ASSIGNMENT 1

PREDICTIVE ANALYTICS

Classification and Regression

Ervin Liu (s10036078)

HR ANALYTICS & AIRBNB SINGAPORE

1. Introduction
2. Methodology
3. Results
4. Observations

INTRODUCTION

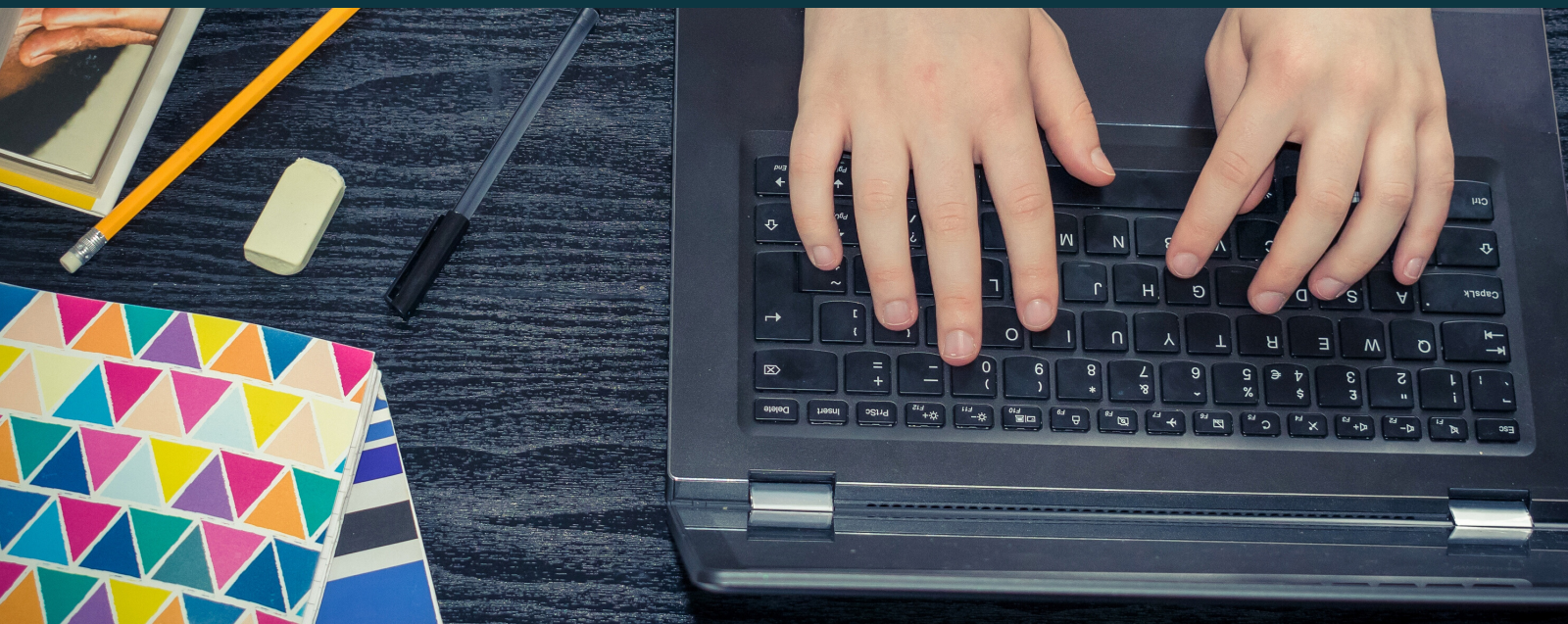
1.1 Business Case Scenario - HR Analytics

The HR Analytics team wants to evaluate the promotion likelihood of employees

- What **factors lead** to an employee's promotion?
- Which **employees** are likely to be promoted?

