AGH UNIVERITY OF SCIENCE AND TECHNOLOGY
FACULTY OF APPLIED MATHEMATICS

The Selection of a Model for Airlines Customer Satisfaction

Joanna Krężel, Anna Matysek, Piotr Mikler, Adam Szczerba

24 styczeń, 2022

Abstract

The project aims to analyze the data about satisfaction of Investico Airlines passengers. Having customer-granular observations about the cruise and a reported satisfaction level for particular aspects of the flight we try to fit models which predict whether they are satisfied with the service or not. The binary classification models used during the project are Adaline, Logistic regression, Random Forest and a Neural Network; out of which our recommendation is Logistic Regression based on explainability and high performance metrics. This document describes the process we undertook and presents the results of data preprocessing, model selection and model validation.

Contents

1 Introduction

During the project we undertook a problem of an Airline Company, which surveyed it's passengers on different aspects of the journey and would like to make inference about parts of their service which impact the overall client satisfaction.

The document presents our approach to answering that question. The first chapter describes the origin and treatment of the data. In the second chapter we talk about different binary classification models, fit and validate them. In the last part of the document their performance is compared on an unseen test set and we conclude with a model recommendation and answer the posed question based on the findings of the project.

2 Data

The data is downloaded from www.kaggle.com and delivered by an anonimized airline organization, called *Invistico Airlines* for the purpose of the project. The dataset consists of customer-granular information about them, their flight and survey votes given for specific parts of the service. The variable of interest is the binary *satisfaction* answer, which denotes whether a customer was overall satisfied or dissatisfied with their flight. To help us do that, we have 129,880 instances of 22 explanatory variables, which we describe below.

Feature	Description	Variable type
Satisfaction	Overall satisfaction	factor, 2 levels
Gender	Gender of the passenger	factor, 2 levels
Customer type	Loyalty of the passenger	factor, 2 levels
Age	The age of a passenger	continuous
Type of travel	Flight purpose	factor, 2 levels
Class	Travel class in the plane	factor, 3 levels
Flight distance	Distance of the journey	continuous
Seat comfort	Survey note for seat comfort	factor, 5 levels
Departure/arrival	Survey note for departure/arrival time convenience	factor, 5 levels
Food and drink	Survey note for food and drinks	factor, 5 levels
Gate location	Survey note for gate location	factor, 5 levels

Feature	Description	Variable type
Inflight WiFi service	Survey note for the inflight wifi	factor, 5 levels
Inflight entertainment	Survey note for inflight entertainment	factor, 5 levels
Online support	Survey note for online support	factor, 5 levels
Ease of online booking	Survey note for online booking	factor, 5 levels
On-board services	Survey note for on-board service	factor, 5 levels
Leg room	Survey note for leg room/space	factor, 5 levels
Baggage handling	Survey note for baggage handling	factor, 5 levels
Checkin service	Survey note for check-in service	factor, 5 levels
Cleanliness	Survey note for cleanliness	factor, 5 levels
Online boarding	Survey note for online boarding	factor, 5 levels
Departure delay	Delay upon departure	continuous
Arrival delay	Delay upon arrival	continuous

2.1 Exploratory Data Analysis

Before data wrangling and modelling, it is good to get an overview of our data set and verify whether the variables meet a commmon sense. We start by summarizing the dataframe to see a handful of basic statistics. For each quantitative variable we get the min, max, mean, median and the IQR. For each categorical variable on the other hand we see the number of observations per category.

```
##        satisfaction      Gender                Customer Type         Age
##  satisfied   :71087   Female:65899   Loyal Customer   :106100   Min.   : 7.00
##  dissatisfied:58793   Male  :63981   disloyal Customer: 23780   1st Qu.:27.00
##                                                                 Median :40.00
##                                                                 Mean   :39.43
##                                                                 3rd Qu.:51.00
##                                                                 Max.   :85.00
##
##          Type of Travel       Class       Flight Distance Seat comfort
##  Personal Travel:40187   Eco     :58309   Min.   :  50    0: 4797
##  Business travel:89693   Business:62160   1st Qu.:1359    1:20949
##                          Eco Plus: 9411   Median :1925    4:28398
```

```
##                                          Mean   :1981    5:17827
##                                          3rd Qu.:2544    2:28726
##                                          Max.   :6951    3:29183
##
##  Departure/Arrival time convenient Food and drink Gate location
##  0: 6664                           0: 5945        2:24518
##  1:20828                           1:21076        3:33546
##  2:22794                           2:27146        4:30088
##  3:23184                           3:28150        1:22565
##  4:29593                           4:27216        5:19161
##  5:26817                           5:20347        0:    2
##
##  Inflight wifi service Inflight entertainment Online support
##  2:27045               4:41879                2:17260
##  0:  132               2:19183                3:21609
##  3:27602               0: 2978                4:41510
##  4:31560               3:24200                5:35563
##  5:28830               5:29831                1:13937
##  1:14711               1:11809                0:    1
##
##  Ease of Online booking On-board service Leg room service Baggage handling
##  3:22418                3:27037          0:  444          3:24485
##  2:19951                4:40675          4:39698          4:48240
##  1:13436                1:13265          3:22467          1: 7975
##  5:34137                2:17174          2:21745          2:13432
##  4:39920                5:31724          5:34385          5:35748
##  0:   18                0:    5          1:11141
##
##  Checkin service Cleanliness Online boarding Departure Delay in Minutes
##  5:27005         3:23984     2:18573         Min.   :   0.00
##  2:15486         4:48795     3:30780         1st Qu.:   0.00
##  4:36481         1: 7768     5:29973         Median :   0.00
##  3:35538         2:13412     4:35181         Mean   :  14.71
##  1:15369         5:35916     1:15359         3rd Qu.:  12.00
##  0:    1         0:    5     0:   14         Max.   :1592.00
##
##  Arrival Delay in Minutes
##  Min.   :   0.00
```

```
##  1st Qu.:   0.00
##  Median :   0.00
##  Mean   :  15.09
##  3rd Qu.:  13.00
##  Max.   :1584.00
##  NA's   :393
```

It's worth noting at this stage that for survey answers describing customer satisfaction level, 0 means *Not Available* and reflects a situation in which the passenger did not provide an answer.

Looking at the data summary we don't see any immediate problems, and overall the data statistics seem to be reasonable. It is noteworthy that the numbers of satisfied and dissatisfied customers in the dataset are comparable, which means that we're not working with an imbalanced binary classification problem.

Moreover, we have almost the same number of males and females, and of business and eco class passengers. However, we have significantly more opinions from loyal customers than from disloyal ones. We note that loyal customers might give higher survey answers because of e.g. benefits due to loyalty programs. What we also notice is the fact that there are missing values in some columns, which we will take care about in the following chapters.

Additionally, we visualize histograms of features to view the distribution of continuous variables.

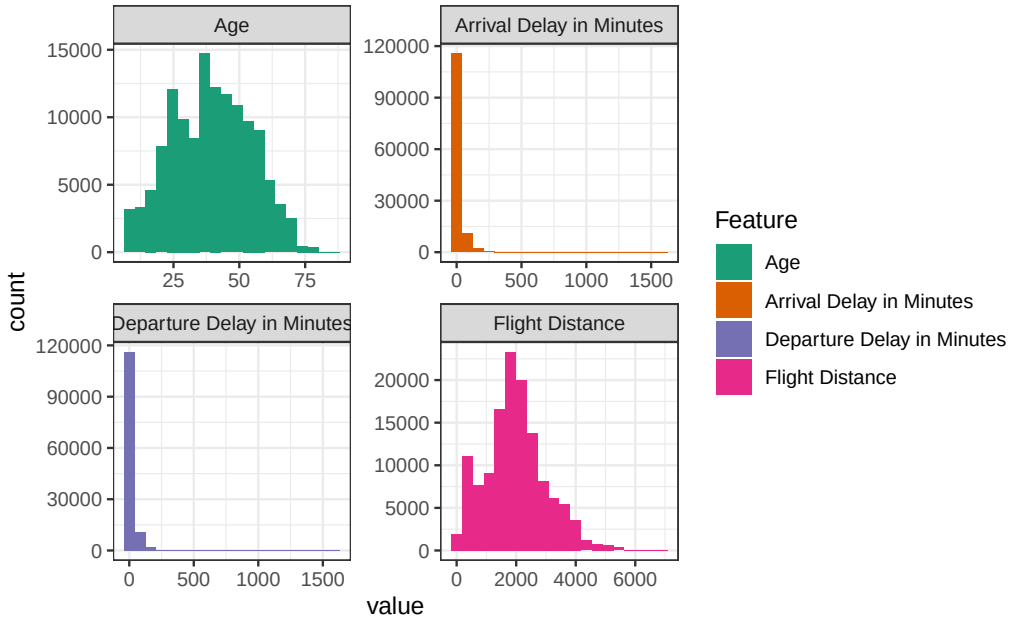

Figure 1: Distributions of continuous variables.

We see that:

- departure and arrivals delays are mostly zero, or small,
- the most common flight distance is around 2,000 kilometers,
- the age varies between very small children to elderly in their 80s, but the biggest number of customers are between their 20s and 60s.

Now, as when we have an initial overview of our data set, we move to data preprocessing.

2.2 Data Preprocessing

As both academia and business point out, the data-related operations typically constitute about 90% of the whole effort in a modeling pipeline. The performance of any model is heavily driven by the quality of it's inputs. It can be proven in a simple trial by combat that even a suboptimal model running on high quality data can oftentimes bring a sophisticated one with poor inputs to it's knees. For that reason it is of utmost importance to pay

extra care and attention to the data which is fed to the decision making models.

During data preprocessing step we will raise questions about the information the data conveys. First, we'll deal with NAs and outliers. Then we will refactor the features using variable binning and select a subset of features displaying predictive power. In the end we will encode the variables and prepare a train-validation split for model development.

2.2.1 Missing data treatment

Table 2: NA breakdown per feature. NAs span a small portion of data

Feature	NA_count	pct_of_data
ScheduleNote	6664	5.13%
FoodNote	5945	4.58%
SeatNote	4797	3.69%
EntertainmentNote	2978	2.29%
LegRoomNote	444	0.34%
ArrivalDelay	393	0.3%
WifiNote	132	0.1%
eBookingNote	18	0.01%
eBoardingNote	14	0.01%
CleanNote	5	0%
ServiceNote	5	0%
GateNote	2	0%
CheckInNote	1	0%
eSupportNote	1	0%

After examining the data missingness it seems we don't have any critical issue related to missing values. *NAs* are present in 12 variables, but they constitute a minuscule portion of a very large dataset (see table 2). We considered employing an imputation strategy based on median, but given that roughly 0.08 of all rows contain an *NA*, and even if we drop them we would still have 119255 observations left to work with. Based on that we decided not to introduce imputed values to the dataset, but rather work with pure data.

