NYC Flights 2013

John Cruz

2023-02-11

Required Libraries

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

Using dplyr

Row Operations

filter()

```
# Flights that departed on January 1
flights |>
  filter(month == 1 & day == 1)
```

```
## # A tibble: 842 x 19
##     year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
##    <int> <int> <int>    <int>      <int>   <dbl>   <int>   <int>   <dbl> <chr>
##  1  2013     1     1      517        515       2     830     819      11 UA
##  2  2013     1     1      533        529       4     850     830      20 UA
##  3  2013     1     1      542        540       2     923     850      33 AA
##  4  2013     1     1      544        545      -1    1004    1022     -18 B6
##  5  2013     1     1      554        600      -6     812     837     -25 DL
##  6  2013     1     1      554        558      -4     740     728      12 UA
##  7  2013     1     1      555        600      -5     913     854      19 B6
##  8  2013     1     1      557        600      -3     709     723     -14 EV
##  9  2013     1     1      557        600      -3     838     846      -8 B6
## 10  2013     1     1      558        600      -2     753     745       8 AA
## # ... with 832 more rows, 9 more variables: flight <int>, tailnum <chr>,
## #   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dttm>, and abbreviated variable names
## #   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
## #   5: arr_delay
```

```
# Flights that departed in January or February
flights |>
  filter(month %in% c(1, 2))
```

```
## # A tibble: 51,955 x 19
##     year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
##    <int> <int> <int>    <int>      <int>   <dbl>   <int>   <int>   <dbl> <chr>
##  1  2013     1     1      517        515       2     830     819      11 UA
##  2  2013     1     1      533        529       4     850     830      20 UA
##  3  2013     1     1      542        540       2     923     850      33 AA
##  4  2013     1     1      544        545      -1    1004    1022     -18 B6
##  5  2013     1     1      554        600      -6     812     837     -25 DL
##  6  2013     1     1      554        558      -4     740     728      12 UA
##  7  2013     1     1      555        600      -5     913     854      19 B6
##  8  2013     1     1      557        600      -3     709     723     -14 EV
##  9  2013     1     1      557        600      -3     838     846      -8 B6
## 10  2013     1     1      558        600      -2     753     745       8 AA
## # ... with 51,945 more rows, 9 more variables: flight <int>, tailnum <chr>,
## #   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dttm>, and abbreviated variable names
## #   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
## #   5: arr_delay
```

arrange()

```
flights |>
  arrange(desc(dep_delay))
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
##    <int> <int> <int>    <int>      <int>   <dbl>   <int>   <int>   <dbl> <chr>
##  1  2013     1     9      641        900    1301    1242    1530    1272 HA
##  2  2013     6    15     1432       1935    1137    1607    2120    1127 MQ
##  3  2013     1    10     1121       1635    1126    1239    1810    1109 MQ
##  4  2013     9    20     1139       1845    1014    1457    2210    1007 AA
##  5  2013     7    22      845       1600    1005    1044    1815     989 MQ
##  6  2013     4    10     1100       1900     960    1342    2211     931 DL
##  7  2013     3    17     2321        810     911     135    1020     915 DL
##  8  2013     6    27      959       1900     899    1236    2226     850 DL
##  9  2013     7    22     2257        759     898     121    1026     895 DL
## 10  2013    12     5      756       1700     896    1058    2020     878 AA
## # ... with 336,766 more rows, 9 more variables: flight <int>, tailnum <chr>,
## #   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dttm>, and abbreviated variable names
## #   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
## #   5: arr_delay
```

distinct()

```
flights |>
  distinct(origin, dest)
```

```
## # A tibble: 224 x 2
##    origin dest 
##    <chr>  <chr>
##  1 EWR    IAH  
##  2 LGA    IAH  
##  3 JFK    MIA  
##  4 JFK    BQN  
##  5 LGA    ATL  
##  6 EWR    ORD  
##  7 EWR    FLL  
##  8 LGA    IAD  
##  9 JFK    MCO  
## 10 LGA    ORD  
## # ... with 214 more rows
```

Column Operations

mutate()