1.2 Target Audience

- HR Department
- Employee Relations team



INTRODUCTION

1.3 Dataset Overview

Target Variable

1. is_promoted

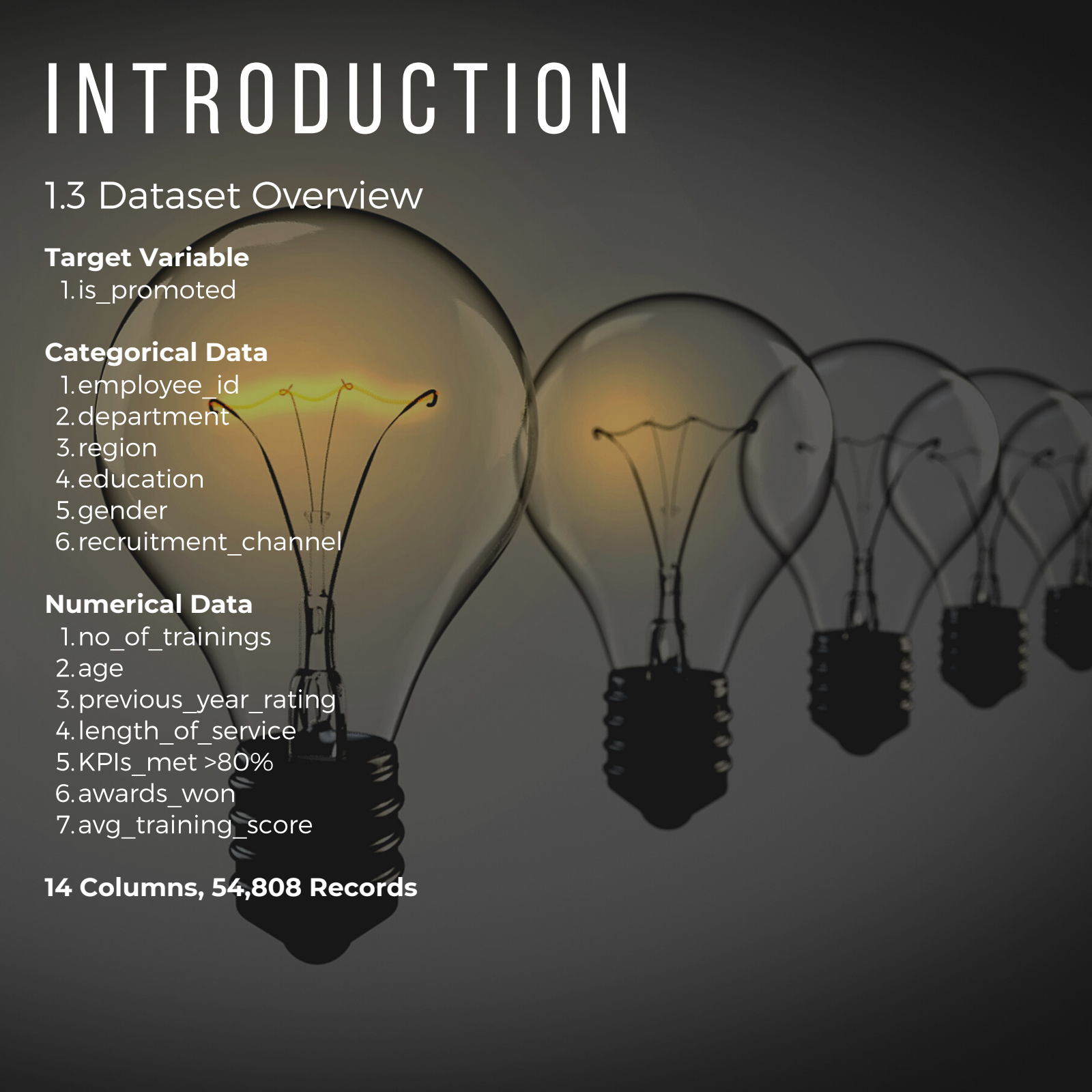
Categorical Data

1. employee_id
2. department
3. region
4. education
5. gender
6. recruitment_channel

Numerical Data

1. no_of_trainings
2. age
3. previous_year_rating
4. length_of_service
5. KPIs_met >80%
6. awards_won
7. avg_training_score

14 Columns, 54,808 Records



METHODOLOGY

2.1 Load Dataframe

Load and check the balance of records by target variable to observe how skewed or evenly split they are.

2.2 Cleanse Data

Remove unnecessary columns and null values

2.3 Stratified Sampling

Conduct a randomized stratified sampling to ensure an even representation of data from both promotees and non-promotees without undue skewing of dataset

2.4 Transform Data

Encode categorical data and normalize dataset in preparation for analysis

2.5 Correlation Analysis

Correlational analysis of independent variables in the dataset through the use of tables, barplots, heatmaps and scatter matrices

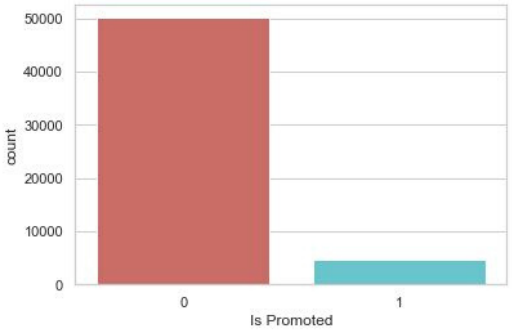
RESULTS

3.1 Load Dataframe

Load and check HR Analytics dataframe

	employee_id	department	region	education	gender	recruitment_channel	no_of_trainings	age	previous_year_rating	length_of_service	KPIs_me >80
0	65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	5.0	8	
1	65141	Operations	region_22	Bachelor's	m	other	1	30	5.0	4	
2	7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	3.0	7	
3	2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	1.0	10	
4	48945	Technology	region_26	Bachelor's	m	other	1	45	3.0	2	

Check balance of target variable (is_promoted)




```
# Express the total count of both classes of the target variable as percentages
count_notpromoted = len(df_hr[df_hr['is_promoted']==0])
count_promoted = len(df_hr[df_hr['is_promoted']==1])

pct_of_notpromoted = count_notpromoted/(count_notpromoted+count_promoted)
print("Percentage of non-promotees is", pct_of_notpromoted*100)

pct_of_promoted = count_promoted/(count_notpromoted+count_promoted)
print("Percentage of promotees is", pct_of_promoted*100)
```

```
Percentage of non-promotees is 91.48299518318494
Percentage of promotees is 8.517004816815064
```

Promotees account for only 8.52% of our dataset. And the ratio of promotees to non-promotees is approximately 9:91

RESULTS

3.2 Cleanse Data

Drop null values and non-useful columns - employee id and recruitment channel

```
df_hr = df_hr.dropna()
df_hr = df_hr.drop(columns=['employee_id','recruitment_channel'])
df_hr.head()
```

	department	region	education	gender	no_of_trainings	age	previous_year_rating	length_of_service	KPIs_met >80%	awards_won?	avg_training_score	is_pro
0	Sales & Marketing	region_7	Master's & above	f	1	35	5.0	8	1	0	49	
1	Operations	region_22	Bachelor's	m	1	30	5.0	4	0	0	60	
2	Sales & Marketing	region_19	Bachelor's	m	1	34	3.0	7	0	0	50	
3	Sales & Marketing	region_23	Bachelor's	m	2	39	1.0	10	0	0	50	
4	Technology	region_26	Bachelor's	m	1	45	3.0	2	0	0	73	

