Data Analysis and Visulation Homework #01

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

```
## -- Attaching packages --------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts ------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

A. Sort flights to find the most delayed flights. Find the flights that left earliest

```
# most delayed flights
flights %>%
  arrange(desc(dep_delay))
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_time sched_dep_time dep_delay arr_time
##    <int> <int> <int>    <int>          <int>     <dbl>    <int>
##  1  2013     1     9      641            900      1301     1242
##  2  2013     6    15     1432           1935      1137     1607
##  3  2013     1    10     1121           1635      1126     1239
##  4  2013     9    20     1139           1845      1014     1457
##  5  2013     7    22      845           1600      1005     1044
##  6  2013     4    10     1100           1900       960     1342
##  7  2013     3    17     2321            810       911      135
##  8  2013     6    27      959           1900       899     1236
##  9  2013     7    22     2257            759       898      121
## 10  2013    12     5      756           1700       896     1058
## # ... with 336,766 more rows, and 12 more variables: sched_arr_time <int>,
## #   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>
```

```
# Find the flights that left earliest
flights %>%
  arrange(dep_delay)
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_time sched_dep_time dep_delay arr_time
##    <int> <int> <int>    <int>          <int>     <dbl>    <int>
##  1  2013    12     7     2040           2123       -43       40
##  2  2013     2     3     2022           2055       -33     2240
##  3  2013    11    10     1408           1440       -32     1549
```

```
##  4  2013     1    11     1900           1930       -30     2233
##  5  2013     1    29     1703           1730       -27     1947
##  6  2013     8     9      729            755       -26     1002
##  7  2013    10    23     1907           1932       -25     2143
##  8  2013     3    30     2030           2055       -25     2213
##  9  2013     3     2     1431           1455       -24     1601
## 10  2013     5     5      934            958       -24     1225
## # ... with 336,766 more rows, and 12 more variables: sched_arr_time <int>,
## #   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>
```

B. Which flights traveled the longest? Which traveled the shortest?

```
# Travelled the longest
flights %>%
  arrange(desc(air_time))
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_time sched_dep_time dep_delay arr_time
##    <int> <int> <int>    <int>          <int>     <dbl>    <int>
##  1  2013     3    17     1337           1335         2     1937
##  2  2013     2     6      853            900        -7     1542
##  3  2013     3    15     1001           1000         1     1551
##  4  2013     3    17     1006           1000         6     1607
##  5  2013     3    16     1001           1000         1     1544
##  6  2013     2     5      900            900         0     1555
##  7  2013    11    12      936            930         6     1630
##  8  2013     3    14      958           1000        -2     1542
##  9  2013    11    20     1006           1000         6     1639
## 10  2013     3    15     1342           1335         7     1924
## # ... with 336,766 more rows, and 12 more variables: sched_arr_time <int>,
## #   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>
```

```
# Travelled the shortest
flights %>%
  arrange(air_time)
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_time sched_dep_time dep_delay arr_time
##    <int> <int> <int>    <int>          <int>     <dbl>    <int>
##  1  2013     1    16     1355           1315        40     1442
##  2  2013     4    13      537            527        10      622
##  3  2013    12     6      922            851        31     1021
##  4  2013     2     3     2153           2129        24     2247
##  5  2013     2     5     1303           1315       -12     1342
##  6  2013     2    12     2123           2130        -7     2211
##  7  2013     3     2     1450           1500       -10     1547
##  8  2013     3     8     2026           1935        51     2131
##  9  2013     3    18     1456           1329        87     1533
## 10  2013     3    19     2226           2145        41     2305
```

