

DATE OF PUBLICATION:
24TH DECEMBER 2023



AIRLINE CUSTOMER SATISFACTION

GROUP 25

AUTHORED BY:
Ali Shihab
Adam Meor Azlan
Paula Rodriguez

TABLE OF CONTENTS

1. Introduction	3
2. EDA & Preparation	6
3. Data Preprocessing	13
4. Modelling & Metrics	18
5. Recommendation	27
6. Contributions	28

INTRODUCTION

1.1 Background

Customer satisfaction is a fundamental performance metric for airlines. Regular reviews of an airline's performance under this metric is a necessary technique to measure overall performance, as well as the factors contributing to its ability to meet revenue goals and the business aims set by key stakeholders. An important task when measuring customer satisfaction is conducting regular customer surveys. These surveys must be of ample size, large enough to be statistically significant and an adequate sample representative of the customer base. Using these surveys, key factor decomposition steps in business analytics can be taken to measure the areas and services where the airline is performing well, as well as areas where the airline needs to improve. Further, it allows airlines to make efficient use of capital by allocating maximally efficient budgets to the most significant contributing services that are the most underperforming.

In 2022, customer satisfaction dropped by 8% due to the rampant cancellations of that year. Despite this, the industry as a whole increased by 1% over the previous year. The necessity of business analytics is in that it can clarify an otherwise counterintuitive result, allowing airlines not to overlook underperforming services despite improvements on the annual term.

1.2 Methodology

The standard methodology of CRISP-DM will be followed, as illustrated below. This begins with gaining a business understanding, establishing the business aims, research objectives, project outcomes and key performance indicators. With reference to this business understanding, the scope of the project is defined.



Subsequently, an understanding of the data, its structure, characteristics and limitations will follow. At each stage, it may be necessary to iteratively revisit prior phases of the cycle to ensure new discoveries are reflected and propagated through our pipeline. This includes data preparation, including cleaning, preprocessing and (as and when appropriate) feature engineering and dimensionality reduction to prepare one or more subsets for comparative modelling. Modelling then follows based on the preparation phase, evaluating based on predefined metrics and iteratively revisiting the previous phases as explained. Deployment is out of the scope of this project, but in such a case, continuous monitoring and regular re-training is required to deal with phenomena such as data drift or concept drift.

Introduction

2.1 The Dataset

The dataset was sourced from Kaggle at <https://www.kaggle.com/datasets/teejmahal20/airline-passenger-satisfaction>. It consists of survey responses by customers of an airline, rating various aspects of their flight experience on scale of 1 to 5. There are 25 features across a training set and test set with cumulative size of 129,880 records. The visualisation of this data in tabular form is below.

2.1.1 Suitability

The dataset was selected as it satisfied the conditions stated under section 1.1. These were that the dataset must be set of customer survey response data that is of significant enough size that it can be deemed representative of the entire customer base. It is also suitable due to being a mix of both quantitative and qualitative survey responses, ensuring that there is variety in the type of data processed to provided a more holistic and robust insight to the customer experience and the patterns and relationships that can be drawn from it.

2.1.2 Complexity

The large number of features allows for ample selection of feature subsets for variable preprocessing techniques and comparisons. However, it is not so high dimensional so as to detrimentally effect the performance of the models, given that there are 129,880 samples for 23 (25 less the redundant ID fields) fields, with a total ratio of over 5,500 samples per feature. Further, given the relatively balanced class of the target variable, the data is suitable because any further class balancing would be optional rather than necessary.

2.1.3 Data Scale

Lastly, the scale of features highly contrasts between the type of the feature. For all discrete ordinal features, the data is within the same order of magnitude. However, the continuous features are in a range that is orders of magnitude larger than the discrete ordinals, but all within the same relative range (save for one feature, "Age"). The number of categorical features is within a reasonable range so as not to require extensive preparation, while also providing enough key customer and flight data.

2.2 Data Overview

In this dataset, there are:

- 14 ordinal discrete features for ratings of in and out-of flight services, such as baggage handling, inflight service, booking service;
- 4 continuous numerical features for customer age, flight distance, flight departure delay and flight arrival delay;
- 3 binary categorical features representing gender, type of flight (business or leisure), and customer type (loyal or disloyal);
- 1 categorical feature representing the flight class (economy, economy plus, business);

Introduction

2 redundant ID fields are also included. Finally, the **target** variable is a binary categorical representing customer **satisfaction**, taking values of “satisfied” or “neutral or dissatisfied”. The target variable is reasonably balanced, with 43% of customers falling under “satisfied”, and 57% “neutral or dissatisfied”. The figure below depicts a tabular view of the data and basic descriptive statistics.

Variable type: character								
	skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
1	Gender	0	1	4	6	0	2	0
2	Customer.Type	0	1	14	17	0	2	0
3	Type.of.Travel	0	1	15	15	0	2	0
4	Class	0	1	3	8	0	3	0
5	satisfaction	0	1	9	23	0	2	0

Variable type: numeric											
	skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
1	X	0	1	44159.	31207.	0	16235.	38964.	71433.	103903	
2	id	0	1	64940.	32493.	1	32471.	64940.	97410.	129880	
3	Age	0	1	39.4	15.1	7	27	40	51	85	
4	Flight.Distance	0	1	1190.	997.	31	414	844	1744	4983	
5	Inflight.wifi.service	0	1	2.73	1.33	0	2	3	4	5	
6	Departure.Arrival.time.convenient	0	1	3.06	1.53	0	2	3	4	5	
7	Ease.of.Online.booking	0	1	2.76	1.40	0	2	3	4	5	
8	Gate.location	0	1	2.98	1.28	0	2	3	4	5	
9	Food.and.drink	0	1	3.20	1.33	0	2	3	4	5	
10	Online.boarding	0	1	3.25	1.35	0	2	3	4	5	
11	Seat.comfort	0	1	3.44	1.32	0	2	4	5	5	
12	Inflight.entertainment	0	1	3.36	1.33	0	2	4	4	5	
13	On.board.service	0	1	3.38	1.29	0	2	4	4	5	
14	Leg.room.service	0	1	3.35	1.32	0	2	4	4	5	
15	Baggage.handling	0	1	3.63	1.18	1	3	4	5	5	
16	Checkin.service	0	1	3.31	1.27	0	3	3	4	5	
17	Inflight.service	0	1	3.64	1.18	0	3	4	5	5	
18	Cleanliness	0	1	3.29	1.31	0	2	3	4	5	
19	Departure.Delay.in.Minutes	0	1	14.7	38.1	0	0	0	12	1592	
20	Arrival.Delay.in.Minutes	393	0.997	15.1	38.5	0	0	0	13	1584	

Dataset after loading - split by datatype (“Character”, “Numeric”) with summary statistics.

Since the target is binary, this is a binary classification problem. Therefore, the following performance metrics will be used:

- **Feature Importance:** necessary to identify most influential factors of satisfaction;
- **Accuracy:** necessary to measure percentage of correctly identified/classified customers;
- **F1-Score:** important in cases of class imbalance & to give equal weighting to precision & recall;
- **ROC-AUC:** maximizing model's ability to differentiate between classes.

2.3 Model Selection

Given the large sample-to-feature ratio and the nature of the problem being a binary classification one, the following models were selected for comparison:

- **Deep Neural Network:** for its superior ability to find non-linear relationships;
- **Support Vector Machine:** for its singular decision boundary and flexibility of the kernels;
- **Decision Tree:** for simplicity and explainability;
- **Random Forest:** for enhanced capabilities on the Decision Tree;
- **Logistic Regression:** for simplicity and as a standard baseline;
- **K-Nearest Neighbours:** for its flexibility and consideration of locality;

EDA & Preparation

3.1 Data preparation

3.1.1 Duplicate Records

Analysis of the unique record of “ID” and “X” columns showed an approximately 26,000 record discrepancy, indicating duplicate records possibly due to the combination of the train and test sets obtained, of which the test set may have been a subset of the train set already. Otherwise, it could have been due to separate indexing of the two sets of data:

```
X:      103904 id:      129880
```

Number of unique records for each column “X” and “ID”.

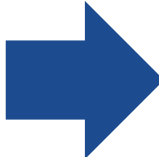
After filtering all duplicate records out using R, the dataset remained the same, indicating that the phenomenon was due to separate indexing. Therefore we remain with 129,880 records, still.

3.1.2 Feature types

Of the 14 discrete ordinal features, 13 features were ratings from 0 to 5 and 1 - “Baggage Handling” - was a rating from 1-5. When scaling, this (minor) difference will be addressed. Given the natural order of ratings features, these can be converted to the R “factor” data type - and then numerical at the scaling or normalisation step - and do not need to be one-hot encoded due to preservation of their natural order.

The continuous features only need to be scaled. Of all categorical features, the binary ones were dichotomous and therefore do not have a natural order, where as the “Class” feature does have a natural order. Thus, neither type of categorical needs one-hot encoding, as binary features can just represent either value using a 0 or 1, and the “Class” feature must retain its natural order. All categorical variables will be converted to R “factor” datatypes and, later, numerical types. The following depicts the transform of the feature types:

\$ Gender	: chr	"Male" "Male" "Female" "Female" ...
\$ CustomerType	: chr	"Loyal Customer" "disloyal Customer" "Loyal Customer"
\$ Age	: int	13 25 26 25 61 26 47 52 41 20 ...
\$ TypeofTravel	: chr	"Personal Travel" "Business travel" "Business travel"
\$ Class	: chr	"Eco Plus" "Business" "Business" "Business" ...
\$ FlightDistance	: int	460 235 1142 562 214 1180 1276 2035 853 1061 ...
\$ Inflightwifiservice	: int	3 3 2 2 3 3 2 4 1 3 ...
\$ DepartureArrivaltimeconvenient	: int	4 2 2 5 3 4 4 3 2 3 ...
\$ EaseofOnlinebooking	: int	3 3 2 5 3 2 2 4 2 3 ...
\$ Gatelocation	: int	1 3 2 5 3 1 3 4 2 4 ...
\$ Foodanddrink	: int	5 1 5 2 4 1 2 5 4 2 ...
\$ Onlineboarding	: int	3 3 5 2 5 2 2 5 3 3 ...
\$ Seatcomfort	: int	5 1 5 2 5 1 2 5 3 3 ...
\$ Inflightentertainment	: int	5 1 5 2 3 1 2 5 1 2 ...
\$ Onboardservice	: int	4 1 4 2 3 3 3 5 1 2 ...
\$ Legroomservice	: int	3 5 3 5 4 4 3 5 2 3 ...
\$ Baggagehandling	: int	4 3 4 3 4 4 4 5 1 4 ...
\$ Checkinservice	: int	4 1 4 1 3 4 3 4 4 4 ...
\$ Inflightservice	: int	5 4 4 4 3 4 5 5 1 3 ...
\$ Cleanliness	: int	5 1 5 2 3 1 2 4 2 2 ...
\$ DepartureDelayinMinutes	: int	25 1 0 11 0 0 9 4 0 0 ...
\$ ArrivalDelayinMinutes	: num	18 6 0 0 0 0 23 0 0 0 ...
\$ satisfaction	: chr	"neutral or dissatisfied" "neutral or dissatisfied"



