BUSN32100 Final Exam

12/03/2022

Research Question Using the dataset from US Department of Transportation in 2015, I want to explore the patterns of flight departure delay. Specifically, answering the following questions:

- First, overall, whether the flight delay time or percentage related to the cities that the flight departure from or arriving, whether the percentage of delay is related to time, including departure month, day of the week and scheduled hour.
- Second, whether the flight departure delay related to air time delay or distance. Does a departure delay also likely to cause air time delay.
- Third, does the flight delay related to the airline. And if so in which month or hour each airline performs better
- Lastly, if we are departing from Chicago, which month, hour and airline will be the best choice
- Access final cleaned dataset here

1 Data Cleaning

1.1 Import data

```
# Load data
files <- list.files(pattern = "*.csv")  #read all csv files
list2env(
  lapply(
    setNames(files, make.names(gsub("_.*|*.csv", "", files))),
    read.csv
  ),
  envir = .GlobalEnv #save as separate df in Global Environment
)
```

```
## <environment: R_GlobalEnv>
```

```
# Check dataset
glimpse(flights)
```

```
## Rows: 5,819,079
## Columns: 31
## $ YEAR                <int> 2015, 2015, 2015, 2015, 2015, 2015, 2015, 2015, 20~
## $ MONTH               <int> 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,~
## $ DAY_OF_WEEK         <int> 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,~
```

```
## $ AIRLINE             <chr> "AS", "AA", "US", "AA", "AS", "DL", "NK", "US", "A~
## $ FLIGHT_NUMBER       <int> 98, 2336, 840, 258, 135, 806, 612, 2013, 1112, 117~
## $ TAIL_NUMBER         <chr> "N407AS", "N3KUAA", "N171US", "N3HYAA", "N527AS", ~
## $ ORIGIN_AIRPORT      <chr> "ANC", "LAX", "SFO", "LAX", "SEA", "SFO", "LAS", "~
## $ DESTINATION_AIRPORT <chr> "SEA", "PBI", "CLT", "MIA", "ANC", "MSP", "MSP", "~
## $ SCHEDULED_DEPARTURE <int> 5, 10, 20, 20, 25, 25, 25, 30, 30, 30, 30, 35, 35,~
## $ DEPARTURE_TIME      <int> 2354, 2, 18, 15, 24, 20, 19, 44, 19, 33, 24, 27, 3~
## $ DEPARTURE_DELAY     <int> -11, -8, -2, -5, -1, -5, -6, 14, -11, 3, -6, -8, 0~
## $ TAXI_OUT            <int> 21, 12, 16, 15, 11, 18, 11, 13, 17, 12, 12, 21, 18~
## $ WHEELS_OFF          <int> 15, 14, 34, 30, 35, 38, 30, 57, 36, 45, 36, 48, 53~
## $ SCHEDULED_TIME      <int> 205, 280, 286, 285, 235, 217, 181, 273, 195, 221, ~
## $ ELAPSED_TIME        <int> 194, 279, 293, 281, 215, 230, 170, 249, 193, 203, ~
## $ AIR_TIME            <int> 169, 263, 266, 258, 199, 206, 154, 228, 173, 186, ~
## $ DISTANCE            <int> 1448, 2330, 2296, 2342, 1448, 1589, 1299, 2125, 14~
## $ WHEELS_ON           <int> 404, 737, 800, 748, 254, 604, 504, 745, 529, 651, ~
## $ TAXI_IN             <int> 4, 4, 11, 8, 5, 6, 5, 8, 3, 5, 4, 7, 4, 5, 4, 4, 4~
## $ SCHEDULED_ARRIVAL   <int> 430, 750, 806, 805, 320, 602, 526, 803, 545, 711, ~
## $ ARRIVAL_TIME        <int> 408, 741, 811, 756, 259, 610, 509, 753, 532, 656, ~
## $ ARRIVAL_DELAY       <int> -22, -9, 5, -9, -21, 8, -17, -10, -13, -15, -30, -~
## $ DIVERTED            <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ CANCELLED           <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ CANCELLATION_REASON <chr> "", "", "", "", "", "", "", "", "", "", "", "", ""~
## $ AIR_SYSTEM_DELAY    <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
## $ SECURITY_DELAY      <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
## $ AIRLINE_DELAY       <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
## $ LATE_AIRCRAFT_DELAY <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
## $ WEATHER_DELAY       <int> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
```

```
# Check supporting dataset
# Airport information
glimpse(airports)
```

```
## Rows: 322
## Columns: 7
## $ IATA_CODE <chr> "ABE", "ABI", "ABQ", "ABR", "ABY", "ACK", "ACT", "ACV", "ACY~
## $ AIRPORT   <chr> "Lehigh Valley International Airport", "Abilene Regional Air~
## $ CITY      <chr> "Allentown", "Abilene", "Albuquerque", "Aberdeen", "Albany",~
## $ STATE     <chr> "PA", "TX", "NM", "SD", "GA", "MA", "TX", "CA", "NJ", "AK", ~
## $ COUNTRY   <chr> "USA", "USA", "USA", "USA", "USA", "USA", "USA", "USA", "USA~
## $ LATITUDE  <dbl> 40.7, 32.4, 35.0, 45.4, 31.5, 41.3, 31.6, 41.0, 39.5, 51.9, ~
## $ LONGITUDE <dbl> -75.4, -99.7, -106.6, -98.4, -84.2, -70.1, -97.2, -124.1, -7~
```

```
# Missing values
colSums(is.na(airports))
```

```
## IATA_CODE   AIRPORT      CITY     STATE   COUNTRY  LATITUDE LONGITUDE
##         0         0         0         0         0         3         3
```

```
# Fill out missing values
na_airport <- airports[is.na(airports$LATITUDE), "IATA_CODE"] # find missing airport IATA code
# Create empty tibble with column names to store the info
airport_info <- tibble(
```

```
  "IATA_CODE" = character(),
  "LATITUDE" = numeric(),
  "LONGITUDE" = numeric()
)
# Function to find the location info for missing values
i <- 1
for (i in seq_along(na_airport)) {
  airport_info[i, 1] <- na_airport[i] # IATA_CODE
  airport_info[i, 2:3] <- airport_location(na_airport[i]) # geo location info
  i <- i + 1
}
# Combine to the original Airport data frame
airports[is.na(airports$LATITUDE), "LATITUDE"] <- airport_info$LATITUDE
airports[is.na(airports$LONGITUDE), "LONGITUDE"] <- airport_info$LONGITUDE

# Airline information
glimpse(airlines)
```

```
## Rows: 14
## Columns: 2
## $ IATA_CODE <chr> "UA", "AA", "US", "F9", "B6", "OO", "AS", "NK", "WN", "DL", ~
## $ AIRLINE   <chr> "United Air Lines Inc.", "American Airlines Inc.", "US Airwa~
```

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

```
## IATA_CODE   AIRLINE
##         0         0
```

Explanation

- There are multiple airport information missing in the dataset provided in Kaggle. Therefore, I used the package in R to fill out the missing information.
- Packages used in the analysis are include in the RMD file but not the PDF.

1.2 Merge Dataset

```
flight_merged <- flights %>%
  left_join(airports, by = c("ORIGIN_AIRPORT" = "IATA_CODE")) %>%
  left_join(airports, by = c("DESTINATION_AIRPORT" = "IATA_CODE")) %>%
  select(-contains("COUNTRY")) %>% # all flights were departing in the US
  clean_names(
    replace = c(`.x` = "_ORIGIN", `.y` = "_DESTINATION")
  ) %>%
  left_join(airlines, by = c("airline" = "IATA_CODE")) %>%
  rename(
    "airline_name" = AIRLINE,
    "scheduled_airtime" = scheduled_time
  ) %>%
  filter(cancelled == 0) %>% # filter out cancelled flights
  select(
```

```
    -contains(c("taxi", "wheels", "elapsed", "arrival", "cancell")),
    -diverted, -year, -c(air_system_delay:weather_delay)
  ) # remove non-informative column
```

1.3 Check Data

```
# Check missing values
colSums(is.na(flight_merged))
```

```
##                 month                   day           day_of_week
##                     0                     0                     0
##               airline         flight_number           tail_number
##                     0                     0                     0
##        origin_airport   destination_airport   scheduled_departure
##                     0                     0                     0
##        departure_time       departure_delay     scheduled_airtime
##                     0                     0                     1
##              air_time              distance        airport_origin
##                 15187                     0                483711
##           city_origin          state_origin       latitude_origin
##                483711                483711                483711
##      longitude_origin   airport_destination      city_destination
##                483711                483711                483711
##     state_destination  latitude_destination longitude_destination
##                483711                483711                483711
##          airline_name
##                     0
```

```
# Remove rows with empty air time information
flight_merged <- flight_merged %>%
  filter(!is.na(air_time))
# Check rows with empty airport info
summary(flight_merged[is.na(flight_merged$airport_origin), "month"])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      10      10      10      10      10      10
```

```
#double check to make sure no missing value is included: colSums(is.na(flight_merged))

# Check duplicated values
flight_dup <- flight_merged %>% get_dupes(
  month, day, scheduled_departure, # same time
  origin_airport, destination_airport, # same location schedule
  airline, tail_number # same flight
)
flight_dup <- flight_dup %>%
  filter(flight_number != 1865) # only keep the last one which is correct

flight_merged <- anti_join(flight_merged, flight_dup, # remove wrong duplicated data
  by = c(
```

```
    "month", "day", "scheduled_departure",
    "origin_airport", "destination_airport",
    "airline", "tail_number", "flight_number"
  )
) %>%
  select(-flight_number)
```

Interpretation

- After removing all na values from other columns, there are still 482872 in columns with missing airport information. It turns out that all of those data are from October. To make sure the time series analysis consistent, I did not remove those values.

