

Programming for Professional Research Using R

Session 3

November 9, 2023

Today

- Pop quiz
- Learn how to:
 - Create simple academic-standard regression output tables using the `stargazer` package
 - Create flexible and easy-to-read tables of any dataset using the `gt` package
 - Create a scatter plot, density plot, and bar chart using the `ggplot2` package
- Practice the above!

Pop Quiz!

<https://pollev.com/marcandreafiorina503>

Pop Quiz 'Tidy' Question

table_a

```
#> # A tibble: 6 x 3
#>   country      year rate
#> * <chr>      <int> <chr>
#> 1 Afghanistan 1999 745/19987071
#> 2 Afghanistan 2000 2666/20595360
#> 3 Brazil      1999 37737/172006362
#> 4 Brazil      2000 80488/174504898
#> 5 China       1999 212258/1272915272
#> 6 China       2000 213766/1280428583
```

table_b

```
#> # A tibble: 12 x 4
#>   country      year type      count
#>   <chr>      <int> <chr>      <int>
#> 1 Afghanistan 1999 cases         745
#> 2 Afghanistan 1999 population 19987071
#> 3 Afghanistan 2000 cases         2666
#> 4 Afghanistan 2000 population 20595360
#> 5 Brazil      1999 cases         37737
#> 6 Brazil      1999 population 172006362
#> # ... with 6 more rows
```

```
table_c
#> # A tibble: 6 x 4
#>   country      year  cases population
#>   <chr>      <int>  <int>      <int>
#> 1 Afghanistan 1999     745    19987071
#> 2 Afghanistan 2000    2666    20595360
#> 3 Brazil      1999   37737   172006362
#> 4 Brazil      2000   80488   174504898
#> 5 China       1999  212258  1272915272
#> 6 China       2000  213766  1280428583
```

Spread across two tibbles

```
table_d1 # cases
#> # A tibble: 3 x 3
#>   country `1999` `2000`
#> * <chr>   <int> <int>
#> 1 Afghanistan     745     2666
#> 2 Brazil          37737    80488
#> 3 China           212258   213766
```

```
table_d2 # population
#> # A tibble: 3 x 3
#>   country      `1999`      `2000`
#> * <chr>      <int>      <int>
#> 1 Afghanistan 19987071    20595360
#> 2 Brazil      172006362    174504898
#> 3 China       1272915272   1280428583
```

Data Visualization -- Simple Regression Table

Regression Tables

Regression tables are very common in economic/policy analysis.

They're very simple to create using R and a software called **LateX** (pronounced latek).

Unless you're getting into academic research, you don't need to know how to properly use LateX. Just enough to:

- Export the LateX script from R
- Copy/paste it into a LateX-reading software, e.g. Overleaf
- Export the pdf or png to share

Predicted Consumption per Capita (2019 PPP USD)

	Any Treatment vs. Control (1)	Women Working Treatment vs. Any Treatment (2)
Any Treatment	12.049** (5.330)	12.155* (6.600)
Women Working Treatment		-0.222 (8.463)
Baseline Control	0.249** (0.101)	0.249** (0.101)
Constant	22.788*** (3.483)	22.791*** (3.489)
Control Mean	27.91	27.91
Observations	761	761
R ²	0.028	0.028
Adjusted R ²	0.025	0.024
Residual Std. Error	44.983 (df = 758)	45.013 (df = 757)
F Statistic	10.925*** (df = 2; 758)	7.275*** (df = 3; 757)
Note:	*p<0.1; **p<0.05; ***p<0.01	

Regression Table -- Step by Step

Run Regression in R

Convert to Exportable Table

```
# Simplest regression format in R

reg_example <- lm(
  outcome_variable ~ independent_variable + control_variables,
  data = dataset
)

# Observe results

reg_example %>% summary()
```


Regression Table -- Step by Step

Run Regression in R

Convert to Exportable Table

Simply do one of these!

```
reg_example_ht <- reg_example %>%  
  huxtable::huxreg()
```

OR

```
reg_example_sg <- reg_example %>%  
  stargazer::stargazer() # Many options to make prettier
```

Regression Table -- Step by Step

Export Huxtable Table

Export Stargazer Table

Some simple options for the Huxtable table:

```
huxtable::quick_latex(  
  reg_example_ht,  
  file = "filepath/filepath/filepath/reg_example_ht.tex"  
)  
  
huxtable::quick_pdf(  
  reg_example_ht,  
  file = "filepath/filepath/filepath/reg_example_ht.pdf"  
)  
  
huxtable::quick_html(  
  reg_example_ht,  
  file = "filepath/filepath/filepath/reg_example_ht.html"  
)
```

Regression Table -- Step by Step

Export Huxtable Table

Export Stargazer Table

You can export a LaTeX script using the 'writeLines' function

```
writeLines(  
  reg_example_sg,  
  "filepath/filepath/filepath/reg_example_sg.tex"  
)
```

To visualize your table, the easiest solution is to:

- Create a free Overleaf account on overleaf.com
- Open a new document
- Copy/paste your .tex output in between the `begin{document}` and `end{document}` lines
- Click compile and then save!

