

Some Essentials for Data Science with R

Derek Beaton

2020 FEB 25

Outline

- ▶ Part 0: Project set up

Outline

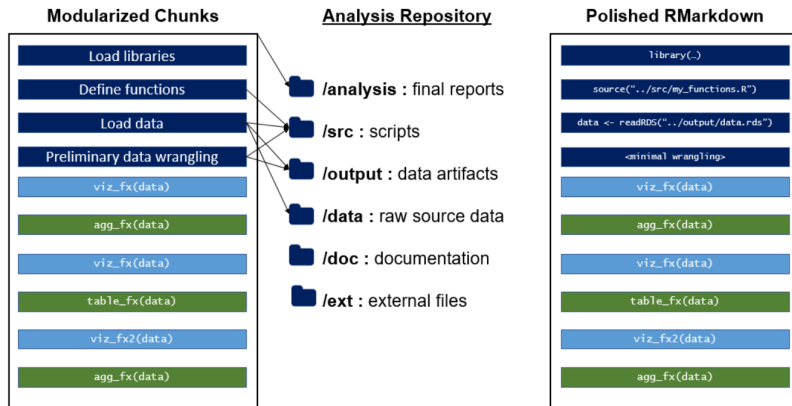
- ▶ Part 0: Project set up
- ▶ Part 1: RStudio, Git, R, and RMarkdown

Outline

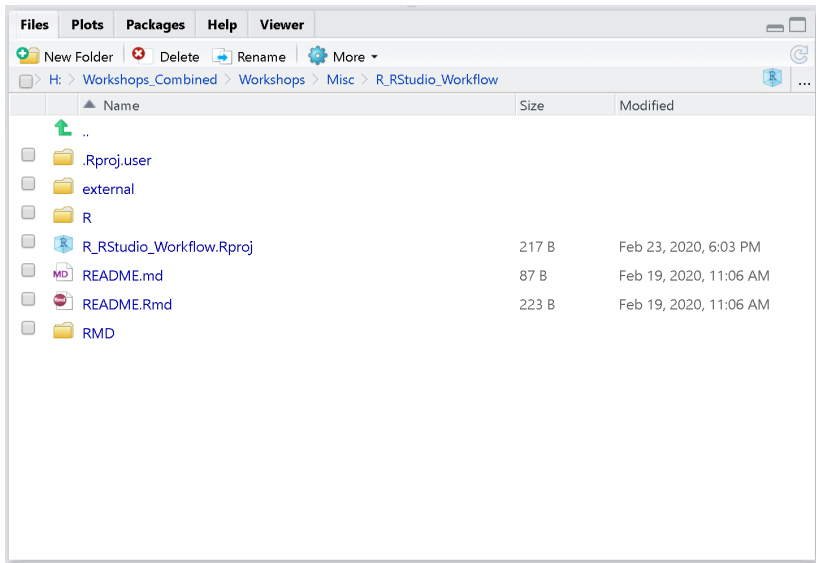
- ▶ Part 0: Project set up
- ▶ Part 1: RStudio, Git, R, and RMarkdown
- ▶ Part 2: Working with data

Part 0: Project set up

Part 0: Project set up



<https://emilyriederer.netlify.com/post/rmarkdown-driven-development/>



Organize your project folders and markdown

- ▶ What works for you?

Organize your project folders and markdown

- ▶ What works for you?
- ▶ What works for your organization or team?

Organize your project folders and markdown

- ▶ What works for you?
- ▶ What works for your organization or team?
- ▶ Maximize utility, minimize complexity

Part 1: RStudio, Git, R, and RMarkdown

RStudio

- ▶ IDE: Integrated development environment

RStudio

- ▶ IDE: Integrated development environment
- ▶ RStudio: Does so much

RStudio

- ▶ IDE: Integrated development environment
- ▶ RStudio: Does so much
 - ▶ We scratch the surface here

RStudio Setup

- ▶ Download R and Rstudio

RStudio Setup

- ▶ Download R and Rstudio
 - ▶ Strongly recommend Microsoft R
(<https://mran.microsoft.com/open>)

RStudio Setup

- ▶ Download R and Rstudio
 - ▶ Strongly recommend Microsoft R (<https://mran.microsoft.com/open>)
 - ▶ Comes with Intel MKL

RStudio Setup

- ▶ Download R and Rstudio
 - ▶ Strongly recommend Microsoft R (<https://mran.microsoft.com/open>)
 - ▶ Comes with Intel MKL
- ▶ Plain R is fine (<https://cran.r-project.org/>)

RStudio Setup

- ▶ Download R and Rstudio
 - ▶ Strongly recommend Microsoft R (<https://mran.microsoft.com/open>)
 - ▶ Comes with Intel MKL
- ▶ Plain R is fine (<https://cran.r-project.org/>)
 - ▶ Can relink to faster libraries

RStudio Setup

- ▶ Download R and Rstudio
 - ▶ Strongly recommend Microsoft R (<https://mran.microsoft.com/open>)
 - ▶ Comes with Intel MKL
- ▶ Plain R is fine (<https://cran.r-project.org/>)
 - ▶ Can relink to faster libraries
- ▶ Download RStudio (<https://www.rstudio.com/>)

RStudio Environment

The screenshot displays the RStudio interface with the following components:

- Script Editor:** Contains R code for data processing, including loading the `ADNI` package, cleaning data, and creating a subset of variables.
- Console:** Shows the output of the executed code, including summary statistics for various variables.
- Environment Pane:** Displays the current environment, showing the `amerge_subset` data frame with 665 observations and 17 variables.

Script Editor Code:

```
## 0.1 Specify the column names and participants you want (ie, baseline visit for all participants with MOCA=1)
adni.cols <- c("RID", "VISCODE", "DX", "AGE", "PTGENDER", "PTEDUCAT", "PTETHCAT", "PTRACCAT", "APOE4", "FDG", "
adni.rows <- c(adnimerge$VISCODE=="b1" & adnimerge$MOCA==16)
amerge_subset <- adnimerge[adni.rows, adni.cols]

#### remove participants with missing data
amerge_subset <- amerge_subset[complete.cases(amerge_subset),]

## 0.2 Bring in modified hachinks
amerge_subset$HMScore <- modhach$HMScore[match(amerge_subset$RID, modhach$RID)]

## 0.3 Manually change variable classes (remove class 'labelled')
amerge_subset$HMScore <- as.numeric(amerge_subset$HMScore)
```

Console Output:

```
## 0.1 Specify the column names and participants you want (ie, baseline visit for all participants with MOCA=1)
Mean :71.92      Mean :10.36
3rd Qu.:176.60   3rd Qu.:18.00
Max. :89.60      Max. :20.00

APOE4      FDG      APOE4      FDG      APOE4      FDG
Min. :0.0000   Min. :0.6983   Min. :0.8385   Min. :0.0000   Min. :0.0000   Min. :16.00
1st Qu.:0.0000   1st Qu.:1.1837   1st Qu.:1.0199   1st Qu.:0.0000   1st Qu.:8.00    1st Qu.:22.00
Median :0.0000   Median :1.2802   Median :1.1105   Median :1.0000   Median :12.00   Median :24.00
Mean :0.5248   Mean :1.2682   Mean :1.1989   Mean :1.2020   Mean :13.80    Mean :23.89
3rd Qu.:1.0000   3rd Qu.:1.3620   3rd Qu.:1.3714   3rd Qu.:2.0000   3rd Qu.:18.00   3rd Qu.:26.00
Max. :2.0000   Max. :1.7011   Max. :2.0256   Max. :5.5000   Max. :46.00    Max. :30.00

HMScore     Hippocampus     MidTemp     mPACTra1158     HMScore
Min. :81721   Min. :12213   Min. :138.6883   Min. :0.0000
1st Qu.:984410   1st Qu.:6510   1st Qu.:18535   1st Qu.:-6.4051   1st Qu.:0.0000
Median :1051621   Median :7223   Median :20186   Median :-2.5250   Median :1.0000
Mean :1105026   Mean :7150   Mean :20302   Mean :-3.6882   Mean :0.588
3rd Qu.:1120570   3rd Qu.:7834   3rd Qu.:22088   3rd Qu.:0.3482   3rd Qu.:1.0000
Max. :11486036   Max. :110602   Max. :32189   Max. :5.3540   Max. :3.0000

> view(amerge_subset)
```

Environment Pane:

amerge_subset: 665 obs. of 17 variables
variable_type_map: num [1:17, 1:3] 0 1 0 0 0 0 1 1 0 ...
Values:
ids: chr [1:665] "2002" "2003" "2007" "2010" "2011" "201...
MOCA: num [1:665] 28 24 23 27 25 26 25 24 24 30 ...
Functions:
scatterplot: function (x, y, x.lim = NA, y.lim = NA, x.lab = "...)

