

Transforming Data in R

Pramod Duvvuri

3/29/2019

Data Manipulation in R

```
library(nycflights13) # Data to be used
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.2.1 --
## v ggplot2 3.1.0      v purrr  0.3.2
## v tibble  2.1.1      v dplyr  0.8.0.1
## v tidyr   0.8.3      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()
```

```
nycflights13::flights
```

```
## # 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     1     517             515           2     830
## 2  2013     1     1     533             529           4     850
## 3  2013     1     1     542             540           2     923
## 4  2013     1     1     544             545          -1    1004
## 5  2013     1     1     554             600          -6     812
## 6  2013     1     1     554             558          -4     740
## 7  2013     1     1     555             600          -5     913
## 8  2013     1     1     557             600          -3     709
## 9  2013     1     1     557             600          -3     838
## 10 2013     1     1     558             600          -2     753
## # ... 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 <dtm>
```

```
attach(flights)
?flights
```

Basics of dplyr

The below functions will be covered in this file.

1. `filter()`
2. `arrange()`
3. `select()`
4. `mutate()`

```
5. summarise()
6. group_by()
```

filter()

```
# Select all flights on January 1st
filter(flights, month == 1, day == 1)
```

```
## # A tibble: 842 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time
##   <int> <int> <int>   <int>         <int>       <dbl>   <int>
## 1  2013     1     1     517           515         2     830
## 2  2013     1     1     533           529         4     850
## 3  2013     1     1     542           540         2     923
## 4  2013     1     1     544           545        -1    1004
## 5  2013     1     1     554           600        -6     812
## 6  2013     1     1     554           558        -4     740
## 7  2013     1     1     555           600        -5     913
## 8  2013     1     1     557           600        -3     709
## 9  2013     1     1     557           600        -3     838
##10  2013     1     1     558           600        -2     753
## # ... with 832 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 <dtm>

jan_1_data <- filter(flights, month == 1, day == 1) # Can use view(jan_1_data) to see the data

# To assing the data to another variable and print to console
(dec25_data <- filter(flights, month == 12, day == 25))
```

```
## # A tibble: 719 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time
##   <int> <int> <int>   <int>         <int>       <dbl>   <int>
## 1  2013    12    25     456           500        -4     649
## 2  2013    12    25     524           515         9     805
## 3  2013    12    25     542           540         2     832
## 4  2013    12    25     546           550        -4    1022
## 5  2013    12    25     556           600        -4     730
## 6  2013    12    25     557           600        -3     743
## 7  2013    12    25     557           600        -3     818
## 8  2013    12    25     559           600        -1     855
## 9  2013    12    25     559           600        -1     849
##10  2013    12    25     600           600         0     850
## # ... with 709 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 <dtm>
```

```
filter(flights, month = 4) # throws an error (must use ==)
```

```
sqrt(2) ^ 2 == 2
```

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

```

1 / 49 * 49 == 1

## [1] FALSE

# Use near() for approximation
near(sqrt(2) ^ 2, 2)

## [1] TRUE

near(1 / 49 * 49, 1)

## [1] TRUE

# Using filter() with logical operators
filter(flights, month == 4 | month == 6)

## # A tibble: 56,573 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>
## 1  2013     4     1     454             500          -6     636
## 2  2013     4     1     509             515          -6     743
## 3  2013     4     1     526             530          -4     812
## 4  2013     4     1     534             540          -6     833
## 5  2013     4     1     542             545          -3     914
## 6  2013     4     1     543             545          -2     921
## 7  2013     4     1     551             600          -9     748
## 8  2013     4     1     552             600          -8     641
## 9  2013     4     1     553             600          -7     725
## 10 2013     4     1     554             600          -6     752
## # ... with 56,563 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 <dtm>

# Alternate way to write the above line of code
apr_jun_data <- filter(flights, month %in% c(4,7))
nrow(apr_jun_data)

## [1] 57755

# Missing Values
NA == NA

## [1] NA

# Adding some context
x <- NA # Age of Person 1
y <- NA # Age of Person 2
x == y # Compare

## [1] NA

is.na(x)

## [1] TRUE

# Handling Missing Values using filter()
df <- tibble(x = c(19,27,32,NA))
filter(df, x > 1) # NA excluded

