# ММО РК1

Выполнил Байбарин Р.Г. ИУ5-21М

Ввод [1]:
```
import pandas as pd
import numpy as np
```

## Метрики

Ввод [2]:
```
train = pd.read_csv('./train.csv')
train.head()
```

Out[2]:

| | POSTED_BY | UNDER_CONSTRUCTION | RERA | BHK_NO. | BHK_OR_RK | SQUARE_FT | READY_TO_I |
|---|---|---|---|---|---|---|---|
| **0** | Owner | 0 | 0 | 2 | BHK | 1300.236407 | |
| **1** | Dealer | 0 | 0 | 2 | BHK | 1275.000000 | |
| **2** | Owner | 0 | 0 | 2 | BHK | 933.159722 | |
| **3** | Owner | 0 | 1 | 2 | BHK | 929.921143 | |
| **4** | Dealer | 1 | 0 | 2 | BHK | 999.009247 | |

Ввод [3]:
```
train.describe()
```

Out[3]:

| | UNDER_CONSTRUCTION | RERA | BHK_NO. | SQUARE_FT | READY_TO_MOVE | |
|---|---|---|---|---|---|---|
| **count** | 29451.000000 | 29451.000000 | 29451.000000 | 2.945100e+04 | 29451.000000 | 2945 |
| **mean** | 0.179756 | 0.317918 | 2.392279 | 1.980217e+04 | 0.820244 | |
| **std** | 0.383991 | 0.465675 | 0.879091 | 1.901335e+06 | 0.383991 | |
| **min** | 0.000000 | 0.000000 | 1.000000 | 3.000000e+00 | 0.000000 | |
| **25%** | 0.000000 | 0.000000 | 2.000000 | 9.000211e+02 | 1.000000 | |
| **50%** | 0.000000 | 0.000000 | 2.000000 | 1.175057e+03 | 1.000000 | |
| **75%** | 0.000000 | 1.000000 | 3.000000 | 1.550688e+03 | 1.000000 | |
| **max** | 1.000000 | 1.000000 | 20.000000 | 2.545455e+08 | 1.000000 | |

Ввод [4]:
```
train.isna().sum()
```

Out[4]:
```
POSTED_BY                0
UNDER_CONSTRUCTION       0
RERA                     0
BHK_NO.                  0
BHK_OR_RK                0
SQUARE_FT                0
READY_TO_MOVE            0
RESALE                   0
ADDRESS                  0
LONGITUDE                0
LATITUDE                 0
TARGET(PRICE_IN_LACS)    0
dtype: int64
```

## Задача №1.

Для набора данных проведите кодирование одного (произвольного) категориального признака с использованием метода "count (frequency) encoding".

Ввод [5]:
```
enc_nom_1 = (train.groupby('POSTED_BY').size()) / len(train)
enc_nom_1
```

Out[5]:
```
POSTED_BY
Builder    0.021120
Dealer     0.621065
Owner      0.357815
dtype: float64
```

Ввод [6]:
```
train['POSTED_BY_ENCODE'] = train['POSTED_BY'].apply(lambda x : enc_nom_1[x
train.head()
```

Out[6]:

| | POSTED_BY | UNDER_CONSTRUCTION | RERA | BHK_NO. | BHK_OR_RK | SQUARE_FT | READY_TO_I |
|---|---|---|---|---|---|---|---|
| **0** | Owner | 0 | 0 | 2 | BHK | 1300.236407 | |
| **1** | Dealer | 0 | 0 | 2 | BHK | 1275.000000 | |
| **2** | Owner | 0 | 0 | 2 | BHK | 933.159722 | |
| **3** | Owner | 0 | 1 | 2 | BHK | 929.921143 | |
| **4** | Dealer | 1 | 0 | 2 | BHK | 999.009247 | |

## Задача №21.

Для набора данных проведите масштабирование данных для одного (произвольного) числового признака с использованием масштабирования по медиане.

Ввод [7]:
```
from sklearn.preprocessing import RobustScaler

data_scaled_temp = RobustScaler().fit_transform(train[["TARGET(PRICE_IN_LAC

new_df = pd.DataFrame(data_scaled_temp)
new_df.columns=["TARGET_SCALED", "SQUARE_FT_SCALED"]

print(new_df)
# data_cs41_scaled = arr_to_df(data_cs41_scaled_temp)
# data_cs41_scaled.describe()
```

```
       TARGET_SCALED  SQUARE_FT_SCALED
0          -0.112903          0.192387
1          -0.177419          0.153601
2          -0.306452         -0.371768
3           0.008065         -0.376745
4          -0.024194         -0.270565
...              ...               ...
29446      -0.274194          2.036285
29447      -0.741935         -0.623708
29448      -0.562903         -0.234245
29449       0.080645         -0.381113
29450      -0.551613         -0.427688

[29451 rows x 2 columns]
```

Ввод [8]:
```
new_train = train.join(new_df)
new_train = new_train[new_train["TARGET_SCALED"]<200]
```

