tigeRs: Princeton's R Group

Kim Kreiss & Angela Li

Workshop 1

Introduction

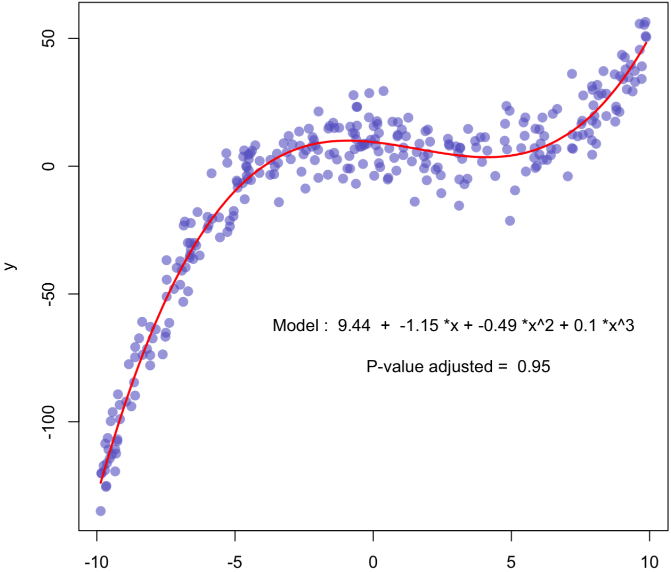
R/RStudio/RMarkdown and Why?

- R is a statistical programming language great for data analysis and data science applications
- RStudio is an Integrated Development Environment (IDE)–basically just a nice interface for using R, writing/running code, and interacting with data and files
- RMarkdown is a file format that lets you combine R code, data, and text that outputs a document, report, slideshow, etc.

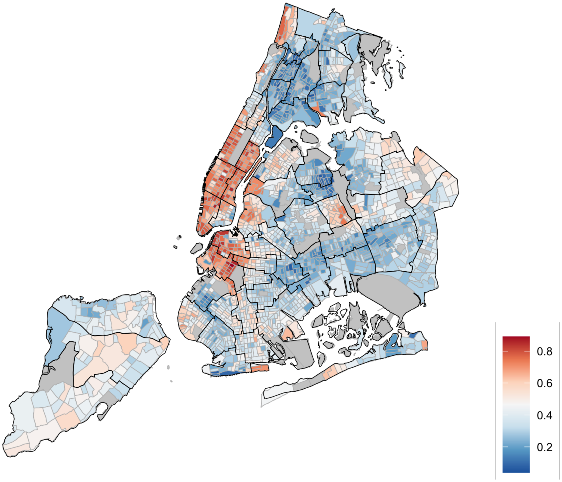
Cool things you can do with R

- Easy and intuitive management of code and data
- Excellent visualization capabilities, including charts, maps, interactive dashboards etc.
- Easily output analysis into a digestible format
- can handle a wide arrange of statistical analysis, data analysis, and data science applications

Some examples



Some examples



Share of Creative Class Workers in NYC Census Tracts

(by PUMA/Community District)

Source: 2019 American Community Survey 5-year estimates
Tracts with less than 30 people are shaded grey.

Some examples

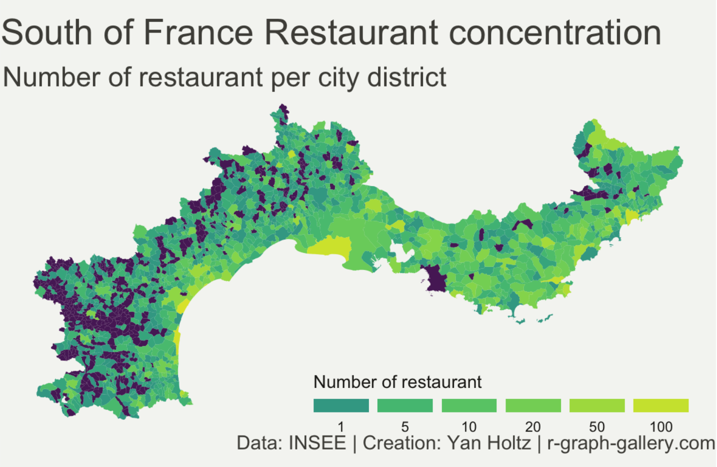

South of France Restaurant concentration
Number of restaurant per city district
Number of restaurant
1
5
10
20
50
100
Data: INSEE | Creation: Yan Holtz | r-graph-gallery.com

Figure 2: 'Source: r-graph-gallery.com'

Some examples

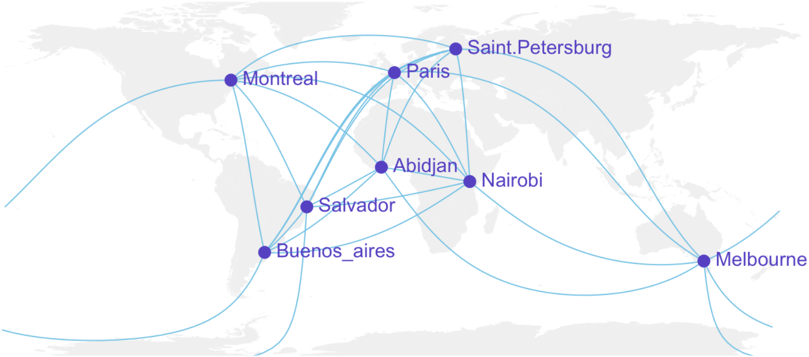


Figure 3: ‘Source: r-graph-gallery.com’

