



# XYZ TOURS DATA MINING ANALYSIS

Data Diggers

Austin Alcott, Matt Ashcroft,  
Adhiraj Roka, Rui Zhang

## I. Introduction

Data Diggers were tasked with a data mining operation for XYZ Tours. The company wished to learn salient patterns about its tours' services and customers, specifically those who are most likely to rebook another tour within 1 year of their trip. The key goals laid out by XYZ tours were:

1. To identify the key-factors (characteristics) that make the 15-25% of the customers to repeat their travel with us.
2. To profile and understand tour customers and then identify prospects from those customers who will likely repeat their travel with us within a year.

Data from 2005-2007 was provided to Data Diggers across 93 variables including age, tour ratings, tour season, tour location, etc. Data was split into a modeling and scoring set. The modeling set was used to build, train, validate, and test the models. It consisted of over 23,000 observations from the three tour years. R Studio was used to perform all parts of the data mining.

The team performed the data mining operation in two phases. In the first phase, a data audit and descriptive analysis was completed. This included cleaning the data, transforming variables, consolidating variables, and choosing salient variables for use in phase II modeling. In the second phase, several exploratory models were built and evaluated for their performance. The team employed many techniques to incrementally improve each of the models. Finally, the best one was selected and applied to the scoring dataset.

In this report, Data Diggers briefly recap the phase I summary that was already delivered and detail the work that was performed in phase II. The final model is given to XYZ tours for use in future forecasting. Finally, a business-oriented descriptive analysis shows how XYZ tours can best utilize the findings of the team. Suggestions for business strategy moving forward accompany this analysis.

## II. Phase I overview

### a. Variable Types

Data auditing and analysis began by checking the structure of each variable. Many nominal variables were initially characterized as numeric, so these had to be changed. The target variable, Book\_12Mo, which indicates whether a customer booked another tour within 12 months of the first, was one of these that needed to be changed. In addition, some nominal variables would be better represented as ordinal ones. For example, rating variables were converted from numeric to nominal and then to ordinal. This is significant for decision tree models which are discussed in the phase II section.

### b. Missing Values

Next, the dataset was searched for missing values represented by {".", "NA", "", "?"}. Out of the 23,459 observations and 93 variables, only 2% was missing. This is a remarkably complete dataset. However, in some variables there existed extraneous levels or categories that should not be there. Some variables that have 1 to 5 scales also included several hundred responses of zero. Others contained negative values. These represent non-responses, not missing values. These levels were consolidated with the other level which had a positive response rate closest to that of the extraneous level. For example, TD\_Overall measures the customer's satisfaction with their tour director on a scale of 1 to 4. The modeling set also contained 264 non-responses for the variable. It turned out that the response rate of those variables was most similar to customers that responded 4 – Excellent. Therefore, the non-responses were combined with that group. Other “yes-no” variables had extraneous levels that were combined with the no response. One such variable was Travel\_Again.

### c. Extraneous Levels

The distribution of positive and negative responses against each category in the nominal variables was checked with mosaic plots. The mosaic plots gave the team an initial idea of which variables might become important to the models. Figure 1 shows the mosaic plot of age. This plot shows that the majority of XYZ Tours' customers are at or near retirement age, and those between the ages of 60-69 are most likely to book another tour within 12 months. More consolidation techniques were explored as methods to improve model results later in the process. These are explained in the "attempts at improvement" section under phase II.

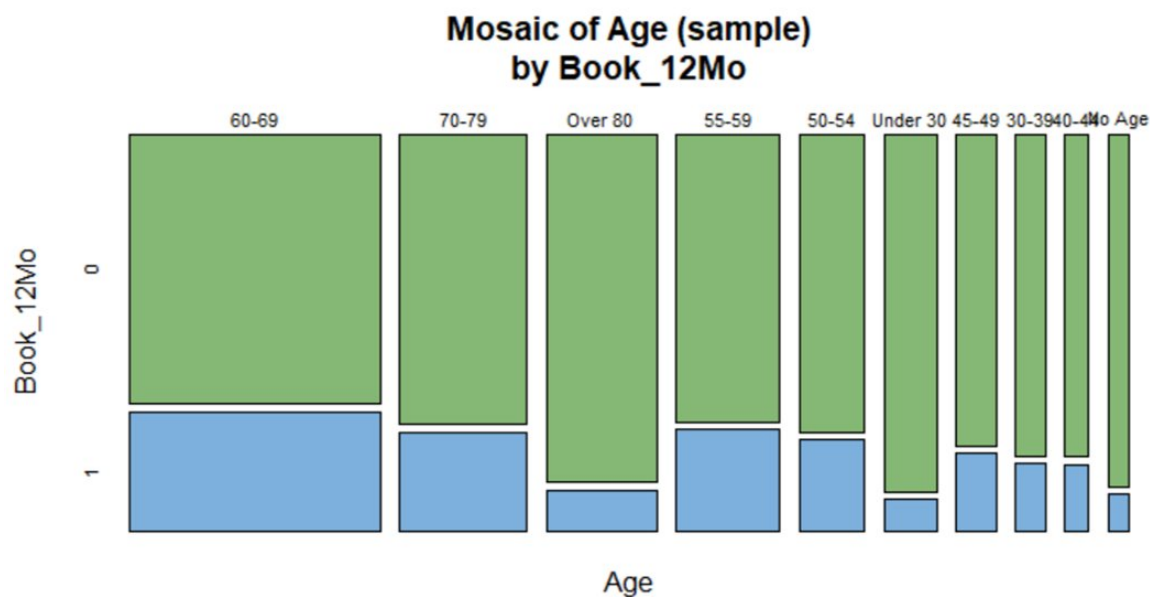


Figure 1: Mosaic plot of age against the Target variable, Book\_12Mo. Older customers are more common and more likely to book until they exceed 80 years old.

#### d. Distributions and transformations

Variable distributions were analyzed with histograms and by non-visual means. If the distribution of a variable was particularly skewed, a transformation was applied to normalize it. In all cases, the Yeo-Johnson transformation was preferred. Transformation was needed on fewer than 10 of the 93 variables in the dataset as seen in Figure 2. The *log* transformation was applied to the variables in the range of -1.5 to 1.5 in the beginning.

```

> TransformParams <- preProcess(tours.xf[split,indx], method=c("YeoJohnson"))
> TransformParams$y
  Grp_Size      Capacity  Fair_Hotels  Fair_Meals
1.140564965 1.460438494 -2.206162564 -2.735666106
Good_Hotels  Good_Meals  Good_GUSS  Good_Optionals
-0.007491901 -0.183275398 -0.861248661 -0.972375834
Excellent_Hotels Excellent_Meals Excellent_GUSS Excellent_Optionals
-0.071361341 -0.406811880 -0.224762937 -0.007475889
Optionals
0.115475965

```

Figure 2: Variables that were transformed with the Yeo-Johnson transformation

#### e. Time variables:

The team converted the time variables to continuous using 'chron' package. This allowed for better visualization of the data. In the graph below, we notice that while there was a lot of variation in depart time (from 6 AM to 7 PM), most tourists arrived at their destination at around 10-11 A.M.

