Machine Learning Airline Satisfaction Dataset

13th January 2022/23

Table of Contents

Table of Figures

Table of Tables

1 Introduction

This report will go into detail of an analysis and the method of an airline satisfaction classification problem. The problem is to be able to identify based on a set of features whether it is dissatisfied, neutral or satisfied feedback from an airline customer. This report goes over the steps taken to analyse the dataset and create a method that will reach the goal of predicting the satisfaction of airline passengers.

2 Exploratory Data Analysis

This section of the report will go over the analysis of the dataset this will include descriptions about the features, class imbalances, missing data, noise, outliers, duplicates, cleaning that may need to be done and a summary of the dataset and what we can take from all the information (linking it to the real world).

2.1 Feature descriptions

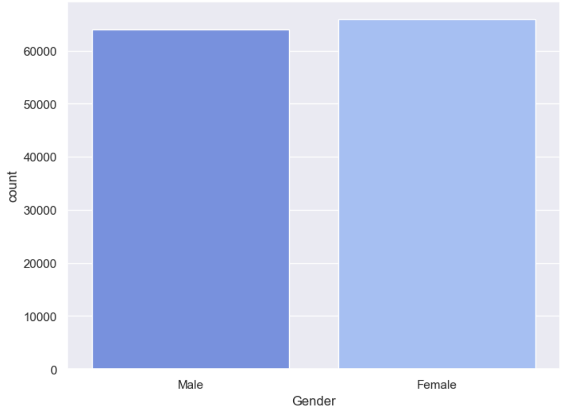
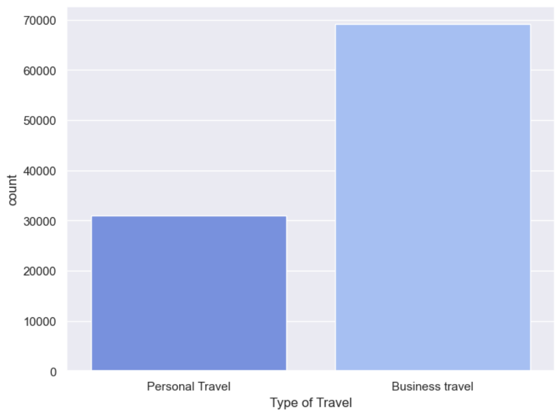
Table 1: Table of Feature description

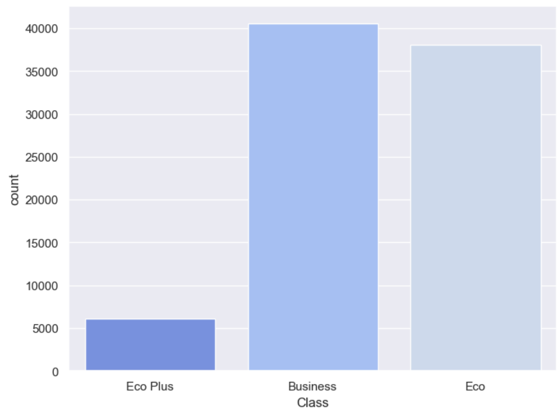
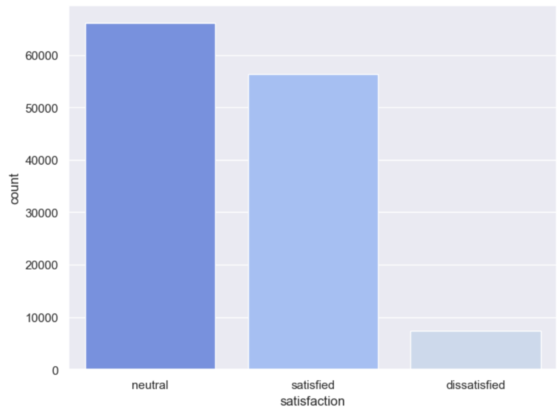
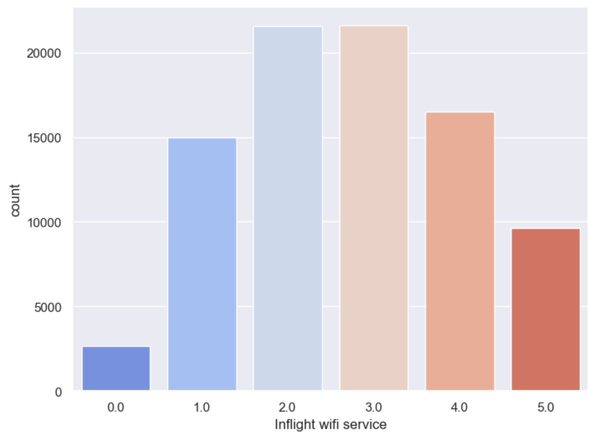
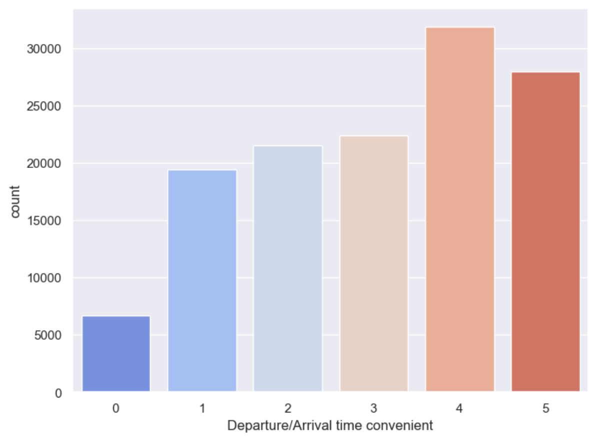
Feature	Description	Data Type
ID	A unique ID for each record	Int64
Gender	A Male or Female category	Object
Customer Type	Loyal Customer or Disloyal Customer category	Object
Age	A range of numbers from 7 to 85	Int64
Type of Travel	Personal or Business travel category	Object
Class	Business, Eco+ and Eco category	Objects
Flight Distance	Distance the flight flew Ranging from 31 to 4983	Int64
Inflight wifi Service	User rating of plane wifi 1-5 category	Float64
Departure/Arrival time convenient	User rating of the plane departure and landing time was convenient 1-5 category	Int64
Ease of Online booking	User rating of how easy it was to book online	Int64
Gate location	User rating of the gate location (easy to find)	Int64
Food and drink	User rating of the food and drink 1 to 5	Int64
Online boarding	User rating of online boarding 1 to 5	Float64
Seat comfort	User rating of how comfortable the seat was 1 to 5	Int64
Inflight entertainment	User rating of how good the entertainment was 1 to 5	Float64
On-board service	User rating of how the on-board service was 1 to 5	Int64
Leg room service	User rating of how much leg room they got 1 to 5	Int64
Baggage handling	User rating of how their baggage was handled 1 to 5	Int64
Checkin service	User rating of their check-in experience 1 to 5	Int64
Inflight service	User rating of how the inflight service was 1 to 5	Int64
Cleanliness	User rating of how clean it was 1 to 5	Int64
Departure Delay in Minutes	A range of value for the departure delay	Float64
Arrival Delay in Minutes	A range of value for the Arrival delay	Float64
Satisfaction	A categorical option for satisfied, neutral and dissatisfied	Object

