

AN INTRODUCTION TO R FOR POLICY ANALYSIS



COURSE SCHEDULE:

Webinar each Monday @ 10am (AEST) ~ 20 July to 17 August

- ~~• Week 1 ~ An introduction to programming, the R language and Rstudio~~
- ~~• Week 2 ~ Doing stuff with data - Importing, Exploring and Summarizing Data~~
- ~~• Week 3 ~ Making pixels pretty (1) - Data Cleaning, Merging and Basic Visualization~~
- **Week 4 ~ Making pixels pretty (2) - Data Cleaning and Visualization**
- Week 5 ~ Bringing together the R programming pipeline

WORKSHOP 3 OUTLINE:

- R in the Wild: Anya Cushnie Mills – M & E Advisor
- Why we viz – example
- Reviewing the basics
- Data viz in R – an applied introduction to ggplot2
- Policy Scenario: County Unemployment in the US
- Workshop evaluation



Why Use R?

Anya Cushnie Mills
PhD Student, Epidemiology and Public Health
Tampere University, Finland
Anya.Cushnie@gmail.com



- ✓ Used R exclusively for 5 years
- ✓ Open access – no license required
- ✓ All numerical analysis possible
- ✓ Huge online community
- ✓ Never have to memorize a code
- ✓ Writing code is just like writing a sentence
- ✓ Several packages to simplify tasks

- **Assessing HIV program outcomes for Jamaica before and after “Treat All”: A retrospective study using the national treatment services database.**

Anya V. Cushnie^{§1}, Ralf Reintjes^{1, 2¶}, Susanna Lehtinen-Jacks^{1¶}, J. Peter Figueroa^{3¶}

- 1 Unit of Health Sciences, Faculty of Social Sciences, Tampere University, Tampere, Finland
- 2 Department of Health Sciences, Hamburg University of Applied Sciences, Hamburg, Germany:
- 3 Department of Community Health and Psychiatry, University of the West Indies, Mona, Jamaica

Analysis was done using *R*, version 3.5.3 and the *finalfit* package was used to generate regression results and plots.

A LITTLE ABOUT ME

- Currently an economist at Cambridge Economic Policy Associates (CEPA).
- Have had a varied career prior to this:
 - ODI Fellowship at Vanuatu's Treasury
 - HM Treasury, UK
 - National Audit Office, UK
 - Department for Transport, UK (working on EU Exit, fun times...)

