Predicting Hotel Reservation Cancellations using Machine Learning

Michael Hotaling

11/21/2020

Abstract

Last minute cancellations are a part of life. We've all had plans that have fallen through at one point in our lives, especially considering todays events, and we've had to cancel a plane ticket or hotel reservation at some point in our lives. Luckily for the consumer, many hotels offer free cancellation policies due to competition. Although it might seem like a minor thing, canceling reservations can become a costly expense for many hotels. For example, in 2018, nearly 40% of 40% of on-the-books revenue was lost due to cancellations. (Funnell, 2019) (Hertzfeld, 2019) (d-edge, 2019)

For my term project, I will be using a dataset containing hotel demand data between July 2015 and August 2017 for two hotels and a total of 12,000 observations. (Antonio, Ana, & Nunes, 2019)

Research Questions

a. Are hotel cancellations predictable based on certain data attributes?
b. Which attributes are best correlated with reservation cancellations?
c. Can we predict how many cancellations there will be based on the time of year?
d. Do most cancellations happen within a certain time frame (e.g. 6 weeks out)?
e. Will predicting cancellations provide any insightful data (cost savings)?
f. Will a model that predicts city hotel reservation cancellations work on data from a resort hotel and vice versa?

The Data

The data is comprised of several attributes, listed below:

```
##  [1] "IsCanceled"                  "LeadTime"
##  [3] "ArrivalDateYear"             "ArrivalDateMonth"
##  [5] "ArrivalDateWeekNumber"       "ArrivalDateDayOfMonth"
##  [7] "StaysInWeekendNights"        "StaysInWeekNights"
##  [9] "Adults"                      "Children"
## [11] "Babies"                      "Meal"
## [13] "Country"                     "MarketSegment"
## [15] "DistributionChannel"         "IsRepeatedGuest"
## [17] "PreviousCancellations"       "PreviousBookingsNotCanceled"
## [19] "ReservedRoomType"            "AssignedRoomType"
## [21] "BookingChanges"              "DepositType"
## [23] "Agent"                       "Company"
## [25] "DaysInWaitingList"           "CustomerType"
## [27] "ADR"                         "RequiredCarParkingSpaces"
## [29] "TotalOfSpecialRequests"      "ReservationStatus"
## [31] "ReservationStatusDate"       "Location"
```

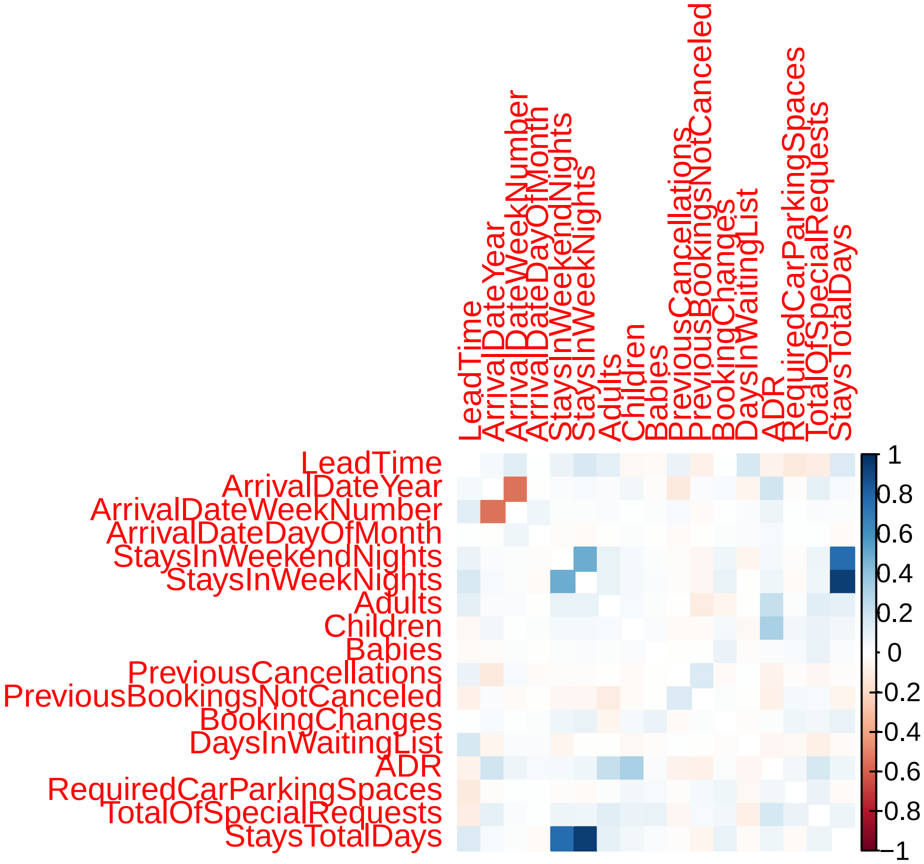
Data cleaning

Before starting, I need to clean the data up slightly. Some values should be converted to factors. I will also be engineering some extra attributes to help us determine when customers book their reservations and when they plan on checking out.

