Kaggle Report

Romauli Butarbutar 12 April 2021

Part 1 - The steps of data analysis

The first part of this project involves conducting the steps of data analysis to generate a set of predictions. There are many types of data including continuous, categorical, text, and date features. I try to categories data for better understanding.

a. Listing/URL descriptors
b. Host descriptors
c. Location descriptors
d. Property descriptors
e. Price descriptor
f. Term of condition descriptor
g. Reviews descriptor
h. Additional descriptor

After review the data, refer to the common assumption of rental price factors, I exclude some of data that is not relevant to estimate the price, so I will not do further analysis as it has same data or all value are null so I assumed it is not significant for example: Additional descriptor, Listing/URL descriptors**.** I just include several important variables including Location descriptors and Property descriptors. I will also analysis some of other relevant variables from Term of condition descriptor and Reviews and Host descriptor. Other important step in data cleaning is to analysis the word count of the room description/summary and amenities.

A. Data Wrangling, Cleaning and Tidying

Visualizing Data Visualizing data can aid in forming an understanding of the data, identifying trends, and spotting anomalies.

```
# ***Examine outliers----
library(ggplot2)
ggplot(data=train,aes(x='',y=price))+
geom_boxplot(outlier.color='red',outlier.alpha=0.5, fill='cadetblue')+
  geom_text(aes(x='',y=median(price),label=median(price)),size=3,hjust=11)+
  xlab(label = '')
```

Missing and Parsing Data

One of the issues is variables not being in the correct format, in this case including:

- last_scraped,host_since,calendar_last_scraped,first_review,last_review : type is character, need to change to date class
- thumbnail_url, medium_url, xl_picture_url,license : logical with NA?

- host_is_superhost,host_has_profile_pic,host_identity_verified,is_location_exact,has_availability,requires_license,instant_bookable,is_business_travel_ready,require_guest_profile_picture,require_guest_phone_verification : chr "f" should be factor?
- square_feet,price,weekly_price,monthly_price,security_deposit,cleaning_fee : data type integers but so many NA value, we need to clean the data
- check country : Uruguay? Country Code:UY, we remove this data. - blank data in several significant data like beds, square_feet. For other blank data will be ignored in this analysis

B. Feature Selection

We will perform several technique of feature selections to get more relevance predictors. This selection will reduce the effect of multicollinearity as theoretically the number of predictors increases, the chance of finding correlations among a predictor or a set of predictors increases that leading to inflation of standard errors of coefficients and erroneous conclusions.

1. **Corrplot**
   The result of Corrplot shows that the Variance Inflating Factor (VIF) in the range of 1<VIF<5 that means no significant of threat of collinearity.
2. **Best Subset Selection**
   After perform best subset selection, the least RSME is about 67.90036
3. **Forward and Hybrid Selection**
   After perform best forward stepwise and hybrid stepwise, the least RSME is about 67.93774, there is no significant different with best subset selection
4. **Shrinkage : Lasso & PCA**
   After perform shrinkage, the least RSME using Lasso method is about 68.38622, higher RMSE than the previous technique, while Dimension Reduction is not working for some errors that I need to find out.

C. Creating Model

I perform some models with several technique to get less RMSE value.

1. **LINEAR REGRESSION MODEL**
   1.1. Linear Regression: Property Descriptors
   1.2. Linear Regression: Location Descriptors
   1.3. Linear Regression: Host Descriptors
   1.4. Linear Regression: Location Descriptors
   1.5. Linear Regression: Condition Descriptors
   1.6. Linear Regression: Significant Variables

   After perform several linear regression of descriptor categories and selected significant independent variables from each model, the least RSME is about 67.89162

2. **TREE MODEL**

2.1. Simple Regression Tree

After perform Simple Regression Tree, the least RSME is about 72.12098, higher RMSE than the previous technique.

2.2. Regression Tree Complex

Let us construct a larger tree by changing the value of cp. In general, the smaller the value of cp, the larger the tree and the greater the complexity of the model. After perform Complex Regression Tree, the least RSME is about 69.49447, lowest RMSE than Simple Regression Tree.

2.3. Advanced Tree

Perform Advanced Tree technique using the default value of cp is 0.01. After perform Advanced Tree, the least RSME is about 51.00288, lowest RMSE than other methods, but with note for file submission is not working and there are some errors that I need to address.

2.4. Tree with Tuning

I also perform Tune the complexity of a tree using 5-fold cross-validation. Here, we will examine cross-validation error for 100 different values of cp. After perform Advanced Tree, the least RSME is about 64.24447, higher than previous method.