Ввод [9]:
```
new_train
```

Out[9]:

| | POSTED_BY | UNDER_CONSTRUCTION | RERA | BHK_NO. | BHK_OR_RK | SQUARE_FT | READY_ |
|---|---|---|---|---|---|---|---|
| **0** | Owner | 0 | 0 | 2 | BHK | 1300.236407 | |
| **1** | Dealer | 0 | 0 | 2 | BHK | 1275.000000 | |
| **2** | Owner | 0 | 0 | 2 | BHK | 933.159722 | |
| **3** | Owner | 0 | 1 | 2 | BHK | 929.921143 | |
| **4** | Dealer | 1 | 0 | 2 | BHK | 999.009247 | |
| **...** | ... | ... | ... | ... | ... | ... | |
| **29446** | Owner | 0 | 0 | 3 | BHK | 2500.000000 | |
| **29447** | Owner | 0 | 0 | 2 | BHK | 769.230769 | |
| **29448** | Dealer | 0 | 0 | 2 | BHK | 1022.641509 | |
| **29449** | Owner | 0 | 0 | 2 | BHK | 927.079009 | |
| **29450** | Dealer | 0 | 1 | 2 | BHK | 896.774194 | |

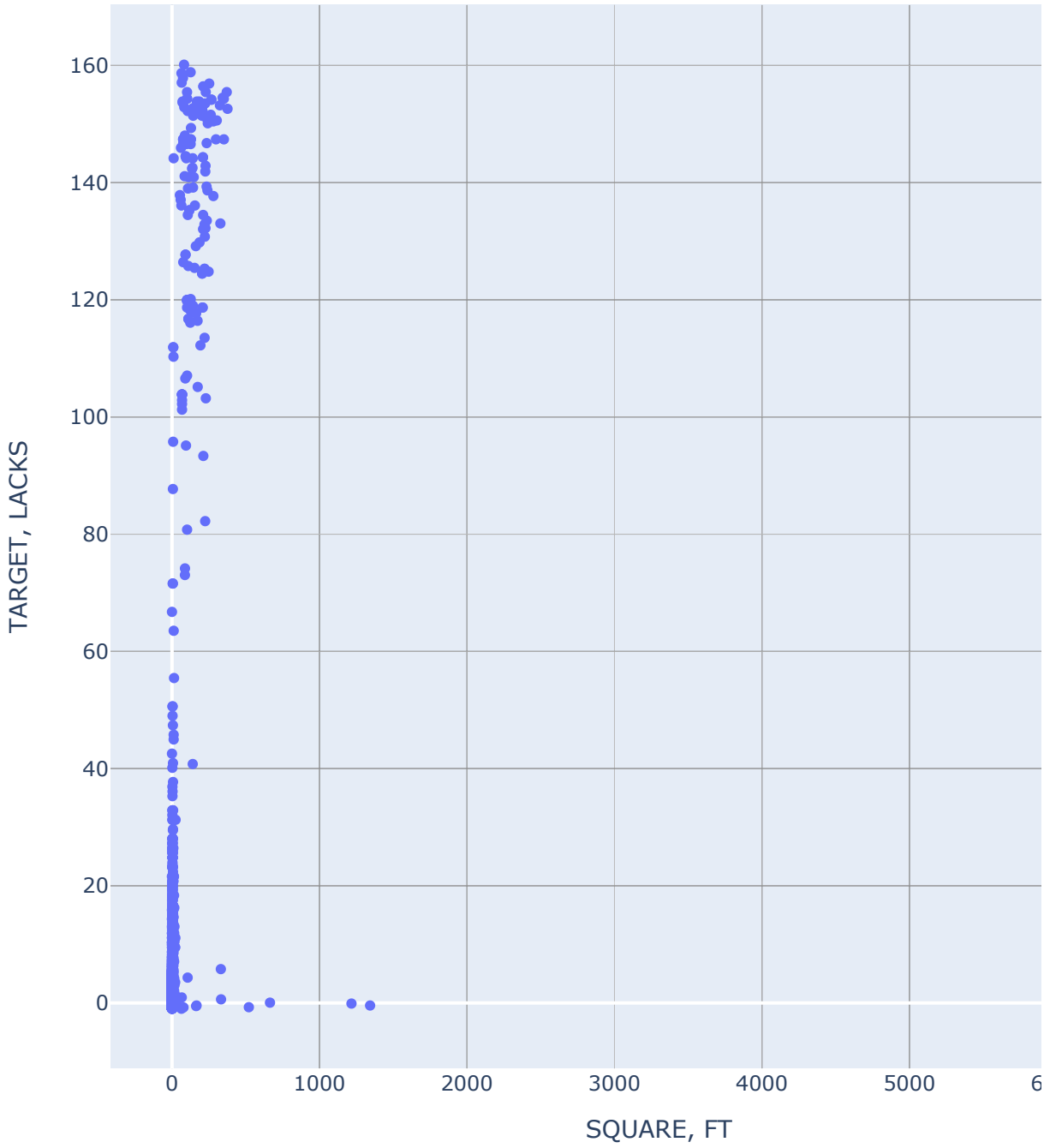
29448 rows × 15 columns

## Диаграмма рассеивания

Ввод [10]:
```
import plotly.graph_objects as go

fig = go.Figure(
    data=[go.Scatter(x=new_train["SQUARE_FT_SCALED"], y=new_train["TARGET_S
    layout=go.Layout(height=800, width=800, xaxis={"title": "SQUARE, FT"},
)
```

Ввод [11]:
```
fig.show()
```




Ввод [ ]:

Ввод [ ]:

Ввод [ ]: