Capstone project
Hotel Booking Analysis

Amritanshu Kumar

Data scientist Enthusiast

INDEX

Introduction

- **Hotel :** A place where you pay to stay when you are on holiday or traveling
- **Types**

1. **City Hotel:** A city hotel is what you probably know best. It provides accommodation and meals to travellers . Often times, people come from all over the world to stay at a hotel so that they can tour around the place that they are staying.

1. **Resort:** A resort is a self-contained commercial establishment that tries to provide most of a vacationer's wants , such as food , drink, swimming lodging ,sports , entertainment , and shopping , on the premises

Data segment

- **Is cancelled :** The Is cancelled column shows the hotel booking cancelled or not where “0” indicates there are no cancellation.
- **Lead time :** The time between actual arrival and reservation.
- **Arrival date year :** In this there are three year of data like 2015,2016,2017.it is telling us about in which year we have booking.
- **Arrival date month :** It includes all the 12 month like ‘January’, ‘february’ , ‘march’ , ‘April’ , ‘may’ , ‘June’ , ‘July’ , ‘august’ , ‘September’ , ‘October’ , ‘November’ , ‘December’.it is telling us about in which month we have booking.
- **Arrival date week number :** It is telling us about in which week from the year we have reservation

Data segment

- **Arrival date day of month :** In this we have aware about in which date we have reservation.
- **Stayed in weekend nights :** How many reservation made and how many person staying in weekend nights.
- **Stayed in weekday nights :** How many reservation made and how many person staying in weekday nights.
- **Meal :** there are many preference per reservation like BB, FB, HB, SC, Undefined.
- **Country :** In this we aware about the where do people come from and staying in our resort or hotel mostly.
- **Peoples :** In this which type of people stayed in our hotel or resort segment wise like Adults, with their families , with their childrens.

Data segment

- **Market segment :** There are many market segment and reservation made and the purpose of reservation like eg,

  direct means any person who reservation made individually,

  corporate means any corporate trip made,

  groups mean any reservation made by group of people etc.
- **Distribution channel :** There are many distribution channel are available and made through the booking (Direct, Corporate, GDS, TA/TO , Underdefined)
- **Is Repeated Guest :** How many guest are visit in hotel or resort repeatedly there are two values given ‘0’ for none and ‘1’ for the guest who visited again

Data segment

- **Reserved room type :** Reserved room type means like which type types of room reserved like there are many types of room ‘A’ , ’B’ , ’C’ ’D’ , ’E’ , ‘F’ , ‘G’ , ‘H’ , ‘L’ , ‘P’.
- **Assigned room type :** It is help us to tell about which type of room assigned to which person.
- **Booking changes :** In this like how many booking changes in which month and year.
- **Deposit type :** The deposit type means like how person deposit their amount regarding reservation and there are three type of deposite like ‘no deposit’ , ‘non refund’ , ‘refundable’.
- **Days in waiting list :** How many waiting days for reservation

Data segment

- **Customer type :** Customer Type means a customer class, a customer sub-class, or a specific group of customers so there are some customer type the given data like ‘transient’ , ‘contract’ , ‘group’ , ‘transient party’.
- **Required car parking :** In this it is telling us about how many car parking are available in there.
- **Total special reservation :** In this segment we aware about how many special reservation are there like eg . ‘birthday party’ , ‘marriage ceremony’ , ‘marriage anniversary’ , ‘seminar’ , ‘office meeting’.

Data segment

- **Reservation status :** The reservation status aware us about what is the current status of Hotel or Resort it is divided into three part

1. **Check out :** The action or an instance of leaving your room in a hotel , Resort, etc., after you are finished staying there.
2. **Cancelled :** In this cancelled means like how many booked rooms were cancel happens.
3. **No show :** There are no reservation there.

Data segment

- **Previous booking not cancelled :** Number of previous bookings not cancelled by the customer prior to the current booking.
- **Booking changes :** number of changes made to the booking from the moment the booking was entered on the PMS until the moment of check-in or cancelltion .
- **Reserved room type :** It is a room which is code of room type.
- **Assigned room type :** code for the type of the room assigned to the booking.
- **Agent :** It means the ID of the travel agency who made the booking for city hotel and resort hotel.

