

PROJECT REPORT

(PROJECT TERM: JULY – DECEMBER 2023 (SEM- 5))

ZILLOW HOME VALUE PREDICTION

COURSE: Fundamentals of Machine Learning

COURSE CODE: INT254

SUBMITTED BY:

Name	Roll No.	Registration No.	Section
Rishabh Deo Singh	RKM014A30	12116007	KM014
Abhishek Rathore	RKM014B36	12101413	KM014
Mohammad Sahil	RKM014A18	12111970	KM014

SUBMITTED TO:

Dr. Dhanpratap Singh (25706)

In partial fulfillment of the requirements for the award of the degree of
“Bachelor Of Technology in Computer Science and Engineering”.



LOVELY
PROFESSIONAL
UNIVERSITY

Phagwara, Punjab – 144001

DECLARATION

We hereby declare that the project titled “Zillow Home Value Prediction” is an authentic record of our own work carried out as requirements of the project for the award of a B. Tech degree in Computer Science and Engineering from Lovely Professional University, Phagwara, under the guidance of Sr. Dhanpratap Singh, during July to December 2023. All the information provided in this project report is based on our own intensive work and is genuine.

Rishabh Deo Singh (12116007)

30th October 2023

Abhishek Rathore (12101413)

Mohammad Sahil (12111970)

CERTIFICATE

This is to certify that the declaration statement made by the students is correct to the best of my knowledge and belief. They have completed the project titled “Zillow Home Value Prediction” under my guidance and supervision. The present work is the result of their original investigation, effort and study. No part of the work has ever been submitted for any other degree at any university. The project is fit for the submission and partial fulfilment of the conditions for the award of a B. Tech degree in Computer Science and Engineering from Lovely Professional University, Phagwara.

Dr. Dhanpratap Singh (25706)
School of Computer Science and Engineering,
Lovely Professional University,
Phagwara, Punjab.
Date: 30th October 2023

ACKNOWLEDGEMENT

It is with our immense gratitude that I acknowledge the support and help of our professor Dr. Dhanpratap Singh (25706), who gave us the golden opportunity to do this wonderful and informative project on Zillow Home Value Prediction. He has always encouraged us in this research and project. Without his continuous guidance and persistent help, this project would not have been a success for us. During this project, we came to know about so many new things related to machine learning models and datasets.

We would also like to thank our university Lovely Professional University, Punjab and the Department of Computer Science and Engineering without which this project would have not been an achievement. We would also like to thank our family, friends, for their support and guidance.

Rishabh Deo Singh (12116007)

Abhishek Rathore (12101413)

Mohammad Sahil (12111970)

Date: 30th October 2023

TABLE OF CONTENTS

S. No.	Title	Page No.
01.	Abstract	06
02.	Introduction	07
03.	Dataset Description	08
04.	Detailed Description	10
05.	Conclusion and Future Scope	21
06.	References	22

ABSTRACT

The Zillow Home Value Prediction project is an initiative of Zillow – the leading real estate marketplace. The primary objective of this project is to develop a predictive model that could accurately estimate the values of residential properties in the United States on the basis of data provided by Zillow. This abstract provides an overview of the project, its aim and its findings.

The aim of this project titled Zillow Home Value Prediction is to predict the sale prices of the houses and improve the log error i.e. the error due to the difference between the actual and the predicted home values. Basically in this model, we are predicting the log error value from a dataset that we have taken from Kaggle. Here we are using the linear regression model of machine learning. A linear regression model is a fundamental statistical and machine learning model used for modelling the relationship between a dependent variable (or target) and one or more independent variables (or predictors).

Our approach here in this project is that first, we import the required libraries and the dataset, next perform Exploratory Data Analysis (EDA), then Feature Engineering and finally build a Linear Regression Model. We are validating the model by calculating different error values such as mean absolute error, mean squared error and root mean squared error. From our prediction, the mean absolute error which is coming is 0.05268 or 5.268%.

