ST3189: ASSESSED COURSEWORK PROJECT

Hotel booking analysis and prediction

Brief Description

A journey through a hotel bookings history data set where we analyze the most important information and build predictive models aiming to predict bookings cancelation

Candidate Number: A15927

Table of Contents

Introduction:

Hotels have it difficult to manage bookings correctly, overbooking has been a common practice in hotel management with the objective of "improve the expected profit by selling the same room several times" (Birkenheuer, 2009 as cited by Zhechev, Todorov 2010). Although overbooking has a lot of advantages for revenue in the short term, but it can affect revenue in the long term with impact on the reputation and customer loyalty of the hotel (Selmi, 2007). The correct application of overbooking is crucial to the hotel industry and if not applied correctly it can be harmful for the company (Zhechev, Todorov 2010). Cancellation policies have a great impact on revenue and although strict cancellation policies secure revenue of no shows and cancellations (M. Velten, 2017) they can affect bookings. Post covid most free cancellation policies have converted on average 4.2 times better (according to rentalscaleup.com) "Free cancellation bookings have surpassed all of the other policies combined." When free cancellation policies are just 30% of their total offerings. Having this in mind we can see the importance of managing bookings correctly. With cancelation fees becoming less attractive to customers, the need of a tool that could predict cancelations becomes almost a necessity for hotels.

The aim of this project is to try to understand main trends in the hotel industry and build predictive models that might be useful for hotel companies to forecast and manage bookings correctly. I will try to predict average daily rates and focus mainly on predicting if a booking will be cancelled. Lastly, I will use PCA to see if we can get better predictions with the available features. Dimensionality is not large given the size of the dataset, but I believe that to achieve better predictive results more features should be added to the dataset. In that case, PCA could be beneficial, preparing data for better performance while keeping the model computable.

Dataset:

The selected data set is the hotel booking demand data set provided by Jesse Mostipak in Kaggle. A modification of this dataset was previously used to write an article about hotel prediction in 2019 by Nuno Antonio, Ana de Almeida and Luis Nunes. This article will be used to compare results as a benchmark in the conclusion.

Because of word restrictions the data set features won't be completely presented and explained in detail in this report, but useful information can be found in the Jupyter notebook. There are 32 features with 11 categorical variables, 20 numerical variables and 1 date and time variable with 119390 rows.

Also, many times reference made to visualizations, tables and information present in the Jupyter notebook that for word limit issues cannot be presented in the report. When this happens, I will try to reference them accordingly.

The structure of the report is the following:
The first step is loading the data and conduct some basic data preparation and data cleaning. Second, make a data exploration analysis leading to a second data preparation step, in which feature selection will be conducted. Followed by a regression analysis to predict average daily rate (referred from now on as adr) with different models, classification to predict booking cancellation and lastly PCA to do dimensionality reduction and rerunning previously used models to compare the performance.

Data Loading and Cleaning:

Loading Data: The data set is loaded (it was downloaded from Kaggle and loaded locally to the Jupyter notebook) The dataset is also saved in the GitHub repository for precaution of the data being deleted from Kaggle for any reason.

Feature analysis: Checking what the data types are and if there are any missing values. A lot of categorical variables are present. At first glance there are 12 object, 4 float64, 16 int64. In a further look into the data, it stands out that most of the variables are discrete, and most could be considered categorical. Only adr, lead time, stays in weekend nights, stays in weeknights and days in waiting list appear to be purely continuous. Almost all the rest are either categorical or discrete.

Missing Values: Most of the missing values come from two columns: company (94% of values missing) and agent (13% of values missing) so both columns where dropped. This is decided because of the large amount of missing data that if imputed could affect the model. Once these features are dropped, the rest of the rows with missing values are dropped, 432 rows (0.4%) which were missing in the column's babies and country.

