ZE GERMANS LIMITED

HOTEL C
H2

Group G - Authors:

Maximilian Maukner, m20200645

Steffen Hillmann, m20200589

Ehsan Meisami Fard, m20201050

GitHub Repository:

https://github.com/ehsanmeisami/cancellation-prediction

Table of Contents

1. Introduction

In the hotel industry, a major part of revenue management is to govern the bookings. A booking or a reservation represents a contract between a customer and the hotel. This contract obligates the hotel to give the customer the right to use the service in the future at an agreed price and most of the time with an option to cancel the contract. Due to the digitalisation of the reservation process and the appearance of Online Travel Agencies, abbreviated as OTAs, over the past decade, all of the hotels have gained a massive market exposure which might consequently increase their revenue. However, this has also led not only to immense price competition among hotels but also to competition concerning the cancellation policy. The price and cancellation policy competition among hotels has made it easier for customers to make one or multiple bookings for the same trip and still look out for other cheaper options as they have the right to exercise their option to cancel their reservation without any sunk costs or other type of consequences.

Overall, there are two main approaches to strategize against cancellations. First is to overbook the rooms to a certain given hotel knowing its cancellation rate. Yet in a case of overestimation of the cancellation rate, the customer ends up without a suitable room to stay and this can lead to trust loss and social reputation damage.

Moreover, the second approach is to implement more strict cancellation policies which might ultimately lead to a decrease in market share and ultimately a loss in demand. As a result, a better prediction of the net demand and better pricing and overbooking policies as well as classifying high cancellation likelihood bookings might lead to better overbookings and thereby increase the total revenue. Moreover, the hotel can exercise preventive methods to diminish the chance of a customer cancelling its booking by offering them a special offer on top of the current settled booking.

In this case, we deal with Hotel chain C which has been highly impacted by the amount of cancellations. Hotel C is a chain with resort and city hotels in Portugal and does not differentiate itself from other independent and non-independent hotel chains. It consists of 2 hotels, hotel 1 and hotel 2 which are named as H1 and H2, respectively. H1 has a current cancellation rate of 27.8% while the H2 has a cancellation of 41.7% and thereby impacted more severely from this issue. In this respect, the manager of the H2 has set a goal to diminish the cancellations of H2 down to 20% and has given us the task to evaluate the possibility of developing a predictive model that is able to project the net demand of the city hotel H2.

In this report, we lay out the steps taken to build a predictive model that identifies reservations that have a high chance of cancellation. Firstly, we illustrate the business and machine learning objectives of this project. Secondly, we move on to the machine learning process which entails the data cleaning and preparation as well as feature engineering and selection. Subsequently, these steps enable us to start with the model creation and assessment. Moreover, we evaluate and weigh up the output of the chosen model and refer the results back to the initial business and machine learning objective. Lastly, we make a few recommendations on how to strategize for deployment including the necessary steps and how to perform them.

2. Business Understanding

2.1. Business Objective

Hotel chain C is encountering a high rate of cancellations after reservation, especially H2. Due to the high costs of the cancellations for the hotel chain, its management has decided to find a better approach to reduce the cancellation by forecasting and identifying the customers that have a high chance of cancelling their reservation. The ultimate goal of the management is to reduce the cancellation rate of H2 down to 20% while as of now approximately 41.7% of the total reservations in H2 lead to a cancellation. Moreover, as a result, at the end of the project, the management of the hotel chain should be able to differentiate and identify reservations that have a high chance of cancellation. Furthermore, the identification of possible booking cancellations would lead to better assessment of the net demand which calculates itself as total cancellations out of the existing bookings subtracted from total bookings. The better prediction of the net demand ultimately maximizes the potential revenue that H2 is able to generate and helps the management to optimize and reform hotels reservation policy. Lastly, through exploratory data analysis numerous insights were gained from the reservation which were cancelled that can be used for further business implications.

2.2. Situation Assessment

The dataset provided by the client has 31 features which consists of information from each customer that made a reservation in the past in H2. The total sample of the reservation is 79330. The data provided consists of information about the reservation itself, such as, lead time, arrival date and the type of room reserved to name a few. Despite useful information about the bookings made, the data lacks socio-demographic information regarding the customer who has made the reservation. We highly recommend Hotel chain C to increase the amount of features collected from the customers who made the booking.

2.3 Machine Learning Goals

Subsequently, we translate the business objectives into a rather more technical data mining objective. In this case, the data mining objective of this project is to create a predictive model that classifies the incoming hotel bookings as either “cancelled” or as “not cancelled”. This enables the hotel manager to accurately predict the net demand and improve cancellation policies and overbooking strategies.

