Hotel Demand Analysis and Predicting Cancellations using Machine Learning Techniques

Juilee Salunkhe [*], Akshay Bhala[**], Raj Desai[**]

Applied Data Science, iSchool
Syracuse University

Poster Presentation : https://www.youtube.com/watch?v=9CyjALbCqds&feature=youtu.be

Dash Application: https://myapphotel.herokuapp.com/

Abstract-
Revenue Management and demand forecasting in the hospitality industry is a challenge faced in this era of internet. It can help estimate the net demand forecast, increase reservation retention, and offer better cancellation policies by predicting the cancellations Using datasets from two hotels, the project aims to understand and analyze bookings cancellations, customer segmentation and satiation, seasonality, perform descriptive analysis to understand patterns, anomalies, etc. and to develop prediction models to classify a hotel booking's likelihood to be canceled or not by employing five machine learning methods (Random forest Classifier, Gradient Boosting, KNN, Logistic Regression and Naïve Bayes). Intents to effectively sift through the signals detected from variables, discover patterns and anomalies and then use that information to make predictions for guest bookings cancellations.

Index Terms- Hotel revenue management, predicting cancellations, Machine learning, Data Science

I. INTRODUCTION

Machine learning can redefine Revenue Management in the Hospitality industry. Room allocation to the right customer at the right moment with optimal price is a challenge in the hospitality industry. With last-minute cancellations and "no shows", there is loss of revenue since the capacity allocation for hotels is no longer optimal. What if the industry can predict which reservations are likely to get canceled beforehand?

Revenue Management magnifies revenues of a company by means of demand-management decisions. With new online travel agencies, websites and with unparalleled levels of data being produced daily this task of revenue managers has become impossible to do manually. Every day new data points are created by new customers searching and comparing 'n' number of hotels on the internet. With wide range of options available, customers tend to cancel bookings only a few days prior to the trip. At that point of time, managers have already made arrangements for the customers like allocating resources, rooms and other settings.

Although as a customer it may seem flexible to cancel bookings of a hotel at the last moment it can have a substantial effect on hotel revenue and will affect pricing and inventory allocations decisions negatively. Cancellations can represent 20% of the total bookings received by hotels (Morales and Wang, 2010). This value increases to 60% in the case of airport/roadside hotels (Liu, 2004). In an attempt to balance losses, hotels resort to the implementation of overbooking strategies and restrictive cancellation policies (Mehrotra and Ruttley, 2006; Smith et al, 2015; Talluri and Van Ryzin, 2005).

By predicting probable cancellations, hotel management can outline which reservations are going to get canceled which can help them to prevent cancellations by giving those specific customers some extra benefits or discounts. Also, it will help hoteliers to model price and resource allocation. Online travel agencies can jot down variables correlated with cancellations and can amend offers and policies accordingly. Using data from hotel's Property Management Systems (PMS), the paper aims to explain and illustrate how machine learning techniques can be applied on the data to predict booking cancellations. Project aims to:

1. To perform descriptive statistics, find anomalies, do exploratory data analysis and find trends, patterns in the data to give customer insights.
2. Identifying which features in hotel PMS's databases contribute to predict a booking cancellation probability.
3. Building a model to classify bookings with high cancellation probability and using this information to forecast cancellations.
4. Evaluate performance of various models and find the best three followed by hyperparameter tuning.

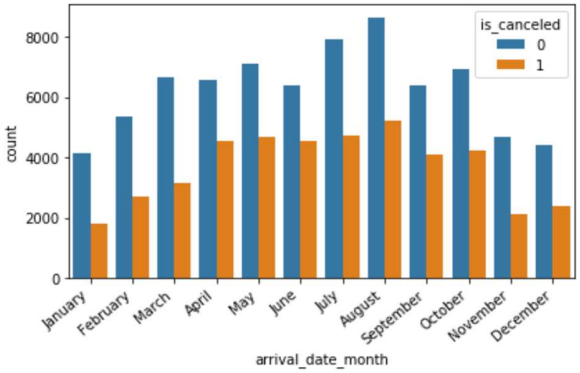

Fig.1. Arrival date by cancellations

II. LITERATURE REVIEW

Machine learning approaches are slowly attracting attention owing to their outstanding predicting power and flexibility in modeling relationships. Machine learning approaches have been widely used in predicting business failure (Gepp et al., 2010; Li and Sun, 2012; Lin et al., 2011), the stock price (Alkhatib et al., 10 2013; Tsai and Wang, 2009), a currency exchange rate (Galeshchuk, 2016; El Shazly and El Shazly, 1999), etc.