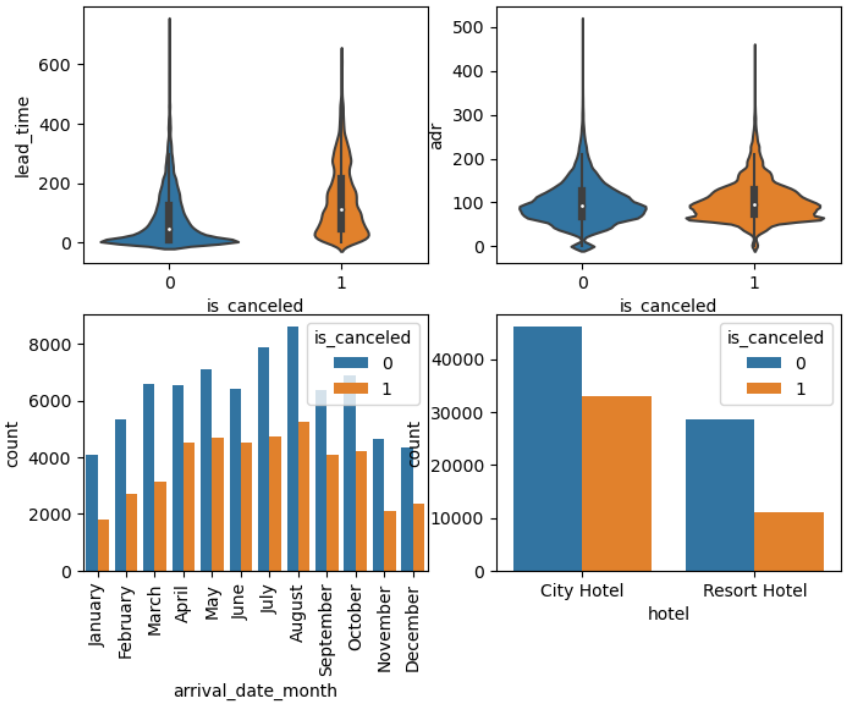
Outliers: Then it follows the search for outliers that need to be eliminated. The data is very skewed in general, but I believe only a few outliers need to be removed. This is mainly because they are wrongly imputed. The only ones removed are in the adr column where we have negative values (considered impossible) and a single value that is 50 times larger than the mean. Zero values for adr are considered since they are most complementary and could be gifts or promotions. Great variability in the lead time variable (variable that counts the number of days between the day the booking is made and the first day of the booking) is present. In this case there is no need to remove any outlier since the data seems to be consistent. With these minor tweaks there are 118896 of the original 119390 rows and 30 of the original 32 features. The next step is the exploratory data analysis in order to get better understanding of the data set and see if there is need to remove or clean any further.

Section Conclusions: The data set is a clean data set that has very few visible errors or missing values, all of which were cleaned or removed without the loss of valuable information. There are some worries about the skewness of some variables which were addressed by computing a logarithmic transformation. Unfortunately, this didn't give better results, so these columns were kept as they originally were in the dataset.

Exploratory Data Analysis and Visualization:

The cancelation rate looks high with a 37% of the reservations being cancelled. Also, the number of cancelations by deposit type are revised. Here very counterintuitive results are found, most of the non-refund type of deposits (99%) are cancelled, while I expected this number to be smaller than the no deposit type (29%). I don't know what might be causing this, it might be an error in the data or a problem with the hotel policy. Further investigation should be made to confirm the validity of this data.

Violin plot (lead time vs is canceled): In the first violin plot very informative data about the lead time is found. There are very different distributions between the cancelled group and the not cancelled group. Most of the not cancelled reservations have shorter lead time while the cancelled group has a higher median and completely different distribution. It looks like a longer lead time has a higher probability of cancelling the booking which makes sense, I expected this to happen, the longer the plan the more probable things will happen that might make you cancel.



Violin plot (adr vs is canceled): Next the exploration of the relationship between adr and cancellation with another violin plot where it doesn't seem to have great differences in the distribution and similar medians. It seems like price is not that important in deciding whether the reservation gets cancelled or not.

