Hotel Bookings Cancellation

BY

Heba Mohamed

PROBLEM STAMENT

Have you ever wondered when the best time of year to book a hotel room is?

What is the optimal length of stay in order to get the best daily rate?

What if you wanted to predict whether or not a hotel was likely to receive a disproportionately high number of special requests?

PROPOSED SOLUTION

The main goal is to generate meaningful estimators from the data set we have and then choose the model that best predicts cancellation by comparing it to the accuracy ratings of several ML models.

- 2 different Logistic Regression models
- Random Forest using K-Folds cross-validation
- Decision Tree model
- XgBoost model

TOOLS

pandas

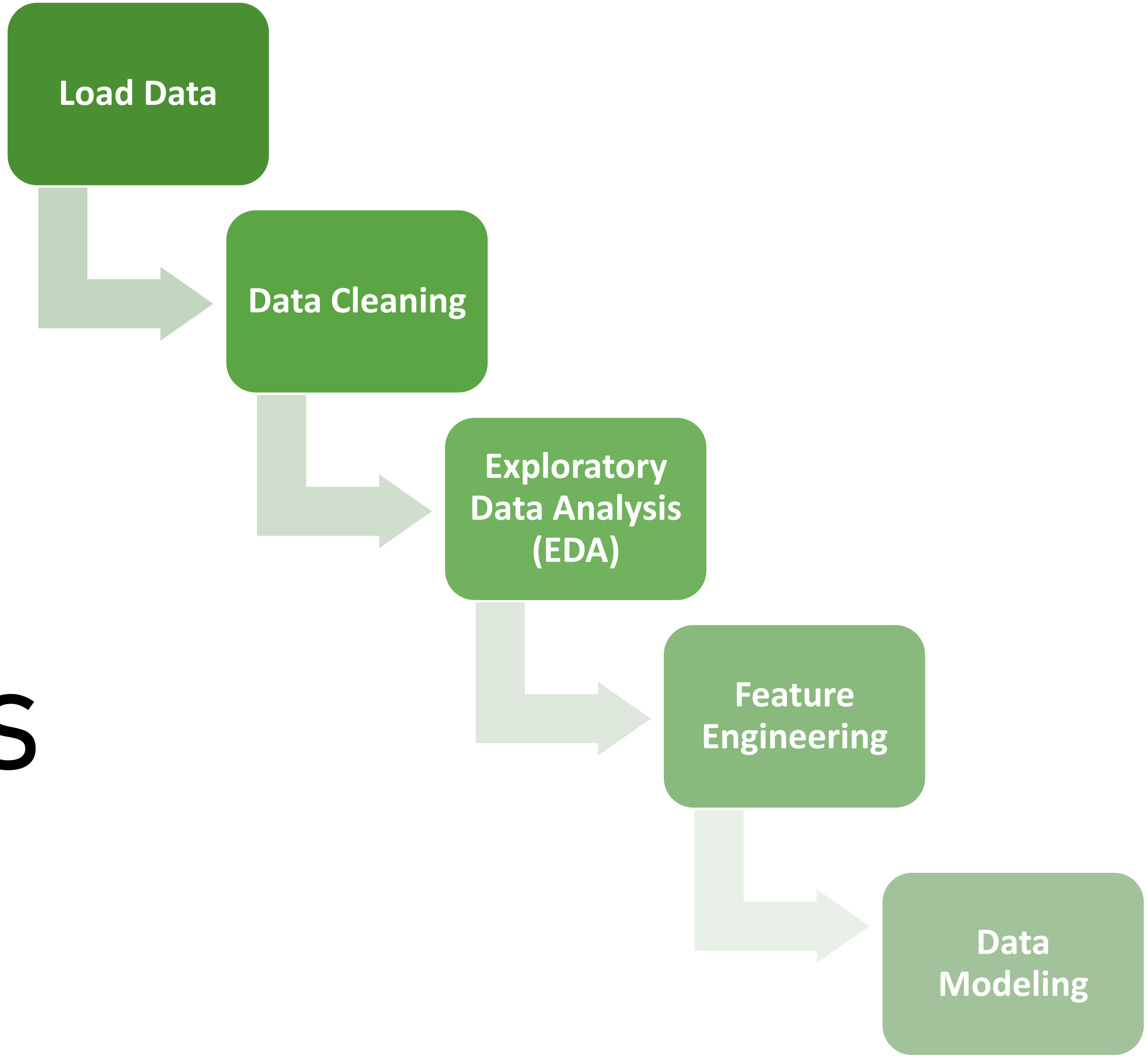
jupyter

scikit learn

plotly

Keras

PROJECT STAGES



Building
Training
Evaluating
Testing

DATASET

shape = (119390, 32)

