Flight Landing Prediction Project: Part 1

Background: Flight landing.
Motivation: To reduce the risk of landing overrun.
Goal: To study what factors and how they would impact the landing distance of a commercial flight.
Data: Landing data (landing distance and other parameters) from 950 commercial flights (not real data set but simulated from statistical models). See two Excel files 'FAA-1.xls' (800 flights) and 'FAA-2.xls' (150 flights).

Initial exploration of the data

Step 1: Loading data files

In order to read in the two files, I installed the "readxl" package and loaded it. After this, the two datasets were read into R using the ***read_excel*** function. The first few rows of the data sets are observed using the code ***head ()***. The output shows that "FAA1.xls" has 8 column variables while "FAA2.xls" has 7 column variables.

R code:

```
install.packages("readxl")
library(readxl)
library(dplyr)
Data1 <- read_excel("FAA1.xls")
Data2 <- read_excel("FAA2.xls")

head(Data1)
head(Data2)
```

Output:

```
# A tibble: 6 x 8
  aircraft duration no_pasg speed_ground speed_air height pitch distance
  <chr>       <dbl>   <dbl>        <dbl>     <dbl>  <dbl> <dbl>    <dbl>
1 boeing       98.5      53         108.      109.   27.4  4.04    3370.
2 boeing      126.       69         102.      103.   27.8  4.12    2988.
3 boeing      112.       61          71.1      NA    18.6  4.43    1145.
4 boeing      197.       56          85.8      NA    30.7  3.88    1664.
5 boeing       90.1      70          59.9      NA    32.4  4.03    1050.
6 boeing      138.       55          75.0      NA    41.2  4.20    1627.

# A tibble: 6 x 7
  aircraft no_pasg speed_ground speed_air height pitch distance
  <chr>       <dbl>        <dbl>     <dbl>  <dbl> <dbl>    <dbl>
1 boeing         53         108.      109.   27.4  4.04    3370.
2 boeing         69         102.      103.   27.8  4.12    2988.
3 boeing         61          71.1      NA    18.6  4.43    1145.
4 boeing         56          85.8      NA    30.7  3.88    1664.
5 boeing         70          59.9      NA    32.4  4.03    1050.
6 boeing         55          75.0      NA    41.2  4.20    1627.
```

Step 2: Check the structure of data sets

The structure of the two datasets are examined using the ***str ()*** function. Data 1 "FAA1.xls" has a sample size of 800 and a total of 8 variables. Data 2 has a sample size of 150, with 7 variables. The two datasets also differ in that Data1 contains the data on Flight duration between taking off and landing, while Data2 doesn't.

Code and Output:

```
> str(Data1)
Classes 'tbl_df', 'tbl' and 'data.frame':       800 obs. of  8 variables:
 $ aircraft    : chr  "boeing" "boeing" "boeing" "boeing" ...
 $ duration    : num  98.5 125.7 112 196.8 90.1 ...
 $ no_pasg     : num  53 69 61 56 70 55 54 57 61 56 ...
 $ speed_ground: num  107.9 101.7 71.1 85.8 59.9 ...
 $ speed_air   : num  109 103 NA NA NA ...
 $ height      : num  27.4 27.8 18.6 30.7 32.4 ...
 $ pitch       : num  4.04 4.12 4.43 3.88 4.03 ...
 $ distance    : num  3370 2988 1145 1664 1050 ...
> str(Data2)
Classes 'tbl_df', 'tbl' and 'data.frame':       150 obs. of  7 variables:
 $ aircraft    : chr  "boeing" "boeing" "boeing" "boeing" ...
 $ no_pasg     : num  53 69 61 56 70 55 54 57 61 56 ...
 $ speed_ground: num  107.9 101.7 71.1 85.8 59.9 ...
 $ speed_air   : num  109 103 NA NA NA ...
 $ height      : num  27.4 27.8 18.6 30.7 32.4 ...
 $ pitch       : num  4.04 4.12 4.43 3.88 4.03 ...
 $ distance    : num  3370 2988 1145 1664 1050 ...
```

Step 3: Merge datasets and remove duplicates

The two datasets are merged using the ***rbind ()*** function. Because the two datasets need to have matching columns for the rbind function to work, a column titled ***"duration"*** with all "NA" values is added to Data 2. After merging, the combined dataset has a total of 950 rows and 8 columns. I checked for duplicates using the code ***sum(dup_idx=="TRUE")***. This shows that there are 100 duplicated rows in the dataset. This is removed as shown in the code below. The new combined dataset with no duplicates is named "Data".

Code:

```
# merge the two data sets
Data_merge <- rbind(Data1,Data2)
dim(Data_merge)
head(Data_merge)

#Are there duplicates? How many?
dup_idx <- duplicated(Data_merge[,-2])
sum(dup_idx == "TRUE")
#100
# remove duplicates
Data <- Data_merge[!dup_idx, ]
dim(Data)
#850 8
```

Step 4: Check the structure and summary statistics of combined data set

The structure of the combined dataset "Data" is checked using the ***str ()*** function. After which, the summary statistics is calculated for each of the variables using the ***summary ()*** function. The output reveals that the combined dataset has a sample size of 850 observations, with a total of 8 column variables.