INTRODUCTION

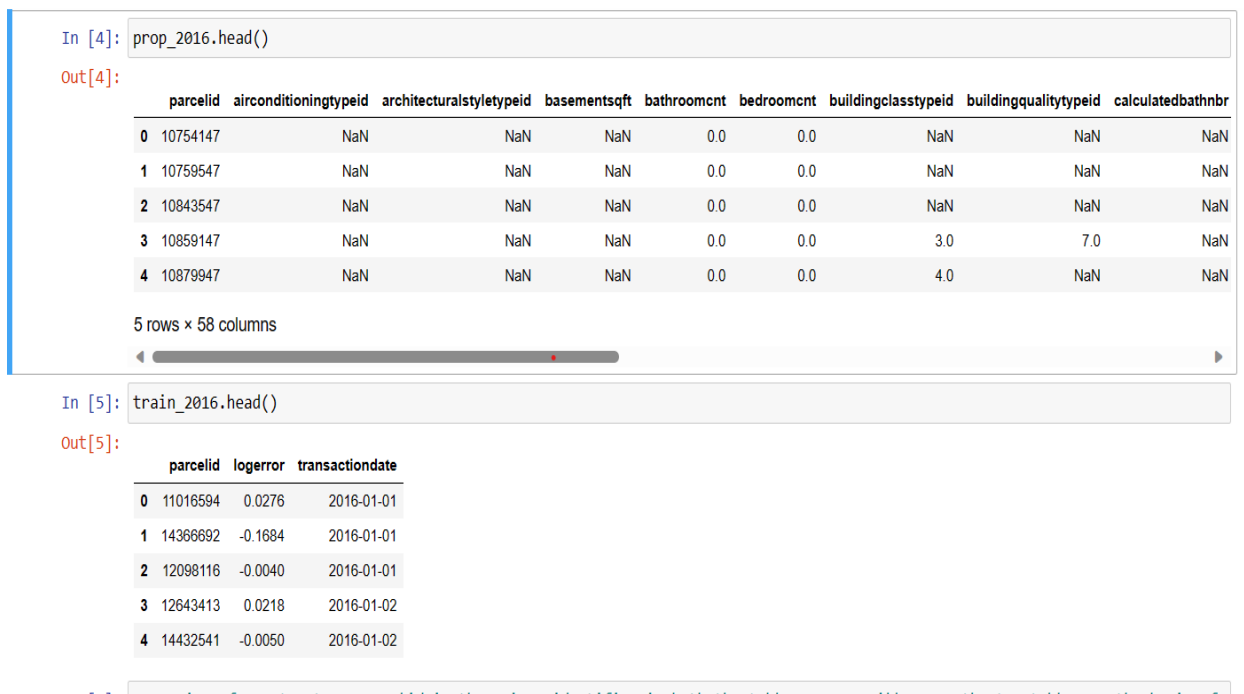
Our project titled Zillow House Value Prediction is based on a United States-based real estate company Zillow. Zillow is an American online real estate company that provides a wide range of services related to real estate and housing. It was founded in 2006 and has become one of the leading online real estate marketplaces in the United States. Zillow is known for its innovative approach to using technology and data to empower homeowners, buyers, sellers, and renters in the real estate market. Some key functionalities of Zillow are Real Estate Listings, Zestimate, Home Buying and Selling tools, Market Trends and Data, Mortgage information, Rental information, Real Estate Research and Blog and so on.

Buying a house that suits their choices is every person's desire, and it is thus known as their dream house. One considers several aspects while purchasing a home, starting from the budget, the location, the number of rooms available, and many more. But how to find a house that satisfies one's requirements? This is not a quick and easy task. But no need to worry; homebuyers can nowadays find their dream home with the click of a button. Zillow is a popular estimator for house evaluation available online. The Zillow Zestimate provides homebuyers with information on the actual worth of the house based on public data. The accuracy of the Zestimate information depends on the location and availability of the data in a specific area.

The aim of this project is to predict the sale prices of houses and improve the log error i.e., the error due to the difference between the actual and the predicted home values. In this project basically, we are predicting log error value by combining two datasets of 2016 and then validating our prediction value with respect to actual value. In this project our approach or the steps we follow to make predictions are Importing the required libraries and the dataset, Merging the two datasets, Exploratory Data Analysis (EDA), Feature Engineering, Model Building and finally Model Validation. For model building, we are using the Linear Regression Model and for validation, we are using three different types of errors namely mean absolute error, mean squared error and root mean squared error.

DATASET DESCRIPTION

The dataset that we are using in this project has been taken from the Kaggle website. There are two data that we are using properties_2016.csv and train_2016_v2.csv. We are merging both these two data into one data so that we can use it in our project. This dataset is provided in one of the Zillow Prize competitions and is a comprehensive dataset that includes a wide range of information related to residential real estate properties in the United States. The dataset is designed for the purpose of predicting the estimated value of these properties. A snapshot of the dataset is –



The screenshot displays two Jupyter Notebook cells. The first cell, labeled 'In [4]:', contains the command `prop_2016.head()`. The output, labeled 'Out[4]:', shows a table with 10 columns: `parcelid`, `airconditioningtypeid`, `architecturalstyletypeid`, `basementsqft`, `bathroomcnt`, `bedroomcnt`, `buildingclasstypeid`, `buildingqualitytypeid`, and `calculatedbathnbr`. The first five rows of data are shown, with `parcelid` values ranging from 10754147 to 10879947. The second cell, labeled 'In [5]:', contains the command `train_2016.head()`. The output, labeled 'Out[5]:', shows a table with 3 columns: `parcelid`, `logerror`, and `transactiondate`. The first five rows of data are shown, with `parcelid` values ranging from 11016594 to 14432541.

	parcelid	airconditioningtypeid	architecturalstyletypeid	basementsqft	bathroomcnt	bedroomcnt	buildingclasstypeid	buildingqualitytypeid	calculatedbathnbr
0	10754147	NaN	NaN	NaN	0.0	0.0	NaN	NaN	NaN
1	10759547	NaN	NaN	NaN	0.0	0.0	NaN	NaN	NaN
2	10843547	NaN	NaN	NaN	0.0	0.0	NaN	NaN	NaN
3	10859147	NaN	NaN	NaN	0.0	0.0	3.0	7.0	NaN
4	10879947	NaN	NaN	NaN	0.0	0.0	4.0	NaN	NaN

5 rows x 58 columns

	parcelid	logerror	transactiondate
0	11016594	0.0276	2016-01-01
1	14366692	-0.1684	2016-01-01
2	12098116	-0.0040	2016-01-01
3	12643413	0.0218	2016-01-02
4	14432541	-0.0050	2016-01-02

The above figure shows two data properties_2016.csv (prop_2016) and train_2016_v2.csv (train_2016). These two data are merged together to use in our Zillow Home Value Prediction Project.