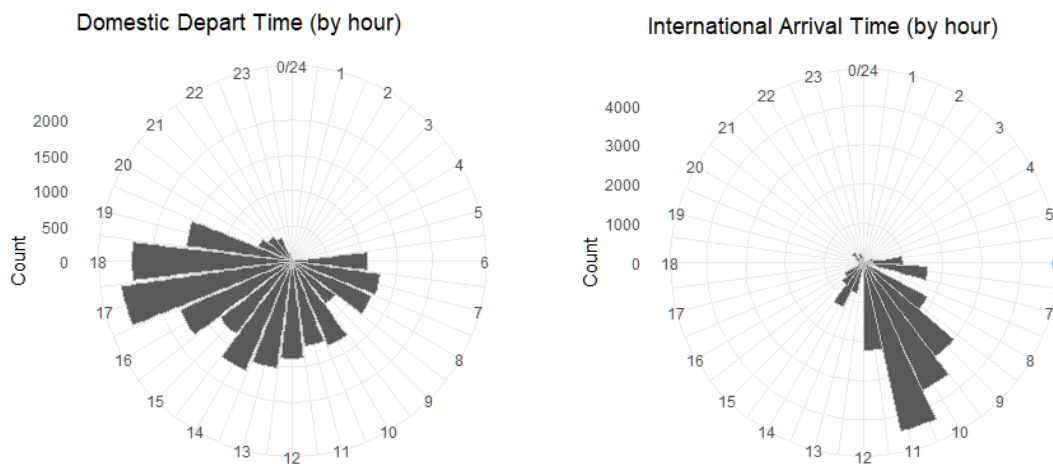


Figure 3: Arrival and departure times

#### f. Variable Importance

The final activity in phase I was variable importance. All variables were ranked for importance using Chi-square, T-statistic, and ROC index. The ROC index was particularly useful because it allows the comparison of both numeric and nominal variables. The top 10 most important variables by ROC index are shown in Table 1.

Table 1: Variable importance by ROC value

Variable Name	ROC Value
Past_Trips	0.6640
Email	0.6340
DB_Enter_Months	0.5754
Overall_Impression	0.5539
Age	0.5531
Pre_Departure	0.5448
Pax_Category	0.5388
Reference	0.5379
Hotels_Avg	0.5346
Excellent_Hotels	0.5329

### III. Phase II

#### a. Model starting points

#### i. Decision Tree

For the Decision Tree model, the team used alternative cutoff/ threshold and f-score (*this was used for all model assessments*) to see how the model performed. Initially, about 43 variables were selected from the variable importance test for this model.

threshold specificity sensitivity  
best 0.2107677 0.7829525 0.4935263

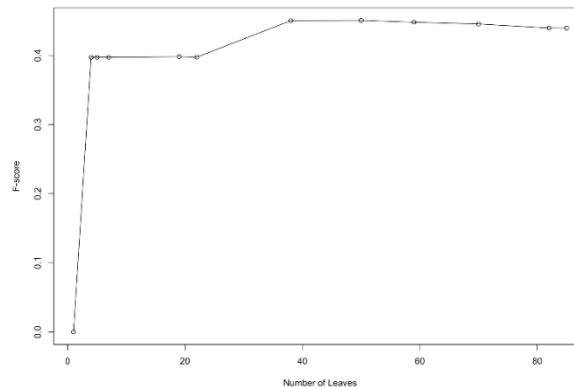


Figure 4: Decision tree pruning and selection

The graph in figure 4 above shows the diagram of cp-table with the number of leaves and corresponding f-scores. This analysis was used this to prune the final decision tree model.

The decision tree achieved an f-score of 0.45. The model had less than 50% sensitivity. It performed similar on the test dataset as well. Due to poor performance, the team decided to move forward with other models.

```
Confusion Matrix and Statistics

tree.class   0   1
0  3564  665
1   988  648

      Accuracy : 0.7182
      95% CI   : (0.7065, 0.7296)
    No Information Rate : 0.7761
    P-Value [Acc > NIR] : 1

      Kappa : 0.2542

  Mcnemar's Test P-Value : 2.377e-15

    Sensitivity : 0.4935
    Specificity : 0.7830
```

Figure 5: Performance statistics of the first decision tree

## ii. Logistic Regression

```
Call:
glm(formula = Book_12Mo ~ Past_Trips + Email + Return_Domestic_Gateway_Groups +
    Age_Con + TourCode_Groups + Overall_Impression + TourDate_Con +
    TravelAgain + Domestic_Depart_Time_Con + State_Con, family = binomial,
    data = tours.mdl[split, vars])

F1
0.5065862
> confusionMatrix(table(reg.class, new.tours[split.valid,]$Book_12Mo),
+   positive = "1")
Confusion Matrix and Statistics

reg.class    0    1
0  3144   390
1  1408   923
```

Figure 6: F-score and confusion matrix from logistic regression

Figure 6 shows the results of stepwise selection, and the confusion matrix for the model. The sensitivity is 0.70; however, this model gives too many false positives.

## iii. ANN

At beginning, our team used 2 hidden layers for ANN, and each layer has 16 neurons. As results, we had overfitting issue. The validation loss increases while training loss decreases.

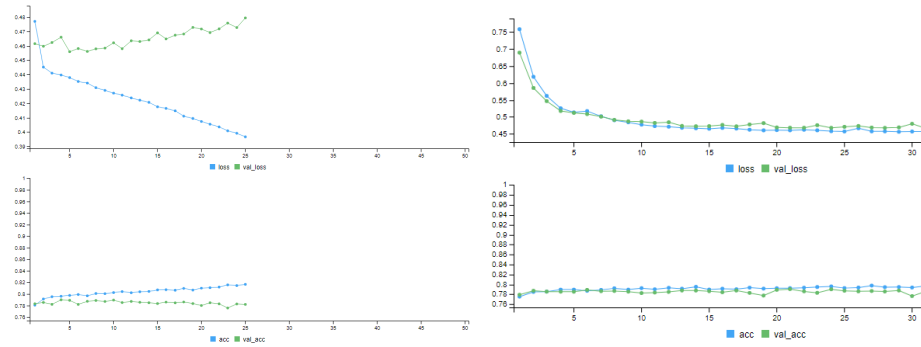


Figure 7: Left: Overfitting issue, right: after adding L2 regularization and dropout function

This issue is solved by applying L2 regularization and using dropout function. L2 regularization is also

called ridge regression.  $\lambda \sum_{j=1}^P \beta_j^2$  is added as penalty term to the loss function. L2 regularization is added into each layer. Moreover, the parameter of dropout function is set as 0.2, which means 20% inputs will be dropped before they go into next layer.