3. BOOTSRAPPING MODELS

Bootstrap Aggregation models generate a large number of bootstrapped samples. A tree is fit to each bootstrapped sample. Predictions are generating as an average of all models (for numerical outcome variables) or the majority group (for categorical outcome variables). When I try to perform Random Forest, Tuned Random Forest, Forest with Ranger and Boosting with cross-validation and Boosting with XGBoost it took so long time, so I decided to cut the process by terminating R.

4. FINAL BOOSTING MODELS

After perform Data Cleaning Complexity, and include some wordcount and amenities, this is my final model with boosting method.

Part 2 - Lessons Learned

The Lesson I have learned from the experience of AirBnB Kaggle Competition including more understanding in doing data analysis such as read the data, perform exploratory data, data wrangling, cleaning and tyding the data. Then perform first data analysis modeling using several linear regression approaches. Following feature selection to gain more relevance variables, then using the result to perform more complex modeling like Tree model, Boosting, Random forest etc. The final step I learn how to find the insight from the data and learn how to communicate the result using data visualization.

Most of my time is spent to find the most relevance variables, fit it into the model and when the model is too complex, it will need more running time of R to process it and get the result. As it said, most of the data scientist time is spent simply finding, cleansing, and organizing data, leaving only small amount of their time to actually perform analysis modelling”

More importantly, I realize that the level of complexity of the model and variables probably could lead to overfitting problem. It is important to use the common knowledge and use a good intuition how to logically select the relevant variables for a model, like this quote:

"It is through science that we prove. But it is through intuition that we discover" — Henri Poincare

Part 3 - Report Summarizing

To summarize this project, I would describe:

A. The insights from exploring the data

According to this kaggle prediction, I can inference the price of an Airbnb rental affected by neighbourhood, room type, number of bedrooms, accommodates, square_feet, etc. Based on the prediction model result, we can suggest the best price that the landlord offer to the renters and we we can also customize the housing recommendation based on customers price preference.Before doing a deep analysis using different approaches of modelling, I think the most significant independent variable to predict the price would be mainly related to LOCATION. However, after analyze the result, some other variables seem to be more significant like number of reviews, score reviews, availability, cancellation policy and the credibility of the host. Also, for such kind of online rental like AirBnB, the detail description of apartment including summary, host_about, neighborhood_overview etc would be affected the popularity, so I suggest to include the detail description/ summary in the listing to gain more users and popularity.

B. Efforts to prepare the data

I need to do the data wrangling, cleaning and tidying several times, go and back to first process and do it again and again. In order to do the analysis of character data type like amenities I should explore and implement so Regular Expression functions in R.
At the first time, there are so many errors and I learn how to solve it using helper from many sources like website and R documentation.

C. Exploring Analysis techniques

As aforementioned, I perform several analysis techniques and I think almost trying all the methods from classes, even I cannot find the lowest RMSE value though. By exploring analysis techniques, I get more handfuls technical skill of Analysis techniques.

D. The Failed Steps or Missteps along the way

- Some of my models like XGBoosting Model, Tuning the Tree are not working and still confuse with the error.
- When I perform Dimension Reduction Technique, I found some errors that I decided to cut the process.

- Performing several techniques of Forward selection and Hybrid selection would result insignificant difference, so I think we just need to choose one for time efficiency.

E. APPENDIX – FINAL CODE FOR SUBMISSION

```
#clear memory
rm(list=ls())

#read the data
train <- read.csv("../Data/train.csv", stringsAsFactors = T)
test <- read.csv("../Data/test.csv", stringsAsFactors = T)

dim(train)
dim(test)

str(train)
str(test)

head(train)

# DATA CLEANING, WRANGLING, TIDYING

#Reformat data, convert data type
library("tidyverse")
library(dplyr)

#Reformat Date Class
train$last_scraped = as.Date(train$last_scraped)
train$host_since = as.Date(train$host_since)
train$calendar_last_scraped = as.Date(train$calendar_last_scraped)
train$first_review = as.Date(train$first_review)
train$last_review = as.Date(train$last_review)
```

Kaggle Report

Romauli Butarbutar 12 April 2021

```
test$last_scraped = as.Date(test$last_scraped)

test$host_since = as.Date(test$host_since)

test$calendar_last_scraped = as.Date(test$calendar_last_scraped)

test$first_review = as.Date(test$first_review)

test$last_review = as.Date(test$last_review)


#Count the days

train$last_scraped_days = as.numeric(as.Date("2021-04-11") - train$last_scrap
ed)

train$host_since_days = as.numeric(as.Date("2021-04-11") - train$host_since)

train$calendar_last_scraped_days = as.numeric(as.Date("2021-04-11") - train$c
alendar_last_scraped)

train$first_review_days = as.numeric(as.Date("2021-04-11") - train$first_revi
ew)

train$last_review_days = as.numeric(as.Date("2021-04-11") - train$last_review
)


test$last_scraped_days = as.numeric(as.Date("2021-04-11") - test$last_scraped
)

test$host_since_days = as.numeric(as.Date("2021-04-11") - test$host_since)

test$calendar_last_scraped_days = as.numeric(as.Date("2021-04-11") - test$cal
endar_last_scraped)

test$first_review_days = as.numeric(as.Date("2021-04-11") - test$first_review
)

test$last_review_days = as.numeric(as.Date("2021-04-11") - test$last_review)




#Character Class

train$summary = as.character(train$summary)

train$space = as.character(train$space)

train$description = as.character(train$description)

train$neighborhood_overview = as.character(train$neighborhood_overview)

train$transit = as.character(train$transit)

train$access = as.character(train$access)

train$house_rules = as.character(train$house_rules)
```