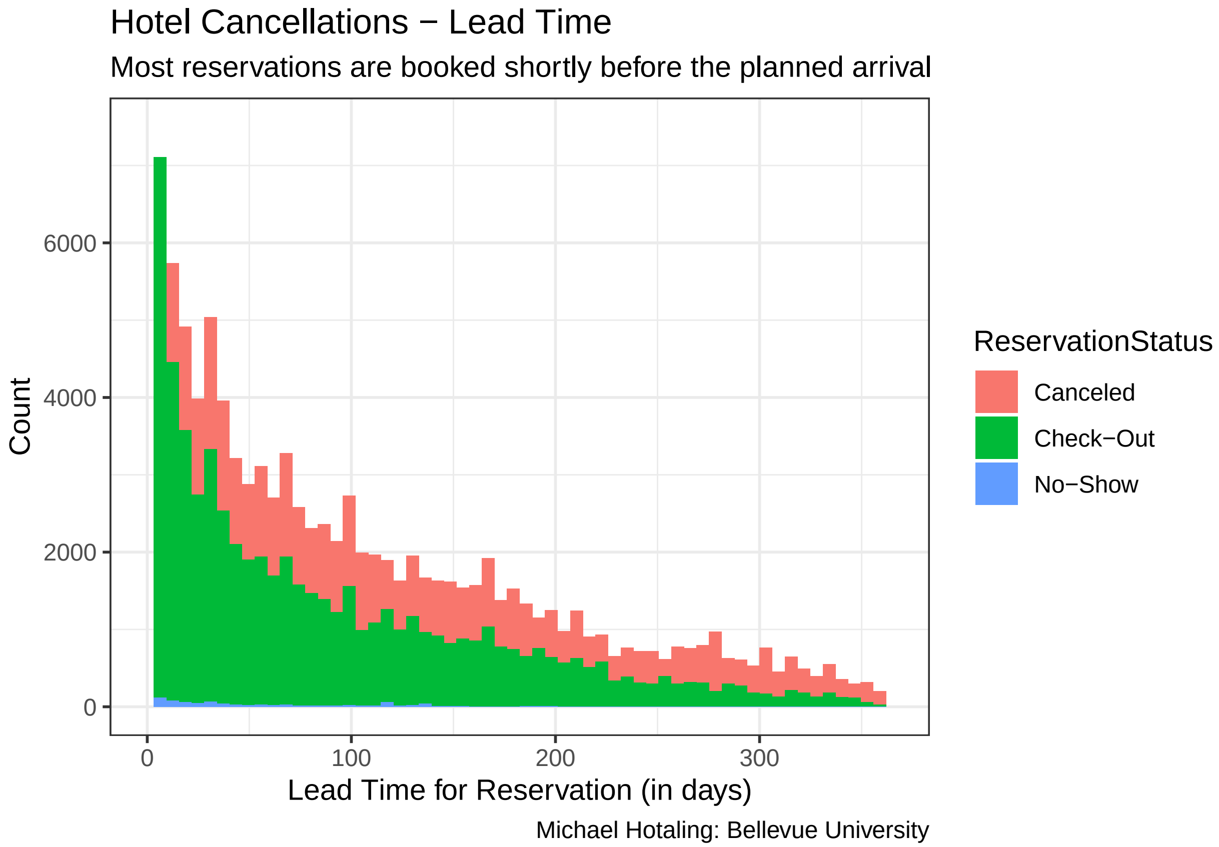
EDA

We will first explore this data using some EDA. For this exercise, I'll be utilizing ggplot2 and corrplot.

Let's first check to see if there are any strong correlations in our data. We can use corrplot to create a correlation map for the numeric values in our dataset.

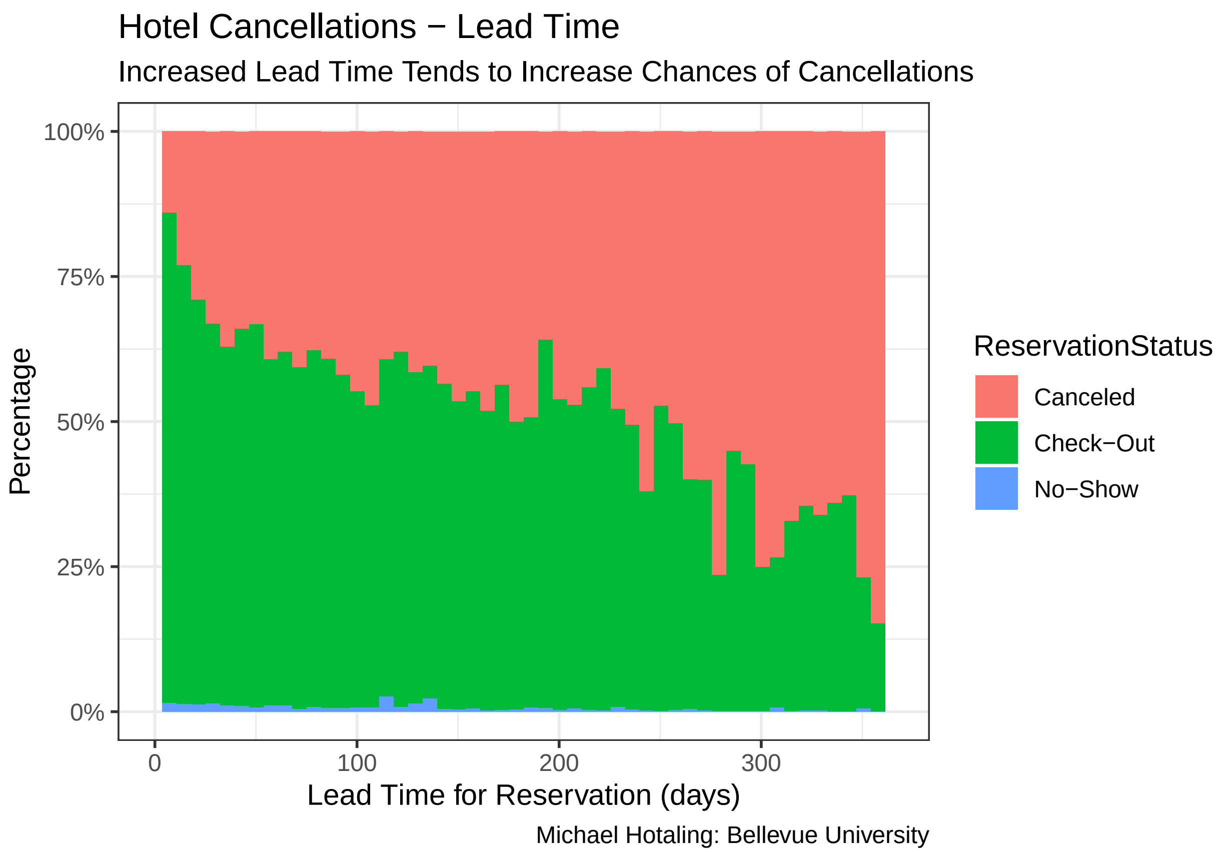


There aren't too many highly correlated values in our dataset, which should prevent any collinearity. Now let's investigate the lead-time for our reservations.