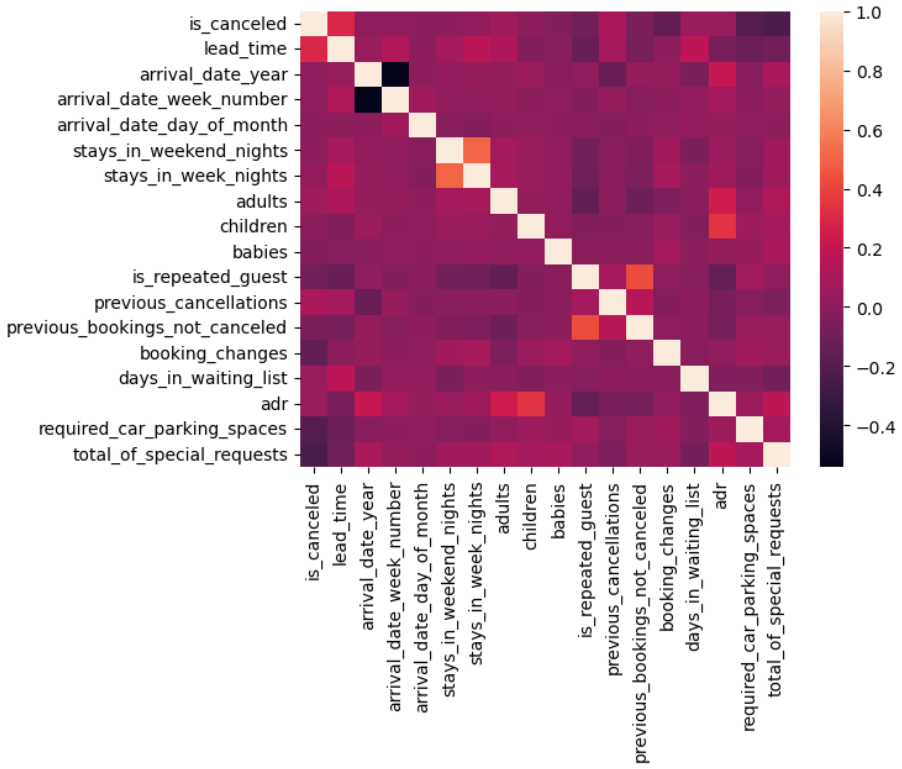
Bar plot (month of reservation vs is canceled): The months influence in cancellations is analyzed here. More bookings lead to more cancellations, but also that the months with the higher cancelled percentage are not the months with the higher booking (table calculated in the Jupyter notebook). As to the reasons of this further analysis and information (not present in the data set) should be gathered to clear this issue.

Bar plot (hotel type vs is canceled): Comparing the two types of hotels provided in the data set (city and resort) there are great differences in the cancelled percentage rate. City hotels have a 42% cancelation rate whereas resort hotel have only 28%. More research should be made to investigate the reasons of these. A speculative opinion might be that resort hotels are more linked to holidays and city hotels might be used in business trips with more probability of cancelation.

Scatter plots: Several scatter plots (in Jupyter Notebook) where plotted to see relationships between continuous variables, some comments on this are that as I expected the higher the lead time the cheaper the adr and the longer the days of the reservation the cheaper the adr.

Other plots: There are several more plots in the Jupyter notebook and more business insights can be found by digging deeper into the analysis like further looking at the box plots and doing more in-depth plotting, but because of word restrictions, there will not be commented here.

Correlation Matrix: To finish to the data analysis part, the last plot is the correlation matrix between all numerical variables.



This plot helps to identify correlation among the variables which help decide to change or drop some in order to avoid multicollinearity and perform better in the regression.

Section Conclusions: A lot of insights can be derived from data analysis, and this should be explored further. I tried to keep the data analysis part short just to be able to understand the data better and to get it prepared for prediction. There are some issues to address, and operational decisions could be made to improve the cancelation rate, like for example, trying to shorten the lead time to avoid cancellations, etc.

Feature Selection and Engineering:

General Decisions: Some decisions were made and maintained throughout the project. In this section I go through the main decisions and the reasons behind them. The variables reservation status and reservation status date where both dropped from all the prediction models. This is because this information is posterior to when a booking is either cancelled or completed. These variables get updated either when someone canceled its stay or the day the customer checks out. Including these variable goes against the objective of the model and leads to serious overfitting of the data because it is very highly correlated with the cancelled variable. Trough out both regression and classification model an iterative approach was used trying different things and modifying them. In the Jupyter notebook for simplicity only the final model was included but steps are described below. As to encoding the same approach was used for all the models. One hot encoder was used for all the categorical variables except for the country variable. Various approaches where used, first the one hot encoder was tried but produced too many features, then changing the feature composition creating a binary variable that was given the value of 1 if the people belonged to PRT category (Referring to Portugal, the country where hotels are located) and 0 otherwise. Lastly the label encoder was selected for this because it used only one feature and didn't change greatly the information provided. This was decided because the results were very similar in all the cases and simplicity was preferred.