However, there has been little discussion on applying machine learning approaches in the hotel industry. Most of the applications of machine learning techniques in the industry focus on hotel online review analysis. For instance, Ma et al. (2018) calculate the effect of hotel online reviews with the user-provided photos using text mining techniques. Moro et al. (2017) use a support vector machine to predict customer online review scores given users profiles. Phillips et al. (2015) use Artificial Neural Network (ANN) to investigate relationships among online client reviews, hotel characteristics, and revenue per available room (RevPar). Besides online review analysis, Yang et al. (2015) use projection pursuit regression (PPR), ANN, SVM, and boosted regression to predict hotel success indicators (RevPar, profit, labor productivity, and efficiency score) given hotels location. Corazza et al. (2014) apply supervised MultiLayer Perceptron (MLP) ANN to simulate the procedure of hotel online booking. Instead of establishing a forecasting model according to the hotel's structure, the authors utilize ANN to figure out data's internal pattern based on existed customer bookings. They construct a function to respond to customers reservation requests and provide alternate solutions.

In summary, there remains a paucity of research on applying machine learning techniques on demand forecasting in hotel settings.

III. DATA UNDERSTANDING

Data was obtained directly from the hotels' PMS databases' servers by executing a TSQL query on SQL Server Studio Manager, the integrated environment tool for managing Microsoft SQL databases. Not all variables in the dataset come from the bookings or change log database tables. Some come from other tables, and some are engineered from different variables from different tables. PNR database structure was built for the airline industry and thus it does not have fields or variables that are specific to the hotel industry. Extensive domain knowledge was fundamental to understand this data and to conduct a good variable selection. A diagram presenting the PMS database tables from where variables were extracted is presented in Fig.2. Table 1 presents a list of all variables extracted.

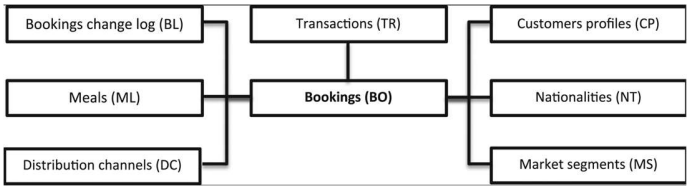

Fig.2. PMS Database

Also, for exploration purposes more features like 'Total Guests' (includes Children, Babies and Adults), 'Given same room type' (binary variable) in 0 and 1s. 1 meaning given same room type as reserved by the customer and 0 meaning not were created.

Table 1.

Variable	Type	Description
hotel	Categorical	Type of hotel. City or Resort type
is_canceled	Integer	In terms of 0 and 1.0 means no cancellation whereas 1 meaning booking canceled
lead_time	Integer	Number of days that elapsed between the entering date of the booking into the PMS and the arrival date
arrival_date_year	Integer	Year of arrival date
arrival_date_month	Categorical	Month of arrival date with 12 categories: "January" to "December"
arrival_date_week_number	Integer	Week number of the arrival date
arrival_date_day_of_month	Integer	Day of the month of the arrival date
stays_in_weekend_nights	Integer	Number of weekend nights (Saturday or Sunday) the guest stayed or booked to stay at the hotel
stays_in_week_nights	Integer	Number of week nights (Monday to Friday) the guest stayed or booked to stay at the hotel
adults	Integer	Number of adults
children	float	Number of children
babies	Integer	Number of babies
meal	Categorical	Type of meal booked. Categories are presented in standard hospitality meal packages: BO, BL and ML Undefined/SC – no meal package;BB – Bed & Breakfast;HB – Half board (breakfast and one other meal – usually dinner);FB – Full board (breakfast, lunch and dinner)
country	Categorical	Country of origin
market_segment	Categorical	Market segment designation. In categories, the term "TA" means "Travel Agents" and "TO" means "Tour Operators"
distribution_channel	Categorical	Booking distribution channel. The term "TA" means "Travel Agents" and "TO" means "Tour Operators"
is_repeated_guest	Integer	Value indicating if the booking name was from a repeated guest (1) or not (0)
previous_cancellations	Integer	Number of previous bookings that were cancelled by the customer prior to the current booking
previous_bookings_not_canceled	Integer	Number of previous bookings not cancelled by the customer prior to the