\$ Gender	: num	1 1 0 0 1 0 1 0 0 1 ...
\$ CustomerType	: num	0 1 0 0 0 0 0 0 1 ...
\$ Age	: int	13 25 26 25 61 26 47 52 41 20 ...
\$ TypeofTravel	: num	1 0 0 0 0 1 1 0 0 0 ...
\$ Class	: num	3 1 1 1 1 2 2 1 1 2 ...
\$ FlightDistance	: int	460 235 1142 562 214 1180 1276 2035 853 1061 ...
\$ Inflightwifiservice	: int	3 3 2 2 3 3 2 4 1 3 ...
\$ DepartureArrivaltimeconvenient	: int	4 2 2 5 3 4 4 3 2 3 ...
\$ EaseofOnlinebooking	: int	3 3 2 5 3 2 2 4 2 3 ...
\$ Gatelocation	: int	1 3 2 5 3 1 3 4 2 4 ...
\$ Foodanddrink	: int	5 1 5 2 4 1 2 5 4 2 ...
\$ Onlineboarding	: int	3 3 5 2 5 2 2 5 3 3 ...
\$ Seatcomfort	: int	5 1 5 2 5 1 2 5 3 3 ...
\$ Inflightentertainment	: int	5 1 5 2 3 1 2 5 1 2 ...
\$ Onboardservice	: int	4 1 4 2 3 3 3 5 1 2 ...
\$ Legroomservice	: int	3 5 3 5 4 4 3 5 2 3 ...
\$ Baggagehandling	: int	4 3 4 3 4 4 4 5 1 4 ...
\$ Checkinservice	: int	4 1 4 1 3 4 3 4 4 4 ...
\$ Inflightservice	: int	5 4 4 4 3 4 5 5 1 3 ...
\$ Cleanliness	: int	5 1 5 2 3 1 2 4 2 2 ...
\$ DepartureDelayinMinutes	: int	25 1 0 11 0 0 9 4 0 0 ...
\$ ArrivalDelayinMinutes	: num	18 6 0 0 0 0 23 0 0 0 ...
\$ satisfaction	: num	0 0 1 0 1 0 0 1 0 0 ...

Data pre-transform (left) and post-transform (right) of categorical to factor and then numerical.

EDA & Preparation

3.1.3 Missing Values

Of the 129,880 samples, only 393 had missing values, equating to approximately 0.3% of the dataset which is quite insignificant. All 393 missing values belonged to the “ArrivalDelayInMinutes” feature. Rudimentary data imputation techniques would involve mean, median or mode imputation - a more robust approach would include prediction of these values by some means, for example, linear regression. Given the relatively insignificant proportion of missing values, a comparison of these approaches would likely cause inconsequential changes in the feature-wise distribution of the data, which means a comparison of these changes will likely yield almost identical results. So, we opted to proceed with more robust approach for imputation, using linear regression, detailed later.

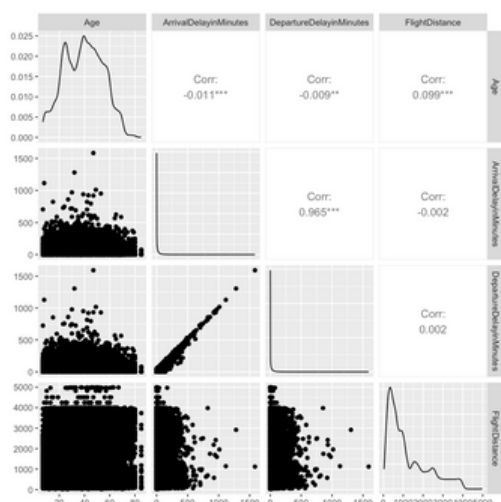
3.2 Descriptive Statistics

3.2.1 Distribution

Continuous Features

Notably, the distribution of both “ArrivalDelayInMinutes” and “DepartureDelayInMinutes” features has significant (mutually comparable) positive skew. Their relative similarity also confirms the intuition that the features are likely extremely collinear, given that both features relate to the delay of the same flight at different points, where planes are unlikely to alter their flight speed, path or other characteristics due to a number of constraints, including fuel consumption, pre-programmed schedules, air traffic and regulations, among others.

Another feature with a skewed distribution is “FlightDistance”, which can be reasoned for due to the general relationship that further flights are more expensive, which less people are able to pay for. As expected, the “Age” feature exhibited an approximately standard Gaussian/Normal distribution. Plots for the distribution and summary statistics are shown below.



Field	Catagorical	Symbols	Name	Min	Mean	Max	Skew
Age	✗ No	-	0	7.00	39.43	85.00	-0.00
ArrivalDelayInMinutes	✗ No	-	0	0.00	15.09	1,584.00	6.67
DepartureDelayInMinutes	✗ No	-	0	0.00	14.64	1,592.00	6.85
FlightDistance	✗ No	-	0	31.00	1,190.21	4,983.00	1.11

Pair & Pearson correlation plot (left) and descriptive statistics (right) including skew of continuous features.

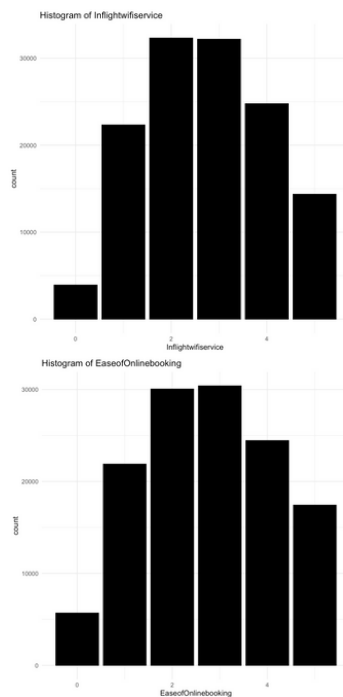
EDA & Preparation

Discrete/Categorical Features

Of all the ratings, none exhibit any significant skew. Further, the categorical features are fairly imbalanced, except for the target features. Example plots of their respective chart and descriptive statistics follow. Note: a more representative statistic description of the discrete features (ratings) might be the relative quantiles, detailed in the initial table describing the data in section 2.2.

Field	Catagorical	Symbols	Name	Min	Mean	Max	Skew
Gender	✓ Yes	2	Female(51%)	-	-	-	-
CustomerType	✓ Yes	2	Loyal Customer(82%)	-	-	-	-
TypeofTravel	✓ Yes	2	Business travel(69%)	-	-	-	-
Class	✓ Yes	3	Business(48%)	-	-	-	-
satisfaction	✓ Yes	2	neutral or dissatisfied(57%)	-	-	-	-

Pair & bar plot (left) and descriptive statistics (right) including skew of continuous features.



Field	Catagorical	Symbols	Name	Min	Mean	Max	Skew
Inflightwifiservice	✗ No	-	0	0.00	2.73	5.00	0.04
DepartureArrivaltimeconvenient	✗ No	-	0	0.00	3.06	5.00	-0.33
EaseofOnlinebooking	✗ No	-	0	0.00	2.76	5.00	-0.02
Gatelocation	✗ No	-	0	0.00	2.98	5.00	-0.06
Foodanddrink	✗ No	-	0	0.00	3.20	5.00	-0.16
Onlineboarding	✗ No	-	0	0.00	3.25	5.00	-0.46
Seatcomfort	✗ No	-	0	0.00	3.44	5.00	-0.49
Inflightentertainment	✗ No	-	0	0.00	3.36	5.00	-0.37
Onboardservice	✗ No	-	0	0.00	3.38	5.00	-0.42
Legroomservice	✗ No	-	0	0.00	3.35	5.00	-0.35
Baggagehandling	✗ No	-	0	1.00	3.63	5.00	-0.68
Checkinservice	✗ No	-	0	0.00	3.31	5.00	-0.37
Inflightservice	✗ No	-	0	0.00	3.64	5.00	-0.69

Example bar plots (left) and descriptive statistics (right) including skew of discrete features.

Intuitively, the more negative the skew of the ratings features, the higher the satisfaction of the customers with those services.

3.3 Customer Satisfaction & Feature Trends

After analysis of the relationship between ratings fields and other symbolic fields, the following section revealed critical insights into customer satisfaction across various airline services. High satisfaction can be observed in entertainment and wifi, baggage handling, and online booking ease, while challenges exist in gate location, seat comfort, and legroom.

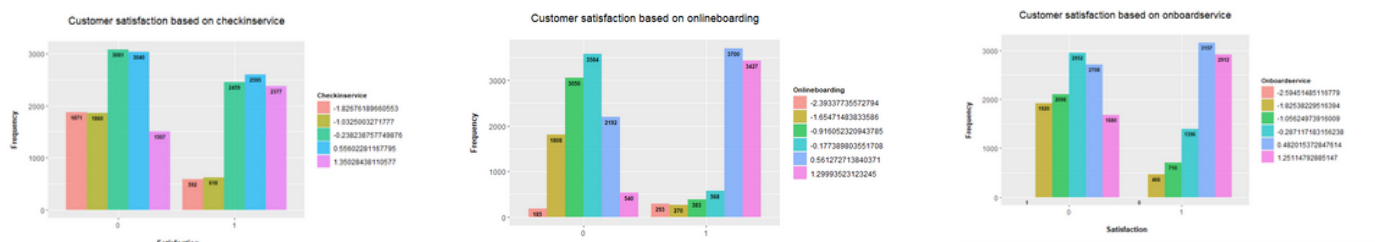
Features like entertainment and wifi stand out as significant positive differentiators, highlighting their importance. However, variability in satisfaction levels in Food and Drink suggests a need for further examination.