```

```
## # A tibble: 3 x 1
##       x
##   <dbl>
## 1    19
## 2    27
## 3    32
```

```
filter(df, is.na(x) | x > 1)
```

```
## # A tibble: 4 x 1
##       x
##   <dbl>
## 1    19
## 2    27
## 3    32
## 4    NA
```

Exercises

Q1. Find all flights that

Q1.1 Had an arrival delay of two or more hours

```
filter(flights, arr_delay >= 120)
```

```
## # A tibble: 10,200 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>
## 1  2013     1     1     811             630          101    1047
## 2  2013     1     1     848             1835         853    1001
## 3  2013     1     1     957             733          144    1056
## 4  2013     1     1    1114             900          134    1447
## 5  2013     1     1    1505            1310          115    1638
## 6  2013     1     1    1525            1340          105    1831
## 7  2013     1     1    1549            1445           64    1912
## 8  2013     1     1    1558            1359          119    1718
## 9  2013     1     1    1732            1630           62    2028
## 10 2013     1     1    1803            1620          103    2008
## # ... with 10,190 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 <dtm>
```

Q1.2 Flew to Houston (IAH or HOU)

```
filter(flights, dest %in% c('IAH', 'HOU'))
```

```
## # A tibble: 9,313 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>
## 1  2013     1     1     517             515           2     830
## 2  2013     1     1     533             529           4     850
## 3  2013     1     1     623             627          -4     933
## 4  2013     1     1     728             732          -4    1041
## 5  2013     1     1     739             739           0    1104
## 6  2013     1     1     908             908           0    1228
```

```
## 7 2013 1 1 1028 1026 2 1350
## 8 2013 1 1 1044 1045 -1 1352
## 9 2013 1 1 1114 900 134 1447
## 10 2013 1 1 1205 1200 5 1503
## # ... with 9,303 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 <dtm>
```

Q1.3 Were operated by United, American, or Delta

```
filter(flights, carrier %in% c('AA','DL','UA'))
```

```
## # A tibble: 139,504 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>
## 1 2013     1     1     517           515           2     830
## 2 2013     1     1     533           529           4     850
## 3 2013     1     1     542           540           2     923
## 4 2013     1     1     554           600          -6     812
## 5 2013     1     1     554           558          -4     740
## 6 2013     1     1     558           600          -2     753
## 7 2013     1     1     558           600          -2     924
## 8 2013     1     1     558           600          -2     923
## 9 2013     1     1     559           600          -1     941
## 10 2013     1     1     559           600          -1     854
## # ... with 139,494 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 <dtm>
```

Q1.4 Departed in summer (July, August, and September)

```
filter(flights, month %in% c(7,8,9))
```

```
## # A tibble: 86,326 x 19
##   year month   day dep_time sched_dep_time dep_delay arr_time
##   <int> <int> <int>   <int>         <int>         <dbl>   <int>
## 1 2013     7     1       1           2029          212     236
## 2 2013     7     1       2           2359           3     344
## 3 2013     7     1      29           2245          104     151
## 4 2013     7     1      43           2130          193     322
## 5 2013     7     1      44           2150          174     300
## 6 2013     7     1      46           2051          235     304
## 7 2013     7     1      48           2001          287     308
## 8 2013     7     1      58           2155          183     335
## 9 2013     7     1     100           2146          194     327
## 10 2013     7     1     100           2245          135     337
## # ... with 86,316 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 <dtm>
```

Q1.5 Arrived more than two hours late, but didn't leave late

```
filter(flights, arr_delay > 120, dep_delay <= 0)
```

```
## # A tibble: 29 x 19
```

```
##      year month   day dep_time sched_dep_time dep_delay arr_time
##      <int> <int> <int>   <int>         <int>      <dbl>   <int>
##  1  2013     1    27    1419           1420        -1     1754
##  2  2013    10     7    1350           1350         0     1736
##  3  2013    10     7    1357           1359        -2     1858
##  4  2013    10    16     657            700        -3     1258
##  5  2013    11     1     658            700        -2     1329
##  6  2013     3    18    1844           1847        -3         39
##  7  2013     4    17    1635           1640        -5     2049
##  8  2013     4    18     558            600        -2     1149
##  9  2013     4    18     655            700        -5     1213
## 10  2013     5    22    1827           1830        -3     2217
## # ... with 19 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 <dtm>
```

