Visualization in R using ggplot2

Angela Zoss

3/20/19

https://github.com/amzoss/ggplot2-DF19

Set up environment

- R
- RStudio
- tidyverse package

Try right now:
Open RStudio
Try running “library(tidyverse)”
Tell me about any errors

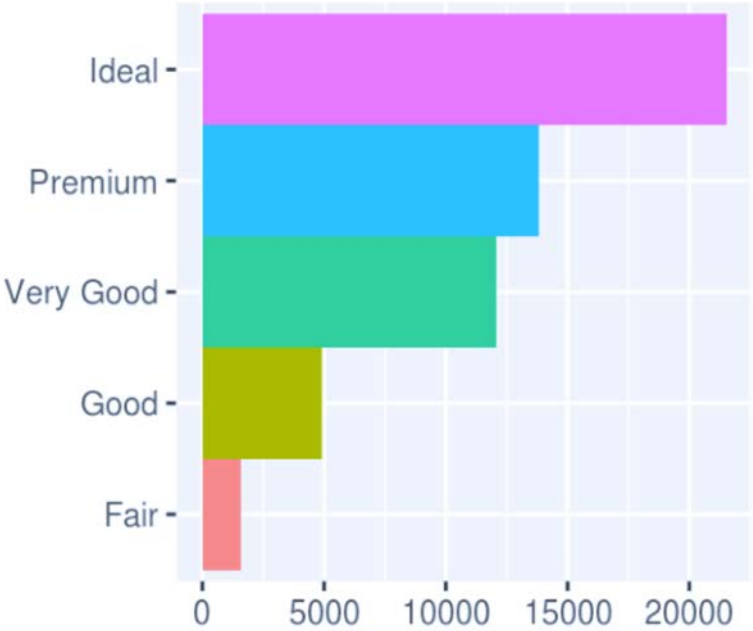
Don’t have it installed?

https://vm-manage.oit.duke.edu/containers

ggplot2

What is ggplot2?

an R package designed to create plots based on a theory of the grammar of graphics.



http://r4ds.had.co.nz/data-visualisation.html

Grammar of graphics

1. DATA: a set of data operations that create variables from datasets
2. TRANS: variable transformations (e.g., rank)
3. SCALE: scale transformations (e.g., log)
4. COORD: a coordinate system (e.g., polar)
5. ELEMENT: graphs (e.g., points) and their aesthetic attributes (e.g., color)
6. GUIDE: one or more guides (axes, legends, etc.).

Wilkinson, Leland. (2005). *The grammar of graphics (2nd ed)*. New York: Springer.

Why ggplot2 instead of base R?

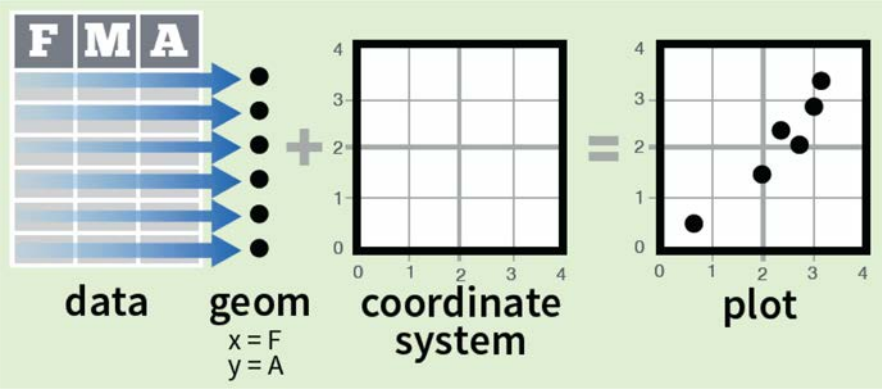
- nice defaults
- easy faceting
- (arguably) more natural syntax
- can switch chart types more easily

"Why I use ggplot2", David Robinson
http://varianceexplained.org/r/why-I-use-ggplot2/

ggplot2: Elements

Basic elements in any ggplot2 visualization

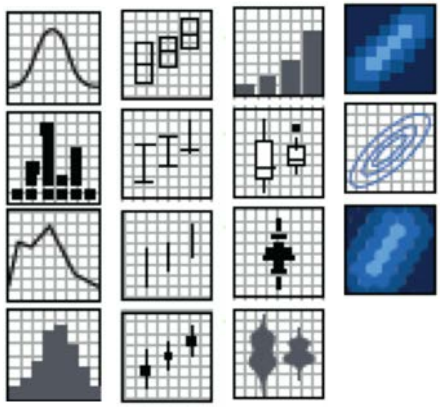
- **data**
- **aes**thetics
  (variable mappings)
- **geom**
  (chart type or shape)
- coordinate system
  (the arrangement of the marks;
  most geoms use default, cartesian)



http://bit.ly/ggplot2-cheatsheet

Types of geoms

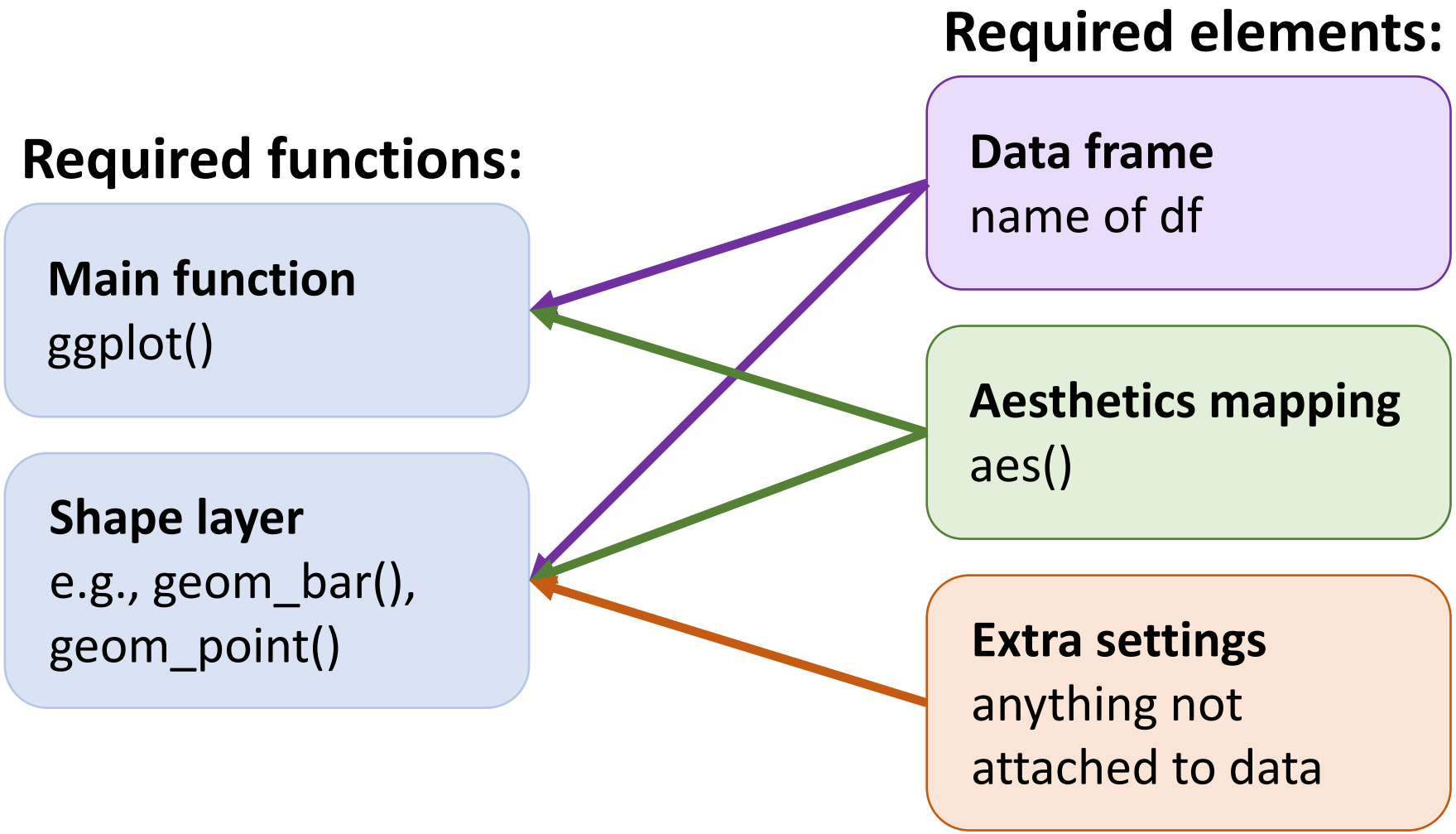
- geom_bar()
- geom_point()
- geom_histogram()
- geom_map()
- etc.



http://bit.ly/ggplot2-cheatsheet

Note: some geoms also include data summary functions.
e.g., the “bar” geom will count data points in each category.

ggplot2: Basic syntax


Required functions:
Main function
ggplot()
Shape layer
e.g., geom_bar(),
geom_point()
Required elements:
Data frame
name of df
Aesthetics mapping
aes()
Extra settings
anything not
attached to data

Template for a simple plot

```
ggplot( data = data frame )  +

    geom_...( mapping = aes(variable mappings),
              non-variable adjustments )
```

Template for a simple plot

Main function

```
ggplot( data = data frame )  +
```

Shape layer

```
geom_...( mapping = aes(variable mappings),
          non-variable adjustments )
```

Step-by-step

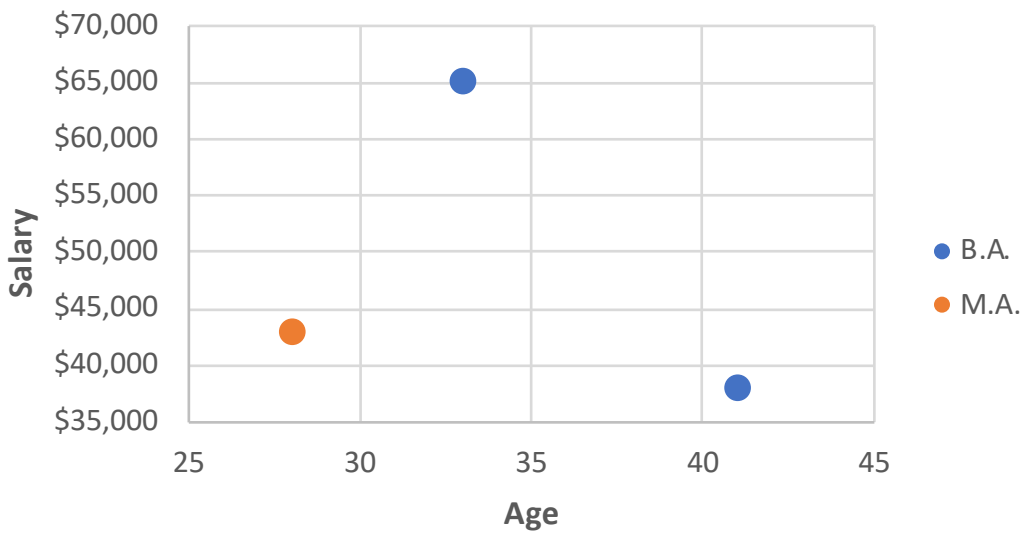
1. Set the data
2. Choose a shape layer
3. Map variables to aesthetics
4. Add non-variable adjustments

1. Set the data

"employees"

Name	Age	Salary	Highest Degree
Jane Smith	33	$65,000	B.A.
Abby Jones	28	$43,000	M.A.
Bridget Carden	41	$38,000	B.A.