Linear Regression Feature Selection: In the case of the regression, the firs step is to drop the target variable (adr). Then perform a regression with all the variables and get to the benchmark of 0.616 Adjusted R-squared. It follows to look at the p-values and to the previous plotted correlation matrix. As expected, there are a lot of problems with the linear regression, but it is only being used as a benchmark to compare for more complex models and to help the feature selection. There are too many variables that make it hard to interpret, also highly correlated variables that could be excluded are present. A more rigorous selection procedure like forward or backward feature selection could be used at this step but considering that the regression model is only computed as a benchmark and the little variation made in the Adjusted R-Squared by dropping some variables this method was preferred.

On the first regression performed 239 features where used. A selection of variables was made; the variable assigned room type (highly correlated with reserved room type), days in waiting list (highly correlated with lead time) and babies and country that were found not significant where dropped. By removing these variables, the data set has only 55 degrees of freedom and an Adjusted R-Squared of 0.607 only 0.009 less than the original model. With these variables the training error was 905 and the test error of 953.

Lastly, another transformation was attempted, since the target variable is skewed, logarithmic transformation was attempted to try to fix this issue and rerunning the regression, but worst results were achieved with an adjusted R-Squared of 0.472. This approach was discarded.

Random Forrest Feature Selection: In this case the feature selection process started with all features excepting the target and iterated over possibilities with different encoders and variables and once again for simplicity decide to keep the model with less variables which performs almost the same as the one with all the variables.

Rest of the models: The same procedure was performed with gboosting and all classification algorithms with always the same principal, feature importance, lowest error and simplest model arriving to a specific feature selection for each model.

Section Conclusions: Feature selection helped in some cases to achieve better prediction and computational performance.

Regression:

The motivation for this task is to predict the average daily rate. A mean squared error of 953 was achieved with our first regressions but it was improved in our random forest task (280) and got a benchmark value of 635 for our untuned gboosting.

Discouraged by the first regression I decided to go for two of the best algorithms in order to see if I could perform better. It did improve consistently the models with only working on feature selection. I will go over both cases:

Random Forest: This ensemble of trees works great in both regression and classification. Having already selected our features now tuning the hyperparameters follows. A grid search with cross validation is conducted. Here computing is an issue. Model execution took long so I had to put restrictions in the number of parameters searched and in the cross-validation folds. The hyperparameters searched for were the number of trees, the maximum depth of the trees and the minimum samples required to split a node. The results gave a slightly worse mean squared error (around 289). I expected the model to perform better with these hyperparameters set but it looks like the model is already preforming almost at its best and the default model is preferred.

Gradient Boosting: This ensemble gave poor performance in the first untuned model. Once again parameters where selected using cross-validation and then fitting the data once again. The parameters selected where the learning rate (step size), maximum depth of each tree and the number of trees. In this case it achieved a mean squared error of 333 almost half of the original model but still not as good as random forest regression.

Section Conclusions: In conclusion I might say that there are two accurate models in estimating adr. Random forest has a slight better performance and can be used to predict adr. Computation limited the model tuning, but it still has good performance.

Classification:

The motivation for this task is to predict if a booking will be cancelled or not. Different algorithms will be presented and trained, and results will be compared. In this case the measure of error will be accuracy since I want to predict all the classes correctly. This is because if only predicting one of the variables correctly is attempted our model wouldn't do the job of helping in order to perform overbooking. Predicting all classes is important in order to be able to overbook. The algorithms selected are the following:

Logistic Regression: In the untuned model an accuracy of 0.8 was achieved. Tuning the parameters penalty, C and the solver better results where accomplished. There is still room for improvement. Specially in predicting cancelled bookings (a 0.65 recall) but for a simple model result are considerably good. The result of the second model are:

	precision	recall	f1-score	support
0	0.82	0.92	0.86	15052
1	0.82	0.65	0.72	8728
accuracy			0.82	23780
macro avg	0.82	0.78	0.79	23780
weighted avg	0.82	0.82	0.81	23780