2. Feature Engineering

2.1 Create a new column with air time delay

```
flight_merged <- flight_merged %>%
  mutate(air_time_delay = air_time - scheduled_airtime)
```

2.2. Formatting date time columns

```
# Function to combine date time info
clean_dttm <- function(year, month, day, time) {
  make_datetime(year, month, day, time %/% 100, time %% 100)
}
# Apply the function to the date set
flight_merged <- flight_merged %>%
  mutate(
    date = format(clean_dttm(2015, month, day, departure_time), "%b-%d"), # create date
    day_of_week = factor(
      as.factor(
        day_of_week # convert to factor column, for plotting
      ),
      labels = c(
        "Mon",
        "Tues",
        "Wed",
        "Thurs",
        "Fri",
        "Sat",
        "Sun"
      )
    ),
    scheduled_departure = hms((scheduled_departure %/% 100 * 3600 # Hour
      + scheduled_departure %% 100 * 60)), # Mins
    scheduled_dep_hour = hour(scheduled_departure) # Create ]sche_dep_hour column
  )
```

2.3 Create a column with departure delay dummy variable

```
flight_merged <- flight_merged %>%
  mutate(departure_delayed_dummy = ifelse(
    departure_delay >= 15,
    1, # if delay exceeds 15 mins, equals 1
    0 # if delay is shorter than 15 mins, equals 0
  )) %>%
  select(
    date, month, day, day_of_week, # select dataset column order
    scheduled_departure, scheduled_dep_hour,
    departure_delay, departure_delayed_dummy,
    scheduled_airtime, air_time_delay, distance,
    airline, airline_name, everything()
  )
# Write to csv
# write_csv(flight_merged, "flight_merged_clean.csv")
```

3. Exploratory Analysis

3.1 Distribution of data

```
as.data.frame(summary(select_if(flight_merged, is.numeric))) %>%
  select(-Var1) %>%
  separate(Freq, into = c("attribute", "value"), sep = ":", extra = "merge") %>%
  mutate(attribute = str_remove(attribute, "\\.")) %>%
  rename(Variable = "Var2") %>%
  pivot_wider(
    names_from = attribute,
    values_from = value
  ) %>%
  select(-contains("NA")) %>%
  filter(!grepl("longitude|latitude", Variable)) %>% #remove geo location columns
  kable(
    caption = "Summary Statistics",
    format.args = list(scientific = FALSE)
  ) %>%
  kable_classic() %>%
  kable_styling(
    position = "center", font_size = 8,
    full_width = T, html_font = "Cambria",
    c("stripend", "bordered")
  )
```

3.1.1 Summary Statistics

```
# Find distribution of month
count(flight_merged, month) %>% arrange(desc(n))
```

Table 1: Summary Statistics

Variable	Min	1st Qu	Median	Mean	3rd Qu	Max
month	1.00	4.00	7.00	6.55	9.00	12.00
day	1.0	8.0	16.0	15.7	23.0	31.0
scheduled_dep_hour	0	9	13	13	17	23
departure_delay	-82	-5	-2	9	7	1988
departure_delayed_dummy	0.000	0.000	0.000	0.184	0.000	1.000
scheduled_airtime	18	85	123	142	174	718
air_time_delay	-216.0	-34.0	-28.0	-28.4	-22.0	189.0
distance	31	373	650	824	1065	4983
departure_time	1	921	1330	1335	1740	2400
air_time	7	60	94	114	144	690

3.1.2 Distribution of Categorical columns

```
##    month      n
## 1      7 514384
## 2      8 503956
## 3      6 492847
## 4      3 492138
## 5      5 489641
## 6     10 482872
## 7      4 479251
## 8     12 469716
## 9     11 462365
## 10     9 462151
## 11     1 457013
## 12     2 407663
```

```
# Find the overall percentage of delayed flights
count(flight_merged, departure_delayed_dummy)
```

```
##   departure_delayed_dummy       n
## 1                       0 4663576
## 2                       1 1050421
```

```
# Find distribution of origin and destination city
count(filter(flight_merged, !is.na(city_destination)), city_origin) %>%
  arrange(desc(n)) %>%
  head(10) %>%
  cbind(count(filter(flight_merged, !is.na(city_destination)), city_destination) %>%
  arrange(desc(n)) %>%
  head(10))
```

```
##          city_origin      n  city_destination      n
## 1            Chicago 355239           Chicago 354342
## 2            Atlanta 343506           Atlanta 343076
## 3  Dallas-Fort Worth 232647 Dallas-Fort Worth 231764
## 4            Houston 195151           Houston 194596
## 5             Denver 193397            Denver 193033
## 6        Los Angeles 192003       Los Angeles 192136
## 7           New York 186497          New York 186114
## 8            Phoenix 145552     San Francisco 145409
```

```
## 9      San Francisco 145491           Phoenix 145378
## 10         Las Vegas 131937         Las Vegas 132124
```

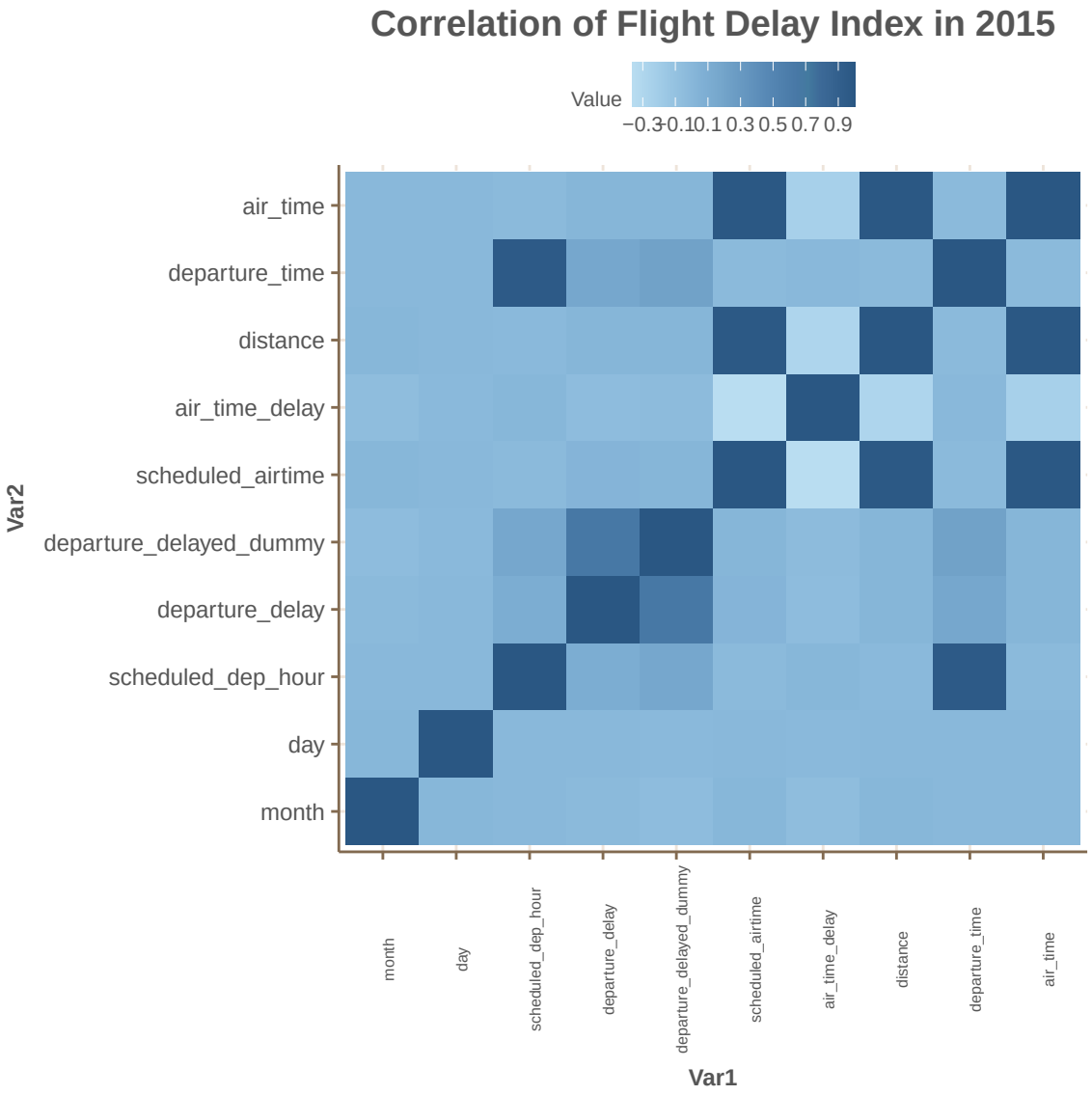
Interpretation

- Based on the analysis above, we can see that most flights are not delayed, while the maximum delay minutes were 1988. The dataset contains flight info from the start of the year to the end of 2015. After cleaning the dataset, there are 5819079 observations left and those data are relatively evenly distributed by Month.
- Chicago has the highest number of flights either by departure or arrival number. The top 10 airports with the highest flight number does not change either by departure or arrival.

