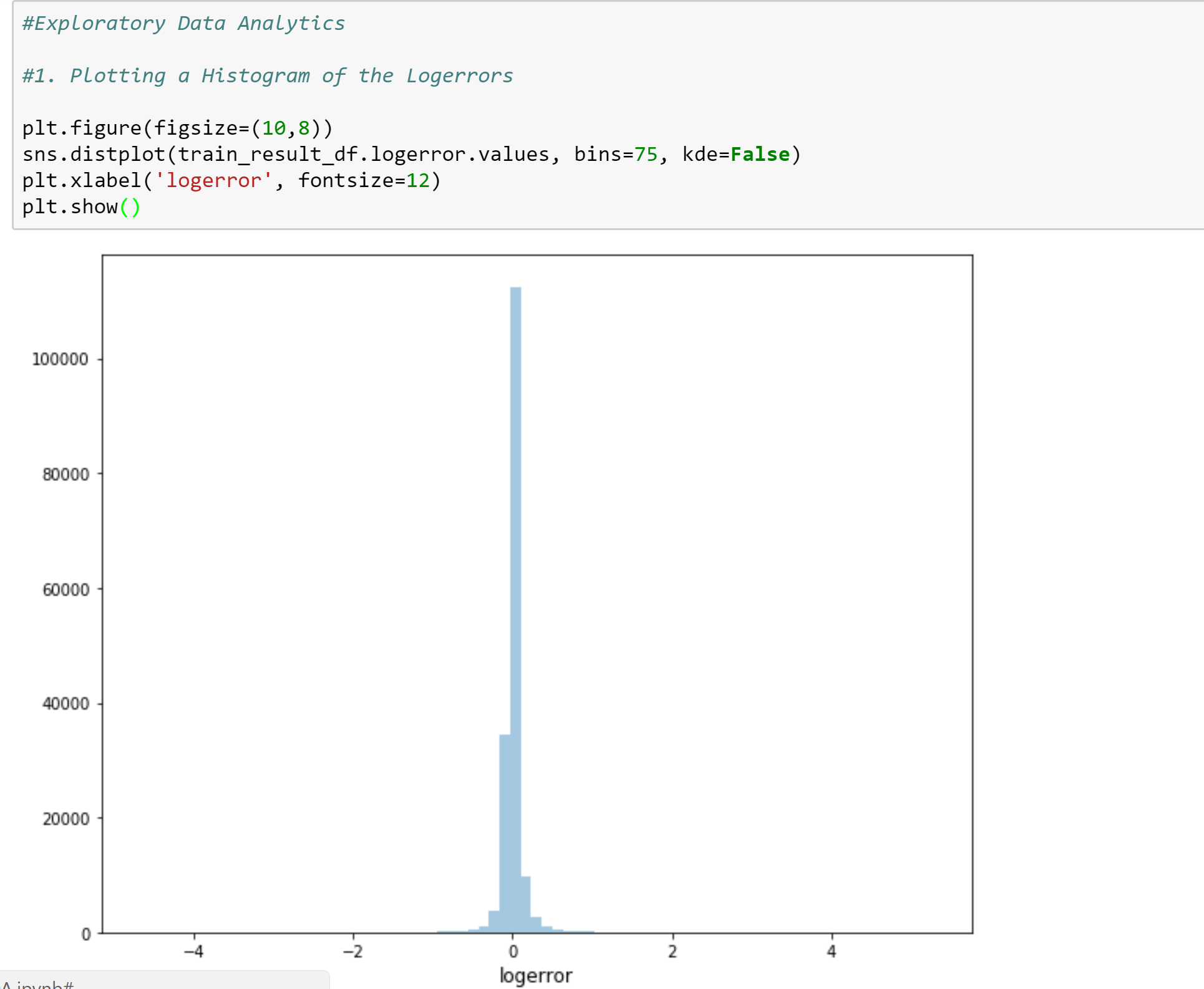
**Zillow Data Set**

**Data Ingestion, Exploratory Data Analysis, Data Wrangling**

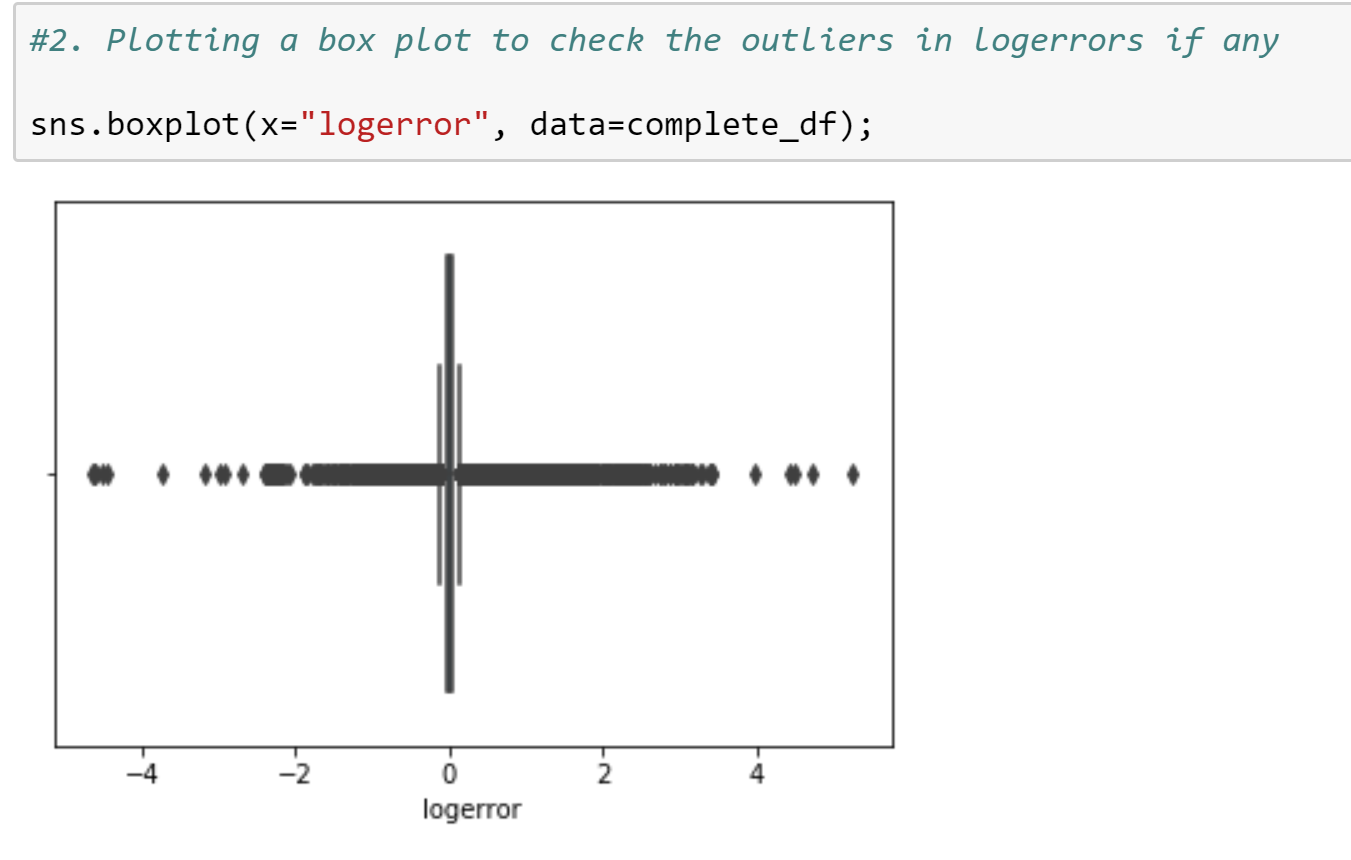
1. Plotting a histogram of the log errors

After merging the files, we are looking for the distribution of log errors in the data set. We see that it’s a normal distribution, but we need to check for anomaly as well.