This classification is represented by a binary variable “IsCancelled” where the values are 1 for “cancelled” and 0 for “not cancelled”. Furthermore, we highly emphasize the metric Recall, which is defined as the proportion of actual positives that were identified correctly. Recall as well as the F1 score are the main metrics that we utilize to measure and evaluate our predictive model and thereby define them as our machine learning success criteria. However, it is also essential not to neglect other important metrics such as Precision, Training and Test accuracy.

3. Machine Learning Process

3.1. Data Understanding

The investigated dataset consists of 79330 observations with 31 features of which 13 are categorical variables and 18 are numerical variables. In this predictive analysis, the target is the binary variable, “IsCanceled”. Each observation represents a hotel booking. The comprehensive bookings due to arrive between the 1st of July of 2015 and the 31st of August 2017, including bookings that effectively arrived and bookings that were canceled.

3.2. Data Preparation

3.2.1. Data Cleaning

As can be seen in the output, most of the variables are categorical. From the article "Hotel booking demand datasets" written by Nuno Antonio, Ana de Almeida and Luis Nunes, it can be assumed that the dataset does not contain any missing data [1]. However, in some categorical variables, such as Agent or Company, “NULL” is presented as one of the categories. This should not be considered as a missing value, but rather as “not applicable”. Therefore, these reservations can be considered as purchased directly by the customers, without any intermediary organizations such as agencies. It was decided to drop the characteristic "Company", as this characteristic has more than 95 percent missing data and therefore does not provide any relevant information.

As the source mentioned that no missing value should exist, it is surprising to find observations with missing values for the Children- (4), Country- (24), MarketSegment- (2), DistributionChannel- (4) feature. Since these only represent less than 0.03% of the data, it was decided that it was safer not to use these observations in our analysis and modeling. As a result, the Children, MarketSegment and DistributionChannel observations with missing values were deleted. The missing values for the Country feature have been filled with the most frequent value of the column. Although the original authors have extensively pre-processed the data for modeling, the data set is not completely clean. After looking closely at each column, it was found that some values were filled with spaces. A function was used to delete unnecessary spaces in each column. The ADR feature refers to the average price per night of the reservation. Therefore, 1208 observations that had a value smaller than zero were deleted.

In addition, 167 observations were deleted where the value for "adults" and "children" was 0. It is possible that the data contains observations where the feature “Adult” equals 0 but have values for children. Furthermore, it is essential to note that the dataset contains duplicates. However, it is possible that multiple bookings with the same features were made on the same day. Since there is no feature like "booking ID", it is not possible to say with certainty that these are real duplicates, which makes the deletion of these "duplicates" questionable. Moreover, the duplicates display a representation of the population and enable the model to generalize for the population and therefore they shall not be removed from the dataset.

The features “DepositType” and “IsCanceled” are correlated in a counter-intuitive way. Over 99 percent of people who paid the entire amount upfront (“Non Refund”) canceled their reservation. It was decided to delete this feature and the accuracy value is almost identical to the one obtained with the "DepositType" feature. It was also found that after deleting this feature, the importance of the "ADR" feature increased.

3.2.2. Exploratory Data Analysis

Before jumping into feature selection and engineering, it is important to extract as much knowledge as possible from the dataset first and look out for unanticipated findings. It is critical to analyse features that could have an impact on the hotel reservation policies as well as booking cancellations. Thus, we focus on several features and try to plot them against our target variable which is either “cancellation” or “not cancelled”. For further exploratory data analysis, the respective jupyter notebook entails numerous plots and interesting findings which you can find in the github repository.

As seen in *Figure 1*, the parking space demanded and the corresponding cancellation rate is illustrated. Here, it is observed that from all bookings that required a parking space, there is not a single one that has been cancelled. We can conclude that bookings that require a parking space will most likely be confirmed and thereby not be cancelled. Moreover, as it is shown in *Figure 2*, previously cancelled bookings have a cancellation rate of 93%. On the other hand, we can assume that a booking that has been cancelled before is more likely to be cancelled again.

