Case Study

Hotel Booking Cancellation Prediction

08/17/2023

Togzhan Nurmukanova

Contents / Agenda

Executive Summary

- Nowadays, the hotel business performance is facing the major problem of booking cancelations, which causes significant losses in revenue. To resolve this problem, extensive research and analysis are needed, therefore the business needs the advice services of the data scientist, helping to develop efficient solutions to predict accurately the booking status of the reservations. In the context of this case study, we look at the situation of the booking cancelations for INN Group Hotel Group.

- The data has been provided for the period of **July 2017 to December 2018** and considers the reservations done through the various channels including both **traditional and online** methods.

Executive Summary

- The brief look at the data provided the following insights as 92% of the reservations have been booked through online and offline channels. Also, the hotel demonstrated seasonal behavior as most bookings were allocated for August, September, and October months.

- The **Decision System** approach is chosen as the most suitable one, as it allows to study many aspects of booking cancellations, as accurate evaluation of the booking cancellations, that causes issues with **the profit losses** as well as evaluation of the not canceled bookings, that contributes to opportunity **growth for the business**. The solution approach was chosen to apply models such as **Decision Tree and Random Forest** with **default parameters** and **hyperparameter tuning that includes pruning/prepruning methods.**

Executive Summary (Cont)

- The obtained results show that the **Decision Tree (Default)** shows the overfitting problem among the Accuracy, Precision, and Recall. Application of the hyperparameter tuning and prepruning/pruning methods enhanced performance by helping to solve an overfitting problem. The precision for the TRUE class of the Training dataset was found as **77%**, and other parameters such as **Recall and Precision have balanced improvement**.

- Comparing the performance of the Random forest (Default) model, we found that it performed well with minimum parameter tuning. After applying the hyperparameter tuning and prepruning/pruning, we observed better performance for Precision as the True Class was measured as **80%**. Also slight decrease between the Training and Test dataset values was observed among all metrics.

Executive Summary

- A detailed look at the prediction features among all the models demonstrated that the **Lead time and Average price per room** were the most important features for predicting booking reservation status.

- Based on the findings of the case study, we can make recommendations for actionable insights findings as well as findings from the ML analysis. Initially, to protect the largest market segment type, the business should pay attention to the online and offline types of bookings and offer them incentives that help to keep their reservations. Also, we want to avoid cancellations for the seasonal time, therefore the business needs to take measures that would not be favorable to make late cancellations for this period.

Executive Summary

- Considering the ML model analysis, the business should take care of the reservations that were done far in advance and offer them a graded policy that would protect bookings from the cancelations. Furthermore, we can offer flexible price policies based on seasonality and offer attractive pricing for customers from the Online and Offline markets.

- To summarize, our case study helped to predict booking the cancelation status by evaluating the factors, that contribute to cancellations. This result was achieved by carefully studying the results and choosing the model with the best performance which was determined as Random Forest with pruning/repruning and hyperparameter tuning applied. It allowed us to obtain a good Training dataset for the True class of precision and balanced improvement on other metrics. The business uses these results to develop measures to prevent booking cancellations and expand opportunities for business growth by attracting new customers.

Business Problem Overview and Solution Approach

- The hotel business is known to be long-time established, competitive business type. The profitability of this business is highly dependent on the actual occupancy happening over time. The highest negative impact for the hotel is the large number of cancelations caused by the cancellations or no show-offs.
- One of the potential ways to establish the cancellation is to determine which factors affect the reservations, that would help to predict which type of reservation is more likely to be canceled. The implication of the predictive model would provide an understanding of the factors influencing the reservation and would help to build the hotel policies enhancing profitability by tackling cancellations and refunds.

Business Problem Overview and Solution Approach (cont)

- By achieving the goal of the accurate prediction of hotel cancellation we would contribute to:

1. effective resource management, by allocating the staff and resources with respect to actual bookings.
2. Customer satisfaction, by approaching the customers that are most likely to cancel and offer them incentives encouraging them to keep their reservations.
3. Financial performance, by converting the right amount of the reservations predicted to be canceled to the new reservations, allows avoiding overbookings and underoccupancies.

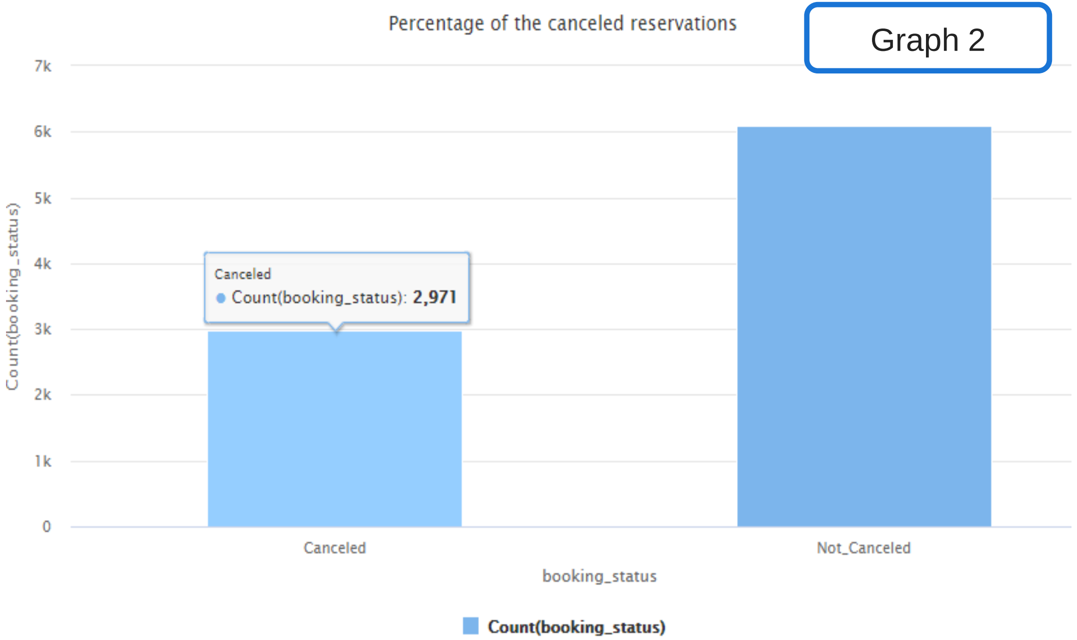
Business Problem Overview and Solution Approach (cont)

- The decision systems would be used to evaluate:

1. the main characteristics of the reservations that have the highest tendency to cancel;
2. Reservations wrongly predicted to be canceled, that help to minimize opportunities for revenue growth.

Univariate Analysis

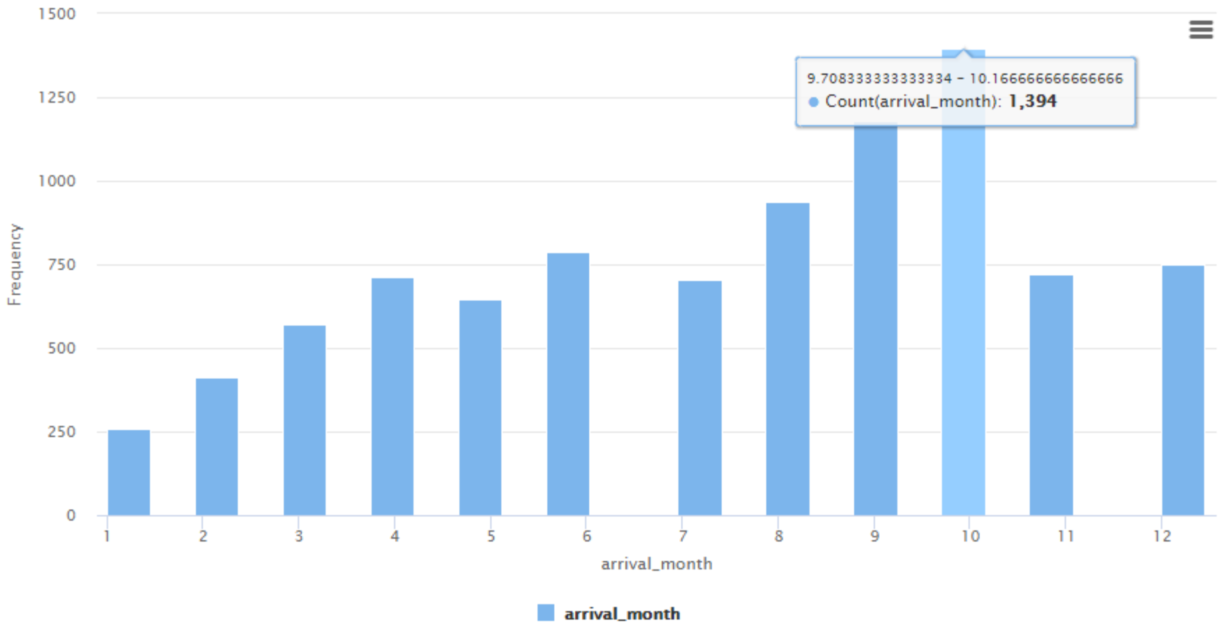
EDA: Percentage of the canceled resevations-Histogram



- The booking status is the target variable, it shows whether the reservation was canceled or not.