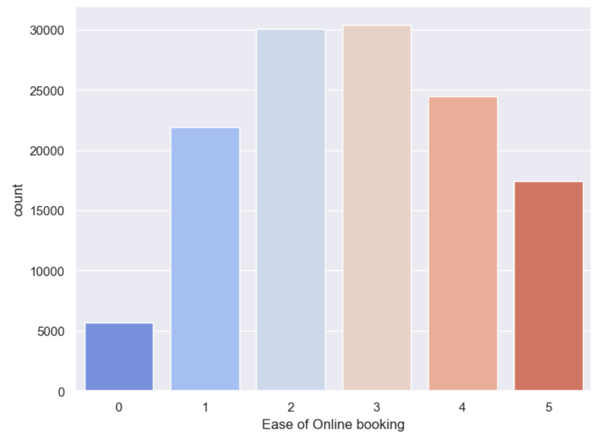
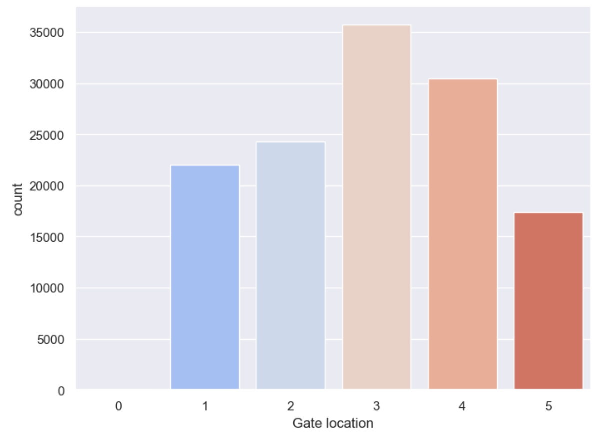
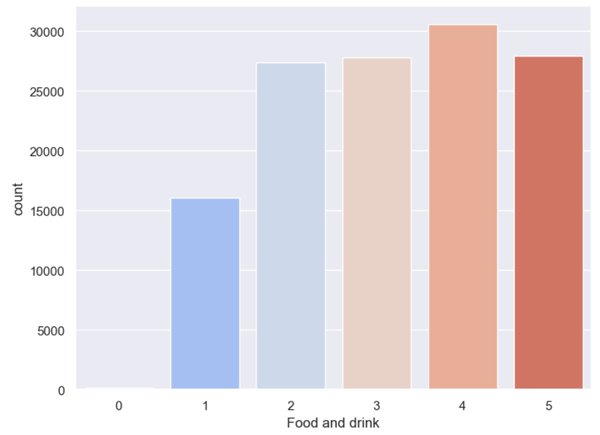
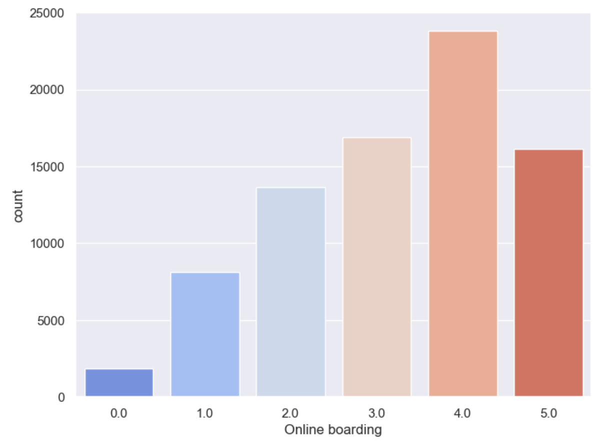
2.2 Key Values

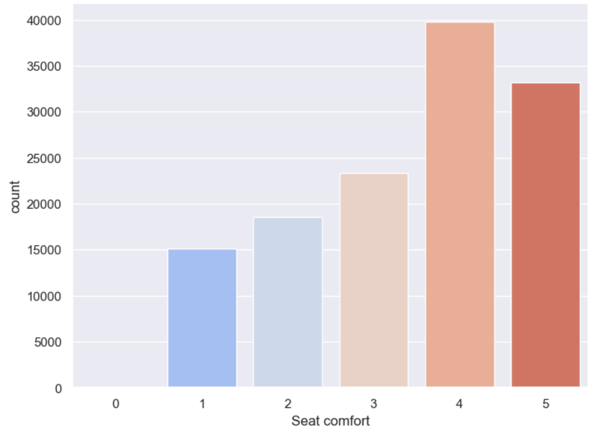
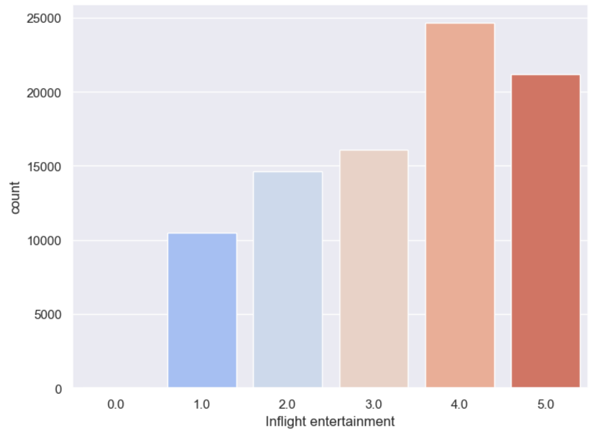
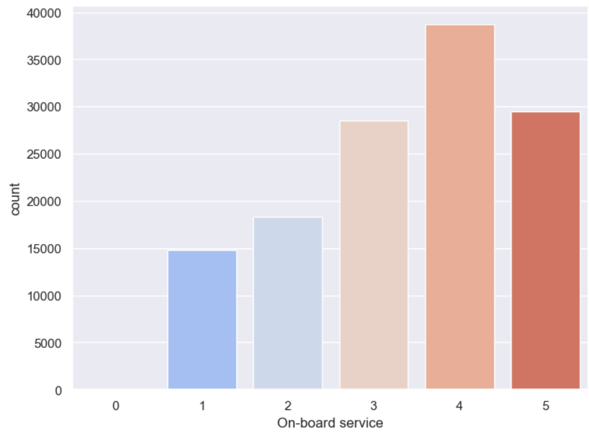
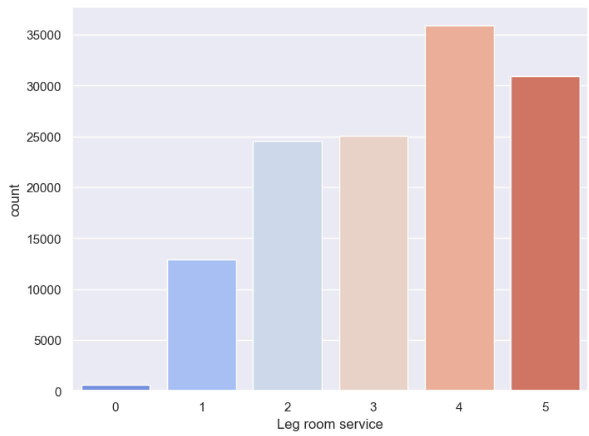
In this dataset there are two types of values categorical values and real values (continues values). This section will look at the values, their class balance, their central tendencies and their variability.