reserved_room_type	Categorical	Code of room type reserved. Code is presented instead of designation for anonymity reasons
assigned_room_type	Categorical	Code for the type of room assigned to the booking. Sometimes the assigned room type differs from the reserved room type due to hotel operation reasons (e.g. overbooking) or by customer request. Code is presented instead of designation for anonymity reasons
booking_changes	Integer	Number of changes/amendments made to the booking from the moment the booking was entered on the PMS until the moment of check-in or cancellation
deposit_type	Categorical	Indication on if the customer made a deposit to guarantee the booking. This variable can assume three categories:No Deposit – no deposit was made
agent	float	ID of the travel agency that made the booking
company	float	ID of the company/entity that made the booking or responsible for paying the booking. ID is presented instead of designation for anonymity reasons
days_in_waiting_list	Integer	Number of days the booking was in the waiting list before it was confirmed to the customer
customer_type	Categorical	Type of booking, assuming one of four categories: Contract - when the booking has an allotment or other type of contract associated to it;Group – when the booking is associated to a group;Transient – when the booking is not part of a group or contract, and is not associated to other transient booking;Transient-party – when the booking is transient, but is associated to at least other transient booking
adr	float	Average Daily Rate
required_car_parking_spaces	Integer	Number of car parking spaces required by the customer
total_of_special_requests	Integer	Number of special requests made by the customer (e.g. twin bed or high floor)
reservation_status	Categorical	Reservation last status, assuming one of three categories:Canceled – booking was canceled by the customer;Check-Out – customer has checked in but already departed;No-Show – customer did not check-in and did inform the hotel of the reason why
reservation_status_date	Categorical	Date at which the last status was set. This variable can be used in conjunction with the ReservationStatus to understand when was the booking canceled or when did the

IV. EXPLORATORY DATA ANALYSIS

Descriptive Statistics show or summarize data in a meaningful way. It is important to understand the data before modeling. Descriptive statistics do not, however, allow us to make conclusions beyond the data we have analyzed or reach conclusions regarding any hypotheses we might have made. They are simply a way to describe our data. The data was observed for central tendency like mean, median, mode, measures of variability like range, interquartile range (IQR), variance, and standard deviation.

	is_canceled	lead_time	arrival_date_year	arrival_date_week_number	arrival_date_day_of_month
count	119390.000000	119390.000000	119390.000000	119390.000000	119390.000000
mean	0.370416	104.011416	2016.156554	27.165173	15.798241
std	0.482918	106.863097	0.707476	13.605138	8.780829
min	0.000000	0.000000	2015.000000	1.000000	1.000000
25%	0.000000	18.000000	2016.000000	16.000000	8.000000
50%	0.000000	69.000000	2016.000000	28.000000	16.000000
75%	1.000000	160.000000	2017.000000	38.000000	23.000000
max	1.000000	737.000000	2017.000000	53.000000	31.000000

Fig.3. Descriptive Statistics

From the data, it was observed that the average cancellations are just 37.04% where it deviates by 48% which means there was lots of variation between cancellations which directly affected the productivity of the hotel. Also, it was observed that 75% of people asked for one special request. Almost 78% of the people book for BB meal- type i.e. Bed and breakfast and about 47.30% of market segment designation is of Online Travel Agents. Descriptive analysis showed that mean values for adults and children were higher in the Resort. This showed us that the resort hotels were a better choice for large families.

Descriptive statistics were followed by Exploratory Data Analysis. Few questions like where does the guest come from? How long do repeated people stay at the hotel? What was the hotel type preferred by customers with a long stay? How was the distribution of customers? Which market segment played a major role? and many such questions about the customers. To understand the nature of cancellations it is very important to learn about the customers who make them. Analysis was done to understand customer type, behavior, preferences, etc. Using univariate and multivariate analysis the data pointed out that about 40.87% of people came from PRT (Portugal) itself. They were followed by customers from the UK (10.20%), France (8.76%), Spain (7.21) and Germany (6.13%). Out of four customer types, bookings made by customers from transient types were the highest (75.1%). More number of cancellations were made by the total guests (including Children and babies with Adults) booking Resort Hotel. Higher number of cancellations came from Transient type customers.

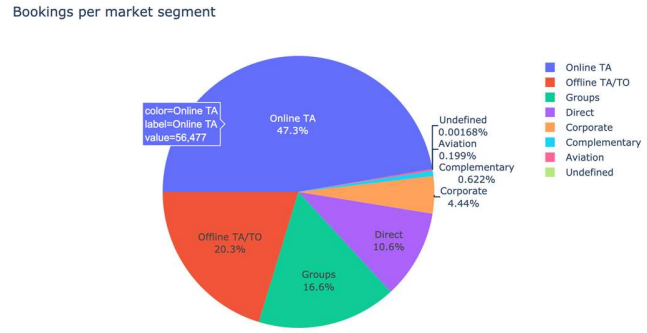

Fig.4.Pie plot of distribution of market segments

When studying customers for revenue management it is also important to know who our loyal customers are. The customers who visit more than once and are willing to pay for the same service again. The study further can be used to find out customer lifetime value and personalization of services for the sector. Revenue management offers hotels a marketing and sales advantage, as it advertises for hyper-targets the buyer persona that is likeliest to visit. After getting an overview of the data, we worked on repeated guests and found out that only 1.62% of the total guests' data are repeated guests. It can be said that the returning customers are less as compared to the new customers.