- *.before* or *.after* “Determine new columns placement in data frame.”
- *.keep* “Control which variables are kept. (‘used’ argument keeps the inputs from your calculations)”

```
flights |>
  mutate(
    gain = dep_delay - arr_delay,
    hours = air_time / 60,
    gain_per_hour = gain / hours,
    .keep = "used"
  )
```

```
## # A tibble: 336,776 x 6
##    dep_delay arr_delay air_time  gain hours gain_per_hour
##        <dbl>     <dbl>    <dbl> <dbl> <dbl>         <dbl>
##  1         2        11      227    -9 3.78          -2.38
##  2         4        20      227   -16 3.78          -4.23
##  3         2        33      160   -31 2.67         -11.6 
##  4        -1       -18      183    17 3.05           5.57
##  5        -6       -25      116    19 1.93           9.83
##  6        -4        12      150   -16 2.5           -6.4 
##  7        -5        19      158   -24 2.63          -9.11
##  8        -3       -14       53    11 0.883         12.5 
##  9        -3        -8      140     5 2.33           2.14
## 10        -2         8      138   -10 2.3           -4.35
## # ... with 336,766 more rows
```

select()

- starts_with("abc"): matches names that begin with "abc".
- ends_with("xyz"): matches names that end with "xyz".
- contains("ijk"): matches names that contain "ijk".
- num_range("x", 1:3): matches x1, x2 and x3.

```r
# Select columns by name
flights |>
  select(year, month, day)
```

```
## # A tibble: 336,776 x 3
##     year month   day
##    <int> <int> <int>
##  1  2013     1     1
##  2  2013     1     1
##  3  2013     1     1
##  4  2013     1     1
##  5  2013     1     1
##  6  2013     1     1
##  7  2013     1     1
##  8  2013     1     1
##  9  2013     1     1
## 10  2013     1     1
## # ... with 336,766 more rows
```

```r
# Select all columns between year and day (inclusive)
flights |>
  select(year:day)
```

```
## # A tibble: 336,776 x 3
##     year month   day
##    <int> <int> <int>
##  1  2013     1     1
##  2  2013     1     1
##  3  2013     1     1
##  4  2013     1     1
##  5  2013     1     1
##  6  2013     1     1
##  7  2013     1     1
##  8  2013     1     1
##  9  2013     1     1
## 10  2013     1     1
## # ... with 336,766 more rows
```

```r
# Select all columns except those from year to day (inclusive)
flights |>
  select(!year:day)
```

```
## # A tibble: 336,776 x 16
##    dep_t~1 sched~2 dep_d~3 arr_t~4 sched~5 arr_d~6 carrier flight tailnum origin
##      <int>   <int>   <dbl>   <int>   <int>   <dbl> <chr>    <int> <chr>   <chr>
##  1     517     515       2     830     819      11 UA        1545 N14228  EWR
##  2     533     529       4     850     830      20 UA        1714 N24211  LGA
##  3     542     540       2     923     850      33 AA        1141 N619AA  JFK
##  4     544     545      -1    1004    1022     -18 B6         725 N804JB  JFK
##  5     554     600      -6     812     837     -25 DL         461 N668DN  LGA
##  6     554     558      -4     740     728      12 UA        1696 N39463  EWR
##  7     555     600      -5     913     854      19 B6         507 N516JB  EWR
##  8     557     600      -3     709     723     -14 EV        5708 N829AS  LGA
##  9     557     600      -3     838     846      -8 B6          79 N593JB  JFK
## 10     558     600      -2     753     745       8 AA         301 N3ALAA  LGA
## # ... with 336,766 more rows, 6 more variables: dest <chr>, air_time <dbl>,
## #   distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>, and abbreviated
## #   variable names 1: dep_time, 2: sched_dep_time, 3: dep_delay, 4: arr_time,
## #   5: sched_arr_time, 6: arr_delay
```

```
# Select all columns that are characters
flights |>
  select(where(is.character))
```

```
## # A tibble: 336,776 x 4
##    carrier tailnum origin dest
##    <chr>   <chr>   <chr>  <chr>
##  1 UA      N14228  EWR    IAH
##  2 UA      N24211  LGA    IAH
##  3 AA      N619AA  JFK    MIA
##  4 B6      N804JB  JFK    BQN
##  5 DL      N668DN  LGA    ATL
##  6 UA      N39463  EWR    ORD
##  7 B6      N516JB  EWR    FLL
##  8 EV      N829AS  LGA    IAD
##  9 B6      N593JB  JFK    MCO
## 10 AA      N3ALAA  LGA    ORD
## # ... with 336,766 more rows
```

rename()

