ST2195 Programming for Data Science

COURSEWORK REPORT

LIANG ZHIKAI
STUDENT ID: 230709733 | UNIVERSITY OF LONDON

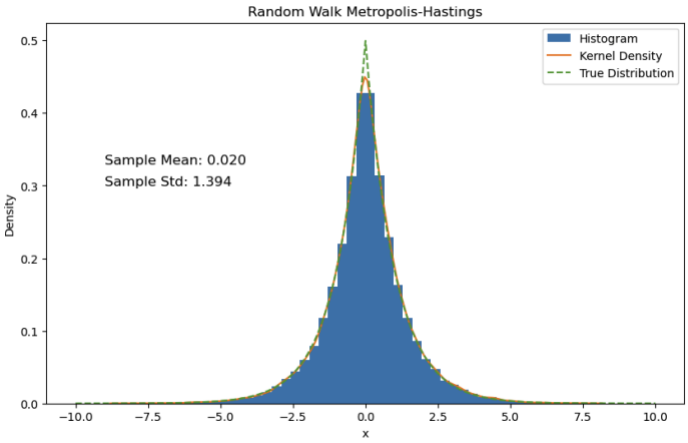
Table of Contents

1a) Apply the random walk Metropolis algorithm using N = 10000 and s = 1. Use the generated samples (x1, . . . xN) to construct a histogram and a kernel density plot in the same figure. Note that these provide estimates of f (x).Overlay a graph of f (x) on this figure to visualise the quality of these estimates. Also, report the sample mean and standard deviation of the generated samples.

To generate the samples, two functions are created. The first function defines the probability density function (PDF) used for sampling with the Metropolis-Hastings algorithm. This PDF is defined as,

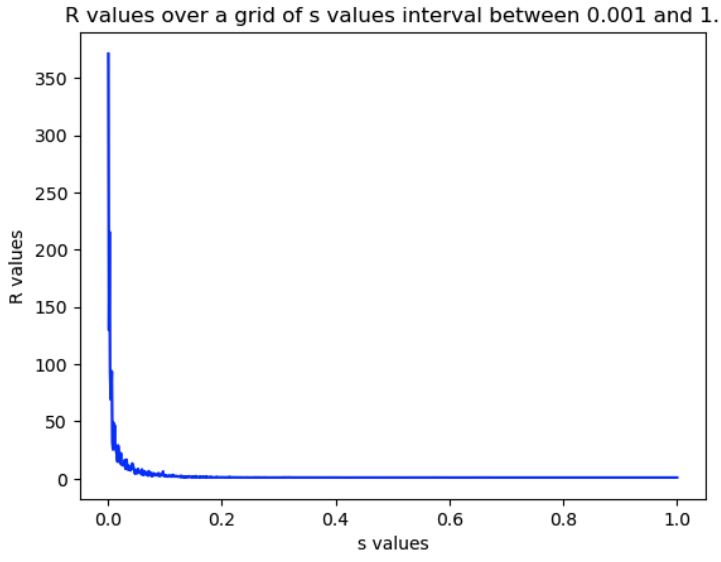
$$\mathbf{f(x)} = \frac{\mathbf{1}}{\mathbf{2}}\mathbf{exp}(-|\mathbf{x}|)$$

The second function implements the Metropolis-Hastings algorithm to generate samples from the specified PDF distribution. The algorithm iterates through N = 10000 iterations, sampling new values 'x_*' selected randomly from a normal distribution centered around the previous sample, with a standard deviation of s = 1. Each proposed sample's probability density is computed, and compared to the previous sample's probability density. If the ratio of these densities exceeds a random number u from a uniform distribution of 0 to 1, the proposed sample is accepted; otherwise, the previous sample is retained. The chosen sample is then stored in an array.



The resulting samples are then used to create a histogram, visually comparing the estimated distribution with the true distribution defined by the PDF. The histogram and Kernel Density Estimate (KDE) closely resemble the true distribution, indicating the algorithm's success in approximating the desired distribution. Sample statistics reveal a mean of 0.02 and a standard deviation of 1.394, providing insights into the central tendency and variability of the generated samples.

1b) In general, values of R hat close to 1 indicate convergence, and it is usually desired for R hat to be lower than 1.05. Calculate the R hat for the random walk Metropolis algorithm with N = 2000, s = 0.001 and J = 4. Keeping N and J fixed, provide a plot of the values of R hat over a grid of s values in the interval between 0.001 and 1.




```
Initial values used to compute R hat: [25 10 11 23]
R hat value for N = 2000, s = 0.001, and J = 4: 371.2913362596153
R hat value for N = 2000, s = 0.5, and J = 4: 1.0202321011170785
```

For this part, a third function is defined to compute the R hat value based on between-chain variance (B) and within chain variance(W). Then we iterate over a grid of s values, generating multiple chains for each s value using the Metropolis-Hastings function created in part (a). The R hat value is then calculated for each set of chains and the results is then stored in a list.