DATA SEGMENT

- Is cancelled
- Arrival date year
- Arrival month
- Arrival date
- Stayed in weekend
- Stayed in weekday night
- Types of meal
- Country wise visitors
- Types of people
- Market segment
- Distribution channel
- Deposit type
- Booking type
- Reesrvation type
- Total special request
- Requiure of car parking

Data cleaning

- Data processing
- Dealing with missing values
- How many booking were cancelled
- The percentage of booking for each year
- Which is bussiest month for hotels or resort
- How many adults stayed in the weekend


Error Free Data
Maintains Data Consistency
Benefits of Data Cleaning
Data Quality
Complete Data
Accuracy and Efficiency

Benefits of data cleaning

- 1. Error-Free Data: When multiple sources of data are combined there may be chances of so much error. Through Data Cleaning, errors can be removed from data. Having clean data which is free from wrong and garbage values can help in performing analysis faster as well as efficiently. By doing this task our considerable amount of time is saved. If we use data containing garbage values, the results won't be accurate. When we don't use accurate data, surely we will make mistakes. Monitoring errors and good reporting helps to find where errors are coming from, and also makes it easier to fix incorrect or corrupt data for future applications.
- 2. Data Quality: The quality of the data is the degree to which it follows the rules of particular requirements. For example, if we have imported phone numbers data of different customers, and in some places, we have added email addresses of customers in the data. But because our needs were straightforward for phone numbers, then the email addresses would be invalid data. Here some pieces of data follow a specific format. Some types of numbers have to be in a specific range. Some data cells might require a selected quite data like numeric, Boolean, etc. In every scenario, there are some mandatory constraints our data should follow. Certain conditions affect multiple fields of data in a particular form. Particular types of data have unique restrictions. If the data isn't in the required format, it would always be invalid. Data cleaning will help us simplify this process and avoid useless data values.

