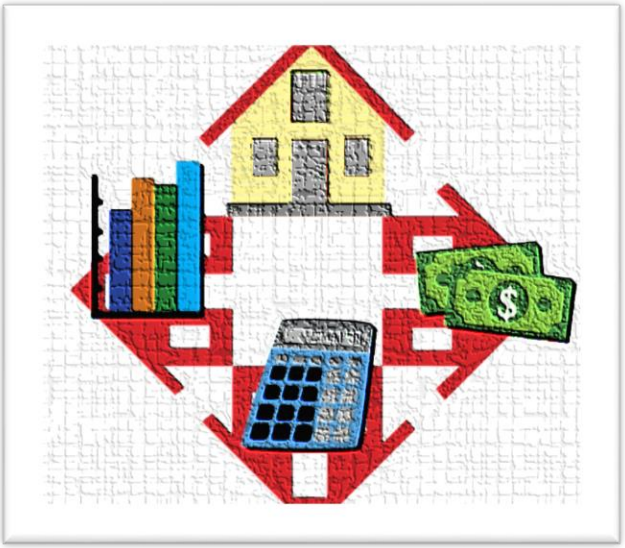
Sberbank Russian Housing Project



CONTENTS

1. Problem Statement

Housing costs demand a significant investment from both consumers and developers. Sberbank, Russia's oldest and largest bank, helps their customers by making predictions about realty prices so renters, developers, and lenders are more confident when they sign a lease or purchase a building.

In this project, we will develop regression algorithms which use a broad spectrum of features to predict realty prices. We will rely on a rich dataset that includes housing data and macroeconomic patterns.

2. Data used

As provided there are 3 different data sources called Macro, Train and Test. For the period of August 2011-June 2015, we have Macro and Train data. Macro contains 100 variables and 30471 observations which are the macroeconomic patterns affecting the housing. Train contains 292 variables and 30471 observations which are a set of housing realty prices. Test contains 291 variables and 30471 observations which are the housing patterns of Russia in the period July 2016-May 2016.

Based on the previous economic trends and housing patterns, we can predict the realty prices for the specified period.

2.1 Variables used:

Some crucial features are listed below.

Features of macro:

- gdp_quart
- gdp_quart_growth
- gdp_annual
- gdp_annual_growth
- salary
- salary_growth
- modern_education_share
- old_education_build_share
- apartment_fund_sqm

Features of train and test:

- price_doc**(target variable)**
- full_sq
- floor
- max_floor
- material
- build_year
- num_room
- kitch_sq

3. Approach

3.1 General Machine Learning Approach:

To speak of our model in a nutshell, it is a black box. We provide the set of inputs which is processed and then we obtain a vector of predictions.
The machine learning algorithm has been trained over time with data from the past. Thus, based on the previous trends, it can predict to the current trend.

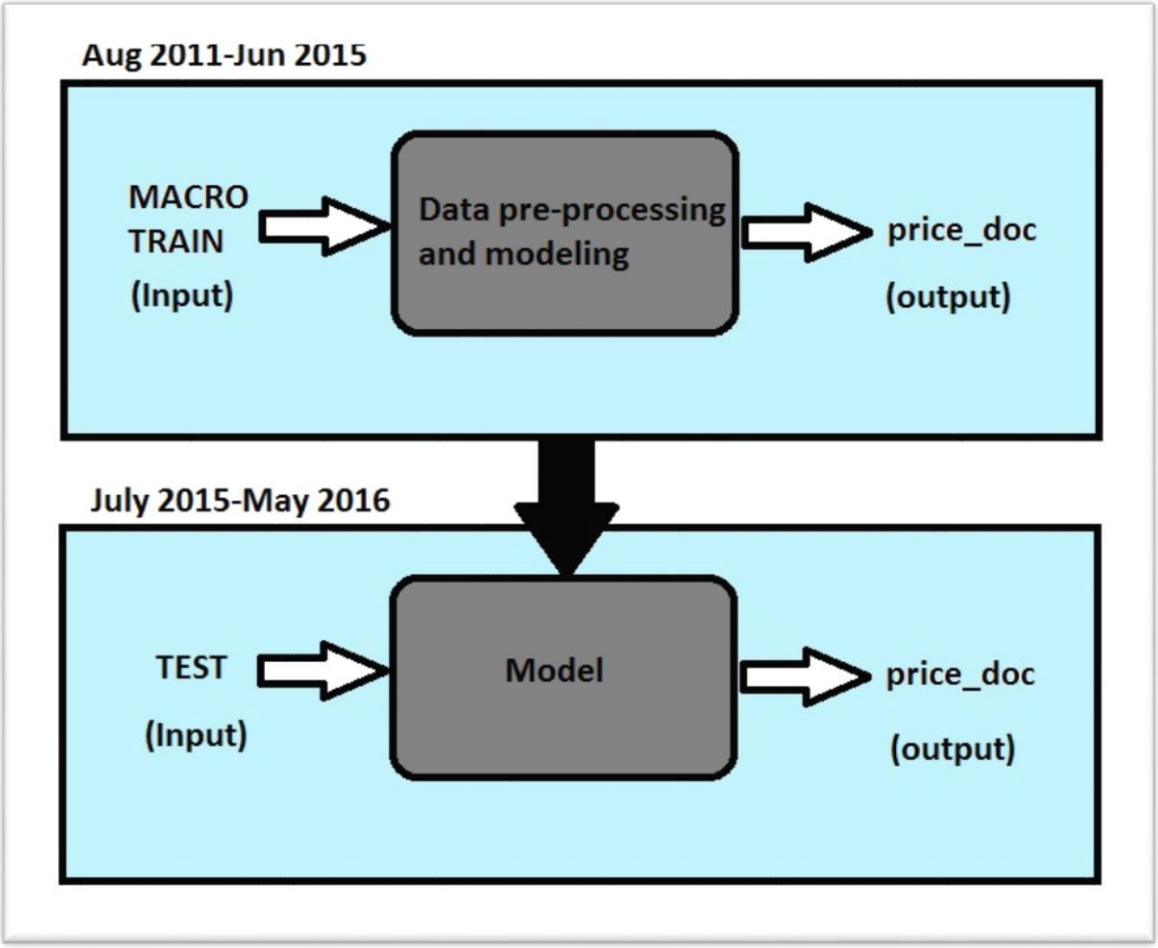


Figure 1 General Machine Learning Model

3.2 Regression Prediction Approach

Majority of the data type is numerical and thus we rely more towards regression modeling.