Below is the description of the key features and information included in this dataset:

1. Parcel ID: A unique identifier for each property, used to associate the property with its specific information.
2. Air Conditioning: Information about the type of air conditioning system installed in the property.
3. Bathroom Count: The total count of bathrooms in the property, including both full and partial bathrooms.
4. Bedroom Count: The total count of bedrooms in the property.
5. Building Quality: A measure of the overall quality of the building's construction and materials.
6. Calculated Finished Square Foot: The total area of the property's living space, measured in square feet.
7. FIPS Code: A Federal Information Processing Standards (FIPS) code that identifies the county in which the property is located.
8. Fireplace Count: The count of fireplaces in the property.
9. Garage Count: The count of garages or parking spaces available on the property.
10. Heating System: Information about the type of heating system installed in the property.
11. Latitude and Longitude: Geographic coordinates indicating the property's location.
12. Lot Size Square Feet: The total area of the property's lot, measured in square feet.
13. Property Tax Amount: The amount of property tax assessed on the property.
14. Year Built: The year the property was constructed.
15. Zestimate: The Zillow Estimate, which is an automated valuation of the property's current market value.
16. Log Error: The natural logarithm of the difference between the Zestimate and the actual sales price of the property.
17. Transaction Date: The date when the property transaction occurred.
18. Raw Census Tract and Block ID: Geographic identifiers that specify the property's location within a census tract.
19. Room Count: The total count of rooms in the property, including bedrooms, bathrooms, and other living spaces.
20. Story Count: The total number of stories or levels in the property.

21. Tax Assessed Year: The year in which the property tax assessment was made.

These are some of the key features in our dataset. The total number of features in properties_2016.csv is 58 and in train_2016_v2.csv is 3. When we merge these two data together the total number of features is 60. The dataset provides valuable information for understanding property characteristics, geographic features, and tax-related data, making it suitable for real estate market analysis and predictive modelling.

DETAILED DESCRIPTION

1- Importing the required libraries and dataset:

The first step includes importing the required libraries and datasets refers to the initial steps in a data analysis data science project or machine learning project where we set our programming environment to work with the necessary tools and data. Libraries or packages are collections of pre-written code and functions that provide additional capabilities for our programming language. Importing a library means making those functions and capabilities available for use in our code. Some key libraries which we imported in our project are:

```
In [49]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [50]: import math
from math import sqrt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.ensemble import RandomForestRegressor
```

Datasets are collections of data that we want to analyze, explore or model. Loading a dataset means bringing the data into our programming environment so that we can work with it. We are loading datasets in our project using pandas library `pd.read_csv()` method as –

```
prop_2016 = pd.read_csv('properties_2016.csv', low_memory=False)
print('Shape of properties_2016 dataset: ', prop_2016.shape)
```

```
Shape of properties_2016 dataset: (2985217, 58)
```

```
train_2016 = pd.read_csv('train_2016_v2.csv', low_memory=False)
print('Shape of train_2016 dataset: ', train_2016.shape)
```

```
Shape of train_2016 dataset: (90275, 3)
```

After importing libraries and loading the datasets we are performing two operations:

- Merging of the two datasets
- Understanding the dataset

We are merging two datasets using the `parcelid` feature. While understanding datasets we came to know that `parcelid` is the unique identifier between the two tables. Therefore, we merged the two tables based on `parcelid` to form our complete dataset and then go ahead with Exploratory Data Analysis techniques.

```
final_data = prop_2016.copy()
final_data = final_data.merge(train_2016, on = 'parcelid')
```

```
final_data.head()
```

2- Exploratory Data Analysis (EDA) –

Exploratory Data Analysis (EDA) is a step in the data analysis process, where the primary focus is to summarize the main characteristics of the data, often with the help of visual and quantitative methods. The goal of EDA is to understand the data, identify patterns, relationships, and anomalies, and extract insights that may lead to further analysis or hypothesis testing.