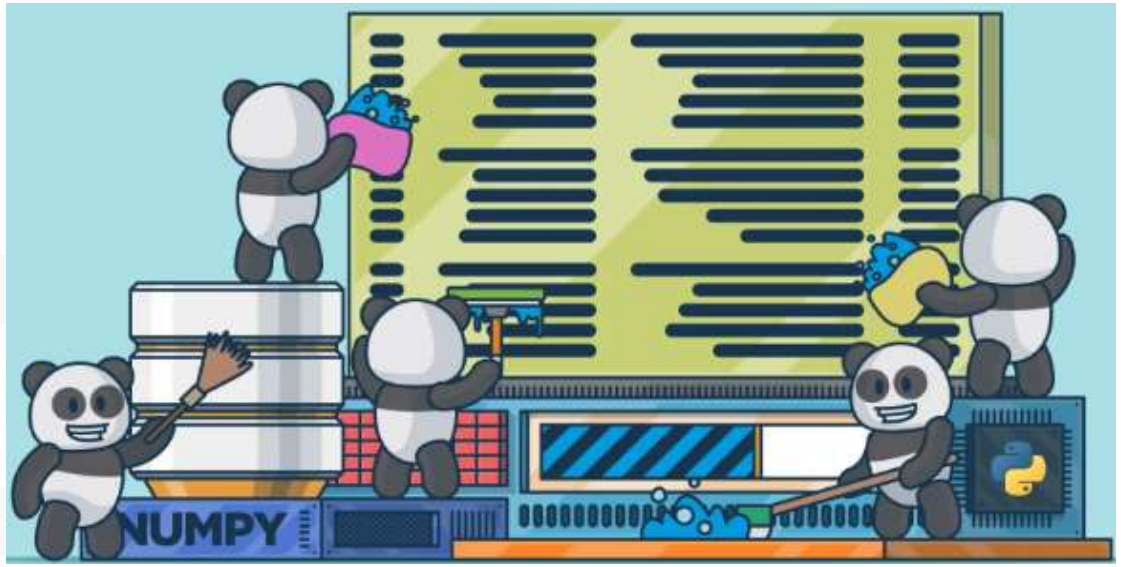
Benefits of data cleaning

- **3. Accurate and Efficient:** Ensuring the data is close to the correct values. We know that most of the data in a dataset are valid, and we should focus on establishing its accuracy. Even if the data is authentic and correct, it doesn't mean the data is accurate. Determining accuracy helps to figure out the data entered is accurate or not. For example, the address of a customer is stored in the specified format, maybe it doesn't need to be in the right one. The email has an additional character or value that makes it incorrect or invalid. Another example is the phone number of a customer. This means that we have to rely on data sources, to cross-check the data to figure out if it's accurate or not. Depending on the kind of data we are using, we might be able to find various resources that could help us in this regard for cleaning.
- **4. Complete Data:** Completeness is the degree to which we should know all the required values. Completeness is a little more challenging to achieve than accuracy or quality. Because it's nearly impossible to have all the info we need. Only known facts can be entered. We can try to complete data by redoing the data gathering activities like approaching the clients again, re-interviewing people, etc. For example, we might need to enter every customer's contact information. But a number of them might not have email addresses. In this case, we have to leave those columns empty. If we have a system that requires us to fill all columns, we can try to enter missing or unknown there. But entering such values does not mean that the data is complete. It would be still being referred to as incomplete.

Benefits of data cleaning

- **5. Maintains Data Consistency:** To ensure the data is consistent within the same dataset or across multiple datasets, we can measure consistency by comparing two similar systems. We can also check the data values within the same dataset to see if they are consistent or not. Consistency can be relational. For example, a customer's age might be 25, which is a valid value and also accurate, but it is also stated as a senior citizen in the same system. In such cases, we have to cross-check the data, similar to measuring accuracy, and see which value is true. Is the client a 25-year old? Or the client is a senior citizen? Only one of these values can be true. There are multiple ways to for your data consistent.
- By checking in different systems.
- By checking the source.
- By checking the latest data.

❖ So basically there are 4 columns company, agent , country and children with missing values



Data Preprocessing

First copy the dataset, so our original dataset remains unchanged

```
[ ]  copy_of_data = hotel_df.copy()
```

1.Dealing with Missing Values

Check if our data contains any missing values

```
[ ]  hotel_df.isnull().sum().sort_values(ascending=False)[:10]
```

```
company                     112593
agent                        16340
country                        488
children                         4
reserved_room_type               0
assigned_room_type               0
booking_changes                  0
deposit_type                     0
hotel                            0
previous_cancellations           0
dtype: int64
```

```
[ ]  hotel_df[['agent','company']] = hotel_df[['agent','company']].fillna(0.0)
     hotel_df
```

We have some features with missing values.

In the agent and the company column, we have id_number for each agent or company, so for all the missing values, we will just replace it with 0.

Exploratory Data Analysis

- Now let's do the fun part, extract the information from our data and try to answer our questions.
  How Many Booking Were Cancelled?
  Let's write the function to get the percentage of different values.

```
[ ] def get_count(series, limit=None):

        '''
        INPUT:
            series: Pandas Series (Single Column from DataFrame)
            limit:  If value given, limit the output value to first limit samples.
        OUTPUT:
            x = Unique values
            y = Count of unique values
        '''

        if limit != None:
            series = series.value_counts()[:limit]
        else:
            series = series.value_counts()

        x = series.index
        y = series/series.sum()*100

        return x.values,y.values
```

- This function takes a series or data frame column and returns the two arrays x is our unique values y is the percentage value of each unique value Now let's use this function on our is_canceled feature and see the result

❑ is_canceled have two unique values: 1 if booking got canceled, else 0.
Now let's plot this result. I will write another function to plot the diagram. The good thing about writing function is that we can reuse the code again and again.

```
x,y = get_count(df_not_canceled['is_canceled'])
x,y
```

```
(array([0]), array([100.]))
```

```
'''
  INPUT:
      x:        Array containing values for x-axis
      y:        Array containing values for y-axis
      x_lable:  String value for x-axis label
      y_lable:  String value for y-axis label
      title:    String value for plot title
      figsize:  tuple value, for figure size
      type:     type of plot (default is bar plot)

  OUTPUT:
      Display the plot
  '''
```

