**A close up of a logo

Description automatically generated**

**Final Report**

**US Airline Passenger Satisfaction Survey**

Yi Yang

Northeastern University

ALY 6040 - 90540

Professor Kasun Samarasinghe

08/11/2020

**INTRODUCTION**

We all have witnessed the growth and change of the US aviation industry in the past 20 years. More and more people rely on the convenience of taking flights to their destinations. While the needs are increasing, some challenges come with them. To seek sustainable growth, US Airlines should take the satisfaction of its passengers seriously and always listen to their feedback to improve their services.

Our business problem is the percentage of satisfaction level of satisfied customers is not very high by observing our dataset.

Therefore, our business goal is to seek ways to increase the level of satisfaction and help the US airline business gain some insights into improving their services.

In this report, we will explore what affects passengers’ satisfaction level the most and try to provide some insights via predictive models. Finally, we will make comparisons between these models and select the best-fitting model for the dataset. We are hoping to offer some recommendations on how to improve US Airline customer service.

**ANALYSIS**

DATA PREPROCESSING

The dataset contains a total of 12,9880 observations with 24 variables. Among those 24 attributes, 5 of them are category and others are integers. The key factor that we are focusing on is the variable: satisfaction\_v2, which represents airline passengers’ satisfaction level and we will explore how each attribute relates to it.

|  |  |  |
| --- | --- | --- |
| Attribute | Attributes Description | Attribute Value Level |
| id | id of the passengers | Range between 0-129880 |
| Satisfaction\_v2 (Dependent Variable) | Satisfaction level | Satisfied, neutral or dissatisfied |
| Gender | Gender of the passengers | Female, Male |
| Customer Type | The customer type | Loyal customer, disloyal customer |
| Age | Age of the passengers | Range between 7 - 85 years old |
| Type of Travel | Purpose of the flight of the passengers | Personal Travel, Business Travel |
| Class | Travel class in the plane of the passengers | Business, Eco, Eco plus |
| Flight Distance | The flight distance of this journey | Range between 50-6951 |
| Seat Comfort | Satisfaction level of Comfortability of the seat in airplane | Rating:0 (least) - 5 (highest) |
| Departure/Arrival time Convenient | Satisfaction level of Convenience of departure or arrival time | Rating:0 (least) - 5 (highest) |
| Food and Drink | Satisfaction level of Food and drink provided on the airplane | Rating:0 (least) - 5 (highest) |
| Gate Location | Satisfaction level of Gate location | Rating:0 (least) - 5 (highest) |
| Inflight WIFI Service | Satisfaction level of Inflight WIFI service | Rating:0 (least) - 5 (highest) |
| Inflight Entertainment | Satisfaction level of Inflight entertainment | Rating:0 (least) - 5 (highest) |
| Online Support | Satisfaction level of Online support | Rating:0 (least) - 5 (highest) |
| Ease of Online Booking | Satisfaction level of Ease of online booking | Rating:0 (least) - 5 (highest) |
| On-board Service | Satisfaction level of On-board service | Rating:0 (least) - 5 (highest) |
| Leg Room Service | Satisfaction level of Leg room service | Rating:0 (least) - 5 (highest) |
| Baggage Handling | Satisfaction level with baggage service | Rating:0 (least) - 5 (highest) |
| Check-in Service | Satisfaction level check-in Service | Rating:0 (least) - 5 (highest) |
| Cleanliness | Satisfaction level of Cleanliness | Rating:0 (least) - 5 (highest) |
| Online Boarding | Satisfaction level with online boarding service | Rating:0 (least) - 5 (highest) |
| Departure Delay in minutes | Departure Delay time in minutes | Range between 0-1592 |
| Arrival Delay in Minutes | Arrival Delay time in minutes | Range between 0-1584 |

*Table 1*. Data Dictionary

To begin with, we first check for class bias in the dataset. From the table below, we can see the proportion of satisfied population and dissatisfied population are approximately the same. Thus, the dataset is not biased. A screenshot of a cell phone

Description automatically generated

*Table 2.* Cross Table of Satisfied /Dissatisfied

Then we checked the completeness of the dataset and whether there are missing values. From the output below, we observed 393 NA values in a column: Arrival Delay in Minutes. To determine the technique of handling missing values, we first calculated the proportion of the NA values by dividing 393 by 129880 which equals to 0.3% that falls between the range of 0 - 0.5% so that we can conclude that this percentage of NAs does not affect our analysis and can be removed.

