R3.1_wrangling.R

aditya mhaske

2023-04-30

```
library(dplyr)
library(ggplot2)
library(nycflights13)
```

```
## alaska_flights <- flights %>%
##   filter(carrier == "AS")
```

```
## portland_flights <- flights %>%
##   filter(dest == "PDX")
## View(portland_flights)
```

```
## btv_sea_flights_fall <- flights %>%
##   filter(origin == "JFK" & (dest == "BTV" | dest == "SEA") & month >= 10)
## View(btv_sea_flights_fall)
```

```
## btv_sea_flights_fall <- flights %>%
##   filter(origin == "JFK", (dest == "BTV" | dest == "SEA"), month >= 10)
## View(btv_sea_flights_fall)
```

```
## not_BTV_SEA <- flights %>%
##   filter(!(dest == "BTV" | dest == "SEA"))
## View(not_BTV_SEA)
```

```
## flights %>% filter(!dest == "BTV" | dest == "SEA")
```

```
## many_airports <- flights %>%
##   filter(dest == "SEA" | dest == "SFO" | dest == "PDX" |
##          dest == "BTV" | dest == "BDL")
```

```
## many_airports <- flights %>%
##   filter(dest %in% c("SEA", "SFO", "PDX", "BTV", "BDL"))
## View(many_airports)
```

```
summary_temp <- weather %>%
  summarize(mean = mean(temp), std_dev = sd(temp))
summary_temp
```

```
## # A tibble: 1 x 2
##    mean std_dev
##   <dbl>   <dbl>
## 1    NA      NA
```

```
summary_temp <- weather %>%
  summarize(mean = mean(temp, na.rm = TRUE),
            std_dev = sd(temp, na.rm = TRUE))
summary_temp
```

```
## # A tibble: 1 x 2
##    mean std_dev
##   <dbl>   <dbl>
## 1  55.3    17.8
```

```
## summary_temp <- weather %>%
##   summarize(mean = mean(temp, na.rm = TRUE)) %>%
##   summarize(std_dev = sd(temp, na.rm = TRUE))
```

```
summary_monthly_temp <- weather %>%
  group_by(month) %>%
  summarize(mean = mean(temp, na.rm = TRUE),
            std_dev = sd(temp, na.rm = TRUE))
summary_monthly_temp
```

```
## # A tibble: 12 x 3
##    month  mean std_dev
##    <int> <dbl>   <dbl>
##  1     1  35.6   10.2
##  2     2  34.3    6.98
##  3     3  39.9    6.25
##  4     4  51.7    8.79
##  5     5  61.8    9.68
##  6     6  72.2    7.55
##  7     7  80.1    7.12
##  8     8  74.5    5.19
##  9     9  67.4    8.47
## 10    10  60.1    8.85
## 11    11  45.0   10.4
## 12    12  38.4    9.98
```

```
diamonds
```

```
## # A tibble: 53,940 x 10
##    carat cut       color clarity depth table price     x     y     z
##    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
##  1  0.23 Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43
##  2  0.21 Premium   E     SI1      59.8    61   326  3.89  3.84  2.31
##  3  0.23 Good      E     VS1      56.9    65   327  4.05  4.07  2.31
##  4  0.29 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63
##  5  0.31 Good      J     SI2      63.3    58   335  4.34  4.35  2.75
##  6  0.24 Very Good J     VVS2     62.8    57   336  3.94  3.96  2.48
##  7  0.24 Very Good I     VVS1     62.3    57   336  3.95  3.98  2.47
##  8  0.26 Very Good H     SI1      61.9    55   337  4.07  4.11  2.53
##  9  0.22 Fair      E     VS2      65.1    61   337  3.87  3.78  2.49
## 10  0.23 Very Good H     VS1      59.4    61   338  4     4.05  2.39
## # ... with 53,930 more rows
```

```r
diamonds %>%
  group_by(cut)
```

```
## # A tibble: 53,940 x 10
## # Groups:   cut [5]
##    carat cut       color clarity depth table price     x     y     z
##    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
##  1  0.23 Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43
##  2  0.21 Premium   E     SI1      59.8    61   326  3.89  3.84  2.31
##  3  0.23 Good      E     VS1      56.9    65   327  4.05  4.07  2.31
##  4  0.29 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63
##  5  0.31 Good      J     SI2      63.3    58   335  4.34  4.35  2.75
##  6  0.24 Very Good J     VVS2     62.8    57   336  3.94  3.96  2.48
##  7  0.24 Very Good I     VVS1     62.3    57   336  3.95  3.98  2.47
##  8  0.26 Very Good H     SI1      61.9    55   337  4.07  4.11  2.53
##  9  0.22 Fair      E     VS2      65.1    61   337  3.87  3.78  2.49
## 10  0.23 Very Good H     VS1      59.4    61   338  4     4.05  2.39
## # ... with 53,930 more rows
```

```r
diamonds %>%
  group_by(cut) %>%
  summarize(avg_price = mean(price))
```

