

**School of InfoComm Technology**

**Machine Learning**

Diploma in Data Science (DS)

Diploma in Information Technology (IT)

October 2023 Semester

**INDIVIDUAL ASSIGNMENT 1**

(30% of Machine Learning Module)

**Deadline for Submission:**

**10th Dec 2023 (Sunday), 2359 Hours**

|  |  |  |
| --- | --- | --- |
| Student Name | : | Isaac Khoo |
| Student Number | : | S10244252C |
| Video Presentation Link | : |  |

**Penalty for late submission:**

10% of the marks will be deducted every day after the deadline.

**NO** submission will be accepted after 17th Dec 2023, 23:59.

Contents

[Summary/Overview 3](#_Toc153136678)

[HR Analytics 4](#_Toc153136679)

[Problem Understanding 4](#_Toc153136680)

[Data Exploration 4](#_Toc153136681)

[Cleanse and Transform the data 16](#_Toc153136682)

[Missing Values 17](#_Toc153136683)

[Outliers 18](#_Toc153136684)

[Numerical Transformation 19](#_Toc153136685)

[Encoding Category Data Columns 23](#_Toc153136686)

[Scaling 26](#_Toc153136687)

[Correlation Analysis 26](#_Toc153136688)

[Export the data 28](#_Toc153136689)

[Summary and Further improvements 28](#_Toc153136690)

[Summary 28](#_Toc153136691)

[Further Improvements 29](#_Toc153136692)

[Airbnb 30](#_Toc153136693)

[Problem Understanding 30](#_Toc153136694)

[Data Exploration 30](#_Toc153136695)

[Clean and Transform the Data 39](#_Toc153136696)

[Missing Values 39](#_Toc153136697)

[Outliers 40](#_Toc153136698)

[Numerical Transformation 42](#_Toc153136699)

[Encode Categorical Data 49](#_Toc153136700)

[Scaling the Data 50](#_Toc153136701)

[Correlation Analysis 52](#_Toc153136702)

[Export the Data 53](#_Toc153136703)

[Summary and Further Improvements 53](#_Toc153136704)

[Summary 53](#_Toc153136705)

[Further Improvements 54](#_Toc153136706)

# Summary/Overview

In this report, we delve into the analysis and exploration of two distinct datasets to gain a comprehensive understanding of the underlying data structures. Our primary objective is to prepare the data for subsequent machine learning modeling. This process involves meticulous data preparation, thorough exploration, and analysis, employing a combination of visualization techniques and statistical approaches.

Dataset A: **HR Analytics**

* Description: This dataset contains employee personal information, education background, past performance etc.
* Features: employee\_id, department, region, education, gender, recruitment\_channel, no\_of\_trainings, age, previous\_year\_rating, length\_of\_service, KPIs\_met >80%, awards\_won?, avg\_training\_score and is\_promoted
* Size: 54807 rows, 14 columns

Dataset B: **Airbnb**

* Description: This dataset describes the listing activity and metrics from year 2013 to 2019. The data file includes the hosts’ information, the condition of listed properties, the reviews etc.
* Features: id, name, host\_id, host\_name, neighbourhood\_group, neighbourhood, latitude, longitude, room\_type, price, minimum\_nights, number\_of\_reviews, last\_review, reviews\_per\_month, calculated\_host\_listings\_count, availability\_365
* Size: 7907 Rows, 16 Columns

I will be performing 3 methodologies on the 2 data sets to be able to prepare the data for subsequent machine learning modeling. The 3 methodologies are:

1. Loading and Exploring the Data:

* Loading: Loading the Dataset
* Visualizations: Utilize charts, graphs, and plots to explore the distribution of data and relationships between variables.
* Statistical Approaches: Employ descriptive statistics and hypothesis testing to derive insights into the dataset.

1. Cleansing and transforming the data:

* Cleaning: Identify and handle missing values, outliers, or inconsistencies.
* Feature Engineering: Create new features or transform existing ones to enhance model performance.
* Encoding: Convert categorical variables into a format suitable for machine learning models.
* Scaling: If necessary, standardize or normalize numerical features to ensure uniformity.

1. Correlation Analysis

In this report I will be documenting my key findings from your exploratory analysis, highlighting patterns, correlations, or anomalies that may impact the machine learning model.