Q1.6 Were delayed by at least an hour, but made up over 30 minutes in flight

```
filter(flights, dep_delay > 60, arr_delay < 30)
```

```
## # A tibble: 181 x 19
##      year month   day dep_time sched_dep_time dep_delay arr_time
##      <int> <int> <int>   <int>         <int>      <dbl>   <int>
##  1  2013     1     3    1850           1745        65     2148
##  2  2013     1     3    1950           1845        65     2228
##  3  2013     1     6    1019            900        79     1558
##  4  2013     1     7    1543           1430        73     1758
##  5  2013     1    12    1706           1600        66     1949
##  6  2013     1    12    1953           1845        68     2154
##  7  2013     1    19    1456           1355        61     1636
##  8  2013     1    21    1531           1430        61     1843
##  9  2013     1    21    1648           1545        63     1939
## 10  2013    10    10    1938           1835        63     2158
## # ... with 171 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 <dtm>
```

Q1.7 Departed between midnight and 6am (inclusive)

```
filter(flights, dep_time >= 0000 & dep_time <= 0600)
```

```
## # A tibble: 9,344 x 19
##      year month   day dep_time sched_dep_time dep_delay arr_time
##      <int> <int> <int>   <int>         <int>      <dbl>   <int>
##  1  2013     1     1     517            515         2       830
##  2  2013     1     1     533            529         4       850
##  3  2013     1     1     542            540         2       923
##  4  2013     1     1     544            545        -1      1004
##  5  2013     1     1     554            600        -6       812
##  6  2013     1     1     554            558        -4       740
##  7  2013     1     1     555            600        -5       913
##  8  2013     1     1     557            600        -3       709
##  9  2013     1     1     557            600        -3       838
## 10  2013     1     1     558            600        -2       753
```

```
## # ... with 9,334 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 <dtm>
```

```
# Using between() for filtering
sum(between(flights$dep_time, 0000, 0600), na.rm = TRUE)
```

```
## [1] 9344
```

arrange()

```
arrange(flights, year, month, day)
```

```
## # 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   1     517           515         2     830
## 2  2013     1   1     533           529         4     850
## 3  2013     1   1     542           540         2     923
## 4  2013     1   1     544           545        -1    1004
## 5  2013     1   1     554           600        -6     812
## 6  2013     1   1     554           558        -4     740
## 7  2013     1   1     555           600        -5     913
## 8  2013     1   1     557           600        -3     709
## 9  2013     1   1     557           600        -3     838
## 10 2013     1   1     558           600        -2     753
## # ... 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 <dtm>
```

```
# Descending Order
arrange(flights, 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 <dtm>
```

```
# Missing Values (stored in the end)
df <- tibble(x = c(5, 2, NA))
```

```
arrange(df, x)
```

```
## # A tibble: 3 x 1
##       x
##   <dbl>
## 1     2
## 2     5
## 3    NA
```

```
arrange(df, desc(x))
```

```
## # A tibble: 3 x 1
##       x
##   <dbl>
## 1     5
## 2     2
## 3    NA
```

Exercises

Q1. How could you use `arrange()` to sort all missing values to the start? (Hint: use `is.na()`).

```
arrange(df, desc(is.na(x)))
```

```
## # A tibble: 3 x 1
##       x
##   <dbl>
## 1    NA
## 2     5
## 3     2
```

Q2. Sort flights to find the most delayed flights. Find the flights that left earliest.

```
arrange(flights, desc(arr_delay), 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     7    22    2257             759       898     121
## 9  2013    12     5     756            1700       896    1058
## 10 2013     5     3    1133            2055       878    1250
## # ... 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 <dtm>
```

Q3. Sort flights to find the fastest flights.

```
arrange(flights, arr_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     5     7    1715           1729        -14    1944
## 2  2013     5    20     719           735        -16     951
## 3  2013     5     2    1947           1949         -2    2209
## 4  2013     5     6    1826           1830         -4    2045
## 5  2013     5     4    1816           1820         -4    2017
## 6  2013     5     2    1926           1929         -3    2157
## 7  2013     5     6    1753           1755         -2    2004
## 8  2013     5     7    2054           2055         -1    2317
## 9  2013     5    13     657           700         -3     908
## 10 2013     1     4    1026           1030         -4    1305
## # ... 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 <dtm>
```