The F-score and confusion matrix is shown in figure 8. Compare with logistic regression, the F score of ANN is slightly improved. The sensitivity of ANN model is a little lower than the sensitivity of regression model, but the false positives are much less in ANN model.

```
F1
0.5150215
> confusionMatrix(table(ann.class,tours.ann
+ positive = "1")
Confusion Matrix and Statistics

ann.class   0   1
 0  3269  413
 1  1282   900

      Accuracy : 0.7109
      95% CI   : (0.6992, 0.7225)
    No Information Rate : 0.7761
    P-Value [Acc > NIR] : 1

      Kappa   : 0.3268

McNemar's Test P-Value : <2e-16

    Sensitivity : 0.6855
    Specificity : 0.7183
```

Figure 8 F-score and confusion matrix of ANN

#### iv. Random Forest

The below graph is the variable importance graph that the team computed from RF for its model development stages using internal down-sampling (not final model). The plot gives us a good idea of variable rankings. Observe that TourDate, Email, PastTrips are potentially useful candidate inputs.



For this model we had 67 total inputs out of which 7 were missing value flags. The f-score=0.54 along with specificity and sensitivity was pretty good compared to other models. Having so many inputs can lead to overfitting issues; hence, the team decided to reduce the number of variables. *(more about this below in the final model part.)*

```

F1
0.5438653
> confusionMatrix(table(RF.class,tours.rf[split.test,]$Book_12Mo),
+                    positive = "1")
Confusion Matrix and Statistics

RF.class   0    1
0  3615  473
1   936  840

Accuracy : 0.7597
95% CI : (0.7486, 0.7706)
No Information Rate : 0.7761
P-Value [Acc > NIR] : 0.9987

Kappa : 0.3857

McNemar's Test P-Value : <2e-16

Sensitivity : 0.6398
Specificity : 0.7943

```

Figure 9: F-score and other performance parameters of the first Random Forest model.

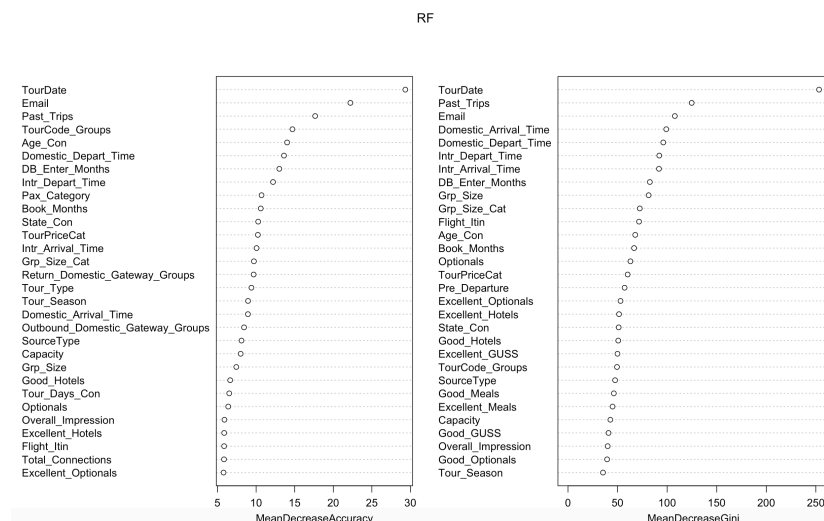


Figure 10: Initial "variable importance" from random forest model first try.

## b. Issues encountered

- The team encountered possible bottleneck model performance issues where the f-score for the models did not show great improvement.
- The team also encountered overfitting issues as in the initial stages for some models, we used about 50-60 potential input variables.

- In order to overcome these issues, the team went back to phase 1 and selected potential candidate variables and consolidated them. For example: The team included return connections along with outbound connections in the model and combined the two into a new variable called Total Connections and consolidated them into two groups using weight of evidence. The team also recategorized large factor variables into smaller levels. For example: for variable TD\_overall, levels 0,1,2 were recoded into 3 looking at percentage of target variables from the mosaic plots; this way we had three levels: 3,4,5.

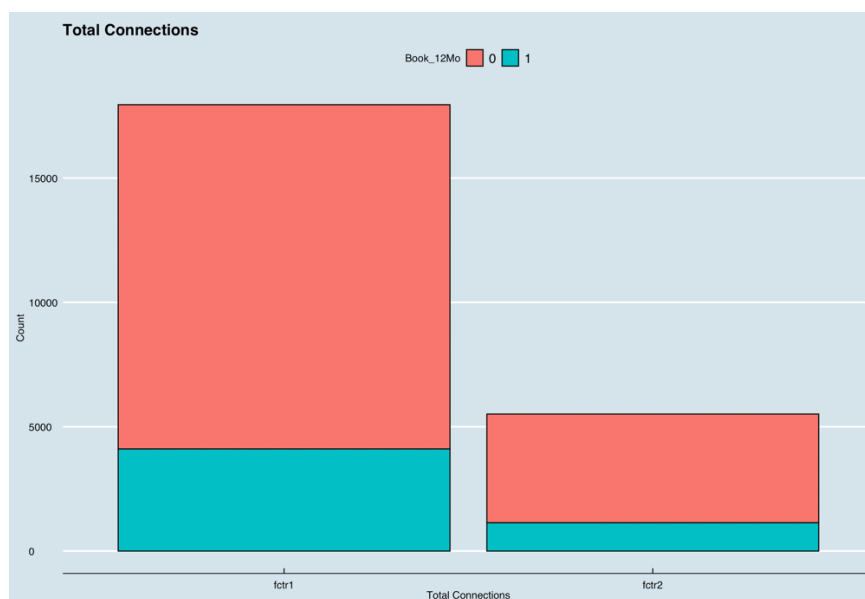


Figure: (consolidated) Total Connections: Return Connections + Outbound Connections

Some of the other variables that the team derived are as follows:

*Data Diggers created 4 new variables: “Grp\_Size\_Ratio”, “hotel\_rating”, “meal\_rating”, and “guss\_rating”*

1. “Grp\_Size\_Ratio”

$$\text{“Grp\_Size\_Ratio”} = \text{“Grp\_Size”} / \text{“Capacity”}$$

The “capacity” variable is the capacity of the tour. The smaller of the ratio the better service can be provided.

