Group Members

Yong Quan Zhi, Tommy (A0195353Y / E0384984@u.nus.edu)

Ritesh Munjal (A0195304H / E0384935@u.nus.edu)

CA 1 Project Report

Problem solving using pattern recognition

Contents

[1 Introduction 2](#_Toc18509457)

[2 Tools and Techniques 3](#_Toc18509458)

[2.1 Tools 3](#_Toc18509459)

[2.2 Techniques 3](#_Toc18509460)

[2.2.1 Regression 3](#_Toc18509461)

[2.2.2 Classification 3](#_Toc18509462)

[3 Model Design 4](#_Toc18509463)

[3.1 Regression 4](#_Toc18509464)

[3.2 Classification 4](#_Toc18509465)

[4 Model Performance 4](#_Toc18509466)

[5 Project Findings 4](#_Toc18509467)

[6 Summary 5](#_Toc18509468)

[References 5](#_Toc18509469)

# 1 Introduction

With advances in artificial intelligence over the last 20 years, researchers have been exploring the possibilities of how much computers can replicate human behavior or execute complex tasks that perhaps most humans are unable to do. With pattern recognition, it is possible to enable a computer to learn without specifically programming it to execute specific tasks. Examples of such tasks include, but are not limited to, self-driving vehicles, giving recommendations on videos(Netflix, youtube), fraud detection, predicting consumer trends and etc.

Drawing inspiration from our previous project in our first semester at NUS ISS, our group has decided to go back to the home rental market and consider how landlords decide a price on renting out their homes.

With over 6 million listings world wide and an annual revenue of $3 billion, which is expected to grow by 250% in 2020, we chose AirBnb as a case study and sourced the internet for datasets on AirBnb listings. The following project report will detail our approach to the project and state our findings.



*AirBnb image, <* [*https://www.wsaw.com/fox/content/news/Stevens-Point-homeowners-can-rent-homes-on-AirBnB-and-VRBO-512899481.html*](https://www.wsaw.com/fox/content/news/Stevens-Point-homeowners-can-rent-homes-on-AirBnB-and-VRBO-512899481.html)*>.*

# 2 Tools and Techniques

## 2.1 Tools

The following items below are the tools and libraries used to implement this project.

|  |
| --- |
| Tools |
| Anaconda |
| Spyder |
| Libraries |
| SKLearn |
| Pandas |
| Numpy |
| Matplotlib |
| Math |
| statsmodel |

## 2.2 Techniques

During the course of the project, we decided to use both classification and regression techniques to not only predict home rental prices, but also to see what insights can be drawn from the data with relation to pricing.

### 2.2.1 Regression

As the problem statement is to be able to predict the daily lease price of a home/room, regression techniques are used as the output will be a type of continuous data. As such, it is also important to choose a good variety of models that are linear, non-linear and of branching types and analyze them to choose the best model. Below are the types of machine learning models that will be used for our regression problem.

Machine Learning Models

1. Multiple Linear Regression Model
2. Polynomial Regression Model
3. Decision Tree Regression Model
4. Random Forest Regression Model

### 2.2.2 Classification

# 3 Model Design Process

The first step in designing our machine learning model is to understand the problem statement that we would like to tackle for this project. In our first semester, we created an intelligent home rental recommendation system which was able to recommend users what home listings they could rent based on their responses from a list of predetermined questions. For this CA1 project, we wanted to see if we can explore the idea of being able to predict the price of a home/room so that owners will be able to list their accommodations on the site and the system will be able to provide a recommendation on pricing.

With our problem statement, we searched the web for an appropriate dataset of at least 30 features and at least 3000 samples. The search led us to Kaggle.com and the chosen dataset was for Airbnb which has about 94 features and over 30000 samples. This was more than enough for the purposes of our project and would pose the right amount of challenges for us to process the data and try to see if we can build a feasible model with it.

The following subsections will dive into our classification and regression approaches and how we went about designing the models.

## 3.1 Regression



The first step was to take a look at the dataset and have an idea of what kind of columns were present and how data is represented for each column. As we would need to try and reduce the number of columns to reduce model complexity, columns that were obviously irrelevant to the purposes of the model were noted down so that they could be programmatically removed in our python script.