Some examples

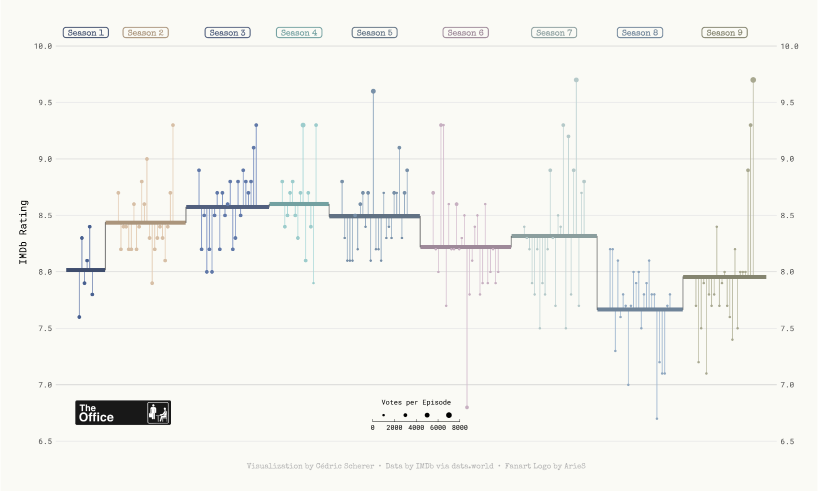
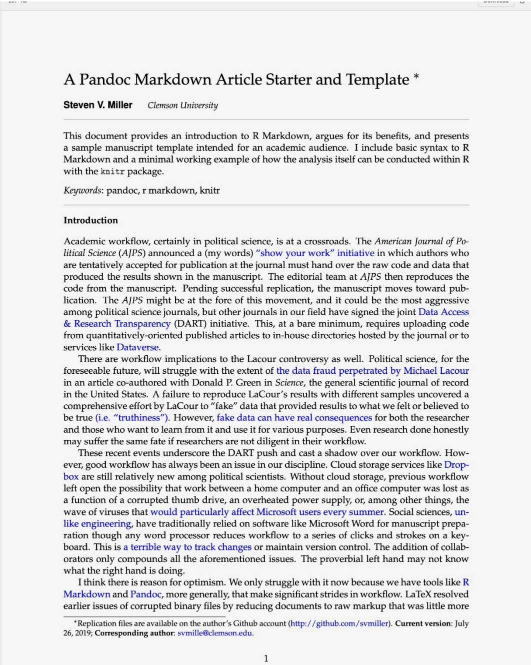


Figure 4: ‘Source: r-graph-gallery.com’

Some examples



A Pandoc Markdown Article Starter and Template [*]

Steven V. Miller *Clemson University*

This document provides an introduction to R Markdown, argues for its benefits, and presents a sample manuscript template intended for an academic audience. I include basic syntax to R Markdown and a minimal working example of how the analysis itself can be conducted within R with the `knitr` package.

Keywords: pandoc, r markdown, knitr

Introduction

Academic workflow, certainly in political science, is at a crossroads. The *American Journal of Political Science* (*AJPS*) announced a (my words) "show your work" initiative in which authors who are tentatively accepted for publication at the journal must hand over the raw code and data that produced the results shown in the manuscript. The editorial team at *AJPS* then reproduces the code from the manuscript. Pending successful replication, the manuscript moves toward publication. The *AJPS* might be at the fore of this movement, and it could be the most aggressive among political science journals, but other journals in our field have signed the joint Data Access & Research Transparency (DART) initiative. This, at a bare minimum, requires uploading code from quantitatively-oriented published articles to in-house directories hosted by the journal or to services like Dataverse.

There are workflow implications to the Lacour controversy as well. Political science, for the foreseeable future, will struggle with the extent of the data fraud perpetrated by Michael Lacour in an article co-authored with Donald P. Green in *Science*, the general scientific journal of record in the United States. A failure to reproduce LaCour's results with different samples uncovered a comprehensive effort by LaCour to "fake" data that provided results to what we felt or believed to be true (i.e. "truthiness"). However, fake data can have real consequences for both the researcher and those who want to learn from it and use it for various purposes. Even research done honestly may suffer the same fate if researchers are not diligent in their workflow.

These recent events underscore the DART push and cast a shadow over our workflow. However, good workflow has always been an issue in our discipline. Cloud storage services like Dropbox are still relatively new among political scientists. Without cloud storage, previous workflow left open the possibility that work between a home computer and an office computer was lost as a function of a corrupted thumb drive, an overheated power supply, or, among other things, the wave of viruses that would particularly affect Microsoft users every summer. Social sciences, unlike engineering, have traditionally relied on software like Microsoft Word for manuscript preparation though any word processor reduces workflow to a series of clicks and strokes on a keyboard. This is a terrible way to track changes or maintain version control. The addition of collaborators only compounds all the aforementioned issues. The proverbial left hand may not know what the right hand is doing.