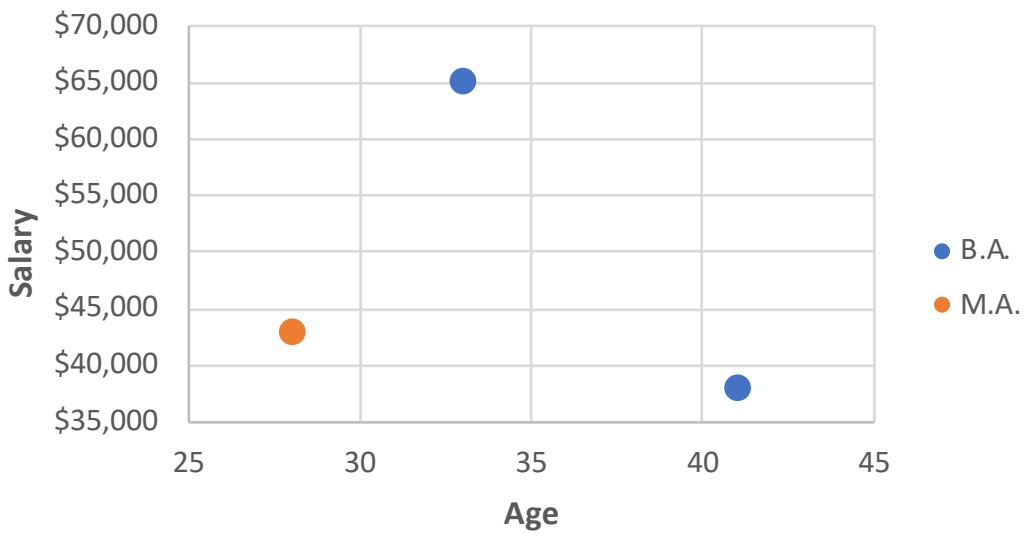
```
ggplot(employees)
```



2. Choose a shape layer

Name	Age	Salary	Highest Degree
Jane Smith	33	$65,000	B.A.
Abby Jones	28	$43,000	M.A.
Bridget Carden	41	$38,000	B.A.

```
ggplot(employees) +
    geom_point()
```

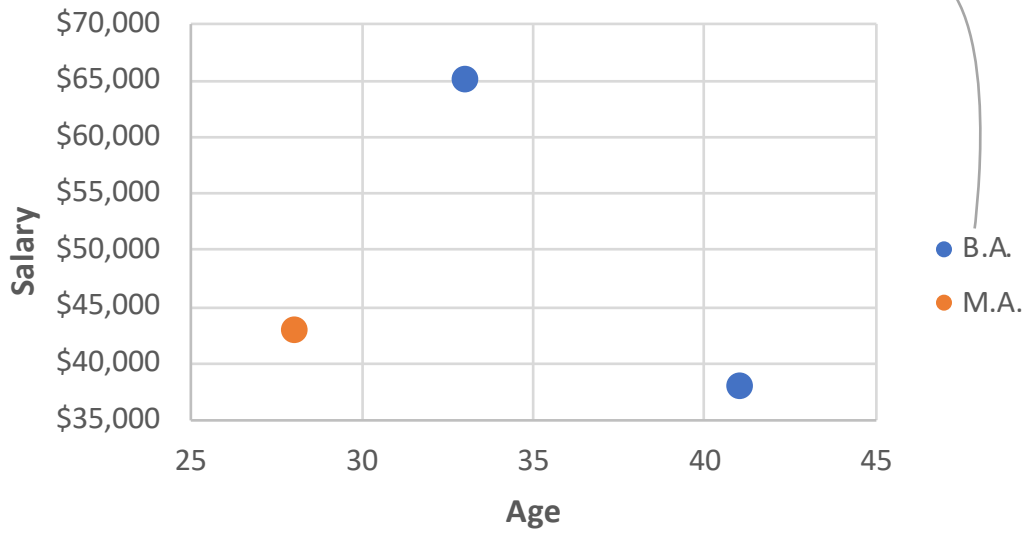


3. Map variables to aesthetics

Name	Age	Salary	Highest Degree
Jane Smith	33	$65,000	B.A.
Abby Jones	28	$43,000	M.A.
Bridget Carden	41	$38,000	B.A.

```
ggplot(employees) +
    geom_point(
        aes(x=age,
            y=salary,
            color=degree))
```

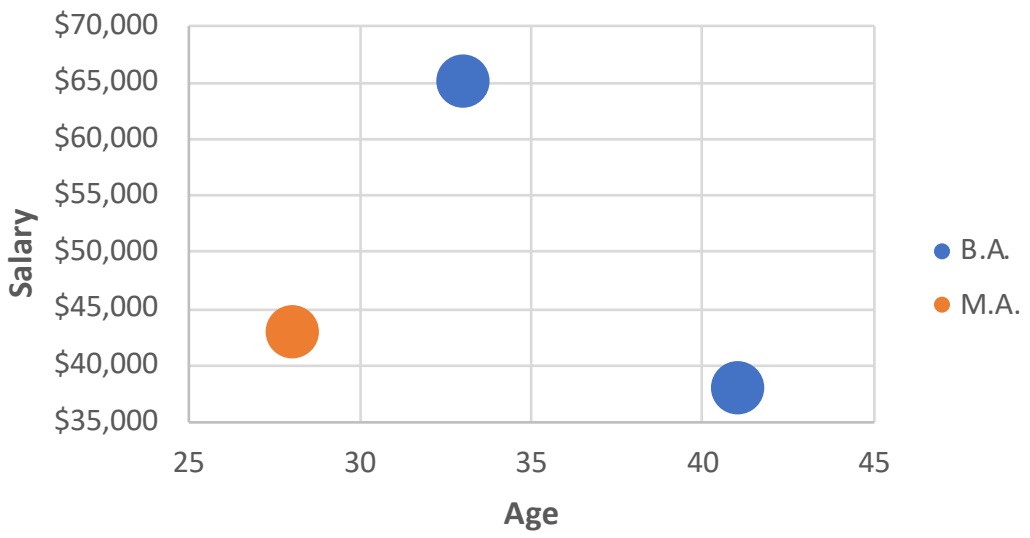
x position y position color



4. Add non-variable adjustments

Name	Age	Salary	Highest Degree
Jane Smith	33	$65,000	B.A.
Abby Jones	28	$43,000	M.A.
Bridget Carden	41	$38,000	B.A.

```
ggplot(employees) +
    geom_point(
        aes(x=age,
            y=salary,
            color=degree),
        size=10)
```



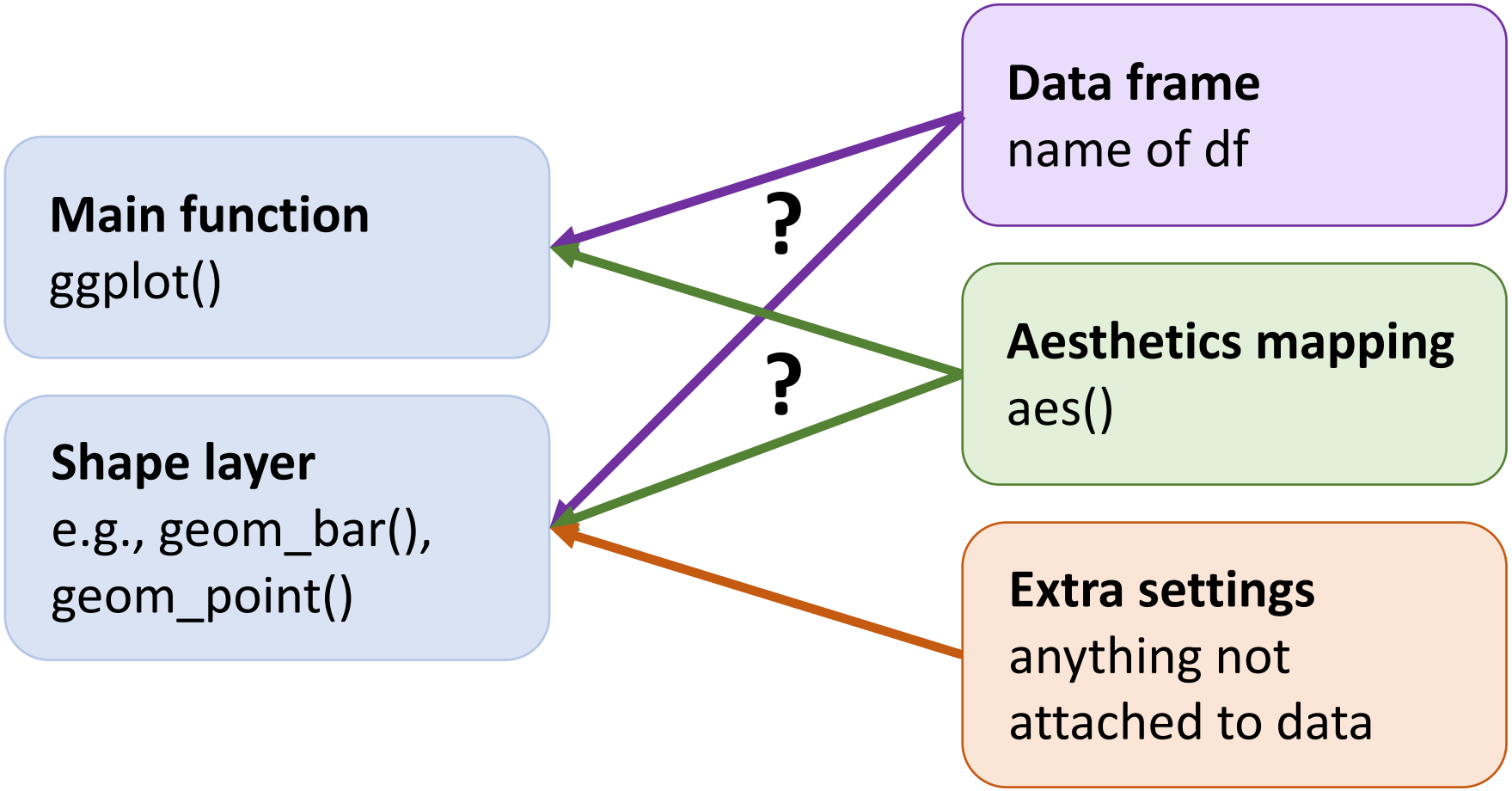
Template for a simple plot

Main function

```
ggplot( data = data frame )  +
```

Shape layer

```
geom_...( mapping = aes(variable mappings),
          non-variable adjustments )
```