Hotel Cancellations − Lead Time

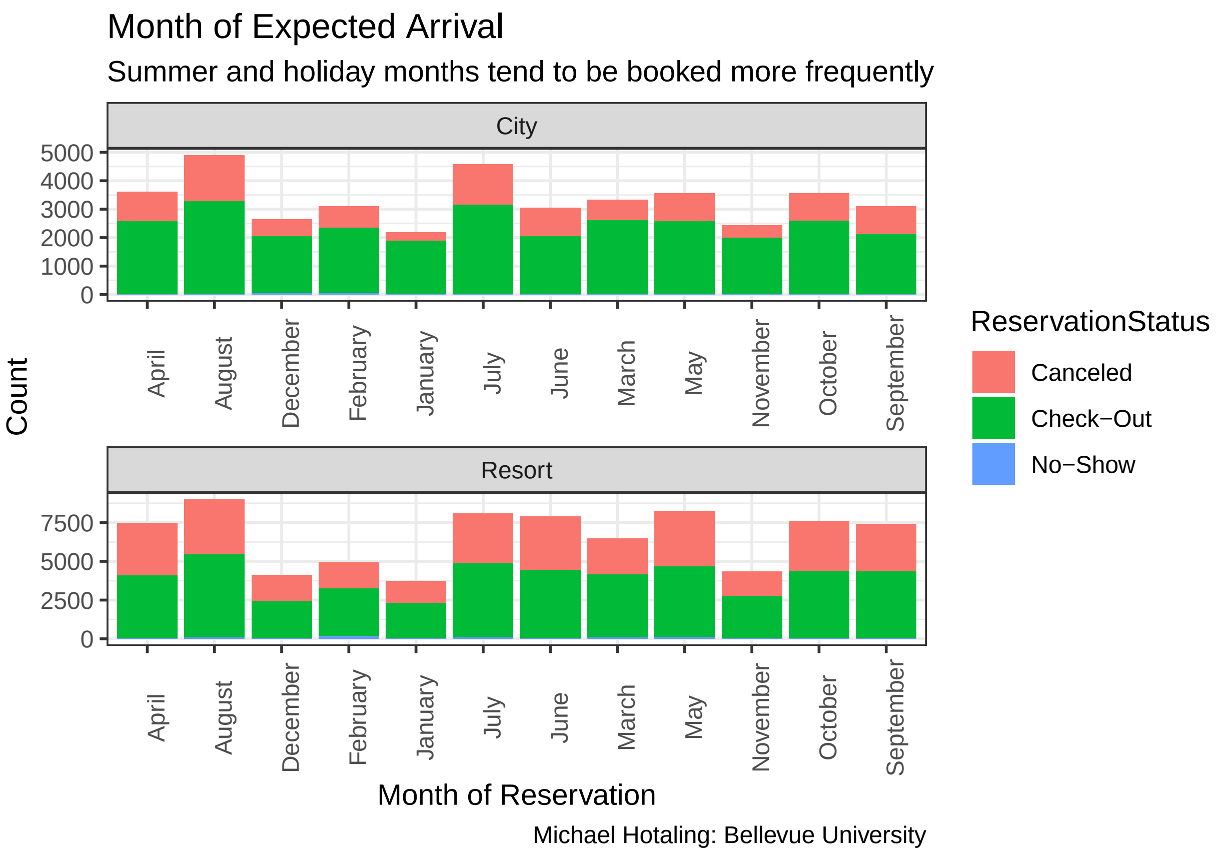
Most reservations are booked shortly before the planned arrival

Michael Hotaling: Bellevue University



From this data, we can see that most of our reservations are made closer to the date of check-in than not. We can also see that reservations made farther out are more likely to fall through.

Next, let's example the time of year people plan to arrive to the hotel.

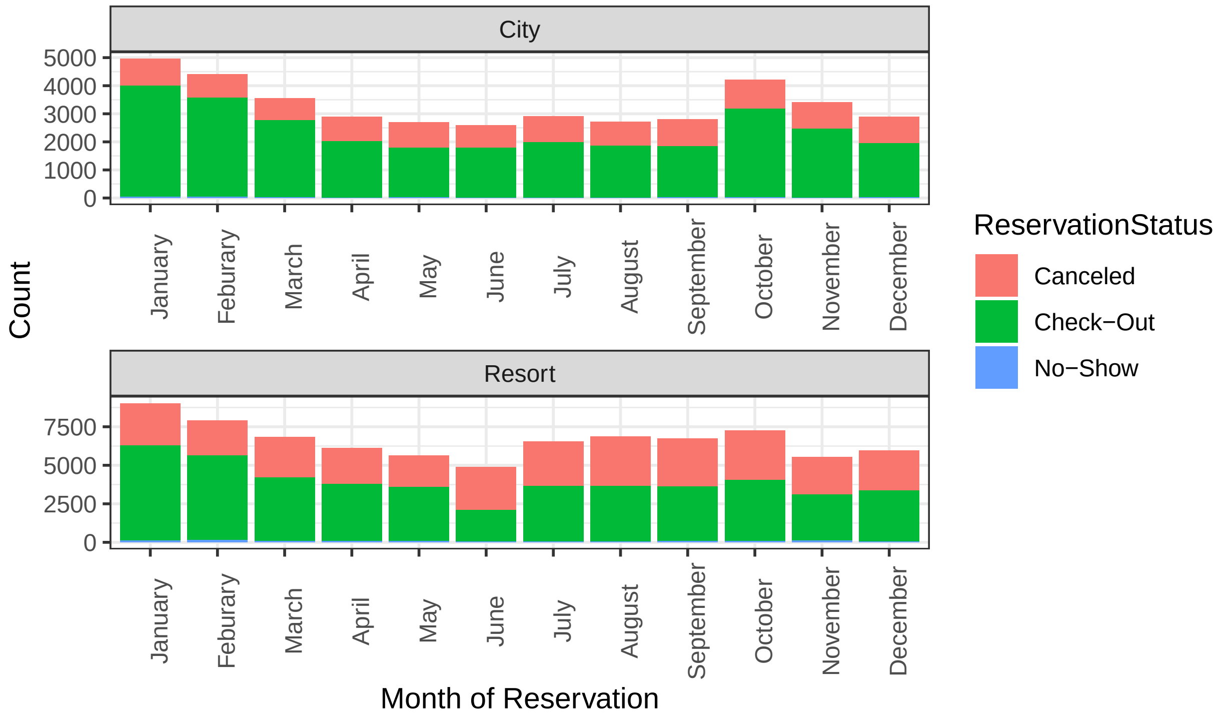


From this graphic, we can see that the most popular times of the year are the summer and holiday months, especially for the resort hotel.

