Final Project Proposal STAT167

Group Name: Statistically Speaking
Shreya Mohan, Kalyani Mantirraju, Crystal Arevalo,
Karen Alvarez, Mason Lam, Eric Yang

2025-04-27

Installation & Packages

```
library(nycflights13)
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(car)
```

```
Loading required package: carData

Attaching package: 'car'

The following object is masked from 'package:dplyr':

    recode

The following object is masked from 'package:purrr':

    some
```

```r
library(dunn.test)
```

Introduction

The primary goal of this research is to explore factors influencing flight delays from New City airports in 2013.

Problem Statement and Motivation

Understanding factors that contribute to flight delays is critical for informing Federal Aviation Administration (FAA) policies and guiding airlines and airports in improving operational efficiency, enhancing weather preparedness, and reducing delays through controllable factors. By analyzing weather conditions, airline differences, holiday effects, fleet age, and airport specific challenges, this research can provide data-driven insights to optimize air travel and ensure compliance with aviation regulations in heavily congested areas like New York City.

Main Research Question

What are the key correlations between flight delays from NYC airports?

Sub-questions:

The following questions will guide the analysis:

1. How do weather conditions affect flight delays?
    a. Are specific weather variables (e.g., precipitation, wind speed, humidity) correlated with departure and arrival delays?
2. How do differences between airlines influence flight delays?
    a. Do certain airlines experience more delays than other, if so, what operational or fleet-related factors contribute to these differences?
    b. How do metrics like cancelation rates and plane speed vary across airlines, and what impact do these metrics have on delays?
3. Are delays more frequent during major holidays?
    a. Are there differences during peak travel periods (e.g., Thanksgiving, Christsmas, New Year's Day)
4. Does the age of the plane affect flight delays?
    a. Do older planes experience more delays compared to newer ones?
    b. Are there specific plane models or manufactures associated with better on-time performance?
5. How do environmental factors like humidity, visibility, and wind affect flight delays?

    a. Are these effects observed across all airports?
6. What impact does precipitation have on specific airports and weather-related delays?
    a. Do airports in regions with higher average precipitation experience more delays?

Datasets

1. Flights dataset: All flights that departed from NYC in 2013

```
head(flights)
```

```
# A tibble: 6 x 19
   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
  <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
1  2013     1     1      517            515         2      830            819
2  2013     1     1      533            529         4      850            830
3  2013     1     1      542            540         2      923            850
4  2013     1     1      544            545        -1     1004           1022
5  2013     1     1      554            600        -6      812            837
6  2013     1     1      554            558        -4      740            728
# i 11 more variables: arr_delay <dbl>, carrier <chr>, flight <int>,
#   tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>,
#   hour <dbl>, minute <dbl>, time_hour <dttm>
```

```
dim(flights)
```

```
[1] 336776     19
```

```
names(flights)
```

```
 [1] "year"           "month"          "day"            "dep_time"
 [5] "sched_dep_time" "dep_delay"      "arr_time"       "sched_arr_time"
 [9] "arr_delay"      "carrier"        "flight"         "tailnum"
[13] "origin"         "dest"           "air_time"       "distance"
[17] "hour"           "minute"         "time_hour"
```

```
str(flights)
```

```
tibble [336,776 x 19] (S3: tbl_df/tbl/data.frame)
 $ year          : int [1:336776] 2013 2013 2013 2013 2013 2013 2013 2013 2013 2013 ...
 $ month         : int [1:336776] 1 1 1 1 1 1 1 1 1 1 ...
 $ day           : int [1:336776] 1 1 1 1 1 1 1 1 1 1 ...
 $ dep_time      : int [1:336776] 517 533 542 544 554 554 555 557 557 558 ...
```

```
 $ sched_dep_time: int [1:336776] 515 529 540 545 600 558 600 600 600 600 ...
 $ dep_delay     : num [1:336776] 2 4 2 -1 -6 -4 -5 -3 -3 -2 ...
 $ arr_time      : int [1:336776] 830 850 923 1004 812 740 913 709 838 753 ...
 $ sched_arr_time: int [1:336776] 819 830 850 1022 837 728 854 723 846 745 ...
 $ arr_delay     : num [1:336776] 11 20 33 -18 -25 12 19 -14 -8 8 ...
 $ carrier       : chr [1:336776] "UA" "UA" "AA" "B6" ...
 $ flight        : int [1:336776] 1545 1714 1141 725 461 1696 507 5708 79 301 ...
 $ tailnum       : chr [1:336776] "N14228" "N24211" "N619AA" "N804JB" ...
 $ origin        : chr [1:336776] "EWR" "LGA" "JFK" "JFK" ...
 $ dest          : chr [1:336776] "IAH" "IAH" "MIA" "BQN" ...
 $ air_time      : num [1:336776] 227 227 160 183 116 150 158 53 140 138 ...
 $ distance      : num [1:336776] 1400 1416 1089 1576 762 ...
 $ hour          : num [1:336776] 5 5 5 5 6 5 6 6 6 6 ...
 $ minute        : num [1:336776] 15 29 40 45 0 58 0 0 0 0 ...
 $ time_hour     : POSIXct[1:336776], format: "2013-01-01 05:00:00" "2013-01-01 05:00:00" ...
```

```
1 glimpse(flights)
```

```
Rows: 336,776
Columns: 19
$ year           <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2~
$ month          <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
$ day            <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
$ dep_time       <int> 517, 533, 542, 544, 554, 554, 555, 557, 557, 558, 558, ~
$ sched_dep_time <int> 515, 529, 540, 545, 600, 558, 600, 600, 600, 600, 600, ~
$ dep_delay      <dbl> 2, 4, 2, -1, -6, -4, -5, -3, -3, -2, -2, -2, -2, -2, -1~
$ arr_time       <int> 830, 850, 923, 1004, 812, 740, 913, 709, 838, 753, 849,~
$ sched_arr_time <int> 819, 830, 850, 1022, 837, 728, 854, 723, 846, 745, 851,~
$ arr_delay      <dbl> 11, 20, 33, -18, -25, 12, 19, -14, -8, 8, -2, -3, 7, -1~
$ carrier        <chr> "UA", "UA", "AA", "B6", "DL", "UA", "B6", "EV", "B6", "~
$ flight         <int> 1545, 1714, 1141, 725, 461, 1696, 507, 5708, 79, 301, 4~
$ tailnum        <chr> "N14228", "N24211", "N619AA", "N804JB", "N668DN", "N394~
$ origin         <chr> "EWR", "LGA", "JFK", "JFK", "LGA", "EWR", "EWR", "LGA",~
$ dest           <chr> "IAH", "IAH", "MIA", "BQN", "ATL", "ORD", "FLL", "IAD",~
$ air_time       <dbl> 227, 227, 160, 183, 116, 150, 158, 53, 140, 138, 149, 1~
$ distance       <dbl> 1400, 1416, 1089, 1576, 762, 719, 1065, 229, 944, 733, ~
$ hour           <dbl> 5, 5, 5, 5, 6, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 5, 6, 6, 6~
$ minute         <dbl> 15, 29, 40, 45, 0, 58, 0, 0, 0, 0, 0, 0, 0, 0, 0, 59, 0~
$ time_hour      <dttm> 2013-01-01 05:00:00, 2013-01-01 05:00:00, 2013-01-01 0~
```

Variables:

- flights (year, month, day, dep_time, arr_time, sched_dep_time, sched_arr_time, dep_delay, arr_delay, carrier, origin, dest, air_time, distance, time_hour)
  - year, month, day : date of departure
  - dep_time, arr_time : actual departure and arrival times in HHMM
  - sched_dep_time, sched_arr_time : scheduled departure and arrival times in HHMM

- dep_delay, arr_delay : departure and arrival delays in minutes
- carrier : two letter carrier abbreviation of the carrier
- origin, dest : origin and destination
- air_time : amount of time spent in air in minutes
- distance : distance between airport in miles
- time_hour : scheduled date and hour of the flight as POSIXct date

2. Airlines dataset: Translation between two letter carrier codes and names

```
head(airlines)
```

```
# A tibble: 6 x 2
  carrier name
  <chr>   <chr>
1 9E      Endeavor Air Inc.
2 AA      American Airlines Inc.
3 AS      Alaska Airlines Inc.
4 B6      JetBlue Airways
5 DL      Delta Air Lines Inc.
6 EV      ExpressJet Airlines Inc.
```

```
dim(airlines)
```

```
[1] 16  2
```

```
names(airlines)
```

```
[1] "carrier" "name"
```

```
str(airlines)
```

```
tibble [16 x 2] (S3: tbl_df/tbl/data.frame)
 $ carrier: chr [1:16] "9E" "AA" "AS" "B6" ...
 $ name   : chr [1:16] "Endeavor Air Inc." "American Airlines Inc." "Alaska Airlines Inc." "JetBlue A
```

```
glimpse(airlines)
```

```
Rows: 16
Columns: 2
$ carrier <chr> "9E", "AA", "AS", "B6", "DL", "EV", "F9", "FL", "HA", "MQ", "O~
$ name    <chr> "Endeavor Air Inc.", "American Airlines Inc.", "Alaska Airline~
```

Variables:

- airlines (carrier, name)
  - carrier : two-letter abbreviation of the airlines
  - name : full name of the airlines

3. Airports dataset: Airport names and locations

```
head(airports)
```

```
# A tibble: 6 x 8
  faa   name                              lat   lon   alt    tz dst   tzone
  <chr> <chr>                           <dbl> <dbl> <dbl> <dbl> <chr> <chr>
1 04G   Lansdowne Airport                41.1 -80.6  1044    -5 A     America/Ne~
2 06A   Moton Field Municipal Airport    32.5 -85.7   264    -6 A     America/Ch~
3 06C   Schaumburg Regional              42.0 -88.1   801    -6 A     America/Ch~
4 06N   Randall Airport                  41.4 -74.4   523    -5 A     America/Ne~
5 09J   Jekyll Island Airport            31.1 -81.4    11    -5 A     America/Ne~
6 0A9   Elizabethton Municipal Airport   36.4 -82.2  1593    -5 A     America/Ne~
```

```
dim(airports)
```

```
[1] 1458    8
```

```
names(airports)
```

```
[1] "faa"   "name"  "lat"   "lon"   "alt"   "tz"    "dst"   "tzone"
```

```
str(airports)
```

```
tibble [1,458 x 8] (S3: tbl_df/tbl/data.frame)
 $ faa  : chr [1:1458] "04G" "06A" "06C" "06N" ...
 $ name : chr [1:1458] "Lansdowne Airport" "Moton Field Municipal Airport" "Schaumburg Regional" "Ran
 $ lat  : num [1:1458] 41.1 32.5 42 41.4 31.1 ...
 $ lon  : num [1:1458] -80.6 -85.7 -88.1 -74.4 -81.4 ...
 $ alt  : num [1:1458] 1044 264 801 523 11 ...
 $ tz   : num [1:1458] -5 -6 -6 -5 -5 -5 -5 -5 -5 -8 ...
 $ dst  : chr [1:1458] "A" "A" "A" "A" ...
 $ tzone: chr [1:1458] "America/New_York" "America/Chicago" "America/Chicago" "America/New_York" ...
 - attr(*, "spec")=
  .. cols(
  ..   id = col_double(),
  ..   name = col_character(),
```

```
..   city = col_character(),
..   country = col_character(),
..   faa = col_character(),
..   icao = col_character(),
..   lat = col_double(),
..   lon = col_double(),
..   alt = col_double(),
..   tz = col_double(),
..   dst = col_character(),
..   tzone = col_character()
.. )
```

```
glimpse(airports)
```

```
Rows: 1,458
Columns: 8
$ faa   <chr> "04G", "06A", "06C", "06N", "09J", "0A9", "0G6", "0G7", "0P2", "~
$ name  <chr> "Lansdowne Airport", "Moton Field Municipal Airport", "Schaumbur~
$ lat   <dbl> 41.13047, 32.46057, 41.98934, 41.43191, 31.07447, 36.37122, 41.4~
$ lon   <dbl> -80.61958, -85.68003, -88.10124, -74.39156, -81.42778, -82.17342~
$ alt   <dbl> 1044, 264, 801, 523, 11, 1593, 730, 492, 1000, 108, 409, 875, 10~
$ tz    <dbl> -5, -6, -6, -5, -5, -5, -5, -5, -5, -8, -5, -6, -5, -5, -5, -5, ~
$ dst   <chr> "A", "A", "A", "A", "A", "A", "A", "A", "U", "A", "A", "U", "A",~
$ tzone <chr> "America/New_York", "America/Chicago", "America/Chicago", "Ameri~
```