3.2 Overall Analysis

```
# Set theme
ggthemr("fresh")
theme <-
  theme(
    plot.title = element_text(size = 14, face = "bold", hjust = 0.5),
    plot.subtitle = element_text(size = 10, face = "italic", hjust = 0.5),
    axis.title = element_text(size = 9, face = "bold"),
    plot.caption = element_text(size = 6, face = "italic", hjust = 0),
    axis.text = element_text(size = 9),
    legend.text = element_text(size = 8),
    legend.title = element_text(size = 8)
  )
# Calculate correlation
corr_mat <- round(cor(select_if(
  select(flight_merged, -contains(c("latitude", "longitude"))), is.numeric
)), 2)
# Reshape correlation data
melted_corr_mat <- melt(corr_mat)
# Plot
ggplot(melted_corr_mat) +
  geom_tile(aes(Var1, Var2, fill = value)) +
  scale_x_discrete(labels = function(x) str_wrap(x, width = 5)) +
  scale_fill_continuous_tableau(
    name = "Value",
    breaks = seq(-0.5, 1, 0.2)
  ) +
  labs(title = "Correlation of Flight Delay Index in 2015") +
  theme +
  theme(axis.text.x = element_text(size = 6, angle = 90)) +
  legend_top()
```

3.2.1 Heatmap Correlation

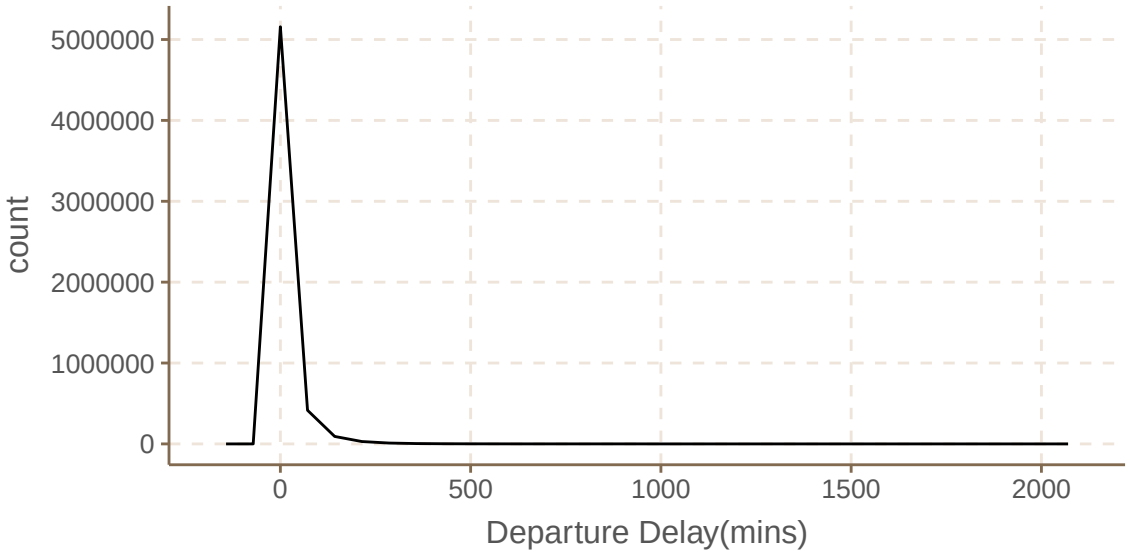


Explanation

- Departure delay is slightly correlated with departure hour, airtime and real departure time.

```
flight_merged %>%
  ggplot() +
  geom_freqpoly(aes(departure_delay)) +
  labs(x = "Departure Delay(mins)")
```

3.2.2 Distribution of delayed time



Explanation

- Most flights were not delayed at departure. And for those delayed flights, most were short delays. However, there is some abnormal delays which was more nearly 2000 mins.

```
# Write function to calculate the percentage by selected group
# function
#one disaggregation variable
get_delay_pct <- function(col, df) {
  # get data frame
  data <- df %>%
    group_by(df[[col]]) %>%
    summarise(delay_pct = mean(departure_delayed_dummy))

  colnames(data) <- c(col, "delay_pct")
  data
}

#multiple disaggregation variables
get_delay_pct_db <- function(col1, col2, df) {
  # get data frame
  data <- df %>%
    group_by(df[[col1]], df[[col2]]) %>%
    summarise(delay_pct = mean(departure_delayed_dummy))
  colnames(data) <- c(col1, col2, "delay_pct")
  data
}

# Generate Plot
get_plt <- function(var, df) {
  plt <- df %>%
    ggplot() +
```

```
    geom_line(aes(factor(df[[var]]), delay_pct,
                  group = 1)) +
    labs(
    y = "Delay Percentage",
    caption = "Data Source: US DOT, Year 2015"
  ) +
    theme
  plt
}
```

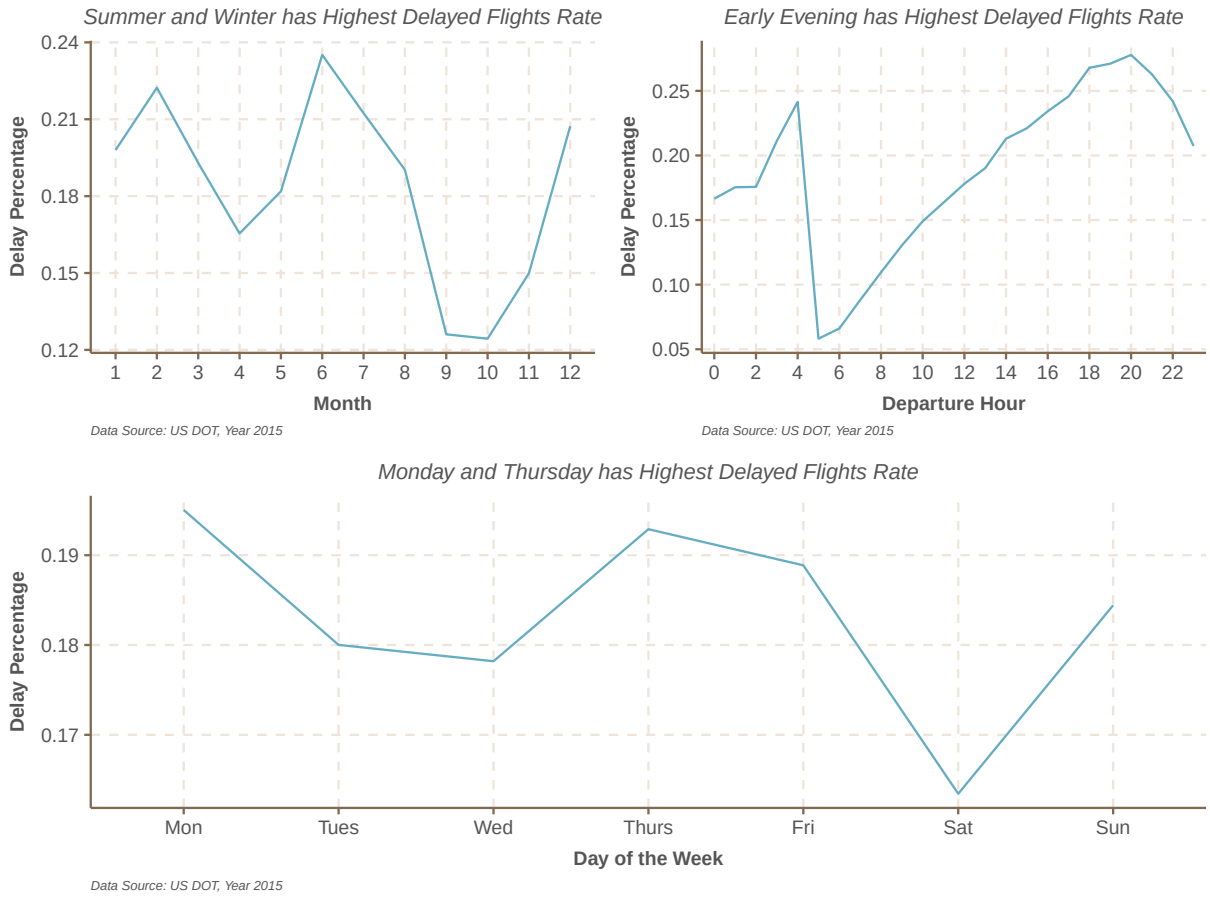
```
# By month
plt_month <- get_plt("month", get_delay_pct("month", flight_merged)) +
  labs(
    subtitle = "Summer and Winter has Highest Delayed Flights Rate",
    x = "Month"
  )

# By day of the week
plt_week <- get_plt("day_of_week", get_delay_pct("day_of_week", flight_merged)) +
  labs(
    subtitle = "Monday and Thursday has Highest Delayed Flights Rate",
    x = "Day of the Week"
  )

# By hour
plt_hour <- get_plt("scheduled_dep_hour", get_delay_pct("scheduled_dep_hour",
                                                        flight_merged)) +
  labs(
    subtitle = "Early Evening has Highest Delayed Flights Rate",
    x = "Departure Hour"
    ) +
  scale_x_discrete(breaks = seq(0, 23, 2))
#Check number of flights
flight_merged %>%
  group_by(scheduled_dep_hour) %>%
  summarise(count = n()) %>%
  arrange(count)
```