The resulting plot displays these R hat values on the y-axis, while the corresponding s values are shown on the x-axis. A steep decline followed by a slow convergence is observed, indicating a successful convergence of the algorithm for the given range of s values. Notably, the R hat value of 1.02 at s = 0.5 is desirable, indicating convergence, as it falls below the threshold of 1.05.

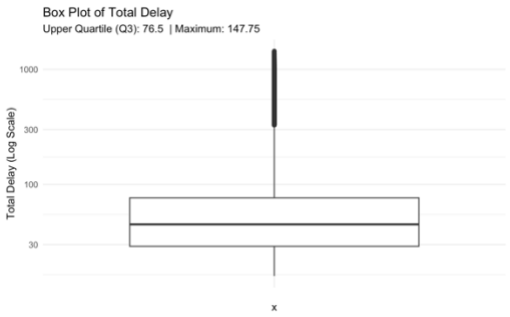
2a) What are the best times and days of the week to minimise delays each year?

This study aims to identify the time period exhibiting the lowest average delay and the smallest proportion of delayed flights in commercial aviation. A flight will be classified as delayed if both its departure and arrival exceed 15 minutes. Through the use of ANOVA, an examination will be conducted to ascertain if the mean total delay remains consistent across different time intervals and days of the week. Total delay in this analysis refers to the average of departure and arrival delay. Subsequently, confidence intervals for the mean delays of each group will be calculated at a 95% confidence level. Furthermore, the study seeks to visually represent the fluctuation in the proportion of delayed flights over the years.

2.1 Data Preparation, Cleaning and Wrangling

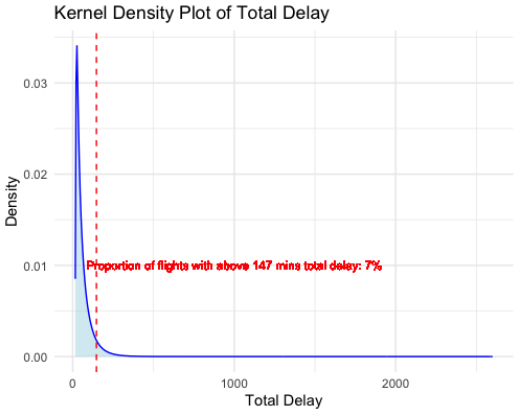
The data analysis process commences by reading and merging flight data covering the years 1998 to 2007. Following this, duplicate observations are eliminated, and the dataset is refined to include only the necessary variables for addressing the question. Subsequently, any missing values attributed to data entry errors are excluded, along with flights that have been diverted or cancelled. Finally, the data undergoes formatting and conversion into the correct data types, including grouping scheduled departure hours into 6 intervals of 4 hours each.

2.2 Data Exploration



Upon observing each selected variable using histograms and Kernel Density plots, it appears that the 'Total_delay' variable may contain extreme outliers. To delve deeper into these outliers, a box plot of total delay is generated for delayed flights. The box plot reveals several outliers with values exceeding 147 minutes, which is the calculated maximum cut-off.

Given that 7% of the flights are outliers, representing a significant proportion, we opt to set a cut-off for extreme values. For this analysis, we define extreme values as any flights delayed for more than one day since outliers that fall between 147 and 1440 minutes are genuine and meaningful. Therefore, we will include data with a total delay of less than 1440 minutes (equivalent to one day) for our analysis. Flights with delays exceeding 1440 minutes will be excluded, as excessively large delays can potentially skew our analysis, particularly when measuring the mean delay.



2.3 Analysis and Data Visualization

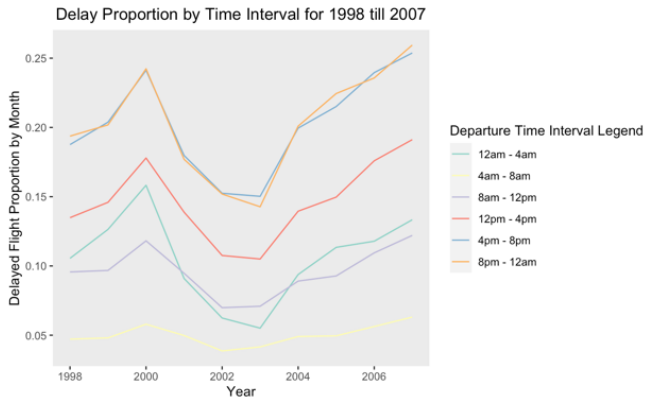
ANOVA

The ANOVA analyses for time intervals and days of the week both yielded statistically significant results with p-values below 0.05 at a 95% confidence level. These findings reject the null hypothesis, suggesting that mean total delay significantly varies across different time intervals and days of the week.