```
## # A tibble: 5 x 2
##   cut       avg_price
##   <ord>         <dbl>
## 1 Fair          4359.
## 2 Good          3929.
## 3 Very Good     3982.
## 4 Premium       4584.
## 5 Ideal         3458.
```

```r
diamonds %>%
  group_by(cut) %>%
  ungroup()
```

```
## # A tibble: 53,940 x 10
##    carat cut       color clarity depth table price     x     y     z
##    <dbl> <ord>     <ord> <ord>   <dbl> <dbl> <int> <dbl> <dbl> <dbl>
##  1  0.23 Ideal     E     SI2      61.5    55   326  3.95  3.98  2.43
##  2  0.21 Premium   E     SI1      59.8    61   326  3.89  3.84  2.31
##  3  0.23 Good      E     VS1      56.9    65   327  4.05  4.07  2.31
##  4  0.29 Premium   I     VS2      62.4    58   334  4.2   4.23  2.63
##  5  0.31 Good      J     SI2      63.3    58   335  4.34  4.35  2.75
##  6  0.24 Very Good J     VVS2     62.8    57   336  3.94  3.96  2.48
##  7  0.24 Very Good I     VVS1     62.3    57   336  3.95  3.98  2.47
##  8  0.26 Very Good H     SI1      61.9    55   337  4.07  4.11  2.53
##  9  0.22 Fair      E     VS2      65.1    61   337  3.87  3.78  2.49
## 10  0.23 Very Good H     VS1      59.4    61   338  4     4.05  2.39
## # ... with 53,930 more rows
```

```
by_origin <- flights %>%
  group_by(origin) %>%
  summarize(count = n())
by_origin
```

```
## # A tibble: 3 x 2
##   origin  count
##   <chr>   <int>
## 1 EWR    120835
## 2 JFK    111279
## 3 LGA    104662
```

```
by_origin_monthly <- flights %>%
  group_by(origin, month) %>%
  summarize(count = n())
```

```
## ‘summarise()‘ has grouped output by ’origin’. You can override using the
## ‘.groups‘ argument.
```

```
by_origin_monthly
```

```
## # A tibble: 36 x 3
## # Groups:   origin [3]
##    origin month count
##    <chr>  <int> <int>
##  1 EWR        1  9893
##  2 EWR        2  9107
##  3 EWR        3 10420
##  4 EWR        4 10531
##  5 EWR        5 10592
##  6 EWR        6 10175
##  7 EWR        7 10475
##  8 EWR        8 10359
##  9 EWR        9  9550
## 10 EWR       10 10104
## # ... with 26 more rows
```

```
by_origin_monthly_incorrect <- flights %>%
  group_by(origin) %>%
  group_by(month) %>%
  summarize(count = n())
