**PROJECT REPORT**

**ON**

**UNITED STATES HOUSE RENT PREDICTION SYSTEM**



Ashish Roy - 001044804

**Abstract**

This report consists of the final findings of the Supervised Machine Learning algorithms used for the prediction of the United States House Rent.

The dataset used for the project has been taken from Kaggle and from iNeuron. The data is then cleaned as per the requirements and then different models were built in Jupyter Notebook (Anaconda 3) to examine the models performance on definite parameters.

After a detail and precise study of the many available algorithms and data, it became obvious that the problem fall under no Regression. The study focuses on various algorithms by using classifiers like- Logistic Regression Classification, K-Nearest Neighbor Classification, Naïve Bayes Classification, and Random Forest Classification.

The major finding is that the machine learning (ML) approach should be suitable for this problem due to many aspects, like a total of 22 features in the dataset.

**Contents**

* Introduction
  + Background
  + Motivation
  + Goal
* Methodologies and Algorithms
  + Data Review
  + Software and Libraries used
  + Data Cleaning
  + Feature Selection
  + Exploratory Data Analysis
  + Methods used
    - Linear Regression
    - K-Nearest Neighbors
    - Decision Tree
    - Random Forest
  + Feature Selection
* Description of Dataset
* Results and Analysis
* Conclusion
* Reference

**Introduction**

**Background**

United States House Rent Prediction aims to predict the house rent the main aim of the project is to build a Machine Learning model to predict the rental prices of a house throughout the United States based on different variables describing the features of the houses. This dataset has multiple features that describes the complete nature of the house and its dynamics, and a target feature ‘Price’ that is to be predicted.

**Motivation**

The importance of house rent as an asset class cannot be overstated. As the number of real estates is increasing day by day, a rent prediction analysis only helps investors assess the earning potential of a specific house in a specific area with specific features and thereby improving the efficiency of the real estate investment in market. This prediction system will also enable the investors to make wise investment decisions so as to maximize on their returns. The motive of this project is to help both landlords and tenants accurately price their rental accommodations. We are striving to make the rental market more transparent and easier to navigate.

**Goal**

We wish to create a model that would tell users what the fair rent of a certain listing (house) would be, in a certain city, with amenities, at any given time. We aim to minimize the difference between the actual rent and the rent estimated by using different types of models. We will evaluate the performance of our model by using different types of machine learning algorithms.

**Methodology & Algorithm**

**Data Review**

The house price dataset is 380 MB in size, it contains 22 columns. After a preliminary study of the available data, we think that this a Regression analysis problem. The potential algorithms we can use for our data are listed below and explained why this model can work efficiently.

**Software and Libraries used**

The dataset is downloaded from iNeuron ML challenge.

Software used: Jupyter Notebooks

That are:-

* Numpy
* Pandas
* Matplotlib
* Seaborn
* Plotly
* Geopandas
* Shapely
* Sklearn

**Data Cleaning:**

The dataset was miscellaneous with many missing and NAN values as the data was very huge. Therefore, many rows had NAN values which can compromise the models, so to sort, all the rows with NAN and missing values the values are removed.The data originally had 20+ columns this can lead to too much noise or distortion in the final dataset many of those columns have empty values(NaN) and missing values too, so we have managed to clean the dataset and buit our model on 22 columns.

**Data Visualization:**

After cleaning the dataset and handling the outliers we have done Data Visualization using different graphs/visuals. Firstly we have shown data visualization and removed the price>4000, then we have done plotting again after encoding the variables of missing values then we handled all the missing values, then followed by the visualization of outliers and removing all the outliers in dataset, Co-relation heatmap is created which shows the highest co-relation of certain columns.

#### **Models Used:**

The dataset is trained on various Regression models to predict the value of the target variables.

Different types of models used in the project are as follows-

* Linear Regression Regressor
* K-Nearest Neighbor Classification
* Decision Tree Regressor
* Random Forest Regressor

**Linear Regression Regressor**

Linear regression is a commonly used algorithm where a single input variable is used to predict the output variable. This also works with multivariable input as well. The relationship between the target variable and input variable is defined as,

y = w0X0 + w1X1 + w2X2 + … + wnXn + b

In Machine learning terms, y is the target variable, Xn are our features, wn are the weights and b is the bias of our model.

In our dataset, as we have features that have direct relation with the target variable, for example the location of a house can increase the rent price, parking space can affect the rent of a house, etc. We believe that this model has the potential to work better with our data and should be considered as a potential model.

This classifier is used to transform its output using the logistic function to return a accurate calculation value. Logistic Regression was not much success with the accuracy of the prediction of the (price) target value of our dataset.

**K-Nearest Neighbor Classification**

We know that KNN can be used for both classification as well as regression problems. It works by predicting values based on similarity of the feature, it assigns a value on the basis of resemblance in the training set. In can potentially work with our data by comparing the similarity of a house in a similar region with similar properties or features. For example, a new data similar to a house in training data that is in the same locality, allows dogs, has laundry in unit and has street parking

This classifier predicts the probability upon checking the K number of nearest neighbors. Applying this algorithm gave a Minimum accuracy score on our dataset.