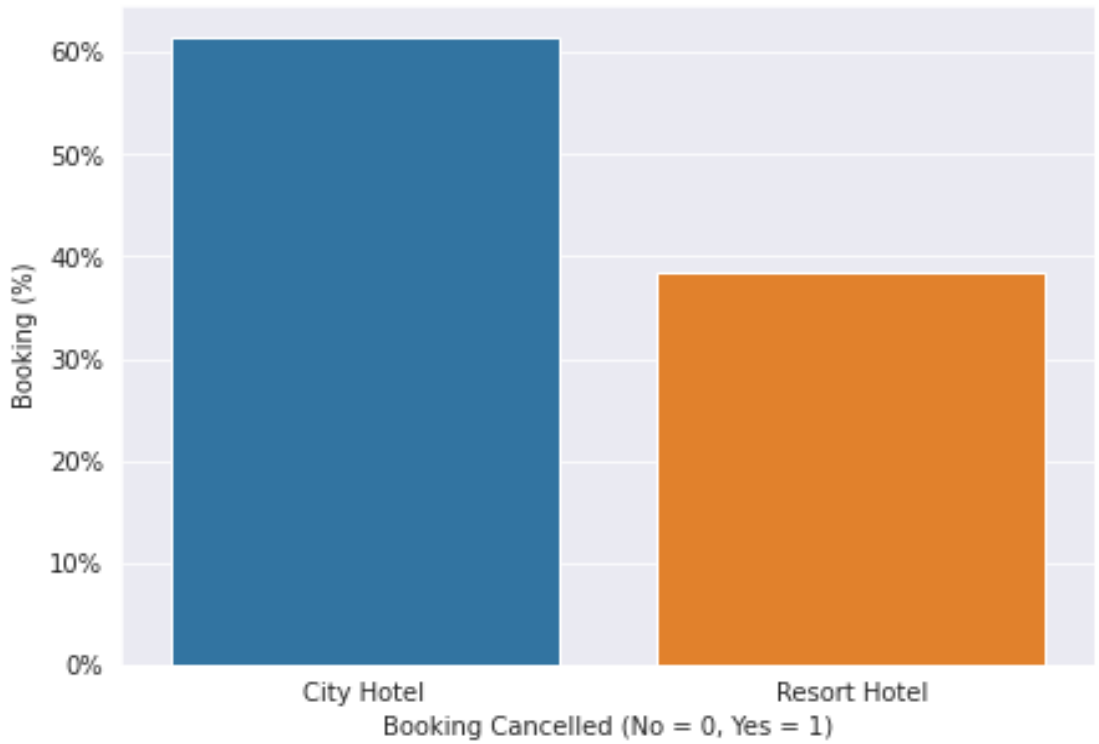
```
def plot(x, y, x_label=None,y_label=None, title=None, figsize=(7,5), type='bar'):

    sns.set_style('darkgrid')

    fig, ax = plt.subplots(figsize=figsize)

    ax.yaxis.set_major_formatter(mtick.PercentFormatter())

    if x_label != None:
        ax.set_xlabel(x_label)

    if y_label != None:
        ax.set_ylabel(y_label)

    if title != None:
        ax.set_title(title)

    if type == 'bar':
        sns.barplot(x,y, ax = ax)
    elif type == 'line':
        sns.lineplot(x,y, ax = ax, sort=False)
```

This function takes two arrays, x, and y and displays the required diagram. The default plot type is a bar plot, but it can also plot the line plot. Optional arguments can be given to display title and labels.

Now let's call the function

```
[25] plot(x,y, x_label='Booking Cancelled (No = 0, Yes = 1)', y_label='Booking (%)')
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variables as keyword
  FutureWarning
```

