Section 1

Sam Frederick
sdf2128@columbia.edu

Downloading Files from Courseworks

Home

Announcements

Assignments

Discussions

Grades

People

Files

Syllabus

Modules

BigBlueButton

Collaborations

Chat

Research Guide

Course Info

Library Reserves

Textbooks

Mailtool

Course Video Recordings (Panopto)

Zoom Class Sessions

Name & Pronoun Recordings

files	Sep 12, 2021			--
gerber_green_larimer_2008.pdf	Sep 19, 2021	Sep 19, 2021		270 KB
holbrook_heidreder_2010.pdf	Sep 4, 2022	Sep 4, 2022	Shigeo Hirano	1.9 MB
minwage.csv	Nov 29, 2021	Nov 29, 2021		14 KB
R commands used in class September 12.pdf	Sunday	Sunday	Shigeo Hirano	1 MB
resume.csv	Sep 16, 2021	Sep 16, 2021		133 KB
section	Thursday			--
sept12_2022.R	Sunday	Sunday	Shigeo Hirano	1 KB
smith_pell_2003.pdf	Sep 26, 2021	Sep 26, 2021		116 KB
social.csv	Oct 26, 2021	Oct 26, 2021		9.1 MB
turnout_class3.csv	Sep 16, 2021	Sep 16, 2021		10.6 MB
turnout.csv	Sep 4, 2022	Sep 4, 2022	Shigeo Hirano	986 bytes
UNprop.csv	Sep 14, 2021	Sep 14, 2021		113 bytes
yeh_et_al_2018.pdf	Sep 26, 2021	Sep 26, 2021		1.7 MB

All My Files

Downloading Files from Courseworks

Downloading Files from Courseworks

files	Sep 12, 2021			--	
gerber_green_larimer_2008.pdf	Sep 19, 2021	Sep 19, 2021		270 KB	
holbrook_heidbreder_2010.pdf	Sep 4, 2022	Sep 4, 2022	Shigeo Hirano	1.9 MB	
minwage.csv	Nov 29, 2021	Nov 29, 2021		14 KB	
R commands used in class September 12.pdf	Sunday	Sunday	Shigeo Hirano	1 MB	
resume.csv	Sep 16, 2021	Sep 16, 2021		133 KB	
section	Thursday			--	
sept12_2022.R	Sunday	Sunday	Shigeo Hirano	1 KB	
smith_pell_2003.pdf	Sep 26, 2021	Sep 26, 2021		116 KB	
social.csv	Oct 26, 2021	Oct 26, 2021		9.1 MB	
turnout_class3.csv	Sep 16, 2021	Sep 16, 2021		10.6 MB	
turnout.csv	Sep 4, 2022	Sep 4, 2022	Shigeo Hirano	986 bytes	⋮
UNpop.csv	Sep 14, 2021	Sep 14, 2021		113 by	Download
yeh_et_al_2018.pdf	Sep 26, 2021	Sep 26, 2021		1.7 MB	

All My Files

Setting Up a Folder for Class

1. Open Finder on Mac or Windows Explorer on Windows

Setting Up a Folder for Class

1. Open Finder on Mac or Windows Explorer on Windows
2. Navigate to Desktop (i.e., click on "Desktop")

Setting Up a Folder for Class

1. Open Finder on Mac or Windows Explorer on Windows
2. Navigate to Desktop (i.e., click on "Desktop")
3. Click "File" > "New Folder"

Setting Up a Folder for Class

1. Open Finder on Mac or Windows Explorer on Windows
2. Navigate to Desktop (i.e., click on "Desktop")
3. Click "File" > "New Folder"
4. Give folder a useful title
   - Example: Research Design

Setting Up a Folder for Class

1. Open Finder on Mac or Windows Explorer on Windows
2. Navigate to Desktop (i.e., click on "Desktop")
3. Click "File" > "New Folder"
4. Give folder a useful title
   - Example: Research Design
5. Use this folder to store course materials

RStudio

Let's open RStudio

Arithmetic

```
2+2
2-2
2*2
2/2
2^3
```

Functions in R

- Functions: perform common tasks for us

Functions in R

- ► Functions: perform common tasks for us

- ► General form:
  - ► `functionname(arguments)`

Functions in R

- Functions: perform common tasks for us

- General form:
  - `functionname(arguments)`

- Getting help for functions:
  - `?functionname`
  - `args(functionname)`

Common Functions

- ▶ sqrt() takes the square root of whatever you put inside of the parentheses

```
sqrt(4)
```

Common Functions

- ▶ sqrt() takes the square root of whatever you put inside of the parentheses

```
sqrt(4)
```

- ▶ c(), or concatenate, combines everything within the parentheses into a vector

```
c(1,2,3)
```

Common Functions

- mean() calculates the average of object within the parentheses

```
mean(c(1,2,3))
```

Common Functions

▶ `mean()` calculates the average of object within the parentheses

```
mean(c(1,2,3))
```

▶ `median()` calculates the median of the object within the parentheses

```
median(c(1,2,3))
```

Common Functions

- ▶ mean() calculates the average of object within the parentheses

```
mean(c(1,2,3))
```

