STA 9750 - Final Project: Airline Fleets

**Savlat Sanoatov**

December 2, 2022

**Abstract:** I wanted to work on data that was familiar to me and so I found ‘Airline Fleet’ dataset from Kaggle. Planespotters.net has a full database on airlines around the world and the airplanes that each owns and operates.This dataset collects the top 100+ airlines in the world (by the size of their fleet). It is combined with information found on Wikipedia on the respective airline’s fleet and the average value/cost of the manufactured airplane. Dataset includes: (a)Parent Airline: i.e. International Airlines Group (IAG) (b)Airline: i.e. Iberia, Aer Lingus, British Airways…etc. which are owned by IAG (c) Aircraft Type: Manufacturer & Model (d)Current: Quantity of airplanes in Operation (e)Future: Quantity of airplanes on order, from planespotter.net (f)Order: Quantity airplanes on order, from Wikipedia (g)Unit Cost: Average unit cost of Aircraft Type, as found by Wikipedia and various google searches (h)Total Cost: Current quantity \* Unit Cost (i)Average Age: Average age of “Current” airplanes by “Aircraft Type”.

## The Data set:

flight = read.csv("C:/Users/savla/OneDrive/Desktop/dataset/Fleet Data.csv")

**Observation** This Fleet Dataset contains 1583 rows and 11 columns with names.

## Load Libraries:

library(tidyverse)  
library(dplyr)  
library(devtools)  
library(readxl)  
library(class)  
library(stats)  
library(caret)  
library(rpart)  
library(rpart.plot)  
library(ggplot2)  
library(readr)  
library(data.table)  
library(ggvis)  
library(openxlsx)  
library(corrplot)

names(flight) = c("Parent\_Airline",  
 "Airline",  
 "Aircraft\_Type",  
 "Current",  
 "Future",  
 "Historic",  
 "Total",  
 "Orders",  
 "Unit\_Cost",  
 "Total\_Cost\_Current",  
 "Average\_Age")  
flight = na.omit(flight)  
colnames<-names(flight)

## Use str(), glipse(), dim() and summary()

glimpse(flight)

***Note:*** Using glipse() function, I was able to see all rows and colums of the dataset with their names.

dim(flight)

***Note:*** Using dim() function, I was able to see the number of Rows(66) and Columns(11)

str(flight)

***Note:*** Using str() function, I was able to see the data.frame.

summary(flight)

***Note:*** Using summary() function, to get summary of the all dataset. For example, it stated the minimum, median and maximum numbers of each column.

## Plotting the dataset

setDT(flight)  
class(flight)

## [1] "data.table" "data.frame"

***Note:*** Assigned setDT function to create a data.table and class function to create a data.frame.

flight[1:5,.(Airline,Aircraft\_Type)]

## Airline Aircraft\_Type  
## 1: Aeroflot Airbus A320  
## 2: Aeroflot Boeing 737  
## 3: Rossiya Airlines Boeing 777  
## 4: Air Algerie Boeing 737  
## 5: Air Berlin Airbus A330

flight[1:5,2:7]

## Airline Aircraft\_Type Current Future Historic Total  
## 1: Aeroflot Airbus A320 71 1 3 75  
## 2: Aeroflot Boeing 737 20 1 16 37  
## 3: Rossiya Airlines Boeing 777 6 5 1 12  
## 4: Air Algerie Boeing 737 31 1 38 70  
## 5: Air Berlin Airbus A330 15 2 3 20

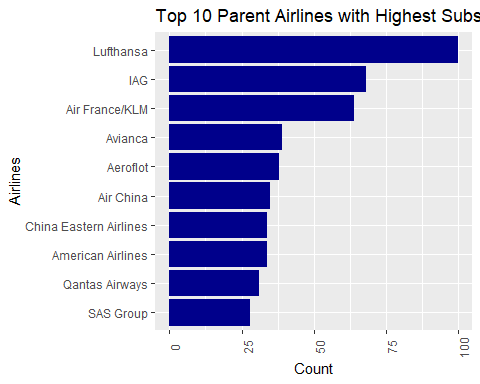
***Note:*** Wanted to find out if assigned functions working properly to make a data table and data frame.