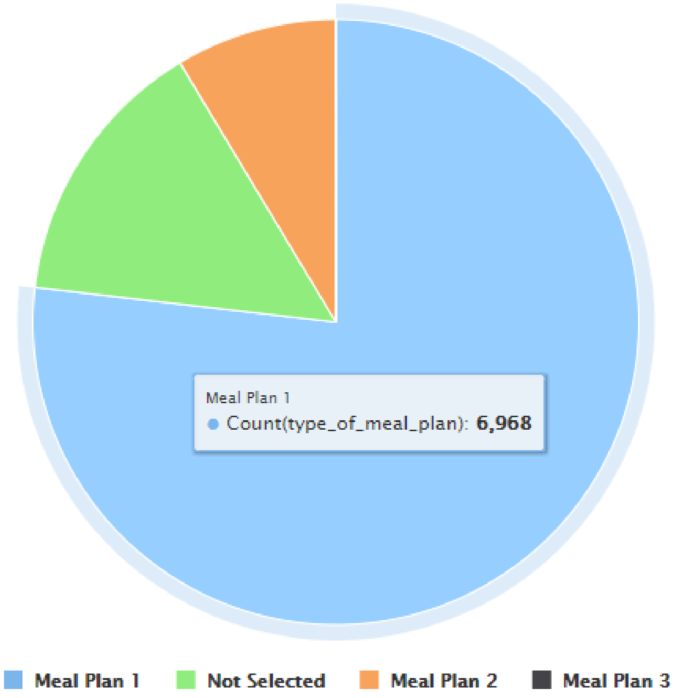
- Approximately **1/3** of the reservations have canceled status

EDA: Month of the arrival date- Histogram



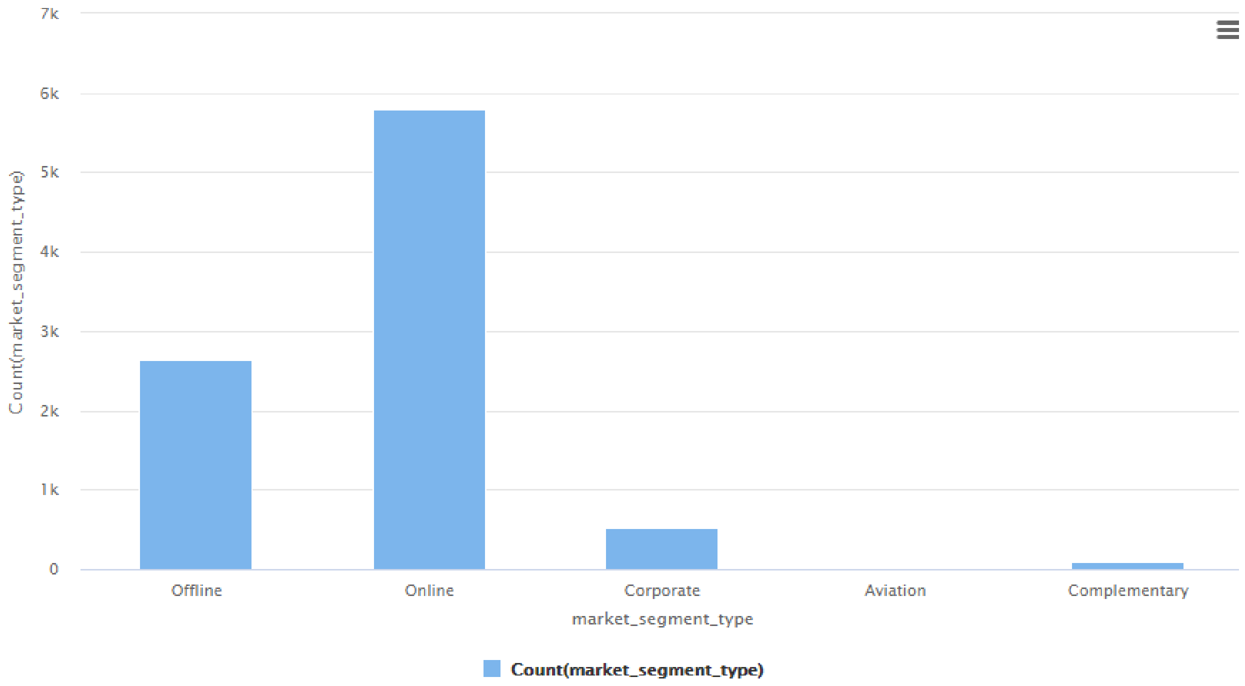
- The most popular season is the time between August to October, which corresponds to **40%** of the reservations done throughout the year.

EDA: Type of a meal plan-Chart



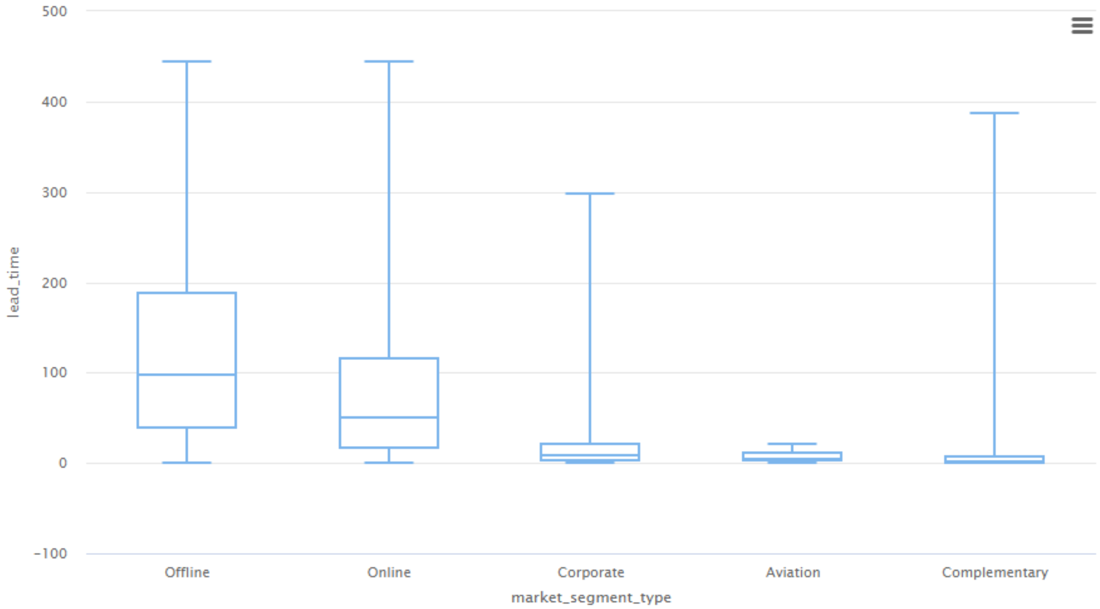
- The majority of the customers prefers breakfast-only and no meal plan selected options.
- **75%** of the booking reservations were opted as the breakfast-only option.

EDA: Market Segment type-Histogram



- **92%** of the booking reservations are taking place through the Online and Offline Channels

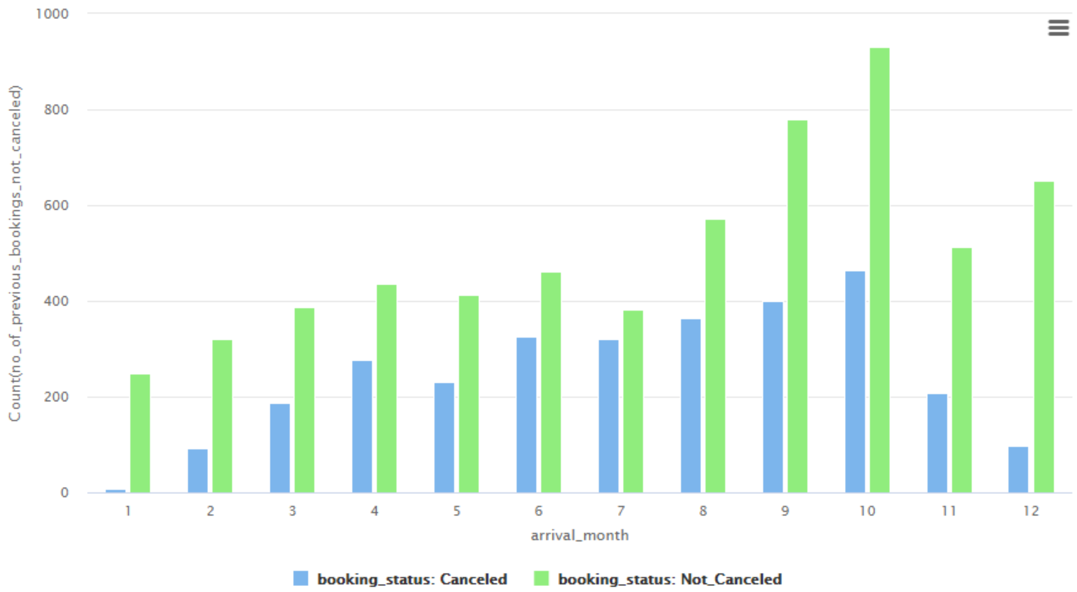
EDA: Lead time across the different market sectors- Box Plot