1. Plotting a box plot of log error

On plotting the box plot, we find out about the anomaly in the log error which is lying outside the box where majority of data is present



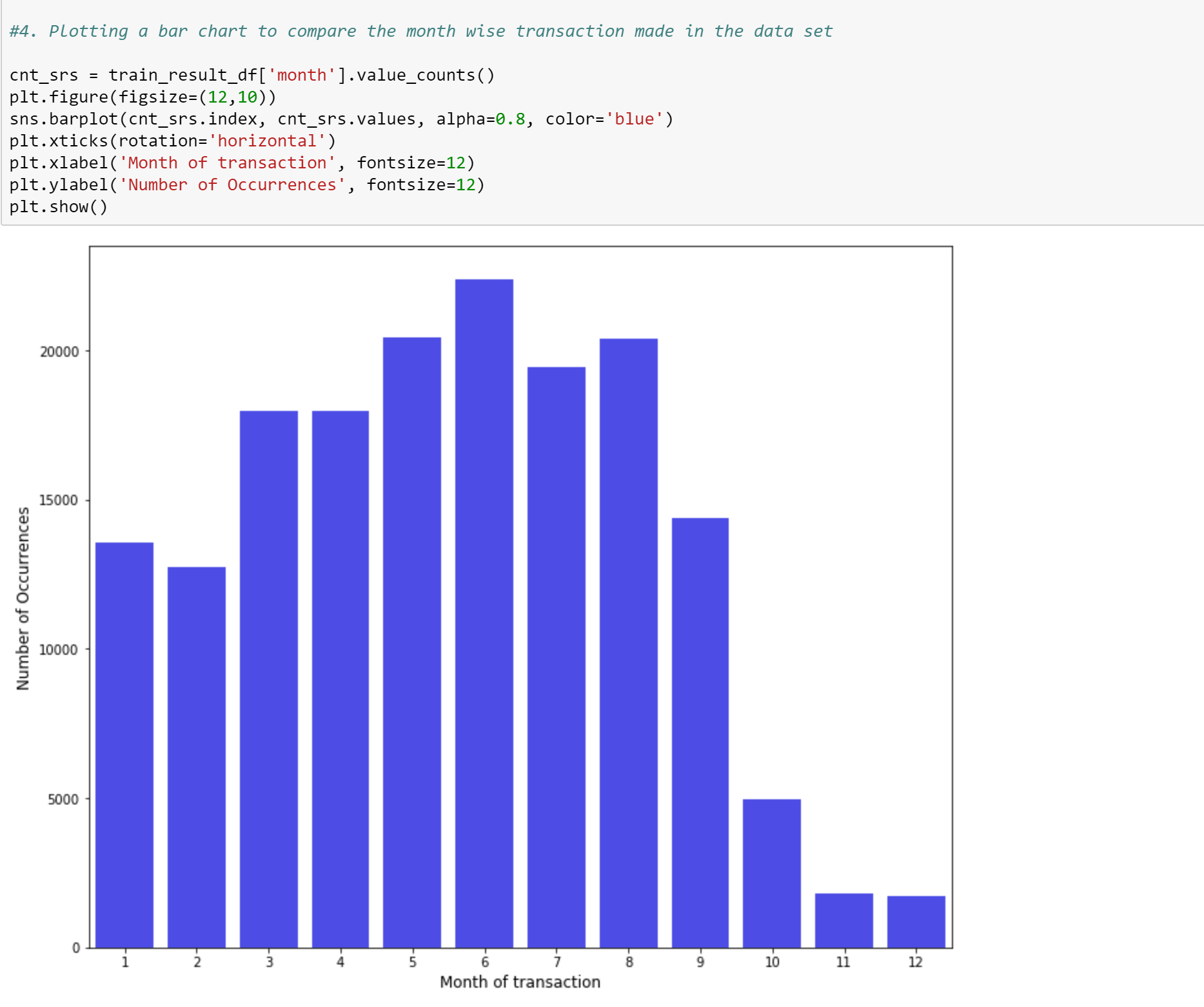
1. Plotting the average log error

We can see a trend in the log error based on the time frame of the transaction, we see a significant dip in the log error during the time April to June of 2016 and March to May of 2017



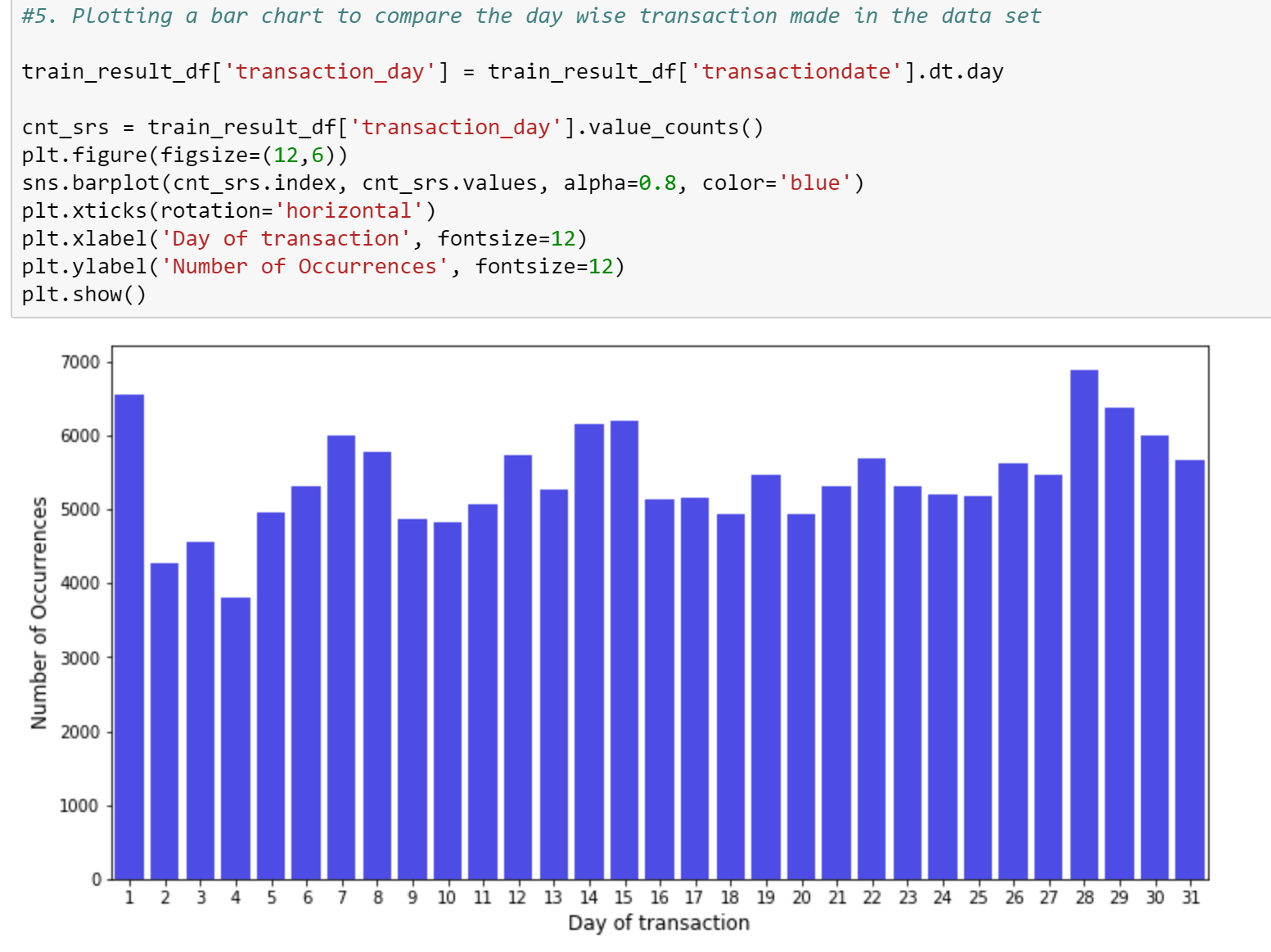
1. Plotting a bar chart for the month wise transaction

On comparing based on the monthly transaction, we infer that when the sales were high the log error was log. Thus, we can see that increase in the sales resulted in decrease of the log error



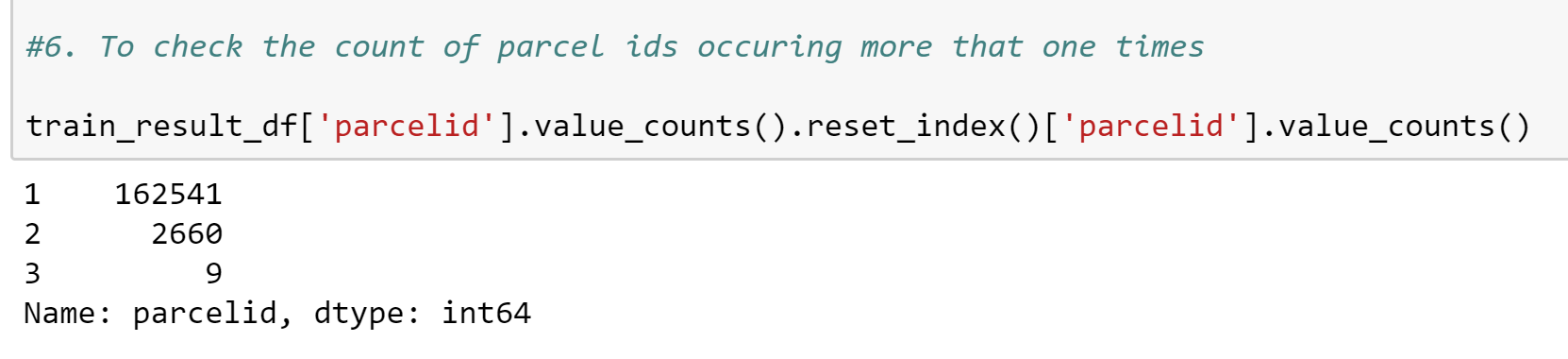
1. Plotting a bar chart for the month wise transaction

To analyze further sales, we can see the trend in terms of the daily transactions that are being made.