- alternative for many columns use *janitor::clean_names()*

```
flights |>
  rename(tail_num = tailnum)
```

```
## # A tibble: 336,776 x 19
##     year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
##    <int> <int> <int>    <int>      <int>   <dbl>   <int>   <int>   <dbl> <chr>
##  1  2013     1     1      517        515       2     830     819      11 UA
##  2  2013     1     1      533        529       4     850     830      20 UA
##  3  2013     1     1      542        540       2     923     850      33 AA
##  4  2013     1     1      544        545      -1    1004    1022     -18 B6
##  5  2013     1     1      554        600      -6     812     837     -25 DL
##  6  2013     1     1      554        558      -4     740     728      12 UA
##  7  2013     1     1      555        600      -5     913     854      19 B6
##  8  2013     1     1      557        600      -3     709     723     -14 EV
##  9  2013     1     1      557         600      -3      838      846       -8 B6
## 10  2013     1     1      558         600      -2      753      745        8 AA
## # ... with 336,766 more rows, 9 more variables: flight <int>, tail_num <chr>,
## #   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dttm>, and abbreviated variable names
## #   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
## #   5: arr_delay
```

relocate()

```
flights |>
  relocate(year:dep_time, .after = time_hour)
```

```
## # A tibble: 336,776 x 19
##    sched_d~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier flight tailnum origin dest
##        <int>   <dbl>   <int>   <int>   <dbl> <chr>    <int> <chr>   <chr>  <chr>
##  1       515       2     830     819      11 UA        1545 N14228  EWR    IAH
##  2       529       4     850     830      20 UA        1714 N24211  LGA    IAH
##  3       540       2     923     850      33 AA        1141 N619AA  JFK    MIA
##  4       545      -1    1004    1022     -18 B6         725 N804JB  JFK    BQN
##  5       600      -6     812     837     -25 DL         461 N668DN  LGA    ATL
##  6       558      -4     740     728      12 UA        1696 N39463  EWR    ORD
##  7       600      -5     913     854      19 B6         507 N516JB  EWR    FLL
##  8       600      -3     709     723     -14 EV        5708 N829AS  LGA    IAD
##  9       600      -3     838     846      -8 B6          79 N593JB  JFK    MCO
## 10       600      -2     753     745       8 AA         301 N3ALAA  LGA    ORD
## # ... with 336,766 more rows, 9 more variables: air_time <dbl>, distance <dbl>,
## #   hour <dbl>, minute <dbl>, time_hour <dttm>, year <int>, month <int>,
## #   day <int>, dep_time <int>, and abbreviated variable names
## #   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
## #   5: arr_delay
```

group_by() & summarize()

```
flights |>
  group_by(month) |>
  summarize(
    delay = mean(dep_delay, na.rm = TRUE),
    count = n()
  )
```

```
## # A tibble: 12 x 3
##    month delay count
##    <int> <dbl> <int>
##  1     1 10.0  27004
##  2     2 10.8  24951
##  3     3 13.2  28834
##  4     4 13.9  28330
##  5     5 13.0  28796
##  6     6 20.8  28243
##  7      7 21.7  29425
##  8      8 12.6  29327
##  9      9  6.72 27574
## 10     10  6.24 28889
## 11     11  5.44 27268
## 12     12 16.6  28135
```

summarize based on tibble and multiple *group_by()*

When you are modifying a tibble that was created by multiple conditions in the previous group_by, you have to use .groups

```
daily <- flights |>
  group_by(year, month, day)

daily_flights <- daily |>
  summarize(
    n = n(),
    .groups = "drop_last"
  )
```

ungroup()

```
daily |>
  ungroup() |>
  summarize(
    delay = mean(dep_delay, na.rm = TRUE),
    flights = n()
  )
```

```
## # A tibble: 1 x 2
##   delay flights
##   <dbl>   <int>
## 1  12.6  336776
```

*slice_* functions

Instead of *n =*, you can use *prop = 0.1* to select 10% of the rows in each group.