A screenshot of a cell phone

Description automatically generated

Also, we drop the id column since it only served as a counting symbol and was not quite related to the satisfaction level in our analysis. Therefore, our dataset is left with a total of 129487 records and 23 attributes.

EXPLORATORY DATA ANALYSIS

First of all, we created a plot to see the distribution of current customers’ satisfaction levels. From Figure 1, there is 54.74% of the population are satisfied with the U.S Airline service and there are 45.26% of the customers are neutral or dissatisfied. As a result, the distribution of satisfied customers and dissatisfied customers are roughly the same.

A screenshot of a cell phone

Description automatically generated

*Figure 1.* Distribution of Satisfied /Dissatisfied Population

To improve the customer’s satisfaction level, we created a correlation matrix between satisfaction\_v2 and other numeric variables to find out which attributes impact satisfaction the most.

A close up of a map

Description automatically generated

*Figure 2.* Correlation Matrix

From the figure above, we can see that inflight entertainment, ease of Online booking, and online support are the top 3 attributes that are highly related to the satisfaction level. To take a closer look, we created three boxplots of these three variables.

A screenshot of a social media post

Description automatically generated**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated** **A screenshot of a cell phone

Description automatically generated**

*Figure 3.* Summary of Highly Correlated Attributes

Based on the boxplots and the summary of these three variables, the mean score is around 3.3-3.5. Hence, it is necessary to improve inflight entertainment, ease of Online booking, and online support service to increase customer satisfaction levels.

Furthermore, we analyzed the demographic characters of the surveyed customers to find target customers. Comparing with male passengers, there are 21.1% more of the female population are satisfied with the U.S Airline services. A screenshot of a cell phone

Description automatically generated

*Figure 4.* Gender vs Satisfaction

In a comparison of travel classes, the business class has the highest satisfaction level with 70.9% satisfied customer and Eco and Eco Plus has approximately consistent satisfaction level of 39.4% and 42.7% satisfaction respectively.

A screenshot of a cell phone

Description automatically generated

*Figure 5.* Travel class vs Satisfaction

By comparing the satisfaction level between loyal customers and disloyal customers, there are 14 % more satisfied customers in loyal customers.

A screenshot of a cell phone

Description automatically generated

*Figure 6.* Customer type vs Satisfaction

Lastly, we can see business travel customers have higher satisfaction than personal travel customers for figure 7.

A screenshot of a cell phone

Description automatically generated

*Figure 7.* Type of travel vs Satisfaction

In summary, we can conclude that female, business class, business travel, and loyal customers tend to be more satisfied with US Airline service.

DATA MODELING

*LINEAR REGRESSION ANALYSIS*

After getting our dataset being prepared, we then divide the original dataset into a training dataset and test dataset, where 70% of the data goes into the training dataset and 30% of the data goes into the test dataset. To perform a Linear regression analysis, we first convert the factorial data including “Gender”, “Customer Type”, “Type of Travel”, and “Class” into numeric datatype, except for our dependent variable, “satisfaction\_v2”. As for our dependent variable, we handle it in different ways in linear regression and logistic regression analysis. For linear regression analysis, the dependent variable has to be continuous. In our case, the attribute, “satisfaction\_v2”, that we are interested in initially, is not qualified for the analysis. As a result, we use the mutate () function to create a new attribute named score, which is the sum of attributes that are shown in satisfaction level (scale 0-5) and made it as outcome variable Y for linear regression. Besides, before performing the linear regression analysis, we checked the correlation between all 10 variables. Based on the graph below, we can see a strong correlation between “Departure Delay in Minutes” and “Arrival Delay in Minutes”. To avoid overfitting, we will drop one of them after the final model was created.

A screenshot of a cell phone

Description automatically generated

*Figure 8.* Correlation Chart

We first created a null model that only contains “score” and a full model that contains “score” and other 10 variables with our training data. Then we applied the stepwise selection with “both” directions to get the final model. The summary of the final model showed it contains 8 variables, and the R2 is equal to 0.2701 and the adjusted R2 is equal to 0.27, which means that only 27 % of the data is described by our linear model.