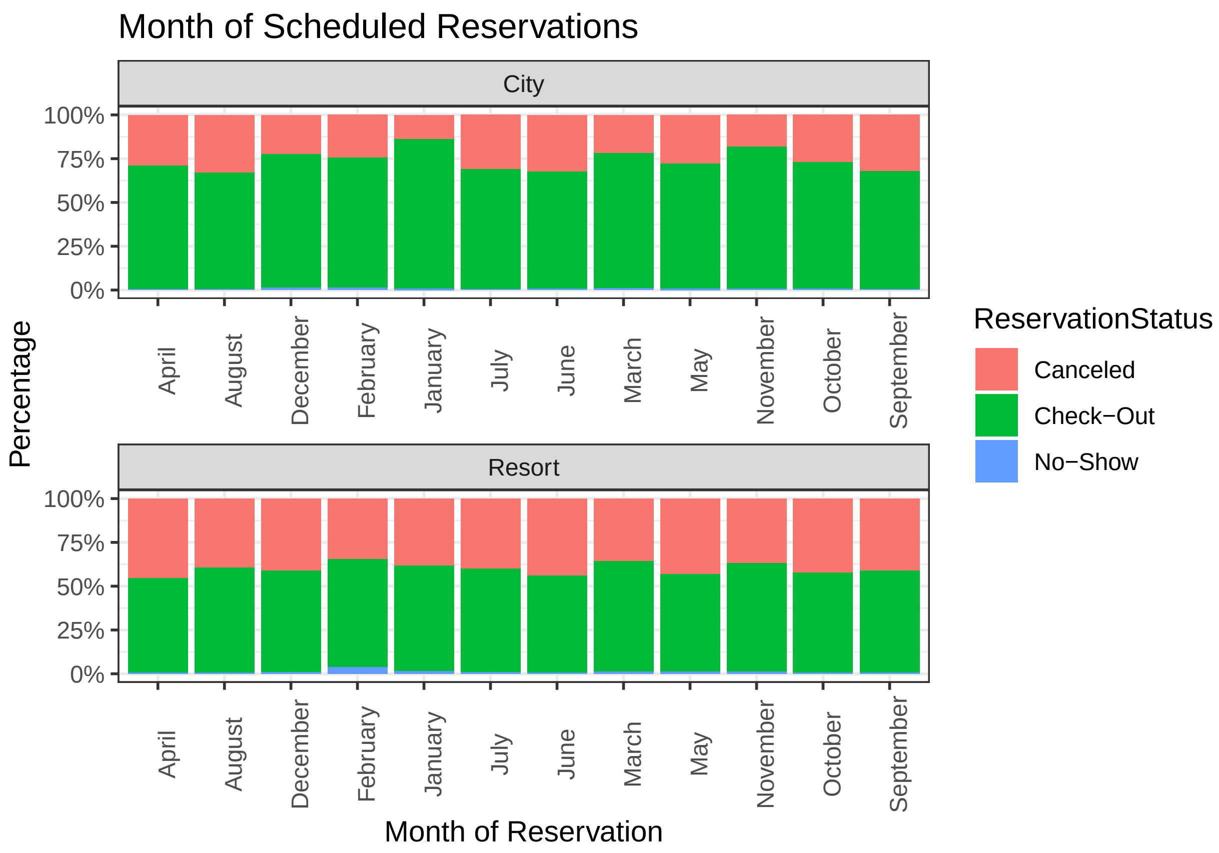
Building off of the lead time and date of reservation, we can investigate to see when most reservations are made.