3.3 Stratified Sampling

As the ratio of promotees to non-promotees is 9:91, randomized stratified sampling was used to achieve a balanced dataset

```
# Concatenate the two datasets
df_hr_new=pd.concat([df_hr1,df_hr0_sample],axis=0)
df_hr_new['is_promoted'].value_counts()
```

```
1    4232
0    4232
Name: is_promoted, dtype: int64
```

```
# Get a statistical overview of the new stratified dataset
df_hr_new.describe()
```

	no_of_trainings	age	previous_year_rating	length_of_service	KPIs_met >80%	awards_won?	avg_training_score	is_promoted
count	8464.000000	8464.000000	8464.000000	8464.000000	8464.000000	8464.000000	8464.000000	8464.00000
mean	1.229088	35.276465	3.629962	6.186555	0.515714	0.065454	67.094518	0.50000
std	0.546848	7.211603	1.210701	4.045155	0.499783	0.247339	14.379556	0.50003
min	1.000000	20.000000	1.000000	1.000000	0.000000	0.000000	41.000000	0.00000
25%	1.000000	30.000000	3.000000	3.000000	0.000000	0.000000	55.000000	0.00000
50%	1.000000	34.000000	4.000000	5.000000	1.000000	0.000000	64.000000	0.50000
75%	1.000000	39.000000	5.000000	8.000000	1.000000	0.000000	80.000000	1.00000
max	6.000000	60.000000	5.000000	34.000000	1.000000	1.000000	99.000000	1.00000

RESULTS

3.4 Transform Data

Categorical data was encoded and the dataset normalized in preparation for analysis

	department	region	education	gender	no_of_trainings	age	previous_year_rating	length_of_service	KPIs_met >80%	awards_won?	avg_training_score	is_promot
0	7	7	2	0	1	35	5.0	8	1	0	49	
1	4	22	1	1	1	30	5.0	4	0	0	60	
2	7	19	1	1	1	34	3.0	7	0	0	50	
3	7	23	1	1	2	39	1.0	10	0	0	50	
4	8	26	1	1	1	45	3.0	2	0	0	73	

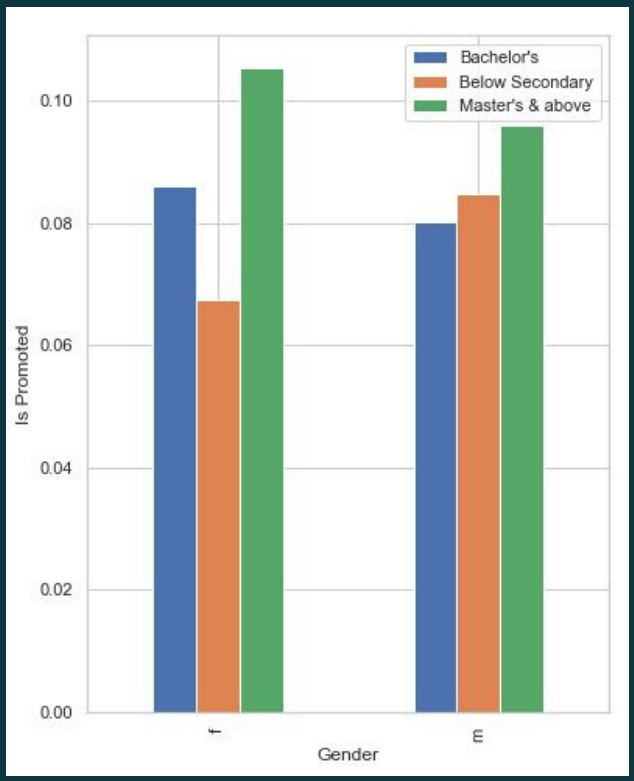
3.5 Correlation Analysis

Comparison of averages of independent numerical variables

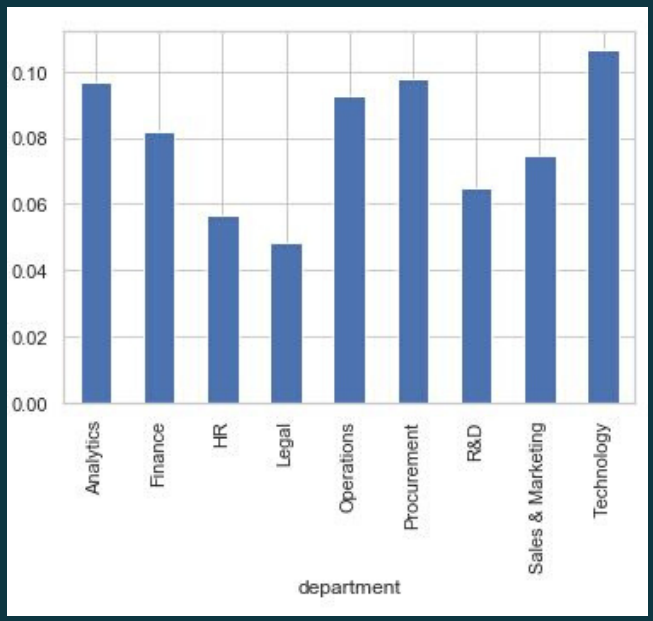
is_promoted	no_of_trainings	age	previous_year_rating	length_of_service	KPIs_met >80%	awards_won?	avg_training_score
0	1.256662	35.641555	3.275907	6.330085	0.323962	0.014000	62.867989
1	1.202977	35.042297	3.984405	6.117202	0.697779	0.119093	71.322779