2.2.2 Feature Engineering

The more is not always the better. Feature engineering is a pre-modeling stage which serves identifying features which are significant and filtering out the ones that are not. *Filtering feature selection methods* allow one to discard redundant features in a model independent way. By reducing the number of variables or discretizing them we simplify the model and increase it's interpretability. It is also a step which tackles multicollinearity which kills stability and predictive power of some models.

2.2.2.1 Continuous Features We have four continuous variables in our dataset: `Age`, `DepartureDelay`, `ArrivalDelay` and `FlightDistance`. We start visualizing their co-dependence structure. We note a high linear relationship between `ArrivalDelay` and `DepartureDelay`, visible both in the correlation matrix (fig. 2) and on their scatterplot (fig. 3). Therefore we can safely drop `ArrivalDelay`, since it doesn't introduce much new information.

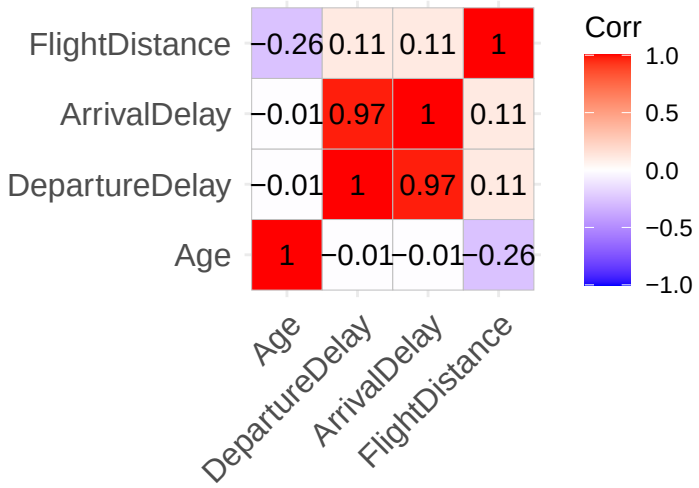

Figure 2: Correlation between continuous scale variables.

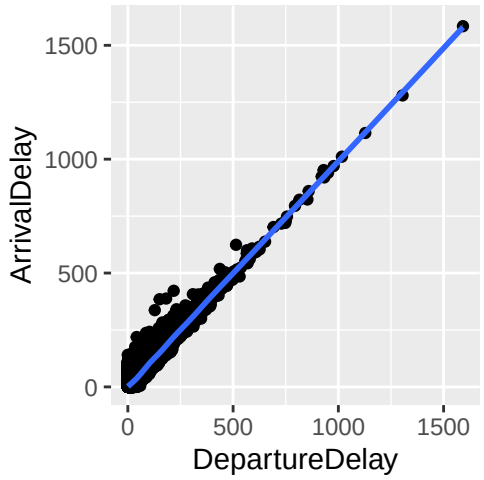

Figure 3: Strong linear relationship between departure delay and arrival delay allows to drop one of them from the dataset.

Next, we're going to examine the loess estimator of satisfaction as a function of the remaining continuous variables to see if any of them looks flat enough to raise suspicion regarding it's utility. Flatness of loess implies that average satisfaction does not change in the explanatory variable, hence the explanatory variable doesn't convey much information. In our case, this is not visible on the figure 4, so we cannot discard any feature based on that. Note, the plot has been generated on a randomized data sub-sample for computational complexity reduction. We made sure however to take a sample large enough, so that the standard errors are tamed, and to ensure the relationship shape is stable regardless of the random seed chosen.

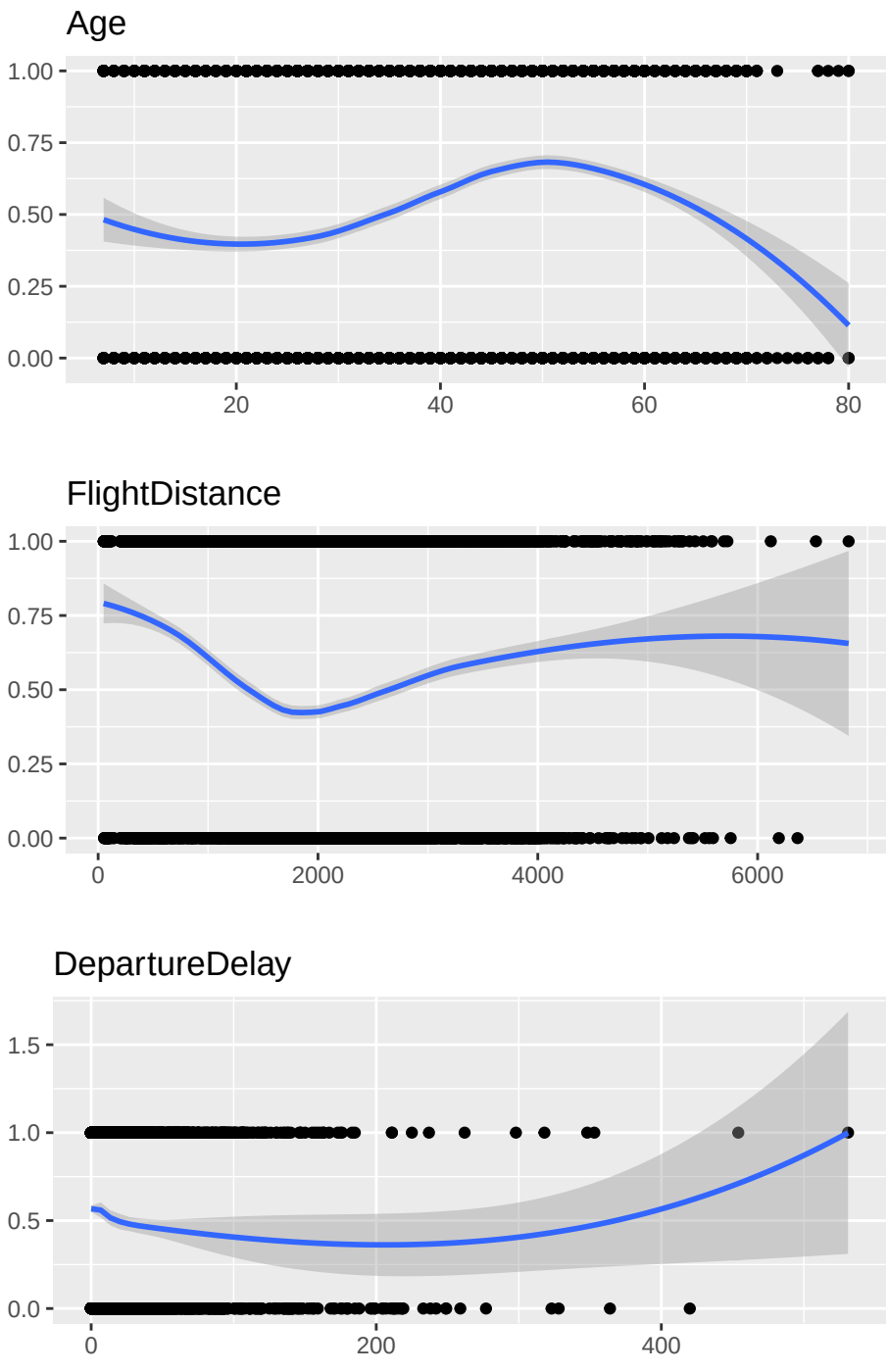

Figure 4: Loess estimator for IsSatisfied as function of continuous features from the dataset.

Now we are going to look at possible binnings of our features. We'll use `woeBinning::woe.binning` function which chooses the binning to maximize the information value of the feature. If the optimized binning will still yield $IV < 0.1$, we will discard the variable. Otherwise we'll analyze the bins to ensure they are not over-optimized to an unreasonable degree. Based on loess plot shapes/regimes the expectation is to have not more than four bins for `Age` and `FlightDistance`, and a maximum of two bins for `DepartureDelay`.

WOE Table for Age

	Final.Bin	Total.Count	Total.Distr.	0.Rate	WOE	IV
1	<= 28	30896	25.9%	58.6%	-51.4	0.068
2	<= 40	29312	24.6%	49.0%	-12.6	0.004
3	<= 59	47672	40.0%	33.9%	50.1	0.096
4	<= Inf	11375	9.5%	53.2%	-29.3	0.008
6	Total	119255	100.0%	45.9%	NA	0.177

WOE Table for FlightDistance

	Final.Bin	Total.Count	Total.Distr.	0.Rate	WOE	IV
1	<= 1359	29839	25.0%	33.3%	53.0	0.067
2	<= 3053	71534	60.0%	53.4%	-30.2	0.055
3	<= Inf	17882	15.0%	36.7%	38.0	0.021
5	Total	119255	100.0%	45.9%	NA	0.143

WOE Table for DepartureDelay

	Final.Bin	Total.Count	Total.Distr.	0.Rate	WOE	IV
1	<= 19	95920	80.4%	44.0%	7.6	0.005
2	<= Inf	23335	19.6%	53.6%	-31.1	0.019
4	Total	119255	100.0%	45.9%	NA	0.024

From the WOE tables above we see that `DepartureDelay` is a variable of low predictive power, hence we won't use it in modelling. For the other variables, as the data binning chosen by the algorithm seems reasonable given the ex-ante expectations, we're going to keep them.

2.2.2.2 Ordinal & Categorical Features The main challenge of the data preparation in this dataset is the proper treatment of passenger survey notes. Take for example the `SeatNote` feature which is a note describing their satisfaction level with the seat. One could ask himself the following questions:

- Are we sure the passenger had seat comfort in mind? Maybe rather their seat location? Or a possibility of choosing the seat?
- Does '$SeatNote$' = 3 imply a negative attitude towards a service? Or it's a moderate 'OK'?
- Is the satisfaction *"difference"* between notes 3 and 2 the same as between notes 5 and 4?
- Is a note '$SeatNote$' = 5 given '$Class$' = 'Eco' the same as '$SeatNote$' = 5 given '$Class$' = '$Business$'?

The same point is valid for any note-type variable in the dataset. The issue boils down to the problem that there is **no universal "unit" of satisfaction**. It is just as non-trivial to measure it as to predict it - simply because everyone perceives satisfaction in a subjective way. Our problem has an additional layer of complexity since we don't have information how precisely the survey questions have been described to the customers - so even if we *did* have some carefully designed satisfaction unit, we cannot be sure if all respondents referred to the same aspects of service when filling out the survey.