Which Month Reservations are Made

Most reservations are made in the Winter

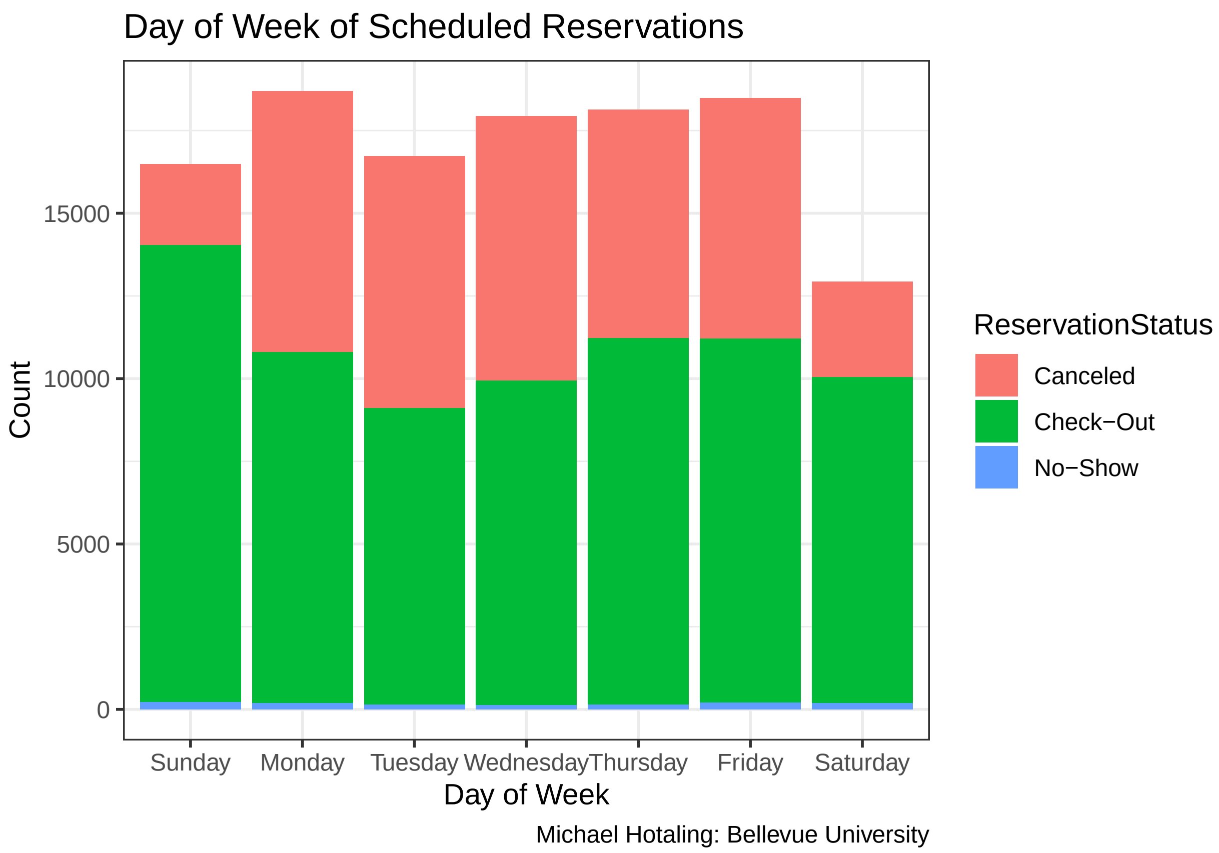


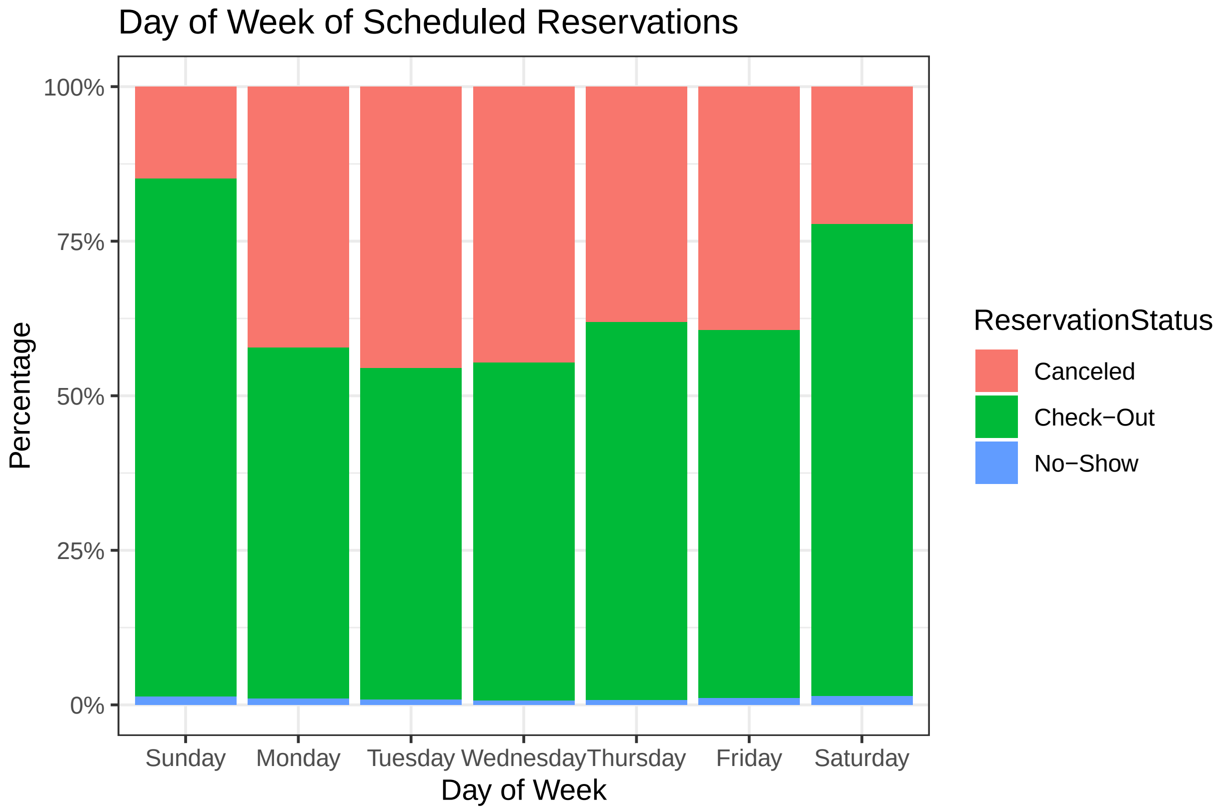
Michael Hotaling: Bellevue University



From this graphic, we can see more reservations are made in the winter months and the time of year the booking was made doesn't have any impact on our cancellation rates.

Similar to the above, we can see what days are reservation are made for.


Day of Week of Scheduled Reservations
Count
15000
10000
5000
0
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Day of Week
ReservationStatus
Canceled
Check−Out
No−Show
Michael Hotaling: Bellevue University



Michael Hotaling: Bellevue University

We seem to have reservations for almost every day of the week, but reservations are less likely to be canceled on Saturdays and Sundays rather than the weekdays.

Machine Learning

Since we will be determining whether or not a reservation will be canceled, a logistic model can be use in conjunction with our data. We can split our data up into training and testing subsets to verify the model is accurate.