In our project in Exploratory Data Analysis, we are exploring the –

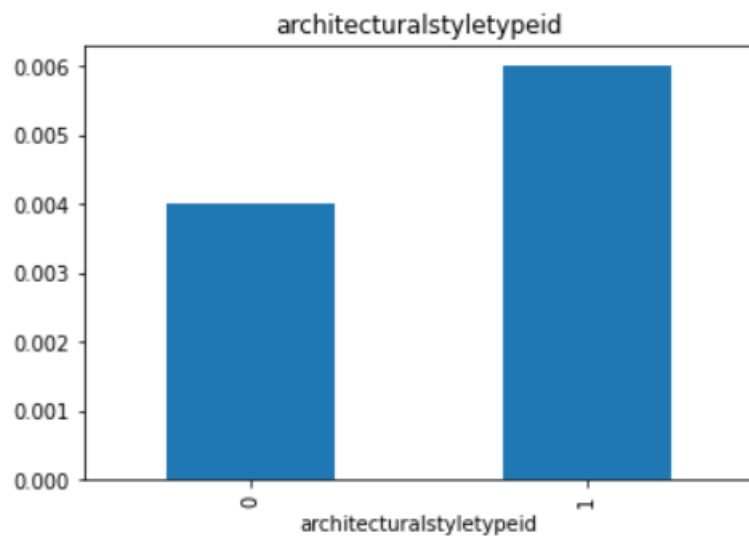
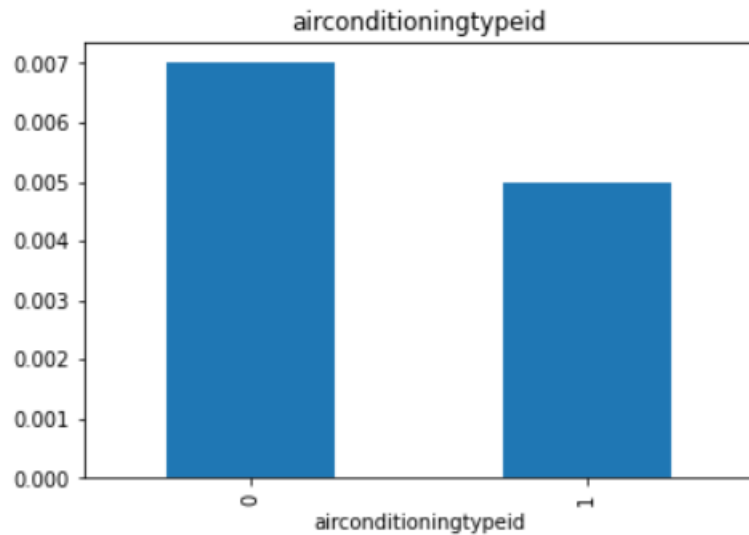
- Missing values
- Distribution of the numerical values
- Outliers
- Distribution of the categorical values
- Potential Relationships between the variables and the target

Firstly, we are checking for missing values null or nan by the following code-

```
missing_values = []  
for var in copy_data.columns:  
    if (copy_data[var].isnull().sum() > 0):  
        missing_values.append(var)
```

```
copy_data[missing_values].isnull().sum()
```

After this, we are visualizing the relation between missing values and log errors by the use of a graph. We are creating a graph for each feature against log error where we are assuming the 1 means missing value and 0 means no missing value. Some examples are –



We are now exploring distribution of numerical values. It refers to how the numeric features in our dataset are spread or distributed. We are performing it by –

```
In [18]: numerical_values = []
for var in copy_data.columns:
    if copy_data[var].dtypes != 'O':
        numerical_values.append(var)

print('Total number of Numerical Data: ', len(numerical_values))

Total number of Numerical Data: 54
```

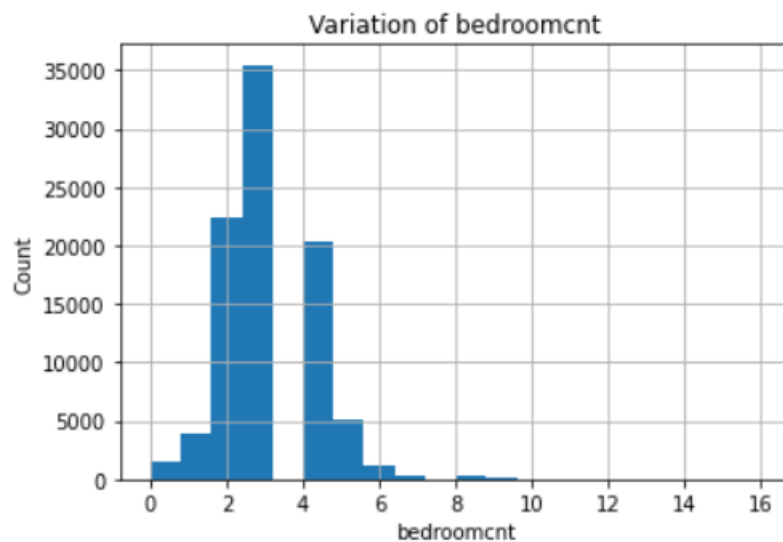
The distribution of categorical variables is the next step of EDA which refers to the qualitative or discrete variables, that represent data that can take on a limited, fixed number of values or categories. These variables are often used to label or group data into distinct classes. We are finding the distribution of categorical variables in our project as –

```
categorical_variables = []
for var in copy_data.columns:
    if copy_data[var].dtypes=='O':
        categorical_variables.append(var)

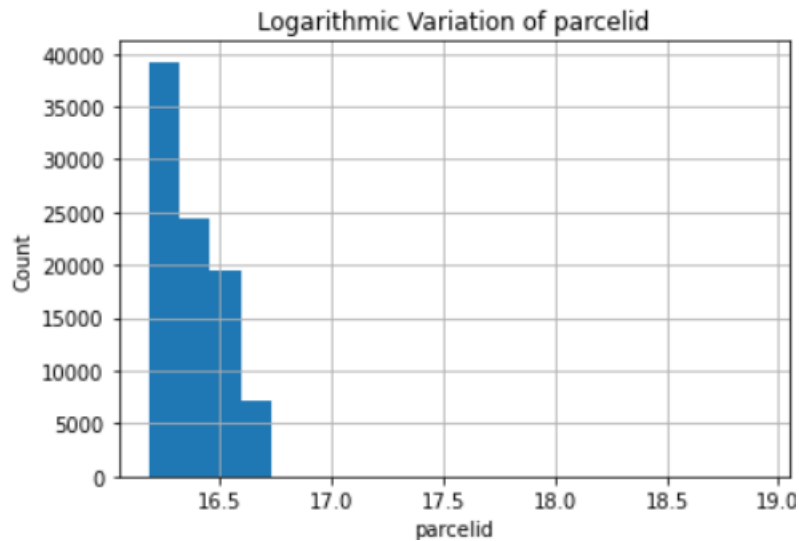
print('Total number of Categorical Variables: ', len(categorical_variables))
```