- ▶ median() calculates the median of the object within the parentheses

```
median(c(1,2,3))
```

- ▶ sum() adds the values in the vector within the parentheses

```
sum(c(1,2,3))
```

Common Functions

- min() calculates the minimum of a vector, and max() calculates the maximum

```
min(c(1,2,3))
max(c(1,2,3))
```

Common Functions

▶ `summary()` gives a variety of these statistics for a vector

```
summary(c(1,2,3))
```

Common Functions

- ▶ summary() gives a variety of these statistics for a vector

```
summary(c(1,2,3))
```

- ▶ length() gives the length of the object within the parentheses

```
length(c(1,2,3))
```

Common Functions

▶ unique() will give all of the unique values in a vector within the parentheses

```
unique(c(1,2,3,1,2,3))
```

Common Functions

- x:z will give you the sequence of integers from number x to number z

```
1:3
```

Common Functions

- `x:z` will give you the sequence of integers from number x to number z

```
1:3
```

- `seq(a, b, by = z)` will give you the sequence of numbers from number a to number b in increments of z

```
seq(1,3, by = 1)
seq(0, 1, by = 0.1)
```

Variables in R

- `<-` is the assignment operator

Variables in R

- <- is the assignment operator

- x<- y *assigns* value y to object x

Variables in R

▶ <- is the assignment operator

▶ x<- y *assigns* value y to object x

```
x <- 1:3
x
```

Variables in R

- `<-` is the assignment operator

- x`<-` y *assigns* value y to object x

```
x <- 1:3
x
```

- x can be whatever we want it to be but can't start with numbers or contain spaces

Variables in R

▶ We can use functions and perform arithmetic operations on our object assigned above

```
mean(x)
summary(x)
```

Types and Classes of Objects in R: Numeric

- ▶ Numeric Data:
  - ▶ Integers

```
class(x)
typeof(x)
```

Types and Classes of Objects in R: Numeric

- Numeric Data:
  - Integers

```
class(x)
typeof(x)
```

- Doubles

```
class(1.3)
typeof(1.3)
```

Types and Classes of Objects in R: Character

▶ Text Data:
  ▶ Characters

```
"a"
class("a")
```

Types and Classes of Objects in R: Factor

- Categorical Data:
  - Store variables with categories as factor variables

Types and Classes of Objects in R: Factor

- ▶ Categorical Data:
  - ▶ Store variables with categories as factor variables
  - ▶ Examples: Party identification, ideology, race, gender

```r
party <- c("Republican", "Democrat",
           "Independent", "Republican")
class(party)
party_factor <- factor(party,
                       levels = c("Democrat",
                                  "Independent",
                                  "Republican"))
class(party_factor)
```

Types and Classes of Objects in R: Data Frames

- ▶ We often want to store multiple vectors containing different data in a cohesive format
  - ▶ data.frame

Types and Classes of Objects in R: Data Frames

- ▶ We often want to store multiple vectors containing different data in a cohesive format
  - ▶ data.frame

- ▶ Columns are often called variables
  - ▶ Accessed using $ operator
  - ▶ Example: surveydata$party

Types and Classes of Objects in R: Data Frames

- ▶ We often want to store multiple vectors containing different data in a cohesive format
  - ▶ data.frame

- ▶ Columns are often called variables
  - ▶ Accessed using $ operator
  - ▶ Example: surveydata$party

- ▶ Rows generally represent the observed values for each observation/individual
  - ▶ Accessed using []
  - ▶ Example: surveydata[1,]

Types and Classes of Objects in R: Data Frames

▶ We often want to store multiple vectors containing different data in a cohesive format
  ▶ data.frame

▶ Columns are often called variables
  ▶ Accessed using $ operator
  ▶ Example: `surveydata$party`

▶ Rows generally represent the observed values for each observation/individual
  ▶ Accessed using []
  ▶ Example: `surveydata[1,]`

▶ Can access both columns and rows using []
  ▶ Example: `surveydata[row, column]`
  ▶ Example: `surveydata$column[row]`

Types and Classes of Objects in R: Logical

- ▶ Logical/Boolean Data:
  - ▶ TRUE or FALSE

```
class(TRUE)
class(FALSE)
class(c(TRUE, TRUE, FALSE))
```

Logical Operators

- `==` tests whether one object is equivalent to another
- `!=` tests whether one object is **not** equivalent to another
- `<` (or `>`) tests whether one object is less than (or greater than) another
- `<=` (or `>=`) tests whether one object is less than or equal to (or greater than or equal to) another

```
2==2
2==3
2!=3
2<3
```

Missing Data

- Missing data often appear as `NA` in R
- `NA` is a special type of logical object
- Sometimes have to remove NA values to calculate desired quantities

```
class(NA)
mean(c(1,2,3, NA))
mean(c(1,2,3, NA),na.rm = T)
```

Missing Data

- Missing data often appear as `NA` in R
- `NA` is a special type of logical object
- Sometimes have to remove NA values to calculate desired quantities

```
class(NA)
mean(c(1,2,3, NA))
mean(c(1,2,3, NA),na.rm = T)
```

- Remember you can use ? to learn more about function arguments

Example