EDA & Preparation

Notable dissatisfaction in Seat Comfort and Legroom outlines a critical area for improvement, and even satisfaction distribution in Departure/Arrival timing highlights the need for differentiation. In particular, Baggage Handling shows operational efficiency, but Gate Location poses a significant pain point requiring targeted improvements.

The class service exhibits high satisfaction in Business Class, reflecting positively on service quality. At the same time, Economy and Economy Plus represent a range of mixed feelings, possibly due to tailored services or differing expectations indicating a need for enhanced experiences for economy travellers. Cleanliness receives generally high satisfaction, emphasizing its crucial role in customer comfort. The comfort and convenience-related fields like seat comfort and legroom show a notable dissatisfaction among all customers. Customer Relations analysis reveals no significant gender-based differences, a strong correlation between loyalty and satisfaction, and higher satisfaction among business travellers.

Additional services offered by the airline, like Ease of Online Booking, receive high satisfaction, while Onboard Services exhibit polarized satisfaction. Other factors, including Food and Drink, Cleanliness, and Online Services, highlight the airline's excellence, but Onboard Service and Wifi display a polarized distribution, requiring attention.

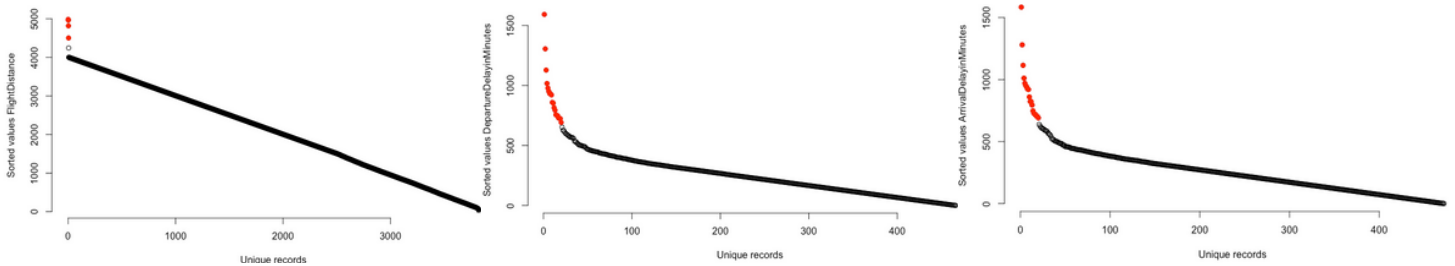


The customer profile analysis reveals satisfaction disparities between loyal and one-time customers, emphasizing the need for robust customer retention strategies. Despite excelling in cleanliness and customer loyalty, challenges in specific dimensions highlight the importance of aligning service delivery with customer expectations. To improve overall customer satisfaction and loyalty, the airline should focus on enhancing economy-class services by addressing comfort and convenience concerns, streamlining check-in processes to match baggage handling standards, and optimizing departure and arrival times to differentiate the airline positively.

3.4 Outlier Detection

Rudimental outlier analysis detected 4 outliers in “FlightDistance”, and 20 in each of the delay-related features using the chi-squared test with a confidence interval of 0.95. Given that the distribution of “FlightDistance” being less skewed, this initially seems plausible. Note: this was performed after imputation of missing values via linear-regression and min-max scaling. The plots for the outliers detected in each feature are shown below.

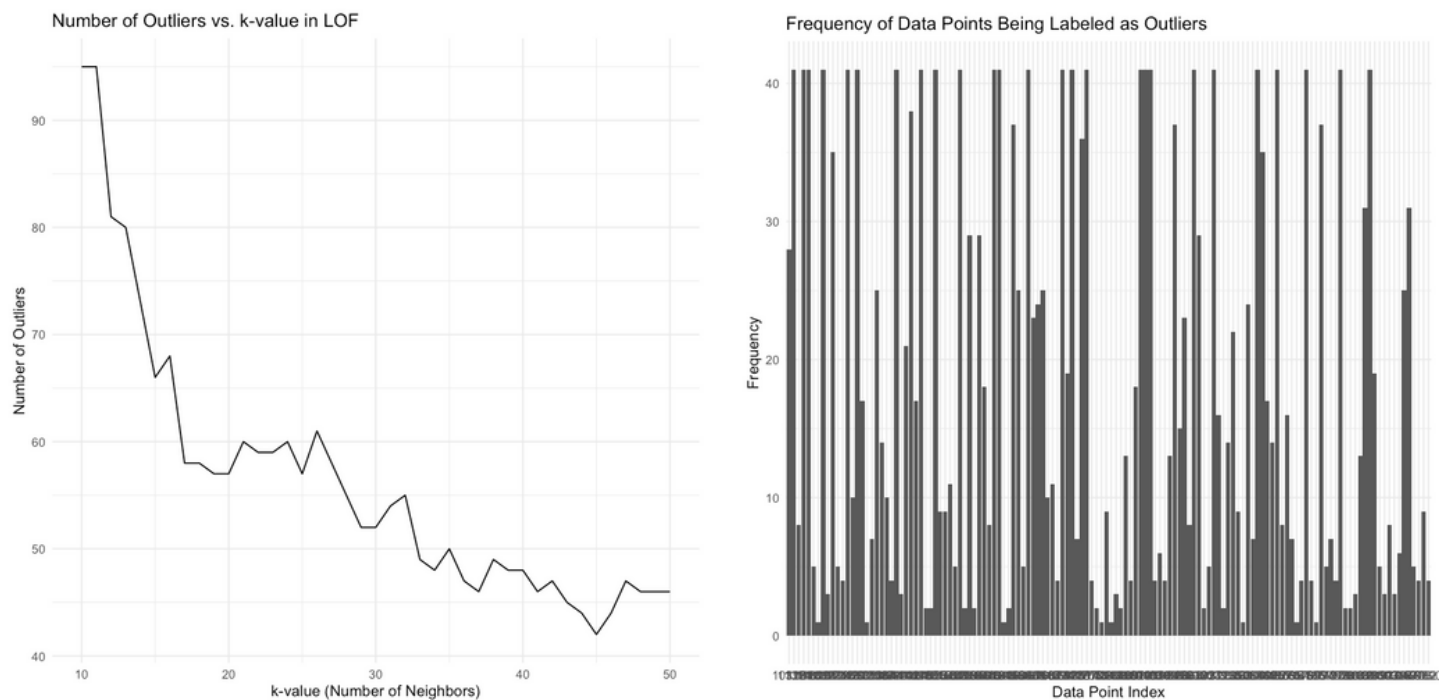
EDA & Preparation



Plots of outliers detected in “FlightDistance” (left), Departure delay (middle), Arrival Delay (right).

However, clearly, more robust outlier detection methods are required. Given the dataset’s skew in the aforementioned features, as well as uncertainty regarding its geometric structure (spherical or non-spherical), a density-based anomaly detection technique seemed most appropriate. Specifically, Local Outlier Factors (LOF) seemed most appropriate, since they operate similarly to DBSCAN in that outliers are determined based on local neighbourhood density (which disregards the geometric shape of the data, unlike distance based techniques such as k -means), but it also considers a “local reachability” of one point from its neighbours, which means that the distance metrics used are asymmetrical, giving us more precise anomaly detection.

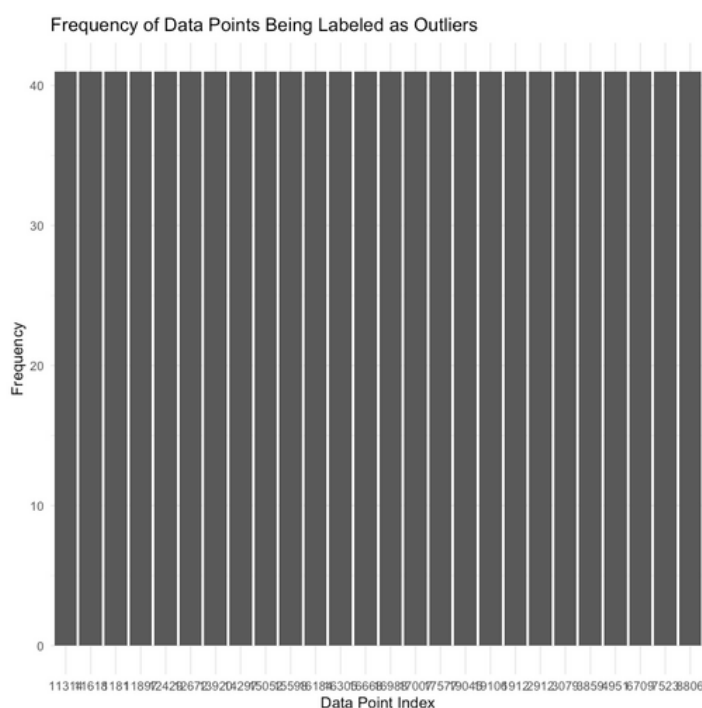
Since LOF is extremely sensitive to the hyper-parameters K (neighbourhood size), a reasonable heuristic range of values for $K \sim [10, 50]$ inclusive was selected and iterated over for further analysis. Below is the resulting plot for the analysis on a subset of 20,000 uniformly sampled rows.



Plot showing outliers found for each K (left) and number of K -values for which each outlier was identified (right).

EDA & Preparation

In the K -value plot (left), an “elbow” occurs at $K = 22$, where the number of outliers found is 57. When looking at the plot on the right, around 120 outliers are repeatedly identified for varying K values, with the a number of the most identified outliers being identified for all values of K . Using a confidence interval of 0.95, 26 outliers are identified, all occurring in 100% of cases. This yields a ~10 times higher proportion of outliers (0.13% as opposed to 0.015%) which, considering the skew of the data previously discussed, appears more consistent. The outliers identified using this method are presented in the bar and histogram plots on the following page. The minimum rate of occurrence is 100% using the previously stated confidence. Empirically, this tells us that, at the very least, our precision rate is upwards of 100%.