RESULTS

3.5 Correlation Analysis



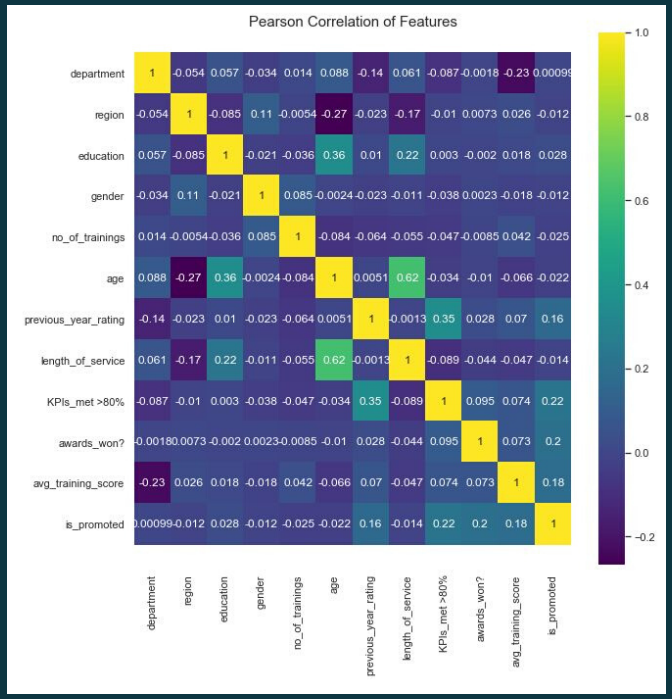
Gender vs is_promoted



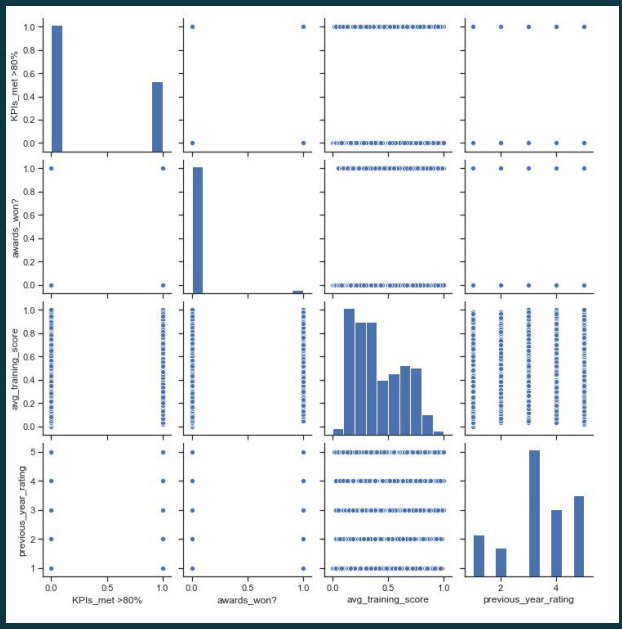
Department vs is_promoted

RESULTS

3.5 Correlation Analysis



Pearson Correlation Heatmap



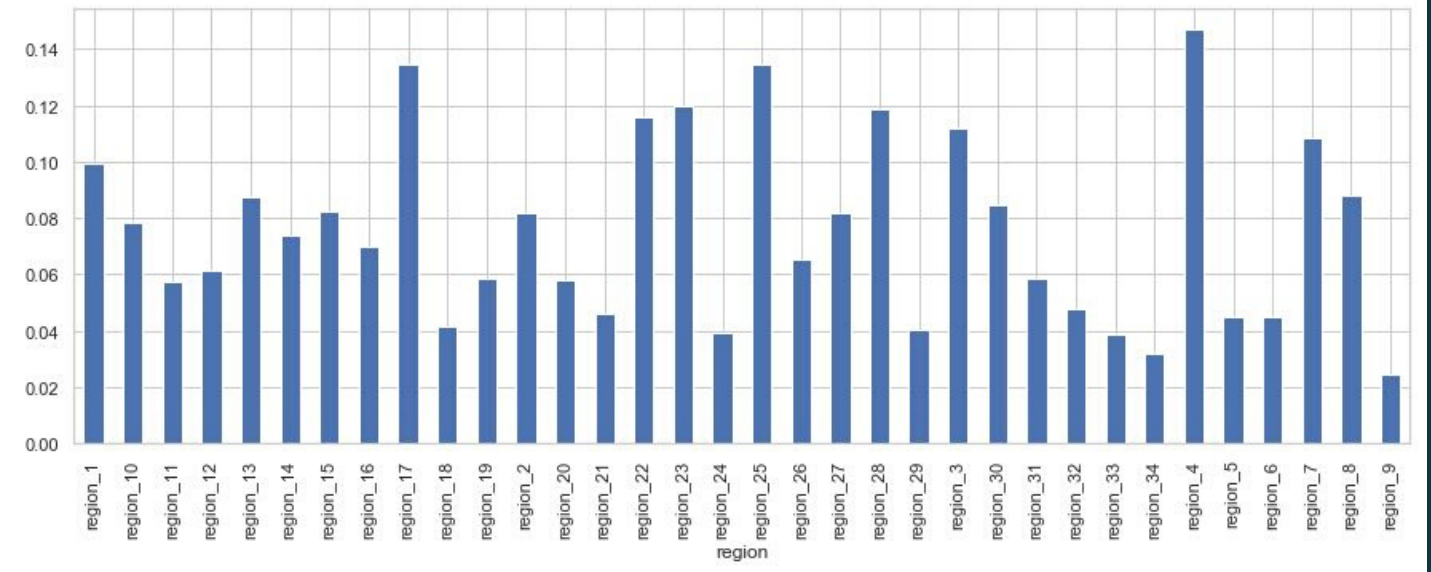
Scatter Matrices

RESULTS

3.5 Correlation Analysis

```
# Comparison of categorical data (Region) against target variable
df_hr.groupby('region').is_promoted.mean().plot.bar(figsize=(15,5))
```

```
<matplotlib.axes._subplots.AxesSubplot at 0xf2e8b5bc08>
```



Promotees vs Region

OBSERVATIONS

1. Promoted employees had a higher average of:
    - previous year's rating
    - meeting >80% of their KPIs
    - awards won
    - training scores

2. Interestingly, the average age, number of trainings received, and length of service of promotees were lower than those of non-promotees.