We aim to overcome this problem, by binning notes into wider classes, depending on how well they explain and affect the overall satisfaction. Since we have a lot of 5-leveled `Note` factors, we strongly believe that not all levels differ in overall satisfaction contribution. In other words, perhaps we could collapse notes of $1, 2\&3$ to one group if they carried similar information. Hence we will again let `woe.binning` automatically select bins and then verify the result.

Since there are 12 `Note` variables, we will only display one of the WOE tables as an example. However for all it has been verified that **adjacent** levels have been binned, so the binning is plausible, and the IV of the newly binned features are above 0.1.

Table 6: WOE table for eBookingNote.

Final.Bin	Total.Count	Total.Distr.	0.Rate	WOE	IV
1	11713	9.8%	86.7%	-203.8	0.329

Final.Bin	Total.Count	Total.Distr.	0.Rate	WOE	IV
3 + 2	38010	31.9%	70.9%	-105.6	0.340
5 + 4	69532	58.3%	25.3%	91.7	0.443
Total	119255	100.0%	45.9%	NA	1.111

The table 7 presents all of the binned features and their IVs. We see that there are features with IV smaller than 0.1 - we are going to drop those from the dataset.

Lastly, a brief look at the spearman rank correlation matrix (fig. 5) shows that there are no highly correlated "note" features among ordinal variables in the dataset.

Table 7: Information Value for all variables.

varName	IV
IsPersonalTravel	0.0532949
FlightDistance	0.1433763
Age	0.1769584
IsFemale	0.1893377
FoodNote	0.2330999
WifiNote	0.2920904
CheckInNote	0.3607163
Class	0.4001969
IsLoyal	0.4537176
CleanNote	0.4555277
BaggageNote	0.4718849
LegRoomNote	0.5714645
eBoardingNote	0.5948280
ServiceNote	0.6151573
eSupportNote	0.9223110
eBookingNote	1.1114520
SeatNote	1.4504018
EntertainmentNote	2.2947658

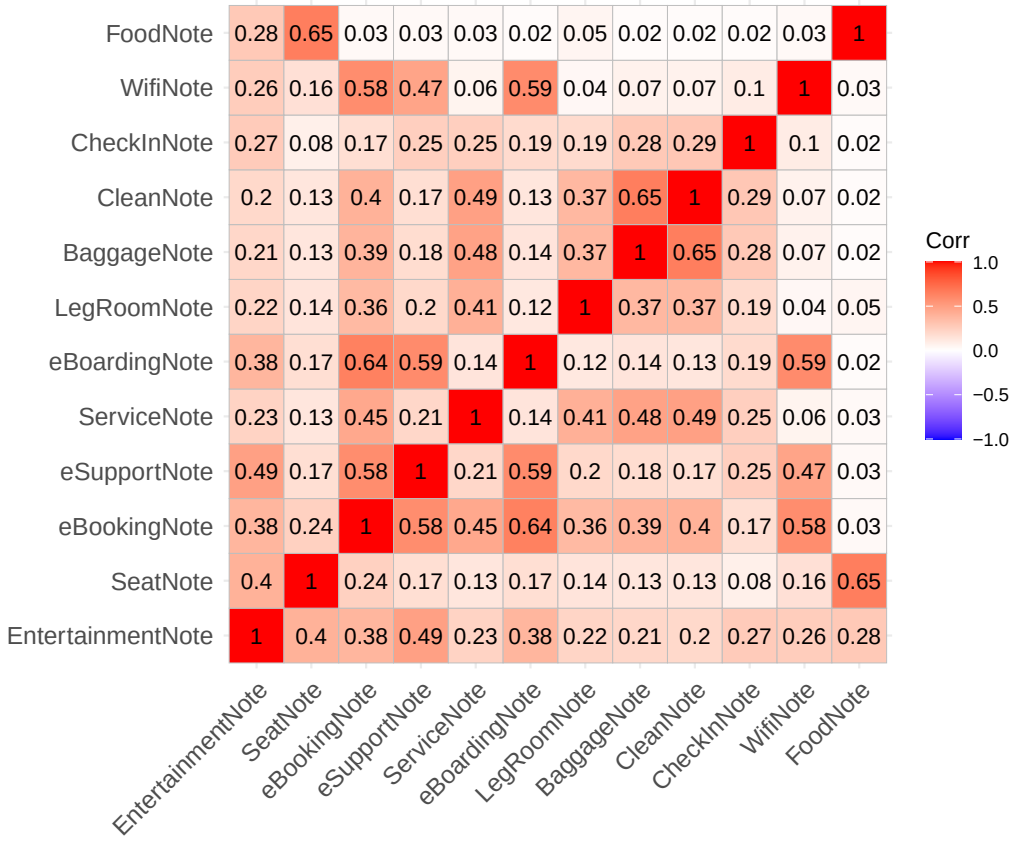

Figure 5: Spearman correlation shows no significant collinear relationships in ordinal variables

2.2.3 Feature Encoding

Some machine learning algorithms which we'll use require numerical data, so we performed **dummy encoding** to transform our data.

That means for each level of factor in our data a separate binary column has been created, and in each factor one of those levels got dropped to avoid having perfect linear relationships.

The resulting, encoded dataframe looks the following way:

```
## Rows: 119,255
## Columns: 34
## $ EntertainmentNote.M <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ EntertainmentNote.H <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ SeatNote.M          <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ SeatNote.H          <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ eBookingNote.H      <fct> 0, 1, 0, 0, 1, 1, 1,~
## $ eBookingNote.M      <fct> 0, 0, 0, 1, 0, 0, 0,~
## $ eSupportNote.M      <fct> 0, 1, 0, 0, 0, 0, 0,~
## $ eSupportNote.H      <fct> 0, 0, 0, 1, 1, 1, 1,~
## $ ServiceNote.M       <fct> 0, 1, 1, 0, 0, 1, 1,~
## $ ServiceNote.L       <fct> 0, 0, 0, 1, 1, 0, 0,~
## $ eBoardingNote.H     <fct> 0, 1, 0, 0, 1, 1, 1,~
## $ eBoardingNote.M     <fct> 0, 0, 0, 1, 0, 0, 0,~
## $ LegRoomNote.M       <fct> 0, 1, 1, 0, 0, 1, 0,~
## $ LegRoomNote.H       <fct> 0, 0, 0, 1, 1, 0, 0,~
## $ BaggageNote.M       <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ BaggageNote.H       <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ CleanNote.M         <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ CleanNote.H         <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ CheckInNote.L       <fct> 0, 0, 1, 1, 1, 0, 0,~
## $ CheckInNote.H       <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ WifiNote.H          <fct> 0, 1, 0, 0, 1, 1, 1,~
## $ WifiNote.M          <fct> 0, 0, 0, 1, 0, 0, 0,~
## $ FoodNote.M          <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ FoodNote.H          <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ Age.30s             <fct> 0, 0, 1, 0, 0, 0, 0,~
## $ Age.40s50s          <fct> 1, 1, 0, 1, 0, 0, 1,~
## $ Age.60plus          <fct> 0, 0, 0, 0, 1, 0, 0,~
## $ FlightDistance.M    <fct> 0, 0, 1, 1, 1, 1, 1,~
## $ FlightDistance.H    <fct> 1, 1, 0, 0, 0, 0, 0,~
## $ Class.Business      <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ Class.EcoPlus       <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ IsSatisfied         <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ IsFemale            <fct> 0, 0, 0, 0, 0, 0, 0,~
## $ IsLoyal             <fct> 1, 1, 1, 1, 1, 1, 1,~
```

2.3 A train-test split of data

At this point, we have 2 dataframes, `Airlines_binned` and `AirlinesEncoded_binned` which we'll provide to the use of modellers. Before we start working on model fitting, we reserve a random 20% of observations as a final test set, for comparing model performances.

Modellers will have the remaining 80% available for their needs regarding model fitting and validation.

3 Baseline Model

Before jumping straight into cutting-edge mathematical models it can be very teaching to fit a simple one and analyze how well it can perform on a given problem. It sets a ground zero for any more complicated models to follow, and gives an idea about how complex is the problem at hand.

Therefore we will fit a simple adaptive linear neuron (Adaline) to our train set and evaluate it's performance on the test set. It is a binary classification model, designed to find a decision boundary for *linearly separable* datasets. However even if the data is not perfectly separable, we can still fit the algorithm and optimize a chosen loss function - *MSE* in our case.

3.1 Fitting and performance

Conceptually, Adaline is just a single layer neural network. It consists of two parts:

- A linear combination of inputs which is piped through an identity activation function - this is used for learning weights
- A unit step decision function - this part enables performing binary predictions.

Adaline optimizes the weights of that linear combination (a.k.a. the *net input*) to arrive at the best fit on the training set.

We will put our own spin on the adaline algorithm to avoid overfitting. Additional *L2* regularization during training will give the model an incentive to opt for smaller weights, and an *early stopping* callback will halt the training

for us, should the validation loss start to plateau and stop decreasing for at least 3 consecutive epochs.

For the baseline model, as it only serves the purposes of testing the waters, we won't be performing costly cross-validation - but rather only compare key model performance indicators across train and test sets to see what we should expect from the challenger models. Figure 6 presents that there are no significant performance differences between training and test sets across a selection of metrics, therefore we conclude that the model is able to generalize the learned rules well for unseen data and is not overfitted.

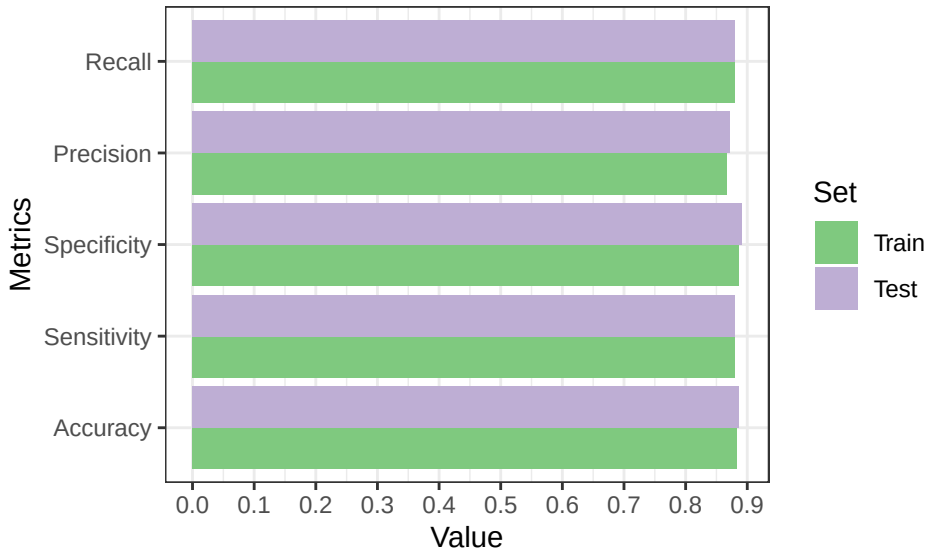

Figure 6: Performance metrics for the baseline Adaline model.