- The reservations done through the **Online** and **Offline** market channels have been booked the most far in advance

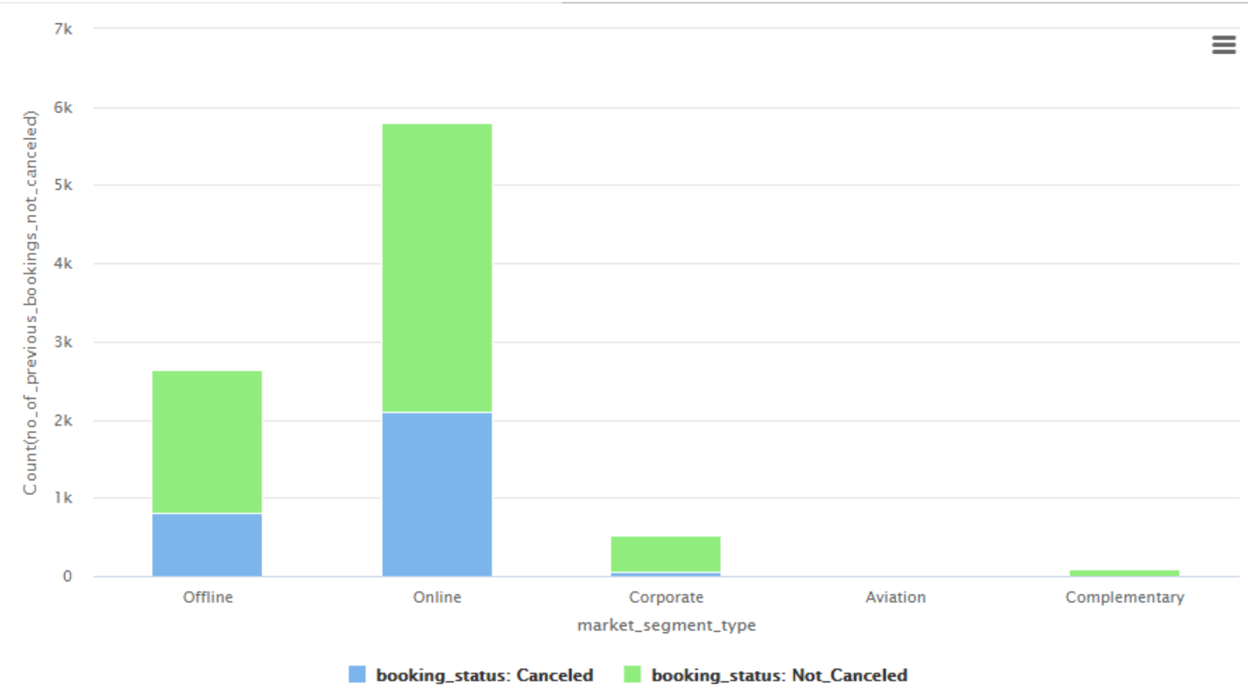
Bivariate Analysis

EDA: Booking Status vs Arrival month- Grouped Bar Plot



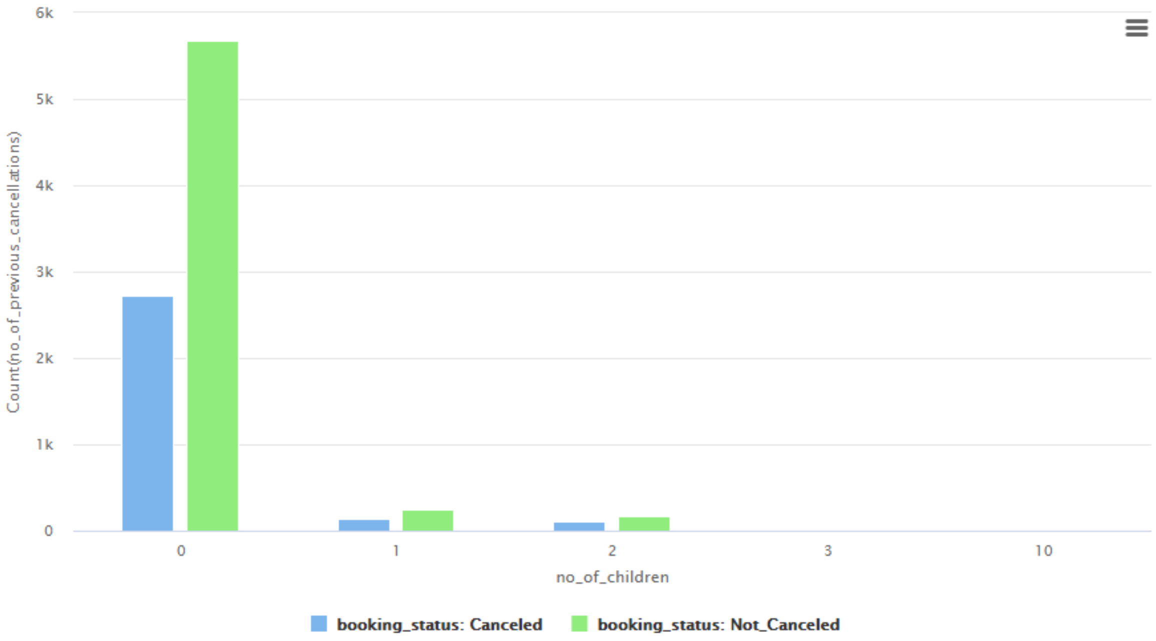
- Approximately **~40%** of the canceled reservations contributed to June, July and August months

EDA: Booking status across the market segment type-Histogram



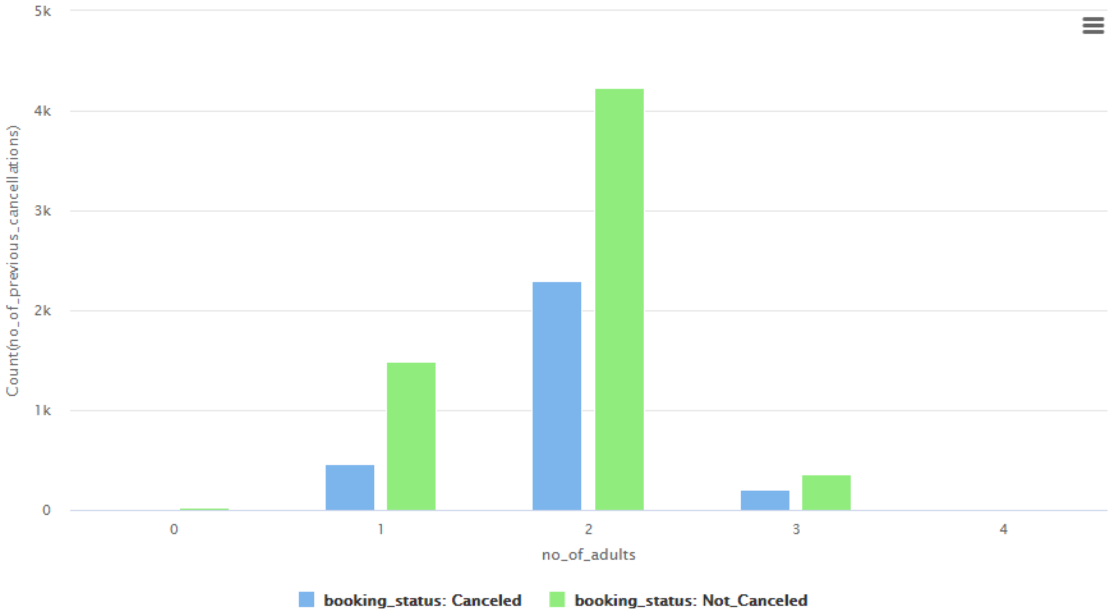
- Reservations with the canceled booking status contribute to **1/3** of the total number of reservations for the Online/ Offline market.

EDA: Previous cancellations vs No of children-Grouped Bar Plot



- Most of the bookings were customers travelling without children.
- The number of children per reservation did not affect the cancelation ratio.

EDA: No of previous cancelations vs No of adults-Histogram



- According to the graph below, the usually the most common number of guests was two. The cancellations for this type of reservation happened more than in half of the cases.