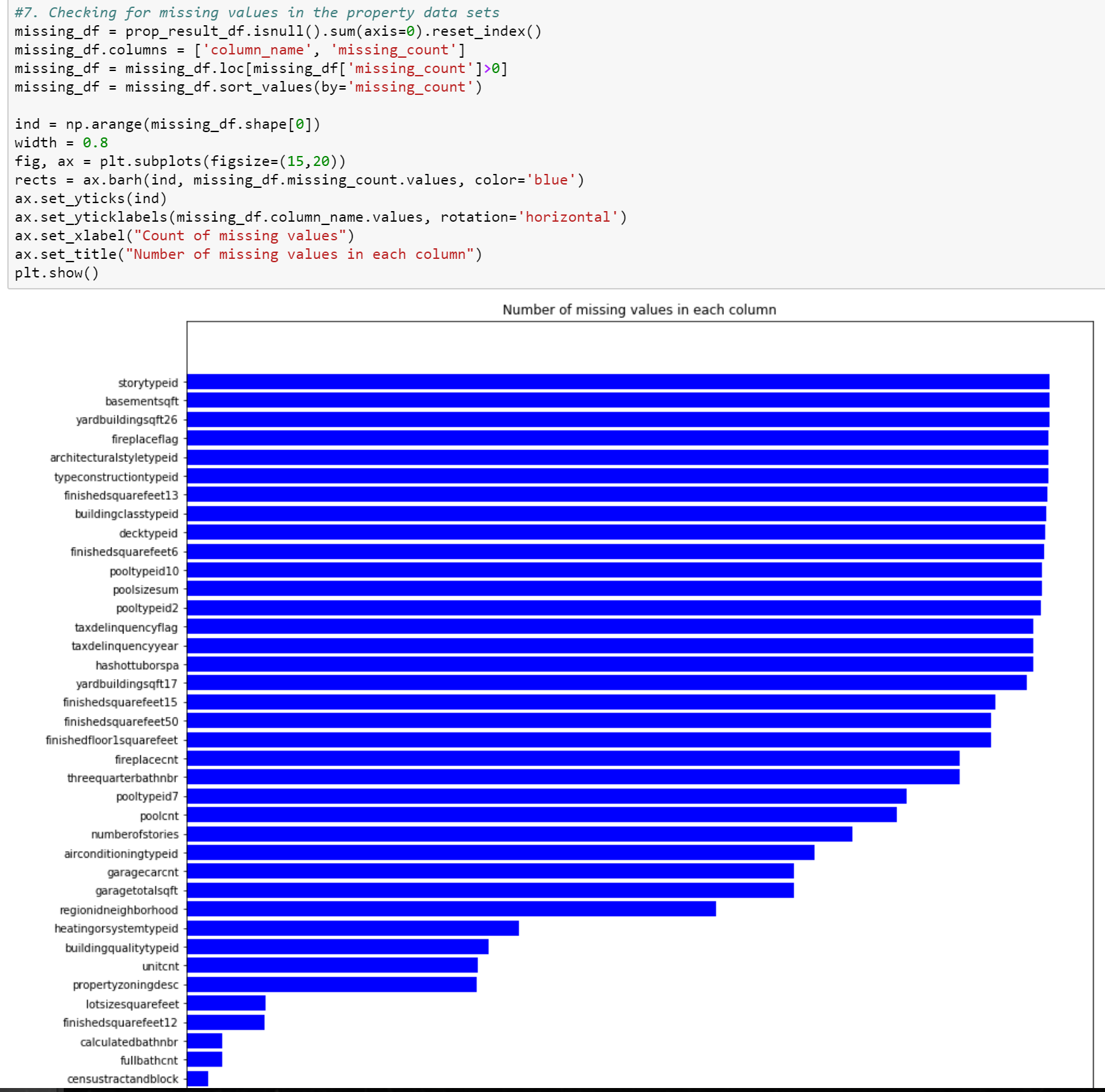
1. To check the count of parcel ids occurring more than one times

We find out that there are many parcel ids that occur more than one-time i.e the same plot is re-sold



1. Checking for missing values in the property data sets

On plotting a bar graph, we can find out about the data sets. We can see the total missing values in each of the column

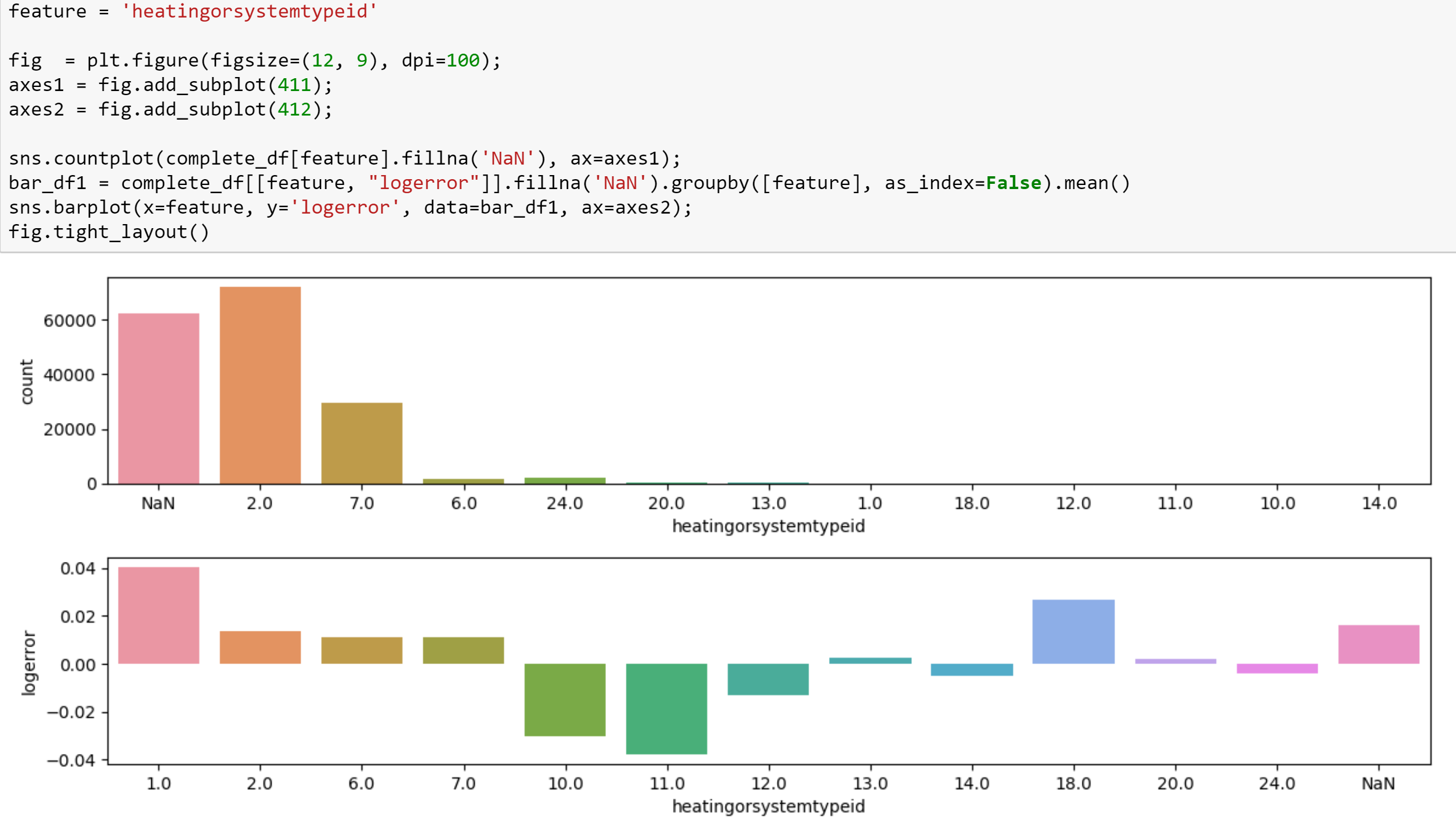


1. Inspect Categorical Data

We want to find out the contribution of the categorical data on the mean log error

Here, we considered airconditioningtypeid, heatingorsystemid and saw the various values of the feature against the log error mean