# HR Analytics

## Problem Understanding

The HR Analytics dataset presents the classification problem of predicting whether an employee will receive a promotion based on various personal and professional characteristics of the employee. The dataset includes information such as an employee's department, education level, gender, and past performance, as well as variables related to training and key performance indicators. By analyzing this data set, we hope to identify patterns and trends that can influence HR department promotion decisions and improve operational efficiency. Understanding and addressing these factors that influence promotion decisions will help HR departments make more informed and effective personnel decisions, which improves overall company results.

## Data Exploration

A screenshot of a computer

Description automatically generated

Let’s look at the data set and have a basic understanding of what we will be working with before we dive in deeper.

A screenshot of a computer

Description automatically generated

.info() shows a summary of the Data Frame. We can see that some of the columns contain null values. We can also see the data type of each column.

A screenshot of a computer

Description automatically generated

I want to display the columns with categorical data. From this I can see that region can be a numerical variable if we stripped 'region\_' and that all of these columns can be easily mapped/encoded if the cardinality of these columns is less (We will understand the cardinality later).

A screenshot of a computer

Description automatically generated

Now I’m displaying the columns with numerical data.

.describe() shows us the statistics of each numerical variable such as the 25th, 50th and 75th quantiles, the mean, standard deviation and the minimum and maximum value. Once we Compare these parameters, we can understand whether our features are in a similar scale or not.

Some Observations I noticed are:

1. There seem to be outliers in no\_of\_training as the mean value is 1.25 but the max value is 10.
2. Employees are in the age range 20-60 years old.
3. previous\_year\_rating is out of 5.
4. The range value of length\_of\_service is wide, it ranges from 1 to 37.
5. KPIs\_met > 80%, awards\_won? and is\_promoted are boolean column as it only has two values which are either 0 or 1.
6. avg\_training\_score ranges from 39 to 99.

It also seems that the columns are in a similar scale.

A screenshot of a computer

Description automatically generated

Now we will observe the cardinality of the variables. We can see that there is a total of 54808 in this dataset which are spread across 9 departments and 34 regions. There are only 3 Education Levels, and we have a few Binary/Boolean columns in this dataset.

**Now we will be using Visualizations to further understand the data.**

The visualizations will help us see and understand the correlations between the columns.

A blue circle with orange triangle and black text

Description automatically generated

This pie chart displays the chances of promotion. From this we can see that the chance of promotion is relatively low.

A graph of blue and orange bars

Description automatically generated

This column chart displays the number of employees in each department and how many of them are promoted and not promoted. From this we can see that there are more people in the Sales & Marketing and Operations departments. Followed by the Technology and Procurement departments, which have the same number of people.

A graph with a line

Description automatically generated

This line graph displays the Promotion rate of each department. From this we can see that the Technology department has the greatest promotion chances, and the Legal department has the lowest promotion chances.

A graph with a line going up

Description automatically generated

This line graph displays the Promotion rate of each region. From this graph we can see that the top 3 regions for promotion rate is region\_4, region\_17 and region\_25. This shows that the employees in these regions are working hard and are getting rewarded for it. However, Region\_9, Region\_34 and Region\_18 are the regions with the lowest promotion rate and Management should meet with employees from these regions to devise a new plan to increase productivity and improve employee happiness.

A graph of a number of employees

Description automatically generated

This graph shows the distribution of Ages for all employees. We can see that the 30-35 age group has the most employees. This tells us that employees choose and want to work for this organization even after obtaining a considerable amount of expertise in the field.

A graph of blue and orange bars

Description automatically generated

This stacked bar chart shows the chances of promotion for each age. From this we can see that the Promotions are pretty much evenly distributed among all ages and that this is a potential column to drop as it does not a driving factor of promotions.

A blue and orange pie chart

Description automatically generated

This pie chart shows the distribution of genders of the employees. From this pie chart, we can tell that there is a big difference in the number of male and female employees. Male employees make about 70% of the workforce. The management should take possible actions to close the gender gap and promote gender equality.

A blue and orange pie chart

Description automatically generated

These 2 pie charts show the promotion chances of each gender. Here we can see that promotions show no discernible gender bias. It is a potential column to drop, since it is not a factor of promotion.

