	0.3268	0.8128	0.0576	0.0520
2	0.2193	0.0931	0.3051	0.7895	0.0559	0.0506

```
lightgbm_model.get_params()
```

```
{'boosting_type': 'gbdt',
 'class_weight': None,
 'colsample_bytree': 1.0,
 'importance_type': 'split',
 'learning_rate': 0.1,
 'max_depth': -1,
 'min_child_samples': 20,
 'min_child_weight': 0.001,
 'min_split_gain': 0.0,
 'n_estimators': 100,
 'n_jobs': -1,
 'num_leaves': 31,
 'objective': None,
 'random_state': 8582,
 'reg_alpha': 0.0,
 'reg_lambda': 0.0,
 'silent': 'warn',
 'subsample': 1.0,
 'subsample_for_bin': 200000,
 'subsample_freq': 0}
```

```
rf_model = create_model('rf')
```

Fold	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	0.2322	0.1388	0.3725	0.7433	0.0664	0.0532
1	0.2276	0.1125	0.3354	0.8027	0.0594	0.0507

```
rf_model.get_params()
```

```
{'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': -1,
 'oob_score': False,
 'random_state': 8582,
 'verbose': 0,
 'warm_start': False}
```

```
gbr_model = create_model('gbr')
```

	MAE	MSE	RMSE	R2	RMSLE	MAPE

```
gbr_model.get_params()
```

```
{'alpha': 0.9,
 'ccp_alpha': 0.0,
 'criterion': 'friedman_mse',
 'init': None,
 'learning_rate': 0.1,
 'loss': 'ls',
 'max_depth': 3,
 'max_features': None,
 'max_leaf_nodes': 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_iter_no_change': None,
 'presort': 'deprecated',
 'random_state': 8582,
 'subsample': 1.0,
 'tol': 0.0001,
 'validation_fraction': 0.1,
 'verbose': 0,
 'warm_start': False}
```

```
et_model = create_model('et')
```

	MAE	MSE	RMSE	R2	RMSLE	MAPE
Mean	0.2383	0.1269	0.3559	0.7539	0.0638	0.0539

```
et_model.get_params()
```

```
{'bootstrap': False,
 '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': -1,
 'oob_score': False,
 'random_state': 8582,
 'verbose': 0,
 'warm_start': False}
```

```
knn_model = create_model('knn')
```

Fold	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	0.2704	0.1754	0.4188	0.6755	0.0760	0.0621
1	0.2684	0.1538	0.3922	0.7303	0.0695	0.0594
2	0.2591	0.1396	0.3737	0.6843	0.0683	0.0595
3	0.2716	0.1530	0.3912	0.7165	0.0695	0.0606
4	0.2639	0.1573	0.3966	0.7181	0.0706	0.0590
5	0.2592	0.1544	0.3930	0.7067	0.0716	0.0595
6	0.2667	0.1485	0.3854	0.6922	0.0689	0.0596
7	0.2837	0.1766	0.4202	0.6525	0.0751	0.0635
8	0.2599	0.1466	0.3828	0.7021	0.0691	0.0585
9	0.2611	0.1417	0.3764	0.7203	0.0685	0.0599
Mean	0.2664	0.1547	0.3930	0.6998	0.0707	0.0602
Std	0.0073	0.0119	0.0149	0.0227	0.0026	0.0015

```
knn_model.get_params()
```

```
{'algorithm': 'auto',
```

```
 'leaf_size': 30,
 'metric': 'minkowski',
 'metric_params': None,
 'n_jobs': -1,
 'n_neighbors': 5,
 'p': 2,
 'weights': 'uniform'}
```

```
ada_model = create_model('ada')
```