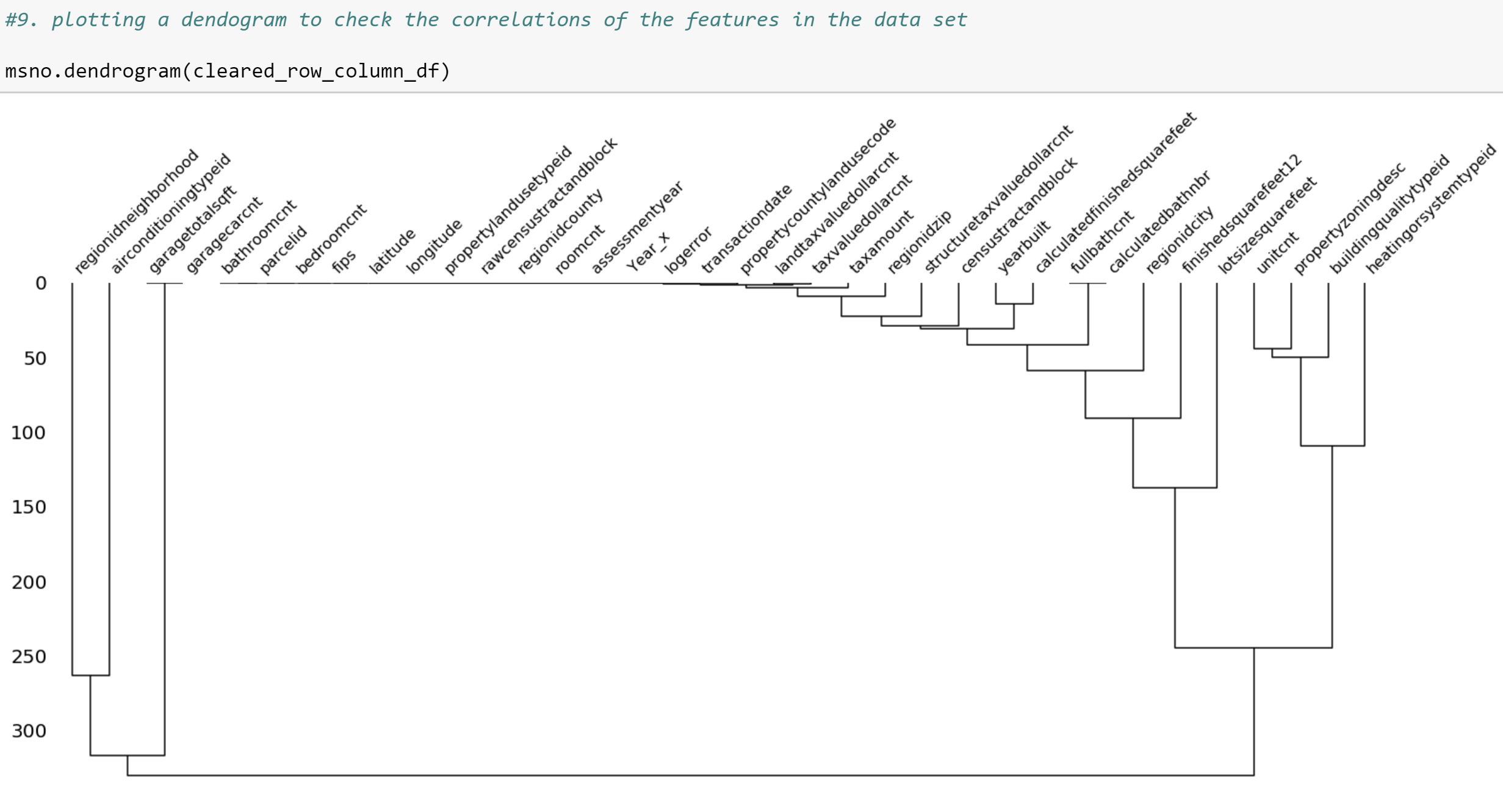




Keeping a threshold and eliminating ones below that – We decided a 70% threshold value. Any columns who has more than 70% of data as null, we simply drop those columns as they won’t be helpful in the model prediction of the zestimate.

1. Plotting a dendogram to check the correlations of the features in the data set

With the help of the dendo-gram from missingno library, we find the features which are correlated to each other



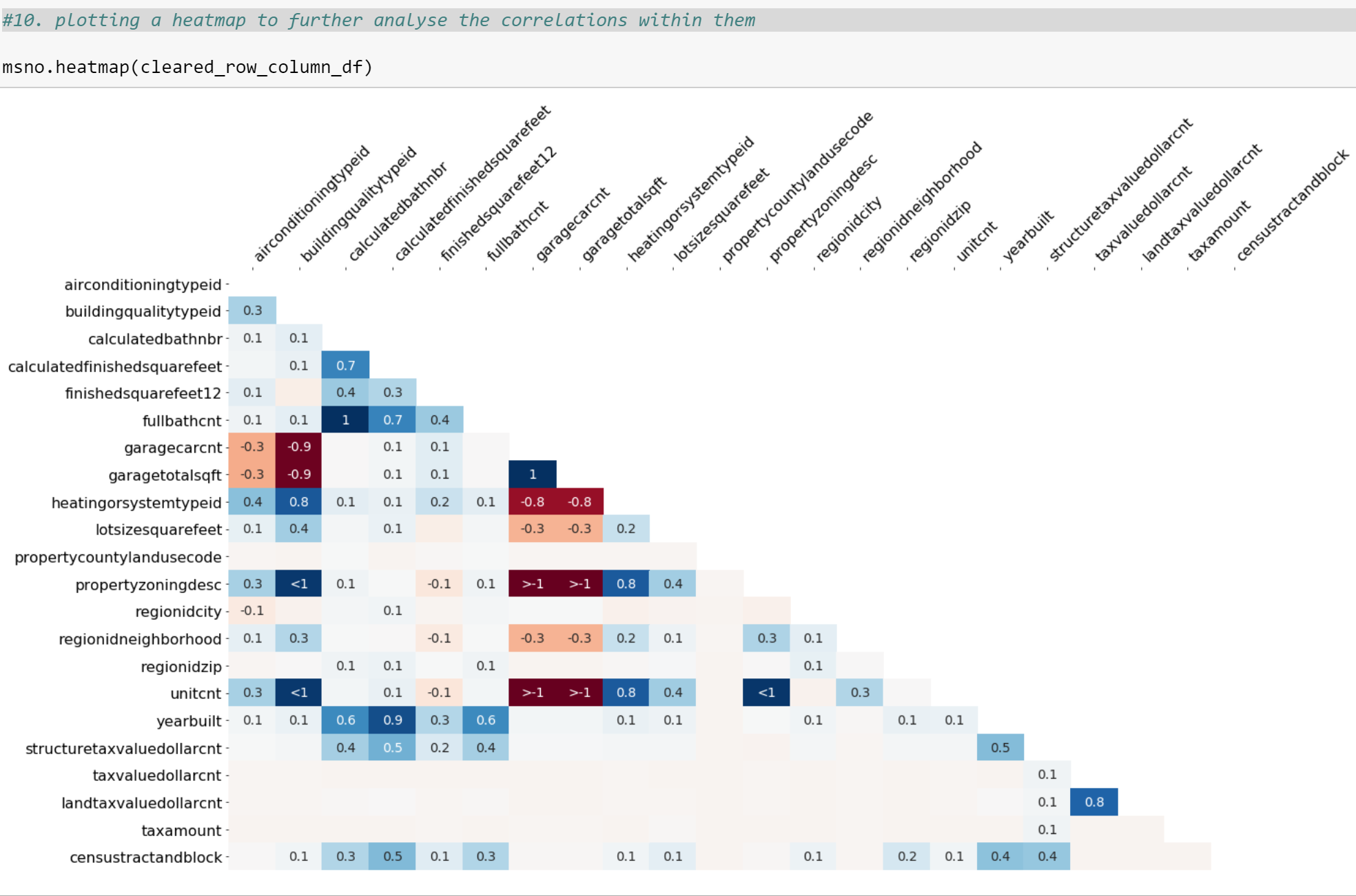
1. Plotting a heatmap to further analyze the correlations within them

Using the heatmap, we further drill down to check the level of correlation between the features and can decide on the missing data analytics. These correlations will help us to predict the missing data

Nullity correlation ranges from -1 (if one variable appears the other definitely does not) to 0 (variables appearing or not appearing have no effect on one another) to 1 (if one variable appears the other definitely also does).

Variables that are always full or always empty have no meaningful correlation, and so are silently removed from the visualization—in this case for instance the datetime and injury number columns, which are completely filled, are not included.