The basic and critical step in the modeling is the data pre-processing. Since we have a huge data sets, we have to tread carefully through the pre-processing till we get cleansed data suitable for feeding our Machine Learning algorithm.

4. Exploration of Variables:

The economic data alone has 100 variables and 30471 observations. We have 96 numerical and 4 categorical variables. The train data has 176 numerical and 16 categorical variables.

4.1 Initial cleaning:

Analyzing **the** data, we observe that there is lot of scope for repairing the data.
For example, the categorical variable "child_on_acc_pre_school" has a comma in place of a decimal point. The fraction of 1000 has been represented inaccurately. We rectify this by replacing the comma by decimal point.
Ex: 45,983 => 45.983
Another example for train can be, replacing shorthanded values for "build_year" by the complete years.
Ex: 17 => 2017

4.2 Checking the distribution of the numerical variables:

Histograms were plotted to check the distribution of the numerical variables. I have provided the graph of macro for the variable "ppi".

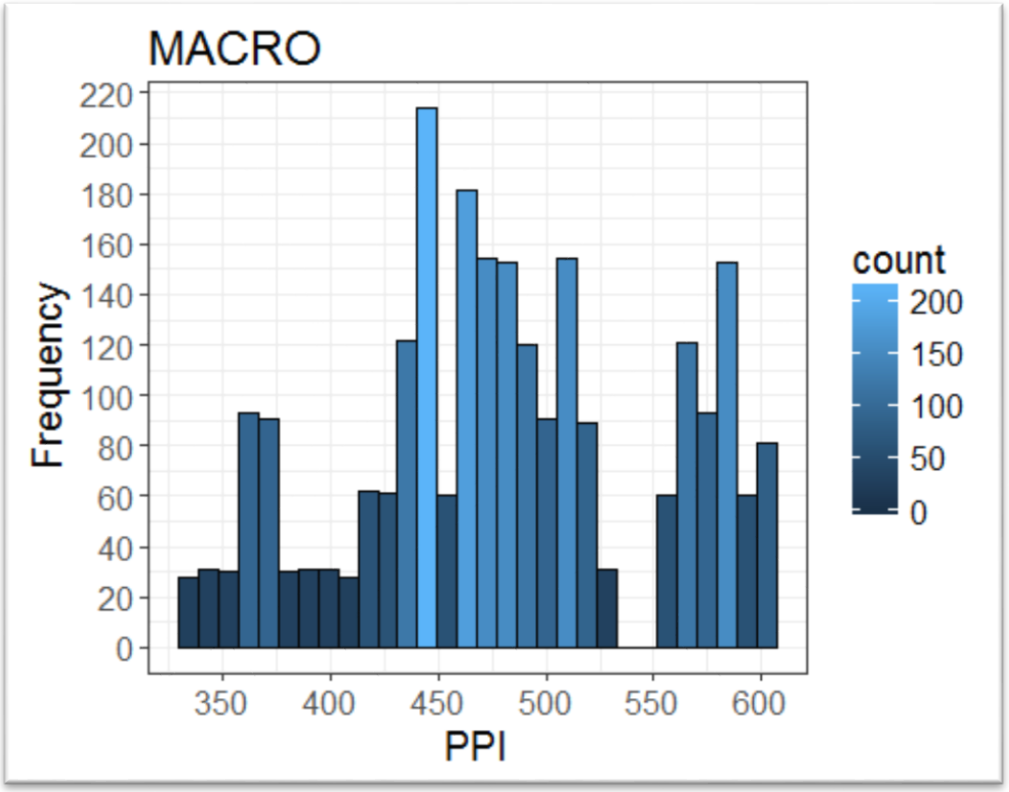


Figure 2 Histogram of macro feature PPI

We observe that the data is closer to normal distribution. We can repeat this with other variables as well.
We repeat this with train variables too. Find the histogram for train variable “raoin_popul”:

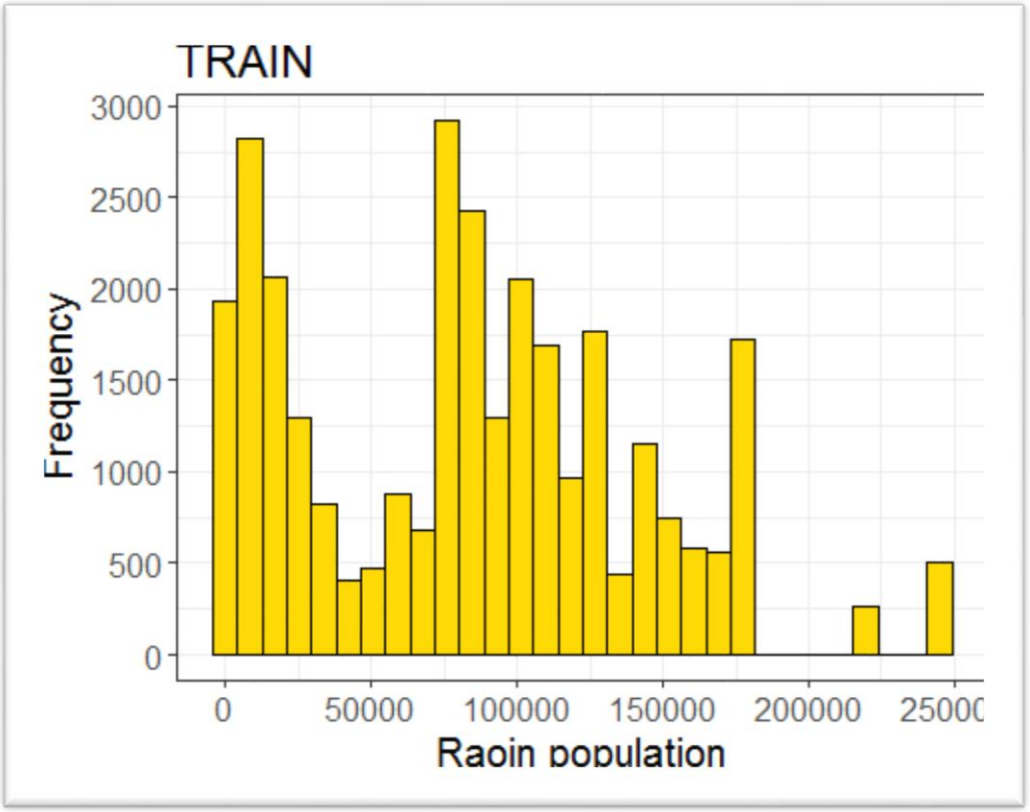


Figure 3 Histogram of ppi train feature

4.3 Checking categorical variables:

	train.ecology	train.sub_area
1	good	Bibirevo
2	Excellent	Tekstil'shhiki
3	Poor	Tekstil'shhiki
4	Good	Mitino
5	Excellent	Basmannoe

Subset of categorical values are represented here.