I think there is reason for optimism. We only struggle with it now because we have tools like R Markdown and Pandoc, more generally, that make significant strides in workflow. LaTeX resolved earlier issues of corrupted binary files by reducing documents to raw markup that was little more

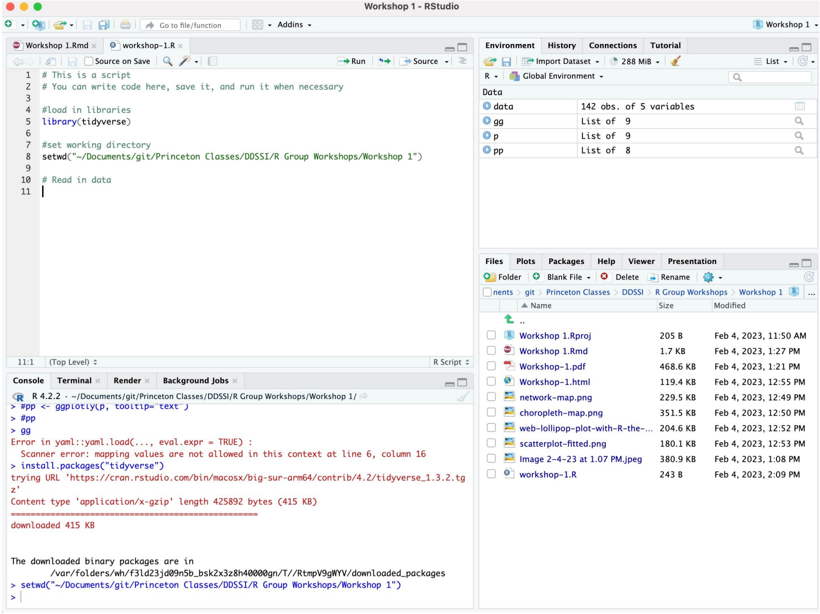
[*] Replication files are available on the author's Github account (http://github.com/svmiller). **Current version**: July 26, 2019; **Corresponding author**: svmille@clemson.edu.

Goals

Today's workshop

- Learn the basics and fundamentals of working in RStudio and RMarkdown
- Learn a framework for reproducible analysis that can be applied to homework, work assignments, etc.
- Go through an analysis together to combine these
- Leave with a file that can serve as a template for future work

RStudio



```r
# This is a script
# You can write code here, save it, and run it when necessary

#load in libraries
library(tidyverse)

#set working directory
setwd("~/Documents/git/Princeton Classes/DDSSI/R Group Workshops/Workshop 1")

# Read in data

```

11:1 (Top Level) R Script

Data

data	142 obs. of 5 variables
gg	List of 9
p	List of 9
pp	List of 8

R 4.2.2 · ~/Documents/git/Princeton Classes/DDSSI/R Group Workshops/Workshop 1/

```
> #pp <- ggplotly(p, tooltip="text")
> #pp
> gg
Error in yaml::yaml.load(..., eval.expr = TRUE) :
  Scanner error: mapping values are not allowed in this context at line 6, column 16
> install.packages("tidyverse")
trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.2/tidyverse_1.3.2.tg
z'
Content type 'application/x-gzip' length 425892 bytes (415 KB)
==================================================
downloaded 415 KB


The downloaded binary packages are in
	/var/folders/wh/f3ld23jd09n5b_bsk2x3z8h40000gn/T//RtmpV9gWYV/downloaded_packages
> setwd("~/Documents/git/Princeton Classes/DDSSI/R Group Workshops/Workshop 1")
> 
```

nents > git > Princeton Classes > DDSSI > R Group Workshops > Workshop 1

Name	Size	Modified
..		
Workshop 1.Rproj	205 B	Feb 4, 2023, 11:50 AM
Workshop 1.Rmd	1.7 KB	Feb 4, 2023, 1:27 PM
Workshop-1.pdf	468.6 KB	Feb 4, 2023, 1:21 PM
Workshop-1.html	119.4 KB	Feb 4, 2023, 12:55 PM
network-map.png	229.5 KB	Feb 4, 2023, 12:49 PM
choropleth-map.png	351.5 KB	Feb 4, 2023, 12:50 PM
web-lollipop-plot-with-R-the-…	204.6 KB	Feb 4, 2023, 12:52 PM
scatterplot-fitted.png	180.1 KB	Feb 4, 2023, 12:53 PM
Image 2-4-23 at 1.07 PM.jpeg	380.9 KB	Feb 4, 2023, 1:08 PM
workshop-1.R	243 B	Feb 4, 2023, 2:09 PM

R Scripts and R Markdown

- R Script (workshop-1.R)
  - A file that only runs normal R code
- RMarkdown Script (workshop-1.Rmd)
  - A different file type that combines code and text to produce a document that includes both
- Both a .R and .Rmd file are provided

Reproducible Analysis

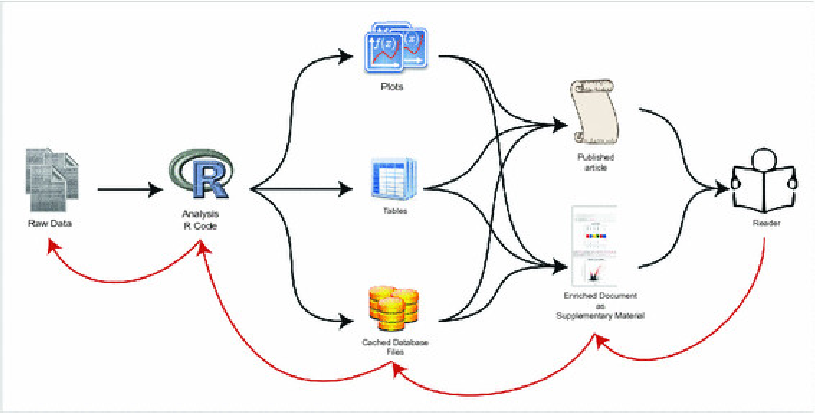


Figure 6: Source: Russo, Righeli, Angelini

Why you should care

- It will save you time and make analyses easy to update/run again
- Eliminates room for mistakes
- Easily shared and validated by others–especially crucial for analysis informing public policy