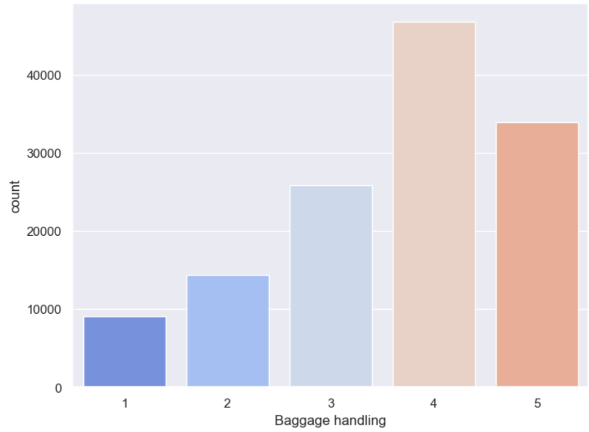
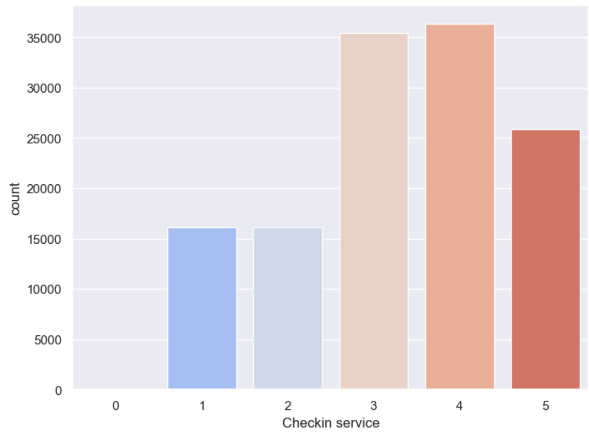
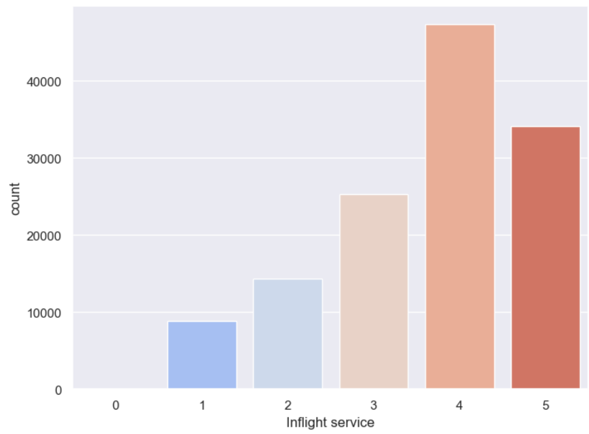
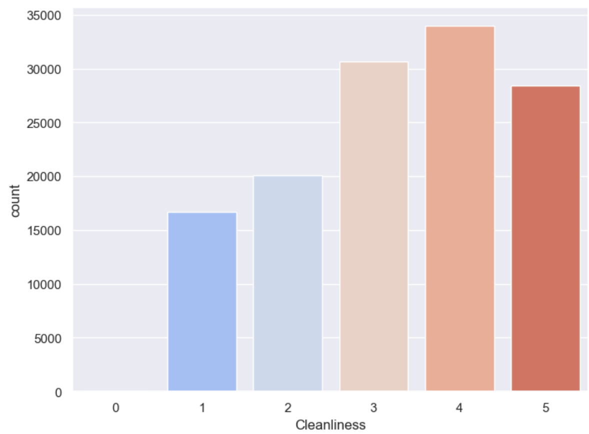
Table 2: Table of categorical features

Feature	Distribution	Count	Central tendencies
Gender		129,880	Mode: Female
Customer Type		129,880	Mode: Loyal Customer
Type of Travel		100,154	Mode: Personal Travel

Class		84727	Mode: Business
Satisfaction		129880	Mode: neutral
Inflight Wifi Service		86,909	Mode: 3.0
Departure/Arrival time convenient		129,880	Mode: 4

Ease of Online booking		129,880	Mode: 3
Gate location		129,880	Mode: 3
Food and drink		129,880	Mode: 4
Online boarding		80,441	Mode: 4

<table>
<tr><td>Seat comfort</td><td>
</td><td>129,880</td><td>Mode:
4</td></tr>
<tr><td>Inflight entertainment</td><td>
</td><td>86,996</td><td>Mode:
4</td></tr>
<tr><td>On-Board service</td><td>
</td><td>129,880</td><td>Mode:
4</td></tr>
<tr><td>Leg room service</td><td>
</td><td>129,880</td><td>Mode:
4</td></tr>
</table>

Baggage handling		129,880	Mode: 4
Checkin service		129,880	Mode: 4
Inflight service		129,880	Mode: 4
Cleanliness		129,880	Mode: 4

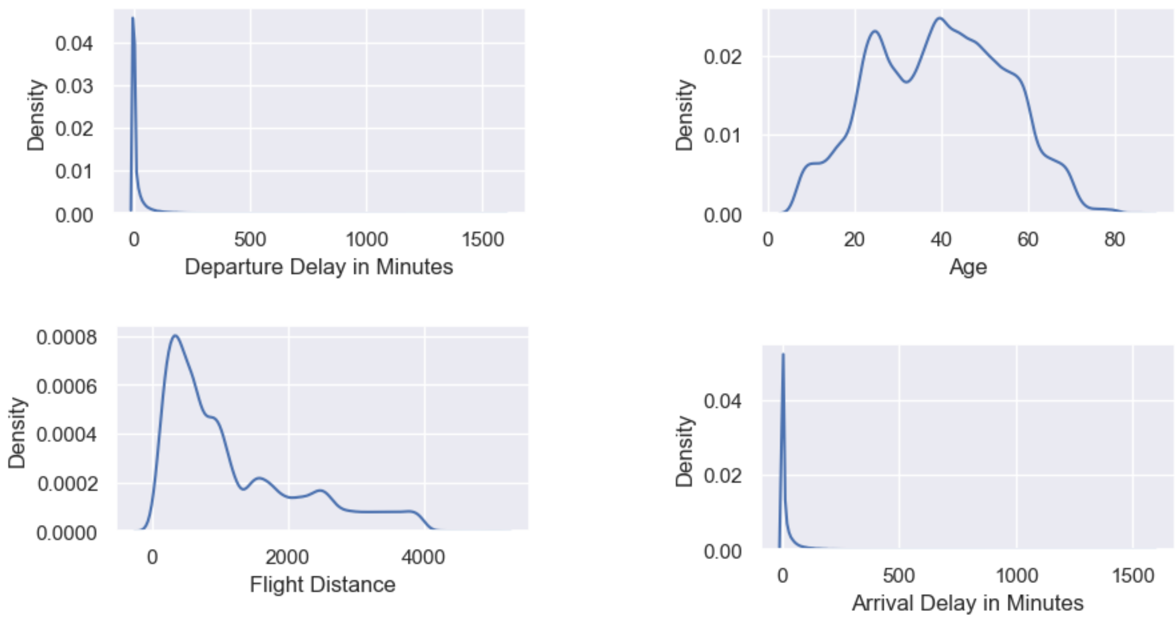


Figure 1 -KDE[i] Density figure of the Continuous Numerical Features

Table 3: Table of Continuous data features

Features	Count	Mean	Median	Range
Departure Delay in Minutes	110259	14.812	0	1592
Arrival Delay in Minutes	57914	15.258	0	1584
Flight Distance	129880	1190.316	844	4952
Age	129880	39.428	40	78

This gives us an idea of the key values within the dataset and helps us understand the data we are using.