In the case of our dataset, since we're working with one-hot-encoded binary features, the magnitudes and signs of fitted adaline weights can be used as proxies for the directions and magnitudes of the influence on the overall satisfaction for a particular feature. More precisely, the weights tell us by how much will the *net input* increase if a particular feature goes from 0 to 1. Once the net input (corrected for bias) exceeds 0.5, this model will predict a satisfied passenger.

The top 3 influencial feature-note pairs (in a positive and negative sense) according to Adaline model are summarized in table 8. One should note

that the coefficients translate to overall satisfaction **indirectly**, since e.g. `SeatNote.H=1` implies `SeatNote.M=0`, and these effects offset in the overall net input. However the coefficients should be sufficient to analyze at least the overall importance of particular features.

We see that solid `Entertaiment` and `Seat` notes can severely sway the satisfaction in the positive direction. On the other hand, we have low `Service` notes that moderately drag down the final satisfaction. The biggest negative coefficients that we see are corresponding to `Food`, but they don't seem make intuitive sense. Here a high food note would impact satisfaction more negatively that a low food note. However it is entirely possible that this is due to some missing variable interactions which Adaline is not designed to pick up.

Table 8: Top 3 positive and negative weights for the Adaline baseline model.

Rank	Feature	Weight
1	EntertainmentNote.H	0.168
2	EntertainmentNote.M	0.134
3	SeatNote.H	0.116
31	FoodNote.M	-0.027
32	ServiceNote.M	-0.030
33	ServiceNote.L	-0.030

Since Adaline's accuracy on the test set is 0.886, we can infer that our data is not hard to separate, or at least it is relatively easy to reach a reasonable performance. For all challenger models we will be aiming to achieve better metrics, which must compensate for additional model complexity.

4 The Challenger Models

4.1 Random Forest

4.1.1 Fitting and performance

As the first modelling option we will try out the Random Forest model. First, we'll consider the below model which includes all of the variables.

```
frm<-IsSatisfied~Class+IsFemale+IsLoyal+Age+FlightDistance+
      EntertainmentNote+SeatNote+eBookingNote+eSupportNote+
      ServiceNote+eBoardingNote+LegRoomNote+BaggageNote+
      CleanNote+CheckInNote+WifiNote+FoodNote
```

```
##
## Call:
##  randomForest(formula = frm, data = Airlines_binned_train, importance = TRUE)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 4
##
##         OOB estimate of  error rate: 6.18%
## Confusion matrix:
##       0     1 class.error
## 0 40963  2777  0.06348880
## 1  3123 48541  0.06044828
```

After running Random Forest with all variables we can already be quite pleased with the results. Only 6.18% of the observations were misqualified thus we obtained a model with 93.82% accuracy.

We denote number of variables sampled at each split as **mtry**. This parameter allows us to optimize the Random Forest to our problem. By optimize we mean choose **mtry** such that value of OOB estimate of error rate is the lowest.

```
##       mtry   OOBError
## 3.OOB    3 0.06360320
## 4.OOB    4 0.06175842
## 6.OOB    6 0.06349839
```

As we see, the value chosen by default is the best one.

Random Forest enables us to determine how important each variable is in the model, two criteria are used:

- MeanDecreaseAccuracy - expresses the accuracy lost by leaving particular variable out of the training set.
- MeanDecreaseGini - measures node impurity at each split, highest purity means that each node contains only elements of a single class.

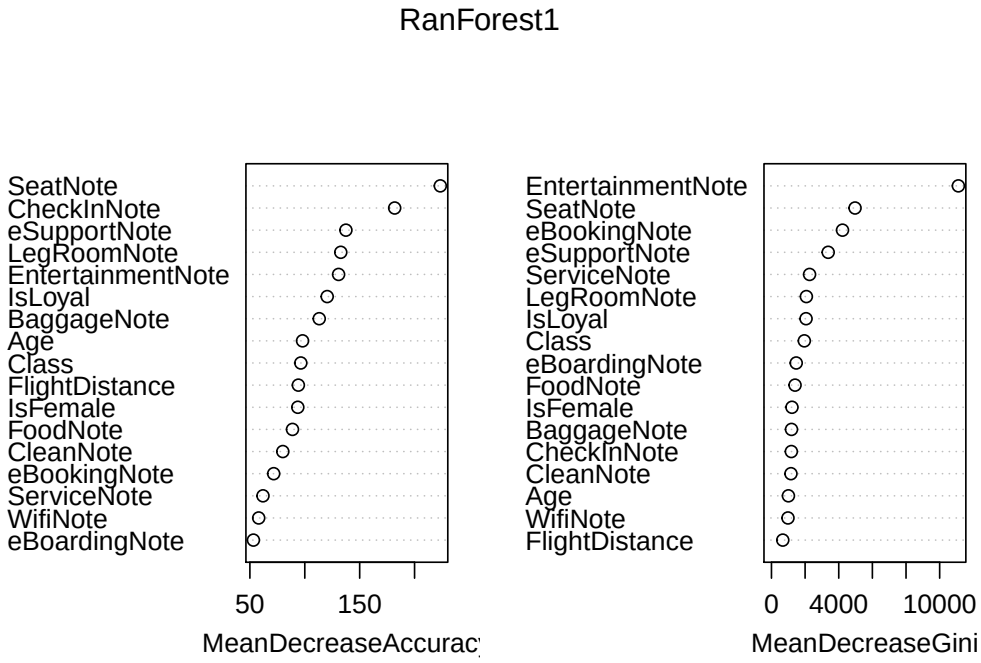

Figure 7: Importance of variables measured by the mean decrease of accuracy and Gini score.

In terms of accuracy of the model, eBoardingNote, WifiNote and ServiceNote seem to be the least important. The most influential are SeatNote, CheckInNote and eSupportNote. The accuracy of the model could drop significantly if we left them out. As for Gini coefficient, the most important are EntertainmentNote, SeatNote and eBookingNote and the least ServiceNote, WifiNote and eBoardingNote.

SeatNote is placed high on both plots, so we may assume that this variable has strong influence on satisfaction level. The same applies to eSupportNote and EntertainmentNote. While thinking of clients' satisfaction, the airline should definitely focus on providing them with comfortable seats and proper online customer support. Moreover, making sure that passengers are entertained throughout the flight (e.g. movies, headphones with music) will definitely help them look back at the flight with nothing but enjoyment.

While thinking of the least important variables we have one immediate candidate - WifiNote - it seems like availability and quality of WiFi service does not matter that much to the customers.

Now, with the help of the ROCurve, we will check the performance of this model.

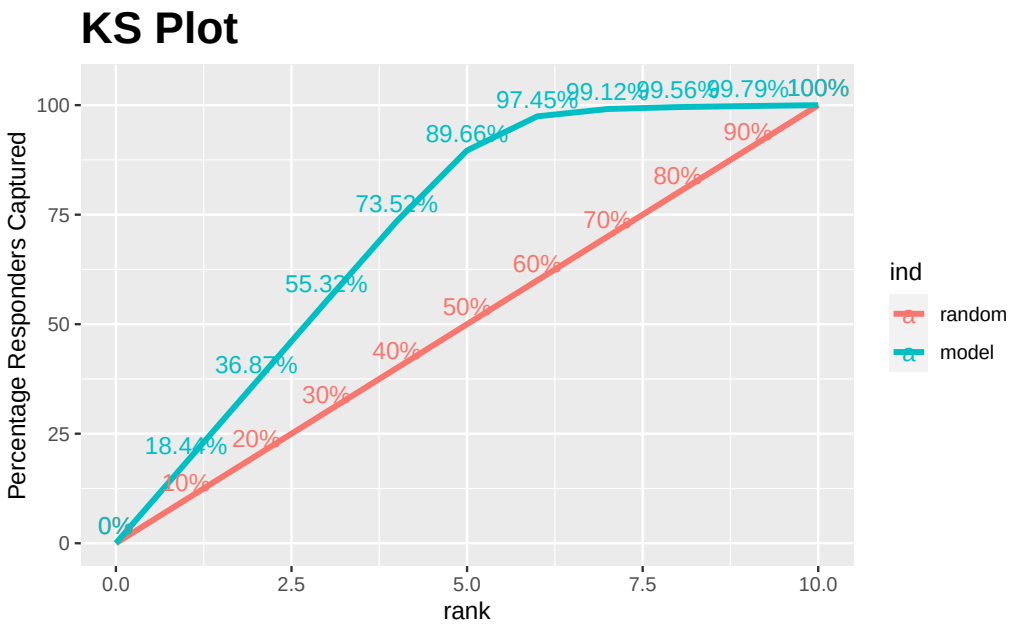

Figure 8: The KS Plot for the random forest.

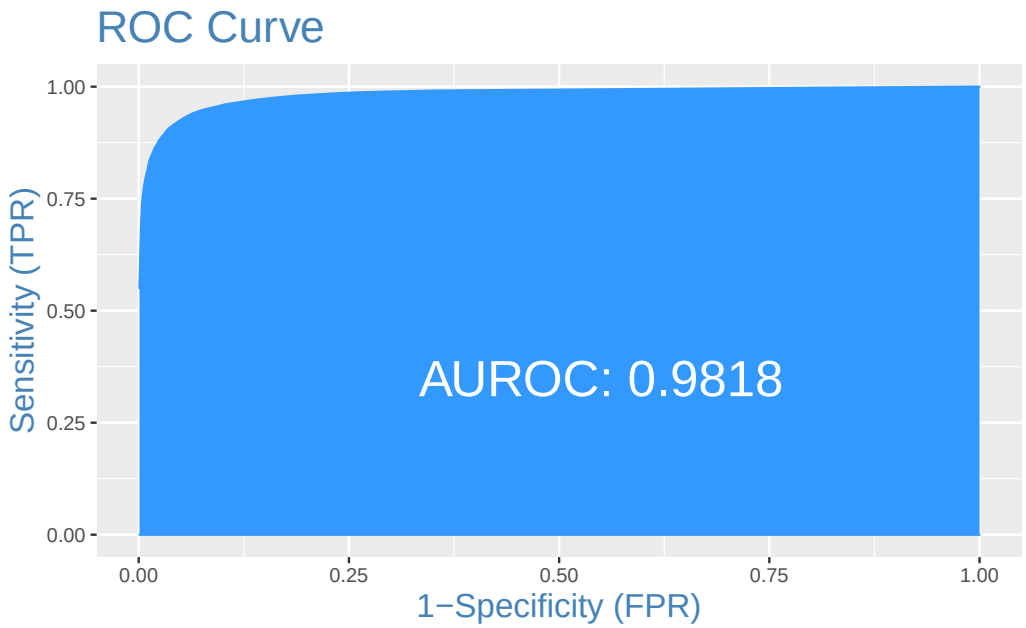

Figure 9: The ROC for the random forest.