Main function
ggplot()
Shape layer
e.g., geom_bar(),
geom_point()
?
?
Data frame
name of df
Aesthetics mapping
aes()
Extra settings
anything not
attached to data

Inheritance

data and aesthetics will carry through from top to bottom

```
ggplot( data = data frame,
        mapping = aes(variable mappings) )
```

+

```
geom_...( aes(add'l variable mappings),
          non-variable adjustments )
```

+

```
geom_...( aes(add'l variable mappings),
          non-variable adjustments )
```

Inheritance

data and aesthetics will carry through from top to bottom

Main function

```
ggplot( data = data frame,
        mapping = aes(variable mappings) )
```

+

Shape layer

```
geom_...( aes(add'l variable mappings),
          non-variable adjustments )
```

+

Shape layer

```
geom_...( aes(add'l variable mappings),
          non-variable adjustments )
```

General inheritance strategy

Main function

```
ggplot( main data frame,
        aes(x, y) )
```

+

Shape layer

```
geom_...( aes(add'l variable mappings),
          non-variable adjustments )
```

+

Shape layer

```
geom_...( aes(add'l variable mappings),
          non-variable adjustments )
```

Working in RStudio

Set up environment

- R
- RStudio
- tidyverse package

Try right now:
Open RStudio
Try running “library(tidyverse)”
Tell me about any errors

Don’t have it installed?

https://vm-manage.oit.duke.edu/containers

Get workshop files

URL: https://github.com/amzoss/ggplot2-DF19

With Git installed

In RStudio:

- Project → New project
- Version Control
- Git
  - Paste in GitHub URL
  - Project directory name: ggplot2-DF19
  - Subdirectory: you choose
- Create Project

Without Git installed

- Click green button to download ZIP
- Unzip files on your laptop

In RStudio:

- Project → New project…
- Existing directory
- Select unzipped folder
- Create Project

Using RStudio

- Projects
- Rmarkdown
- Cheat sheets

https://www.rstudio.com/resources/cheatsheets/#rmarkdown

Why Rmarkdown?

- Plots show up inline
- Easier to incorporate explanatory text and materials
- Like to be able to easily run one chunk at a time

Caution: Running things out of order can mean your code won’t work again later. Clear your environment often and run code chunks in order to be safe.

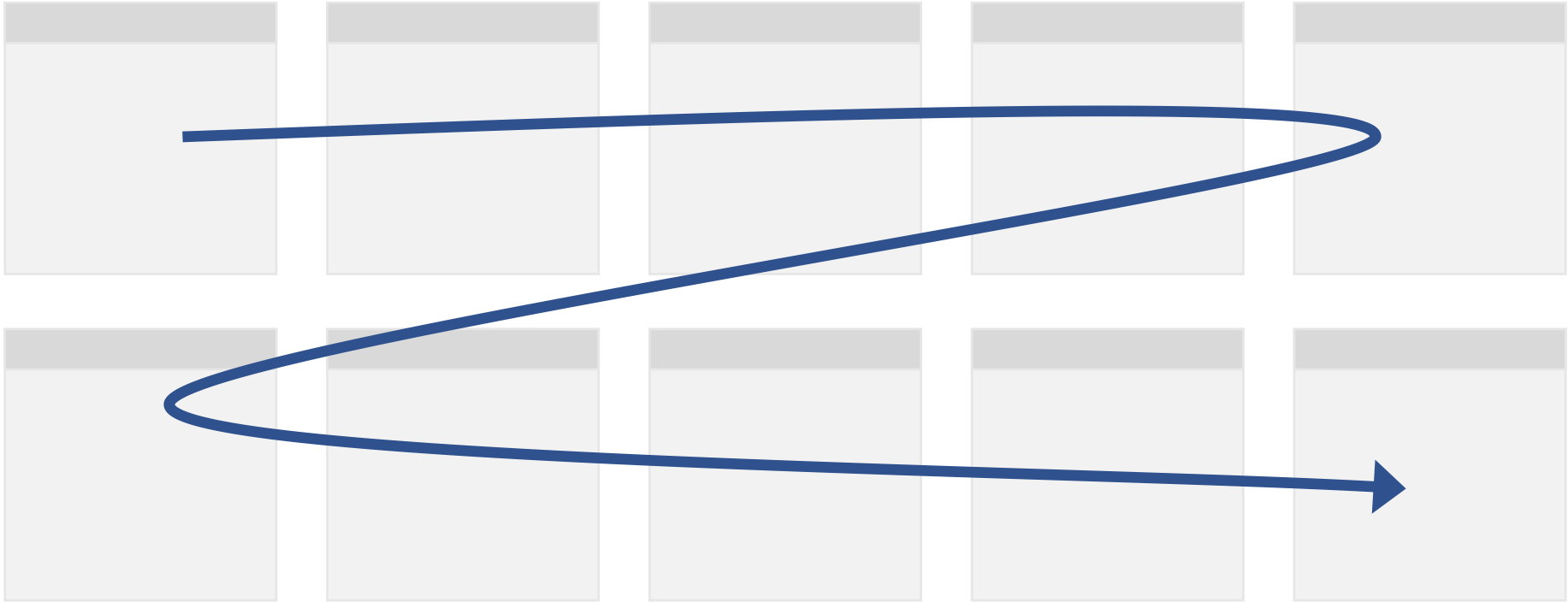
Game of Thrones character ratings

https://www.nytimes.com/interactive/2017/08/09/upshot/game-of-thrones-chart.html

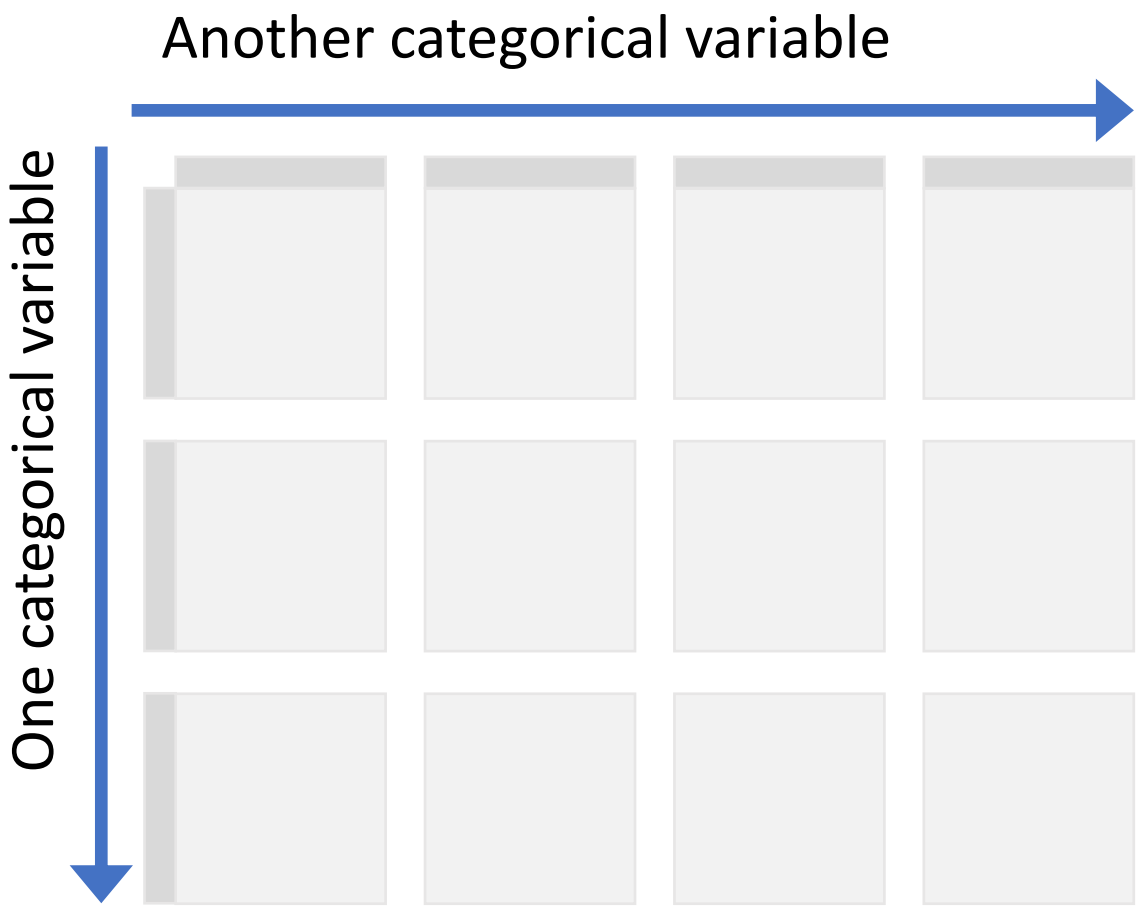
Creating repeated charts

Facet_wrap

```
+ facet_wrap(~variable)
```



Facet_grid




```
+ facet_grid(yvar~xvar)
+ facet_grid(.~xvar)
+ facet_grid(yvar~.)
```