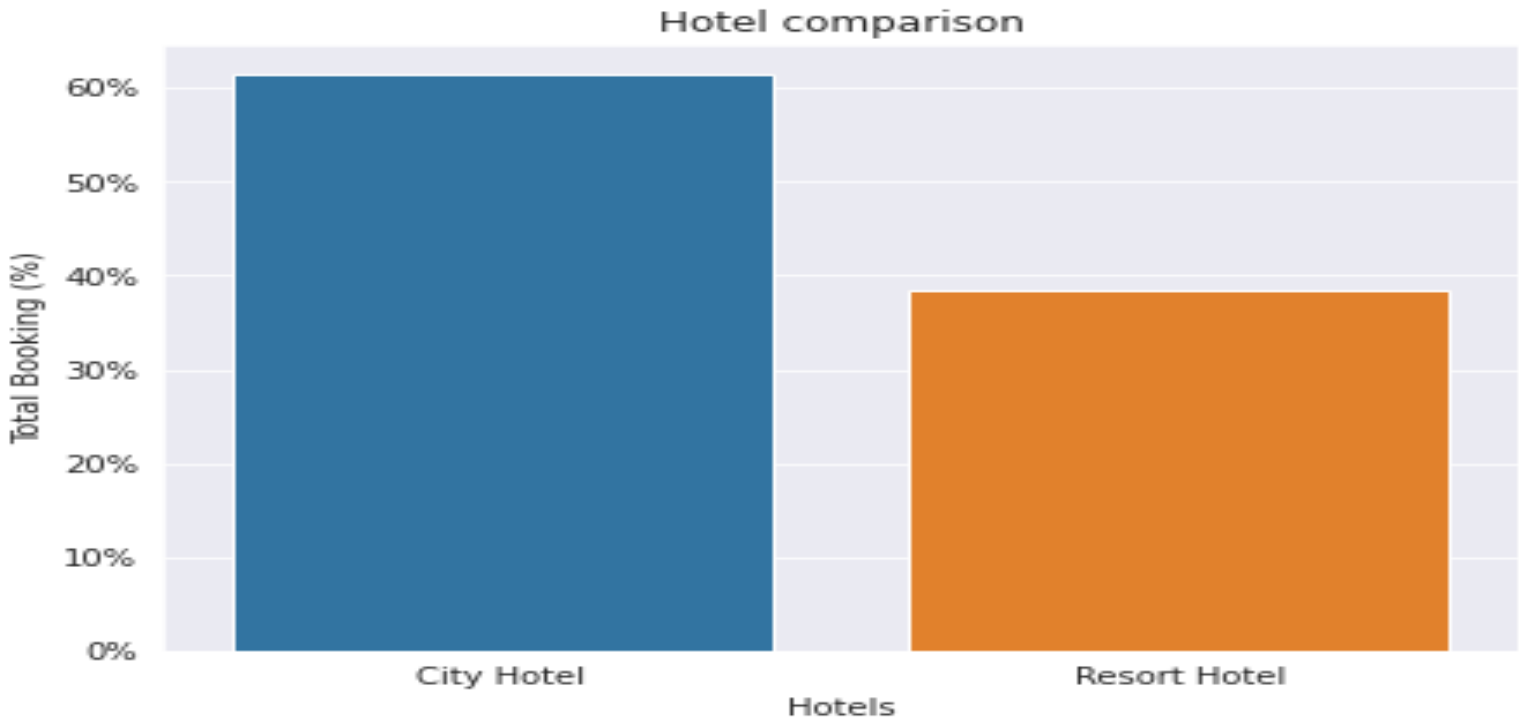


❖What is the booking ratio between Resort Hotel and City Hotel?

Let's answer another question, how many bookings were made for each type of hotel.
We can now reuse the functions that we created earlier. All we have to do is to pass the dataframe column to get_count() function and pass its result (x and y array) to plot function.

```
[26] x,y = get_count(df_not_canceled['hotel'])
     plot(x,y, x_label='Hotels', y_label='Total Booking (%)', title='Hotel comparison')
```

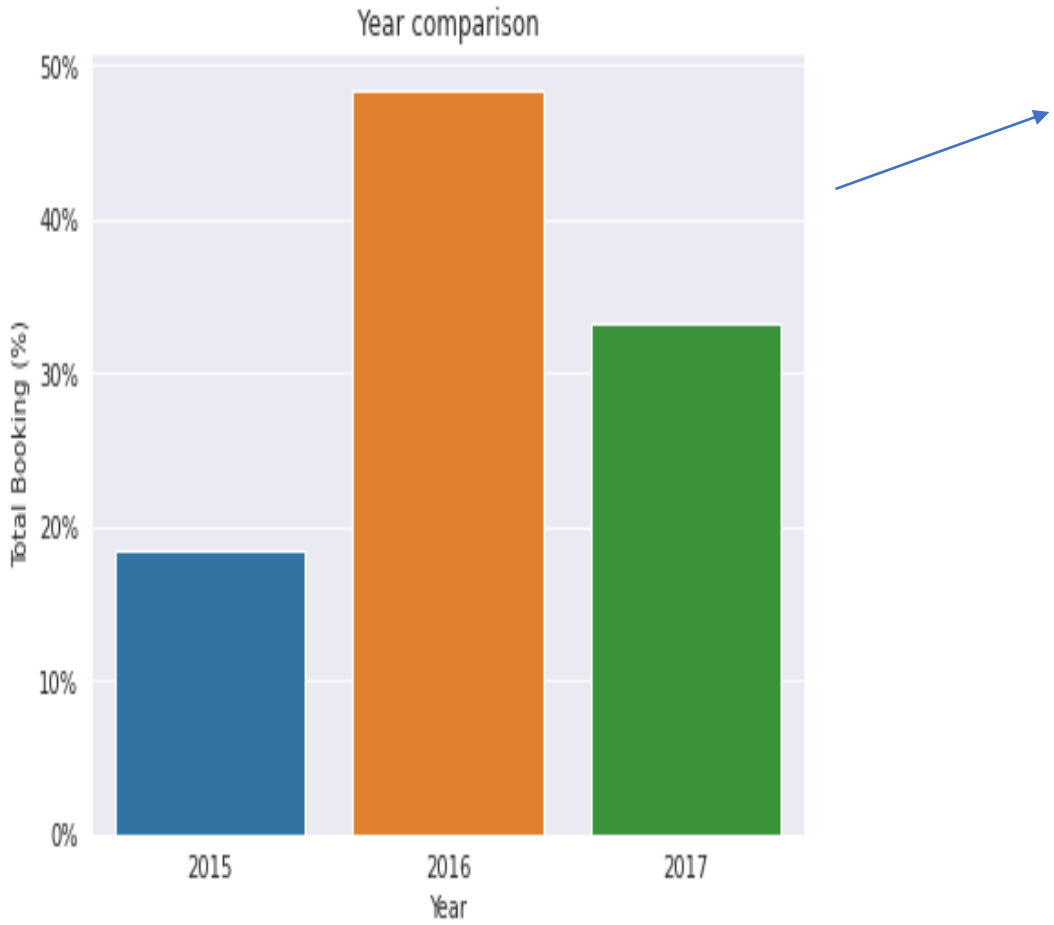
```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pa
  FutureWarning
```