5. Dealing with missing values:

There are multiple missing values in the macro and train datasets.

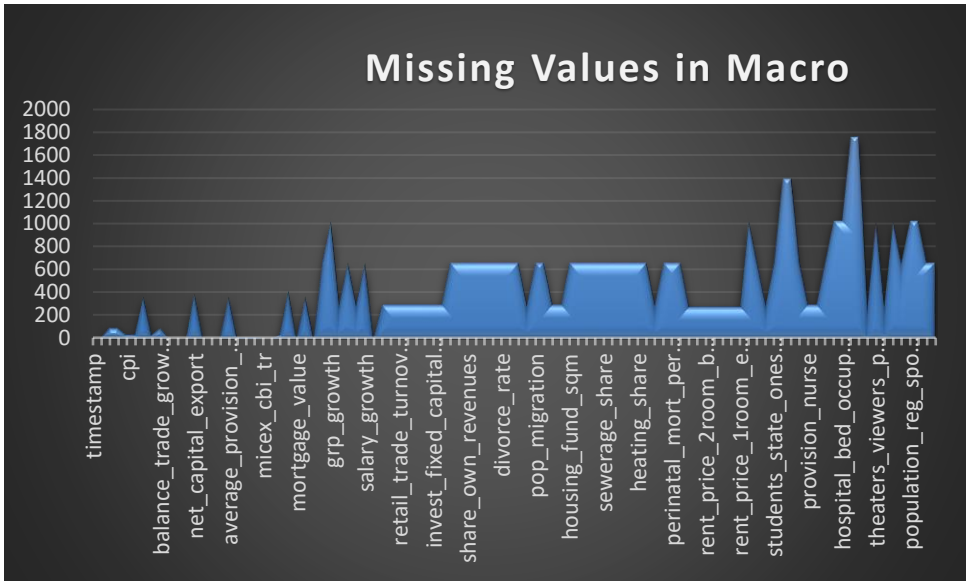


Figure 4 Missing value plot in Macro

We notice a trend where most of the numerical variables have missing data whereas categorical have very few missing data based on above plot.
In the below graph, we can see the spread of missing values in the train dataset.
In the consistent plot, we can see the proportion of missing values with variables.
Based on these trends we identify the variables that needs tending and eliminate or impute the missing values.

