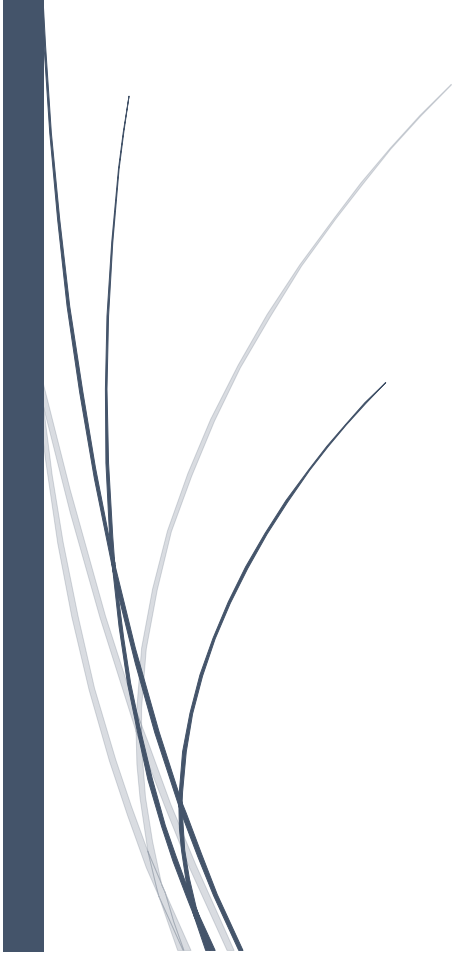
1/2/2022

PROJECT REPORT R-LANGUAGE:

Airline Dataset



Contents

Problem statement:

The airline dataset contains data of flights in the US and their Arrival and departure delay information.

We use the following data to find the expected delays in the arrival time of flights through methods of Data processing and analytics.

This information is helpful for airline carriers to better plan for the worst case scenarios, also the customers can better see which airline is the best.

Wonder question: Have you ever been stuck in an airport because your flight was delayed or canceled and wondered if you could have predicted it if you'd had more data?

Problem Type: Since we are interested in the arrival delay in minutes, we will use regression to predict the arrival delay which may be caused due to some

underlying factors like if there is a delay in departure of flight than it may have arrival delay too etc.

Techniques and strategy: We will be using correlation filter-based feature selection for our model; we will also be doing resampling using cross validation and will also be using sequential forward selection as a tool for feature selection. We will be using some graphical plots to visualize and understand the data.

About the data:

The data consists of flight arrival and departure details for all commercial flights within the USA, from October 1987 to April 2008. This is a large dataset: there are nearly 120 million records in total, but due to limitations of resources, we sampled the data from the years 2007 and 2008 only.

Dataset link:
https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/HG7NV7

Dataset attributes:

The dataset contain 29 attributes which are as follows.

```
 [1] "Year"              "Month"             "DayofMonth"        "DayOfWeek"         "DepTime"
 [6] "CRSDepTime"        "ArrTime"           "CRSArrTime"        "UniqueCarrier"     "FlightNum"
[11] "TailNum"           "ActualElapsedTime" "CRSElapsedTime"    "AirTime"           "ArrDelay"
[16] "DepDelay"          "Origin"            "Dest"              "Distance"          "Cancelled"
[21] "Diverted"          "CarrierDelay"      "WeatherDelay"      "NASDelay"          "SecurityDelay"
[26] "LateAircraftDelay"
```

We will be predicting the arrival delay for each flight using the attributes which are most important.

Data Preparation:

The data was divided into several csv's from year 1987 till 2008. We have chosen year 2007 and 2008 data for our analysis. The 2007 data consists of approximately 7 million observations and the 2008 data consist of approximately 2 million observations. We used data.table library to read our data fast using fread function.

We combined the 2 datasets into one and named it airline dataset.