```
## # ... with 336,766 more rows, and 12 more variables: sched_arr_time <int>,
## #   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>
```

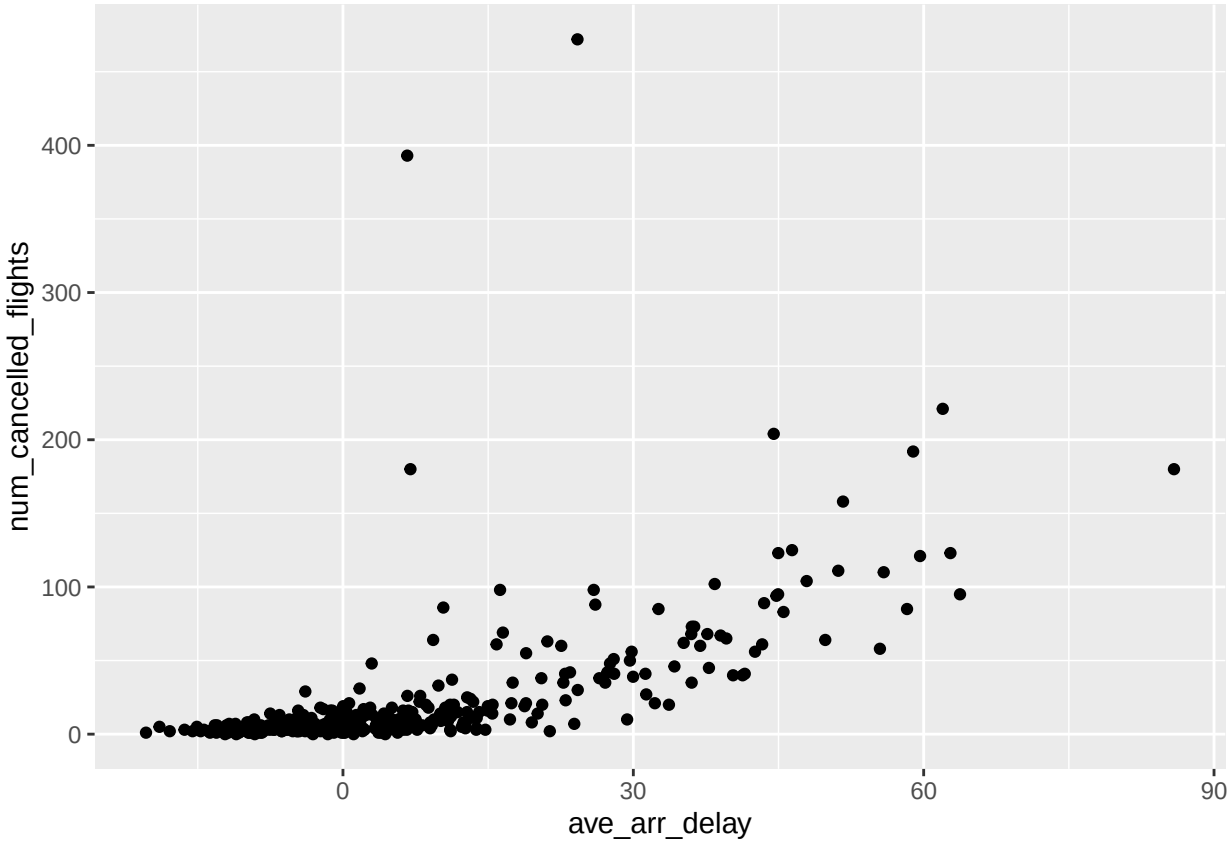
C. Find flights that were delayed by at least an hour, but made up over 30 minutes in flight