by_origin_monthly_incorrect
```

```
## # A tibble: 12 x 2
##    month count
##    <int> <int>
##  1     1 27004
##  2     2 24951
##  3     3 28834
##  4     4 28330
##  5     5 28796
## 6      6 28243
## 7      7 29425
## 8      8 29327
## 9      9 27574
## 10    10 28889
## 11    11 27268
## 12    12 28135
```

```
weather <- weather %>%
  mutate(temp_in_C = (temp - 32) / 1.8)
```

```
summary_monthly_temp <- weather %>%
  group_by(month) %>%
  summarize(mean_temp_in_F = mean(temp, na.rm = TRUE),
            mean_temp_in_C = mean(temp_in_C, na.rm = TRUE))
summary_monthly_temp
```

```
## # A tibble: 12 x 3
##    month mean_temp_in_F mean_temp_in_C
##    <int>          <dbl>          <dbl>
##  1     1           35.6           2.02
##  2     2           34.3           1.26
##  3     3           39.9           4.38
##  4     4           51.7          11.0
##  5     5           61.8          16.6
##  6     6           72.2          22.3
##  7     7           80.1          26.7
##  8     8           74.5          23.6
##  9     9           67.4          19.7
## 10    10           60.1          15.6
## 11    11           45.0           7.22
## 12    12           38.4           3.58
```

```
flights <- flights %>%
  mutate(gain = dep_delay - arr_delay)
```

```
gain_summary <- flights %>%
  summarize(
    min = min(gain, na.rm = TRUE),
    q1 = quantile(gain, 0.25, na.rm = TRUE),
    median = quantile(gain, 0.5, na.rm = TRUE),
    q3 = quantile(gain, 0.75, na.rm = TRUE),
    max = max(gain, na.rm = TRUE),
    mean = mean(gain, na.rm = TRUE),
    sd = sd(gain, na.rm = TRUE),
    missing = sum(is.na(gain))
  )
gain_summary
```

```
## # A tibble: 1 x 8
##     min    q1 median    q3   max  mean    sd missing
##   <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>   <int>
## 1  -196    -3      7    17   109  5.66  18.0    9430
```

```
ggplot(data = flights, mapping = aes(x = gain)) +
  geom_histogram(color = "white", bins = 20)
```

```
## Warning: Removed 9430 rows containing non-finite values (‘stat_bin()‘).
```

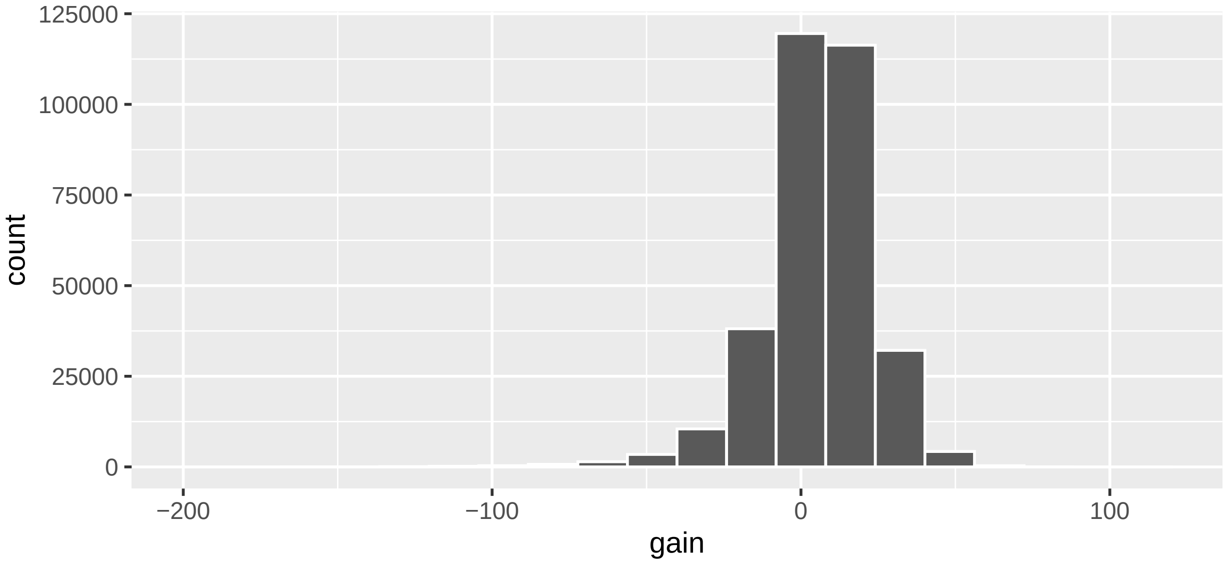

Figure 1: Histogram of gain variable.