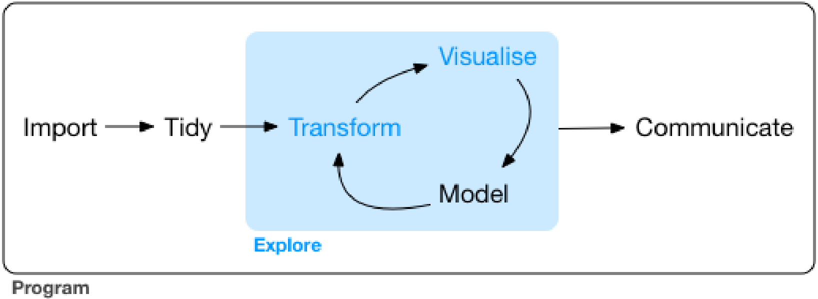
What to keep in mind for our purposes

- The ideal: someone else (including and especially you, at a later date) should be able to
  1. Read your code file and understand what you did and why (code comments are everyone's friend)
  2. Re-run your code without making any edits and produce the same results
- In practice:
  - Any data cleaning, transformations to the data, edits, analyses, etc. should be documented in an R script
  - No edits should be made manually!!! (don't edit your data in excel, don't manually enter or copy/paste output, etc.)

Tidyverse

- A collection of R packages made for data/statistical analysis with the same underlying structure, intuitive syntax and philosophy
- Great for working with and manipulating a wide variety of datasets
- We will use tidyverse today

Tidyverse


Import
Tidy
Transform
Visualise
Model
Communicate
Explore
Program

Learn!

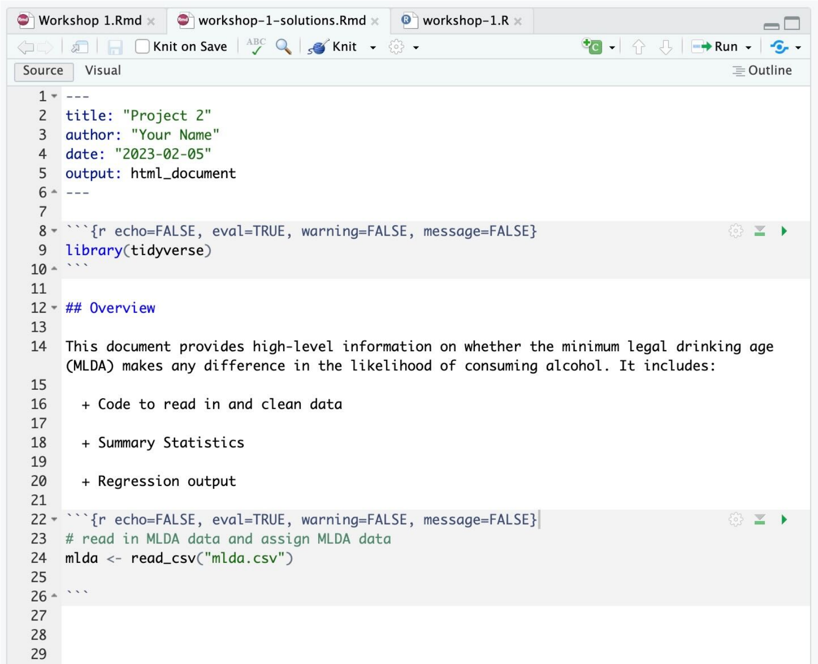
What we will do today

- Go through and reproduce a shortened version of Project 2 from last semester's 507c class
- use tidyverse to:
  - read in data
  - clean/transform data
  - generate summary statistics
  - run a regression model
  - output results in a summary memo document

Files

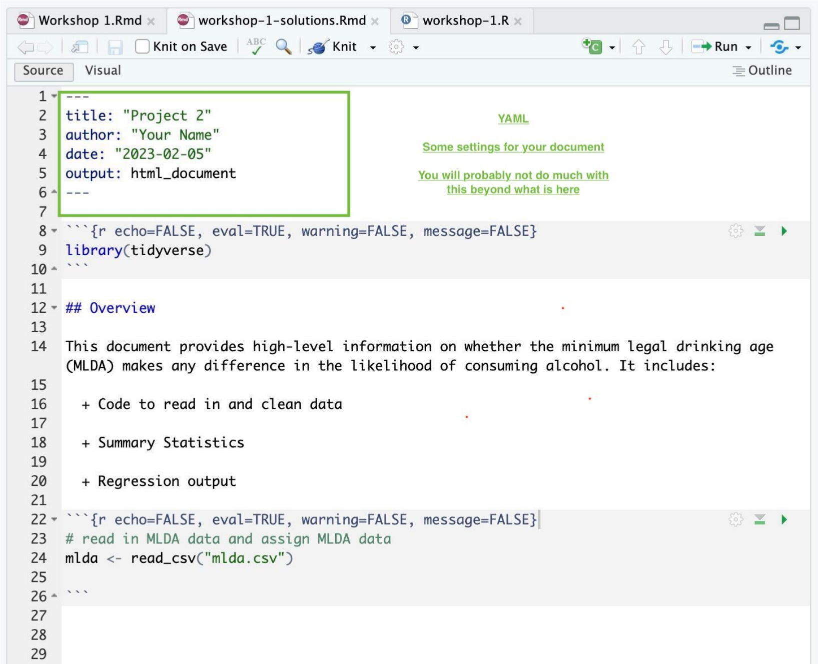
- workshop-1.R
  - this is just an R script with all the code for your reference later
- workshop-1-skeleton.Rmd
  - an RMarkdown code skeleton for you to fill out as we go
- workshop-1-solutions.Rmd
  - the same RMarkdown code file as above, but with all the code and solutions

