Predictive Analytics: Project Report

Morgan Gant
Rachael Doyle

April 13, 2023

Abstract

When looking to go on a trip, having a place to stay is one of the most important things to consider. Most people book hotel reservations to ensure a room(s). However, life happens, and schedules are always changing. Canceling a trip might not affect the traveler but it can negatively impact a hotel who had the room(s) reserved. With the information we have obtained from an online hotel booking channel, we will analyze the given variables, engineer more variables, build models, tune the model parameters, and evaluate the accuracy of predicting of a hotel guest will cancel their reservation or not.

Table of Contents*

Project Introduction

People can stay in hotels for many different reasons, whether it be for a week-long vacation at a nice resort, for a weekend getaway for sports, or even just a single night for a place to rest, but usually these hotel rooms are booked via a reservation. The reservation system is in place so that someone may reserve a room for however many nights they need and keep a record of the price. Some hotels have rewards systems in place to try and get more customers to consistently stay with them and use these reservation systems. Despite the many reservations hotels receive, they also are losing money in the way of cancellations. In 2019 on average 40% of the reservations in hotels were cancelled before the reservation date.

With this high number of cancellations hotels are wondering if there is a way to predict the possibility of a reservation being cancelled. What factors come into play with cancellation, how many days are leading up to the reservation, the amount of people staying in the room, or even the price of the room being stayed in. Using these features, along with a few others seen in the data set we plan to engineer variables, use hyper parameter tuning, and build models that will predict the likelihood of a reservation being cancelled. The goal of this project is to build a model that will best predict the likelihood of a hotel reservation being cancelled, and show which factors are the primary cause for these cancellations.

https://www.hotelmanagement.net/tech/study-cancelation-rate-at-40-as-otas-push-free-change-policy (Shares percentage of how many reservations are cancelled)

Data Description

For this project, a dataset of hotel reservation information was analyzed. This dataset was taken from Kaggle, an online site that contains public data sets. This data set consists of 36,276 rows and 19 columns/variables with no missing values. Each row represents information given from a guest when making a reservation online.

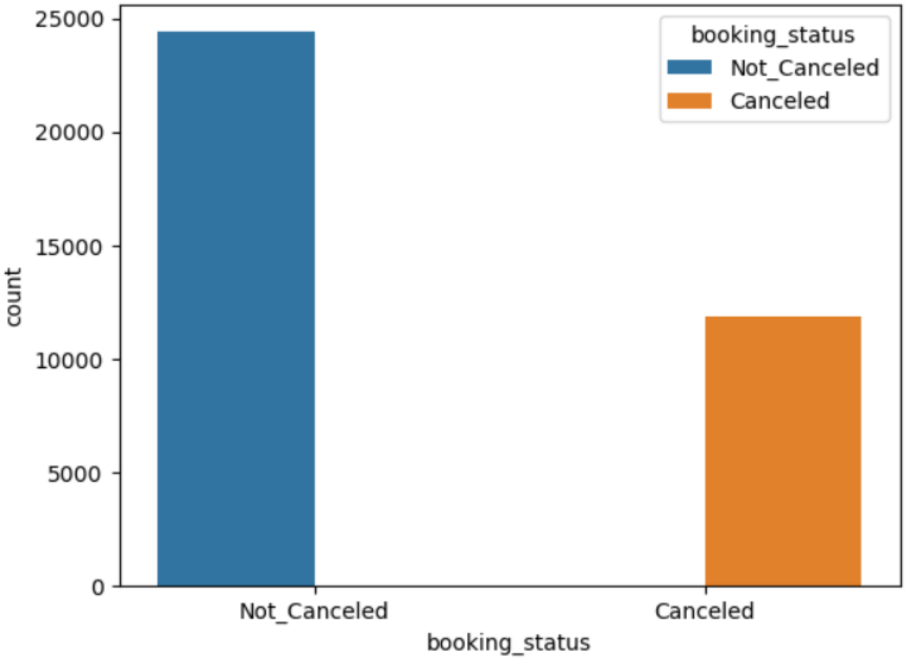
Variable:	Description:	Missing Values:
Booking_ID	Unique identifier of each booking	0
no_of_adults	Number of adults	0
no_of_children	Number of Children	0
no_of_weekend_nights	Number of weekend nights (Saturday or Sunday) the guest	0