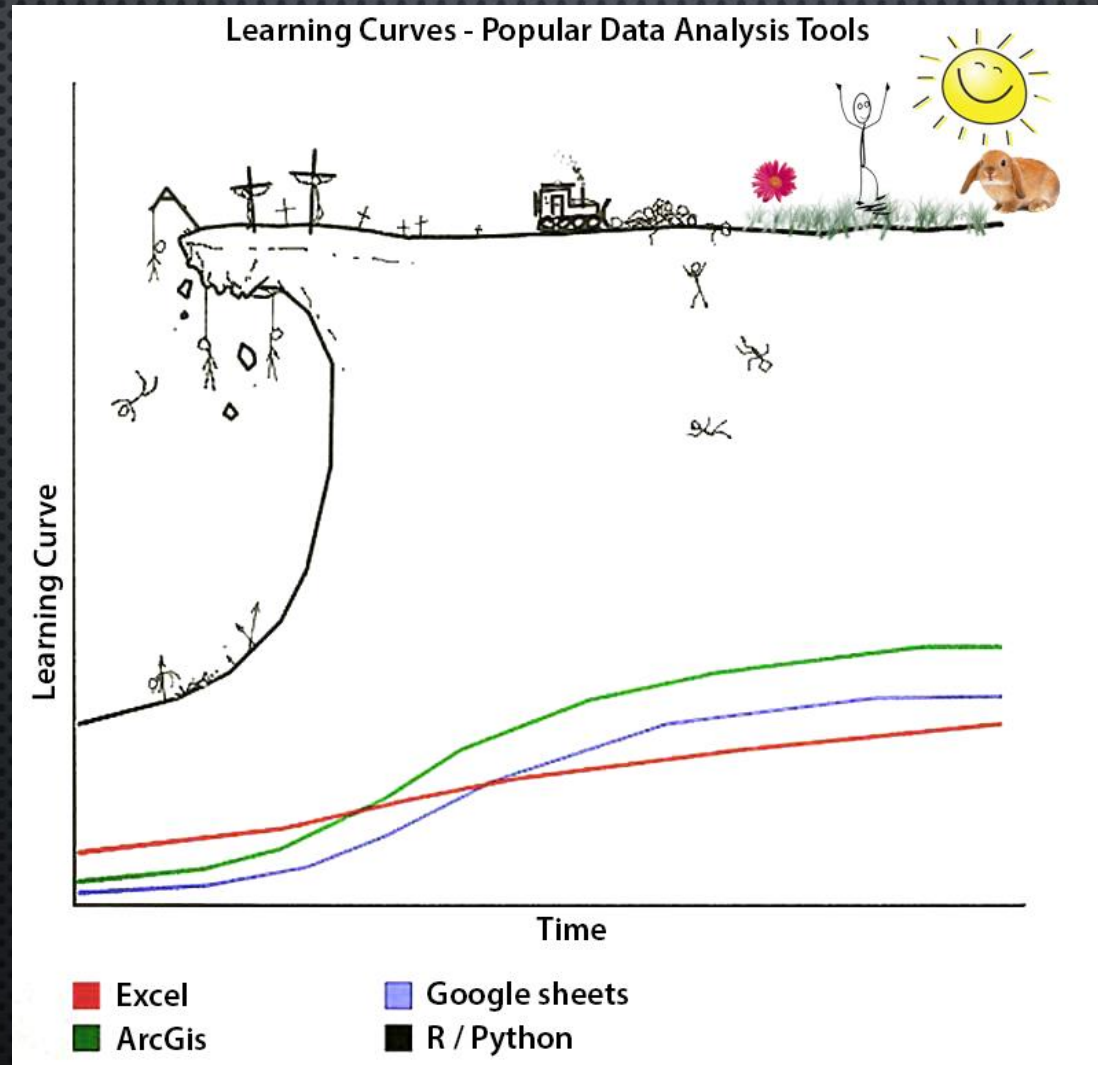
WHY I LEARN R

- It follows me where ever I go.
- It is easier than using Excel for complex tasks.
- It can do more than Excel.

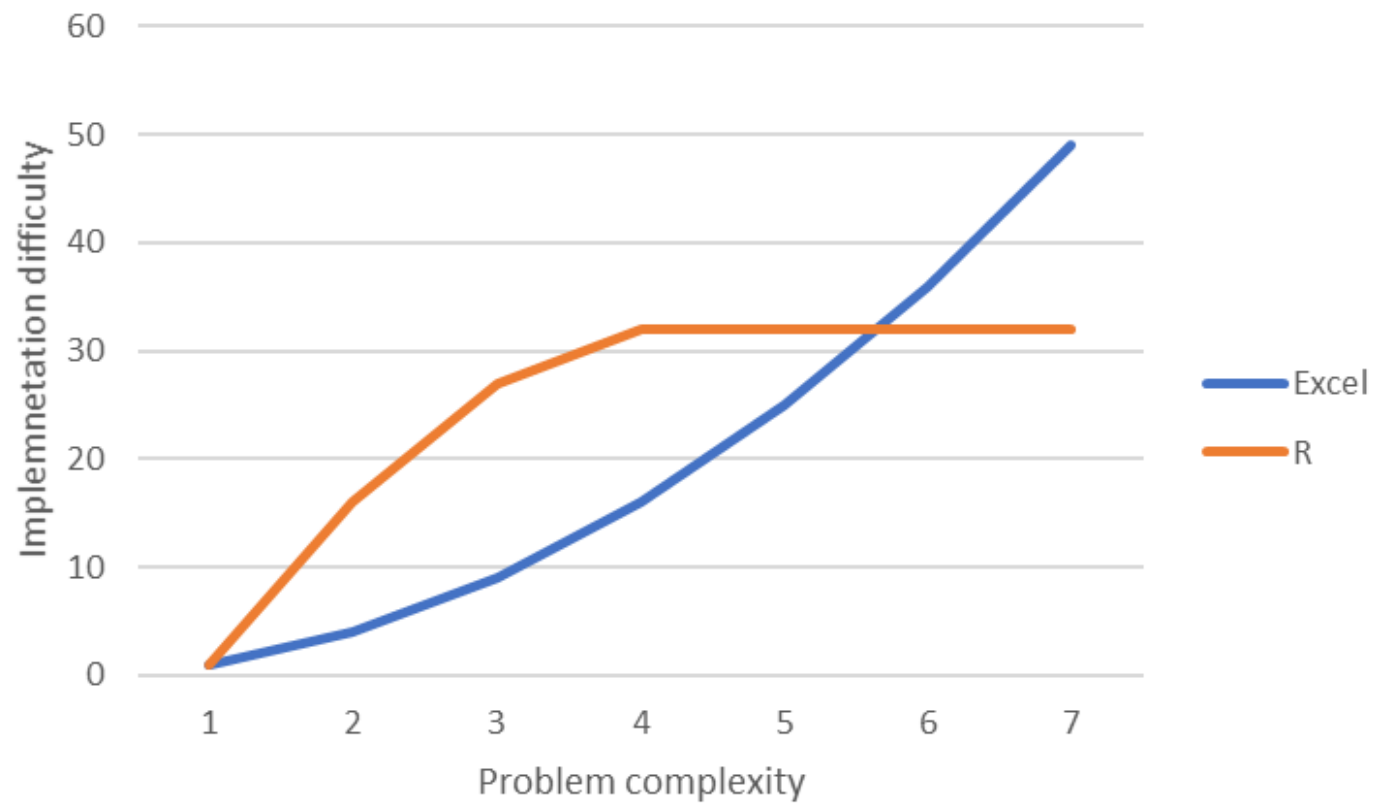
Link to my 'first' ever R output: [Map of Medical Facilities.html](#)