Fold	MAE	MSE	RMSE	R2	RMSLE	MAPE
0	0.3346	0.1964	0.4432	0.6366	0.0841	0.0809
1	0.3332	0.1846	0.4297	0.6763	0.0780	0.0766
2	0.3129	0.1663	0.4079	0.6238	0.0771	0.0748
3	0.3295	0.1877	0.4332	0.6523	0.0804	0.0774
4	0.3354	0.1899	0.4358	0.6598	0.0811	0.0792
5	0.3412	0.1968	0.4436	0.6263	0.0831	0.0814
6	0.3252	0.1810	0.4255	0.6248	0.0788	0.0759
7	0.3493	0.2144	0.4631	0.5781	0.0841	0.0804
8	0.3570	0.2105	0.4588	0.5721	0.0850	0.0838
9	0.3241	0.1793	0.4234	0.6461	0.0804	0.0776
Mean	0.3342	0.1907	0.4364	0.6296	0.0812	0.0788
Std	0.0121	0.0137	0.0157	0.0316	0.0026	0.0027

```
ada_model.get_params()
```

```
{'base_estimator': None,
 'learning_rate': 1.0,
 'loss': 'linear',
 'n_estimators': 50,
 'random_state': 8582}
```

```
# catboost_params = {
#     'iterations': 6000,
#     'learning_rate': 0.005,
#     'depth': 4,
#     'l2_leaf_reg': 1,
#     'eval_metric':'RMSE',
#     'early_stopping_rounds': 200,
#     'random_seed': 42
```

```
# }

# br_params = {
#     'n_iter': 304,
#     'tol': 0.16864712769300896,
#     'alpha_1': 5.589616542154059e-07,
#     'alpha_2': 9.799343618469923,
#     'lambda_1': 1.7735725582463822,
#     'lambda_2': 3.616928181181732e-06
# }

lightgbm_params = {
        'boosting_type': 'gbdt',
        'class_weight': None,
        'colsample_bytree': 1.0,
        'importance_type': 'split',
        'learning_rate': 0.1,
        'max_depth': -1,
        'min_child_samples': 20,
        'min_child_weight': 0.001,
        'min_split_gain': 0.0,
        'n_estimators': 100,
        'n_jobs': -1,
        'num_leaves': 31,
        'objective': None,
        'random_state': 8582,
        'reg_alpha': 0.0,
        'reg_lambda': 0.0,
        'silent': 'warn',
        'subsample': 1.0,
        'subsample_for_bin': 200000,
        'subsample_freq': 0

}

rf_params = {
      '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': -1,
      'oob_score': False,
```

```
      'random_state': 8582,
      'verbose': 0,
      'warm_start': False
  }

gbr_params = {
      'alpha': 0.9,
      'ccp_alpha': 0.0,
      'criterion': 'friedman_mse',
      'init': None,
      'learning_rate': 0.1,
      'loss': 'ls',
      'max_depth': 3,
      'max_features': None,
      'max_leaf_nodes': 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_iter_no_change': None,
      'presort': 'deprecated',
      'random_state': 8582,
      'subsample': 1.0,
      'tol': 0.0001,
      'validation_fraction': 0.1,
      'verbose': 0,
      'warm_start': False
  }


et_params = {
      'bootstrap': False,
      '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': -1,
      'oob_score': False,
      'random_state': 8582,
      'verbose': 0,
      'warm_start': False
```

```
 }

knn_params = {
          'algorithm': 'auto',
          'leaf_size': 30,
          'metric': 'minkowski',
          'metric_params': None,
          'n_jobs': -1,
          'n_neighbors': 5,
          'p': 2,
          'weights': 'uniform'
  }

ada_params = {
    'base_estimator': None,
    'learning_rate': 1.0,
    'loss': 'linear',
    'n_estimators': 50,
    'random_state': 8582
 }

# ridge_params = {
#     'alpha': 631.1412445239156
# }
```

```
models = {

    "lightgbm": LGBMRegressor(**lightgbm_params),
    "rf": RandomForestRegressor(**rf_params),
    "gbr":  GradientBoostingRegressor(**gbr_params),
    "et": ExtraTreesRegressor(**et_params),
    "knn": KNeighborsRegressor(**knn_params),
    "ada": AdaBoostRegressor(**ada_params)

}
```

```
for name, model in models.items():
  model.fit(train_final, log_target)
  print(name + " trained.")
```

```
lightgbm trained.
rf trained.
gbr trained.
et trained.
knn trained.
ada trained.
```

```
results = {}