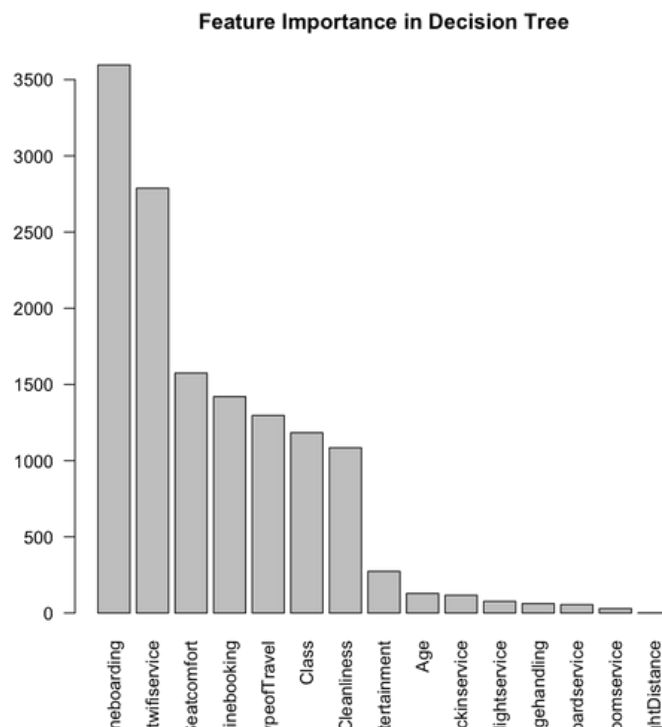
Plot showing the top 26 outliers, identified as outliers in all 41 iterations values of K .

The outliers that were identified were dealt with as detailed in section 4.

3.5 Feature Significance

Feature significance analysis using decision trees found “OnlineBoarding” to be most significant, while “Legroomservice” to be of least significance. However, despite this, online boarding exhibits one of the more significant negative skews, indicating that, while it definitely has room for improvement, perhaps great return on investment would be gained from improving another service. For example, the second most significant feature was “Inflightwifiservice”, which exhibits almost no skew - perhaps this should be an area of focus. The feature significance plot is shown below.

EDA & Preparation



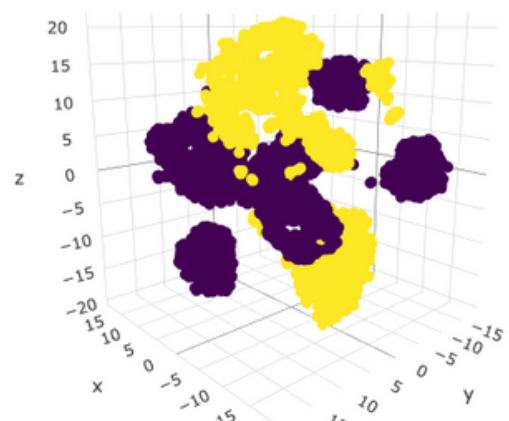
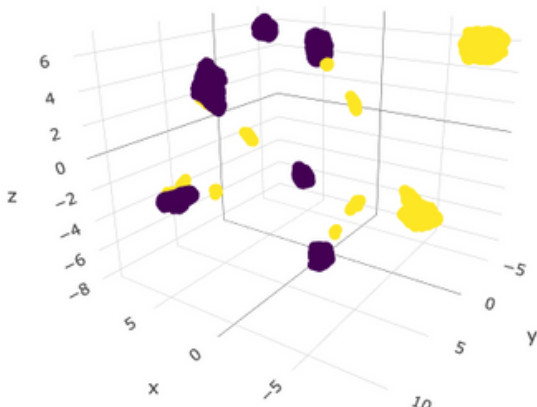
Feature significance plot of the nontrivially contributing features.

3.6 3D Visualisation

In our outlier analysis using LOF, a K value of 43 was used to identify the 22 outliers. Using this same value for parameters in a 3D Uniform Manifold Approximation & Projection (UMAP), $n_neighbours$, and a t-Distributed Stochastic Neighbour Embedding, $perplexity$, the following plots illustrate a uniformly sampled subset of the imputed data (due to resource constraints) of 20,000 samples. When coloured by the sample's Satisfaction value, there is extremely high separability.

3D UMAP Plot

3D tSNE Plot



20,000 uniformly sampled subset projected via UMAP (left) and tSNA (right) ito 3D, coloured by Satisfaction.

Data Preprocessing

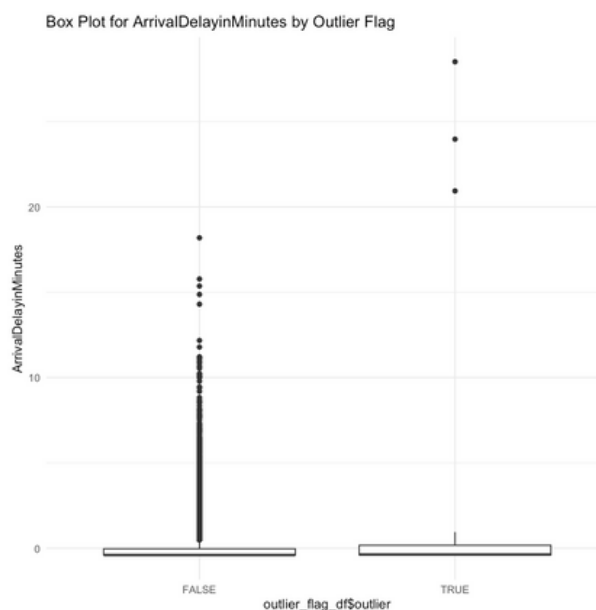
4.1 Feature Imputation

4.1.1 Missing Values

As previously mentioned, the missing values of “ArrivalDelayinMinutes”, referenced in section 3, were imputed via linear regression. The main feature used to make this prediction was “DepartureDelayinMinutes” due to its high collinearity (0.965 Pearson’s correlation coefficient) with the target variable. “FlightDistance”, “TypeofTravel” and “CustomerType” all seemed to improve the predictive performance of the model. The performance metric used was Coefficient of Determination (CoD, R-squared), and the model achieved a maximum CoD value of ~0.93. The model was then used to predict the missing values of “ArrivalDelayinMinutes”, which were then imputed using these values.

4.1.2 Outliers

After missing values were imputed, the data was standardised by z-score to preserve the intra-feature relationships and reduce the impact of the high skew of the delay-related features. The outliers were then analysed and it was shown to have detected the extremes of the continuous features. An example boxplot of the ArrivalDelayinMinutes feature illustrates this below.



Boxplot of ArrivalDelayinMinutes displaying outliers detected.

Since LOF detects sample-wise outliers rather than feature-wise outliers, these outliers will be removed. The resulting data will be compared against the data generated via standard imputation methods such as median imputation following from the chi-squared test for feature-wise anomalies, by comparing the performance of the classification models on the datasets.

Data Preprocessing

4.2 Dimensionality Reduction

Dimensionality reduction may be required (though not necessarily) due to the use of the 20,000 sample subset of the data, which reduces the sample-feature ratio to less than 1000. Due to the variety of the data being a mix of 1) continuous (time/distance features); 2) rank-ordered categorical (ratings features); 3) dichotomous binary categorical (gender, customer type, type of flight), correlation analysis is a significant and complex task, as a mix of correlation coefficients (Pearson's, Spearman's, Kendall's Tau), contribution measures (Cramer's V) and Analysis of Variation (ANOVA), among other techniques, would be required to gauge the full scope of relationships between features. A more straight forward approach is using an autoencoder to encode the features into a similar or lower dimensional latent space. This allows us to encode nonlinear relationships, as well as reduce all adequately encoded features down into a separable representation. Given that it's a continuous space, Pearson's or Spearman's coefficients can be used on the latent representation to analyse the correlation of the encoded features.

Another technique was previously already featured, which is UMAP and tSNE. As previously demonstrated, the projections produce highly separable embeddings of the data in even as low as 3 dimensions. This may indicate the significance of a few features dominating over the rest. Autoencoder latent space analysis is shown below, following Section 4.2.1 on Factor Analysis.

4.2.1 Factor Analysis of Delay Features

Factor analysis is a technique used to derive underlying "factors" that explain patterns or correlations among different variables in our dataset, and is particularly useful for variables that already seem to be interconnected by some latent feature/s. We can therefore use it to reduce the overall dimensionality of our data.

Standard methods of factor analysis assume that variables are continuous, therefore with our dataset, we are limited to only 4 variables where factor analysis can be applied: "Age", "FlightDistance", "DepartureDelayinMinutes", and "ArrivalDelayinMinutes". It also assumes that outliers have been removed, and that the data exhibits Gaussian statistical properties. As such, we applied the analysis after outlier removal, z-scale standardisation, and only on the delay-related fields - intuitively, there is a relationship that can be assumed between them, a priori. We also know from our correlation analysis that the delay fields share a Pearson's correlation of 0.965 and are therefore highly correlated with each other. Factor analysis works by identifying unobserved variables that influence the observed variables, and allows for the representation of high dimensional data in a lower dimensional space, with minimal information loss. After extracting these unobserved variables, known as factors, the factor scores are then calculated for each data point, which can be used in place of the original data point. In our case, we would then combine our arrival and departure delay fields into a single field.

After factor analysis is applied on these two fields, the results show us that these two fields share a commonality of 0.97, with the standardized loadings being 0.98. This confirms our previous highly correlative behaviour of these fields, and show that both variables are strongly correlated with this factor. The proportion of variance explained by this factor is also high at 0.97.

Data Preprocessing

	MR1	h2	u2	com
DepartureDelayinMinutes	0.98	0.97	0.034	1
ArrivalDelayinMinutes	0.98	0.97	0.034	1

Factor loadings, h2 (communality), u2(uniqueness) of the respective delay fields.

