ST2195 COURSEWORK

Report for Part 2

SRN: 220457675

Total number of pages:8 (Including TOC and References)

Table of Contents

2. Harvard Dataverse

For both R and Python, the 10 years of data (1999 to 2008) was put into a data frame called 'ontime_df' for easier access and display. Other data such as airports, carriers, plane-data and variable-description was also put into a data frame. As 'ontime_df' was used across the question, cleaning was done in advance. Extra columns such as 'Date' and 'TotalDelay' were created where the Year, Month and Day was put together to form 'Date', and ArrDelay and DepDelay was added to form 'TotalDelay'. To reduce the size of the data, unnecessary columns was left out, leaving only important ones in 'ontime_df'. The newly cleaned 'ontime_df' will be used to answer the question.

2.1 Best times and days of the week to minimise delays each year

In order not to affect the original 'ontime_df', a copied version titled 'p2aontime_df' was created. Any NA values was cleared so that it will not affect the analysis. The times of the week will be defined as the stages of the day titled 'DateandTime', which has 4 categories of 'Morning', 'Afternoon', 'Evening' and 'Night'. A 'TimeStamp' column was created including both date and time, where it was then passed through a function to sort the timings into the individual stage. The days of the week will be defined as the normal 'Monday' to 'Sunday'. Originally, it was labelled as '1' to '7', where it was then renamed to the days of the week.

To find the best times of the week to minimise delays each year, 'p2aontime_df' was filtered by each year, followed by 0 cancellations, 0 diversions, and a 'TotalDelay' of more than 0. 0 cancellations and 0 diversion meant that flights that successfully took off and land from their intended departure airport to their intended arrival airport are used in the analysis. As a delay that has occurred, 'TotalDelay' will be positive, as a negative number will imply that a delay has not occurred. An average delay will be calculated for each 'DateandTime', which will then be grouped and arranged in ascending order. A bar graph for each year is plotted. From the R and Python graph (Figure 2.1) below, where both have a y-axis of 'Average Delay' and x-axis of 'Times', the results generated are similar. It can be seen that for every year from 1999 to 2008, the best time to minimise delay is taking a flight in the Morning, as it has the lowest average delay of about 30 minutes. The graph also showed that the worst time is Night, where every year it has a more than an average delay of 90 minutes. Additionally, Afternoon is the second-best time, followed by Evening which is the third best time.

To find the best days of the week to minimise delays each year, the same filtering was used as for best times. Instead of calculating an average delay for each 'DateandTime', it is calculated for each 'DayOfWeek', which will also be grouped and arranged in ascending order. A bar graph for each year is also plotted. From the R and Python graph (Figure 2.1.1) below, both have a y-axis of 'Average Delay' and an x-axis of 'Days'. The results generated are also similar. From 1999 to 2001, the best day to fly was on Tuesday, with an average delay of 43 minutes, while the worst day to fly was on Friday, with an average delay of 50 minutes. From 2002 to 2007, the best day to fly was Saturday, with an average delay of about 42 minutes, and the worst day was a mixture of days. Interestingly, for 2006 and 2007, the average delay rose to about 50 minutes, which is similar to the best days from 1999 to 2001. In 2008, Wednesday became the best day to fly, which can be shared with Thursday as the results are 1 minute apart. The worst day is Friday, where the average delay is 64 minutes. It can be inferred that the average delay for all days slowly increased as years went by, where average delay for the best day also slowly increased.