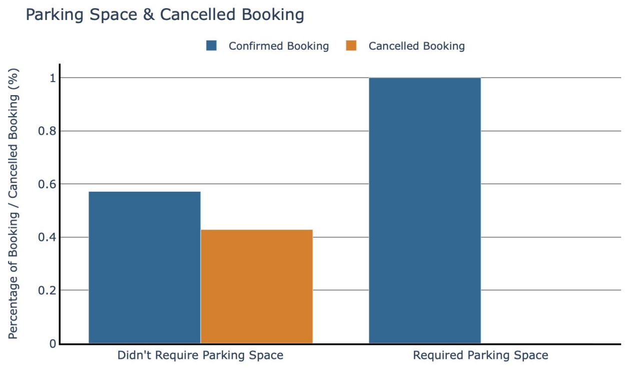


Figure 1: Parking Space & Cancelled Bookings

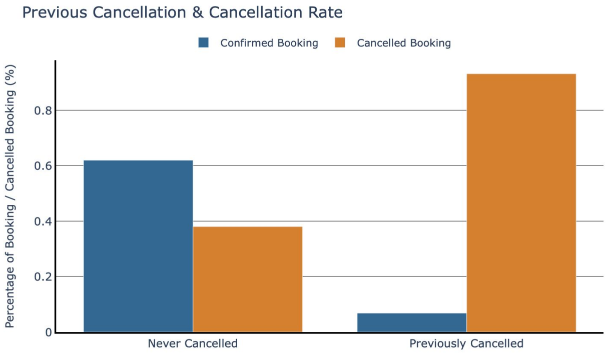


Figure 2: Previous Cancellations & Cancellation Rate

3.2.3. EDA Recommendation

In this section, we would like to briefly touch upon the two findings found and give H2 and its management recommendation. Firstly, concerning the parking spaces, we suggest the H2 to partner up with car rental companies to offer customers a bundle of room and car for an attractive price. The notion behind this strategy is to attract more customers alike and thereby decrease the cancellation rate. Subsequently, concerning the previous cancellations, we believe that setting up a booking payment in advance should encounter this issue and moreover an implementation of a non-refundable deposit for bookings that have been previously cancelled could help to improve the cancellation rates in this respect.

3.2.4. Feature Engineering

While comparing the data types to the expected data types, 2 discrepancies were found. The feature "Children" as well as "ResevationStatusDate" should have the datatype integer and datetime, respectively. Therefore, the type of these two features was changed. The original dataset provides an arrival date with day, month, and year each in a separate feature. For analysis purposes, it is easier to have all of those elements combined into one feature called "ArrivalDate". Next, the number of total days was calculated by adding the weekday and weekend stays to obtain information on the length of the customer booking. After that, features containing 0 days were checked and it was concluded not to delete these observations, since it is possible to book a hotel room only for one day and thus the arrival and departure dates are the same. Furthermore, the features "RequiredCarParkingSpaces", "PreviousCancellations", "BookingChanges" were simplified into 2 segments since these columns contain a high cardinality. To be more precise, 87 percent of the customers never change their booking, 97 percent of the customers do not need a parking space and 93 percent have never cancelled their booking previously.

3.2.5. Feature Selection

Once the optimal model with the ideal parameters is found, another major decision arises, namely the feature selection. By applying feature selection, we are able to reduce model complexity, training time and overfitting while increasing the generalization ability. However, this might lead to an information loss which causes the models to perform worse than before.

Despite the advantages of such technique, it is quite vital to consider the performance of the model in terms of accuracy and thus a certain reasonable trade-off should be considered. As a result, feature selection techniques will be implemented to gain insights regarding the reservations yet it will not be utilized in our model due to the previously mentioned reasons. A useful tool for this is the RFECV module of the sklearn.feature_selection library, which identifies the most important features and the optimal amount. In order to analyze the selected features as accurately as possible, several models were selected, including the Permutation Based Feature Importance, as well as the built in feature selection from Random Forest. After a closer look, it was found that the selected features of the different models overlap a lot so 20 features were selected finally which can be found in the jupyter notebook. In the end, Permutation Importance was used because it is model independent and works well with algorithms that are not from Scikit-Learn, such as XGBoost, LightGBM.

