**Kaggle Report – AirBnb Dataset (Sushmitha Mudda)**

The Kaggle competition for predicting AirBnB data was more than just coding. It was 80% research, 10% strategy, and the rest 10% of it was coding. Given more time and a better understanding of other prediction models, there’s surely potential for a better model.

Initially, there were 90 variables (excluding price) related to the property, host, and reviews for over 35,000 Airbnb rentals in New York. The problem assigned to us was to construct a model using the dataset given and use it to predict the price of a set of Airbnb rentals.

I spent the first couple of days just reading and understanding the data. I tried to find ways around cleaning all the data together, choosing random variables (based on my own opinion of what might be relevant) for the regression model. Although my RMSE score did get lower (82.58), the data was messy and it wasn’t cleaned properly.

I then put down all these variables in a word document, and went through each one. I first filtered out variables that were surely not relevant. Apart from collapsing levels of a factor, transforming non-numeric variables to numeric format, and imputing missing values, I started researching more on how we could use variables such as name, summary, notes and amenities to predict price. I started reading up on some other blogs on how people went about the Airbnb problem, and came across blogs like this(<https://towardsdatascience.com/predicting-airbnb-prices-with-deep-learning-part-1-how-to-clean-up-airbnb-data-a5d58e299f6c> ) which suggested splitting the amenities column into multiple Boolean dummy variables. For variables such as name and space, I used nchar to convert them into the number of characters. For variables like host\_location, I noticed that a good number of hosts didn’t live in New York. I started wondering if that would be a good predictor to determine price, as hosts who didn’t live close to their Airbnb, might have chosen to price it lower.

This led me to the categorical variables, I spent a lot of time figuring out how to visualize them and reduce their levels. I wanted to reduce the number of levels by condensing them into a small number of variables, as the drop levels function wasn’t working out well in the beginning. After a good amount of time on stack overflow, I came across creating a function for combining levels with low frequency count (<https://stackoverflow.com/questions/34385340/combining-low-frequency-counts/34385807#34385807>). After modifying the code, I then created a Condense function to reduce the number of levels by assigning a threshold, where if the values were beneath the threshold, then they fell under the assigned newName.

Although, I wanted to do more with variables such as zip code and neighbourhood\_cleansed, I worked on them towards the last couple of days, so, I didn’t have the time. I cleaned zip code to replace longer and incorrect zip codes into the right one, and assigned the zip codes with NA a zip code in Manhattan, as most of the missing zip codes were Manhattan-based.

When it came to modelling the data, I must say, I played around a lot with regression and trees. Initially, I started with linear regression and it was harder for me to work with as it took me a lot of time picking my clean variables and sub setting them individually towards the end, I wanted a clean dataset in which the model could run(price~.) with all the chosen variables. I tried forward selection and PCA for data reduction. This was turning out to be impossible with the huge dataset that I had cleaned, also because I wanted to include all the amenities variables, so until the last week, I was working with picking and selecting variables. I started reading up on boosting models during the same time and was considering using gbm, and in the last class, Professor Lala mentioned h2o, which worked out best for me, as manually tuning the gbm model started giving me varied scores and also, discarding irrelevant variables (e.g. is\_business\_travel\_ready).

There was one thing mentioned in class about time. I know with more time, cleaning my data further and testing out other models, I could have done a better job in reducing my RMSE. But I must say, I learnt a lot during the past couple of weeks, mostly with creating functions, and exploring different prediction models.