Q4. Which flights travelled the longest? Which travelled the shortest?

```
arrange(flights, air_time) # shortest flights
```

```
## # 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 <dtm>
```

```
arrange(flights, desc(air_time)) # longest flights
```

```
## # 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 <dtm>
```

```
select()
```

```
# Used to select subset of columns/features
select(flights, 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
```

```
# Alternate way to get same subset of data
select(flights, 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
```

```
# All columns but a few (omit a few columns)
select(flights, -(year:day))
```

```
## # A tibble: 336,776 x 16
##   dep_time sched_dep_time dep_delay arr_time sched_arr_time arr_delay
##   <int>         <int>         <dbl>   <int>         <int>         <dbl>
## 1     517           515           2     830           819           11
## 2     533           529           4     850           830           20
## 3     542           540           2     923           850           33
## 4     544           545          -1    1004          1022          -18
## 5     554           600          -6     812           837          -25
## 6     554           558          -4     740           728           12
## 7     555           600          -5     913           854           19
```

```
## 8      557      600      -3      709      723      -14
## 9      557      600      -3      838      846      -8
## 10     558      600      -2      753      745       8
## # ... with 336,766 more rows, and 10 more variables: carrier <chr>,
## #   flight <int>, tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>,
## #   distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

```
select(flights, -c(year,month,day))
```

```
## # A tibble: 336,776 x 16
##   dep_time sched_dep_time dep_delay arr_time sched_arr_time arr_delay
##   <int>      <int>      <dbl>   <int>      <int>      <dbl>
## 1      517        515         2     830        819         11
## 2      533        529         4     850        830         20
## 3      542        540         2     923        850         33
## 4      544        545        -1    1004       1022        -18
## 5      554        600        -6     812        837        -25
## 6      554        558        -4     740        728         12
## 7      555        600        -5     913        854         19
## 8      557        600        -3     709        723        -14
## 9      557        600        -3     838        846         -8
## 10     558        600        -2     753        745          8
## # ... with 336,766 more rows, and 10 more variables: carrier <chr>,
## #   flight <int>, tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>,
## #   distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

```
select(flights, -year, -month, -day)
```

```
## # A tibble: 336,776 x 16
##   dep_time sched_dep_time dep_delay arr_time sched_arr_time arr_delay
##   <int>      <int>      <dbl>   <int>      <int>      <dbl>
## 1      517        515         2     830        819         11
## 2      533        529         4     850        830         20
## 3      542        540         2     923        850         33
## 4      544        545        -1    1004       1022        -18
## 5      554        600        -6     812        837        -25
## 6      554        558        -4     740        728         12
## 7      555        600        -5     913        854         19
## 8      557        600        -3     709        723        -14
## 9      557        600        -3     838        846         -8
## 10     558        600        -2     753        745          8
## # ... with 336,766 more rows, and 10 more variables: carrier <chr>,
## #   flight <int>, tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>,
## #   distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dtm>
```

```
rename()
```

```
# Renaming variables using rename()
rename(flights, tail_number = tailnum)
```

```
## # 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     1     517        515         2     830
```

```
## 2 2013 1 1 533 529 4 850
## 3 2013 1 1 542 540 2 923
## 4 2013 1 1 544 545 -1 1004
## 5 2013 1 1 554 600 -6 812
## 6 2013 1 1 554 558 -4 740
## 7 2013 1 1 555 600 -5 913
## 8 2013 1 1 557 600 -3 709
## 9 2013 1 1 557 600 -3 838
## 10 2013 1 1 558 600 -2 753
## # ... with 336,766 more rows, and 12 more variables: sched_arr_time <int>,
## #   arr_delay <dbl>, carrier <chr>, flight <int>, tail_number <chr>,
## #   origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>,
## #   minute <dbl>, time_hour <dtm>
```