```
flights %>%
  filter(dep_delay > 60, arr_delay - dep_delay > 30) %>%
  arrange(desc(dep_delay))
```

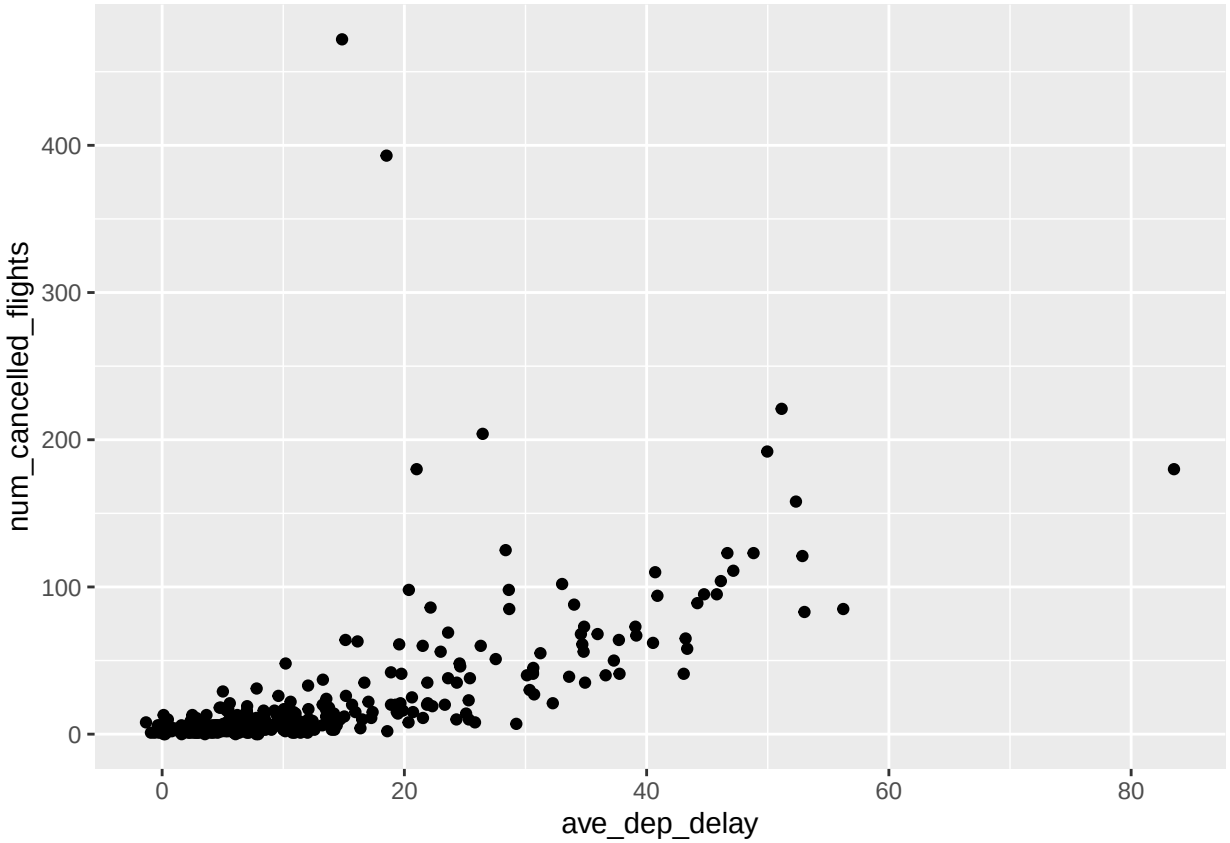
```
## # A tibble: 1,924 x 19
##     year month   day dep_time sched_dep_time dep_delay arr_time
##    <int> <int> <int>    <int>          <int>     <dbl>    <int>
##  1  2013    12    14      830           1845       825     1210
##  2  2013     7     7     2059           1030       629      106
##  3  2013     7    21     1555            615       580     1955
##  4  2013     7    27     1456            600       536     1649
##  5  2013     6    13     2242           1515       447      232
##  6  2013     3    18     2239           1516       443      139
##  7  2013     6    30     1842           1125       437     2229
##  8  2013     6     7     2359           1700       419      201
##  9  2013     6    13     1627            959       388     1815
## 10  2013     6    13     2127           1459       388      125
## # ... with 1,914 more rows, and 12 more variables: sched_arr_time <int>,
## #   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>
```

D. Look at the number of cancelled flights per day. Is there a pattern? Is the proportion of cancelled flights related to the average delay?