Kaggle Report

Romauli Butarbutar 12 April 2021

```
train$host_location = as.character(train$host_location)
train$host_about = as.character(train$host_about)
train$host_verifications = as.character(train$host_verifications)
train$amenities = as.character(train$amenities)
train$neighbourhood_cleansed = as.character(train$neighbourhood_cleansed)

test$summary = as.character(test$summary)
test$space = as.character(test$space)
test$description = as.character(test$description)
test$neighborhood_overview = as.character(test$neighborhood_overview)
test$transit = as.character(test$transit)
test$access = as.character(test$access)
test$house_rules = as.character(test$house_rules)
test$host_location = as.character(test$host_location)
test$host_about = as.character(test$host_about)
test$host_verifications = as.character(test$host_verifications)
test$amenities = as.character(test$amenities)
test$neighbourhood_cleansed = as.character(test$neighbourhood_cleansed)

#exclude data
train <- train[train$country_code != 'UY',]
test <- test[test$country_code != 'UY',]

#check blank data and impute missing value
blank_data = colSums(is.na(train))
blank_data[blank_data > 0]

#train
for (i in 1:ncol(train)) {
  if (is.numeric(train[,i])) {
    train[is.na(train[,i]), i] = mean(train[,i], na.rm = TRUE)
  }
  if (is.factor(train[,i])) {
```

Kaggle Report

Romauli Butarbutar 12 April 2021

```
    train[,i] = addNA(train[,i])
  }
}
#test
for (i in 1:ncol(test)) {
  if (is.numeric(test[,i])) {
    test[is.na(test[,i]), i] = mean(test[,i], na.rm = TRUE)
  }
  if (is.factor(test[,i])) {
    test[,i] = addNA(test[,i])
  }
}

# colSums(is.na(test))
sum(is.na(train$summary))
sum(is.na(test$summary))



#Word count for charater data types
library(ngram)
library(dplyr)
train = train %>%
  rowwise() %>%
  mutate(wc_summary = wordcount(summary),
         wc_space = wordcount(space),
         wc_description = wordcount(description),
         wc_neighborhood_overview = wordcount(neighborhood_overview),
         wc_transit = wordcount(transit),
         wc_access = wordcount(access),
         wc_house_rules = wordcount(house_rules),
         wc_host_location = wordcount(host_location),
         wc_host_about = wordcount(host_about),
         wc_host_verifications = wordcount(host_verifications),
```

Kaggle Report

Romauli Butarbutar 12 April 2021

```
         wc_amenities = wordcount(amenities),
         wc_neighbourhood_cleansed = wordcount(neighbourhood_cleansed)
  )

test = test %>%
  rowwise() %>%
  mutate(wc_summary = wordcount(summary),
         wc_space = wordcount(space),
         wc_description = wordcount(description),
         wc_neighborhood_overview = wordcount(neighborhood_overview),
         wc_transit = wordcount(transit),
         wc_access = wordcount(access),
         wc_house_rules = wordcount(house_rules),
         wc_host_location = wordcount(host_location),
         wc_host_about = wordcount(host_about),
         wc_host_verifications = wordcount(host_verifications),
         wc_amenities = wordcount(amenities),
         wc_neighbourhood_cleansed = wordcount(neighbourhood_cleansed)
  )



#Check different types of amenities to be included in the model analysis usin
g Regular Expressions functions
# Then we will choose the common searching amenities to included in rental


# head(train$amenities)
# head(test$amenities)


library(stringr)
library(qdapTools)
library(eply)
# install.packages("eply")
```