everything()

```
# everything()
select(flights, air_time, everything()) # moves the air_time column to the beginning
```

```
## # A tibble: 336,776 x 19
##   air_time year month day dep_time sched_dep_time dep_delay arr_time
##   <dbl> <int> <int> <int> <int> <int> <dbl> <int>
## 1 227 2013 1 1 517 515 2 830
## 2 227 2013 1 1 533 529 4 850
## 3 160 2013 1 1 542 540 2 923
## 4 183 2013 1 1 544 545 -1 1004
## 5 116 2013 1 1 554 600 -6 812
## 6 150 2013 1 1 554 558 -4 740
## 7 158 2013 1 1 555 600 -5 913
## 8 53 2013 1 1 557 600 -3 709
## 9 140 2013 1 1 557 600 -3 838
## 10 138 2013 1 1 558 600 -2 753
## # ... with 336,766 more rows, and 11 more variables: sched_arr_time <int>,
## #   arr_delay <dbl>, carrier <chr>, flight <int>, tailnum <chr>,
## #   origin <chr>, dest <chr>, distance <dbl>, hour <dbl>, minute <dbl>,
## #   time_hour <dtm>
```

Exercises

Q1. Brainstorm as many ways as possible to select dep_time, dep_delay, arr_time, and arr_delay from flights

```
select(flights, c(dep_time, dep_delay, arr_time, arr_delay))
```

```
## # A tibble: 336,776 x 4
##   dep_time dep_delay arr_time arr_delay
##   <int> <dbl> <int> <dbl>
## 1 517 2 830 11
## 2 533 4 850 20
## 3 542 2 923 33
## 4 544 -1 1004 -18
## 5 554 -6 812 -25
## 6 554 -4 740 12
```

```
## 7      555      -5      913      19
## 8      557      -3      709     -14
## 9      557      -3      838      -8
## 10     558      -2      753       8
## # ... with 336,766 more rows
```

Q2. What happens if you include the name of a variable multiple times in a select() call?

```
select(flights, c("arr_time", "arr_time")) # no error
```

```
## # A tibble: 336,776 x 1
##   arr_time
##   <int>
## 1      830
## 2      850
## 3      923
## 4     1004
## 5      812
## 6      740
## 7      913
## 8      709
## 9      838
## 10     753
## # ... with 336,766 more rows
```

Q3. What does the one_of() function do? Why might it be helpful in conjunction with this vector?

```
vars <- c("year", "month", "day", "dep_delay", "arr_delay")
select(flights, one_of(vars))
```

```
## # A tibble: 336,776 x 5
##   year month   day dep_delay arr_delay
##   <int> <int> <int>     <dbl>     <dbl>
## 1  2013     1     1         2         11
## 2  2013     1     1         4         20
## 3  2013     1     1         2         33
## 4  2013     1     1        -1        -18
## 5  2013     1     1        -6        -25
## 6  2013     1     1        -4         12
## 7  2013     1     1        -5         19
## 8  2013     1     1        -3        -14
## 9  2013     1     1        -3         -8
## 10 2013     1     1        -2          8
## # ... with 336,766 more rows
```

Q4. Does the result of running the following code surprise you? How do the select helpers deal with case by default? How can you change that default?

```
select(flights, contains("TIME")) # ignore.case can be set to FALSE (default TRUE)
```

```
## # A tibble: 336,776 x 6
##   dep_time sched_dep_time arr_time sched_arr_time air_time
##   <int>         <int>     <int>         <int>     <dbl>
## 1     517           515      830           819      227
## 2     533           529      850           830      227
## 3     542           540      923           850      160
## 4     544           545     1004          1022      183
## 5     554           600      812           837      116
```

```
## 6      554      558      740      728      150
## 7      555      600      913      854      158
## 8      557      600      709      723      53
## 9      557      600      838      846      140
## 10     558      600      753      745      138
## # ... with 336,766 more rows, and 1 more variable: time_hour <dtm>
```

