HW 7

Abe Olsson

2025-03-05

Load and Inspect the data

```
# Load dataset
data(flights)

# View first few rows
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>
```

```
# Check structure of the dataset
glimpse(flights)  # From dplyr
```

```
## 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~
```

```
#str(flights)     # From Base R
```

Data Wrangling with dplyr

Cleaning the data

```
## # A tibble: 1 x 19
##    year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>    <int>          <int>     <int>    <int>          <int>
## 1     0     0     0     8255              0      8255     8713              0
## # i 11 more variables: arr_delay <int>, carrier <int>, flight <int>,
## #   tailnum <int>, origin <int>, dest <int>, air_time <int>, distance <int>,
## #   hour <int>, minute <int>, time_hour <int>
```

```
## # A tibble: 1 x 19
##    year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##   <int> <int> <int>    <int>          <int>     <int>    <int>          <int>
## 1     0     0     0        0              0         0        0              0
## # i 11 more variables: arr_delay <int>, carrier <int>, flight <int>,
## #   tailnum <int>, origin <int>, dest <int>, air_time <int>, distance <int>,
## #   hour <int>, minute <int>, time_hour <int>
```

Adding new columns

```
# Create a new column: Total Delay (Departure + Arrival)
flights_clean1 <- flights_clean %>%
  mutate(total_delay = dep_delay + arr_delay)

# Summarize average delay by airline
flights_clean1 %>%
  group_by(carrier) %>%
  summarise(mean_delay = mean(total_delay, na.rm = TRUE)) %>%
  arrange(desc(mean_delay))
```

```
## # A tibble: 16 x 2
##    carrier mean_delay
##    <chr>        <dbl>
##  1 F9            42.1
##  2 FL            38.7
##  3 EV            35.6
##  4 YV            34.5
##  5 WN            27.3
##  6 OO            24.5
```

```
##  7 9E           23.8
##  8 B6           22.4
##  9 MQ           21.2
## 10 UA           15.6
## 11 VX           14.5
## 12 DL           10.9
## 13 AA            8.93
## 14 US            5.87
## 15 HA           -2.01
## 16 AS           -4.10
```

```
#Add the names to the df, to make it readable
flights_clean1 %>%
  left_join(airlines, by = "carrier") %>%
  group_by(carrier, name) %>%
  summarise(mean_delay = mean(total_delay, na.rm = TRUE), .groups = "drop") %>%
  arrange(desc(mean_delay))
```

```
## # A tibble: 16 x 3
##    carrier name                        mean_delay
##    <chr>   <chr>                            <dbl>
##  1 F9      Frontier Airlines Inc.           42.1
##  2 FL      AirTran Airways Corporation      38.7
##  3 EV      ExpressJet Airlines Inc.         35.6
##  4 YV      Mesa Airlines Inc.               34.5
##  5 WN      Southwest Airlines Co.           27.3
##  6 OO      SkyWest Airlines Inc.            24.5
##  7 9E      Endeavor Air Inc.                23.8
##  8 B6      JetBlue Airways                  22.4
##  9 MQ      Envoy Air                        21.2
## 10 UA      United Air Lines Inc.            15.6
## 11 VX      Virgin America                   14.5
## 12 DL      Delta Air Lines Inc.             10.9
## 13 AA      American Airlines Inc.            8.93
## 14 US      US Airways Inc.                   5.87
## 15 HA      Hawaiian Airlines Inc.           -2.01
## 16 AS      Alaska Airlines Inc.             -4.10
```

```
# Filter flights where departure delay is greater than 30 minutes and calculate total delay
flights_filtered <- flights_clean %>%
  filter(dep_delay > 30) %>%
  mutate(total_delay = dep_delay + arr_delay)

# Summarize average delay by airline
delay_summary <- flights_filtered %>%
  group_by(carrier) %>%
  summarise(mean_delay = mean(total_delay, na.rm = TRUE)) %>%
  arrange(desc(mean_delay))

# Add the airline names for readability
final_result <- delay_summary %>%
  left_join(airlines, by = "carrier") %>%
  arrange(desc(mean_delay))
```