Optimized Parameters of Final Model- Random Forest (Pruned)

The ML Model with the best performance were established as Random Forest (Pruned), with the following parameters:

Splitting criterion: **Information Gain**
Number of trees :**10**
Maximal Depth: **10**

Voting strategy: Majority
Use Local random seed
Local Random Seed**: 1**

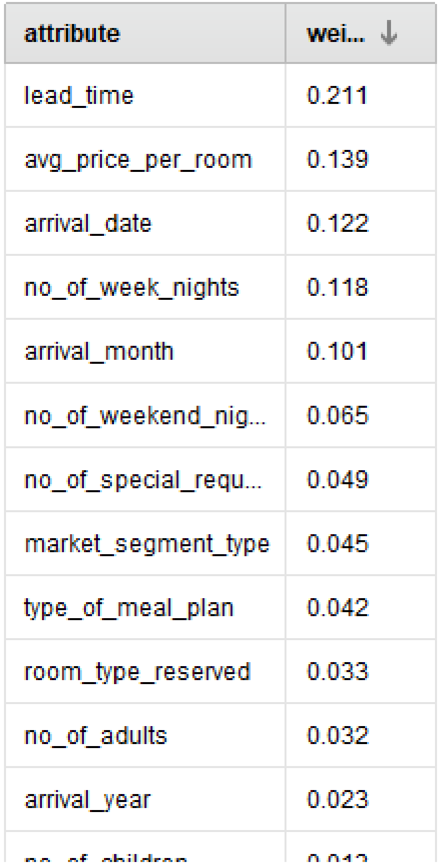
Apply Pruning
Minimal Leaf Size: **2**
Minimal Size of Split: **4**
No of prepruning alternatives: **3**

Apply Prepruning
Confidence: **0.1**

Final ML model –Random Forest (Pruned)

The most important features used for predictions are **Lead time, Average price per room, Arrival date, no of week nights, arrival month, and no of weekend nights**, which contributes to **75%** of the impurity level reduction.

The rest of the features with the overall weight of **25%**, have the smallest impact on the booking status of the reservation.



attribute	wei... ↓
lead_time	0.211
avg_price_per_room	0.139
arrival_date	0.122
no_of_week_nights	0.118
arrival_month	0.101
no_of_weekend_nig...	0.065
no_of_special_requ...	0.049
market_segment_type	0.045
type_of_meal_plan	0.042
room_type_reserved	0.033
no_of_adults	0.032
arrival_year	0.023
no_of_children	0.013

Model Results

Model	Train Accuracy	Test Accuracy	Train Recall	Test Recall	Train Precision	Test Precision
Decision Tree	99.65	83.20	99.61	81.58	99.61	80.82
Decision Tree-PRUNED	87.16	84.38	84.76	81.79	85.77	82.43
Random Forest	89.62	86.07	86.73	82.59	89.33	85.10
Random Forest Pruned	88.94	85.96	86.54	83.31	88.02	84.04

- The Random Forest is the ML with the best performance as it achieved a good value for **Test Precision of 84.04%** and at the same time possesses balanced improvement in Recall metrics.

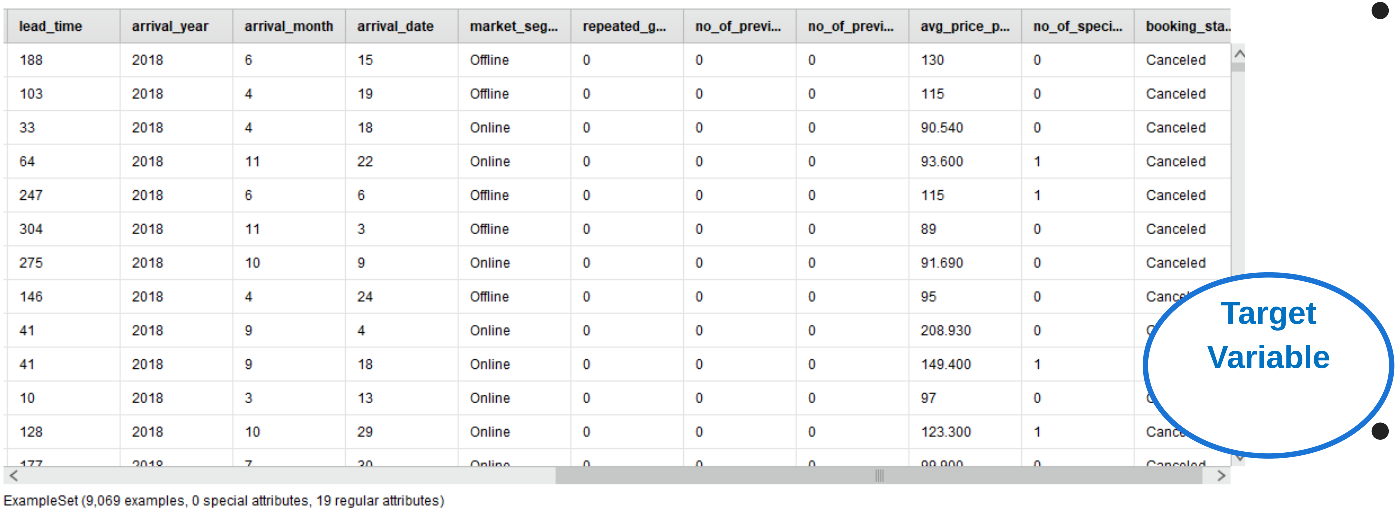
- The difference between the Training and Test Recall is less than 10%, which shows **no overfitting problem.**

APPENDIX

Data Background

- Source of Data: INN Hotels Group Database
- Collection Method: The data was collected through various booking channels, both involved new technologies such as Online booking as well as traditional booking.
- Period of Collection: July 2017- December 2018
- Context: the room type reserved has been encoded by INN Hotel Group for the privacy reasons
- Data Collection tools: online reservations were collected via the website, and the rest through traditional booking channels

Data Contents: Overview



lead_time	arrival_year	arrival_month	arrival_date	market_seg...	repeated_g...	no_of_previ...	no_of_previ...	avg_price_p...	no_of_speci...	booking_sta...
188	2018	6	15	Offline	0	0	0	130	0	Canceled
103	2018	4	19	Offline	0	0	0	115	0	Canceled
33	2018	4	18	Online	0	0	0	90.540	0	Canceled
64	2018	11	22	Online	0	0	0	93.600	1	Canceled
247	2018	6	6	Offline	0	0	0	115	1	Canceled
304	2018	11	3	Offline	0	0	0	89	0	Canceled
275	2018	10	9	Online	0	0	0	91.690	0	Canceled
146	2018	4	24	Offline	0	0	0	95	0	Cance
41	2018	9	4	Online	0	0	0	208.930	0	C
41	2018	9	18	Online	0	0	0	149.400	1	
10	2018	3	13	Online	0	0	0	97	0	C
128	2018	10	29	Online	0	0	0	123.300	1	Canc
177	2018	7	30	Online	0	0	0	99.900	0	Canceled

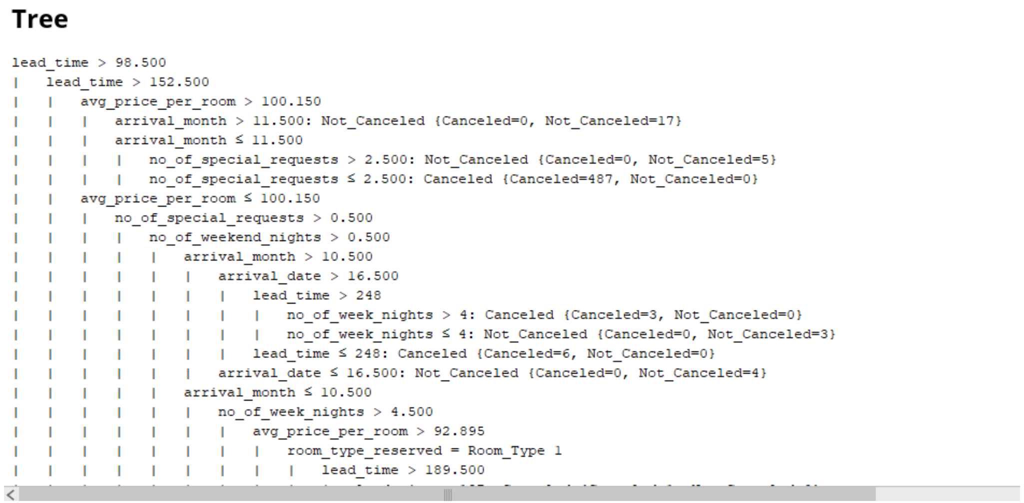
ExampleSet (9,069 examples, 0 special attributes, 19 regular attributes)

- **9,069 Rows**: Each row represent s booking reservation at the hotel.
- **19 Columns:** Each column represents the attribute/feature of the reservation.