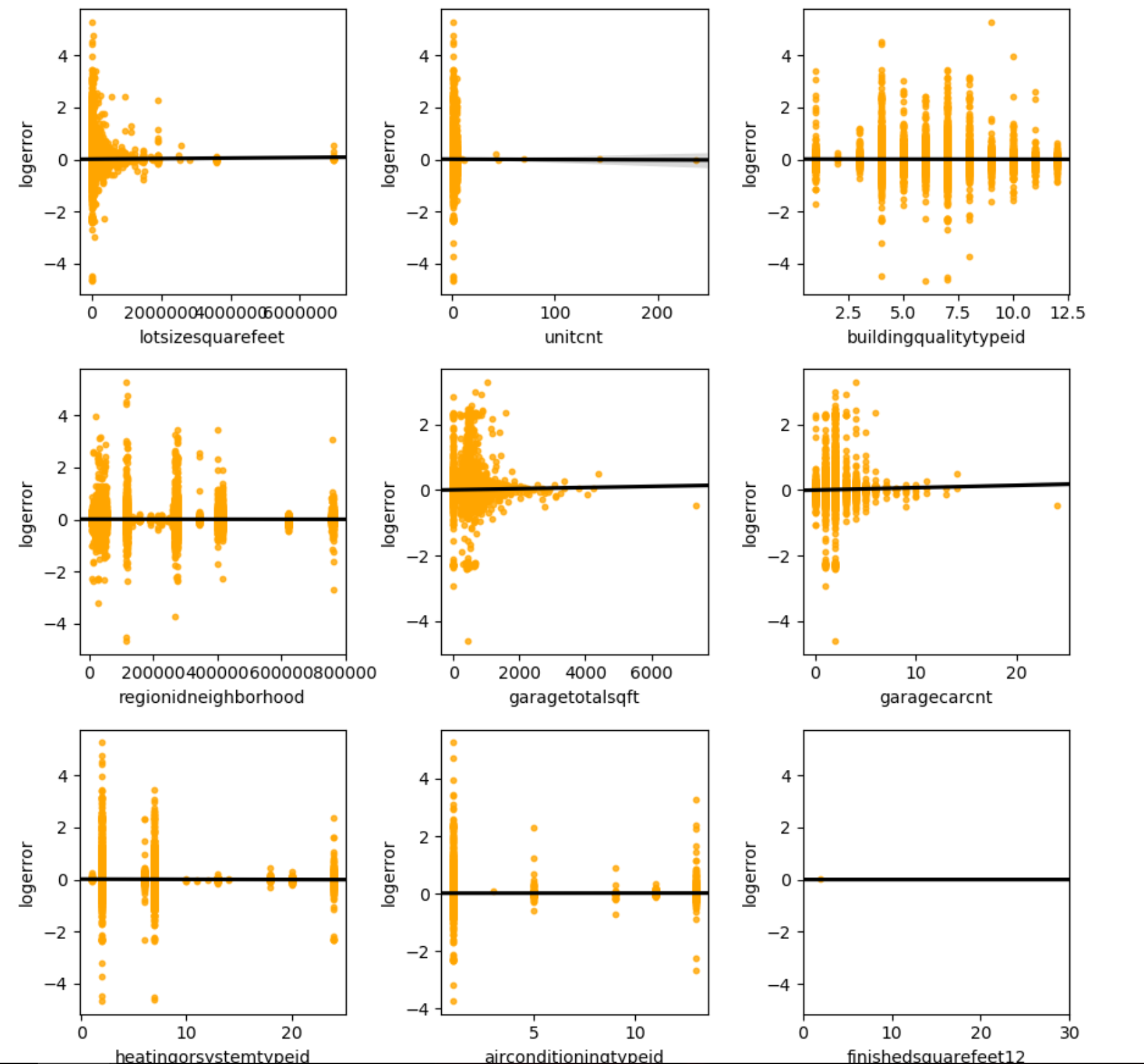
Entries marked -1 are have a correlation that is close to being exactingly negative or positive, but is still not quite perfectly so. This points to a small number of records in the dataset which are erroneous. For example, in this dataset the correlation between VEHICLE CODE TYPE 3 and CONTRIBUTING FACTOR VEHICLE 3 is <1, indicating that, contrary to our expectation, there are a few records which have one or the other, but not both.



1. House Characteristics

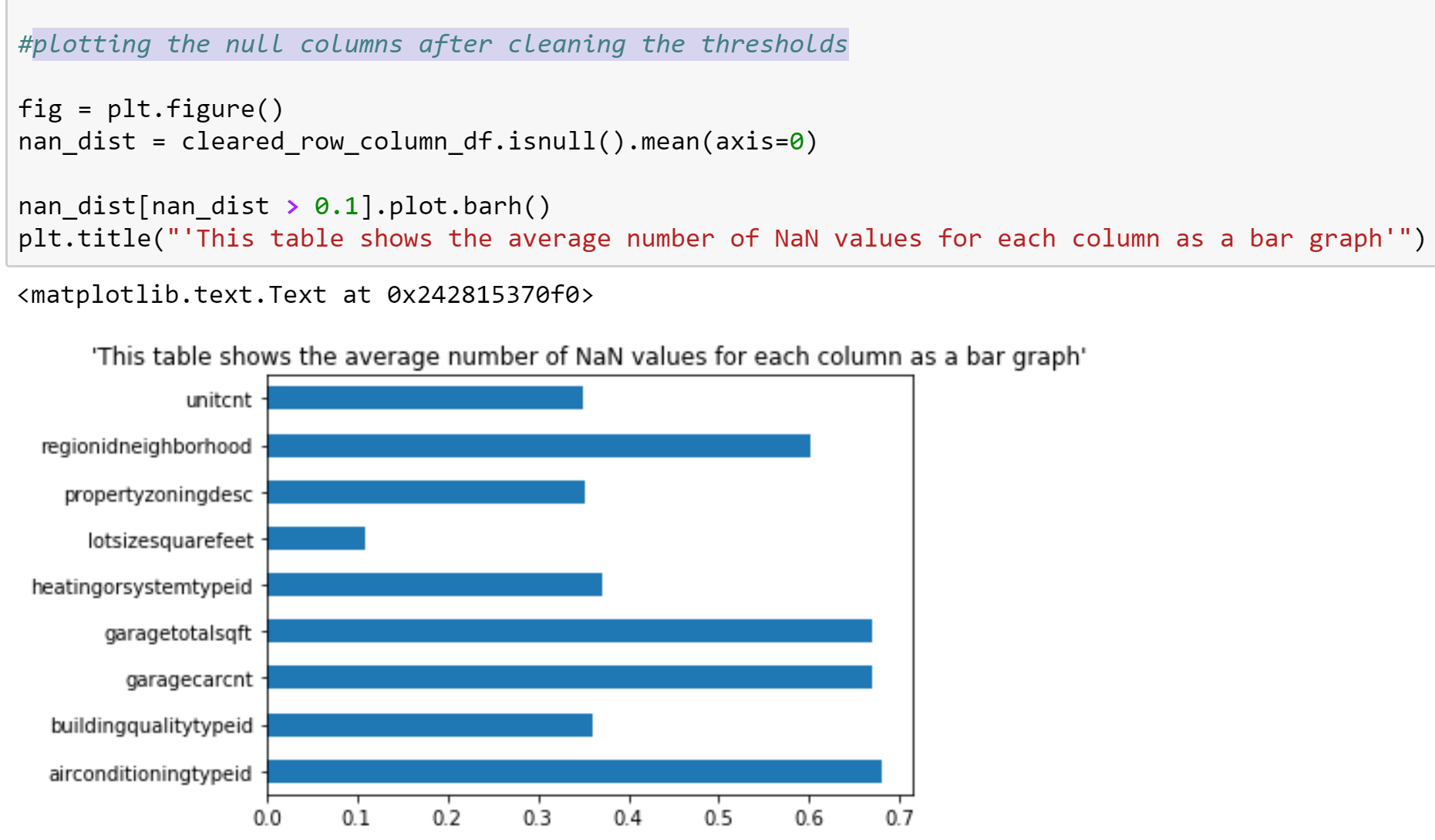
Here we plot multiple scatter plots with various house characteristics and log error to see the variance