kf = KFold(n_splits=10)
```

```
for name, model in models.items():
  result = np.exp(np.sqrt(-cross_val_score(model, train_final, log_target, scoring='neg_mean_
  results[name] = result
```

```
results
```

```
{'ada': array([1.56412888, 1.57315608, 1.5268779 , 1.54416327, 1.59088728,
       1.57466331, 1.56970221, 1.55004307, 1.59832445, 1.56416976]),
 'et': array([1.4069218 , 1.40578309, 1.40920668, 1.40389325, 1.41057606,
       1.4545936 , 1.46559079, 1.40487521, 1.42670761, 1.39519394]),
 'gbr': array([1.40678654, 1.41995737, 1.39928798, 1.40224187, 1.41530953,
       1.44684974, 1.44166933, 1.40166465, 1.4299494 , 1.40781107]),
 'knn': array([1.46106291, 1.45982877, 1.4495952 , 1.45669869, 1.45808888,
       1.50556293, 1.49132939, 1.45180344, 1.48019883, 1.4524395 ]),
 'lightgbm': array([1.36633279, 1.37604611, 1.3546289 , 1.35680099, 1.37330857,
       1.39947899, 1.38582641, 1.34864126, 1.37882352, 1.35685877]),
 'rf': array([1.37445297, 1.38525645, 1.37848126, 1.3766658 , 1.38288635,
       1.4219893 , 1.42239227, 1.36728841, 1.39843947, 1.36984206])}
```

```
for name, result in results.items():
    print("----------\n" + name)
    print(np.mean(result))
    print(np.std(result))
```

```
----------
lightgbm
1.3696746307827268
0.015154551842660482
----------
rf
1.387769432549703
0.019082718617765858
----------
gbr
1.4171527474529755
0.016229856804389296
----------
et
1.4183342037012485
0.02230352834573544
----------
knn
1.4666608532793897
0.018083552429733293
----------
ada
1.5656116211718458
0.020178114252141378
```

```
final_predictions = (
    0.4 * np.exp(models['lightgbm'].predict(test_final)) +
    0.2 * np.exp(models['rf'].predict(test_final)) +
```

```
    0.2 * np.exp(models['gbr'].predict(test_final)) +
    0.1 * np.exp(models['et'].predict(test_final)) +
    0.1 * np.exp(models['knn'].predict(test_final))+
    0.5 * np.exp(models['ada'].predict(test_final))
 )
```

```
final_predictions
```

```
    array([ 94.39593618, 376.65888227, 282.319958  , ...,  83.41278399,
            90.49628851,  94.84847391])
```

```
#train_final.dtypes
```

```
#baseline_model =CatBoostRegressor(verbose=0)
```

```
#baseline_model.fit(train_final,log_target)
```

```
# kf = KFold(n_splits=10)
# results = cross_val_score(baseline_model,train_final,log_target,scoring='neg_mean_squared_e
```

```
# -results
```

```
# -results.mean()
```

```
# plt.figure(figsize=(16,10))
# sns.displot(-results)
```

```
# np.exp(np.sqrt(np.mean(-results)))
```

```
# target.describe()
```

```
log_target
```

```
    0        3.665355
    1        4.787492
    2        4.127134
    3        4.553877
    4        3.931826
               ...
    13315    5.442418
    13316    5.991465
    13317    4.094345
    13318    6.190315
```

```
13319    2.833213
```

```
#predictions = np.exp(baseline_model.predict(test_final))
```

```
submission = pd.concat([test_columns,pd.Series(final_predictions , name = 'price')],axis = 1)
```

```
submission
```

	area_type	availability	location	size	society	total_sqft	bath	bal
0	Super built-up Area	Ready To Move	Brookefield	2 BHK	Roeekbl	1225	2.0	
1	Plot Area	Ready To Move	Akshaya Nagar	9 Bedroom	NaN	2400	9.0	
2	Plot Area	18-Apr	Hennur Road	4 Bedroom	Saandtt	1650	5.0	
3	Super built-up Area	Ready To Move	Kodichikkanahalli	3 BHK	Winerri	1322	3.0	
4	Super built-up Area	Ready To Move	Konanakunte	2 BHK	AmageSa	1161	2.0	
...	...	...	...	...	...	...	...	
1475	Super built-up	Ready To	Vittasandra	2 BHK	Prlla C	1246	2.0	

```
submission.to_csv('/content/House Price Prediction in Bangaluru/Submission_conbined.csv')
```

```
#print(np.__version__)
```

```
# CatboostRegressor-0.87395
```

✓ 0s completed at 5:45 AM