**Decision Tree Regressor**

Decision tree breaks down data into smaller and smaller subsets until the lowest leaf node is reached. This is done by splitting the tree with decision nodes and leaf nodes for every new point that the problem is connected to.

This model is the potential model because we have features like dogs\_allowed, parking\_options, laundary\_options, etc. that could be the deciding factor of the house’s rent. For each decision in our data set, a new branch would be created leading to accurate prediction.

Decision tree is highly useful in classification problems where the total number of features is very high. As well as where the total number of rows are too high. A decision tree is represented

as upside down where its root is at the top of the tree then it splits into branches and when it cannot further split then the end branch is called as decision. Growing a decision tree requires to choose features and conditions to select optimal tree which has maximum prediction. The tree is grown arbitrary.

Applying this algorithm gave us the second highest accuracy.

**Random Forest Regressor**

Random forest gave us the highest accuracy if compared with decision tree, in terms of getting result both have given satisfactory results. With accuracy score of 87%.

Random Forest is a collection of decision trees. As explained in the Decision Tree Regression, it is based on the decision from all the inputs. In Random Forest, it randomly picks K data points from the data set to build a decision tree associated with these data points. We have to choose the number of decision trees we want, and the above steps are repeated. This helps to minimize the

risk of over-fitting the data points. Thus Random Forest is most efficient algorithm which gave us the highest accuracy results.

#### **Feature Selection**

We selected features that have the maximum contribution to our prediction variable by this way we can reduce the computation time and accuracy of our model.

**Visualization**

Visualization is done at wherever it is necessary. Bar graph, line graphs, pie charts,scatterplots, pyplot, maps etc. are used to represent data.

**Dataset Description**

List of Attributes:

| Features / Columns | Description |
| --- | --- |
| id | Unique id of every House in the dataset |
| url | URL of the house listing on craigslist |
| region | Regional location of the house like county, name of the place, etc. |
| region\_url | URL to get the regional information |
| price | Rent of the house which is our target variable |
| type | Type of house like Apartment, condo, duplex, flat, etc. |
| sqfeet | Size of house in Square-feet |
| beds | Number of Bedrooms in the house |
| baths | Number of Bathrooms in the house |
| cats\_allowed | Are cats allowed? 0 for No and 1 for YES |
| dogs\_allowed | Are dogs allowed? 0 for No and 1 for YES |
| smoking\_allowed | Is smoking allowed? 0 for No and 1 for YES |
| wheelchair\_access | Has wheelchair accessibility? 0 for No and 1 for YES |
| electric\_vehicle\_charge | Does the house have electrical vehicle charging? 0 for No and 1 for YES |
| comes\_furnished | Furnished or not? 0 for No and 1 for YES |
| laundary\_options | Type of laundry service available like in unit, hookups, on site, etc. |
| parking\_options | Type of parking available for the house like carport, attached garage, street parking, etc. |
| image\_url | URL of the house’s image |
| description | Description of the house |
| lat | Latitude location of the house |
| long | Longitude location of the house |
| state | State where the house is located |

**Data Source**

The dataset has been taken from iNeuron:

Source: https://challenge-ineuron.in/mlchallenge.php

**Analysis**

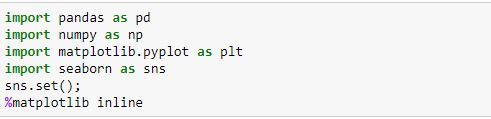
**Data Exploration**

The dataset has 22 columns. The aim of the project is to build a Machine Learning model to predict the rental prices of a house throughout the United States based on different variables describing the features of the houses. This dataset has multiple features that describes the

complete nature of the house and its dynamics, and a target feature ‘Price’ that is to be predicted.We managed to drop several columns and rows.

Following is the detailed explanation and screenshots of how the idea worked:

**1)Firstly we imported all the necessary libraries**



**2)To get more clear insight of data we entered the command data.describe()**

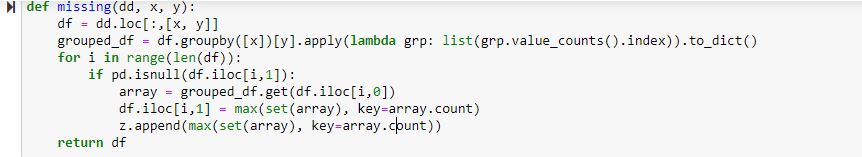


**3)Data Cleaning and Feature Selection**

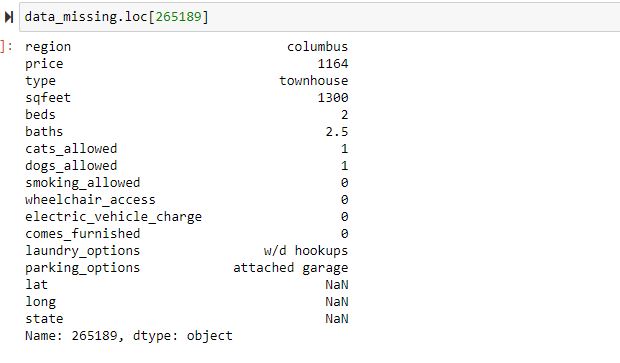
**Dropping Unwanted Columns:**



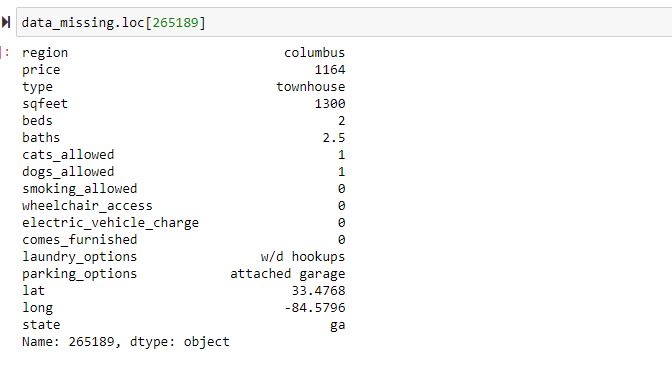
**We have created a function that fills missing values by grouping similar items of a column:**



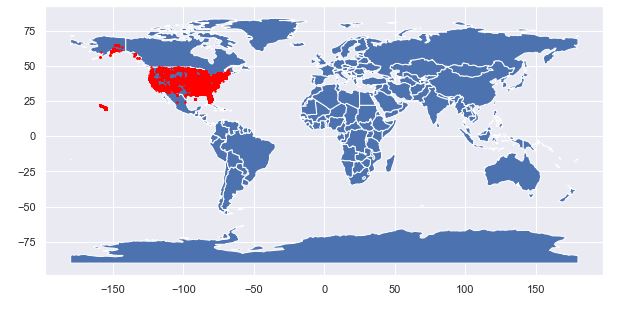
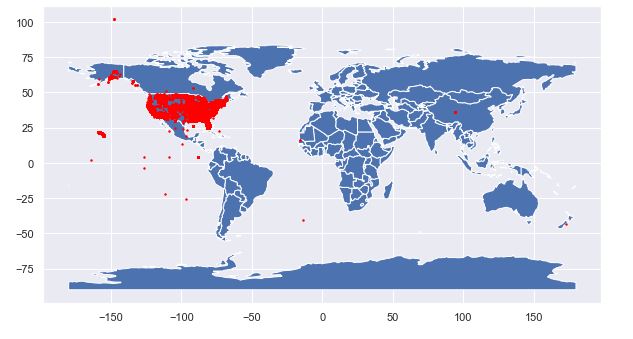
**Values before function was applied:**

****

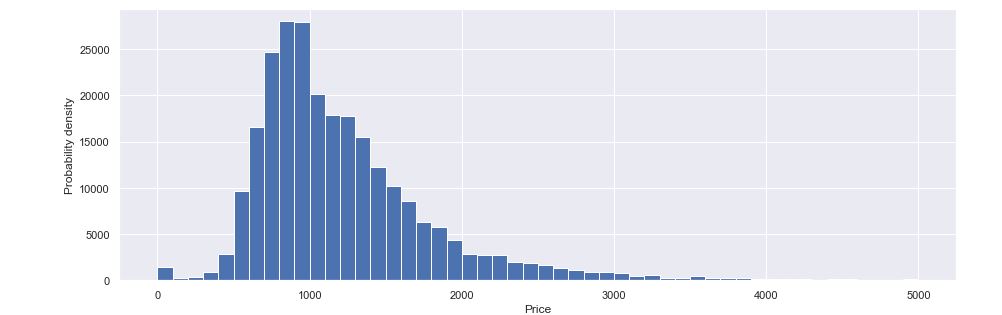
**Verifying the function has filled the missing value:**