RMarkdown File Structure: Overview



~~~
---
title: "Project 2"
author: "Your Name"
date: "2023-02-05"
output: html_document
---

```{r echo=FALSE, eval=TRUE, warning=FALSE, message=FALSE}
library(tidyverse)
```


## Overview

This document provides high-level information on whether the minimum legal drinking age (MLDA) makes any difference in the likelihood of consuming alcohol. It includes:

  + Code to read in and clean data

  + Summary Statistics

  + Regression output

```{r echo=FALSE, eval=TRUE, warning=FALSE, message=FALSE}
# read in MLDA data and assign MLDA data
mlda <- read_csv("mlda.csv")

```



~~~

RMarkdown File Structure: YAML



~~~
---
title: "Project 2"
author: "Your Name"
date: "2023-02-05"
output: html_document
---

```{r echo=FALSE, eval=TRUE, warning=FALSE, message=FALSE}
library(tidyverse)
~~~

Overview

This document provides high-level information on whether the minimum legal drinking age
(MLDA) makes any difference in the likelihood of consuming alcohol. It includes:

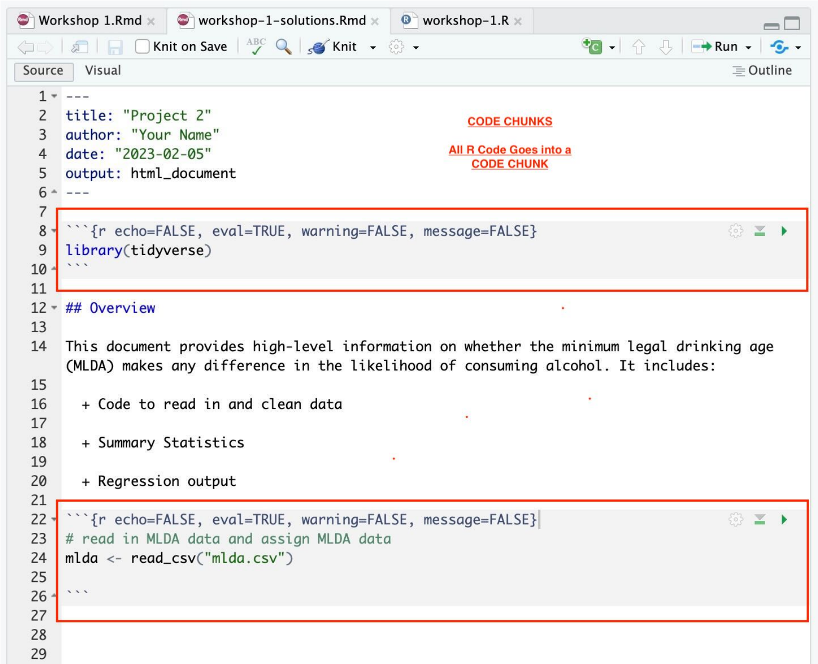
  + Code to read in and clean data

  + Summary Statistics

  + Regression output

```{r echo=FALSE, eval=TRUE, warning=FALSE, message=FALSE}
# read in MLDA data and assign MLDA data
mlda <- read_csv("mlda.csv")

```
```

YAML

Some settings for your document

You will probably not do much with this beyond what is here
```

# RMarkdown File Structure: Code



~~~
---
title: "Project 2"
author: "Your Name"
date: "2023-02-05"
output: html_document
---

```{r echo=FALSE, eval=TRUE, warning=FALSE, message=FALSE}
library(tidyverse)
~~~

## Overview

This document provides high-level information on whether the minimum legal drinking age (MLDA) makes any difference in the likelihood of consuming alcohol. It includes:

  + Code to read in and clean data

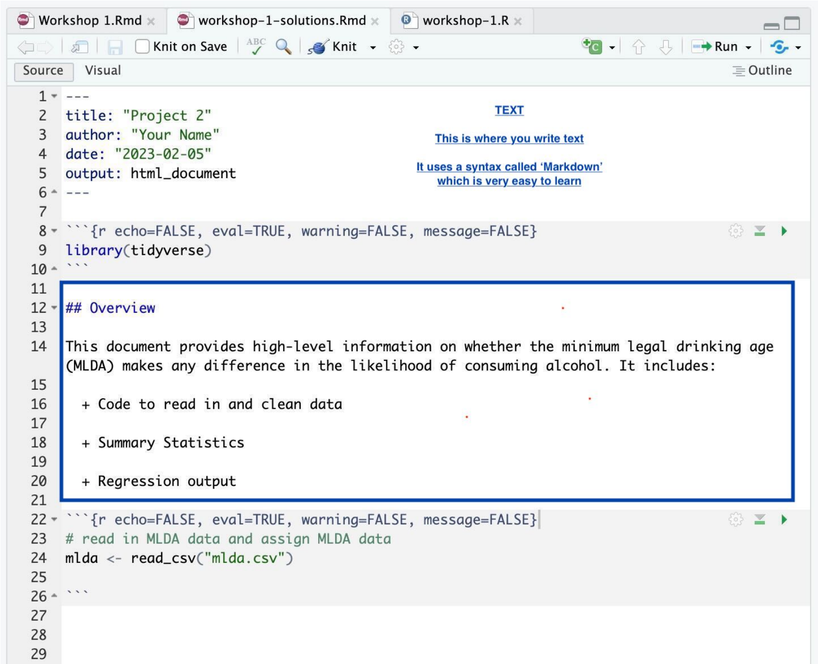
  + Summary Statistics

  + Regression output

```{r echo=FALSE, eval=TRUE, warning=FALSE, message=FALSE}
# read in MLDA data and assign MLDA data
mlda <- read_csv("mlda.csv")