Code:

```
#Check the structure of the combined data set.
str(Data)
#n=850; number of variables=8
#Provide summary statistics for each variable.
summary(Data)
```

Output:

```
'data.frame':   850 obs. of  8 variables:
 $ aircraft    : chr  "airbus" "airbus" "airbus" "airbus" ...
 $ no_pasg     : num  36 38 40 41 43 44 45 45 45 45 ...
 $ speed_ground: num  47.5 85.2 80.6 97.6 82.5 ...
 $ speed_air   : num  NA NA NA 97 NA ...
 $ height      : num  14 37 28.6 38.4 30.1 ...
 $ pitch       : num  4.3 4.12 3.62 3.53 4.09 ...
 $ distance    : num  251 1257 1021 2168 1321 ...
 $ duration    : num  172 188 93.5 123.3 109.2 ...
```

```
  aircraft            duration         no_pasg      speed_ground      speed_air
Length:850         Min.   : 14.76   Min.   :29.0   Min.   : 27.74   Min.   : 90.00
Class :character   1st Qu.:119.49   1st Qu.:55.0   1st Qu.: 65.90   1st Qu.: 96.25
Mode  :character   Median :153.95   Median :60.0   Median : 79.64   Median :101.15
                   Mean   :154.01   Mean   :60.1   Mean   : 79.45   Mean   :103.80
                   3rd Qu.:188.91   3rd Qu.:65.0   3rd Qu.: 92.06   3rd Qu.:109.40
                   Max.   :305.62   Max.   :87.0   Max.   :141.22   Max.   :141.72
                   NA's   :50                                       NA's   :642
     height           pitch          distance
Min.   :-3.546   Min.   :2.284   Min.   :  34.08
1st Qu.:23.314   1st Qu.:3.642   1st Qu.: 883.79
Median :30.093   Median :4.008   Median :1258.09
Mean   :30.144   Mean   :4.009   Mean   :1526.02
3rd Qu.:36.993   3rd Qu.:4.377   3rd Qu.:1936.95
Max.   :59.946   Max.   :5.927   Max.   :6533.05
```

Step 5: Summary/Observations

- The dataset consists of two types of aircraft: Airbus and Boeing
- Over 3/4 of the dataset on speed_air is missing. In addition, 50 data points are missing for the duration variable.
- The summary statistics show that the minimum height of an aircraft when it is passing over the threshold of the runway (in meters) is negative. This may be due to incorrect recording/input of the data.

- The data includes some abnormal ground speed (the normal ground speed as specified the data dictionary should be between 30MPH and 140MPH).
- The maximum air speed of an aircraft is greater than 140MPH. This is abnormal based on the definition of the variable. In addition, the range of the distance is quite large, with the maximum landing distance exceeding the length of the airport runway which is typically less than 6000 feet.

Data Cleaning and further exploration

Step 6: Remove abnormal values

There are abnormal values in the data set. I attempt to remove abnormal values and create a new dataset with only the normal values.

Rather than filtering out abnormal values and rows with missing values which would lead to the removal of over 600 rows, I considered filtering out the normal data from the combined dataset to create a new dataset with only normal values. Because only two variables "Speed_air" and "duration" have missing values, if the rows with missing values are removed from the dataset, the resulting dataset would be less than ¼ of the original dataset. This would also lead to the loss of data points in the other column variables.

First, I created a new dataset named "clean_data" which would consist of the normal values across all the variables. Then, I filtered out the missing values in the duration variable and rows with duration greater than 40 minutes from the original combined data.

I filtered out the missing values in the speed_air variable and rows where the air speed is greater than or equal to 30MPH or less than or equal to 140MPH. More so, rows where the ground speed is greater than or equal to 30MPH or less than or equal to 140MPH are retained in the new dataset. Then, rows where the aircraft height is greater than or equal to 6 are retained. Finally, rows with landing distance less than 6000ft are included.

In summary, a total of 19 rows were removed from the original combined data set, leading to a reduced dataset of 831 rows, with 8 column variables. This new dataset is titled "clean_data".

Code:

```
clean_data <- Data %>%
  filter((is.na(duration) | duration > 40) &
           (is.na(speed_air) | (speed_air >= 30 & speed_air <= 140)) &
           (speed_ground >= 30 & speed_ground <= 140) &
           (height >= 6) &
          (distance < 6000))
```

Step 7: structure of the new combined data set

Using the ***str ()*** and the ***summary ()*** functions, the new data set has 831 observations with 8 variables. The summary of the variables reveals that there are no abnormal values in the dataset, but speed_air and duration variables consist of missing values. The rows with missing values are not removed because the removal of an entire row due to a missing value in one or two columns would lead to the loss of more than ¾ of the entire dataset.

Code and Output:

```
> str(clean_data)
Classes 'tbl_df', 'tbl' and 'data.frame':       831 obs. of  8 variables:
 $ aircraft    : chr  "boeing" "boeing" "boeing" "boeing" ...
 $ duration    : num  98.5 125.7 112 196.8 90.1 ...
 $ no_pasg     : num  53 69 61 56 70 55 54 57 61 56 ...
 $ speed_ground: num  107.9 101.7 71.1 85.8 59.9 ...
 $ speed_air   : num  109 103 NA NA NA ...
 $ height      : num  27.4 27.8 18.6 30.7 32.4 ...
 $ pitch       : num  4.04 4.12 4.43 3.88 4.03 ...
 $ distance    : num  3370 2988 1145 1664 1050 ...
> summary(clean_data)
   aircraft            duration         no_pasg       speed_ground
 Length:831         Min.   : 41.95   Min.   :29.00   Min.   : 33.57
 Class :character   1st Qu.:119.63   1st Qu.:55.00   1st Qu.: 66.20
 Mode  :character   Median :154.28   Median :60.00   Median : 79.79
                    Mean   :154.78   Mean   :60.06   Mean   : 79.54
                    3rd Qu.:189.66   3rd Qu.:65.00   3rd Qu.: 91.91
                    Max.   :305.62   Max.   :87.00   Max.   :132.78
                    NA's   :50
   speed_air          height           pitch          distance
 Min.   : 90.00   Min.   : 6.228   Min.   :2.284   Min.   :  41.72
 1st Qu.: 96.23   1st Qu.:23.530   1st Qu.:3.640   1st Qu.: 893.28
 Median :101.12   Median :30.167   Median :4.001   Median :1262.15
 Mean   :103.48   Mean   :30.458   Mean   :4.005   Mean   :1522.48
 3rd Qu.:109.36   3rd Qu.:37.004   3rd Qu.:4.370   3rd Qu.:1936.63
 Max.   :132.91   Max.   :59.946   Max.   :5.927   Max.   :5381.96
 NA's   :628
```