Helpful data manipulation

About %>%

- Loads automatically with tidyverse
- Used throughout tidyverse (except for ggplot2)
- Pushes data from the left into the function on the right

```
data_frame %>% function(args)
```

filter

Select a subset of rows

```
data %>% dplyr::filter(name == "John")
```

https://www.rstudio.com/resources/cheatsheets/#dplyr

select

Select a subset of columns

```
data %>% dplyr::select(id, name, age)
```

```
data %>% dplyr::select(-count)
```

https://www.rstudio.com/resources/cheatsheets/#dplyr

drop_na

Remove rows with NA values, either in any column or in specified columns

```
data %>% drop_na()
```

```
data %>% drop_na(age)
```

https://www.rstudio.com/resources/cheatsheets/#import

Star Wars character data

https://dplyr.tidyverse.org/reference/starwars.html

Working with text variables

Text variables

In R, "character" variables

Gender	Age	Household Income	Education
Response	Response	Response	Response
Male	18-29		High school degree
Male	18-29	$0 - $24,999	Bachelor degree
Male	18-29	$0 - $24,999	High school degree
Male	18-29	$100,000 - $149,999	Some college or Associate degree
Male	18-29	$100,000 - $149,999	Some college or Associate degree
Male	18-29	$25,000 - $49,999	Bachelor degree
Male	18-29		High school degree
Male	18-29		High school degree
Male	18-29	$0 - $24,999	Some college or Associate degree
Male	18-29	$25,000 - $49,999	Some college or Associate degree
Male	18-29	$25,000 - $49,999	Bachelor degree
Male	30-44	$50,000 - $99,999	Graduate degree
Male	18-29		High school degree
Male	18-29	$0 - $24,999	Some college or Associate degree
Male	18-29	$50,000 - $99,999	Bachelor degree

Factors

- Specify “levels” for a categorical variable
- Specify the order of those levels
- Specify whether the factor is “ordered”

```
month_levels <- c( "Jan", "Feb",
"Mar", "Apr", "May", "Jun", "Jul",
"Aug", "Sep", "Oct", "Nov", "Dec" )

y1 <- factor(x1,
             levels = month_levels)
```

https://r4ds.had.co.nz/factors.html

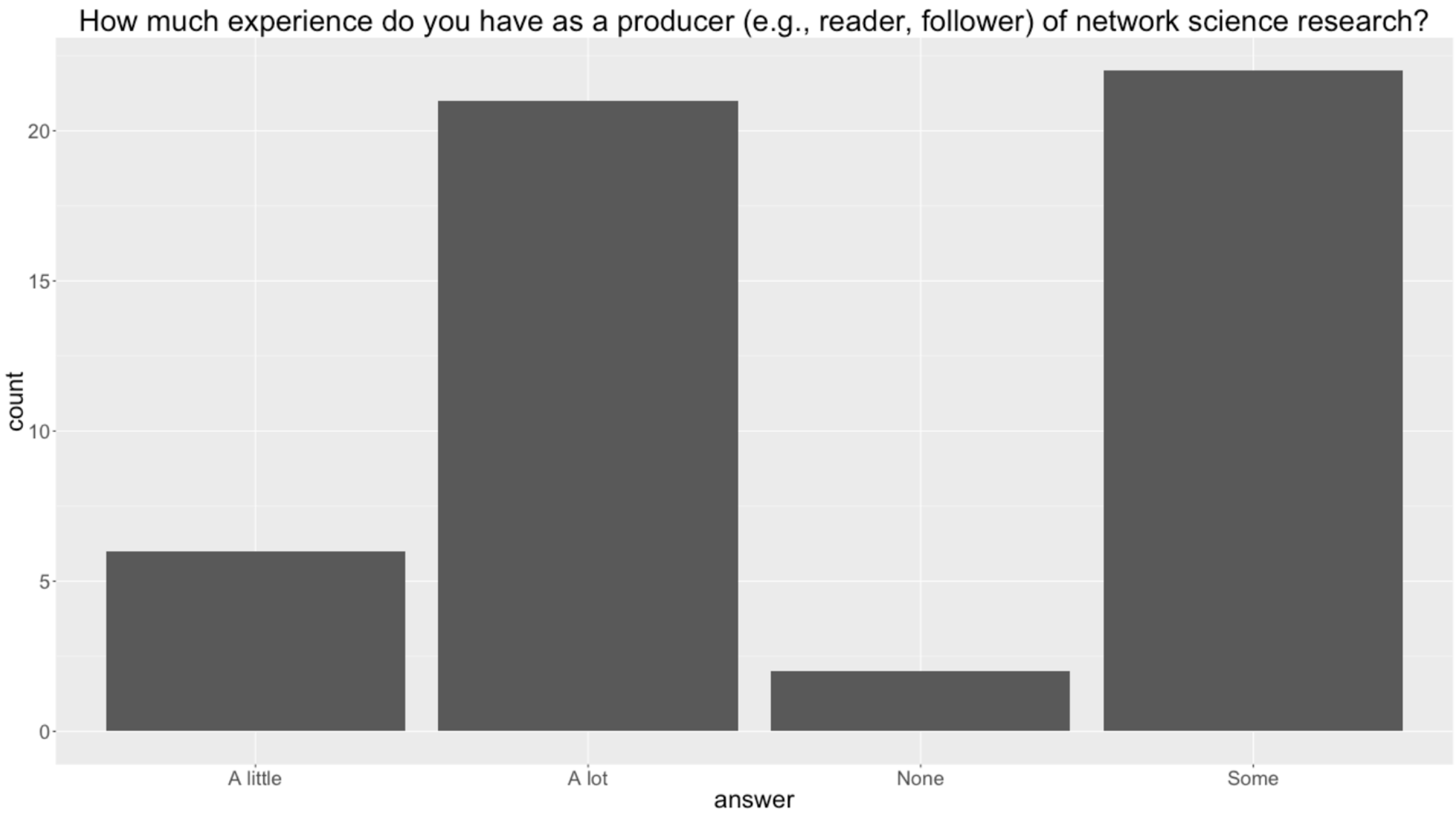
forcats package: helpful functions

- as_factor(char_var): convert a character variable to a factor
- fct_infreq(factor): sort a factor by frequency, descending
- fct_reorder(factor, num_var): sort a factor by a second, numerical variable

https://www.rstudio.com/resources/cheatsheets/#forcats

Principles for Effective Visualizations

Principle 1: Order matters


How much experience do you have as a producer (e.g., reader, follower) of network science research?
count
20
15
10
5
0
A little
A lot
None
Some
answer