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
hotel	is_cancele	lead_time	arrival_dat	arrival_dat	arrival_dat	arrival_dat	stays_in_w	stays_in_w	adults	children	babies	meal	country	market_se	distributio	is_repeate	previous_c	previous_b	reserved_r	assigned_r	booking_c	deposit_ty
Resort Hot	0	342	2015	July	27	1	0	0	2	0	0	BB	PRT	Direct	Direct	0	0	0	C	C	3	No Deposi
Resort Hot	0	737	2015	July	27	1	0	0	2	0	0	BB	PRT	Direct	Direct	0	0	0	C	C	4	No Deposi
Resort Hot	0	7	2015	July	27	1	0	1	1	0	0	BB	GBR	Direct	Direct	0	0	0	A	C	0	No Deposi
Resort Hot	0	13	2015	July	27	1	0	1	1	0	0	BB	GBR	Corporate	Corporate	0	0	0	A	A	0	No Deposi
Resort Hot	0	14	2015	July	27	1	0	2	2	0	0	BB	GBR	Online TA	TA/TO	0	0	0	A	A	0	No Deposi
Resort Hot	0	14	2015	July	27	1	0	2	2	0	0	BB	GBR	Online TA	TA/TO	0	0	0	A	A	0	No Deposi
Resort Hot	0	0	2015	July	27	1	0	2	2	0	0	BB	PRT	Direct	Direct	0	0	0	C	C	0	No Deposi
Resort Hot	0	9	2015	July	27	1	0	2	2	0	0	FB	PRT	Direct	Direct	0	0	0	C	C	0	No Deposi
Resort Hot	1	85	2015	July	27	1	0	3	2	0	0	BB	PRT	Online TA	TA/TO	0	0	0	A	A	0	No Deposi
Resort Hot	1	75	2015	July	27	1	0	3	2	0	0	HB	PRT	Offline TA/	TA/TO	0	0	0	D	D	0	No Deposi
Resort Hot	1	23	2015	July	27	1	0	4	2	0	0	BB	PRT	Online TA	TA/TO	0	0	0	E	E	0	No Deposi
Resort Hot	0	35	2015	July	27	1	0	4	2	0	0	HB	PRT	Online TA	TA/TO	0	0	0	D	D	0	No Deposi
Resort Hot	0	68	2015	July	27	1	0	4	2	0	0	BB	USA	Online TA	TA/TO	0	0	0	D	E	0	No Deposi
Resort Hot	0	18	2015	July	27	1	0	4	2	1	0	HB	ESP	Online TA	TA/TO	0	0	0	G	G	1	No Deposi
Resort Hot	0	37	2015	July	27	1	0	4	2	0	0	BB	PRT	Online TA	TA/TO	0	0	0	E	E	0	No Deposi
Resort Hot	0	68	2015	July	27	1	0	4	2	0	0	BB	IRL	Online TA	TA/TO	0	0	0	D	E	0	No Deposi
Resort Hot	0	37	2015	July	27	1	0	4	2	0	0	BB	PRT	Offline TA/	TA/TO	0	0	0	E	E	0	No Deposi
Resort Hot	0	12	2015	July	27	1	0	1	2	0	0	BB	IRL	Online TA	TA/TO	0	0	0	A	E	0	No Deposi
Resort Hot	0	0	2015	July	27	1	0	1	2	0	0	BB	FRA	Corporate	Corporate	0	0	0	A	G	0	No Deposi
Resort Hot	0	7	2015	July	27	1	0	4	2	0	0	BB	GBR	Direct	Direct	0	0	0	G	G	0	No Deposi
Resort Hot	0	37	2015	July	27	1	1	4	1	0	0	BB	GBR	Online TA	TA/TO	0	0	0	F	F	0	No Deposi
Resort Hot	0	72	2015	July	27	1	2	4	2	0	0	BB	PRT	Direct	Direct	0	0	0	A	A	1	No Deposi
Resort Hot	0	72	2015	July	27	1	2	4	2	0	0	BB	PRT	Direct	Direct	0	0	0	A	A	1	No Deposi
Resort Hot	0	72	2015	July	27	1	2	4	2	0	0	BB	PRT	Direct	Direct	0	0	0	D	D	1	No Deposi
Resort Hot	0	127	2015	July	27	1	2	5	2	0	0	HB	GBR	Offline TA/	TA/TO	0	0	0	D	I	0	No Deposi
Resort Hot	0	78	2015	July	27	1	2	5	2	0	0	BB	PRT	Offline TA/	TA/TO	0	0	0	D	D	0	No Deposi

DATA CLEANING – Dealing with nulls

```
column: children          Nulls:      4     Precentage: 0.00%
column: country           Nulls:    488     Precentage: 0.41%
column: agent             Nulls:  16340     Precentage: 13.69%
column: company           Nulls: 112593     Precentage: 94.31%
```

A 94.31% of company column are missing values. It seems that the best option is dropping the company column.

There are 334 unique agents, since there are too many agents, they may not be predictable. I will decide what to do about the agent after the correlation section.