1. Plotting the columns with null values after cleaning the thresholds

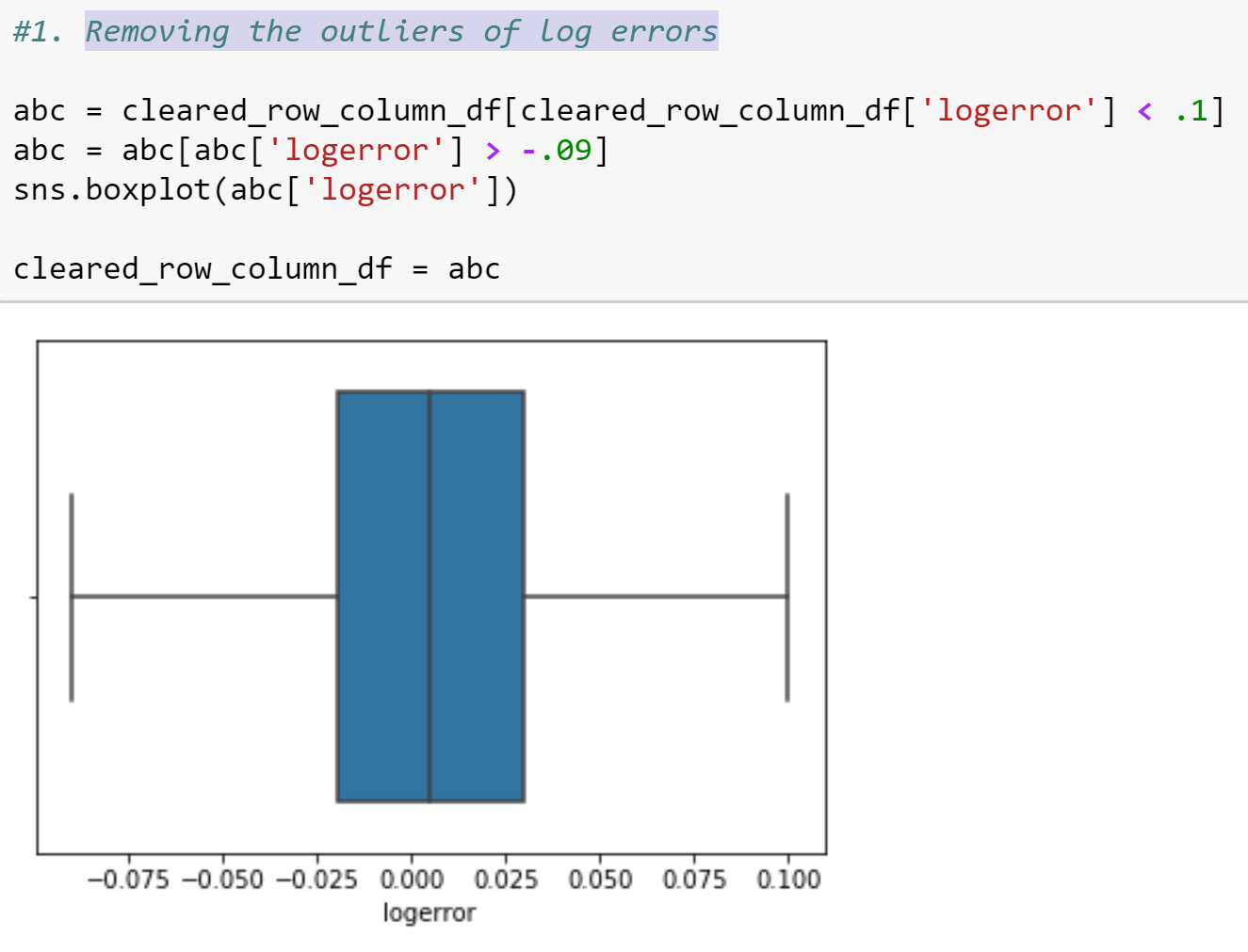
This will be the data that we need to clean and fill in order to feed it to the prediction model



**Data Cleansing**

1. Removing the outliers of log errors

To predict the log error better, we remove the outliers that we found out in the data analysis.



1. Filling RegionIdNeighbourHood

As we found out in the correlation analysis, the missing values here are directly or indirectly correlated to each other and thus using those we can predict the missing data.

Here, the regionidneighborhood is highly correlated to the latitude and longitude and thus using them we can predict the missing values.

We used a NearestCentroid algorithm to predict the regionidneighborhood based on the particular latitude and longitude

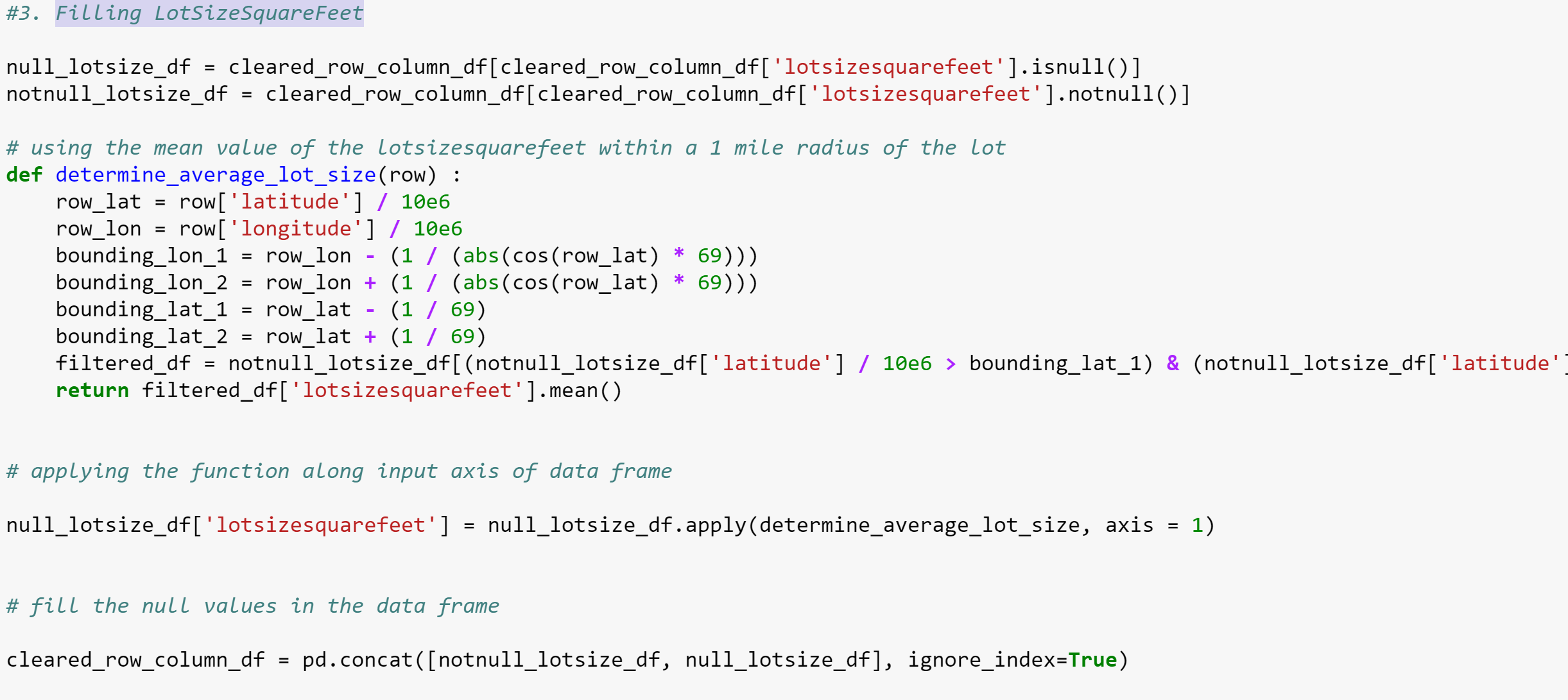
After making a few changed in the model, we predicted the values and filled the missing values with the predicted values.



1. Filling LotSizeSquareFeet

Similarly, we found out that lotsizesquarefeet is correlated to latitude and longitude of the area.

Here, we are checking with in the 1 mile radius and predicting the missing values with the mean of the available data in the radius.



1. Filling BuildingQualityTypeId

Here, we are using the median value of BuildingQualityTypeId to fill the missing data

1. Filling HeatingOrSystemTypeId

To predict this, we used the correlated features and used K-Nearest Neighbors Classifier.

We trained this model with the train data set and predicted the values with the prediction method to fill the missing values.