****

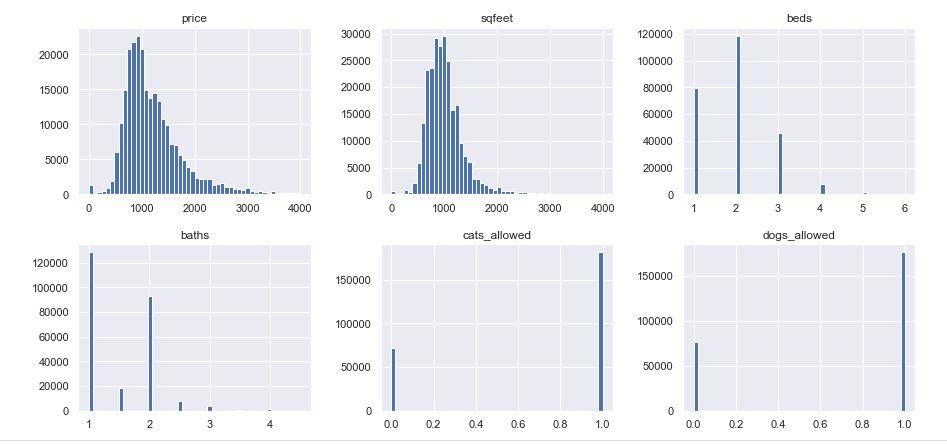
**Outliers Removal:**

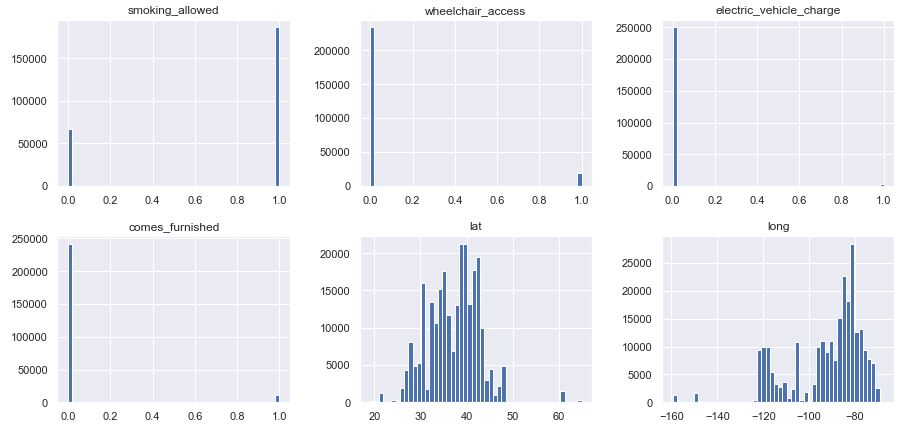
****

**EDA:**

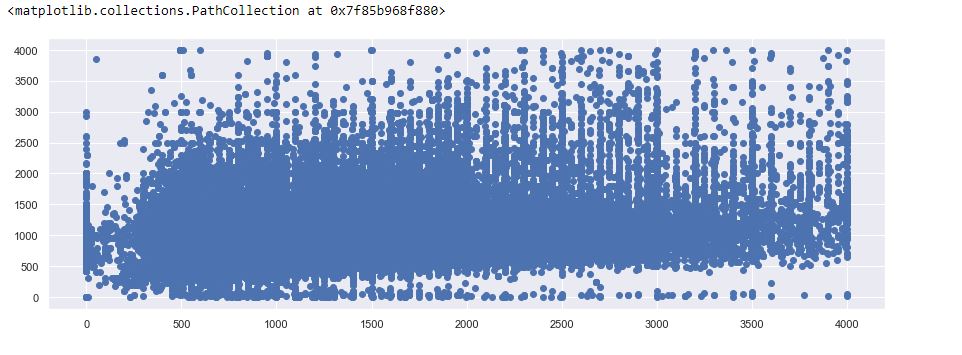
****

**Removing Price > $4000:**

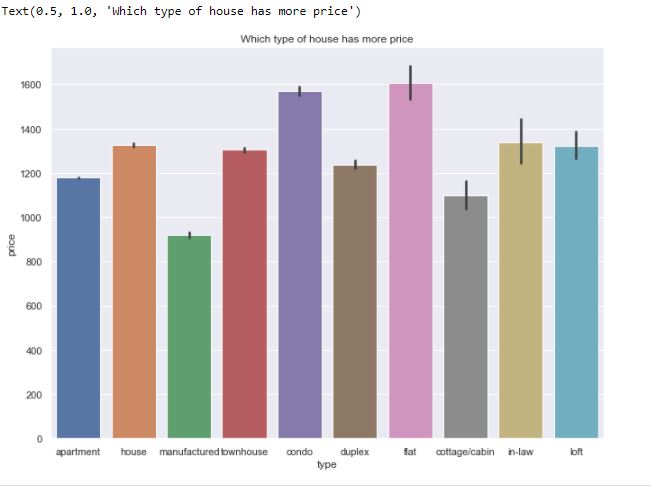
****

*Nationally, 91 percent of new homes built in 2018 were less than 4,000 square feet, with a median size of 2,386 square feet, according to the Census Bureau’s Survey of Construction. Restricting sqfeet in out data to 4000 square feet.Removed the price which is greater than $4000*

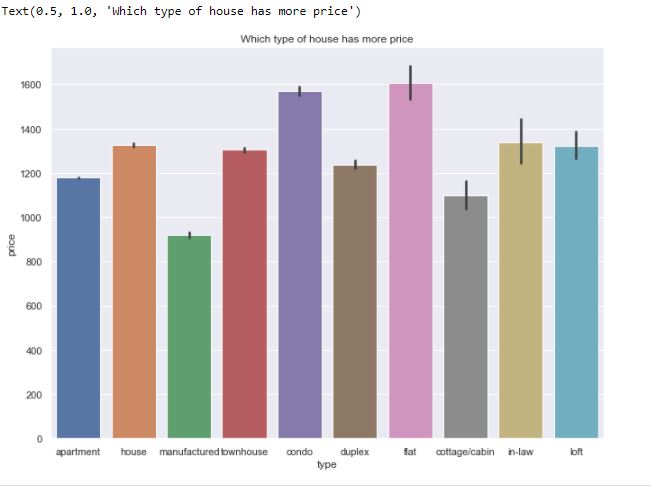
**Plotting histogram after encoding variables:**