	stayed or booked to stay at the hotel	
no_of_week_nights	Number of weeknights (Monday to Friday) the guest stayed or booked to stay at the hotel	0
type_of_meal_plan	Type of meal plan booked by the customer:	0
required_car_parking_space	Does the customer require a car parking space? (0 - No, 1- Yes)	0
room_type_reserved	Type of room reserved by the customer. The values are ciphered (encoded) by INN Hotels.	0
lead_time	Number of days between the date of booking and the arrival date	0
arrival_year	Year of arrival date	0
arrival_month	Month of arrival date	0
arrival_date	Date of the month	0
market_segment_type	Market segment designation.	0
repeated_guest	Is the customer a repeated guest? (0 - No, 1- Yes)	0
no_of_previous_cancellations	Number of previous bookings that were canceled by the customer prior to the current booking	0
no_of_previous_bookings_not_canceled	Number of previous bookings not canceled by the customer prior to the current booking	0
avg_price_per_room	Average price per day of the reservation; prices of the rooms are dynamic. (in euros)	0
no_of_special_requests	Total number of special requests made by the customer (e.g. high floor, view from the room, etc)	0

booking_status	Flag indicating if the booking was canceled or not.	0

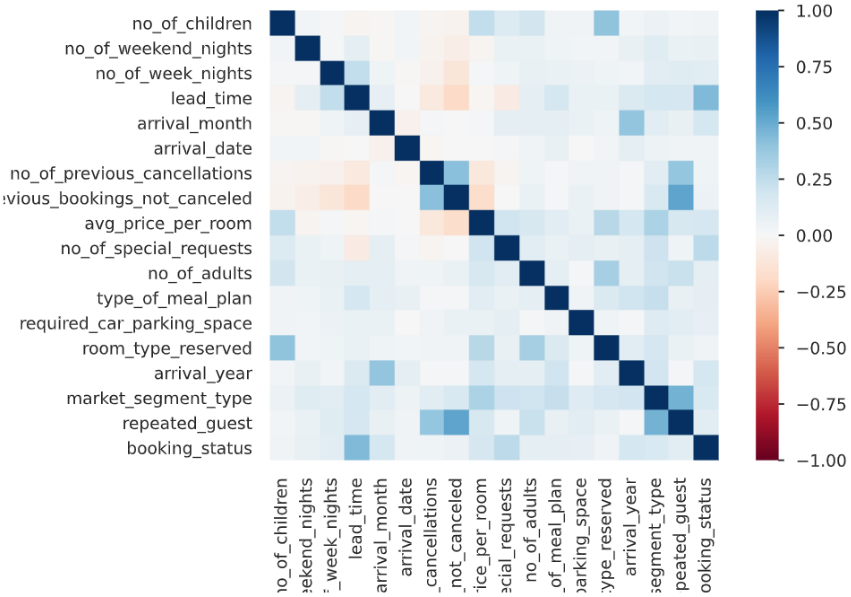
Exploratory Analysis

Figure 1



The above graph displays the target variable we are focusing on, the number of reservations that are cancelled or not cancelled. There is almost double the number of non-cancellations compared to cancellations, telling us the data is imbalanced.

Figure 2



The above chart displays the correlation values between each of the variables. Most of the data is not correlated, which is seen in the lightly shaded boxes, however there are a few correlations that may be worth noting, as seen with the darker shaded boxes.