3. Promotees were quite evenly split between both genders and employees with a Master's Degree & Above formed the largest subgroup for both.

4. On average, the highest number of promotees came from the Technology department, followed by Analytics and Procurement.

5. On average, most promotees came from Region 4, Region 17 and Region 25.

6. The top three variables that show a positive correlation with the target variable are - KPIs>80%, awards won and average training scores

INTRODUCTION

1.1 Business Case Scenario - Airbnb Singapore

The Airbnb Singapore Analytics team is exploring factors that affect rental listing prices.

- What **factors strongly contribute** to an rental listing price?

1.2 Target Audience

- Airbnb Singapore - Business Development and Host Relations teams
- Rental Hosts

INTRODUCTION

1.3 Dataset Overview

Target Variable

1. price

Categorical Data

1. id
2. name
3. host_id
4. host_name
5. neighbourhood_group
6. neighbourhood
7. room_type

Numerical Data

1. latitude
2. longitude
3. minimum_nights
4. number_of_reviews
5. last_review
6. reviews_per_month
7. calculated_host_listings_count
8. availability_365

16 Columns, 7,907 Records

METHODOLOGY

2.1 Load Dataframe

Load and check the data types and shape

2.2 Cleanse Data

Remove unnecessary columns and replace null values of 'reviews_per_month' column with column mean.

2.3 Remove outliers

Initial exploration of the data revealed an unsually high ceiling for some columns compared their mean and upper quartile. Examine and remove outliers to avoid skewing of data when training our predictive model.

2.4 Transform Data

Encode categorical data

2.5 Correlation Analysis

Correlational analysis of independent variables in the dataset through the use of histograms, scatterplots, heatmaps, scatter matrices.

RESULTS

3.1 Load Dataframe

Load and check Airbnb Singapore dataframe

```
# Check dataframe import
df_airbnb.head()
```

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_review
0	49091	COZICOMFORT LONG TERM STAY ROOM 2	266763	Francesca	North Region	Woodlands	1.44255	103.79580	Private room	83	180	
1	50646	Pleasant Room along Bukit Timah	227796	Sujatha	Central Region	Bukit Timah	1.33235	103.78521	Private room	81	90	1
2	56334	COZICOMFORT	266763	Francesca	North Region	Woodlands	1.44246	103.79667	Private room	69	6	2
3	71609	Ensuite Room (Room 1 & 2) near EXPO	367042	Belinda	East Region	Tampines	1.34541	103.95712	Private room	206	1	1
4	71896	B&B Room 1 near Airport & EXPO	367042	Belinda	East Region	Tampines	1.34567	103.95963	Private room	94	1	2

3.2 Cleanse Data

Remove unnecessary columns and replace null values of 'reviews_per_month' column with column mean.

```
# Fillna for 'reviews_per_month' with column mean
avg_reviews = df_airbnb['reviews_per_month'].mean(axis=0)
df_airbnb['reviews_per_month'].replace(np.nan, avg_reviews, inplace=True)

# Drop unnecessary columns
df_airbnb = df_airbnb.drop(columns=['id','name','host_id','host_name','last_review','calculated_host_listings_count'])
df_airbnb.head(10)
```

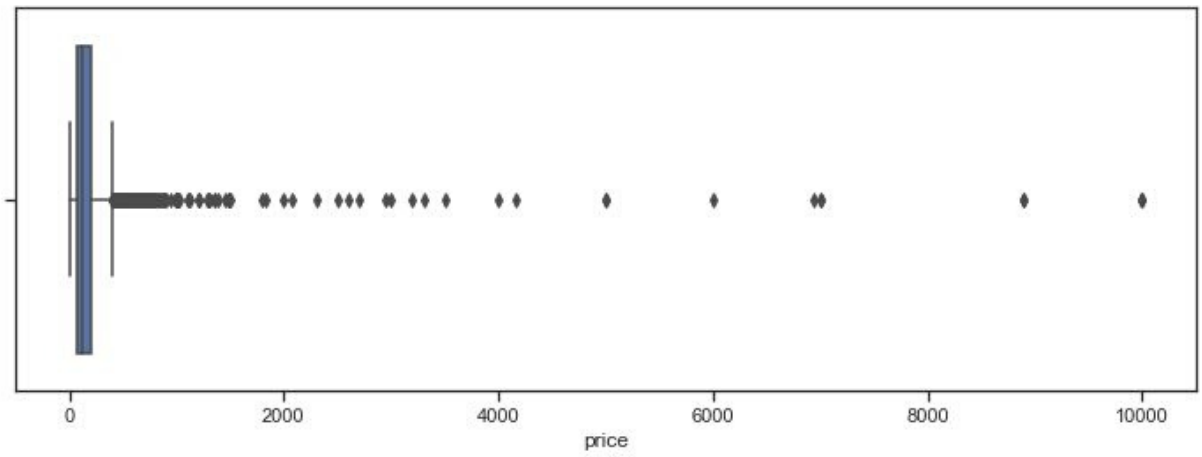
	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_reviews	reviews_per_month	availability_365
0	North Region	Woodlands	1.44255	103.79580	Private room	83	180	1	0.01	365
1	Central Region	Bukit Timah	1.33235	103.78521	Private room	81	90	18	0.28	365
2	North Region	Woodlands	1.44246	103.79667	Private room	69	6	20	0.20	365
3	East Region	Tampines	1.34541	103.95712	Private room	206	1	14	0.15	353
4	East Region	Tampines	1.34567	103.95963	Private room	94	1	22	0.22	355