Link to something I made recently: <https://island-economist.shinyapps.io/sutherlandplanning/>

R VERSUS EXCEL



R VERSUS EXCEL



R VERSUS EXCEL

Average if versus median if

```
=AVERAGEIF(A1:A6,"Apple",B1:B6)
```

```
Data %>% group_by(variable) %>% summarise(mean = mean(variable2))
```

```
{=MEDIAN(IF($A$1:$A$6="Apple",IF($B$1:$B$6<>""),$B$1:$B$6)))}
```

```
Data %>% group_by(variable) %>% summarise(median = median(variable2))
```


WHY WE VIZ

Summary statistics alone can hide complexity

Good data viz also:

- Helps information be more quickly interpreted
- Highlights relationships, trends and patterns
- Allows us to focus on either the forest or the trees
- Helps us guide good policy by telling a 'data story'

R has a reputation for being able to generate an unlimited range of high quality graphics

REFRESHER : THE BASICS

COMMON ERRORS

Unmatched parenthesis: ie not including a close bracket to tell R the arguments passed to a function have been supplied and to apply the function – will tell you with ‘+’

Misplaced commas: eg not separating arguments by commas when applying a function

Applying functions to an incompatible element or data object type – eg trying to multiply text by 2

"cannot open" – attempts to read something that doesn't exist or isn't accessible (eg because the name or directory is wrong or R can't access it due to it being locked/open in another program).

"could not find function" – not loading the library needed for a function or misspelling the name of the function (eg Plot instead of plot).

"Error in eval" – cause by references to something that doesn't exist (eg misspelling an object name)

"no applicable method" – trying to apply a function to an object/element type that doesn't support it

"subscript out of bounds" – telling R to access data that doesn't exist/ is out of bounds (eg asking for row 34 in a 20 row vector)

When data/elements are the wrong type they can be converted using:

`as.data.frame()`, `as_tibble()`, `as_vector()`, `as_character()`, `as_numeric()`, etc

DPLYR VERBS

Provides a grammar for basic data manipulation:

- **select** – pick the columns you want
- **filter** – choose the relevant observations
- **mutate** – create new variables
- **group_by** – define groups you're interested in for summarise
- **summarise** – create summary statistics
- **arrange** – sort your data

enter '?summarise' to see help and a list of summary statistics

TIDY DATA:

Vehicle	Efficiency	Power	Year
Mazda RX4	21	110	2001
Mazda RX4 Wag	21	110	2015
Merc 230	22.8	95	2015
Merc 280	19.2	123	1983
Datsun 710	22.8	93	2017

- ✓ Each variable is a column.
- ✓ Each observation a row.
- ✓ Each experiment/survey a separate table

'TIDYR' CAN HELP YOU GET DATA IN THE RIGHT SHAPE

`pivot_longer()`



`pivot_wider()`



`separate()`



`unite()`



(This can also be useful for making data useful for pivot tables in excel)

MAKING DATA LONGER

`pivot_longer(data, cols)`

cols = columns to reshape/pivot from wide to long (eg a variable/measurement type).