A graph of blue and orange bars

Description automatically generated

This column chart shows the chances of promotion based off gender and their Education Level. From this we can see that Almost 2/3 of the employees holds a Bachelor's Degree and Employees with Master's & above have a higher chance of promotion. The rankings of Gender and Education for Promotions are as follows:

1. Female with Master's & above
2. Male with Master's & above
3. Male with Below Secondary
4. Female with Bachelor's
5. Male with Bachelor's
6. Female with Below Secondary

A pie chart with numbers and text

Description automatically generated

This chart shows the Distribution of Recruitment Channels for the employees. We can see that 55.6% of employees were hired through other means. 2.1% of workers were employed as result of a reference and 42.4% of employees were hired via sourcing. This indicates that a very small proportion of people are referred.

A graph of a bar chart

Description automatically generated with medium confidence

This chart shows the Chances of promotion based off the method they were recruited. From this we can see that even though the number of employees hired through referrals is the lowest, their promotion rate is the greatest. We can tell that employees only suggest individuals who are expected to be very good and do well at their jobs.

A graph with a line

Description automatically generated

This chart shows the Promotion Rate based off the Number of trainings the employees have been through. This chart shows that the Promotion rates decreases as the number of trainings completed increases. Promotions and the number of trainings completed may be negatively correlated.

A graph of different colored rectangular shapes

Description automatically generated

This column chart displays the chances of promotion based on the employees previous years rating. From this we can see that the higher the employees previous years rating, the higher the chances of promotion.

A graph with blue rectangles

Description automatically generated

This chart shows the chances of promotion based off if the employee has won awards before. From this we can see that employees who have won awards have a higher chance of being promoted than those who have not.

A graph with a line

Description automatically generated

This line graph displays the promotion rate based on how long the employee has worked for the company. From this we can notice that there are no trends. This may be a Potential column to drop, since it is not a factor of promotion.

A graph with blue rectangles

Description automatically generated

This column chart shows the promotion rate based off the employees KPIs Met which are greater than 80%. From this we can see that those employees with KPIs Met that is greater than 80%, have a very high chance of getting promoted compared to those who have lesser than 80% KPIs Met.

## Cleanse and Transform the data

A screenshot of a computer

Description automatically generated

First I started off by renaming the columns. I want to change the following column names:

KPIs\_met >80% to met\_KPIs

awards\_won? to won\_awards

### Missing Values

A screenshot of a computer program

Description automatically generated

Now I’ll be identifying the missing values and filling them in. We start off with identifying the columns with missing values. We can see that the column “education” and “previous\_year\_rating” have missing values. Firstly, we will start by filling in the missing values in the “education” column. If we impute the missing values in the education column as 'Bachelor's' simply because it is the most frequent value, the analysis will be incorrect. The column education indicates whether the employee has received a certain level of education, and we cannot just impute null values and assign an education level to an employee on our own. The employee may or may not have achieved the assigned level and this will result in inaccurate analysis. Therefore, we will fill the null values with 'unknown'. Now we will deal with the missing values in the “length of service” column. Just now, we identified that the values in the column length\_of\_service ranges from 1 to 37. This could explain why some employees do not have previous\_year\_rating as they have just joined the company and have yet to receive a rating as it’s their first year. To check this, we will check if those employees with null previous\_year\_rating has their length\_of\_service as 1. The 4th cell runs this check. After running the cell, we can see that all employees with null previous\_year\_rating joined the company this year as their length\_of\_service is all 1. Since they have yet to receive a rating, we will fill the null values with 0.

### Outliers

A screenshot of a graph

Description automatically generated

Now I want to check for outliers, so I plotted a boxplot for each numerical column.

From this we can see that all the numerical data are discrete variables. It also does not make sense to handle outliers in no\_of\_training, age and length\_of\_service because they are date time related variables, and outliers that are present in the dataset can be justified.

won\_awards and is\_promoted are boolean columns hence we should also not handle outliers.

Therefore, I will not handle any outliers.

### Numerical Transformation

A group of blue and white graphs

Description automatically generated

I plotted a histogram for each numerical column to be able to see the distribution of the data. From this I have decided to transform the columns “age”, “length\_of\_service” and “avg\_training\_score”.