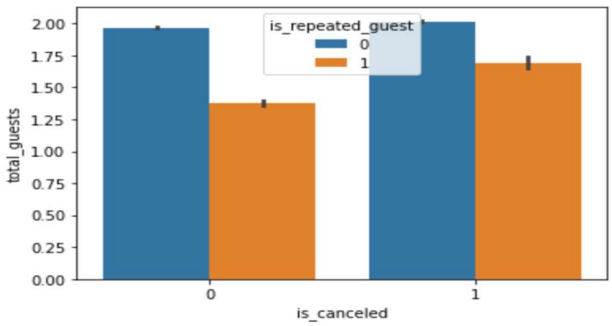

Fig.5. Total Guests v/s cancellations

Analysis showed that most of the customers on average stayed from about 1-4 days. Of those very few were repeated customers. It seems that repeated guests do not find hotel/ resort promising. Repeated guests tend to

stay more on weekend nights than weeknights. The concept of lead time is also crucial when it comes to bookings in the hotel industry. Booking Lead Time is the period of time between when a guest makes a reservation, and the actual check-in date. The general decrease of lead times and the increase of last-minute bookings certainly makes life harder for hotel revenue managers. As a shorter booking lead time can force a hotel to begin to use more costly sources of business as inventory becomes distressed and can act as a leading-edge symptom of declining cycle strength. A longer Booking Lead Time should result in higher Guest Paid ADR.

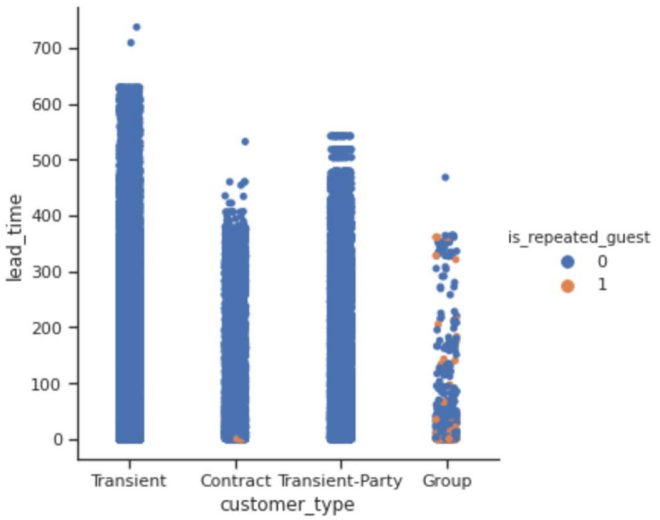

Fig.6. Customer Type v/s Repeated Guests

Also, a greater number of repeated guests were from customer type Groups. They showed less lead time leading to less ADR and hence generates less revenue for the hotel. Planning and forecasting are much easier when there is more time between booking and check-in so it's a good idea to encourage customers to book comfortably ahead of time. Reserved room type and Assigned room type were compared to see whether there were any differences in the allocation of room type and to check whether it is one of the reasons for the cancellations. About 12.49% of the customers were not given the same room type they reserved. However, it is observed that the number of cancellations was not affected by such entries. There can be other factors responsible as well as the probability of hotel assigning a better choice room type to the customer than the previously reserved room type cannot be neglected. Also, for Resort hotel, the prices are very high in the summer season during the months of May, June, July, August reaching up to 180 Euros per night. Since it's a vacation time so the customers are more likely to plan their vacations in the rest of the year during long weekend where the prices are in the range 50-80 Euros per night.

V. DATA PRE-PROCESSING

Real-world data coming from different sources is generally incomplete, noisy and inconsistent to some levels. Data pre-processing includes data cleaning, data transformation and data reduction based on the dataset. Some specified Machine Learning and Deep Learning models need information in a specified format, for example, Random Forest algorithm does not support null values, therefore, to execute random forest algorithm null values has to be managed from the original raw data set. The data was cleaned by finding out null values and imputing them. Four variables namely Company, Agent, Country and children had null values. As observed, 94.3% of 'Company' variable

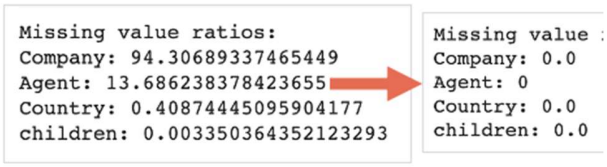

Fig.7. Missing Values and Imputation

are missing values. Therefore, there are less available values for the imputation process. The best option was to drop company column. For children variable, there were only 4 null entries. Having null entries can be comprehended same as having no child or 0 children altogether. Therefore, it was best to substitute 0 in place of null entries. For direct bookings there are no agents required, null values can be values corresponding to Direct bookings in the agent variable. About 13.68% of the data in agent variable constitutes missing values, values can't be deleted or not taken into consideration as it can be important for prediction. Therefore, substitution of an ID ['000'] was implemented in order to use the data for the prediction. The null values which corresponded to direct bookings were replaced by the ID [000] whereas the remaining 10333 null values are NULL which did not correspond to Direct bookings

were imputed by taking mode since very less percentage of Null values were substituted. The 488 entries were dropped entries constituted only 0.4% of the variable data. An outlier is a data point that is distant from other similar points. They may be due to variability in the measurement or may indicate experimental errors.