The model has an AUC of 0.9826568 on the train set and 0.9371992 on the test set. As we observe a drop of AUC presume that this model shows signs of overfitting to the train set.

We decided to explore some options to improve the model but did not succeed. Excluding any variables resulted in making accuracy worse, and still not stable on the test set. Adding interactions didn't change accuracy or performance of the model at all. We therefore concluded on the original model as the for Random Forest representative.

We can now approach validation part.

4.1.2 Validation

We will validate the model using the Monte Carlo Cross Validation. We will draw 100 times a training data-subset containing of 70% of observations, fit the model and then study the AUC on the train and test sets.

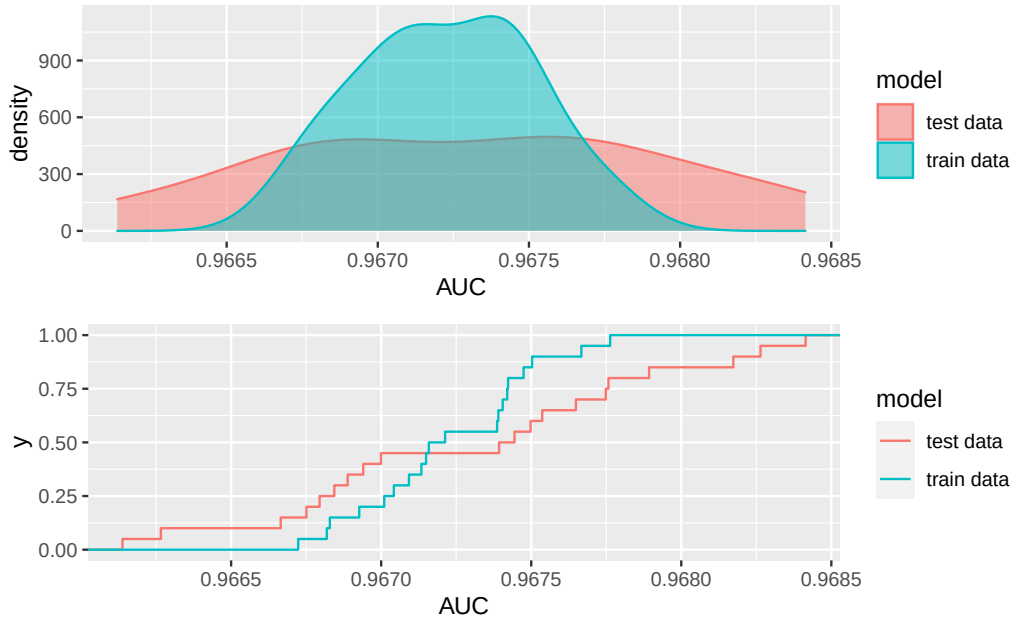

Figure 10: The results of the cross validation for the random forest.

Observed values for the AUC of the test data have the median equal to 0.9674186, the average equal to 0.9673029 and standard deviation equal to 6.4559167×10^{-4}. Even though from the plots it seems like we achieve a larger variance of AUCs on test set, it is nevertheless still a small number. This means our model has worse, yet stable performance on the test set. We are content with the results, as we have AUCs of training and test sets keep in line with each other. The model may be a bit over-fitted to train set, but manages to provide a consistent performance on the test set.

4.2 Logistic Regression

4.2.1 Fitting and performance

The next model that we will study is a logistic regression. We start from the most basic model which takes all the variables.

```
frm<-IsSatisfied~Class+IsFemale+IsLoyal+Age+FlightDistance+
      EntertainmentNote+SeatNote+eBookingNote+eSupportNote+
      ServiceNote+eBoardingNote+LegRoomNote+BaggageNote+
      CleanNote+CheckInNote+WifiNote+FoodNote
```

```
##
## Call:
## glm(formula = frm, family = "binomial", data = Airlines_binned_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.1699  -0.3623   0.0313   0.3099   3.6301
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -7.21962    0.08599 -83.961  < 2e-16 ***
## ClassBusiness       1.34302    0.02605  51.550  < 2e-16 ***
## ClassEcoPlus        0.03494    0.04385   0.797 0.425568
## IsFemale1           0.97658    0.02355  41.466  < 2e-16 ***
## IsLoyal1            1.91910    0.03742  51.283  < 2e-16 ***
## Age30s             -0.02214    0.03196  -0.693 0.488560
## Age40s50s           0.21030    0.03105   6.774 1.25e-11 ***
## Age60plus          -0.37541    0.04313  -8.704  < 2e-16 ***
## FlightDistanceM    -0.19019    0.02919  -6.516 7.21e-11 ***
## FlightDistanceH    -0.13764    0.03939  -3.494 0.000475 ***
## EntertainmentNoteM  1.82628    0.02847  64.152  < 2e-16 ***
## EntertainmentNoteH  2.99102    0.04336  68.979  < 2e-16 ***
## SeatNoteM           0.85488    0.03330  25.670  < 2e-16 ***
## SeatNoteH           5.13222    0.11098  46.246  < 2e-16 ***
## eBookingNoteH       1.64792    0.06294  26.180  < 2e-16 ***
## eBookingNoteM       0.85178    0.06052  14.073  < 2e-16 ***
## eSupportNoteM       0.22928    0.03314   6.918 4.59e-12 ***
## eSupportNoteH       0.47526    0.03731  12.739  < 2e-16 ***
## ServiceNoteH        0.72232    0.04724  15.290  < 2e-16 ***
## ServiceNoteM        0.22391    0.04594   4.874 1.09e-06 ***
## eBoardingNoteH      0.19938    0.03785   5.267 1.38e-07 ***
## eBoardingNoteM      0.32782    0.03447   9.509  < 2e-16 ***
```

```
## LegRoomNoteM        0.03922    0.04776   0.821 0.411536
## LegRoomNoteH        0.75533    0.04772  15.827  < 2e-16 ***
## BaggageNoteM        0.10721    0.03359   3.192 0.001412 **
## BaggageNoteH        0.46100    0.03863  11.932  < 2e-16 ***
## CleanNoteM          0.06595    0.03497   1.886 0.059276 .
## CleanNoteH          0.48344    0.03958  12.214  < 2e-16 ***
## CheckInNoteM        0.33644    0.02874  11.706  < 2e-16 ***
## CheckInNoteH        0.90613    0.03728  24.307  < 2e-16 ***
## WifiNoteH          -0.13901    0.05062  -2.746 0.006025 **
## WifiNoteM           0.12726    0.04902   2.596 0.009435 **
## FoodNoteM          -0.48163    0.03440 -14.000  < 2e-16 ***
## FoodNoteH          -0.51322    0.04304 -11.924  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 131599  on 95403  degrees of freedom
## Residual deviance:  51904  on 95370  degrees of freedom
## AIC: 51972
##
## Number of Fisher Scoring iterations: 7
```

We notice that according to this model:

- business class passengers have a much higher satisfaction level than eco plus class passengers, who in their turn have a higher satisfaction level than eco class ones,
- satisfaction level is higher for females and loyal customers,
- passengers at the age of 40s and 50s have a higher satisfaction level than the others,
- satisfaction level is lower for long and medium flight distances than for the short ones,
- the higher entertainment, seat, eBooking, eSupport, service, leg room, clean, baggage and check-in notes, the higher the satisfaction level.

What we find unintuitive is that taking into account wifi, food and eBoarding the satisfaction level is higher for medium notes than for the high ones. This may be a consequence of variable interactions: wifi and food are usually

available during longer flights. However, the longer the flight, the more people are tired, uncomfortable etc. Nevertheless, we don't have enough background information to make such conclusions. We decide to leave these variables in our model for now and check its performance, but later we will try do improve it.

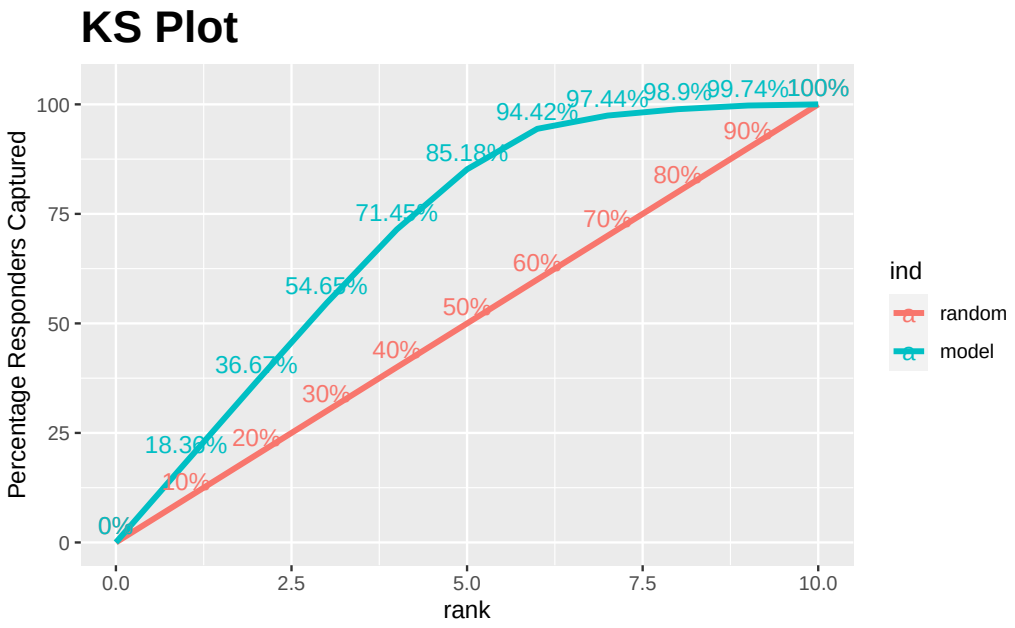

Figure 11: The KS plot for the first logistic regression model.