airline\_count<-flight[,.(Count=.N),by=.(Parent\_Airline)]  
airline\_count[1:10]

## Parent\_Airline Count  
## 1: Aegean Airlines 10  
## 2: Aeroflot 38  
## 3: Aerolineas Argentinas 12  
## 4: Air Algerie 13  
## 5: Air Arabia 4  
## 6: Air Astana 9  
## 7: Air Berlin 12  
## 8: Air Canada 24  
## 9: Air China 35  
## 10: Air Europa 10

**Analysis:** Airline Fleet dataset has ‘Parent Airline’ and ‘Airline’ columns. Airline columns is subsidiary of Parent company. Used “airline\_count’ function to count the number of subsidiaries in each Parent Airline company. For example, Parent”Aeroflot’ has 3 subsidiary airlines.

top10<-flight[,.(Count=.N),by=(Parent\_Airline)][order(-Count)][1:10]  
ggplot(top10,aes(x=reorder(Parent\_Airline,Count),y=Count),fill=Parent\_Airline)+geom\_bar(stat="identity",fill="blue4")+  
theme(legend.position="none",axis.text.x=element\_text(angle=90))+labs(title="Top 10 Parent Airlines with Highest Subsidiary Airlines ", x="Airlines")+coord\_flip()



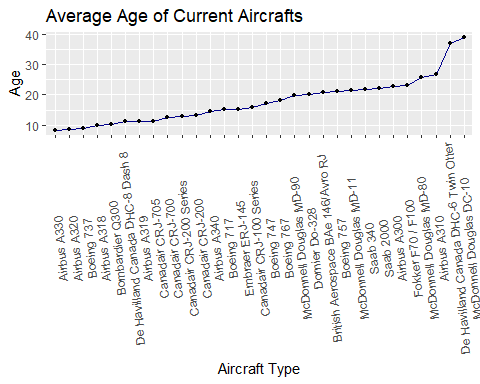
**Analysis:** Used ggplot function to plot the top 10 Parent Airline with highest subsidiary airlines on bar graph using original dataset. Based on the graph, *Lufthansa Airlines* has the highest subsidiary airline companies followed by IAG, Air France, KLM and Avianca.

Chart, bar chart

Description automatically generated

**Analysis:** Used ggplot function to plot the top 10 Parent Airline with highest subsidiary airlines on bar graph used dataset after removing all N/A’s. Based on the graph, *Lufthansa Airlines* still has the highest subsidiary airline companies followed by Turkish Airlines, Air China, Aeroflot, and Korean Air.

top\_age<-flight[,.(Age=mean(Average\_Age,na.rm=TRUE)),by=.(Aircraft\_Type)][order(-Age)][1:30]  
ggplot(top\_age,aes(x=reorder(Aircraft\_Type,Age),y=Age,group=1))+geom\_line(color="blue4")+geom\_point(size=1)+theme(legend.position="none",axis.text.x=element\_text(angle=95))+labs(x="Aircraft Type",title="Average Age of Current Aircrafts")



**Analysis:** Used ggplot function, to plot the linear graph to original dataset to show the Average Age of the Aircrafts currently or been used by Airlines in their fleeet. This line graph shows us that Airbus flights are the newest aircrafts used by airlines in the industry. Specifically, Airbus A330 is newest, while McDonnell Douglas DC-10 is almost 40 years old.