A screenshot of a computer code

Description automatically generated

Firstly, I will start by preparing for the numerical transformation. I start by creating a copy of the data frame. Then I will create a function to see the variable distributions. A screenshot of a computer

Description automatically generated

Next, I will be performing Square root transformation, Cube root transformation, Log Transformation and Reciprocal transformation. As you can see new columns have been created with the transformed value of each variable.

A screenshot of a computer

Description automatically generated

Now I am comparing the original distribution for the column age, I think the reciprocal transformation works best for the column "age".

A screenshot of a computer

Description automatically generated

I am also comparing to the original distribution for the column length of service, I think the cube root transformation works best for the column "length\_of\_service".A screenshot of a computer screen

Description automatically generated

I will also compare the original distribution; I think either the reciprocal or log transformation works best for the column "avg\_training\_score" as they both produced similar results. But the transformed value of reciprocal method is more similar to the values of the other columns therefore I will choose to do the reciprocal method over the log method.

Based on these comparisons, I have concluded that we should do:

* reciprocal for "age" and "avg\_training\_score"
* cube root for "length\_of\_service".

A screenshot of a computer

Description automatically generated

Now I will be applying the transformation methods to the data frame.

### Encoding Category Data Columns

A screenshot of a computer program

Description automatically generated

We will be encoding these columns.

A screenshot of a computer

Description automatically generated

For department column, I used Mapping to encode the column.

A screenshot of a computer

Description automatically generated

For region column we can convert it from a categorical column to a numerical column by stripping the value 'region\_' that is Infront of every value. For instance, the record 'region\_7', once you remove 'region\_', it will be left with 7. We can then convert this column to int.

A screenshot of a computer

Description automatically generated

For education column, I used mapping to encode the column.

A screenshot of a computer

Description automatically generated

For gender column, I used encoding to map the column

A screenshot of a computer

Description automatically generated

For recruitment map column I used mapping to encode the column.

### Scaling

I will not be carrying out scaling as all the columns except employee\_id is in similar scale and we will be dropping employee\_id. Therefore, scaling is not needed.

## Correlation Analysis

A screenshot of a computer

Description automatically generated

A screenshot of a graph

Description automatically generated

A screenshot of a computer

Description automatically generated

I will be dropping the columns employee\_id, gender, age, recruitment\_channel, length\_of\_service and region as based on the data exploration and the correlation analysis, these columns have little to none correlation with our target variable 'is\_promoted'. The visualization analysis also revealed that some of these columns are not a factor of promotion.

## Export the data

A screenshot of a computer

Description automatically generated

Exported the data to a new CSV file and displayed the final data set.

## Summary and Further improvements

### Summary

To summarise what I have learned, discovered and done so far for this HR Dataset, I have:

Renamed Column Names

* KPIs\_met >80% to met\_KPIs
* awards\_won? to won\_awards

Missing Values

* education column: Replaced missing values with value 'unknown'
* previous\_year\_rating column: Filled null values with 0

Outliers

* Did not handle Outliers as those columns with outliers such as “age”, “no\_of\_training”, “length\_of\_service” are justifiable.

Numerical data transformation

* reciprocal transformation for columns 'age' and 'avg\_training\_score'
* cube root transformation for columns 'length\_of\_service'

Categorical Data Encoding

* I did mapping to encode the categorical data in columns department, education, gender, recruitment\_channel
* For region column, I replaced the value in the column to convert region column from a categorical column to a numberical column. For example, a record of 'region\_7', I removed 'region\_', and it will be left with 7. I then convert this column to int.

Scaling

* I did not perform Scaling as all columns are similar in scale.

Dropping of columns

* Dropped the columns employee\_id, Gender, age, recruitment\_channel, length\_of\_service, region as they had little to no correlation to our target variable 'is\_promoted’.

### Further Improvements

For further improvements I could have made to the dataset, It was hard to decide which method is better because we do not have a model result, therefore I chose the methods solely based on my knowledge on what each method does and how they suit my dataset.