Missing Values:

Data contains missing values in some of the columns. The percentage of missing values in each column is shown in the picture attached.

```
              Year              Month         DayofMonth          DayOfWeek            DepTime        CRSDepTime
       0.00000000         0.00000000         0.00000000         0.00000000         2.28795078         0.00000000
           ArrTime        CRSArrTime      UniqueCarrier          FlightNum            TailNum ActualElapsedTime
       2.51993613         0.00000000         0.00000000         0.00000000         0.00000000         2.51993613
  CRSElapsedTime            AirTime           ArrDelay           DepDelay             Origin               Dest
       0.01423429         2.51993613         2.51993613         2.28795078         0.00000000         0.00000000
        Distance             TaxiIn            TaxiOut          Cancelled  CancellationCode           Diverted
       0.00000000         0.71218170         0.65473655         0.00000000         0.00000000         0.00000000
     CarrierDelay       WeatherDelay           NASDelay      SecurityDelay LateAircraftDelay
       18.33524479        18.33524479        18.33524479        18.33524479        18.33524479
```

We can see that last 4 columns contains about 18% missing values; we can discard them. Also, column cancellation code has 97% empty spaces, so its better to discard this column.

Column Filter:

We have already removed cancellation code from the data, other than that taxi in and taxi out coloumns are also not required for our analysis so we will discard them. The final attributes are as follows.

```
 [1] "Year"              "Month"             "DayofMonth"        "DayOfWeek"         "DepTime"
 [6] "CRSDepTime"        "ArrTime"           "CRSArrTime"        "UniqueCarrier"     "FlightNum"
[11] "TailNum"           "ActualElapsedTime" "CRSElapsedTime"    "AirTime"           "ArrDelay"
[16] "DepDelay"          "Origin"            "Dest"              "Distance"          "Cancelled"
[21] "Diverted"          "CarrierDelay"      "WeatherDelay"      "NASDelay"          "SecurityDelay"
[26] "LateAircraftDelay"
```

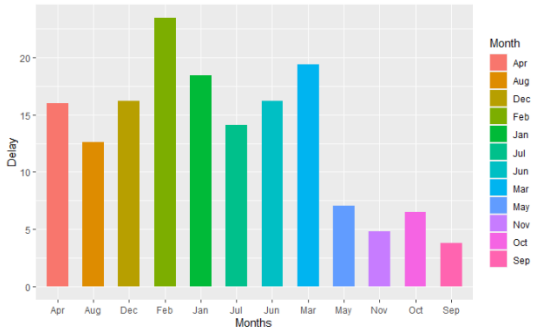
We discarded them using following line of code:

```
airline<-airline[,TaxiIn:=NULL]
airline<-airline[,TaxiOut:=NULL]
```

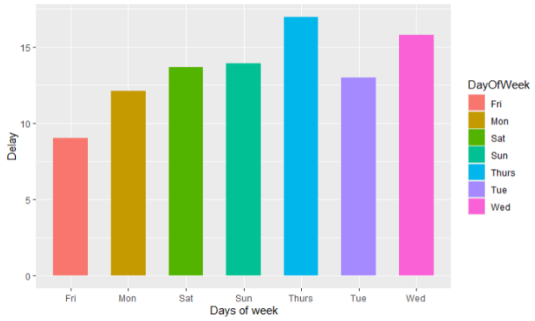
Answer to the Questions:

1. **When is the best time of day, day of the week, and time of year to fly to minimize delays?**
   For this we used summarize and group by functions from dplyr library to compute the mean arrival delay for each month, day of week, day of month and departure time. The one with the minimum mean would be the best month, year, day because the delay seems to be less in this period. Following are the graphs:

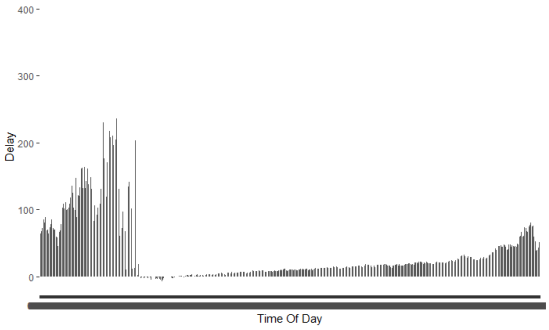
Months



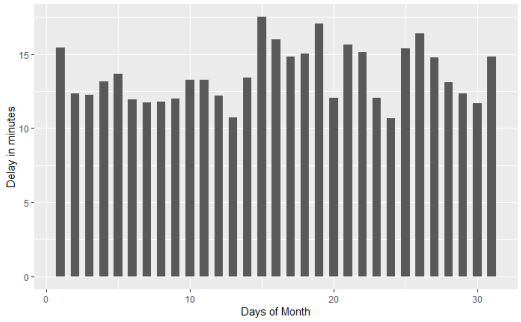
Day of week



Time of day



Day of month

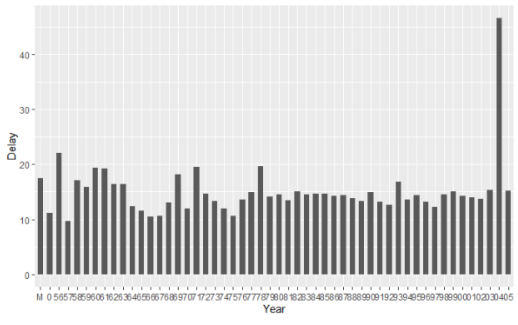


The first graph shows less average arrival delay in September while the second one shows less average arrival delay on Friday. The third one shows that the early hours (from midnight) shows higher delay than the later hours among which 5 am till 9 am is the best time where arrival delay is in negative. The fourth graph shows that 24th day of the month has the least average arrival delay. These are the month, day of week, time and day of month which are suitable to fly in.

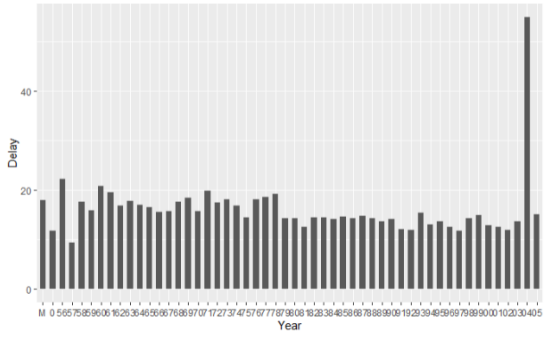
2. **Do older planes suffer more delays?**

   For this we joined plane data with airline data using the flights tail number. Then plotted the year of manufacture against the delays. We summarized the number of delays in the same manner as we did in question 1. Here, we fetched the mean arrival and departure delays for each year of manufacture. Following are the 2 graphs:

Departure Delay



Arrival Delay



From plot 1 we can see that the departure delay is fluctuating during the early years but after 1978 it declines and then it shows a consistency except for year 2004 where it is at its peak. Here we changed the levels of year as 56,57 etc. to have a clearer view of the graph.

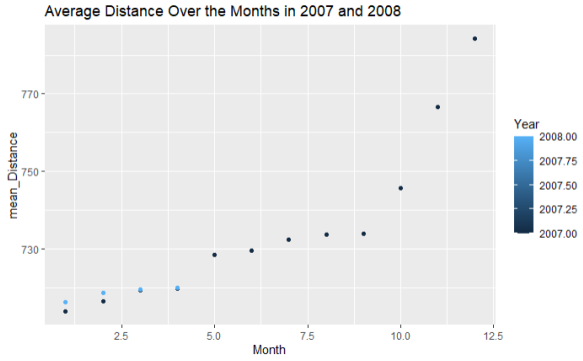
From plot 2 we can observe that the arrival delay is gradually decreasing after year 1980 which shows that older plains do have greater delays as compared to newer except for the plains which were manufactured in 2004 because their arrival delay rate is much higher.

The graph shows a highly gradual decline in the maximum arrival delay time for each day which translates older planes that have been issued earlier suffer more delays than newer planes.

3. **How does the number of people flying between different locations change over time?**
For this we visualize the change in distance, the people travelling and the number of flights over the years. The logic was to summarize the average distance, count of carriers and sum of distance for each month and year. This was done using summarize and group by functions of dplyr library.
    - Change in average distance over the years:

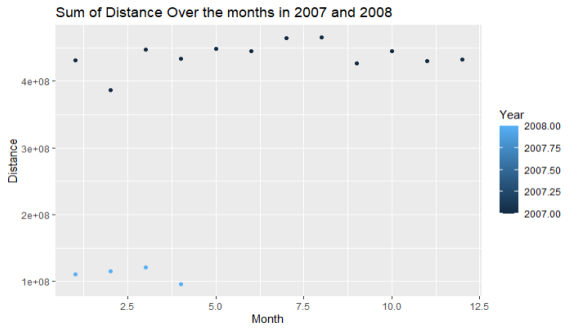
A tibble: 16 x 3 Groups: Month [12]