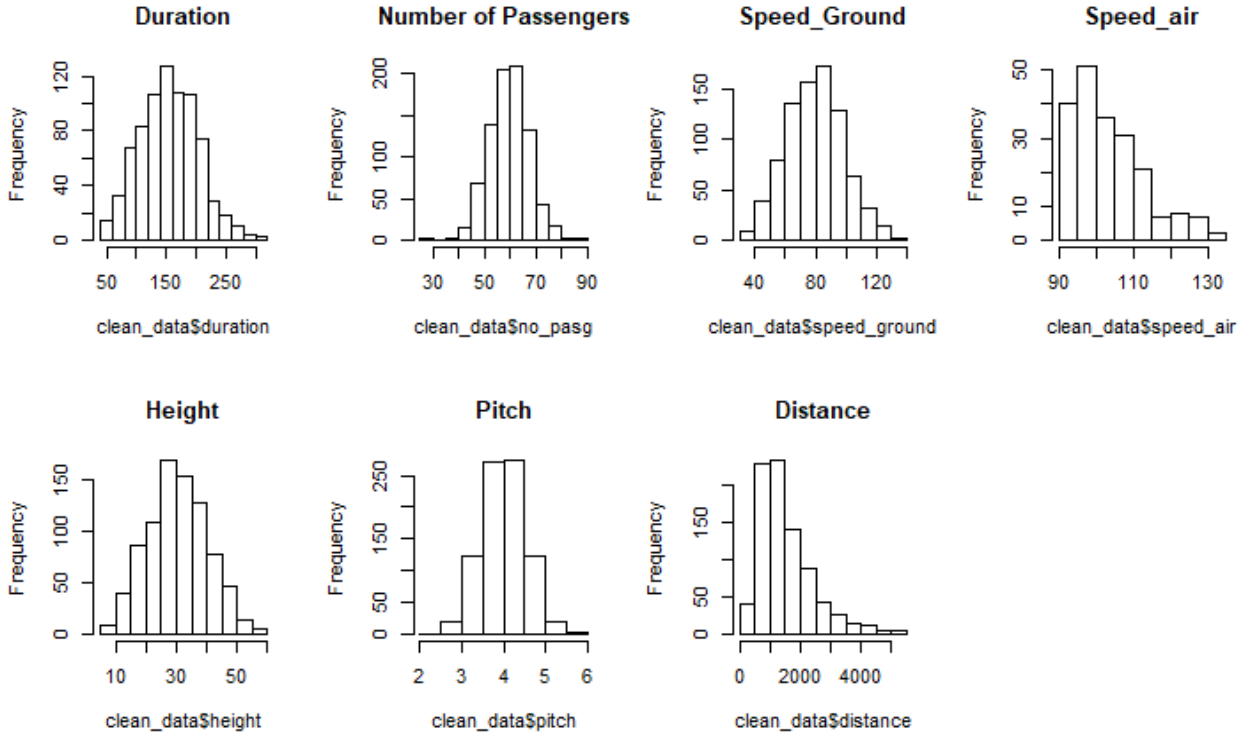
Step 8: Histogram of the variables

The histogram of the variables is constructed using the ***ggplot2*** and the ***HMISC*** packages. The histogram of the duration, number of passengers, speed_ground, height, and pitch variables are symmetric, suggesting that the variables follow a normal distribution. However, the histogram for speed_ground and distance are not symmetric and skewed to the right.

Code:

```
library(ggplot2)
library(Hmisc)
hist.data.frame(clean_data[-1])

par("mfrow"=c(2, 4))
hist(clean_data$duration, main = "Duration")
hist(clean_data$no_pasg, main = "Number of Passengers")
hist(clean_data$speed_ground, main = "Speed_Ground")
hist(clean_data$speed_air, main = "Speed_air")
hist(clean_data$height, main = "Height")
hist(clean_data$pitch, main = "Pitch")
hist(clean_data$distance, main = "Distance")
```

Output:



Step 9: Summary

- There are 50 missing values in the duration variable
- There are 628 missing values in the speed_air variable.
- The histogram of the duration, number of passengers, speed_ground, height, and pitch variables are symmetric, suggesting that the variables follow a normal distribution. However, the histogram for speed_ground and distance are not symmetric and skewed to the right.
- The range between the landing distance is very high.

Initial analysis for identifying important factors that impact the response variable "landing distance"

Step 10: Pairwise Correlation

First, I recoded the "aircraft" variable which is a factor variable to 0/1. Airbus =0; Boeing =1. Then, the pairwise correlation is calculated using the ***cor()*** function. Due to the presence of "NAs" in some of the variables(Speed_air and duration) , ***use="complete.obs"*** is included in the ***cor()*** function.

Code:

```
#first, recode aircraft as 0/1
clean_data$aircraft[clean_data$aircraft == "airbus"] <- 0
clean_data$aircraft[clean_data$aircraft == "boeing"] <- 1
clean_data$aircraft <- as.numeric(clean_data$aircraft)
str(clean_data$aircraft)

cor(clean_data$aircraft, clean_data$distance, use="complete.obs")
cor(clean_data$duration, clean_data$distance, use="complete.obs")
cor(clean_data$no_pasg, clean_data$distance, use="complete.obs")
cor(clean_data$speed_ground, clean_data$distance, use="complete.obs")
cor(clean_data$speed_air, clean_data$distance, use="complete.obs")
cor(clean_data$height, clean_data$distance, use="complete.obs")
cor(clean_data$pitch, clean_data$distance, use="complete.obs")
```

Table 1 shows that there's a positive linear relationship between the following pairs: Landing distance and Speed_ground; Landing distance and Speed_air; Landing distance and type of aircraft; Landing distance and Height; and Landing distance and Pitch. Meanwhile, Speed_air and Speed_ground have the strongest positive linear relationship with landing distance.