The percentage of booking for each year

```
[16] x,y = get_count(df_not_canceled['arrival_date_year'])
     plot(x,y, x_label='Year', y_label='Total Booking (%)', title='Year comparison')
```

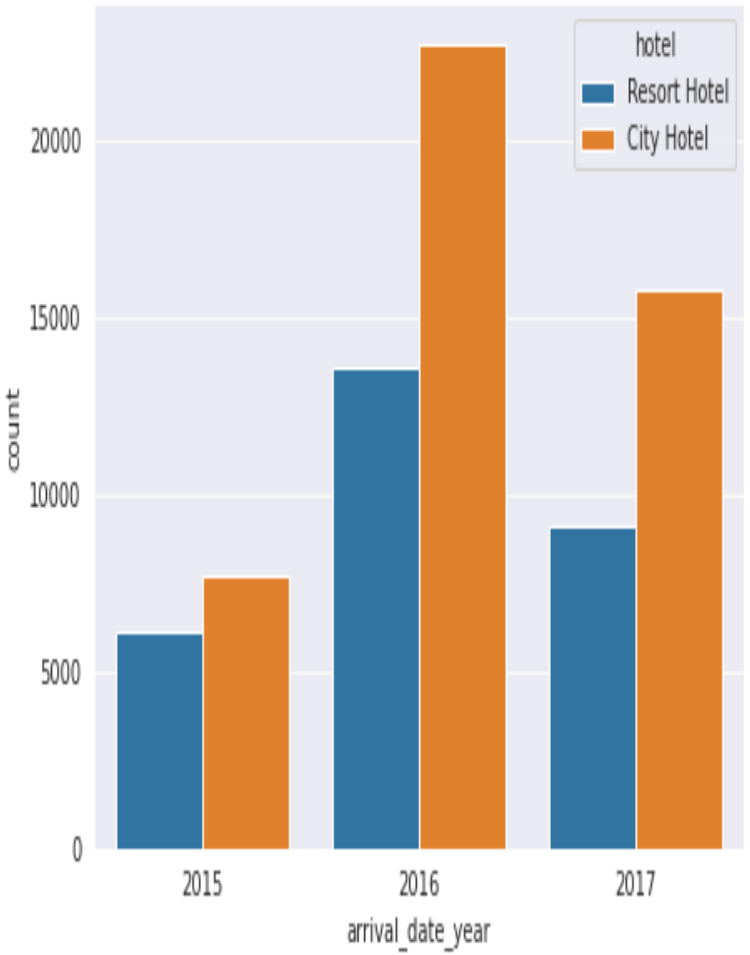
```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning:
  FutureWarning
```



- 2015 : This year almost cover the approx. 20 percentage of booking.
- 2016 : this year almost cover the approx. 50 percentage of booking.
- 2017 : This year almost cover the approx. 30 to 35 percentage of booking

Let's separate it by the hotel and then plot the diagram. We will change our code to display the countplot.

```
plt.subplots(figsize=(7,5))
sns.countplot(x='arrival_date_year', hue='hotel',  data=df_not_canceled);
```



- 2015 : In the column graph the resort hotel contains approx. 6000 of booking and the city hotel contains around 7000 of bookings
- 2016 : : In the column graph the resort hotel contains approx. 13000 of booking and the city hotel contains around 23000 of bookings
- 2017 : : In the column graph the resort hotel contains approx. 9000 of booking and the city hotel contains around 16000 of booking

➢ Which is the busiest month for hotels?