****

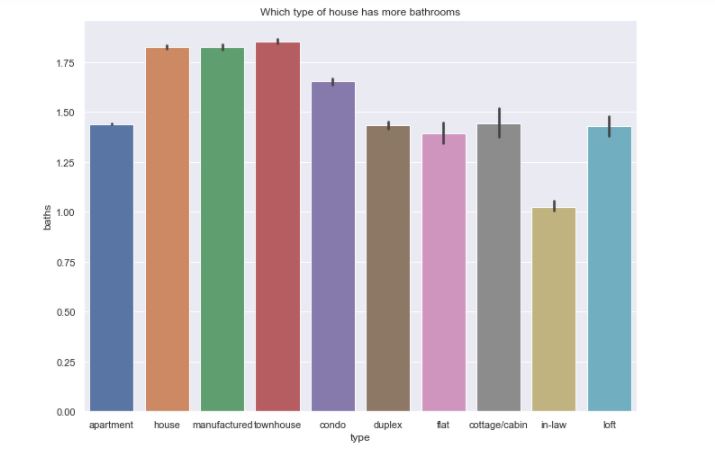
**Displaying which type of house has more price:**

****

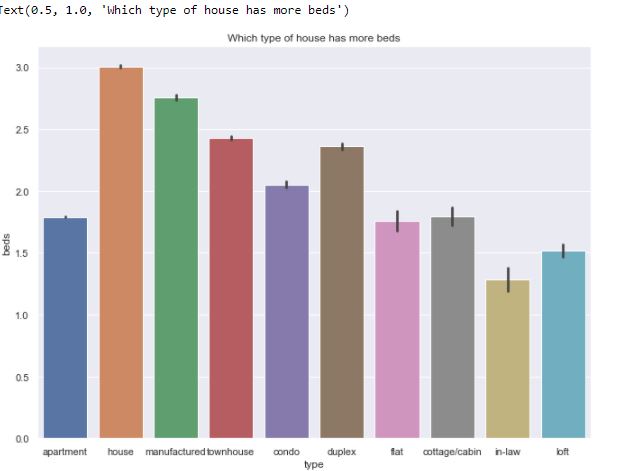
**Displaying which type of house has more sqfeet:**

****

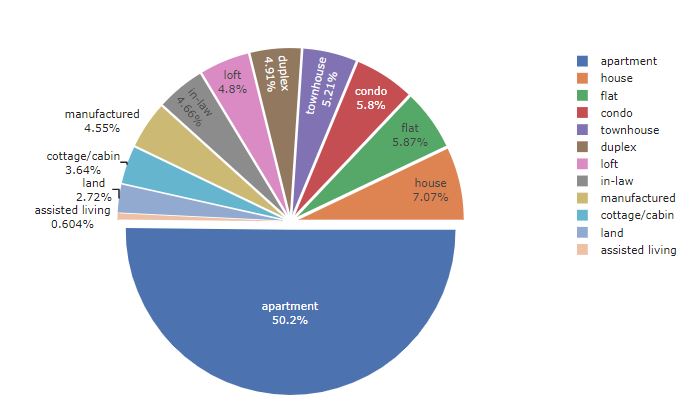
**Displaying which house has more number of Bathrooms:**