eg: to aid time series analysis when years stored by column



MAKING DATA WIDER

```
pivot_wider(data, id_cols = NULL, names_from = 'name',  
values_from = 'value')
```

id_cols = columns that uniquely identifies observations

names_from = the name of the column(s) to make wide

values_from = the column where values are stored

eg to make variables 'long' so they're easier to access



SPLITTING COLUMNS

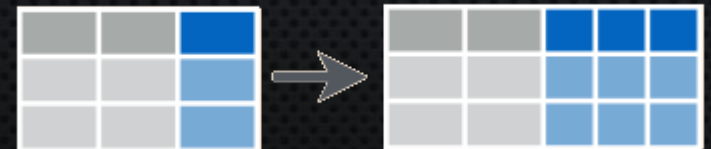
`separate(data,col,into,sep)`

col: column where data should be split

sep: character where data should be split

into: names of columns pieces to be split into

eg: to split first and last names into separate columns



MERGING MULTIPLE COLUMNS

`unite(data, col, ..., sep="_")`

col = name of the new column with merged data

... = columns to merge into one

sep = separator to use between values (defaults to "_")

eg: create a date from separate day/month/year columns



STACKING, BINDING AND JOINING DATA

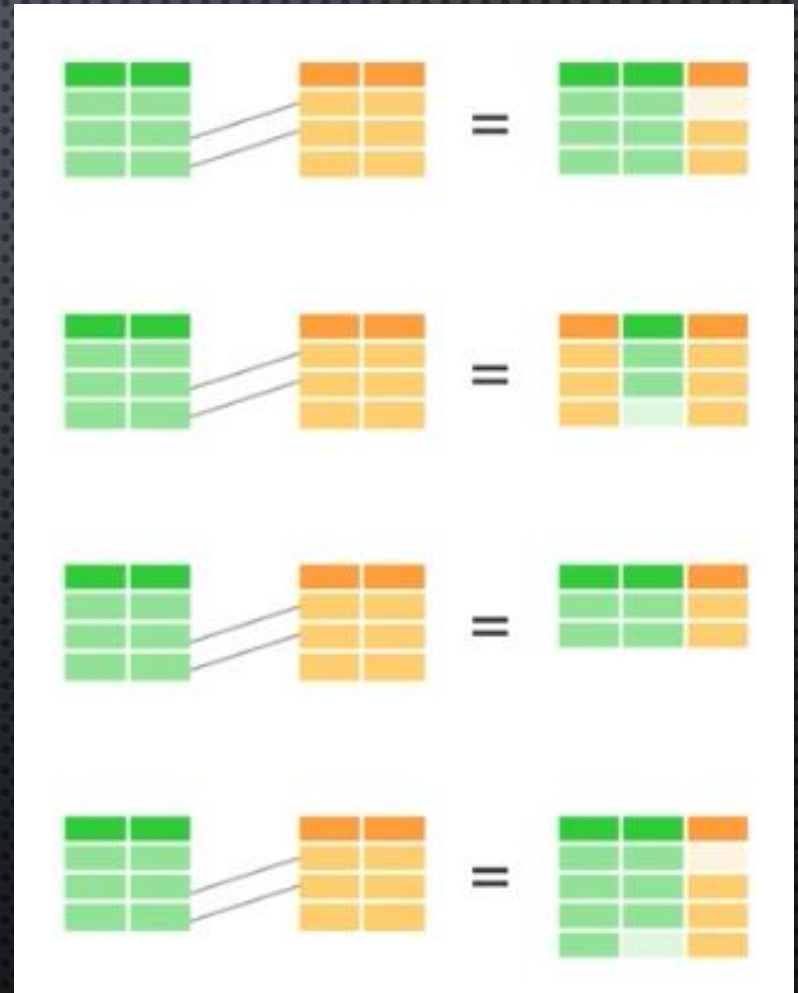
DPLYR JOINS:

`left_join(x,y)`

`right_join(x,y)`

`inner_join(x,y)`

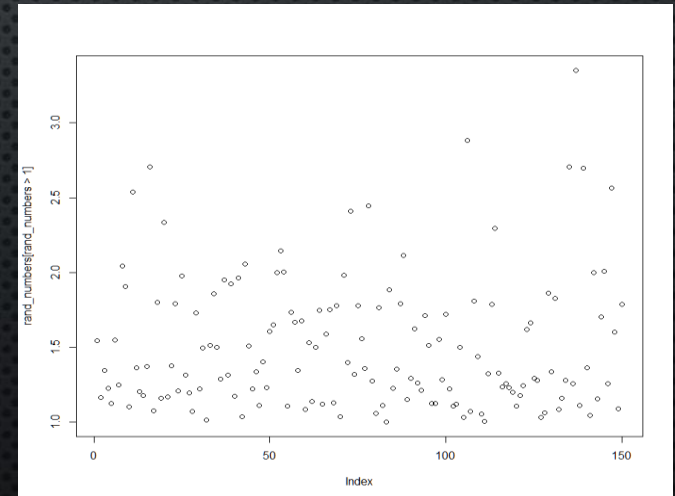
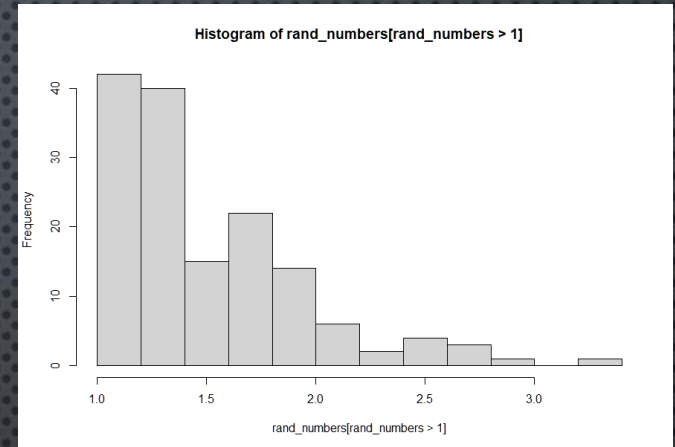
`full_join(x,y)`



BASE PLOTS

- Are built into R
- Are fast and useful
- Don't have to be ugly
- Uses commands, rather than using a 'grammar'
 - Eg `plot()`, `barplot()`, `hist()`, `plot.ts()` etc

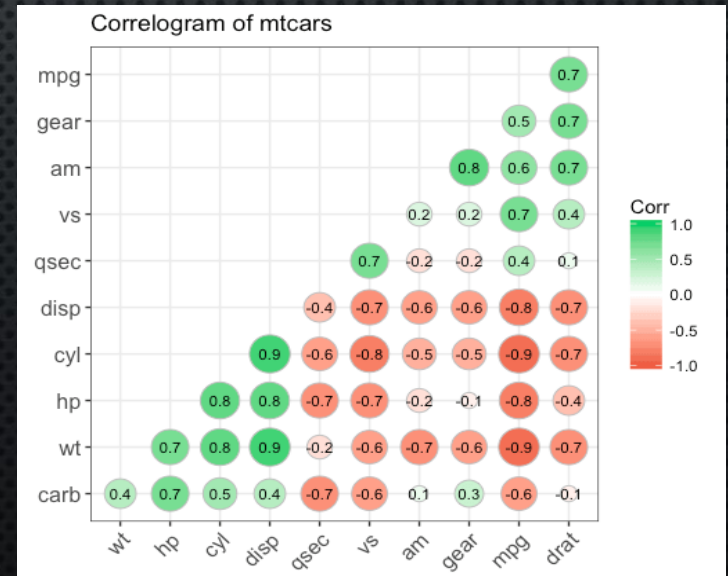
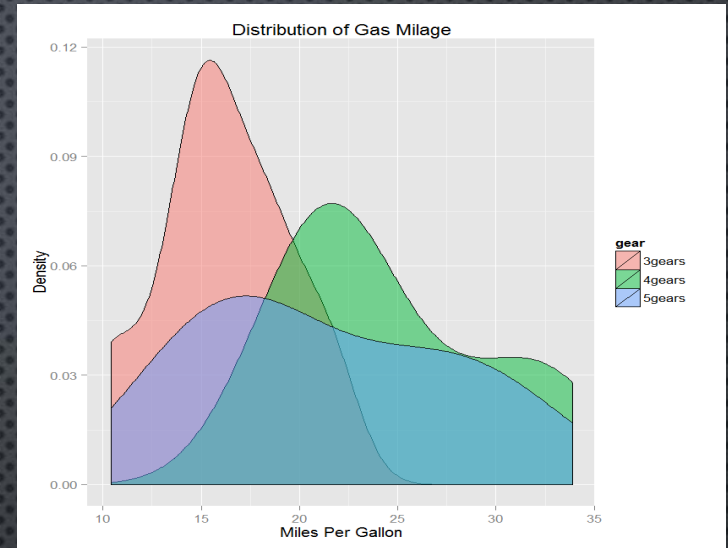
(Great for **exploratory analysis**)



GGPLOT:

- Requires the ggplot2 package
- Can be slower
- Tries to be prettier by default
- Has a consistent logic and 'grammar'

(Great for **explanatory analysis**)



GGPLOT2 – KEY GRAMMAR

GGLOT2 is based on the ‘grammar of graphics’, the idea that every graph is comprised of the same basic components

- Data;
- A geometry: ‘**geom**’ and coordinate system.
- Aesthetics: ‘**aes()**’.