```
ave_delayed_flights = flights %>%
  group_by(year,month,day) %>%
  summarize(ave_dep_delay=mean(dep_delay,na.rm=TRUE),
            ave_arr_delay=mean(arr_delay,na.rm=TRUE))
#
cancelled_flights = flights %>%
  group_by(year,month,day) %>%
  summarize(num_cancelled_flights=sum(is.na(dep_delay)))
#
flights01 = merge(ave_delayed_flights,cancelled_flights)
#
ggplot(data=flights01) +
  geom_point(mapping=aes(x=ave_arr_delay,y=num_cancelled_flights))
```



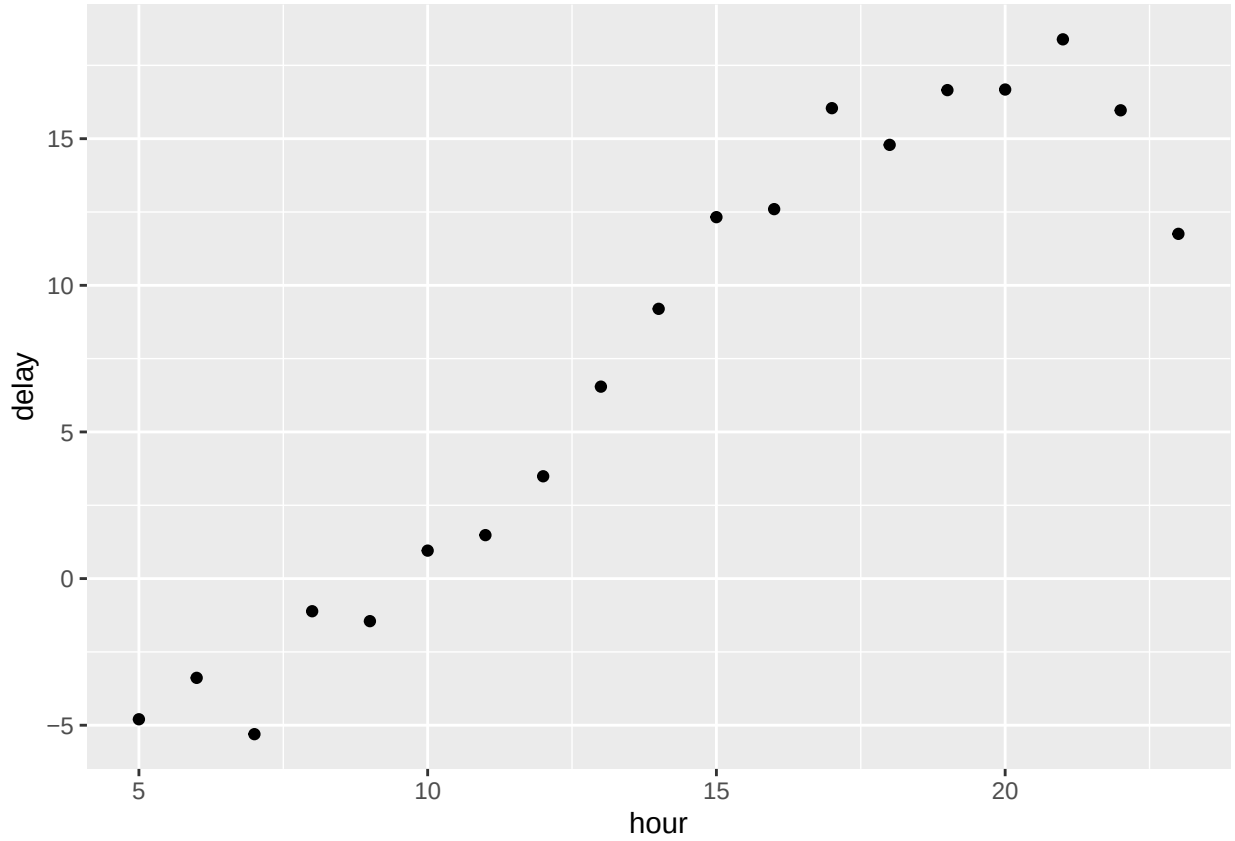