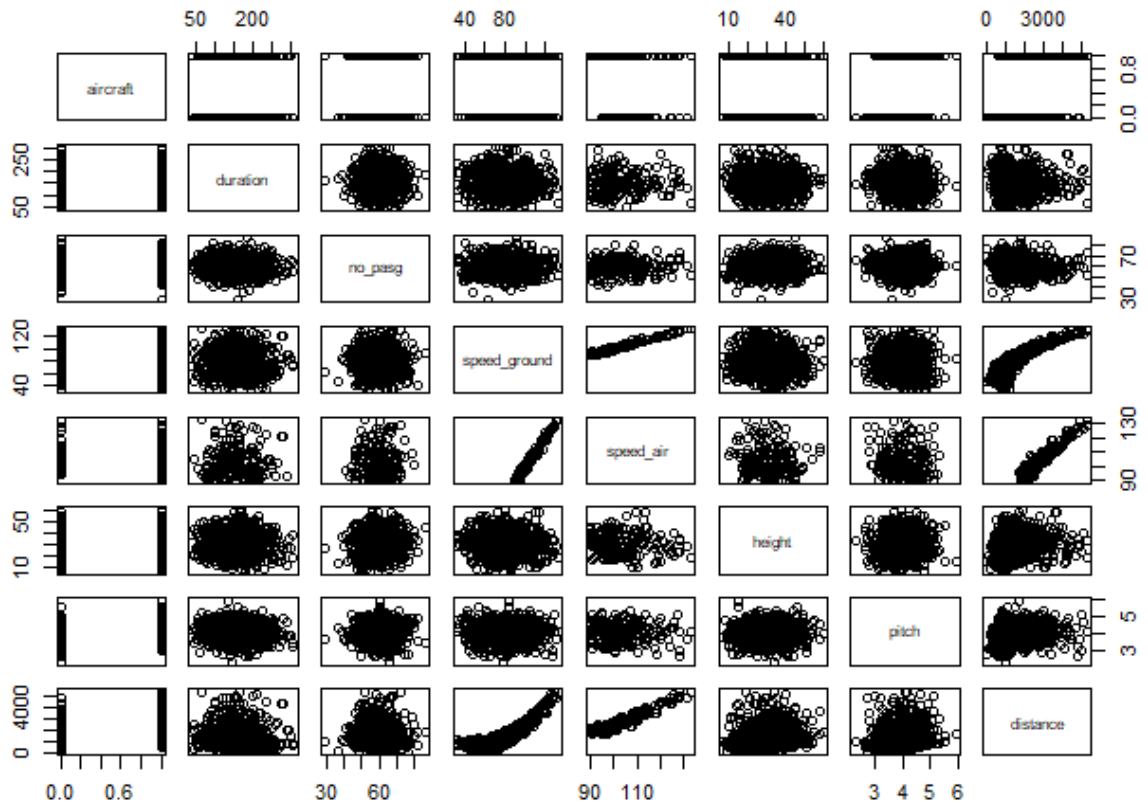
Duration and number of passengers are negatively related to Landing distance. This relationship is quite weak given the size of the correlation coefficient.

Table 1: Correlation between Landing Distance and the x's

Variables	*Size of Correlation*	*Direction*
Speed_air	0.9421	Positive
Speed_ground	0.8662	Positive
Aircraft	0.2381	Positive
Height	0.0994	Positive
Pitch	0.0870	Positive
Duration	0.0514	Negative
No_pasg	0.0178	Negative

Step11: X-Y scatter plots.

The matrix scatterplot is constructed using the code ***pairs (clean_data [,2:8], main="Scatterplot Matrix")***. The scatterplot is consistent with the results in Step10. It confirms the presence of a strong positive linear relation between landing distance and speed_air; landing distance and speed_ground. However, the strength and direction of correlation between landing distance and the other variables aren't very clear in the scatterplot.



Step 12: Yes, the airplane make was recoded as 0/1 and included in Steps 10-11.

Regression using a single factor each time

Step 13: To regress Y (landing distance) on each of the X variables individually, the ***lm ()*** function is used. In order of significance, as shown in Table 2, the air speed and ground speed of the aircraft are the most significant factors. At 0.05 level of significance, air speed, ground speed, type of aircraft, height, and pitch are significant factors in predicting the landing distance. On the other hand, Duration and number of passengers aren't significant factors in predicting landing distance. Note that the sample size (n) in each of the models vary given that some variables have missing values while others do not.

Code:

```
model1 <- lm(clean_data$distance~clean_data$aircraft, data=clean_data)
summary(model1)
model2 <- lm(clean_data$distance~clean_data$no_pasg, data=clean_data)
summary(model2)
model3 <- lm(clean_data$distance~clean_data$speed_ground, data=clean_data)
summary(model3)
model4 <- lm(clean_data$distance~clean_data$speed_air, data=clean_data)
summary(model4)
model5 <- lm(clean_data$distance~clean_data$height, data=clean_data)
summary(model5)
model6 <- lm(clean_data$distance~clean_data$pitch, data=clean_data)
summary(model6)
model7 <- lm(clean_data$distance~clean_data$duration, data = clean_data)
summary(model7)
```

Table 2: Regression of Y (landing distance) on each of the X variables.

Variables	*P-value (order of sig.)*	*Direction of regression coefficient*
Speed_ground	<2e-16	Positive
Speed_air	<2e-16	Positive
Aircraft	3.53e-12	Positive
Height	0.00412	Positive
Pitch	0.012081	Positive
Duration	0.151	Negative
No_pasg	0.609	Negative

Step14: Standardize each X variable. Regress Y (landing distance) on each of the X' variables.

All the X variables are standardized using the ***scale ()*** function. This function standardizes the X variables such that ***x' = (x - mean(x)) / sd(x)***. This function also ignores missing values. I confirmed that the new X variables have mean of 0 and standard deviation of 1. Then, Y is regressed on the standardized X variables.