Figure 3

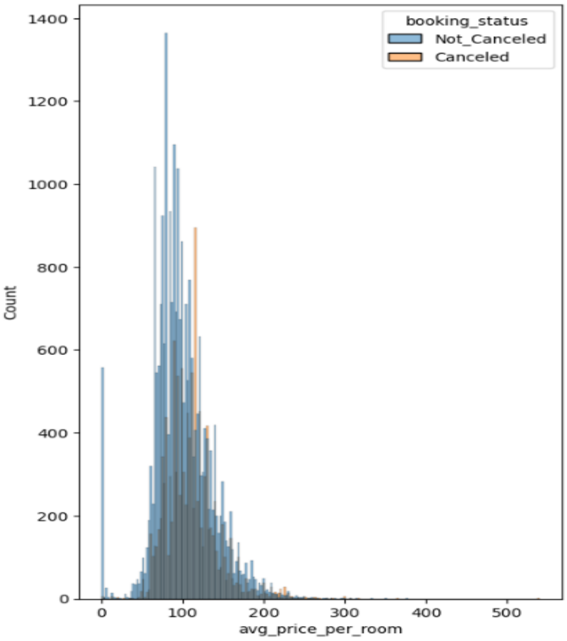
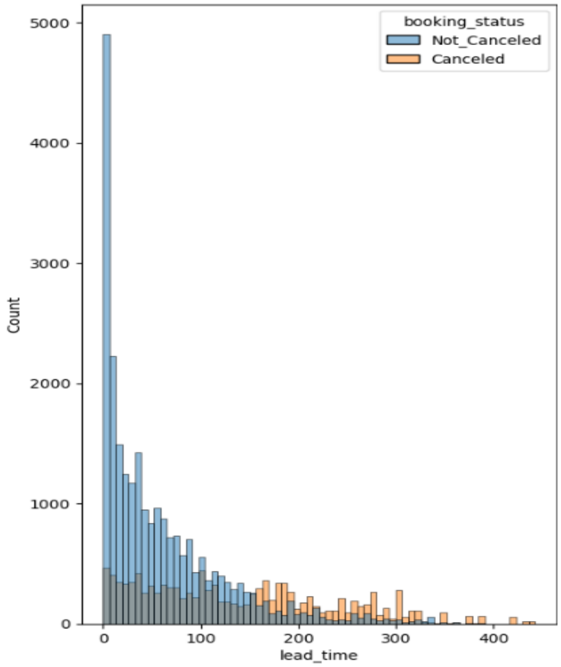


Figure 3 shows the distribution of the average price per room and the count of how many reservations were cancelled and not cancelled, corresponding with the prices. This is slightly skewed right, with most of the data being in the range 50 to 200 for the average room price.

Figure 4



The above graph shows the distribution of lead time and the count of how many reservations were cancelled and not cancelled. As a reminder, lead time is the amount of time the reservation was scheduled before the day it was needed. The data is right-skewed, as lead time increased there were more cancellations than non-cancellations. This variable would be beneficial for using a boxcox transformation when engineering features.

Feature Engineering

To start, we engineered new features that may be important in creating the highest performing predictive model when predicting on booking status. The engineered variable descriptions are below:

Variable:	Description:	Missing Values:
interaction_1	Lead time multiplied by the number of special requests	0
interaction_2	Lead time multiplied by the average price per room	0

interaction_3	Average price per room multiplied by the number of special requests	0
interaction_4	Taken from a decision tree and setting stipulations for interaction_2, number of special requests and market segment type online	0
lead_time^2	Lead time squared	0
no_of_weekend_nights^3	Number of weekend nights cubed	0
par_child	Number of adults multiplied by the number of children	0
diff_night	Difference between number of weekend nights and weekday nights	0
market_segment_type_Online	Dummy variable created by a type of market segment in the data set	0
market_segment_type_Offline	Dummy variable created by a type of market segment in the data set	0
market_segment_type_Corporate	Dummy variable created by a type of market segment in the data set	0
market_segment_type_Complementary	Dummy variable created by a type of market segment in the data set	0
market_segment_type_Aviation	Dummy variable created by a type of market segment in the data set	0
room_type_reserved_Room_Type 7	Dummy variable created by a type of market segment in the data set	0
room_type_reserved_Room_Type 6	Dummy variable created by a type of market segment in the data set	0
room_type_reserved_Room_Type 5	Dummy variable created by a type of market segment in the data set	0
room_type_reserved_Room_Type 4	Dummy variable created by a type of market segment in the data set	0
room_type_reserved_Room_Type 3	Dummy variable created by a type of market segment in the data set	0
room_type_reserved_Room_Type 2	Dummy variable created by a type of market segment in the data set	0

room_type_reserved_Room_Type 1	Dummy variable created by a type of market segment in the data set	0
type_of_meal_plan_Not Selected	Dummy variable created by a type of market segment in the data set	0
type_of_meal_plan_Meal Plan 3	Dummy variable created by a type of market segment in the data set	0
type_of_meal_plan_Meal Plan 2	Dummy variable created by a type of market segment in the data set	0
type_of_meal_plan_Meal Plan 1	Dummy variable created by a type of market segment in the data set	0

Hyper Parameter Tuning

Grid Search CV

Model:	Tuning Paramters:
XGBoost	'colsample_bytree': 1, 'gamma': 0.3, 'learning_rate': 0.01, 'max_depth': 7, 'min_child_weight': 5, 'n_estimators': 500, 'subsample': 0.8
Random Forest	'max_depth': 7, 'min_samples_leaf': 5, 'min_samples_split': 15, 'n_estimators': 500
Gradient Boosting	'learning_rate': 0.01, 'max_depth': 7, 'min_samples_leaf': 7, 'min_samples_split': 10, 'n_estimators': 500

