ST2195 Programming for Data Science
Coursework Project: Helena Jackson

Contents

0.1 General Comments on the Data and pre-analysis Observations

This report is prepared based on "Data Expo 2009: Airline on time data" from the Harvard Dataverse. Years 2003-2005 inclusive are selected[1] to form the basis of this analysis.

This dataset is composed of the following variables:[2]

Variable	Description, values taken	Variable Type
Year	2003-2005	Numerical
Month	1-12	Numerical
DayofMonth	1-31	Numerical
DayOfWeek	1 (Monday) - 7 (Sunday)	Numerical
DepTime	Actual departure time (local, hhmm)	Numerical
CRSDepTIme	Scheduled departure time (local, hhmm)	Numerical
ArrTime	Actual arrival time (local, hhmm)	Numerical
CRSArrTime	Scheduled arrival time (local, hhmm)	Numerical
UniqueCarrier	Unique carrier code	Categorical
FlightNum	Flight number	Categorical
TailNum	Plane tail number: aircraft registration, unique aircraft identifier	Categorical
ActualElapsedTime	Actual elapsed time in minutes	Numerical
CRSElapsedTime	Scheduled elapsed time in minutes	Numerical
AirTime	In minutes	Numerical
ArrDelay	Arrival delay in minutes. A flight is counted as "on time" if it operated less than 15 minutes later the scheduled time shown in the carriers' Computerized Reservations Systems (CRS).	Numerical
DepDelay	Departure delay, in minutes	Numerical
Origin	Origin IATA airport code	Categorical
Dest	Destination IATA airport code	Categorical

[1] Originally, years 2001-2003 inclusive were selected. However, the impact of 9/11 on flights during 2001 was “a 31.6 percent reduction in travel volume in September of 2001 compared to that same month in 2000” (D. E. Clark, J M. McGibany & A. Myers from M. J. Morgan (ed.), The Impact of 9/11 on Business and Economics). Even when the data was analysed across three years, the drop in the total number of flights in September of 2001 was so pronounced that conclusions drawn across the entire dataset were significantly skewed. In the interests of using the data to predict delays in ‘normal’ flight paths - i.e. outside the scope of such unusual and tragic events, years 2003-2005 were selected instead.

[2] Variable description found in the Harvard Dataverse additional files

Distance	In miles	Numerical
TaxiIn	Taxi in time, in minutes	Numerical
TaxiOut	Taxi out time in minutes	Numerical
Cancelled	Was the flight cancelled? Binary 0=no, 1=yes	Numerical
CancellationCode	Reason for cancellation (A = carrier, B = weather, C = NAS, D = security)	Categorical
Diverted	Was the flight diverted? 1 = yes, 0 = no	Numerical
CarrierDelay	In minutes. Carrier delay is within the control of the air carrier. Examples of occurrences that may determine carrier delay are: aircraft cleaning, aircraft damage, awaiting the arrival of connecting passengers or crew, baggage, bird strike, cargo loading, catering, computer, outage-carrier equipment, crew legality (pilot or attendant rest), damage by hazardous goods, engineering inspection, fueling, handling disabled passengers, late crew, lavatory servicing, maintenance, oversales, potable water servicing, removal of unruly passenger, slow boarding or seating, stowing carry-on baggage, weight and balance delays.	Numerical
WeatherDelay	In minutes: Weather delay is caused by extreme or hazardous weather conditions that are forecasted or manifest themselves on point of departure, enroute, or on point of arrival.	Numerical
NASDelay	In minutes: Delay that is within the control of the National Airspace System (NAS) may include: non-extreme weather conditions, airport operations, heavy traffic volume, air traffic control, etc.	Numerical
SecurityDelay	In minutes: Security delay is caused by evacuation of a terminal or concourse, re-boarding of aircraft because of security breach, inoperative screening equipment and/or long lines in excess of 29 minutes at screening areas.	Numerical
LateAircraftDelay	In minutes: Arrival delay at an airport due to the late arrival of the same aircraft at a previous airport. The ripple effect of an earlier delay at downstream airports is referred to as delay propagation.	Numerical

Three consecutive years were selected on the basis that, should there have been any other ‘freak’ events (weather, political events, natural disasters) that affected a portion of the data in one year, there is sufficient unaffected data to neutralise the effect and provide a more unbiased analysis.