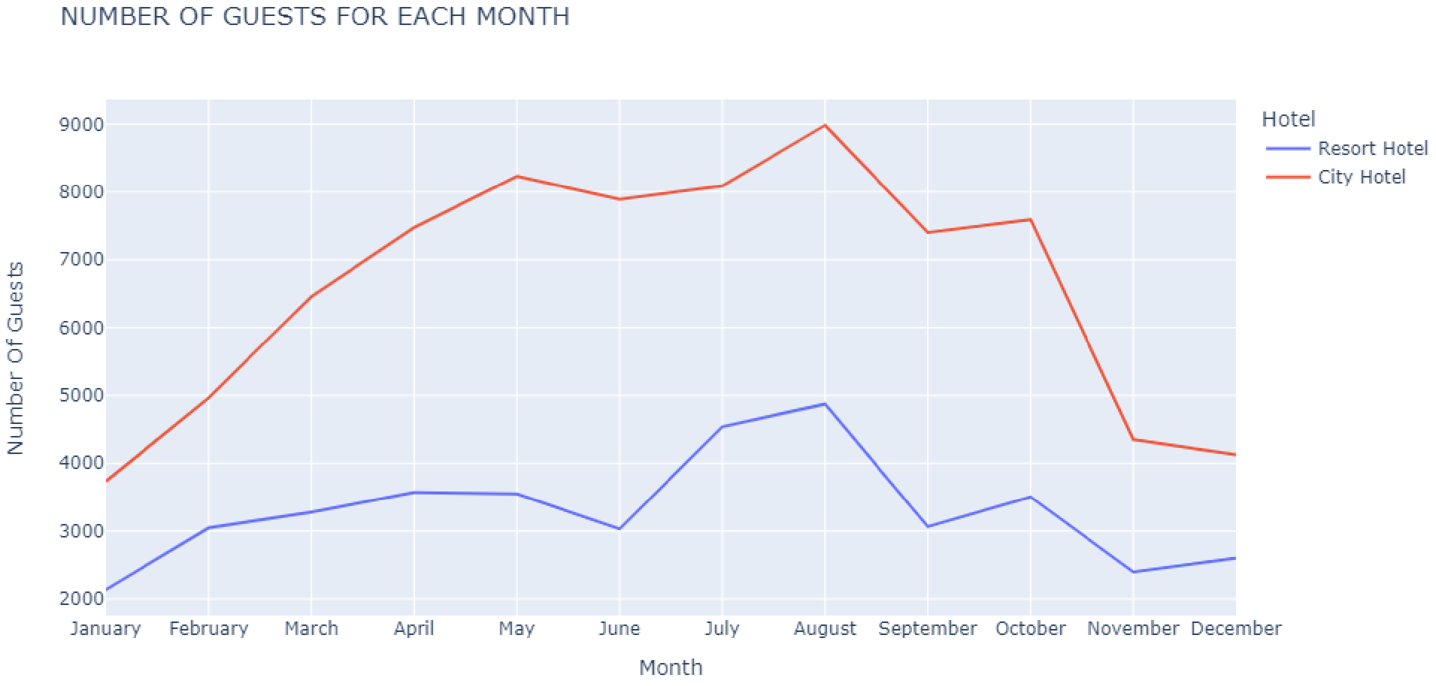
We have also 4 missing values in the children column. If there is no information about children those customers do not have any children.

We have also only 0.41% missing values in the country column. we can simply drop them.

Exploratory Data Analysis (EDA)

- *What is the busiest month?*
- *What is the busiest hotel?*

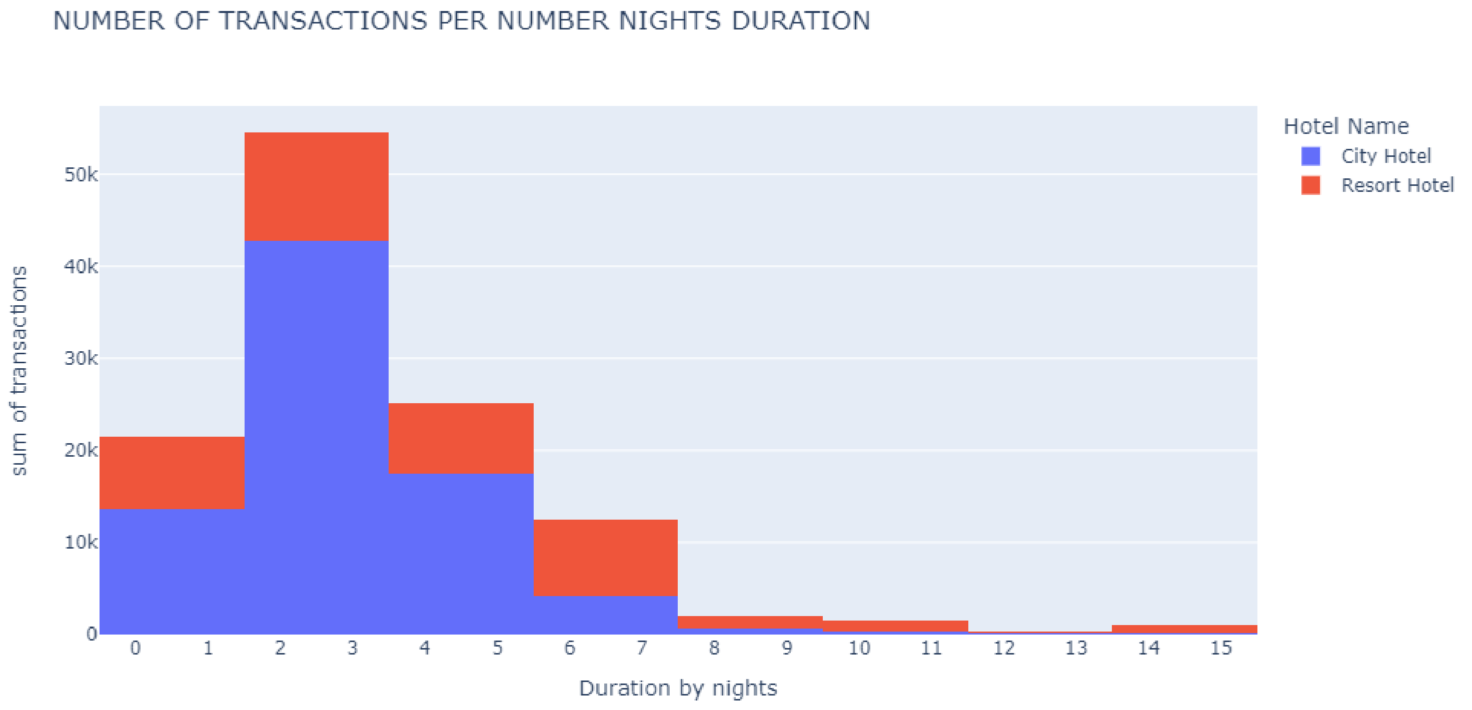
- ❑ *Seems that the August is the busiest month.*
- ❑ *Resort Hotel has less guests than City hotel.*