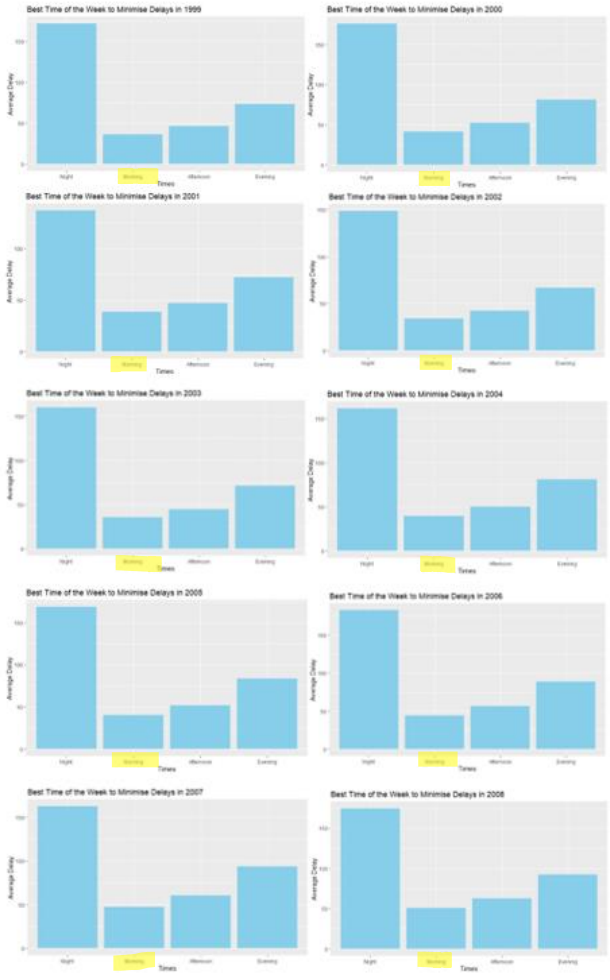

Figure 2.1a: Best Times (R)

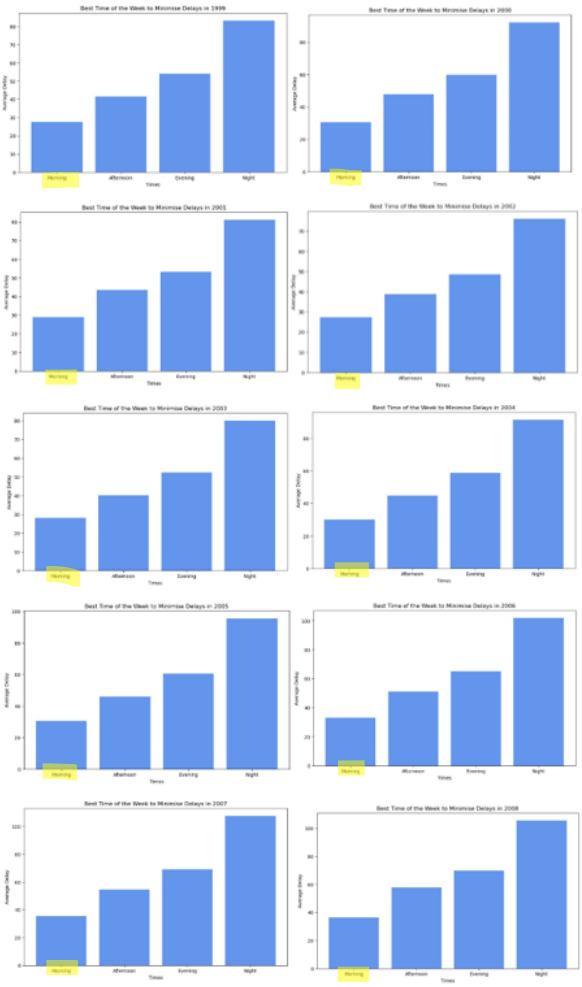

Figure 2.1b: Best Times (Python)