RStudio Environment

~/workshops/2019_Rstudio_Magic-master - RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function

Source on Save

Run

Source

Environment History Connections Git

Global Environment

Data

amerge_subset 665 obs. of 17 variables

variable_type_map

num [1:17, 1:3] 0 1 0 0 0 0 1 1 0 ...

Values

ids chr [1:665] "2002" "2003" "2007" "2010" "2011" "201...

MOCA num [1:665] 28 24 23 27 25 26 25 24 24 30 ...

Functions

scatterplot function (x, y, x.lim = NA, y.lim = NA, x.lab = "...)

CONSOLE

```
## 0.1 Specify the column names and participants you want (ie, baseline visit for all participants with MOCA=1
8 adni.cols <- c("RID", "VISCODE", "DX", "AGE", "PTGENDER", "PTEDUCAT", "PTETHCAT", "PTRACCAT", "APOE4", "FDG",
9 adni.rows <- c(adnimerge$VISCODE=="b1" & adnimerge$MOCA==16)
10 amerge_subset <- adnimerge[adni.rows, adni.cols]
11
12 ##### remove participants with missing data
13 amerge_subset <- amerge_subset[complete.cases(amerge_subset),]
14
15 ## 0.2 Bring in modified hachinks1
16 amerge_subset$HMSCORE <- modhach$HMSCORE[match(amerge_subset$RID, modhach$RID)]
17
18 ## 0.3 Manually change variable classes (remove class 'labelled')
19 <-
20 <-
```

Console Terminal Jobs

```
~/workshops/2019_Rstudio_Magic/ >>
Mean :71.92 Mean :10.36
3rd Qu.:176.60 3rd Qu.:18.00
Max. :89.60 Max. :20.00
APOE4 FDG APOE4 CDRSB ADAS13 MOCA
Min.:0.0000 Min.:0.6983 Min.:0.8385 Min.:0.0000 Min.:8.0 Min.:16.00
1st Qu.:0.0000 1st Qu.:1.1837 1st Qu.:1.0199 1st Qu.:0.0000 1st Qu.:8.0 1st Qu.:22.00
Median:0.0000 Median:1.2802 Median:1.1105 Median:1.0000 Median:12.0 Median:24.00
Mean :0.5248 Mean :1.2682 Mean :1.1989 Mean :1.202 Mean :13.8 Mean :23.89
3rd Qu.:1.0000 3rd Qu.:1.3620 3rd Qu.:1.1714 3rd Qu.:2.0000 3rd Qu.:18.0 3rd Qu.:26.00
Max. :2.0000 Max. :1.7011 Max. :2.0256 Max. :5.500 Max. :46.0 Max. :30.00
Mholatrain Hippocampus MidTemp mPACCtra1158 HMSCORE
Min. : 817421 Min. :12213 Min. : -38.6983 Min. :0.0000
1st Qu.: 984410 1st Qu.: 6510 1st Qu.:18535 1st Qu.: -6.4051 1st Qu.:0.0000
Median :1051621 Median : 7223 Median :20186 Median : -2.5250 Median :1.0000
Mean :1105026 Mean : 7150 Mean :20302 Mean : -3.6882 Mean :0.588
3rd Qu.:1120570 3rd Qu.: 7834 3rd Qu.:22088 3rd Qu.: -0.3482 3rd Qu.:1.0000
Max. :11486036 Max. :110602 Max. :32189 Max. : 5.3540 Max. :3.000
> view(amerge_subset)
>
```

Files Plots Packages Help Viewer

New Folder Delete Rename More

Home workshops 2019_Rstudio_Magic

Name Size Modified

Environment 52 B May 12, 2019, 11:33 AM

2019_Rstudio_Magic.Rproj 218 B May 12, 2019, 6:30 PM

external

mic

output

R

README.md 42 B May 12, 2019, 11:29 AM

Rmd

RStudio Environment

The screenshot displays the RStudio interface with the following components:

- Script Editor:** Contains R code for loading the `ADNI` dataset, cleaning it, and creating a subset of variables. The code includes comments and function calls like `library(ADNIMERGE)`, `admi.rows <- c(adnimerge$VISCODE=="b1")`, and `amerge_subset <- adnimerge[admi.rows, admi.cols]`.
- Console:** Shows the output of the code execution, including summary statistics for various variables such as `PTACCAT`, `APOE4`, `FDG`, `AV45`, `CDR5B`, `ADAS13`, `MOCA`, `lmoledbrain`, `Hippocampus`, `MidTemp`, `mPACCtra115B`, and `HMSCORE`.
- Environment Pane:** Displays the objects in the global environment, including `amerge_subset` (665 obs. of 17 variables) and `scatterplot` (a function).
- Files, Plots, Packages, Help, Viewer Pane:** A red box highlights this pane, which shows the file explorer, plots, packages, help, and viewer. The file explorer shows the current project structure, including `2019_Rstudio_Magic.Rproj` and `2019_Rstudio_Magic.R`.

FILES, PLOTS, HELP

RStudio Environment

The screenshot displays the RStudio interface with the following components:

- Script Editor:** Contains R code for loading the `ADNI` dataset, cleaning it, and creating a subset of variables. The code includes comments and function calls like `library(ADNIMERGE)`, `admi.rows <- c(adnimerge$VISCODE=="b1")`, and `amerge_subset <- adnimerge[admi.rows, admi.cols]`.
- Console:** Shows the output of the code execution, including summary statistics for various variables such as `AP0E4`, `FDG`, `AV45`, `CDR5B`, `ADAS13`, `MOCA`, `ihofetbrain`, `Hippocampus`, `MidTemp`, `hPACCtra115b`, and `hMSCORE`.
- Environment Pane (Highlighted):** Displays the current environment, showing the `amerge_subset` data frame with 665 observations and 17 variables. It also lists the functions loaded in the environment, including `scatterplot`.

VARIABLES, HISTORY, VERSION CONTROL

RStudio Environment

The screenshot displays the RStudio interface with the following components:

- Code Pane:** Contains R code for loading data and cleaning it. A red box highlights the code from line 1 to 19. The word "CODE" is written in large red letters over the code.
- Console:** Shows the output of the R code, including summary statistics for various variables.
- Environment:** Displays the current data environment, showing the 'amerge_subset' object with 665 observations and 17 variables.

CODE

```
1 library(ADNIMERGE)
2
3 #####
4 ## Load and clean data
5 #####
6
7 ## 0.1 Specify the column names and participants you want (ie, baseline visit for all participants with MOCA=1
8 adni.cols <- c("RID", "VISCODE", "DX", "AGE", "PTGENDER", "PTEDUCAT", "PTETHCAT", "PTRACCAT", "APOE4", "FDG",
9 adni.rows <- c(adnimerge$VISCODE=="b1" & adnimerge$MOCA==16)
10 amerge_subset <- adnimerge[adni.rows, adni.cols]
11
12 ##### remove participants with missing data
13 amerge_subset <- amerge_subset[complete.cases(amerge_subset),]
14
15 ## 0.2 Bring in modified hachinks1
16 amerge_subset$HMSCORE <- modhach$HMSCORE[match(amerge_subset$RID, modhach$RID)]
17
18 ## 0.3 Manually change variable classes (remove class 'labelled')
19
20
```