(These concepts are outlined in the ggplot2 cheat sheet)

GGPLOT2 – KEY GRAMMAR

Geoms

Defining the geometry of the plot

Aesthetics

Defining the aesthetic mappings

Labels

Defines labelling

Themes

Changes how plots look

Facets

Dividing plots based on groups

GGPLOT2 – LAYERS ON A CANVAS

Define the canvas:

```
ggplot(data=titanic_data, aes(x=age))
```

Specify a geometry:

```
ggplot(data=titanic_data, aes(x=age)) +  
  geom_histogram()
```

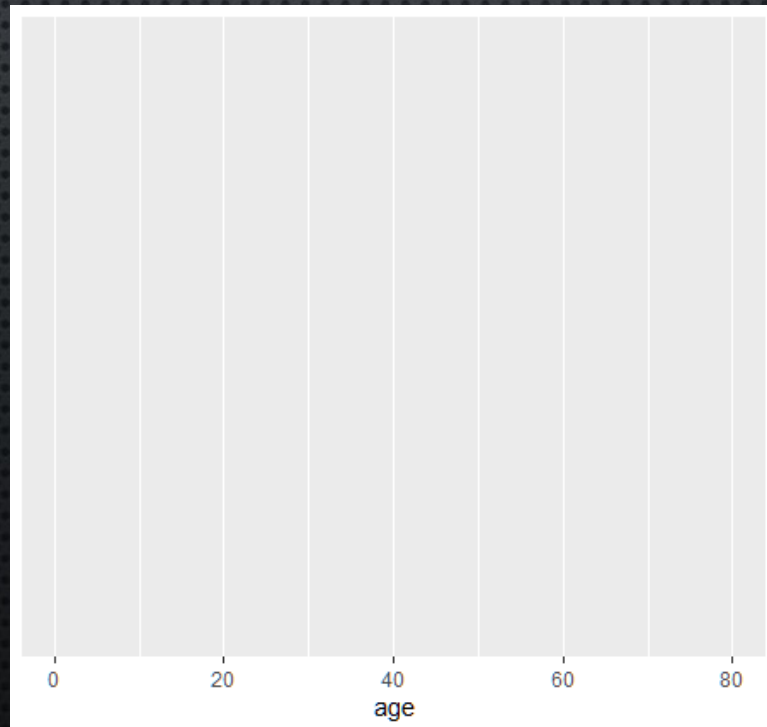
Add layers:

```
ggplot(data=titanic_data, aes(x=age)) +  
  geom_histogram() +  
  labs(y="A Histogram")
```

GGPLOT2 – LAYERS ON A CANVAS

Specify data and aesthetics:

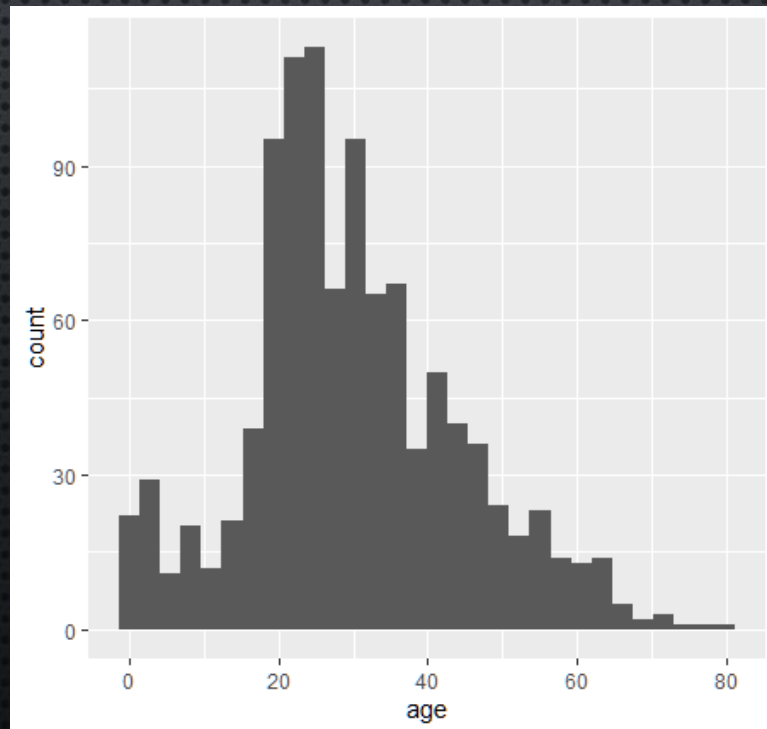
```
ggplot(data=titanic_data, aes(x=age))
```