Variables:

- airports (faa, name, lat, lon)
  - faa : FAA airport code
  - name : usual name of the airport
  - lat, lon : location of airport

4. Planes dataset: Construction information about each plane

```
head(planes)
```

```
# A tibble: 6 x 9
  tailnum  year type               manufacturer model engines seats speed engine
  <chr>   <int> <chr>              <chr>        <chr>   <int> <int> <int> <chr>
1 N10156   2004 Fixed wing multi ~ EMBRAER      EMB-~       2    55    NA Turbo~
2 N102UW   1998 Fixed wing multi ~ AIRBUS INDU~ A320~       2   182    NA Turbo~
3 N103US   1999 Fixed wing multi ~ AIRBUS INDU~ A320~       2   182    NA Turbo~
4 N104UW   1999 Fixed wing multi ~ AIRBUS INDU~ A320~       2   182    NA Turbo~
5 N10575   2002 Fixed wing multi ~ EMBRAER      EMB-~       2    55    NA Turbo~
6 N105UW   1999 Fixed wing multi ~ AIRBUS INDU~ A320~       2   182    NA Turbo~
```

```r
dim(planes)
```

```
[1] 3322    9
```

```r
names(planes)
```

```
[1] "tailnum"      "year"         "type"         "manufacturer" "model"
[6] "engines"      "seats"        "speed"        "engine"
```

```r
str(planes)
```

```
tibble [3,322 x 9] (S3: tbl_df/tbl/data.frame)
 $ tailnum     : chr [1:3322] "N10156" "N102UW" "N103US" "N104UW" ...
 $ year        : int [1:3322] 2004 1998 1999 1999 2002 1999 1999 1999 1999 1999 ...
 $ type        : chr [1:3322] "Fixed wing multi engine" "Fixed wing multi engine" "Fixed wing multi e
 $ manufacturer: chr [1:3322] "EMBRAER" "AIRBUS INDUSTRIE" "AIRBUS INDUSTRIE" "AIRBUS INDUSTRIE" ...
 $ model       : chr [1:3322] "EMB-145XR" "A320-214" "A320-214" "A320-214" ...
 $ engines     : int [1:3322] 2 2 2 2 2 2 2 2 2 2 ...
 $ seats       : int [1:3322] 55 182 182 182 55 182 182 182 182 182 ...
 $ speed       : int [1:3322] NA NA NA NA NA NA NA NA NA NA ...
 $ engine      : chr [1:3322] "Turbo-fan" "Turbo-fan" "Turbo-fan" "Turbo-fan" ...
```

```r
glimpse(planes)
```

```
Rows: 3,322
Columns: 9
$ tailnum      <chr> "N10156", "N102UW", "N103US", "N104UW", "N10575", "N105UW~
$ year         <int> 2004, 1998, 1999, 1999, 2002, 1999, 1999, 1999, 1999, 199~
$ type         <chr> "Fixed wing multi engine", "Fixed wing multi engine", "Fi~
$ manufacturer <chr> "EMBRAER", "AIRBUS INDUSTRIE", "AIRBUS INDUSTRIE", "AIRBU~
$ model        <chr> "EMB-145XR", "A320-214", "A320-214", "A320-214", "EMB-145~
$ engines      <int> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, ~
$ seats        <int> 55, 182, 182, 182, 55, 182, 182, 182, 182, 182, 55, 55, 5~
$ speed        <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N~
$ engine       <chr> "Turbo-fan", "Turbo-fan", "Turbo-fan", "Turbo-fan", "Turb~
```

Variables:

- planes (year, type, manufacturer, model, engines, seats, speed, engine)
  - year : year manufactured
  - type : type of plane
  - manufacturer, model : manufacturer and model
  - engines, seats : number of engines and seats
  - speed : average cruising speed in mph
  - engine : type in engine

5. Weather dataset: Hourly meterological data for each airport

```
head(weather)
```

```
# A tibble: 6 x 15
  origin  year month   day  hour  temp  dewp humid wind_dir wind_speed wind_gust
  <chr>  <int> <int> <int> <int> <dbl> <dbl> <dbl>    <dbl>      <dbl>     <dbl>
1 EWR     2013     1     1     1  39.0  26.1  59.4      270      10.4         NA
2 EWR     2013     1     1     2  39.0  27.0  61.6      250       8.06        NA
3 EWR     2013     1     1     3  39.0  28.0  64.4      240      11.5         NA
4 EWR     2013     1     1     4  39.9  28.0  62.2      250      12.7         NA
5 EWR     2013     1     1     5  39.0  28.0  64.4      260      12.7         NA
6 EWR     2013     1     1     6  37.9  28.0  67.2      240      11.5         NA
# i 4 more variables: precip <dbl>, pressure <dbl>, visib <dbl>,
#   time_hour <dttm>
```

```
dim(weather)
```

```
[1] 26115    15
```

```
names(weather)
```

```
 [1] "origin"     "year"       "month"      "day"        "hour"
 [6] "temp"       "dewp"       "humid"      "wind_dir"   "wind_speed"
[11] "wind_gust"  "precip"     "pressure"   "visib"      "time_hour"
```

```
str(weather)
```

```
tibble [26,115 x 15] (S3: tbl_df/tbl/data.frame)
 $ origin    : chr [1:26115] "EWR" "EWR" "EWR" "EWR" ...
 $ year      : int [1:26115] 2013 2013 2013 2013 2013 2013 2013 2013 2013 2013 ...
 $ month     : int [1:26115] 1 1 1 1 1 1 1 1 1 1 ...
 $ day       : int [1:26115] 1 1 1 1 1 1 1 1 1 1 ...
 $ hour      : int [1:26115] 1 2 3 4 5 6 7 8 9 10 ...
 $ temp      : num [1:26115] 39 39 39 39.9 39 ...
 $ dewp      : num [1:26115] 26.1 27 28 28 28 ...
 $ humid     : num [1:26115] 59.4 61.6 64.4 62.2 64.4 ...
 $ wind_dir  : num [1:26115] 270 250 240 250 260 240 240 250 260 260 ...
 $ wind_speed: num [1:26115] 10.36 8.06 11.51 12.66 12.66 ...
 $ wind_gust : num [1:26115] NA NA NA NA NA NA NA NA NA NA ...
 $ precip    : num [1:26115] 0 0 0 0 0 0 0 0 0 0 ...
 $ pressure  : num [1:26115] 1012 1012 1012 1012 1012 ...
 $ visib     : num [1:26115] 10 10 10 10 10 10 10 10 10 10 ...
 $ time_hour : POSIXct[1:26115], format: "2013-01-01 01:00:00" "2013-01-01 02:00:00" ...
```

```r
glimpse(weather)
```

```
Rows: 26,115
Columns: 15
$ origin     <chr> "EWR", "EWR", "EWR", "EWR", "EWR", "EWR", "EWR", "EWR", "EW~
$ year       <int> 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013, 2013,~
$ month      <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
$ day        <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
$ hour       <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15, 16, 17, 18, ~
$ temp       <dbl> 39.02, 39.02, 39.02, 39.92, 39.02, 37.94, 39.02, 39.92, 39.~
$ dewp       <dbl> 26.06, 26.96, 28.04, 28.04, 28.04, 28.04, 28.04, 28.04, 28.~
$ humid      <dbl> 59.37, 61.63, 64.43, 62.21, 64.43, 67.21, 64.43, 62.21, 62.~
$ wind_dir   <dbl> 270, 250, 240, 250, 260, 240, 240, 250, 260, 260, 260, 330,~
$ wind_speed <dbl> 10.35702, 8.05546, 11.50780, 12.65858, 12.65858, 11.50780, ~
$ wind_gust  <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, 20.~
$ precip     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ pressure   <dbl> 1012.0, 1012.3, 1012.5, 1012.2, 1011.9, 1012.4, 1012.2, 101~
$ visib      <dbl> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10,~
$ time_hour  <dttm> 2013-01-01 01:00:00, 2013-01-01 02:00:00, 2013-01-01 03:00~
```

Variables:

- weather (origin, year, month, day, hour, temp, dewp, humid, wind_dir, wind_speed, wind_gust, precip, pressure, visib, time_hour)
  - origin : weather station
  - year, month, day, hour : time of recording
  - temp, dewp : temperature and dew point in Fahrenheit
  - humid : relative humidity
  - wind_dir, wind_speed, wind_gust : wind direction in degrees, wind speed and gust in mph
  - precip : precipitation in inches
  - pressure : sea level pressure in millibars
  - visib : visibility in miles
  - time_hour : date and hour of the recording as POSIXct date

We can join the data tables by combining them through similar attributes, such as combining flights : time_hour with weather : time_hour.

EDA:

Summary Statistics & Check for Missing Values:

```r
summary(flights)
```

```
     year          month             day           dep_time    sched_dep_time
Min.   :2013   Min.   : 1.000   Min.   : 1.00   Min.   :   1   Min.   : 106
1st Qu.:2013   1st Qu.: 4.000   1st Qu.: 8.00   1st Qu.: 907   1st Qu.: 906
Median :2013   Median : 7.000   Median :16.00   Median :1401   Median :1359
Mean   :2013   Mean   : 6.549   Mean   :15.71   Mean   :1349   Mean   :1344
3rd Qu.:2013   3rd Qu.:10.000   3rd Qu.:23.00   3rd Qu.:1744   3rd Qu.:1729
Max.   :2013   Max.   :12.000   Max.   :31.00   Max.   :2400   Max.   :2359
                                                NA's   :8255
  dep_delay          arr_time    sched_arr_time   arr_delay
Min.   : -43.00   Min.   :   1   Min.   :   1   Min.   : -86.000
1st Qu.:  -5.00   1st Qu.:1104   1st Qu.:1124   1st Qu.: -17.000
Median :  -2.00   Median :1535   Median :1556   Median :  -5.000
Mean   :  12.64   Mean   :1502   Mean   :1536   Mean   :   6.895
3rd Qu.:  11.00   3rd Qu.:1940   3rd Qu.:1945   3rd Qu.:  14.000
Max.   :1301.00   Max.   :2400   Max.   :2359   Max.   :1272.000
NA's   :8255      NA's   :8713                  NA's   :9430
  carrier             flight       tailnum             origin
Length:336776      Min.   :   1   Length:336776      Length:336776
Class :character   1st Qu.: 553   Class :character   Class :character
Mode  :character   Median :1496   Mode  :character   Mode  :character
                   Mean   :1972
                   3rd Qu.:3465
                   Max.   :8500

    dest              air_time        distance         hour
Length:336776      Min.   : 20.0   Min.   :  17   Min.   : 1.00
Class :character   1st Qu.: 82.0   1st Qu.: 502   1st Qu.: 9.00
Mode  :character   Median :129.0   Median : 872   Median :13.00
                   Mean   :150.7   Mean   :1040   Mean   :13.18
                   3rd Qu.:192.0   3rd Qu.:1389   3rd Qu.:17.00
                   Max.   :695.0   Max.   :4983   Max.   :23.00
                   NA's   :9430
    minute        time_hour
Min.   : 0.00   Min.   :2013-01-01 05:00:00.00
1st Qu.: 8.00   1st Qu.:2013-04-04 13:00:00.00
Median :29.00   Median :2013-07-03 10:00:00.00
Mean   :26.23   Mean   :2013-07-03 05:22:54.64
3rd Qu.:44.00   3rd Qu.:2013-10-01 07:00:00.00
Max.   :59.00   Max.   :2013-12-31 23:00:00.00
```

```
1 colSums(is.na(flights))
```

```
         year          month            day       dep_time sched_dep_time
            0              0              0           8255              0
    dep_delay       arr_time sched_arr_time      arr_delay        carrier
         8255           8713              0           9430              0
       flight        tailnum         origin           dest       air_time
```

```
           0          2512             0             0          9430
    distance          hour        minute     time_hour
           0             0             0             0
```

```
summary(airlines)
```

```
   carrier              name
 Length:16          Length:16
 Class :character   Class :character
 Mode  :character   Mode  :character
```

```
colSums(is.na(airlines))
```

```
carrier    name
      0       0
```

```
summary(airports)
```

```
     faa                name               lat              lon
 Length:1458        Length:1458        Min.   :19.72   Min.   :-176.65
 Class :character   Class :character   1st Qu.:34.26   1st Qu.:-119.19
 Mode  :character   Mode  :character   Median :40.09   Median : -94.66
                                       Mean   :41.65   Mean   :-103.39
                                       3rd Qu.:45.07   3rd Qu.: -82.52
                                       Max.   :72.27   Max.   : 174.11
      alt               tz             dst              tzone
 Min.   : -54.00   Min.   :-10.000   Length:1458        Length:1458
 1st Qu.:  70.25   1st Qu.: -8.000   Class :character   Class :character
 Median : 473.00   Median : -6.000   Mode  :character   Mode  :character
 Mean   :1001.42   Mean   : -6.519
 3rd Qu.:1062.50   3rd Qu.: -5.000
 Max.   :9078.00   Max.   :  8.000
```

```
colSums(is.na(airports))
```

```
  faa  name   lat   lon   alt    tz   dst tzone
    0     0     0     0     0     0     0     3
```

```
summary(planes)
```

```
   tailnum               year         type           manufacturer
 Length:3322        Min.   :1956   Length:3322        Length:3322
 Class :character   1st Qu.:1997   Class :character   Class :character
 Mode  :character   Median :2001   Mode  :character   Mode  :character
                    Mean   :2000
```

```
                  3rd Qu.:2005
                  Max.   :2013
                  NA's   :70
   model              engines          seats           speed
Length:3322        Min.   :1.000   Min.   :  2.0   Min.   : 90.0
Class :character   1st Qu.:2.000   1st Qu.:140.0   1st Qu.:107.5
Mode  :character   Median :2.000   Median :149.0   Median :162.0
                   Mean   :1.995   Mean   :154.3   Mean   :236.8
                   3rd Qu.:2.000   3rd Qu.:182.0   3rd Qu.:432.0
                   Max.   :4.000   Max.   :450.0   Max.   :432.0
                                                   NA's   :3299
   engine
Length:3322
Class :character
Mode  :character
```

```
colSums(is.na(planes))
```

```
    tailnum         year         type manufacturer        model      engines
          0           70            0            0            0            0
      seats        speed       engine
          0         3299            0
```

```
summary(weather)
```

```
   origin               year          month             day
Length:26115       Min.   :2013   Min.   : 1.000   Min.   : 1.00
Class :character   1st Qu.:2013   1st Qu.: 4.000   1st Qu.: 8.00
Mode  :character   Median :2013   Median : 7.000   Median :16.00
                   Mean   :2013   Mean   : 6.504   Mean   :15.68
                   3rd Qu.:2013   3rd Qu.: 9.000   3rd Qu.:23.00
                   Max.   :2013   Max.   :12.000   Max.   :31.00

     hour            temp            dewp            humid
Min.   : 0.00   Min.   : 10.94   Min.   :-9.94   Min.   : 12.74
1st Qu.: 6.00   1st Qu.: 39.92   1st Qu.:26.06   1st Qu.: 47.05
Median :11.00   Median : 55.40   Median :42.08   Median : 61.79
Mean   :11.49   Mean   : 55.26   Mean   :41.44   Mean   : 62.53
3rd Qu.:17.00   3rd Qu.: 69.98   3rd Qu.:57.92   3rd Qu.: 78.79
Max.   :23.00   Max.   :100.04   Max.   :78.08   Max.   :100.00
                NA's   :1        NA's   :1       NA's   :1
   wind_dir       wind_speed       wind_gust         precip
Min.   :  0.0   Min.   :   0.000   Min.   :16.11   Min.   :0.000000
```

```
 1st Qu.:120.0   1st Qu.:   6.905   1st Qu.:20.71   1st Qu.:0.000000
 Median :220.0   Median :  10.357   Median :24.17   Median :0.000000
 Mean   :199.8   Mean   :  10.518   Mean   :25.49   Mean   :0.004469
 3rd Qu.:290.0   3rd Qu.:  13.809   3rd Qu.:28.77   3rd Qu.:0.000000
 Max.   :360.0   Max.   :1048.361   Max.   :66.75   Max.   :1.210000
 NA's   :460     NA's   :4          NA's   :20778
    pressure          visib          time_hour
 Min.   : 983.8   Min.   : 0.000   Min.   :2013-01-01 01:00:00.0
 1st Qu.:1012.9   1st Qu.:10.000   1st Qu.:2013-04-01 21:30:00.0
 Median :1017.6   Median :10.000   Median :2013-07-01 14:00:00.0
 Mean   :1017.9   Mean   : 9.255   Mean   :2013-07-01 18:26:37.7
 3rd Qu.:1023.0   3rd Qu.:10.000   3rd Qu.:2013-09-30 13:00:00.0
 Max.   :1042.1   Max.   :10.000   Max.   :2013-12-30 18:00:00.0
 NA's   :2729
```

```
colSums(is.na(weather))
```

```
   origin      year     month       day      hour      temp      dewp
        0         0         0         0         0         1         1
    humid  wind_dir wind_speed  wind_gust    precip  pressure     visib
        1       460         4     20778         0      2729         0
time_hour
        0
```

Planes Dataset EDA

```
dim(planes)
```

```
[1] 3322    9
```

```
colnames(planes)
```

```
[1] "tailnum"      "year"         "type"         "manufacturer" "model"
[6] "engines"      "seats"        "speed"        "engine"
```

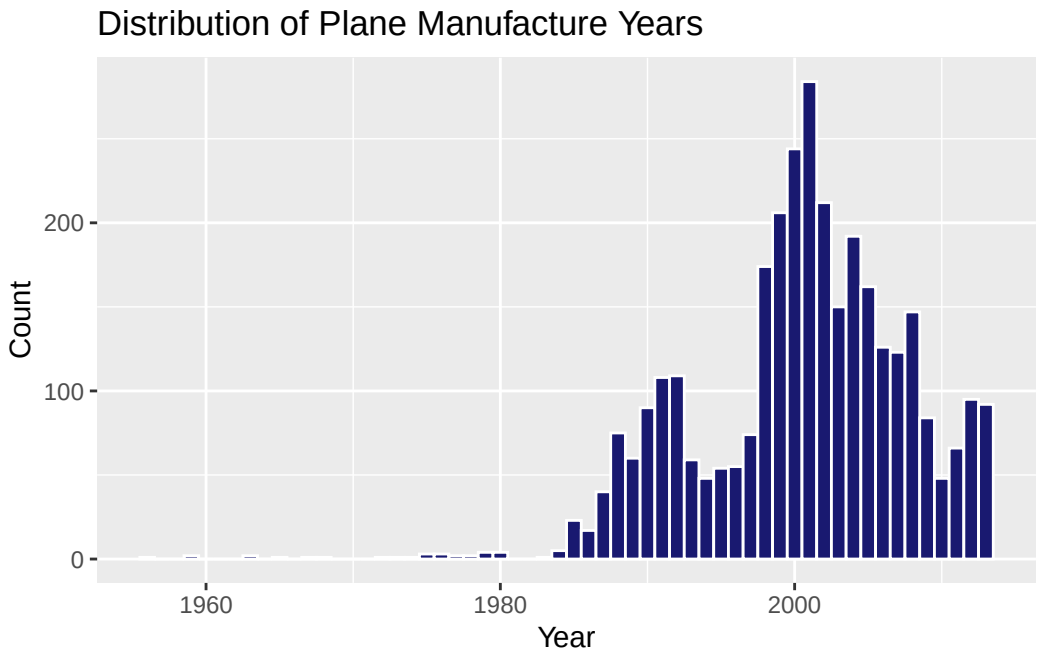
```
planes %>%
  count(manufacturer, sort = TRUE) %>%
  top_n(10) %>%
  ggplot(aes(x = reorder(manufacturer, n), y = n)) +
  geom_col(fill = "darkgreen") +
  coord_flip() +
  labs(title = "Top 10 Plane Manufacturers", x = "Manufacturer", y = "Number of Planes")
```

```
Selecting by n
```

The visualization above shows the top 10 plane manufacturers present in the data-set. Boeing has the largest amount of planes with approximately 1750 planes, and Airbus has the second most with approximately 400 planes.

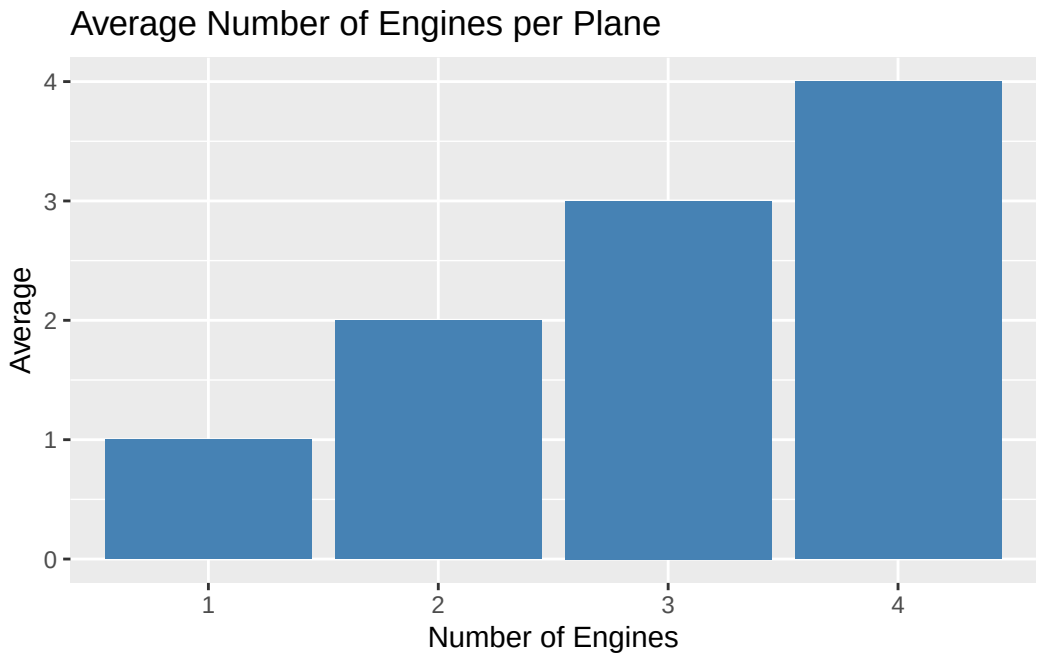
```
ggplot(planes, aes(x = year)) +
  geom_histogram(binwidth = 1, fill = "midnightblue", color = "white") +
  labs(title = "Distribution of Plane Manufacture Years", x = "Year", y = "Count")
```

```
Warning: Removed 70 rows containing non-finite outside the scale range
(`stat_bin()`).
```



This histogram shows the distribution of plane manufacture years, with the majority of planes built between the mid-1990s and early 2000s. There is a notable peak around the year 2000, indicating a surge in plane production during that period.

```
avg_engines <- planes %>%
  group_by(engines) %>%
  summarise(avg = mean(engines, na.rm = TRUE))

# Create the bar plot
ggplot(avg_engines, aes(x = factor(engines), y = avg)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(title = "Average Number of Engines per Plane", x = "Number of Engines", y = "Average")
```



Airlines Dataset EDA

Dimensions and column names of the airlines dataset

```
dim(airlines)
```

```
[1] 16  2
```

```
colnames(airlines)
```

```
[1] "carrier" "name"
```

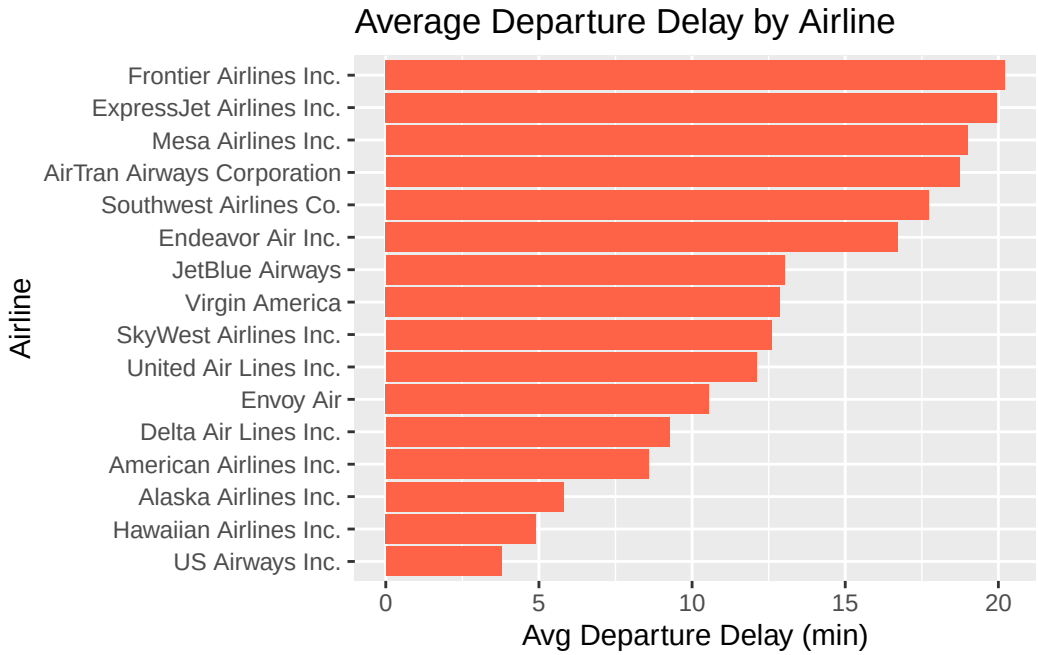
Viewing all the Unique Airlines:

```
airlines %>%
  arrange(name)
```

```
# A tibble: 16 x 2
   carrier name
   <chr>   <chr>
 1 FL      AirTran Airways Corporation
 2 AS      Alaska Airlines Inc.
 3 AA      American Airlines Inc.
```

