2440016804 - Rio Pramana - LA01 - Assignment 2

Import libraries & read downloaded dataset from https://www.kaggle.com/jojoker/singapore-airbnb

In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
# Importing the dataset, downloaded file is in the same folder
csv_path = "listings.csv"
listings_df = pd.read_csv(csv_path)
```

Checking the dataset

In [3]:
```
listings_df.shape
```

Out[3]:
```
(7907, 16)
```

In [4]:
```
listings_df.head(5)
```

Out[4]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_
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	
2	56334	COZICOMFORT	266763	Francesca	North Region	Woodlands	1.44246	103.79667	Private room	69	6	
3	71609	Ensuite Room (Room 1 & 2) near EXPO	367042	Belinda	East Region	Tampines	1.34541	103.95712	Private room	206	1	

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_
4	71896	B&B Room 1 near Airport & EXPO	367042	Belinda	East Region	Tampines	1.34567	103.95963	Private room	94	1	

In [5]:
```
listings_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7907 entries, 0 to 7906
Data columns (total 16 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   id                              7907 non-null   int64
 1   name                            7905 non-null   object
 2   host_id                         7907 non-null   int64
 3   host_name                       7907 non-null   object
 4   neighbourhood_group             7907 non-null   object
 5   neighbourhood                   7907 non-null   object
 6   latitude                        7907 non-null   float64
 7   longitude                       7907 non-null   float64
 8   room_type                       7907 non-null   object
 9   price                           7907 non-null   int64
 10  minimum_nights                  7907 non-null   int64
 11  number_of_reviews               7907 non-null   int64
 12  last_review                     5149 non-null   object
 13  reviews_per_month               5149 non-null   float64
 14  calculated_host_listings_count  7907 non-null   int64
 15  availability_365                7907 non-null   int64
dtypes: float64(3), int64(7), object(6)
memory usage: 988.5+ KB
```

1. Extracting independent variables and dependent variables

Berdasarkan deskripsi dataset dan deskripsi setiap kolom dataset pada https://docs.google.com/spreadsheets/d/1iWCNJcSutYqpULSQHlNyGInUvHg2BoUGoNRIGa6Szc4/edit#gid=982310896 (Data Dictionary dari source dataset), yang merupakan **dependent variable** adalah **price**, 14 kolom lainnya adalah **independent variables**

Untuk memudahkan extraction, kolom price yang berada di tengah-tengah dataset akan dipindahkan ke bagian paling akhir

In [6]:
```
listings_new = listings_df.copy()
cols_at_end = ['price']
listings_new = listings_new[[c for c in listings_new if c not in cols_at_end]
        + [c for c in cols_at_end if c in listings_new]]
listings_new.head(5)
```

Out[6]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	minimum_nights	number_of_reviews
0	49091	COZICOMFORT LONG TERM STAY ROOM 2	266763	Francesca	North Region	Woodlands	1.44255	103.79580	Private room	180	1
1	50646	Pleasant Room along Bukit Timah	227796	Sujatha	Central Region	Bukit Timah	1.33235	103.78521	Private room	90	18
2	56334	COZICOMFORT	266763	Francesca	North Region	Woodlands	1.44246	103.79667	Private room	6	20
3	71609	Ensuite Room (Room 1 & 2) near EXPO	367042	Belinda	East Region	Tampines	1.34541	103.95712	Private room	1	14
4	71896	B&B Room 1 near Airport & EXPO	367042	Belinda	East Region	Tampines	1.34567	103.95963	Private room	1	22

Extracting Independent Variables

In [7]:
```
#Extracting independent variables:
x = listings_new.iloc[:,:-1].values #Extract semua kolom kecuali kolom terakhir
print(x)
```

```
[[49091 'COZICOMFORT LONG TERM STAY ROOM 2' 266763 ... 0.01 2 365]
 [50646 'Pleasant Room along Bukit Timah' 227796 ... 0.28 1 365]
 [56334 'COZICOMFORT' 266763 ... 0.2 2 365]
 ...
 [38109336 '[ Farrer Park ] New City Fringe CBD Mins to MRT' 281448565
  ... nan 3 173]
 [38110493 'Cheap Master Room in Central of Singapore' 243835202 ... nan
```