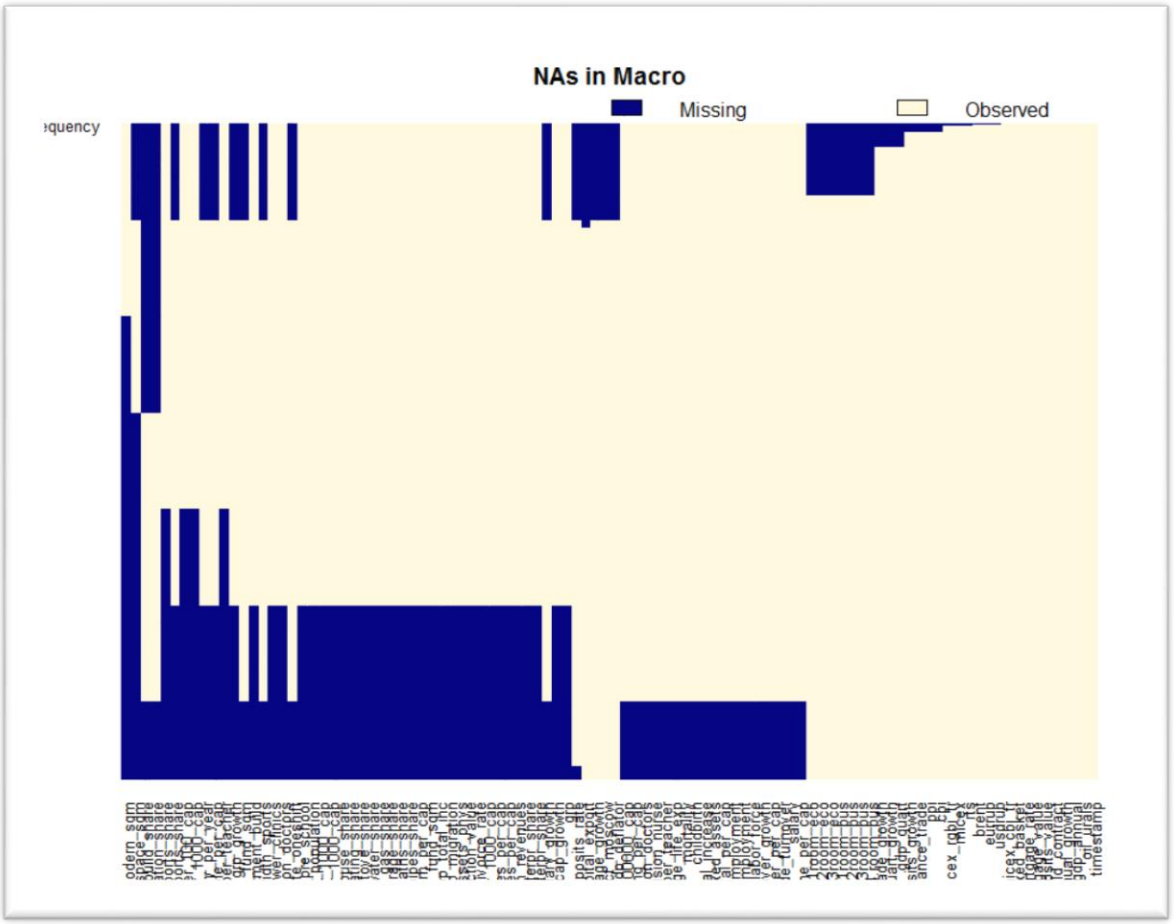


Figure 5 Missing value frequency plot of variables

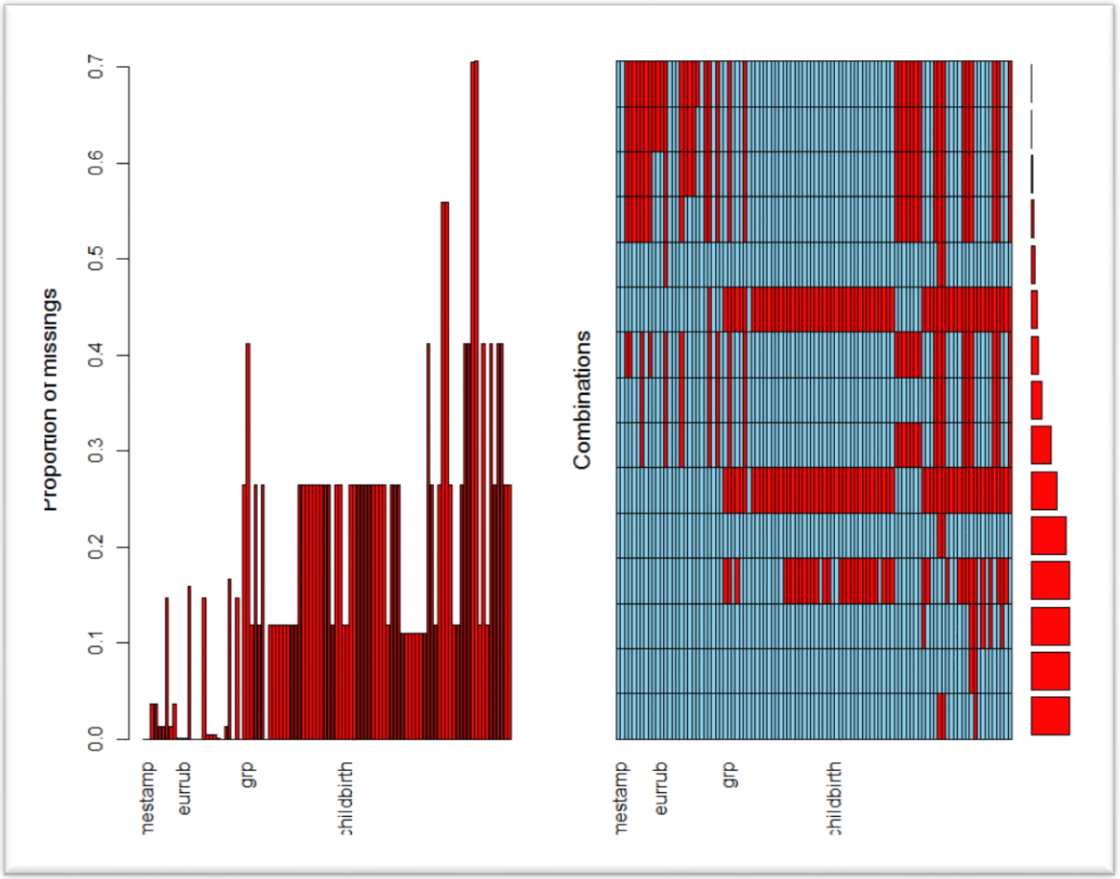


Figure 6 Proportion of missing values

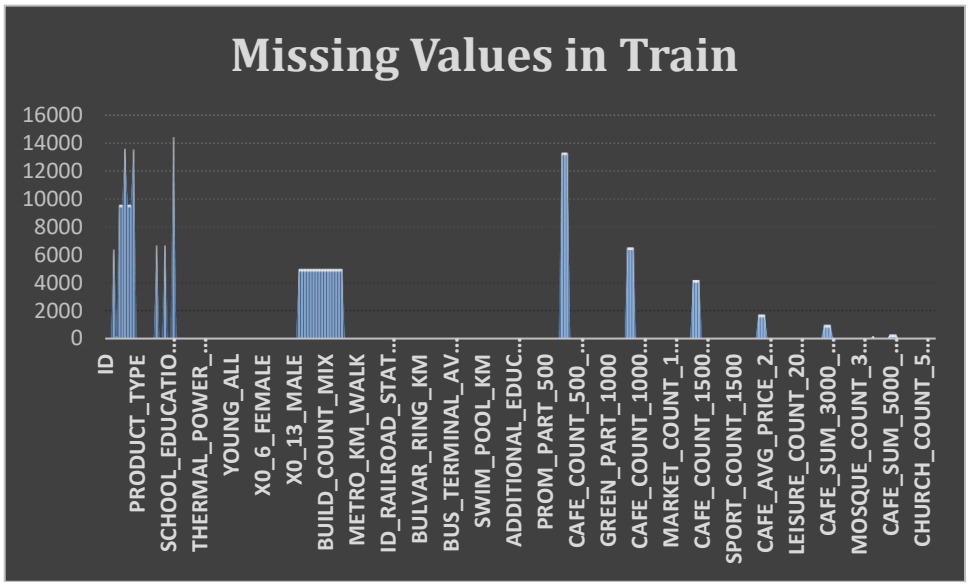


Figure 7 Missing values in Train data

Above is the inference for train variables. There are treated similar to macro.

We choose to impute rather than eliminate the variables, because we have sufficient and significant data which can used to deduce the missing values and thus increase the quality and accuracy of the model.

5.1 Dealing with categorical missing values:

Macro:
Replace missing values with mode of the variable
Train:
No missing values present.