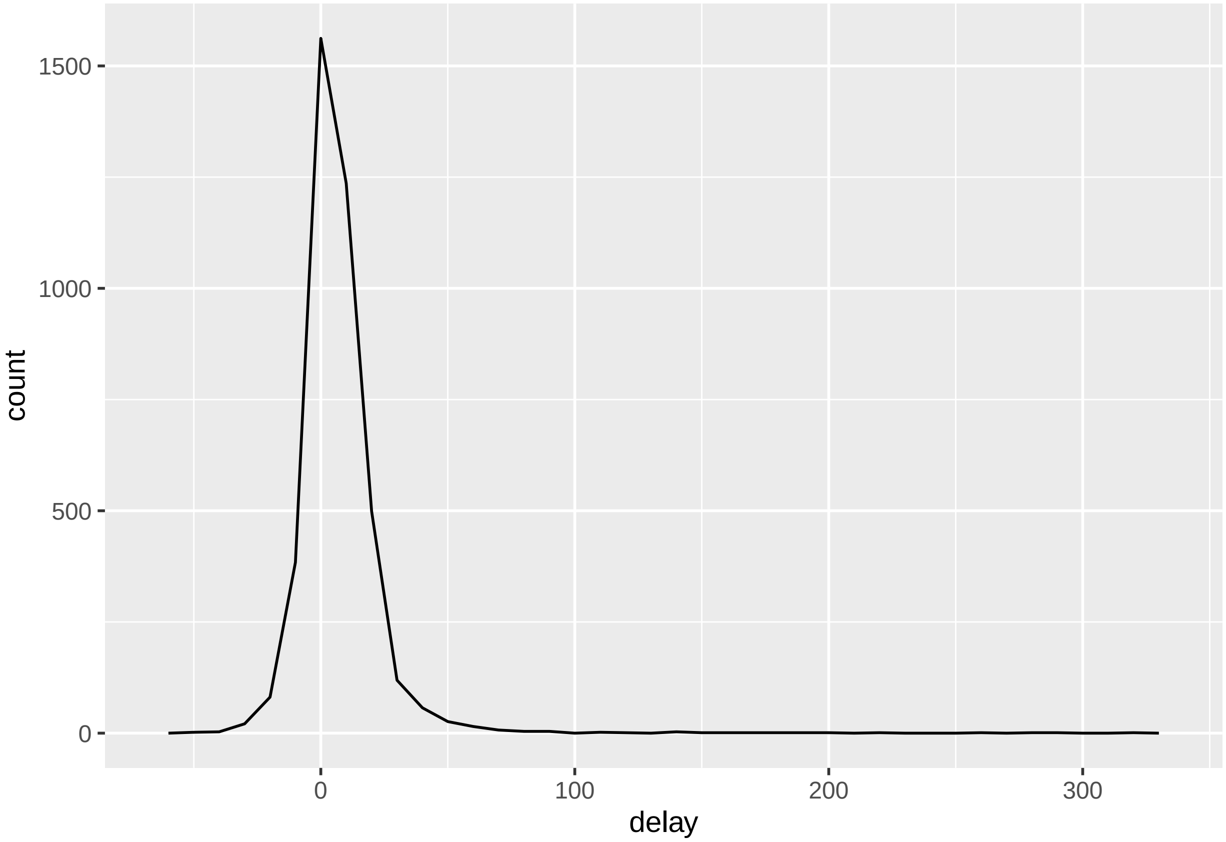
- df |> slice_head(n = 1) takes the first row from each group.
- df |> slice_tail(n = 1) takes the last row in each group.
- df |> slice_min(x, n = 1) takes the row with the smallest value of x.
- df |> slice_max(x, n = 1) takes the row with the largest value of x.
- df |> slice_sample(n = 1) takes one random row.

```
flights |>
  group_by(dest) |>
  slice_max(arr_delay, prop = 0.05)
```