```

```

CODE CHUNKS

All R Code Goes into a CODE CHUNK
```

RMarkdown File Structure: Text



~~~
---
title: "Project 2"
author: "Your Name"
date: "2023-02-05"
output: html_document
---

```{r echo=FALSE, eval=TRUE, warning=FALSE, message=FALSE}
library(tidyverse)
~~~

Overview

This document provides high-level information on whether the minimum legal drinking age (MLDA) makes any difference in the likelihood of consuming alcohol. It includes:

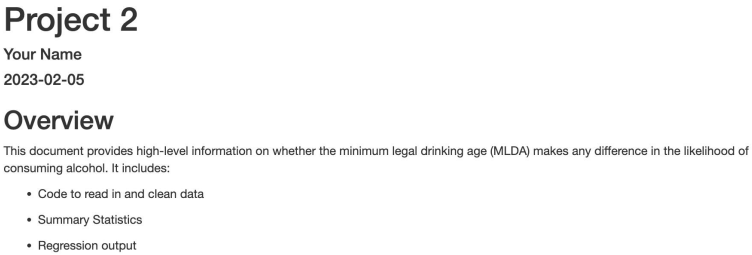
  + Code to read in and clean data

  + Summary Statistics

  + Regression output

```{r echo=FALSE, eval=TRUE, warning=FALSE, message=FALSE}
# read in MLDA data and assign MLDA data
mlda <- read_csv("mlda.csv")

```
```

TEXT

This is where you write text

It uses a syntax called 'Markdown' which is very easy to learn
```

# Rmarkdown: Output



# Project 2

### Your Name

### 2023-02-05

## Overview

This document provides high-level information on whether the minimum legal drinking age (MLDA) makes any difference in the likelihood of consuming alcohol. It includes:

- Code to read in and clean data
- Summary Statistics
- Regression output

# R Programming Basics

```
# object assignment
# (strings, numbers, dataframes, lists, etc.)
this_is_an_object <- "object"

# this is how you inspect an object
this_is_an_object
```

```
## [1] "object"
```

```
# another assignment
x <- 2^3

# another inspection
x
```

```
## [1] 8
```

- Naming is case sensitive
- Must start with a letter and have no spaces
- i_suggest_using_this_format
- ButSomePeopleDoThis
- objects will be loaded into your environment on the right!

Set Up: Working Directory, Package Loading

```r
#set working directory
setwd("~/Documents/workshop-1")

# you will only need to install packages once
install.packages("tidyverse")
install.packages("kableExtra")
install.packages("stargazer")
install.packages("knitr")
install.packages("rdrobust")

# you will do this whenever
# you need to load in and use tidyverse
library(tidyverse)
library(kableExtra)
library(stargazer)
library(knitr)
library(rdrobust)
```

Read in and inspect our data

```
mlda <- read_csv("mlda.csv")
mlda
```

```
## # A tibble: 61,263 x 11
##    hs_diploma hispanic white black emplo~1 married  male days_21 perc_~2 drink~3
##         <dbl>    <dbl> <dbl> <dbl>   <dbl>   <dbl> <dbl>   <dbl>   <dbl>   <dbl>
##  1          0        1     0     0       1       1     1    1601   0.548       0
##  2          1        1     0     0       1       0     0    1024   1.10        1
##  3          1        0     0     1       1       0     0    2455   6.58        1
##  4          0        1     0     0       1       0     1    -590   0           0
##  5          0        1     0     0       0       1     0    1815   0           0
##  6          1        0     0     1       1       1     0    2424   0           0
##  7          1        0     0     1       1       0     0    1116   3.29        1
##  8          1        0     0     1       1       0     0    2942   0.548       0
##  9          1        1     0     0       1       0     0    2516   3.29        1
## 10          1        0     0     1       1       0     0    2516  57.0         1
## # ... with 61,253 more rows, 1 more variable: student <dbl>, and abbreviated
## #   variable names 1: employed, 2: perc_days_drink, 3: drinks_alcohol
```

- Use the viewer to view more of the data (click on it or run code below)

```
View(mlda)
```

Some quick data exploration

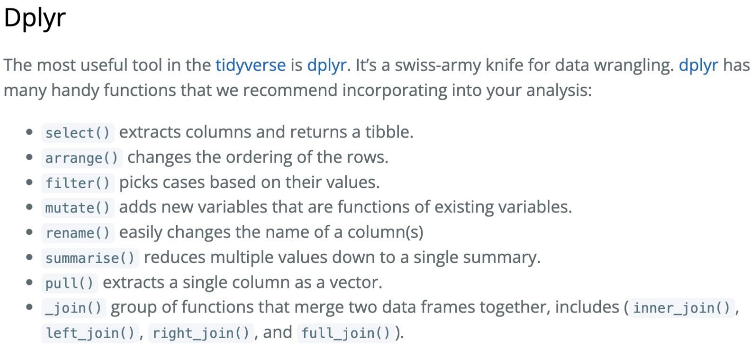
```
mean(mlda$student)
```