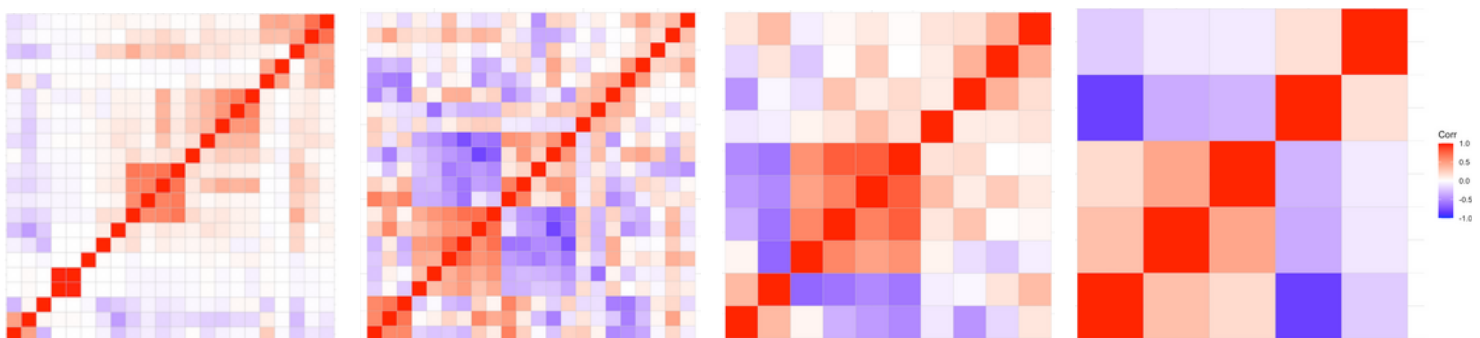
	MR1
SS loadings	1.93
Proportion Var	0.97

Sum of squared loadings, and proportion of variance explained by the factor.

Overall, these results underline the strong correlation of the two delay fields, and successfully explains most of the variance between them, meaning that we can reduce our overall dimensionality by 1. Due to the projection of our data via the factor analysis, the final factor scores would naturally lead to both positive and negative values, with very high variance. In order to equalise the order of magnitude of the feature again, we apply z-scale standardisation once more on these factor scores, in order for the data to be ready for our algorithms. Ultimately, the delay features were replaced with a single latent feature representing them, reducing the number of features down from 22 to 21.

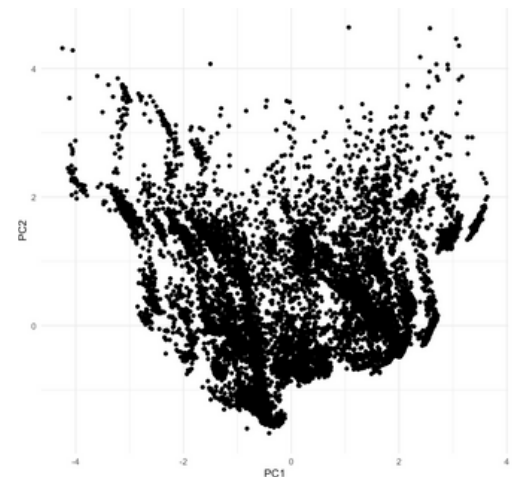
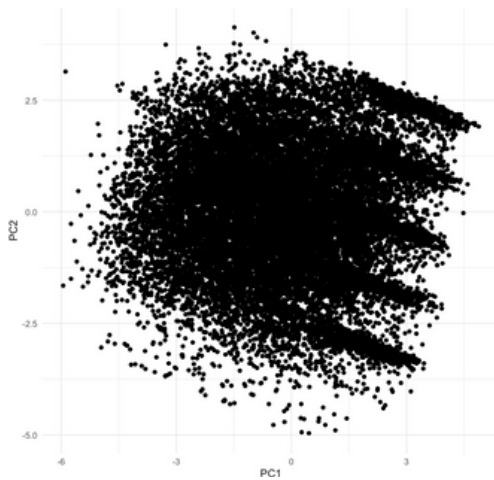
4.2.2 Autoencoder

Autoencoder embedding proved ineffective due to the difficulty in minimising the reconstruction error. 3 autoencoder architectures were applied, each with layers of varying numbers of neurons depending on the size of the latent space that was being encoded to. The latent space sizes were 21 (a re-embedding of the same dimensionality), 10, and 5. Each layer used batch normalisation and ReLu activation functions, with the output layer of the decoder being a sigmoid activation function, and Mean Square Error was used as the loss function. Of the 3 models, of course, the one with the latent space of 21 neurons had the lowest MSE output on both training and test sets. Despite this, all encodings seemed to generate embeddings with increased Pearson's correlation coefficients when compared to a (purely **illustrative**) correlation matrix of the original data, rendering them useless. The following plots depict their correlation and visualisations of their first 2 principle components. Plots of proportion of variance explained is included only for the data encoded by the 10-neuron autoencoder for illustrative purposes, as they were not useful



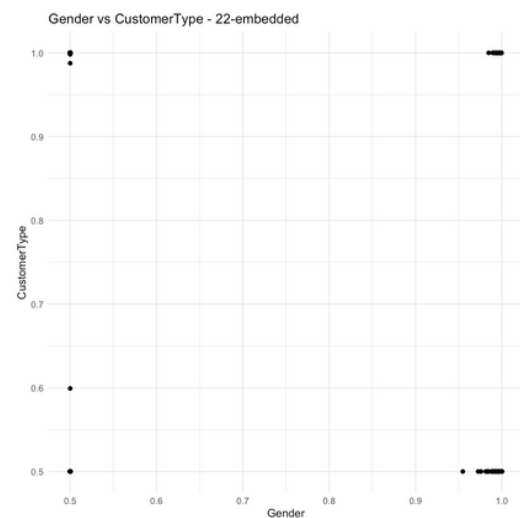
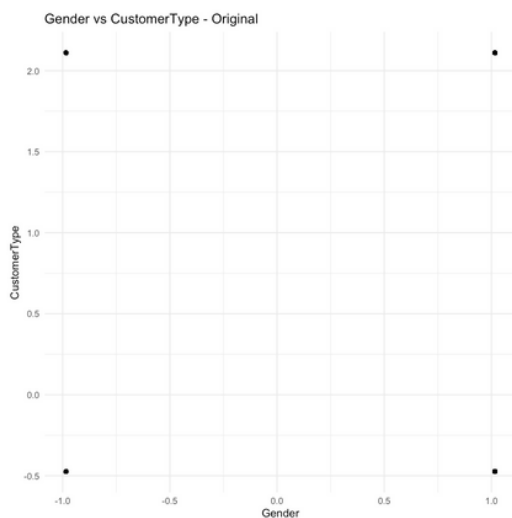
Pearson's correlation heatmaps of original (left), 21 (second), 10 (third), 5 (right) embedded feature embeddings.

Data Preprocessing



Principal Component 1 and 2 of the original (left) data and the 21-dimension encoded (right) data.

The following plots of the first 2 features of the original data (left) and the 21-dimension encoded data (right) show that there is a nontrivial reconstruction error that is not addressed, despite no reduction in the latent space.



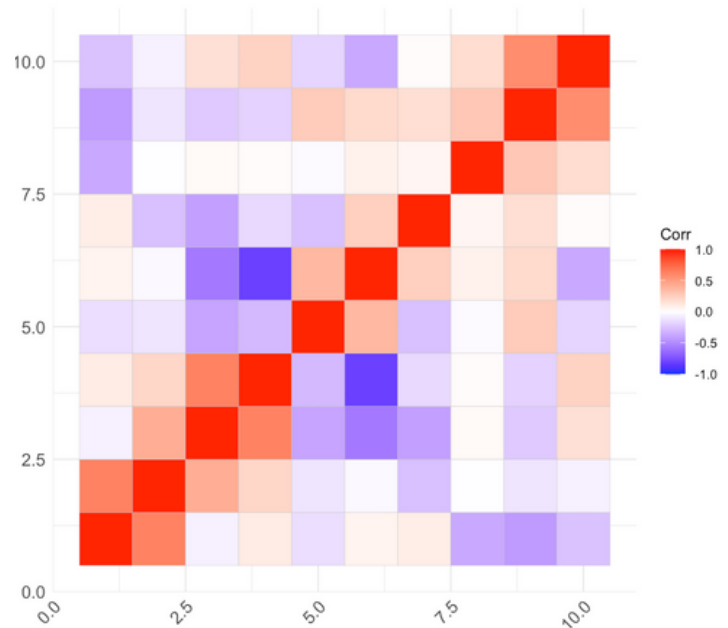
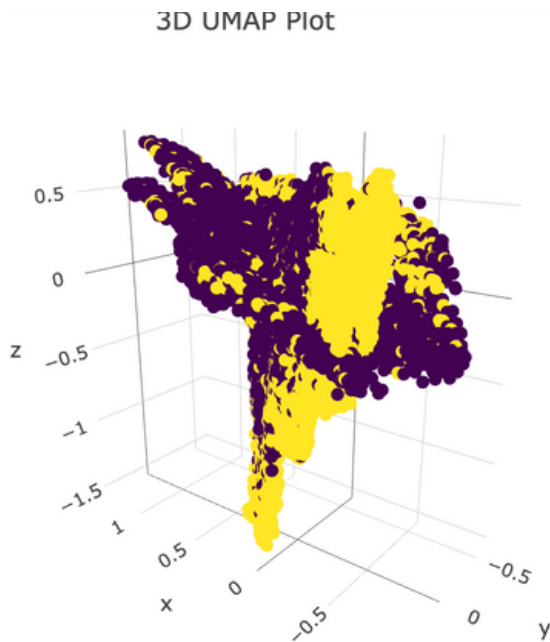
Gender-CustomerType plot of original (left) data and 21-dimension encoded (right) data.

Despite both variables being binary categorical, the plot on the right shows characteristics of continuity in both Gender and CustomerType features, that is, variance about one of the gender values, as well as variance about one of the CustomerType values. Perhaps further extensive investigation could yield better results, but this is out of the scope of this project, so we proceed with UMAP.

4.2.3 UMAP Reduction

The following plots show the Pearson's correlation heatmap for the reduction of the original data to 10 dimensions using the same parameters as previously stated.

Data Preprocessing



3D plot of first 3 components of 10D UMAP reduction coloured by satisfaction (left), and correlation heatmap (right).

In comparison to the correlation heatmap produced by the autoencoder encoding the data into 10D, this one shows more feature-wise independence in the data. A comparison of all techniques will be required to test the efficacy of the three data set - the original, the auto-encoder reduced 10D data, and the UMAP reduced 10D data. An outline of their features and processing are detailed in the table below.