```
 2 30]
[38112762 'Amazing room with private bathroom walk to Orchard' 28788520
 ... nan 7 365]]
```

Extracting Dependent Variable

In [8]:
```
#Extracting dependent variable:
y = listings_new.iloc[:,15].values #Extract kolom terakhir
print(y)
```

```
[83 81 69 ... 58 56 65]
```

2. handling missing data (Replacing missing data with the mean value)

Check which column(s) has missing data

In [9]:
```
listings_new.isnull().sum()
```

Out[9]:
```
id                                  0
name                                2
host_id                             0
host_name                           0
neighbourhood_group                 0
neighbourhood                       0
latitude                            0
longitude                           0
room_type                           0
minimum_nights                      0
number_of_reviews                   0
last_review                      2758
reviews_per_month                2758
calculated_host_listings_count      0
availability_365                    0
price                               0
dtype: int64
```

In [10]:
```
listings_new.head(5)
```

Out[10]:

id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	minimum_nights	number_of_reviews

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	minimum_nights	number_of_reviews
0	49091	COZICOMFORT LONG TERM STAY ROOM 2	266763	Francesca	North Region	Woodlands	1.44255	103.79580	Private room	180	1
1	50646	Pleasant Room along Bukit Timah	227796	Sujatha	Central Region	Bukit Timah	1.33235	103.78521	Private room	90	18
2	56334	COZICOMFORT	266763	Francesca	North Region	Woodlands	1.44246	103.79667	Private room	6	20
3	71609	Ensuite Room (Room 1 & 2) near EXPO	367042	Belinda	East Region	Tampines	1.34541	103.95712	Private room	1	14
4	71896	B&B Room 1 near Airport & EXPO	367042	Belinda	East Region	Tampines	1.34567	103.95963	Private room	1	22

Pada dataset, terdapat 3 kolom yang memiliki missing data, yaitu kolom **name**, **neighbourhood_group**, dan **room_type**.

Handle missing data on 'name' column

Untuk kolom name, kita menghandle missing data dengan mereplace missing data tersebut menggunakan mode dari kolom name karena kolom name berisi categorical data

In [11]:
```
listings_new.name.value_counts()
```

Out[11]:
```
Luxury hostel with in-cabin locker - Single mixed     13
Studio Apartment - Oakwood Premier                     9
Inviting & Cozy 1BR APT 3 mins from Tg Pagar MRT       9
Stylish 1BR Located 7 mins from Tg Pagar MRT           8
City-located 1BR loft apartment *BRAND NEW*            8
                                                      ..
Boonlay 16sqm Cosy Master Room for Rent                1
Tanjong Pagar Pristine Studio Apartment                1
lavLoftbed *RmT, no-sharing, wifi, mrt                 1
Newly furnished spacious room                          1
```

```
Amazing room with private bathroom walk to Orchard     1
Name: name, Length: 7457, dtype: int64
```

In [12]:
```
listings_new.name.mode()
```

Out[12]:
```
0    Luxury hostel with in-cabin locker - Single mixed
dtype: object
```

Mode dari kolom name adalah 'Luxury hostel with in-cabin locker - Single mixed', maka missing value pada kolom ini akan direplace dengan value tersebut. Untuk mengaksesnya, menggunakan [0] dibelakang mode

In [13]:
```
listings_new.name.mode()[0]
```

Out[13]:
```
'Luxury hostel with in-cabin locker - Single mixed'
```

In [14]:
```
listings_new['name'].fillna(listings_new['name'].mode()[0], inplace = True)
```

Mengecek apakah kolom name yang valuenya missing sudah diganti dengan modenya:

In [15]:
```
listings_new.name.value_counts()
```

Out[15]:
```
Luxury hostel with in-cabin locker - Single mixed     15
Inviting & Cozy 1BR APT 3 mins from Tg Pagar MRT       9
Studio Apartment - Oakwood Premier                     9
Superhost 1BR APT in the heart of Tg Pagar             8
Stylish 1BR Located 7 mins from Tg Pagar MRT           8
                                                      ..
Boonlay 16sqm Cosy Master Room for Rent                1
Tanjong Pagar Pristine Studio Apartment                1
lavLoftbed *RmT, no-sharing, wifi, mrt                 1
Newly furnished spacious room                          1
Amazing room with private bathroom walk to Orchard     1
Name: name, Length: 7457, dtype: int64
```

Mengecek apakah masih ada missing value pada kolom name:

In [16]:
```
listings_new.isnull().sum()
```

Out[16]:
```
id                                  0
```