****

**Displaying which type of house has more number of Bedrooms:**

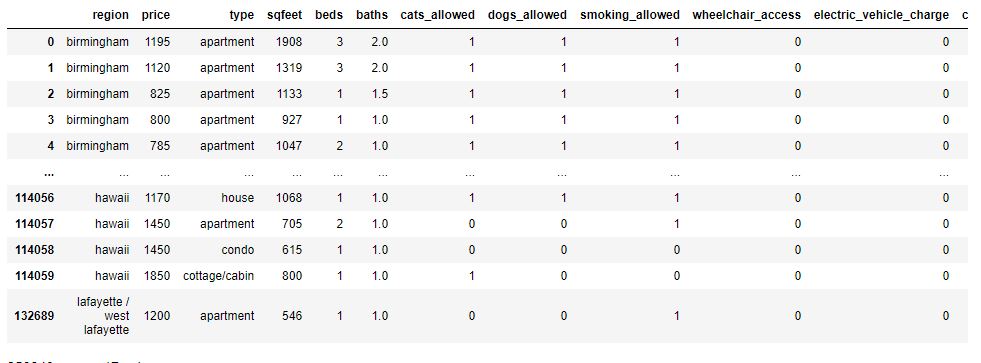
****

**Displaying polyplot to show overall %:**

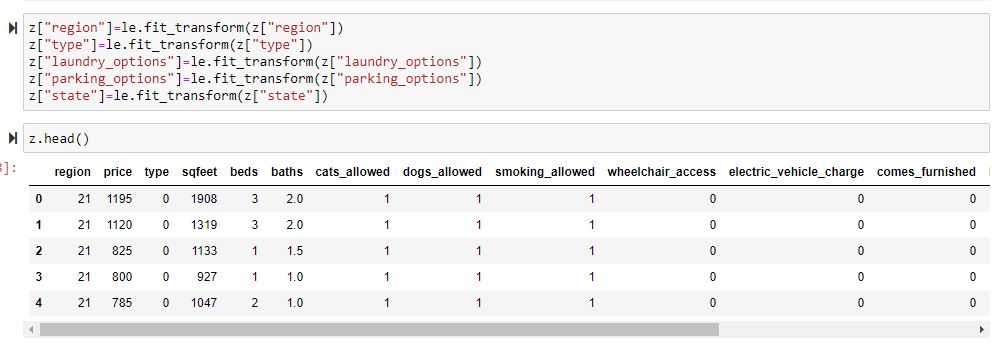
****

**Data Processing:**

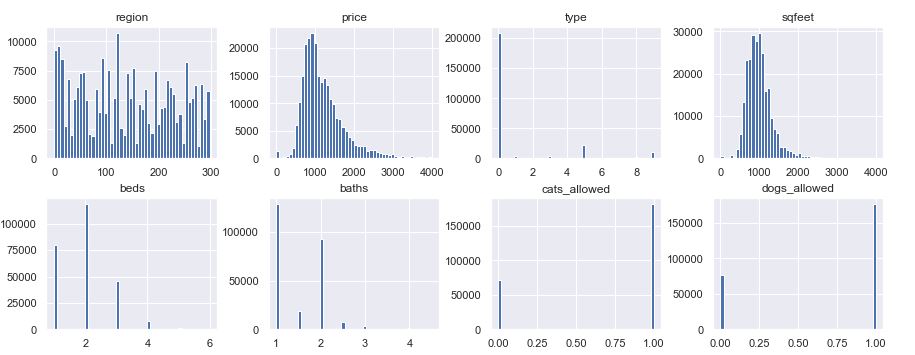
****

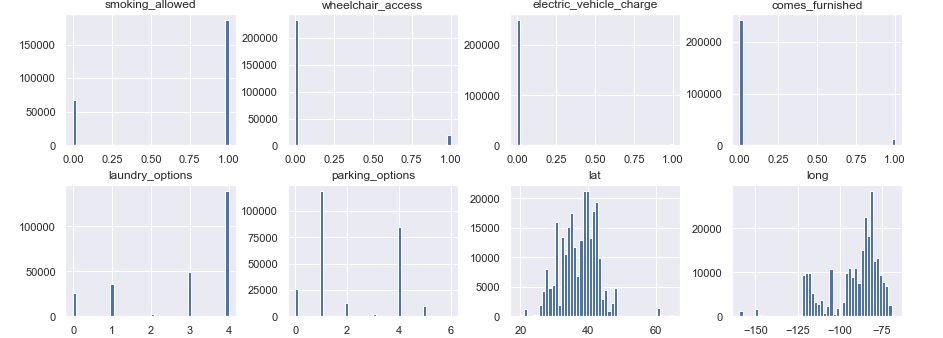
****

**Fit-Transform:**

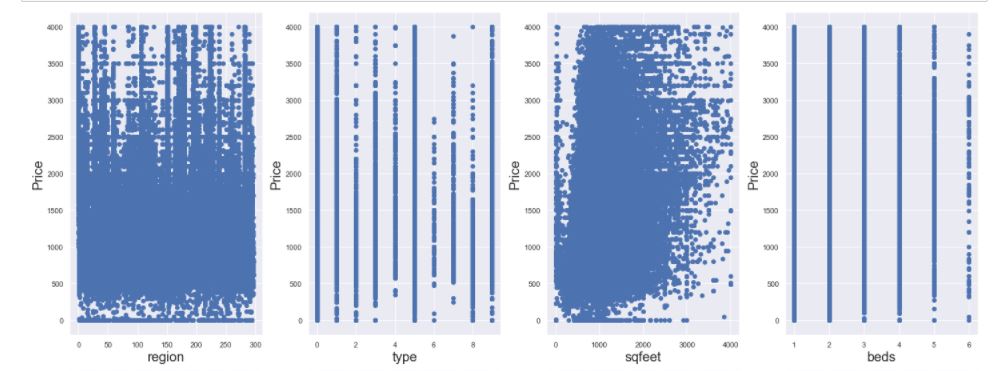
****

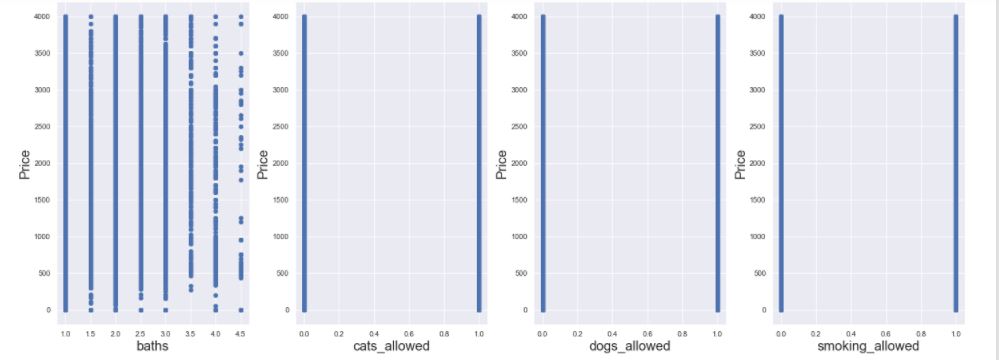
**All the data is converted to numerical format:**