```
## [1] 0.1004848
```

```
mean(mlda$drinks_alcohol)
```

```
## [1] 0.6249449
```

Transform, explore, and visualize



Dplyr

The most useful tool in the tidyverse is dplyr. It's a swiss-army knife for data wrangling. dplyr has many handy functions that we recommend incorporating into your analysis:

- `select()` extracts columns and returns a tibble.
- `arrange()` changes the ordering of the rows.
- `filter()` picks cases based on their values.
- `mutate()` adds new variables that are functions of existing variables.
- `rename()` easily changes the name of a column(s)
- `summarise()` reduces multiple values down to a single summary.
- `pull()` extracts a single column as a vector.
- `_join()` group of functions that merge two data frames together, includes (`inner_join()`, `left_join()`, `right_join()`, and `full_join()`).

Figure 7: "Source: https://hbctraining.github.io/Intro-to-R/lessons/tidyverse_data_wrangling.html"

Tidyverse Function Syntax

Table 1: Useful dplyr functions and syntax

Function	Syntax
select()	select(df, var1, var2, ...)
mutate()	mutate(df, new_var = old_var + 5)
filter()	filter(df, var1 == value)
rename()	rename(df, new_name = old_name)
summarise()	summarise(df, mean_var1 = mean(var1))
if_else	if_else(condition, true, false)

Transform and explore

```
# add a new variable for age in years and a heavy drinker variable if they drink more than 50%
mlda <- mutate(mlda, age_years = 21 + days_21/365, heavy_drinker = if_else(perc_days_drink > 75, 1, 0))

# filter to just underage drinkers
underage <- filter(mlda, days_21 < 0 & drinks_alcohol == 1)

# demographic characteristics of underage drinkers
summ_stats_underage <- summarise(underage,
          min_age = min(age_years),
          mean_age = mean(age_years),
          median_age = median(age_years),
          mean_perc_days_drink = mean(perc_days_drink),
          median_perc_days_drink = median(perc_days_drink))

summ_stats_underage
```

```
## # A tibble: 1 x 5
##   min_age mean_age median_age mean_perc_days_drink median_perc_days_drink
##     <dbl>    <dbl>      <dbl>                <dbl>                  <dbl>
## 1    17.7     19.7       19.8                 13.3                   6.58
```

Use the pipe operator to combine all of these into one ‘dplyr chain’

- The pipe operate %>% takes output from on function and ‘pipes’ it into another function

```
x <- paste("A", "String")
print(x)

print(paste("A", "String"))

paste("A", "String") %>%
  print(.)
```

- The period denotes where the previous output should be the argument in the new function

```
summ_stats <- mutate(mlda, age_years = 21 + days_21/365, heavy_drinker = if_else(perc_days_drink > 75, 1,
  filter(., days_21 < 0 & drinks_alcohol == 1) %>%
  summarise(.,min_age = min(age_years),
          mean_age = mean(age_years),
          median_age = median(age_years),
          mean_perc_days_drink = mean(perc_days_drink),
          median_perc_days_drink = median(perc_days_drink))

summ_stats
```

```
## # A tibble: 1 x 5
##   min_age mean_age median_age mean_perc_days_drink median_perc_days_drink
##     <dbl>    <dbl>      <dbl>                <dbl>                  <dbl>
## 1    17.7     19.7       19.8                 13.3                   6.58
```

```
# now let's look at summary stats of heavy drinkers vs non heavy drinkers:

# demographic characteristics of heavy drinkers vs non-heavy drinkers
summ_stats_heavy <- mlda %>%
```

Try yourself

- Create a variable that denotes someone as an underage drinker (they are less than 21 years old and drink alcohol)
- Use the skeleton code below to calculate the share of underage drinkers that are a student, married, male, Hispanic, Black, or white

```
summ_stats2 <- mutate(mlda, underage_drinker = if_else()) %
  filter() %>%
  summarise()
```

Export as a table

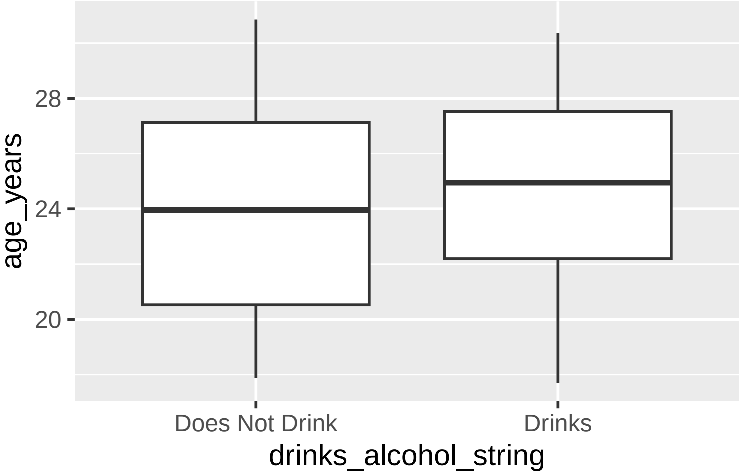
```r
library(kableExtra)

kableExtra::kable(summ_stats_underage,
      digits=1, caption="Drinking Patterns among Underage Drinkers",
      col.names = c("Minimum Age", "Mean Age", "Median Age", "Avg Percent Days Drank",
                    "Median Percent Days Drank"))
```

Table 2: Drinking Patterns among Underage Drinkers

Minimum Age	Mean Age	Median Age	Avg Percent Days Drank	Median Percent Days Drank
17.7	19.7	19.8	13.3	6.6

Age distribution of drinkers vs non-drinkers?