A close up of a newspaper

Description automatically generated

*Figure 9.* Summary of Linear Regression Model

Using the predict () function, we predicted the score on the test dataset and created a plot between the observed value and the predicted value.

A screenshot of a cell phone

Description automatically generated

*Figure 10.* Plot Between Actual Value and Predicted Value

From the plot here, it matches with our conclusion that our variables are not highly linearly correlated. We also created a histogram of residuals, and the histogram suggests that the residuals (the error terms) are normally distributed, so our data follows linear regression assumptions.

A picture containing screenshot

Description automatically generated

*Figure 11.* Histogram of Residuals

Finally, we calculated the R2 and RMSE to evaluate our linear regression model. The output shows that the R2 is equal to 0.2748838 and RMSE is equal to 8.043188, which indicates that the model is not efficient.

*LOGISTIC REGRESSION ANALYSIS*

Logistic regression is another example of classification techniques that is used to predict when the outcome variable Y is binary categorical. As a result, with the ifelse() function, we transformed the variable, “satisfaction\_v2”, as a binary variable, where “satisfied” are marked as “1” and “neural or dissatisfied” are marked as “0”.

Then we used glm() function in R to build a logistic model to predict “satisfaction\_v2” from the other 22 independent attributes.

A close up of text on a white background

Description automatically generated

*Figure 12.* Summary of the Logistic Regression Model

Based on the summary statistic of the logistic model that we built based on training data, we can tell from each attribute’s coefficients and the p-values that they are all significantly important in predicting “satisfaction\_v2” since all p-values are quite small.

Then, we predicted the test dataset using the model we just built. By using optimalCutoff() function, we achieved our probability cutoff is around 0.577. With the optimal cutoff, we also created a confusion matrix and calculate the accuracy of the logistic model.

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

Based on the result above, we observed the accuracy of 83.7% and the misclassification rate as 16.3%. Moreover, the proportion of actual positive cases that are correctly identified is 87.6% and the proportion of actual negative cases that are correctly identified is 79.6%.

Lastly, we plotted the Receiver Operating Characteristics Curve (ROC) to trace the percentage of true positives accurately predicted by the given logit model as the prediction probability cutoff is lowered from 1 to 0. Since the ROC value is equal to 0.7823, we can conclude the logistic model is moderately efficient.

A screenshot of a cell phone

Description automatically generated

*Figure 13.* ROC Curve

*RIDGE REGRESSION ANALYSIS*

To improve our models, we applied regularization with the ridge regression model. With the same step as the linear regression model, we first split the data into 70% training data and 20% test data. Then we transformed x from a variable within a data frame into a matrix and transformed y from a variable within a data frame into a vector. With the default setting of alpha, nlambda, and lambda.min.ratio, we created a ridge model and generated a plot between Log(λ) and mean squared error. From the plot, we found out the best lambda is 0.4771427.

*A screenshot of a computer

Description automatically generated*

*Figure 14.* Lambda Plot

Using our ridge regression model with the best-fitting parameter and lambda estimates from training data, we predicted the score in the test dataset. To evaluate the ridge regression model, we calculated the R2 and RMSE, where R2 equal to 0.97 and RMSE equal to 0.03.

*DECISION TREE MODEL*

With our dataset, we tried to build a classification tree to model to predict values of y, satisfication\_v2. Firstly, we split the data into 70% training data and 30% test data and named them as “train” and “test”. Then we grow the tree with the rpart() function and “class” method for a classification tree based on the training dataset and plot it with the repart.plot() function.

A picture containing clock

Description automatically generated

*Figure 15*. Tree Plot

To validate the current model, we applied the printcp() and plotcp() functions. From the summary below, we can see only 4 variables are used to build the tree and they are “Ease of Online booking”, “Inflight entertainment”, “Online support”, and “Seat comfort”. Moreover, from the list of cp values of the report, we can select the one having the least cross-validated error and use it to prune the tree.A screenshot of text

Description automatically generated

*Figure 16*. Summary Report for Decision Tree Model

The plotcp function provides a graphical representation of the cross-validated error summary as below.

