Investigate Business Hotel using Data Visualization

Created by:
Bagus Ariobimo
bagusariobimo7@gmail.com
https://www.linkedin.com/in/bagus-ariobimo
https://github.com/riyouuyt

"A data enthusiast who has completed a course in this field and is ready to start his career. Have a excellent understanding in statistics, programming, and data processing. Proficient in using tools such as Python, R, and SQL. Able to collect, clean and analyze data with the necessary techniques. Have skills in data visualization and simple statistical modeling. Creative problem solver and has a passion for learning. Ready to contribute to data-driven projects and collaborate in teams. Committed to further developing data science skills and achieving significant results in data analysis."

Supported by:
Rakamin Academy
Career Acceleration School
www.rakamin.com

Overview

"It is very important for a company to always analyze its business performance. On this occasion, we will delve deeper into business in the hospitality sector. Our focus is to find out how our customers behave in making hotel reservations, and its relationship to the rate of cancellation of hotel reservations. We will present the results of the insights we find in the form of data visualization to make it easier to understand and more persuasive."

Data Overview

Business Statement 🏢:

As a member of the Data Scientist team at a hotel company, the main goal of this Mini Project is to provide insight and in-depth understanding of our hotel's business performance. 🚀 Through data exploration, we will analyze customer behavior in booking hotel tickets, look for factors that influence hotel ticket booking cancellations, and identify opportunities to improve our services and profitability. 📊 The results of this analysis will be presented using data visualization and data storytelling to help the management team make smarter and more strategic decisions in managing our hotel business.

Goals 🎯:

- Analyze customer behavior in booking hotel tickets, including booking patterns, length of stay, and room preferences.
- Identify factors that contribute to hotel ticket booking cancellations, such as price, room type, and booking period.
- Present discovered insights through informative and easy-to-understand data visualizations. 📈
- Provide recommendations to the hotel management team based on analysis findings to improve service and profitability. 💼
- Create powerful data stories to help management teams make smarter, data-driven decisions. 📖

Data Preparation

Project Data Column Information:

- **Hotel type**: (Resort Hotel or City Hotel).
- **is_canceled**: Indicates if the order was canceled (1 = Yes, 0 = No).
- **lead_time**: Number of days between booking date and arrival date.
- **arrival_date_year**: The year the customer arrived.
- **arrival_date_month**: The month the customer arrived.
- **arrival_date_week_number**: Week number in the year of arrival.
- **arrival_date_day_of_month**: Arrival day in month.
- **stays_in_weekend_nights**: Number of weekend nights spent by the customer.
- **stays_in_weekdays_nights**: Number of weeknights spent by the customer.
- **adults**: The number of adults in the booking.
- **children**: Number of children in the booking.
- **babies**: Number of babies in the order.
- **meal:** The type of meal ordered.
- **city:** Destination city code.

Data Preparation

- **market_segment**: Customer market segment.
- **distribution_channel**: Order distribution channel.
- **is_repeated_guest**: Signs if the customer is a repeat guest (1 = Yes, 0 = No).
- **previous_cancellations**: Number of orders previously canceled by the customer.
- **previous_bookings_not_canceled**: Number of previous bookings that were not canceled by the customer.
- **booking_changes**: Number of changes made to the booking.
- **deposit_type**: Type of deposit paid (No Deposit, Non Refund, or Refundable).
- **agent**: ID of the agent who placed the order.
- **company**: Company ID if the order was made by a company.
- **days_in_waiting_list**: The number of days in the waiting list before the booking is confirmed.
- **customer_type**: Customer type (Transient, Contract, or Group).
- **adr**: Average Daily Rate, average daily rate.
- **required_car_parking_spaces**: Number of parking spaces required by the customer.
- **total_of_special_requests**: Number of special requests submitted by customers.
- **reservation_status**: Reservation status (Canceled, Check-Out, or No-Show).