```
flights <- flights %>%
  mutate(
    gain = dep_delay - arr_delay,
    hours = air_time / 60,
    gain_per_hour = gain / hours
  )
```

```
freq_dest <- flights %>%
  group_by(dest) %>%
  summarize(num_flights = n())
freq_dest
```

```
## # A tibble: 105 x 2
##    dest  num_flights
##    <chr>       <int>
##  1 ABQ           254
##  2 ACK           265
##  3 ALB           439
##  4 ANC             8
##  5 ATL         17215
##  6 AUS          2439
##  7 AVL           275
##  8 BDL           443
##  9 BGR           375
## 10 BHM           297
## # ... with 95 more rows
```

```
freq_dest %>%
  arrange(num_flights)
```

```
## # A tibble: 105 x 2
##    dest  num_flights
##    <chr>       <int>
##  1 LEX             1
##  2 LGA             1
##  3 ANC             8
##  4 SBN            10
##  5 HDN            15
##  6 MTJ            15
##  7 EYW            17
##  8 PSP            19
##  9 JAC            25
## 10 BZN            36
## # ... with 95 more rows
```

```
freq_dest %>%
  arrange(desc(num_flights))
```

```
## # A tibble: 105 x 2
##    dest  num_flights
##    <chr>       <int>
##  1 ORD         17283
##  2 ATL         17215
##  3 LAX         16174
##  4 BOS         15508
##  5 MCO         14082
##  6 CLT         14064
##  7 SFO         13331
##  8 FLL         12055
##  9 MIA         11728
## 10 DCA          9705
## # ... with 95 more rows
```

```
## View(airlines)
```

```
## flights_joined <- flights %>%
##   inner_join(airlines, by = "carrier")
## View(flights)
## View(flights_joined)
```

```
## View(airports)
```

```
## flights_with_airport_names <- flights %>%
##   inner_join(airports, by = c("dest" = "faa"))
## View(flights_with_airport_names)
```

```
named_dests <- flights %>%
  group_by(dest) %>%
  summarize(num_flights = n()) %>%
  arrange(desc(num_flights)) %>%
  inner_join(airports, by = c("dest" = "faa")) %>%
  rename(airport_name = name)
named_dests
```

```
## # A tibble: 101 x 9
##    dest  num_flights airport_name             lat    lon   alt    tz dst   tzone
##    <chr>       <int> <chr>                  <dbl>  <dbl> <dbl> <dbl> <chr> <chr>
##  1 ORD         17283 Chicago Ohare Intl      42.0  -87.9   668    -6 A     Amer~
##  2 ATL         17215 Hartsfield Jackson At~  33.6  -84.4  1026    -5 A     Amer~
##  3 LAX         16174 Los Angeles Intl        33.9 -118.    126    -8 A     Amer~
##  4 BOS         15508 General Edward Lawren~  42.4  -71.0    19    -5 A     Amer~
##  5 MCO         14082 Orlando Intl            28.4  -81.3    96    -5 A     Amer~
##  6 CLT         14064 Charlotte Douglas Intl  35.2  -80.9   748    -5 A     Amer~
##  7 SFO         13331 San Francisco Intl      37.6 -122.     13    -8 A     Amer~
##  8 FLL         12055 Fort Lauderdale Holly~  26.1  -80.2     9    -5 A     Amer~
##  9 MIA         11728 Miami Intl              25.8  -80.3     8    -5 A     Amer~
## 10 DCA          9705 Ronald Reagan Washing~  38.9  -77.0    15    -5 A     Amer~
## # ... with 91 more rows
```

```
## flights_weather_joined <- flights %>%
##   inner_join(weather, by = c("year", "month", "day", "hour", "origin"))
## View(flights_weather_joined)
```

```
## joined_flights <- flights %>%
##   inner_join(airlines, by = "carrier")
## View(joined_flights)
```

```
## glimpse(flights)
```

```
## flights %>%
##   select(carrier, flight)
```

```
## flights_no_year <- flights %>% select(-year)
```

```
## flight_arr_times <- flights %>% select(month:day, arr_time:sched_arr_time)
## flight_arr_times
```

```
## flights_reorder <- flights %>%
##   select(year, month, day, hour, minute, time_hour, everything())
## glimpse(flights_reorder)
```

```
## flights %>% select(starts_with("a"))
## flights %>% select(ends_with("delay"))
## flights %>% select(contains("time"))
## flights_time_new <- flights %>%
##   select(dep_time, arr_time) %>%
##   rename(departure_time = dep_time, arrival_time = arr_time)
## glimpse(flights_time_new)
```

```
## named_dests %>% top_n(n = 10, wt = num_flights)
```

```
## named_dests  %>%
##   top_n(n = 10, wt = num_flights) %>%
##   arrange(desc(num_flights))
```