ANALYSIS OF TWO SIGMA RENTAL DATA

**CS 636: Data Analytics with R Programming**

By

Bhavneet Singh (bs432)

**MASTER OF SCIENCE IN COMPUTER SCIENCE**

**New Jersey Institute of Technology**

INDEX

**1. Project Overview..................................................................................................................................... 3**

**2. Data Information ……................................................................................................................................. 4**

**3. Source Code ............................................................................................................................................. 5**

**4. Data Pre-processing ...................................................................................................................................8**

**5. Data Modeling ....................................................................................................................................... . .9**

**6. Screenshots ............................................................................................................................................ . . 10**

Project Overview

The goal of the project is to find out on the basis of the attributes listed, which apartment would be of the buyer’s interest.

**Data Information**

**Files:**

**train.json** - the training set

**test.json** - the test set

**sample\_submission.csv** - a sample submission file in the correct format

**images\_sample.zip** - listing images organized by listing\_id

**Data fields:**

**bathrooms**: number of bathrooms of the apartment

**bedrooms**: number of bathrooms of the apartment

**building\_id:** Building ID number

**created** **description:** Description of the apartment

**display\_address:** Address where the apartment is located

**features**: a list of features about this apartment

**latitude and longitude:** Geographical location of the apartment

**listing\_id:** ID number of the apartment

**manager\_id:** ID number of the person who is the manager of the apartment.

**photos**: a list of photo links.

**price**: in USD

**street\_address:** The location of the apartment

**interest\_level**: this is the target variable. It has 3 categories: 'high', 'medium', 'low'

**Source Code**

library(UsingR)

library(RJSONIO)

library(nnet)

kaggledataRaw = fromJSON("C:\\Users\\Jagpreet Singh\\Documents\\homework3\\train.json\\train.json")

kaggleDataTest = fromJSON("C:\\Users\\Jagpreet Singh\\Downloads\\test.json\\test.json")

#head(KaggleData)

KaggleData = kaggledataRaw[['interest\_level']]

Bathroom = kaggledataRaw$bathrooms

Bedroom = kaggledataRaw$bedrooms

Address = kaggledataRaw$display\_address

Building\_ID = kaggledataRaw$building\_id

Created = kaggledataRaw$created

Description = kaggledataRaw$description

Features = kaggledataRaw$features

Latitude = kaggledataRaw$latitude

Listing\_ID = kaggledataRaw$listing\_id

Longitude = kaggledataRaw$longitude

Manager\_ID = kaggledataRaw$manager\_id

Photos = kaggledataRaw$photos

Price = kaggledataRaw$price

Street\_address = kaggledataRaw$street\_address

Interest=kaggledataRaw$interest\_level

KaggleData\_TEST = kaggleDataTest[['interest\_level']]

Bathroom\_TEST = kaggleDataTest$bathrooms

Bedroom\_TEST = kaggleDataTest$bedrooms

Address\_TEST = kaggleDataTest$display\_address

Building\_ID\_TEST = kaggleDataTest$building\_id

Created\_TEST = kaggleDataTest$created

Description\_TEST = kaggleDataTest$description

Features\_TEST = kaggleDataTest$features

Latitude\_TEST = kaggleDataTest$latitude

Listing\_ID\_TEST = kaggleDataTest$listing\_id

Longitude\_TEST = kaggleDataTest$longitude

Manager\_ID\_TEST = kaggleDataTest$manager\_id

Photos\_TEST = kaggleDataTest$photos

Price\_TEST = kaggleDataTest$price

Street\_address\_TEST = kaggleDataTest$street\_address

Interest\_TEST=kaggleDataTest$interest\_level

#

# i=1

# num\_rat=c(0)

#

#

# while(i<=length(KaggleData))

# {

# if(KaggleData[[i]]=="high")

# {

# num\_rat[i] = "1"

# }

# else if(KaggleData[[i]]=="medium" | KaggleData[[i]]=="low")

# {

# num\_rat[i] = "0.5"

# }

#

# i=i+1

# }

real\_data<- data.frame(Interest,Bathroom,Bedroom,Latitude,Longitude,Price)

real\_datatest=data.frame(Interest\_TEST,Bathroom\_TEST,Bedroom\_TEST,Latitude\_TEST,Longitude\_TEST,Price\_TEST)

head(real\_data)

nrow(Bedroom\_TEST)

set.seed(897)

idxTrain <- sample(nrow(real\_data),as.integer(nrow(real\_data)\*0.5))

train.real\_data = real\_data[idxTrain,]

test.real\_data = real\_data[-idxTrain,]

# glm\_fit<- glm(num\_rat~Bathroom+Bedroom+Price+Latitude+Longitude,data = train.real\_data,family = "binomial")

# summary(glm\_fit)

# predict\_fit<- predict(glm\_fit,test.real\_data,type = "response")

# prediction <- rep(0,length(predict\_fit))

# prediction[predict\_fit > 0.5] <- 1

# prediction[predict\_fit < 0.5] <- 0

# accuracy=sum(diag(table(test.real\_data$num\_rat,prediction)))/nrow(test.real\_data)

# accuracy

multi=multinom(Interest~.,data = train.real\_data,family = "multinomial")

summary(multi)

predict\_fit<- predict(multi,test.real\_data)

accuracy=sum(diag(table(test.real\_data$Interest,predict\_fit)))/nrow(test.real\_data)

accuracy

z= summary(multi)$coefficients/summary(multi)$standard.errors

p = (1-pnorm(abs(z),0,1))\*2

coeff = summary(multi)$coefficients

std.err = summary(multi)$standard.errors

t = cbind(t(coeff),t(std.err),t(z),t(p))

t= unique(t)

t

rm(list = ls())

**Data Preprocessing:**

The dataset had many variables to look at. Like Description, Building ID, Photos, etc.

We did not consider those variables while regressing. Because from our prediction we thought that, these values would not have too much effect on the prediction and thereby would have not affected the accuracy.

So the only variables that we took into our considertation were:

Interest, Bathroom, Bedroom, Latitude, Longitude, Price.

We compared Interest against the rest of the variables.

**Data Modelling:**

We were first just going to compare the interest level with the rest of the variables using a linear logistic regression approach. That is by putting 0 values for the "low” and “ medium” and 1 for the “high” .

But then we thought of using a multinomial logistic model. So we considered, medium, low and high. This gave us a better prediction and accuracy.

**SCREENSHOTS:**