3.3. Modeling

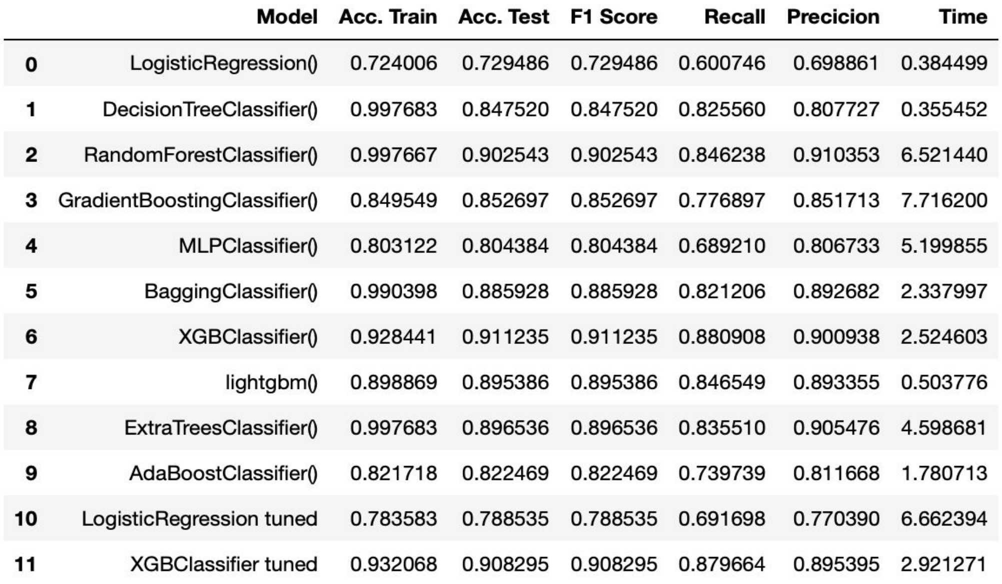
3.3.1. Model Selection

Model selection was performed by running the same data through some of the most common classification algorithms in the Scikit-learn library, as well as LightGBM and XGBoost. LightGBM uses a novel technique of Gradient-based One-Side Sampling (GOSS) to filter out the data instances for finding a split value while XGBoost uses a pre-sorted algorithm and histogram-based algorithm for computing the best split [2]. We will examine these two algorithms in more detail later on since they provide the best results for our predictive model.

Moreover, we created a function that computes the train accuracy, test accuracy, F1 score, precision, recall, and runtime to compare all algorithms with each other. However, for the final selection one single measure will be taken into account, namely Recall which computes the ratio of positive classes correctly detected. This measure was selected since one of the business goals is to classify as many cancellations as possible correctly. As a result, the model that displays the highest recall value will be selected as our cancellation classification model.

3.3.2. Build and Assess Model

The data will be splitted into a train and a test with a corresponding size of 70 and 30 percent of the entire dataset. Further, to address the problem of potential overfitting, 5-fold cross-validation was applied for every required evaluation. Most of the models are still overfitting, however hyperparameter tuning on some of them will fix the overfitting condition. As the result of training and following testing of the models, the final scores are shown in the table below.



	Model	Acc. Train	Acc. Test	F1 Score	Recall	Precicion	Time
0	LogisticRegression()	0.724006	0.729486	0.729486	0.600746	0.698861	0.384499
1	DecisionTreeClassifier()	0.997683	0.847520	0.847520	0.825560	0.807727	0.355452
2	RandomForestClassifier()	0.997667	0.902543	0.902543	0.846238	0.910353	6.521440
3	GradientBoostingClassifier()	0.849549	0.852697	0.852697	0.776897	0.851713	7.716200
4	MLPClassifier()	0.803122	0.804384	0.804384	0.689210	0.806733	5.199855
5	BaggingClassifier()	0.990398	0.885928	0.885928	0.821206	0.892682	2.337997
6	XGBClassifier()	0.928441	0.911235	0.911235	0.880908	0.900938	2.524603
7	lightgbm()	0.898869	0.895386	0.895386	0.846549	0.893355	0.503776
8	ExtraTreesClassifier()	0.997683	0.896536	0.896536	0.835510	0.905476	4.598681
9	AdaBoostClassifier()	0.821718	0.822469	0.822469	0.739739	0.811668	1.780713
10	LogisticRegression tuned	0.783583	0.788535	0.788535	0.691698	0.770390	6.662394
11	XGBClassifier tuned	0.932068	0.908295	0.908295	0.879664	0.895395	2.921271

Figure 3: Model Performance

One can clearly see that the XGBoost algorithm performs best in terms of Recall (e.g. ratio of cancelled bookings correctly classified) with 88% while meeting the overfitting condition, since the training and test accuracies are close to each other. Further, the confusion matrix for the XGBoost classifier can be seen in *Figure 4* which enables to observe the accuracy by comparing the actual and predicted classes.