```
name                                0
host_id                             0
host_name                           0
neighbourhood_group                 0
neighbourhood                       0
latitude                            0
longitude                           0
room_type                           0
minimum_nights                      0
number_of_reviews                   0
last_review                      2758
reviews_per_month                2758
calculated_host_listings_count      0
availability_365                    0
price                               0
dtype: int64
```

Kolom name sudah tidak ada missing value lagi dan direplace dengan value modenya

Handle missing data on 'last_review' column

Untuk kolom last_review, kita menghandle missing data dengan mereplace missing data tersebut menggunakan mode dari kolom last_review karena kolom last_review berisi categorical data

In [17]:
```
listings_new.last_review.value_counts()
```

Out[17]:
```
2019-08-12    152
2019-08-11    128
2019-08-13    110
2019-08-10     87
2019-08-08     78
             ...
2016-12-03      1
2016-01-18      1
2016-07-27      1
2017-08-19      1
2019-03-22      1
Name: last_review, Length: 1001, dtype: int64
```

Mode dari kolom last_review:

In [18]:
```
listings_new.last_review.mode()
```

Out[18]:
```
0    2019-08-12
dtype: object
```

Replace missing values:

In [19]:
```
listings_new['last_review'].fillna(listings_new['last_review'].mode()[0], inplace = True)
```

Mengecek apakah kolom last_review yang valuenya missing sudah diganti dengan modenya:

In [20]:
```
listings_new.last_review.value_counts()
```

Out[20]:
```
2019-08-12    2910
2019-08-11     128
2019-08-13     110
2019-08-10      87
2019-08-08      78
              ...
2016-12-03       1
2016-01-18       1
2016-07-27       1
2017-08-19       1
2019-03-22       1
Name: last_review, Length: 1001, dtype: int64
```

Mengecek apakah masih ada missing value pada kolom last_review:

In [21]:
```
listings_new.isnull().sum()
```

Out[21]:
```
id                                  0
name                                0
host_id                             0
host_name                           0
neighbourhood_group                 0
neighbourhood                       0
latitude                            0
longitude                           0
room_type                           0
minimum_nights                      0
number_of_reviews                   0
last_review                         0
reviews_per_month                2758
calculated_host_listings_count      0
```

```
availability_365                0
price                           0
dtype: int64
```

Kolom last_review sudah tidak ada missing value lagi dan direplace dengan value modenya

Handle missing data on 'reviews_per_month' column

Untuk kolom reviews_per_month, kita menghandle missing data dengan mereplace missing data tersebut menggunakan mode dari kolom reviews_per_month karena lebih optimal jika kita menggunakan reviews_per_month yang paling sering muncul untuk menghindari kemungkinan penurunan akurasi dalam jumlah yang besar

In [22]:
```
listings_new.reviews_per_month.value_counts()
```

Out[22]:
```
1.00    172
0.04    104
0.08     96
0.05     93
0.12     92
       ...
4.02      1
3.92      1
3.52      1
3.57      1
8.00      1
Name: reviews_per_month, Length: 527, dtype: int64
```

Mode dari kolom reviews_per_month:

In [23]:
```
listings_new.reviews_per_month.mode()
```

Out[23]:
```
0    1.0
dtype: float64
```

Replace missing values:

In [24]:
```
listings_new['reviews_per_month'].fillna(listings_new['reviews_per_month'].mode()[0], inplace = True)
```

Mengecek apakah kolom reviews_per_month yang valuenya missing sudah diganti dengan modenya:

In [25]:
```
listings_new.reviews_per_month.value_counts()
```

Out[25]:
```
1.00    2930
0.04     104
0.08      96
0.05      93
0.10      92
        ...
4.02       1
3.92       1
3.52       1
3.57       1
8.00       1
Name: reviews_per_month, Length: 527, dtype: int64
```

Mengecek apakah masih ada missing value pada kolom reviews_per_month:

In [26]:
```
listings_new.isnull().sum()
```

Out[26]:
```
id                                0
name                              0
host_id                           0
host_name                         0
neighbourhood_group               0
neighbourhood                     0
latitude                          0
longitude                         0
room_type                         0
minimum_nights                    0
number_of_reviews                 0
last_review                       0
reviews_per_month                 0
calculated_host_listings_count    0
availability_365                  0
price                             0
dtype: int64
```

Kolom reviews_per_month sudah tidak ada missing value lagi dan direplace dengan value modenya

3. Encoding Categorical data for neighbourhood_group variable and room_type variable

In [27]:

```
listings_new.neighbourhood_group.value_counts()
```

Out[27]:
```
Central Region       6309
West Region           540
East Region           508
North-East Region     346
North Region          204
Name: neighbourhood_group, dtype: int64
```

In [28]:
```
listings_new.room_type.value_counts()
```

Out[28]:
```
Entire home/apt    4132
Private room       3381
Shared room         394
Name: room_type, dtype: int64
```

Pada kolom neighbourhood_group, terdapat 5 kategori yang tidak berhubungan (bukan ordinal). Pada kolom room_type, terdapat 3 kategori juga yang tidak berhubungan (bukan ordinal). Maka, kita gunakan One Hot Encoding untuk encoding data pada kedua kolom tersebut.

Akan digunakan OneHotEncoder dan ColumnTransformer:

In [29]:
```
from sklearn.preprocessing import OneHotEncoder
from sklearn.compose import ColumnTransformer
```

Sebelum melakukan encoding pada x, saya extract ulang variable x sehingga datanya terupdate (Menggunakan data setelah dilakukan handling missing data)

In [30]:
```
#Extracting independent variables:
x = listings_new.iloc[:,:-1].values #Extract semua kolom kecuali kolom terakhir
print(x)
```

```
[[49091 'COZICOMFORT LONG TERM STAY ROOM 2' 266763 ... 0.01 2 365]
 [50646 'Pleasant Room along Bukit Timah' 227796 ... 0.28 1 365]
 [56334 'COZICOMFORT' 266763 ... 0.2 2 365]
 ...
 [38109336 '[ Farrer Park ] New City Fringe CBD Mins to MRT' 281448565
  ... 1.0 3 173]
 [38110493 'Cheap Master Room in Central of Singapore' 243835202 ... 1.0
  2 30]
 [38112762 'Amazing room with private bathroom walk to Orchard' 28788520
  ... 1.0 7 365]]
```

Encoding:

In [31]:
```
ct = ColumnTransformer([("Neighbourhood Group & Room Type", OneHotEncoder(), [4,8])], remainder = 'passthrough')
#[4,8] menunjukkan kolom yang diencode, kolom neighbourhood_group dan room_type berada pada kolom nomor 4 dan 8
x = ct.fit_transform(x)
print(x)
```

```
[[0.0 0.0 1.0 ... 0.01 2 365]
 [1.0 0.0 0.0 ... 0.28 1 365]
 [0.0 0.0 1.0 ... 0.2 2 365]
 ...
 [1.0 0.0 0.0 ... 1.0 3 173]
 [1.0 0.0 0.0 ... 1.0 2 30]
 [1.0 0.0 0.0 ... 1.0 7 365]]
```

4. Splitting the Dataset into the Training set and Test set

In [32]:
```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test= train_test_split(x, y, test_size= 0.2, random_state=0)
```

5. Print x_train, x_test, y_train and y_test

In [33]:
```
#x_train: features for the training data
print(x_train)
```

```
[[1.0 0.0 0.0 ... 0.19 67 358]
 [1.0 0.0 0.0 ... 0.06 1 0]
 [1.0 0.0 0.0 ... 1.0 18 89]
 ...
 [1.0 0.0 0.0 ... 0.03 1 0]
 [0.0 0.0 0.0 ... 1.0 1 0]
 [1.0 0.0 0.0 ... 1.0 1 362]]
```

In [34]:
```
#x_test: features for testing data
print(x_test)
```

```
[[1.0 0.0 0.0 ... 0.23 1 324]
 [0.0 0.0 0.0 ... 0.81 4 361]
```

```
[1.0 0.0 0.0 ... 1.3 27 345]
...
[1.0 0.0 0.0 ... 1.18 84 0]
[1.0 0.0 0.0 ... 0.58 67 363]
[1.0 0.0 0.0 ... 1.58 18 77]]
```

In [35]:

```
#y_train: Dependent variables for training data
print(y_train)
```

```
[100 119  99 ...  85 200 135]
```

In [36]:

```
#y_test: Independent variable for testing data
print(y_test)
```

```
[131  62  83 ...  82 150  99]
```