2.3 Class balance

From the previous set of figures, we can see that through-out the categorical data it has an even balance except for category 0 as some data has it as an option and others do not. As well as this we can see in the continuous data that the departure delay and arrival delay is skewed towards 0. The report will now go into more detail about satisfaction.


satisfaction
neutral
satisfied
dissatisfied
count
Inflight wifi service
Departure/Arrival time convenient
Ease of Online booking
Gate location
Food and drink
Online boarding
Seat comfort
Inflight entertainment
On-board service
Leg room service
Baggage handling
Checkin service
Inflight service
Cleanliness
Gender
Male
Female

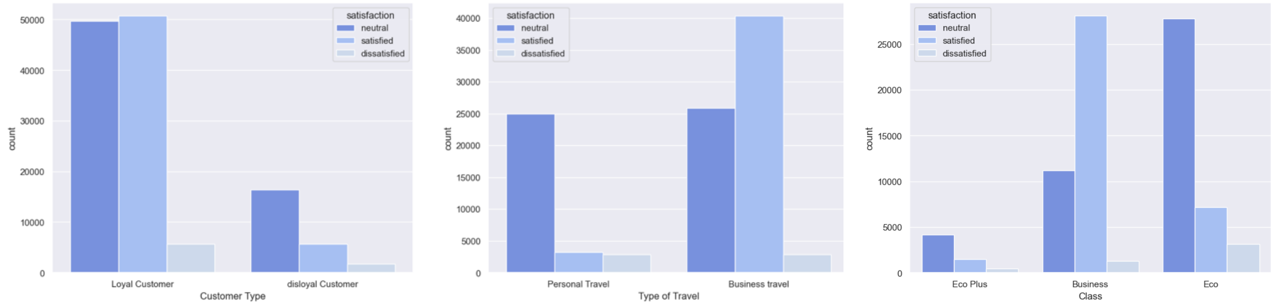


Figure 2 - Figure of Categorical data with a hue of satisfaction

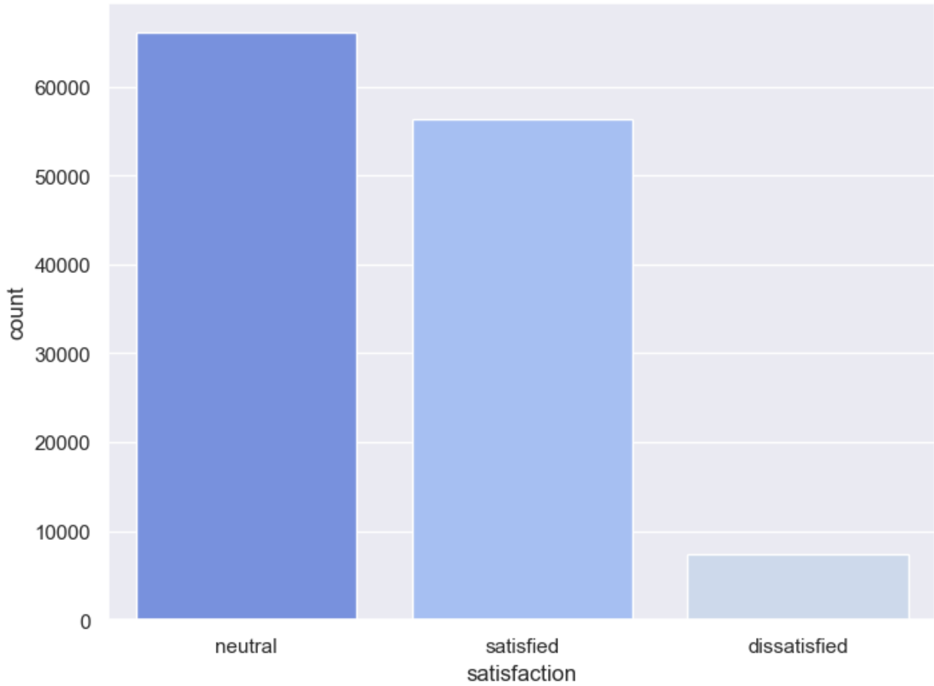


Figure 3 - Figure of Satisfaction Distribution

As we can see in *Figure 3* there is a big class imbalance between dissatisfied and the other two classes, neutral and satisfied. Neutral having the most with 66,080, satisfied next with 56,428 and dissatisfied last with 7,372. As well as this we can see in *Figure 2* that the satisfaction spread is what we would expect with all the data. With a higher satisfaction count on the higher ratings.

2.4 Missing Data

Table 4: Table of Missing Values

ID	0	On-board service	0
Gender	0	Leg room service	0
Customer Type	0	Baggage handling	0
Age	0	Checkin service	0
Type of Travel	29726	Inflight service	0
Class	45153	Cleanliness	0
Flight Distance	0	Departure Delay in Minutes	19621
Inflight wifi service	42971	Arrival Delay in Minutes	71966
Departure/Arrival time convenient	0	satisfaction	0
Ease of Online booking	0		

Gate location	0
Food and drink	0
Online boarding	49439
Seat comfort	0
Inflight entertainment	42884