Wherever applicable outliers can be excluded to eliminate noisy data from the dataset. If not removed they can alter the accuracy of models resulting in poorer outputs. Outliers can mislead the training process. With the help of the univariate method, the outliers were visualized. With the use of boxplots distribution of data was displayed. Outliers are those values of a variable that fall far from the central point, the median. The Outliers were treated using IQR (Inter-Quartile Range). Outlier Detection was done by identifying the lower-bound and upper-bound of the data. Any values which were less than lower-bound or greater than upper bound were deleted and plotted a scatterplot to see the result.

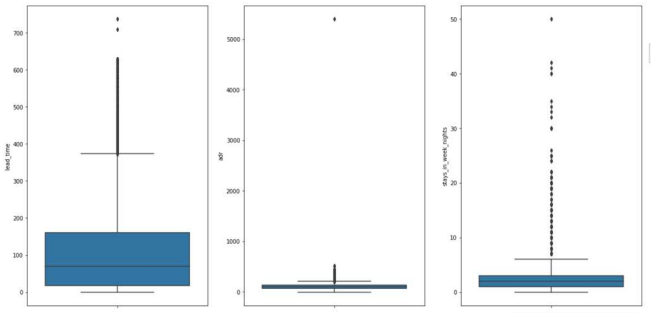

Fig.8. Boxplots of Lead-time, Average Daily Rate and Stays in weeknights for Outlier Detection

Data transformation was done with respect to categorical data in our dataset. Since variables like meal, hotel, market segment, etc. had no ordinal variables so no ranking method was used. Instead, one-hot encoding was performed on all categorical variables. In one-hot encoding, new columns are created for each unique value. Data transformation created new columns and increased the dimension of the data. It is desirable to reduce the number of input variables to both reduce the computational cost of modeling and, in some cases, to improve the performance of the model. Since the reservation status variable was directly proportional to 'is_canceled', the 'reservation status' variable was deleted and not taken into consideration since it can cause accuracy to reach 100%.

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.2078	1.36e-12	1.53e+11	0.000	0.208	0.208
lead_time	-9.645e-17	1.4e-17	-6.909	0.000	-1.24e-16	-6.91e-17
arrival_date_year	-3.207e-16	2.15e-15	-0.149	0.882	-4.54e-15	3.9e-15
arrival_date_week_number	-1.067e-15	2.52e-15	-0.423	0.672	-6e-15	3.87e-15
arrival_date_day_of_month	1.366e-16	3.78e-16	0.361	0.718	-6.05e-16	8.78e-16
stays_in_weekend_nights	1.976e-16	1.2e-15	0.164	0.870	-2.16e-15	2.56e-15
stays_in_week_nights	-2.68e-17	7.66e-16	-0.035	0.972	-1.53e-15	1.47e-15
is_repeated_guest	5.881e-16	6.28e-15	0.094	0.925	-1.17e-14	1.29e-14
previous_cancellations	-9.527e-17	1.17e-15	-0.081	0.935	-2.39e-15	2.2e-15
previous_bookings_not_canceled	-3.112e-17	7.19e-16	-0.043	0.965	-1.44e-15	1.38e-15
booking_changes	3.725e-16	1.61e-15	0.232	0.817	-2.78e-15	3.52e-15
agent	-1.927e-16	1.37e-17	-14.061	0.000	-2.2e-16	-1.66e-16
days_in_waiting_list	-7.686e-15	4.88e-26	-1.58e+11	0.000	-7.69e-15	-7.69e-15
adr	1.71e-17	3.67e-17	0.465	0.642	-5.49e-17	8.91e-17
required_car_parking_spaces	1.135e-15	4.23e-15	0.268	0.789	-7.16e-15	9.43e-15
total_of_special_requests	-2.851e-16	1.42e-15	-0.201	0.841	-3.07e-15	2.5e-15
total_guest	-2.721e-16	1.65e-15	-0.165	0.869	-3.51e-15	2.96e-15

Fig.9. Feature Importance by Logistic Regression

To see the impact of other variables more clearly 'reservation_status' variable did not make it to the final dataset for modeling. After running logistic regression, the model was significant as the p-value was less than 0.05. The variables having p-value greater than 0.05 were removed. Variables like 'previous_cancellations', 'previous_bookings_not_canceled','booking_changes',' required_car_parking_spaces','total_of_special_requests' and many more were dropped.