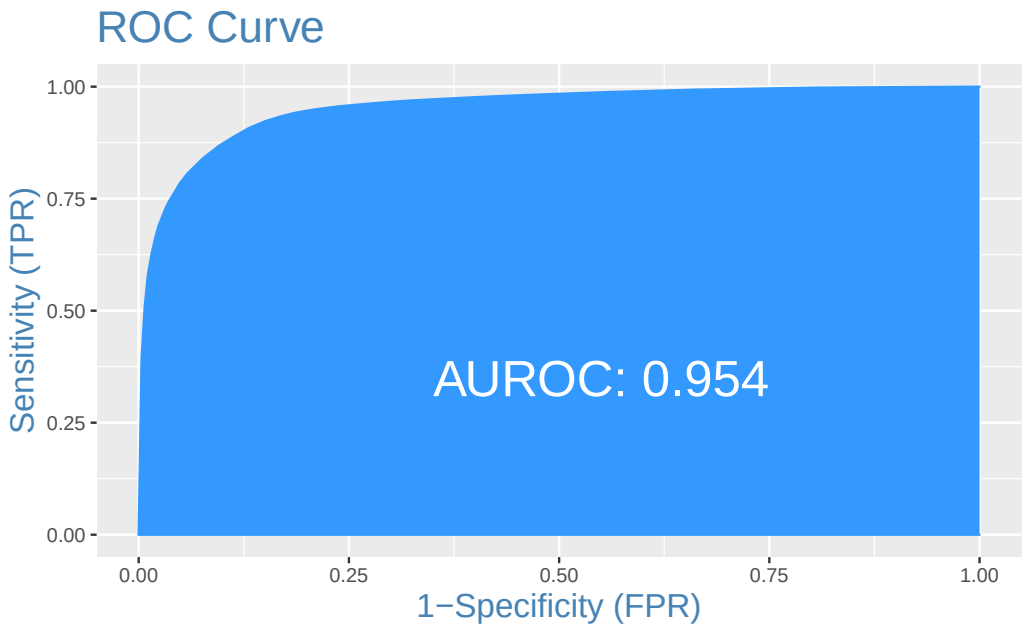

Figure 12: The ROC for the first logistic regression model.

The AUC on the training data is 0.9542947 and on the testing data 0.9541048. The difference is marginal. KS plot and the ROC curve also look great. This model is already a good challenger.

However, we decided to add interactions between variables as some of them seem natural. After many attempts we opt for including the interactions between gender&loyalty, gender&class, gender&age and class&flight distance. We tried also for example class&food or class&service, but they didn't improve our model.

What is more, we decided to not include WifiNote as after adding interactions it was the least significant feature and the relationship was counterintuitive. Thus we get the final logistic regression model, which is summarized below.

```
frm2<-IsSatisfied~Class+IsFemale+IsLoyal+Age+FlightDistance+
      EntertainmentNote+SeatNote+eBookingNote+eSupportNote+
      ServiceNote+eBoardingNote+LegRoomNote+BaggageNote+
      CleanNote+CheckInNote+FoodNote+IsFemale*IsLoyal+
```

```
      IsFemale*Age+IsFemale*Class+Class*FlightDistance
```

```
##
## Call:
## glm(formula = frm2, family = "binomial", data = Airlines_binned_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.1219  -0.3482   0.0296   0.3028   3.5915
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -7.23908    0.09822 -73.699  < 2e-16 ***
## ClassBusiness                 1.74811    0.06208  28.159  < 2e-16 ***
## ClassEcoPlus                  0.42848    0.11666   3.673 0.000240 ***
## IsFemale1                     1.15594    0.07710  14.992  < 2e-16 ***
## IsLoyal1                      1.40869    0.05355  26.307  < 2e-16 ***
## Age30s                        0.21245    0.04727   4.494 6.99e-06 ***
## Age40s50s                     0.55289    0.04442  12.447  < 2e-16 ***
## Age60plus                    -0.33849    0.06615  -5.117 3.10e-07 ***
## FlightDistanceM              -0.22820    0.04753  -4.801 1.58e-06 ***
## FlightDistanceH              -0.44938    0.08464  -5.310 1.10e-07 ***
## EntertainmentNoteM            1.75566    0.02894  60.658  < 2e-16 ***
## EntertainmentNoteH            2.84328    0.04440  64.040  < 2e-16 ***
## SeatNoteM                     0.84514    0.03383  24.985  < 2e-16 ***
## SeatNoteH                     5.15478    0.11218  45.949  < 2e-16 ***
## eBookingNoteH                 1.55519    0.05784  26.888  < 2e-16 ***
## eBookingNoteM                 0.92479    0.05355  17.270  < 2e-16 ***
## eSupportNoteM                 0.16166    0.03321   4.868 1.13e-06 ***
## eSupportNoteH                 0.42142    0.03776  11.161  < 2e-16 ***
## ServiceNoteH                  0.69248    0.04817  14.377  < 2e-16 ***
## ServiceNoteM                  0.17398    0.04644   3.747 0.000179 ***
## eBoardingNoteH                0.22947    0.03721   6.168 6.94e-10 ***
## eBoardingNoteM                0.33404    0.03464   9.644  < 2e-16 ***
## LegRoomNoteM                 -0.02540    0.04874  -0.521 0.602319
## LegRoomNoteH                  0.66935    0.04891  13.686  < 2e-16 ***
## BaggageNoteM                  0.11801    0.03457   3.414 0.000640 ***
```

```
## BaggageNoteH                    0.50288    0.03998  12.577  < 2e-16 ***
## CleanNoteM                      0.10369    0.03610   2.872 0.004076 **
## CleanNoteH                      0.55564    0.04111  13.516  < 2e-16 ***
## CheckInNoteM                    0.34630    0.02908  11.909  < 2e-16 ***
## CheckInNoteH                    0.94897    0.03817  24.862  < 2e-16 ***
## FoodNoteM                      -0.47547    0.03481 -13.660  < 2e-16 ***
## FoodNoteH                      -0.51726    0.04323 -11.966  < 2e-16 ***
## IsFemale1:IsLoyal1              1.08540    0.07105  15.276  < 2e-16 ***
## IsFemale1:Age30s               -0.39527    0.06544  -6.041 1.54e-09 ***
## IsFemale1:Age40s50s            -0.57123    0.06321  -9.038  < 2e-16 ***
## IsFemale1:Age60plus            -0.06771    0.09023  -0.750 0.453013
## ClassBusiness:IsFemale1        -1.53533    0.05220 -29.412  < 2e-16 ***
## ClassEcoPlus:IsFemale1         -0.48121    0.09760  -4.930 8.20e-07 ***
## ClassBusiness:FlightDistanceM   0.49425    0.06107   8.093 5.82e-16 ***
## ClassEcoPlus:FlightDistanceM   -0.19410    0.10774  -1.802 0.071615 .
## ClassBusiness:FlightDistanceH   0.73526    0.09554   7.696 1.41e-14 ***
## ClassEcoPlus:FlightDistanceH   -0.12560    0.20768  -0.605 0.545328
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 131599  on 95403  degrees of freedom
## Residual deviance:  50220  on 95362  degrees of freedom
## AIC: 50304
##
## Number of Fisher Scoring iterations: 7
```

We notice that according to our model:

- business class passengers have a much higher satisfaction level than eco plus class passengers, who in their turn have a higher satisfaction level than eco class ones,
- satisfaction level is higher for females and loyal customers,
- passengers at the age of 40s and 50s have a higher satisfaction level than the others,
- the higher entertainment, seat, eBooking, eSupport, service, leg room, clean, baggage and check-in notes, the higher the satisfaction level,
- generally, the longer the flight distance, the less satisfied people are.

However, this is not the case in business cass, where people are more satisfied on longer distances,

- females travelling business and eco plus class are less satisfied than the ones from eco class and females over 60 are more satisfied than the younger ones.

Now we will study the performance of the model.

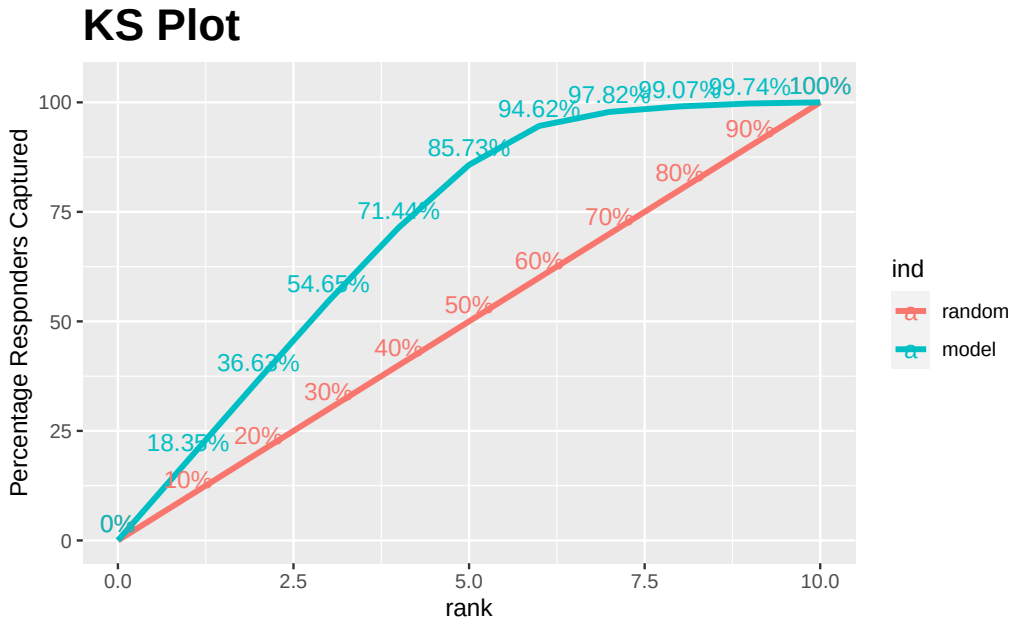

Figure 13: The KS plot for the second logistic regression model.