You can also install the `tinytex` package and use `pdftolatex` to save a PDF file.

Data Visualization -- Descriptive Statistics -- Tables

Descriptive Statistics Tables

Thankfully, not every R position requires academic-standard tables or use of LaTeX.

It is still useful, however, to be able to communicate descriptive statistics about data.

GEWE El Salvador Baseline							
Module L -- Female HoH -- Time Use -- Weekdays							
	Mean (SD)	Median (Q1, Q3)	Min - Max	# Obs (% Group)	# in Group		
Daily Time Spent (Hours)							
Agriculture (Household)	0.24 (0.81)	0 (0, 0)	0 - 10	1,275 (100%)	1,275		
Childcare	0.97 (1.49)	0 (0, 1.6)	0 - 14.25	1,275 (100%)	1,275		
Chores	4.76 (2.54)	4.58 (3, 6.48)	0 - 13.75	1,275 (100%)	1,275		
Collecting Water	0.17 (0.48)	0 (0, 0)	0 - 4.25	1,275 (100%)	1,275		
Collecting Wood	0.2 (0.54)	0 (0, 0)	0 - 4.42	1,275 (100%)	1,275		
Eating	2 (1.06)	1.75 (1.29, 2.5)	0 - 9.25	1,275 (100%)	1,275		
Leisure/Religion	2.39 (2.04)	2 (0.75, 3.5)	0 - 14.5	1,275 (100%)	1,275		

There are countless R packages to help do this. Today, we're looking at the `gt` package. It's simple to use and it's very easy to create good-looking tables using it.

`gt` exports into `.png`, `.pdf`, or `.html`. You can add interactive elements, plots within columns.

Descriptive Statistics Table -- Step-by-Step

We will mainly use the example in the script for this. To summarize, the steps are:

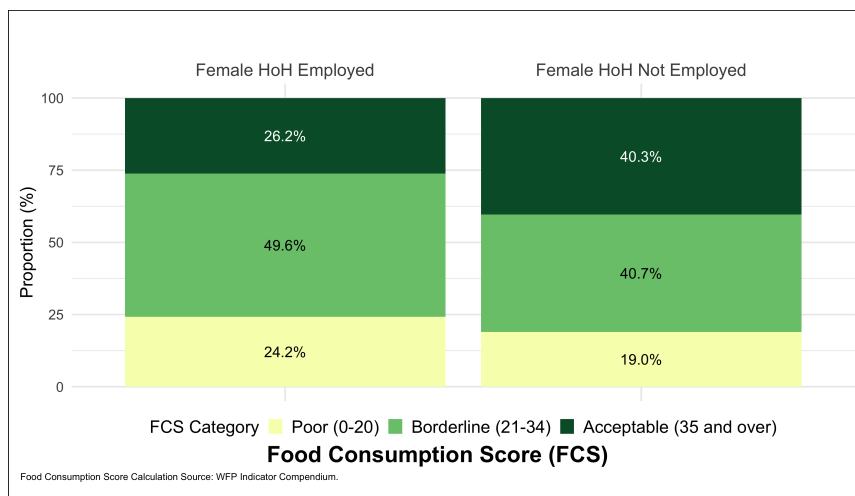
- Create a dataset you want to export
- Run the dataset through the `gt()` function to create a `gt` object
- Customize the table using functions from the `gt` package (see online for further things you can do). Examples of what you can do include:
 - Modify column names -- `cols_label()`
 - Modify borders -- `tab_style()`, `cell_borders()`
 - Add colors conditional on cell value -- `data_color()`
 - Add title/subtitle -- `tab_header()`
- Export the table using `gtsave()`

Data Visualization -- Descriptive Statistics -- Plots

Descriptive Stats Plots

`ggplot2` is the gold standard in data visualization in data work. It's one of the main reason that people use R over other programming languages.

Very simple syntax and allows you to add elements very easily.



You can use `ggplot2` to create any type of plot you can think of.

I've included a lot of links at the end of these slides to explore the possibilities of `ggplot2` further. Strongly recommend you use them or at least save them somewhere.