```
ggplot(data=flights01) +
  geom_point(mapping=aes(x=ave_dep_delay,y=num_cancelled_flights))
```



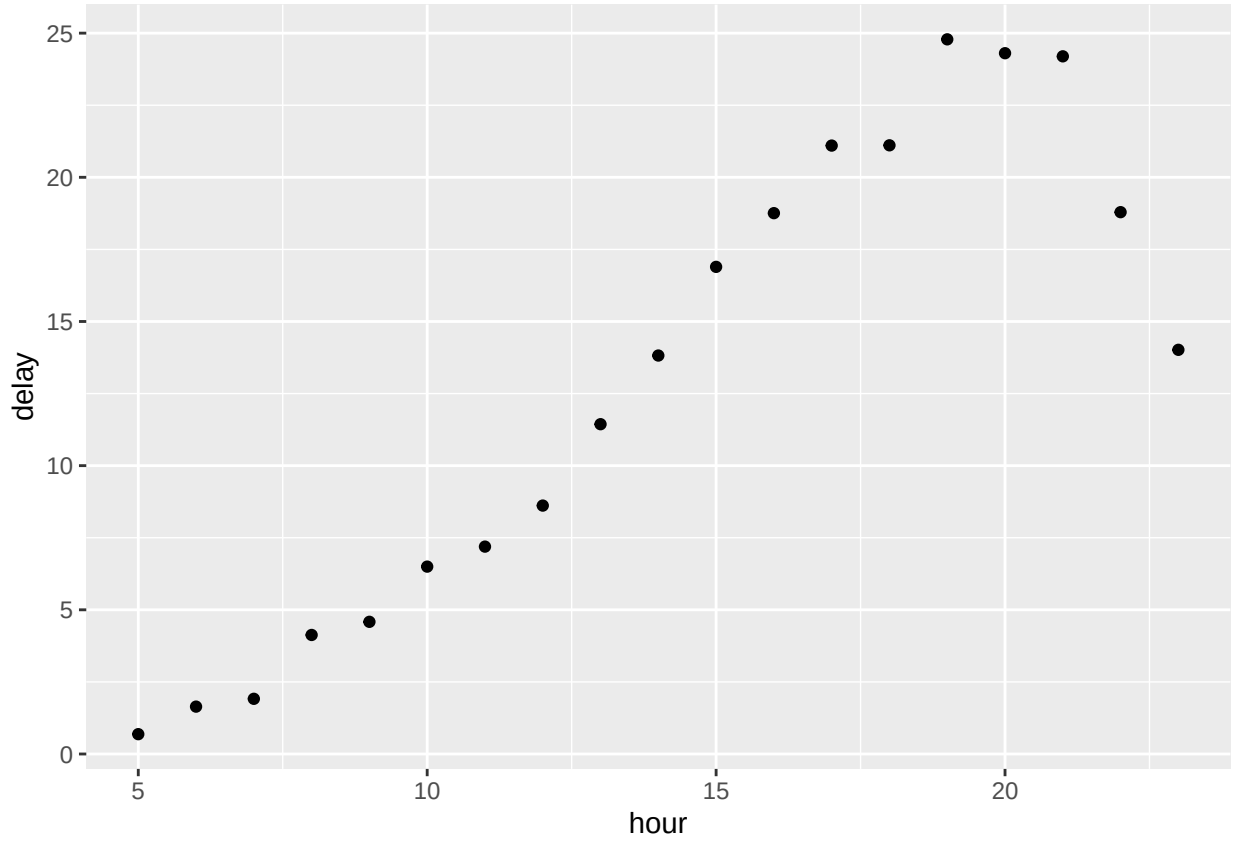
Yes, the number of cancelled flights are related to the average arrival deplay and average departure deplay from the graphs. We can say the average deparure time and average arrival deplay increase, the number of cancelled flights increases.