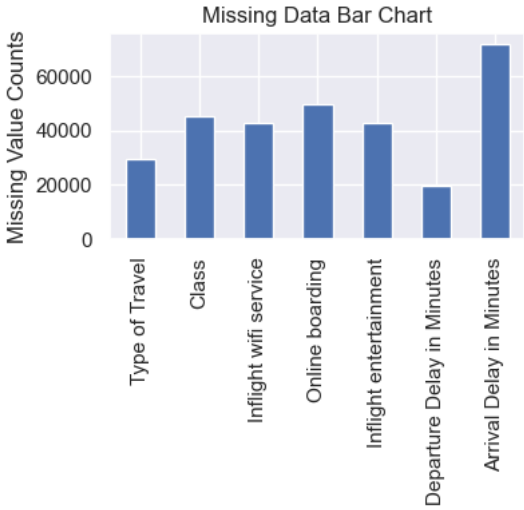


Figure 4 - Missing value count

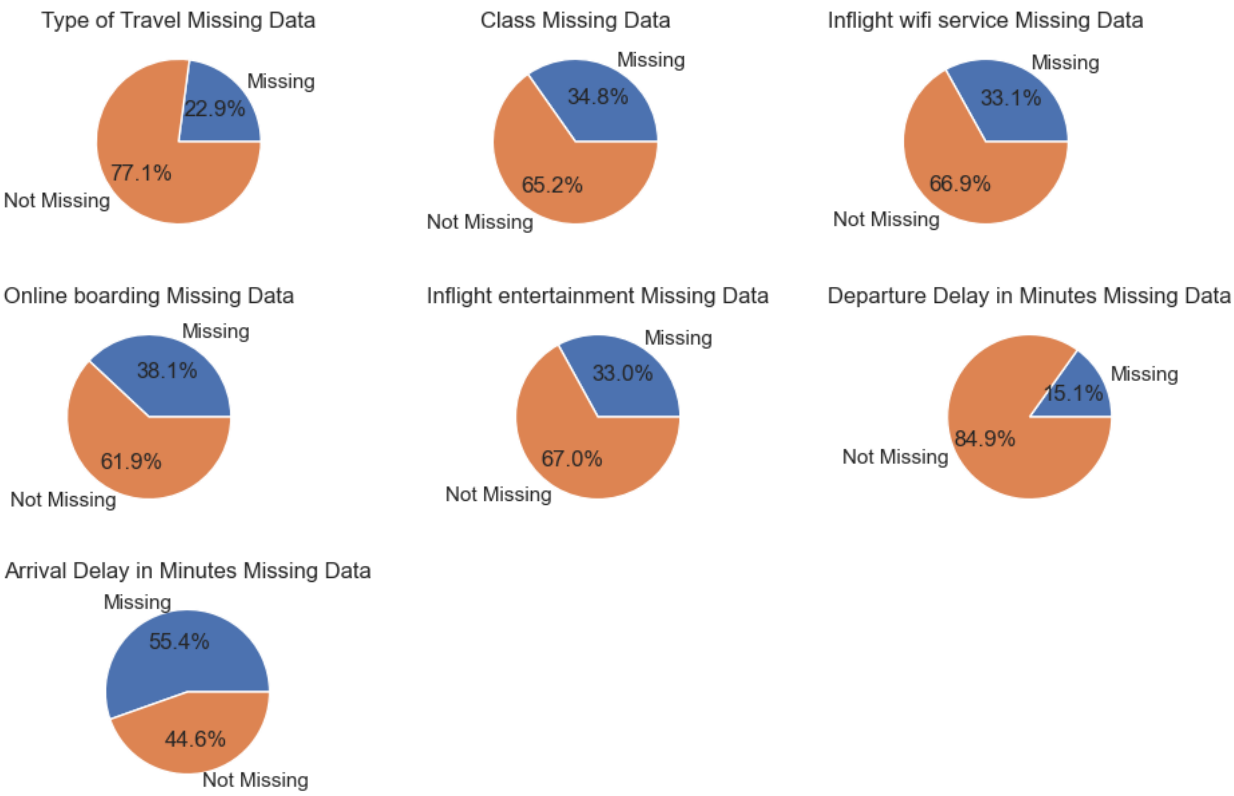


Figure 5 - Percent of category that is missing values

As *Table 4 & Figure 4* shows there are 7 features with missing values. Some are categorical and some are real numbers. *Figure 5* tells us that 5 of the categories have missing values for over 33% and the highest being 55.4%

2.5 Duplicates

After analysing the data, the analysis found 0 duplicates in the dataset. But if there were duplicates this would be fine as it could just be coincidence and with the amount of data there is it would have not been noticeable.

2.6 Noise & Outliers

With this dataset there can only be outliers on the continuous data.

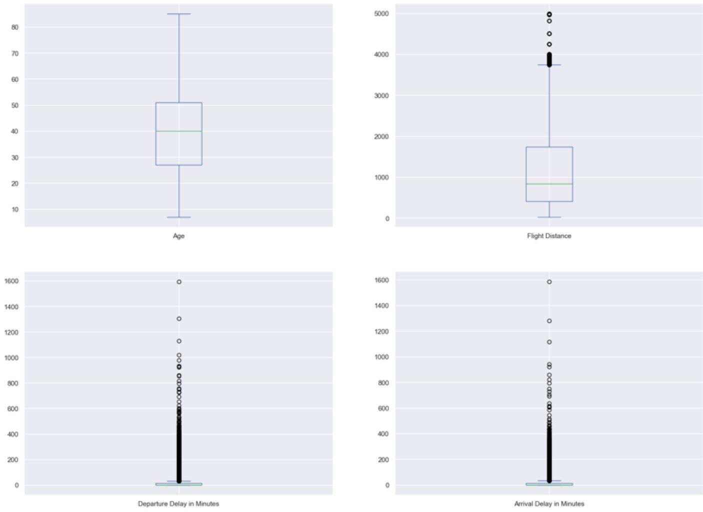


Figure 6 - Boxplot for continuous data

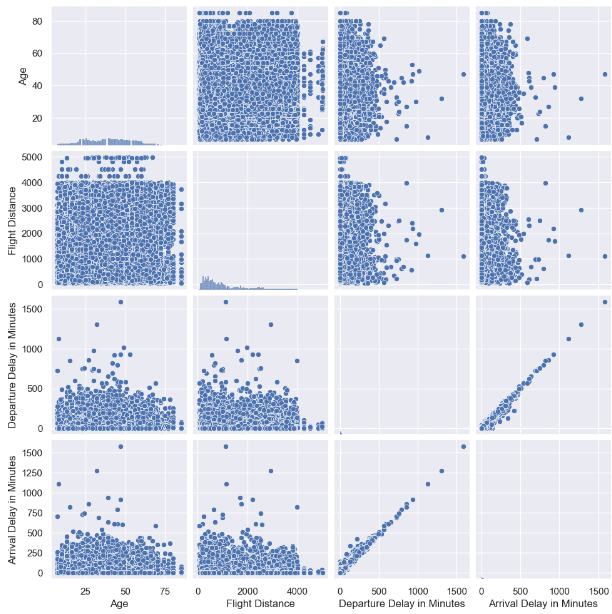


Figure 7 - Pair plot[ii] for continuous data

Both *Figure 6* & *Figure 7* show that there are outliers in the dataset. These outliers may be accidental inputs such as putting in a longer delay time or could just mean that an extremely delayed plane is very rare. As well as this in *Table 2 there is a range that goes from 0-5 but there are very few cases of an actual 0 rating. This may be because they data is not set up properly or there were errors in the input but it is very likely that the 0s are noise in the categorical data.*

2.7 Correlation analysis

Correlation analysis[iii] Glen, S. (2021). will allow me to see if there is any correlation between features. The categorical data was mapped onto values which allowed it to also be in the correlation analysis.