3.2.3 Delayed percentage by time

```
## # A tibble: 24 x 2
##    scheduled_dep_hour  count
##                 <int>  <int>
##  1                  4    526
##  2                  3    767
##  3                  2   1383
##  4                  1   5091
##  5                  0  14504
##  6                 23  43646
##  7                 22 115358
##  8                  5 115647
```

```
##  9                 21 183406
## 10                 20 253612
## # ... with 14 more rows
```

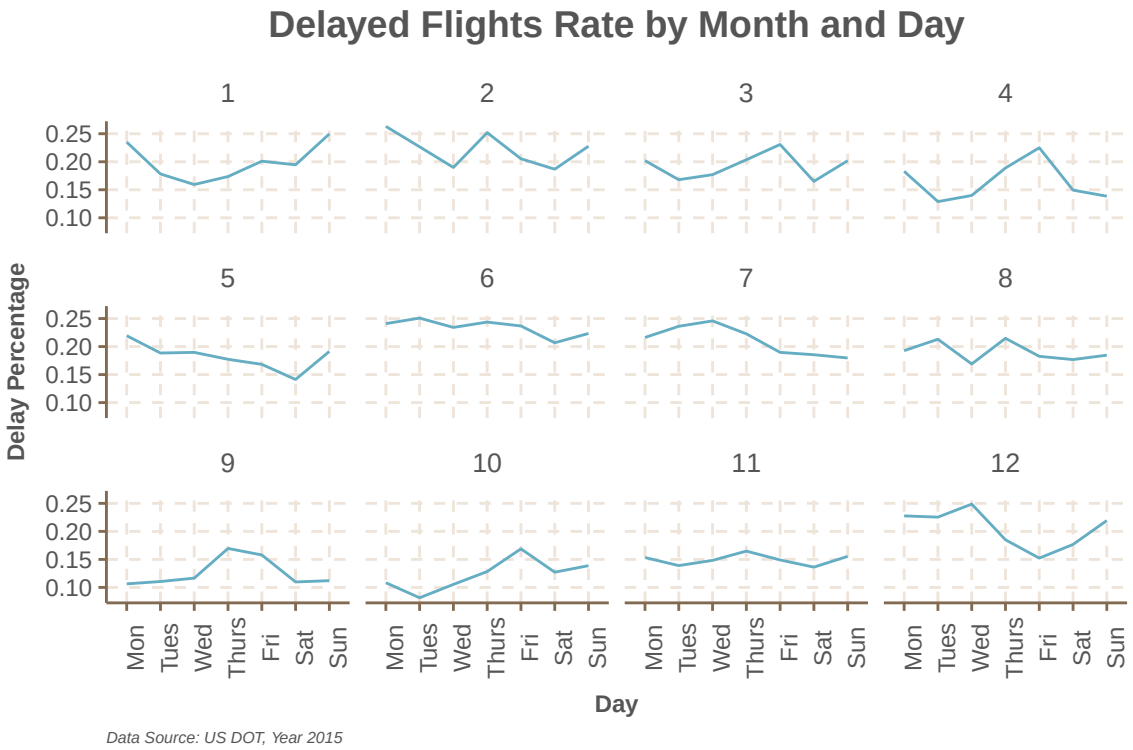
```
(plt_month | plt_hour) /
  plt_week
```



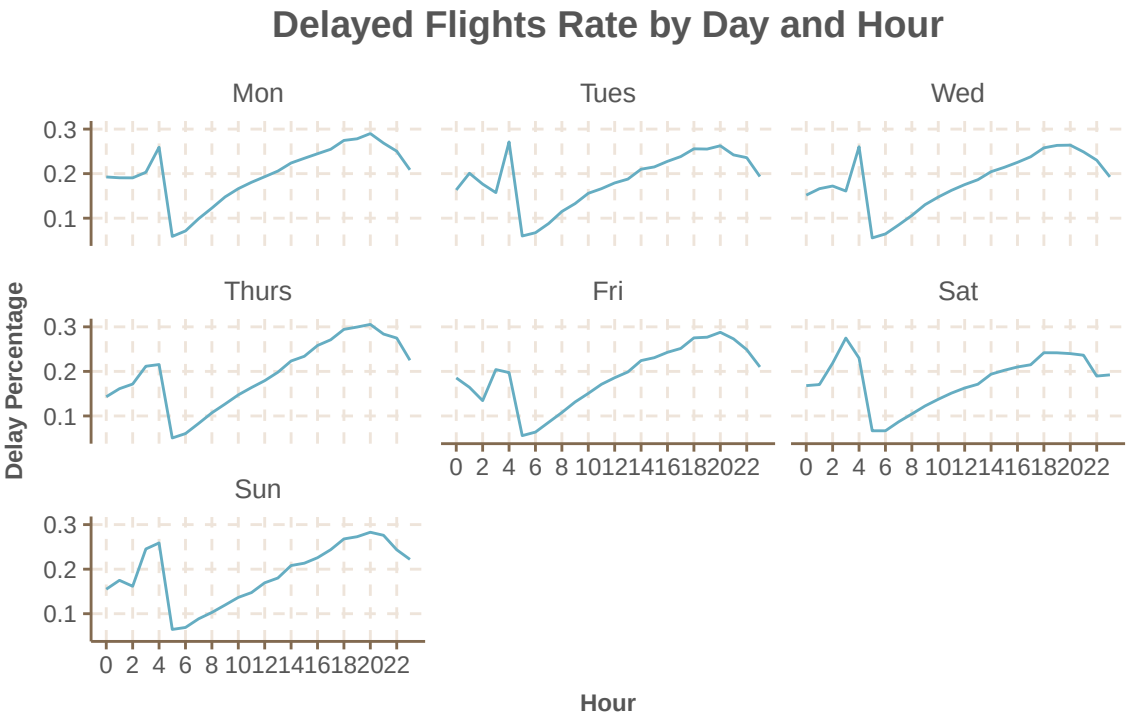
Explanation

- June has the highest delay rate, followed by other months in summer and February. Delay rate in Autumn is pretty low.
- During the week, midweek has higher delay rate. The rate then goes down and increases sharply on Sunday and Monday. The trend is potentially related to work schedule. Near the end of the week people need to come back to home city or plans to go out during the weekend, then before workday starts they need to take flights back. As the demand increases, delay could be more likely to happen.
- During the day, 4am-6am has the lowest delay rate, but by checking the data we know that the number of flights are also significantly less compared to any other time period. Starting early morning, the delay rate continues to increase, peaking at early night around 8pm. Besides, 2am has the second highest delay rate.

```r
# By month and day
get_delay_pct_db("month", "day_of_week", flight_merged) %>%
  ggplot() +
  geom_line(aes(day_of_week, delay_pct,
                group = 1)) +
  labs(
    title = "Delayed Flights Rate by Month and Day",
    x = "Day",
    y = "Delay Percentage",
    caption = "Data Source: US DOT, Year 2015"
  ) +
  facet_wrap(~month) +
  theme +
  theme(axis.text.x = element_text(angle = 90))
```




```r
# By month and hour
get_delay_pct_db("scheduled_dep_hour", "day_of_week", flight_merged) %>%
  ggplot() +
  geom_line(aes(scheduled_dep_hour, delay_pct)) +
  scale_x_continuous(breaks = seq(0, 23, 2)) +
  labs(
    title = "Delayed Flights Rate by Day and Hour",
    x = "Hour",
    y = "Delay Percentage",
    caption = "Data Source: US DOT, Year 2015"
  ) +
  facet_wrap(~day_of_week) +
  theme
```