Data Preprocessing

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119390 entries, 0 to 119389
Data columns (total 29 columns):
 #   Column                          Non-Null Count   Dtype
---  ------                          --------------   -----
 0   hotel                           119390 non-null  object
 1   is_canceled                     119390 non-null  int64
 2   lead_time                       119390 non-null  int64
 3   arrival_date_year               119390 non-null  int64
 4   arrival_date_month              119390 non-null  object
 5   arrival_date_week_number        119390 non-null  int64
 6   arrival_date_day_of_month       119390 non-null  int64
 7   stays_in_weekend_nights         119390 non-null  int64
 8   stays_in_weekdays_nights        119390 non-null  int64
 9   adults                          119390 non-null  int64
 10  children                        119386 non-null  float64
 11  babies                          119390 non-null  int64
 12  meal                            119390 non-null  object
 13  city                            118902 non-null  object
 14  market_segment                  119390 non-null  object
 15  distribution_channel            119390 non-null  object
 16  is_repeated_guest               119390 non-null  int64
 17  previous_cancellations          119390 non-null  int64
 18  previous_bookings_not_canceled  119390 non-null  int64
 19  booking_changes                 119390 non-null  int64
 20  deposit_type                    119390 non-null  object
 21  agent                           103050 non-null  float64
 22  company                         6797 non-null    float64
 23  days_in_waiting_list            119390 non-null  int64
 24  customer_type                   119390 non-null  object
 25  adr                             119390 non-null  float64
 26  required_car_parking_spaces     119390 non-null  int64
 27  total_of_special_requests       119390 non-null  int64
 28  reservation_status              119390 non-null  object
dtypes: float64(4), int64(16), object(9)
memory usage: 26.4+ MB
```

During the data preprocessing phase, several crucial steps were taken to ensure the data's quality and usability:

- **Removing Duplicate Data**: Duplicate records were identified and eliminated from the dataset. This step guarantees that each entry in the dataset is unique, preventing any redundancy in the analysis.
- **Correcting Data Types**: Some columns required adjustments in their data types to accurately represent the information they contain. This ensures that calculations and operations on the data are performed correctly.
- **Handling Invalid Data**: Invalid or missing data values were addressed by appropriate employing strategies. For example, zero values were filled in for missing values in certain columns, and 'unknown' values were assigned to others where data was unavailable.
- **Dropping Unnecessary Data**: Columns that did not contribute significantly to the analysis or were irrelevant to the project's objectives were removed. This streamlines the dataset, making it more focused and efficient for further analysis.

These meticulous data preprocessing steps were executed to prepare the data for in-depth analysis and to ensure the accuracy and integrity of the insights derived from it.

Data Preprocessing

1. Removing the duplicate data

The dataset originally had 119,210 rows and 29 columns. After removing duplicate rows, the dataset size is reduced to 85,953 rows while retaining the same number of columns. This reduction in dataset size indicates the presence of duplicate records in the original dataset. Removing duplicates improves data integrity and quality, ensuring that the data used for analysis is free from redundancy and data entry errors.

2. Handling a Missing Values

In the dataset, there are missing values present in several columns. Here is a brief summary of the columns with missing data:

- **company (Float64):** This column has the highest number of missing values, with 81,019 entries (approximately 94.07% of the total) being null. It likely indicates that most entries in this column do not involve a company.
- **agent (Float64):** There are 11,941 missing values (approximately 13.86% of the total) in this column. These missing values suggest that a substantial number of entries do not have an associated agent.

- **city (Object)**: Around 450 entries (approximately 0.52% of the total) have missing values in the 'city' column. These entries likely represent cases where city information was not available.children (Float64): A small number of entries (4 in total) have missing values (approximately 0.0046% of the total) in the 'children' column. This might indicate that these customers did not specify the number of children when booking.It's important to address these missing values appropriately during data preprocessing to ensure accurate and meaningful analysis.

3. Handling Incorrect Data Types

In this data preprocessing step, the data types of three columns in the DataFrame were converted to 'int64' as follows:

- children Column: The 'children' column was converted to 'int64' data type, ensuring that it contains integer values.
- agent Column: Similarly, the 'agent' column was also converted to 'int64' data type, ensuring it contains integer values.
- company Column: Lastly, the 'company' column was converted to 'int64' data type, making sure it consists of integer values.By performing these data type conversions, the DataFrame now contains these columns as integers, which may be more suitable for certain types of analysis and computations.

4. Handling Incorrect Missing Values

After preprocessing the 'meal' column, the data values have been cleaned and reduced to the following refined categories:

- **'Breakfast'**
- **'Full Board'**
- **'Dinner'**
- **'No Meal'**

The 'Undefined' category has been gracefully omitted from the dataset.

For more information about the project, click here