Methods I have not managed to implement include:

* Handling Outliers
* One-Hot Encoding
* Label Encoding
* Scaling

# Airbnb

## Problem Understanding

The Airbnb dataset presents a regression problem in which we aim to predict the daily rental price of a listed property based on various characteristics of the property. The dataset includes information such as the location, type of room, number of reviews and availability, as well as the host's information and listing activity. In this analysis, we will be specifically focusing on the central region of the data in order to better understand the characteristics and trends of listings in this area. By analyzing the data for this region, we hope to identify patterns and trends that can inform hosts' pricing decisions and help them optimize their listings for the central region of the platform. When hosts establish fair and competitive prices for their homes, it can improve occupancy rates and platform performance overall. This can be achieved by understanding and addressing the issues that drive rental costs.

## Data Exploration

A screenshot of a computer

Description automatically generated

We will start by observing and looking at the data set so as to have a basic understanding before we dive in deeper.

A screenshot of a computer code

Description automatically generated

Next since for our problem understanding, we will only be considering data from the central region in the dataset.

A screenshot of a computer

Description automatically generated

We will remove listings that are not available for bookings.

A screenshot of a computer

Description automatically generated

I will be dropping the following columns as they are not useful factors to help us predict "price".

* id
* name
* host\_id
* host\_name
* neighbourhood\_group
* last\_review

A screenshot of a computer

Description automatically generated

.info() shows a summary of the Data Frame. We can tell that some of the columns contain null values. We can also see the data type of each column.

A screenshot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

We will want to be able to see which columns contain categorical data and numerical data.

.describe() will help show us the statistics of each numerical variable, e.g., the 25th, 50th and 75th quantiles, the mean, standard deviation and minimum and maximum value. Comparing these parameters we can quickly understand whether our features are in a similar scale. In this case, we can see that they are not on a similar scale.

For example, Price takes values 0-10000 whereas number\_of\_reviews take values 0-307, and latitude takes values 1.243870-1.365830.

A screenshot of a computer program

Description automatically generated

Next, I will be able to view the cardinality of the data. From what we can see, There are 20 Neighbourhoods in central region, latitude and longitude are expected to have high cardinality, there are only 3 room types, there are only 364 different prices despite having more than 6300 listing, this suggest that many listing have the same price and not all the listing is available all day, different listing has different availability and different minimum night requirement.

A graph of a number of numbers

Description automatically generated with medium confidence

This visualisation allows us to see the distribution of prices in the data and understand the range of prices and the frequency of listings at different price points. The visualisation shows the overall distribution of prices in the data, including the number of listings at different price points and the overall shape of the distribution. The distribution is skewed to the right, meaning that there are more listings with lower prices and fewer listings with higher prices. We can also see that most of the price is within the 50 to 220 range.

A graph of a number of people

Description automatically generated

This visualisation is a bar chart showing the average price of listings by neighbourhood. We can compare the average prices of listings in different neighbourhoods and see which neighbourhoods have higher or lower average prices.

We can see that, listings in the Bukit Timah and Marine Parade area have a lower price compared to listings in Orchard and Southern Island area.

A diagram of a distribution of a number of points

Description automatically generated with medium confidence

We are able to observe the correlation between price, latitude, and longitude in the data from this visualization. We can see that the listings at the bottom are more costly. The central region contains a larger cluster of postings.

A graph of a bar chart

Description automatically generated with medium confidenceWe may compare the average listing costs for various room kinds using this visualization to determine whether room types are typically more or less expensive. While a complete house or apartment often costs more, shared rooms are the most affordable.

A graph with a line

Description automatically generated

From this graph we can observe how the average cost of listings changes according to the minimum number of nights needed to reserve them thanks to this visualization. There isn't any particular pattern that indicates this isn't a significant element influencing listing prices.

A graph showing a number of blue lines

Description automatically generated

This graph allows us to see how the average price of listings varies based on the number of listings a host has. There is no specific trend which suggests that this is not a strong factor affecting the price of listings.

A graph showing a number of reviews

Description automatically generated

This visualization allows us to see how the average price of listings varies based on the number of reviews they have received. We can see that there is no specific trend which suggests that this is not a strong factor affecting the price of listings.

A graph showing a blue line

Description automatically generated

This visualization allows us to see how the average price of listings varies based on the number of reviews per month they receive. There is no specific trend which suggests that this is not a strong factor affecting the price of listings.