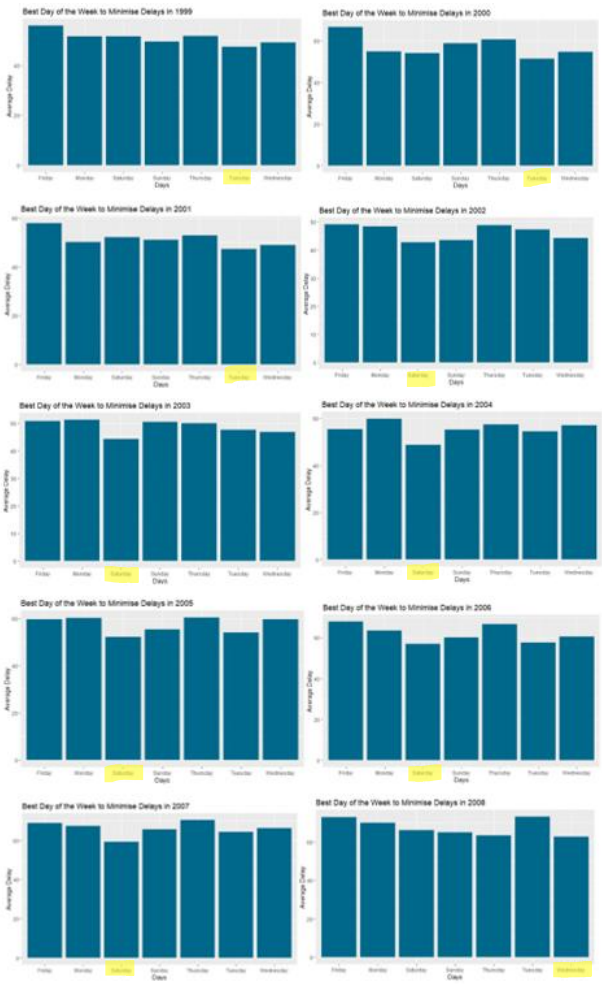

Figure 1.1.1a: Best Days (R)

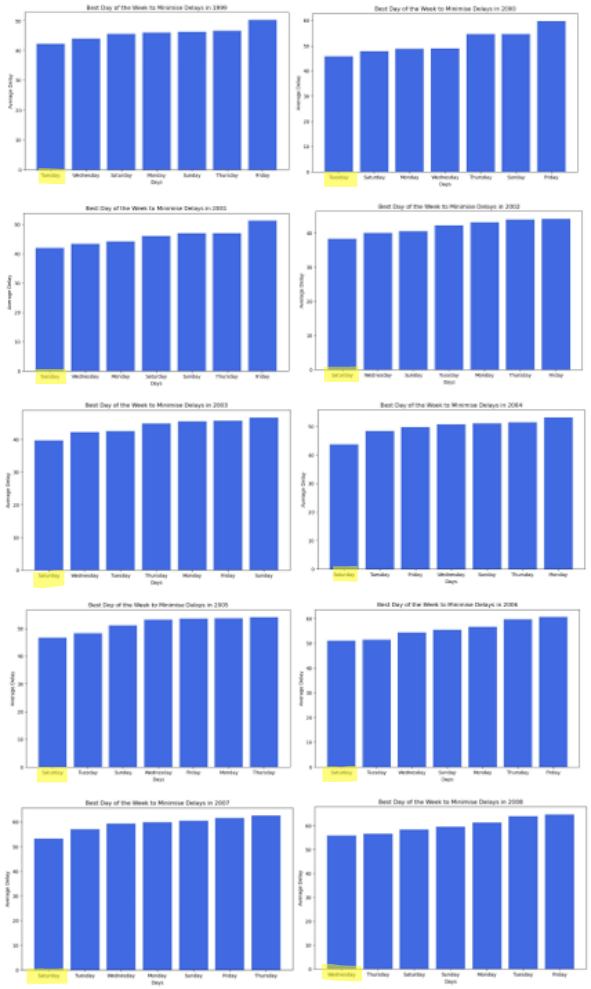

Figure 2.1.1b: Best Days (Python)

2.2 Evaluate whether older planes suffer more delays

For 2.2, 'planes_df' was used. In the original data frame, it had blank cells, where I filled it with NA values before removing it. 'Ontime_df' was subset to choose necessary columns such as the 'Year', 'TailNum' and 'year' (manufacturing year), where NA values was dropped as well. Both data frames were then joined by 'TailNum', where further NA values and unnecessary columns was dropped. To find older planes, range of the manufacturing years was generated (1956, 2008) where it is then segmented by a threshold of 1998 as a plane manufactured within 10 years was considered new, while anything more than 10 years was considered old[1]. There were some unrelated data generated, such as '0000', 'None' and '0'. These were filtered off as it does not make sense that it is part of the manufacturing year. A category of new and old planes was created, where the average delay was calculated, and then grouped by 'Year' and manufacturing years.