GGPLOT2 – LAYERS ON A CANVAS

Add the geometry:

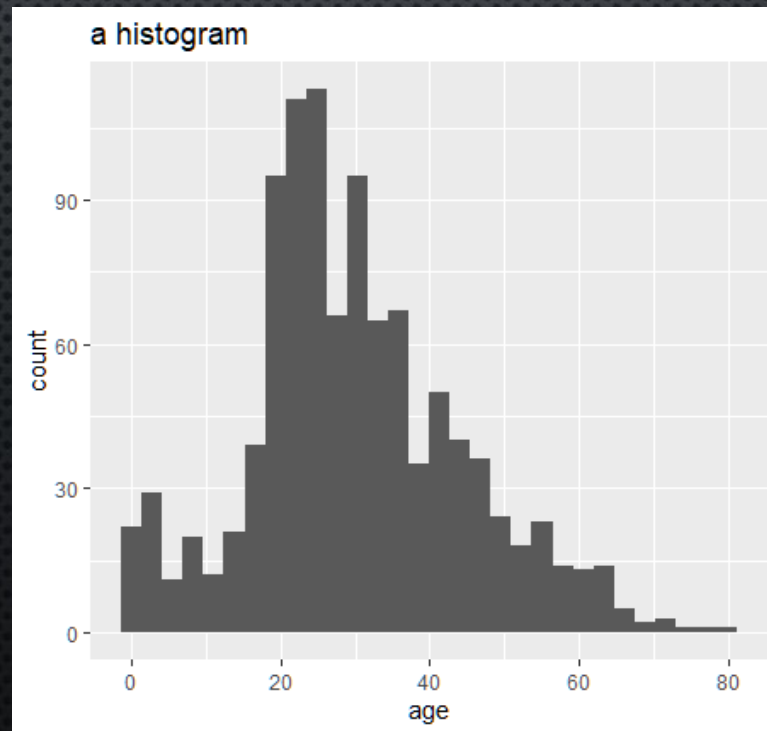
```
ggplot(data=titanic_data, aes(x=age)) +  
  geom_histogram()
```



GGPLOT2 – LAYERS ON A CANVAS

Add labels:

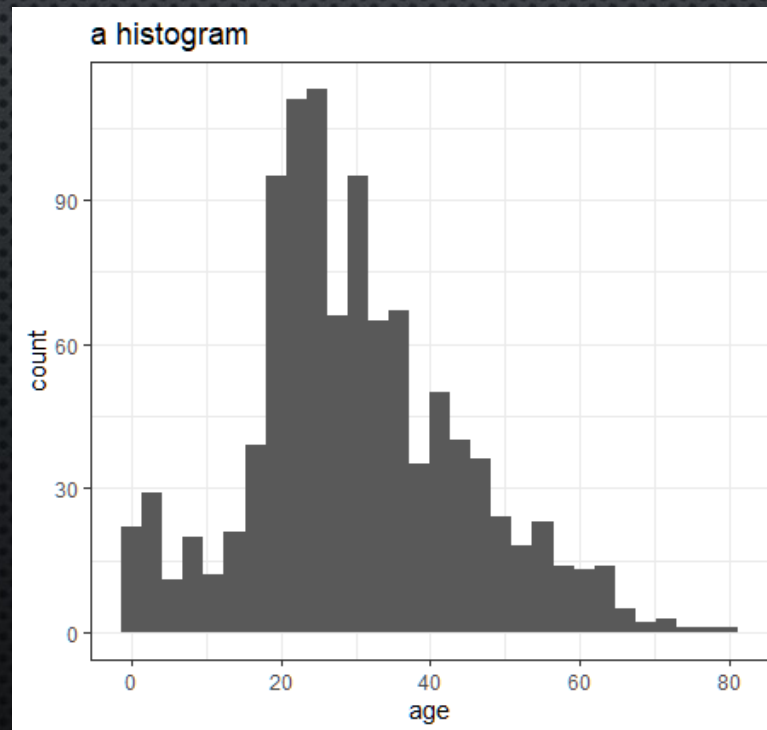
```
ggplot(data=titanic_data, aes(x=age)) + geom_histogram() +  
labs(title="a histogram")
```