The purpose of this report is to answer each of the five questions below, which primarily explore delays in flights across the US from 2003-2005 .I explain my interpretation and decision-making to answer each question in R / Python sequentially. I begin with some general observations about the data.

0.1.1 Definition of Delay

In the data, a flight is counted as "on time" if it operated less than 15 minutes later than the time scheduled in the Computerised Reservations Systems (CRS). Delays in the dataset exist in the form of Departure Delays and Arrival Delays.

Plotting mean departure delays and mean arrival delays for each carrier(below), note that delays at arrival are generally lower than at departure. Flight carriers must adjust their speed to make up time on arrival. It therefore makes sense to focus on arrival delays - this is the factor most likely to affect passengers.

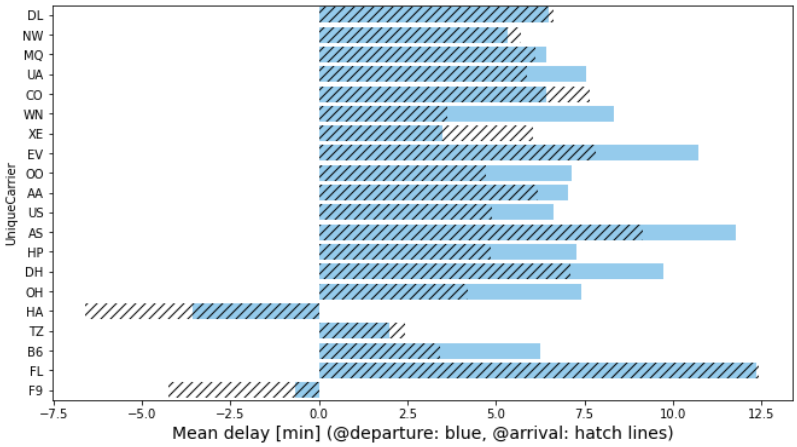


Fig. I: Barplot of departure delays(blue) vs arrival delays (hatched) by aircraft carrier.

Clearly, delayed departures do not necessarily equate to delayed arrival.

For this reason, I will focus primarily on Arrival Delays in this analysis.

0.2.0 Data Wrangling, Cleaning and Pre-Processing

Initial data importing, descriptive statistics and sampling was carried out in Python, and then the structures imported into R (e.g. the sample dataset to ensure consistent analysis and visualisation). Code is included in the R Script to construct a random sample of n=10,000 to ensure replicability but, for practical purposes, I have used the sample created with Python in my own R code analysis. The csv files of flight data in 2003-2005 are imported. Once it has been verified that the column structures are uniform across all three**,** the data are concatenated into an aggregated (pandas) dataframe we will call **'flights'**.

0.2.1 A brief summary of the large dataset

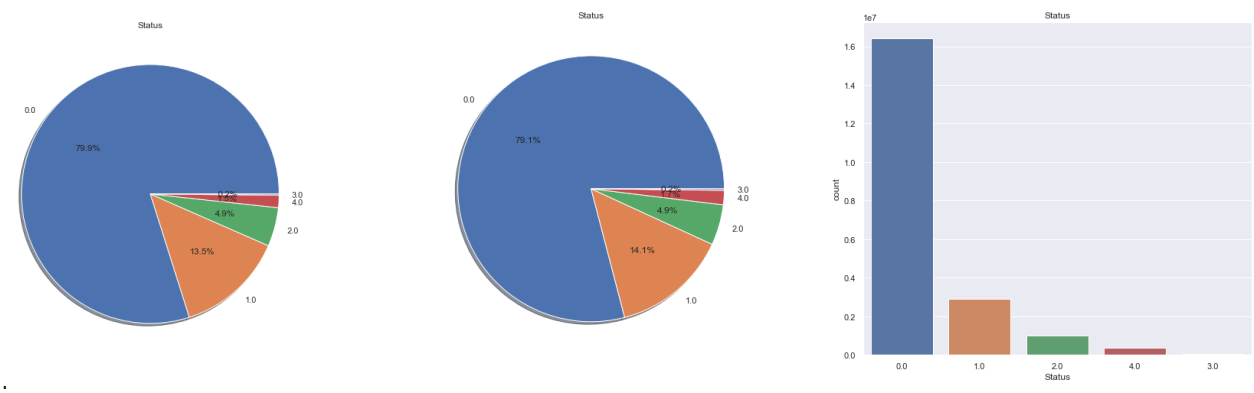
'flights' is a pandas dataframe consisting of objects, integer and float-type data. It consists of 20,758,406 items and uses more than 4GB of memory - a data set of this size will run tasks slowly and execute them using a significant amount of processing power and RAM. We will take a sample of appropriate size to run analyses more easily, while ensuring data completeness / lack of noise as far as possible.

0.2.2 Taking an appropriate sample

Take a random sample of 10,000 values with replacement to ensure that each selection is independent of the next. Does a sample with n = 10,000 ensure complete enough data while reducing the processing power sufficiently?

0.2.3 Consider delays: it is sensible to evaluate the sample suitability based on its fit of delay proportions to the population data. If the sample presents delay proportions that are a close fit to the population proportions we will accept that it is sufficient for our analyses. 79.1% of flights in **flights** arrived on time or with a delay of fewer than 15 minutes. We create a 'Status' measure to represent whether the flight was on time (0), slightly delayed (1), highly delayed (2), diverted (3), or cancelled (4).

Fig. II & III: comparison of Delay Status in 'flights' and 'sample' reveals the proportions of delay types are very similar



.

0.2.4 Numerical justification of use of sample with a Chi-Squared Goodness-of-fit test:
Constructing a Chi-Squared Goodness-of-fit test:

H_0: a random sample of n=10,000 is a suitable sample from which to carry out inference on the population
H_1: a random sample of n=10,000 is not a suitable sample from which to carry out inference on the population

Expected values = means of 17 numerical variables in the 'flights' dataset
Observed values = means of 17 numerical variables in the 'sample' dataset

With $\chi 2 = \sum(Oi - Ei)2/Ei$
We find $\chi 2$ for p = 0.999, dof = n-1 =16 is 0.0714. This value is clearly well below the critical threshold value of 5.142. We therefore accept the null and proceed with the choice of sample.

0.2.5 Dealing with missing values

We first find the total number of NA or NULL values in the sample, and then find NAs as percentages of each column. The larger the proportion of NAs, the more noise will be added by imputing or filling missing values.

There is a low proportion of mission values in most columns (2% missing in columns "DepTime", "ArrTime", "ActualElapsedTime", "AirTime", "ArrDelay" & "DepDelay"). 13% of Delay reasons, categorised into "CarrierDelay", "WeatherDelay", "NASDelay", "SecurityDelay". "LateAircraftDelay" respectively missing. These are all numerical variable and so we will impute them - comparison reveals that the mean and median are generally similar, so we will use the mean to impute. SimpleImputer or .fillna() are both methods that could be used for this.

9 values are missing from the 'TailNum' column; we will impute them with the modal item in this column (categorical).

A very high proportion in 'Cancellation Code' (99%). Drop this column; there is no sensible method of imputing with nearly all values missing. Imputed values would introduce only noise, not preserve any information.

Then write the sample to csv to use for subsequent analysis in R.

1.0 When is the best time of day, day of the week, and time of year to fly to minimise delays?

1.1 Arrival Delays

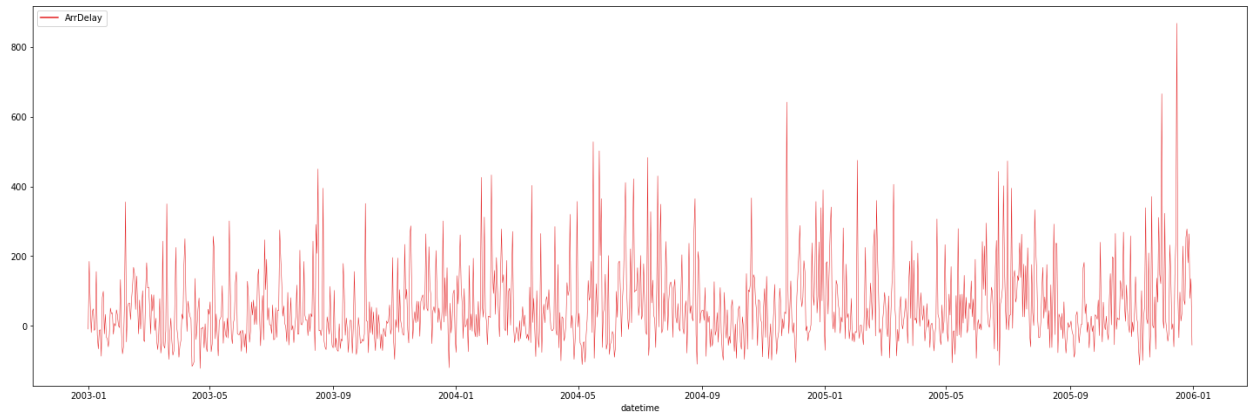
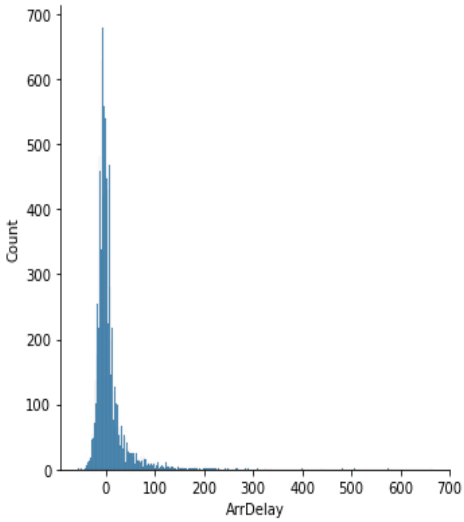


Fig. IV: Arrival Delay counts over the time period covered in the sample data.

1.2 Distribution and Characteristics of Arrival Delays



The majority of delays are short - this fits with our earlier finding that approximately 79.1% of flights in 'flights' are on time, and 79.9% in 'sample' are ontime. The median Arrival Delay is *-1* minutes - by percentile count, flights actually arrive *early* on average.

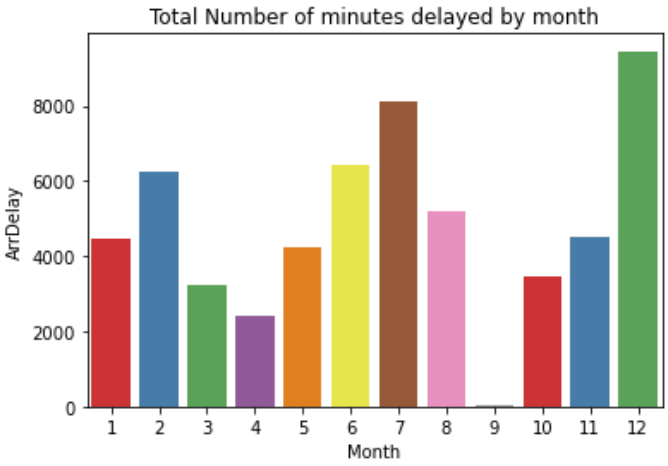
The longer delays, while unusual, skew the distribution considerably. Skewness = 5.532881- delay times are heavily right-skewed.

Fig V: distribution of Arrival Delay times in minutes

1.3 Delays by month:

The Months with the highest delays and total minutes delayed are July, August and December. Peak holiday seasons at both axes of the year perhaps unsurprisingly increase demand for internal flights. A higher total volume of flights is likely to lead to higher numbers of delayed flights, and longer delay times on average.

The month with the lowest average delay and total number of delays is September, by a considerable margin.



1.3.1 Delays by Month: Conclusion: 'Off-season' months present the lowest average delay and total delay time. September has the lowest total delay time and variance in delay times. Conclude that March is the best month in which to fly to minimise delays.

1.4 Delays by Day

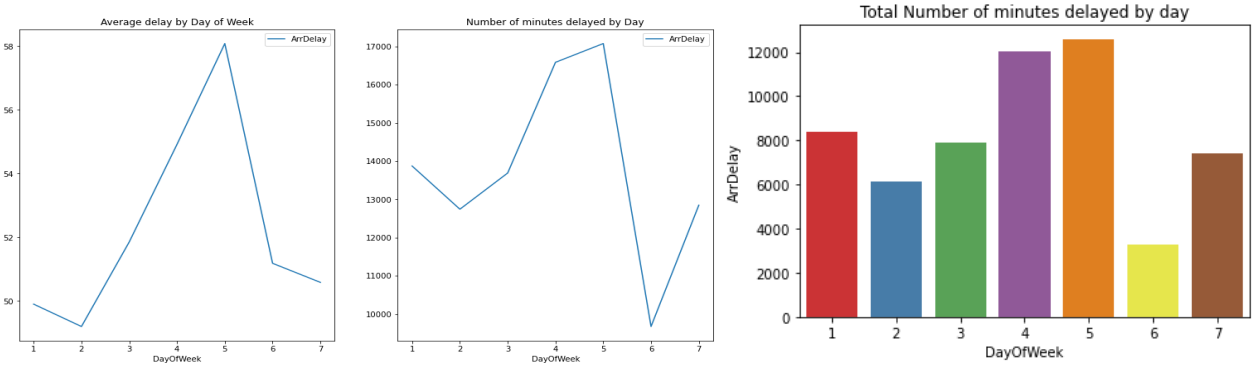


Fig. VII, VIII: Average delay per Day Of Week and total minutes delayed by Day of Week across the sample

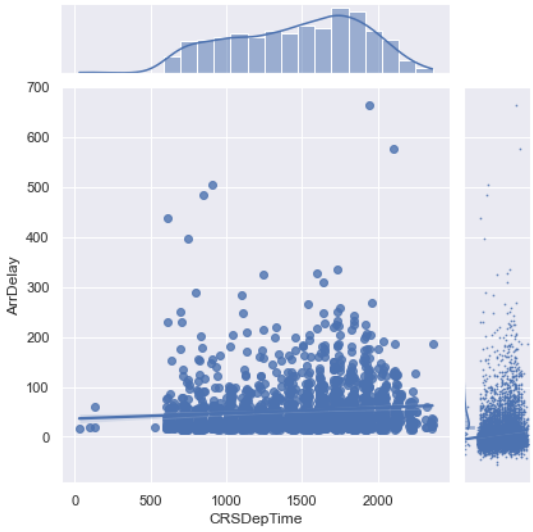
Day of Week	1	2	3	4	5	6	7
Variance in ArrDelay Totals	1664	2213	2429	2724	3128	3224	3799

Friday suffers the highest average and total delays and has the greatest variance in delay times. Saturdays have the lowest average and total delay times over the three years, but variance in delay times across all flights on Saturdays is high. On *average*, the delay may be low but this could be influenced by extreme(ly low or high) data values. We cannot say with certainty that flights on Saturday will *always* have low delay times.

Sunday has similarly low average delay (1 minute more on average than Saturday), and the lowest variance in delay times of all days, but the total number of delayed minutes on Sundays is high. This could be explicable - e.g. there are on average more flights on Sundays than Saturdays. Even though average delay times are very similar, the aggregate increase in flight volume leads to a higher sum of minutes of delay.

1.4.1 Conclusion: Saturday is the best choice, but may be subject to fluctuations. Avoiding Friday and midweek flights is certainly the best way to minimise delays by day.

1.5 Delays by Time of Day (Hour)



The CRSDepTime, the scheduled departure time,is measured against Arrival delays.

It is harder to produce a specific ‘recommendation’ in this case: there are thousands of unique Departure Times across the three year period. Attempting to reproduce the same table of average / total / variance for delays yields a much larger, more complicated and less useful table. We cannot draw conclusions from it as with our other data.

1.5.1 Conclusion: Plotting the variables with a regression line, we note that the average delay tends to increase with the departure time of day. Flights leave on time in the morning and delays grow cumulatively - so take an AM flight to minimise the chances of delay.

Fig. IX: scatter plot of scheduled departure times vs arrival delays with regression line

2.0 Do older planes suffer more delays?

Exploration of this question uses to the **sample** dataset of 10,000 values as well as the **plane_data** dataset provided by the Harvard Dataverse. **plane_data** provides information about each of 4480 unique aircraft tail numbers. It contains the variables ‘type’, ‘manufacturer’, ‘issue_date’, ‘status’ (valid tail number or not), ‘aircraft type’, ‘engine type’ and ‘year’ [of manufacture].

For the purposes of this question, we need only to refer to the age of the aircraft and how it relates to delays. The **sample** and **plane_data** data frames are merged on the intersect of their respective ‘TailNum’ columns. In Python, the following dataframes are used to analyse the relationship between aircraft age and arrival delays:

tailnums:
A dataframe to identify the flights recorded in **sample** with the unique tail number of each aircraft and information about its issue date and year.

Head of tailnums dataframe

datetime	UniqueCarrier	TailNum	ArrDelay	Status	CarrierDelay	year
NaT	WN	N723	32	1	6	NaN
2001-12-27	CO	N57852	-6	0	0	2001
2002-07-01	XE	N12900	-37	0	0	2001
2007-01-10	AS	N782AS	-10	0	0	1994

tailnums_decades:
The dataframe produced when he **tailnums** dataframe is resampled and grouped by *decade* with sums of the delay totals to provide a 'top-level' view of any trends in the data:

Head of tailnums_decades

datetime	ArrDelay	Status	CarrierDelay
1976-01-01	812	24	521
1986-01-01	9235	401	3935
1996-01-01	26161	1130	11137

2.1 Visualisation of delays by aircraft age

Fig. X: Scatter plot of Arrival delay by plane manufacture year (in Python)

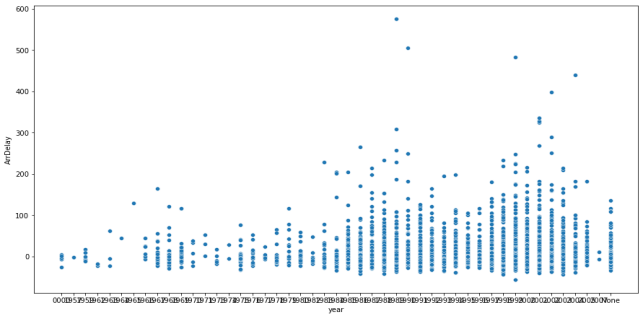
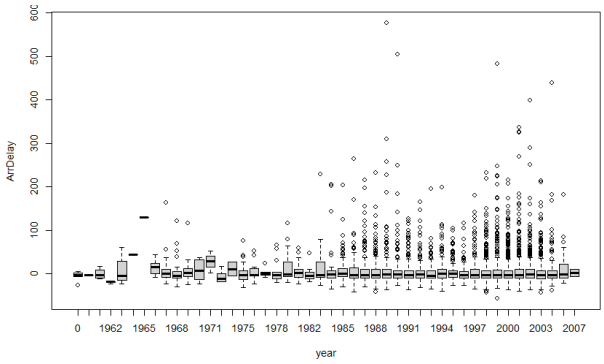


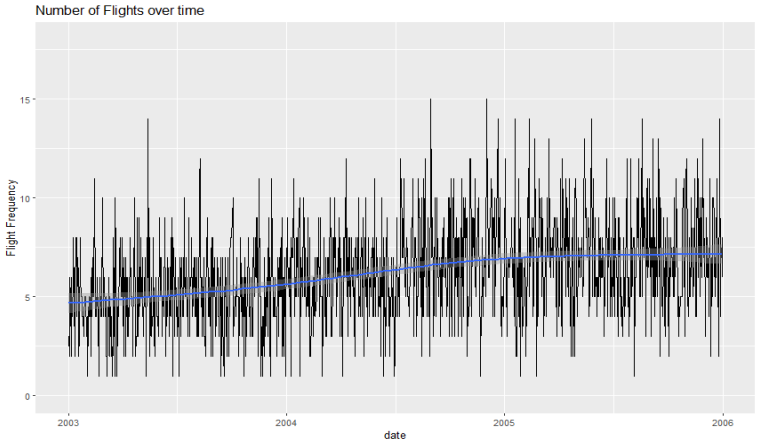
Fig. XI Box plot of Arrival delay by plane manufacture year (in R)



2.2 Conclusion

Surprisingly, the data appears to show, not that older planes suffer more delays, but that (relatively) *newer* planes suffer higher delay times. The boxplot diagram produced in R shows that, although the median delay times do not change drastically with newer aircraft ages, there is much more variation in delay times and higher proportions of 'extreme' high delay times. Why? We do not know, although, to offer one possible explanation, it is plausible that newer planes are more in demand and used more frequently. Higher usage rates of new planes could associate the newer planes with higher rates of delays. However, this is simply speculation. In conclusion, though, we definitely *cannot* conclude that older planes suffer more delays based on this analysis of the data.

3.0 How does the number of people flying between different locations change over time?



To answer this question, we begin by examining if the *total* number of flights changes over time.

Fig XII: Line plot of flight volume over the sample period in R. There is a clear increase in the frequency of flights over the three year sample period between all locations.