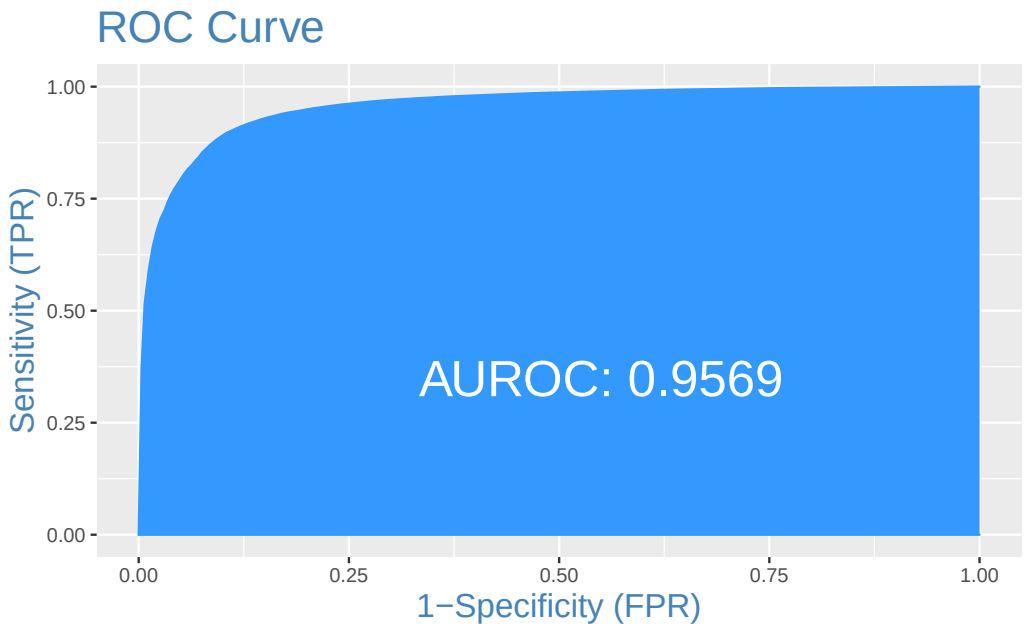

Figure 14: The ROC for the second logistic regression model.

This logistic regression model has an AUC of 0.9573007 on the training data and 0.956471 on the testing data. The difference is again marginal. The KS is 0.7923236. Moreover, we can see (on the ROC curve graph) that our classifier is not far from being perfect. Therefore, we will move forward to the validation part.

4.2.2 Validation

To validate the model we will use the cross validation method and opt for the Monte Carlo Cross Validation.

We use the Monte Carlo Cross Validation with a test data-set that spans 30% of our observations and 70% in the training data-set. We will draw 100 times a training data-set of 0.7 and study the AUC on the testing data-set.

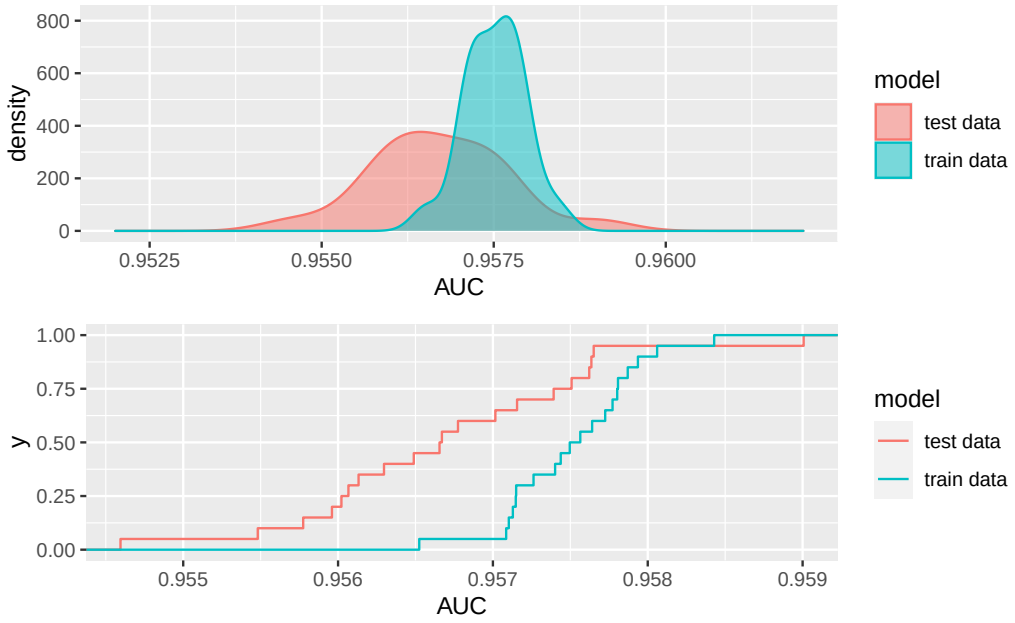

Figure 15: The histogram for the AUC of the randomised data-sets with the Monte Carlo cross validation for the second logistic regression model.

We are very satisfied with the performance of our model. The median of the observed values for the AUC of the test data is 0.9566629, the average is 0.9566953 with a standard deviation of 9.7626465×10^{-4}. We see that dispersion of AUCs during cross-validation is minuscule, and similar between cross-validated test and train sub-sets. We find these results more than satisfactory.

4.3 Neural Network model

Although we can't use Neural Networks to explain customer satisfaction, we also made an attempt to solve the problem with a neural network to understand the upper bound for performance that is achievable on the given problem.

The network consists of the following layers:

- input with 34 nodes,

- 3 hidden ones with 24, 16, 8 nodes respectively,
- output with 2 nodes.

In each layer we have used ReLU (Rectified Linear Unit) activation function, and to regularize the model we applied a dropout of 0.3.

```
## Model: "sequential_1"
## _________________________________________________________________
##  Layer (type)                 Output Shape              Param #
## =================================================================
##  dense_5 (Dense)              (None, 34)                1156
##
##  dropout_3 (Dropout)          (None, 34)                0
##
##  dense_4 (Dense)              (None, 24)                840
##
##  dropout_2 (Dropout)          (None, 24)                0
##
##  dense_3 (Dense)              (None, 16)                400
##
##  dropout_1 (Dropout)          (None, 16)                0
##
##  dense_2 (Dense)              (None, 8)                 136
##
##  dropout (Dropout)            (None, 8)                 0
##
##  dense_1 (Dense)              (None, 2)                 18
##
## =================================================================
## Total params: 2,550
## Trainable params: 2,550
## Non-trainable params: 0
## _________________________________________________________________
```

4.3.1 Fitting and performance

We gave the model 25 epochs to fit to the binned and encoded dataset, with a batch size of 64. During fitting, a sample of 0.1 was used a as a control/validation data to make sure we're not overfitting the model.

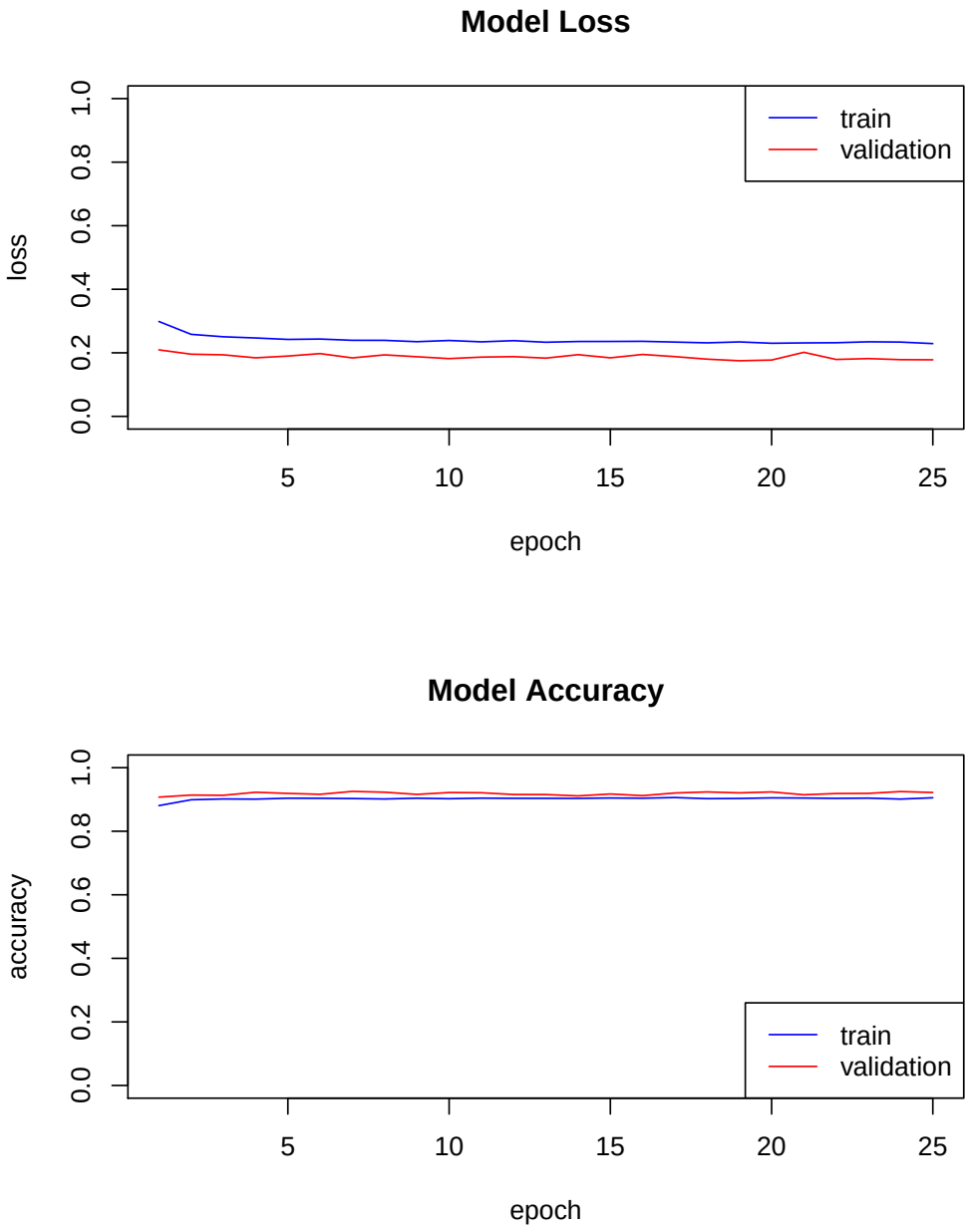

Figure 16: The entire accuracy of the model was achieved in the first epoch of neural network training.

Surprisingly, the neural network has achieved most of its performance after only one epoch. This can be seen in the charts above, and explainable by high level of data separability - which was the reason why even the baseline model achieved high performance.