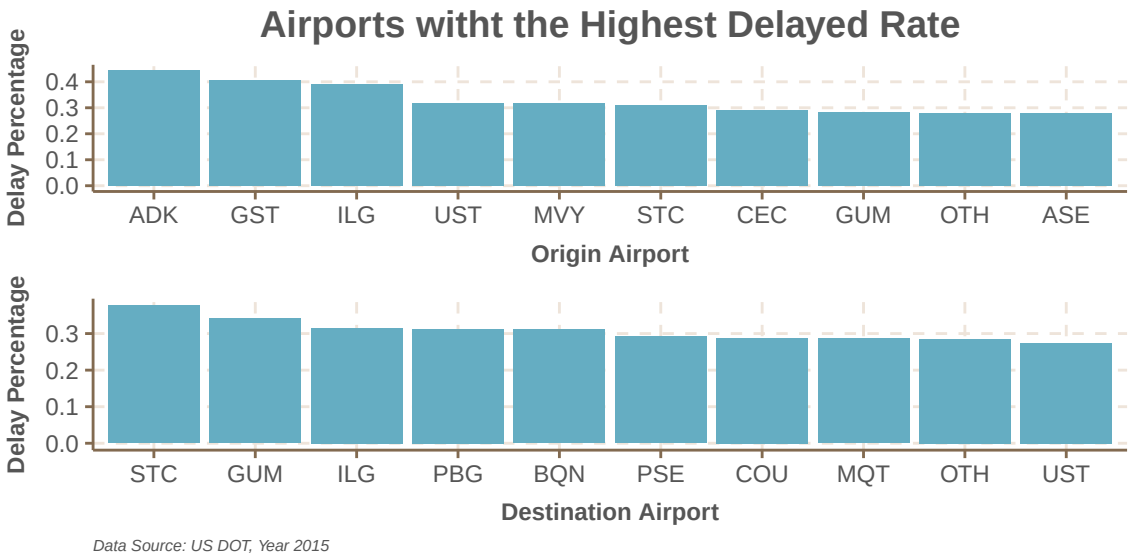
Explanation

- Distribution of delay rate during the week varies across month. In general, summer months have high delay rate throughout the week, with a slightly higher rate during the beginning of a week. In December, early days during the week are more likely to have delay. All other months follow the overall trend.
- The delay pattern by hour is quite similar during throughout the week.

```
# By origin
plt_delay_ori_h <- get_delay_pct("origin_airport", filter(flight_merged, month!= 10)) %>%
  slice_max(order_by = delay_pct, n = 10) %>%
  ggplot() +
  geom_col(aes(fct_reorder(origin_airport, desc(delay_pct)),
               delay_pct),
           fill = "#65ADC2") +
  labs(
    title = "Airports witht the Highest Delayed Rate",
    x = "Origin Airport",
    y = "Delay Percentage") +
  theme
plt_delay_des_h <- get_delay_pct("destination_airport", filter(flight_merged, month!= 10)) %>%
  slice_max(order_by = delay_pct, n = 10) %>%
  ggplot() +
  geom_col(aes(fct_reorder(destination_airport, desc(delay_pct)),
               delay_pct),
```

```
          fill = "#65ADC2") +
  labs(
    x = "Destination Airport",
    y = "Delay Percentage",
    caption = "Data Source: US DOT, Year 2015") +
  theme

ggarrange(plt_delay_ori_h, plt_delay_des_h,
          nrow = 2)
```

3.2.4 Delayed percentage by origin and destination

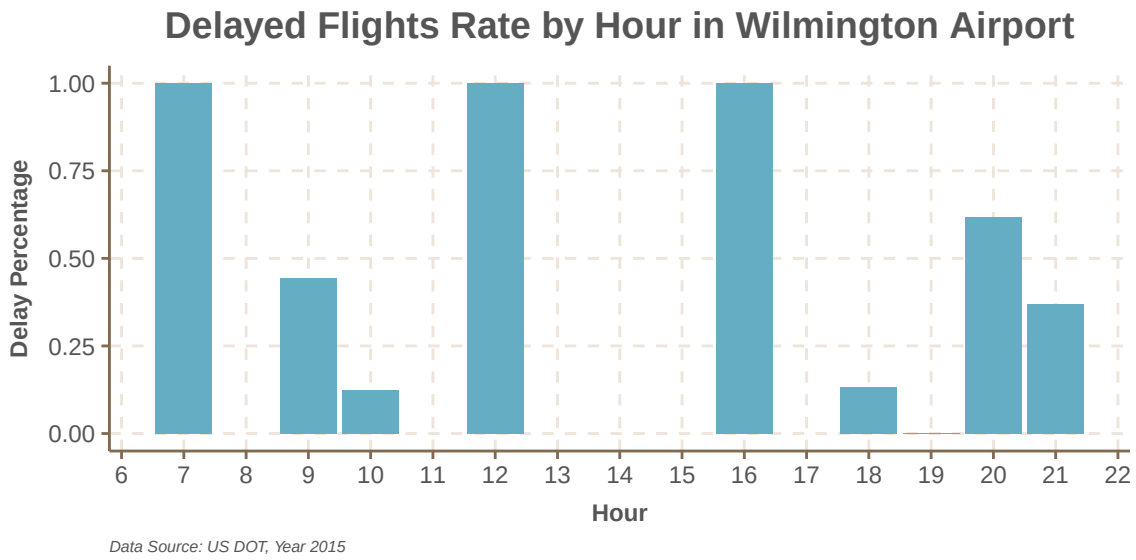



```
# ILG By month
get_delay_pct("month", filter(flight_merged, origin_airport == "ILG")) %>%
  arrange(desc(delay_pct))
```

```
## # A tibble: 4 x 2
##   month delay_pct
##   <int>     <dbl>
## 1     3     0.5
## 2     2     0.444
## 3     4     0.36
## 4     1     0.269
```

```
# By hour
get_delay_pct("scheduled_dep_hour", filter(flight_merged, origin_airport == "ILG")) %>%
  ggplot() +
  geom_col(aes(scheduled_dep_hour, delay_pct)) +
  scale_x_continuous(breaks = seq(1, 24, 1)) +
  labs(
    title = "Delayed Flights Rate by Hour in Wilmington Airport",
    x = "Hour",
    y = "Delay Percentage",
```

```
    caption = "Data Source: US DOT, Year 2015"
  ) +
  theme
```




```
# Destination
get_delay_pct("destination_airport", filter(flight_merged, origin_airport == "ILG")) %>%
  arrange(desc(delay_pct))
```

```
## # A tibble: 2 x 2
##   destination_airport delay_pct
##   <chr>                   <dbl>
## 1 MCO                     0.443
## 2 TPA                     0.294
```

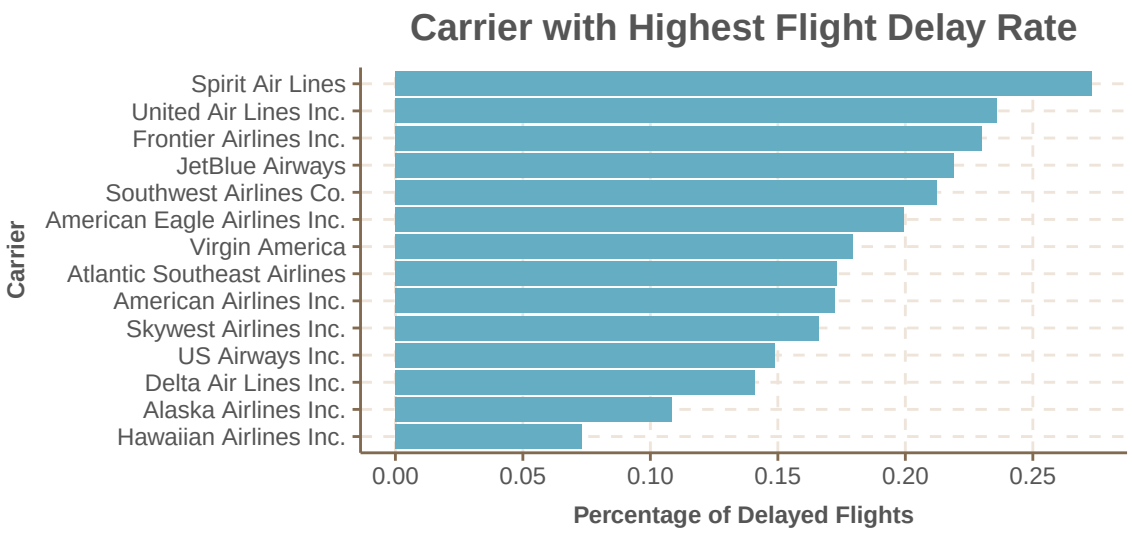
Explanation

- ADK airport has the highest delay rate both by origin and destination. By looking at the data, we find out that flights to or from ILG is most likely to delay in 7am, 12pm and 16pm, with flights to *Orlando(MCO)* or *Tampa(TPA)*

```
# Carriers with the highest delay rate in 2015
get_delay_pct("airline_name", flight_merged) %>%
  arrange(desc(delay_pct)) %>%
  ggplot() +
  geom_col(aes(x = fct_reorder(airline_name, delay_pct),
               y = delay_pct)
  ) +
  scale_y_continuous(breaks = seq(0, 0.25, 0.05)) +
  labs(title = "Carrier with Highest Flight Delay Rate",
       caption = "Data Source: US Department of Transportation, Year 2015",
```

```
      x = "Carrier",
      y = "Percentage of Delayed Flights") +
  coord_flip() +
  theme
```