Decision Trees: The following method used was decision trees. The base benchmark of the untuned model is an overall accuracy of 0.84. One of the main advantages of this model was its fast execution compared to more complex models being able to predict in just seconds with almost the same or better accuracy than other models. These made it easier to do a more thorough grid search for the best hyperparameters. In this case tunning the hyperparameters by cross-validation lead to a similar accuracy than the default model. The resulted for the tuned model are:

	precision	recall	f1-score	support
0	0.86	0.82	0.84	15052
1	0.72	0.77	0.74	8728
accuracy			0.80	23780
macro avg	0.79	0.80	0.79	23780
weighted avg	0.81	0.80	0.81	23780

Bagging: Using the previously calculated decision tree as base model this ensemble technique aims for better performance. We also tuned the hyperparameters number of estimators, maximum

samples and maximum features by cross validation and used the model to predict for booking cancelations. Once again, this model used small computing power which allowed it to execute faster and was easier to work with. The results were very satisfactory. It performs very well in predicting with an overall accuracy of 0.88 at test. The results where:

	precision	recall	f1-score	support
0	0.89	0.93	0.91	15052
1	0.87	0.80	0.83	8728
accuracy			0.88	23780
macro avg	0.88	0.86	0.87	23780
weighted avg	0.88	0.88	0.88	23780

Random Forest Classifier: This model gave the highest prediction in the feature preparation part with an 0.89 accuracy once again we used the usual cross validation technique to search for the number of estimators, the max depth, the minimum of samplers required to split and the minimum samples per leaf. Computability was once again a problem where we had to reduce the number of parameters in order to be able to compute the model with acceptable runtime. Once again, the values very similar in accuracy.

	precision	recall	f1-score	support
0	0.89	0.94	0.91	15052
1	0.88	0.80	0.84	8728
accuracy			0.89	23780
macro avg	0.88	0.87	0.88	23780
weighted avg	0.89	0.89	0.89	23780

Section Conclusions: The result of tunning the models have much influence in some models whilst improved other models a few points. Computation was a limitation, and this could influence model performance. Results are still very good reaching test values of almost 90% and over 80% in all cases.

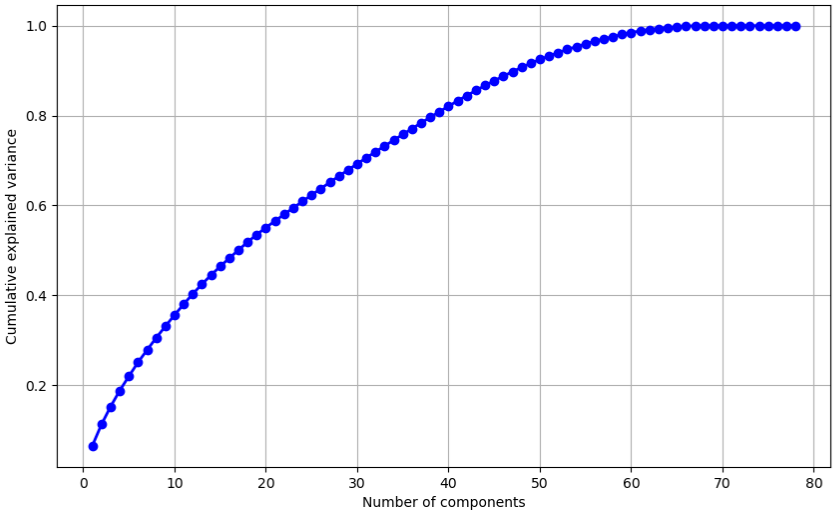
Unsupervised Learning (PCA):

Originally, I wanted to do a cluster analysis of clients in order to get more insights about them and detect hidden information. When I started to do so I realized that it didn't make much sense because there wasn't enough information about them to make appropriate clusters. I decided to adjust the objective and use PCA instead. I know the number of features is not big given the size of the data set, but the aim of this project is to give hotels a tool that can help them predict (with little information) their cancellations. So, this tool might be used by smaller datasets that might need to reduce the dimensionality of their models. I also think that more information (specifically client information) could be gathered and used to predict in which case the number of features could be large enough and PCA could also be needed. One of the main drawbacks of PCA is the loss of interpretation of its parameters as they become transformed, but I don't think this is a problem since

the models are mainly used to predict are not the ones with the simpler interpretation. I also think it is good that we can get rid of the correlations between many variables.