A close up of a map

Description automatically generated

*Figure 17.* Cross-validation results of Tree Model

Then we prune the tree to create an optimal decision tree by setting the optimal cp value associated with the minimum error, and with the plotcp(pruned\_tree), we get the same graph as tree\_model, which indicates that our initial decision model is the optimal decision tree for our dataset.

After the validation check, we made a prediction with both training dataset and test dataset based on the tree model. We then created confusion tables based on training prediction and test prediction and calculate the accuracy for both. The result shows that the accuracy for the training dataset is 86.40% and 86.49% for the test dataset. Hence, we concluded our model is not overfitting.

*RANDOM FOREST MODEL*

Having a similar concept with a decision tree, Random Forest works by creating multiple decision trees and then combining the output generated by each of the decision trees to predict an outcome variable. It helps reduce the correlation between the decision trees and removes the bias that decision trees might introduce in the system. Below is the Random Forest model that we built based on our dependent variable “satisfaction\_v2” and other variables. A picture containing bird

Description automatically generated

*Figure 18.* Random Forest Model with default parameters

By default, the number of trees is 500, and the number of variables tried at each split is 4 in this case, and the Out of bag (OOB) error rate is 4.28%. Then we predicted the training and test data.

A screenshot of a cell phone

Description automatically generated

*Figure 19.* Prediction on test data

We can tell that the training dataset has an accuracy of 99.99% while the test dataset has an accuracy of 95.8%. And then we plotted the error rate in terms of the number of trees.

A picture containing screenshot

Description automatically generated

*Figure 20.* Number of trees vs Error rate

We can tell from Figure 20 above that the error rate tends to be stabilized as the number of trees gets after 300. To optimize our accuracy, we used a tuning algorithm to find the best mtry in terms of optimizing the highest accuracy.

A screenshot of a cell phone

Description automatically generated

*Figure 21.* Optimal mtry vs Accuracy

We observe from Figure 21 that this model can have the highest accuracy of 96% as the number of variables per tree(mtry) is 9 or 10, but the accuracy seems not improving too much compared to what we achieved by default.

After finding the most optimal parameters, we built an improved random forest model with improved accuracy of 4.09% shown below.A screenshot of a cell phone

Description automatically generated

*Figure 22.* Improved Random Forest model

A screenshot of a cell phone

Description automatically generated

*Figure 23.* Prediction on test data with the optimal model

We also used importance () and varImpPlot() functions to check and plot the importance of independent variables.

A screenshot of text

Description automatically generated

*Figure 24.* Importance of variables

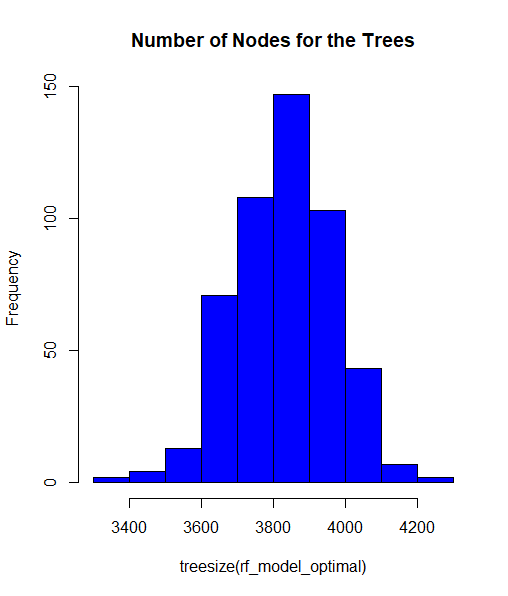
A screenshot of a social media post

Description automatically generated

*Figure 25.* Random Forest Variable Importance Plot

Figure 24 and 25 show the importance of each variable in the model by the accuracy and the Gini Index. The two graphs show that “Checking.service” and “Inflight.entertainment” are most relevant to our model’s accuracy, followed by “Seat.Comfort”.

Besides, we can plot the distribution of the number of nodes for the trees generated by our improved random forest model.



*Figure 26.* Distribution of the number of nodes for the trees

We can tell from Figure 26 that tree size 3900 has the greatest number of nodes around 150.