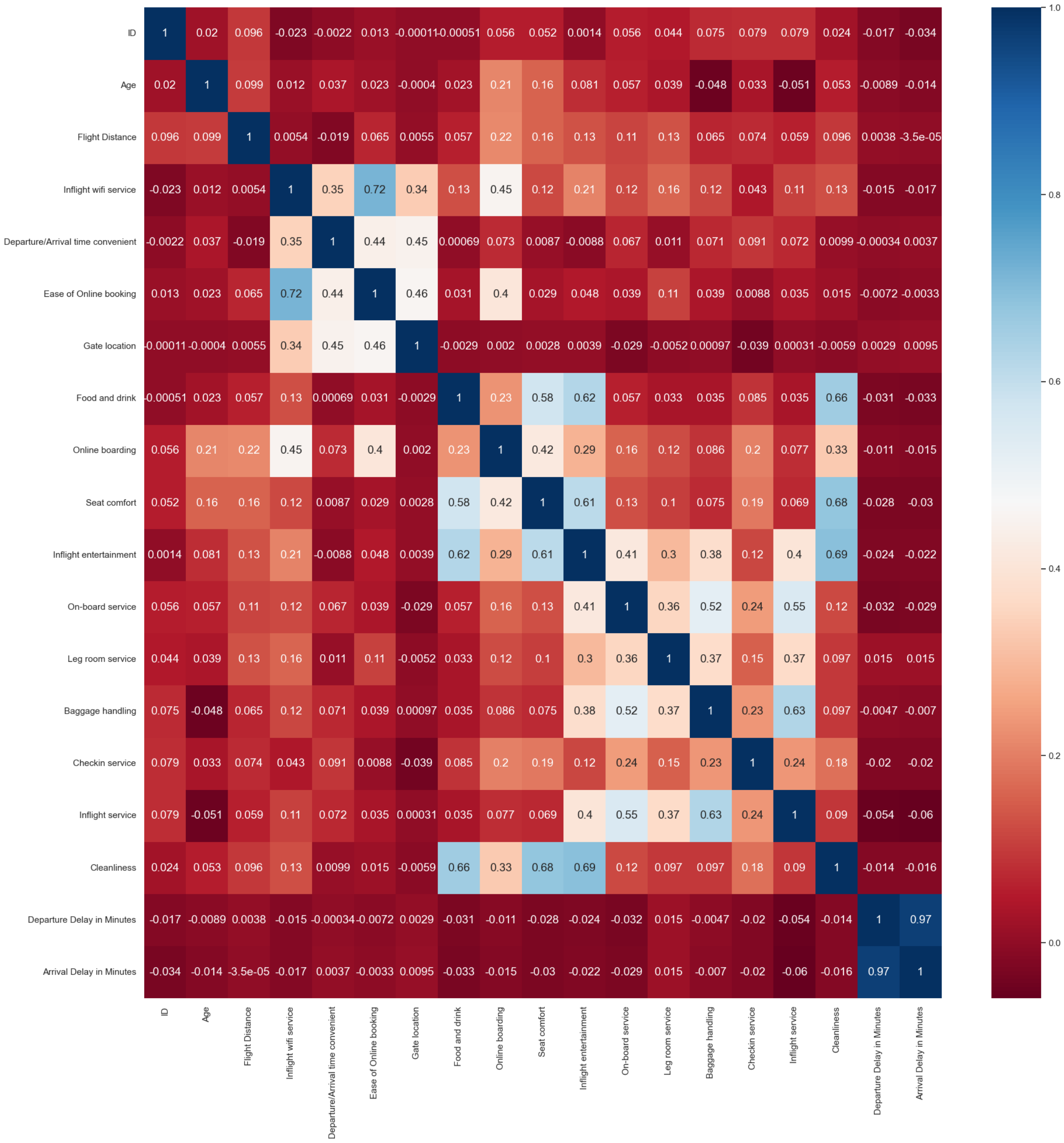


Figure 8 - Pearson correlation analysis[iv]

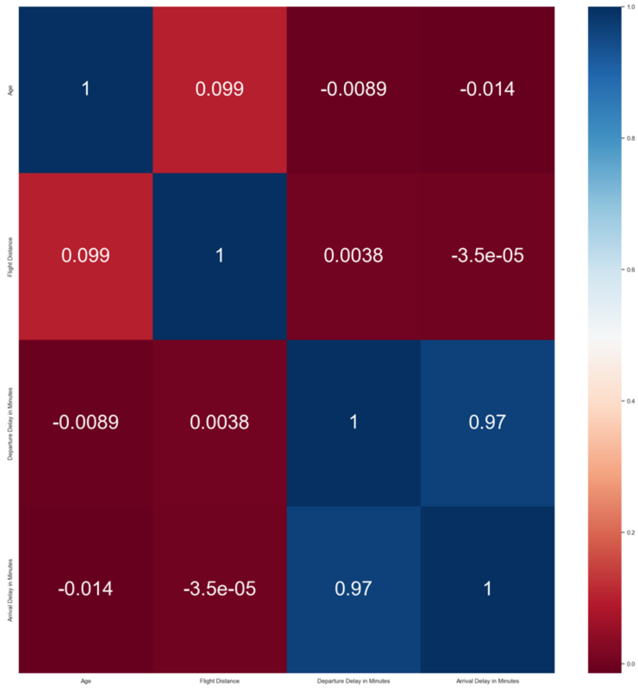
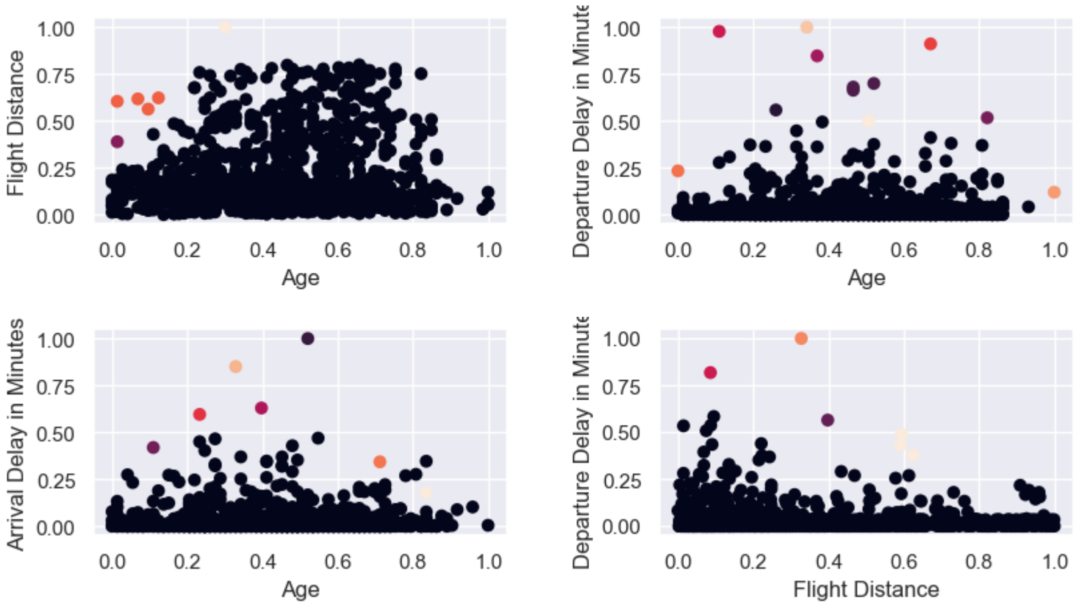


Figure 9 - Pearson correlation analysis with Continuous values

Figure 8 & *Figure 9* shows there is a very strong correlation with Departure Delay in minutes and Arrival Delay in minutes. *Figure 7* also shows that there is a strong correlation between Departure and Arrival Delay in minutes. Figure 8 also shows there is some positive correlation between cleanliness and Inflight entertainment, Seat comfort and food and drink.