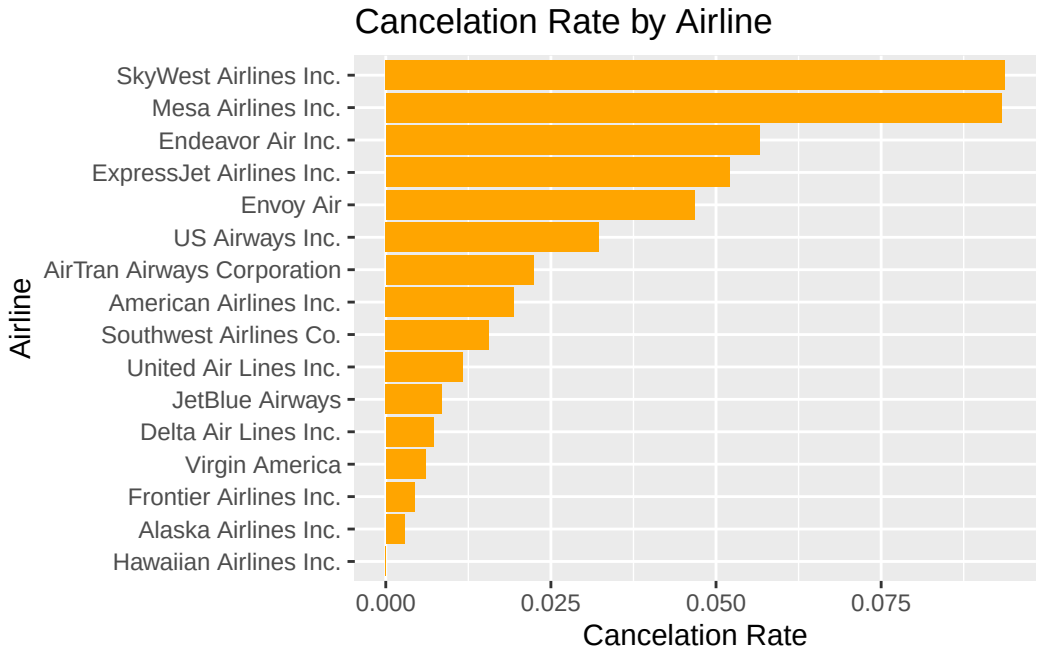
```
 4 DL      Delta Air Lines Inc.
 5 9E      Endeavor Air Inc.
 6 MQ      Envoy Air
 7 EV      ExpressJet Airlines Inc.
 8 F9      Frontier Airlines Inc.
 9 HA      Hawaiian Airlines Inc.
10 B6      JetBlue Airways
11 YV      Mesa Airlines Inc.
12 OO      SkyWest Airlines Inc.
13 WN      Southwest Airlines Co.
14 US      US Airways Inc.
15 UA      United Air Lines Inc.
16 VX      Virgin America
```

```r
# Join flights and airline names
flights_airlines <- flights %>%
  left_join(airlines, by = "carrier")

# Average delay metrics
avg_delays <- flights_airlines %>%
  group_by(name) %>%
  summarise(
    avg_dep_delay = mean(dep_delay, na.rm = TRUE),
    avg_arr_delay = mean(arr_delay, na.rm = TRUE),
    flights = n()
  )

# Plot: Departure Delay
ggplot(avg_delays, aes(x = reorder(name, avg_dep_delay), y = avg_dep_delay)) +
  geom_col(fill = "tomato") +
  coord_flip() +
  labs(
    title = "Average Departure Delay by Airline",
    x = "Airline",
    y = "Avg Departure Delay (min)"
  )
```



We can see that on average, Frontier Airlines has the most departure delay at around 20 min, with ExpressJet roughly around the same 20 minutes. Less than half the Airlines seem to be past the 13 minute delay mark.

```
cancel_rate <- flights_airlines %>%
  mutate(cancelled = is.na(dep_delay)) %>%
  group_by(name) %>%
  summarise(cancel_rate = mean(cancelled), total_flights = n())

ggplot(cancel_rate, aes(x = reorder(name, cancel_rate), y = cancel_rate)) +
  geom_col(fill = "orange") +
  coord_flip() +
  labs(
    title = "Cancelation Rate by Airline",
    x = "Airline",
    y = "Cancelation Rate"
  )
```



As we can see from above, Skywest Airlines Inc has the highest cancellation rate, with Mesa Airlines very closely behind, and a huge drop off at Endeavor Air Inc. ## Planes Dataset EDA

```
dim(planes)
```

```
[1] 3322    9
```

```
colnames(planes)
```

```
[1] "tailnum"      "year"         "type"         "manufacturer" "model"
[6] "engines"      "seats"        "speed"        "engine"
```

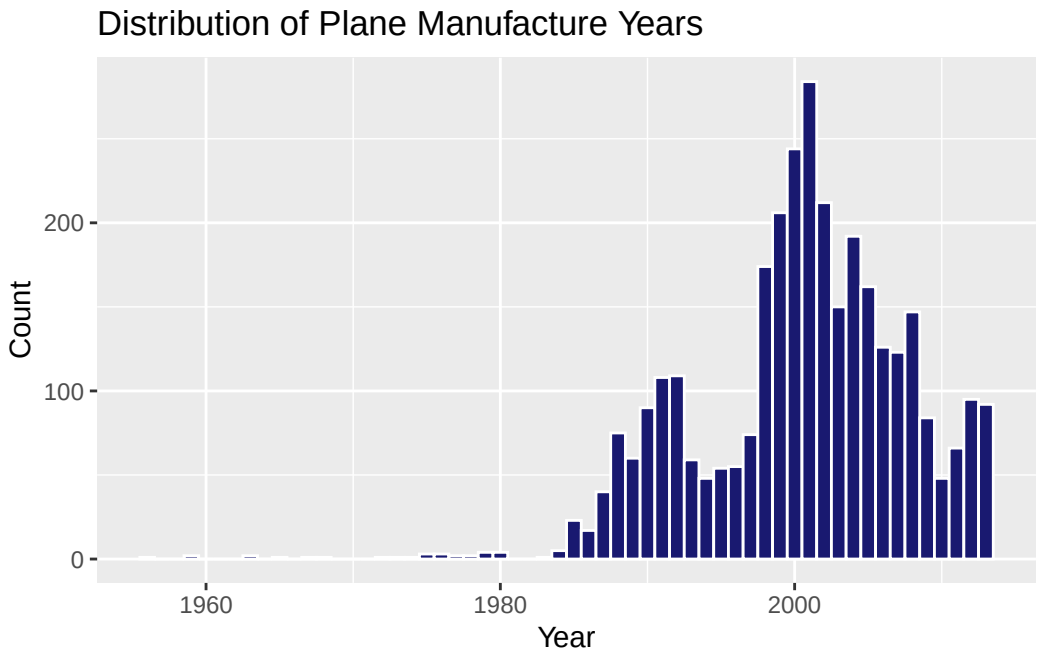
```
planes %>%
  count(manufacturer, sort = TRUE) %>%
  top_n(10) %>%
  ggplot(aes(x = reorder(manufacturer, n), y = n)) +
  geom_col(fill = "darkgreen") +
  coord_flip() +
  labs(title = "Top 10 Plane Manufacturers", x = "Manufacturer", y = "Number of Planes")
```

```
Selecting by n
```

The visualization above shows the top 10 plane manufacturers present in the data-set. Boeing has the largest amount of planes with approximately 1750 planes, and Airbus has the second most with approximately 400 planes.

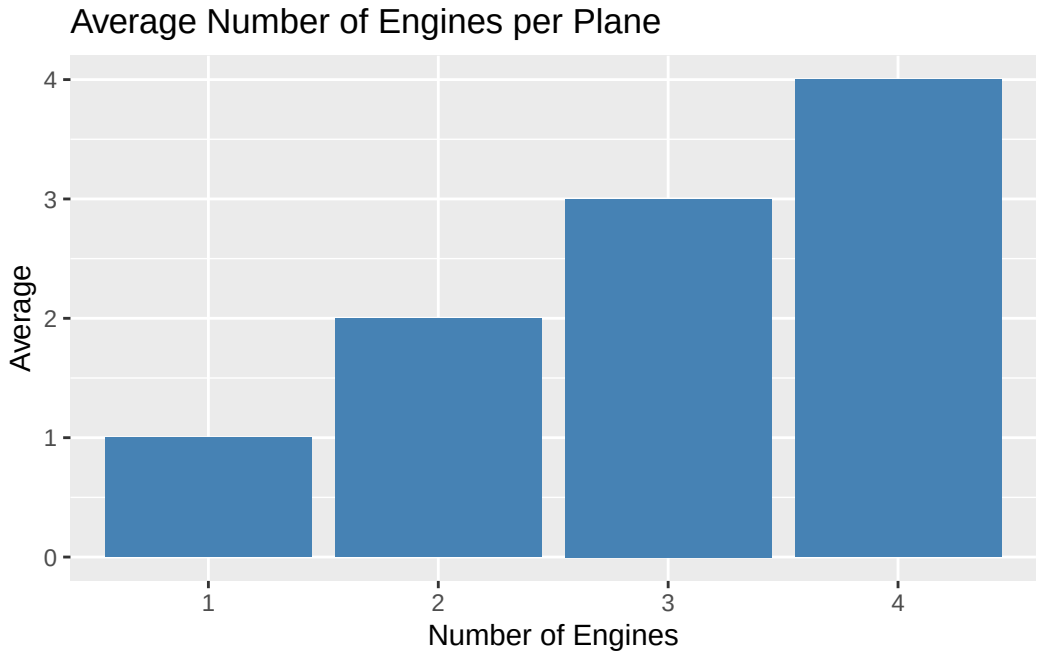
```
ggplot(planes, aes(x = year)) +
  geom_histogram(binwidth = 1, fill = "midnightblue", color = "white") +
  labs(title = "Distribution of Plane Manufacture Years", x = "Year", y = "Count")
```

```
Warning: Removed 70 rows containing non-finite outside the scale range
(`stat_bin()`).
```



This histogram shows the distribution of plane manufacture years, with the majority of planes built between the mid-1990s and early 2000s. There is a notable peak around the year 2000, indicating a surge in plane production during that period.

```r
avg_engines <- planes %>%
  group_by(engines) %>%
  summarise(avg = mean(engines, na.rm = TRUE))

# Create the bar plot
ggplot(avg_engines, aes(x = factor(engines), y = avg)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(title = "Average Number of Engines per Plane", x = "Number of Engines", y = "Average")
```



FLIGHTS EDA:

```
1 head(flights)
```

```
# A tibble: 6 x 19
   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
  <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
1  2013     1     1      517            515         2      830            819
2  2013     1     1      533            529         4      850            830
3  2013     1     1      542            540         2      923            850
4  2013     1     1      544            545        -1     1004           1022
5  2013     1     1      554            600        -6      812            837
6  2013     1     1      554            558        -4      740            728
# i 11 more variables: arr_delay <dbl>, carrier <chr>, flight <int>,
#   tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>,
#   hour <dbl>, minute <dbl>, time_hour <dttm>
```

Most of our analysis is based on how other variables and datasets affect and compare to the flights dataset. We are seeing how the arrival time, departure delay time, departure time, arrival delay time, and other variables are affected.