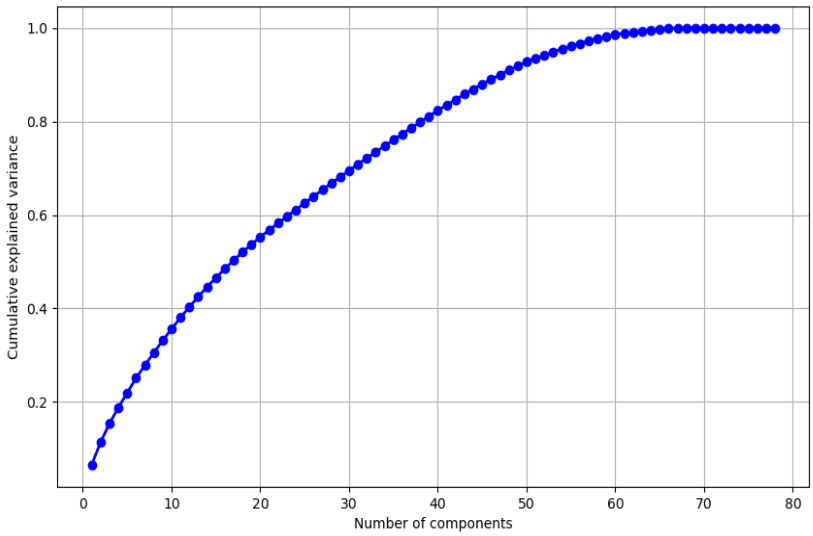
For the reasons stated above I decided to take this PCA approach. The goal is to choose the best working model (in both cases random forest as a regressor and as a classification task) and see if with PCA we can accomplish better or at least similar performance.

Random Forest Regressor with PCA: PCA is executed over all possible features using the previous encoding. For PCA to be effective it is necessary to scale the variables; the standard scaler is selected. A plot of the cumulative explained variance versus the number of components:



Around 60 number of components the reach 100% threshold of the explained variance is reached. Less components could be chosen but because the data set is still large, the first 60 were kept. Then the random forest algorithm is run and once again with cross validation the best hyperparameters are chosen. The mean squared error is 403 which is considerably worse than the best without PCA.

Random Forest Classifier With PCA: The same procedure than with regression is applied. Scale the dataset and do PCA. The only difference is the target variable (adr) used before and in this case the target variable for classification (is canceled) is used. The results are the following:



Once again it can be appreciated that the 100% cumulative explained variance is reached around 60 number of components. The first 60 components are selected and then proceed to train the model and tune

the hyperparameters with cross-validation. The model performs very similar to the random forest classifier trained with the original features:

	precision	recall	f1-score	support
0	0.81	0.96	0.88	15052
1	0.89	0.60	0.72	8728
accuracy			0.83	23780
macro avg	0.85	0.78	0.80	23780
weighted avg	0.84	0.83	0.82	23780

Section Conclusions: In the classification task with PCA it still doesn't work great on predicting cancelled bookings, but it works better than random and has an overall accuracy of 0.83 which can lead to think that PCA could be useful if the features where to be expanded or more information would be collected on customers. There is improvement to make in regression and I wouldn't recommend using PCA unless many more features are added.

Conclusion:

This project started with a bookings data set with the aim of understanding how bookings work and what improvement could be done to this process. Investigating the subject from other sources I found that cancelations are a very delicate subject and have a great impact on the profitability of the hotel business. I centered this project in trying to predict the average daily rate, whether a booking gets cancelled or not and doing a cluster analysis of the customers to understand them better. Along the way changes were made to adapt certain objectives and beliefs to what the data available was saying.

The data set with more than 100.000 rows and only 32 features was a very large one. From this data set useful information was extracted, here are some of the ideas. One concern was that algorithms did not perform much better by tuning hyperparameters. One of the main reasons for this might be that given the size of the data set we could not use a large grid of hyperparameters or use a large cross validation fold parameter and the default version of this algorithms adapted well to the data. Computability issues could be fixed by using random search instead of grid, or by working with samples of the dataset. I preferred to work with the whole data and using grid search because I see this exercise as a potential tool that other hotels could use, and they might have smaller datasets or if needed more computing power could be used to search for more parameters with more folds.