2.8 Clustering analysis



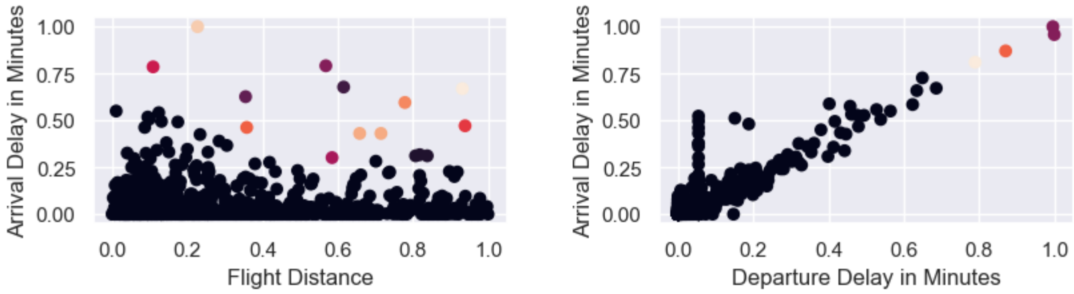


Figure 10 - DBSCANs[v] cluster analysis (continuous data)

Figure 10 shows all possible pairs for a DBSCAN of the continuous data. The figures show that there is only 1 large cluster in each with a few outliers. With more adjustment there would be no outliers. The graphs continue to show a strong positive correlation with Departure Delay in Minutes and Arrival in Minutes.

2.9 Summary of Exploratory Data Analysis

The analysis of the data shows that this is a large dataset with 22 features. And a significant amount of missing data. To be able to use this data and run a machine learning model it will need significant data prepping. As the dataset has lots of missing values and a few strong correlations that can be removed. The data also suggest that how delayed a plane takes off is will determine how delayed it is when it lands. The data also suggest that passengers are most likely to be satisfied when it is clean, there is good entertainment, their seat is comfy and good food and drink.

3 Method

The objective of the method in this report is to create a classification model that can predict a planes passenger satisfaction with an 85%+ Accuracy. As well as this I want to create a model that is efficient and for that I will need to select certain features that have the biggest impact on the accuracy of the model. Another big objective for the report is to be able to input valid and realistic data into the features with missing data.

3.1 Data Processing

In this Report the first step taken to bring the data to a useable state for multiple classifications was to drop[vi] the ID feature and the Arrival Delay in minutes. Arrival Delay in minutes was dropped because it was missing over half of its data and had a strong correlation to Departure Delay in minutes, so Arrival delay and Departure delay were basically the same.

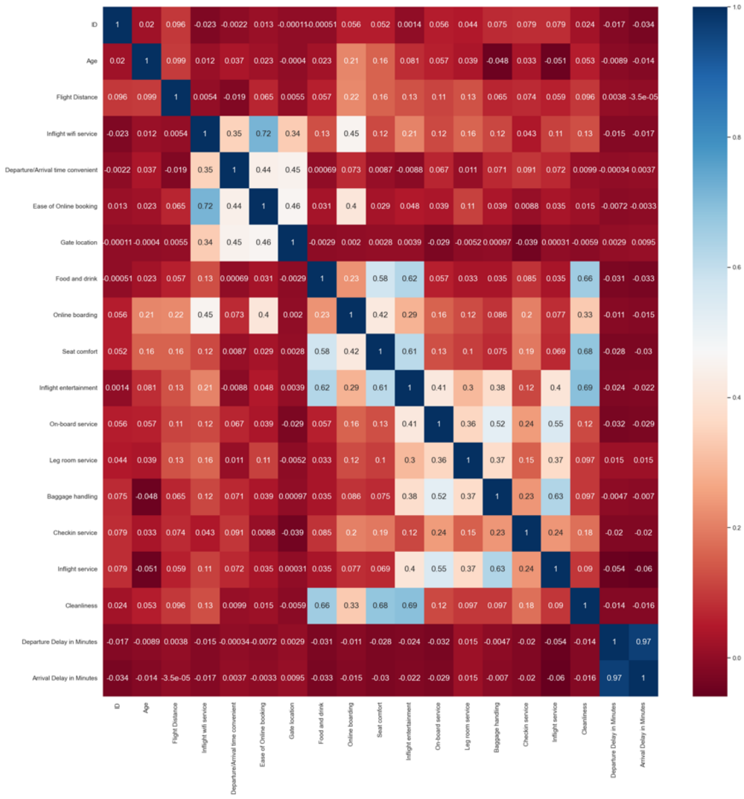


Figure 11 - Pearson correlation analysis

The Next step the report took was a heatmap (*Figure 11*) analysis where I dropped anything with over a 65% this is because it has a strong correlation[vii] DeZyre. (2020). and so did not need two of the same things. This led me to drop Ease of online booking as it had a strong correlation with inflight wifi service with and Cleanliness because it had multiple strong correlations with Food and drink, Seat comfort and Inflight entertainment.

The next data processing was filling the Departure Delay in minutes, missing data. I chose to impute the median[viii] into this feature as it will mitigates the effects of outliers and as there were only 19k missing values will not change the feature that much.

The next step in processing the data is to fill the missing values of all the categorical data using the mode here seemed appropriate if it wasn't so much data that was missing. For most of the features over 1/3 of the data was missing to fill the mode here could massively skew the results. So instead of filling with the mode a KNN[ix] Brownlee, J. (2020). was used for the rest of the missing data. For this the KNN needed all the data mapped to numbers so it could run and find the nearest neighbour. KNN felt like the right choice for the data as it could use hamming distance to measure the distance to and from the categorical data.
The last thing that was done to prepare the data was to run a feature selection[x]. As there are quite a lot of features in this dataset it was best to run a feature selection so that the classifiers aren't spending too much time and resources on features that are irrelevant. For the feature selection it was chosen to make 2 datasets one with the highest 5 features and the other with the 8 highest features. This was done so that we could compare the results from the feature selection in time and accuracy.

3.2 Classification algorithms

The Chosen classifier algorithms are KNN[xi], Random Forest[xii] and SVM[xiii]. KNN was chosen as it does not need training time so is very useful on large datasets and it can handle categorical data. Random Forest was chosen because it can handle large datasets, categorical data very well and is good at feature selection so when it runs with the full data frame it will be able to narrow it down. I chose SVM[xiv] because SVMs are known for achieving high accuracy and unbalanced data.
Each algorithm will also be run with a set with 5 features, 8 features and all the features this is to optimise the accuracy and find where it is best. As well as this will optimise each classifies parameters to find the optimal in each algorithm.

3.3 Validation method

For the report Cross-validation[xv] was chosen because it will help generalise to unseen data, allows for better hyperoperator tuning such as random search which was used in the report allowing the models to find the optimal hyperparameters training the data better. Using Cross-validation on large datasets can lead to more accurate, robust, and generalizable models.

3.4 Evaluation method

The evaluation will be done on accuracy[xvi] (as it is a simple and easy to understand metric that gives an overall idea of how well a model is performing.), confusion matrix[xvii] (It provides a detailed breakdown of the model's performance), and MSE[xviii], MAE[xix], ME[xx] values (as they are commonly used and aren't affected much by outliers). Doing the evaluation like this allows the evaluation to give an overall look at how well the classification models ran.