So - our conclusions about certain destinations should be made in the light of the fact that there was an *overall* increase in the demand for domestic flights in the USA between 2003 and 2005. I will aim to do this

by considering the *monthly* flight frequencies in several of the most popular airports to see if there are any cyclical trends in the data.

3.1 Popularity of Destinations and Departure Airports

There are 250 unique destinations in the sample dataset. The five most popular airports across the whole sample are ATL, ORD, DFW, LAX, IAH - both for departures and arrivals. I focus my analysis on the top 3 airports - ATL (Atlanta International, Georgia); ORD (Chicago O'Hare, Illinois); and DFW (Dallas Fort Worth, Texas).

Fig. XII: destination airports with highest number of flights

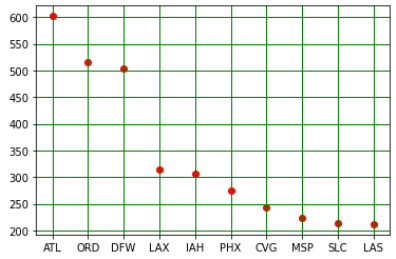
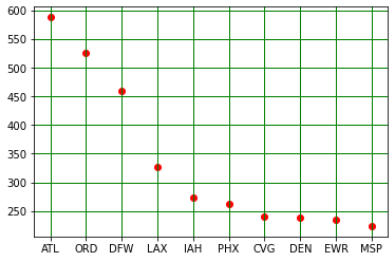
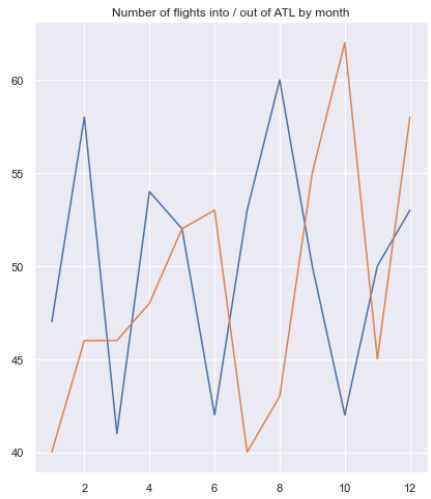


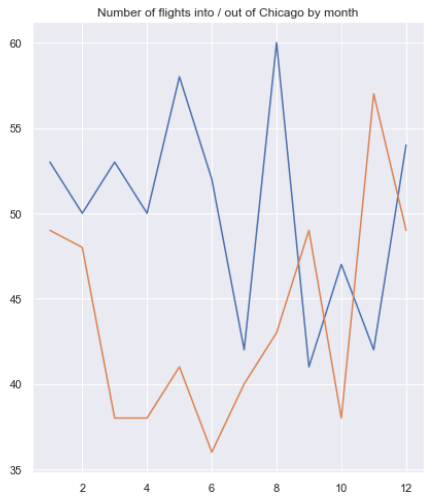
Fig. XIII: origin airports with highest number of flights

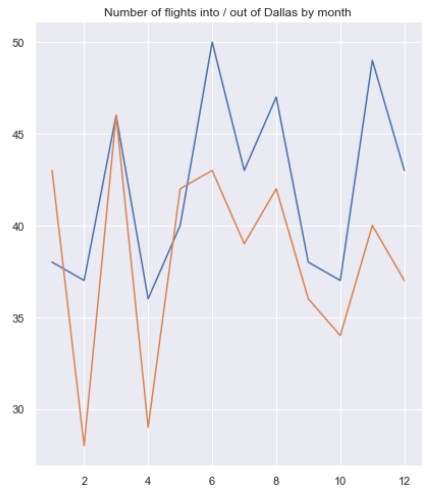


3.2 Comparing volume of flights by month

Each of the three graphs compare monthly volume of flights into and out of the three most popular airports in the sample dataset (blue = departures, orange = arrivals). Departures mostly closely match the arrivals by month in Dallas: are there any seasonal trends that might incite people to visit *and* leave Texas in certain months? As a southern state, probably popular with holidaymakers, it makes sense that the summer months and holiday periods would increase the number of arrivals. Departures also peak in these months - short holidays of less than a month will mean that flights into and out of Dallas are at a peak during this busy season. Conversely, arrivals in Chicago and Atlanta are at a yearly low during summer. In both cases, residents may be keen to holiday elsewhere.