Data Contents (cont)

- Data Structure: Types of data (polynomial, integer, Binomial, Real)
- Format of the data: CVS
- Description of Columns: **Polynomial**-booking id, type of meal plan, the room reserved, no of adults, market segment type; **Integer**-no of adults, no of children, no of weekend nights, no of week nights, lead time, arrival year, arrival month, arrival date, no of previous cancellations, no of bookings not canceled, no of special requests; **Binomial**- required car parking space, repeated guests, booking status, **Real**-average price per room.
- Metadata: **no missing values**, **booking Id** are unique, we can **drop this column**
- Quality and Limitations: there are some data entries for not canceled reservations with **0 average price cost.**

Model Building - Decision Tree (Unpruned)

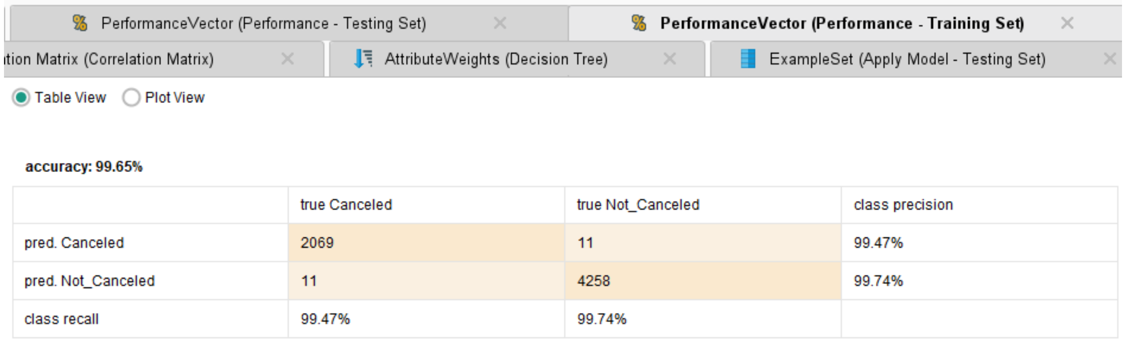


Tree

```
lead_time > 98.500
|   lead_time > 152.500
|   |   avg_price_per_room > 100.150
|   |   |   arrival_month > 11.500: Not_Canceled {Canceled=0, Not_Canceled=17}
|   |   |   arrival_month ≤ 11.500
|   |   |   |   no_of_special_requests > 2.500: Not_Canceled {Canceled=0, Not_Canceled=5}
|   |   |   |   no_of_special_requests ≤ 2.500: Canceled {Canceled=487, Not_Canceled=0}
|   |   avg_price_per_room ≤ 100.150
|   |   |   no_of_special_requests > 0.500
|   |   |   |   no_of_weekend_nights > 0.500
|   |   |   |   |   arrival_month > 10.500
|   |   |   |   |   |   arrival_date > 16.500
|   |   |   |   |   |   |   lead_time > 248
|   |   |   |   |   |   |   |   no_of_week_nights > 4: Canceled {Canceled=3, Not_Canceled=0}
|   |   |   |   |   |   |   |   no_of_week_nights ≤ 4: Not_Canceled {Canceled=0, Not_Canceled=3}
|   |   |   |   |   |   |   lead_time ≤ 248: Canceled {Canceled=6, Not_Canceled=0}
|   |   |   |   |   |   arrival_date ≤ 16.500: Not_Canceled {Canceled=0, Not_Canceled=4}
|   |   |   |   |   arrival_month ≤ 10.500
|   |   |   |   |   |   no_of_week_nights > 4.500
|   |   |   |   |   |   |   avg_price_per_room > 92.895
|   |   |   |   |   |   |   |   room_type_reserved = Room_Type 1
|   |   |   |   |   |   |   |   |   lead_time > 189.500
```

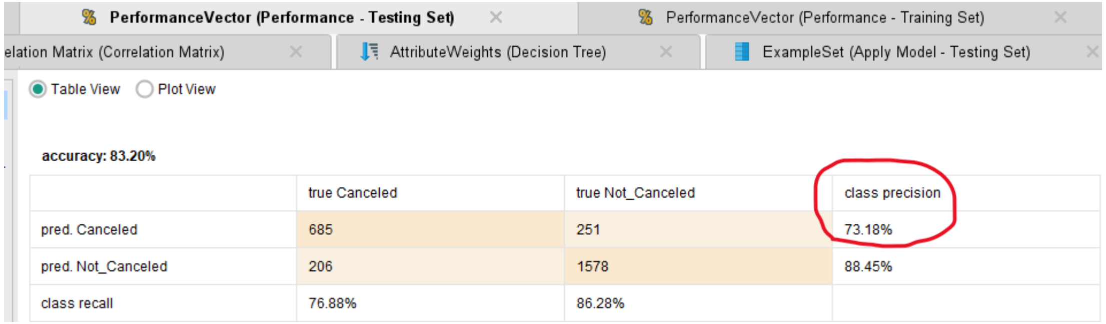
- the top three variables used to build the decision tree are Lead time, Average price per room and Arrival Month/ Number Special Requests.
- Gini Index was chosen as the splitting criterion and the maximal depth of the tree was set as 100.

Model Building - Decision Tree (Unpruned)



accuracy: 99.65%

	true Canceled	true Not_Canceled	class precision
pred. Canceled	2069	11	99.47%
pred. Not_Canceled	11	4258	99.74%
class recall	99.47%	99.74%	



accuracy: 83.20%

	true Canceled	true Not_Canceled	class precision
pred. Canceled	685	251	73.18%
pred. Not_Canceled	206	1578	88.45%
class recall	76.88%	86.28%	

- The results for the training dataset show a difference in accuracy of more than 10% which indicates an ***overfitting problem***, when the model performs well on the training dataset but does perform poor on the test dataset

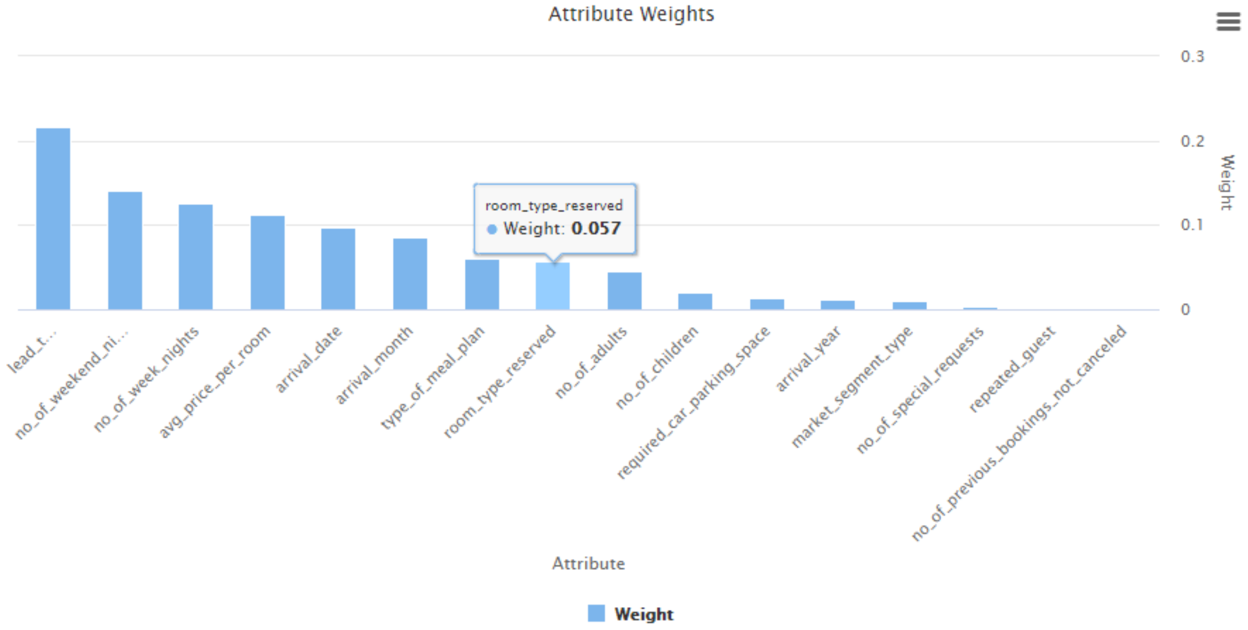
Model Building - Decision Tree(Unpruned)

Model	Train Accuracy	Test Accuracy	Train Recall	Test Recall	Train Precision	Test Precision
Decision Tree	99.65	83.20	99.61	81.58	99.61	80.82

- high variance between the train and the test dataset for both Precision and Recall parameters. To overcome this problem, the hyperparameter tuning is needed.