Chart, line chart

Description automatically generated

**Analysis:** Used ggplot function, to plot the linear graph to dataset after removing all N/A’S to show the Average Age of the Aircrafts currently or been used by Airlines in their fleet. This line graph shows us that Boeing 787 Dreamliner is newest, while Boeing 757 is almost 20 years old.

flight[,c(1,4:7)][,lapply(.SD,mean,na.rm=TRUE),by=Parent\_Airline][order(-Total)][1:5]

## Parent\_Airline Current Future Historic Total  
## 1: Southwest Airlines 718 38 195 952  
## 2: Ryanair 366 10 52 240  
## 3: Skywest Airlines 166 NaN 33 199  
## 4: Gol Linhas Aéreas 124 NaN 67 191  
## 5: IndiGo 124 10 16 150

**Analysis:** I wanted to find out the the top Parent Airlines by thier fleet size. This table shows us that the “Southwest Airlines has the largest fleet size based on the dataset.

flight[,lapply(.SD,sum,na.rm=TRUE),.SDcols=4:8,by=Aircraft\_Type][1:5]

## Aircraft\_Type Current Future Historic Total Orders  
## 1: Airbus A319 1188 28 326 1517 91  
## 2: Airbus A320 3299 112 928 4319 2506  
## 3: Airbus A321 1231 54 173 1458 1090  
## 4: ATR 42/72 439 5 425 866 59  
## 5: Boeing 737 5328 101 3873 9301 2300

**Analysis:** I also wanted to find out what is the most common used Aircraft type and found out this data. Based on this data, “Boeing 737” and “Airbus A320” are the most common aircraft types used in the industry by the airlines.

## Question: Separate Training and Test Data Sets

nrows <- NROW(flight)  
set.seed(35)  
index <- sample(1:nrows, 0.67 \* nrows)  
  
train <- flight[index,]  
test <- flight[-index,]  
rbind(dim(train),dim(test))

## [,1] [,2]  
## [1,] 1060 11  
## [2,] 523 11

**Analysis:** Created a training data set corresponding to 67% of the available data.

## Question 3: Applying KNN

• Perform a KNN prediction of Class as a function. • Produce the confusion matrix. • Calculate sensitivity, specificity, accuracy, and precision.

flight$Current = as.numeric(flight$Current)  
flight$Future = as.numeric(flight$Future)  
flight$Historic = as.numeric(flight$Historic)  
flight$Total = as.numeric(flight$Total)  
flight$Orders = as.numeric(flight$Orders)

**Analysis:** Applied the K-means clustering to convert them to numerical values

Current.mean = mean(flight$Current)  
Future.mean = mean(flight$Future)  
Historic.mean = mean(flight$Historic)  
Total.mean = mean(flight$Total)  
Orders.mean = mean(flight$Orders)  
  
  
Current.sd = sd(flight$Current)  
Future.sd = sd(flight$Future)  
Historic.sd = sd(flight$Historic)  
Total.sd = sd(flight$Total)  
Orders.sd = sd(flight$Orders)

**Analysis:** Performed a prediction to find the mean and standard deviation for Current, Future, Historic, Total, and Orders.

**Mean:**  
Current ~ 74.5  
Future ~ 3.6515  
Historic ~ 25.0757  
Total ~ 103.2727  
Orders ~ 50.5303

**Standard Deviation:**  
Current ~ 108.18  
Future ~ 5.0459  
Historic ~ 45.7845  
Total ~ 149.5194  
Orders ~ 72.1367

flight$Current = (flight$Current - Current.mean) / Current.sd  
flight$Future = (flight$Future - Future.mean) / Future.sd  
flight$Historic = (flight$Historic - Historic.mean) / Historic.sd  
flight$Total = (flight$Total - Total.mean) / Total.sd  
flight$Orders = (flight$Orders - Orders.mean) / Orders.sd

**Analysis:** Performed a prediction to find out the standard deviation of all rows of the Airline Fleet data set.

flight$Current = scale(flight$Current)  
flight$Future = scale(flight$Future)  
flight$Historic = scale(flight$Historic)  
flight$Total = scale(flight$Total)  
flight$Orders = scale(flight$Orders)

**Analysis:** Performed the K-Means Clustering to scale the function of the data set.

ind = sample(2, nrow(flight), replace=TRUE, prob=c(0.75,0.25))

**Analysis:** Used the index ind to find know if the condition is true.

flight.training = flight[ind==1,2:6]  
flight.test = flight[ind==2,2:6]  
flight.trainLabels = flight[ind==1,11]  
flight.testLabels = flight[ind==2,11]  
flight = na.omit(flight)