Exploratory Data Analysis (EDA)

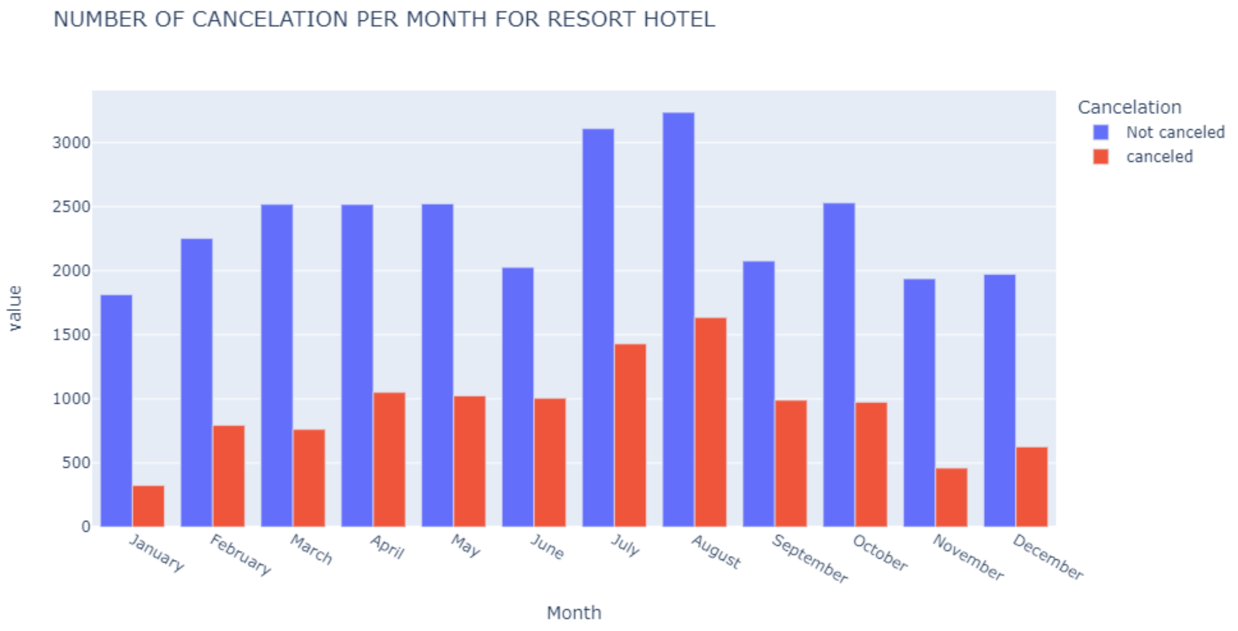
- *What is the number of guests for each time duration (per night)?*
- *What is the hotel type with more time spent?*