5.2 Dealing with numerical missing values:

I have used below methods to handle missing values. Both the methods can handle numerical and categorical variables in their own fashion. But since we do not have much categorical missing values and to reduce the computational overhead, we segregate numerical values and impute them.

a. missForest

Using the missForest function present under RandomForest package, I could successfully eliminate all the missing values. It's a non-parametric imputation method which can be used for mixed data types. missForest builds a random forest model for every variable. Then it uses the model to predict missing values in the variable with the help of observed values.
It yields OOB (out of bag) imputation error estimate.

b. aregImpute

Using aregImpute under Hmisc package, I could eliminate the missing values. This method too can be used for numerical and categorical variables. It's a combination of three methods: bootstrapping, additive regression and predictive mean matching. In bootstrapping, different bootstrap resamples are used for each of multiple imputations. Then, a flexible additive model (e.g, non-parametric regression method) is fitted on samples taken with replacements from original data and missing values are predicted using non-missing values. Then, it uses predictive mean matching to impute missing values.

I have implemented both the methods in my project. Hmisc yielded the results quicker as my dataset is huge.

6. Creating New Features

Our feature set is humongous and it needs dimensional reduction. But we have one key step to do before that. We will concentrate on extracting new prominent features from the existing ones. Since we must analyze the impact of these variables on the target variable, we must keep dimensional reduction on hold.

Here are some illustrations of how new features can eventually help make our model better.

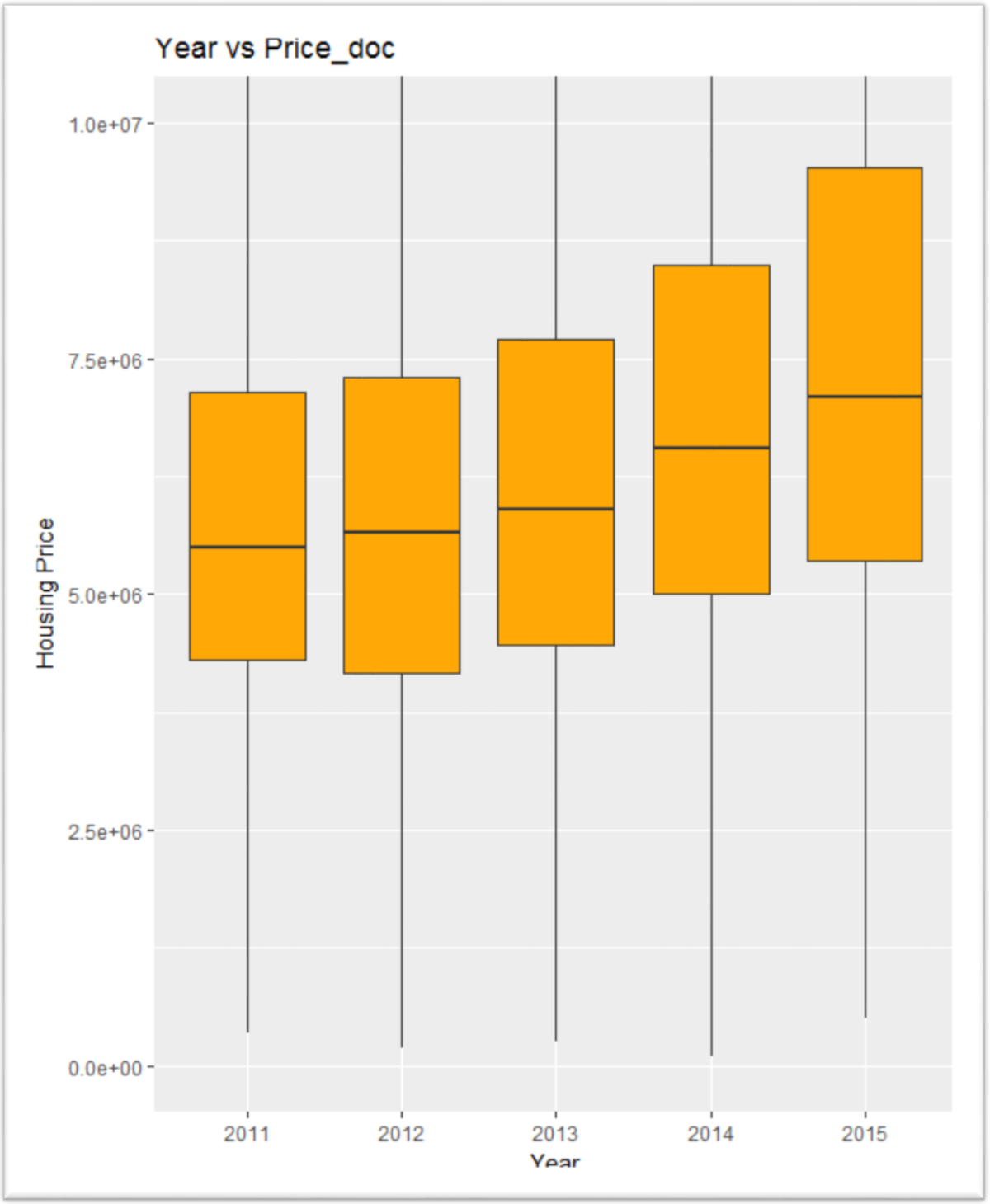


Figure 8 Boxplot- Housing Price increases with passing year

For example, the "timestamp" in macro is of no significance to the prediction. But we can extract the Day, Month and Year.