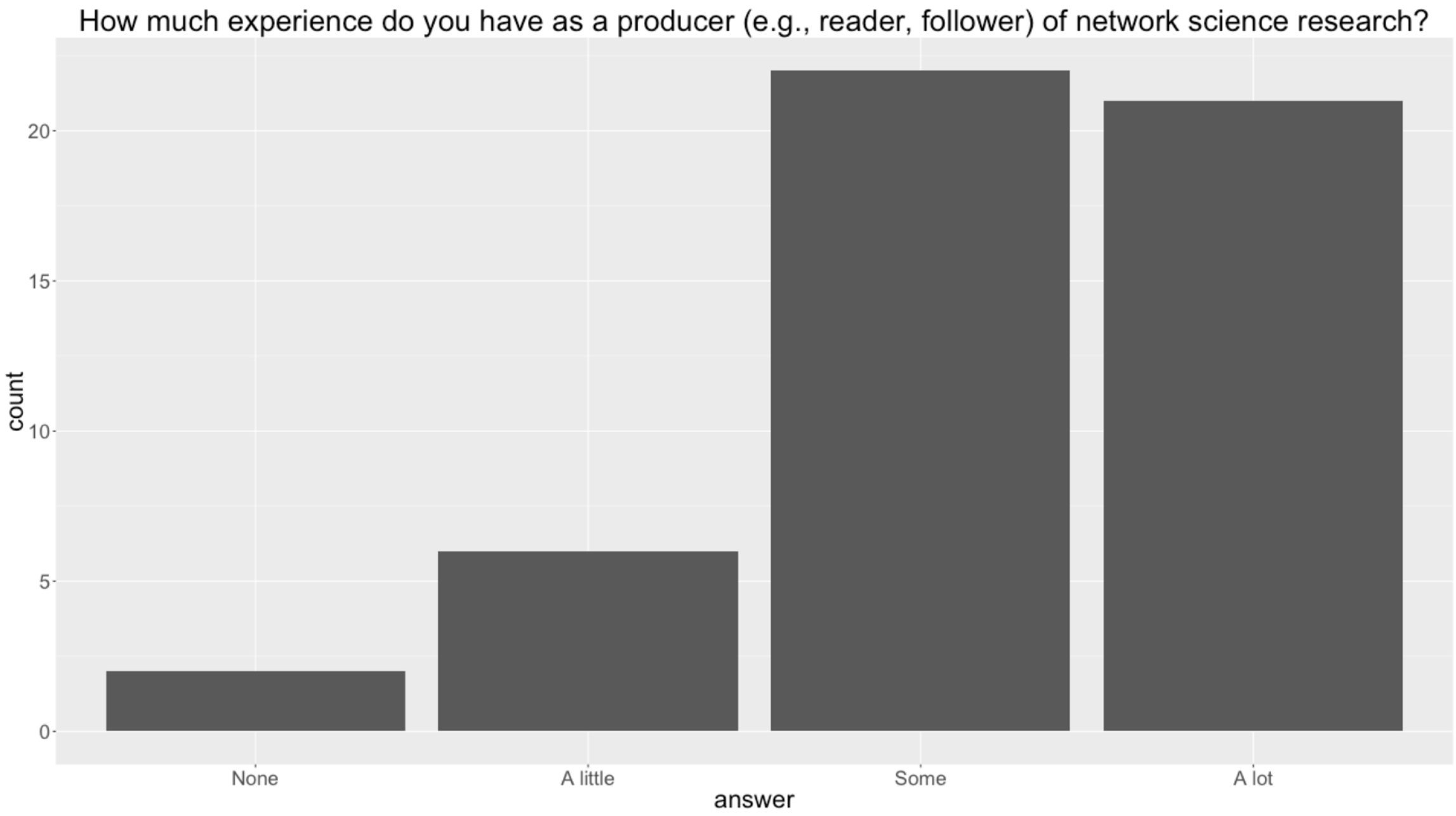
Order by meaning

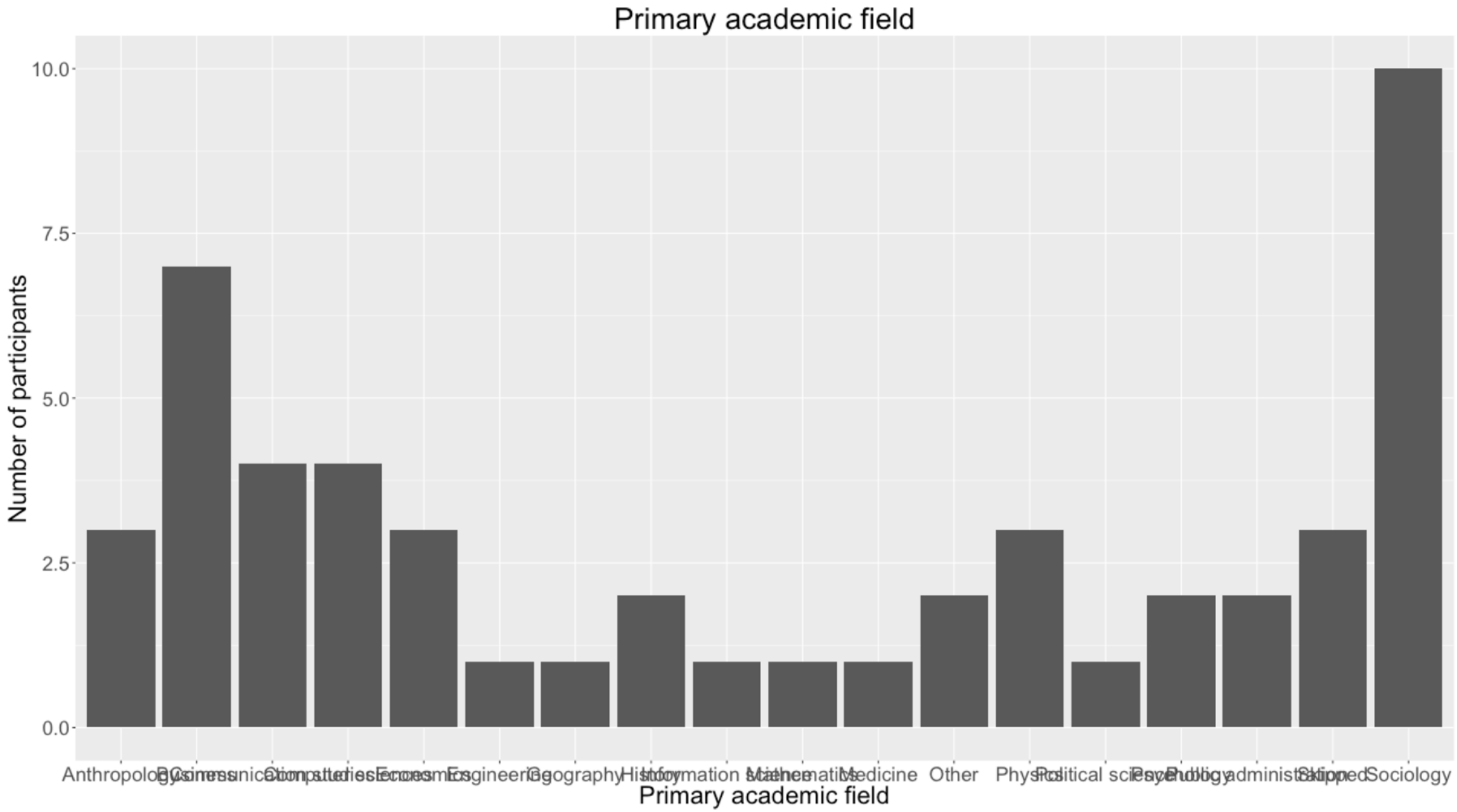
1. Treat variable as factor

```
data$answer <- fct_relevel(as_factor(data$answer),
                           c("None", "A little",
                             "Some", "A lot"))
```

2. Reorder levels manually

3. Specify levels in order


How much experience do you have as a producer (e.g., reader, follower) of network science research?
count
20
15
10
5
0
None
A little
Some
A lot
answer

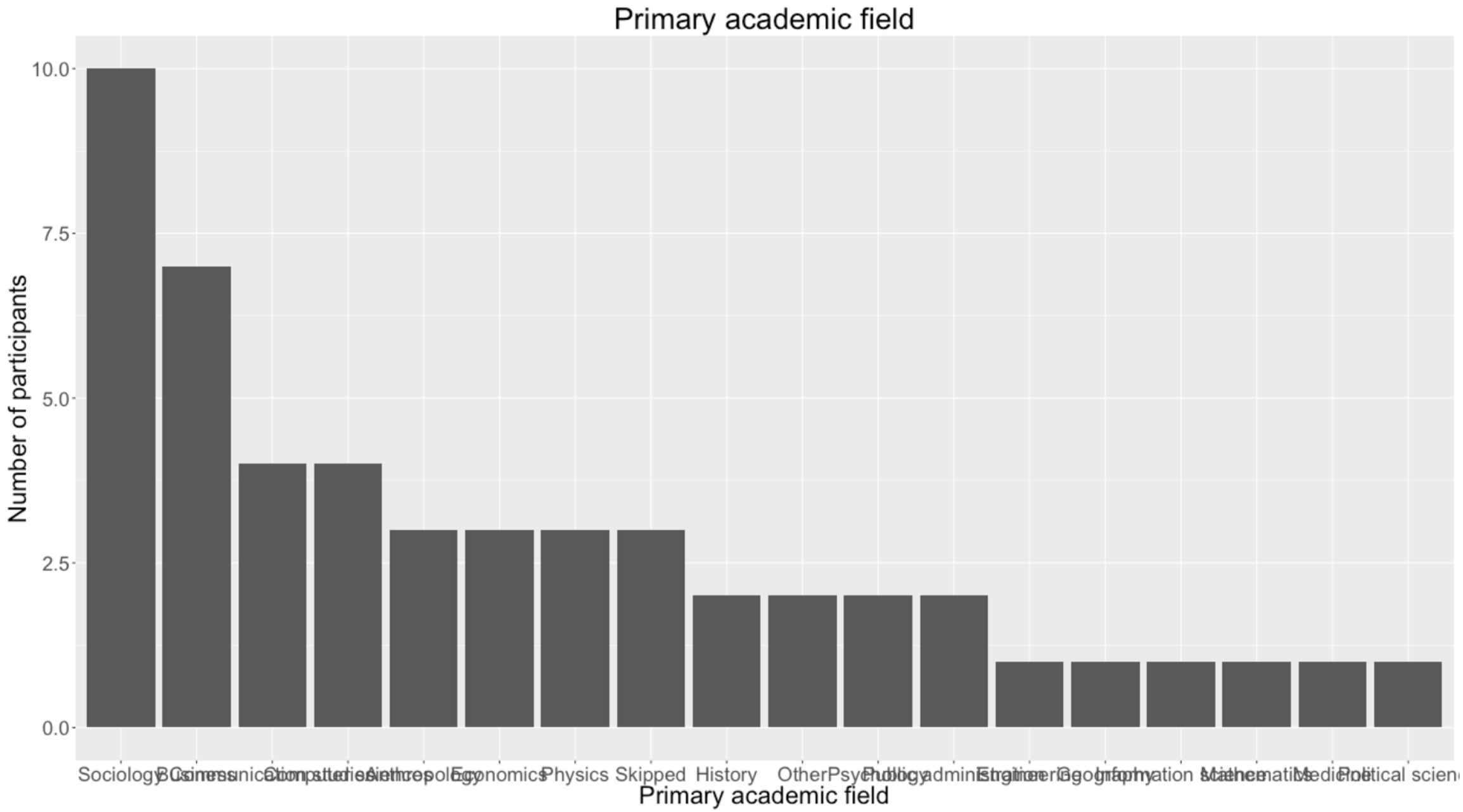

Primary academic field
Number of participants
10.0
7.5
5.0
2.5
0.0
Anthropology
Business
Communication
Computer sciences
Economics
Engineering
Geography
History
Information sciences
Mathematics
Medicine
Other
Physics
Political science
Psychology
Public administration
Skipped
Sociology
Primary academic field

Order by value (using forcats)

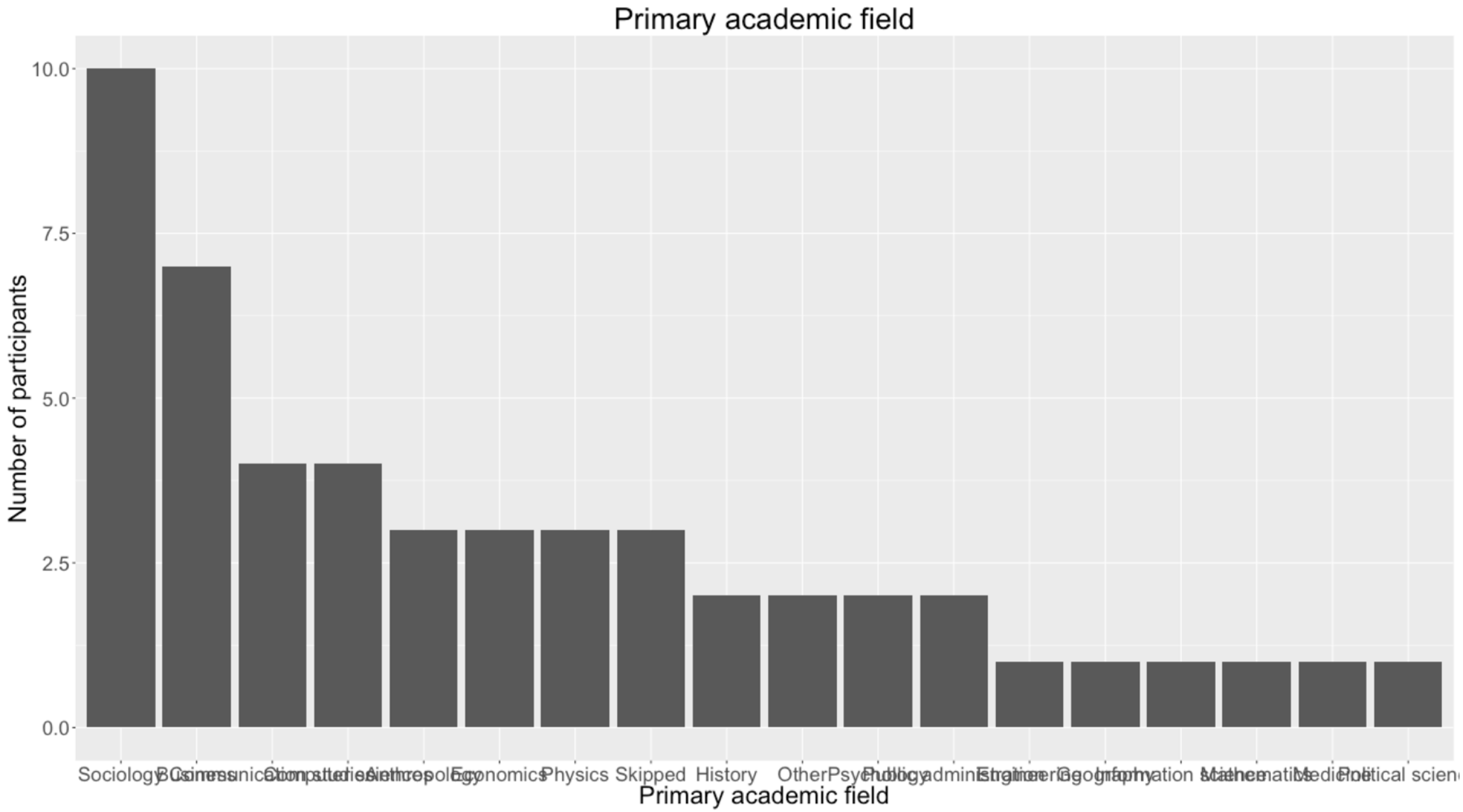
```
data$academic_field <-
     fct_infreq(as_factor(data$academic_field))
```