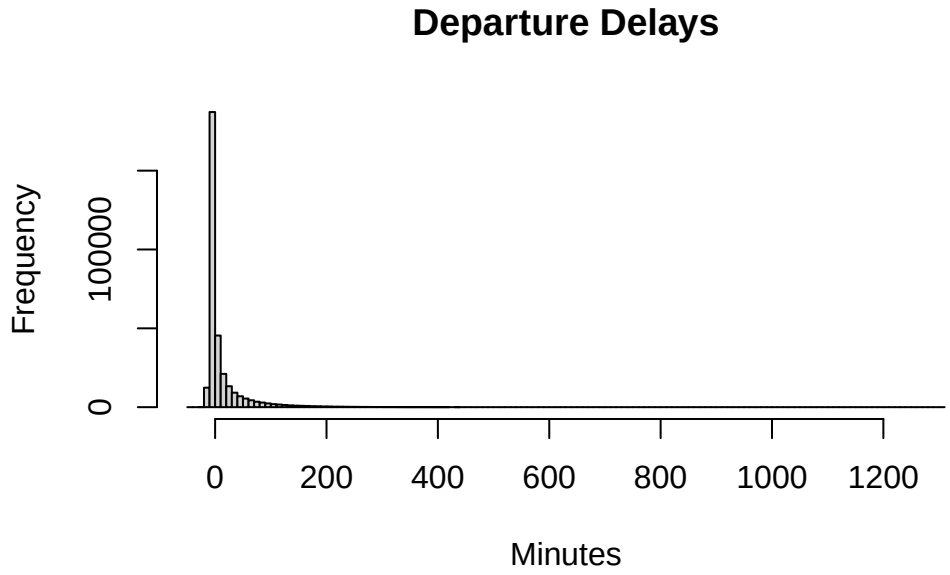
#flights that were not canceled ## We will be using these the not_canceled data for the rest of the EDA

```r
not_canceled <- filter(flights, !is.na(dep_delay), !is.na(arr_delay))
not_canceled
```

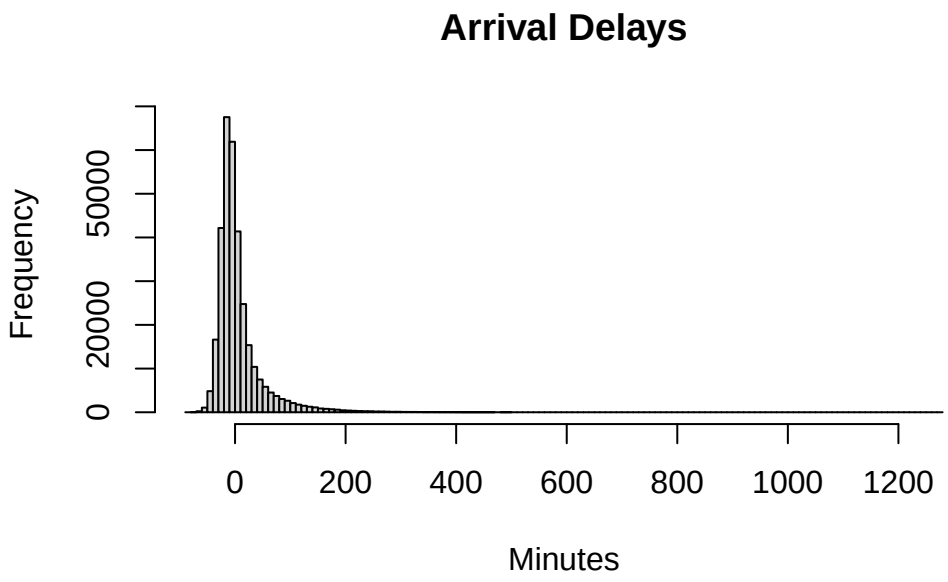
```
# A tibble: 327,346 x 19
    year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
   <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
 1  2013     1     1      517            515         2      830            819
 2  2013     1     1      533            529         4      850            830
 3  2013     1     1      542            540         2      923            850
 4  2013     1     1      544            545        -1     1004           1022
 5  2013     1     1      554            600        -6      812            837
 6  2013     1     1      554            558        -4      740            728
 7  2013     1     1      555            600        -5      913            854
 8  2013     1     1      557            600        -3      709            723
 9  2013     1     1      557            600        -3      838            846
10  2013     1     1      558            600        -2      753            745
# i 327,336 more rows
# i 11 more variables: arr_delay <dbl>, carrier <chr>, flight <int>,
#   tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>,
#   hour <dbl>, minute <dbl>, time_hour <dttm>
```

#Basic delay analysis ## Distribution and Proportion of delayed flights that were not canceled

```r
#histograms
hist(not_canceled$dep_delay, breaks=100, main = "Departure Delays",xlab = "Minutes")
```



```r
hist(not_canceled$arr_delay, breaks=100, main ="Arrival Delays", xlab = "Minutes")
```




```r
#proportions
mean(not_canceled$dep_delay>0, na.rm=TRUE)
```

```
[1] 0.3902446
```

```r
mean(not_canceled$arr_delay>0, na.rm=TRUE)
```

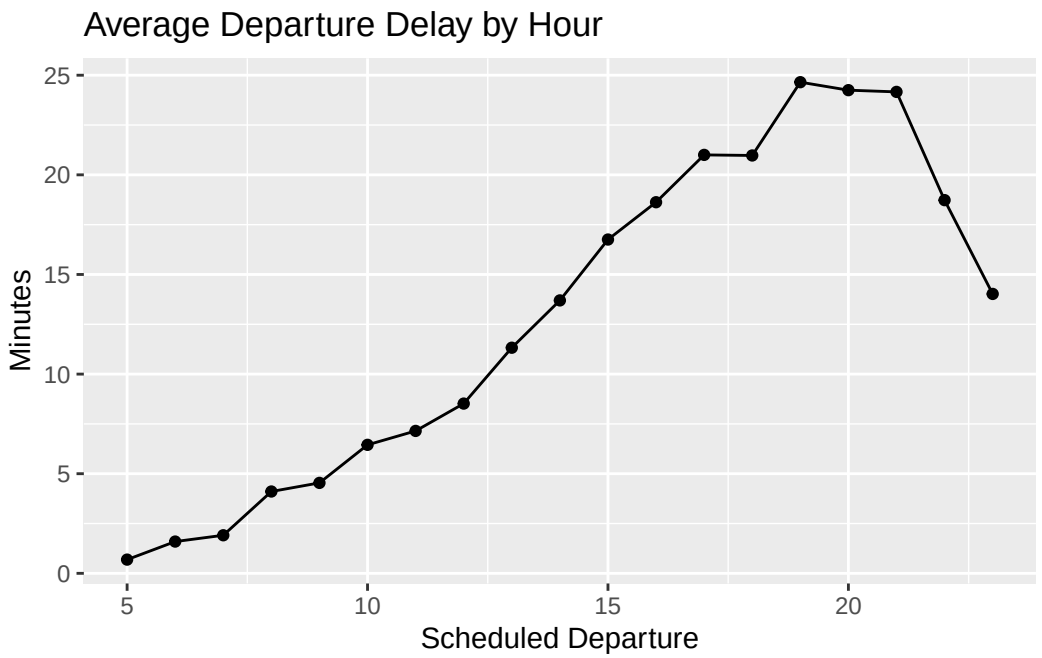
```
[1] 0.4063101
```

#most of the departure delays do not go over 200 minutes and the arrival delays have very few delays past 200 minutes.

#Delay patterns

```r
#convert time to hours
not_canceled$dep_hour <-floor(not_canceled$sched_dep_time/100)
not_canceled$arr_hour <-floor(not_canceled$sched_arr_time/100)

#plot
not_canceled |>
  group_by(dep_hour) |>
  summarize(mean_dep_delay = mean(dep_delay, na.rm=TRUE))|>
```

```
  ggplot(aes(x=dep_hour, y =mean_dep_delay))+
  geom_line()+
  geom_point()+
  labs(title = "Average Departure Delay by Hour", x="Scheduled Departure", y="Minutes")
```



#We can see that many of the delays happen further in the day and peak at about 18 hours and then it descends from there.

#Delays by Airport

```
not_canceled |>
  group_by(origin)|>
  summarize(avg_dep_delay= mean(dep_delay, na.rm=TRUE), avg_arr_delay= mean(arr_delay, na.rm=TRUE))
```

```
# A tibble: 3 x 3
  origin avg_dep_delay avg_arr_delay
  <chr>          <dbl>         <dbl>
1 EWR             15.0          9.11
2 JFK             12.0          5.55
3 LGA             10.3          5.78
```

#EWR has the highest average departure and arrival delay followed by JFK and then LGA

#Ranking airlines by delay

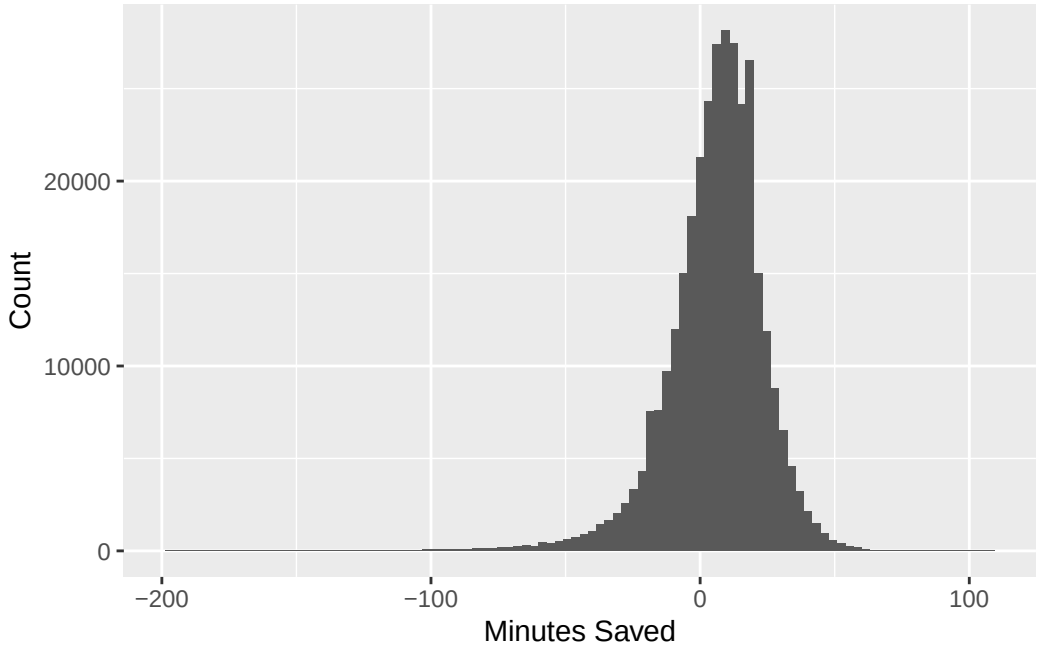
```r
not_canceled|>
  group_by(carrier)|>
  summarize(avg_dep_delay = mean(dep_delay, na.rn=TRUE))|>
  arrange(desc(avg_dep_delay))
```

```
# A tibble: 16 x 2
   carrier avg_dep_delay
   <chr>           <dbl>
 1 F9              20.2
 2 EV              19.8
 3 YV              18.9
 4 FL              18.6
 5 WN              17.7
 6 9E              16.4
 7 B6              13.0
 8 VX              12.8
 9 OO              12.6
10 UA              12.0
11 MQ              10.4
12 DL               9.22
13 AA               8.57
14 AS               5.83
15 HA               4.90
16 US               3.74
```

#F9 has the highest average departure delay at 20 hours.