2. “hotel\_rating”

In the phase I, the team found if a customer stayed multiple hotels it is possible for a customer to give some hotels excellent rating while giving other hotels poor rating. From figure 10, one can see that poor rating has a greater effect than excellent rating. In the dataset, the formula of “Hotels\_Avg” is

$$\text{Hotel Avg} = \frac{1 * \text{number of PoorHotel} + 2 * \text{FairHotel} + 3 * \text{GoodHotel} + 4 * \text{ExcellentHotel}}{\text{Total number}}$$

This formula does not consider that poor rating has more effects. Therefore, a new variable “hotel\_rating” was created, and the formula is

$$\text{Hotel\_rating} = -4 * \text{Poor\_Hotels} + \text{Good\_Hotels} + 2 * \text{Excellent\_Hotels}$$

Here, there is more penalty for poor rating, and the coefficient for “Fair\_Hotels” is 0 since the fair hotel rating has little effect on repeat customer rate.

There are some customers that didn’t give any rating; our team treat these observations as non-response group, and it gives valuable information. Since the customers in this group didn’t give any rating, their “hotel\_rating” will be 0. Meanwhile, It is possible to get 0 score for the new formula. To prevent the 0 score mix with the nonresponse group, the score will be added by 100. Therefore, the formula will be:

$$\text{Hotel\_rating} = 100 - 4 * \text{Poor\_Hotels} + \text{Good\_Hotels} + 2 * \text{Excellent\_Hotels}$$

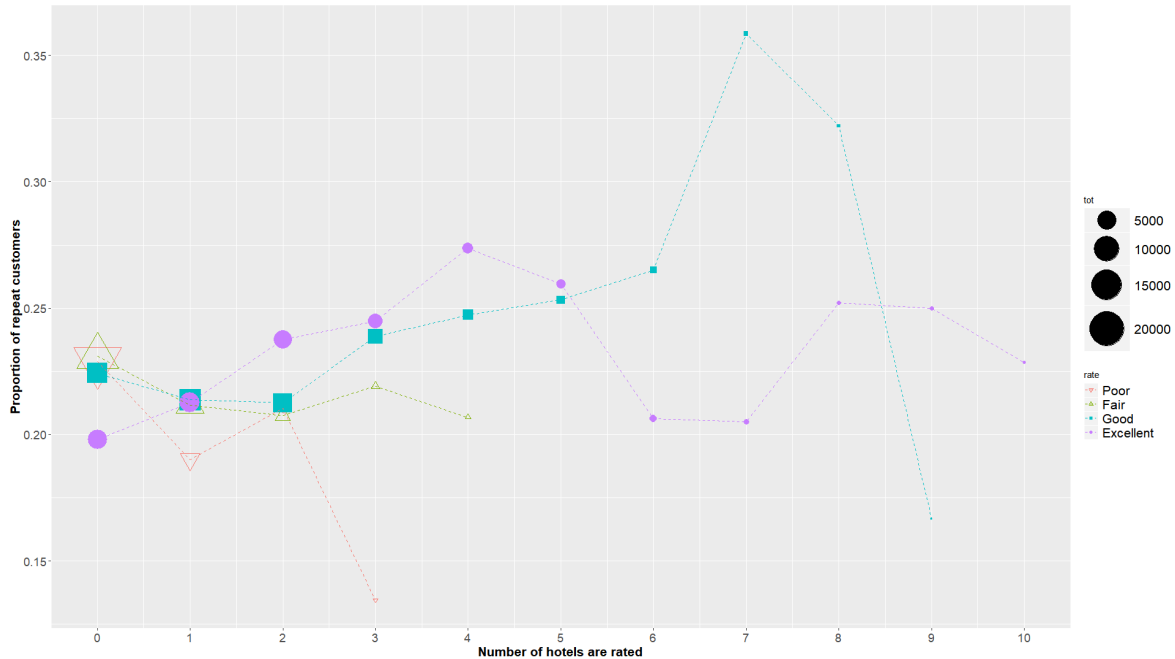


Figure 11: Effect of hotel ratings on repeat customer rate

We use the same formula, but change the coefficient, and apply them to “meal\_rating” and “guss\_rating”.

For “meal\_rating”, the formula is  $50 + (-4)*Poor\_Meals - 2*Fair\_Meals + Good\_Meals$  since it was found that both “fair\_meals” and “poor\_Meals” would reduce repeat customer rate.

For “guss\_rating”, the formula is  $50 - Poor\_GUSS - 2*Fair\_GUSS$ , since “Poor\_GUSS” and “Fair\_GUSS” would greatly hurt business while “Good\_GUSS” and “Excellent\_GUSS” have little help for the business.

### c. Attempts at Improvement

#### i. Parameter tuning

To improve the models, the team first attempted to tune the parameters of both the neural network and the random forest. This effort did not prove very fruitful and better improvement came from returning to phase I and doing more data manipulation. Nevertheless, the details of the parameter tuning are described below.

#### 1. Artificial Neural Network

First, the parameters of the neural network were tuned. In a neural network, the main parameters that can be tuned are the number of neurons in each layer, the number of hidden layers, and the activation function for each layer. The number of layers and neurons were tuned by trial and error since there is no perfect technique for optimizing the number of either. Extremes of high and low numbers of neurons were tested. Eventually, the best performing number of neurons was 256. This is surprising since generally the number of neurons is similar to the number of input variables, which was only 45 including 5 missing value flags.

The number of hidden layers was tuned over a smaller range from 2 to 5. The best number of layers turned out to be 3 hidden layers, plus the input layer and the output layer. The more layers involved, the more the model behaves like deep learning. The activation function of one of the three hidden layers was adjusted. The default activation function used was the hyperbolic tangent function. However, one of the three hidden layers had its activation function changed to the “softplus” function to smooth the results. Softplus is a very smooth function. The learning rate parameter is another potential avenue of exploration, but it was not altered during this analysis.