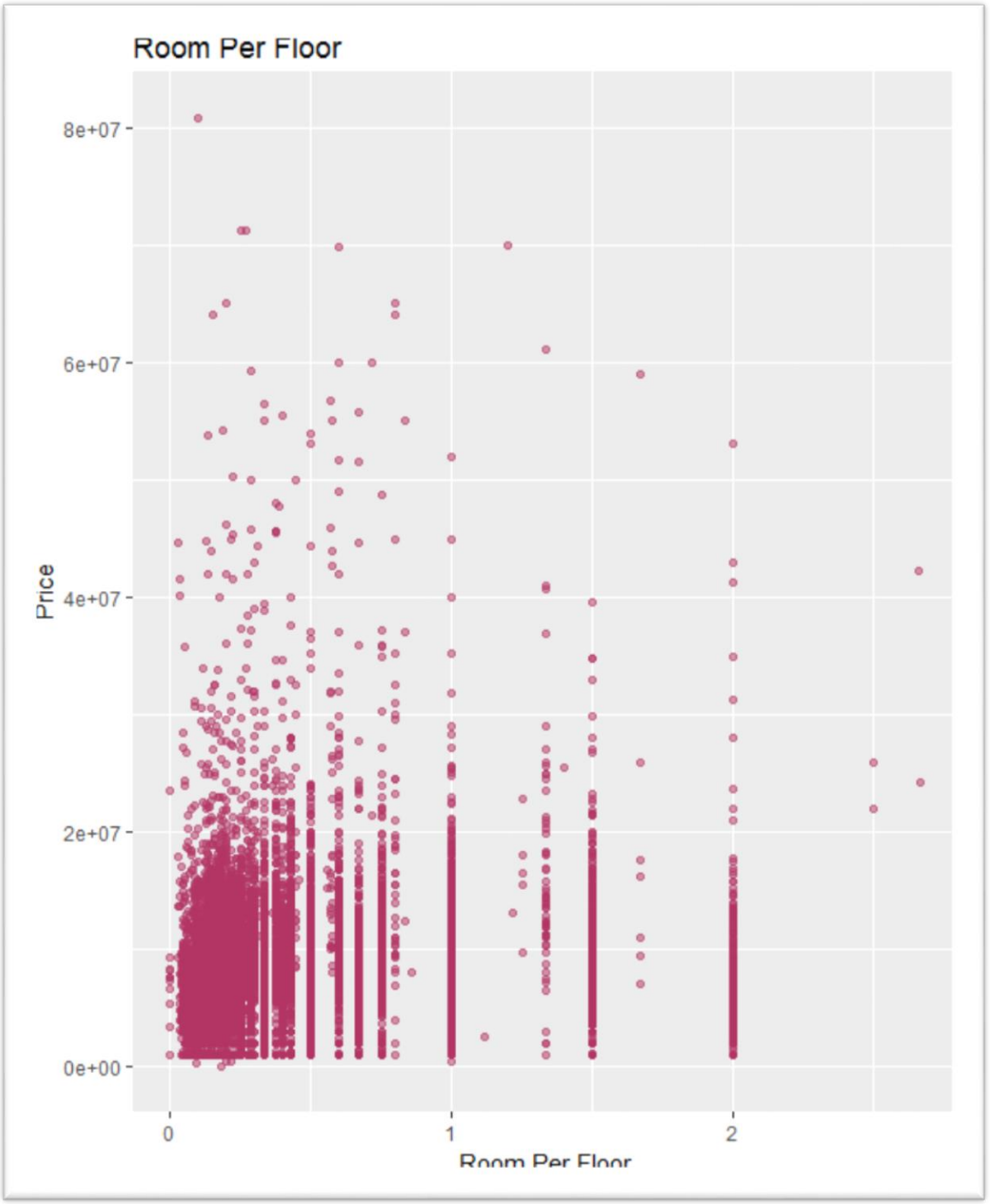


Figure 9 Room Area affecting price

See how each passing year, the housing price is increasing. Another example could be creating a variable “room_per_floor” by combining variables “num_room” by “floor”.

.

7. Feature Selection and Dimensional Reduction

Now we merge the economic and train data by “timestamp”. We can progress and select the most notable features and eliminate the ones that do not contribute much to our model.

7.1 Outlier Analysis

Outlier are the inconsistent or inaccurately values in our dataset. We can detect them when they vary disproportionally from the rest of the dataset. Such observations can be removed as they hamper the model functioning.

We have an illustration of data points in variable “salary”. It gives a rough idea of outliers present in it.

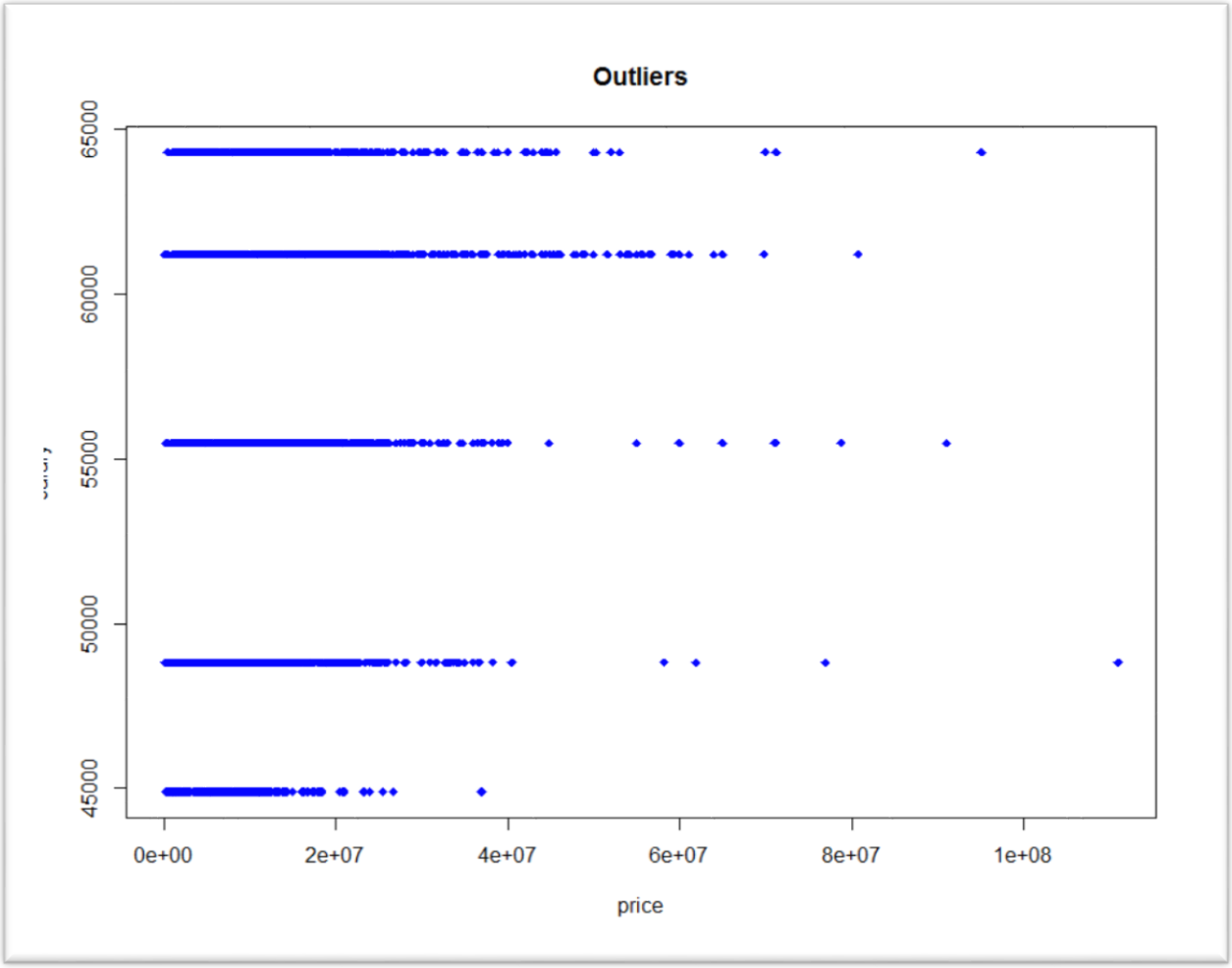


Figure 10 Outliers of “salary”

Boxplots help depict outliers. If a data point lies beyond the fences of the boxplot, say it considered as on outlier.