Randomized Search CV

Model:	Tuning Paramters:
XGBoost	'colsample_bytree': 1, 'gamma': 0.3, 'learning_rate': 0.01, 'max_depth': 7, 'min_child_weight': 5, 'n_estimators': 500,

	'subsample': 0.8,
Random Forest	'max_depth': 7 'min_samples_leaf': 7, 'min_samples_split': 15, ' n_estimators': 100,
Gradient Boosting	'learning_rate': 0.01, 'max_depth': 7, 'min_samples_leaf': 5, 'min_samples_split': 10, 'n_estimators': 500

Optuna

Model:	Tuning Paramters:
XGBoost	'n_estimators': 1392, 'min_child_weight': 5, 'learning_rate': 0.08125570672227159, 'gamma': 4.428304894029277, 'subsample': 1, 'colsample_bytree': 1, 'max_depth': 10
Random Forest	'n_estimators': 142, 'min_samples_split': 6, 'min_samples_leaf': 9, 'max_depth': 10
Gradient Boosting	'n_estimators': 1957, 'min_samples_split': 23, 'min_samples_leaf': 12, 'max_depth': 10

Using hyper-parameter dictionaries and the XGBoost, Random Forest, and Optuna frameworks we found the above hyper parameters to be optimal for each of the model types using F1 score to evaluate results. Now we plan to use these hyper parameters to build the final models for predicting on the testing set, and eventually to build an ensemble learner.

Predictive Modeling/Ensemble Method

Figure 5

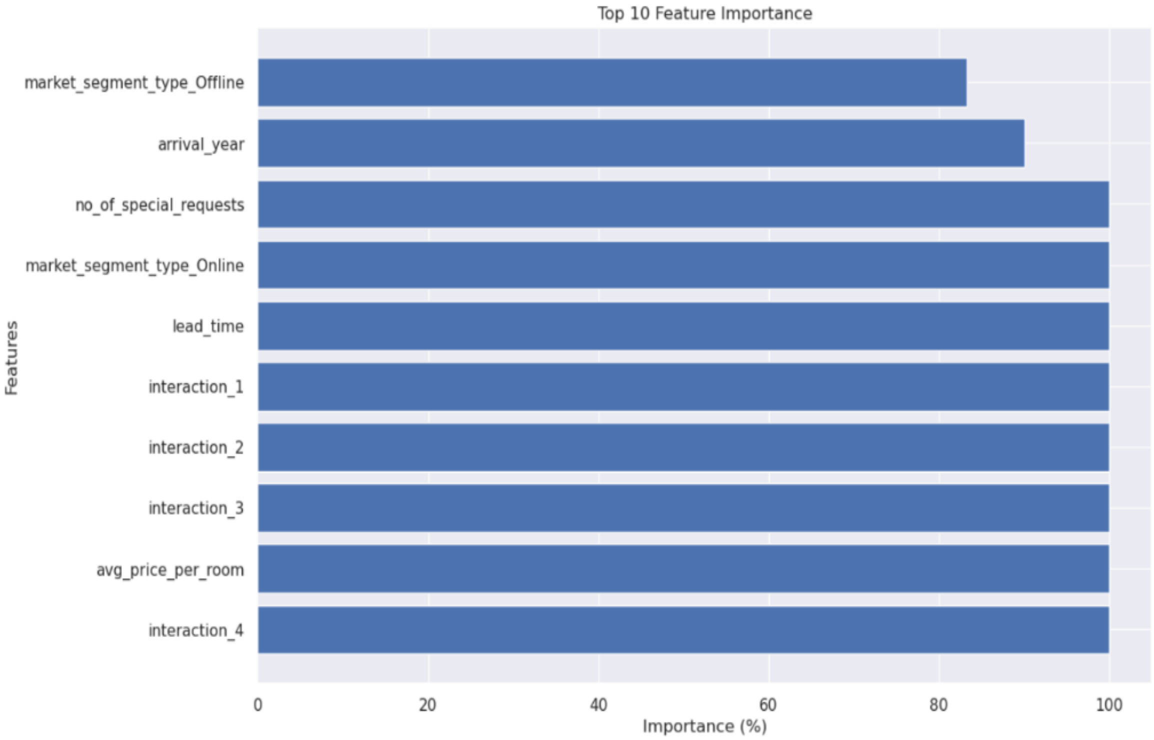
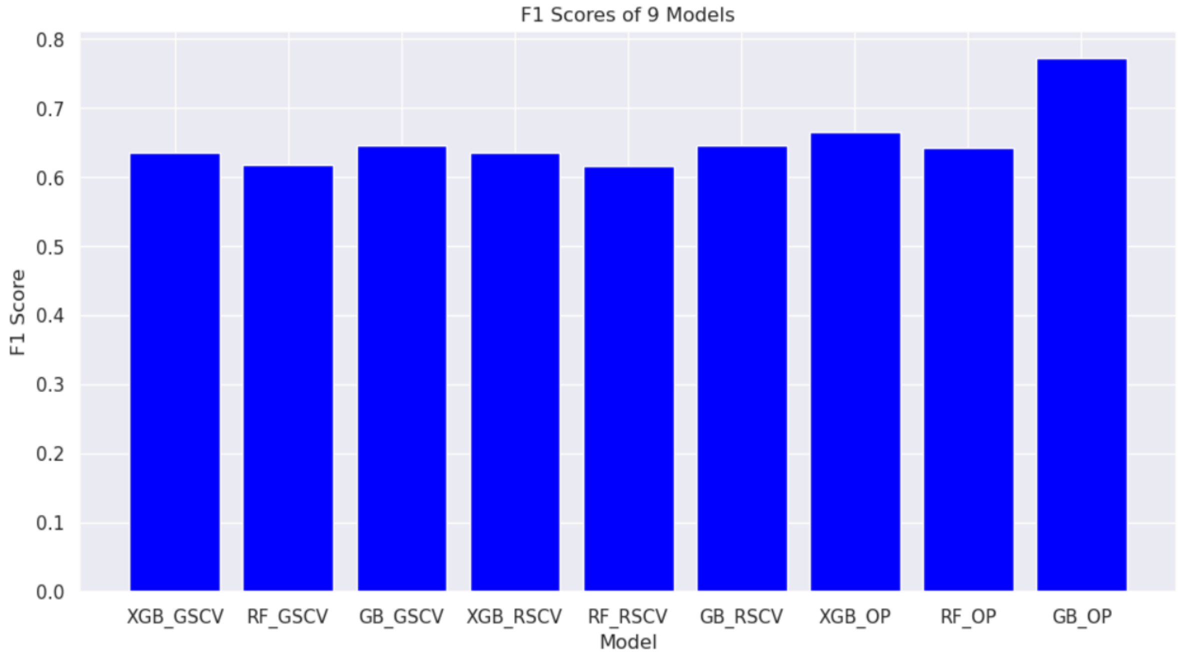