E. What time of the day should you fly if you want to avoid delays as much as possible.

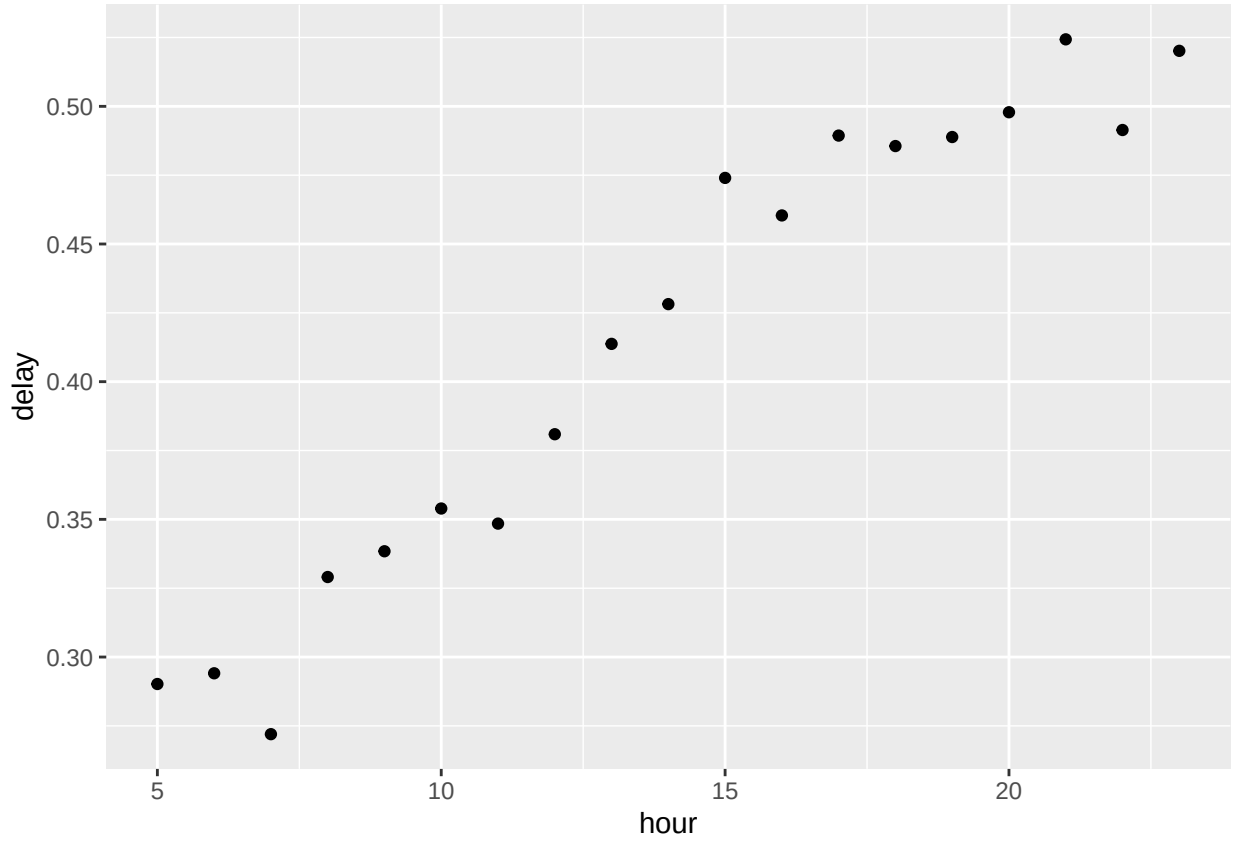
```
flights %>%
  group_by(hour) %>%
  filter(!is.na(arr_delay)) %>%
  summarise(delay = mean(arr_delay)) %>%
  ggplot(aes(x = hour, y = delay)) + geom_point()
```



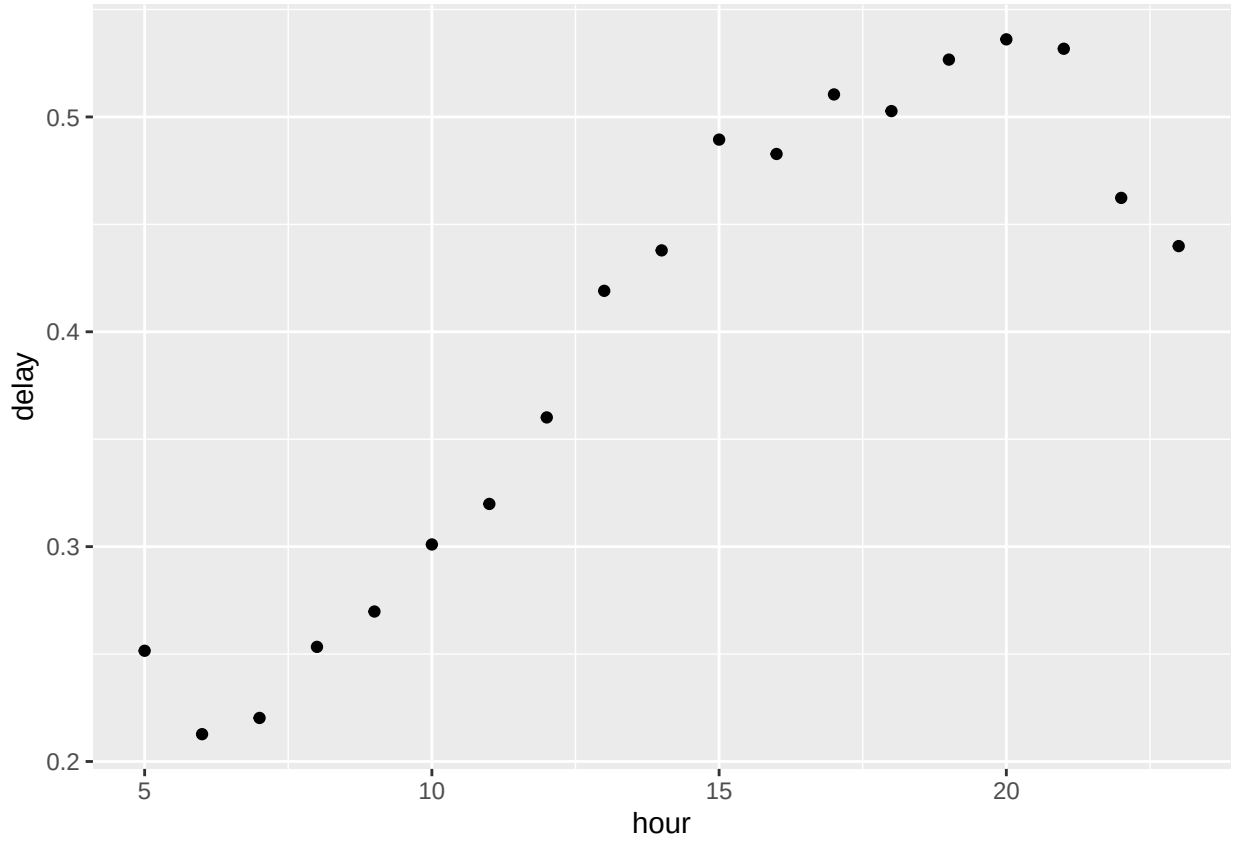