Dataset	Features	Processing	Scaling	Usage
Original	All original features - Delay Features Substituted (21 Features)	LOF Outlier Removal LinReg Imputation Scaling Factor Substitution	Z-scale Standardisation	All Models
Encoded10	Autoencoder Latent Space (10 Dimensions)	LinReg Imputation LOF Outlier Removal Scaling Factor Substitution Dim. Reduction	Z-scale Standardisation	SVM, Logistic Regression
UMAP10	UMAP Projection (10 Components)	LinReg Imputation LOF Outlier Removal Scaling Factor Substitution Dim. Reduction	Z-scale Standardisation	SVM, Logistic Regression

Table summary of the 3 datasets compared.

Modelling & Metrics

5.1 Deep Neural Network - Ali

5.1.1 Architecture

Three different model architectures were tested - 1 for the original data of 21 features, and 2 for the autoencoder and UMAP embedded data of 10 features each. The model for the original data consisted of 6 layers, batch normalisation, ReLU activation, a dropout layer with 30% probability, and a sigmoid activation acting on a single output neuron at the output layer. The only change made to the other 2 models is that the number of layers was reduced from 6 to 5 and 4 respectively to reduce model complexity and reduce overfitting.

The loss function used was Binary Cross-entropy Loss, in order to not only train for accuracy, but also train for confidence (probability values) through use of the log-likelihood function which penalises high confidence incorrect predictions by assigning high loss values to the output probability.

5.1.2 Data Preparation & Regularisation

Prior to input, the data was split into stratified train and tests subsets. Both subsets were standardised using the z-scale metrics of the train subset to avoid data leakage. The train subset was then split again used stratified k-fold cross-validation, with 5 validation folds, in order to further prevent overfitting. This new train-validation split was also standardised using the z-scale metrics of this new train set, to prevent data leakage. All partitioning occurred with an 80/20 split.

Both dropout and k-fold cross-validation aided in preventing overfitting. Unfortunately, due to resource constraints (Tensorflow/Keras incompatibility with Silicon Macs meant using PyTorch, which has no GridSearchCV implementation), hyperparameter tuning was not available via Grid Search. Instead, empirical testing was used to obtain optimal hyper parameters.

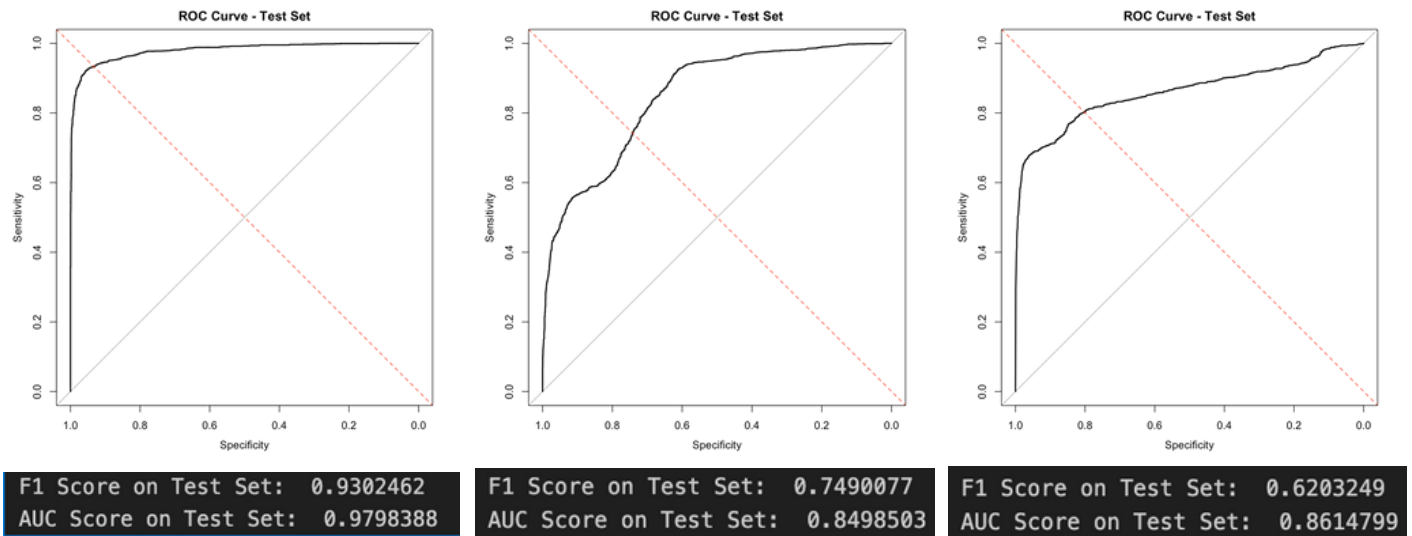
5.1.3 Results & Performance Metrics

Needless to say, the model performing on the original feature data, with 21 features, performed the best by far. It achieved a test accuracy of 94.05%. Ultimately, only the 2nd model architecture was used for both the UMAP embedded data and the Autoencoder embedded data. It achieved 49.75% and 73.13% accuracy, respectively. The rest of the performance metrics are visualised below.

The data shows that the model which classified the original feature data is exceptionally performant, given that the ROC-AUC is ~ 0.98 which shows that it has an extremely low false positive rate and very high sensitivity. Coupled with its 94.05% accuracy on the test set, there is strong indication that it is almost ideal. The F1-score of ~ 0.93 shows that it has exceptional ability to differentiate between classes while also remaining sensitive to them.

The curves of the other 2 datasets are interesting; in the case of the UMAP data, which performed the worst in terms of accuracy at 49.75%, it still maintained a relatively high AUC of ~ 0.86 . This seems to conflict with its harmonic mean of ~ 0.62 and may be worth investigating.

Modelling & Metrics



F1 score and AUC for the original feature data (left), Autencoder data (middle), and UMAP data (right).

The Autoencoder dataset exhibits a similar AUC as the UMAP data, at ~0.85, but as expected, its F1-score is higher, at ~0.75, matching its accuracy of 73.13%. Perhaps further investigation into the encoding process, or encoding to a high dimensional space, like 15D, may produce better results as it reduces the reconstruction error and distorts the statistical properties of the data much less.

5.2 Support Vector Machine - Ali

5.2.1 Data Preparation & Regularisation

The data was prepared identically to that which was prepared for DNN classification.

5.2.2 Architecture

Empirical testing showed a Radial Basis Function/Gaussian kernel to perform best. Default cost & gamma values were used, where cost was 1 and gamma was $1/(\text{dimension of data} = 23)$. The cost of 1 was appropriate due to the observed ease of separability of the data, which mean hard margins were suitable to use and ensure high accuracy by more strongly penalising misclassified datapoints. The default gamma value of $1/23$ seemed to generate sufficiently maximised results, and did not seem to have significantly high influence on the performance when it took small values within the range of the default.

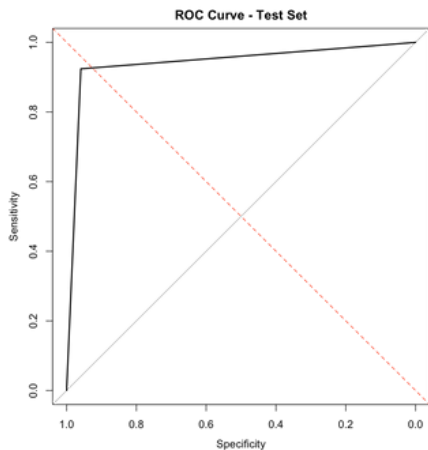
5.2.3 Results & Performance Metrics

Given the significant difference in performance of the DNN between datasets, only the original features were used for the SVM. The SVM outperformed the DNN, with a test accuracy of 94.35%. Interestingly, the F1 score was severely lowered despite its marginal improvement in accuracy. The plots below show the key performance metrics.

The high ROC-AUC score may be due to the sizeable imbalance between positive and negative instances of the satisfaction class, with the positive being almost 1.5x times the negative. Naturally, the True Negative rate of the classifier would grow with this skew, whereas the True Positive may not.

Modelling & Metrics

A high True Negative rate, as opposed to a high True Positive rate, could mask the underperformance of the classifier and artificially inflate the AUC value. In this case, F1 may be the differentiator.



Actual		
Predicted	0	1
0	2134	135
1	91	1640

F1 Score on Test Set: 0.1123128

AUC Score on Test Set: 0.9415224

ROC for the original data (left), F1 & AUC (lower), and Confusion Matrix (right) for SVM on original features..

5.3 Conclusion for DNN vs SVM - Ali

This data shows that, while the SVM was more performant in the accuracy metric, it was only marginally better. On the other hand, the combination of its significantly worse performance in the F1 metric, as well as its marginally worse AUC value, indicate that perhaps the DNN is a better choice over it for classification. However, there is further scope for investigation with grid search in the Autoencoder, the DNN and SVM models in order to extract optimal feature representations and performance. Further, UMAP and tSNE proved to be ineffective techniques for dimensionality reduction. UMAP was slightly more suitable given the mathematical logic of tSNE being stochastic projections onto less dimensions using t-distributions - in the case of UMAP, the gradient descent over its cost function (similarity differences) provides a mathematically more stable and consistent result for embedding, though trivially so.

5.4 Logistic Regression - Adam

5.4.1 Data Preparation

For the auto-encoded and UMAP datasets, both were needed to be transformed into a data frame from a matrix and umap object respectively. They were both also missing the satisfaction field, and needed it to be attached to pass into the logistic regression algorithm. From here, all datasets were then split into train and test datasets with the 80/20 split.

5.4.2 Regularisation

With logistic regression, the main way to prevent overfitting is by tuning the regularisers and their strength. Logistic regression allows the usage of the L1 (ridge regression), L2 (lasso regression), and a combination of L1 and L2 (elasticnet) regularisers.

Modelling & Metrics

This mix is controlled by the alpha variable where $\alpha = 0$ is only using L1, $\alpha = 1$ is only using L2, and anything in between 0-1 is a mix. In my model, I only include L2 regularisation as when experimenting with different alpha values from 0-1, $\alpha=1$ performed best which means complete L2 regularisation. The next variable is lambda which controls the strength of the regularisers, by applying a penalty to the coefficients of the model, where a larger value imposes a larger penalty, where it reduces the coefficients towards zero. For this, I used a cross validation method where I iteratively test out 100 lambda values between 0.01 and 0.0001. The best lambda value would then be used on the final model where results will be compared.