Moreover, since we know that feature selection leads to an information loss (as already mentioned in part 3.2.3.) it is vital to test the significance of this consequence. Therefore, we implemented multiple feature selections techniques to find out the most important features for our results evaluation. It is important to note that we have not used feature selection to create our model as this results in worse model performance than with all features included.

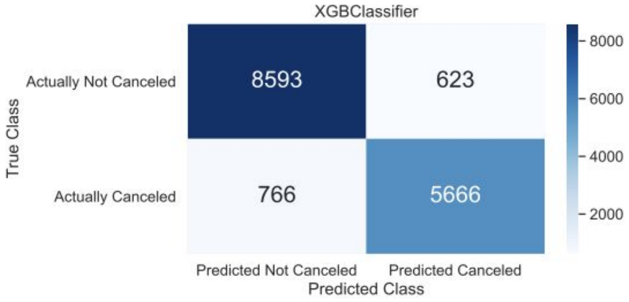


Figure 4: Confusion Matrix XGBoost

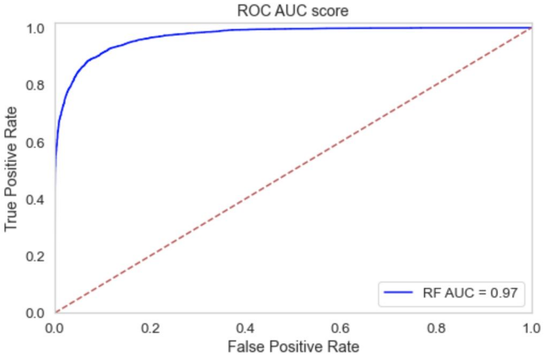


Figure 5: ROC AUC Scores

4. Evaluation

4.1. Evaluate Results

As mentioned in the previous section, out of the total 12 models trained and tested, we have decided to use the XGBoost algorithm due to its performance in terms of Recall. This classification technique has resulted in the best outcome for this specific business objective and the dataset provided from the hotel management.

By classifying reservation cancellations the management is able to better predict the net demand and thereby overbook by to a certain amount depending on the amount of incoming possible cancellations. This strategy can be utilized to boost the hotel revenue and maximize room occupation. Furthermore, along with the made recommendations, this predictive model can be used to optimize the booking and refund policies.

Moreover, features and dimensions resulting from feature importance techniques can be utilized to analyse the certain characteristics of the customers that have been vital to our predictive model.

4.2. Next Steps

As of now, the predictive model should help the management of H2 to decrease the cancellation down to 20%. Yet, it is vital to make the predictive model accessible to every staff member with sufficient technical knowledge. It is important that the model can continuously be used and operated by the client. As a result, we built and deployed an integrated ML system that can predict the chance of not cancelling through a simple web application.

5. Deployment

By implementing a web application, it is possible to check at any time whether a future customer will cancel the booking with a certain probability or not. The defined model enables the management to take action on bookings that are identified as potentially cancelable. Besides, the development of this predictive model enables Michael to reduce revenue losses resulting from booking cancellations and minimize the risks associated with overbooking. In addition to these goals, the main benefit for the management is to implement less rigid cancellation policies and restrictions without increasing uncertainty. Besides, there is also a high potential to generate more bookings and thus increase revenue.

In addition to the predictive model, existing bookings can also be classified. Hotels can use these clusters to better prepare for their guests, target them more effectively, and give them a sense of cancellation risk.

Lastly, the predictive model requires maintenance after a period of time. Due to changes in the data structure as well as in customer behaviour, the model needs to be tested and reevaluated again to sustain the level of accuracy it is currently offering.

6. Conclusion

Overall, we were able to help HC2 and its management achieve their initial business objective, which was to predict the likelihood of a booking being canceled and as a result to reduce H2's cancellation rate to 20%. Before model building and selection, we analyzed the entire data set to better understand the characteristics of a canceled booking, which would allow HC2 to develop targeted and preventative strategies. Besides, we performed data cleaning by removing outliers and data transformation by converting impractical features into meaningful and targetable information. Also, the model evaluation was performed based on accuracy and recall, as our machine learning goal was to correctly classify as many actual cancellations as possible. The XGBoost Classifier was hereby the best performing model with a recall of 88%. Furthermore, the challenge was not only to come up with a predictive model but also to develop and deploy an environment which the model can be used very easily.

7. References

[1] https://www.sciencedirect.com/science/article/pii/S2352340918315191

[2] https://towardsdatascience.com/lightgbm-vs-xgboost-which-algorithm-win-the-race-1ff7dd4917d