1. Treat variable as factor

2. Order factor by inverse frequency

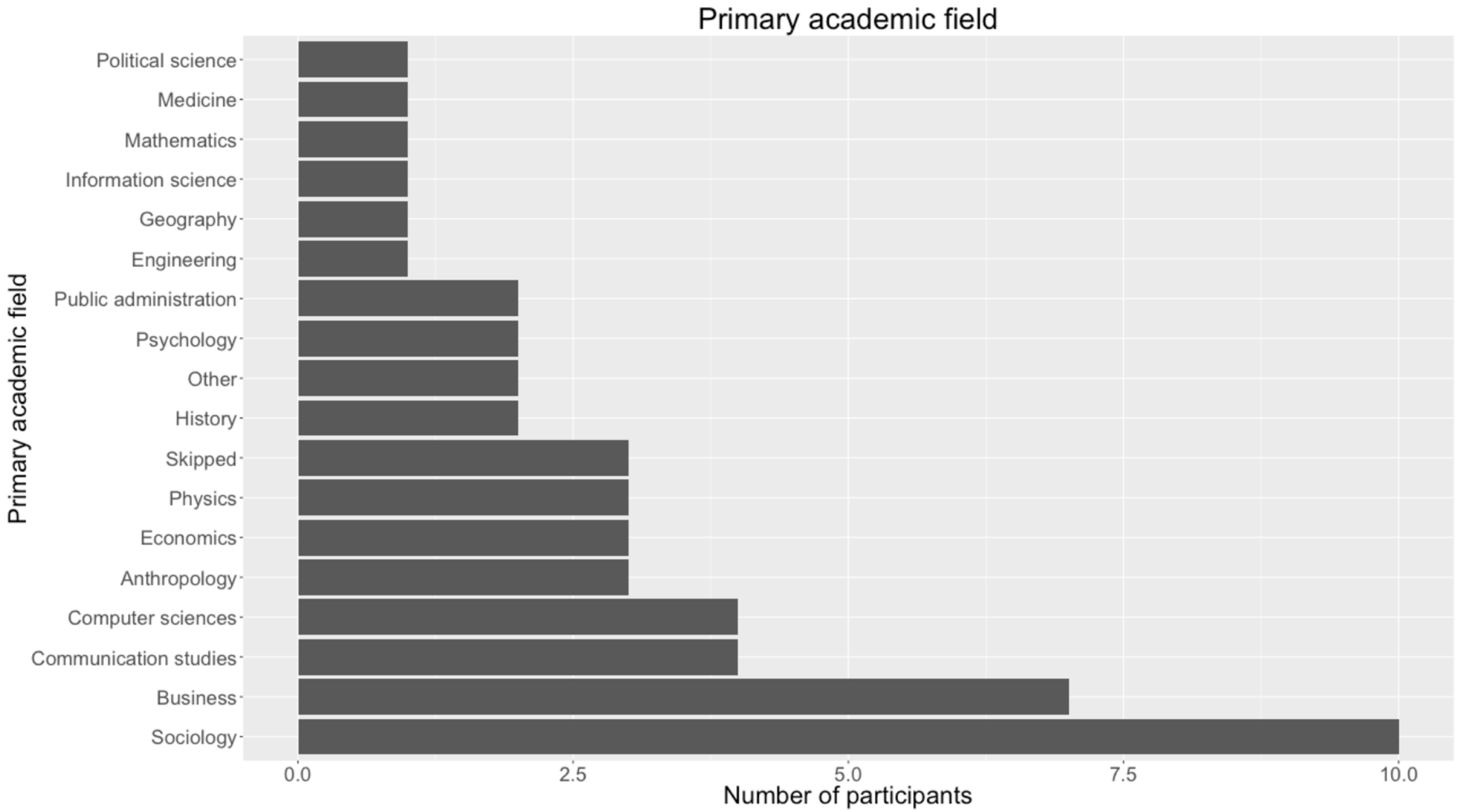

Primary academic field
Number of participants
10.0
7.5
5.0
2.5
0.0
Sociology
Business
Communication
Computer science
Anthropology
Economics
Physics
Skipped
History
Other
Psychology
Public administration
Engineering
Geography
Information science
Mathematics
Medicine
Political scien
Primary academic field

Principle 2:
Put long categories on y-axis


Primary academic field
Number of participants
10.0
7.5
5.0
2.5
0.0
Sociology
Business
Communication
Computer science
Anthropology
Economics
Physics
Skipped
History
Other
Psychology
Public administration
Engineering
Geography
Information science
Mathematics
Medicine
Political scien
Primary academic field

Flip the axes

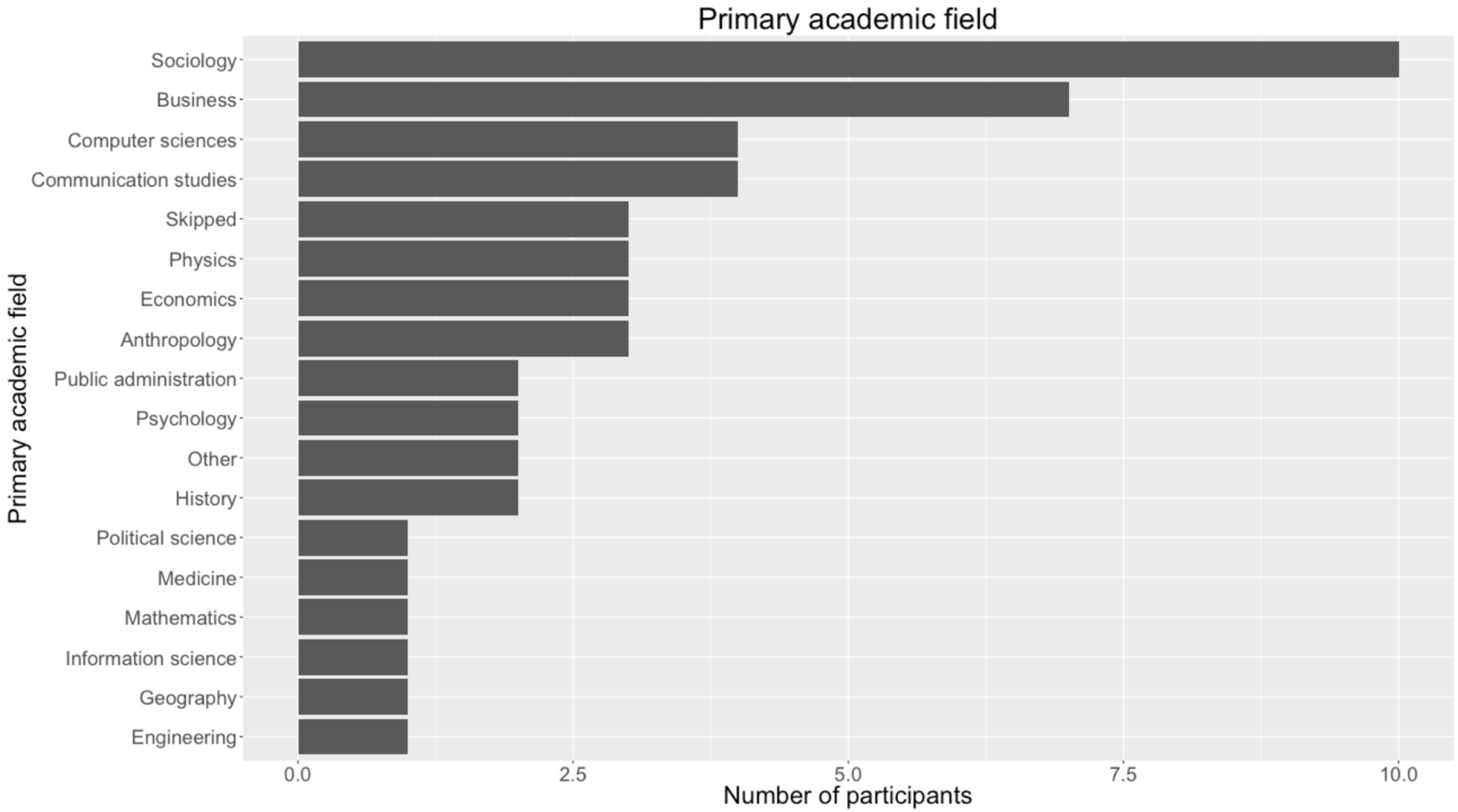
```
+ coord_flip()
```


Primary academic field
Political science
Medicine
Mathematics
Information science
Geography
Engineering
Public administration
Psychology
Other
History
Skipped
Physics
Economics
Anthropology
Computer sciences
Communication studies
Business
Sociology
Primary academic field
0.0
2.5
5.0
7.5
10.0
Number of participants