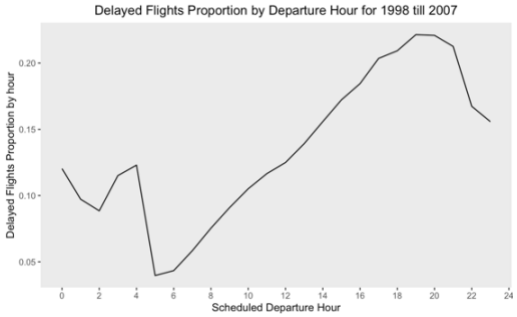
When is the best time to fly to minimize delay?

	2.5 %	97.5 %
(Intercept)	6.060467	6.232400
DepartureBins4am - 8am	-4.958427	-4.781406
DepartureBins8am - 12pm	-1.873582	-1.698936
DepartureBins12pm - 4pm	2.313496	2.488234
DepartureBins4pm - 8pm	6.947838	7.122651
DepartureBins8pm - 12am	5.955965	6.134950

The 95% confidence interval suggests that mean total delay is lowest for the 4am to 8am time interval with end points (-4.95mins, -4.78mins). To determine the lowest delay proportion, Data will be grouped by departure intervals and year to compute delay proportions for each interval. Results will be visualized via a line graph.



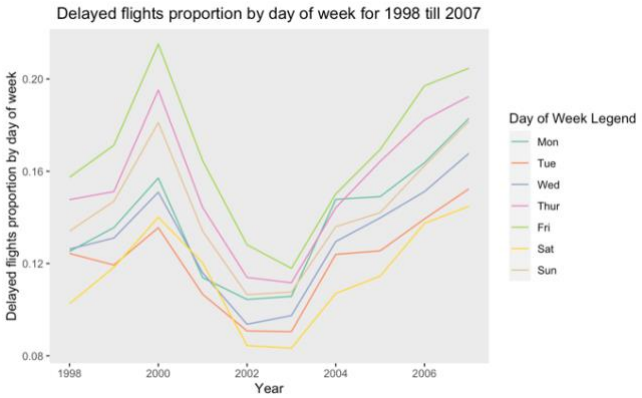
The line graph on the left illustrates that the period between 4am to 8am consistently experiences the lowest delay proportion throughout the years. To pinpoint the optimal hour for minimizing delays, another line graph that displays the delay proportion by hour will be ploted. The plot clearly highlights that 5am boasts the lowest delay proportion. Furthermore, we observe that delays gradually increase and peak around 6 pm. This pattern aligns with the phenomenon where delays from earlier flights can cascade and affect subsequent flights, leading to increased delays as the day progresses.

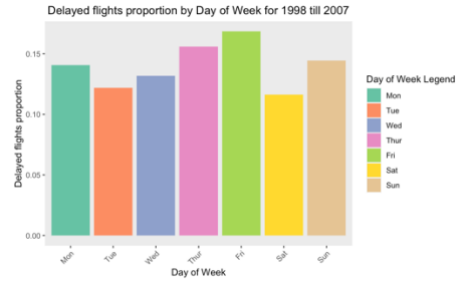


When is the best day of week to fly to minimize delay?

	2.5 %	97.5 %
(Intercept)	7.99469711	8.0357779
DayOfWeekTue	-1.78862447	-1.7303627
DayOfWeekWed	-0.68871924	-0.6305255
DayOfWeekThur	1.57306327	1.6311967
DayOfWeekFri	2.43346520	2.4915472
DayOfWeekSat	-2.69146184	-2.6312344
DayOfWeekSun	0.06111193	0.1199376