Total number of Categorical Variables: 6

After analyzing the categorical variables, we are now analyzing the discrete variables. Discrete variables in ml refers to those variables that have a limited number of distinct values, often categorically representing different groups or options. We are analyzing discrete variable by a histogram –

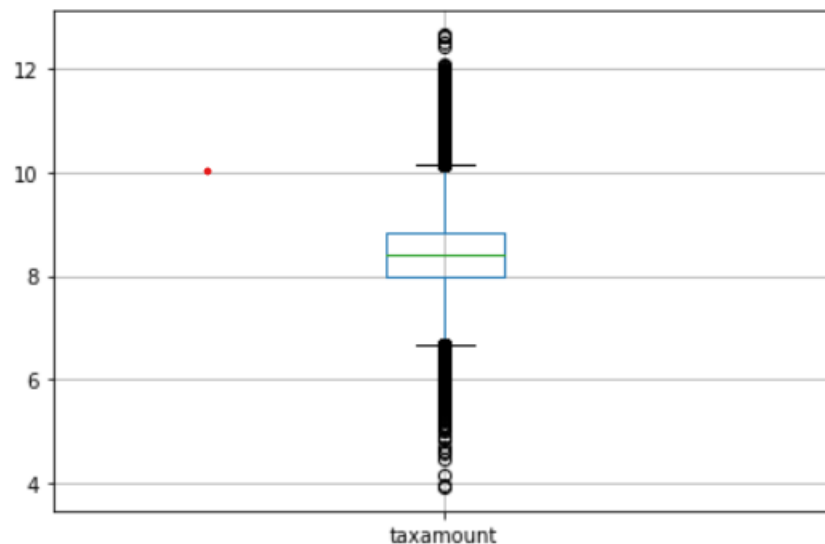


In our data analysis we came to know that some of the variables are not normally distributed so we are evaluating whether a logarithmic transformation of the variables returns values that will follow normal distribution or not.



After performing logarithmic transformation and analyzing graph we came to know that a better spread of the values for only few variables when we use the logarithmic transformation.

At the final step of the EDA we are analyzing outliers. Outliers are basically data points that deviate significantly from the majority of the data in a dataset. They are observations that are unusual or exceptional in some way and do not conform to the general patterns or trends present in the data.



The above graph shows the outliers in a taxamount feature. From observations we see that there are outliers in some of the variables that we remove with feature engineering and build a model on top of it.

3- Feature Engineering –

Feature Engineering refers to the process of creating new features or modifying existing features in a dataset to improve the performance of machine learning models. It is a crucial step in the data preprocessing and model development pipeline. It leads to better model accuracy, generalization, and interpretability. After doing feature engineering we are saving our data in new csv file named Zillow_Feature_Engineering_Dataset.csv and importing this dataset for further project.

Duplicate check – When we are checking duplicate id in our data we observe that there are 125 rows of duplicate data with respect to the parcelid. So, we drop all these rows by following code –

```
data.drop_duplicates(subset ="parcelid", keep = 'first', inplace = True)
```

```
data.shape
```

```
(90150, 60)
```

In the original data the shape is (90275, 60) but after removing duplicate rows the shape is (90150, 60).

Missing values – Next we are analyzing columns with missing values and dropping all those columns which have more than 60% of missing values as these columns have no use in prediction model and they unnecessary create more complexity. After dropping these columns the new shape of our data comes out to be (90150, 31).

Elapsed Time – We are now analyzing the elapsed time means all the time columns and converting them into one column basically we are capturing elapsed time as –

```
# Capture Elapsed Time
```

```
data['yeardifference'] = data['assessmentyear'] - data['yearbuilt']
```

After dropping these columns our data reduces to (90150, 29).