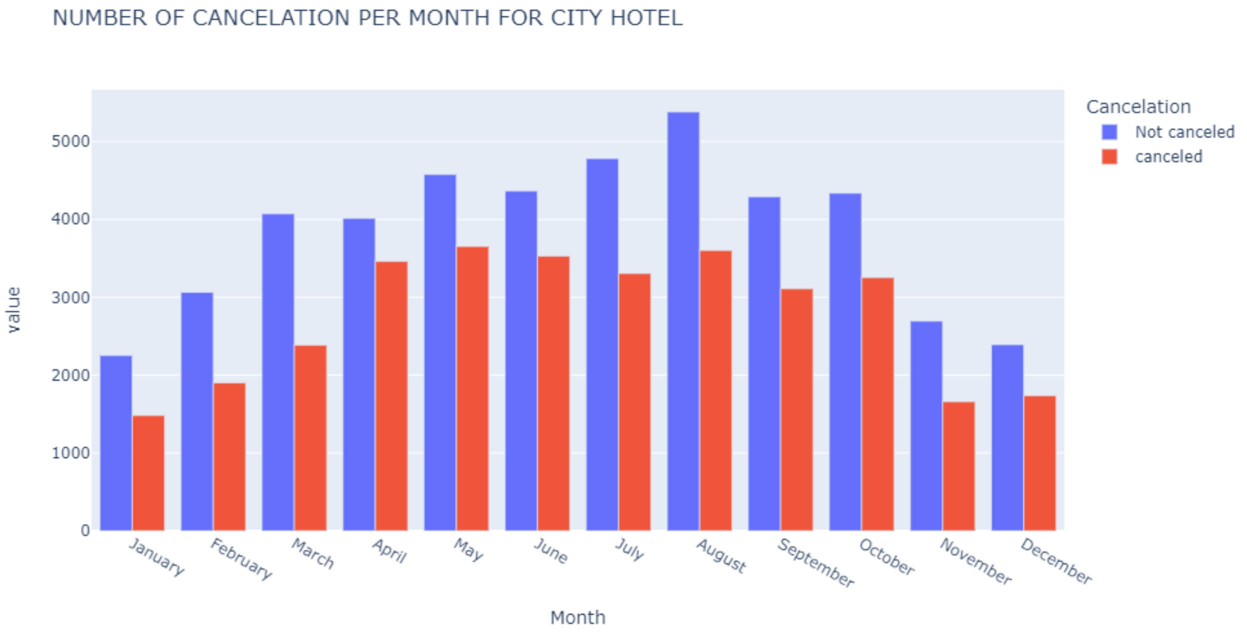
❑ *Most people do not seem to prefer to stay at the hotel for more than 1 week. But it seems normal to stay in Resort hotels for up to 15 days.*



Exploratory Data Analysis (EDA)