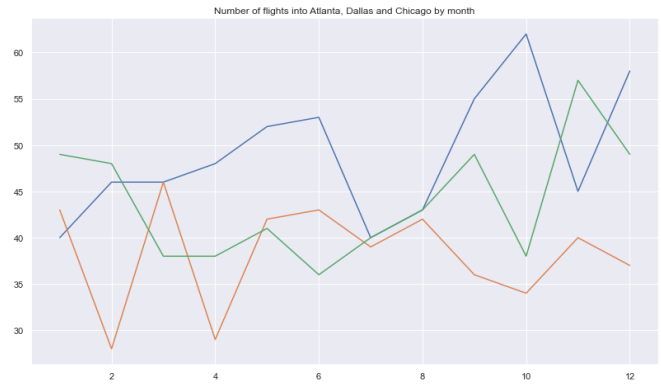




Finally, we consider the comparison of departures and arrivals across all three airports similarly. As we expect, volume fluctuates considerably by month. Thus, although these observations are by no means confirmed, the trends in monthly volume of arrivals and departures for these destinations fits into what we know and can learn about them. Flight volume is heavily seasonal, and is likely to be affected by the climate - and desirability - of each location across the USA over the year.

Fig. XVII: volume of departures from Atlanta, Dallas and Chicago by month
Blue = ATL, orange = DFW, green = ORD

Fig XVIII volume of arrivals to Atlanta, Dallas and Chicago by month.
Blue = ATL, orange = DFW, green = ORD



4.0 Can you detect cascading failures as delays in one airport create delays in others?

To answer this question, we consider the 'LateAircraftDelay' factor in the sample dataset. There are 145 instances of late aircraft delay (LAD) in the sample. We can also use the **tailnums** data frame created in question 2 to isolate the plane tail number in instances of LAD for onward delays.

We expect to find instances of cascading delays - whether specifically linked to LAD or not. In question 1, we highlighted that Arrival Delays tend to increase over the course of a day as delays accumulate. These delays could be due to any reason, but show that delays do influence one another.

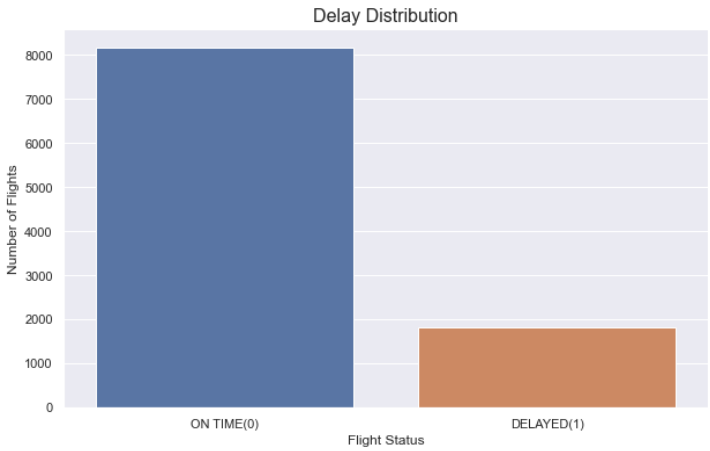
Fig. IXX: Delays plotted by category by month

Fig. IXX clearly shows a high proportion of delays are caused by delays from incoming aircraft, which fits our note on cumulative daily delays. If (as Q3 seemed to suggest), the *net* volume of air traffic increases in summer months, this would explain a higher volume of cascading delays.

Basic methodology to detect cascading delays could be to filter the tailnums dataset by instances of LateAircraftDelay, using the 'TailNum' index to find subsequent flights by the same plane on the given date.

If we can show a pattern of delays for a given 'TailNum' that suffers a delay at one airport, we can successfully detect cascading delays.

5.0 Use the available variables to construct a model that predicts delays.



Delay is a binary classification.

To view work on models using

- svm.SVR(),
- Logistic regression()
- DecisionTreeClassifier() - roc auc score of 95.3%
- linear_model.SGDRegressor(),
- linear_model.BayesianRidge(),
- linear_model.LassoLars(),
- linear_model.ARDRegression(),
- linear_model.PassiveAggressiveRegressor(),
- linear_model.TheilSenRegressor(),
- linear_model.LinearRegression()]
- Random Forest

Classifiers, see work in the Python notebook.