1. Dropping PropertyZoningDesc

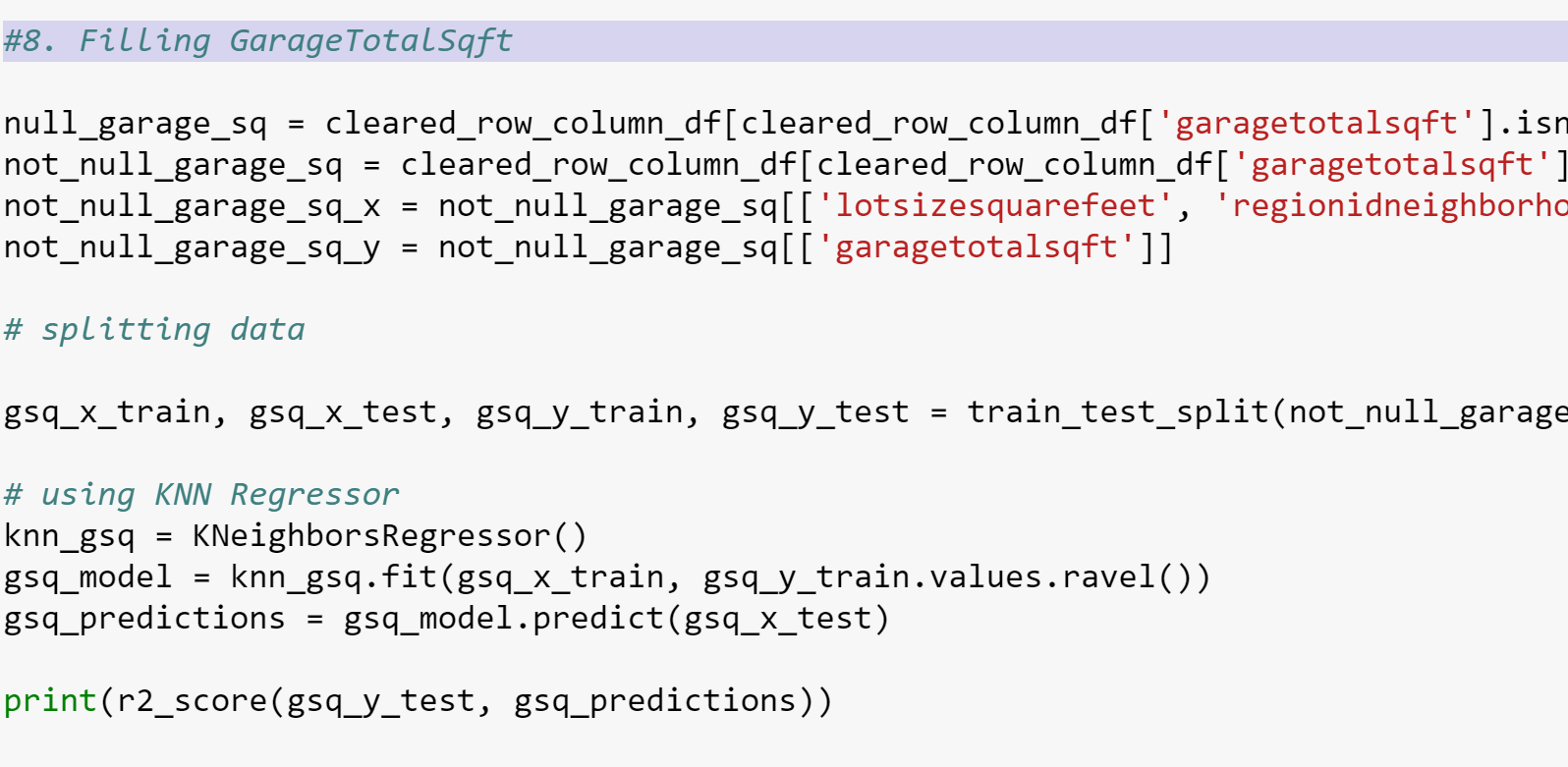
We simply dropped this feature as this was not going to be used in the prediction algorithms. It was of type String and that cannot be used in the models

1. Filling UnitCnt

For Unit count we used the KNN classifier as well to predict the missing values and filled it in the data frame

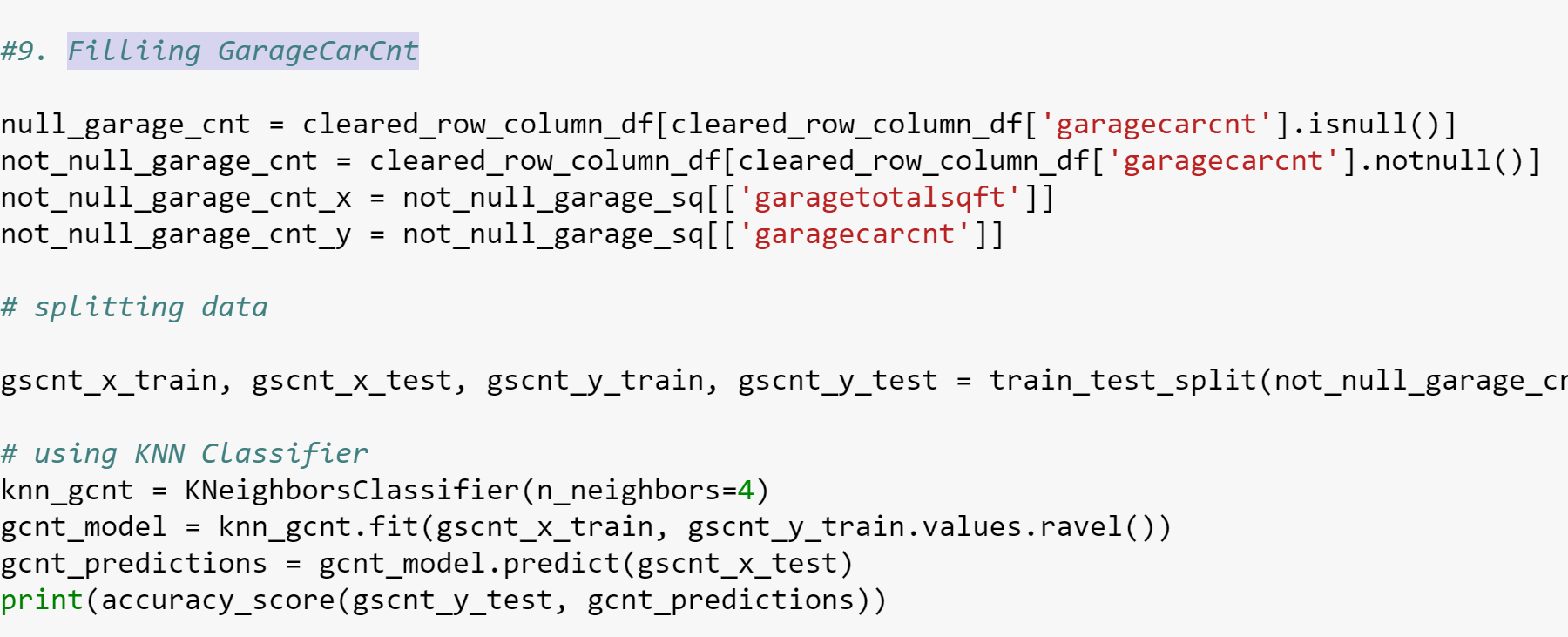
1. Filling GarageTotalSqft

We are using a K Nearest Neighbors Regressor as the value to be predicted is a continuous value



1. Filliing GarageCarCnt

We used a KNN classifier



1. Filling AirconditioningTypeId

We are using another KNN Classifier to predict the missing values here.