Console

```
~/workshops/2019_Rstudio_Magic/ >
      Mean      :71.92      Mean      :10.36
      3rd Qu.:176.60      3rd Qu.:18.00
      Max.     :89.60      Max.     :20.00

APOE4      FDG      APOE4      ADAS13      MOCA
Min.:0.0000 Min.:0.6983 Min.:0.8385 Min.:0.0000 Min.:0.0000
1st Qu.:0.0000 1st Qu.:1.1837 1st Qu.:1.0199 1st Qu.:0.0000 1st Qu.:16.00
Median:0.0000 Median:1.2802 Median:1.1105 Median:1.0000 Median:24.00
Mean :0.5248 Mean :1.2682 Mean :1.1989 Mean :1.202 Mean :13.8 Mean :23.89
3rd Qu.:1.0000 3rd Qu.:1.3620 3rd Qu.:1.1714 3rd Qu.:2.000 3rd Qu.:18.0 3rd Qu.:26.00
Max. :2.0000 Max. :1.7011 Max. :2.0256 Max. :5.500 Max. :46.0 Max. :30.00

Mholatrain Hippocampus MidTemp mPACCtra118b HMSCORE
Min. : 81721 Min. : 12213 Min. : -38.6883 Min. : 0.0000
1st Qu.: 984410 1st Qu.: 6510 1st Qu.:18535 1st Qu.: -6.4051 1st Qu.:0.0000
Median :1051621 Median : 7223 Median :20186 Median : -2.5250 Median :1.0000
Mean :11057026 Mean : 7150 Mean :20302 Mean : -3.6882 Mean :0.588
3rd Qu.:11205710 3rd Qu.: 7834 3rd Qu.:22088 3rd Qu.: -0.3482 3rd Qu.:1.0000
Max. :11486036 Max. :110602 Max. :32189 Max. : 5.3540 Max. :3.000

> view(amerge_subset)
>
```

Environment

Object	Class	Attributes
amerge_subset	data.frame	665 obs. of 17 variables
variable_type_map	list	[1:17, 1:3] 0 1 0 0 0 0 1 1 0 ...
ids	chr	[1:665] "2002" "2003" "2007" "2010" "2011" "201..."
MOCA	num	[1:665] 28 24 23 27 25 26 25 24 24 30 ...
scatterplot	function	function (x, y, x.lim = NA, y.lim = NA, x.lab = "...")

RStudio Environment

~/workshops/2019_Rstudio_Magic-master - RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function Addins

DATA VIEWER

	DX	AGE	PTGENDER	PTEDUCAT	PTETHCAT	PTRACCAT	APOE4	FDG	AV45	CDRSB	ADAS13	MOCA	WholeBrain
2002	MCI	64.8	Male	18	Not Hisp/Latino	White	0	1.2091908	0.9794523	2.5	4	28	1123556.8
2003	MCI	65.6	Female	18	Not Hisp/Latino	White	0	1.2899625	1.1646374	2.0	11	24	1070369.5
2007	MCI	85.4	Female	20	Hisp/Latino	White	0	1.3058182	1.4495250	2.5	9	23	920710.1
2010	MCI	62.9	Female	20	Not Hisp/Latino	Other	1	1.3121151	1.1472848	0.5	6	27	986402.9
2011	MCI	69.9	Female	14	Not Hisp/Latino	White	0	1.4537199	1.0537930	1.5	7	25	967822.5
2018	MCI	76.4	Female	18	Not Hisp/Latino	White	0	1.3148491	1.0525191	1.5	10	26	1004817.0
2022	MCI	66.0	Male	18	Not Hisp/Latino	Other	1	1.2031270	1.3135914	1.5	6	25	1173068.2
2023	MCI	61.9	Female	14	Not Hisp/Latino	White	0	1.4000446	1.0299761	1.0	6	24	969957.1
2031	MCI	72.5	Male	16	Not Hisp/Latino	White	0	1.3404430	0.9939887	2.0	10	24	1059879.5
2036	MCI	66.7	Female	14	Not Hisp/Latino	White	0	1.2892910	1.0300795	1.0	5	30	1019101.0
2037	MCI	75.8	Male	16	Not Hisp/Latino	White	1	1.3074956	1.4389912	0.5	20	20	1104797.3
2042	MCI	69.5	Male	20	Not Hisp/Latino	White	0	1.2083193	1.0655846	1.5	18	23	1061388.4
2043	MCI	72.2	Female	20	Not Hisp/Latino	White	1	1.2781158	1.2040191	2.0	8	27	1032110.3

Showing 110/13 of 685 entries

Environment History Connections Git

Global Environment

Data

amerge_subset 665 obs. of 17 variables

variable_type_map num [1:17, 1:3] 0 1 0 0 0 0 1 1 0 ...

Values

ids chr [1:665] "2002" "2003" "2007" "2010" "2011" "201..."

MOCA num [1:665] 28 24 23 27 25 26 25 24 24 30 ...

Functions

scatterplot function (x, y, x.lim = NA, y.lim = NA, x.lab = "...")

Files Plots Packages Help Viewer

New Folder Delete Rename More

Home workshops 2019_Rstudio_Magic

Name	Size	Modified
Renviron	52 B	May 12, 2019, 11:33 AM
2019_Rstudio_Magic.Rproj	218 B	May 12, 2019, 6:50 PM
external		
mic		
output		
R		
README.md	42 B	May 12, 2019, 11:29 AM
Rmd		

```
~/workshops/2019_Rstudio_Magic/ >
  Mean : 71.92
  3rd Qu.: 176.60
  Max. : 89.60

  APOE4      FDG      AV45      CDRSB      ADAS13      MOCA
Min. :0.0000 Min. :0.6983 Min. :0.8385 Min. :0.0000 Min. : 0.0 Min. :16.00
1st Qu.:0.0000 1st Qu.:1.1837 1st Qu.:1.0199 1st Qu.:0.0000 1st Qu.: 8.0 1st Qu.:22.00
Median :0.0000 Median :1.2802 Median :1.1105 Median :1.0000 Median :12.0 Median :24.00
Mean :0.5248 Mean :1.2682 Mean :1.1989 Mean :1.2020 Mean :13.8 Mean :23.89
3rd Qu.:1.0000 3rd Qu.:1.3620 3rd Qu.:1.3714 3rd Qu.:2.0000 3rd Qu.:18.0 3rd Qu.:26.00
Max. :2.0000 Max. :1.7011 Max. :2.0256 Max. :5.500 Max. :46.0 Max. :30.00

  WholeBrain
Min. : 817421 Min. : 3731 Min. :12213 Min. : -38.6883 Min. :0.0000
1st Qu.: 984410 1st Qu.: 6510 1st Qu.:18535 1st Qu.: -6.4051 1st Qu.:0.0000
Median :1051621 Median : 7223 Median :20186 Median : -2.5250 Median :1.0000
Mean :1105026 Mean : 7150 Mean :20302 Mean : -3.6882 Mean :0.588
3rd Qu.:1120570 3rd Qu.: 7834 3rd Qu.:22088 3rd Qu.: -0.3482 3rd Qu.:1.0000
Max. :11486036 Max. :110602 Max. :32189 Max. : 5.3540 Max. :3.000

> view(amerge_subset)
>
```

RStudio is more

- ▶ Not just an IDE (integrated development environment)

RStudio is more

- ▶ Not just an IDE (integrated development environment)
- ▶ A company

RStudio is more

- ▶ Not just an IDE (integrated development environment)
- ▶ A company
- ▶ A community

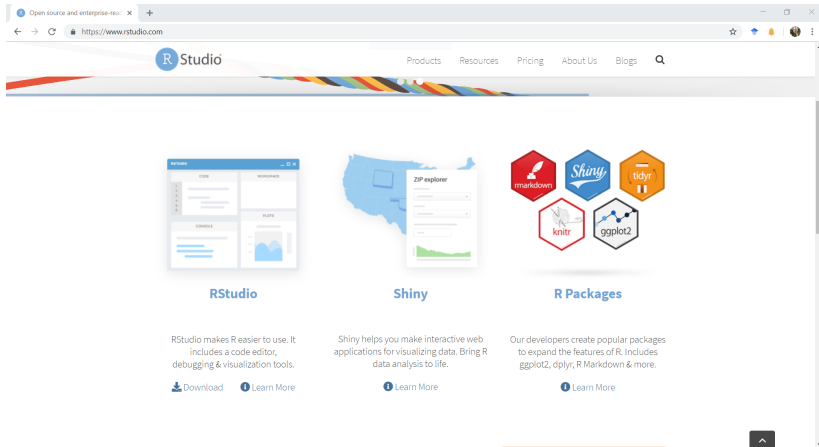
RStudio is more

- ▶ Not just an IDE (integrated development environment)
- ▶ A company
- ▶ A community
- ▶ A conference

RStudio is more

- ▶ Not just an IDE (integrated development environment)
- ▶ A company
- ▶ A community
- ▶ A conference
- ▶ A centralized resource

RStudio Resources



The screenshot shows the RStudio website homepage. At the top is a navigation bar with the RStudio logo and links for Products, Resources, Pricing, About Us, and Blogs. Below the navigation bar is a large graphic featuring a stylized map of the United States with a colorful, multi-colored line passing through it. The main content area is divided into three columns, each representing a different RStudio product or resource. The first column is for RStudio, the second for Shiny, and the third for R Packages. Each column includes a representative image, a title, a brief description, and a link to learn more or download.