```
viz <- mlda %>%
  mutate(drinks_alcohol_string = if_else(drinks_alcohol == 1, "Drinks", "Does Not Drink")) %>%
  ggplot(aes(x=drinks_alcohol_string, y=age_years)) +
  geom_boxplot()
viz
```



Run a Model

- Use a linear probability model to predict the effect of age in years on the likelihood of drinking alcohol
- we can use lm()

```
# general form
lpm1 <- lm(formula = y ~ x_1 + x_2 + ... x_n, data = df)
```

```
# lpm of years in age on likelihood of drinking alcohol
lpm1 <- lm(formula = drinks_alcohol ~ age_years, data = mlda )
summary(lpm1)
```

```
##
## Call:
## lm(formula = drinks_alcohol ~ age_years, data = mlda)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7365 -0.5766  0.3128  0.3852  0.4922
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.1999271  0.0140917   14.19   <2e-16 ***
## age_years   0.0173927  0.0005712   30.45   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4805 on 61261 degrees of freedom
## Multiple R-squared:  0.01491,    Adjusted R-squared:  0.01489
## F-statistic: 927.3 on 1 and 61261 DF,  p-value: < 2.2e-16
```

Exercise

- Now, add an age squared and age cubed variable to your dataset and rerun

```
mlda <- mlda %>%
  mutate()

lpm2 <- lm(formula = , data = )
```

- Then add additional demographic controls to your model and rerun

```
lpm3 <- lm(formula = drinks_alcohol ~ , data =)
```

Use Stargazer to output your results

```
library(stargazer)

stargazer(lpm1, lpm2, type = "html",
          dep.var.labels = "Probabilty of Drinking", header=F)
```

Table 3:

	Dependent variable:	
	Probabilty of Drinking	
	(1)	(2)
age_years	0.017*** (0.001)	1.697*** (0.104)
age_sq		$-$0.065*** (0.004)
age_cu		0.001*** (0.0001)
Constant	0.200*** (0.014)	$-$13.996*** (0.817)
Observations	61,263	61,263
R^2	0.015	0.031
Adjusted R^2	0.015	0.031
Residual Std. Error	0.481 (df = 61261)	0.476 (df = 61259)
F Statistic	927.281*** (df = 1; 61261)	661.294*** (df = 3; 61259)
Note:		*p$<$0.1; **p$<$0.05; ***p$<$0.01

Output model results

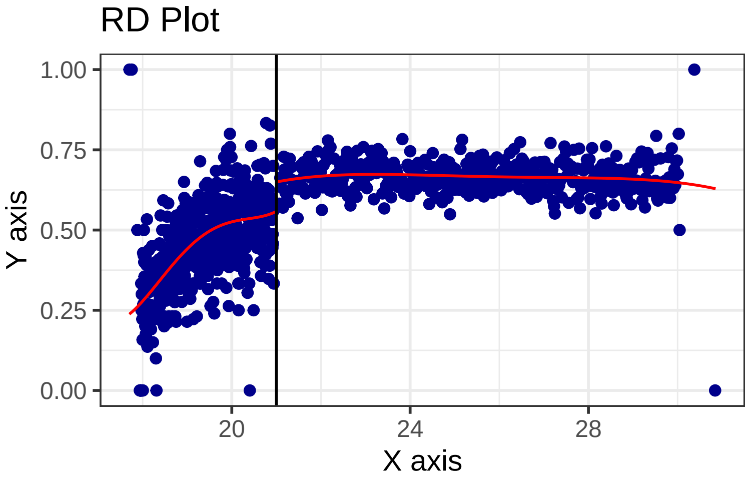
Table 4:

	Dependent variable:		
	Probabilty of Drinking with Controls		
	(1)	(2)	(3)
age_years	0.017***	1.697***	1.710***
	(0.001)	(0.104)	(0.102)
age_sq		−0.065***	−0.066***
		(0.004)	(0.004)
age_cu		0.001***	0.001***
		(0.0001)	(0.0001)
student			−0.025***
			(0.007)
male			0.179***
			(0.004)
Constant	0.200***	−13.996***	−14.167***
	(0.014)	(0.817)	(0.803)
Observations	61,263	61,263	61,263
R^2	0.015	0.031	0.065
Adjusted R^2	0.015	0.031	0.065
Residual Std. Error	0.481 (df = 61261)	0.476 (df = 61259)	0.468 (df = 61257)
F Statistic	927.281*** (df = 1; 61261)	661.294*** (df = 3; 61259)	855.417*** (df = 5; 61257)
Note:			*p<0.1; **p<0.05; ***p<0.01

Visualize the discontinuity

```
library(rdrobust)
rdplot(mlda$drinks_alcohol,mlda$age_years,c=21)
```

```
## [1] "Mass points detected in the running variable."
```



Citations

Citations

Russo, Francesco & Righelli, Dario & Angelini, Claudia. (2016). Advantages and Limits in the Adoption of Reproducible Research and R-Tools for the Analysis of Omic Data. Lecture Notes in Computer Science. 9874. 245-258. 10.1007/978-3-319-44332-4_19.