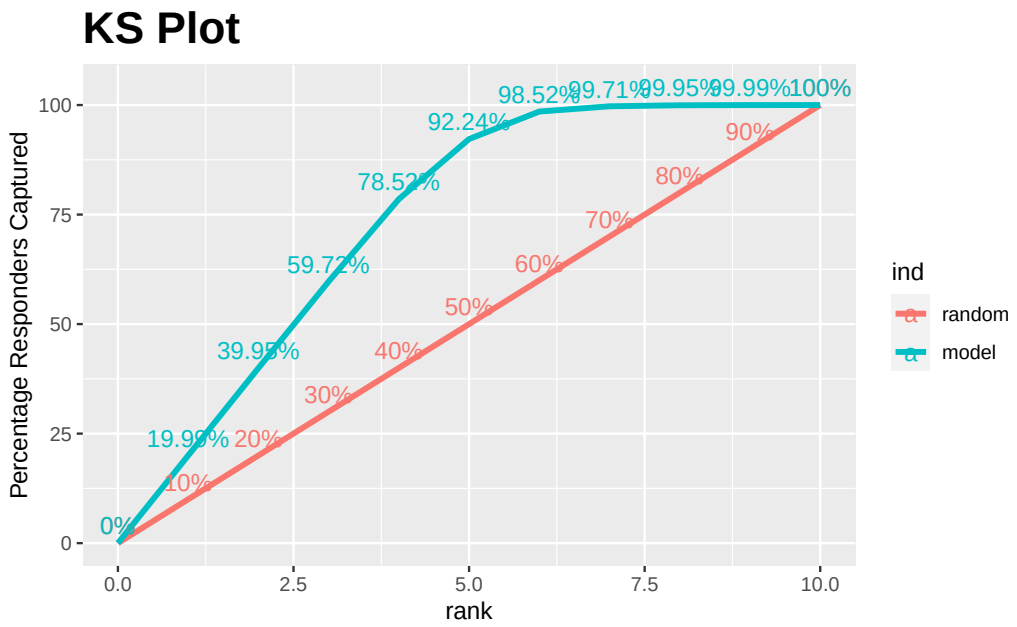

Figure 17: The KS for the neural network.

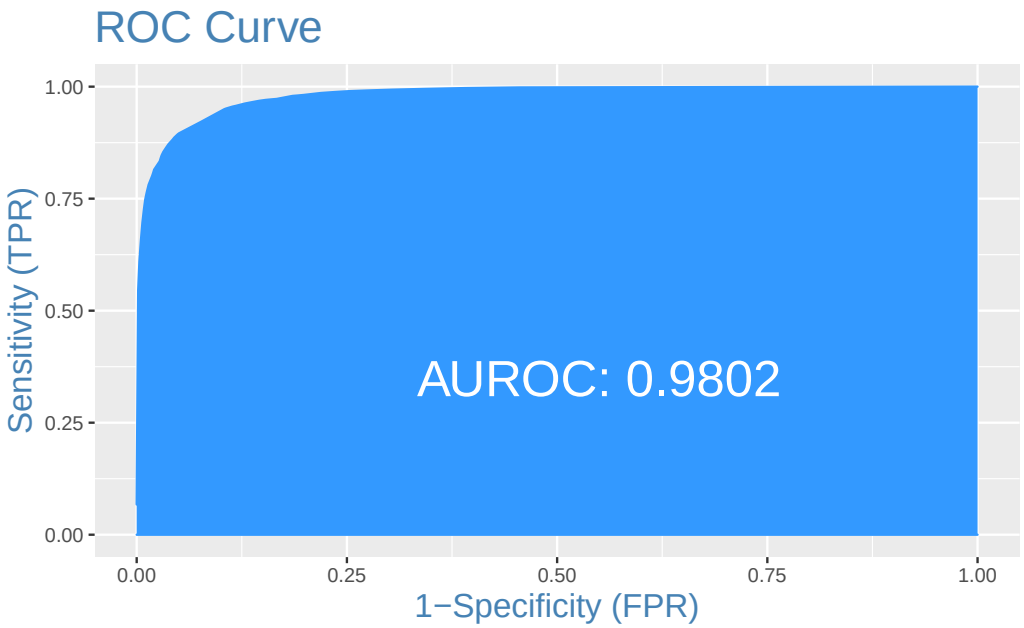

Figure 18: The ROC for the neural network.

4.3.2 Validation

To keep consistency with other models, we have also used Monte Carlo Cross Validation for the Neural Network despite the high computational cost.

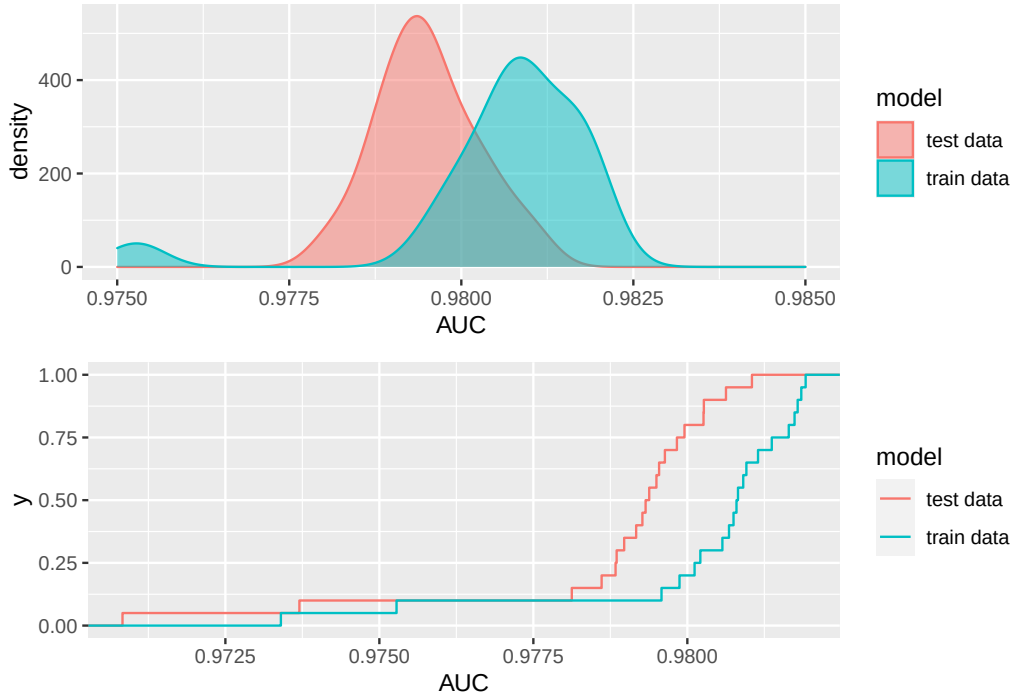

Figure 19: The results of cross validation for the neural network.

Validation showed that the model is highly effective.

The AUC for the neural network model on training data has:

- the median equal to 0.9808101 for train data and 0.9793519 for test data,
- the average equal to 0.9802706 for train data and 0.9787867 for test data,
- the standard deviation equal to 0.0021555 for train data and 0.0023823 for test data.

Moreover, the density plots behave as one would expect - they have a distribution of similar dispersion, but the test data is slightly displaced to the left. Nonetheless, this displacement is acceptable and explainable since in the end model is expected to perform better on the train dataset.

5 Conclusion

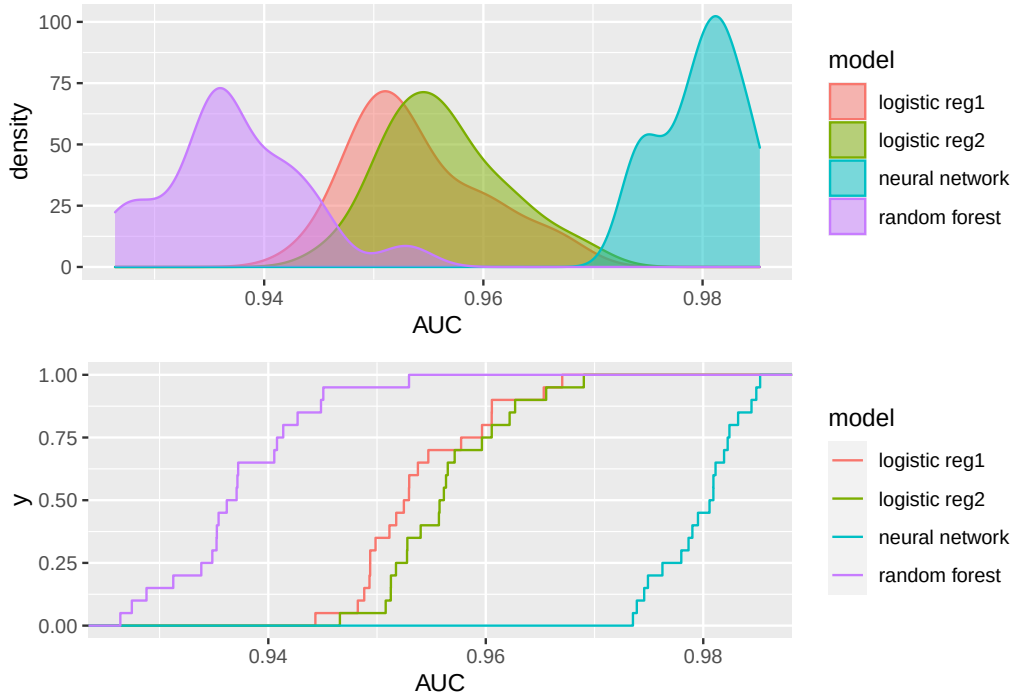

Figure 20: The kernel density for the observed areas under the curve (top) and the cumulative probability density functions (bottom)for the challenger models. All AUCs shown are for the test data only.

In the plots, we showed the accuracy of models with the data they had not seen before. As we can see, all challengers models performed well. The neural network model is the most accurate, but we recommend logistic regression 2 for production - since it is transparent, and explainable so its operation can provide insight about customer preferences for the Airlines Company.

Moreover, the effects achieved by the logistic regression model are of very high performance metrics.

In the following table we summarise these results:

Model	Mean AUC on Test Data	Mean AUC on Training Data
logistic regression 1	0.9539931	0.9542947
logistic regression 2	0.9564216	0.9571064
random forest	0.9372222	0.9672647
neural network	0.9798081	0.9795287

6 Bibliography

De Brouwer, Philippe J.S. 2020. The Big r-Book: From Data Science to Learning Machines and Big Data. New York: John Wiley & Sons, Ltd.