Open source and enterprise-ready | <https://www.rstudio.com>

RStudio

Products Resources Pricing About Us Blogs

RStudio

RStudio makes R easier to use. It includes a code editor, debugging & visualization tools.

[Download](#) [Learn More](#)

Shiny

Shiny helps you make interactive web applications for visualizing data. Bring R data analysis to life.

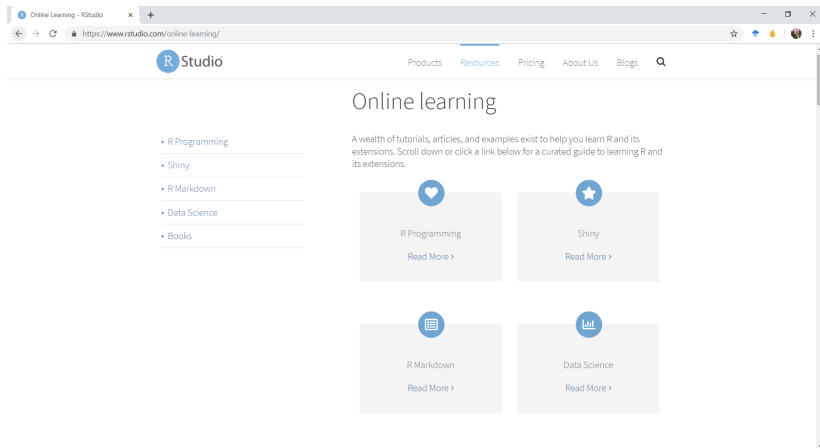
[Learn More](#)

R Packages

Our developers create popular packages to expand the features of R. Includes ggplot2, dplyr, R Markdown & more.

[Learn More](#)

RStudio Resources



The screenshot shows the RStudio website's 'Online Learning' section. The browser's address bar displays 'https://www.rstudio.com/online-learning/'. The website's navigation bar includes links for 'Products', 'Resources' (which is highlighted), 'Pricing', 'About Us', and 'Blogs', along with a search icon. On the left side, there is a vertical list of links: 'R Programming', 'Shiny', 'R Markdown', 'Data Science', and 'Books'. The main content area is titled 'Online learning' and contains a paragraph stating: 'A wealth of tutorials, articles, and examples exist to help you learn R and its extensions. Scroll down or click a link below for a curated guide to learning R and its extensions.' Below this text are four cards arranged in a 2x2 grid. Each card features a blue circular icon at the top, a title, and a 'Read More >' link. The cards are: 1) 'R Programming' with a heart icon, 2) 'Shiny' with a star icon, 3) 'R Markdown' with a document icon, and 4) 'Data Science' with a bar chart icon.

Online Learning - RStudio

https://www.rstudio.com/online-learning/


RStudio

Products Resources Pricing About Us Blogs

Online learning


A wealth of tutorials, articles, and examples exist to help you learn R and its extensions. Scroll down or click a link below for a curated guide to learning R and its extensions.

- [R Programming](#)
- [Shiny](#)
- [R Markdown](#)
- [Data Science](#)
- [Books](#)




R Programming

[Read More >](#)




Shiny

[Read More >](#)



R Markdown

[Read More >](#)




Data Science

[Read More >](#)


RStudio Resources

Cheatsheets - RStudio

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

 RStudio

ProductsResourcesPricingAbout UsBlogs



RStudio Cheat Sheets

The cheat sheets below make it easy to learn about and use some of our favorite packages. From time to time, we will add new cheat sheets to the gallery. If you'd like us to drop you an email when we do, let us know by clicking the button to the right.

SUBSCRIBE TO CHEAT SHEET UPDATES HERE


- RStudio IDE
- R Markdown
- Shiny
- Package Development

- Data Import
- Data Transformation with dplyr
- Data Visualization with ggplot2
- Apply functions with purrr

- Deep Learning with Keras
- Data Science in Spark with Sparklyr
- String manipulation with stringr
- Dates and times with lubridate

Python with R and Reticulate Cheat Sheet

The reticulate package provides a comprehensive set of tools for interoperability between Python and R. With reticulate, you can call Python from R in a variety of ways including importing Python modules into R scripts, writing R Markdown Python chunks, sourcing Python scripts, and using Python interactively within the RStudio IDE. This cheatsheet will remind you how. Updated 4/19.



RStudio Setup

- ▶ For set up:
<https://jennybc.github.io/2014-05-12-ubc/r-setup.html>

RStudio Setup

- ▶ For set up:
<https://jennybc.github.io/2014-05-12-ubc/r-setup.html>
- ▶ For R projects, see

RStudio Setup

- ▶ For set up:
<https://jennybc.github.io/2014-05-12-ubc/r-setup.html>
- ▶ For R projects, see
 - ▶ <https://support.rstudio.com/hc/en-us/articles/200526207-Using-Projects>

RStudio Setup

- ▶ For set up:
<https://jennybc.github.io/2014-05-12-ubc/r-setup.html>
- ▶ For R projects, see
 - ▶ <https://support.rstudio.com/hc/en-us/articles/200526207-Using-Projects>
 - ▶ <https://r4ds.had.co.nz/workflow-projects.html>

R Projects

Compartmentalize & collaborate:

- ▶ RStudio projects

R Projects

Compartmentalize & collaborate:

- ▶ RStudio projects
 - ▶ “RStudio projects make it straightforward to divide your work into multiple contexts, each with their own working directory, workspace, history, and source documents.”

R Projects

Compartmentalize & collaborate:

- ▶ RStudio projects
 - ▶ “RStudio projects make it straightforward to divide your work into multiple contexts, each with their own working directory, workspace, history, and source documents.”
 - ▶ specific projects

R Projects

Compartmentalize & collaborate:

- ▶ RStudio projects
 - ▶ “RStudio projects make it straightforward to divide your work into multiple contexts, each with their own working directory, workspace, history, and source documents.”
 - ▶ specific projects
 - ▶ R package development

R Projects

Compartmentalize & collaborate:

- ▶ RStudio projects
 - ▶ “RStudio projects make it straightforward to divide your work into multiple contexts, each with their own working directory, workspace, history, and source documents.”
 - ▶ specific projects
 - ▶ R package development
 - ▶ cloning from (e.g., Git) repos

New Project

Create Project



New Directory

Start a project in a brand new working directory



Existing Directory

Associate a project with an existing working directory



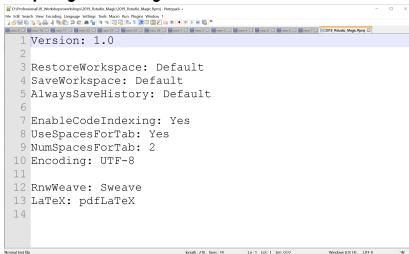
Version Control

Checkout a project from a version control repository



Cancel

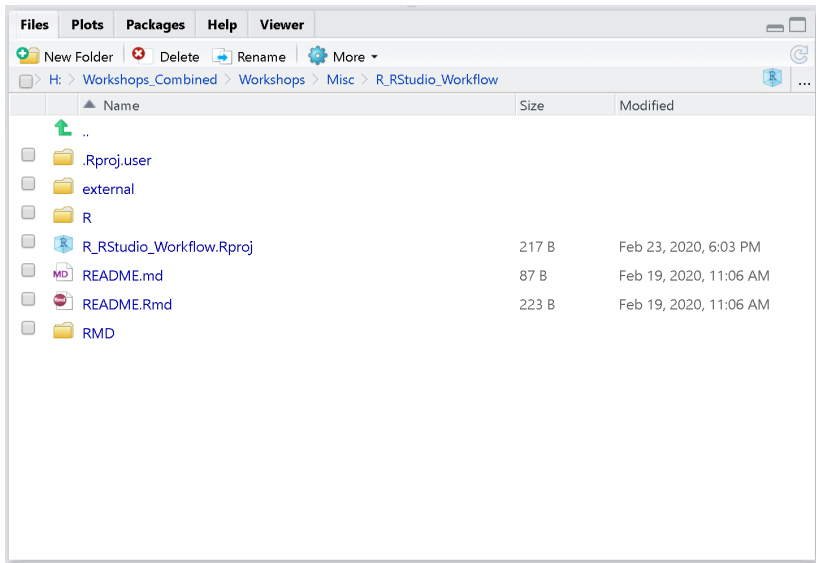
.Rproj files: just a text file with some parameters for start up