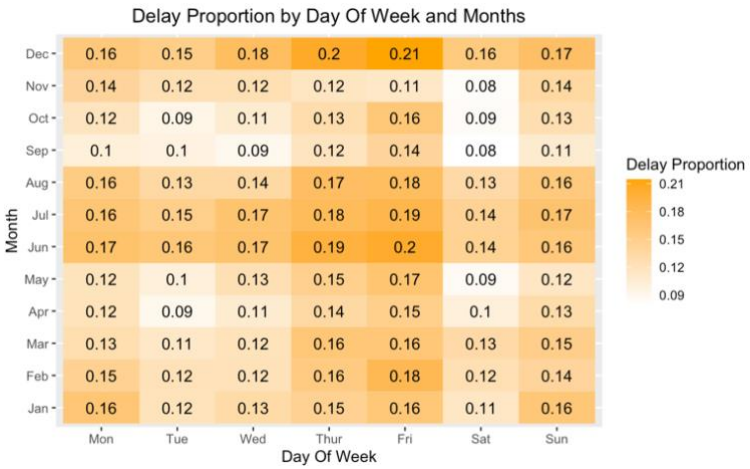
The 95% confidence interval suggests that the mean total delay is the lowest for Saturday, with endpoints (-2.69 mins, -2.63 mins). A bar chart will be plotted to visualize the delay proportion by day of the week to ascertain if Saturday indeed exhibits the lowest delay proportion. Following this, an evaluation will be conducted to determine if the delay proportion remains consistent throughout the years using a line graph.





Both the bar chart and line graph demonstrate that Saturday consistently has the lowest delay proportion throughout the years, except for the years 2000 and 2001, where Tuesday exhibits the lowest delay proportion.

Is the best day of week to fly the same for different months?



To answer this question, we grouped the data by day of the week and month, calculating delay proportions accordingly. The generated heatmap offers a comprehensive overview of delay variations. It confirms that Saturday generally experiences the lowest delay proportion across most months. However, for certain months, Tuesday either matches or exceeds Saturday's performance in terms of delay proportion. Additionally, the heatmap highlights September as the optimal month for flying due to consistently low delay proportions.

Conclusion

The analysis suggests that the most favourable times and days of the week to minimize delays each year are flying in the morning, specifically from 5 to 6am, on either Saturday or Tuesday. Additionally, flying during the month of September further minimizes delays.

2b) Evaluate whether older planes suffer more delays on a year-to-year basis.

To address this question, we will utilize a combination of statistical techniques, particularly linear regression analysis and Pearson's correlation coefficient. Additionally, we will visualize the data using correlation matrix, bar plots, and scatter plots to examine the delay proportion across different aircraft ages and investigate the relationship between plane age and total delay.

3.1 Data Preparation, Cleaning and Wrangling

The data cleaning process begins by selecting the necessary variables from the flights data created in part (a). Subsequently, supplementary plane data is imported and merged with the flights data. A new variable, 'Plane-age', is created by calculating the difference between the year of the flight and the year the plane was manufactured. Missing values arise during this process, either due to absence of plane information in the supplementary data or missing information on the year of manufacture. To handle the latter, we extract the year from the issue date and use an if-else statement to substitute missing values with the year of the issue date. As there are no other methods to address missing values caused by unavailable information in the plane data, these observations are omitted from our analysis.

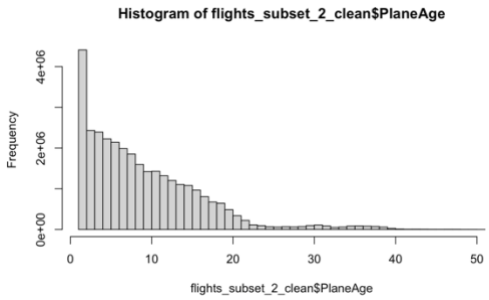
Missing values are also identified for delay variables, hence we impute them with zeroes. A pivot table summarizing the mean delay for each delay variable is generated to investigate the presence of NAs within these variables. The table indicates that mean delays are consistently zero from 1998 to 2002, while the values for 2003 differ from those of 2004 to 2007. This discrepancy suggests a possible reason: delays for each reason were not recorded before 2004.