3.2.5 Top Cases by unique carrier



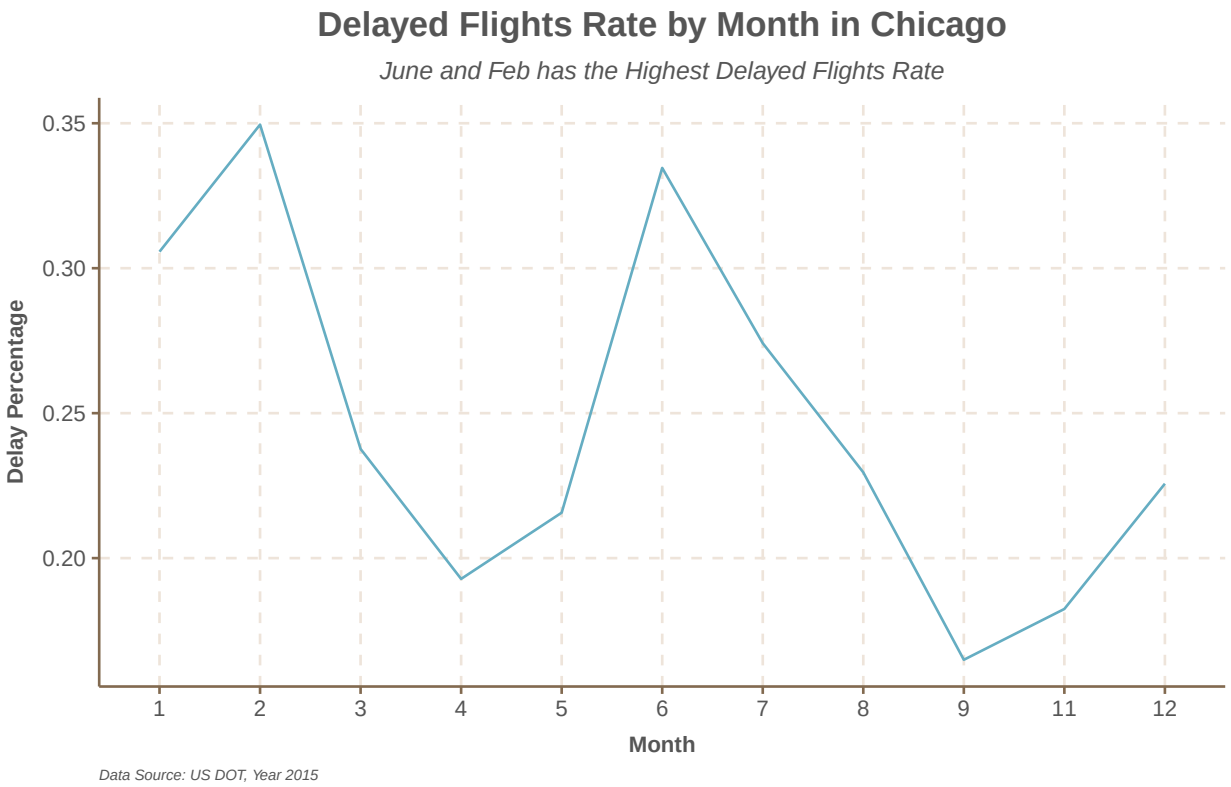

```
# Analysis:
#flight_merged %>% filter(departure_delay > 15) %>%
# mutate(departure_delay = winsorise(departure_delay, probs = c(0, 0.9))) %>%
# ggplot(aes(cut_interval(departure_delay, 6), air_time_delay)) + geom_boxplot()
```

3.2.6 Relationship of delay time and air time *Explanation*

- By running the analysis above, the distribution of air time delay does not seem to have a strong correlation with departure delay. The average air time delay covnerges below 0.

3.3 Analysis of Chicago Airports

```
chicago <- flight_merged %>%
  filter(city_origin == "Chicago")
# By month
get_plt("month", get_delay_pct("month", chicago)) +
  labs(
    title = "Delayed Flights Rate by Month in Chicago",
    subtitle = "June and Feb has the Highest Delayed Flights Rate",
    x = "Month"
  ) +
  theme
```




```
# Which Flight to choose in Feb or June
chicago %>%
  filter(month %in% c(2, 6)) %>%
  group_by(day_of_week, airline) %>%
  summarise(delay_pct = mean(departure_delayed_dummy)) %>%
  slice_min(order_by = delay_pct)
```

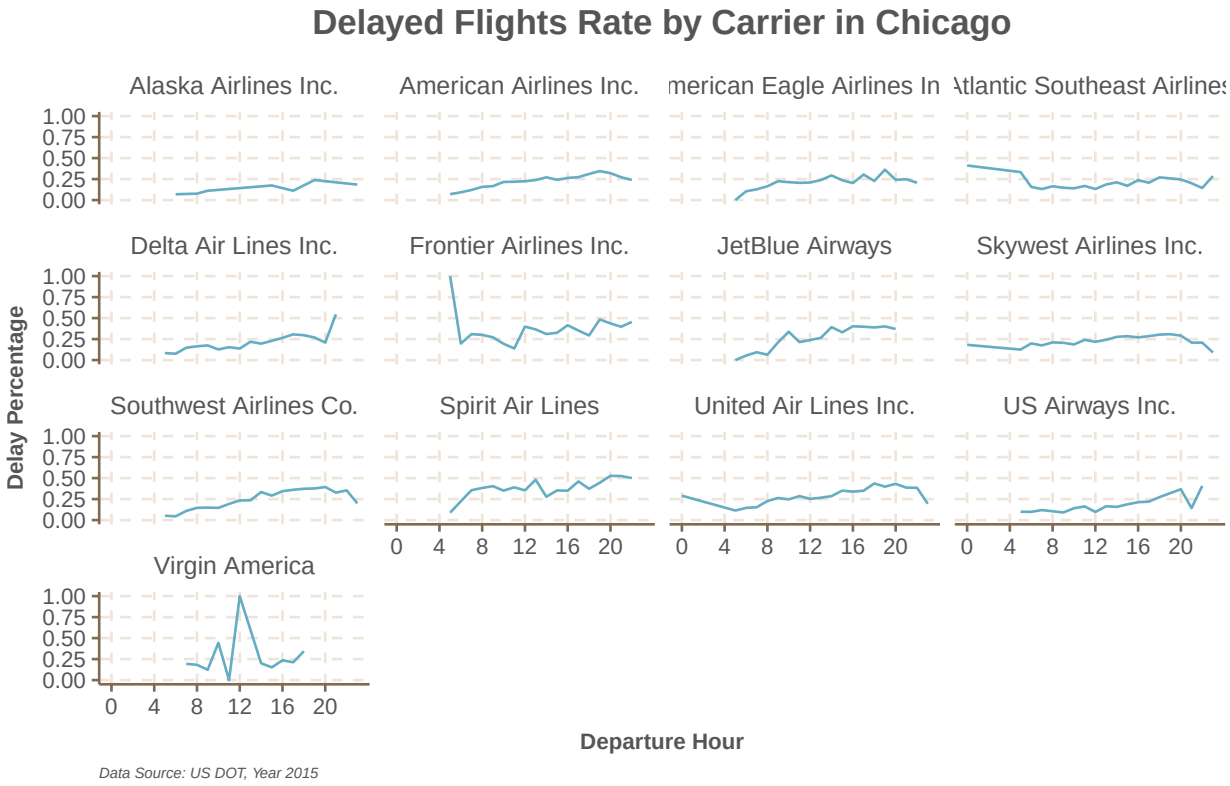
```
## # A tibble: 7 x 3
## # Groups:   day_of_week [7]
##   day_of_week airline delay_pct
##   <fct>       <chr>       <dbl>
## 1 Mon         AS          0.214
## 2 Tues        AS          0.05
## 3 Wed         US          0.0903
## 4 Thurs       AS          0.179
## 5 Fri         AS          0.179
## 6 Sat         US          0.134
## 7 Sun         AS          0.114
```

```
chicago %>%
  filter(month %in% c(2, 6)) %>%
  group_by(scheduled_dep_hour, airline) %>%
  summarise(delay_pct = mean(departure_delayed_dummy)) %>%
  slice_min(order_by = delay_pct)
```

```
## # A tibble: 19 x 3
```

```
## # Groups:   scheduled_dep_hour [19]
##    scheduled_dep_hour airline delay_pct
##                 <int> <chr>       <dbl>
##  1                  5 MQ          0
##  2                  6 AS          0.0333
##  3                  7 B6          0.0833
##  4                  8 B6          0.0588
##  5                  9 US          0.125
##  6                 10 B6          0.143
##  7                 11 F9          0.172
##  8                 12 US          0.154
##  9                 13 EV          0.255
## 10                 14 B6          0
## 11                 15 AS          0.193
## 12                 16 VX          0.222
## 13                 17 AS          0.123
## 14                 18 US          0.337
## 15                 19 AS          0.193
## 16                 20 DL          0
## 17                 21 US          0.0952
## 18                 22 F9          0.25
## 19                 23 OO          0
```

```
# By carrier
get_delay_pct_db("scheduled_dep_hour", "airline_name", chicago) %>%
  ggplot() +
  geom_line(aes(scheduled_dep_hour, delay_pct)) +
  scale_x_continuous(breaks = seq(0, 23, 4)) +
  labs(
    title = "Delayed Flights Rate by Carrier in Chicago",
    x = "Departure Hour",
    y = "Delay Percentage",
    caption = "Data Source: US DOT, Year 2015"
  ) +
  facet_wrap(~airline_name) +
  theme
```

