1. Preprocessing:

   I. Check the nulls then remove nulls in ‘bathroom’ and ‘bedroom’ and ‘latitude’ and ‘longitude‘ columns , because number of rows which are removed is very small with respect to total number of rows.

   II. Check outliers in the numerical columns and check if the numerical features are normally distributed or not, if not we apply log function to square feet column, ‘longitude’ column has negative values to we can’t apply log or square root or exp. If there are outliers greater than the max value in box-plot we replace it with the max value in box-plot and if there are outliers less than minimum value of box plot, we replace it with the minimum value in box-plot.

   III. String columns are converted to lower case, then replace the nulls with median of each column.

   IV. Columns which have small number of distinct values we apply one hot encoder but if columns have large number of distinct values we apply label encoder.

   V. We create new column that have total number of rooms (feature engineering).

   VI. Check the ‘bathroom’ , ‘bedroom’ and ‘total number of rooms’, if these columns have floating point then remove it and convert it to integer.

   VII. Change the format of price display column by removing any character except numbers and convert it to integer. Remove price column as it is a redundant column.

   VIII. Check the distribution of the target if not normal apply boxcox.

   IX. At the end we apply standard scalar to the selected column in the feature selection.

2. Feature selection:

   I. Apply correlation and calculate the correlation of all features with the target.

   II. If correlation <= abs(0.1) we drop these features and the remaining features are ‘bathroom’ , ‘bedrooms’ , ‘state’ , ‘longitude’ , ‘square_feet’ and ‘total_numberofrooms’.

III. Remove ‘currency’ , ‘fee’ , ‘category’ because each column has only one value.

3. Models:

| **Random Forest** | **XG-Boost** | **Linear Regression** |
|---|---|---|
| operates by constructing a multitude of decision trees. | Gradient Boosting: Like other boosting algorithms, XG-Boost works by building a series of decision trees sequentially. However, unlike Random Forest, which builds independent trees in parallel, XG-Boost builds trees sequentially, with each tree learning from the mistakes of the previous ones. | Linear regression is a fundamental statistical technique used to model the relationship between a dependent variable and one or more independent variables. It assumes a linear relationship between the independent variables and the dependent variable. |
| Mean Squared Error (Train): 0.0002864404034994118 | Mean Squared Error (Train): 0.00019933386874848255 | Mean Squared Error (Train): 0.0005232460824986067 |
| Mean Squared Error (Validation): 0.00031694837098030133 | Mean Squared Error (Validation): 0.00024838715478999807 | Mean Squared Error (Validation): 0.0005389928508322267 |
| Mean Squared Error (Test): 0.00031312123569111487 | Mean Squared Error (Test): 0.0002268181751608491 | Mean Squared Error (Test): 0.0004981081350122221 |
| R-squared Score (Train): | R-squared Score (Train): 0.735413566925379 | R-squared Score (Train): |

| | | |
|---|---|---|
| 0.6197924360461122 | | 0.30546767863487256 |
| R-squared Score (Validation): 0.590653766714143 | R-squared Score (Validation): 0.6792021807987266 | R-squared Score (Validation): 0.303878128246034 |
| R-squared Score (Test): 0.5460385964495257 | R-squared Score (Test): 0.6711602874218658 | R-squared Score (Test): 0.2778456319292242 |
| Cross-Validation Mean Squared Error: 0.00031227078337437754 | Cross-Validation Mean Squared Error: 0.00023502579911358876 | Cross-Validation Mean Squared Error: 0.0005273512806130166 |

Polynomial regression=used when no linear realtion between features and target

Error and score:

Mean Squared Error (Train): 0.0003318817967933198

Mean Squared Error (Validation): 0.00034840633190197476

Mean Squared Error (Test): 0.0003436392663055157

R-squared Score (Train): 0.5594756607732314

R-squared Score (Validation): 0.550025074506916

R-squared Score (Test): 0.5017937275867305

Hyperparameter: 1-Rondom-Forest:

a) number of estimators=number of trees build, best value of this hyperparameter is 50 because we cannot increase the number of trees to prevent over fitting and cannot decrease number of trees too much to prevent underfitting.

b) max depth: best value of this hyperparameter is 6 because we cannot increase the depth to prevent over fitting and cannot decrease depth too much to prevent underfitting.

c) max features: The algorithm considers a maximum of "max_features" features at each split .

2-XG-Boost :

a) alpha=prevent overfiting

Size of test set is 20% , train set is 80% from them 20% for validation.

Teamid=6

| | |
|---|---|
| باسل اسلام شاكر محمد | 2021170112 |
| أحمد يحيى شلبي محمد | 2021170054 |
| اسلام محمد ابوسريع عوف | 2021170070 |
| إياد عمرو عبد الهادي إسماعيل | 2021170106 |
| أحمد إسماعيل محمود سيد | 20201701045 |