Kaggle Report

Romauli Butarbutar 12 April 2021

```
train$amenities = gsub("[^[:alnum:]]", "", train$amenities)
train$amenities = gsub('([[:upper:]])', ' \\1', train$amenities)
train$amenities =  strsplit(train$amenities," ")
train = cbind(train, mtabulate(train$amenities))

test$amenities = gsub("[^[:alnum:]]", "", test$amenities)
test$amenities = gsub('([[:upper:]])', ' \\1', test$amenities)
test$amenities =  strsplit(test$amenities," ")
test = cbind(test, mtabulate(test$amenities))



# neighbourhood_cleansed : multiple factoral levels
train_nc = train %>%
  group_by(neighbourhood_cleansed) %>%
  summarize(n = n(), meanPrice = mean(price)) %>%
  arrange(desc(n))

train = merge(train, train_nc, by = c("neighbourhood_cleansed", "neighbourhoo
d_cleansed"))

## Just get 50 level
train$neighbourhood_cleansed = as.character(train$neighbourhood_cleansed)
train = train %>%
  mutate(level_nc = ifelse(n > 150, neighbourhood_cleansed, "other"))
train$neighbourhood_cleansed = as.factor(train$neighbourhood_cleansed)
train$level_nc = as.factor(train$level_nc)

test_nc = test %>%
  group_by(neighbourhood_cleansed = neighbourhood_cleansed) %>%
  summarize(n = n()) %>%
  arrange(desc(n))
```

Kaggle Report

Romauli Butarbutar 12 April 2021

```
test = merge(test, test_nc, by = c("neighbourhood_cleansed", "neighbourhood_c
leansed"))


test$neighbourhood_cleansed = as.character(test$neighbourhood_cleansed)

test = test %>%

  mutate(level_nc = ifelse(n > 30, neighbourhood_cleansed, "other"))

test$level_nc = as.factor(test$level_nc)


## create price_mean_c for score_data

train2 = data.frame(neighbourhood_cleansed = train_nc$neighbourhood_cleansed,

                    meanPrice = train_nc$meanPrice)


test = merge(test, train2, by = c("neighbourhood_cleansed", "neighbourhood_cl
eansed"), all.x = TRUE)


## find NA value

#sum(is.na(test$meanPrice))

test[is.na(test$meanPrice),]$meanPrice = mean(test$meanPrice, na.rm = TRUE)


# neighbourhood_group_cleansed: multiple factoral levels

train3 = train %>%

  group_by(neighbourhood_group_cleansed = neighbourhood_group_cleansed) %>%

  summarize(meanPriceGC = mean(price))


train = merge(train, train3, by = c("neighbourhood_group_cleansed", "neighbou
rhood_group_cleansed"))

test = merge(test, train3, by = c("neighbourhood_group_cleansed", "neighbourh
ood_group_cleansed"), all.x = TRUE)


## check NA value

sum(is.na(test$meanPriceGC))


# DATA MODELLING
```

Kaggle Report

Romauli Butarbutar 12 April 2021

```
library(gbm)

set.seed(1708)


boostModelFinal = gbm(price ~ meanPrice+meanPriceGC + level_nc + bedrooms + r
oom_type + property_type + bathrooms + beds

                      + accommodates + cleaning_fee + monthly_price + securit
y_deposit + minimum_nights + maximum_nights  + neighbourhood_group_cleansed

                      + host_is_superhost + availability_30 + availability_60
+ availability_90 + availability_365

                      + review_scores_rating + number_of_reviews + last_revie
w_days + first_review_days + review_scores_cleanliness + review_scores_accura
cy

                      + wc_transit + wc_summary+ wc_description+ wc_host_abou
t + wc_neighborhood_overview #word count

                      + host_listings_count + host_since_days + reviews_per_m
onth + host_has_profile_pic

                      + extra_people + guests_included + cancellation_policy

                      + Airconditioning + Dryer + Elevator + Familykidfriendl
y + Freestreetparking #amenities

                      + Hairdryer + Iron + Oven + Refrigerator + Shampoo + Se
lfcheckin  #amenities

               ,data = train, distribution = "gaussian",

               n.trees = 30000,

               interaction.depth = 5,

               shrinkage = 0.005,

               n.minobsinnode = 5)


summary(boostModelFinal)


## predict train dataset

predboostModelFinal = predict(boostModelFinal, n.trees = 30000)

RMSE = sqrt(mean((predboostModelFinal-train$price)^2)); RMSE

# [1] 37.50427


## predict scoring dataset

predboostModelFinal_test = predict(boostModelFinal, n.trees = 30000, newdata
= test)
```

Kaggle Report

Romauli Butarbutar 12 April 2021

```
RMSE = sqrt(mean((predboostModelFinal_test-train$price)^2)); RMSE

# [1] 139.3448


#FILE SUBMISSION

submission <- data.frame(id = test$id, price = predboostModelFinal_test)

write.csv(x = submission, file = "Model Boost Final Predictions 2.csv", row.n
ames = FALSE)


subMix = read.csv('Model Boost Final Predictions 2.csv')

sum(is.na(subMix))
```