A graph with blue lines

Description automatically generated

This visualization allows us to see how the average price of listings varies based on the number of days they are available for booking. We can observe that there is no specific trend which suggests that this is not a strong factor affecting the price of listings.

## Clean and Transform the Data

### Missing Values

A screenshot of a computer

Description automatically generated A screenshot of a computer program

Description automatically generated

Firstly, we will start by handling the missing values. We will start by identifying the column with missing values. We can see that there are missing values in column “reviews\_per\_month”.

.describe() helps us notice that the minimum value for number\_of\_reviews is 0. This might be the reason for null values in reviews\_per\_month. To check this, I ran a code in the 3rd cell to check if all rows with null values in the ‘reviews\_per\_month’ have a value of 0 in the ‘number\_of\_reviews’ column. The result shows that we were right. With this confirmation, we can safely input the missing values in 'reviews\_per\_month' as 0. Lastly, we will check if there are still missing values, and we can see that there are no more missing values and all past missing values have been replaced.

### Outliers

A screenshot of a computer

Description automatically generated

To check for outliers, I plotted boxplot for each of the numerical columns. We can see that there are outliers present in all the columns except 'availability\_365'.

Some things to take not are:

* 'latitude' and 'longitude' should not be handled as the values in these columns represents a specific location, any alteration will make it lose its accuracy.
* we should not touch 'price' as it is our target variable. Therefore, I will not be handling the outliers in 'price' as well.
* 'calculated\_host\_listing', there are reasons to justify why there are outliers in this column. The host can be some sort of agency who help clients manage their listing on airbnb and therefore they have a lot of listing compared to other hosts.
* I will look further into 'minimum\_nights', 'number\_of\_reviews' and 'reviews\_per\_month' before deciding if I should handle outliers in these columns.

A screenshot of a computer

Description automatically generated

The host of these listings may be looking for long-term occupants which might be why the minimum\_nights value is listed higher compared to other listings which are meant for shorter stays. Hence, I will not be handling the outliers in this column as I feel like it makes sense to have outliers in minimum\_night column for listings that are meant for longer stays.

A screenshot of a computer

Description automatically generated

For the columns number\_of\_reviews and reviews\_per\_month, the 2 columns are correlated so we will be analysing them together.

A listing is more popular than the others if it has a higher number of reviews. Given that the listing activity and metrics in this dataset span the years 2013 through 2019, it makes sense that the "number\_of\_review" would be this high given the listing has been listed for a longer time. This particular listing has been around for 4.7years, having a total number of 307 reviews in this 4.7years is not absurd. With this in mind, the outlier values in number\_of\_review and reviews\_per\_month could be because the listing have been listed for some time already and have gathered more reviews compared to new listing.

Therefore, we will not be handling the outliers in these columns.

In conclusion, I will not be handling any outlier values as the outlier values makes sense to exist. It is not very justifiable for me to just remove the columns or replace the value as it can affect the accuracy of the data.

### Numerical Transformation

A group of blue and white graphs

Description automatically generated

I started by plotting histogram for each numerical column so we will be able to identify which column we want to perform transformation in. I have decided to perform transformation in the column’s minimum\_nights, number\_of\_reviews, reviews\_per\_month, calculated\_host\_listings\_count, availability\_365. A screenshot of a computer

Description automatically generated

Firstly we will start by creating a new copy of the dataframe. Next, I will be performing Square root transformation, Cube root transformation, Log Transformation and Reciprocal transformation. As you can see new columns have been created with the transformed value of each variable. This indicates that the transformations are done. A screenshot of a computer

Description automatically generated

Next we will be doing numerical transformation for minimum\_nights. However, after comparing to the original distribution, I think that none of the transformation makes it better so "minimum\_nights" should not be transformed as the transformed distribution is worse than the original distribution.

A screenshot of a computer

Description automatically generated

Next for number\_of\_reviews, after comparing to the original distribution, I think the log transformation works best for "number\_of\_reviews". However, because this column consist of value 0, we should do cube root transformation instead.

A screenshot of a computer

Description automatically generated

I also think that we should do cube transformation for similar reasons.