Next, we proceeded to handle missing data. Through inspection, about a third of the 30000 samples had columns with missing data. It was possible to use data imputation to deal with the missing data, either by replacing it with the mode or mean value of the affected column. In the end, we opted to drop the rows with missing data. This was to ensure the replacing of the mean or mode of the data would not affect the model greatly and to see what could be achieved without the samples with missing data.

Before we proceeded to assign the target and independent variables, we shuffled the data as it was arranged by region and this could have affected our model’s performance as the data could have be split improperly and given the training process an inaccurate spread of samples. This issued was realized only after training had already begun and will be elaborated more later.

Once the target and independent variables had already been assigned, columns were checked to see which were categorical and should be encoded with one hot encoding. After that, the ordinal features were simply encoded.

Now that we had columns of features that were of numerical value it was time to see if we could further trim down the number of columns in order to reduce model complexity and improve performance. The p-value of each column with respect to the target variable was used to eliminate undesired columns. The method used for this was backward elimination and the statsmodel library was used to facilitate this process. Initially, the significance level was set at 0.05, however, as there were still too many columns left it was further reduced to 0.000000001.

The last bit was to scale the features and split them into training and test data. The four machine learning regression models used were multiple linear, polynomial, decision tree and random forest.

## 3.2 Classification

# 4 Model Performance

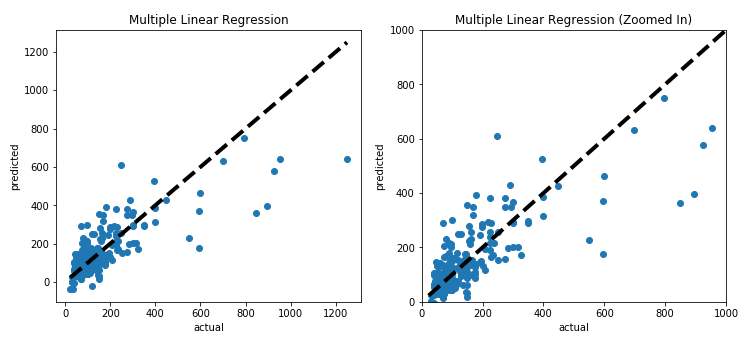
## 4.1 Regression

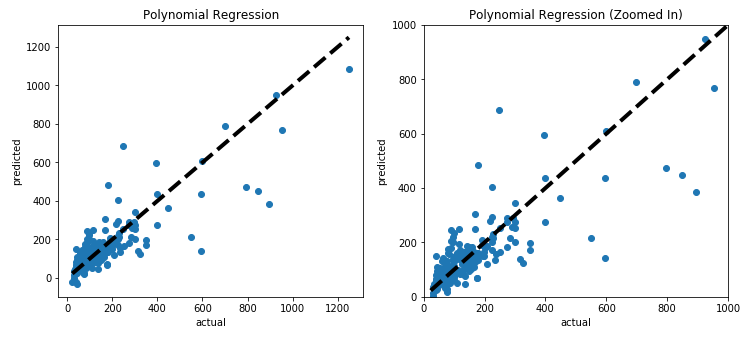
This section will display the model’s performance summary, scatter and distribution plots between predicted and actual Y-values. An analysis for them will be provided in the project findings section below.

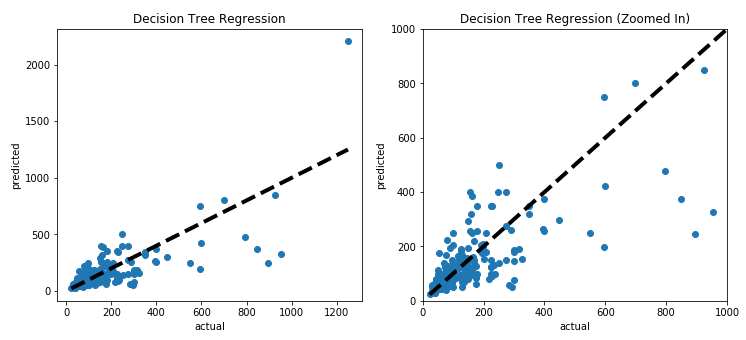
Model Summary (Rounded to nearest 2 decimals)