The image shows a Notepad++ window titled "C:\Users\user\Documents\RStudio-Magic\RStudio-Magic.Rproj - Notepad++". The window contains a text file with the following content:

```
1 Version: 1.0
2
3 RestoreWorkspace: Default
4 SaveWorkspace: Default
5 AlwaysSaveHistory: Default
6
7 EnableCodeIndexing: Yes
8 UseSpacesForTab: Yes
9 NumSpacesForTab: 2
10 Encoding: UTF-8
11
12 RnwWeave: Sweave
13 LaTeX: pdfLaTeX
14
```

The status bar at the bottom indicates "Normal text file", "length: 278", "lines: 14", "Ln: 1", "Col: 1", "Sel: 0/0", "Windows (2/15)", "UTF-8", and "OK".



Git

What is Git?

- ▶ Version control (like SVN, a.k.a. subversion)

What is Git?

- ▶ Version control (like SVN, a.k.a. subversion)
- ▶ Traditionally for developers/software

What is Git?

- ▶ Version control (like SVN, a.k.a. subversion)
- ▶ Traditionally for developers/software
- ▶ Now more common to “track changes”



Derek
derekbeaton

Edit profile

Post doc at Rotman Research
Institute/Baycrest
Rotman/Baycrest
Toronto
www.derekbeaton.com

★ 100

Organizations



Overview Repositories 24 Projects 0 Packages 0 Stars 0 Followers 22 Following 0

Pinned

OutS

OutS and Robert Shusterman

0 6

ONDRiApps

A (preprocessor) home for ONDRiB curated ShinyApps

0 3

Marvel Cinematic Universe Network

0

GSVD

0 7 4

GPLS

a home for generalized partial least squares

0 2 2

mudler_report

Fetch from cloud:mudler_report

Some r code to visualize the Mudler Report

0

Customize your pins

289 contributions in the last year

Contribution settings



Contribution activity

300

← → ↻ 🏠 <https://github.com/dewkbaator/GOV2/commits/master> ... 🗨️ ⚙️ 🌐 📄

◀ Commits on Nov 14, 2019

deck is a dummy
dewkbaator committed on Nov 14, 2019 Verified [865696](#) [C](#)

◀ Commits on Aug 21, 2019

I removed the warning() calls because they are annoying.
dewkbaator committed on Aug 21, 2019 [8a20787](#) [C](#)

◀ Commits on Aug 20, 2019

caught a small mistake in the documentation.
dewkbaator committed on Aug 20, 2019 [29a0f2a](#) [C](#)

Inclusion of new data (beer tasting notes) and a variety of small cha...
dewkbaator committed on Aug 20, 2019 [9c5a89a](#) [C](#)

◀ Commits on Aug 19, 2019

added beer tasting notes so that there is another ordinal data set av...
dewkbaator committed on Aug 19, 2019 [8a5a899](#) [C](#)

◀ Commits on Aug 14, 2019

updates to small items in documentation and a new data set
dewkbaator committed on Aug 14, 2019 [6a6a33a](#) [C](#)

◀ Commits on Aug 13, 2019

no need for defaults because I check for missing parameters
dewkbaator committed on Aug 13, 2019 [879a486](#) [C](#)

◀ Commits on Aug 6, 2019

fix error.
dewkbaator committed on Aug 6, 2019 [8dca47b](#) [C](#)

◀ Commits on Jul 6, 2019

final documentation update.
dewkbaator committed on Jul 6, 2019 [9f29aef](#) [C](#)

Github

- ▶ As students: You can get free pro accounts

Github

- ▶ As students: You can get free pro accounts
- ▶ And you really really should

Github

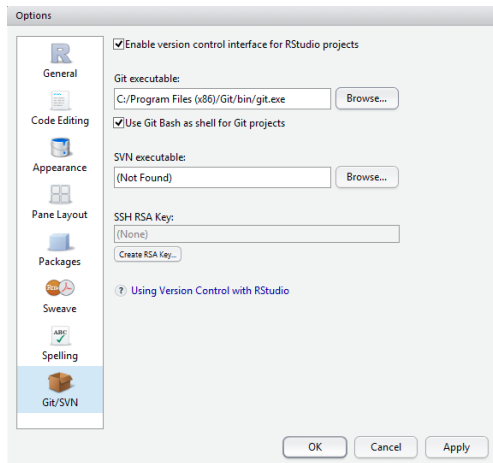
- ▶ As students: You can get free pro accounts
- ▶ And you really really should
- ▶ <https://education.github.com/pack>

Git & R Projects

The premiere Git & R resource: <https://happygitwithr.com/>

Git & R Projects

Download git and link executable within RStudio



Git basics

- ▶ Pull or Fetch: get latest from a repository

Git basics

- ▶ Pull or Fetch: get latest from a repository
- ▶ Commit: make a history of your local changes

Git basics

- ▶ Pull or Fetch: get latest from a repository
- ▶ Commit: make a history of your local changes
- ▶ Push: send your commits to a repository

R

What is R?

- ▶ R is general purpose programming

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”
- ▶ R is a collection of tools

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”
- ▶ R is a collection of tools
 - ▶ Pre-packaged software at your disposal

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”
- ▶ R is a collection of tools
 - ▶ Pre-packaged software at your disposal
- ▶ R is free (as in beer and speech)

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”
- ▶ R is a collection of tools
 - ▶ Pre-packaged software at your disposal
- ▶ R is free (as in beer and speech)
 - ▶ No cost, no restrictions

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”
- ▶ R is a collection of tools
 - ▶ Pre-packaged software at your disposal
- ▶ R is free (as in beer and speech)
 - ▶ No cost, no restrictions
 - ▶ E.g., Microsoft (nee Revolution) R

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”
- ▶ R is a collection of tools
 - ▶ Pre-packaged software at your disposal
- ▶ R is free (as in beer and speech)
 - ▶ No cost, no restrictions
 - ▶ E.g., Microsoft (nee Revolution) R
- ▶ R is a functional language

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”
- ▶ R is a collection of tools
 - ▶ Pre-packaged software at your disposal
- ▶ R is free (as in beer and speech)
 - ▶ No cost, no restrictions
 - ▶ E.g., Microsoft (nee Revolution) R
- ▶ R is a functional language
 - ▶ Mathematical functions

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”
- ▶ R is a collection of tools
 - ▶ Pre-packaged software at your disposal
- ▶ R is free (as in beer and speech)
 - ▶ No cost, no restrictions
 - ▶ E.g., Microsoft (nee Revolution) R
- ▶ R is a functional language
 - ▶ Mathematical functions
 - ▶ Pass expressions and functions to and from functions

What is R?