GGPLOT2 – LAYERS ON A CANVAS

Give it a theme:

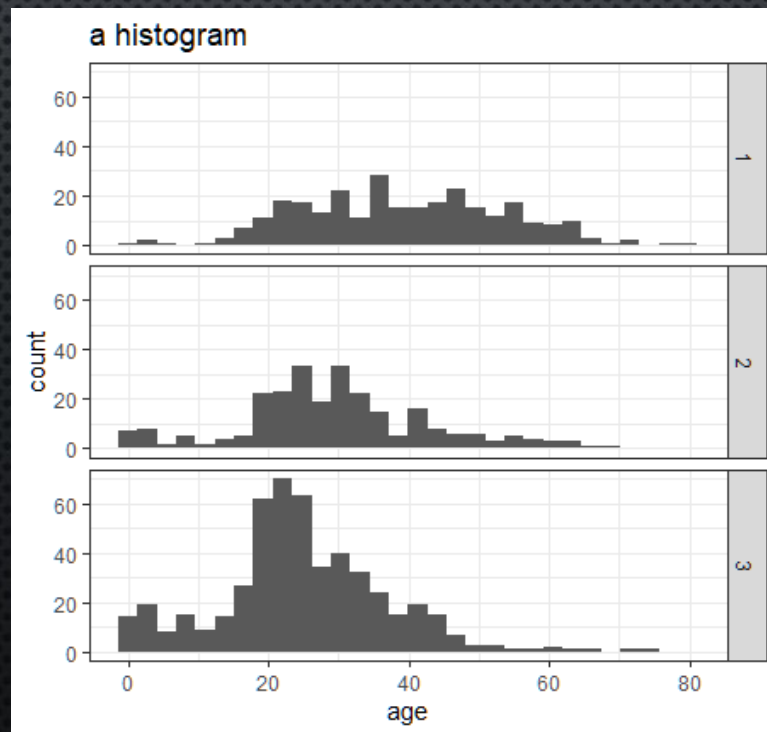
```
ggplot(data=titanic_data, aes(x=age)) + geom_histogram() +  
labs(title="a histogram") +  
theme_bw()
```



GGPLOT2 – LAYERS ON A CANVAS

Split by facets:

```
ggplot(data=titanic_data, aes(x=age)) + geom_histogram() +  
labs(title="a histogram") + theme_bw() +  
facets_grid(pclass ~ .)
```



GGPLOT2 – TIPS + COMMON ERRORS

- If your ggplot code goes over multiple lines make sure the + is on the line above.
- `geom_bar()` makes the height of the bar proportional to the count of the cases in each group. If you want to plot the values of the data use `geom_col()` instead.
- `geom_line()` connects observations in order of the variable on the x axis. If your line graph looks funny you probably meant to use `geom_path()` instead.
- If you want to 'zoom' to some part of your graph use the function `coord_cartesian()`. Using `xlim()` or `ylim()` will drop data from the plot.
- A common cause of issues is encoding character variables as factors. If something odd is happening check that your character variable isn't a factor.

POLICY SCENARIO: US POVERTY AND UNEMPLOYMENT

US COUNTY UNEMPLOYMENT AND POVERTY

Still convinced winning graphs will help them win the election, the President has asked for more help for another speech. In particular, the President has asked:

- To show how unemployment has changed when comparing states that voted Republican vs. Democrat in the presidential election
 - Does this change if we change the time period used?
 - What time period makes the most sense given the question?
- Which counties have contributed the most to recent increases in the number of unemployed in New England?
- Which states were hardest hit in terms of unemployment over the GFC?
 - Which counties contributed the most to this?

SUGGESTED SWIRL EXERCISES

- **Before next week complete**

- ~~• Subsetting Vectors~~

- ~~• Matrices and Data Frames~~

- ~~• Looking at Data~~

- ~~• Dates and Times~~

- **Base Graphics**

- **Logic**

- **Functions**



WEBINAR EVALUATION

(See chat for evaluation link)

AN INTRODUCTION TO R FOR POLICY ANALYSIS