4 Results and Discussion

4.1 Accuracy, MSE, MAE, ME

Table 5: Accuracy MSE MAE ME table for models

Model	Accuracy	MSE	MAE	ME
Random Forest (5 features)	0.766871	0.268111	0.246024	2.0
Random Forest (8 features)	0.835777	0.178365	0.168488	2.0
Random Forest (All features)	0.879399	0.125209	0.120854	2.0
KNN (5 Features)	0.702531	0.341875	0.317611	2.0
KNN (8 Features)	0.725255	0.313178	0.291014	2.0
KNN (All Features)	0.699598	0.340981	0.316017	2.0
SVM (5 Features)	0.735962	0.299616	0.276693	2.0
SVM (8 Features)	0.773876	0.251981	0.234288	2.0
SVM (All Features)	0.803157	0.216177	0.203636	2.0

The table shows that the Random Forest with all the features had the best Accuracy and MSE, MAE but all had an ME of 2. The ME of 2 makes sense as if the model does get some wrong it doesn't have to get it wrong by much to be the wrong class.

4.2 Confusion matrix

Table 6: Random Forest model with 5 features

	Dissatisfied	Neutral	Satisfied
Dissatisfied	21	1195	217
Neutral	239	10840	2002
Satisfied	67	2323	8813

Table 7: Random Forest model with 8 features

	Dissatisfied	Neutral	Satisfied
Dissatisfied	5	1308	120
Neutral	46	11946	1089
Satisfied	7	1636	9560

Table 8: Random Forest model with All features

	Dissatisfied	Neutral	Satisfied
Dissatisfied	0	1377	56
Neutral	0	12519	562
Satisfied	0	1057	10146

Table 9: KNN model with 5 features

	Dissatisfied	Neutral	Satisfied
Dissatisfied	7	1150	276
Neutral	45	10659	2377
Satisfied	36	3972	7195

Table 10: KNN model with 8 features

	Dissatisfied	Neutral	Satisfied
Dissatisfied	6	1163	264
Neutral	50	10729	2302
Satisfied	21	3399	7783

Table 11: KNN model with All features

	Dissatisfied	Neutral	Satisfied
Dissatisfied	12	1122	299
Neutral	65	10464	2552
Satisfied	22	3746	7435

Table 12: SVM model with 5 features

	Dissatisfied	Neutral	Satisfied
Dissatisfied	0	4713	1179
Neutral	0	41659	10697
Satisfied	0	10694	33923

Table 13: SVM model with 8 features

	Dissatisfied	Neutral	Satisfied
Dissatisfied	0	4982	910
Neutral	0	44230	8126
Satisfied	0	9172	35445

Table 14: SVM model with All features

	Dissatisfied	Neutral	Satisfied
Dissatisfied	0	5247	645
Neutral	0	46709	5647
Satisfied	0	8763	35854

The confusion matrix shows that all the models struggle to predict the dissatisfied column. This is likely because of the class imbalance. This is best shown in the Random Forest model where it gives up all together trying to predict for the dissatisfied class.

4.3 Discussion

After running these models, the pros and cons to each are clear. Random Forest really struggled with the dissatisfied class and eventually stopped predicting for it. But this worked in its favour as it got over the goal 85% but only on the ALL features one and this came at the cost of execution time. KNN performed well on execution time, but its accuracy did not meet the goal of 85% in any of its different variations. SVM worked ok with its accuracy better than KNN but not as good as random forest, but it had execution times that were far longer than the other 2. If you want the best Random Forest, got the highest accuracy and in a short time. The data tells us you can predict an airline passenger satisfaction to key values with the key 5 being Age, Class, Flight Distance, Inflight wifi service and Online boarding.

5 Conclusions and future work

Overall, the Random Forest was the best model for time and for accuracy as with all features it got over 85% which was the goal. The report taught that length of training does not mean better results. Making the dataset optimised for the model will significantly help the model process the data and give better results. This report also shows that you can predict the satisfaction of an airline passenger to a reasonable degree with a machine learning model. If I had more time I would try and improve the dataset so that it fits to an SVM model better.

6 References

[i] Pydata.org. (2012). *seaborn. kdeplot — seaborn 0.9.0 documentation*. [online] Available at: https://seaborn.pydata.org/generated/seaborn.kdeplot.html.

[ii] seaborn.pydata.org. (n.d.). *seaborn.pairplot — seaborn 0.10.1 documentation*. [online] Available at: https://seaborn.pydata.org/generated/seaborn.pairplot.html.

[iii] Glen, S. (2021). *Correlation in Statistics: Correlation Analysis Explained*. [online] Statistics How To. Available at: https://www.statisticshowto.com/probability-and-statistics/correlation-analysis/.

[iv] seaborn.pydata.org. (n.d.). *seaborn.heatmap — seaborn 0.10.1 documentation*. [online] Available at: https://seaborn.pydata.org/generated/seaborn.heatmap.html.

[v] scikit-learn. (n.d.). *sklearn.cluster.DBSCAN*. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html#sklearn.cluster.DBSCAN.

[vi] pandas.pydata.org. (n.d.). *pandas.DataFrame.drop — pandas 1.2.4 documentation*. [online] Available at: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.drop.html.

[vii] DeZyre. (2020). *How to drop out highly correlated features in Python?* -. [online] Available at: https://www.projectpro.io/recipes/drop-out-highly-correlated-features-in-python.

[viii] scikit-learn.org. (n.d.). *6.4. Imputation of missing values — scikit-learn 0.22.2 documentation*. [online] Available at: https://scikit-learn.org/stable/modules/impute.html.

[ix] Brownlee, J. (2020). *kNN Imputation for Missing Values in Machine Learning*. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/knn-imputation-for-missing-values-in-machine-learning/.

[x] Scikit-learn.org. (2019). *1.13. Feature selection — scikit-learn 0.21.3 documentation*. [online] Available at: https://scikit-learn.org/stable/modules/feature_selection.html.

[xi] scikit-learn developers (2019). *sklearn.neighbors.KNeighborsClassifier — scikit-learn 0.22.1 documentation*. [online] Scikit-learn.org. Available at: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html.

[xii] Scikit-learn (2018). *3.2.4.3.1. sklearn.ensemble.RandomForestClassifier — scikit-learn 0.20.3 documentation*. [online] Scikit-learn.org. Available at: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html.

[xiii] scikit-learn developers (2019). *sklearn.svm.SVC — scikit-learn 0.22 documentation*. [online] Scikit-learn.org. Available at: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html.

[xiv] scikit learn (2018). *1.4. Support Vector Machines — scikit-learn 0.20.3 documentation*. [online] Scikit-learn.org. Available at: https://scikit-learn.org/stable/modules/svm.html.

[xv] SciKit-Learn (2009). *3.1. Cross-validation: evaluating estimator performance — scikit-learn 0.21.3 documentation*. [online] Scikit-learn.org. Available at: https://scikit-learn.org/stable/modules/cross_validation.html.

[xvi] Scikit-learn.org. (2019). *sklearn.metrics.accuracy_score — scikit-learn 0.22 documentation*. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html.

[xvii] Scikit-learn.org. (2019). *sklearn.metrics.confusion_matrix — scikit-learn 0.21.3 documentation*. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html.

[xviii] scikit-learn.org. (n.d.). *sklearn.metrics.mean_squared_error — scikit-learn 0.24.2 documentation*. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_squared_error.html.

[xix] scikit-learn. (n.d.). *sklearn.metrics.mean_absolute_error*. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_absolute_error.html.

[xx] scikit-learn. (n.d.). *sklearn.metrics.max_error*. [online] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.max_error.html [Accessed 16 Jan. 2023].