VI. DATA MODELING AND EVALUATION

Machine learning is the process of mathematical algorithms learning patterns or trends on previously recorded data observations and then makes a prediction or classification. In this work, we are examining only binary classification (e.g. Y = 1,0), which is a form of supervised learning in which an algorithm aims to classify which category an input belongs to. Supervised learning can be described as taking an input vector comprised of n-features and mapping it to an associated target value or class label. The term "supervised" originated from the concept that the training and testing datasets contain a response label and the algorithm observes the input vector and attempts to learn a probability distribution to predict 'y' given 'x'.

Models were developed using different classification algorithms and then selecting the one(s) that present better performance indicators. Because the label "is_canceled" could only assume binary values (0: no; 1: yes), the following two-class classification algorithms were chosen:

- Logistic Regression
- Random Forest Classifier

• Gradient Boosting
• Naïve Bayes
• K-nearest Neighbors

1. Logistic Regression

Logistic regression is named after the function, the logistic function which is an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1. Logistic regression models the probability of the default class (e.g. the first class). The probability prediction is transformed into a binary value (0 or 1) using the logistic function in order to actually make a probability prediction. Since Logistic regression assumes no error in the output variable (y), removing outliers is the first step in preparing the training data for this model. Like linear regression, the model can overfit if the data has multiple highly correlated inputs. Therefore, the reservation status variable was removed from the data. Since 'is_canceled' and 'reservation status' were highly correlated. The base model was trained on the training data with default parameter settings, and the accuracy came to be 75% and the F-1 Score of 67%.

2. Random Forest Classifier

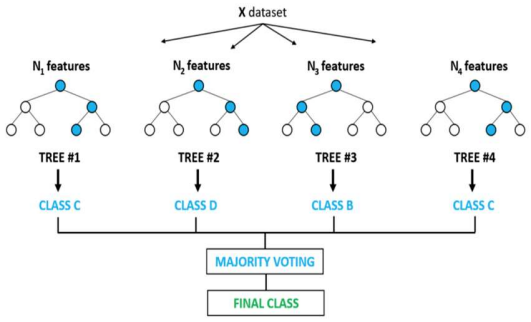

Fig.10. Example of ensemble of Random forest

Random Forest is an ensemble-based learning algorithm that uses multiple trees to average (regression) or compute majority votes (classification) in the terminal leaf nodes when making a prediction. It is built off the idea of bootstrap aggregation, which is a method for resampling with replacement in order to reduce variance. For instance, while using the random forest for classification, each tree will give an estimate of the probability of the class label, the probabilities will be averaged over the 'n' trees and the highest yields the predicted class label (fig.9). The base model was trained on the training data, and the accuracy came to be 80.97% and F1-score is 79%.

3. Gradient Boosting

Gradient Boosting Machine is an ensemble method consisting of multiple decision trees where each decision tree is built sequentially one after the other and the final ensemble model is produced by taking the weighted average of predictions made by each base classifier. GBTs iteratively train decision trees in order to minimize a loss function. Like decision trees, GBTs handle categorical features, extend to the multiclass classification setting, do not require feature scaling, and are able to capture non-linearities and feature interactions. On each iteration, the algorithm uses the current ensemble to predict the label of each training instance and then compares the prediction with the true label. The dataset is re-labeled to put more emphasis on training instances with poor predictions. Thus, in the next iteration, the decision tree will help correct for previous mistakes. The specific mechanism for re-labeling instances is defined by a loss function. With each iteration, GBTs further reduce this loss function on the training data. GBT uses Log Loss for classification tasks. We begin running the base model with default values and got an F-1 score of 71% and an accuracy of 76.65%.

4. Naïve Bayes

Naive Bayes is a statistical classification technique based on Bayes Theorem. It is one of the simplest supervised learning algorithms. Naive Bayes classifier assumes that the effect of a particular feature in a class is independent of other features. Even if these features are interdependent, these features are still considered independently. This assumption simplifies computation, and that's why it is considered as naive. This assumption is called class conditional independence. Naive Bayes classifier calculates the probability of an event by calculating the prior probability for given class labels. Finding the Likelihood probability with each attribute for each class followed by putting these values in Bayes Formula and calculating posterior probability. Finally

looking for a class that has a higher probability, given the input belongs to the higher probability class. The training data was used to train the model and base model with default parameter settings and using Gaussian Naïve Bayes performed with an accuracy of 69.04% and an F-1 Score of 66%.