The table reveals that the ground speed, air speed, type of aircraft, height, and pitch have a positive and significant impact on the aircraft's landing distance, with the ground speed having that largest impact. However, the duration and the number of passengers have a negative and non-significant relationship with the landing distance of the aircraft.

Code:

```
clean_data_norm <- as.data.frame(scale(clean_data[,-8]))
clean_data_norm$distance <- clean_data$distance
summary(clean_data_norm)

y <- clean_data_norm$distance
x1 <- clean_data_norm$aircraft
x2 <- clean_data_norm$no_pasg
x3 <- clean_data_norm$speed_ground
x4 <- clean_data_norm$speed_air
x5 <- clean_data_norm$height
x6 <- clean_data_norm$pitch
x7 <- clean_data_norm$duration
#Check
sd(x1);sd(x2); sd(x3); sd(x4, na.rm=TRUE); sd(x5); sd(x6); sd(x7,na.rm=TRUE)

model11 <- lm(y~x1, data=clean_data_norm)
summary(model11)

model12 <- lm(y~x2, data=clean_data_norm)
summary(model12)

model13 <- lm(y~x3, data=clean_data_norm)
summary(model13)

model14 <- lm(y~x4, data=clean_data_norm)
summary(model14)

model15 <- lm(y~x5, data=clean_data_norm)
summary(model15)

model16 <- lm(y~x6, data=clean_data_norm)
summary(model16)

model17 <- lm(y~x7, data=clean_data_norm)
summary(model17)
```

Table 3: Regression of Y (landing distance) on each of the Standardized X variables.

Variables	*Size of regression coef*	*Direction of regression coefficient*
Speed_ground	776.45 (sig)	Positive
Speed_air	774.35 (sig)	Positive
Aircraft	213.46 (sig)	Positive
Height	89.11 (sig)	Positive
Pitch	78.01 (sig)	Positive
Duration	46.48 (not sig.)	Negative
No_pasg	15.92(not sig.)	Negative

Step 15: Summary and Ranking of Variables

When we compare Tables 1,2,3, it is seen that the results are consistent. The positive relationship between landing distance and ground speed, air speed, type of aircraft, height, and pitch is shown by the sign of the correlation and regression coefficient. The size of the relationship is also revealed in the size of the correlation coefficient and the regression coefficient. In the same vein, Tables 1,2, and 3 show that duration and the number of passengers is negatively related to the landing distance. However, not significant as shown in Table 3.

More so, the ground speed, air speed, type of aircraft are the most important factors. Based on the three tables, the factors are ranked based on their relative importance in determining the landing distance.

Table 0: All the factors based on their relative importance in determining the landing distance

Variables	*Rank of Importance*
Speed_ground	1
Speed_air	2
Aircraft	3
Height	4
Pitch	5
Duration	6
No_pasg	7
*****Most important =1, least important=7***	

Check collinearity

Step 16: Compare the regression coefficients of the three models

Using the unstandardized/original X variables, the ***lm ( )*** function is employed in constructing the three models. The correlation between speed_ground and speed_air is also investigated.

When the landing distance is regressed individually on Speed_air and Speed_ground as in Model 1 and 2, both variables are significant with positive regression coefficients. However, when y is regressed on both x's as in Model 3, only the coefficient of Speed_air remains significant with a positive regression coefficient. The regression coefficient of Speed_ground isn't significant, and the sign changed from positive to negative. The change in significance and sign of regression coefficient could be due to the presence of a strong positive linear relationship between Speed_ground and Speed_air (correlation coef. = 0.9879).

In summary, the air speed and the ground speed of the aircraft have a significant and positive effect on landing distance if they are individually considered. However, when they are both included in the model, the air speed of the aircraft is still positively and significantly related to the landing distance, but the ground speed has a negative and insignificant effect on the landing distance.

For model selection, I would choose Model 1. Although, Model 2 and 3 offer higher R-squared, Model 3 suggests the problem of collinearity. Model 2 on the other hand consists of Speed_air which has over ¾ of the data points missing. Hence, I would select Model 1 and keep Speed_ground as a variable in predicting the landing distance.

Summary Table

	Model 1	**Model 2**	**Model 3**
Speed_ground	41.44 (sig)		-14.37
Speed_air		79.53 (sig)	93.96 (sig)
R-squared	0.7504	0.8875	0.8883
Adjusted R-squared	0.7501	0.8870	0.8871
Correlation (Speed_air, Speed_ground)		0.9879	
Note: The X variables are unstandardized			

Code and Output:

```
> model1 <- lm(clean_data$distance~clean_data$speed_ground, data=clean_data)
> summary(model1)

Call:
lm(formula = clean_data$distance ~ clean_data$speed_ground, data = clean_dat
a)

Residuals:
    Min      1Q  Median      3Q     Max
-897.09 -319.16  -72.09  210.83 1798.88

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)            -1773.9407    67.8388  -26.15   <2e-16 ***
clean_data$speed_ground    41.4422     0.8302   49.92   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 448.1 on 829 degrees of freedom
Multiple R-squared:  0.7504,    Adjusted R-squared:  0.7501
F-statistic:  2492 on 1 and 829 DF,  p-value: < 2.2e-16

> extractAIC(model1) #10148
[1]     2.00 10148.53
```

```
> model12 <- lm(clean_data$distance~clean_data$speed_air, data=clean_data)
> summary(model12)

Call:
lm(formula = clean_data$distance ~ clean_data$speed_air, data = clean_data)

Residuals:
    Min      1Q  Median      3Q     Max
-776.21 -196.39    8.72  209.17  624.34

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)          -5455.709    207.547  -26.29   <2e-16 ***
clean_data$speed_air    79.532      1.997   39.83   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 276.3 on 201 degrees of freedom
  (628 observations deleted due to missingness)
Multiple R-squared:  0.8875,    Adjusted R-squared:  0.887
F-statistic:  1586 on 1 and 201 DF,  p-value: < 2.2e-16

> extractAIC(model12)
[1]    2.000 2284.334

> model13 <- lm(clean_data$distance~clean_data$speed_air+ clean_data$speed_gr
ound, data=clean_data)
> summary(model13)

Call:
lm(formula = clean_data$distance ~ clean_data$speed_air + clean_data$speed_gr
ound,
    data = clean_data)

Residuals:
    Min      1Q  Median      3Q     Max
-819.74 -202.02    3.52  211.25  636.25

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)           -5462.28     207.48 -26.327  < 2e-16 ***
clean_data$speed_air     93.96      12.89   7.291 6.99e-12 ***
clean_data$speed_ground  -14.37      12.68  -1.133    0.258
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 276.1 on 200 degrees of freedom
  (628 observations deleted due to missingness)
Multiple R-squared:  0.8883,    Adjusted R-squared:  0.8871
F-statistic:   795 on 2 and 200 DF,  p-value: < 2.2e-16

> extractAIC(model13)
[1]    3.000 2285.035
> #collinearity check
> cor(clean_data$speed_air,clean_data$speed_ground,use="complete.obs")
[1] 0.9879383
```

