

Transformation

Prof. Eric A. Suess

Chapter 4 Data Transformation

The 5 verbs of data wrangling

- Pick observations by their values (**filter()**).
- Reorder the rows (**arrange()**).
- Pick variables by their names (**select()**).
- Create new variables with functions of existing variables (**mutate()**).
- Collapse many values down to a single summary (**summarise()**).
- (**group_by()**)

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

We will continue to work with the *flights* dataset that is in the *ggplot2* package.

```
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>
```

Change the code from the Transformation presentation to using the pipe `%>%`.

`filter()`

```
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>
```

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>
```

arrange()

```
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>
```

select()

```
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
```

select()

```
select(flights, time_hour, air_time, everything())
```

```
## # A tibble: 336,776 x 19
##   time_hour          air_time year month   day dep_time sched_dep_time
##   <dtm>              <dbl> <int> <int> <int>   <int>         <int>
## 1 2013-01-01 05:00:00      227  2013     1     1     517           515
## 2 2013-01-01 05:00:00      227  2013     1     1     533           529
## 3 2013-01-01 05:00:00      160  2013     1     1     542           540
## 4 2013-01-01 05:00:00      183  2013     1     1     544           545
## 5 2013-01-01 06:00:00      116  2013     1     1     554           600
## 6 2013-01-01 05:00:00      150  2013     1     1     554           558
## 7 2013-01-01 06:00:00      158  2013     1     1     555           600
## 8 2013-01-01 06:00:00       53  2013     1     1     557           600
## 9 2013-01-01 06:00:00      140  2013     1     1     557           600
## 10 2013-01-01 06:00:00      138  2013     1     1     558           600
## # ... with 336,766 more rows, and 12 more variables: dep_delay <dbl>,
## #   arr_time <int>, sched_arr_time <int>, arr_delay <dbl>, carrier <chr>,
## #   flight <int>, tailnum <chr>, origin <chr>, dest <chr>, distance <dbl>,
## #   hour <dbl>, minute <dbl>
```

mutate()

```
flights_sml <- select(flights,
  year:day,
  ends_with("delay"),
  distance,
```

```

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

```

```

## # 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

```

summarize()

```

summarise(flights, delay = mean(dep_delay, na.rm = TRUE))

```

```

## # A tibble: 1 x 1
##   delay
##   <dbl>
## 1  12.6

```

```

by_day <- group_by(flights, year, month, day)
summarise(by_day, delay = mean(dep_delay, na.rm = TRUE))

```

```

## # A tibble: 365 x 4
## # Groups:   year, month [?]
##   year month   day delay
##   <int> <int> <int> <dbl>
## 1  2013     1     1  11.5
## 2  2013     1     2  13.9
## 3  2013     1     3  11.0
## 4  2013     1     4   8.95
## 5  2013     1     5   5.73
## 6  2013     1     6   7.15
## 7  2013     1     7   5.42
## 8  2013     1     8   2.55
## 9  2013     1     9   2.28
## 10 2013     1    10   2.84
## # ... with 355 more rows

```

Combining multiple operations using functions and assignment <-

```
by_dest <- group_by(flights, dest)
delay <- summarise(by_dest,
  count = n(),
  dist = mean(distance, na.rm = TRUE),
  delay = mean(arr_delay, na.rm = TRUE)
)
delay <- filter(delay, count > 20, dest != "HNL")
```

Combining multiple operations using functions and assignment <-, note the ggplot “piping” using the +

```
ggplot(data = delay, mapping = aes(x = dist, y = delay)) +
  geom_point(aes(size = count), alpha = 1/3) +
  geom_smooth(se = FALSE)
```

It looks like delays increase with distance up to ~750 miles and then decrease. Maybe as flights get longer there's more ability to make up delays in the air?

geom_smooth() using method = 'loess' and formula 'y ~ x'

Combining multiple operations with the pipe %>%

Does this code read better?

```
delays <- flights %>%
  group_by(dest) %>%
  summarise(
    count = n(),
    dist = mean(distance, na.rm = TRUE),
    delay = mean(arr_delay, na.rm = TRUE)
  ) %>%
  filter(count > 20, dest != "HNL")

head(delays)
```

```
## # A tibble: 6 x 4
##   dest  count  dist delay
##   <chr> <int> <dbl> <dbl>
## 1 ABQ     254  1826   4.38
## 2 ACK     265   199   4.85
## 3 ALB     439   143  14.4
## 4 ATL  17215   757  11.3
## 5 AUS   2439  1514   6.02
## 6 AVL     275   584   8.00
```