5.4.3 Other Model Parameters

The other parameter changed in the models were the family parameter which specifies the type of model to be fitted on logistic regression. For the context of our dataset, the predicted value is satisfaction with 0/1 which is a binary option. Therefore, the logistic regression model is a binary model, and the family parameter was therefore set to binomial. Additionally, the chosen threshold that determines the overall output of the logistic regression model was chosen to be 0.5. These values were empirically tested at 0.5, 0.6, 0.7, 0.8, and the threshold of 0.5 performed significantly better than the others, and thus was chosen for the final models threshold.

5.4.4 Results

Surprisingly, all datasets performed relatively similar on logistic regression. Despite EDA showing us that the 21 dimensions used in our original dataset were not redundant and were not highly correlated, logistic regression proved to be versatile on all 3 datasets

F1 Score on Test Set: 0.8586

F1 Score on Test Set: 0.7668

F1 Score on Test Set: 0.8291

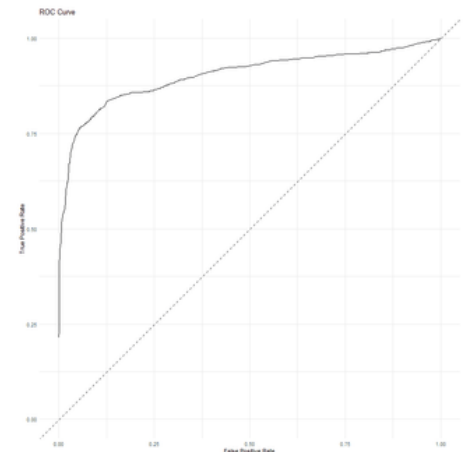
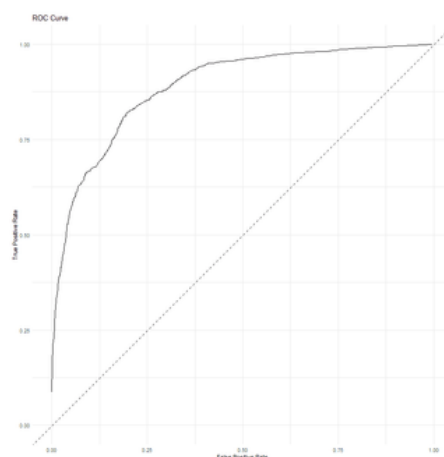
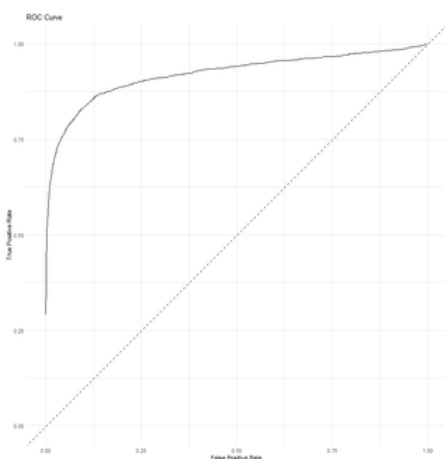
F1 scores for the original data (left), Autoencoder data (middle), and UMAP data (right)

AUC Score on Test Set: 0.930
Accuracy: 0.88275"

AUC Score on Test Set: 0.9022
Accuracy: 0.8615"

AUC Score on Test Set: 0.889
Accuracy: 0.80225"

AUC and accuracy scores for the original data (left), Autoencoder data (middle), and UMAP data (right)



ROC for the original data (left), Autoencoder data (middle), and UMAP data (right)

Modelling & Metrics

The test accuracies were roughly 88.3%, 80.2%, and 86.2% respectively for the original, autoencoder, and umap data. The F1 scores followed the same pattern with 0.859, 0.767, and 0.829 respectively. Although the AUC scores also followed the same pattern with 0.930, 0.889, and 0.902 respectively, autoencoder showed a big jump and nearly matched UMAP's AUC score. This might suggest that although the original and UMAP datasets performed better when the probability threshold was manually set at 0.5, the autoencoder dataset performs slightly better at distinguishing satisfied and unsatisfied classes over the range of all possible probability thresholds. This might indicate that there is a chance that there exists a better threshold that could make the autoencoder data perform better. On the other hand the AUC score could potentially be skewed due to the imbalance between classes. Specifically, an abundance of negative cases could make AUC appear optimistic as the model is simply good at identifying the negative cases. This applies to our data where 57% of customer responses were neutral or dissatisfied.

predicted_classes		
	0	1
0	2107	202
1	267	1424

predicted_classes		
	0	1
0	1908	387
1	404	1301

predicted_classes		
	0	1
0	2102	191
1	363	1344

Confusion matrices for the original data (left), Autoencoder data (middle), and UMAP data (right)

There is a decline in F1 score and accuracy between the original data model and the reduced dimensionality models, which is expected due to the loss of information occurred during dimensionality reduction. However, this does still demonstrate that the most important features were still preserved the UMAP representation, as the decline was only very slight at ~2%. Autoencoder still performed decently, trailing by ~6% behind UMAP, demonstrating that UMAP's dimensionality reduction slightly outperformed the autoencoder. The time taken (including testing 100 lambda values) for each dataset was 9.36s, 7.11s, and 7.17s respectively.

Optimal lambda: 3e-04

Optimal lambda: 1e-04

Optimal lambda: 1e-04

Optimal lambda values for the original data (left), Autoencoder data (middle), and UMAP data (right)

The optimal lambda values of 0.0003, 0.0001, and 0.0001 for the models respectively, indicate that the umap and autoencoder data did not overfit as much as the original dataset, which is to be expected due to the reduced dimensionality. performed best on a significantly higher regulariser strength compared to the umap data. Overall, all lambda values were relatively low meaning that overfitting was not an issue for the logistic regression model. It is interesting to see that the autoencoder and UMAP's lambda value were the exact same despite the slight gap in performance. This suggests that although the autoencoder and UMAP helps prevent overfitting, UMAP is superior than autoencoder at dimensionality reduction within the context of logistic regression.

Overall, these results illustrate logistic regression's robustness when modeling with different dimensionality of data. It is still able to deliver results even with a reduction in dimensionality. However, due to its efficiency, a dataset with larger dimensions could still be utilized, for better performance.

5.5 Random Forest - Adam

5.5.1 Data Preparation

The data was prepared in the same way as logistic regression.

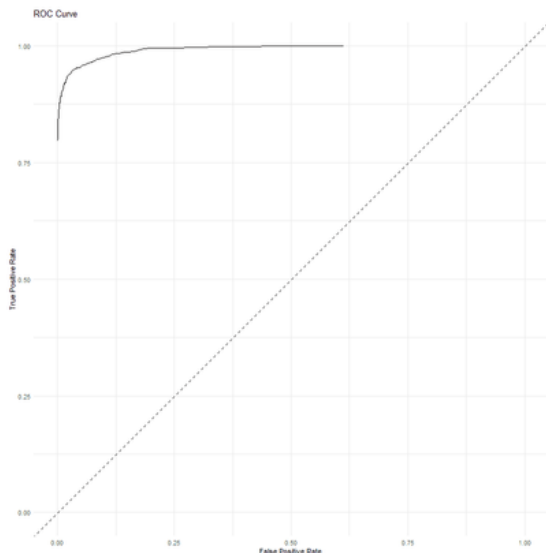
5.5.2 Random Forest Size

With random forest, the main parameter to be tuned is the forest size. Forest sizes of 50, 100, 200, 300, 400, 500 trees were all empirically tested and the one that performed the best was 300.

Modelling & Metrics

5.5.3 Results

The results indicate that random forest performed better than logistic regression, with test accuracy of 95.8%, F1 score of 0.964 and AUC score of 0.992. The time taken to run the model was at 8.23s.



ROC for random forest

```
F1 Score on Test Set: 0.9638
AUC Score on Test Set: 0.9923
Accuracy on Test Set: 0.95825
```

F1 score, AUC score, and accuracy for random forest

```
      0      1
0 8762 301
1 495 6442
```

Confusion matrix for random forest

Anything below 300 trees resulted in a slight decrease in performance (94.7% accuracy vs 95.8% accuracy for 50 trees vs 300 trees). Anything greater than 300 resulted in a stagnant performance at the cost of efficiency. For example, 300 trees took 8.23s to run whereas 500 trees took 13.57s. This is despite the performance of both trees being extremely similar to each other, with accuracies of 95.8% and 95.7% respectively.

Interestingly, the very high AUC score of 0.992 suggests that random forest does extremely well in separability in distinguishing between the positive and negative classes. However, like with logistic regression this metric may be skewed due to our slight class imbalance. As expected, both accuracies and f1 scores were also higher than logistic regression. This demonstrates that the model successfully learnt from the imbalanced data, and maintains a competitive performance.

5.6 Conclusion for LR vs RF - Adam

Overall, the results unsurprisingly show that random forest outperformed logistic regression in every metric. Accuracy improved over 7% from 88.3% to 95.8%, F1 score also improved greatly from 0.859 to 0.964, and AUC score slightly improved from 0.930 to 0.992. However, random forest did take a lot longer than logistic regression. With 300 trees, random forest took 8.23s to run compared to logistic regression which took 0.0927s for a single run. However, tests do show that a decrease in trees can significantly increase the efficiency of random forest, whilst only sacrificing a bit of performance. Due to the imbalanced data, the most important metrics to consider are accuracy and F1 score. Between the two, the better choice is clearly random forest. The trade off between performance and efficiency, by going from random forest to logistic regression is not worth it due to the simplicity of logistic regression. Logistic regression should only be used as a baseline model and cannot compete with other complex models.

Further investigations could include seeing how random forest would perform on the reduced dimensions datasets. It would be interesting to see how a more complex model would perform when given simpler data.

Modelling & Metrics

5.7 K-nearest neighbors - Paula

5.7.1 Data Preparation

The dataset was loaded and partitioned into training and testing sets with a 70/30 split. The satisfaction variable was ensured to have valid factor levels.

5.7.2 Regularisation

Regularisation in the kNN model involves adjusting the hyperparameter 'k.' The value of 'k' (set to 15 in this case) influences the number of neighbours considered during prediction. K is set as an arbitrary selection but to determine the optimal 'k' value for your specific dataset, I had to explore a range of values and use cross-validation to evaluate their performance. Adjusting 'k' and observing changes in model accuracy, precision, recall, or F1 score.