Descriptive Statistics Table -- Step-by-Step

Dataset	Convert to Plot	Add Something	Make It More Interesting
---------	-----------------	---------------	--------------------------

Start with a dataset you want to visualize

```
mtcars %>% head()
```

##		mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
##	Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
##	Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
##	Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
##	Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
##	Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
##	Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Descriptive Statistics Table -- Step-by-Step

Dataset

Convert to Plot

Add Something

Make It More Interesting

```
mtcars %>%  
  ggplot()
```

Descriptive Statistics Table -- Step-by-Step

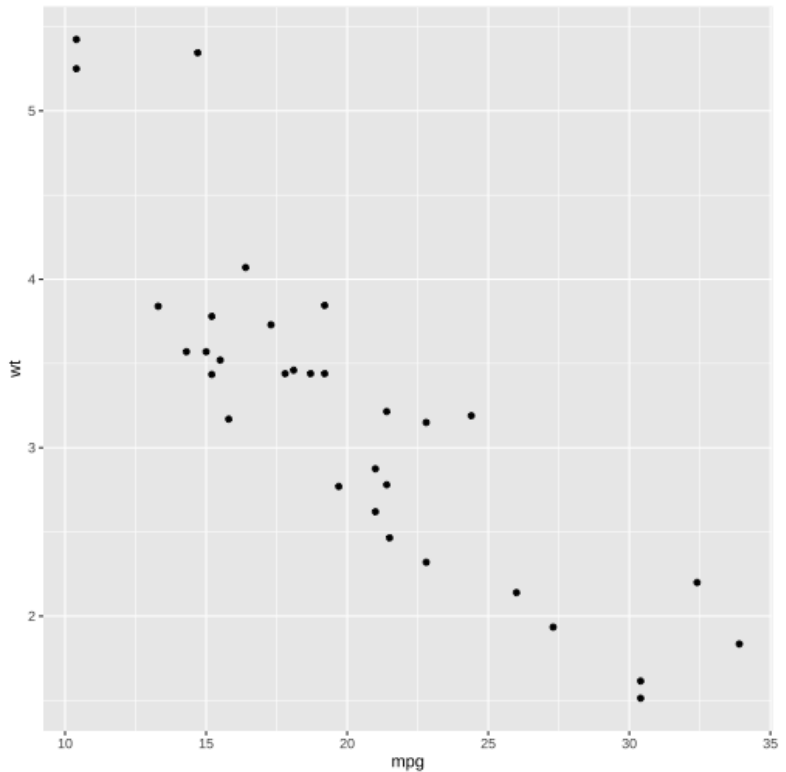
Dataset

Convert to Plot

Add Something

Make It More Interesting

```
mtcars %>%  
  ggplot() +  
  geom_point(  
    aes(x = mpg, y = wt)  
  )
```



