Hotel booking demand dataset

Objectives:

The goal of this project is to

1-Exploring the hotel booking data and finding answers for some questions such as:

- what is the hotel types and which one is more demand?
- Which type of hotel has the highest number of cancellations?
- what is the busy months?
- How many guest from each country?
- What is the percentage of repeat customers?
- How many Customers with the history of cancellation?

2- Create a model to predict of the customer they cancel the booking or not.

Design:

First, I download the data from Kaggle website and read it. Then, I cleaned and explored the data. After that I showed the data to find the relation between the features. Then, I implemented three models .

Data:

This data set contains booking information for a city hotel and a resort hotel, and includes information such as when the booking was made, length of stay, the number of adults, children, and/or babies, and the number of available parking spaces, etc

This dataset includes 119390 entries with 32 features, here some clearly for the features:

Features :	Description :
is_canceled	Value indicating if the booking was canceled (1) or not (0)
lead_time	Number of days that elapsed between the entering date of the booking into the PMS and the arrival date
deposit_type	Indication on if the customer made a deposit to guarantee the booking. This variable can assume three categories: No
days_in_waiting_list	Number of days the booking was in the waiting list before it was confirmed to the customer
customer_type	Type of booking, assuming one of four categories:Transient - Transient-Party - Contract - Group
adr	Average Daily Rate as defined by dividing the sum of all lodging transactions by the total number of staying nights

Algorithms:

I started to clean the dataset by:

- Drop the duplicate rows.
- Drop the rows with column "***adult***=0 "
- Converting the datatype of ***children*** and ***agent*** from float to integer.
- Remove column ***company, days_in_waiting_list, arrival_date_year, assigned_room_type, booking_changes,reservation_status, ,days_in_waiting_list'.***
- Replace null values with 0 in ***agent*** feature.
- Replace NULL with "unknown" in column ***country***, after I finish exploring my data I decided to Remove this column.
- Fill null value in ***adr*** with mean.

EDA & Modelling:

I did some exploration on the data to find the answers for these questions above. After that I implemented a linear & classification algorithms models ⟶ "***Random Forest Classifier, Decision Tree Classifier, Logistic Regression***" to predict our target (is cancelled). I used these models to take the following features to predict if the customer will cancel the reservation or not:

'hotel', 'meal','market_segment','distribution_channel','reserved_room_type', 'deposit_type', 'customer_type', 'reservation_status_date',lead_time,arrival_date_month,arrival_date_week_number,arrival_date_day_of_month, stays_in_weekend_nights,stays_in_week_nights,adults ,children,babies,is_repeated_guest ,previous_cancellations,previous_bookings_not_canceled,agent ,adr ,required_car_parking_spaces,total_of_special_requests

To see the performance of these models I split my data into 80%(train- validation) sets /20% test set, and fit the models on train set, and test it on the validation and test sets.

The following is the accuracy Report for each model:

Logistic Regression:

	precision	recall	f1-score
0	0.95	1.00	0.97
1	0.00	0.00	0.00
accuracy			0.95

Decision Tree Classifier

	precision	recall	f1-score
0	0.96	0.91	0.93
1	0.17	0.33	0.22
accuracy			0.88

Random Forest Classifier

	precision	recall	f1-score
0	0.96	1.00	0.98
1	1.00	0.33	0.50
accuracy			0.96

These are the comparison accuracy scores of all the models arranged:

	Model	Score
2	Random Forest Classifier	0.964912
0	Logistic Regression	0.947368
1	Decision Tree Classifier	0.859649

The best Model is **Random Forest Classifier** has test accuracy of **96%.**

Tools:

- ***numpy, pandas*** for data manipulation.
- ***matplotlib and seaborn,*** for plotting.
- ***sklearn*** for modling .
- The worked done through ***Jupyter*** notebook using ***python.***