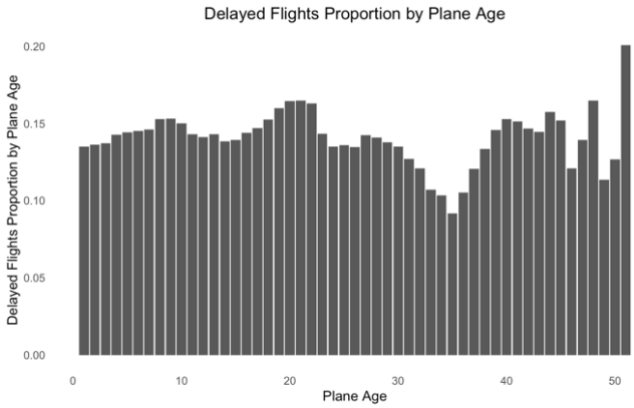
Year <int>	CarrierDelay_Mean <dbl>	NASDelay_Mean <dbl>	SecurityDelay_Mean <dbl>	WeatherDelay_Mean <dbl>
1998	0.000000	0.000000	0.00000000	0.0000000
1999	0.000000	0.000000	0.00000000	0.0000000
2000	0.000000	0.000000	0.00000000	0.0000000
2001	0.000000	0.000000	0.00000000	0.0000000
2002	0.000000	0.000000	0.00000000	0.0000000
2003	1.343892	1.860594	0.01265166	0.3116771
2004	2.695139	3.502191	0.02564770	0.7209017
2005	3.066259	3.438686	0.02016995	0.6740737
2006	3.463124	3.656690	0.03160729	0.6930303
2007	3.959775	3.876239	0.02431610	0.7889269

Table 1: Mean delay for delay variables grouped by year

3.2 Data Exploration

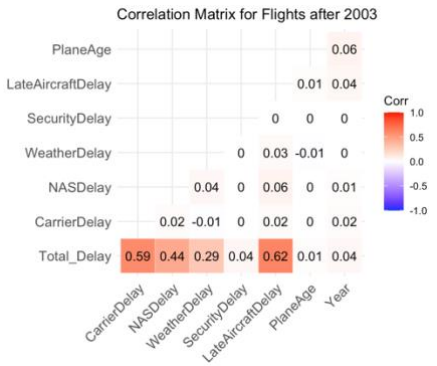
Histograms are plotted for the variables to observe their distributions. A discrepancy is noted in the distribution of plane age, as it includes instances of negative ages and aircraft ages ranging from 2003 to 2007, which is not practically feasible. Further investigation reveals that the negative ages stem from zeroes in the manufactured year variable, possibly due to data entry errors. Once again the year from the issue date will be substituted for the zero values. Flights with negative or zero ages will be excluded from analysis since there are no alternative methods to determine their real age. Another histogram is plotted to ensure that there are no discrepancies in the plane age variable, while also observing its distribution. The resulting histogram shows that the frequency for newer aircraft is higher than for older aircraft, which is expected as older aircraft are often replaced by newer ones. To ensure the accuracy of our analysis, a contingency table is generated to verify that there are enough observations for each aircraft age.





To investigate whether older aircraft experience higher delay proportions, the data is grouped by plane age and the delay proportion is computed. The results are visualized using a bar plot. However, the bar plot does not reveal clear patterns indicating that older aircraft result in higher delay proportions. Further analysis is required to draw conclusive insights.

Pearson's correlation coefficient



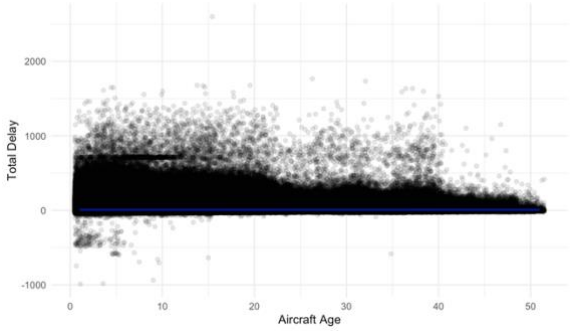

```
        Pearson's product-moment correlation

data:  selected_column$PlaneAge and selected_column$Total_Delay
t = 34.138, df = 20968017, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.007026923 0.007882926
sample estimates:
       cor
0.007454926
```

Considering that delay variables were not recorded before 2004, a correlation matrix is plotted for data after 2003 to examine the relationships between variables. The Pearson's correlation coefficient value of 0.007 between plane age and total delay suggests a very weak positive relationship. The p-value being less than 0.05 indicates that the relationship is statistically significant. However, while there is a statistically significant correlation between both variables, the correlation is very weak, suggesting that there is little practical significance to the relationship between them. Furthermore, moderate to strong correlations between total delay and both carrier and late aircraft delay are observed. This suggests that factors relating to the carrier or aircraft may be stronger predictors for delay.