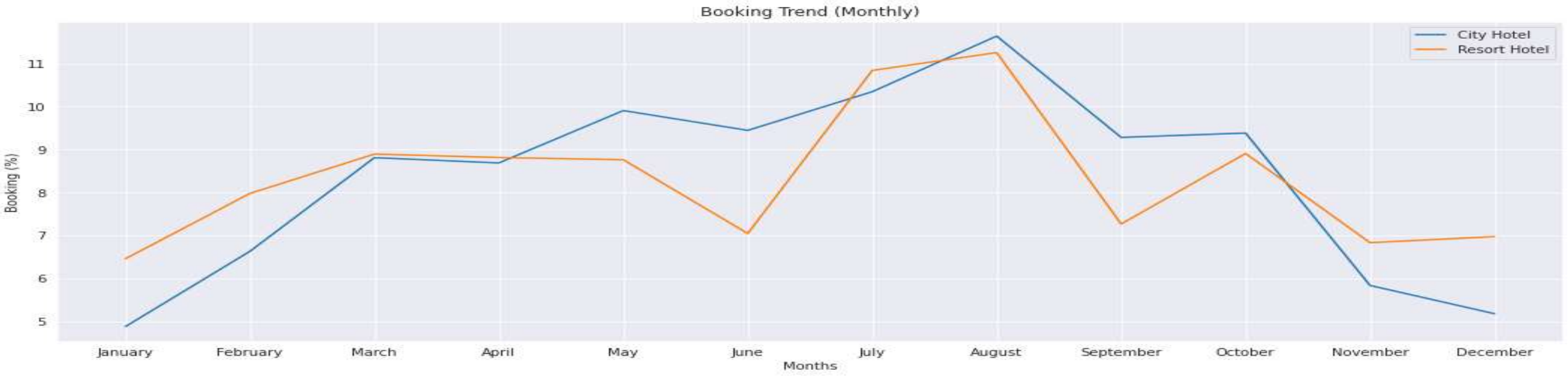
To answer this question, we will select the arrival_date_month feature and get its value count. Now the resulting data will not be sorted according to month order so we have to sort it. We will make the new list with the names of months in order to sort our data according to this list.

We will display the Lineplot to display the trend.

```
[18] new_order = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September',
                  'October', 'November', 'December']

     sorted_months = df_not_canceled['arrival_date_month'].value_counts().reindex(new_order)

     x = sorted_months.index
     y = sorted_months/sorted_months.sum()*100


     #sns.lineplot(x, y.values)
     plot(x, y.values, x_label='Months', y_label='Booking (%)', title='Booking Trend (Monthly)', type='line', figsize=(18,6))
```

➢ Comparison between resort hotel and city hotel reservation

```
[19] new_order = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October',
                  'November', 'December']
     ## there are  only City Hotel
     sorted_months = df_not_canceled.loc[hotel_df.hotel=='City Hotel' ,'arrival_date_month'].value_counts().reindex(new_order)

     x1 = sorted_months.index
     y1 = sorted_months/sorted_months.sum()*100

     ## there are only Resort Hotel
     sorted_months = df_not_canceled.loc[hotel_df.hotel=='Resort Hotel' ,'arrival_date_month'].value_counts().reindex(new_orde

     x2 = sorted_months.index
     y2 = sorted_months/sorted_months.sum()*100
     ## Draw the line plot
     fig, ax = plt.subplots(figsize=(18,6))

     ax.set_xlabel('Months')
     ax.set_ylabel('Booking (%)')
     ax.set_title('Booking Trend (Monthly)')

     sns.lineplot(x1, y1.values, label='City Hotel', sort=False)
     sns.lineplot(x1, y2.values, label='Resort Hotel', sort=False)

     plt.show()
```



AlmaBetter

Pandas

Thank You