5.7.3 Results

The kNN model, trained on customer satisfaction data, exhibits commendable performance across various evaluation metrics. The overall accuracy of the model is 91.2%, implying that it correctly classifies a substantial portion of instances. Precision, representing the positive predictive value, is at 89.1%, indicating the reliability of the model in predicting positive instances. Notably, the model demonstrates a high sensitivity of 96.0%, showcasing its efficacy in accurately identifying satisfied customers. Specificity, denoting the true negative rate, stands at 85.1%, reflecting the model's ability to appropriately identify instances of dissatisfaction. The harmonized F1 score, a balance between precision and recall, is impressive at 94.4%, emphasizing the robustness of the model's predictive capabilities.

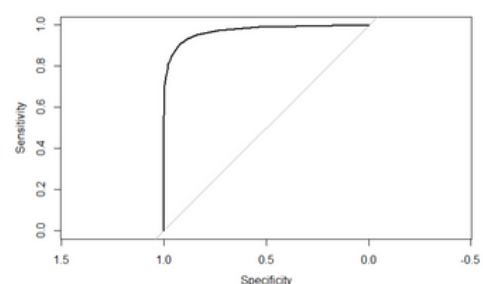
The Area Under the Receiver Operating Characteristic (ROC) Curve, a metric often used to evaluate binary classifiers, yields a score of 97.1%. This high AUC score signifies excellent discrimination ability, indicating the model's proficiency in distinguishing between satisfied and unsatisfied customers. The Kappa statistic, measuring agreement between predicted and actual classes, is noteworthy at 81.97%. This suggests a substantial agreement beyond random chance. Furthermore, the balanced accuracy, accounting for sensitivity and specificity, stands at 90.56%, providing a comprehensive perspective on the model's overall performance. Examining prevalence and detection rates, it is observed that the prevalence of positive instances is 55.85%. The detection rate, indicating the proportion of correctly identified instances, is at 53.63%. McNemar's Test, assessing the significance of differences between predicted and actual classes, yields a p-value less than 2.2×10^{-16} , signifying a significant distinction.

```
      Reference
Prediction  0    1
0      3218  395
1      133  2254
```

Confusion Matrix for KNN

```
[1] "Accuracy: 0.912"
[1] "F1 Score: 0.944281524926686"
[1] "AUC Score: 0.970526481449"
```

F1 score, AUC score, and accuracy for KNN



ROC for KNN

Modelling & Metrics

5.8 Random Forest - Paula

5.8.1 Data Preparation

The data was prepared in the same way as K-nearest neighbour.

5.8.2 Random Forest Size

The tree size in a Random Forest, governed by the `ntree` parameter, plays a crucial role in determining the overall complexity of the ensemble model. A Random Forest is an ensemble of decision trees, and increasing the number of trees can lead to improved model performance up to a certain point. Numerous tree sizes were tested, and the results were compared. The size that achieved an optimal performance was 100.

5.8.3 Other Model Parameters

Additionally, the number of variables randomly sampled as candidates at each split in the decision tree construction is set to 3. By setting this hyperparameter, at each split, the algorithm considers a subset of three randomly chosen predictor variables to make a decision.

5.8.4 Results

The Random Forest model, trained on customer satisfaction data, delivers exceptional predictive performance across a spectrum of evaluation metrics, underscoring its robustness in discerning customer satisfaction. The overall accuracy of the model stands at an impressive 94.6%, reflecting a substantial proportion of accurately classified instances. Precision, indicating the positive predictive value, is notable at 89.1%, emphasizing the model's reliability in predicting instances of customer satisfaction.

In terms of sensitivity, the model exhibits a high rate of 96.1%, showcasing its proficiency in accurately identifying satisfied customers. Simultaneously, specificity, representing the true negative rate, is commendable at 92.1%, highlighting the model's effectiveness in appropriately identifying instances of dissatisfaction. The harmonized F1 score, a balanced measure of precision and recall, reaches an impressive 95%, affirming the model's robust predictive capabilities, particularly in managing imbalanced classes.

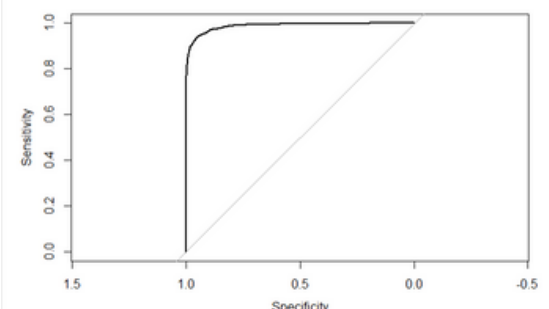
The Area Under the Receiver Operating Characteristic (ROC) Curve, a metric assessing discrimination ability, attains a noteworthy score of 98.8%. This indicates the model's excellence in distinguishing between satisfied and unsatisfied customers. The Kappa statistic, measuring agreement beyond chance, stands at 89%, underscoring substantial agreement between predicted and actual classes.

	Reference	
Prediction	0	1
0	3222	193
1	129	2456

Confusion Matrix for RF

```
[1] "Accuracy: 0.9463333333333333"
[1] "F1 Score: 0.95009671179884"
[1] "AUC Score: 0.988322930371635"
```

F1 score, AUC score, and accuracy for RF



ROC for RF

Modelling & Metrics

5.9 Conclusion for KNN and RF - Paula

In comparing the Random Forest and kNN models for predicting customer satisfaction, both demonstrate commendable performance. The Random Forest model achieves a high overall accuracy of 94.6%, with superior precision, sensitivity, and specificity. It boasts a remarkable F1 score of 95%, AUC of 98.8%, and Kappa of 89%, indicating exceptional predictive capabilities and agreement between predicted and actual classes. On the other hand, the kNN model exhibits notable accuracy (91.2%), sensitivity, and precision, with an AUC of 97.1% and Kappa of 81.97%. While performing well, it falls slightly short of the Random Forest model in various metrics.

Comparing the two models, the Random Forest model stands out with higher overall accuracy, precision, specificity, and discrimination ability. The exceptional AUC score and Kappa statistic further indicate its superiority in agreement between predicted and actual classes. While the kNN model performance is notable, the Random Forest model's higher metrics suggest it may be better suited for predicting customer satisfaction in this context. The nuanced strengths of the Random Forest model, especially in handling imbalanced classes and achieving high precision, contribute to its overall superiority in this predictive task.

6. Comparison of Models

Given the business problem statement, part of which involves gaining an understanding of the key drivers of satisfaction, the areas of interest, and the areas that need attention, a need for an explainable model arises, or at least one which provides this breakdown of components. Clearly, complex black-boxes such as DNNs are out of the question in this sense. Further, the DNN was clearly not a top performer, which makes this a relatively easy decision to make.

On the topic of performance, the clearly superior model is that of Random Forest, with 95.8% accuracy and, crucially, excellent F1 score of 0.964, and almost flawless AUC at 0.992. The performance increase over non-tree-based models may have come due to the relatively simple relationships between the predictors and the target, and thus its high separability and potentially high entropy. The Logistic Regression, KNN & SVM models, while strong performers, did underperform in comparison, possibly due to their increased reliance on separability and distance based metrics, as opposed to the Tree-based models' sensitivity to entropy.

Ultimately, with respect to hardware and resource limitations, the business would benefit from the relative simplicity of the Tree-based models, which require less compute power and no parallelisation. All things considered, Random Forest is the best usage with no compromises in any facet.

Recommendations

7. Recommendation to Senior Management

Recommendations to senior management on the technical process include the following:

- Further investigation into feature importance and factor analysis to identify consistently highly contributing features;
- Further investigation into hyper parameter tuning for even more accurate models using grid search, and for the derivation of a new, more concentrated feature space;

Recommendations to senior management on the business problem of understanding and improving the factors driving customer satisfaction includes focusing on the features with both the most positive skew and simultaneously highest importance. In particular:

- Inflight wifi service was identified as the 2nd most important factor, yet it was the only feature with positive skew. The belief is that efficient allocation of capital and resources includes investing in and improving the inflight wifi service, as it is likely to generate the great improvement in satisfaction for given unit of currency or other resource.
- Ease of online booking was identified as the 4th most important feature, with approximately similar importance to inflight wifi service, yet it has the 2nd most positive skew (or least negative, at -0.2). Similarly, investing in this **after** the inflight wifi service would be the best use of company capital and resources. Improvements are dependant on further qualitative analysis of the exact issues and frustrations faced by the customers.

To improve overall customer satisfaction, the airline could also focus on enhancing the comfort and making departure and arrival times more convenient for travellers.

Lastly, it seems that the majority of neutrality or dissatisfaction seems to be approximately equally distributed (save for the 2 most important features mentioned earlier) among the rest of the services. Further, analysis of the distributions of flight-type against satisfaction showed that business-class customers seemed to be least neutral or dissatisfied, intuitively. Economy customers were the most neutral or dissatisfied, closely followed by Economy Plus, indicating that allocation of resources should be focused in the Economy flight class. Given that the vast majority of customers fly Economy, this would likely generate the greatest return on investment, in terms of profit, customer satisfaction rating and likely competitive advantage index.

Contributions

Contributing Member	Original Contributions - Sections	Original Contributions - Visualisations	Contributing Edits - Sections	Contributing Edits - Visualisations
Ali	1*, 2*, 3.1*-3.2, 3.4*-3.6, 4.1*-4.2, 4.2.2-4.2.3, 5.1*-5.3, 6, 7	1*, 2*, 3.1*-3.2, 3.4*-3.6, 4.1*-4.2, 4.2.2-4.2.3, 5.1*-5.3	4.2.1, 3.3	N/A
Adam	4.2.1, 5.4*, 5.5*, 5.6*	4.2.1, 5.4*, 5.5*, 5.6*	N/A	N/A
Paula	3.3, 5.7*, 5.8*, 5.9*	3.3, 5.7*, 5.8*, 5.9*	7	N/A
Arshia	N/C	N/C	N/C	N/C

N/C : No contribution

* : Including all subsections

Statement of Contributions

All members agreed to the table above detailing contributions. However some members request a meeting with the module professor to go over the nuances.

The group agrees that Arshia did not contribute at all, and was largely absent.

Adam and Paula believe that their grades should not be balanced and that all rebalancing should come from reducing Arshia's grade to boost Ali's grade.

Ali does not agree and has provided a statement detailing the contributions.