mutate()

```
# Using subset of flight data
flights_sml <- select(flights,
  year:day,
  ends_with("delay"),
  distance,
  air_time
)

# Adding new columns

mutate(flights_sml,
  gain = dep_delay - arr_delay,
  speed = round(distance / air_time * 60,3)
)
```

```
## # A tibble: 336,776 x 9
##   year month   day dep_delay arr_delay distance air_time   gain speed
##   <int> <int> <int>     <dbl>     <dbl>     <dbl>     <dbl> <dbl> <dbl>
## 1  2013     1     1         2         11     1400      227     -9   370.
## 2  2013     1     1         4         20     1416      227    -16   374.
## 3  2013     1     1         2         33     1089      160    -31   408.
## 4  2013     1     1        -1        -18     1576      183     17   517.
## 5  2013     1     1        -6        -25      762      116     19   394.
## 6  2013     1     1        -4         12      719      150    -16   288.
## 7  2013     1     1        -5         19     1065      158    -24   404.
## 8  2013     1     1        -3        -14      229       53     11   259.
## 9  2013     1     1        -3         -8      944      140      5   405.
## 10 2013     1     1        -2          8      733      138    -10   319.
## # ... with 336,766 more rows
```

```
# Use columns just created
mutate(flights_sml,
  gain = dep_delay - arr_delay,
  hours = air_time / 60,
  gain_per_hour = gain / hours # gain created above
)
```

```
## # A tibble: 336,776 x 10
##   year month   day dep_delay arr_delay distance air_time   gain hours
##   <int> <int> <int>     <dbl>     <dbl>     <dbl>     <dbl> <dbl> <dbl>
## 1  2013     1     1         2         11     1400      227     -9  3.78
## 2  2013     1     1         4         20     1416      227    -16  3.78
## 3  2013     1     1         2         33     1089      160    -31  2.67
## 4  2013     1     1        -1        -18     1576      183     17  3.05
## 5  2013     1     1        -6        -25      762      116     19  1.93
```

```
## 6 2013 1 1 -4 12 719 150 -16 2.5
## 7 2013 1 1 -5 19 1065 158 -24 2.63
## 8 2013 1 1 -3 -14 229 53 11 0.883
## 9 2013 1 1 -3 -8 944 140 5 2.33
## 10 2013 1 1 -2 8 733 138 -10 2.3
## # ... with 336,766 more rows, and 1 more variable: gain_per_hour <dbl>
```

transmute()

```
# Keep only newly created columns
transmute(flights_sml,
  gain = dep_delay - arr_delay,
  hours = air_time / 60,
  gain_per_hour = gain / hours
)
```

```
## # A tibble: 336,776 x 3
##   gain hours gain_per_hour
##   <dbl> <dbl>         <dbl>
## 1   -9 3.78         -2.38
## 2  -16 3.78         -4.23
## 3  -31 2.67        -11.6
## 4   17 3.05          5.57
## 5   19 1.93          9.83
## 6  -16 2.5          -6.4
## 7  -24 2.63         -9.11
## 8   11 0.883        12.5
## 9    5 2.33          2.14
## 10 -10 2.3         -4.35
## # ... with 336,766 more rows
```

Aggregate Functions

```
y <- c(1, 2, 2, NA, 3, 4)
min_rank(y)
```

```
## [1] 1 2 2 NA 4 5
```

```
row_number(y)
```

```
## [1] 1 2 3 NA 4 5
```

```
dense_rank(y)
```

```
## [1] 1 2 2 NA 3 4
```

Exercises

```
transmute(flights, air_time, arr_time - dep_time)
```

```
## # A tibble: 336,776 x 2
##   air_time `arr_time - dep_time`
##   <dbl>         <int>
```

```
## 1      227      313
## 2      227      317
## 3      160      381
## 4      183      460
## 5      116      258
## 6      150      186
## 7      158      358
## 8       53      152
## 9      140      281
## 10     138      195
## # ... with 336,766 more rows

#transmute(flights, arr_time, sched_arr_time, arr_delay, dep_time, sched_dep_time, dep_delay)

transmute(flights, dep_time, sched_dep_time, dep_delay)

## # A tibble: 336,776 x 3
##   dep_time sched_dep_time dep_delay
##   <int>      <int>      <dbl>
## 1      517          515          2
## 2      533          529          4
## 3      542          540          2
## 4      544          545         -1
## 5      554          600         -6
## 6      554          558         -4
## 7      555          600         -5
## 8      557          600         -3
## 9      557          600         -3
## 10     558          600         -2
## # ... with 336,766 more rows

fl_df <- mutate(flights, total_delay = arr_delay + dep_delay)
transmute(arrange(fl_df, desc(total_delay)), total_delay)

## # A tibble: 336,776 x 1
##   total_delay
##   <dbl>
## 1      2573
## 2      2264
## 3      2235
## 4      2021
## 5      1994
## 6      1891
## 7      1826
## 8      1793
## 9      1774
## 10     1753
## # ... with 336,766 more rows

1:3 + 1:10 # error
```