## 2. Random Forest

To tune the random forest, we tuned two parameters: `mtry` and `ntree`. `Ntree` determines how many trees are built and tested. This number was varied between 200 and 2000 but results did not improve significantly after 400 trees so that was the final number used to build the random forest. `Mtry` is the number of variables that are made available to the tree at every split. This parameter is set significantly lower than the number of input variables so that the algorithm does not just build the same tree 400 times. `Mtry` was varied from 2 to 10 but the best results came with `mtry=5`. This means that the model had 5 random variables to choose from at each split. None of the changes in the parameter tuning had a significant effect on the F-score of the models. The next section describes how the return to data preparation had a more significant effect.

ii. Returning to phase I

The team tried two types of consolidation for the input variables. The table below shows the potential candidate variables that were consolidated using one of the two types:

Table 2: Types of consolidation used for each variable

Variables	Weight of Evidence (WOE)	Decision Tree
Tour_Days	✓	
Grp_size_ratio_ = Grp_Size/Capacity	✓	
TourDate	✓	
Recommend_GAT	✓	
Hotels_Avg	✓	
Meals_Avg	✓	
GUSS_Avg	✓	
Optionals_Avg	✓	
Age	✓	
State	✓	
Email	✓	
Optionals	✓	
Domestic_Depart and Arrival_Time	✓	
Intr_Depart and Arrival_Time	✓	
Return and Outbound_Connect_Time_Mins_1 & 2	✓	
Total Connections	✓	
TourCode		✓
Outbound Domestic Gateway		✓
Return Domestic Gateway		✓

1. Weight-of-Evidence Consolidation:

Weight-of-Evidence was applied for consolidation. We use “Information” package to compute WOE values. Once WOE values were computed, we could plot WOE vs variable levels to find the levels have similar WOE values and put these levels into one group. The size of each new group should not less than 5% of total observations.

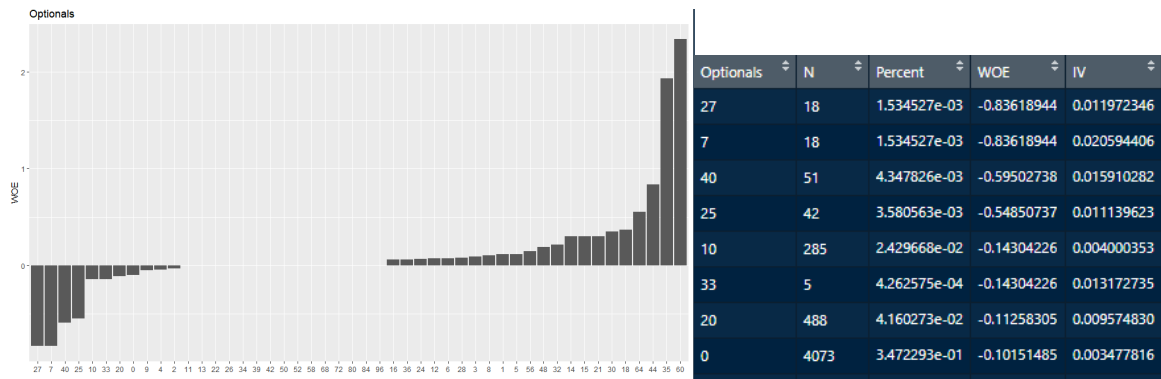


Figure 12: WOE of "Optionals"

In figure 11, the left plot is WOE of "Optionals" variable, and the table on the right side includes information of sample size, WOE value, and information value. Level "27" and "7" have similar WOE values, but there are only 36 observations in these two levels. Therefore, more levels which have similar WOE values need to be added to form a new group. Figure 4 shows the results of variable "Optionals" after consolidation. "Optionals" is consolidated into 4 levels, and the level follows repeat customer rate.

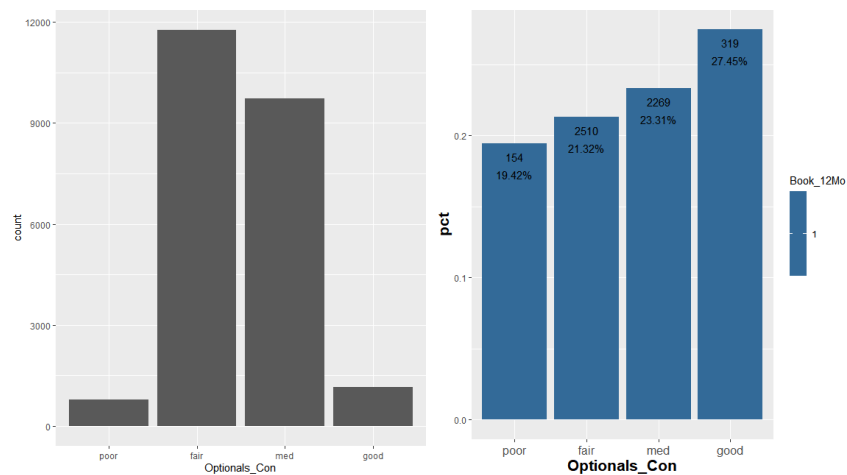


Figure 13: Histograms of "Optionals" after consolidation

## 2. Decision Tree Consolidation

The team used Decision Tree consolidation to recategorize groups. Using this consolidation, the team grouped the categories that display similar associations with the responses into one bin. This was achieved using the *tree.bins()* function.

Example: In figure 13, it can be observed that Tour Code has 127 levels and so many factor classes are tedious to visualize.

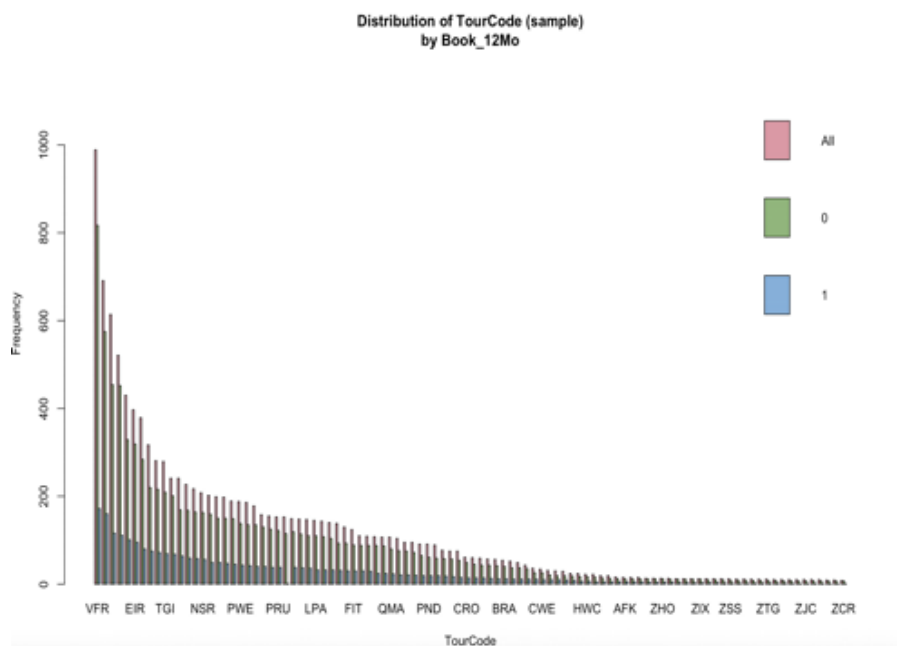


Figure 14: Tour Code histogram

Using the decision tree rules, TourCode was consolidated into 4 groups as seen below. Similarly, the team tried this consolidation for variables like Outbound and Return Domestic Gateway.