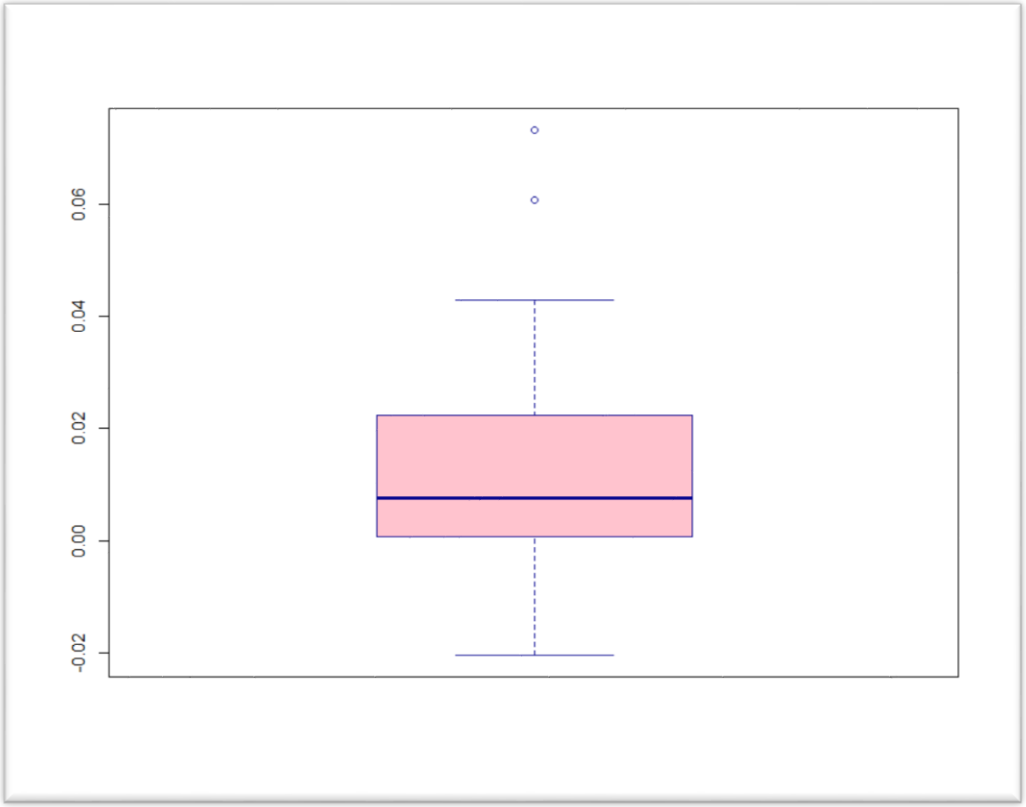


Figure 11 Boxplot of "salary" indicating Outlier

The points lying outside the fences of the plots can be removed. This will reduce the dimension of our data.

7.2 Check for Collinearity and Variance

There are high chances of multicollinearity in high dimensional data. Below is an illustration of correlation of a sample of the dataset.

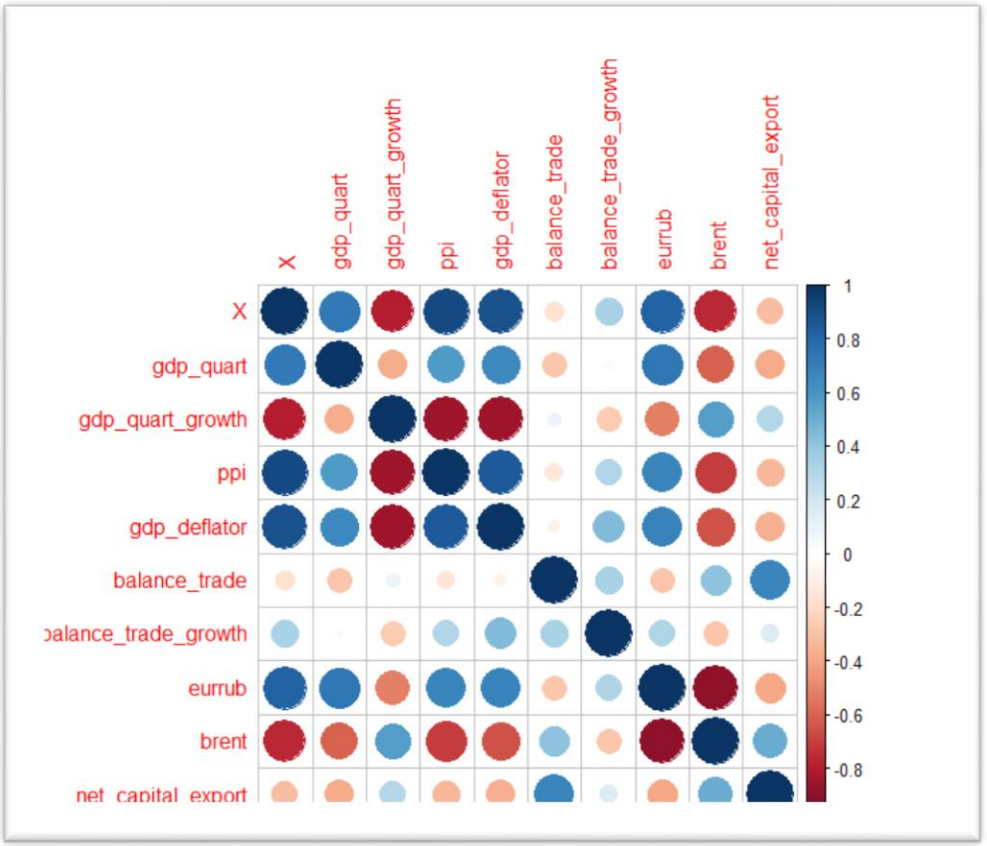


Figure 11 Correlation plot

Notice the trends of correlation between the variables. So we can sense that these variables, for example in this case, “ppi” and “balance trade” are highly correlated.