Month <chr>	Year <int>	mean_Distance <dbl>
Apr	2007	713.9315
Apr	2008	716.3321
Aug	2007	716.4162
Dec	2007	718.6711
Feb	2007	719.2449
Feb	2008	719.6551
Jan	2007	719.6724
Jan	2008	719.8923
Jul	2007	728.4205
Jun	2007	729.6443



From the table, we can see that the average distance has increased over the year. From the graph we can also see that the average distance is increasing.

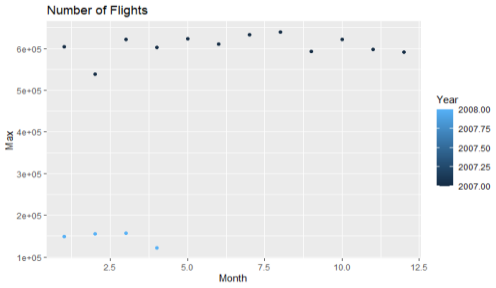
- Change in the number of people travelling:



We can see from the plot that the sum of distance has significantly decreased in 2008 as compared to 2007. This shows that less people travelled during this period.

- Change in the number of flights:

Month <int>	Year <int>	Max <int>
1	2007	604582
1	2008	148807
2	2007	538878
2	2008	156237
3	2007	621057
3	2008	157370
4	2007	602317
4	2008	122169
5	2007	623326
6	2007	609838



From the table and the graph, we can conclude that the number of flight also significantly declined over the years.

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

For this we subsetted the data with departure time greater than zero and then resubsetted this data where arrival time is greater than zero and then calculated the arrival vs departure percentage.

```
subset<- filter(airline,DepDelay>0)
arrdela<- table(subset$ArrDelay)
arrdela<- data.frame(arrdela)

subset2<- filter(subset,ArrDelay<0)
subset3<- filter(subset,ArrDelay>0)

print(paste0("The percentage of Departure Delays Causing Arrival Delays :",+nrow(subset3)/nrow(subset)*100,"%"))
print(paste0("The percentage of Departure Delays Not Causing Arrival Delays :",+nrow(subset2)/nrow(subset)*100,"%"))
```

```
| "The percentage of Departure Delays Causing Arrival Delays :80.8492803688307%"
| "The percentage of Departure Delays Not Causing Arrival Delays :17.2330668195408%"
```

There is a 80.849% chance that delay on the departure airport will create a delay on the arrival airport as well, therefore cascading failures do seem to exist.

Data Pre-Processing and Modelling:

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

The 5th and the last question is to create a model using the available variables. This part consists of 3 parts:

- Pre-processing.
- Modelling.
- Evaluation and findings.

Data Pre-processing:

The data pre-processing includes the down sampling of data, we picked only 5% of the data for our model because the data is so big. We already removed missing values and irrelevant columns. The next step was to only select integer or numeric columns for the modelling. Using mlr3 library we created a regression task out of our final data set with the following features:

```
[1] "ActualElapsedTime" "AirTime"          "CRSElapsedTime"    "Cancelled"         "CarrierDelay"
[6] "DayofMonth"        "DepDelay"         "Distance"          "Diverted"          "FlightNum"
11] "LateAircraftDelay" "NASDelay"         "SecurityDelay"     "WeatherDelay"      "Year"
```

The target attribute is Arrival delay.