3.3 Analysis and Data Visualization

Linear regression and Scatter plot

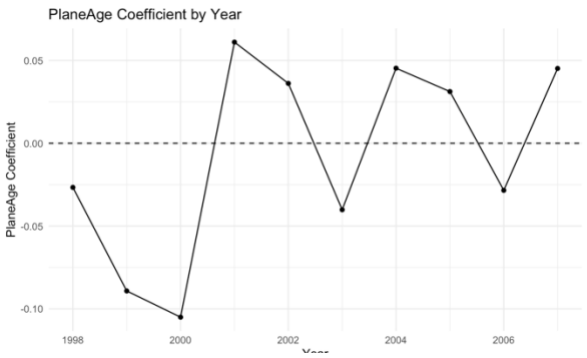
```
Call:
lm(formula = Total_Delay ~ PlaneAge, data = flights_subset_2_clean)

Residuals:
    Min      1Q  Median      3Q     Max
-997.62  -14.22   -9.16    0.36 2591.21

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 8.1091290  0.0094944  854.10   <2e-16 ***
PlaneAge    0.0118215  0.0007999   14.78   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 32.68 on 32179310 degrees of freedom
Multiple R-squared:  6.788e-06, Adjusted R-squared:  6.757e-06
F-statistic: 218.4 on 1 and 32179310 DF,  p-value: < 2.2e-16
```



The R-squared value of 0.000006 suggests that plane age explains only 0.0006% of the total delay variance. Therefore, we conclude that plane age, as a variable, is not a strong predictor of total delay. The coefficient of 0.01 indicates a very weak relationship, suggesting that total delay increases by only 0.01 minutes when plane age increases by one unit. However, the associated p-value, being less than 0.05, suggests that plane age does have a statistically significant effect on predicting total delay. Nevertheless, this effect is very weak. The blue line on the scatter plot illustrates the relationship between both variables. However, due to the weak relationship, the gradient of the line cannot be observed.

Do older aircraft tend to cause more delays from year to year?

To address this question, we create a line graph illustrating the coefficients of plane age over the years to assess whether older aircraft contribute to more delays over time. We achieve this by iterating through different yearly datasets, storing coefficient values in a data frame, and using this data to plot the line graph. The line graph reveals fluctuations in the coefficient: in some years, the relationship between plane age and delays is negative, while in others, it is positive. However, the coefficient values consistently remain small and almost negligible throughout the years.



Conclusion

The study suggests that older planes do experience more delays overall, however the relationship is very weak. Other factors relating to Carrier or Aircraft are more likely to play a more substantial role in determining total delay. As nearly 50% of the information is omitted due to the absence of plane information during the data cleaning process, the assumption is that this missing data does not contain any crucial information required for this analysis.

2c) For each year, fit a logistic regression model for the probability of diverted US flights using as many features as possible from attributes of the departure date, the scheduled departure and arrival times, the coordinates and distance between departure and planned arrival airports, and the carrier. Visualize the coefficients across years.

This section aims to develop a logistic regression model for predicting diverted flights, focusing on simplicity through feature selection and multicollinearity detection. Initially, we'll include all potentially relevant features and assess multicollinearity using the Variance Inflation Factor (VIF). Subsequently, we'll use chi-square tests and ANOVA to examine highly correlated variables and eliminate redundant ones. Next, we'll evaluate the remaining features' importance using the Area Under the Curve (AUC) metric, assessing their ability to distinguish between diverted and non-diverted flights. Regularization will then be applied by tuning hyperparameters alpha and lambda to potentially improve AUC scores. Finally, we'll visualize the coefficients of the final model and perform sensitivity analysis on sensitivity and specificity versus threshold values. This systematic approach aims to create an effective and interpretable logistic regression model for predicting flight diversions, while adhering to the principle of parsimony.

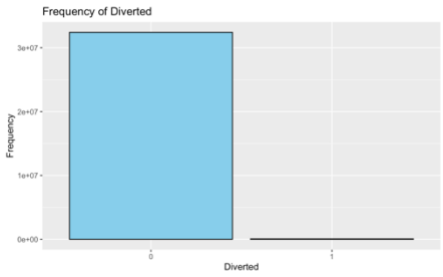
4.1 Data Preparation, Cleaning and Wrangling

Predictors	Description	Data Type
Distance	Distance between origin and destination airport.	int
DepartureInterval	Scheduled Departure time grouped into 6 intervals of 4 hours.	category
ArrivalInterval	Scheduled Arrival time grouped into 6 intervals of 4 hours.	category
Month	Month of flight	category
DayOfWeek	Day of flight (Mon, Tue, etc)	category
Carrier	Carrier that operates the flight	category
lat_Origin	Latitude coordinate of origin airport	float
long_Origin	Longitude coordinate of origin airport	float
lat_Dest	Latitude coordinate of destination airport	float
long_Dest	Longitude coordinate of destination airport	float
Year	Year of flight	int
engine_type	Type of engine	category
aircraft_type	Type of aircraft	category
manufacturer	Manufacturer of the aircraft	category
PlaneAge	Age of aircraft	float