Transforming Variables – Next we are transforming incorrectly scaled variables such as latitude and longitude has been multiplied to 10^6 , rawcensustractandblock to 10^6 and censustractandblock to 10^{12} .

```
[n [183]: data[['latitude', 'longitude']] = (data[['latitude', 'longitude']])/(10**6)
data['censustractandblock'] = (data['censustractandblock'])/(10**12)
data['rawcensustractandblock'] = (data['rawcensustractandblock'])/(10**6)
```

Replacing Missing values – We are replacing missing values in the original variable with the mode as –

```
def replace_missing_data(df, miss_var):
    print("Missing Values - Mode of features")

    for var in miss_var:
        data[var] = data[var].fillna(data[var].mode()[0])
    return data

missing_var = []
for var in data.columns:
    if data[var].isnull().sum() > 0:
        missing_var.append(var)
data = replace_missing_data(data, missing_var)
```

Missing Values - Mode of features

	parcelid	bathroomcnt	bedroomcnt	buildingqualitytypeid	calculatedbathnbr	calculatedfinishedsquarefeet	finishedsquarefeet12	fips
0	17073783	2.5	3.0	7.0	2.5	1264.0	1264.0	6111.0
1	17088994	1.0	2.0	7.0	1.0	777.0	777.0	6111.0
2	17100444	2.0	3.0	7.0	2.0	1101.0	1101.0	6111.0
3	17102429	1.5	2.0	7.0	1.5	1554.0	1554.0	6111.0
4	17109604	2.5	4.0	7.0	2.5	2415.0	2415.0	6111.0
5	17125829	2.5	4.0	7.0	2.5	2882.0	2882.0	6111.0

In the above figure, it is shown clearly that the null value in buildingqualitytypeid is replaced by mode value 7.0

Encoding Categorical Variables – We are now encoding the two categorical variables propertycountylandusecode and propertyzoningdesc.

propertycountylandusecode_labels	propertyzoningdesc_labels
54	609
55	609
50	609
49	609
50	609
50	609
50	609

Multicollinearity – At the end, we are checking for multicollinearity and dropping those columns which are highly correlated as they does not affect the result. We are removing these columns and after removal, the final dataset shape is(70260, 21) on which we build our Linear Regression Model.