Variable selection based on ranking in Table 0.

Step 17: Fit Models based on Table 0

The ***lm ()*** function is used to construct the models. The X variables are included in the model based on the ranking in Table 0. The table below shows the regression coefficient for each of the models, the R-squared and the AIC which is gotten using the ***extractAIC( )*** function are also reported. Note that the sample size (n) in each of the models vary given that some variables have missing values while others do not.

The table and plot show that as more variables are included, the R-squared value increased. However, there's no increase in R-squared from Model 4 to 5, suggesting that the inclusion of pitch added no new information or improvement to the model. This is also evident given that the adjusted R-squared decreased.

Results Summary Table:

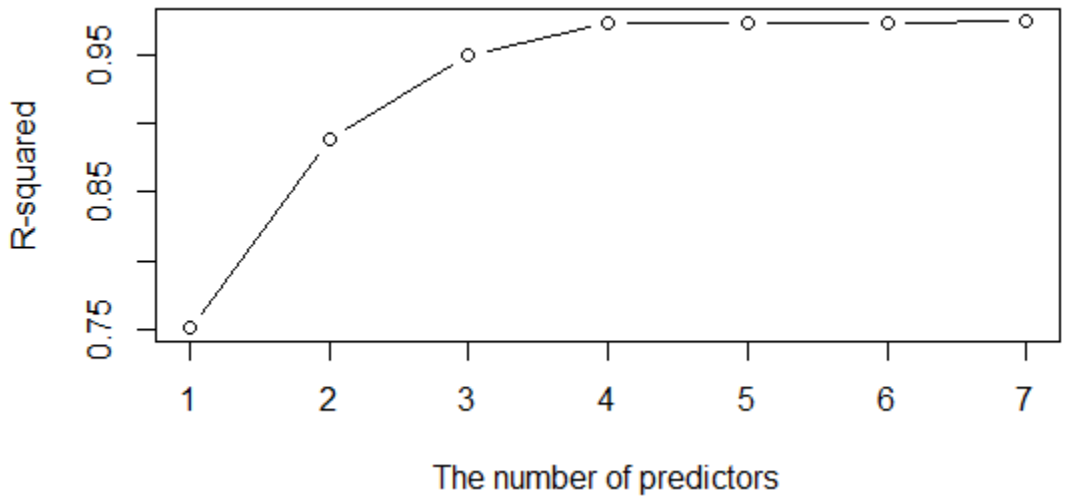
```
model14a <- lm(clean_data$distance~clean_data$speed_ground, data=clean_data)
summary(model14a)
extractAIC(model14a)

model14b <- lm(clean_data$distance~clean_data$speed_ground+clean_data$speed_air, data=clean_data)
summary(model14b)
extractAIC(model14b)

model14c <- lm(clean_data$distance~clean_data$speed_ground+clean_data$speed_air+clean_data$aircraft, data=clean_data)
summary(model14c)
extractAIC(model14c)

model15 <- lm(clean_data$distance~clean_data$speed_ground+clean_data$speed_air+clean_data$aircraft+clean_data$height
              , data=clean_data)
summary(model15)
extractAIC(model15)

model16 <- lm(clean_data$distance~clean_data$speed_ground+clean_data$speed_air+clean_data$aircraft+clean_data$height
              +clean_data$pitch, data=clean_data)
summary(model16)
extractAIC(model16)

model17 <- lm(clean_data$distance~clean_data$speed_ground+clean_data$speed_air+clean_data$aircraft+clean_data$height
              +clean_data$pitch+ clean_data$duration, data=clean_data)
summary(model17)
extractAIC(model17)

model18 <- lm(clean_data$distance~ clean_data$speed_ground+clean_data$speed_air+clean_data$aircraft+
                clean_data$height+clean_data$pitch+ clean_data$duration+ clean_data$no_pasg, data=clean_data)
summary(model18)
extractAIC(model18)
```

```
r.squared.1<- summary(model14a)$ r.squared ; print(r.squared.1)
r.squared.2<- summary(model14b)$ r.squared ; print(r.squared.2)
r.squared.3<- summary(model14c)$ r.squared ; print(r.squared.3)
r.squared.4<- summary(model15)$ r.squared ; print(r.squared.4)
r.squared.5<- summary(model16)$ r.squared ; print(r.squared.5)
r.squared.6<- summary(model17)$ r.squared ; print(r.squared.6)
r.squared.7<- summary(model18)$ r.squared ; print(r.squared.7)
plot(c(1:7),c(r.squared.1,r.squared.2,r.squared.3,r.squared.4,r.squared.5,r.squared.6,
              r.squared.7),type="b",ylab ="R-squared", xlab ="The number of predictors")
```

Output and Summary Result:

	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7
Speed_ground	41.44	-14.37	-11.27	-5.89	-6.07	-3.64	-3.55
Speed_air		93.96	92.36	88.05	88.23	85.64	85.55
Aircraft			413.19	426.96	428.48	439.41	437.94
Height				13.64	13.63	13.68	13.68
Pitch					-4.05	-12.94	-13.49
Duration						0.15	0.13
No_pasg							-1.98
R-squared	0.7504	0.8883	0.9497	0.9738	0.9738	0.9744	0.9747
Adjusted R-squared	0.7501	0.8871	0.949	0.9733	0.9732	0.9736	0.9738
AIC	10148.53	2285.035	2124.92	1994.29	1996.24	1919.45	1919.31
No of observations	831	203	203	203	203	195	195

R-squared vs number of x variables



Model 4 Output:

```
Call:
lm(formula = clean_data$distance ~ clean_data$speed_ground +
    clean_data$speed_air + clean_data$aircraft + clean_data$height,
    data = clean_data)

Residuals:
    Min      1Q  Median      3Q     Max
-289.38  -99.24   14.36   92.48  335.88

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)          -6390.230    109.865 -58.164   <2e-16 ***
clean_data$speed_ground    -5.886      6.183  -0.952    0.342
clean_data$speed_air       88.049      6.275  14.032   <2e-16 ***
clean_data$aircraft       426.955     19.185  22.255   <2e-16 ***
clean_data$height          13.640      1.010  13.511   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 134.3 on 198 degrees of freedom
  (628 observations deleted due to missingness)
Multiple R-squared:  0.9738,    Adjusted R-squared:  0.9733
F-statistic:  1843 on 4 and 198 DF,  p-value: < 2.2e-16

> extractAIC(model15)
[1]    5.000 1994.294
```

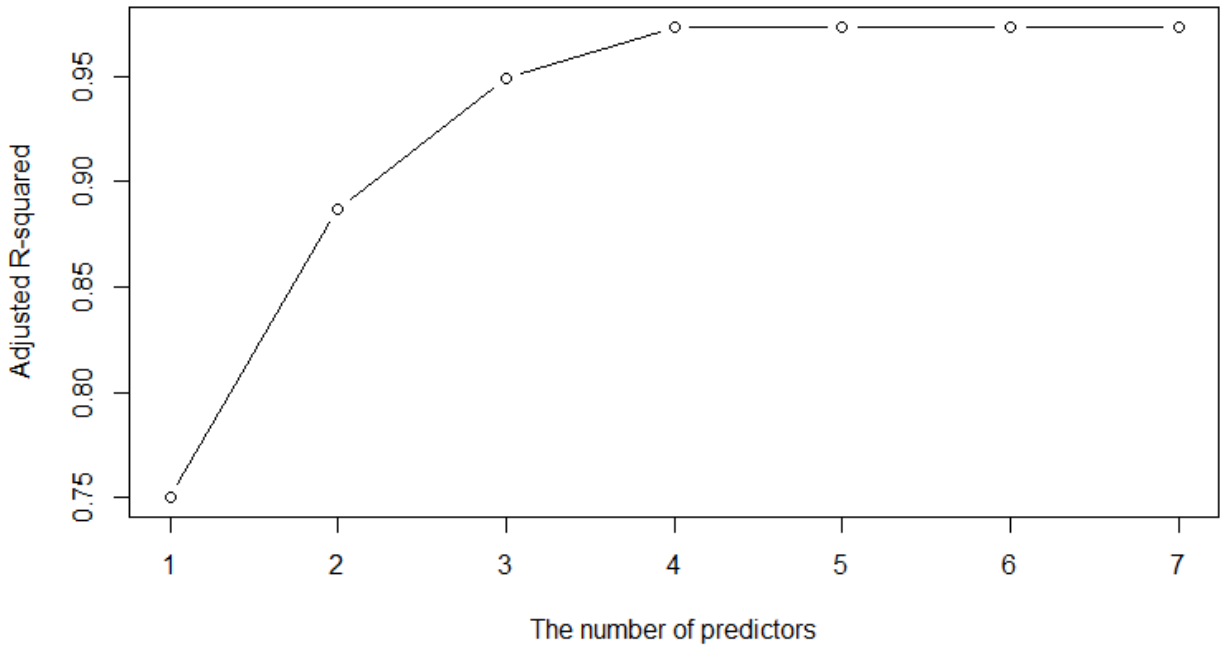
Step 18: Plot Adjusted R-squared vs Number of Variables

The adjusted R-squared for all the models are lower than the R-squared values. Although, the pattern is like that of the R-squared, as p increases, the adjusted r-squared value slightly decreased from Model 4 to Model 5. This implies that the addition of pitch as a new variable brought no new information to the model. This inclusion didn't help improve the model either.

Code:

```
adj.r.squared.1<- summary(modell4a)$ adj.r.squared ; print(adj.r.squared.1)
adj.r.squared.2<- summary(modell4b)$ adj.r.squared ; print(adj.r.squared.2)
adj.r.squared.3<- summary(modell4c)$ adj.r.squared ; print(adj.r.squared.3)
adj.r.squared.4<- summary(modell5)$ adj.r.squared ; print(adj.r.squared.4)
adj.r.squared.5<- summary(modell6)$ adj.r.squared ; print(adj.r.squared.5)
adj.r.squared.6<- summary(modell7)$ adj.r.squared ; print(adj.r.squared.6)
adj.r.squared.7<- summary(modell8)$ adj.r.squared ; print(adj.r.squared.7)

plot(c(1:7),c(adj.r.squared.1,adj.r.squared.2,adj.r.squared.3,adj.r.squared.4,
              adj.r.squared.5,adj.r.squared.6,
              adj.r.squared.7),type="b",ylab ="Adjusted R-squared", xlab ="The number of predictors")
```