Descriptive Statistics Table -- Step-by-Step

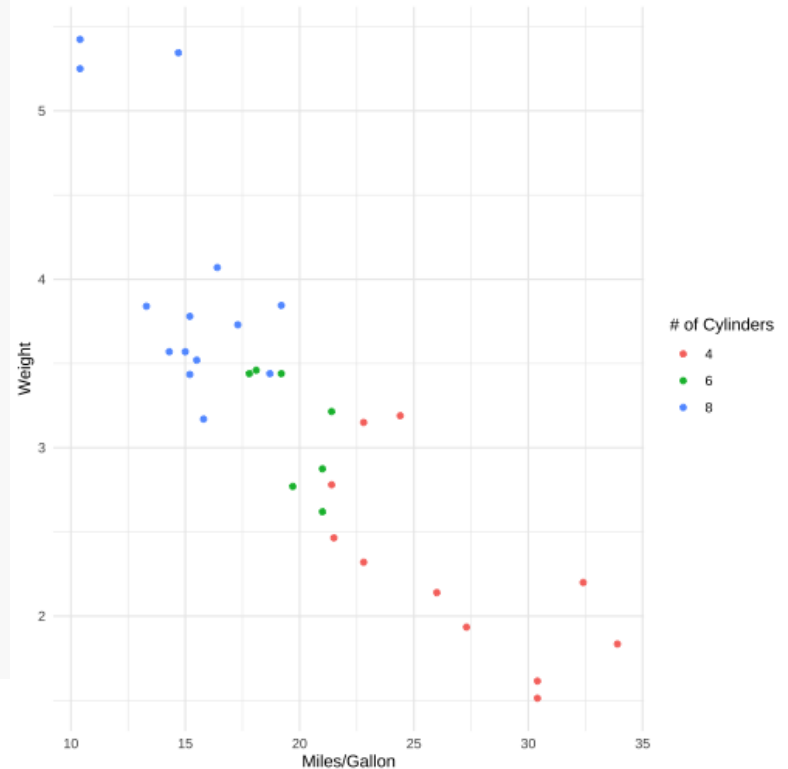
Dataset

Convert to Plot

Add Something

Make It More Interesting

```
mtcars %>%  
  ggplot() +  
  geom_point(  
    aes(  
      x = mpg, y = wt,  
      color = factor(cyl)  
    )  
  ) +  
  xlab("Miles/Gallon") +  
  ylab("Weight") +  
  scale_color_discrete(  
    name = "# of Cylinders"  
  ) +  
  theme_minimal()
```



Practical Exercise -- Using the World Values Survey Dataset

World Values Survey

Background

"The survey, which started in 1981, seeks to use the most rigorous, high-quality research designs in each country. The WVS consists of nationally representative surveys conducted in almost 100 countries which contain almost 90 percent of the world's population, using a common questionnaire. [...] WVS seeks to help scientists and policy makers understand changes in the beliefs, values and motivations of people throughout the world."

Survey Contents

- Social values, attitudes & stereotypes
- Societal well-being
- Social capital, trust and organizational membership
- Economic values
- Corruption
- Migration
- Post-materialist index
- Science & technology
- Religious values
- Security
- Ethical values & norms
- Political interest and political participation
- Political culture and political regimes
- Demography

Today's practical component

1. Successfully run the code in the `session_3_template.R` script
2. Attempt the challenges at the bottom of the script!
3. Optional assignment: complete as many of the challenges as you'd like and submit to me for review in the next two weeks

Links

Syllabus:

https://mfiorina.github.io/sais_r_course/syllabus/r_course_syllabus.html

Session 1: https://mfiorina.github.io/sais_r_course/session_1/session_1.html

Session 2: https://mfiorina.github.io/sais_r_course/session_2/session_2.html

Tables

Marek Hlavac, [“stargazer: beautiful LATEX, HTML and ASCII tables from R statistical output”](#)

Thomas Mock, [“gt - a \(G\)rammar of \(T\)ables”](#)

Plots

Alicia Horsch, [“A quick introduction to ggplot2”](#)

RStudio, [RStudio Cheatsheets](#)

Explore R Further

More Complex Data Manipulation

Iterative coding (using loops for repetitive code) – purrr’s map function is your friend. I recommend Thomas Mock, [“Functional programming in R with Purrr”](#) to get you started.

User-made functions in R – At some point, it will become time-effective to create your own functions to apply to your work. Hadley Wickham, [Advanced R Chapter 6 – Functions](#).

Publishing Your R Work

RStudio, [“Introduction to RMarkdown”](#). Summarizes the uses and utility of the RMarkdown framework.

Yihui Xie, [“xaringan Presentations” – book chapter](#) and [presentation](#).
Introduction to xaringan , a package that allows you to create slide decks using R. Also explore the [xaringanExtra package](#).

With RMarkdown, create books using [bookdown](#) or a blog using [blogdown](#).

Explore R Further

Data Visualization Using Plots

The R community organizes **“Tidy Tuesday”**. This is a weekly challenge where users are provided a dataset and participants then swap graphs and scripts used to create their visualizations.

David Robinson’s **Tidy Tuesday live screencasts** on YouTube. The perfect resource to follow along and try to replicate a professional coder’s scripts.

Yan Holtz and Conor Healy, **“From Data to Viz”**. An amazing repository of methods to create different data visualizations using R.

Explore R Further

Geospatial Data Visualization

Robin Lovelace, Jakub Nowosad, and Jannes Muenchow, **Geocomputation with R**. A great introduction to manipulating geospatial data (shapefiles and rasters) in R.

Edzer Pebesma, **“Simple Features for R”**. An introduction to the sf package, commonly used for geospatial work in R.

Edzer Pebesma, **“Plotting Simple Features”**. How to use sf and ggplot2 to visualize data using maps.

Explore R Further

For those interested in conducting data work in the development world: Kristoffer Bjørkefur, Luiza Cardoso de Andrade, Benjamin Daniels, and Maria Ruth Jones, **Development Research in Practice – The DIME Analytics Data Handbook**. A comprehensive account of tools and instruments to conduct quantitative development research.

For those looking for more hands-on, real-world data work: Ben Baldwin, **“A beginner’s guide to nflfastR”**. How to download and explore NFL play-by-play data. This is how I learnt how to use R. Further tutorials using this data can be found at the **“Open Source Football” blog**.