**Analysis:** Created the data subsets, the training, and the test, with the labels being kept separately

prediction\_knn = knn(train = flight.training, test = flight.test, cl = flight.trainLabels, k=3)

**Analysis:** Performed the KNN prediction of the flight.test data using prediction function.

(confusionMatrix = table(Actual\_Value = flight.testLabels, Predicted\_Value = prediction\_knn))

**Analysis:** Performed confusion matrix of the breast cancer data using confusionmatrix function.

sensitivity(confusionMatrix)

## [1] 0.8871123

**Analysis:** The sensitivity is calculated to be 88.71% and it looks good.

specificity(confusionMatrix)

## [1] 0.9216341

**Analysis:** The specificity is calculated, to be 92.16% and it looks good.

accuracy(confusionMatrix)

## [1] 0.9013521

**Analysis:** The accuracy is calculated, to be 90.1% and it looks good.

precision(confusionMatrix)

## [1] 0.9127424

**Analysis:** The precision is calculated, to be 91.27% and it looks good.

## Logistic Regression

flight.training$Class = flight.trainLabels

***Note*:** Assigned training class to the train labels function.

logregressionflight = glm(Class ~ Current + Future + Historic + Total + Orders, data = flight.training, family = 'binomial')

**Observation:** The logregressioncancer data set is list of 13.

predictionlr = predict(flight\_fit, flight.test, type = 'response')

flight.test$predicted = ifelse(predictionlr>0.7, TRUE, FALSE)

(confusionMatrix\_lr = table(Actual\_Value = flight.testLabels, predicted\_value = predictionlr>0.7))

**Observation:** Performed confusion matrix of the breast cancer data using *confusionmatrix* function.

sensitivity(confusionMatrix\_lr)

## [1] 0.8193124

**Observation:** The sensitivity is calculated, and it is 81.93% but it’s less so compared to applying knn.

specificity(confusionMatrix\_lr)

## [1] 0.9501425

**Observation:** The specificity is calculated to be 95.01% and it increased compared to applying knn.

accuracy(confusionMatrix\_lr)

## [1] 0.8914524

**Observation:** The accuracy is calculated, and it is 89.14% but it’s less so compared to applying knn.

precision(confusionMatrix\_lr)

## [1] 0.921787

**Observation:** The precision is calculated to be 92.17% and it increased compared to applying knn.

#using lm function for linear regression  
lm(Current ~ Total, data = flight)

##   
## Call:  
## lm(formula = Current ~ Total, data = flight)  
##   
## Coefficients:  
## (Intercept) Total   
## 4.881e-17 9.805e-01

flight\_fit = lm(Current ~ Total, data = flight)  
#getting the summary of flight fit to plot  
summary(flight\_fit)

##   
## Call:  
## lm(formula = Current ~ Total, data = flight)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.11004 -0.03808 -0.00272 0.08223 0.41810   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.881e-17 2.437e-02 0.00 1   
## Total 9.805e-01 2.456e-02 39.93 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.198 on 64 degrees of freedom  
## Multiple R-squared: 0.9614, Adjusted R-squared: 0.9608   
## F-statistic: 1595 on 1 and 64 DF, p-value: < 2.2e-16

#plotting the date   
plot(Current ~ Total, data = flight, main = "Current Fleets and Total Fleets")  
#adding linear line of best fit using abline() function  
abline(flight\_fit)

Chart, scatter chart

Description automatically generated

**Analysis:** Based on the linear regression graph, there are more current in use fleets than stored or ordered fleets.

#using lm function for linear regression  
lm(Current ~ Future, data = flight)

##   
## Call:  
## lm(formula = Current ~ Future, data = flight)  
##   
## Coefficients:  
## (Intercept) Future   
## -1.283e-16 7.331e-01

flight\_fit = lm(Current ~ Future, data = flight)  
#getting the summary of flight\_fit to plot  
summary(flight\_fit)