We also split our task in 70/30 test and train data.

```
# Train test split
train = sample(task$nrow, 0.7 * task$nrow)
test_set = setdiff(seq_len(task$nrow), train)
```

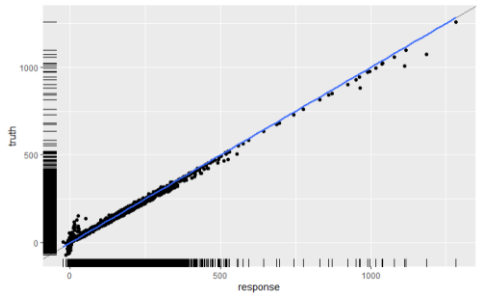
The next step was the feature selection, we used filter based approach to pick out the best features for our data. We used correlation as a tool to judge the predictors importance towards the target variable which is Arrival delay. Following are the results of correlation:

feature <chr>	score <dbl>
DepDelay	0.9355320217
LateAircraftDelay	0.6073489683
CarrierDelay	0.5726728436
NASDelay	0.4863730001
WeatherDelay	0.2805770195
Year	0.2802342964
ActualElapsedTime	0.0908877743
FlightNum	0.0316333866
AirTime	0.0278787379
SecurityDelay	0.0272953321

We selected attributes whose score is greater than 0.2.

Model:

Since we need to predict the arrival delays in minutes, we used linear regression model, later we some metrics to evaluate these. We did this using mlr3 library learners, we used linear regression learner to train on the train data and predicted on the test data.




```
# setting the learner
learner_ranger = lrn("regr.lm")

# Training and testing our model
p=learner_ranger$train(new_tsk, row_ids = train)

prediction = learner_ranger$predict(new_tsk, row_ids = test_set)
```

The plot shows the actual vs predicted arrival delay values. This graph gives a good representation of the predicted arrival delays. Other than this, we can see that the coefficient of the model here.

We also used cross validation technique to check the model with 3 folds, here we selected the attributes based on the sequential forward selection instead of the correlation filter. There was a slight increase in r square value.

CarrierDelay <lgl>	DepDelay <lgl>	LateAircraftDelay <lgl>	NASDelay <lgl>	WeatherDelay <lgl>	Year <lgl>	regr.rsq <dbl>	runtime_learners <dbl>
TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	0.96212020	0.56

Model Evaluation:

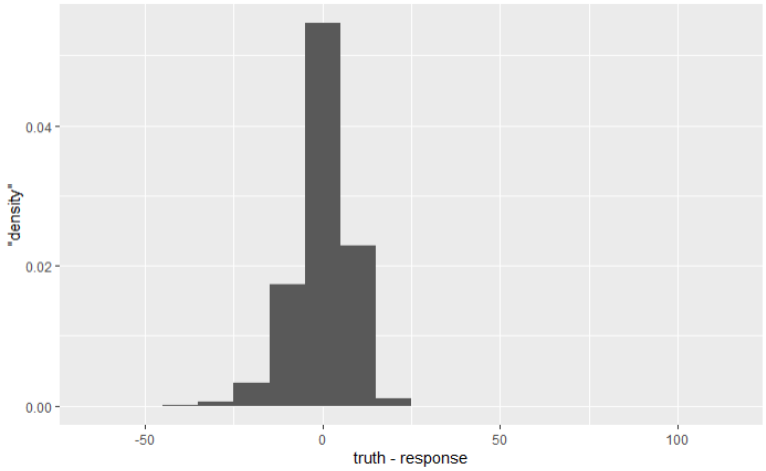
We used root mean square, mean square errors and the r squared value to evaluate the model.

```
> print(paste0("The r square value is : ",prediction$score(msr("regr.rsq"))))
[1] "The r square value is : 0.96135995563662"
> print(paste0("The mean square error value is : ",prediction$score(msr("regr.mse"))))
[1] "The mean square error value is : 68.6566648529059"
> print(paste0("The root mean square error value is : ",prediction$score(msr("regr.rmse"))))
[1] "The root mean square error value is : 8.28593174319617"
```

We can see that the r square value is about 1 which indicates that the model outperformed in predicting the arrival delays.

Model Findings:

From the model, we can perfectly predict the arrival delays using the stated attributes like departure delays etc. It also shows a very good relationship with the actual data and the test data. The following histogram shows the range of predicted arrival delay values.



The values less than 0 shows that the flight reached before the arrival time which is the best case scenario. The values greater than 0 shows the arrival delay. The model also shows that the departure delay is one of the big reason for arrival delay while other factors like weather delay security delays are also some of the causes of high arrival delays.

We can see that with and without re sampling, our linear regression model outperformed with r square value of 0.96.