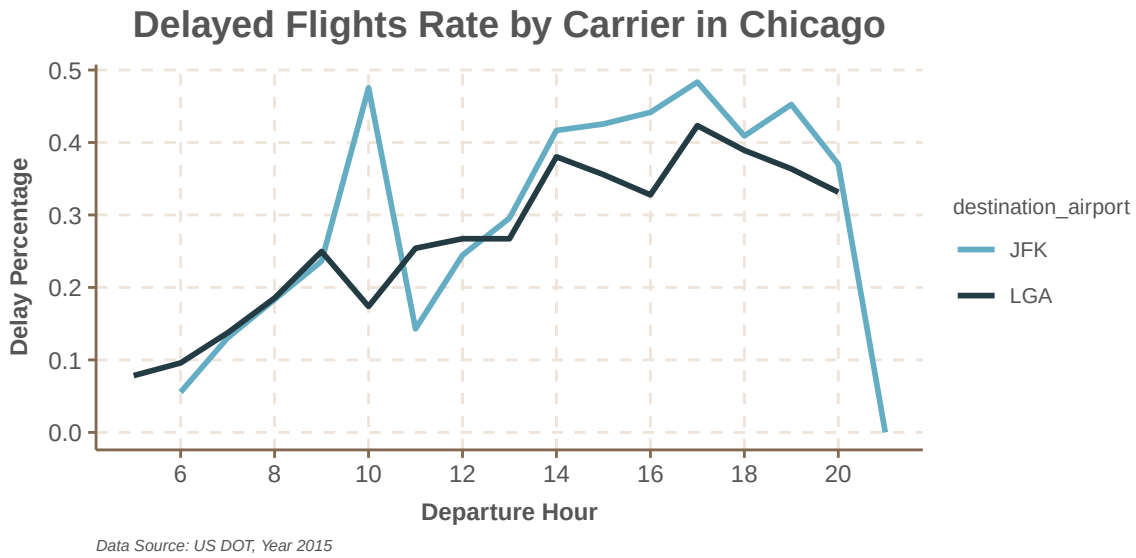



```
# By destination: if we're going to New  York
get_delay_pct_db("destination_airport", "airline_name", filter(chicago, city_destination == "New York")) %>%
  arrange(delay_pct, destination_airport)
```

```
## # A tibble: 8 x 3
## # Groups:   destination_airport [2]
##   destination_airport airline_name           delay_pct
##   <chr>               <chr>                      <dbl>
## 1 LGA                 Skywest Airlines Inc.      0.175
## 2 JFK                 Delta Air Lines Inc.       0.182
## 3 LGA                 American Airlines Inc.     0.217
## 4 LGA                 Spirit Air Lines           0.277
## 5 LGA                 Southwest Airlines Co.     0.313
## 6 LGA                 United Air Lines Inc.      0.320
## 7 JFK                 JetBlue Airways            0.321
## 8 JFK                 American Airlines Inc.     0.329
```

```
get_delay_pct_db("destination_airport", "scheduled_dep_hour",
                 filter(chicago, city_destination == "New York")) %>%
  arrange(delay_pct) %>%
  ggplot() +
  geom_line(
    aes(scheduled_dep_hour, delay_pct,
      color = destination_airport
    ),
    size = 1
  ) +
```

```
  scale_x_continuous(breaks = seq(0, 23, 2)) +
  labs(
    title = "Delayed Flights Rate by Carrier in Chicago",
    x = "Departure Hour",
    y = "Delay Percentage",
    caption = "Data Source: US DOT, Year 2015"
  ) +
  theme
```



Explanation

- In general, flights in June and February has the highest delay rate, which is likely caused by the weather condition and number of people on travel during the summer. The delay rate is also higher than the average across the nation.
- All flights have a higher delay rate at night and all carriers perform similarly on flight delay, while VX has an abnormally high delay rate compare to others around noon. Also, Frontier Airline is not a good choice in particular if we want to leave Chicago in early morning like 4am. Among all others, *JetBlue*, *Delta* and *Spirit* are not good choices as they have higher delay rate overall.
- If we want to go New York, *Skywest Airlines Inc.* to *LFA* and *Delta Air Lines Inc.* to *JFK* has a lower delay rate on average. And the delay rate at *JFK* is higher in most cases during the day except between *10am-12am*

4. Model

```
flight_merged <- flight_merged %>%
  mutate(day_of_week = as.numeric(day_of_week)) # convert to numeric for model
#a.Basic model
#Does departure delay related to departure time
reg <- lm(departure_delay ~ month + day_of_week + scheduled_dep_hour, data = flight_merged)
stargazer(reg, type = "text")
```

```
##
## ================================================
##                          Dependent variable:
##                      ---------------------------
##                            departure_delay
## ------------------------------------------------
## month                         -0.238***
##                                (0.005)
##
## day_of_week                   -0.232***
##                                (0.008)
##
## scheduled_dep_hour            0.844***
##                                (0.003)
##
## Constant                      0.781***
##                                (0.061)
##
## ------------------------------------------------
## Observations                  5,713,997
## R2                              0.013
## Adjusted R2                     0.013
## Residual Std. Error      36.700 (df = 5713993)
## F Statistic         24,645.000*** (df = 3; 5713993)
## ================================================
## Note:                  *p<0.1; **p<0.05; ***p<0.01
```

```
#b.Improve model
reg_full <- lm(departure_delay ~ month + day_of_week + scheduled_dep_hour
               + airline + air_time*distance, data = flight_merged)
stargazer(reg_full, type = "text")
```

```
##
## ================================================
##                          Dependent variable:
##                      ---------------------------
##                            departure_delay
## ------------------------------------------------
## month                         -0.274***
##                                (0.005)
##
## day_of_week                   -0.234***
##                                (0.008)
##
## scheduled_dep_hour            0.849***
##                                (0.003)
##
## airlineAS                     -7.570***
##                                (0.098)
##
## airlineB6                     2.030***
##                                (0.084)
##
## airlineDL                     -1.550***
```

```
##                                 (0.059)
##
## airlineEV                      0.423***
##                                 (0.068)
##
## airlineF9                      3.590***
##                                 (0.129)
##
## airlineHA                      -6.940***
##                                 (0.143)
##
## airlineMQ                      1.830***
##                                 (0.084)
##
## airlineNK                      6.440***
##                                 (0.116)
##
## airlineOO                      -0.553***
##                                 (0.067)
##
## airlineUA                      5.090***
##                                 (0.068)
##
## airlineUS                      -3.680***
##                                 (0.095)
##
## airlineVX                      -0.385**
##                                 (0.155)
##
## airlineWN                      1.720***
##                                 (0.055)
##
## air_time                       0.015***
##                                 (0.001)
##
## distance                       0.002***
##                                (0.0002)
##
## air_time:distance             -0.00001***
##                                (0.00000)
##
## Constant                       -1.830***
##                                 (0.094)
##
## ------------------------------------------------
## Observations                   5,713,997
## R2                               0.019
## Adjusted R2                      0.019
## Residual Std. Error      36.500 (df = 5713977)
## F Statistic         5,894.000*** (df = 19; 5713977)
## ================================================
## Note:                   *p<0.1; **p<0.05; ***p<0.01
```