- *What is the number of cancellations according to the month in both hotels?*



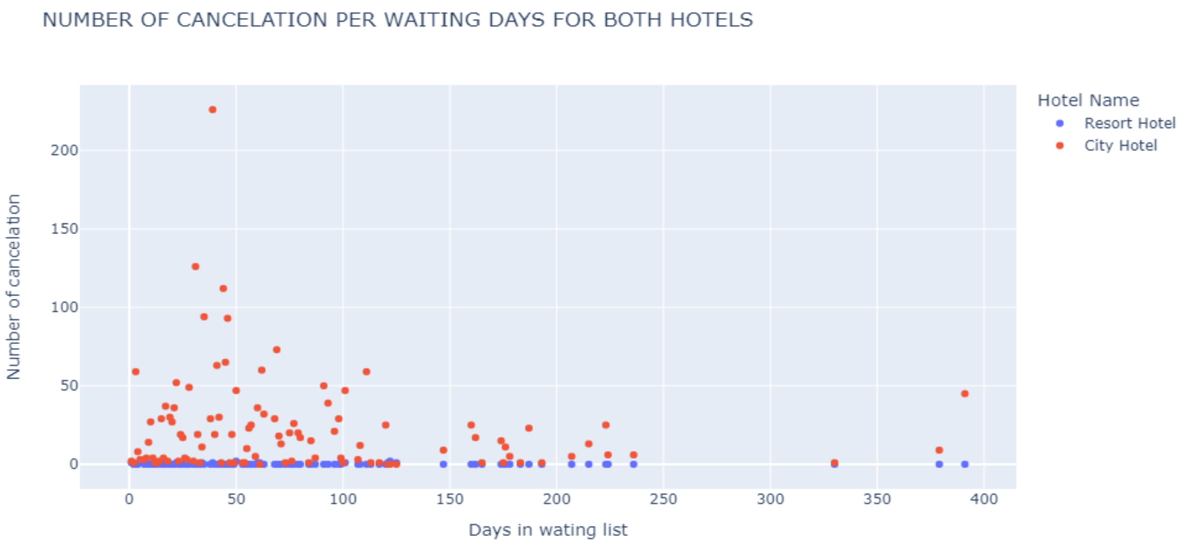


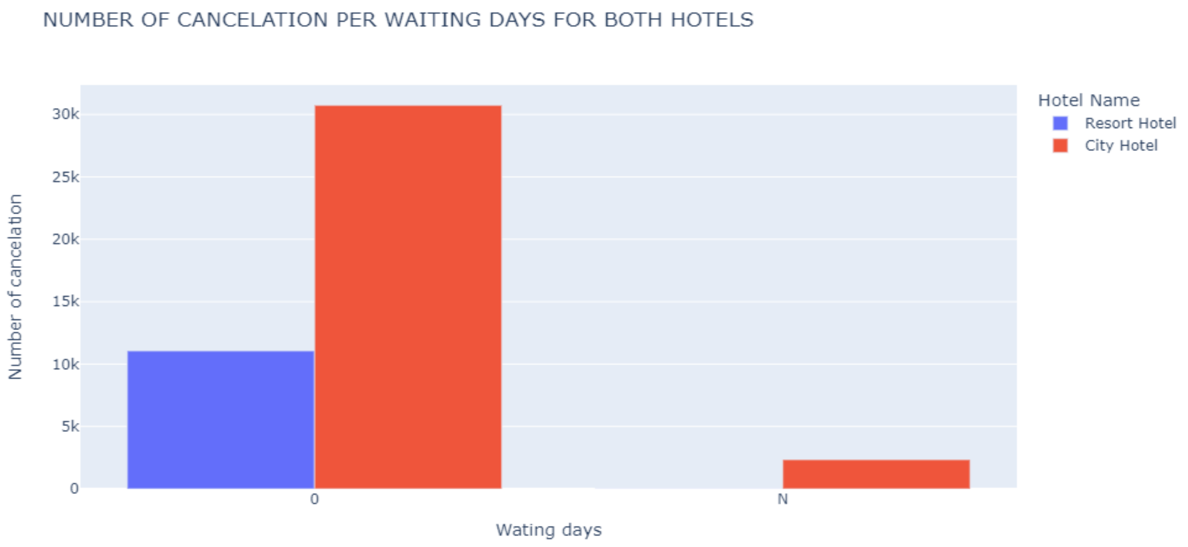
Exploratory Data Analysis (EDA)

- *What is the number of cancellations according to customer type in both hotels?*

Exploratory Data Analysis (EDA)

- *What is the number of cancellations according to waiting days type in both hotels?*
- *What is the number of cancellations of 0 waiting days and n waiting days in both hotels?*





FEATURE ENGINEERING

1. Adding New Features
2. Drop Useless Features
3. Handling The Categorical Features
4. Correlation Checking
5. Remove Features With Low Correlation With Cancelation

FEATURE ENGINEERING – Adding 4 new features

- ***is_family***

$$x = (adults > 0 \,\&\, children > 0) \mid (adults > 0 \,\&\, babies > 0)$$

$$isfamily(x) = \begin{cases} 1, & x = 1 \\ 0, & x = 0 \end{cases}$$

- ***total_customer***

$$\text{totalcustomers} = \text{adults} + \text{children} + \text{babies}$$

- ***deposit_given***

$$depositgiven(x) = \begin{cases} 1, & x = '\text{Refundable}' \mid\mid '\text{No Deposit}' \\ 0, & x = '\text{Non Refund}' \end{cases}$$

- ***total_nights***

$$totalnights = stays\_in\_weekend\_nights + stays\_in\_week\_nights$$

FEATURE ENGINEERING – Drop useless features

I created new features more expressive than this one, so I'll drop the following columns:

- *adults*
- *babies*
- *children*
- *deposit_type*
- *reservation_status_date*

FEATURE ENGINEERING – Handling the categorical features

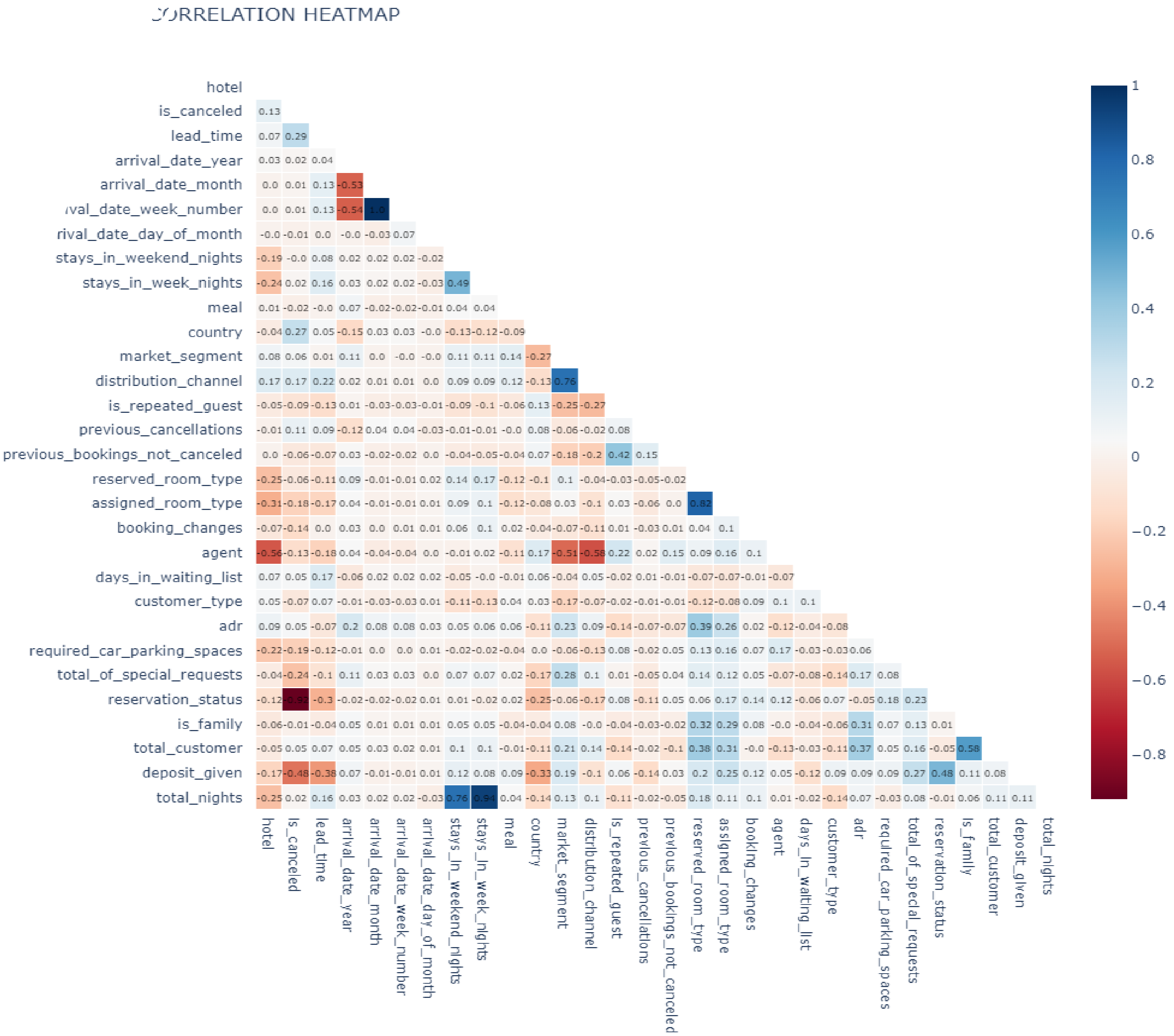
Replace the (hotel, arrival_date_month) features with numerical values manually.