From the graphs below (Figure 2.2), where for both R and Python is also similar, both have a y-axis of 'Average Delay' and a x-axis of 'Year Category'. It can be seen that the older planes (in orange for Python and blue for R) had more average delays from 1999 to 2003, before having the same average delays as newer planes in the middle of 2003, and ultimately having lesser average delays as compared to newer planes from 2004 onwards. Thus, it can be concluded that older planes do suffer more delays in the first half of the 10 years, whereas newer planes suffer more delays in the second half of the 10 years.

However, it can also be inferred that the average delays for both older and newer planes increased from 1999 to 2008. The newer planes had the lowest average delay in 2002, while the older planes had the lowest average delay in 2003. Even though newer planes suffered more delays in the second half of the 10 years, the older planes also have an average delay that is almost similar to the newer planes, hence resulting in not a big difference between the two.

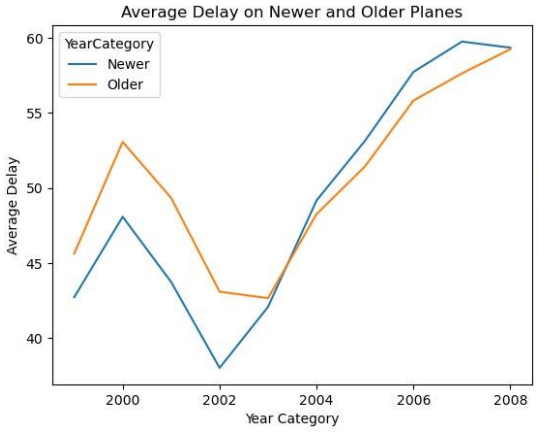

Figure 2.2a: Older Planes Delay (Python)

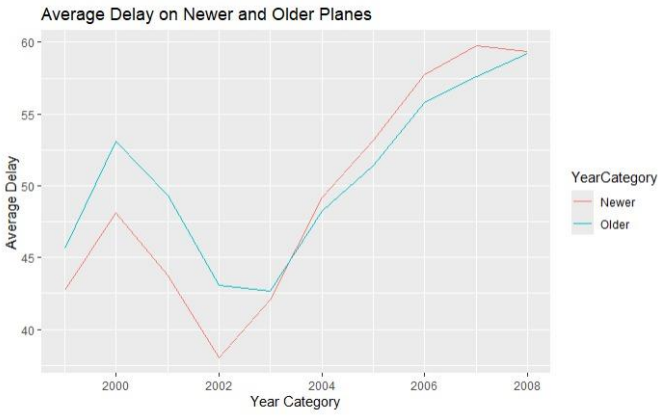

Figure 2.2b Older Planes Delay (R)

[1] Paramount Business Jet Website

2.3 Logistic Regression Model

Columns that contained the features from 'ontime_df' was selected, as well as IATA codes and coordinates from 'airports_df'. They were then joined via the IATA codes to form a new data frame, before removing unnecessary columns such as the 'Origin' airport and 'Dest' airport. The 'Dates', originally in datetime structure, was changed into integer format, which will be used as a numerical feature. Only 'UniqueCarrier' will be used as a categorical feature. For the probability of diverted US flights, the dependent variable (target) will be 'Diverted', while the independent variable (features) will be all 6 features listed in the question.