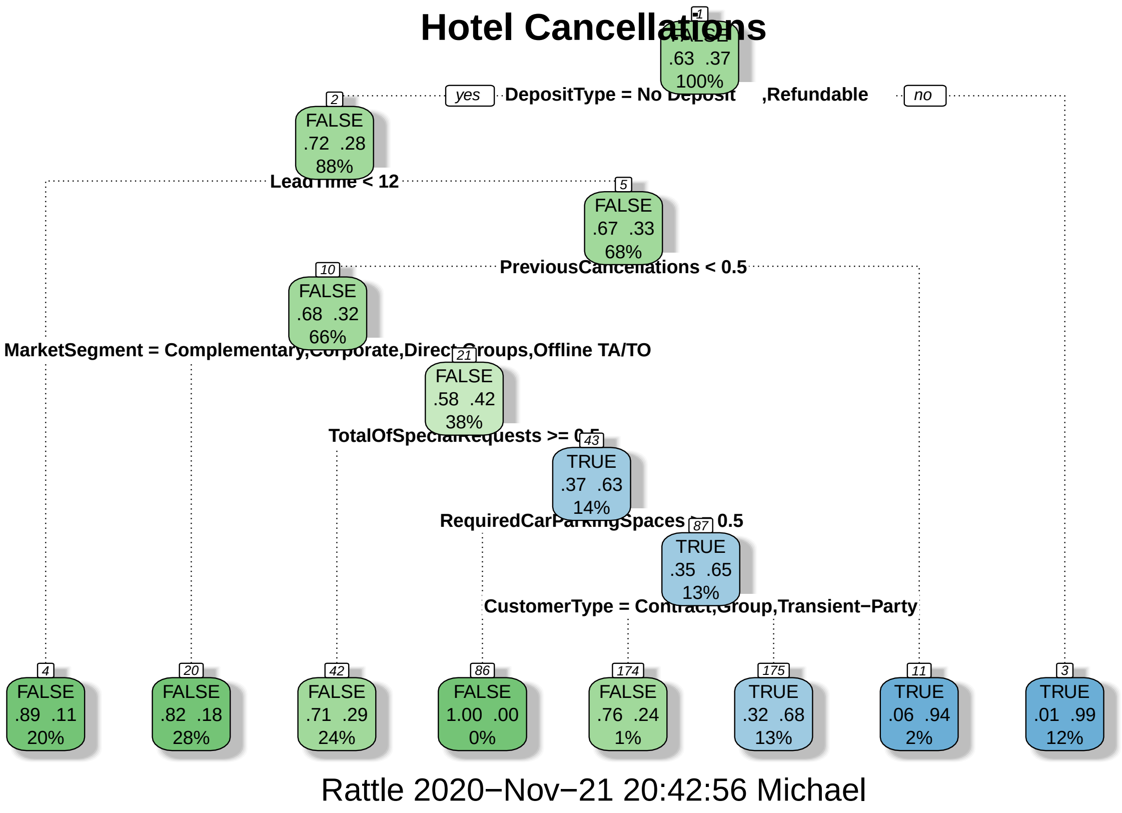
```
##
## Call:
## glm(formula = IsCanceled ~ LeadTime + StaysInWeekendNights +
##     StaysInWeekNights + Adults + IsRepeatedGuest + PreviousCancellations +
##     PreviousBookingsNotCanceled + BookingChanges + RequiredCarParkingSpaces +
##     TotalOfSpecialRequests + ReservationDayName + DayResMadeName +
##     MonthResMade + CheckoutDateName, family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -8.4904  -0.8516  -0.4462   0.8941   5.2648
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -2.711e+00  6.125e-02 -44.259  < 2e-16 ***
## LeadTime                     4.374e-03  8.639e-05  50.632  < 2e-16 ***
## StaysInWeekendNights        -2.739e-01  1.088e-02 -25.174  < 2e-16 ***
## StaysInWeekNights            1.249e-01  5.491e-03  22.743  < 2e-16 ***
## Adults                       3.374e-01  1.813e-02  18.614  < 2e-16 ***
## IsRepeatedGuestTRUE         -1.524e+00  9.694e-02 -15.725  < 2e-16 ***
## PreviousCancellations        2.687e+00  6.843e-02  39.269  < 2e-16 ***
## PreviousBookingsNotCanceled -4.463e-01  2.739e-02 -16.298  < 2e-16 ***
## BookingChanges              -6.001e-01  1.876e-02 -31.992  < 2e-16 ***
## RequiredCarParkingSpaces    -2.876e+03  9.154e+05  -0.003 0.997494
## TotalOfSpecialRequests      -6.385e-01  1.238e-02 -51.557  < 2e-16 ***
## ReservationDayNameMonday     1.444e+00  3.554e-02  40.623  < 2e-16 ***
## ReservationDayNameTuesday    1.753e+00  3.853e-02  45.490  < 2e-16 ***
## ReservationDayNameWednesday  1.562e+00  3.752e-02  41.615  < 2e-16 ***
## ReservationDayNameThursday   1.180e+00  3.739e-02  31.570  < 2e-16 ***
## ReservationDayNameFriday     1.309e+00  3.554e-02  36.842  < 2e-16 ***
## ReservationDayNameSaturday   5.698e-01  3.863e-02  14.749  < 2e-16 ***
## DayResMadeNameMonday        -6.342e-02  3.236e-02  -1.960 0.050002 .
## DayResMadeNameTuesday        9.203e-03  3.185e-02   0.289 0.772626
## DayResMadeNameWednesday     -7.321e-02  3.190e-02  -2.295 0.021760 *
## DayResMadeNameThursday      -1.614e-01  3.101e-02  -5.206 1.93e-07 ***
## DayResMadeNameFriday        -2.746e-01  3.150e-02  -8.718  < 2e-16 ***
## DayResMadeNameSaturday      -2.493e-01  3.134e-02  -7.955 1.79e-15 ***
## MonthResMadeFeburary         4.914e-02  3.771e-02   1.303 0.192569
## MonthResMadeMarch            3.834e-01  3.863e-02   9.926  < 2e-16 ***
## MonthResMadeApril            5.954e-01  3.981e-02  14.956  < 2e-16 ***
## MonthResMadeMay              5.694e-01  4.071e-02  13.987  < 2e-16 ***
## MonthResMadeJune             1.103e+00  4.209e-02  26.212  < 2e-16 ***
## MonthResMadeJuly             6.026e-01  3.906e-02  15.426  < 2e-16 ***
## MonthResMadeAugust           7.254e-01  3.927e-02  18.475  < 2e-16 ***
## MonthResMadeSeptember        5.767e-01  3.972e-02  14.520  < 2e-16 ***
## MonthResMadeOctober          3.387e-01  3.746e-02   9.041  < 2e-16 ***
## MonthResMadeNovember         6.584e-01  3.949e-02  16.674  < 2e-16 ***
## MonthResMadeDecember         7.405e-01  3.968e-02  18.663  < 2e-16 ***
## CheckoutDateNameMonday       1.681e-02  3.277e-02   0.513 0.607952
## CheckoutDateNameTuesday     -2.261e-01  3.329e-02  -6.792 1.10e-11 ***
## CheckoutDateNameWednesday   -1.376e-01  3.377e-02  -4.075 4.59e-05 ***
## CheckoutDateNameThursday    -1.178e-01  3.245e-02  -3.629 0.000284 ***
## CheckoutDateNameFriday      -7.487e-02  3.158e-02  -2.371 0.017750 *
## CheckoutDateNameSaturday     1.447e-02  3.150e-02   0.459 0.645917
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 110179  on 83572  degrees of freedom
## Residual deviance:  83746  on 83533  degrees of freedom
## AIC: 83826
##
## Number of Fisher Scoring iterations: 12
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE  TRUE
##      FALSE 20029  5829
##      TRUE   2521  7438
##
```

```
##                Accuracy : 0.7669
##                  95% CI : (0.7625, 0.7712)
##     No Information Rate : 0.6296
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.4731
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.8882
##             Specificity : 0.5606
##          Pos Pred Value : 0.7746
##          Neg Pred Value : 0.7469
##              Prevalence : 0.6296
##          Detection Rate : 0.5592
##    Detection Prevalence : 0.7219
##       Balanced Accuracy : 0.7244
##
##        'Positive' Class : FALSE
##
```

We have a accuracy rating of about 75%, but many false positive errors.