Figure 5 shows the computed average of top 10 variable appearances using RFECV and XGBoost, Random Forest and Gradient Boost as the estimators. In the final modeling/ensemble stage we used the top eight features to build the models. We did XGBoost, Random Forest Classifier, and Gradient Boosting models. With these models we predicted on the validation data set and will use these combined likelihoods to create an ensemble. We will then use this ensemble to predict on the testing data set. All results are shown below:

Figure 6



Based on the above chart we see that we have F1 scores varying from lower .6 values to upwards of .75 values. This value is seen with the Gradient boosting model done in Optuna, therefore this is our best model overall.

Conclusion

Because of the money lost from hotel reservation cancellations, the possibility of having a machine learning model to predict the likelihood of cancellations is huge. Throughout this project we followed the process of exploratory analysis, variable engineering, hyper-parameter tuning, and creating varying models to determine reoccurring patterns in the cancellations data set, in hopes of creating a model that will successfully flag potential reservation cancellations. Our best model without running the ensemble (that will be done in the final report), was using the Gradient Boosting Classifier and Optuna framework to predict on the variable booking status. It had an F1 score of .77. This isn't very good considering 1.0 is the best but we're hoping after the ensemble we can improve it!

Implications and Further Questions

One of our first findings was when we were doing feature engineering, we wanted to use the BoxCox transformation on the variable lead_time but we realized that it contained zeros and the logarithmic function used in the transformation is not defined for these values. So, we decided to engineer a feature from that variable by squaring it to see if it provided any significance for our models and status on booking. Throughout the modeling process we ran into an issue with how our y_train variable was being stored so we had to use an encoder to change the way it was stored so it could be executed. We aren't satisfied with overall performance but if we had more time to experiment with the variables, we could try to increase the performance of predicting if an individual will cancel their hotel room.