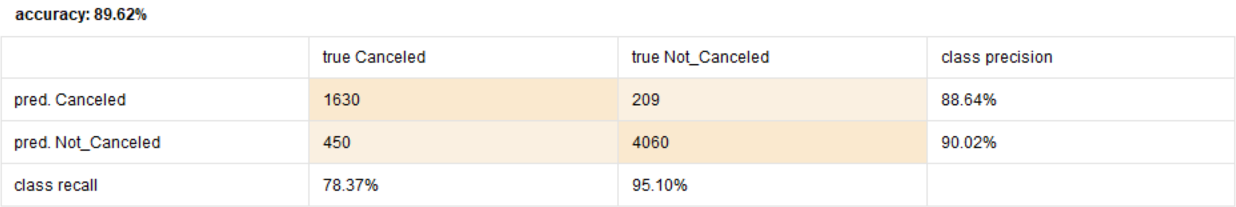
- The overall performance of the Decision Tree model (Default), indicates an ability to provide general performance, by correctly identifying the majority of the bookings that contribute to cancellations, with nearly **73% Precision for the TRUE class.**

Feature Importance Decision Tree (Unpruned)



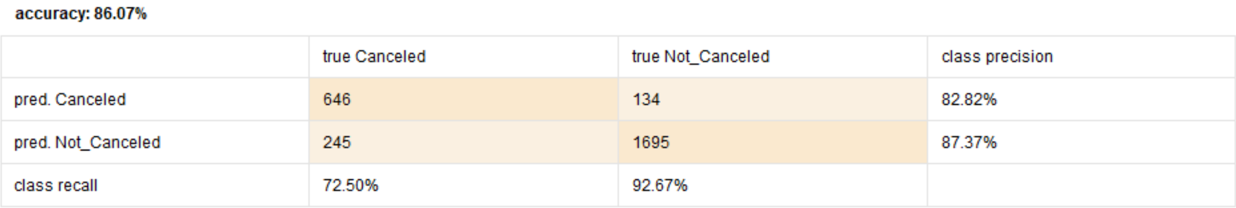
- **The lead time, no of weekend nights, no of week nights and average price per room** are considered the most valuable predictors of the booking status.

Model Building - Random Forest (Unpruned)



accuracy: 89.62%

	true Canceled	true Not_Canceled	class precision
pred. Canceled	1630	209	88.64%
pred. Not_Canceled	450	4060	90.02%
class recall	78.37%	95.10%	



accuracy: 86.07%

	true Canceled	true Not_Canceled	class precision
pred. Canceled	646	134	82.82%
pred. Not_Canceled	245	1695	87.37%
class recall	72.50%	92.67%	

- Criterion- Gini Index, the number of trees: 10 and maximal depth:10
- The overfitting problem was resolved with minimum hyperparameter tuning performed.

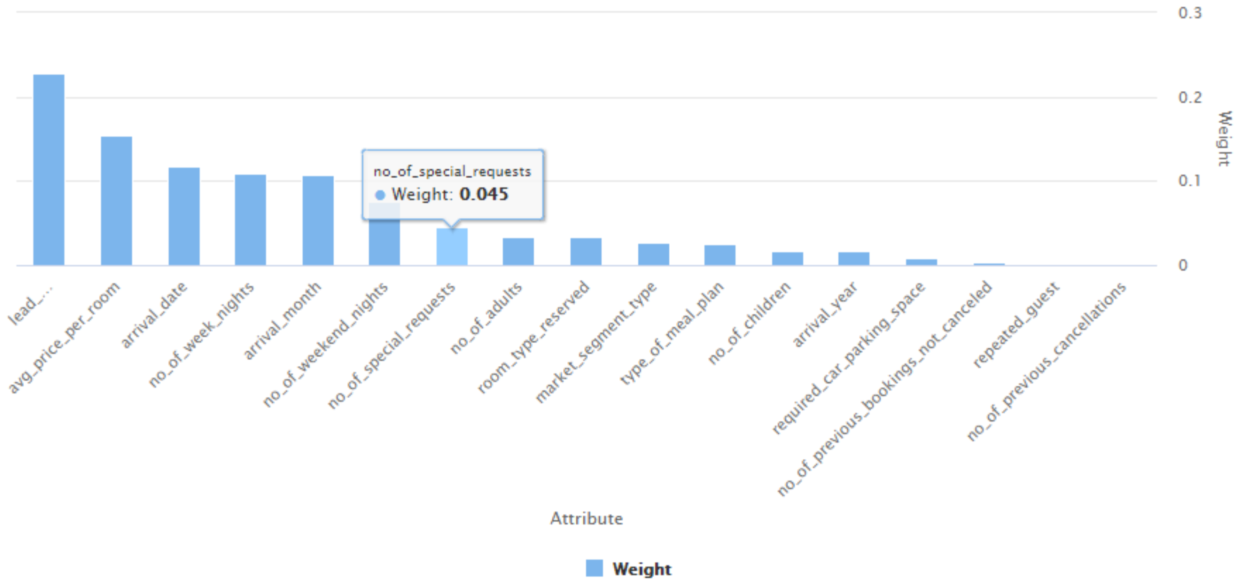
Model Building - Random Forest (Unpruned)

Model	Train Accuracy	Test Accuracy	Train Recall	Test Recall	Train Precision	Test Precision
Random Forest	89.62	86.07	86.73	82.59	89.33	85.10

- Precision and Recall values both for the Training and Test datasets demonstrated balanced improvement.

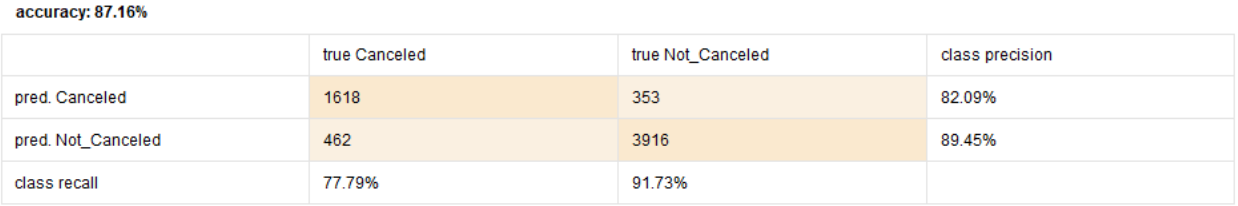
- The general goal was achieved by correctly identifying the majority of the bookings that contribute to cancellations, with nearly **83% Precision for the TRUE class.**

Feature Importance Random Forest (Unpruned)



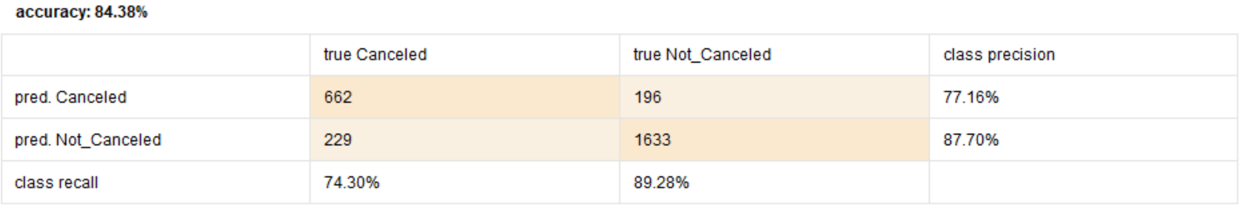
- The most important features were determined as **lead time, average price per room, arrival date, number of week nights, arrival nights.**

Model Performance Evaluation and Improvement - Decision Tree (Pruned)



accuracy: 87.16%

	true Canceled	true Not_Canceled	class precision
pred. Canceled	1618	353	82.09%
pred. Not_Canceled	462	3916	89.45%
class recall	77.79%	91.73%	



accuracy: 84.38%

	true Canceled	true Not_Canceled	class precision
pred. Canceled	662	196	77.16%
pred. Not_Canceled	229	1633	87.70%
class recall	74.30%	89.28%	

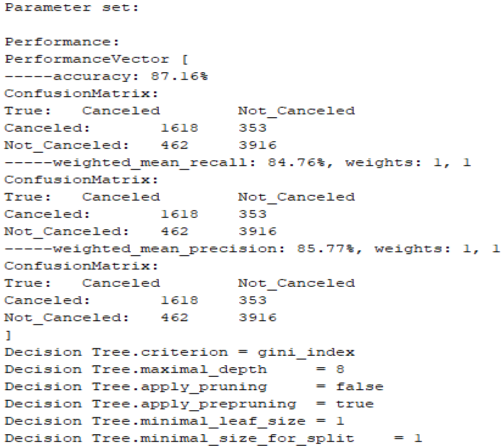
- Optimizing Model Parameters: **Criterion** (Gain Ratio, Gini Index, Information Gain), Max. Depth 5 to 8, **Apply pruning** true/false, **apply prepruning** true/false, **min. leaf size** 1 to 10, **min split size** 1 to 30