Correlated variables are redundant to the model and just an overhead. Thus they can be either combined or reduced to one distinct variable.

7.3 Select key features with VIF

Finally , we can use VIF function to detect important variables and remove the rest which do not contribute much to our model.

In this project using VIF, 156 variables were left and used to model!

8. Hypothesis Testing

8.1 T-TEST Result

Applied to numerical data

```
        One Sample t-test

data:  df_num
t = 173.76, df = 6520800, p
-value < 2.2e-16
alternative hypothesis: tru
e mean is not equal to 0
95 percent confidence inter
val:
 131052.3 134042.5
sample estimates:
mean of x
 132547.4
```

8.2 CHI-SQUARE TEST Result

```
Chi-squared test for given
probabilities

data:  cat
X-squared = 15706, df = 116
0, p-value < 2.2e-16
```

Since p-value is not <0.05 we reject the null hypothesis.

9. Predictive Regression Model

We have been provided with a test dataset called “test” of which observations are present. We should predict housing prices for all these observations which correspond to July 2015 - May2016.

Since my problem statement is a regression problem, I opt to build a Linear Regression Model first.

9.1 Linear Regression Model.

Model was built on training data and tested on test data.

The coefficient of determination R^2 measures how well the model fits data.

The MSE and RMSE were used to determine the fit of the model too.

The model is 95 % accurate.

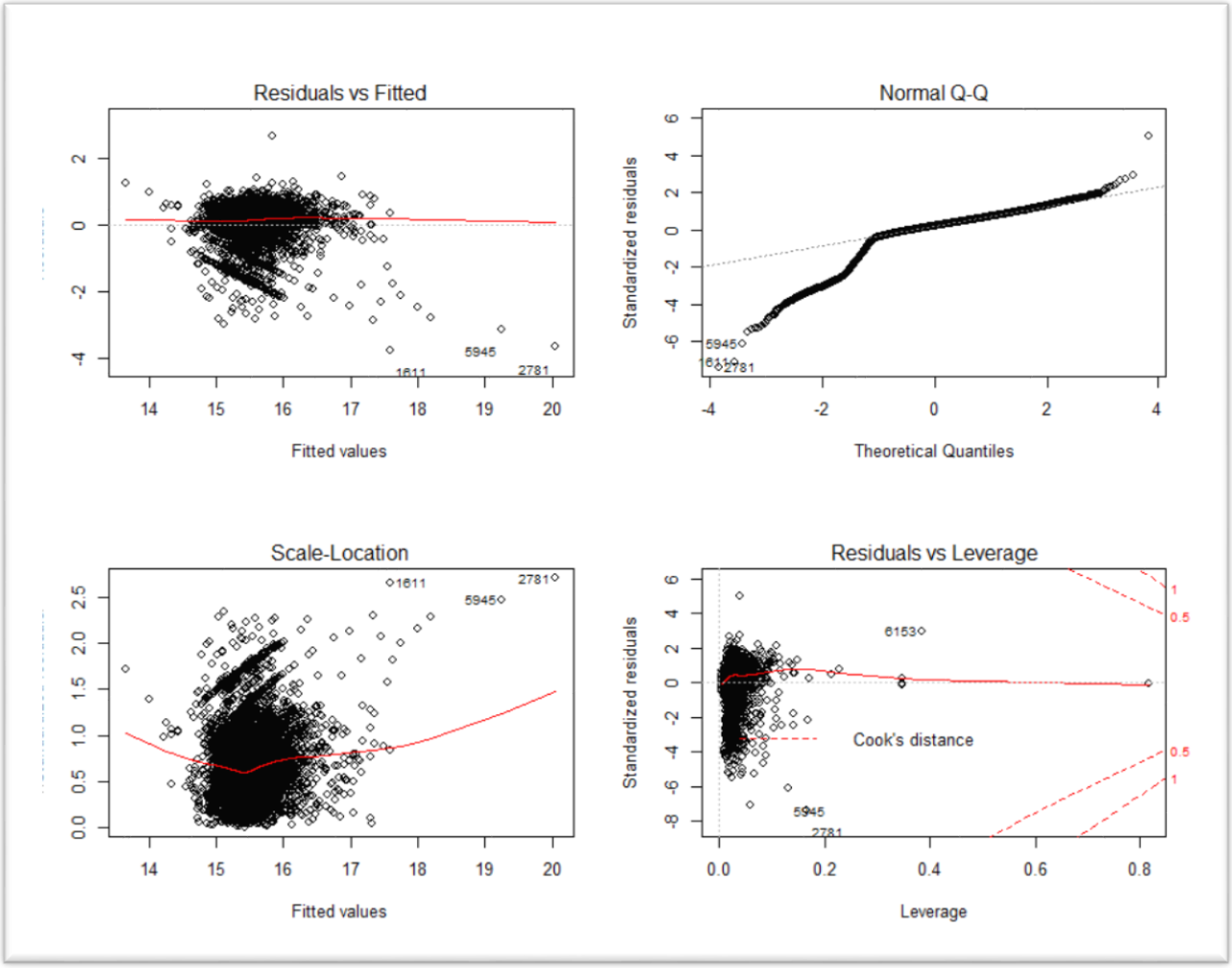


Figure 12 Linear Regression Diagnostics.

```
Residual standard error: 0.
481 on 30295 degrees of fre
edom
Multiple R-squared:  0.3707
,       Adjusted R-squared:
0.3671
F-statistic:   102 on 175 a
nd 30295 DF,  p-value: < 2.
2e-16
```

10. Conclusion:

I have chosen the result of better performance and accuracy yielding Linear Regression model and the housing prices for Sberbank data has been predicted.

_____THANK YOU_____