- ▶ R is general purpose programming
 - ▶ Design around & for statistics
 - ▶ “for and by statisticians”
- ▶ R is a collection of tools
 - ▶ Pre-packaged software at your disposal
- ▶ R is free (as in beer and speech)
 - ▶ No cost, no restrictions
 - ▶ E.g., Microsoft (nee Revolution) R
- ▶ R is a functional language
 - ▶ Mathematical functions
 - ▶ Pass expressions and functions to and from functions
 - ▶ and Turing Complete

Assignment

```
# allowed but not preferred  
a_variable = 10 + 1  
# preferred  
a_variable <- 10 + 1  
# a bonus  
10 + 1 -> a_variable
```

Dots

```
# allowed but not preferred  
a.variable = 10 + 1  
  ## dots have 2 meanings in R,  
    ## with a 3rd in the tidyverse  
  
# preferred  
a_variable <- 10 + 1
```

“Reserved” characters

- ▶ c, q, t, C, D, I, F, and T (via https://www.johndcook.com/blog/r_language_for_programmers/)

“Reserved” characters

- ▶ c, q, t, C, D, I, F, and T (via https://www.johndcook.com/blog/r_language_for_programmers/)
- ▶ Except that these can be redefined

R: Data Structures

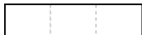
R: Data Structures

single type

multiple types

1D

Vector

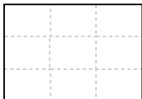


List

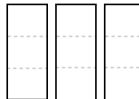


2D

Matrix

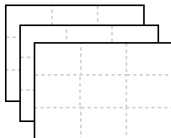


Data frame



nD

Array



See <https://rstudio-education.github.io/hopr/r-objects.html>

VECTOR

```
a_vector <- c(2, 0, 2, 0, 0, 2, 2, 5)
```

```
a_vector[1]  
>2
```

```
a_vector[4]  
>0
```

2	1
0	2
2	3
0	4
0	5
2	6
2	7
5	8

MATRIX

```
a_matrix <- matrix(c(2, 0, 2, 0, 0, 2, 2, 5), nrow = 4, ncol = 2)
```

```
a_matrix[1,1]  
>2
```

```
a_matrix[1,2]  
>0
```

```
a_matrix[4,2]  
>5
```

```
a_matrix[4,]  
>0 5
```

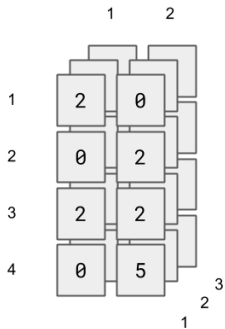
	1	2
1	2	0
2	0	2
3	2	2
4	0	5

ARRAY

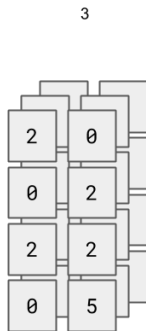
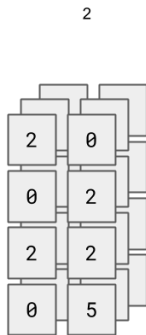
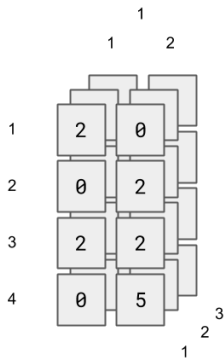
```
an_array[1,1,1]  
>2
```

```
an_array[1,4,2]
```

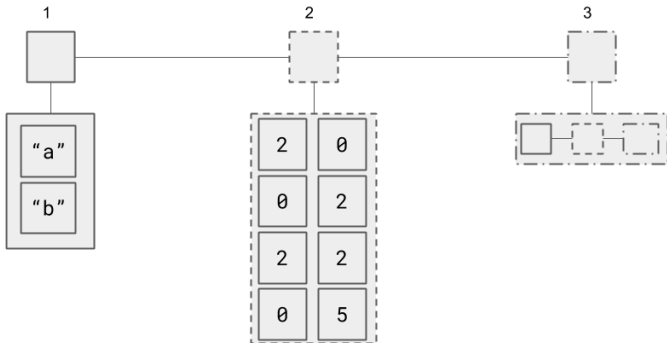
```
an_array[, ,1]
```



ARRAY



LIST



DATA FRAMES

1	2	3	
2	"A"	T	1
0	"C"	T	2
1	"D"	F	3
0	"C"	T	4

R: Data Structures

- ▶ `list[[1]]` or `list$name`

R: Data Structures

- ▶ `list[[1]]` or `list$name`
- ▶ `data.frame[[1]][1]` or `data.frame[1,1]` or `data.frame$name`

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal
 - ▶ Includes NaN, Inf, -Inf

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal
 - ▶ Includes NaN, Inf, -Inf
 - ▶ character

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal
 - ▶ Includes NaN, Inf, -Inf
 - ▶ character
 - ▶ logical

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal
 - ▶ Includes NaN, Inf, -Inf
 - ▶ character
 - ▶ logical
 - ▶ includes NA, T, TRUE, F, FALSE

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal
 - ▶ Includes NaN, Inf, -Inf
 - ▶ character
 - ▶ logical
 - ▶ includes NA, T, TRUE, F, FALSE
- ▶ factor

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal
 - ▶ Includes NaN, Inf, -Inf
 - ▶ character
 - ▶ logical
 - ▶ includes NA, T, TRUE, F, FALSE
- ▶ factor
 - ▶ factors are usually not your friends

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal
 - ▶ Includes NaN, Inf, -Inf
 - ▶ character
 - ▶ logical
 - ▶ includes NA, T, TRUE, F, FALSE
- ▶ factor
 - ▶ factors are usually not your friends
 - ▶ with `read.csv(): stringsAsFactors = F` or convert these

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal
 - ▶ Includes NaN, Inf, -Inf
 - ▶ character
 - ▶ logical
 - ▶ includes NA, T, TRUE, F, FALSE
- ▶ factor
 - ▶ factors are usually not your friends
 - ▶ with `read.csv(): stringsAsFactors = F` or convert these
 - ▶ `stringsAsFactors = F` as default in R 4.0.0

R: Data types

- ▶ All of them are here: <https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Objects>
- ▶ The most common you'll use:
 - ▶ numeric
 - ▶ real or decimal
 - ▶ Includes NaN, Inf, -Inf
 - ▶ character
 - ▶ logical
 - ▶ includes NA, T, TRUE, F, FALSE
- ▶ factor
 - ▶ factors are usually not your friends
 - ▶ with `read.csv(): stringsAsFactors = F` or convert these
 - ▶ `stringsAsFactors = F` as default in R 4.0.0
 - ▶ or use tibbles in the tidyverse

R: factor disasters

```
a_numeric_vector <- c(3, 0, 1, -2, 2, 5, 5, 2, 1)
(a_numeric_vector + 1)
```

```
## [1] 4 1 2 -1 3 6 6 3 2
```

```
a_numeric_vector <- c(3, 0, 1, -2, 2, 5, 5, 2, 1)
(a_numeric2factor_vector <- as.factor(a_numeric_vector))
```

```
## [1] 3 0 1 -2 2 5 5 2 1
```

```
## Levels: -2 0 1 2 3 5
```

```
a_numeric_vector <- c(3, 0, 1, -2, 2, 5, 5, 2, 1)
(a_numeric2factor_vector <- as.factor(a_numeric_vector))
```

```
## [1] 3 0 1 -2 2 5 5 2 1
## Levels: -2 0 1 2 3 5
```

```
(as.numeric(a_numeric2factor_vector))
```

```
## [1] 5 2 3 1 4 6 6 4 3
```

```
(as.numeric(a_numeric2factor_vector) + 1)
```

```
## [1] 6 3 4 2 5 7 7 5 4
```



```
a_numeric_vector <- c(3, 0, 1, -2, 2, 5, 5, 2, 1)
(a_numeric2factor_vector <- as.factor(a_numeric_vector))
```

```
## [1] 3 0 1 -2 2 5 5 2 1
## Levels: -2 0 1 2 3 5
```

```
(as.character(a_numeric2factor_vector))
```

```
## [1] "3" "0" "1" "-2" "2" "5" "5" "2" "1"
```

```
(as.numeric(as.character(a_numeric2factor_vector)))
```

```
## [1] 3 0 1 -2 2 5 5 2 1
```

Cheatsheet for base R

Base R Cheat Sheet

Getting Help

Accessing the help files

?mean

Get help of a particular function.

help.search('weighted mean')

Search the help files for a word or phrase.

help(package = 'dplyr')

Find help for a package.

More about an object

str(iris)

Get a summary of an object's structure.

class(iris)

Find the class an object belongs to.

Using Libraries

install.packages('dplyr')

Download and install a package from CRAN.

library(dplyr)

Load the package into the session, making all its functions available to use.

dplyr::select

Use a particular function from a package.

data(iris)

Load a built-in dataset into the environment.

Working Directory

getwd()

Find the current working directory (where inputs are found and outputs are sent).

setwd('C://file/path')

Change the current working directory.

Use projects in RStudio to set the working directory to the folder you are working in.