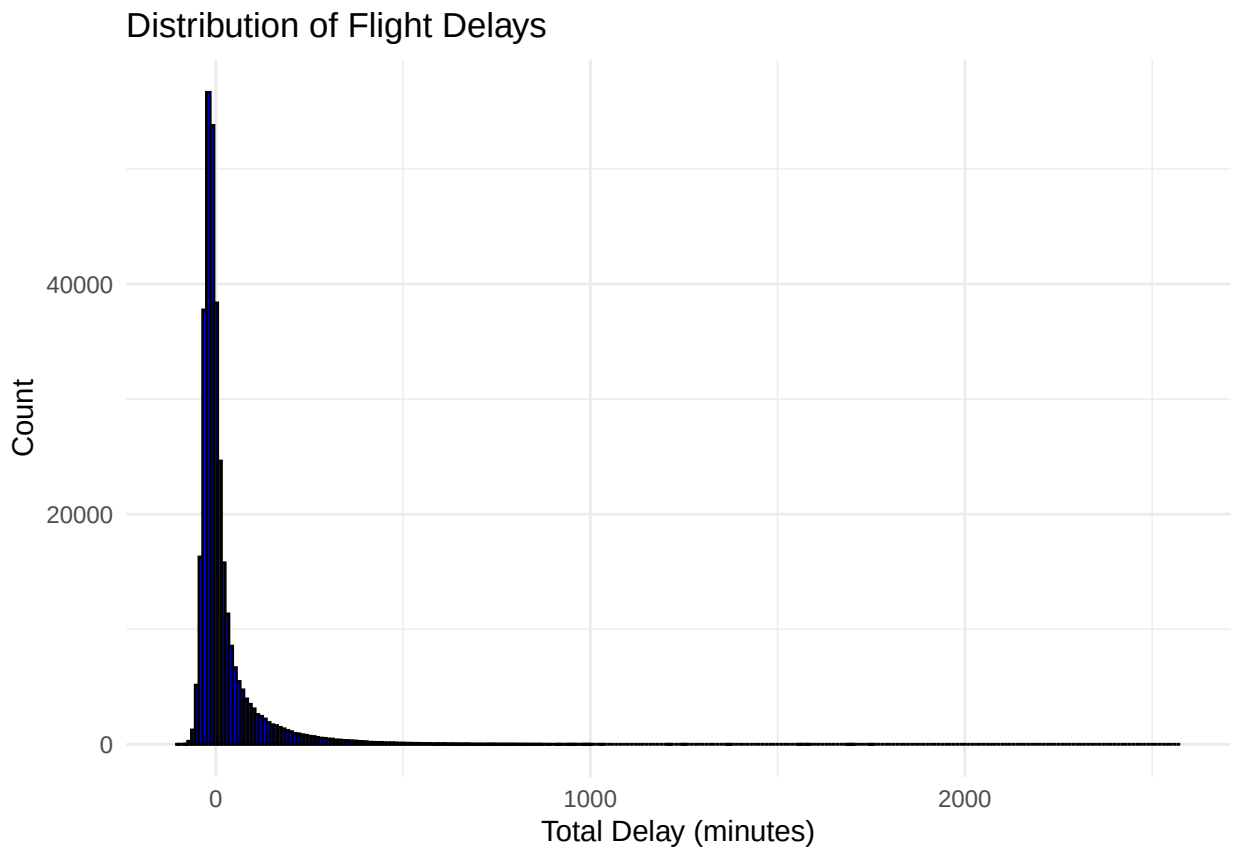
```r
# Display the final result
final_result
```

```
## # A tibble: 16 x 3
##    carrier mean_delay name
##    <chr>        <dbl> <chr>
##  1 HA            306. Hawaiian Airlines Inc.
##  2 VX            206. Virgin America
##  3 FL            205. AirTran Airways Corporation
##  4 F9            202. Frontier Airlines Inc.
##  5 OO            200. SkyWest Airlines Inc.
##  6 YV            189. Mesa Airlines Inc.
##  7 DL            180. Delta Air Lines Inc.
##  8 9E            176. Endeavor Air Inc.
##  9 WN            173. Southwest Airlines Co.
## 10 EV            173. ExpressJet Airlines Inc.
## 11 AA            172. American Airlines Inc.
## 12 B6            166. JetBlue Airways
## 13 MQ            166. Envoy Air
## 14 AS            161. Alaska Airlines Inc.
## 15 UA            157. United Air Lines Inc.
## 16 US            155. US Airways Inc.
```

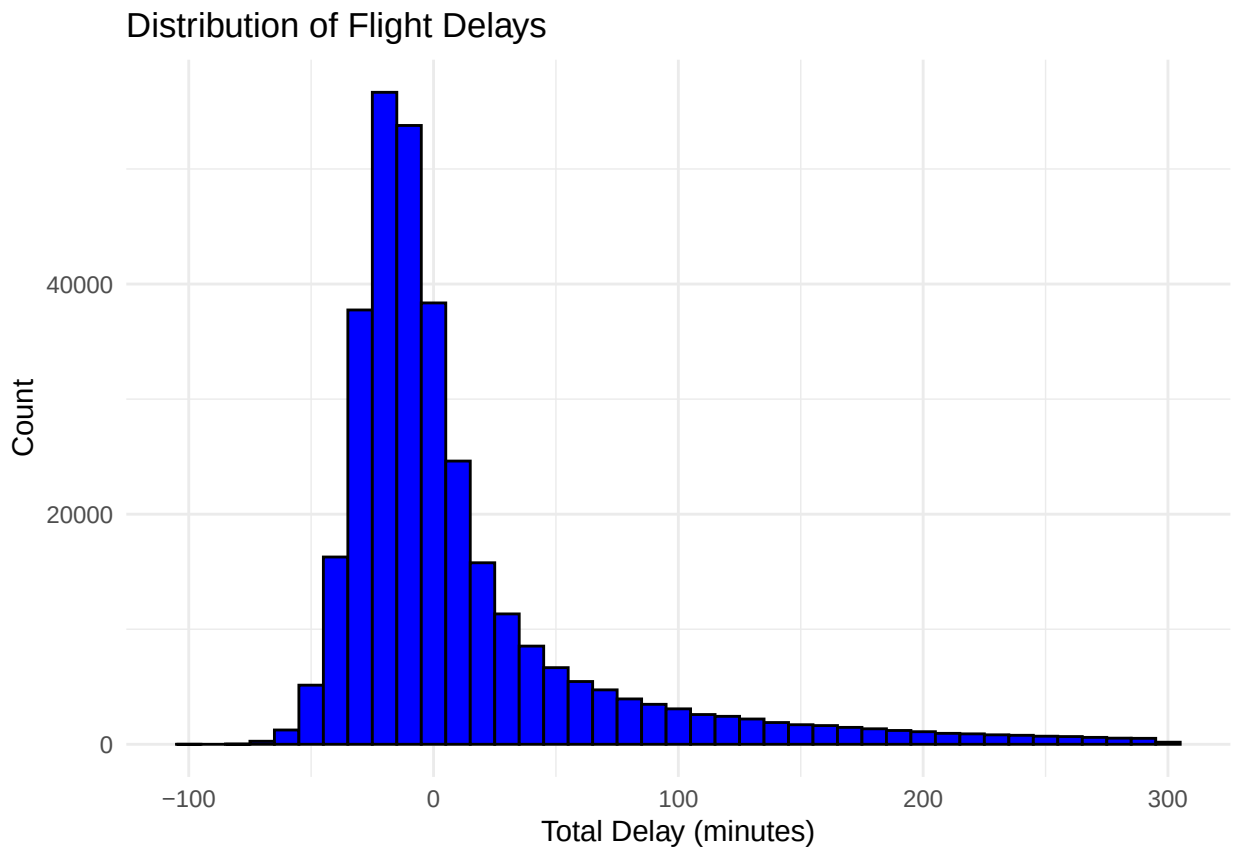
Plotting

Plot 1 - Histogram of delays

```r
ggplot(flights_clean1, aes(x = total_delay)) +
  geom_histogram(binwidth = 10, fill = "blue", color = "black") +
  labs(title = "Distribution of Flight Delays", x = "Total Delay (minutes)", y = "Count") +
  theme_minimal()
```

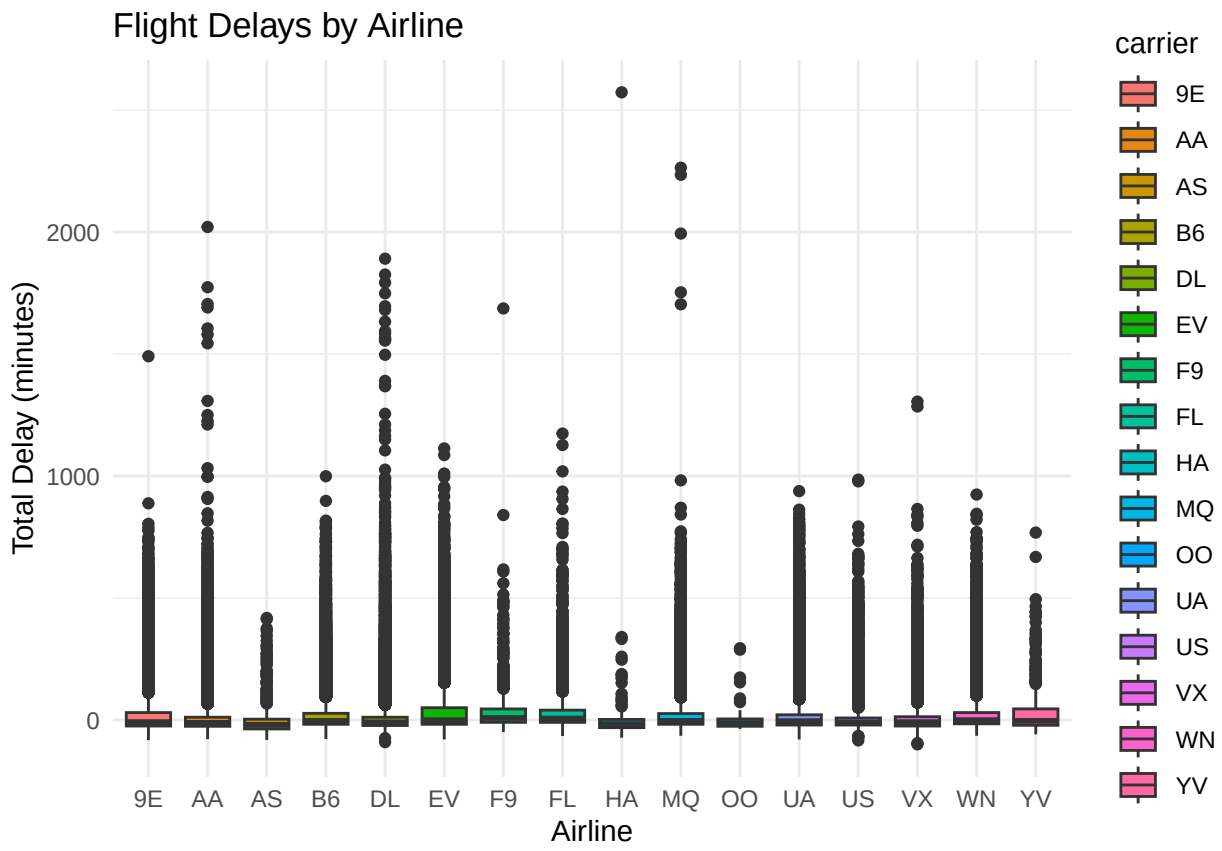



```
#let's remove the tail from the delays
flights_clean1 %>% filter(total_delay < 300 ) %>%
  ggplot( aes(x = total_delay)) +
  geom_histogram(binwidth = 10, fill = "blue", color = "black") +
  labs(title = "Distribution of Flight Delays", x = "Total Delay (minutes)", y = "Count") +
  theme_minimal()
```

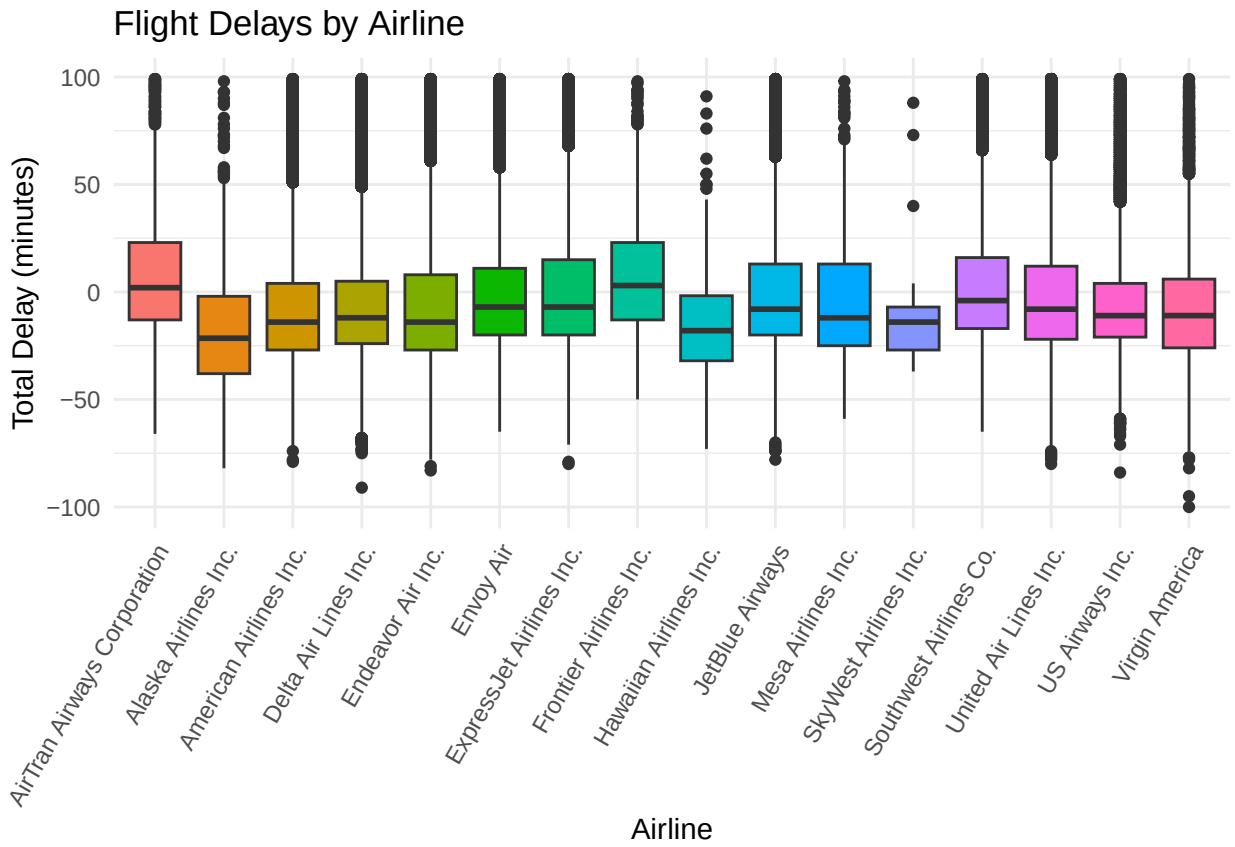


Plot 2 -

```
#Lets just make a box plot of delays
ggplot(flights_clean1, aes(x = carrier, y = total_delay, fill = carrier)) +
  geom_boxplot() +
  labs(title = "Flight Delays by Airline", x = "Airline", y = "Total Delay (minutes)") +
  theme_minimal()
```



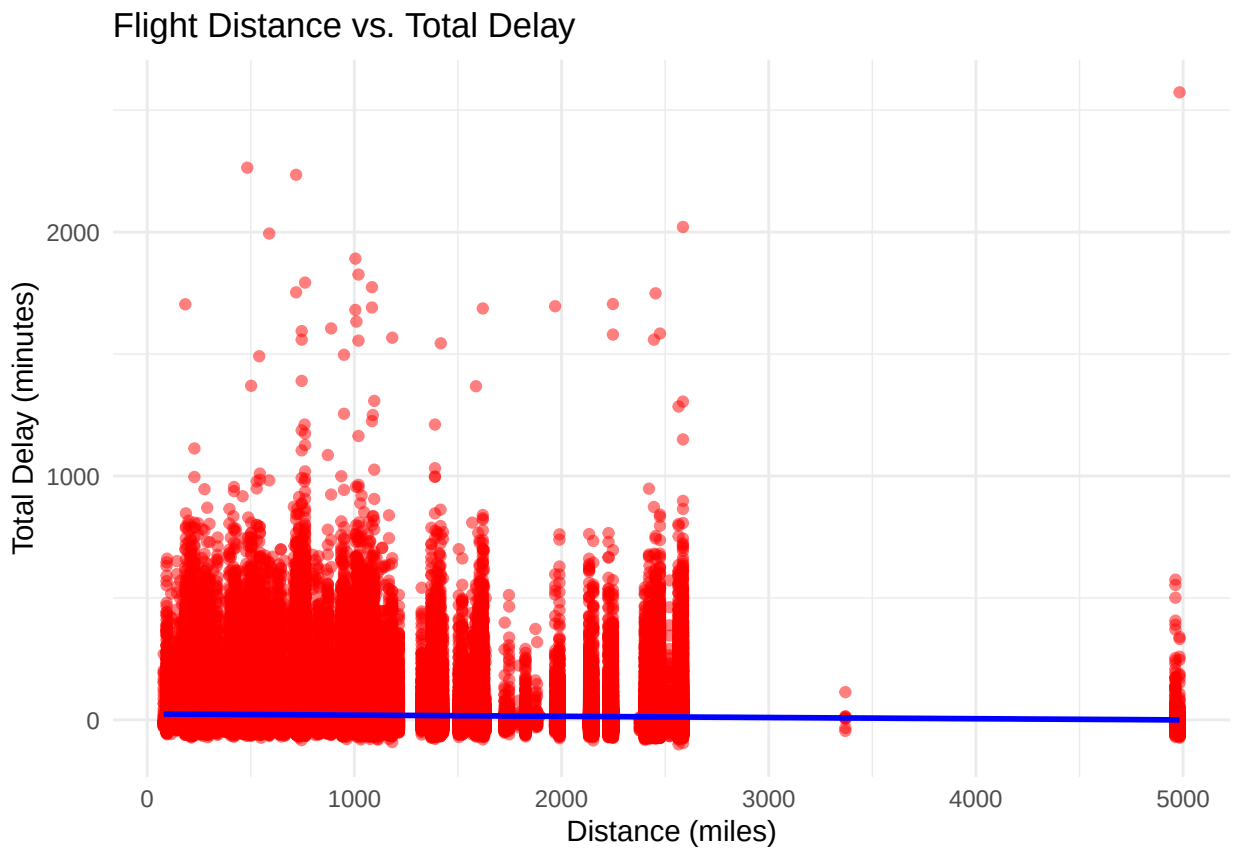

```
#Lets try to clean this up a little
flights_clean1 %>%
  filter( total_delay < 100 ) %>%
  left_join(airlines, by = "carrier") %>%
  ggplot( aes(x = name, y = total_delay, fill = name)) +
  geom_boxplot() +
  labs(title = "Flight Delays by Airline", x = "Airline", y = "Total Delay (minutes)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 60, hjust = 1), legend.position = "none")
```



Regression Example

```
ggplot(flights_clean1, aes(x = distance, y = total_delay)) +
  geom_point(alpha = 0.5, color = "red") +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "Flight Distance vs. Total Delay", x = "Distance (miles)", y = "Total Delay (minutes)") +
  theme_minimal()
```

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



This does not look very helpful, lets try to clean this up a bit

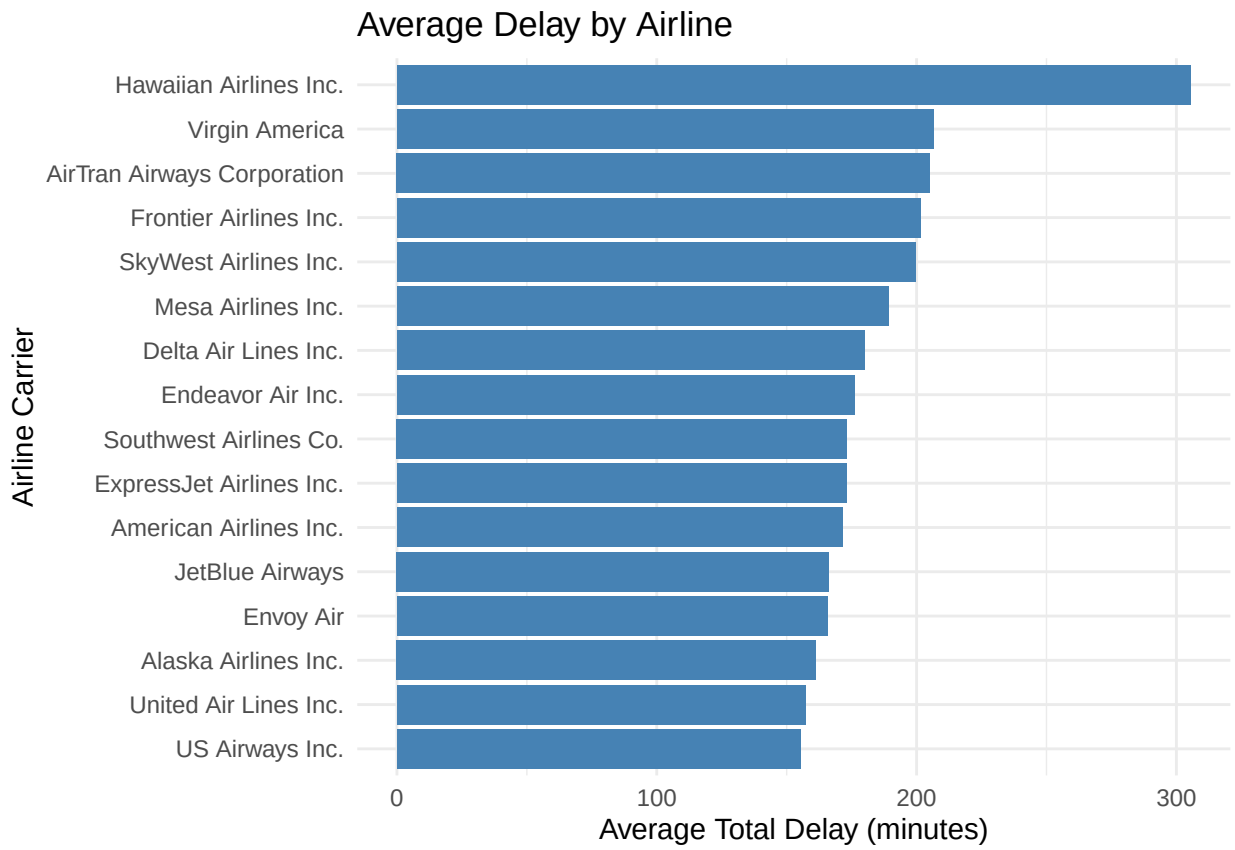
```
# Remove outliers: Filter out extreme values
flights_clean_no_outliers <- flights_clean1 %>%
  filter(distance < quantile(distance, 0.99), total_delay < quantile(total_delay, 0.99))

# Plot without outliers
distancevdelayplot <- ggplot(flights_clean_no_outliers, aes(x = distance, y = total_delay)) +
  geom_point(alpha = 0.5, color = "red") +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "Flight Distance vs. Total Delay (No Outliers)", x = "Distance (miles)", y = "Total Delay
  theme_minimal()
```

Plotting

Plot 3 - Histogram of delays

```
ggplot(final_result, aes(x = reorder(name, mean_delay), y = mean_delay)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(
    title = "Average Delay by Airline",
    x = "Airline Carrier",
    y = "Average Total Delay (minutes)"
  ) +
```

```
  theme_minimal() +
  coord_flip()
```