We can try another machine learning algorithm known as the Decision Tree to attempt to get a better score.



```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE  TRUE
##      FALSE 20992  5246
##      TRUE   1558  8021
##
##                Accuracy : 0.81
##                  95% CI : (0.8059, 0.8141)
##     No Information Rate : 0.6296
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.568
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9309
##             Specificity : 0.6046
##          Pos Pred Value : 0.8001
##          Neg Pred Value : 0.8374
##              Prevalence : 0.6296
##          Detection Rate : 0.5861
##    Detection Prevalence : 0.7326
##       Balanced Accuracy : 0.7677
##
##        'Positive' Class : FALSE
##
```

Using a Decision Tree, we were able to reclaim 5% more accuracy, increasing our total accurate to 81%. We were also able to reduce the amount of false positive errors in our analysis.

Answering the Research Questions.

Research Questions

a. Are hotel cancellations predictable based on certain data attributes?

we were able to achieve an accuracy of 81% when attempting to predict whether or not a customer might cancel their reservation, indicating that hotel cancellations are predictable to some extent. Additional data might help us improve accuracy.

b. Which attributes are best correlated with reservation cancellations?

Since we made two different models, we can go over each one.

The logistic model was able to use most of the attributes we fed it. The attributes that were most highly correlated with cancellation were lead-time, number of nights stayed, number of adults, if the customer was a repeated guest and didn't cancel a booking before, The day the booking was made for, the time of month the booking is for, and several others. I believe that the number of attributes we passed into the model was much too high, and it might have caused some overfitting. Going back over the data and analyzing some of the redudant attributes might help the model become more accurate

the Decision Tree only uses a few attributes, such as Deposit Type, Lead Time, Market segmentation, and a few others. The fact that the model can achieve a much better fit to our test data in comparison to our logistic model is quite impressive

c. Can we predict how many cancellations there will be based on the time of year?

I originally wanted to create a Poisson Regression model to predict the seasonality and cyclicality of reservation cancellations, but I wasn't able to do it. This is something that I would be interesting in solving at a later time, but for now, we will leave this question open.

d. Do most cancellations happen within a certain time frame (e.g. 6 weeks out)?

When I first was analyzing this data, I didn't realize the date of cancellation wasn't included. I won't be able to answer this question without having access to the date of cancellation. We could in theory create a survival analysis using that data which would provide an interesting insight into more risk assessment models for our customers.

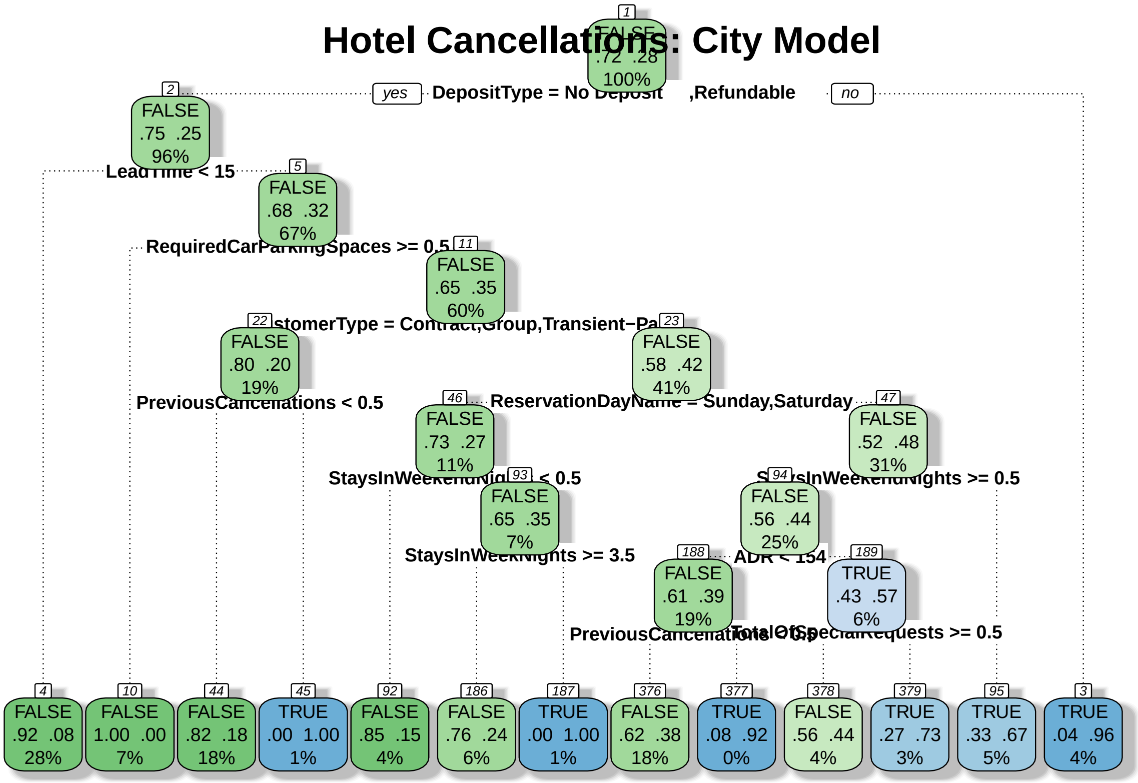
We can see from the first two EDA graphics that reservations were much more likely to fall through as time went on, but there isn't a general cutoff that I was expecting to see.

e. Will predicting cancellations provide any insightful data (cost savings

Being able to predict whether or not a customer might cancel can have a huge economic impact. Typically, when a reservation is made, the room is locked until that customer leaves or the customer cancels their reservation. If a customer cancels their reservation last minute and customers which were looking for a room before weren't able to get one, revenue is lost. If we can successfully predict whether or not a customer will cancel their reservation, we can use that data to "soft reserve" a room by still allowing other bookings. When the customer does cancel, the revenue from that room isn't lost.

f. Will a model that predicts city hotel reservation cancellations work on data from a resort hotel and vice versa?

We can easily test this by creating the models and comparing their performance on data from the opposing hotel.