RESULTS

3.3 Remove outliers

Initial exploration of the data revealed an unsually high ceiling for some columns compared to their mean and upper quartile, including the target variable 'price'

```
# Use a box plot for outliers in the target variable (price)
plt.figure(figsize=(12,4))
sns.boxplot(df_airbnb['price'])
plt.show()
```



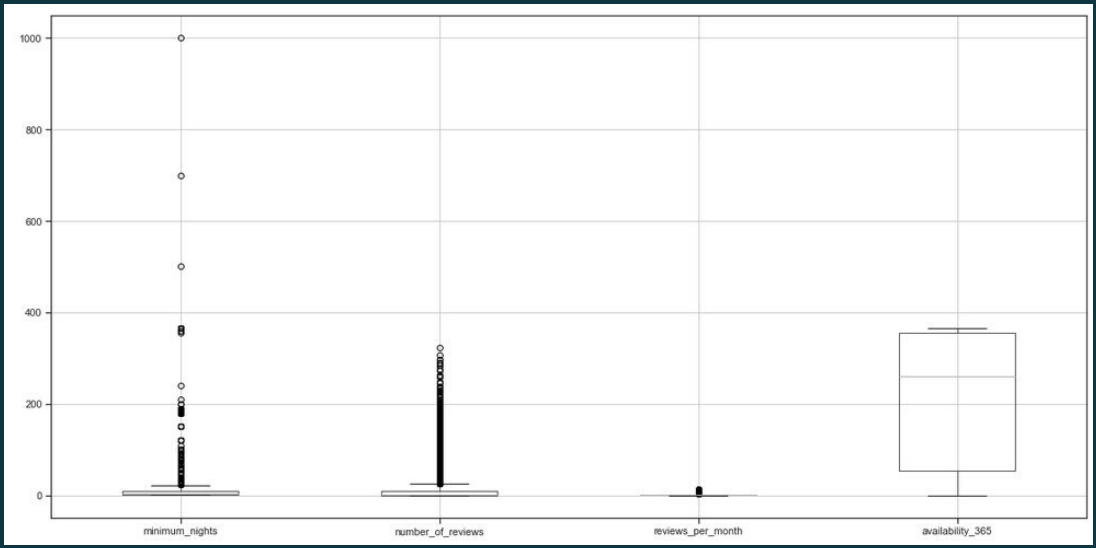

```
df_airbnb.groupby(['neighbourhood_group','room_type']).price.mean().unstack(0)
```

neighbourhood_group	Central Region	East Region	North Region	North-East Region	West Region
room_type					
Entire home/apt	224.581862	209.358779	188.433962	191.671875	334.178082
Private room	114.380117	117.234973	81.992958	79.878676	117.825397
Shared room	59.198276	187.090909	107.666667	55.000000	106.125000

RESULTS

3.3 Remove outliers

Outliers were also discovered in the 'minimum_nights', 'number_of_reviews' and 'reviews_per_month' columns.



The dataset IQR range was calculated and the outliers removed, resulting in an approximately 3,700-row (~47%) reduction of the dataset

```
# Remove the outliers and assign the new values to the df_airbnb dataframe
df_airbnb = df_airbnb[~((df_airbnb < (Q1 - 1.5 * IQR)) |(df_airbnb > (Q3 + 1.5 * IQR))).any(axis=1)]
df_airbnb.describe()
```

	latitude	longitude	price	minimum_nights	number_of_reviews	reviews_per_month	availability_365
count	4131.000000	4131.000000	4131.000000	4131.00000	4131.000000	4131.000000	4131.000000
mean	1.306059	103.855470	147.312031	4.57008	3.699104	0.718112	206.270879
std	0.016562	0.026769	87.281739	4.95262	5.647622	0.459606	151.437176
min	1.256390	103.781130	0.000000	1.00000	0.000000	0.020000	0.000000
25%	1.294670	103.840730	79.000000	1.00000	0.000000	0.250000	35.000000
50%	1.309550	103.851960	132.000000	3.00000	1.000000	0.940000	264.000000
75%	1.315560	103.871700	200.000000	6.00000	5.000000	1.043669	356.000000
max	1.361580	103.926440	400.000000	22.00000	25.000000	2.090000	365.000000