5. K-Nearest Neighbors

When KNN is used for classification, the output can be calculated as the class with the highest frequency from the K-most similar instances. Each instance in essence votes for their class and the class with the most votes is taken as the prediction. Class probabilities can be calculated as the normalized frequency of samples that belong to each class in the set of K most similar instances for a new data instance. Since there are even number of classes (e.g. 2) it is a good idea to choose a K value with an odd number to avoid a tie. K value of 3 is taken to run the base model and the Accuracy and F-1 Score were 74.14% and 69% respectively.

VII. Model Comparison

More robust approaches like Ensemble method algorithms worked well with class data. Classification metrics like F1 score, precision, recall and accuracy, and true and false positive rates are examined on the portion of data that was held out during training and for evaluating how well the model classifies the input feature vector using fivefold cross-validation.

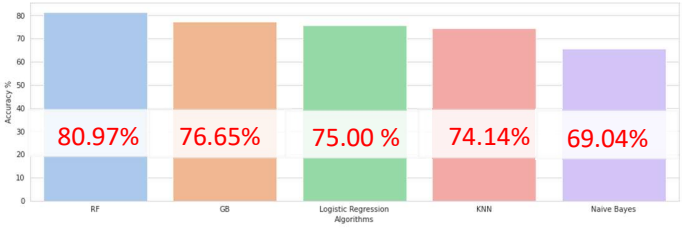

Fig.11. Model comparison (Accuracy)

The top three models Random Forest, Gradient Boosting and Logistic Regression in terms of accuracy and F1 score were selected and further tuned.

Hyperparameter	Logistic Regression	Description
c		The trade-off parameter of logistic regression that determines the strength of the regularization is called C. Inverse of regularization strength; must be a positive float. Like in support vector machines, smaller values specify stronger regularization.
Penalty		Used to specify the norm used in the penalization. The 'newton-cg', 'sag' and 'lbfgs' solvers support only l2 penalties. 'elastic net' is only supported by the 'saga' solver. If 'none' (not supported by the liblinear solver), no regularization is applied.
Solver		There are various kind of solvers which are used to find the global minima instead of local minima in a curve which helps us to decide the optimum value of weights.
Hyperparameter	**Random Forest**	**Description**
n_estimators		This is the number of trees you want to build before taking the maximum voting or averages of predictions.
max_features		The number of features to consider while searching for a best split. These will be randomly selected.
max_depth		The max depth of a tree. Used to control over-fitting as higher depth will allow model to learn relations very specific to a sample.
min_samples_split		Defines the minimum number of samples (or observations) which are required in a node to be considered for splitting. Higher values prevent a model from learning relations which might be highly specific to the sample selected for a tree. Too high values can lead to underfitting.
min_samples_leaf		Defines the minimum samples (or observations) required in a terminal node or leaf. Generally lower values should be chosen for imbalanced class problems because the regions in which the minority class will be in majority will be very small.
bootstrap		Method for sampling data points (with or without replacement)
Hyperparameter	**Gradient Boosting**	**Description**
Learning_rate		This determines the impact of each tree on the outcome. GBM works by starting with an initial estimate which is updated using the output of each tree. The learning parameter controls the magnitude of this change in the estimates. Lower values are generally preferred as they make the model robust to the specific characteristics of tree and thus allowing it to generalize well. Lower values would require higher number of trees to model all the relations and will be computationally expensive.
max_depth		Bounds the maximum depth of the tree.
n_estimators		The number of sequential trees to be modeled. Though GBM is robust at higher number of trees but it can still overfit at a point. Hence, this should be tuned using CV for a learning rate.

Fig.12. Hyperparameters used for tuning models

VIII. Hyperparameter Tuning and Results

After tuning hyperparameters with different combinations, there was an increase in the Accuracy and F1 scores for all the three models. Random Forest has the highest accuracy and F1 score with a maximum depth of 100 trees, minimum samples leaf of 1, and n_estimators of 300. Random forest was followed by Gradient Boosting and Logistic Regression. Further, as observed from fig.14 the confusion matrix of Random Forest shows that the Random Forest has the minimum number of False-Negatives.

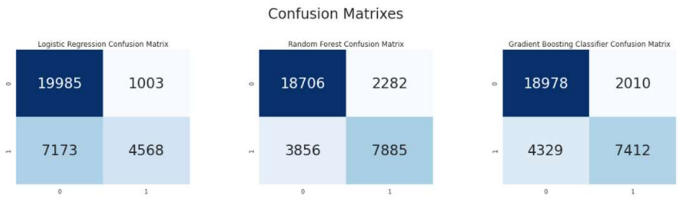

Fig.13. Confusion Matrices

Hyperparameter	boostrap	max_depth	max_features	min_samples_leaf	min_samples_split	n_estimators	Accuracy	Precision	Recall	F1-score
1	True	50	1	5	8	200	74%	86%	64%	64%
2	True	10	0.5	1	2	100	77%	80%	71%	72%
3	**True**	**100**	**0.5**	**1**	**4**	**300**	**81%**	**80%**	**78%**	**79%**