In python, the new data frame 'probability' was filtered by each year, before the pre-processing pipelines. Here, the numerical and categorical features, and the target variable was listed. The features then undergo transforming, where they were imputed to fill any missing values. The numerical features were also scaled to standardise the data, while the dummy variables were created for the categorical feature. Then, training and testing sets was created, where 50% of the data selected underwent training, and the other 50% was used for testing. To find the ideal result, parameters was created and searched, before inputting it in the Logistic Regression model. To know the performance of the model, a ROC (Receiver Operating Characteristic Curve) was plotted where Sensitivity is against Specifity, as well as it's AUC (Area Under Curve) value. It is the most ideal when the curve is as close to the top left corner, or AUC = 1. It is the least ideal when AUC = 0.5. From the graph (Figure 2.3.1) below, all 10 years showed similar AUC values of about 0.64. Model 1 (1999) is the most ideal with an AUC = 0.66, while model 3 is the least ideal with an AUC = 0.62. Additionally, we want to visualise coefficients over the years. Coefficients and features of each year was generated, and as the results for all 10 years were shown, it was shown that different years had different number of features. This was caused by the 'UniqueCarrier' feature, where there was different US[2] IATA codes in each year. To plot a more concise graph, common carriers amongst all 10 years was picked out while the uncommon ones was removed. They were not removed beforehand as removing it may cause the prediction of diverted flights to be less accurate. Coefficients of the model means that for any change in the feature, the probability of the target happening will be multiplied by that amount. From the graph (Figure 2.3.2) below, it can be seen that all 10 years have a similar shape, where the carriers, scheduled departure time and scheduled arrival time fluctuates more. The rest of the features are relatively stable. This means that the three features are the ones that mostly impact the probability of diverted US flights. Depending on what is the scheduled departure and arrival time, and what airline is operating, the probability of diversion varies.

In R, the new data frame 'logistic_regression_model' was also filtered by each year, before training and testing set was added. 50% of the data was used for training, while the other half was used for testing. To calculate the classification error, a task was set up, where the same target and features was listed. Classification error refers to the amount of incorrectly classified objects divided by the total number of objects. The lower the classification error, the better the model is. From the graph (Figure 2.3.3) below, it can be seen that all models have a low classification error, where model 4 (2002) has the lowest classification error of 0.001597, while model 2 (2000) has the highest classification error of 0.002510. The logistic regression model was added, where missing values was substituted with the mean of the variable's values. Training of the training set is set, followed by generating the prediction. To visualise the coefficients over the years, coefficients and their respective feature was generated, shaped into a data frame, where the same removal of extra carriers was done. The features were listed in a new column so that it matches the coefficients, before removing the intercept row as only coefficients are needed. From the graph (Figure 2.3.4) below, it can be seen that only the carriers' coefficients fluctuations all throughout the 10 years, while the numerical features are relatively the same. This also means that the carriers are the ones that mostly impact the probability of diverted US flights. Carrier AS (Alaska Airlines) had high positive coefficients with the exception of 2007 and 2008, where it increases the probability of diversion, while carrier DL (Delta Air Lines) had negative coefficients for all 10 years, decreasing the probability of diversion.

[2] Federal Aviation Administration website

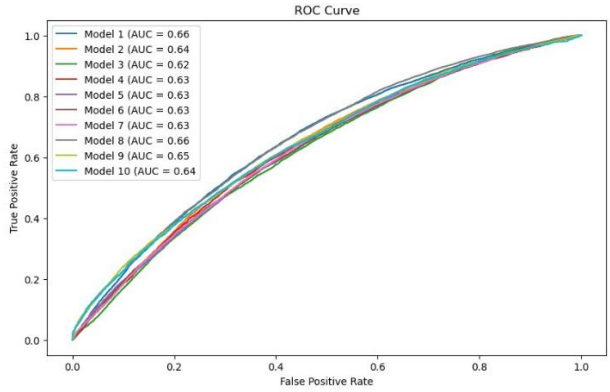

Figure 2.3.1: ROC Graph (Python)