RESULTS

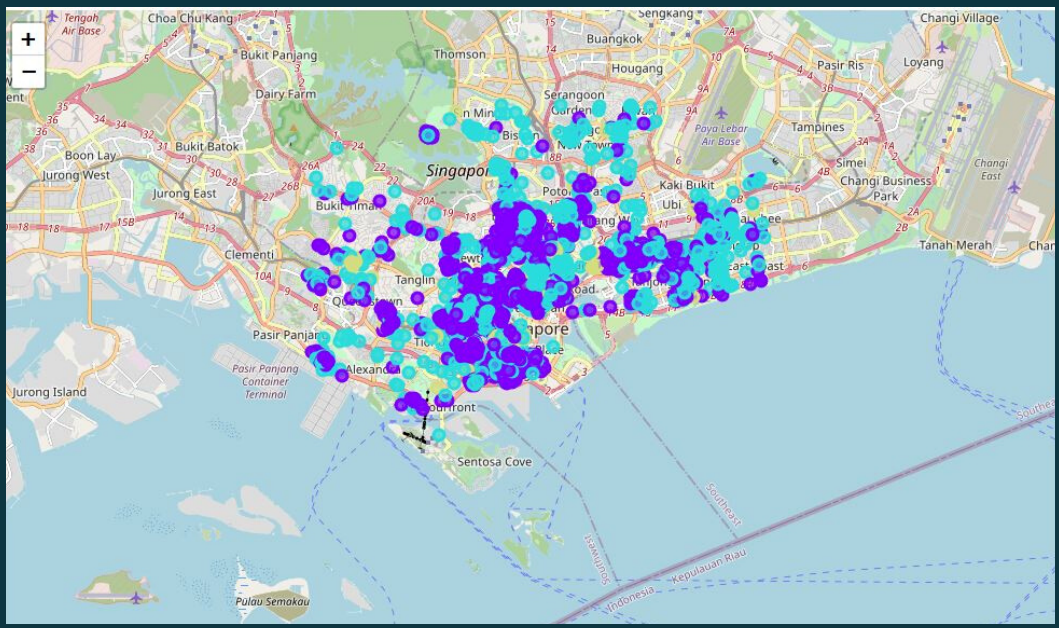
3.4 Transform Data

Categorical data was encoded in preparation for analysis

```
# Encode Neighbourhoods
df_airbnb['neighbourhood'] = df_airbnb['neighbourhood'].map({'Ang Mo Kio': 1, 'Bedok': 2, 'Bishan': 3, 'Bukit Batok': 4,
                                                             'Bukit Merah': 5, 'Bukit Panjang': 6, 'Bukit Timah': 7, 'Cen
                                                             'Choa Chu Kang': 9, 'Clementi': 10, 'Downtown Core': 11, 'Ge
                                                             'Hougang': 13, 'Jurong East': 14, 'Jurong West': 15, 'Kallan
                                                             'Lim Chu Kang': 17, 'Mandai': 18, 'Marina South': 19, 'Marin
                                                             'Museum': 21, 'Newton': 22, 'Novena': 23, 'Orchard': 24, 'Ou
                                                             'Punggol': 27, 'Queenstown': 28, 'River Valley': 29, 'Rochor
                                                             'Sembawang': 31, 'Sengkang': 32, 'Serangoon': 33, 'Singapore
                                                             'Southern Islands': 35, 'Sungei Kadut': 36, 'Tampines': 37,
                                                             'Toa Payoh': 39, 'Tuas': 40, 'Western Water Catchment': 41,
                                                             'Yishun':43
                                                             } ).astype(int)
```

```
# Encode Neighbourhood Groups
df_airbnb['neighbourhood_group'] = df_airbnb['neighbourhood_group'].map({'Central Region': 1,'East Region': 2,
                                                                         'North Region': 3, 'North-East Region': 4
                                                                        } ).astype(int)
```

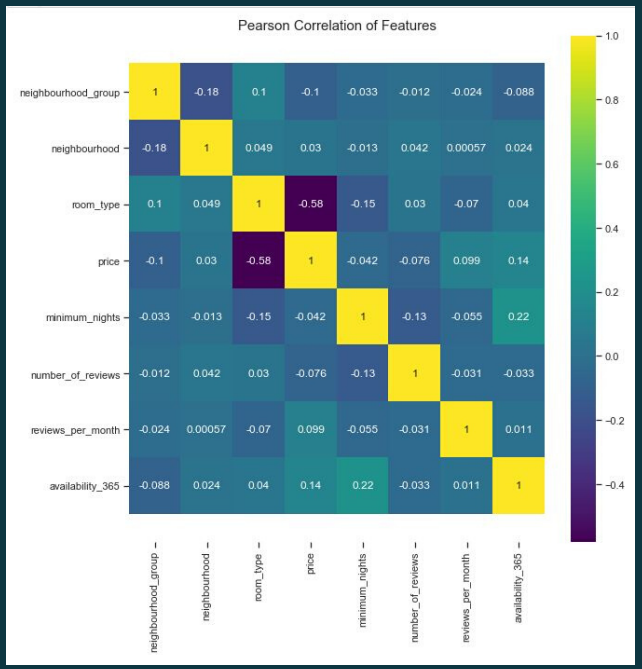
Geographical representation of reduced dataset using **Folium**.



Different colours represent different room types. Majority of listings belong to Entire Home and Private Room rentals