Model Performance Evaluation and Improvement - Decision Tree (Pruned)

Model	Train Accuracy	Test Accuracy	Train Recall	Test Recall	Train Precision	Test Precision
Decision Tree-PRUNED	87.16	84.38	84.76	81.79	85.77	82.43

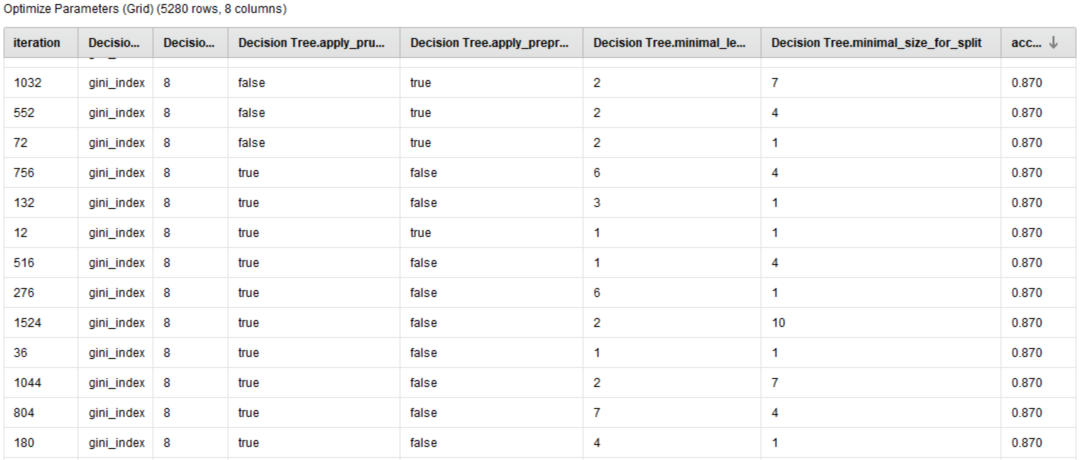
- To resolve the high variance for the train and test dataset of the Decision Tree (Default), we applied the following techniques as pruning, reduction of tree depth, and tuning of other hyperparameters. The improvement of both Precision and Recall for Training and Test showed the improvement.
- This helped to increase value of the Precision of True class, as our **77.16%** of true canceled reservations have been classified correctly.

Model Performance Evaluation and Improvement - Decision Tree (Pruned)



```
Parameter set:

Performance:
PerformanceVector [
-----accuracy: 87.16%
ConfusionMatrix:
True:   Canceled         Not_Canceled
Canceled:        1618    353
Not_Canceled:    462     3916
-----weighted_mean_recall: 84.76%, weights: 1, 1
ConfusionMatrix:
True:   Canceled         Not_Canceled
Canceled:        1618    353
Not_Canceled:    462     3916
-----weighted_mean_precision: 85.77%, weights: 1, 1
ConfusionMatrix:
True:   Canceled         Not_Canceled
Canceled:        1618    353
Not_Canceled:    462     3916
]
Decision Tree.criterion = gini_index
Decision Tree.maximal_depth      = 8
Decision Tree.apply_pruning      = false
Decision Tree.apply_prepruning   = true
Decision Tree.minimal_leaf_size = 1
Decision Tree.minimal_size_for_split    = 1
```



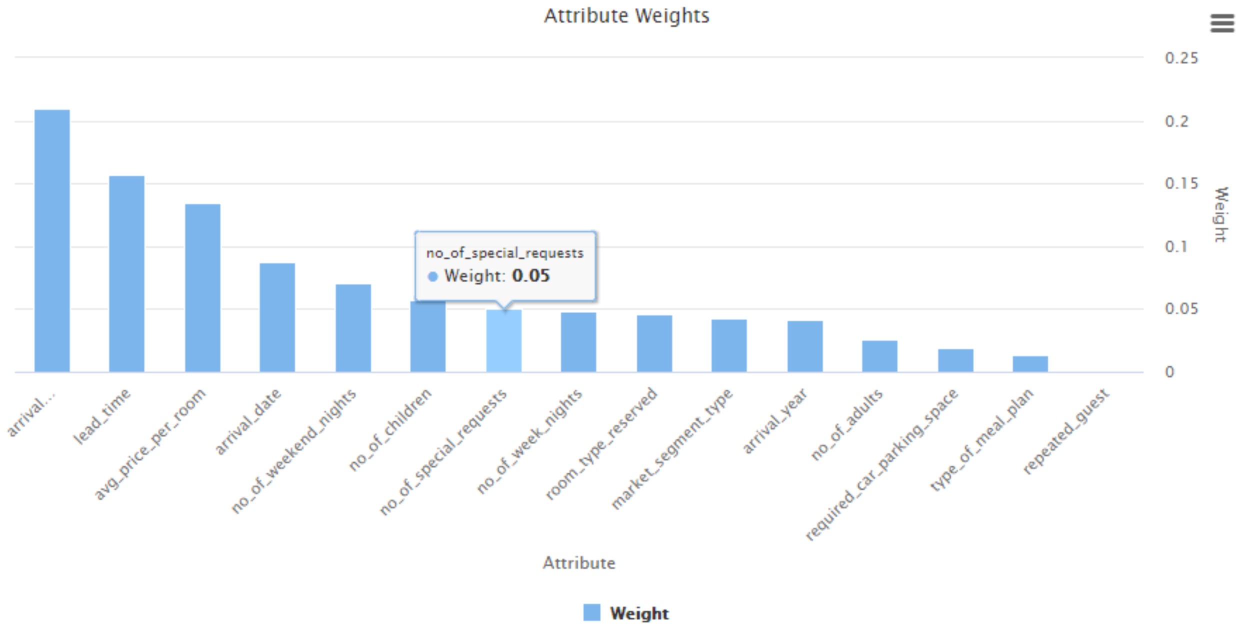
Optimize Parameters (Grid) (5280 rows, 8 columns)

iteration	Decisio...	Decisio...	Decision Tree.apply_pru...	Decision Tree.apply_prepr...	Decision Tree.minimal_le...	Decision Tree.minimal_size_for_split	acc... ↓
1032	gini_index	8	false	true	2	7	0.870
552	gini_index	8	false	true	2	4	0.870
72	gini_index	8	false	true	2	1	0.870
756	gini_index	8	true	false	6	4	0.870
132	gini_index	8	true	false	3	1	0.870
12	gini_index	8	true	true	1	1	0.870
516	gini_index	8	true	false	1	4	0.870
276	gini_index	8	true	false	6	1	0.870
1524	gini_index	8	true	false	2	10	0.870
36	gini_index	8	true	false	1	1	0.870
1044	gini_index	8	true	false	2	7	0.870
804	gini_index	8	true	false	7	4	0.870
180	gini_index	8	true	false	4	1	0.870

- Based on the optimized parameter (Grid), we can see that significant improvements were achieved by tuning parameters such as Gini Index and Maximal Depth, whereas the pruning /prepruning, minimal leaf size, and minimal size for split provided minor improvements.

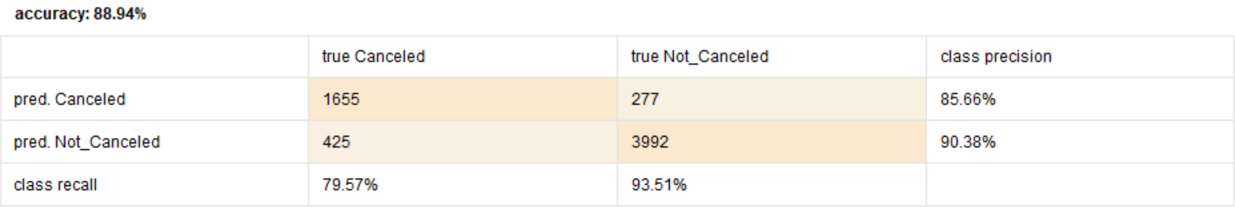
Note: *You can use more than one slide if needed*

Feature Importance Decision Tree (Pruned)



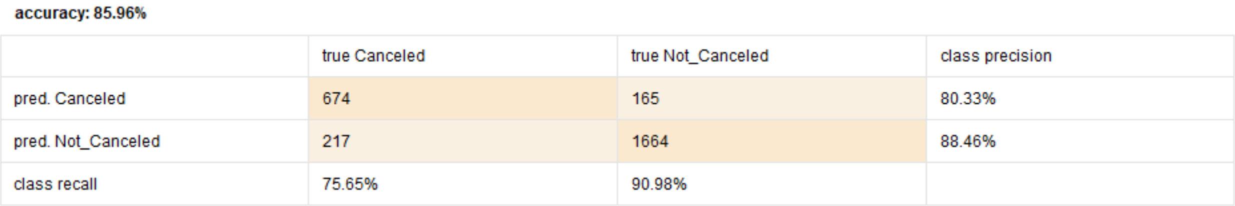
- Arrival month, lead time, average price per room, arrival date, number of weekend nights and number of the special requests.