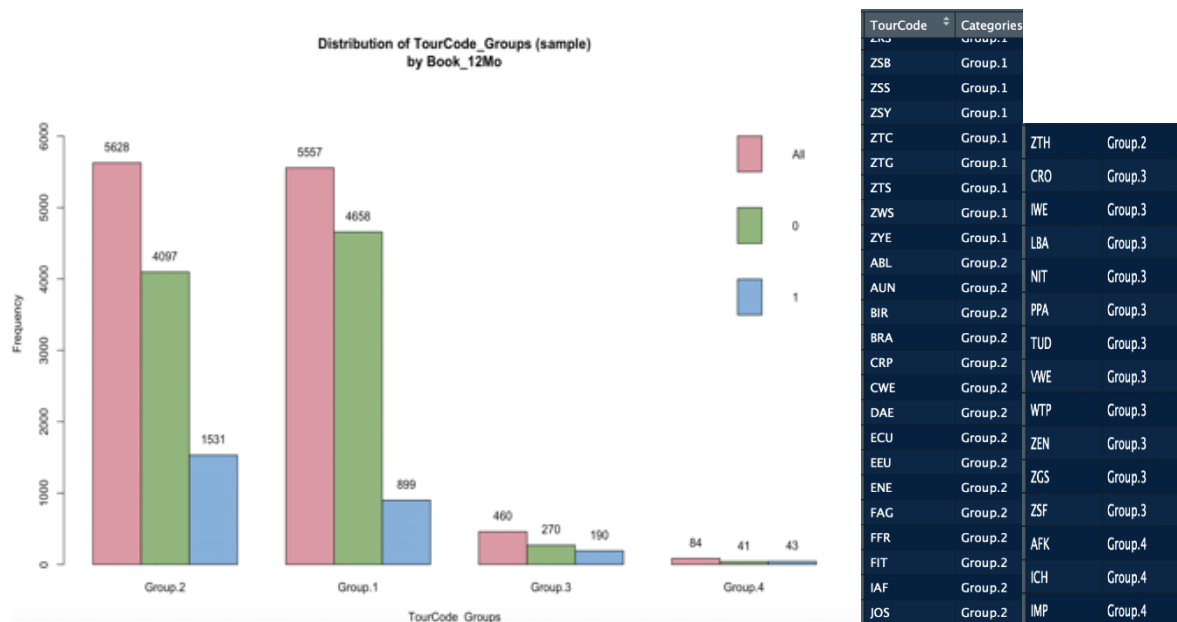


Figure 15: Consolidation of Tour Codes

### iii. Downsampling and Upsampling

Since there was class imbalance in the dataset: 0 (has not booked within 12 months) = 77.6% and 1 (has booked within 12 months) = 22.4%, the team also tried sampling techniques to balance the data.




Using down-Sample, the team randomly sampled a data set so that all classes have the same frequency as the minority class and apply it to the whole dataset as seen in figure 15. Similarly, using Up-Sample, the team randomly sampled with replacement to make the class distributions equal.



Figure 16: Effect of upsampling and downsampling on model performance

Table 3: F-scores of different models under downsampling and upsampling

Model	Sampling Technique	F-score(test)
Logistic Regression	Up-Sampling	0.5022564
Logistic Regression	Down-Sampling	0.4909947
ANN	Up-Sampling	0.5031813
ANN	Down-Sampling	0.4992932
Random Forest	Up-Sampling	0.4242121
Random Forest	Down-Sampling	0.4276543

In table 3, observe that the F-scores for Logistic Regression  and ANN  went up in external Up-Sampling as compared to external down-sampling. However, in the case of Random Forest , the F-score went down for Up-sampling compared to down-Sampling.

#### IV. Final Model

##### a. F-Score

```
> fscore
      F1
0.5499838
> confusionMatrix(Table(RF.class,tours.rf[split.test,$Book_12Mo],positive = "1")
Confusion Matrix and Statistics

RF.class   0    1
 0  3623  463
 1   928  850

      Accuracy : 0.7628
      95% CI   : (0.7517, 0.7736)
    No Information Rate : 0.7761
    P-Value [Acc > NIR] : 0.9928

      Kappa : 0.3938

  Mcnemar's Test P-Value : <2e-16

    Sensitivity : 0.6474
    Specificity : 0.7961
   Pos Pred Value : 0.4781
   Neg Pred Value : 0.8867
    Prevalence : 0.2239
    Detection Rate : 0.1450
Detection Prevalence : 0.3032
   Balanced Accuracy : 0.7217

 'Positive' Class : 1
```

Figure 17: F-score of the final RF model

The f-score is about **0.55** for this Random Forest and is better as compared to the previous RF model. Definitely consolidation and reducing variables (reduced to 45 inputs with 5 missing value flags) improved the model. Sensitivity and specificity are also better for this model compared to the previous ones.