#Check to see if the flights that were delayed made up the time in the air

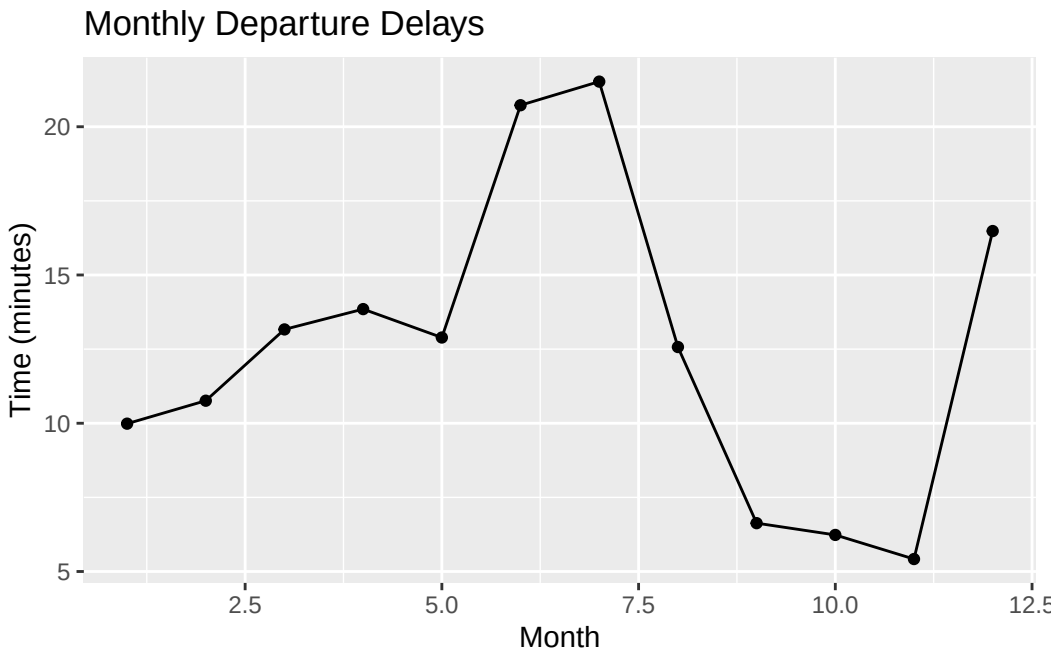
```r
not_canceled|>
  mutate(made_up_time = dep_delay- arr_delay)|>
  ggplot(aes(x=made_up_time))+
  geom_histogram(bins=100)+
  labs(titel="Made up Time in Air", x= "Minutes Saved", y="Count")
```



#We can see that the majority of the flights did not save any minutes on the arrival delay and actually ended up being delayed more. Some flights did in fact save minutes but it was less than 50% of all flights.

#Delays by Month

```
not_canceled|>
  group_by(month)|>
  summarize(mean_dep_delay=mean(dep_delay, na.rm=TRUE))|>
  ggplot(aes(x=month, y=mean_dep_delay))+
  geom_line()+
  geom_point()+
  labs(title = "Monthly Departure Delays", x="Month", y="Time (minutes)")
```

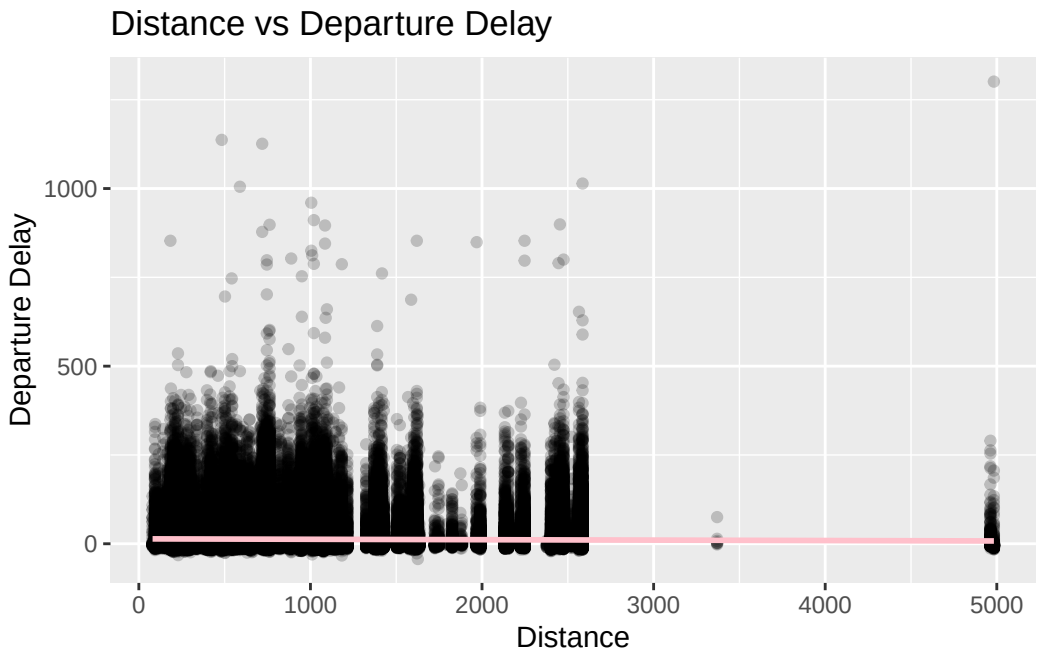


#we see that the majority of flights are delayed from May to mid July, and there is another peak at December. The months with the shorest delays are September and October.

#Does distance affect the amount of delays?

```
ggplot(not_canceled, aes(x=distance, y=dep_delay))+
  geom_point(alpha=0.2)+
  geom_smooth(method = "lm", se=TRUE,color= "Pink")+
  labs(title = "Distance vs Departure Delay", x="Distance", y="Departure Delay")
```

```
`geom_smooth()` using formula = 'y ~ x'
```



#We can see that there is not much of an effect of Distance on Departure delay.

#Flights traveled the longest by distance

```
longest_Distance <- not_canceled|>
arrange(desc(distance))|>
select(carrier, origin, dest)
longest_Distance
```

```
# A tibble: 327,346 x 3
   carrier origin dest
   <chr>   <chr>  <chr>
 1 HA      JFK    HNL
 2 HA      JFK    HNL
 3 HA      JFK    HNL
 4 HA      JFK    HNL
 5 HA      JFK    HNL
 6 HA      JFK    HNL
 7 HA      JFK    HNL
 8 HA      JFK    HNL
 9 HA      JFK    HNL
10 HA      JFK    HNL
# i 327,336 more rows
```

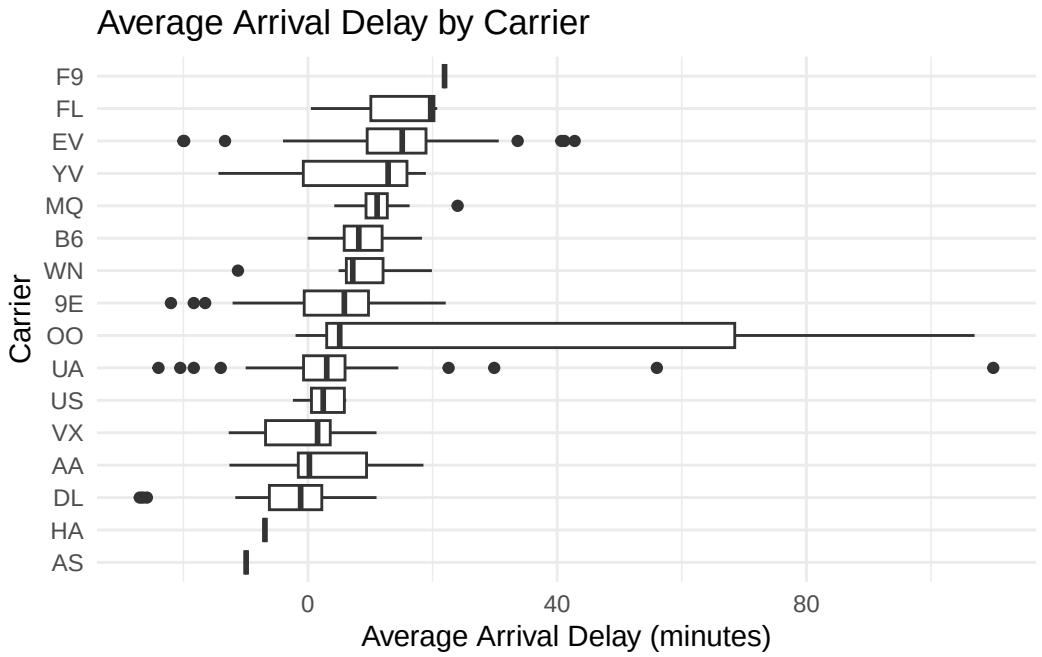
#We see that HA is the carrier with the longest flights and they all start at JFK airport and land at HNL.

#arrival delays per carrier

```
carrier_Dest<-not_canceled |>
  group_by(carrier, dest) |>
  summarize(avg_arr_Delay = mean(arr_delay, na.rm= TRUE), .group= "drop")
```

```
`summarise()` has grouped output by 'carrier'. You can override using the
`.groups` argument.
```

```
  ggplot(carrier_Dest, aes(x = reorder(carrier, avg_arr_Delay, median), y= avg_arr_Delay))+
  geom_boxplot()+
  coord_flip()+
  labs(title = "Average Arrival Delay by Carrier",x = "Carrier", y= "Average Arrival Delay (minutes)"
  theme_minimal()
```



Analysis Approach Plan:

Assumptions: All variables are independent

The process of analysis will involve data cleaning after forming our question, basic exploration of the data, comparison of certain datasets with other datasets, visualization of the data, and an interpretation of the data/results. Cleaning of the data will deal with tasks like handling empty cells/columns and NA values. When it comes to exploratory data analysis, we plan on using tools such as histograms and boxplots to gain an understanding of the data and identify patterns and relationships. The statistical

analysis that we plan on performing with the data will most likely involve making comparisons between groups to compare airlines, times, and other metrics to make our overall claim. For example, we might be comparing trends in time performance by weeks or month between different airlines to gain a better understanding of how differences in airlines affect delays. In terms of data visualization, we will most likely be using line graphs for trends over time when it comes to comparing flight time under different variables and heatmaps/scatterplots for flight delays to help communicate our findings. Finally, interpretation of the data will involve us answering the proposed question by summarizing our statistics/findings as well as through the presentation of graphical evidence.

Analysis:

Question 4: Does the age of the plane affect flight delays?

- Within this main question we will perform hypothesis tests to answer the two following sub-questions:

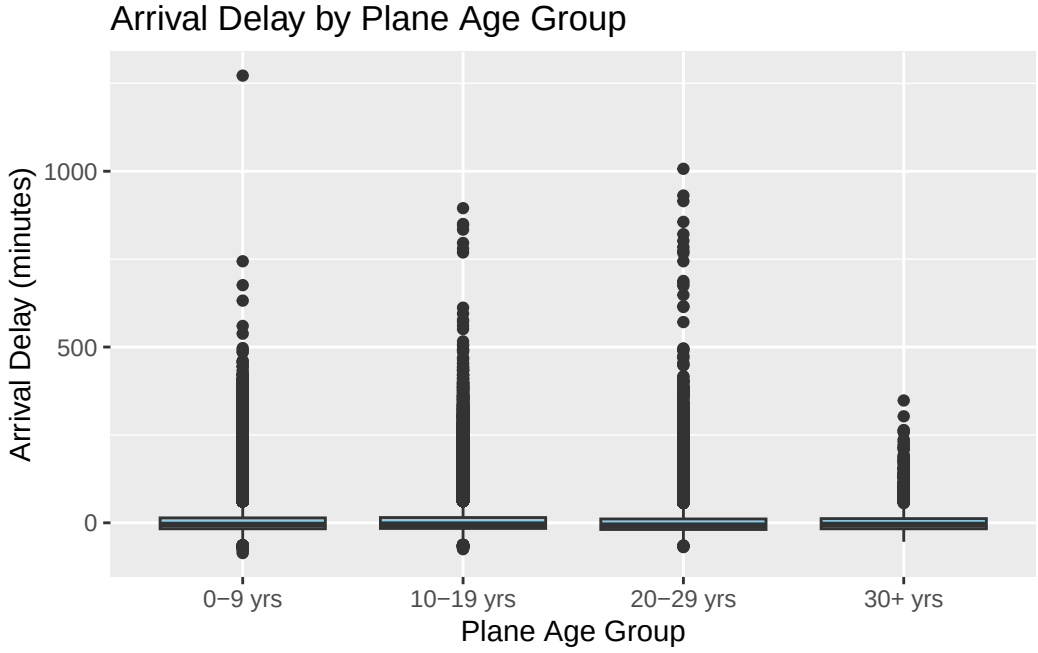
1. Do older planes experience more delays compared to newer ones?

```r
# Join flights with planes to get plane manufacture year
planes_fixed <- planes %>%
  rename(plane_year = year)

flights_planes <- flights %>%
  inner_join(planes %>% rename(plane_year = year), by = "tailnum") %>%
  filter(!is.na(plane_year), !is.na(arr_delay)) %>%
  mutate(
    plane_age = 2013 - plane_year,
    age_group = case_when(
      plane_age < 10 ~ "0-9 yrs",
      plane_age < 20 ~ "10-19 yrs",
      plane_age < 30 ~ "20-29 yrs",
      TRUE ~ "30+ yrs"
    )
  )

head(flights_planes)
```

```
# A tibble: 6 x 29
   year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
  <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
1  2013     1     1      517            515         2      830            819
2  2013     1     1      533            529         4      850            830
3  2013     1     1      542            540         2      923            850
4  2013     1     1      544            545        -1     1004           1022
5  2013     1     1      554            600        -6      812            837
6  2013     1     1      554            558        -4      740            728
# i 21 more variables: arr_delay <dbl>, carrier <chr>, flight <int>,
```

```
#   tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>,
#   hour <dbl>, minute <dbl>, time_hour <dttm>, plane_year <int>, type <chr>,
#   manufacturer <chr>, model <chr>, engines <int>, seats <int>, speed <int>,
#   engine <chr>, plane_age <dbl>, age_group <chr>
```

```r
ggplot(flights_planes, aes(x = age_group, y = arr_delay)) +
  geom_boxplot(fill = "skyblue") +
  labs(
    title = "Arrival Delay by Plane Age Group",
    x = "Plane Age Group",
    y = "Arrival Delay (minutes)"
  )
```




```r
#install.packages("dunn.test")

# Ensure age_group is factor
flights_planes$age_group <- as.factor(flights_planes$age_group)

# Summary stats
summary_stats <- flights_planes %>%
  group_by(age_group) %>%
  summarise(
    mean_delay = mean(arr_delay, na.rm = TRUE),
    count = n()
  )
print(summary_stats)
```

```
# A tibble: 4 x 3
  age_group mean_delay  count
  <fct>          <dbl>  <int>
1 0-9 yrs         7.36 103366
2 10-19 yrs       7.61 133479
3 20-29 yrs       4.00  35412
4 30+ yrs         5.54   1596
```

```r
# Normality test with sample per group
set.seed(123)
normality_test <- flights_planes %>%
  group_by(age_group) %>%
  summarise(
    sample_delays = list(sample(arr_delay[!is.na(arr_delay)], min(5000, n()), replace = FALSE)),
    shapiro_p = shapiro.test(unlist(sample_delays))$p.value
  ) %>%
  select(-sample_delays)
print(normality_test)
```

```
# A tibble: 4 x 2
  age_group shapiro_p
  <fct>         <dbl>
1 0-9 yrs    5.81e-68
2 10-19 yrs  1.49e-69
3 20-29 yrs  4.87e-75
4 30+ yrs    2.11e-45
```

```r
# Levene's Test for homogeneity of variances
levene_result <- leveneTest(arr_delay ~ age_group, data = flights_planes)
print(levene_result)
```

```
Levene's Test for Homogeneity of Variance (center = median)
          Df F value  Pr(>F)
group      3  3.4683 0.01542 *
      273849
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Since homogeneity is violated, run Kruskal-Wallis test (non-parametric)
kruskal_result <- kruskal.test(arr_delay ~ age_group, data = flights_planes)
print(kruskal_result)
```

```

    Kruskal-Wallis rank sum test

data:  arr_delay by age_group
Kruskal-Wallis chi-squared = 514.57, df = 3, p-value < 2.2e-16
```

```r
# post-hoc test for Kruskal-Wallis (Dunn test) if significant
dunn.test(flights_planes$arr_delay, flights_planes$age_group, method = "bonferroni")
```

```
  Kruskal-Wallis rank sum test

data: x and group
Kruskal-Wallis chi-squared = 514.566, df = 3, p-value = 0

                           Comparison of x by group
                                 (Bonferroni)
Col Mean-|
Row Mean |    0-9 yrs   10-19 yr   20-29 yr
---------+---------------------------------
10-19 yr |  -2.397838
         |     0.0495
         |
20-29 yr |   19.95894   22.22150
         |    0.0000*    0.0000*
         |
 30+ yrs |   0.872657   1.268699  -3.942415
         |     1.0000     0.6136    0.0002*

alpha = 0.05
Reject Ho if p <= alpha/2
```

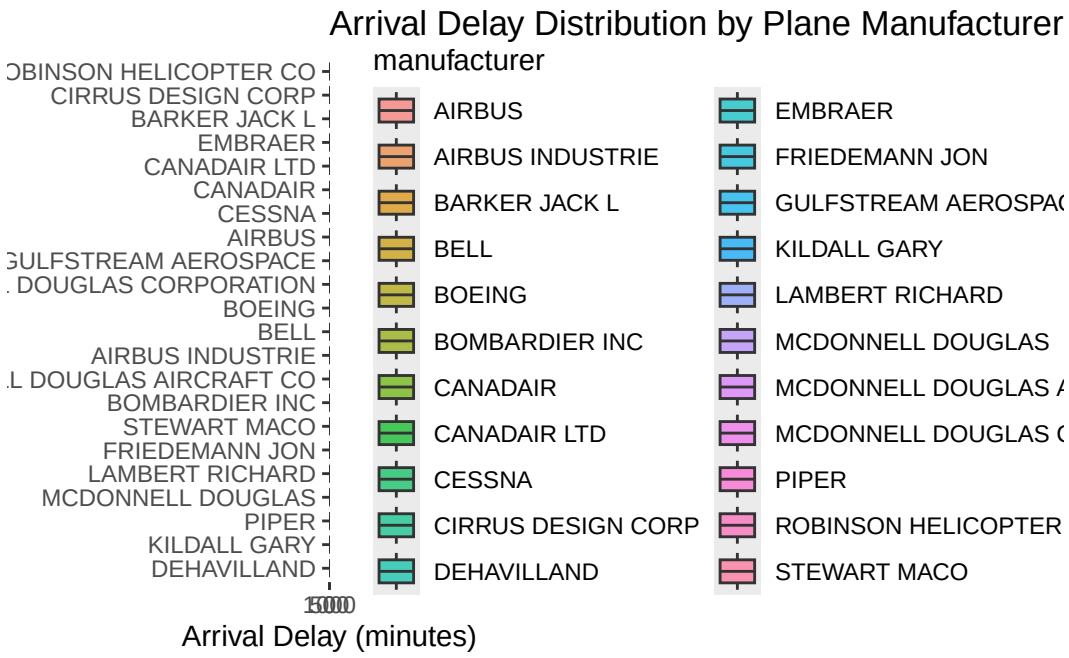
The Kruskal-Wallis test revealed a highly significant difference in arrival delays across plane age groups ($p < 2.2e\text{-}16$), indicating that at least one group's delay distribution differs from the others. Post-hoc pairwise comparisons using Dunn's test with Bonferroni correction showed that planes aged 20–29 years experience significantly different delay patterns compared to both the 0–9 and 10–19 year groups (adjusted p-values < 0.001). Additionally, planes aged 30+ years differ significantly from the 20–29 year group (adjusted $p = 0.0002$), but do not differ significantly from the younger 0–9 or 10–19 year groups. The difference between the 10–19 and 0–9 year groups was borderline significant (adjusted $p = 0.0495$). In summary, planes aged 20–29 years tend to have notably different arrival delays compared to most other age groups, highlighting a possible link between this age range and on-time performance issues.

2. Are there specific plane models or manufactures associated with better on-time performance?

```r
flights_manufacturer <- flights %>%
  inner_join(planes, by = "tailnum") %>%
  filter(!is.na(manufacturer), !is.na(arr_delay)) %>%
  group_by(manufacturer) %>%
  filter(n() > 50)
```

```r
ggplot(flights_manufacturer, aes(x = reorder(manufacturer, arr_delay, FUN = median), y = arr_delay, f
  geom_boxplot(outlier.size = 0.5, alpha = 0.7) +
  coord_flip() +   # horizontal for readability
```

```
  labs(
    title = "Arrival Delay Distribution by Plane Manufacturer",
    x = "Manufacturer",
    y = "Arrival Delay (minutes)"
  )
```




```
# filter manufacturers with more than 50 flights
flights_manufacturer <- flights %>%
  inner_join(planes, by = "tailnum") %>%
  filter(!is.na(manufacturer), !is.na(arr_delay)) %>%
  group_by(manufacturer) %>%
  filter(n() > 50) %>%
  ungroup()

# Convert manufacturer to factor to avoid warnings in tests
flights_manufacturer$manufacturer <- as.factor(flights_manufacturer$manufacturer)

# Normality test per manufacturer group
set.seed(123)
normality_test <- flights_manufacturer %>%
  group_by(manufacturer) %>%
  summarise(
    sample_delays = list(sample(arr_delay, min(5000, n()), replace = FALSE)),
    shapiro_p = shapiro.test(unlist(sample_delays))$p.value
  ) %>%
```

```
  select(-sample_delays)
print(normality_test)
```

```
# A tibble: 22 x 2
   manufacturer       shapiro_p
   <fct>                  <dbl>
 1 AIRBUS              1.03e-66
 2 AIRBUS INDUSTRIE    1.17e-71
 3 BARKER JACK L       1.20e-15
 4 BELL                2.94e-11
 5 BOEING              4.68e-67
 6 BOMBARDIER INC      8.57e-69
 7 CANADAIR            2.56e-47
 8 CANADAIR LTD        1.16e-11
 9 CESSNA              1.67e-30
10 CIRRUS DESIGN CORP  1.70e-22
# i 12 more rows
```

```
# Levene's Test for homogeneity of variance
leveneTest(arr_delay ~ manufacturer, data = flights_manufacturer)
```

```
Levene's Test for Homogeneity of Variance (center = median)
          Df F value    Pr(>F)
group     21   43.17 < 2.2e-16 ***
      278672
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Kruskal-Wallis test
kruskal_result <- kruskal.test(arr_delay ~ manufacturer, data = flights_manufacturer)
cat("\nKruskal-Wallis test result:\n")
```

```
Kruskal-Wallis test result:
```

```
print(kruskal_result)
```

```
    Kruskal-Wallis rank sum test

data:  arr_delay by manufacturer
Kruskal-Wallis chi-squared = 3770.9, df = 21, p-value < 2.2e-16
```

```
# If Kruskal-Wallis is significant, do Dunn's test for pairwise comparisons
dunn_result <- dunn.test(flights_manufacturer$arr_delay, flights_manufacturer$manufacturer, method =
```

```
  Kruskal-Wallis rank sum test

data: x and group
Kruskal-Wallis chi-squared = 3770.9154, df = 21, p-value = 0
```

Comparison of x by group
(Bonferroni)

Col Mean- / Row Mean	AIRBUS	AIRBUS I	BARKER J	BELL	BOEING	BOMBARDI
AIRBUS I	9.931131					
	0.0000*					
BARKER J	-2.525521	-3.585438				
	1.0000	0.0389				
BELL	1.203695	0.669759	2.212547			
	1.0000	1.0000	1.0000			
BOEING	12.65471	0.975794	3.684397	-0.623082		
	0.0000*	1.0000	0.0265	1.0000		
BOMBARDI	11.68778	2.756355	3.919185	-0.498747	2.240740	
	0.0000*	0.6751	0.0103*	1.0000	1.0000	
CANADAIR	-6.410582	-8.948789	-0.123357	-2.490554	-9.256539	-9.683654
	0.0000*	0.0000*	1.0000	1.0000	0.0000*	0.0000*
CANADAIR	-1.781623	-2.429688	-0.203980	-2.063333	-2.488113	-2.635713
	1.0000	1.0000	1.0000	1.0000	1.0000	0.9697
CESSNA	-1.650045	-3.315613	1.250314	-1.651213	-3.475462	-3.835224
	1.0000	0.1056	1.0000	1.0000	0.0589	0.0145*
CIRRUS D	-2.501401	-3.640381	0.145408	-2.152896	-3.747300	-3.998451
	1.0000	0.0314	1.0000	1.0000	0.0206*	0.0074*
DEHAVILL	3.309480	2.783866	4.090625	1.515670	2.738733	2.614826
	0.1080	0.6204	0.0050*	1.0000	0.7124	1.0000
EMBRAER	-31.90230	-41.17496	-0.537743	-2.744796	-50.63987	-39.18125
	0.0000*	0.0000*	1.0000	0.6993	0.0000*	0.0000*
FRIEDEMA	0.149903	-0.353436	1.218851	-0.717146	-0.397816	-0.514335
	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
GULFSTRE	-1.424144	-2.879936	1.207840	-1.621899	-3.016807	-3.336904
	1.0000	0.4594	1.0000	1.0000	0.2950	0.0978