Model Performance Evaluation and Improvement – Random Forest (Pruned)



accuracy: 88.94%

	true Canceled	true Not_Canceled	class precision
pred. Canceled	1655	277	85.66%
pred. Not_Canceled	425	3992	90.38%
class recall	79.57%	93.51%	



accuracy: 85.96%

	true Canceled	true Not_Canceled	class precision
pred. Canceled	674	165	80.33%
pred. Not_Canceled	217	1664	88.46%
class recall	75.65%	90.98%	

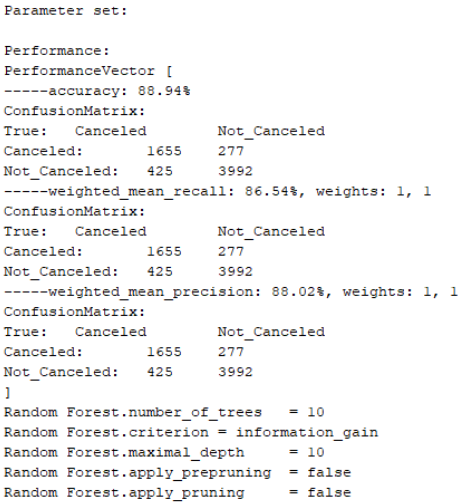
- Optimizing Model Parameters: **Criterion** (Gain Ratio, Gini Index, Information Gain), Max. Depth 1 to 10, **Apply pruning** true/false, confidence 0.1, **apply prepruning** true/false, **min. leaf size 2**, **min split size 4, no of prepruning alternatives 3,** use local random seed, local random seed 1

Model Performance Evaluation and Improvement - Random Forest (Pruned)

Model	Train Accuracy	Test Accuracy	Train Recall	Test Recall	Train Precision	Test Precision
Random Forest Pruned	88.94	85.96	86.54	83.31	88.02	84.04

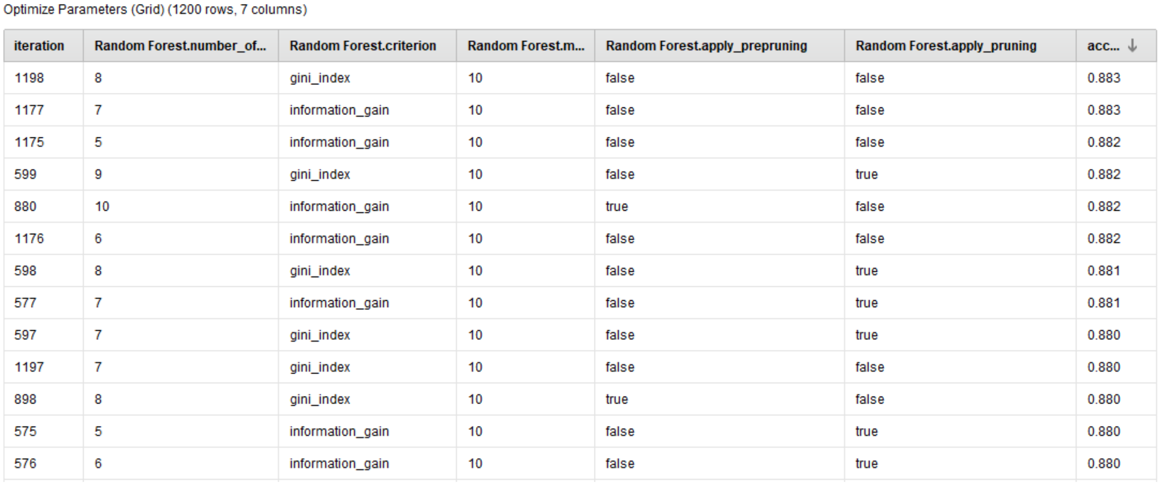
- After applying pruning and hyperparameter tuning to the Random Forest Model (Unpruned), we saw that this has slightly improved the values for all the metrics, by reducing the difference between the Training and Test dataset.
- Furthermore, we observed that the Random Forest (Pruned) model demonstrated good performance for the Precision of the True class, by achieving an increase of **80.33%**

Model Performance Evaluation and Improvement - Random Forest (Pruned)



```
Parameter set:

Performance:
PerformanceVector [
-----accuracy: 88.94%
ConfusionMatrix:
True:    Canceled        Not_Canceled
Canceled:        1655    277
Not_Canceled:    425     3992
-----weighted_mean_recall: 86.54%, weights: 1, 1
ConfusionMatrix:
True:    Canceled        Not_Canceled
Canceled:        1655    277
Not_Canceled:    425     3992
-----weighted_mean_precision: 88.02%, weights: 1, 1
ConfusionMatrix:
True:    Canceled        Not_Canceled
Canceled:        1655    277
Not_Canceled:    425     3992
]
Random Forest.number_of_trees    = 10
Random Forest.criterion = information_gain
Random Forest.maximal_depth      = 10
Random Forest.apply_prepruning   = false
Random Forest.apply_pruning      = false
```

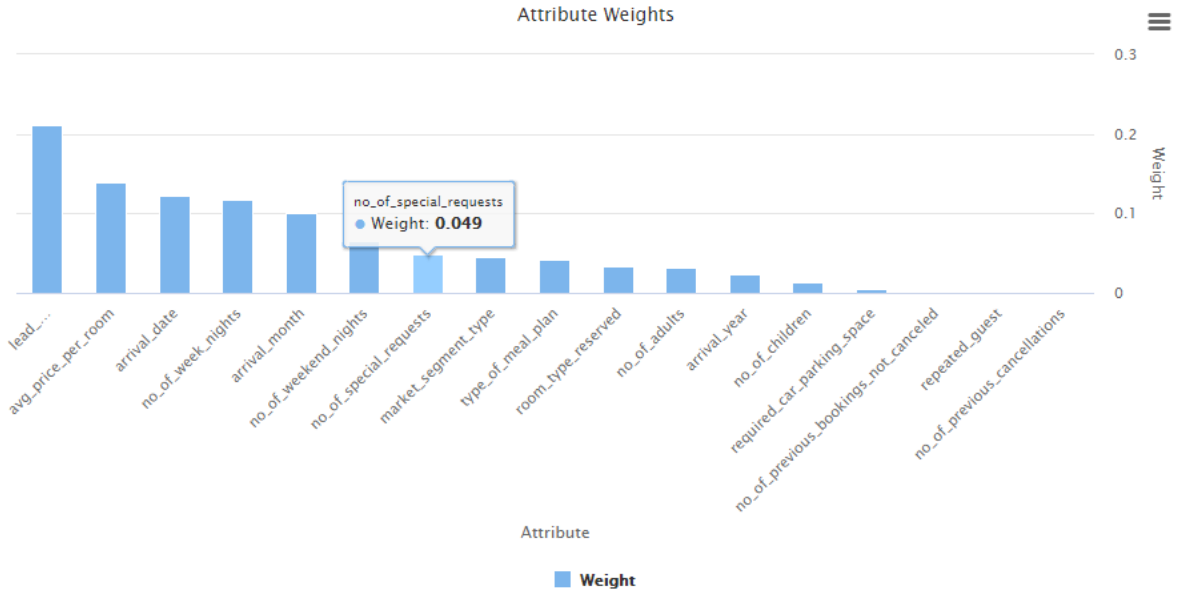


Optimize Parameters (Grid) (1200 rows, 7 columns)

iteration	Random Forest.number_of...	Random Forest.criterion	Random Forest.m...	Random Forest.apply_prepruning	Random Forest.apply_pruning	acc... ↓
1198	8	gini_index	10	false	false	0.883
1177	7	information_gain	10	false	false	0.883
1175	5	information_gain	10	false	false	0.882
599	9	gini_index	10	false	true	0.882
880	10	information_gain	10	true	false	0.882
1176	6	information_gain	10	false	false	0.882
598	8	gini_index	10	false	true	0.881
577	7	information_gain	10	false	true	0.881
597	7	gini_index	10	false	true	0.880
1197	7	gini_index	10	false	false	0.880
898	8	gini_index	10	true	false	0.880
575	5	information_gain	10	false	true	0.880
576	6	information_gain	10	false	true	0.880

- Similar to the Decision Tree hyperparameter tuning, we can see that significant improvements were achieved by tuning parameters such as Information Gain and Maximal Depth, whereas the pruning /prepruning, minimal leaf size, and minimal size for split provided minor improvements.

Feature Importance- Random Forest (Pruned)



- the most important features are Lead time, Average price per room, and arrival date remained the same as for the Random Forest (Default) model.

Happy Learning !