##### b. Lift and ROC Graph

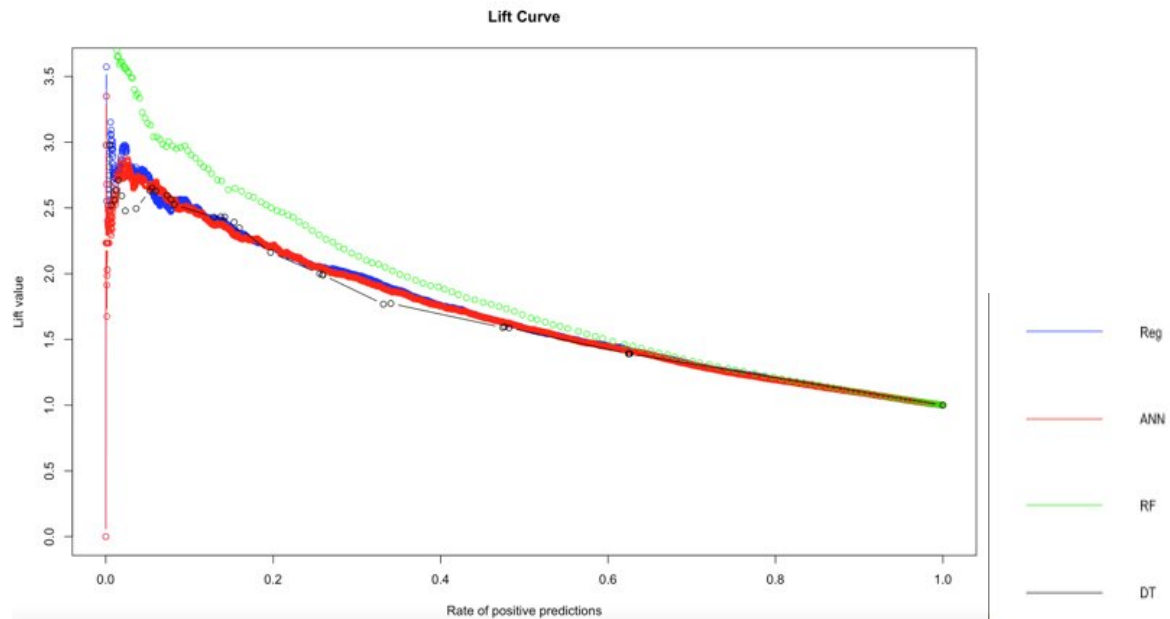


Figure 18: Lift Curve comparing Decision Tree, Regression, Neural Network, and Random Forest models

The Lift curve above shows that the Random Forest model performed the best out of the 4 models that the team tried, with a RF lift value of 2.47 for the 20th percentile. This means the proportion of primary outcome cases in the top ~20% is about 147% more likely to have booked the tour within the 12 months than a randomly selected 20% of cases.

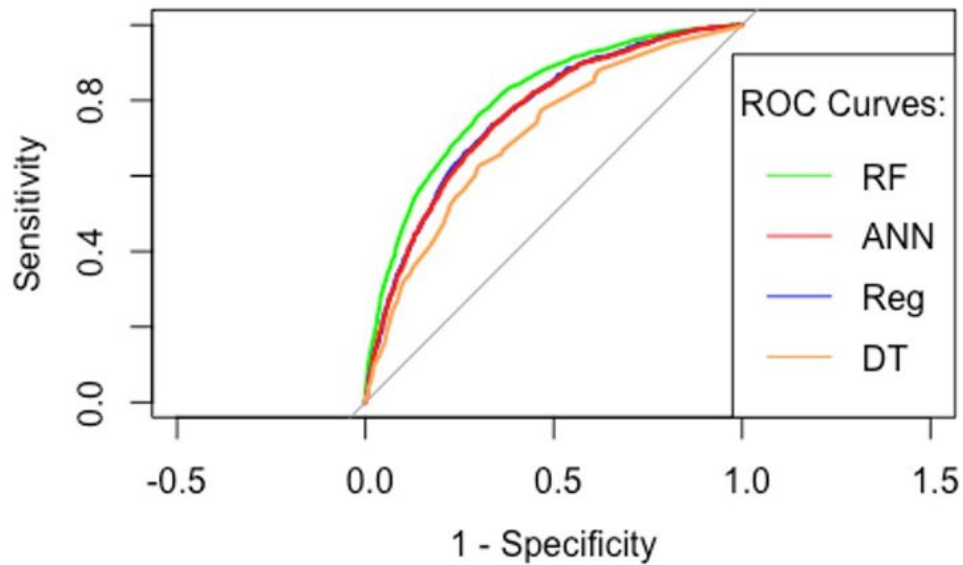


Figure 19: ROC Curve comparing Decision Tree, Regression, Neural Network, and Random Forest

The ROC curve shows that the Random Forest model performed the best. It has the highest area under the curve compared to other models.

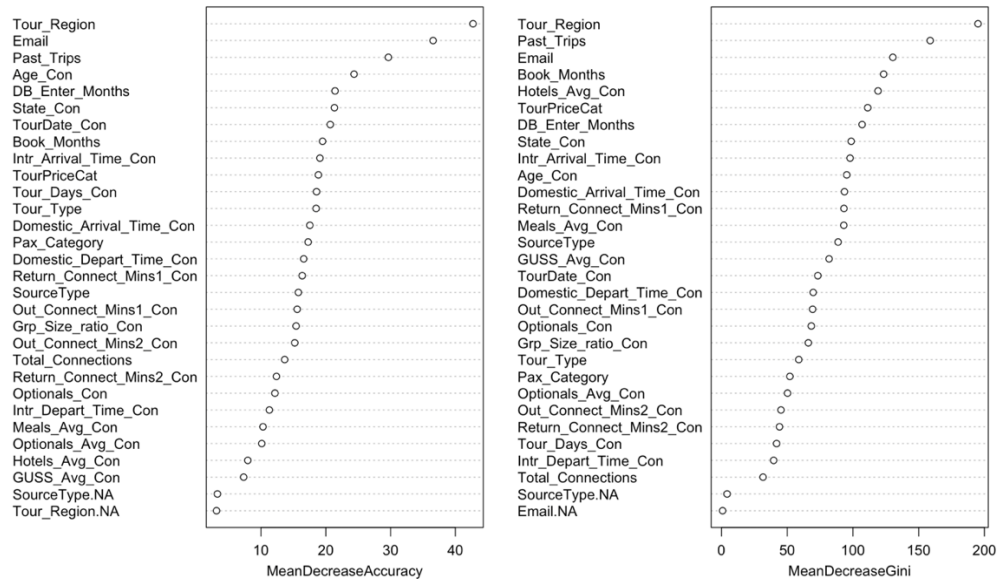


Figure 20: Plots of Mean Decrease Accuracy and Mean Decrease GINI for the final RF Model. These show which variables were most important to the model.

## V. Analysis

### a. Customer Profiling

After finalizing the model, the team is able to draw some conclusions about the types of customers that are most likely to book a follow-up trip with XYZ tours. These findings may translate into tangible actions the company can use to improve its customer retention. First, customers who are in retirement age, between (60 and 79) are not only the most numerous customers, they are also the most likely to rebook, indicating that seniors of this age travel frequently. Customers over age 80 are not as likely to rebook, however. If the goal is to maximize repeat customers, care should be taken to cater to the needs of these customers. Tours should not be particularly strenuous, for instance.

Next, customers who planned their trip further in advance (over 12 months) are the most likely to book another trip within a year, while customers who booked a few months beforehand are the least likely. This makes sense as the further in advance a customer tends to book their trips, the more likely it is that their next booking falls within a year. Consider two travelers that both take a trip every 18 months. Traveler A books trips 12 months ahead of time while Traveler B books only 3 months ahead of time. Traveler A will book their next trip within 6 months of their last one while traveler B will book 15 months after their last one. These customers could both be repeat customers of XYZ tours, but the target variable Book\_12Mo will not capture traveler B as a repeat customer. Perhaps this is cause for XYZ tours to reconsider their target variable.