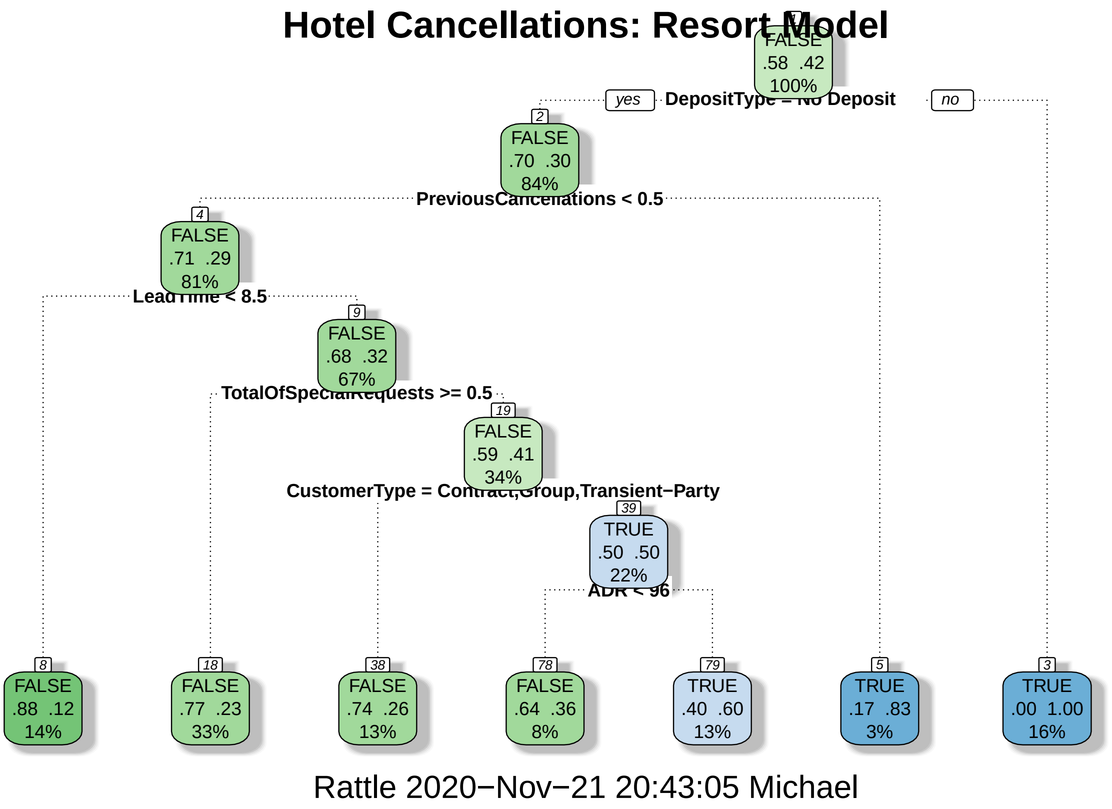


Rattle 2020−Nov−21 20:42:59 Michael

```
## Confusion Matrix and Statistics
##
##             Reference
## Prediction FALSE  TRUE
##      FALSE 43225 12556
##      TRUE   3003 20546
##
##                Accuracy : 0.8039
##                  95% CI : (0.8011, 0.8066)
##     No Information Rate : 0.5827
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5795
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9350
##             Specificity : 0.6207
##          Pos Pred Value : 0.7749
##          Neg Pred Value : 0.8725
##              Prevalence : 0.5827
##          Detection Rate : 0.5449
##    Detection Prevalence : 0.7032
##       Balanced Accuracy : 0.7779
##
##        'Positive' Class : FALSE
##
```

Our City Model results work pretty well on our resort hotel data. we have an accuracy of about 80%.

Let's try building our model using resort hotel data and testing it against the city hotel.



Hotel Cancellations: Resort Model

Rattle 2020−Nov−21 20:43:05 Michael

```
## Confusion Matrix and Statistics
```

```
##
##           Reference
## Prediction FALSE  TRUE
##      FALSE 26572  6762
##      TRUE   2366  4360
##
##                Accuracy : 0.7721
##                  95% CI : (0.768, 0.7762)
##     No Information Rate : 0.7224
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.3532
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9182
##             Specificity : 0.3920
##          Pos Pred Value : 0.7971
##          Neg Pred Value : 0.6482
##              Prevalence : 0.7224
##          Detection Rate : 0.6633
##    Detection Prevalence : 0.8321
##       Balanced Accuracy : 0.6551
##
##        'Positive' Class : FALSE
##
```

Our model is still relatively accurate at 77% accuracy, but we have a lot more false positive values than our previous model.

We can also try to use our logistic regression regression to see if we get different results.

Model Trained on City Data:

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE  TRUE
##      FALSE 42424 16681
##      TRUE   3804 16421
##
##                Accuracy : 0.7418
##                  95% CI : (0.7387, 0.7448)
##     No Information Rate : 0.5827
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.438
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9177
##             Specificity : 0.4961
##          Pos Pred Value : 0.7178
##          Neg Pred Value : 0.8119
##              Prevalence : 0.5827
##          Detection Rate : 0.5348
##    Detection Prevalence : 0.7451
##       Balanced Accuracy : 0.7069
##
##        'Positive' Class : FALSE
##
```

Model Trained on Resort Data:

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE  TRUE
##      FALSE 23676  4987
##      TRUE   5262  6135
##
##                Accuracy : 0.7442
##                  95% CI : (0.7399, 0.7484)
##     No Information Rate : 0.7224
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.367
##
##  Mcnemar's Test P-Value : 0.006799
##
##             Sensitivity : 0.8182
##             Specificity : 0.5516
##          Pos Pred Value : 0.8260
##          Neg Pred Value : 0.5383
##              Prevalence : 0.7224
##          Detection Rate : 0.5910
##    Detection Prevalence : 0.7155
##       Balanced Accuracy : 0.6849
##
##        'Positive' Class : FALSE
##
```

Both models show around 75% accuracy, which is about where our other model was.

Both models seem to show some general accuracy as oppose to random pick, but if deployed at a real hotel, the model should be used on that hotels data since the reason for booking the hotel may be different depending on the location of the hotel.

Resources

d-edge. (2019, October 4). HOW ONLINE HOTEL DISTRIBUTION IS CHANGING IN EUROPE: A Deep-dive into European Hotel Distribution trends 2014-2018. Retrieved from d-edge: Hospitality Solutions: https://www.d-edge.com/how-online-hotel-distribution-is-changing-in-europe/

Funnell, R. (2019, May 10). The real cost of 'free' cancellations. Retrieved from Triptease: https://triptease.com/blog/the-real-cost-of-free-cancellations/

Hertzfeld, E. (2019, April 23). Study: Cancellation rate at 40% as OTAs push free change policy. Retrieved from Hotel Management: https://www.hotelmanagement.net/tech/study-cancelation-rate-at-40-as-otas-push-free-change-policy

Ward, B. (2019, Feburary 12). Introduction to Tidyverse : readr, tibbles, tidyr & dplyr. Retrieved from Medium.com: https://medium.com/@brianward1428/introduction-to-tidyverse-7b3dbf2337d5