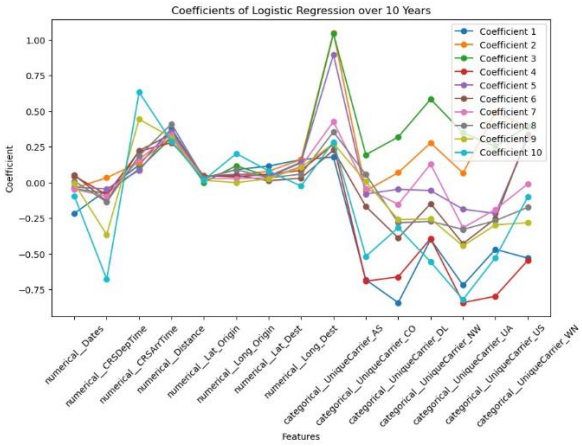

Figure 2.3.2: Coefficients over 10 years (Python)

```
<TaskClassif:model1> (5527884 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.002448315
```

```
<TaskClassif:model2> (5683047 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.002509922
```

```
<TaskClassif:model3> (5967780 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.002169316
```

```
<TaskClassif:model4> (5271359 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.001596553
```

```
<TaskClassif:model5> (6488540 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.001774821
```

```
<TaskClassif:model6> (7129270 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.001947464
```

```
<TaskClassif:model7> (7140596 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.001964542
```

```
<TaskClassif:model8> (7141922 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.002305542
```

```
<TaskClassif:model9> (7453215 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.002312559
```

```
<TaskClassif:model10> (2389217 x 10)
* Target: Diverted
* Properties: twoclass
* Features (9):
  - int (4): CRSArrTime, CRSDepTime, Date, Distance
  - dbl (4): Lat_Dest, Lat_Origin, Long_Dest, Long_Origin
  - fct (1): UniqueCarrier
 classif.ce
0.002347212
```

Figure 2.3.3: Classification Error (R)

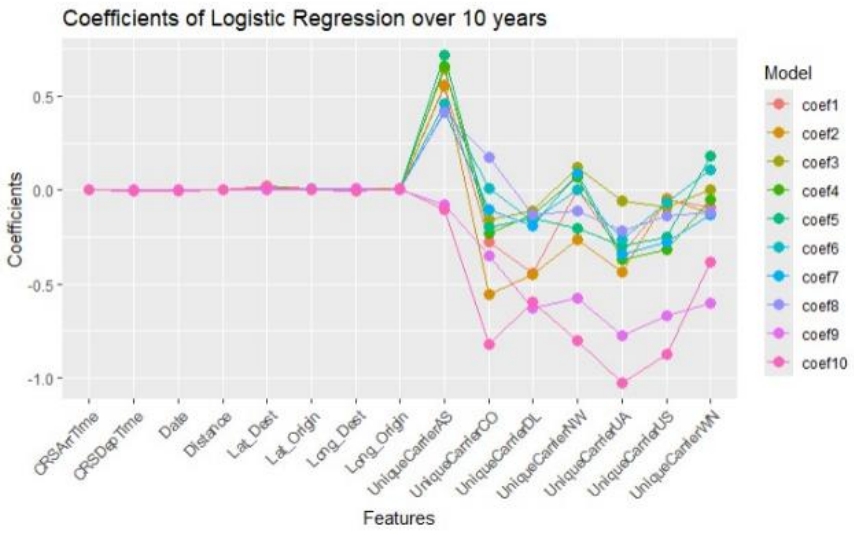

Figure 2.3.4: Coefficients over 10 years (R)

3 References

Is the age of an aircraft a safety factor? (n.d.). Paramount Business Jets.

https://www.paramountbusinessjets.com/faq/age-of-aircraft-safety-factor#:~:text=An%20aircraft's%20age%20is%20based,Standard%20aircraft%20%3D%2010%2D20%20years

ASQP : Carrier codes and Names - ASPMHelp. (n.d.).

https://aspm.faa.gov/aspmhelp/index/ASQP___Carrier_Codes_And_Names.html