property	parcelid	1	0.028	0.014	0.43	0.051	0.88	-0.2	-0.8	0.87	-0.21	0.046	0.88	0.17	-0.88	0.52	0.52	-0.033	0.037	0.87	0.025	-0.22	0.85	-0.21
	bathroomcnt	0.028	1	0.58	-0.3	0.72	0.096	-0.42	0.026	0.034	0.023	0.017	0.098	-0.025	-0.096	0.16	0.04	0.071	0.39	0.099	0.046	-0.46	0.094	0.018
	bedroomcnt	0.014	0.58	1	-0.094	0.65	0.062	-0.15	0.067	0.055	-0.18	-0.38	0.064	-0.017	-0.062	0.15	0.2	0.092	0.23	0.068	0.035	-0.076	0.0066	0.043
	buildingqualitytypeid	0.43	-0.3	-0.094	1	-0.21	0.42	0.28	-0.4	0.37	-0.23	-0.21	0.42	0.049	-0.42	0.19	0.28	0.049	-0.083	0.42	-0.0077	0.23	0.39	-0.19
	finishedsquarefeet12	0.051	0.72	0.65	-0.21	1	0.13	-0.3	0.0085	0.069	-0.12	-0.2	0.13	-0.044	-0.13	0.19	0.13	-0.051	0.5	0.13	0.051	-0.27	0.071	0.00085
	fips	0.88	0.096	0.062	0.42	0.13	1	-0.29	-0.68	0.71	-0.2	0.019	1	0.053	-1	0.67	0.65	-0.044	0.075	0.99	0.019	-0.29	0.97	-0.32
	heatingorsystemtypeid	-0.2	-0.42	-0.15	0.28	-0.3	-0.29	1	0.046	-0.17	-0.12	-0.2	-0.3	0.016	0.29	-0.33	-0.17	-0.042	-0.14	-0.3	-0.014	0.52	-0.33	0.024
	latitude	-0.8	0.026	0.067	-0.4	0.0085	-0.68	0.046	1	-0.59	0.17	-0.047	-0.68	-0.16	0.68	-0.12	-0.4	0.004	-0.12	-0.67	-0.019	0.037	-0.67	0.29
	longitude	0.87	0.034	0.055	0.37	0.069	0.71	-0.17	-0.59	1	0.21	-0.029	0.71	0.13	0.71	0.61	0.39	-0.032	-0.035	0.7	0.023	-0.23	0.68	0.13
	lotssquarefeet	-0.21	0.023	-0.18	-0.23	-0.12	-0.2	0.12	0.17	-0.21	1	0.35	-0.2	-0.019	0.2	-0.13	-0.13	-0.024	-0.083	-0.2	-0.002	-0.16	-0.14	0.062
	propertylandusetypeid	-0.046	0.017	-0.38	-0.21	-0.2	0.019	-0.2	-0.047	-0.029	0.35	1	0.02	0.056	-0.019	-0.018	-0.2	-0.46	-0.071	0.017	0.0021	-0.45	0.092	0.017
	rawcensustractandblock	0.88	0.098	0.064	0.42	0.13	1	-0.3	-0.68	0.71	-0.2	0.02	1	0.06	-1	0.67	0.65	-0.044	0.074	0.99	0.019	-0.3	0.97	-0.31
	regionidcity	0.17	-0.025	-0.017	0.049	-0.044	0.053	0.016	-0.16	0.13	0.019	0.056	0.06	1	-0.053	-0.001	0.0034	-0.013	0.015	0.058	0.011	-0.037	0.042	0.19
	regionidcounty	-0.88	-0.096	-0.062	-0.42	-0.13	-1	0.29	0.68	-0.71	0.2	-0.019	-1	-0.053	1	-0.67	-0.65	0.044	-0.075	-0.99	-0.019	0.29	-0.97	0.32
	regionidzip	0.52	0.16	0.15	0.19	0.19	0.67	-0.33	-0.12	0.61	-0.13	-0.018	0.67	-0.001	-0.67	1	0.42	-0.064	-0.042	0.67	0.016	-0.42	0.64	-0.093
	roomcnt	0.52	0.04	0.2	0.28	0.13	0.65	-0.17	-0.4	0.39	-0.13	-0.2	0.65	0.0034	-0.65	0.42	1	-0.035	-0.03	0.66	0.016	0.021	0.62	-0.21
	unitcnt	-0.033	0.071	0.092	0.049	-0.051	-0.044	-0.042	0.004	-0.032	-0.024	-0.46	-0.044	-0.013	0.044	-0.064	-0.035	1	0.01	-0.044	-0.0087	0.11	0.042	-0.011
	taxamount	-0.037	0.39	0.23	0.083	0.5	0.075	-0.14	-0.12	-0.035	-0.083	-0.071	0.074	0.015	-0.075	-0.042	-0.03	0.01	1	0.069	-0.024	-0.13	0.033	0.016
	censustractandblock	0.87	0.099	0.068	0.42	0.13	0.99	-0.3	-0.67	0.7	-0.2	0.017	0.99	0.058	-0.99	0.67	0.66	-0.044	0.069	1	0.02	-0.29	0.97	-0.31
	logerror	0.025	0.046	0.035	-0.0077	0.051	0.019	-0.014	-0.019	0.023	-0.002	0.0021	0.019	0.011	-0.019	0.016	0.016	-0.0087	-0.024	0.02	1	-0.028	0.016	-0.014
	yeardifference	-0.22	-0.46	-0.076	0.23	-0.27	-0.29	0.52	0.037	-0.23	-0.16	-0.45	-0.3	-0.037	0.29	-0.42	0.021	0.11	-0.13	-0.29	-0.028	1	-0.32	-0.018
	propertycountylandusecode_labels	0.85	0.094	0.0066	0.39	0.071	0.97	0.33	0.67	0.68	-0.14	0.092	0.97	0.042	-0.97	0.64	0.62	0.042	0.033	0.97	0.016	0.32	1	0.31
	propertyzoningdesc_labels	-0.21	0.018	0.043	-0.19	0.00085	-0.32	0.024	0.29	-0.13	0.062	0.017	-0.31	0.19	0.32	-0.093	-0.21	-0.011	-0.016	-0.31	-0.014	-0.018	-0.31	1
		parcelid	bathroomcnt	bedroomcnt	buildingqualitytypeid	finishedsquarefeet12	fips	heatingorsystemtypeid	latitude	longitude	lotssquarefeet	propertylandusetypeid	rawcensustractandblock	regionidcity	regionidcounty	regionidzip	roomcnt	unitcnt	taxamount	censustractandblock	logerror	yeardifference	propertycountylandusecode_labels	propertyzoningdesc_labels

4- Linear Regression Model –

We are now building a linear regression model on our data. For this we first divide our data into two variables X and y where X consists of all features except target feature (logerror) and y consists of target feature(logerror). We are dividing these X and y into train-test set in the ratio 80:20 i.e. 80% for training and 20% for testing our data using sklearn library.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state = 42)

X_train.shape, X_test.shape
((56208, 20), (14052, 20))

y_train.shape, y_test.shape
((56208,), (14052,))
```

The model –

```
# Linear Regression Model

linear_regression = LinearRegression()
linear_regression.fit(X_train, y_train)