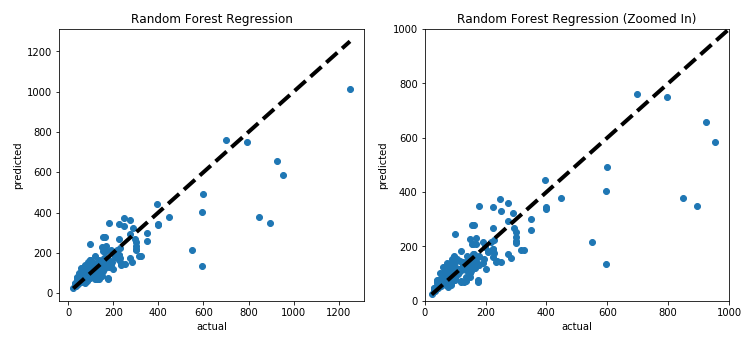
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Multiple Linear | Polynomial | Decision Tree | Random Forest |
| Explained Variance | 0.61 | 0.71 | 0.46 | 0.46 |
| Mean absolute sq error | 66.96 | 53.47 | 66.17 | 66.17 |
| Root mean sq error (RMSE) | 108.26 | 93.81 | 127.29 | 127.29 |
| Median absolute error | 44.74 | 27.92 | 30.0 | 30.0 |
| R2 score | 0.61 | 0.71 | 0.46 | 0.75 |

Scatter Plots (Predicted vs Actual)

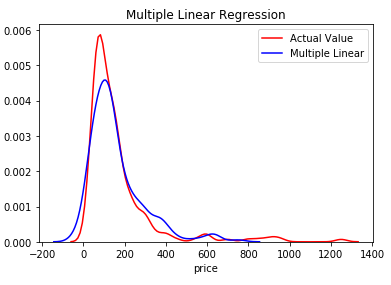


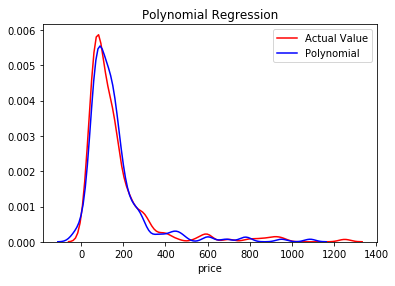


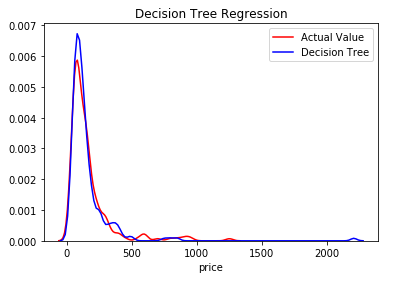


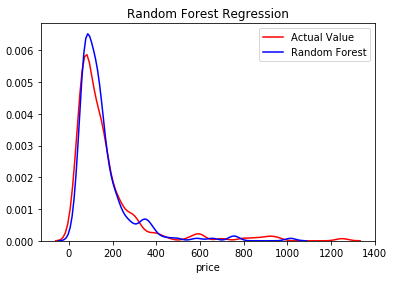


Distribution Plots (Predicted vs Actual)









## 4.2 Classification

# 5 Project Findings

## 5.1 Regression

The main performance metric that will be used for our regression models will be based on mean squared error(MSE). We used root mean squared error(RMSE) in printing out our results so that it is easier for us to read and understand the figures.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Multiple Linear | Polynomial | Decision Tree | Random Forest |
| Explained Variance | 0.61 | 0.71 | 0.46 | 0.46 |
| Mean absolute sq error | 66.96 | 53.47 | 66.17 | 66.17 |
| Root mean sq error (RMSE) | 108.26 | 93.81 | 127.29 | 127.29 |
| Median absolute error | 44.74 | 27.92 | 30.0 | 30.0 |
| R2 score | 0.61 | 0.71 | 0.46 | 0.75 |

## 5.2 Classification

# 6 Summary

# References

1. <https://ipropertymanagement.com/airbnb-statistics/#targetText=The%20target%20market%20for%20Airbnb,guest%20arrivals%20at%20Airbnb%20listings.>
2. <https://www.sas.com/en_sg/insights/analytics/machine-learning.html>
3. <https://towardsdatascience.com/machine-learning-general-process-8f1b510bd8af>
4. <https://www.youtube.com/watch?v=wpQiEHYkBys>