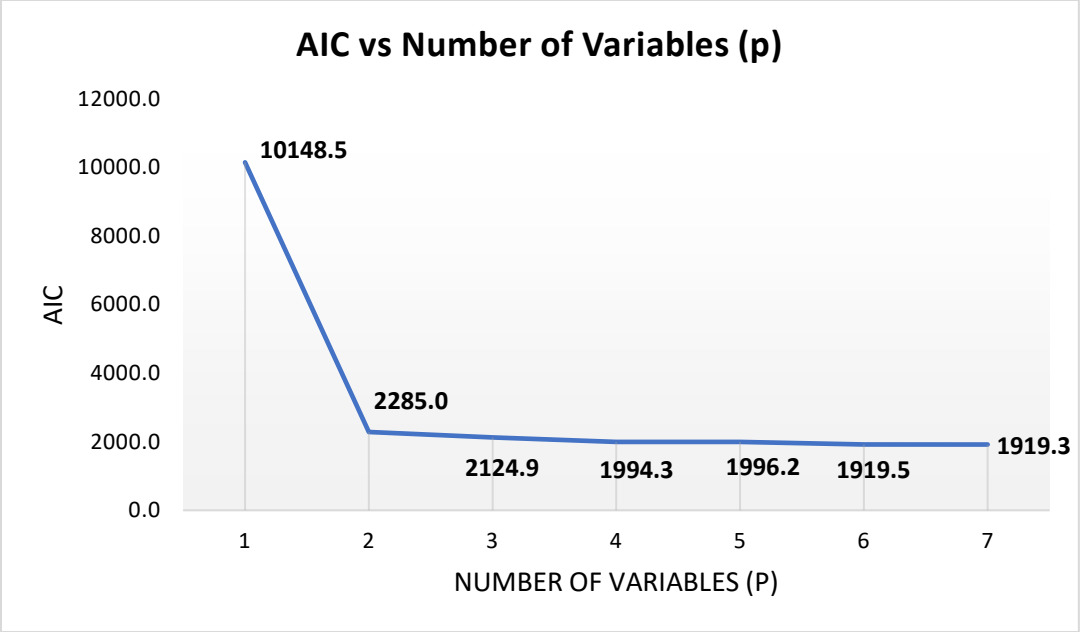
Adjusted R-squared vs number of x variables



Step 19: Plot AIC vs Number of Variables

The ***extractAIC( )*** function is used in extracting the AIC value for each of the models. The code is shown in Step 17. As shown in the plot below, the AIC value decreased from Model 1 to 4. While there was an increase from Model 4 to 5. Given that the AIC value depends on the number of observations, it is difficult to compare all the models because they have different number of observations. However, Model 2 to 5 have the same number of observations while model 5 and 6 have lower number of observations.

Comparing models 2,3,4, and 5, the model with the smallest AIC value is model 4. The increase in the AIC value in model 5 suggests that the inclusion of a new variable didn't provide a new information or improvement of the model. Similar conclusion was made in Steps 17 and 18. Hence, the most preferred model is Model 4, with the following X variables: Speed_ground, Speed_air, Aircraft, and Height.



Step 20. Compare the results in Steps 18-19, what variables would you select to build a predictive model for LD?

Based on the results and decisions in Steps 18-19, I would select Speed_ground, Speed_air, type of aircraft, and height as variables in building a predictive model for LD. However, I would test to confirm the presence of multicollinearity. If collinearity exists between Speed_ground and Speed_air, I will consider Speed_ground instead of Speed_air because of the large number of missing observations.

Variable selection based on automate algorithm.

Step 21:

Given the high number of missing observations in the Speed_air variable and the strong correlation with Speed_ground, Speed_air is removed in this step. This is because the ***stepAIC ()*** function in the ***MASS package*** wouldn't work with missing values. Looking at the output, the model with the lowest AIC value of 9691.2 is that in which the x's are speed_ground, aircraft, and height. This is consistent with the selected variables in Steps 17-20.

Compared to the result in Step 19, the best model here has the same variables as in the manually selected model. The variables selected manually are consistent with that selected using the automate algorithm.

In conclusion, the following variables are the most significant in predicting landing distance: Speed_ground, Aircraft, and Height. In other words, the ground speed of the aircraft, the make of the aircraft, and the height of the aircraft when it is passing over the threshold of the runway are the three most important and significant factors in predicting the aircraft's landing distance.

Code:

```
install.packages("MASS")
require(MASS)
Model1_LM <- lm(distance ~ 1, data = clean_data[,-5])
fit_maxim <- lm(distance~.,data=clean_data[,-5])

Model1_LM <- stepAIC(Model1_LM, direction = 'forward',
                     scope=list(upper=fit_maxim,lower= Model1_LM))
```

Output:

```
Start:  AIC=11299.8
distance ~ 1

               Df Sum of Sq       RSS   AIC
+ speed_ground  1 480561690 157699570 10104
+ aircraft      1  33759132 604502127 11220
+ height        1   6866417 631394842 11256
+ pitch         1   3010731 635250529 11262
+ duration      1   1685114 636576145 11263
<none>                      638261260 11263
+ no_pasg       1    181284 638079976 11265

Step:  AIC=10148.53
distance ~ speed_ground

           Df Sum of Sq       RSS     AIC
+ aircraft  1  47102191 110597379  9810.8
+ height    1  14123617 143575953 10027.6
+ pitch     1   8246571 149453000 10061.0
<none>                  157699570 10103.6
+ no_pasg   1    154554 157545016 10104.8
+ duration  1     50570 157649000 10105.4

Step:  AIC=9854.77
distance ~ speed_ground + aircraft

           Df Sum of Sq       RSS    AIC
+ height    1  15048298  95549081 9691.2
<none>                  110597379 9810.8
+ pitch     1    182007 110415372 9811.4
+ no_pasg   1     41575 110555804 9812.5
+ duration  1      9394 110587985 9812.7

Step:  AIC=9735.37
distance ~ speed_ground + aircraft + height

           Df Sum of Sq      RSS    AIC
<none>                  95549081 9691.2
+ no_pasg   1    120379 95428702 9692.2
+ pitch     1     71174 95477907 9692.6
+ duration  1      4446 95544635 9693.2
```