##   
## Call:  
## lm(formula = Current ~ Future, data = flight)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.5081 -0.4421 -0.1285 0.2947 2.5185   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.283e-16 8.437e-02 0.000 1   
## Future 7.331e-01 8.502e-02 8.623 2.59e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.6854 on 64 degrees of freedom  
## Multiple R-squared: 0.5374, Adjusted R-squared: 0.5302   
## F-statistic: 74.35 on 1 and 64 DF, p-value: 2.588e-12

#plotting the date   
plot(Current ~ Future, data = flight, main = "Current Fleets and Future Fleets")  
#adding linear line of best fit using abline() function  
abline(flight\_fit)

Chart, scatter chart

Description automatically generated

**Analysis:** Based on the plotted linear regression graph, the coefficients show us that the line crosses below the origin and future value expected to be higher than now.

#using lm function for linear regression  
lm(Historic ~ Total, data = flight)

##   
## Call:  
## lm(formula = Historic ~ Total, data = flight)  
##   
## Coefficients:  
## (Intercept) Total   
## 6.679e-17 8.682e-01

flight\_fit = lm(Historic ~ Total, data = flight)  
#getting the summary of flight\_fit to plot  
summary(flight\_fit)

##   
## Call:  
## lm(formula = Historic ~ Total, data = flight)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.21684 -0.21653 -0.00306 0.08879 2.89510   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.679e-17 6.155e-02 0 1   
## Total 8.682e-01 6.203e-02 14 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.5001 on 64 degrees of freedom  
## Multiple R-squared: 0.7538, Adjusted R-squared: 0.7499   
## F-statistic: 195.9 on 1 and 64 DF, p-value: < 2.2e-16

#plotting the date   
plot(Historic ~ Total, data = flight, main = "Historic Fleets and Total Fleets")  
#adding linear line of best fit using abline() function  
abline(flight\_fit)

Chart, scatter chart

Description automatically generated

**Analysis:** Based on the plotted linear regression graph, it crosses above the origin which coefficients are positive. The historic fleets are less than current in use and future ordered fleets.

#using lm function for linear regression  
lm(Orders ~ Current, data = flight)

##   
## Call:  
## lm(formula = Orders ~ Current, data = flight)  
##   
## Coefficients:  
## (Intercept) Current   
## -4.879e-17 6.514e-01

flight\_fit = lm(Orders ~ Current, data = flight)  
#getting the summary of flight\_fit to plot  
summary(flight\_fit)

##   
## Call:  
## lm(formula = Orders ~ Current, data = flight)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.9551 -0.3551 -0.2169 0.1129 4.5465   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -4.879e-17 9.412e-02 0.000 1   
## Current 6.514e-01 9.484e-02 6.868 3.17e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7646 on 64 degrees of freedom  
## Multiple R-squared: 0.4243, Adjusted R-squared: 0.4153   
## F-statistic: 47.17 on 1 and 64 DF, p-value: 3.168e-09

#plotting the date   
plot(Orders ~ Current, data = flight, main = "Ordered Fleets and Current Fleets")  
#adding linear line of best fit using abline() function  
abline(flight\_fit)

Chart, scatter chart

Description automatically generated

**Analysis:** Based on the plotted regression graph compares current in use and ordered aircrafts by airlines. As we can see, more dots are closer to origin which means line crosses below the origin.

plot(flight$Current,flight$Orders)  
abline(flight\_fit)  
legend("topright",  
legend=paste("R-squared = ",  
format(summary(flight\_fit)$adj.r.squared,  
digits=2)))

Chart, scatter chart

Description automatically generated

**Analysis:** This is plotted linear regression is Current and Orders which R-squared to be 0.42. Based on the coefficients line crosses below origin.

plot(flight$Current,flight$Future)  
abline(flight\_fit)  
legend("topright",  
legend=paste("R-squared = ",  
format(summary(flight\_fit)$adj.r.squared,  
digits=2)))

Chart, scatter chart

Description automatically generated

**Analysis:** This is plotted linear regression is Current and Future orders which R-squared to be 0.42. Based on the coefficients line crosses origin.