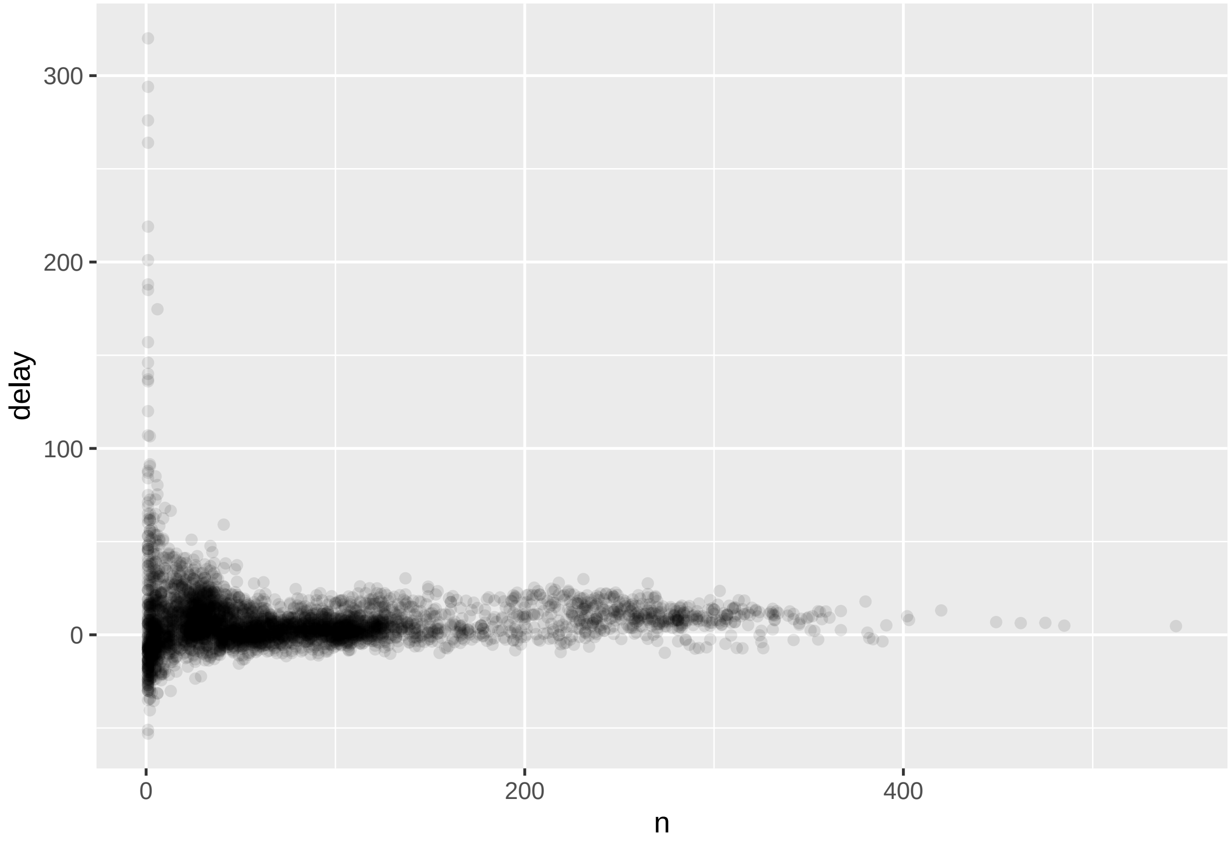
```
## # A tibble: 16,931 x 19
## # Groups:   dest [97]
##     year month   day dep_time sched_de~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
##    <int> <int> <int>    <int>      <int>   <dbl>   <int>   <int>   <dbl> <chr>
##  1  2013     7    22     2145       2007      98     132    2259     153 B6
##  2  2013    12    14     2223       2001     142     133    2304     149 B6
##  3  2013    10    15     2146       2001     105     106    2248     138 B6
##  4  2013     7    23     2206       2007     119     116    2259     137 B6
##  5  2013    12    17     2220       2001     139     120    2304     136 B6
##  6  2013     7    10     2025       2007      18     105    2259     126 B6
##  7  2013     7    30     2212       2007     125      57    2259     118 B6
##  8  2013     7    28     2038       2007      31      56    2259     117 B6
##  9  2013    12     8     2049       2001      48      58    2304     114 B6
## 10  2013     9     2     2212       2007     125      48    2259     109 B6
## # ... with 16,921 more rows, 9 more variables: flight <int>, tailnum <chr>,
## #   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dttm>, and abbreviated variable names
## #   1: sched_dep_time, 2: dep_delay, 3: arr_time, 4: sched_arr_time,
## #   5: arr_delay
```

```r
delays <- flights |>
  filter(!is.na(arr_delay), !is.na(tailnum)) |>
  group_by(tailnum) |>
  summarize(
    delay = mean(arr_delay, na.rm = TRUE),
    n = n()
  )

ggplot(delays, aes(x = delay)) +
  geom_freqpoly(binwidth = 10)
```