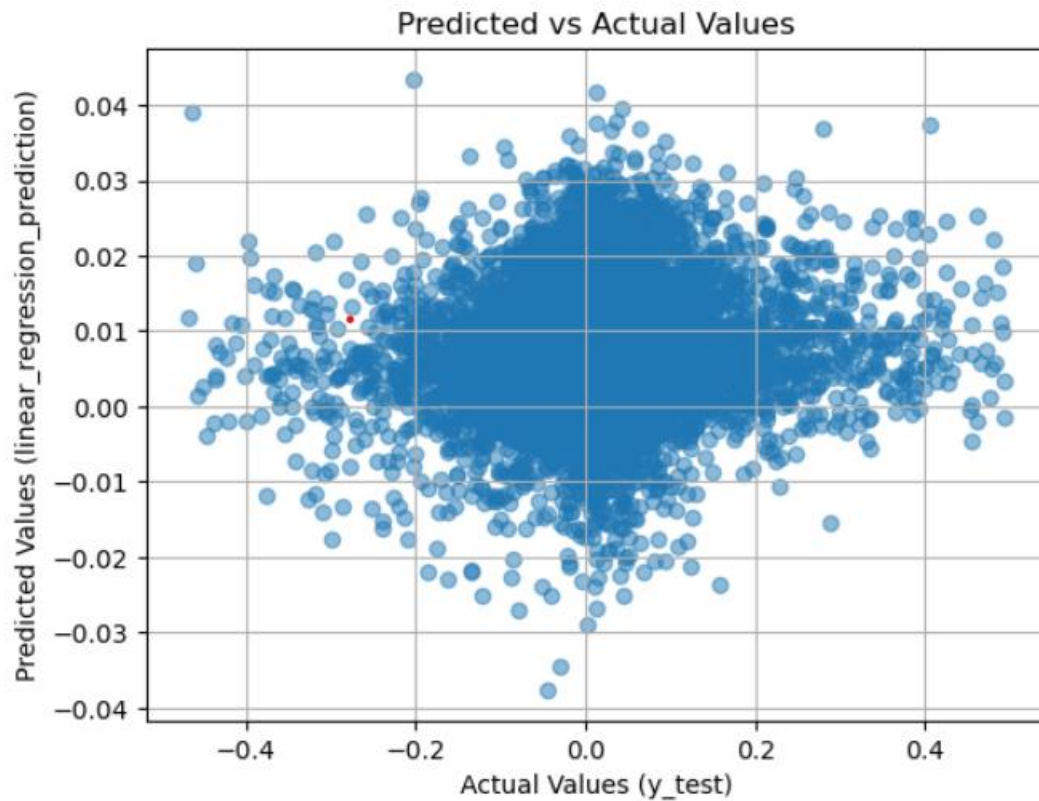
▼ LinearRegression
LinearRegression()

linear_regression_prediction = linear_regression.predict(X_test)

linear_regression_prediction
array([-0.00565632, -0.0007824 ,  0.01668563, ...,  0.01559079,
        0.00250482,  0.00433461])

y_test
14808    0.3358
27489    0.2135
69881    0.0159
36967    0.0000
9803     0.0488
...
62060   -0.0182
24167    0.1231
-----
```

The above figure shows the predicted values as well as the y_{test} value. The graph showing Predicted vs Actual Values is –



Calculation of errors: Mean Absolute error = 0.052683398591761826 or 5.27%
Mean Squared Error = 0.0071510871956413785 or 0.72%
Root Mean Squared Error = 0.08456410110467312 or 8.4%

CONCLUSION AND FUTURE SCOPE

The results obtained above are very promising and effective. This approach finds The logerror value with very less error as we can see above. The predicted values are very close to the actual values as we can see in graph shown above. Our model works satisfactorily and all the intermediate steps are also showing good promises. We have learnt many new things such as EDA, Feature Engineering, Data Preprocessing, Linear Regression Model and Validating Model.

Though our model works satisfactorily there is a scope of future improvement in our project. We can test other advanced models of machine learning on this dataset to improve efficiency and correctness of logerror value. Additional machine learning models, including deep learning and ensemble techniques, can be tested for improved predictive accuracy. More advanced feature engineering techniques can be explored to create new variable or transform existing ones. We can also incorporate external data like integrating external datasets such as local economic data, neighbourhood crime rates, or school quality to enhance the model's predictive power. We can also utilize real-time data sources to keep the model updated and relevant.

In conclusion, the Zillow Home Value Prediction project has the potential to offer valuable tools and insights for homebuyers, sellers, and real estate professionals. With continuous development and refinement, it can serve as a reliable resource for property valuation and real estate decision-making. The project's future scope includes advancing the predictive model, incorporating additional data sources, addressing privacy and ethical concerns to provide a comprehensive and reliable solution for property value prediction.

REFERENCES

- [1] Kaggle Zillow Prize: Zillow's Home Value Prediction (Zestimate), 2017.
Available: <https://www.kaggle.com/c/zillow-prize-1>

- [2] Simple Exploration Notebook – Zillow Prize by DKSDMS4.
Available: <https://www.kaggle.com/code/dksdms4/simple-exploration-notebook-zillow-prizenotebook-zillow-prize>

- [3] Zillow's Home Value Prediction (Zestimate) by Amitabh Priyadarshi.
Available:
<https://www.kaggle.com/code/amitabhpriyadarshi/zillow-s-home-value-prediction-zestimate>

- [4] Datasets - <https://www.kaggle.com/competitions/zillow-prize-1/data>