KILDALL	2.757056	2.276466	3.556354	1.245048	2.234952	2.122105
	0.6736	1.0000	0.0434	1.0000	1.0000	1.0000
LAMBERT	0.991503	0.501741	1.960196	-0.083066	0.458850	0.344931
	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
MCDONNEL	17.38324	13.29676	6.924364	1.102706	13.23871	11.77878
	0.0000*	0.0000*	0.0000*	1.0000	0.0000*	0.0000*
MCDONNEL	8.613098	2.774816	4.054407	-0.410345	2.380882	0.901520
	0.0000*	0.6379	0.0058*	1.0000	1.0000	1.0000
MCDONNEL	2.534726	0.188565	3.360667	-0.611980	-0.017347	-0.556644
	1.0000	1.0000	0.0898	1.0000	1.0000	1.0000
PIPER	4.547819	3.713007	5.146813	1.439755	3.643210	3.443095
	0.0006*	0.0237*	0.0000*	1.0000	0.0311	0.0664
ROBINSON	-2.003523	-3.126982	0.463251	-1.948603	-3.231432	-3.481042
	1.0000	0.2040	1.0000	1.0000	0.1423	0.0577
STEWART	1.799910	1.310077	2.694919	0.513029	1.267453	1.153014
	1.0000	1.0000	0.8132	1.0000	1.0000	1.0000

Col Mean- Row Mean	CANADAIR	CANADAIR	CESSNA	CIRRUS D	DEHAVILL	EMBRAER
CANADAIR	-0.152935					
	1.0000					
CESSNA	2.139120	1.064927				
	1.0000	1.0000				
CIRRUS D	0.326588	0.313165	-1.138783			
	1.0000	1.0000	1.0000			
DEHAVILL	4.536502	3.695902	3.656535	4.056727		
	0.0007*	0.0253	0.0295	0.0057*		
EMBRAER	-0.979205	-0.089631	-3.171492	-0.791015	-4.826300	
	1.0000	1.0000	0.1752	1.0000	0.0002*	
FRIEDEMA	1.385788	1.211819	0.620677	1.148166	-2.182791	1.602771
	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
GULFSTRE	1.950500	1.050907	0.013166	1.096731	-3.600436	2.786985
	1.0000	1.0000	1.0000	1.0000	0.0367	0.6145
KILDALL	3.896446	3.278208	3.109032	3.515032	-0.198882	4.143972
	0.0113*	0.1207	0.2168	0.0508	1.0000	0.0039*

LAMBERT	2.180970	1.866283	1.417459	1.899699	-1.532327	2.405077
	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
MCDONNEL	15.09122	4.542870	8.296791	7.203749	-1.024365	29.24464
	0.0000*	0.0006*	0.0000*	0.0000*	1.0000	0.0000*
MCDONNEL	9.591538	2.732172	4.014678	4.139773	-2.522928	25.90790
	0.0000*	0.7267	0.0069*	0.0040*	1.0000	0.0000*
MCDONNEL	6.289239	2.396971	2.838717	3.374760	-2.678874	9.335853
	0.0000*	1.0000	0.5232	0.0853	0.8532	0.0000*
PIPER	6.328748	4.203350	4.818408	5.158643	-0.376350	6.957186
	0.0000*	0.0030*	0.0002*	0.0000*	1.0000	0.0000*
ROBINSON	0.748966	0.543767	-0.742171	0.330825	-3.850482	1.243623
	1.0000	1.0000	1.0000	1.0000	0.0136*	1.0000
STEWART	2.975839	2.511849	2.193966	2.643250	-0.940646	3.213603
	0.3375	1.0000	1.0000	0.9484	1.0000	0.1514

Col Mean- Row Mean	FRIEDEMA	GULFSTRE	KILDALL	LAMBERT	MCDONNEL	MCDONNEL
GULFSTRE	-0.607105					
	1.0000					
KILDALL	1.892094	3.067257				
	1.0000	0.2495				
LAMBERT	0.606492	1.394767	-1.274542			
	1.0000	1.0000	1.0000			
MCDONNEL	2.019089	7.344407	-0.670132	1.124655		
	1.0000	0.0000*	1.0000	1.0000		
MCDONNEL	0.595487	3.512714	-2.039462	-0.264204	-9.930511	
	1.0000	0.0512	1.0000	1.0000	0.0000*	
MCDONNEL	0.385683	2.566185	-2.194748	-0.453073	-6.721640	-0.897308
	1.0000	1.0000	1.0000	1.0000	0.0000*	1.0000
PIPER	2.223891	4.666325	-0.118841	1.449297	0.915747	3.287409
	1.0000	0.0004*	1.0000	1.0000	1.0000	0.1168
ROBINSON	-0.956128	-0.719028	-3.325453	-1.710445	-6.661752	-3.626467
	1.0000	1.0000	0.1020	1.0000	0.0000*	0.0332
STEWART	1.186262	2.161880	-0.708115	0.571952	-0.321318	1.070653

```
         |     1.0000     1.0000     1.0000     1.0000     1.0000     1.0000
Col Mean-|
Row Mean |   MCDONNEL      PIPER   ROBINSON
---------+---------------------------------
   PIPER |   3.446159
         |     0.0657
         |
ROBINSON |  -2.916703  -4.856862
         |     0.4086    0.0001*
         |
STEWART  |   1.245277  -0.751627   2.452362
         |     1.0000     1.0000     1.0000

alpha = 0.05
Reject Ho if p <= alpha/2
```

```r
#Instead of printing all pairwise comparisons (which is long), filter and print only significant comp
if (kruskal_result$p.value < 0.05) {
  sig_comparisons <- data.frame(
    comparison = dunn_result$comparisons,
    p_value = dunn_result$P.adjusted
  ) %>% filter(p_value < 0.05)

  cat("\nSignificant pairwise differences between manufacturers (Dunn's test):\n")
  print(sig_comparisons)
} else {
  cat("\nNo significant differences between manufacturers detected by Kruskal-Wallis test.\n")
}
```

```
Significant pairwise differences between manufacturers (Dunn's test):
                                          comparison       p_value
1                            AIRBUS - AIRBUS INDUSTRIE  3.520163e-21
2                      AIRBUS INDUSTRIE - BARKER JACK L  3.886724e-02
3                                  AIRBUS - BOEING  1.216946e-34
4                             BARKER JACK L - BOEING  2.647770e-02
5                           AIRBUS - BOMBARDIER INC  1.699897e-29
6                    BARKER JACK L - BOMBARDIER INC  1.026202e-02
7                                 AIRBUS - CANADAIR  1.674346e-08
8                       AIRBUS INDUSTRIE - CANADAIR  4.151109e-17
9                                 BOEING - CANADAIR  2.438948e-18
10                          BOMBARDIER INC - CANADAIR  4.086806e-20
11                            BOMBARDIER INC - CESSNA  1.448942e-02
12           AIRBUS INDUSTRIE - CIRRUS DESIGN CORP  3.144302e-02
13                     BOEING - CIRRUS DESIGN CORP  2.064534e-02
14               BOMBARDIER INC - CIRRUS DESIGN CORP  7.364085e-03
15                     BARKER JACK L - DEHAVILLAND  4.968936e-03
16                          CANADAIR - DEHAVILLAND  6.605991e-04
```

```
17                              CANADAIR LTD - DEHAVILLAND  2.530688e-02
18                                   CESSNA - DEHAVILLAND  2.952718e-02
19                         CIRRUS DESIGN CORP - DEHAVILLAND  5.747859e-03
20                                       AIRBUS - EMBRAER 2.865086e-221
21                               AIRBUS INDUSTRIE - EMBRAER  0.000000e+00
22                                       BOEING - EMBRAER  0.000000e+00
23                                 BOMBARDIER INC - EMBRAER  0.000000e+00
24                                    DEHAVILLAND - EMBRAER  1.606518e-04
25                       DEHAVILLAND - GULFSTREAM AEROSPACE  3.669247e-02
26                             BARKER JACK L - KILDALL GARY  4.343220e-02
27                                  CANADAIR - KILDALL GARY  1.127444e-02
28                                   EMBRAER - KILDALL GARY  3.942482e-03
29                                AIRBUS - MCDONNELL DOUGLAS  1.276418e-65
30                      AIRBUS INDUSTRIE - MCDONNELL DOUGLAS  2.791820e-38
31                         BARKER JACK L - MCDONNELL DOUGLAS  5.058163e-10
32                                 BOEING - MCDONNELL DOUGLAS  6.056356e-38
33                       BOMBARDIER INC - MCDONNELL DOUGLAS  5.799314e-30
34                            CANADAIR - MCDONNELL DOUGLAS  2.136569e-49
35                          CANADAIR LTD - MCDONNELL DOUGLAS  6.409492e-04
36                               CESSNA - MCDONNELL DOUGLAS  1.235401e-14
37                    CIRRUS DESIGN CORP - MCDONNELL DOUGLAS  6.765853e-11
38                               EMBRAER - MCDONNELL DOUGLAS 6.067009e-186
39                  GULFSTREAM AEROSPACE - MCDONNELL DOUGLAS  2.387074e-11
40                    AIRBUS - MCDONNELL DOUGLAS AIRCRAFT CO  8.213450e-16
41            BARKER JACK L - MCDONNELL DOUGLAS AIRCRAFT CO  5.805203e-03
42                 CANADAIR - MCDONNELL DOUGLAS AIRCRAFT CO  1.002327e-19
43                   CESSNA - MCDONNELL DOUGLAS AIRCRAFT CO  6.875356e-03
44       CIRRUS DESIGN CORP - MCDONNELL DOUGLAS AIRCRAFT CO  4.015349e-03
45                  EMBRAER - MCDONNELL DOUGLAS AIRCRAFT CO 6.266295e-146
46        MCDONNELL DOUGLAS - MCDONNELL DOUGLAS AIRCRAFT CO  3.542126e-21
47                  CANADAIR - MCDONNELL DOUGLAS CORPORATION  3.684733e-08
48                   EMBRAER - MCDONNELL DOUGLAS CORPORATION  1.157090e-18
49        MCDONNELL DOUGLAS - MCDONNELL DOUGLAS CORPORATION  2.075422e-09
50                                         AIRBUS - PIPER  6.260617e-04
51                                 AIRBUS INDUSTRIE - PIPER  2.365569e-02
52                                    BARKER JACK L - PIPER  3.060154e-05
53                                          BOEING - PIPER  3.109937e-02
54                                       CANADAIR - PIPER  2.854665e-08
55                                    CANADAIR LTD - PIPER  3.037573e-03
56                                         CESSNA - PIPER  1.671373e-04
57                                CIRRUS DESIGN CORP - PIPER  2.873011e-05
58                                         EMBRAER - PIPER  4.009423e-10
59                               GULFSTREAM AEROSPACE - PIPER  3.541626e-04
60                      DEHAVILLAND - ROBINSON HELICOPTER CO  1.361576e-02
61                MCDONNELL DOUGLAS - ROBINSON HELICOPTER CO  3.125217e-09
62 MCDONNELL DOUGLAS AIRCRAFT CO - ROBINSON HELICOPTER CO  3.318608e-02
63                           PIPER - ROBINSON HELICOPTER CO  1.377457e-04
```

The Kruskal-Wallis test indicates a highly significant difference in arrival delays across plane manu-

facturers (^{2} = 3770.9, df = 21, p < 2.2e-16). This strongly suggests that not all manufacturers have the same on-time performance. The post-hoc Dunn's test with Bonferroni correction highlights specific pairwise differences. Notably, Airbus aircraft exhibit significantly lower arrival delays compared to Boeing, Bombardier, and Canadair. Additionally, some other manufacturers like Barker Jack L also show significant differences with Boeing and Bombardier.These results indicate that certain manufacturers are associated with better punctuality.

Alternative Strategies & Back Up Plan:

As a backup idea, we are planning on seeing if there is any correlation between the amount of delays present in the different airports. Our data deals with the airports EWR, JFK, and LGA which are all different airports within New York City. Our first question is to figure out if the JFK airport has a different amount of delays compared to LGA or EWR if there is a higher amount of precipitation in the JFK area. Although all the airports are in New York, within the different areas of the city, there can be different amounts of precipitation and rainfall that occur. Our second question is to decide whether the different airports have different models of planes and if the difference affects the amounts of delays. For example if a plane is older or a different configuration, does that lead to more delays due to cleaning or maintenance? And lastly, our third question is whether the three different airports have different airlines coming in and out and if these differing airlines affect the amount of delays present on a given day. For example, if Delta services one airport and not another, does that increase or decrease the amount of total delays for an airport. These questions can be further investigated if our first set of questions are not approved or if we need more content to explore within our project. These sets of backup questions will further explore the flight data we have.