Table 2: Table of predictors that are selected initially

The data preparation stage involves cleaning and wrangling the data before model training. Initially, the flights data is merged with supplementary data containing Carrier, Airport, and plane information. Airports information is merged twice, once for destination and once for origin, using different keys for each merge. Variable names are then renamed to improve interpretability. Subsequently, missing values in the plane information are addressed using methods similar to those employed in previous cleaning processes. To simplify categorical variables like 'engine_type', 'manufacturer', and 'aircraft_type', which contain numerous groups, we group them into smaller categories based on frequency, facilitating clearer interpretation of results. Finally, all variables are converted to the appropriate data type to ensure compatibility for model training.

4.2 Data Exploration and Feature Selection



Value <fctr>	Frequency <int>
0	32417158
1	70169

Table 3: Diverted Contingency Table

Histograms and Kernel Density plots are generated for all variables to examine their distributions and identify any potential outliers that may require further investigation. A particular focus is placed on the outcome variable 'Diverted', which is found to be heavily imbalanced, with diverted flights accounting for only 0.002% of the dataset. Given this severe class imbalance, accuracy is deemed an inadequate evaluation metric, as a model could achieve high accuracy simply by predicting the majority class. Therefore, the Area Under the Curve (AUC) is chosen as the evaluation metric. AUC considers both sensitivity and specificity, providing a more comprehensive assessment of the model's performance, particularly in imbalanced datasets.

<u>VIF, Chi-square Test, ANOVA and Cramer's V Results</u>

Given the presence of high VIF values across various variables, subsequent analyses revealed significant associations ($p < 0.05$ at a 95% confidence level) between 'Carrier' and other variables exhibiting high VIF. Notably, Cramer's V values exceeded 0.25, indicating strong associations. In light of these findings, it is recommended to retain only 'Carrier' and 'DepartureInterval' among the variables with high VIF values.

	GVIF	Df	GVIF^(1/(2*Df))
Distance	1.798902	1	1.341232
DepartureInterval	7.946266	5	1.230315
ArrivalInterval	8.013087	5	1.231346
Month	1.009556	11	1.000432
DayOfWeek	1.003859	6	1.000321
Carrier	2243.128542	20	1.212750
lat_Origin	1.207190	1	1.098722
long_Origin	1.531860	1	1.237683
lat_Dest	1.195817	1	1.093534
long_Dest	1.469125	1	1.212075
Year	1.553209	1	1.246278
engine_type	15.251350	3	1.574773
aircraft_type	4.168982	1	2.041808
manufacturer	800.285383	5	1.951302
PlaneAge	3.704998	1	1.924837

Table 4: VIF results

<u>Feature Selection and Hyperparameter Tuning</u>

Given the large dataset size, a smaller 5% sample is chosen for model training and testing to ensure efficiency without compromising accuracy. This sample is then further split into 70% for training and 30% for testing, stratified by the 'Diverted' variable to address its imbalance.

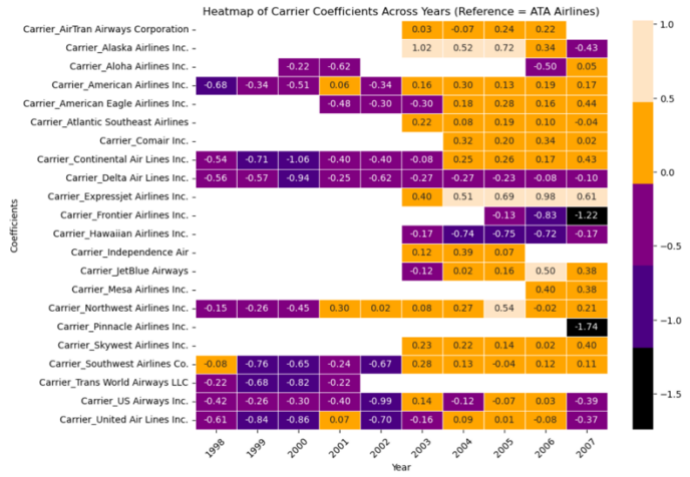
Model Type	Features	Cross Validation AUC
Model with single feature	Distance	0.616
	DepartureInterval	0.520
	Month	0.564
	DayOfWeek	0.501
	Carrier	0.567
	lat_Origin	0.505
	long_Origin	0.525
	lat_Dest	0.508
	long_Dest	0.524
Model with all but 1 features	Distance removed	0.593
	DepartureInterval removed	0.628
	Month removed	0.612
	DayOfWeek removed	0.630
	Carrier removed	0.623
	lat_Origin removed	0.627
	long_Origin removed	0.630
	lat_Dest removed	0.630
	long_Dest removed	0.629
Full Model	All Features	0.635
Partial Model (Selected Features)	Distance, Month, Carrier, long_Dest, long_Origin	0.630

