Introduction to R for Data Management and Analysis

Marcel Ramos, MPH

Thursday, June 14, 2018

Notes on last Thursday's lecture

- Examples with pipes
- Aggregate function
- Formulas

*Using the **nycflights13** dataset*

```
library(nycflights13); library(dplyr)
flights %>% group_by(carrier) %>%
  summarise(avg_depdelay = mean(dep_delay, na.rm = TRUE),
            count = n()) %>% left_join(airlines) %>%
    arrange(avg_depdelay) %>% head
```

```
## # A tibble: 6 x 4
##   carrier avg_depdelay count name
##   <chr>          <dbl> <int> <chr>
## 1 US              3.78 20536 US Airways Inc.
## 2 HA              4.90   342 Hawaiian Airlines Inc.
## 3 AS              5.80   714 Alaska Airlines Inc.
## 4 AA              8.59 32729 American Airlines Inc.
## 5 DL              9.26 48110 Delta Air Lines Inc.
## 6 MQ             10.6  26397 Envoy Air
```

Reshaping data using ***gather***

```
data(iris); library(tidyr)
longdata <- gather(tbl_df(iris), key = measure, n,
  Sepal.Length:Petal.Width) %>% separate(measure, c("type",
    "dimension"))
longdata %>% group_by(Species, type, dimension) %>%
  summarise(avg_dim = mean(n, na.rm = TRUE))
```

```
## # A tibble: 12 x 4
## # Groups:   Species, type [?]
##    Species    type  dimension avg_dim
##    <fct>      <chr> <chr>       <dbl>
##  1 setosa     Petal Length      1.46
##  2 setosa     Petal Width       0.246
##  3 setosa     Sepal Length      5.01
##  4 setosa     Sepal Width       3.43
##  5 versicolor Petal Length      4.26
```

Pew example

```r
library(readr)
(pew <- read_csv("../Data/pew.csv"))
```

```
## Parsed with column specification:
## cols(
##   religion = col_character(),
##   `<$10k` = col_integer(),
##   `$10-20k` = col_integer(),
##   `$20-30k` = col_integer(),
##   `$30-40k` = col_integer(),
##   `$40-50k` = col_integer(),
##   `$50-75k` = col_integer(),
##   `$75-100k` = col_integer(),
##   `$100-150k` = col_integer(),
##   `>150k` = col_integer(),
##   `Don't know/refused` = col_integer()
```

Gather dataset

```r
pew %>% gather(income, n, -religion) %>% head
```

```
## # A tibble: 6 x 3
##   religion                income      n
##   <chr>                   <chr>   <int>
## 1 Agnostic                <$10k      27
## 2 Atheist                 <$10k      12
## 3 Buddhist                <$10k      27
## 4 Catholic                <$10k     418
## 5 Don’t know/refused <$10k      15
## 6 Evangelical Prot        <$10k     575
```

income, religion : variables to gather n : variable in cells -religion means all except religion

Using group_by

```
pew %>% gather(income, n, -religion) %>%
  group_by(income) %>% summarise(totals = sum(n))
```

```
## # A tibble: 10 x 2
##    income                   totals
##    <chr>                     <int>
##  1 <$10k                      1930
##  2 >150k                      2608
##  3 $10-20k                    2781
##  4 $100-150k                  3197
##  5 $20-30k                    3357
##  6 $30-40k                    3302
##  7 $40-50k                    3085
##  8 $50-75k                    5185
##  9 $75-100k                   3990
## 10 Don't know/refused         6121
```

Using group_by

```r
pew %>% gather(income, n, -religion) %>%
  group_by(religion) %>% summarise(totals = sum(n))
```

```
## # A tibble: 18 x 2
##    religion                totals
##    <chr>                    <int>
##  1 Agnostic                   826
##  2 Atheist                    515
##  3 Buddhist                   411
##  4 Catholic                  8054
##  5 Don’t know/refused         272
##  6 Evangelical Prot          9472
##  7 Hindu                      257
##  8 Historically Black Prot   1995
##  9 Jehovah's Witness          215
## 10 Jewish                     682
```

Plotting and Graphing

- Exploratory Data Analysis
- Base graphics
- Intro `ggplot2`
- Saving graphics

Plotting systems in R

- 'Base' graphics
- lattice
- ggplot2

Exploratory Data Analysis

- Informal representation data
- Looking for patterns, outliers, etc.

Types of graphs

- Historgram
- Scatterplot
    - Scatterplot matrix
- Boxplots
- Violin plots (`ggplot2`)
- Q-Q plots

`par` *function*

- Check parameters for graphing

ggplot2 - *Grammar of Graphics*

- Different syntax
- Powerful operations

Saving output to file

- Formats
  - PDF
  - SVG
  - PNG/TIFF

End in `dev.off()`

`ggplot2` graphics may require a `print` before it gets rendered in the file.