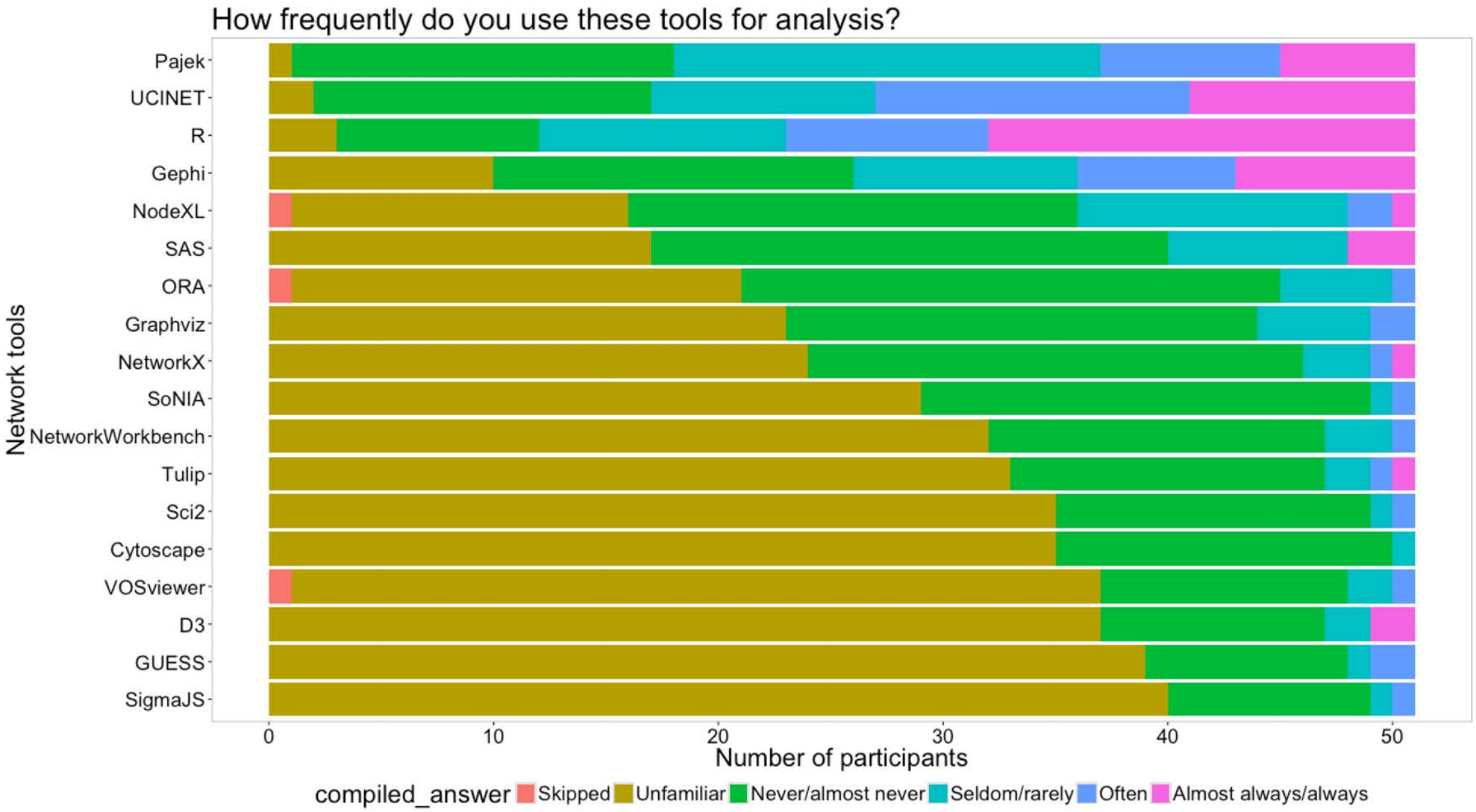
Oops!

```
data$academic_field <-
     fct_rev(fct_infreq(
       as_factor(data$academic_field)))
```

Have to reverse the order of the levels


Primary academic field
Sociology
Business
Computer sciences
Communication studies
Skipped
Physics
Economics
Anthropology
Public administration
Psychology
Other
History
Political science
Medicine
Mathematics
Information science
Geography
Engineering
Primary academic field
0.0
2.5
5.0
7.5
10.0
Number of participants

Principle 3:
Select meaningful colors

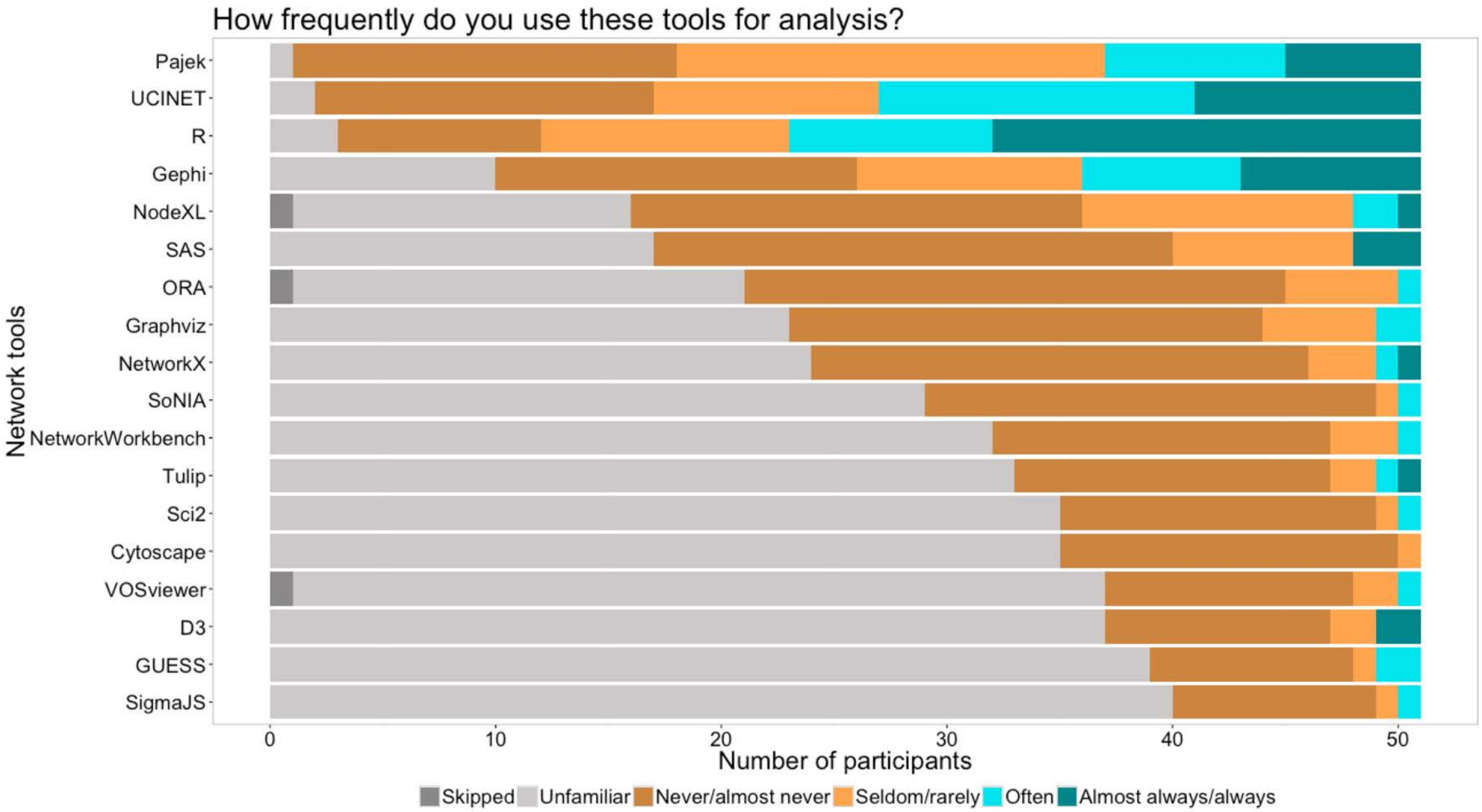

How frequently do you use these tools for analysis?
Network tools
Pajek
UCINET
R
Gephi
NodeXL
SAS
ORA
Graphviz
NetworkX
SoNIA
NetworkWorkbench
Tulip
Sci2
Cytoscape
VOSviewer
D3
GUESS
SigmaJS
0
10
20
30
40
50
Number of participants
compiled_answer
Skipped
Unfamiliar
Never/almost never
Seldom/rarely
Often
Almost always/always

Select colors manually, or use alternate palette

```
scale_fill_manual(
    values=c("snow4","snow3",
             "tan3","tan1",
             "turquoise2","turquoise4"))

scale_fill_manual(
    values=c("#fee391","#fe9929", "#cc4c02"))

# Also see package RColorBrewer
scale_fill_brewer(palette="BrBG")
```