```
#
flights %>%
  group_by(hour) %>%
  filter(!is.na(dep_delay)) %>%
  summarise(delay = mean(dep_delay)) %>%
  ggplot(aes(x = hour, y = delay)) + geom_point()
```




```
flights %>%
  group_by(hour) %>%
  filter(!is.na(arr_delay)) %>%
  summarise(delay = mean(arr_delay>0)) %>%
  ggplot(aes(x = hour, y = delay)) + geom_point()
```




```
#
flights %>%
  group_by(hour) %>%
  filter(!is.na(dep_delay)) %>%
  summarise(delay = mean( dep_delay>0)) %>%
  ggplot(aes(x = hour, y = delay)) + geom_point()
```



We have four graphs of arrival delay and departure delay. From those four graphs, we can find if people take plane in the morning, between 5am - 10am, they will aviods delays as much as possible.

F. For each destination, compute the total minutes of delay. For each flight, compute the proportion of the total delay for its destination.

```
# For each destination, compute total minutes of delay
flights %>%
  filter(arr_delay>0) %>%
  group_by(dest) %>%
  summarize(total_minutes=sum(arr_delay>0)) %>%
  arrange(desc(total_minutes))
```

```
## # A tibble: 103 x 2
##    dest  total_minutes
##    <chr>         <int>
##  1 ATL            7946
##  2 ORD            6198
##  3 LAX            5967
##  4 CLT            5838
##  5 MCO            5545
##  6 FLL            5212
##  7 SFO            4941
##  8 BOS            4743
##  9 DCA            4003
## 10 MIA            3855
## # ... with 93 more rows
```

```r
# For each fligh, compute the proportion of the total delay for its destination.
flights %>%
  filter(arr_delay>0) %>%
  group_by(dest) %>%
  mutate(arr_delay_prop = arr_delay / sum(arr_delay)) %>%
  select(carrier, flight, tailnum, origin, dest, arr_delay_prop) %>%
  arrange(desc(arr_delay_prop))
```

```
## # A tibble: 133,004 x 6
## # Groups:   dest [103]
##    carrier flight tailnum origin dest  arr_delay_prop
##    <chr>    <int> <chr>   <chr>  <chr>          <dbl>
##  1 UA         887 N528UA  EWR    ANC            0.629
##  2 UA         385 N806UA  EWR    MTJ            0.594
##  3 VX          55 N839VA  JFK    PSP            0.472
##  4 EV        5383 N398CA  LGA    SBN            0.424
##  5 EV        5383 N761ND  LGA    SBN            0.4
##  6 UA         441 N817UA  EWR    HDN            0.361
##  7 UA         568 N436UA  EWR    BZN            0.314
##  8 UA        1506 N16701  EWR    JAC            0.283
##  9 UA         355 N474UA  EWR    HDN            0.269
## 10 EV        5325 N611QX  LGA    CHO            0.241
## # ... with 132,994 more rows
```