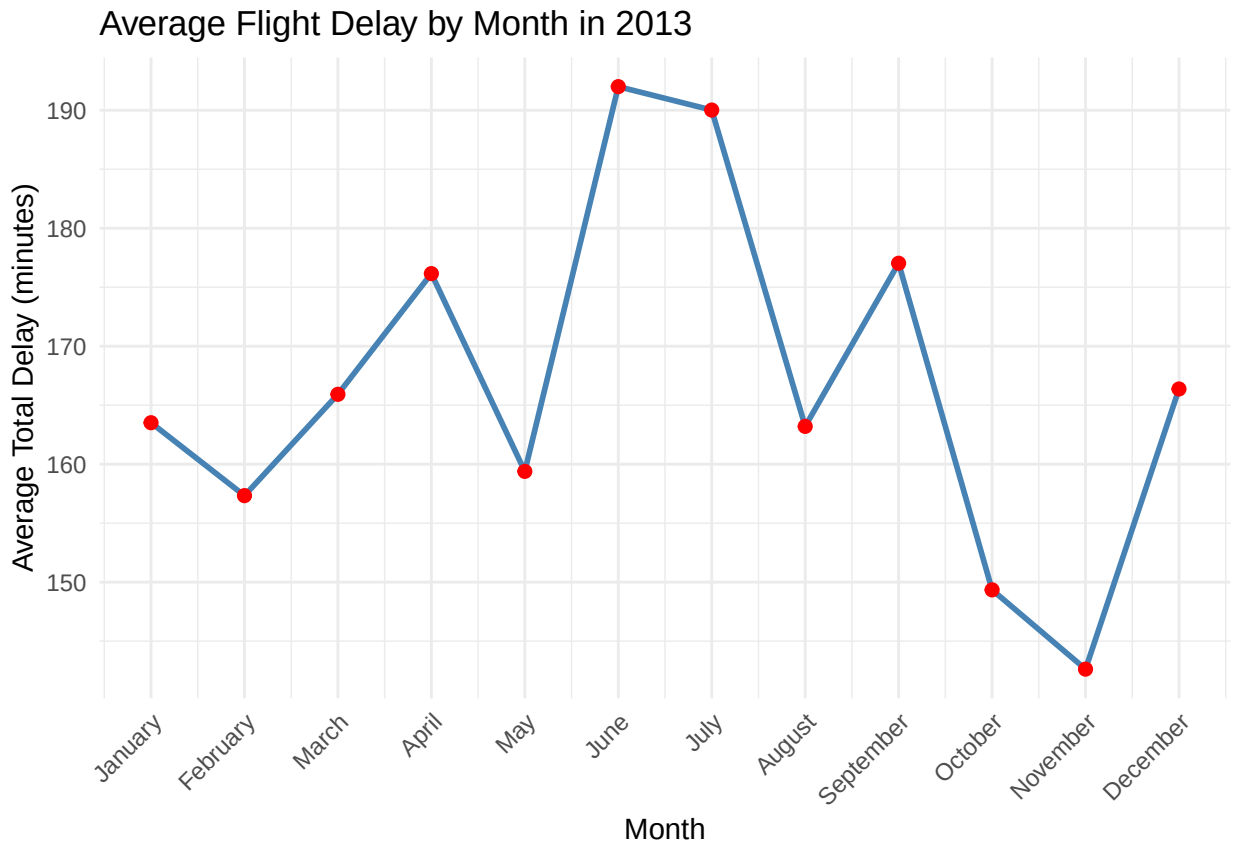


Plot 4

```
# Calculate average delay by month
monthly_delay <- flights_clean %>%
  filter(dep_delay > 30) %>%  # Only include flights with departure delay > 30 min
  mutate(total_delay = dep_delay + arr_delay) %>%
  group_by(month) %>%
  summarise(mean_delay = mean(total_delay, na.rm = TRUE), .groups = "drop") %>%
  arrange(month)

# Create a line plot
ggplot(monthly_delay, aes(x = month, y = mean_delay)) +
  geom_line(color = "steelblue", size = 1) +
  geom_point(color = "red", size = 2) +  # Add points to highlight each month
  labs(
    title = "Average Flight Delay by Month in 2013",
    x = "Month",
    y = "Average Total Delay (minutes)"
  ) +
  scale_x_continuous(breaks = 1:12, labels = month.name) +  # Convert numbers to month names
```

```
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotate x-axis labels slightly
```

```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```



Exploratory Questions

1. What is the average delay by airline?

After removing the delays less than 30 minutes, we are able to visualize the more significant delays that would impact travelers.

```
## # A tibble: 16 x 3
##    carrier mean_delay name
##    <chr>        <dbl> <chr>
##  1 HA            306. Hawaiian Airlines Inc.
##  2 VX            206. Virgin America
##  3 FL            205. AirTran Airways Corporation
##  4 F9            202. Frontier Airlines Inc.
```