Using LabelEncoder with the following columns:

- meal
- distribution_channel
- reserved_room_type
- assigned_room_type
- agent
- customer_type
- reservation_status
- market_segment

FEATURE ENGINEERING

Correlation Checking



SAVE DATASET

Before saving I dropped all features with high impact of cancellation

total_nights	*is_family*
arrival_date_week_number	*stays_in_weekend_nights*
arrival_date_month	*agent*

DATA MODELING

Normalize numerical features.

Splitting dataset to train and test sets.

Build Logistic Regression with reservation status feature

Build Logistic Regression without reservation status feature

Build Random Forest using K-Folds cross-validation

Build Decision Tree model

Build XgBoost model

Build Logistic Regression

$$J(\theta) = \frac{1}{2}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2.$$

$$h\theta(X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X)}}$$

Model Creation → Model Training → Model Testing → Model Evaluation

Results of Logistic Regression With Reservation Status Feature

```
#### The Logistic Regression (with reservation_status feature) ####

Accuracy Score of Logistic Regression is:
98.93%

Confusion Matrix of Logistic Regression is:
[[22331     7]
 [  375 12958]]

Classification Report of Logistic Regression is:
              precision    recall  f1-score   support

           0       0.98      1.00      0.99     22338
           1       1.00      0.97      0.99     13333

    accuracy                           0.99     35671
   macro avg       0.99      0.99      0.99     35671
weighted avg       0.99      0.99      0.99     35671


################################# End #################################
```

Results of Logistic Regression Witout Reservation Status Feature

```
#### The Logistic Regression (without reservation_status feature) ####

Accuracy Score of Logistic Regression is:
80.12%

Confusion Matrix of Logistic Regression is:
[[20888  1450]
 [ 5641  7692]]

Classification Report of Logistic Regression is:
              precision    recall  f1-score   support

           0       0.79      0.94      0.85     22338
           1       0.84      0.58      0.68     13333

    accuracy                           0.80     35671
   macro avg       0.81      0.76      0.77     35671
weighted avg       0.81      0.80      0.79     35671


################################# End #################################
```

Results of Decision Tree Classifier

with max_depth = 15

```
######################## The Decision Tree Classifier ###############

Accuracy Score of Logistic Regression is:
83.88%

Confusion Matrix of Logistic Regression is:
[[19815  2523]
 [ 3227 10106]]

Classification Report of Logistic Regression is:
              precision    recall  f1-score   support

           0       0.86      0.89      0.87     22338
           1       0.80      0.76      0.78     13333

    accuracy                           0.84     35671
   macro avg       0.83      0.82      0.83     35671
weighted avg       0.84      0.84      0.84     35671


################################### End ###############################
```

Results of XgBoost Classifier

With parameters as the table bellow.