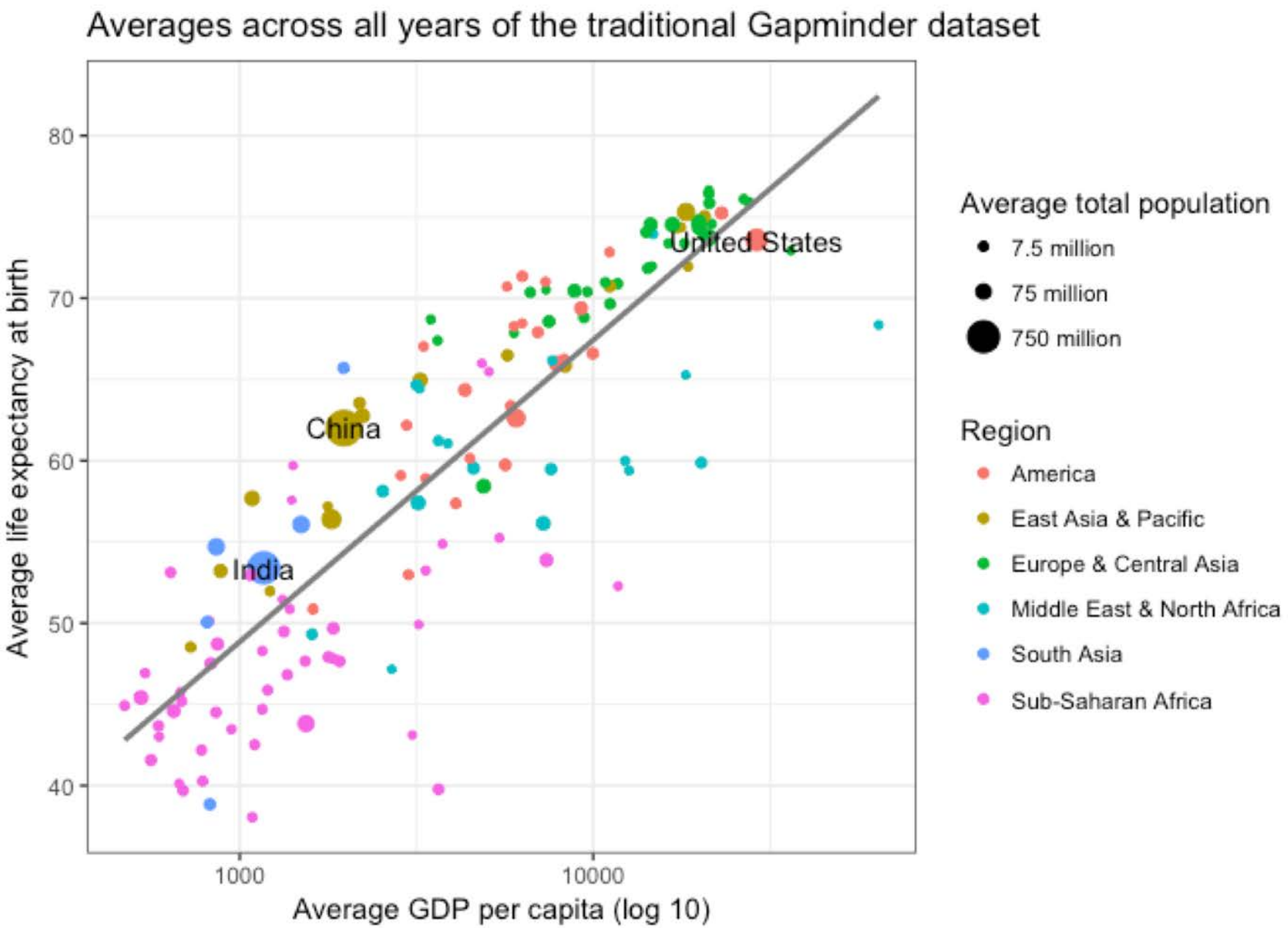

How frequently do you use these tools for analysis?
Network tools
Pajek
UCINET
R
Gephi
NodeXL
SAS
ORA
Graphviz
NetworkX
SoNIA
NetworkWorkbench
Tulip
Sci2
Cytoscape
VOSviewer
D3
GUESS
SigmaJS
0
10
20
30
40
50
Number of participants
Skipped
Unfamiliar
Never/almost never
Seldom/rarely
Often
Almost always/always

Star Wars opinion survey

https://fivethirtyeight.com/features/americas-favorite-star-wars-movies-and-least-favorite-characters/

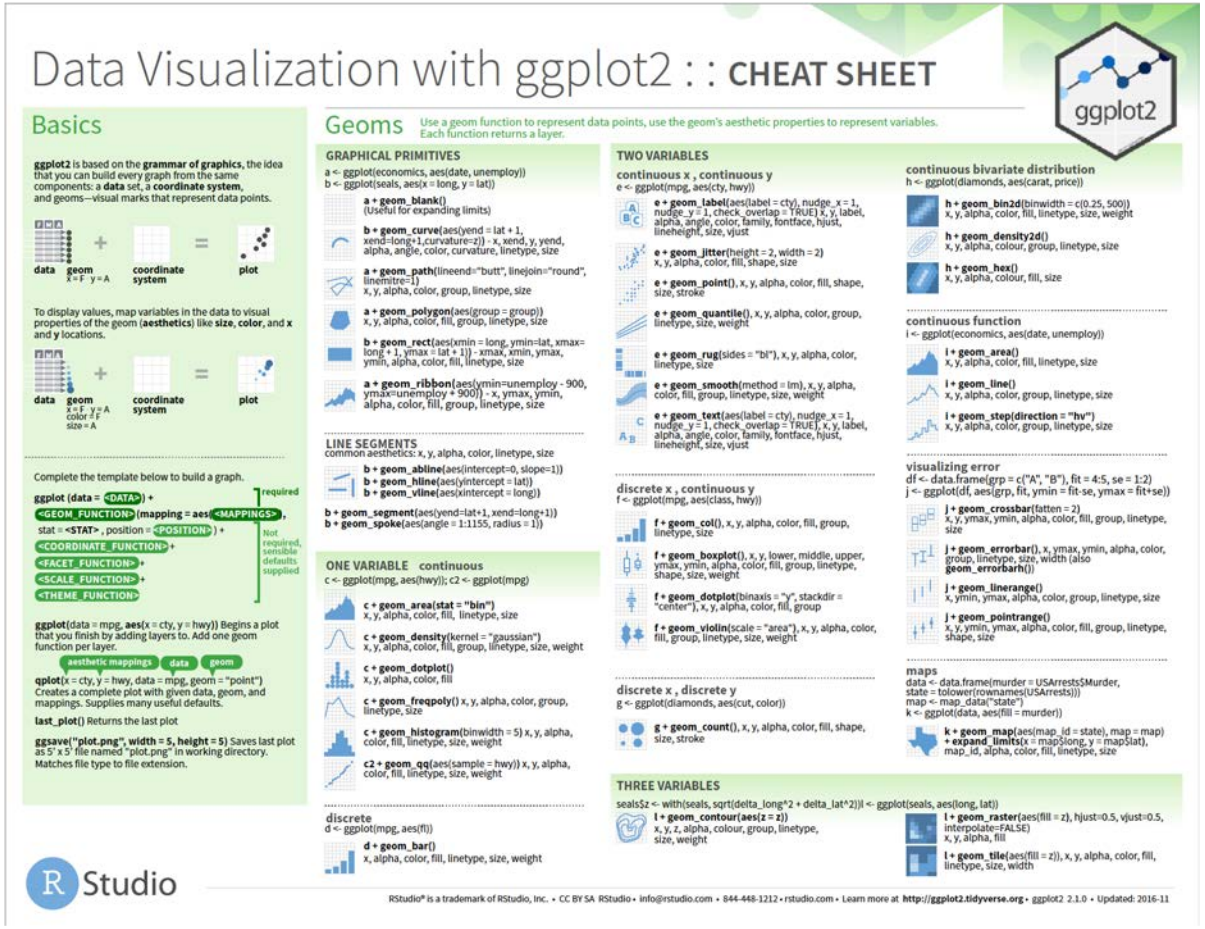
Gapminder Data

http://www.gapminder.org/


Averages across all years of the traditional Gapminder dataset
Average life expectancy at birth
80
70
60
50
40
1000
10000
Average GDP per capita (log 10)
United States
China
India
Average total population
7.5 million
75 million
750 million
Region
America
East Asia & Pacific
Europe & Central Asia
Middle East & North Africa
South Asia
Sub-Saharan Africa

Saving charts out

ggplot2 Cheat Sheet

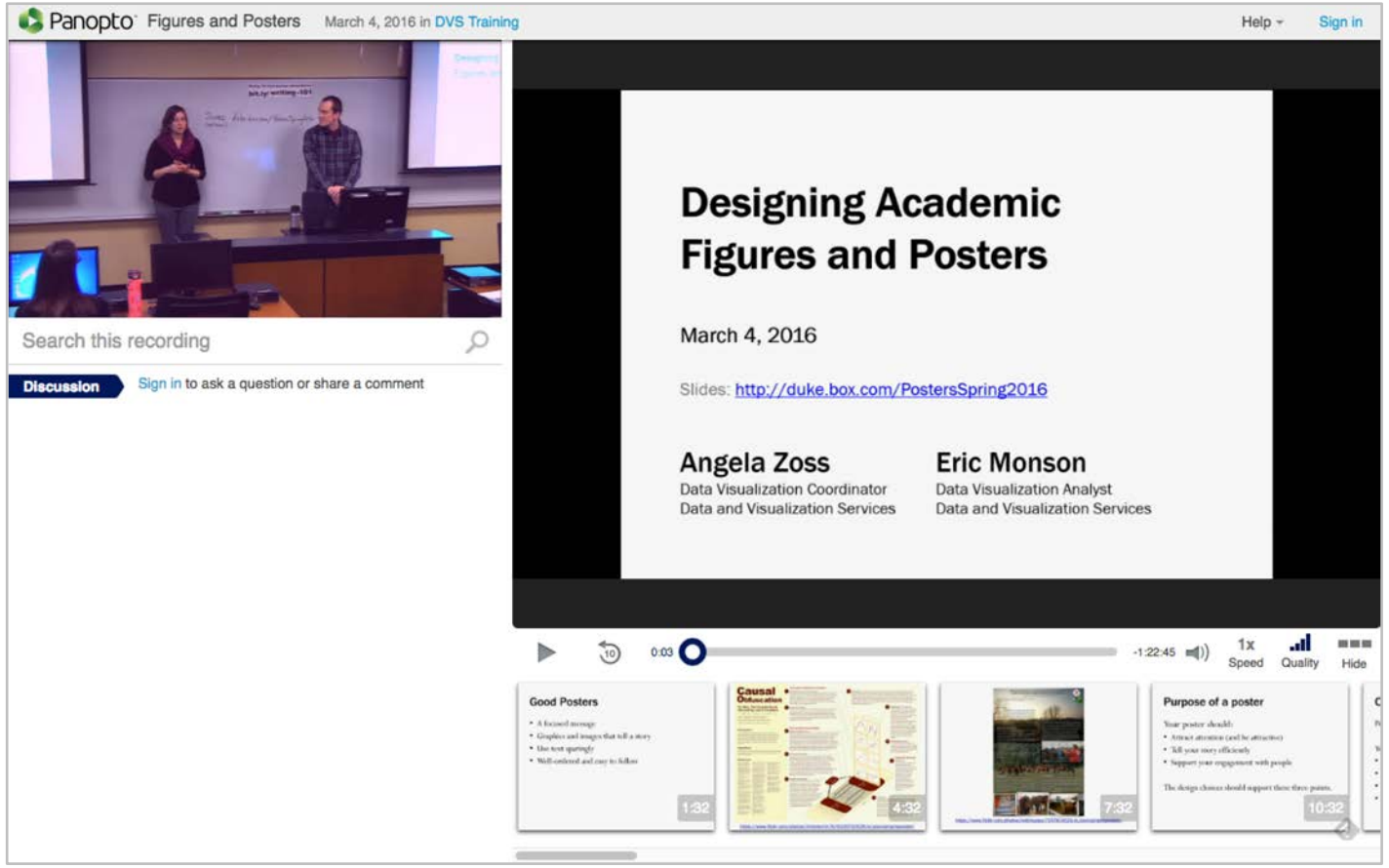


https://www.rstudio.com/resources/cheatsheets/#ggplot2

ggplot2 Resources

- General ggplot2 information
  http://ggplot2.tidyverse.org/
- R Graphics Cookbook (recipes for plots)
  http://www.cookbook-r.com/Graphs/index.html
- R for Data Science (online book that includes ggplot2)
  http://r4ds.had.co.nz/
- ggplot2: Elegant Graphs for Data Analysis (book by Hadley Wickham)
  http://ggplot2.org/book/
- ggplot2 cheatsheet (also in RStudio)
  http://bit.ly/ggplot2-cheatsheet

Videos of past workshops



http://bit.ly/DVSvideos

Questions?

angela.zoss@duke.edu