```
##  5 OO           200. SkyWest Airlines Inc.
##  6 YV           189. Mesa Airlines Inc.
##  7 DL           180. Delta Air Lines Inc.
##  8 9E           176. Endeavor Air Inc.
##  9 WN           173. Southwest Airlines Co.
## 10 EV           173. ExpressJet Airlines Inc.
## 11 AA           172. American Airlines Inc.
## 12 B6           166. JetBlue Airways
## 13 MQ           166. Envoy Air
## 14 AS           161. Alaska Airlines Inc.
## 15 UA           157. United Air Lines Inc.
## 16 US           155. US Airways Inc.
```

- Which airline has the highest average delay? Does this surprise you? Why or why not?

The Airline with the highest delay is Hawaiian Airlines Inc. with an average delay of 305.5 minutes. This is not suprising because the flight travel time is very long, they are more prone to storms across the US and Pacific Ocean and any delay from another flight may impact that flight to Hawaii.

2. Is there a relationship between flight distance and delay?

Yes.

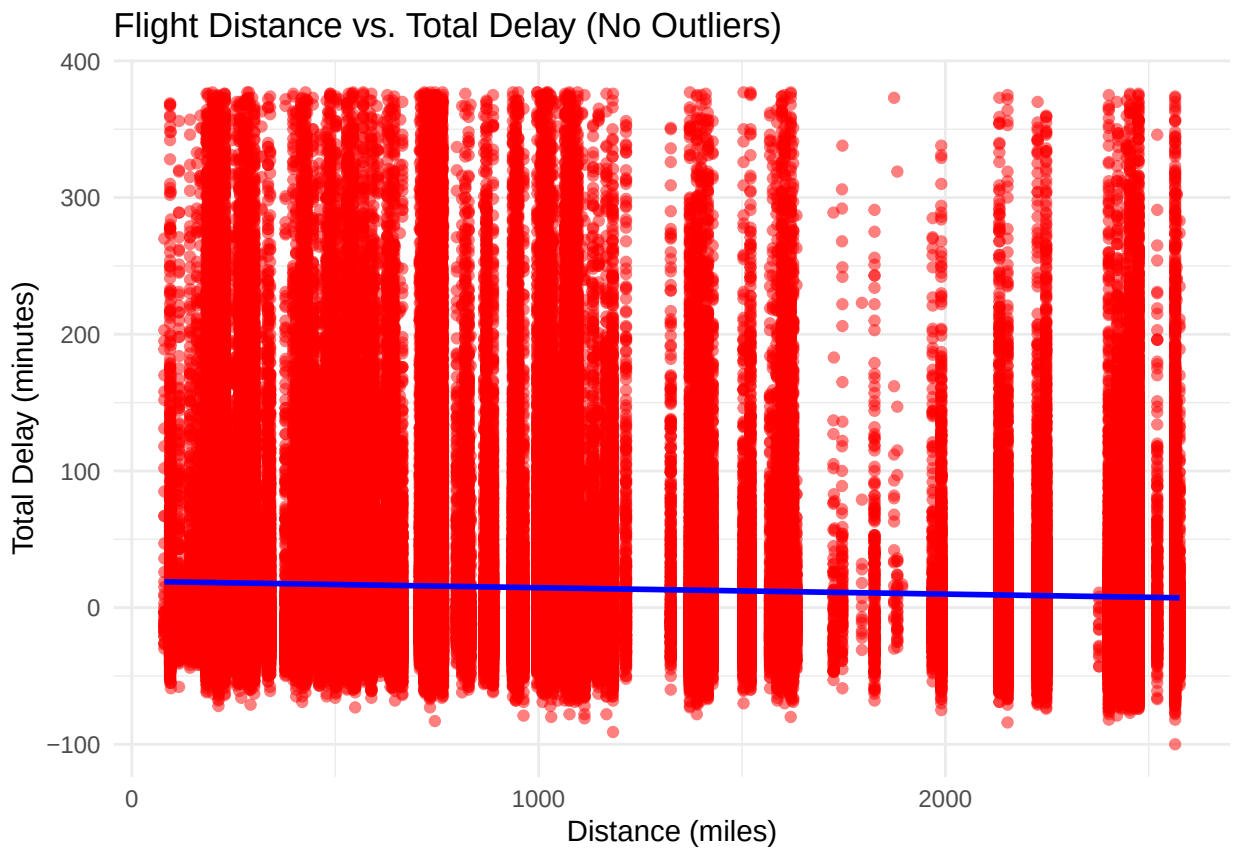
- From the scatter plot of distance vs. total delay, what can you infer about the relationship between flight distance and delay? Is it linear or non-linear?

It can be inferred that there are certain flights, usually longer distance flights, will have less of a delay. Maybe this is due to stricter schedules or more technicians are on staff to make sure there aren't any delays. The most probable might be that longer distance flights fly at a higher altitude and at a higher speed than shorter distance flights.