PARAMETER	VALUE
Booster	'gbtree' uses tree-based model.
learning_rate	0.1
max_depth	15
n_estimators	500

```
######################## The Decision Tree Classifier ###############

Accuracy Score of Logistic Regression is:
83.88%

Confusion Matrix of Logistic Regression is:
[[19815  2523]
 [ 3227 10106]]

Classification Report of Logistic Regression is:
              precision    recall  f1-score   support

           0       0.86      0.89      0.87     22338
           1       0.80      0.76      0.78     13333

    accuracy                           0.84     35671
   macro avg       0.83      0.82      0.83     35671
weighted avg       0.84      0.84      0.84     35671


################################## End ##############################
```

Random Forest Classifier using Grid Search CV

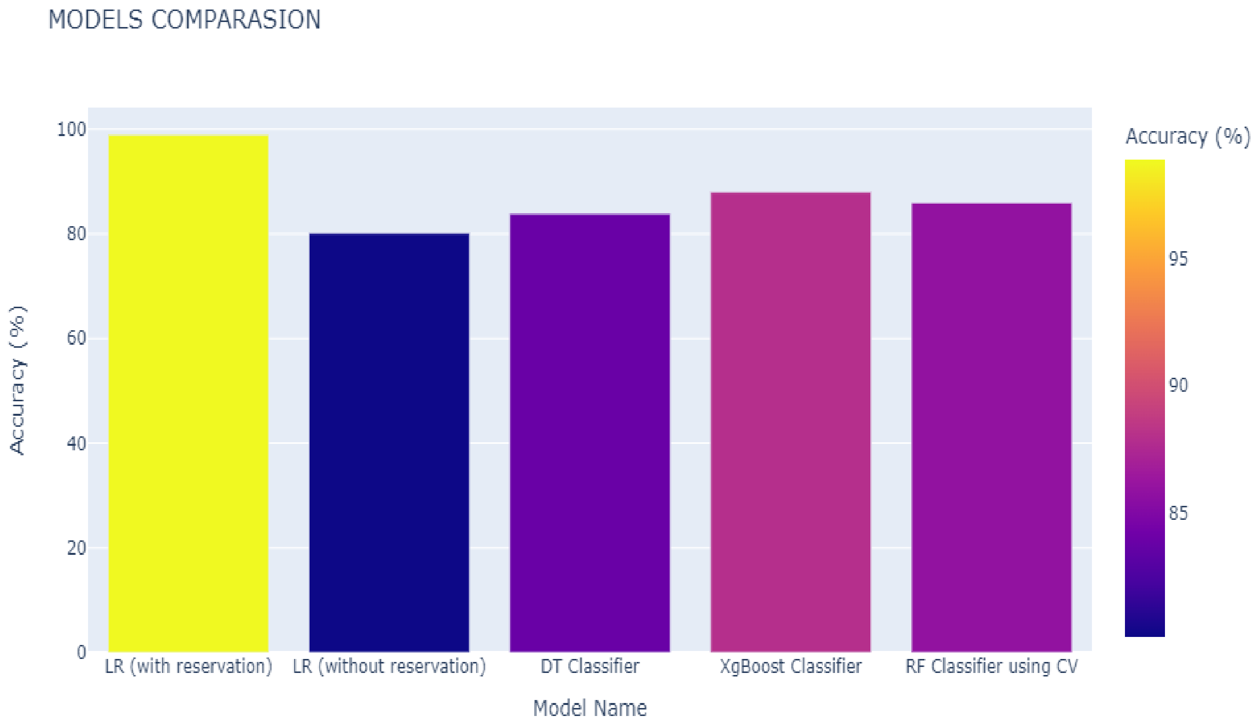
With parameters as the table bellow.

PARAMETER	VALUE
max_depth	[16,18,20]
n_estimators	[100,500]
min_samples_split	[2,5]
CV	5

```
######################## The Random Forest Classifier ########################

Accuracy Score of Logistic Regression is:
86.98%

Confusion Matrix of Logistic Regression is:
[[20863  1475]
 [ 3169 10164]]

Classification Report of Logistic Regression is:
              precision    recall  f1-score   support

           0       0.87      0.93      0.90     22338
           1       0.87      0.76      0.81     13333

    accuracy                           0.87     35671
   macro avg       0.87      0.85      0.86     35671
weighted avg       0.87      0.87      0.87     35671


################################## End ##################################
```

CONCLUSION

MODEL NAME	ACCURACY
LR (with reservation)	98.93%
LR (without reservation)	80.12%
DT Classifier	83.88%
XgBoost Classifier	87.97%
RF Classifier using CV	86.98%



Thanks
Any Question?

REFERENCES

1. Hotel booking demand datasets. (2019, February 1). ScienceDirect. Retrieved December 17, 2021, from https://www.sciencedirect.com/science/article/pii/S2352340918315191#bib2
2. Data Science Project Lifecycle | Lifecycle of Data Science Project. (2021, July 6). Analytics Vidhya. Retrieved December 17, 2021, from https://www.analyticsvidhya.com/blog/2021/05/introduction-to-data-science-project-lifecycle/
3. Pant, A. (2021, December 7). Introduction to Logistic Regression - Towards Data Science. Medium. Retrieved December 17, 2021, from https://towardsdatascience.com/introduction-to-logistic-regression-66248243c148
4. A.V.I.H.A. (n.d.). report.docx - XGBoost Algorithm In Machine learning. Course Hero. Retrieved December 17, 2021, from https://www.coursehero.com/file/79258686/reportdocx/
5. Koehrsen, W. (2019, December 10). Hyperparameter Tuning the Random Forest in Python - Towards Data Science. Medium. Retrieved December 17, 2021, from https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74