There are some relations between variables like lead time and cancelation or adr, mechanisms could be built in order to try to have shorter lead time aiming to keep cancelations down or contacting clients to confirm reservations every certain period could be beneficial to free space. Specially if the hotel has a refundable or no deposit policy. Some inconsistency in our data was found that needs further exploration like for example the cancelations for nonrefundable bookings.

In the prediction of adr with different models random forest gave us the best results. This tool could either be used to predict profitability or facilitated to potential customers so they can see what their adr might be. More could be built into this tool to create a search algorithm so customers can look

for the best combination of variables and look to minimize price given those variables, like for example searching the cheapest stay for two adults in a range of different dates.

The focus of the project was in predicting whether bookings would be cancelled. Here I tried various algorithms which gave similar outcomes, once again random forest standing out as the best one in terms of results but with other viable options if speed was considered an issue (for example decision trees gave good results with very short computation runtime). The following results raised from the best performing model:

	precision	recall	f1-score	support
0	0.89	0.94	0.91	15052
1	0.88	0.80	0.84	8728
accuracy			0.89	23780
macro avg	0.88	0.87	0.87	23780
weighted avg	0.88	0.89	0.88	23780

An overall 89% accuracy with the highest value as well in predicting almost 80% of the cancelled bookings. I find this model to perform better than what I first expected when I started the project. If hotels could consistently predict their bookings like this, they would have a great tool to work with their bookings and they could even increase profitability by having high occupancy rates managing correctly their overbookings, avoiding most risks of damaging the hotel image. I believe that this proved a success and further analysis should be pursued in this area and safe procedures could be constructed around this tool.

In the introduction I also mentioned a paper that worked with a variation of this data set in a similar task. This paper was found after the project was conducted but I don't want to miss to mention that they had results above 90% of accuracy. They worked with more complex models and a more detailed data set, they also used 10 folds in their cross validation. This motivates me that better results could be achieved and will be pursued.

Lastly, I failed to effectively cluster our clients. I believe more information should be gathered in this area in order to do so. This information could not only be useful to understand better the customers but also to reinforce the previous models. Having failed to cluster I decided to use PCA to reduce dimensionality and set the steppingstones for what the classifier should look like if more features where added. Here we performed PCA in both models (regression and classification) and found similar results in the dimensionality reduction. In both cases we could explain the 100% of variability with around 60 dimensions. I then re-run the models and found that in the regression the model performance decreased significantly. In the case of the classification task, very similar performance was achieved which is a good sign and if more features were added or a smaller dataset would be used that needs less features PCA could be a good approach.

Although I have clear that these models and conclusions might not transfer correctly to other datasets it might still be a good place to start from to work towards predicting adr and booking cancelations.

References:

Data set: (https://www.kaggle.com/datasets/jessemostipak/hotel-booking-demand).

Birkenheuer, G.; Brinkmann, A.; Karl H. (2009), The Gain of Overbooking. Job Scheduling Strategies for Parallel Processing, Volume 5798/2009, pp. 80-100

Nuno Antonio, Ana de Almeida and Luis Nunes (2019): Predicting Hotel Booking Cancellation to Decrease Uncertainty and Increase Revenue (https://www.sciencedirect.com/science/article/pii/S2352340918315191).

Rental Scale Up https://www.rentalscaleup.com/fully-refundable-cancellation-policies-see-booking-demand-at-almost-the-same-levels-as-pre-pandemic/

Selmi, N. (2007), Yield Management, a Technological Innovation in Services: Impacts on Hotels and Customers. Tourism, Mobility and Technology TTRA , Europe Conference 22 to 25 April 2007.

Todorov, Andrey and Zhechev, Vladimir (2010), The impact of overbooking on hotels' operation management; https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1698103

Velten (2017), Cancellation Policies in Combination With Scarcity and Social Proof Appeals: A study into the effects of cancellation policies and persuasion cues on consumer responses within the online booking industry; https://essay.utwente.nl/72502/1/Velten_MA_BMS.pdf