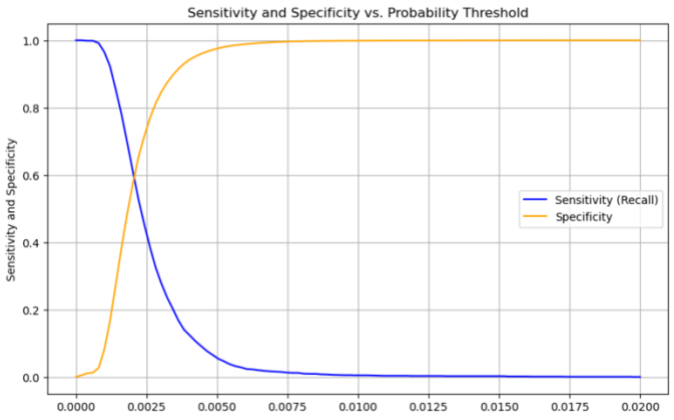
Table 5: Summary of AUC scores

To evaluate the importance of each feature, two methods are employed. Firstly, logistic models are fitted individually for each feature, and cross-validation is performed to calculate the mean AUC scores. Secondly, logistic models are fitted for feature sets containing all predictors except one, and cross-validation is again employed to compute mean AUC scores. These scores are compared to a value of 0.5 to assess if the model features perform better than chance. Functions are created for each method, iterating through each predictor and performing cross-validation. The computed AUC scores are then stored in a dataframe. The performance of all models is summarized in **Table 5**. The selection of features is based on their individual AUC scores and their scores when removed from the full model. 'Distance', 'Carrier', and 'Month' emerge as the top three features, as they have the highest AUC values for the single feature set, and their removal from the full model results in a significant drop in AUC scores. Both longitude variables also demonstrate moderate performance. 'DayOfWeek', 'DepartureInterval', and both latitude variables are excluded due to their marginal explanatory power, and their interaction with other variables does not improve their explanatory power, as evidenced by the AUC scores when they are removed from the full model. Finally, a logistic model is fitted for the remaining variables, achieving an AUC value of 0.63, which is close to that of the full model's AUC value of 0.635.

A machine learning pipeline is established, which involves scaling numeric variables and encoding categorical variables. Subsequently, a grid search is conducted to optimize lambda and alpha parameters (logistic_C for Python), with the objective of maximizing AUC scores. However, the tuning process reveals that AUC scores remain unchanged across different levels of alpha and lambda values. This suggests that employing a penalized model does not enhance the model's effectiveness in predicting flight diversions. Hence, we will employ general logistic regression for the final model. The final trained model consisting of the selected feature performed notably well on the test set, achieving a test AUC value of 0.626.

4.3 Data Visualization





A heatmap was created to visualize the coefficients of 'Carrier' across different years. The process involved extracting the coefficients similar to how the coefficient of 'PlaneAge' was obtained in part (b). The heatmap shows an interesting finding that the coefficients for most carriers are negative before 2003 with ATA airlines as reference. This suggests that most carriers are more likely to experience diversions compared to ATA airlines after 2003. Additionally, a sensitivity analysis was conducted to visualize the trade-off between sensitivity and specificity. The final model reached a balanced performance by adjusting the classification threshold to approximately 0.002. This adjustment led to both sensitivity and specificity levels reaching around 60%.

References

Wickham, Hadley. n.d. *R for Data Science (2e).* https://r4ds.hadley.nz.

Mckinney, Wes. 2023. *Python for Data Analysis, 3e.* https://wesmckinney.com/book/.

Bernd Bischl, Raphael Sonabend, Lars Kotthoff, Michel Lang. n.d. *Applied Machine Learning Using mlr3 in R.* https://mlr3book.mlr-org.com.

Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. 2021. *An Introduction to Statistical Learning with Application R, Second Edition.* Springer.

2024. *Scikitlearn.* https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html.

LLC, Statcorp. 2016. *Introduction to Bayesian statistics, part2: MCMC and the Metropolis-Hastings algorithm.*