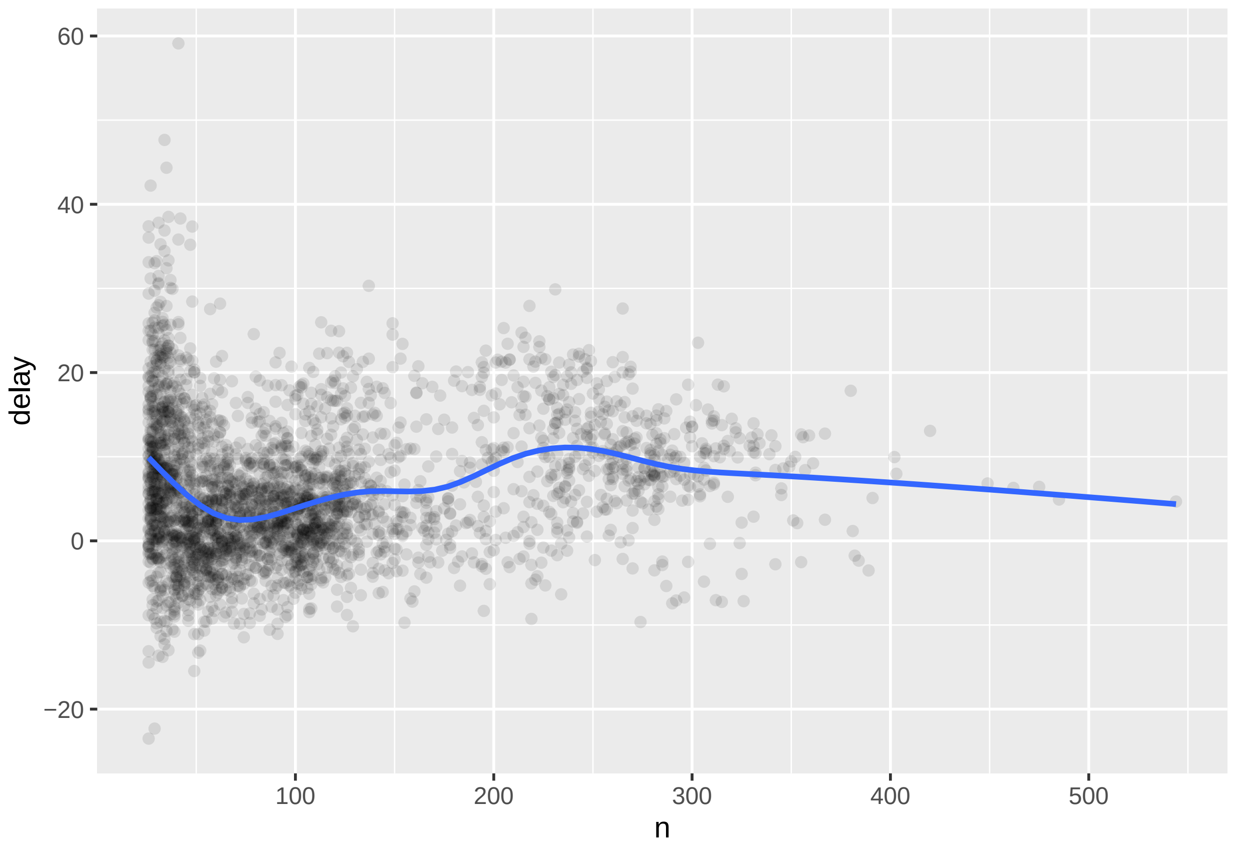


```r
ggplot(delays, aes(x = n, y = delay)) +
  geom_point(alpha = 1/10)
```



```
delays |>
  filter(n > 25) |>
  ggplot(aes(x = n, y = delay)) +
  geom_point(alpha = 1/10) +
  geom_smooth(se = FALSE)
```

```
## ‘geom_smooth()‘ using method = ’gam’ and formula = ’y ~ s(x, bs = "cs")’
```


60
40
20
0
−20
delay
100
200
300
400
500
n