Vectors

Creating Vectors

c(2, 4, 5)	2 4 5	Join elements into a vector
2:6	2 3 4 5 6	An integer sequence
seq(2, 3, by=0.5)	2.0 2.5 3.0	A complex sequence
rep(1:2, times=3)	1 2 1 2 1 2	Repeat a vector
rep(1:2, each=3)	1 1 1 2 2 2	Repeat elements of a vector

Vector Functions

sort(x)	rev(x)
Return x sorted.	Return x reversed.
table(x)	unique(x)
See counts of values.	See unique values.

Selecting Vector Elements

By Position

x[4]	The fourth element.
x[-4]	All but the fourth.
x[2:4]	Elements two to four.
x[-(2:4)]	All elements except two to four.
x[c(1, 5)]	Elements one and five.

By Value

x[x == 10]	Elements which are equal to 10.
x[x < 0]	All elements less than zero.
x[x %in% c(1, 2, 5)]	Elements in the set 1, 2, 5.

Named Vectors

x['apple']	Element with name 'apple'.
-------------------	----------------------------

Programming

For Loop

```
for (variable in sequence){  
  Do something  
}
```

Example

```
for (i in 1:4){  
  j <- i + 10  
  print(j)  
}
```

While Loop

```
while (condition){  
  Do something  
}
```

Example

```
while (i < 5){  
  print(i)  
  i <- i + 1  
}
```

If Statements

```
if (condition){  
  Do something  
} else {  
  Do something different  
}
```

Example

```
if (i > 3){  
  print('Yes')  
} else {  
  print('No')  
}
```

Functions

```
function_name <- function(var){  
  Do something  
  return(new_variable)  
}
```

Example

```
square <- function(x){  
  squared <- xxx  
  return(squared)  
}
```

Reading and Writing Data

Input	Output	Description
df <- read.table('file.txt')	write.table(df, 'file.txt')	Read and write a delimited text file.
df <- read.csv('file.csv')	write.csv(df, 'file.csv')	Read and write a comma separated value file. This is a special case of read.table/write.table.
load('file.Rdata')	save(df, file = 'file.Rdata')	Read and write an R data file, a file type special for R.

Conditions

a == b	Are equal	a > b	Greater than	a >= b	Greater than or equal to	is.na(a)	is missing
a != b	Not equal	a < b	Less than	a <= b	Less than or equal to	is.null(a)	is null

Tidyverse

- ▶ tidyverse: “an opinionated collection of R packages designed for data science [that] share an underlying design philosophy, **grammar, and data structures.**”

Tidyverse

- ▶ tidyverse: “an opinionated collection of R packages designed for data science [that] share an underlying design philosophy, **grammar, and data structures.**”
 - ▶ A sublanguage or dialect

Tidyverse

- ▶ tidyverse: “an opinionated collection of R packages designed for data science [that] share an underlying design philosophy, **grammar, and data structures.**”
 - ▶ A sublanguage or dialect
- ▶ Strongly built around a style:

Tidyverse

- ▶ tidyverse: “an opinionated collection of R packages designed for data science [that] share an underlying design philosophy, **grammar, and data structures.**”
 - ▶ A sublanguage or dialect
- ▶ Strongly built around a style:
 - ▶ objects are nouns

Tidyverse

- ▶ tidyverse: “an opinionated collection of R packages designed for data science [that] share an underlying design philosophy, **grammar, and data structures.**”
 - ▶ A sublanguage or dialect
- ▶ Strongly built around a style:
 - ▶ objects are nouns
 - ▶ functions are verbs

Tidyverse

- ▶ tidyverse: “an opinionated collection of R packages designed for data science [that] share an underlying design philosophy, **grammar, and data structures.**”
 - ▶ A sublanguage or dialect
- ▶ Strongly built around a style:
 - ▶ objects are nouns
 - ▶ functions are verbs
- ▶ Core packages:

Tidyverse

- ▶ tidyverse: “an opinionated collection of R packages designed for data science [that] share an underlying design philosophy, **grammar, and data structures.**”
 - ▶ A sublanguage or dialect
- ▶ Strongly built around a style:
 - ▶ objects are nouns
 - ▶ functions are verbs
- ▶ Core packages:
 - ▶ ggplot2, dplyr, tidyr, readr, tibble, stringr, purrr

Tidyverse

- ▶ tidyverse: “an opinionated collection of R packages designed for data science [that] share an underlying design philosophy, **grammar, and data structures.**”
 - ▶ A sublanguage or dialect
- ▶ Strongly built around a style:
 - ▶ objects are nouns
 - ▶ functions are verbs
- ▶ Core packages:
 - ▶ ggplot2, dplyr, tidyr, readr, tibble, stringr, purrr
 - ▶ <https://www.tidyverse.org/>

tidyverse cheatsheet

R For Data Science Cheat Sheet

Tidyverse for Beginners

Learn More R for Data Science Interactively at www.datacamp.com



Tidyverse

The tidyverse is a powerful collection of R packages that are actually data tools for transforming and visualizing data. All packages of the tidyverse share an underlying philosophy and common APIs.

The core packages are:



- **ggplot2**, which implements the grammar of graphics. You can use it to visualize your data.



- **dplyr** is a grammar of data manipulation. You can use it to solve the most common data manipulation challenges.



- **tidyr** helps you to create tidy data or data where each variable is in a column, each observation is a row and each value is a cell.



- **readr** is a fast and friendly way to read rectangular data.



- **purrr** enhances R's functional programming (FP) toolkit by providing a complete and consistent set of tools for working with functions and vectors.



- **tibble** is a modern re-imagining of the data frame.



- **stringr** provides a cohesive set of functions designed to make working with strings as easy as possible



- **forcats** provide a suite of useful tools that solve common problems with factors.

You can install the complete tidyverse with:

```
> install.packages("tidyverse")
```

Then, load the core tidyverse and make it available in your current R session by running:

```
> library(tidyverse)
```

Note: there are many other tidyverse packages with more specialized usage. They are not loaded automatically with `library(tidyverse)`, so you'll need to load each one with its own `load` library().

Useful Functions

```
> tidyverse_conflicts() Conflicts between tidyverse and other packages
> tidyverse_deps() List all tidyverse dependencies
> tidyverse_log() Get tidyverse logs, using ASCII or unicode characters
> tidyverse_packages() List all tidyverse packages
> tidyverse_update() Update tidyverse packages
```

Loading in the data

```
> library(datasets) Load the datasets package
> library(ggmapr) Load the ggmapr package
> attach(iris) Attach its data to the R search path
```

dplyr

Filter

`filter()` allows you to select a subset of rows in a data frame.

```
> iris %>%
  filter(Species=="virginica") Select iris data of species "virginica"
> iris %>%
  filter(Species=="virginica", Sepal.Length > 6) Select iris data of species "virginica" and sepal length greater than 6.
```

Arrange

`arrange()` sorts the observations in a dataset in ascending or descending order based on one of its variables.

```
> iris %>%
  arrange(Sepal.Length) Sort in ascending order of sepal length
> iris %>%
  arrange(desc(Sepal.Length)) Sort in descending order of sepal length
```

Combine multiple `dplyr` verbs in a row with the pipe operator `%>%`:

```
> iris %>%
  filter(Species=="virginica") %>% Filter for species "virginica"
  arrange(desc(Sepal.Length)) then arrange in descending order of sepal length
```

Mutate

`mutate()` allows you to update or create new columns of a data frame.

```
> iris %>%
  mutate(Sepal.Length=Sepal.Length*10) Change Sepal.Length to be in millimeters
> iris %>%
  mutate(SLmm=Sepal.Length*10) Create a new column called SLmm
```

Combine the verbs `filter()`, `arrange()`, and `mutate()`:

```
> iris %>%
  filter(Species=="virginica") %>% Summarize to find the median sepal length
  mutate(SLmm=Sepal.Length*10) %>% Filter for virginica then summarize the median sepal length
  arrange(desc(SLmm))
```

Summarize

`summarize()` allows you to turn many observations into a single data point.

```
> iris %>%
  summarize(medianSL=median(Sepal.Length)) Summarize to find the median sepal length
> iris %>%
  filter(Species=="virginica") %>% Filter for virginica then summarize the median sepal length
  summarize(medianSL=median(Sepal.Length))
```

You can also summarize multiple variables at once:

```
> iris %>%
  filter(Species=="virginica") %>% Summarize to find the median sepal length
  summarize(medianSL=median(Sepal.Length), maxSL=max(Sepal.Length))
```