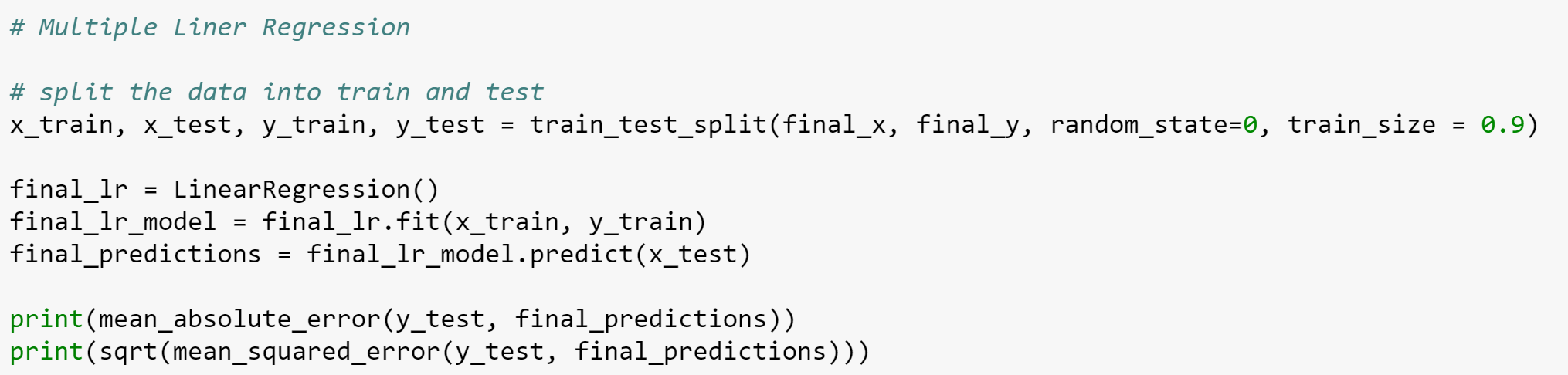
**Prediction model**

1. Multiple Linear Regression

Interpretability

Computational Overhead

Accuracy

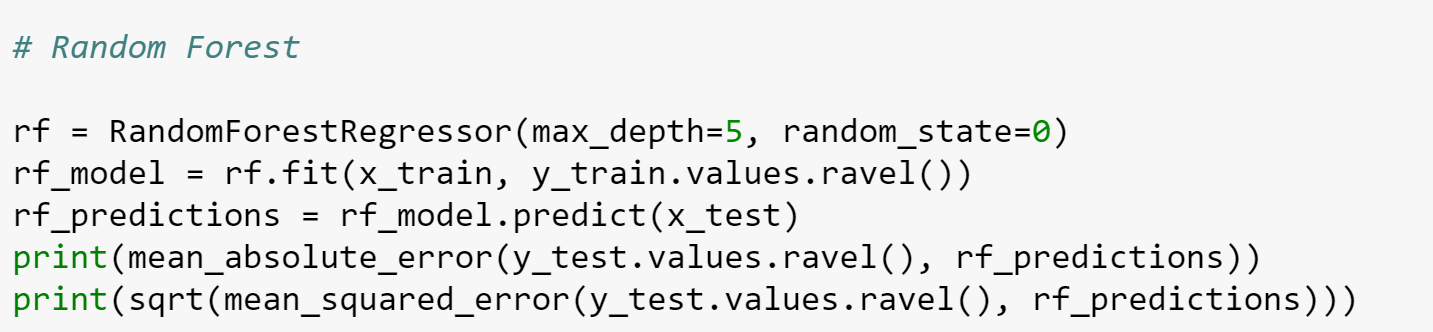


1. Random Forest

Interpretability

Computational Overhead

Accuracy

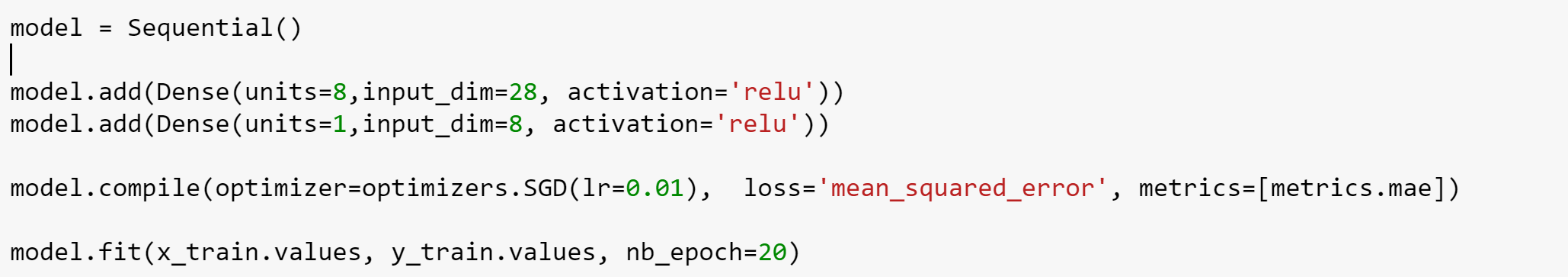


1. Neural Networks

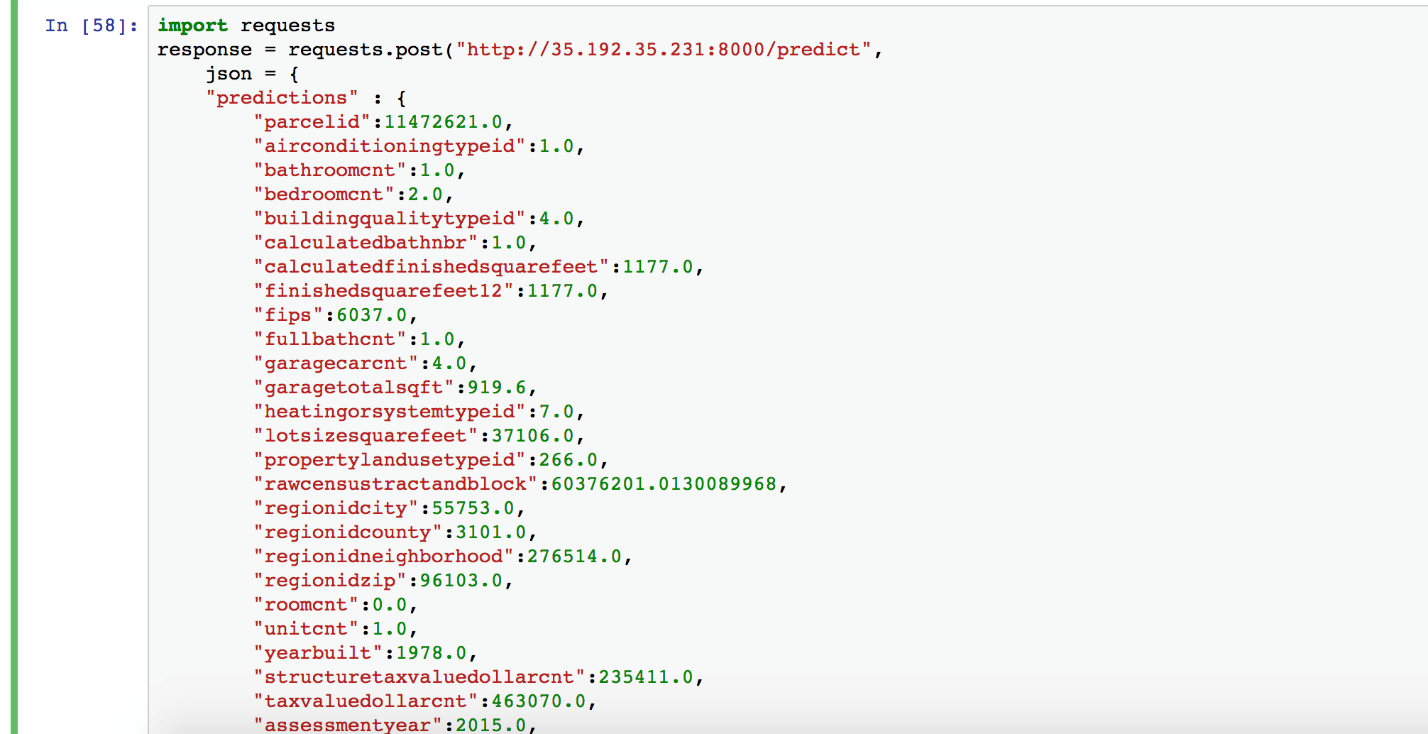
Interpretability

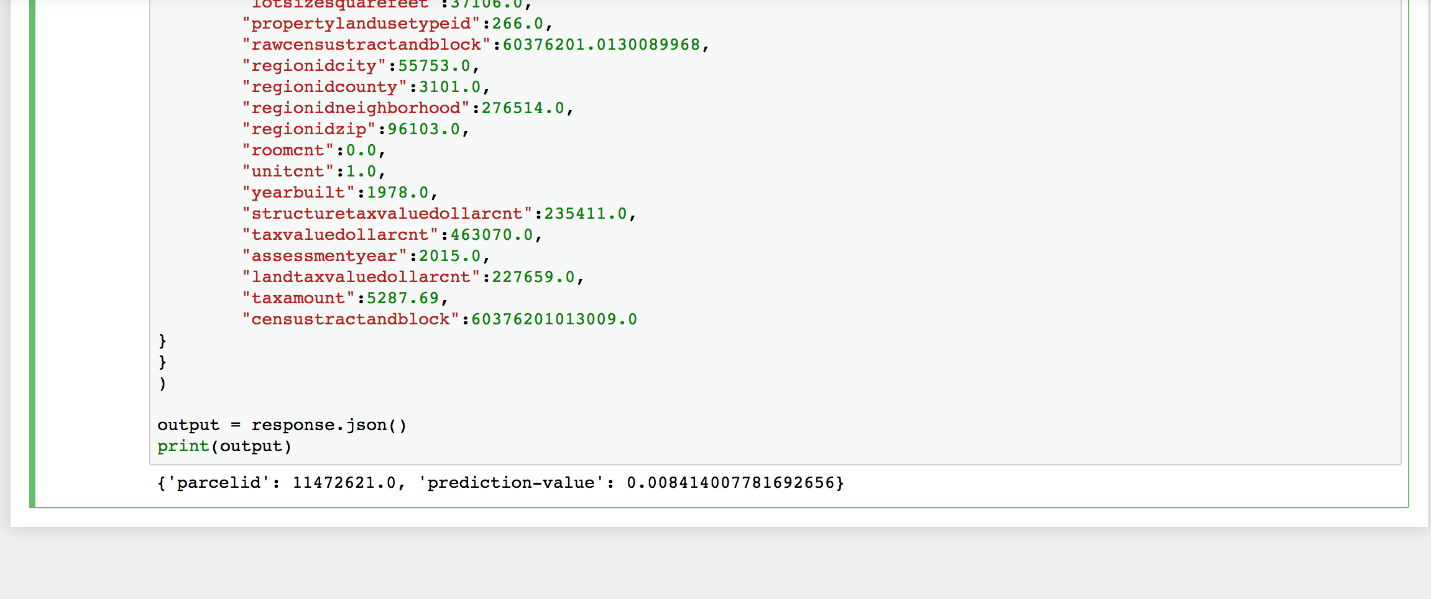
Computational Overhead

Accuracy



**Model Deployment**





**Enhancing the Rest API Search**