*GRADIENT BOOSTING MACHINE (GBM) MODEL*

To perform a GBM model, we did the same step of splitting the data into 70% training data and 30% test data. With the gbm() function, we set the formula as y=satisfication\_v2 and the rest variables as x, distribution as “gaussian”( squared error), data as the train data, shrinkage as 0.001, cv.folds as 5, and the rest arguments as default. Since we have a large sample, we first set the number of trees as 10000. After about an hour, the output shows the best cross-validation iteration was 10,000 with 10,000 iterations, which means we need to increase the number of trees. With the first gbm model, we calculate the minimum CV RMSE is equal to 0.3167633, which indicates on average our model is about 0.3167633off from the actual satisfaction level. We also plot loss function as a result of 10,000 trees added to the ensemble.

A picture containing screenshot

Description automatically generated

*Figure 27*. GBM Model Plot

From the plot above, the green line plots the validation error and the blue dashed line shows the optimum iteration. Since GBMs can overfit so our goal is to find the optimal number of trees that minimize the loss function of interest with cross-validation. Thus, we create a second gbm model with 15,000 trees, and the output shows the best cross-validation iteration was 14,986 within 15,000 iterations. A screenshot of a cell phone

Description automatically generated

The process took around five hours to run. The same step with the first gbm model, we calculated the CV RMSE, and the error has decreased from 0.3167633to 0.2019807.

A screenshot of a cell phone

Description automatically generated

*Figure 28.* Second GBM Model Plot

From the plot here, we can see both the training error and the validation error decreases as the number of iterations increased around 15,000. Comparing with manually tweaking hyperparameters one at a time, a better option is to perform a grid search that iterates over every combination of hyperparameter values and allows us to assess which combination tends to perform well. We first create a hyperparameter grid and construct our grid of hyperparameter combinations. We’re going to search across 81 combinations with varying learning rates and tree depth.

To perform grid search, we first we randomized the data use the train.fraction to take a random percent from the train data. We set the number of trees equal to 10,000, shrinkage as 0.01,0.05, and 0.1, interaction depth as 3,5, and 7, number of minobsinnode as 5, 7, and 10, and bag fraction as 0.65, 0.8 and 1, and the rest parameters as default. From the first search, we list the top 10 models to evaluate their performance.A screenshot of a cell phone

Description automatically generated*Figure 29*. Result Display

By looking up the top 10 models result here, we can refine our search and adjust grid to produce better results in grid research. With the same for loop as before, we performed another grid search with 15,000 trees. From the following result, we get similar result but with lower RMSE.

A screenshot of a cell phone

Description automatically generated

*Figure 30*. Result Display

Based on the finding from the top 10 models, we apply those specific parameters in our training dataset as our final model. For the final GBM model, we set the number of trees as 14,986, interaction.depth as 5, shrinkage as 0.1, n.minobsinnode as 5, bag.fraction as 0.65, train.fraction as 0.65, and the rest as default.

A close up of text on a white background

Description automatically generated

After re-running our final model, we created visuals to show the variables that have the largest impact on satisfaction level for the second GBM model and the final GBM model.

A screenshot of a cell phone

Description automatically generated

*Figure 31*. Variable Importance for GBM2 Model

A screenshot of a cell phone

Description automatically generated

*Figure 32*. Variable Importance for Final GBM Model

Lastly, we made a prediction in both the GBM2 model and the final GBM model and compared their prediction accuracy. By using the equation, we found out that the RMSE of the GBM2 model is equal to 0.1991802, and the RMSE of the final GBM model is equal to 0.1884692. Hence, the final GBM model has higher predictive accuracy than the GBM2 model.

**CONCLUSION**

Comparing with the Linear Regression Model and GBM Model, Ridge Regression Model has higher R2 and lower RMSE. In a comparison of predictive accuracy, Random Forest Model has the highest accuracy with an accuracy rate of up to 96% and fewer misclassification errors.

In summary, based on the exploratory analysis and predictive models, we can conclude that inflight entertainment, online service, and seat comfort level are attributes that significantly impact on customer satisfaction. Thus, US Airlines must ensure a higher quality of services in those services to reach a higher level of customer satisfaction.

**REFERENCE**

D, J. (2018, June 10). Passenger Satisfaction. Retrieved July 12, 2020, from https://www.kaggle.com/johndddddd/customer-satisfaction?select=satisfaction.xlsx