A screenshot of a graph

Description automatically generated

After comparing to the original distribution, cube root transformation seems to work best for "calculated\_host\_listings\_count".

A screenshot of a computer

Description automatically generated

After comparing the original distribution, I think the reciprocal transformation works best for "availability\_365".

In conclusion, Based on these comparisons, I have decided that we should do:

* cube root transformation for "number\_of\_reviews", "reviews\_per\_month" and "calculated\_host\_listings\_count"
* reciprocal transformation for "availability\_365"
* not transform "minimum\_nights"

Now, let's apply the transformation methods to their respective columns in the dataframe.

A screenshot of a computer

Description automatically generated

### Encode Categorical Data

A screenshot of a computer

Description automatically generated

We will be performing encoding on the columns neighbourhood and room\_type.

A screenshot of a computer

Description automatically generated

For neighbourhood column we will be performing Label encoding to encode this column.

A screenshot of a computer

Description automatically generated

For room\_type column we will be performing mapping to encode the column.

### Scaling the Data

A group of blue and white graphs

Description automatically generated

Firstly we start by displaying all the column distributions using a histogram. From what we can see, the MinMax scaler will suit our dataframe better as most of the variables do not follow a normal distribution.

A screenshot of a computer program

Description automatically generated

Time to start scaling the data. Firstly, we will start by dropping the the column “price” as we wouldn’t want to scale the target variable because it will affect the interpretability of the model as the prediction will no longer be in the original units of the target variable. Hence, we will drop it first. After which, we will start with the scaling of the data.

A comparison of a graph

Description automatically generated

Once done scaling we will compare the variable distributions of the data before and after scaling. From what we can see the scaling was successful.

A screenshot of a computer

Description automatically generated

Lastly, we will add the column “price” back as it is out target variable.

## Correlation Analysis

A screenshot of a computer

Description automatically generated

.corr() will help us display the correlations between the different variables.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Based on the data exploration and the correlation analysis, I have decided to drop the 'calculated\_host\_listing\_count' column as it has almost no relation to our target variable 'price'.

A white background with black text

Description automatically generated

Lastly I compared the Original Dataset and the final dataset to see the changes before and after filtering cleaning and transforming the data.

## Export the Data



Exported the data to a new CSV file and displayed the final data set.

## Summary and Further Improvements

### Summary

To summarise what I have learned, discovered and done so far for this Listing Dataset, I have:

* Filtered Rows

Filtered the original dataset to only contain Central Region Listings (neighbourhood\_group) and Listings that are available for booking (Available\_365)

* Dropped unnecessary columns

Dropped the columns, id, name, host\_id, neighbourhood\_group and last\_review

* Missing Values

Missing values in "reviews\_per\_month" column were filled Na with 0.

* Outliers

Outliers were spotted in the columns:

* + latitude
  + longitude
  + price
  + minimum\_nights
  + number\_of\_reviews
  + reviews\_per\_month
  + calculated\_host\_listings\_count

I chose not handle any outliers as outliers are justifiable in these columns.

* Numerical Data Transformation

I performed the following transformations:

1. Cube root transformation: "number\_of\_reviews", "reviews\_per\_month" and "calculated\_host\_listings\_count”.
2. Reciprocal transformation: "availability\_365"
3. Did not transform: "minimum\_nights”.

* Categorical Data Encoding

Performed on columns “neighbourhood” and “room\_type”.

Label encoder was used on column “neighbourhood”.

Mapping was used for column “room\_type”.

* Scaling

MinMax Scaling performed for all columns except "price" (which is target variable)

* Dropping of Columns

Dropped "calculated\_host\_listings\_count" column as it had almost no relation to our target variable "price".

### Further Improvements

Similarly, to the improvements made to the previous HR dataset. It was hard to decide which method is better because we do not have a model result, therefore I chose the methods solely based on my knowledge on what each method does and how they suit my dataset.

Methods I have not managed to implement include:

* Handle Outliers (Chose not to)
* One-Hot Encoding
* Target Mean Encoding
* Standard Scaling

It's important to note that I performed numerical transformation. Given how many machine learning algorithms are sensitive to the scale of input data, I believe this is a crucial step. However, I am not certain about my application of numerical transformation as I do not have a model result to validate that this step is necessary.