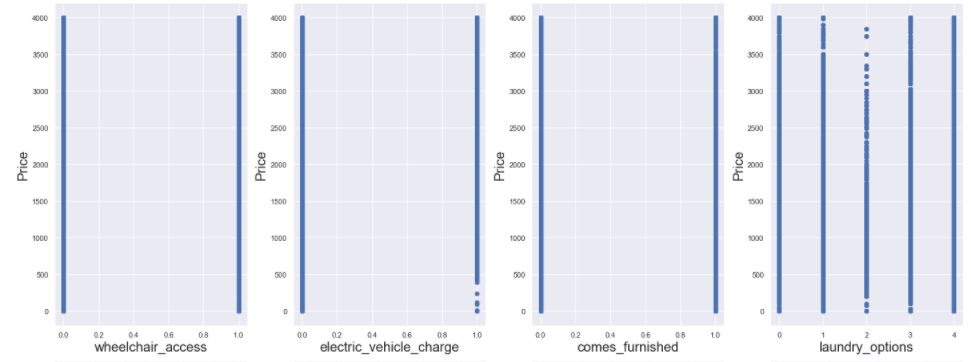
****

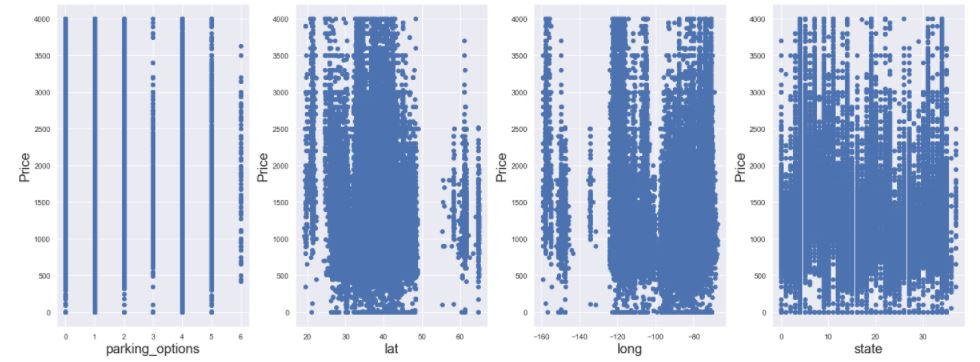
****

**Columns w.r.t Target Variable:**

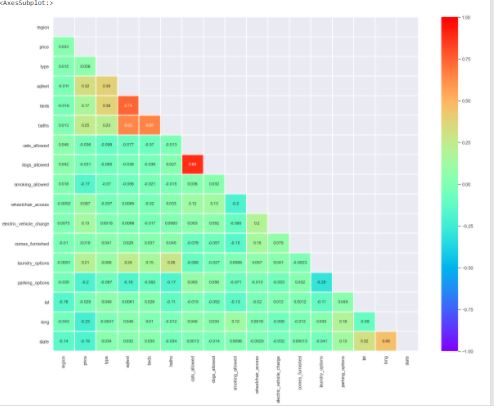
****

****

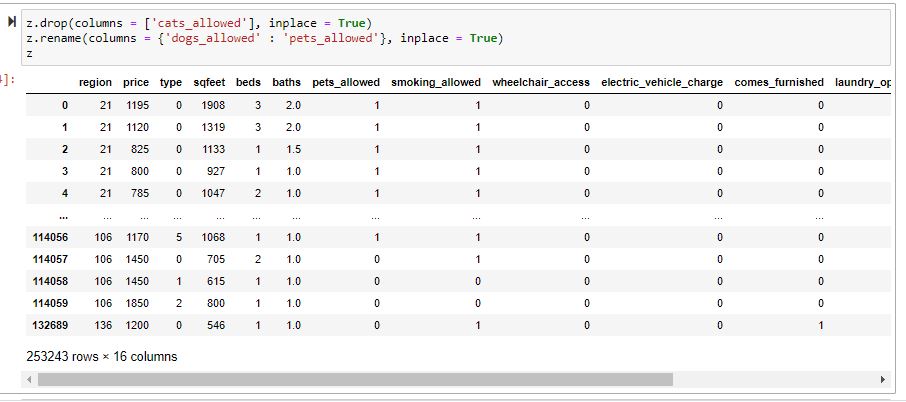
****



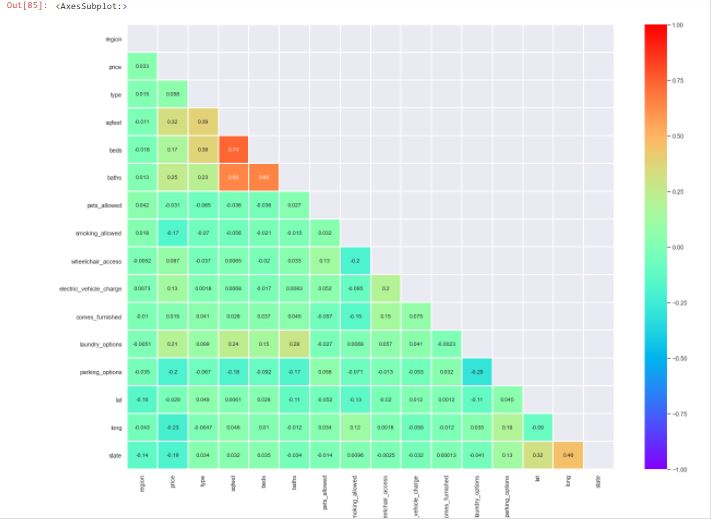
**Co-relation Matrix:**

****

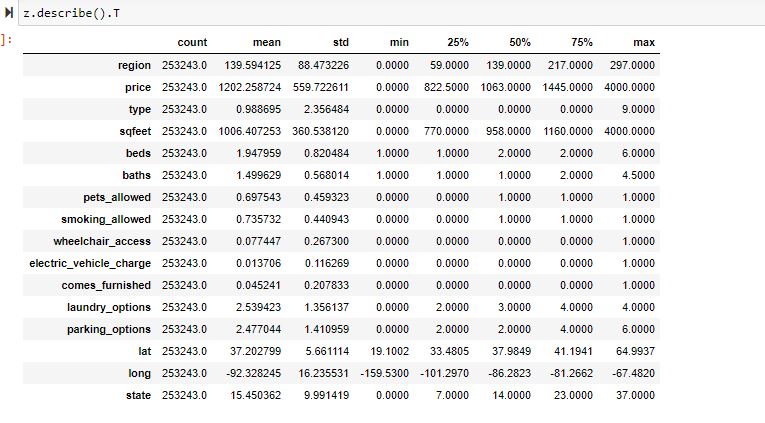
From the above graph we can see that cat\_allowed and dogs\_allowed have very high co-relation therefore we are dropping one column.

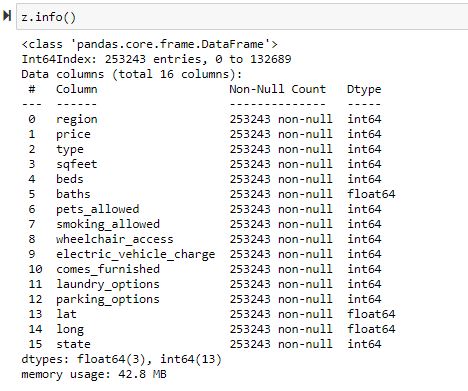
****

**Again displaying the co-relation matrix:**



**Final data description and size:**

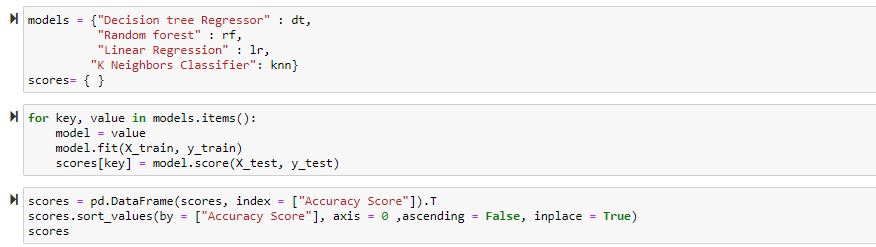
****

****

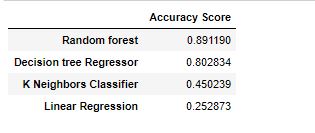
**Modelling and Data Prediction:**

****

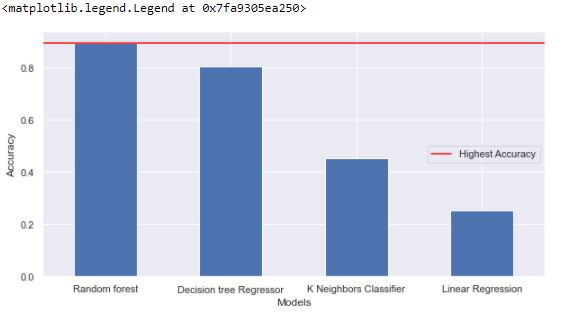
**We have created a dictionary of models:**



**Checking Accuracy Score:**

****

**Visualization of Accuracy Score:**

****

**Conclusion:**

We have done all the necessary Dataprocessing and Calculations, of all the Algorithms used Random Forest has the highest accuracy score for Prediction of Rents.

**References:**

* A Technical Report on Real Estate Rent prediction: <http://mars.gmu.edu/bitstream/handle/1920/11644/rafatirad_realestate.pdf?sequence=1&isAllowed=y>
* <https://towardsdatascience.com/interactive-visualizations-in-jupyter-notebook-3be02ab2b8cd>
* <https://towardsdatascience.com/ai-and-real-state-renting-in-amsterdam-part-1-5fce18238dbc>
* <https://towardsdatascience.com/exploratory-data-analysis-with-pandas-profiling-de3aae2ddff3>
* <https://towardsdatascience.com/overfitting-and-underfitting-in-machine-learning-89738c58f610>
* https://www.analyticsvidhya.com/blog/2021/01/a-quick-overview-of-regression-algorithms-in-machine-learning/