The emailing campaign was very effective; of those customers who made their email available, 30% rebooked within 12 months, compared to 11% of customers who did not give their email. This is a significant difference and shows that XYZ tours should take extra care to collect email addresses.

The number of trips a customer has previously taken was an indicator for rebooking; the more trips, the more likely they were to rebook within the year. XYZ might consider giving a voucher after a customer's

first trip to increase the likelihood that they will book one more trip. Then, they will be more likely to continue to book after that.

#### b. Interaction Terms

XYZ Tours asked specifically about interaction terms. Only one interaction term was explicitly introduced into the logistic regression model. All other types of models automatically generate interactions between variables. The term that was introduced to the regression model was called group size ratio. It was the size of the tour group divided by the capacity of the tour. The theory behind this interaction was that if customers felt less crowded on the tour, they would be more comfortable and be more likely to book another tour. However, since the logistic regression model was outperformed by the random forest model, this interaction did not end up being used. Since so many of the variables are not numeric, it is difficult to introduce many meaningful interactions. Other major alterations to variables came during the consolidation step. Variables such as State had many categories consolidated as discussed in section III.

#### c. Takeaways and Conclusions

In addition to the customer profiling discussed above, some conclusions can be drawn about XYZ Tours operations and how they can be tailored to improve repeat customer rate. The Mean GINI Decrease and Mean accuracy decrease plots show that the variables discussed in the customer profiling – Age, Email, and Past Trips – are important variables to the random forest model as well. However, the top variable in both lists is TourRegion which describes where the tour was located. The mosaic plot in Figure 21 shows that tours to regions labeled CNE, FS, MD, AF, and especially CR were more successful at generating repeat customers. Since Tour\_Region was so important to the model, it would make sense that Tour\_Code would also be important. However, since these variables are so linked, only one was used to avoid collinearity. Tour\_Code was eliminated from the variable pool.

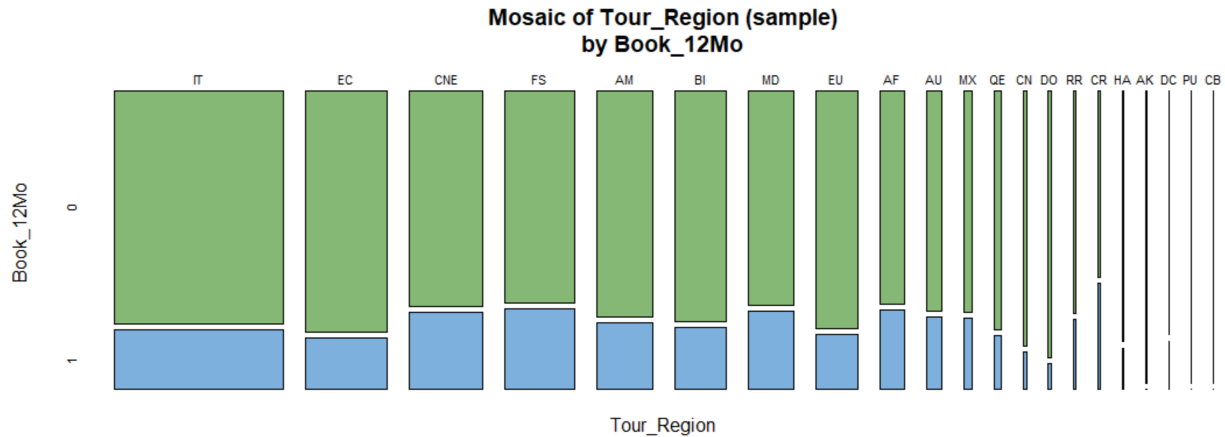


Figure 21: Mosaic plot of Tour\_Region versus Book\_12Mo

One theory may be that the tour directors for those regions are particularly successful in getting repeat customers. If this were the case, one would expect to see TD\_Overall as a more important variable. TD\_Overall does not appear to be important to the model, but that could also be due to the fact that most customers seemed to rate their director highly as shown in Figure 22. The mosaic plot shows that a higher Tour Director rating did not lead to a higher incidence of repeat customers. Nevertheless, the tour directors or organizers in regions CNE, FS, AF, MD, and CR may have some insights into what makes their tours so successful. It would be worthwhile for XYZ tours to consult them for ideas.

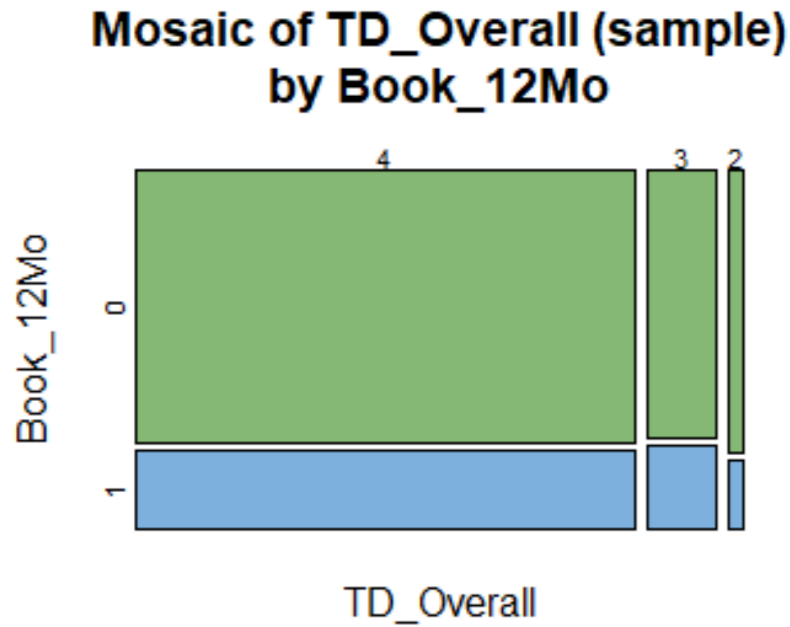


Figure 22: Mosaic of Tour Director ratings against the target variable.

In summary, Data Diggers recommend that XYZ Tours, if their goal is to maximize repeat customers, should focus on catering to retirees, should take care to ensure that customer emails are given, and should consider giving vouchers to customers after their first trip to encourage them to take one more. Also, a different target variable could be considered. The current target does not capture customers who are repeat customers but have more than 1 year between their first trip and when they book their second. Perhaps a better target would be Book\_24Mo or Book\_36Mo.