```
print(distancevdelayplot)
```

This relationship is non-linear. If it were linear it would consistently increase or decrease, forming a clear stright line trend in the scatter-plot.

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



3. How do delays vary across months?

- From the line plot showing the average delay by month, do you notice any seasonal trends in flight delays? Are delays higher in certain months?

Typically, in the summer months there are longer delays. This is most likely due to weather patterns during summer months in North America.

4. What are the implications of long delays?

- If we categorize delays as "Short", "Medium", and "Long", what might be the operational or customer experience impacts of long delays? How might airlines work to reduce these delays?
####Operational and Customer Experience Impacts

####Short Delays (0-30 minutes)

#####Operational Impact: Minimal; airlines can often make up time in the air or optimize turnaround times.

#####Customer Experience: Mild inconvenience but usually tolerated by passengers.

####Medium Delays (31-120 minutes)

#####Operational Impact: Potential gate congestion, crew scheduling conflicts, and increased fuel costs due to rerouting.

#####Customer Experience: Annoyance, possible missed connections, and frustration for business travelers.

####Long Delays (>120 minutes)

####Operational Impact: Major disruptions, flight rescheduling, cascading delays across airline networks, and increased costs for compensation and rebooking.

####Customer Experience: High frustration, missed international connections, overnight stays, and a negative perception of the airline.

Strategies to Reduce Delays #####Better Scheduling & Buffer Time: Airlines can build small buffers into flight schedules to absorb minor delays.

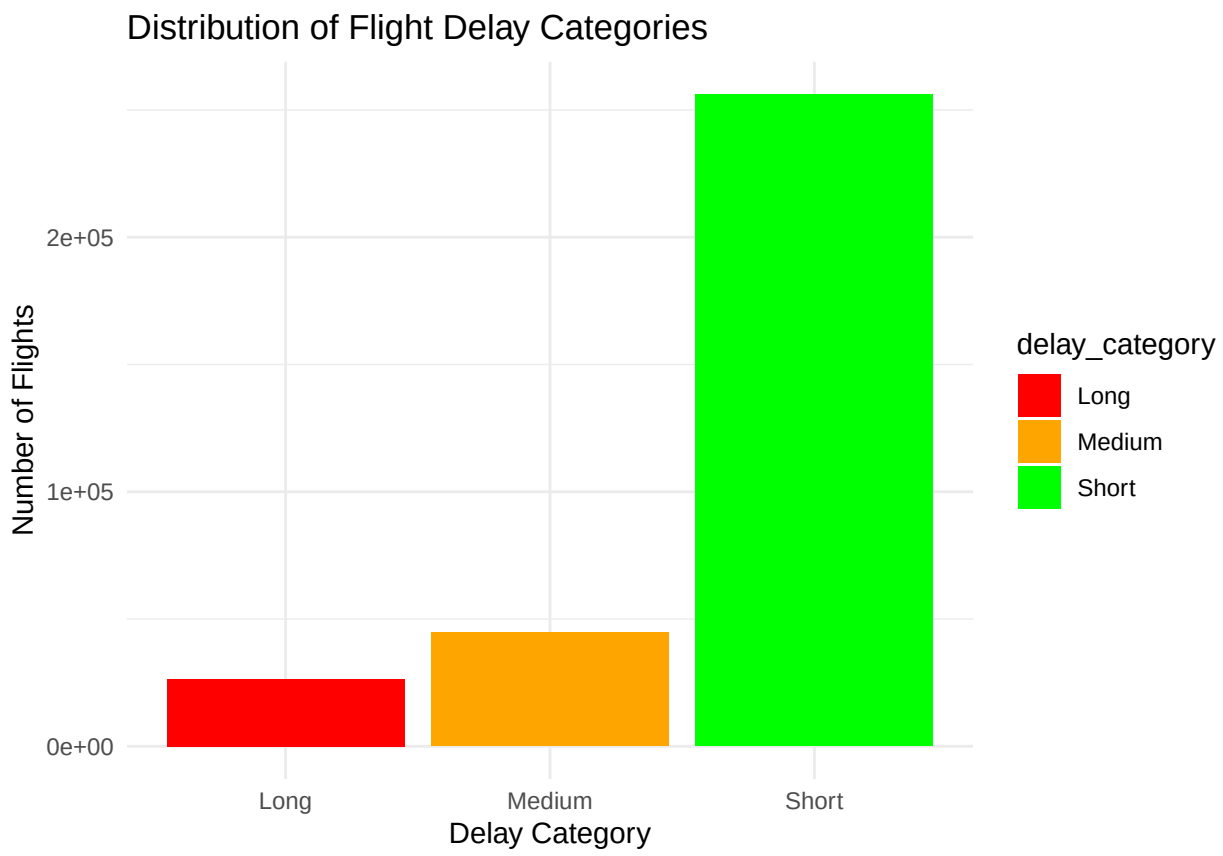
#####Improved Air Traffic Management: Coordinating with air traffic control to optimize takeoffs and landings.

#####Predictive Analytics: Using data to forecast potential delay risks and proactively adjust schedules.

#####Ground Efficiency: Faster turnaround times and better baggage handling to avoid extended gate holds.

#####Customer Communication: Providing real-time updates, alternative flights, and compensation to improve passenger experience.

```
# Categorize delays
flights_clean1 <- flights_clean1 %>%
  mutate(delay_category = case_when(
    total_delay <= 30 ~ "Short",
    total_delay > 30 & total_delay <= 120 ~ "Medium",
    total_delay > 120 ~ "Long"
  ))

# Count the number of flights in each category
delay_counts <- flights_clean1 %>%
  group_by(delay_category) %>%
  summarise(count = n(), .groups = "drop")

# Plot the distribution of delay categories
ggplot(delay_counts, aes(x = delay_category, y = count, fill = delay_category)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Distribution of Flight Delay Categories",
    x = "Delay Category",
    y = "Number of Flights"
  ) +
  theme_minimal() +
  scale_fill_manual(values = c("Short" = "green", "Medium" = "orange", "Long" = "red"))
```



Linear Regression Model

```
# Ensure necessary columns are available
flights_clean1 <- flights_clean1 %>%
  mutate(total_delay = dep_delay + arr_delay)

# Build a linear regression model
delay_model <- lm(total_delay ~ dep_delay + distance, data = flights_clean1)

# Display model summary
print(summary(delay_model))
```

```
##
## Call:
## lm(formula = total_delay ~ dep_delay + distance, data = flights_clean1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -107.944  -11.069   -2.019    8.728  205.562
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.213e+00  5.560e-02  -57.78   <2e-16 ***
## dep_delay    2.018e+00  7.823e-04 2579.54   <2e-16 ***
```

```
## distance    -2.551e-03  4.259e-05  -59.88   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.93 on 327343 degrees of freedom
## Multiple R-squared:  0.9532, Adjusted R-squared:  0.9532
## F-statistic: 3.334e+06 on 2 and 327343 DF,  p-value: < 2.2e-16
```

Coorelation

```r
# Compute correlation between flight distance and total delay
correlation_value <- cor(flights_clean1$distance, flights_clean1$total_delay, use = "complete.obs")

# Print correlation value
print(correlation_value)
```

```
## [1] -0.04379817
```

This value is very close to 0. It is slightly negative which indicates a very weak negative coorelation. As flight distance increases, total delay time slightly decreases.