

Introduction to R for Biologists

Day 1 – Intro to R and the Tidyverse Ecosystem

Developed by Rachael Cox

Welcome to Class!!



Alexandra Lukasiewicz
(she/they)
PhD Candidate
Contreras Lab

Course Summary

Day 1: Introduction to R syntax and the Rstudio Interface

Day 2: Making plots in the Tidyverse (ggplot)

Day 3: Manipulating data in the Tidyverse (dplyr)

Day 4: Statistical Applications of R

Course Summary

Day 1: Introduction to R syntax and the Rstudio Interface

Day 2: Making plots in the Tidyverse (ggplot)

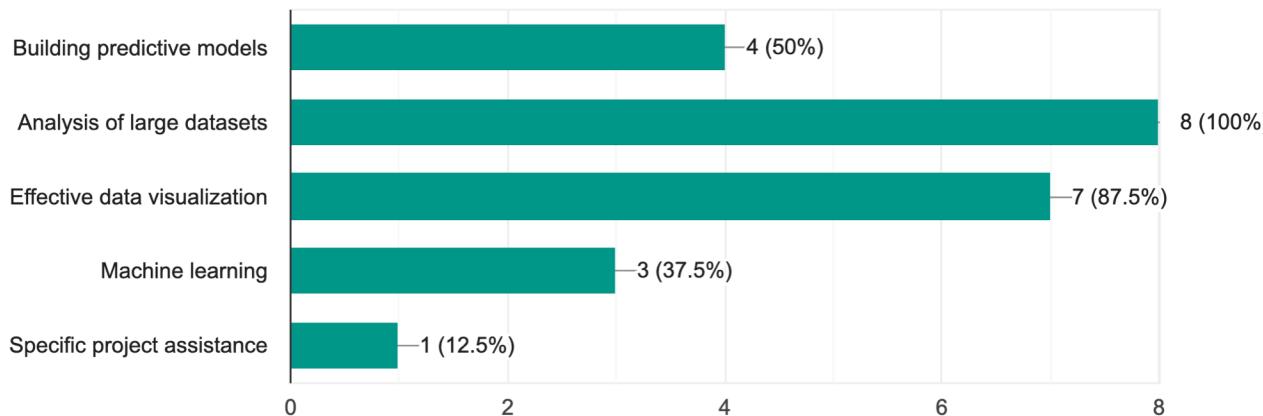
Day 3: Manipulating data in the Tidyverse (dplyr)

Day 4: Statistical Applications of R



Are there specific topics you would like to be covered during the course?

8 responses



Course Format

Concept Slides and Code Along Activities

- **Lecture -> Coding -> Lecture -> Coding**
- Check email for link to my Github
 - Download zip file for day 1
 - https://github.com/ajlukasiewicz/CBRS_R_Intr_o_2024/day_1_material

Course Format

Philip Sweet / Intro_R_Day1 Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file Add file Code

Local Codespaces New

Clone

HTTPS SSH GitHub CLI

https://github.com/philipjsweet/Intro_R_D

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

Help people interested in this repository understand your project by adding a README.md

This screenshot shows a GitHub repository page for 'philipjsweet / Intro_R_Day1'. The repository is public and contains one branch ('main') and no tags. The repository has 1 branch and 0 tags. The main content area displays three files: 'Day_1 Worksheet', 'day1.pdf', and 'mushrooms.csv', all of which were 'Set up repo'. A light blue callout box at the bottom left encourages adding a README.md. On the right, there's a 'Code' dropdown menu with options for 'Local' and 'Codespaces', and a 'Clone' section with links for HTTPS, SSH, and GitHub CLI, along with a copy link. Below that is a section for opening the repository with GitHub Desktop. At the very bottom, a yellow box highlights the 'Download ZIP' button.

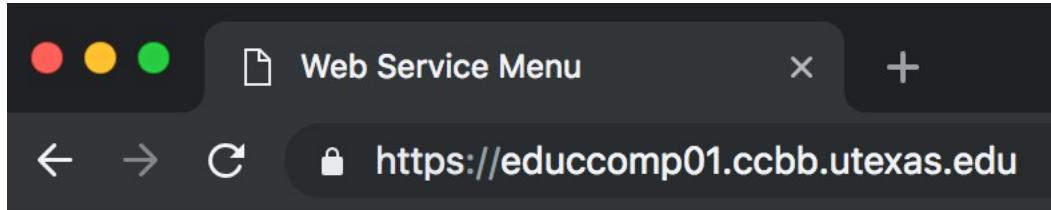
Day 1 Outline

1. How to get set up using R (On the POD and Locally)
2. How and why to use RStudio & R Markdown (.Rmd)
3. Basics of programming
 - Data types
 - Functions
 - Troubleshooting
4. Intro to the Tidyverse
 - Tidy vs untidy data
 - Tidyverse-specific functions