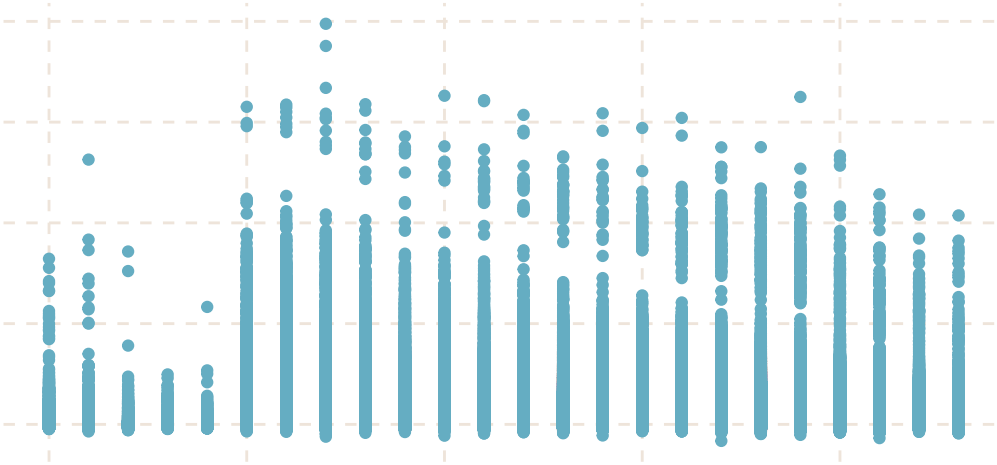
```
# Output residual and RMSE
summary(reg_full)$r.squared
```

```
## [1] 0.0192
```

```
rmse(reg_full, flight_merged)
```

```
## [1] 36.5
```

```
# Plot
ggplot(reg_full, aes(x = scheduled_dep_hour, y = departure_delay)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)
```



```
#c.Train dataset
sample <- sample.split(flight_merged$month, SplitRatio = 0.6)
train  <- subset(flight_merged, sample == TRUE)
test   <- subset(flight_merged, sample == FALSE)
# Run regression again
reg_train <- lm(departure_delay ~ month + day_of_week + scheduled_dep_hour
              + airline + air_time*distance, data = train)
#View result
summary(reg_train)
```

```
##
## Call:
## lm(formula = departure_delay ~ month + day_of_week + scheduled_dep_hour +
##     airline + air_time * distance, data = train)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -84.2  -14.2   -9.0   -1.5 1983.1
```

```
##
## Coefficients:
##                         Estimate   Std. Error t value             Pr(>|t|)
## (Intercept)        -1.845846647  0.120613240  -15.30 < 0.0000000000000002 ***
## month              -0.273887581  0.005908083  -46.36 < 0.0000000000000002 ***
## day_of_week        -0.237159420  0.009928899  -23.89 < 0.0000000000000002 ***
## scheduled_dep_hour  0.858076203  0.004098693  209.35 < 0.0000000000000002 ***
## airlineAS          -7.547173299  0.127094095  -59.38 < 0.0000000000000002 ***
## airlineB6           2.047515058  0.107781750   19.00 < 0.0000000000000002 ***
## airlineDL          -1.668819598  0.075835138  -22.01 < 0.0000000000000002 ***
## airlineEV           0.336909941  0.087623122    3.84              0.00012 ***
## airlineF9           3.472403335  0.166971292   20.80 < 0.0000000000000002 ***
## airlineHA          -7.019641747  0.185181943  -37.91 < 0.0000000000000002 ***
## airlineMQ           1.784163526  0.108429104   16.45 < 0.0000000000000002 ***
## airlineNK           6.433122928  0.149794223   42.95 < 0.0000000000000002 ***
## airlineOO          -0.697810862  0.086401174   -8.08  0.00000000000000067 ***
## airlineUA           5.058836703  0.087014275   58.14 < 0.0000000000000002 ***
## airlineUS          -3.730458163  0.122860987  -30.36 < 0.0000000000000002 ***
## airlineVX          -0.370207910  0.199449917   -1.86              0.06343 .
## airlineWN           1.679598287  0.071030420   23.65 < 0.0000000000000002 ***
## air_time            0.016012762  0.001657571    9.66 < 0.0000000000000002 ***
## distance            0.001930331  0.000204116    9.46 < 0.0000000000000002 ***
## air_time:distance  -0.000008442  0.000000321  -26.31 < 0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.5 on 3428380 degrees of freedom
## Multiple R-squared:  0.0195, Adjusted R-squared:  0.0195
## F-statistic: 3.59e+03 on 19 and 3428380 DF,  p-value: <0.0000000000000002
```

```
#d.Logistic Regression
reg_lo <- glm(departure_delayed_dummy ~ month + day_of_week + scheduled_dep_hour
  + airline + air_time * distance, data = train, family = "binomial")
summary(reg_lo)
```

```
##
## Call:
## glm(formula = departure_delayed_dummy ~ month + day_of_week +
##     scheduled_dep_hour + airline + air_time * distance, family = "binomial",
##     data = train)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.253  -0.681  -0.551  -0.426   2.912
##
## Coefficients:
##                         Estimate    Std. Error z value             Pr(>|z|)
## (Intercept)        -2.7535197677  0.0090696596 -303.60 < 0.0000000000000002 ***
## month              -0.0317035304  0.0004261289  -74.40 < 0.0000000000000002 ***
## day_of_week        -0.0144005756  0.0007167636  -20.09 < 0.0000000000000002 ***
## scheduled_dep_hour  0.0928804174  0.0003038617  305.67 < 0.0000000000000002 ***
## airlineAS          -0.6027135253  0.0109703822  -54.94 < 0.0000000000000002 ***
## airlineB6           0.2442864043  0.0074589046   32.75 < 0.0000000000000002 ***
## airlineDL          -0.2665110966  0.0057986927  -45.96 < 0.0000000000000002 ***
```

```
## airlineEV           0.0467786457  0.0064426039    7.26     0.00000000000038 ***
## airlineF9           0.2532233315  0.0112627608   22.48 < 0.0000000000000002 ***
## airlineHA          -0.8681680435  0.0188871656  -45.97 < 0.0000000000000002 ***
## airlineMQ           0.2226246522  0.0076999041   28.91 < 0.0000000000000002 ***
## airlineNK           0.5305501111  0.0096871032   54.77 < 0.0000000000000002 ***
## airlineOO          -0.0189056428  0.0064101294   -2.95               0.0032 **
## airlineUA           0.3687149226  0.0060279589   61.17 < 0.0000000000000002 ***
## airlineUS          -0.2930136955  0.0094393907  -31.04 < 0.0000000000000002 ***
## airlineVX           0.0149142335  0.0144880980    1.03               0.3033
## airlineWN           0.2539847517  0.0050965544   49.83 < 0.0000000000000002 ***
## air_time            0.0007441562  0.0001177650    6.32     0.00000000026328 ***
## distance            0.0002666698  0.0000146312   18.23 < 0.0000000000000002 ***
## air_time:distance  -0.0000008955  0.0000000249  -35.99 < 0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3271445  on 3428399  degrees of freedom
## Residual deviance: 3133525  on 3428380  degrees of freedom
## AIC: 3133565
##
## Number of Fisher Scoring iterations: 5
```

```
# Prediction
#Plot
fitted.results <- predict(reg_lo, test, type = "response")
fitted.results <- ifelse(fitted.results > 0.5, 1, 0)
# Accuracy
confusionMatrix(test$departure_delayed_dummy, fitted.results)
```

```
##         0      1
## 0 1865107 420201
## 1     215     74
```

```
sensitivity(test$departure_delayed_dummy, fitted.results) # true positive rate
```

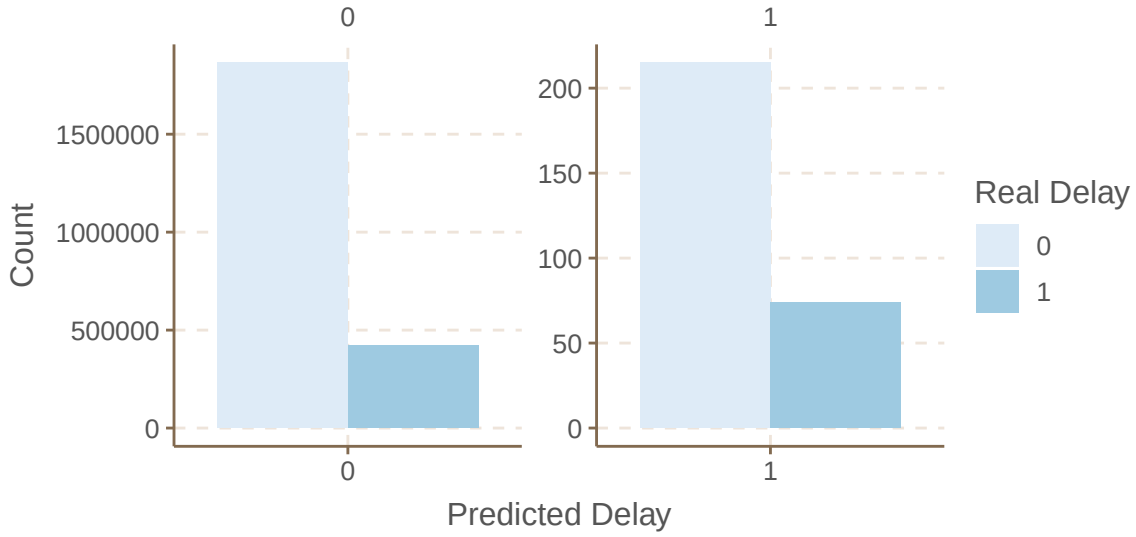
```
## [1] 0.000176
```

```
specificity(test$departure_delayed_dummy, fitted.results) # true negative rate
```

```
## [1] 1
```

```
# Plot
prediction <- matrix(nrow = 2285597, ncol = 2)
prediction[, 1] <- fitted.results
prediction[, 2] <- test$departure_delayed_dummy

as.data.frame(prediction) %>%
  group_by(V1, V2) %>%
  summarise(count = n()) %>%
```

```
ggplot(aes(factor(V1), count)) +
geom_col(aes(fill = factor(V2)),
         position = "dodge") +
scale_fill_brewer("Real Delay") +
labs(x = "Predicted Delay",
     y = "Count") +
facet_wrap(~V1,
           scales = "free")
```



Explanation

- The coefficient of scheduled departure hour on departure delay is 0.841 and is significant given alpha equals 0.01, which means given all other conditions the same, a flight 1 hour late will have an expected increase at 0.841 min for departure delay.
- The coefficient of month and day of the week are both negative, which means the later in a month or in a day of the week, the more likely we need to expect a higher delay for departure. The result aligns with the exploratory analysis in part 3.
- For airline, given all others equal, we will expect 6.520 mins increase on departure delay on average if taking a flight with *Spirit Air Lines*. Other airlines which has an obvious negative effect on departure delay includes *Frontier Airlines Inc.*, *JetBlue Airways*, *United Air Lines Inc.*. Most of the airlines are cheap airlines, so it makes sense that flight delay is more likely to happen. In contrast, a flight with *Hawaiian Airlines Inc.* will have an expected decrease at 8.5 mins for departure delay.
- However, the R square is **0.019**, which performs very poor. It means the departure delay time for flights in US in 2015 does not significantly related to departure hour, although it's significant at 1% level. As shown in the plot, the prediction model cannot predict the real departure delay time well.
- With the logistic regression however, the overall accuracy turns out to be good, which is 81.6%. However, sensitivity is super low at the level of 0.03%, the prediction is not good at predicting delay as well.