Random Forest

Hyperparameter	Learning_rate	max_depth	n_estimators	Accuracy	Precision	Recall	F1-score
1	0.1	6	100	79%	79%	73%	75%
2	0.5	3	200	79%	78%	74%	75%
3	**0.1**	**9**	**300**	**81%**	**80%**	**77%**	**78%**

Gradient Boosting

Hyperparameter	c	Penalty	Accuracy	Precision	Recall	F1-score
1	0.1	l2	75%	78%	67%	68%
2	5.0	l1	75%	78%	67%	68%
3	**1.0**	**l2**	**75%**	**78%**	**67%**	**68%**

Logistic Regression

Fig.14. Hyperparameter tuning and results

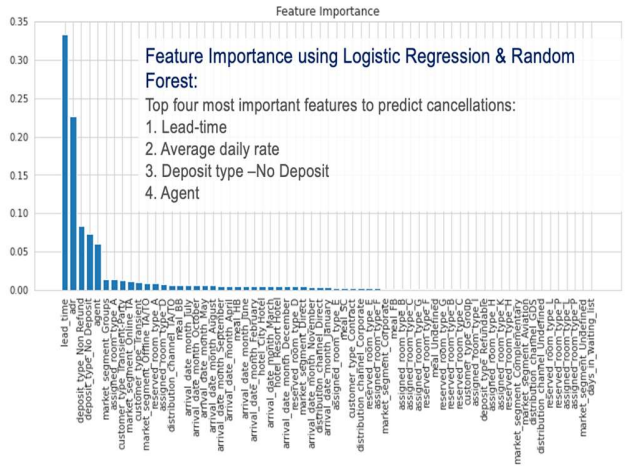

Fig.15. Feature Importance by Random Forest

Feature importance is about evaluating the relationship between each input variable and the target variable using statistics and selecting those input variables that have the strongest relationship with the target variable. Embedded methods like Random Forest combine the qualities of filter and wrapper methods with a built-in feature selection method. Observations and features are randomly extracted from the dataset. Therefore, whole data is not visited by the tree/s and so trees are not correlated and hence less prone to overfitting. After the division of the dataset, the importance of each feature is derived from how "pure" each of the buckets is. Random Forest and Logistic Regression was used for feature importance. The top four features were Lead-time, Average daily rate, Deposit type - No Deposit, and Agent (ID of the Agent who booked).

IX. CONCLUSION

Algorithm which best predicted the cancellations is Random forest with 81% Accuracy. With low False-Negatives and high Recall, it performed better than Gradient Boosting. The Top features to predict cancellations were Lead time, ADR, Deposit type (No deposit), Agent, etc. The Hotel should also target more customers from all over the world other than just Europe. More focus should be given to repeated guests as they can significantly improve revenue in the future. It can also predict Dynamic Pricing in the future.

REFERENCES

[1] Thumbs up? Sentiment Classification using Machine Learning Techniques Bo Pang and Lillian Lee, Shivakumar Vaithyanathan

[2] Kimes, S. E. (2010). The future of hotel revenue management. Cornell Hospitality Reports, 10(14).

[3] Hayes, D. K., & Miller, A. A. (2011). Revenue management for the hospitality industry. Hoboken, NJ, USA: John Wiley & Sons, Inc

Terms and definitions

Metric	Formula
Accuracy	(TP + TN) / (TP + TN + FP + FN)
True Positive Rate (TPR)	TP / (TP + FN)
False Positive Rate (FPR)	FP / (FP + TN)
Precision	TP / (TP + FP)
Recall	TP / (TP + FN)
Area Under the Curve (AUC)	Integral area of plotting TPR vs FPR

Accuracy: Measure of outcome correctness. Measures the proportion of true results (True Positives and True Negatives) among the total number of predictions.

Precision: Measures the proportion of True Positives against the sum of all positive predictions (True Positives and False Positives).

F1 Score: Measure of prediction accuracy, which is the harmonic means of precision and recall.

Recall: Measure of relevant predictions that are retrieved. It can be interpreted as the probability of a randomly selected prediction could be a True Positive.

Outcome: A variable which one's wanting to predict. Also known as response variable, dependent variable, or label.

Lead Time: Time (usually measured in days) between a booking's date of placement in the hotel and the guest's expected arrival date.

AUTHORS

First Author – Juilee Salunkhe, M.S. in Applied Data Science, Syracuse University (jsalunkh@syr.edu).
Second Author – Akshay Bhala, M.S. in Applied Data Science, Syracuse University (abhala@syr.edu).
Third Author – Raj Desai, M.S. in Applied Data Science, Syracuse University (rtdesai@syr.edu).