RESULTS

3.5 Correlation Analysis

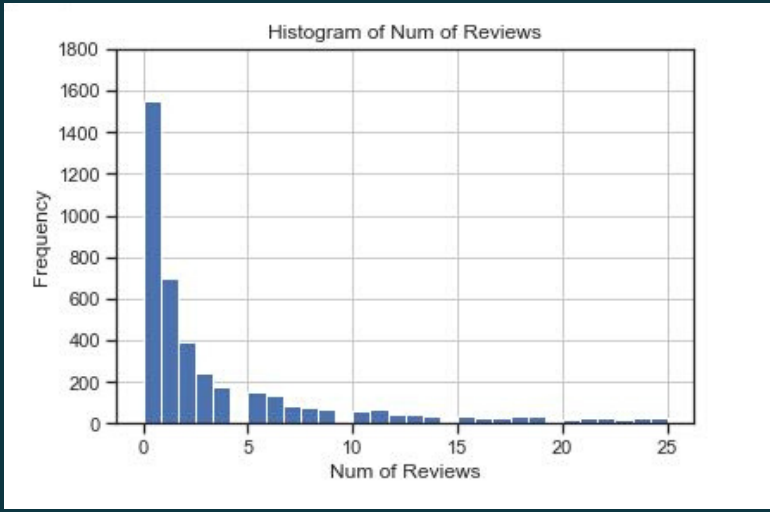


Pearson Correlation Heatmap

Scatter Matrices

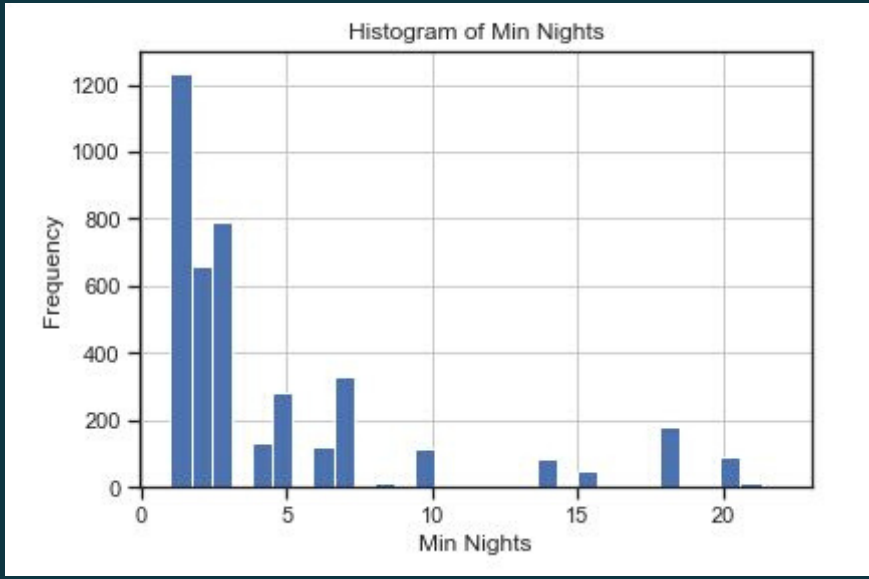
RESULTS

3.5 Correlation Analysis



number_of_reviews histogram

minimum_nights histogram



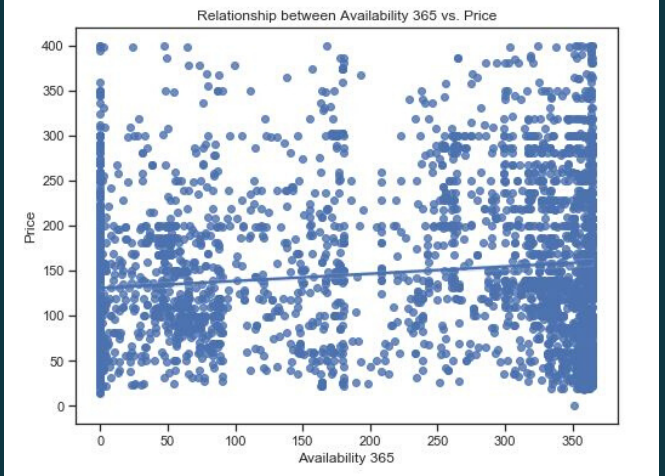
RESULTS

3.5 Correlation Analysis

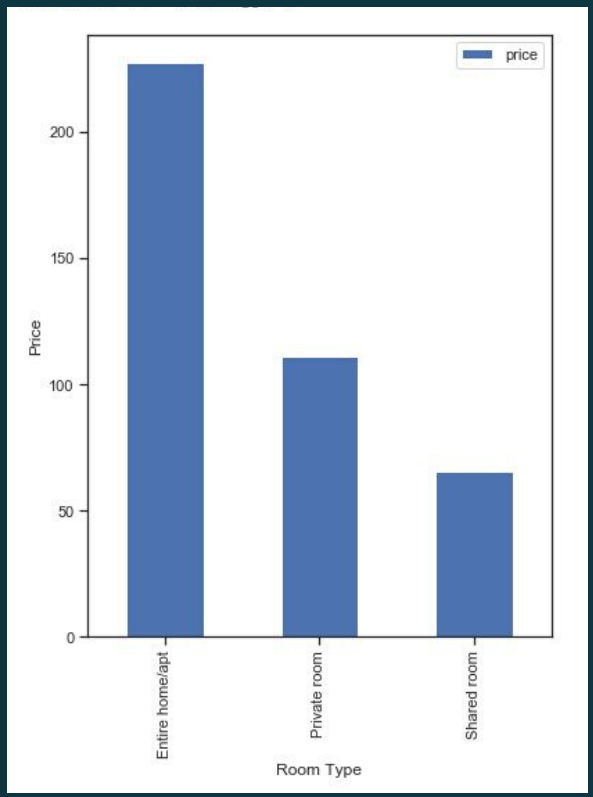
```
# Availability 365 vs Price
plt.figure(figsize=(8,6))
sns.regplot(x='availability_365', y='price', data=df_airbnb)

plt.title('Relationship between Availability 365 vs. Price')
plt.xlabel('Availability 365')
plt.ylabel('Price')
```

```
Text(0, 0.5, 'Price')
```



availability_365 vs price



room type vs price

OBSERVATIONS

1. Room types are strongly correlated with the target variable, while neighbourhood group and availability 365 show a slight correlation.

2. Room type average prices reveal a distinct differences between each, with Entire Home rentals holding the highest price - about twice that of a private room and four times that of a shared room.

3. Pricing of room types vary according to the different neighbourhood groups. This information could be used to market different listings to different customers based on location and price, e.g. Entire Homes in the Central region for expatriates, Private/Shared rooms in the West Region for students, etc.

4. A large proportion of our dataset listings posted < 5 minimum night stay threshold and hold < 5 guest reviews.

5. Although Availability 365 counts among the top three variables with a correlation to the target variable, a scatter plot does not indicate any kind of linear relationship.

6. Rental review ratings is an important info gap here. This could prove to have a strong correlation to price and help improve the accuracy of our model.