`group_by()` allows you to summarize within groups instead of summarizing the entire dataset:

```
> iris %>%
  group_by(Species) %>% Find median and max sepal length of each species
  summarize(medianSL=median(Sepal.Length), maxSL=max(Sepal.Length))
```

```
> iris %>%
  filter(Sepal.Length>6) %>% Find median and max petal length of each species with sepal length > 6
  summarize(medianPL=median(Petal.Length), maxPL=max(Petal.Length))
```

ggplot2

Scatter plot

Scatter plots allow you to compare two variables within your data. To do this with `ggplot2`, you use `geom_point()`:

```
> iris_small <- iris %>%
  filter(Sepal.Length > 5)
> ggplot(iris_small, aes(x=Petal.Length, y=Petal.Width)) + Compare petal width and length
  geom_point()
```

Additional Aesthetics

• Color

```
> ggplot(iris_small, aes(x=Petal.Length, y=Petal.Width, color=Species)) +
  geom_point()
```

• Size

```
> ggplot(iris_small, aes(x=Petal.Length, y=Petal.Width, color=Species, size=Sepal.Length)) +
  geom_point()
```

Faceting

```
> ggplot(iris_small, aes(x=Petal.Length, y=Petal.Width)) +
  geom_point() + facet_wrap(~Species)
```

Line Plots

```
> by_year <- ggplot(iris_small)
  group_by(year) %>% summarize(medianSL=median(Sepal.Length))
> ggplot(by_year, aes(x=year, y=medianSL)) +
  geom_line() + expand_limits(y=0)
```

Bar Plots

```
> by_species <- iris %>%
  filter(Sepal.Length>6) %>%
  group_by(Species) %>% summarize(medianPL=median(Petal.Length))
> ggplot(by_species, aes(x=Species, y=medianPL)) +
  geom_col()
```

Histograms

```
> ggplot(iris_small, aes(x=Petal.Length)) +
  geom_histogram()
```

Box Plots

```
> ggplot(iris_small, aes(x=Species, y=Sepal.Length)) +
  geom_boxplot()
```

DataCamp
Learn R for Data Science Interactively



RMarkdown

RMarkdown

- ▶ A simple markdown language

RMarkdown

- ▶ A simple markdown language
- ▶ Create documents (or slides, websites, books, notebooks, etc. . .)

RMarkdown

- ▶ A simple markdown language
- ▶ Create documents (or slides, websites, books, notebooks, etc. . .)
 - ▶ <https://bookdown.org/yihui/rmarkdown/>

RMarkdown

- ▶ A simple markdown language
- ▶ Create documents (or slides, websites, books, notebooks, etc. . .)
 - ▶ <https://bookdown.org/yihui/rmarkdown/>
 - ▶ See also: <https://ryanpeek.github.io/2020-02-20-10-tips-to-souping-up-rmarkdown/>

RMarkdown

- ▶ A simple markdown language
- ▶ Create documents (or slides, websites, books, notebooks, etc. . .)
 - ▶ <https://bookdown.org/yihui/rmarkdown/>
 - ▶ See also: <https://ryanpeek.github.io/2020-02-20-10-tips-to-souping-up-rmarkdown/>
- ▶ “Literate programming”

RMarkdown

- ▶ A simple markdown language
- ▶ Create documents (or slides, websites, books, notebooks, etc. . .)
 - ▶ <https://bookdown.org/yihui/rmarkdown/>
 - ▶ See also: <https://ryanpeek.github.io/2020-02-20-10-tips-to-souping-up-rmarkdown/>
- ▶ “Literate programming”
 - ▶ Text, headers, sections

RMarkdown

- ▶ A simple markdown language
- ▶ Create documents (or slides, websites, books, notebooks, etc. . .)
 - ▶ <https://bookdown.org/yihui/rmarkdown/>
 - ▶ See also: <https://ryanpeek.github.io/2020-02-20-10-tips-to-souping-up-rmarkdown/>
- ▶ “Literate programming”
 - ▶ Text, headers, sections
 - ▶ Figures, tables

RMarkdown

- ▶ A simple markdown language
- ▶ Create documents (or slides, websites, books, notebooks, etc. . .)
 - ▶ <https://bookdown.org/yihui/rmarkdown/>
 - ▶ See also: <https://ryanpeek.github.io/2020-02-20-10-tips-to-souping-up-rmarkdown/>
- ▶ “Literate programming”
 - ▶ Text, headers, sections
 - ▶ Figures, tables
 - ▶ Code to generate those

RMarkdown

Generate reports:

- ▶ HTML

RMarkdown

Generate reports:

- ▶ HTML
- ▶ Word

RMarkdown

Generate reports:

- ▶ HTML
- ▶ Word
- ▶ PDF

RMarkdown

Generate reports:

- ▶ HTML
- ▶ Word
- ▶ PDF
 - ▶ With LaTeX

RMarkdown

```
---  
title: "Hello R Markdown"  
author: "Awesome Me"  
date: "2018-02-14"  
output: html_document  
---
```

This is a paragraph in an R Markdown document.

Below is a code chunk:

```
```{r}  
fit = lm(dist ~ speed, data = cars)
b = coef(fit)
plot(cars)
abline(fit)
```
```

RMarkdown Don'ts

- ▶ Don't hardcode values or absolute file paths

RMarkdown Don'ts

- ▶ Don't hardcode values or absolute file paths
 - ▶ see `here::here()`

RMarkdown Don'ts

- ▶ Don't hardcode values or absolute file paths
 - ▶ see `here::here()`
 - ▶ Use projects (`.Rproj`)

RMarkdown Don'ts

- ▶ Don't hardcode values or absolute file paths
 - ▶ see `here::here()`
 - ▶ Use projects (`.Rproj`)
- ▶ Don't do complicated or expensive stuff

RMarkdown Don'ts

- ▶ Don't hardcode values or absolute file paths
 - ▶ see `here::here()`
 - ▶ Use projects (`.Rproj`)
- ▶ Don't do complicated or expensive stuff
 - ▶ Database queries

RMarkdown Don'ts

- ▶ Don't hardcode values or absolute file paths
 - ▶ see `here::here()`
 - ▶ Use projects (`.Rproj`)
- ▶ Don't do complicated or expensive stuff
 - ▶ Database queries
 - ▶ Resampling

All together

Within RStudio

- ▶ Integration with version control (git or SVN)

Within RStudio

- ▶ Integration with version control (git or SVN)
- ▶ R Markdown

Within RStudio

- ▶ Integration with version control (git or SVN)
- ▶ R Markdown
 - ▶ Save and execute code

Within RStudio

- ▶ Integration with version control (git or SVN)
- ▶ R Markdown
 - ▶ Save and execute code
 - ▶ Generate high quality reports that can be shared

Within RStudio

- ▶ Integration with version control (git or SVN)
- ▶ R Markdown
 - ▶ Save and execute code
 - ▶ Generate high quality reports that can be shared
 - ▶ Create presentations (like this one!)

Within RStudio

- ▶ Integration with version control (git or SVN)
- ▶ R Markdown
 - ▶ Save and execute code
 - ▶ Generate high quality reports that can be shared
 - ▶ Create presentations (like this one!)
 - ▶ See https://github.com/derekbeaton/Workshops/tree/master/Misc/R_RStudio_Workflow

Within RStudio

- ▶ Integration with version control (git or SVN)
- ▶ R Markdown
 - ▶ Save and execute code
 - ▶ Generate high quality reports that can be shared
 - ▶ Create presentations (like this one!)
 - ▶ See https://github.com/derekbeaton/Workshops/tree/master/Misc/R_RStudio_Workflow
- ▶ Python, D3 (JavaScript), SQL, Shiny, LaTeX, HTML/CSS

Within RStudio

- ▶ Integration with version control (git or SVN)
- ▶ R Markdown
 - ▶ Save and execute code
 - ▶ Generate high quality reports that can be shared
 - ▶ Create presentations (like this one!)
 - ▶ See https://github.com/derekbeaton/Workshops/tree/master/Misc/R_RStudio_Workflow
- ▶ Python, D3 (JavaScript), SQL, Shiny, LaTeX, HTML/CSS
- ▶ And so much more

Part 2: Working with data

Part 2: Working with data

We'll move to R scripts and another RMarkdown document