Access R Studio through your web browser

1. <https://gsafcomp01.ccbb.utexas.edu/>
2. <https://gsafcomp02.ccbb.utexas.edu/>

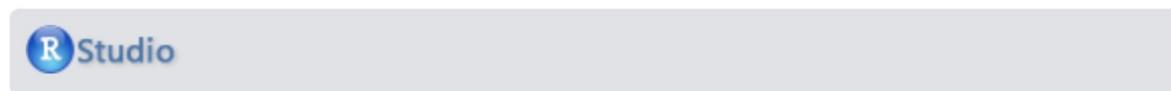
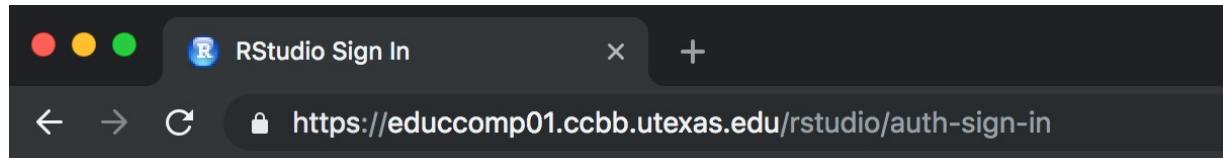
Select RStudio



Please choose one of the following applications:

- [RStudio](#) ←—————
- [Jupyterhub](#)

Sign in with your student# and password



Refer to “Student Accounts” file for your username

A close-up of the RStudio sign-in form. A red arrow points to the "Username:" field, which is circled in red. The form includes fields for "Username" and "Password", a "Stay signed in" checkbox, and a "Sign In" button.

R: The premier data analysis and visualization platform

Step 1: install R

<https://cran.r-project.org/>



The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper

R Studio: A nice user interface for R

Step 2: Install RStudio

<https://www.rstudio.com/products/rstudio/download/>



The screenshot shows the R Studio interface. On the left, the code editor displays an R Markdown file named 'class1.Rmd'. The code includes R global options, library imports (knitr), and a header for an in-class worksheet. It also contains a date entry and two sections of explanatory text. The middle pane shows the 'Environment' tab, which is currently empty. The bottom right corner features a 'R Resources' sidebar with links to various R-related resources like 'Learning R Online' and 'RStudio IDE Support'.

Code in class1.Rmd:

```
1  ```{r global_options, include=FALSE}
2  library(knitr)
3  opts_chunk$set(fig.align="center", fig.height=4, fig.width=4)
4  ```
5  ##In-class worksheet 1
6
7  **Jan 17, 2017**
8
9
10 Much of the work in this class will be done via R Markdown documents. R Markdown documents are documents that combine text, R code, and R output, including figures. They are a great way to produce self-contained and documented statistical analyses.
11
12 In this first worksheet, you will learn how to do some basic markdown editing. After you have made a change to the document, press "Knit HTML" in R Studio and see what kind of a result you get.
13
14 Edit only below this line.
15
```

Environment is empty

R Resources

- Learning R Online
- CRAN Task Views
- R on StackOverflow
- Getting Help with R
- RStudio IDE Support
- RStudio Cheat Sheets
- RStudio Tip of the Day
- RStudio Packages
- RStudio Products

Note about local installation

Installing R locally will require you to install additional packages, such as "Tidyverse" and "DESeq"

I will add the installation code for these in the markdown files

R Markdown

R Markdown: Open the markdown

The screenshot shows the RStudio interface running in a web browser at gsafcomp01.ccbb.utexas.edu/rstudio/. The browser's address bar and various tabs are visible at the top.

Console Pane:

```
R version 3.6.3 (2020-02-29) -- "Holding the Windsock"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(ggthemes)
Session restored from your saved work on 2023-May-25 21:29:05 UTC (3 hours ago)
> |
```

Environment Pane:

Environment is empty

Files Pane:

| Name | Size | Modified |
|------|------|----------|
| R | | |

R Markdown: Open the markdown

The screenshot shows the RStudio interface running in a web browser at gsafcomp01.ccbb.utexas.edu/rstudio/. The top navigation bar includes links for Function reference, Binary classification, Keitz Chamber, Morpheus, UT Account Information, Grep, Learn to purrr, Bike Rides in Texas, and Other Bookmarks. The RStudio menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The left sidebar has tabs for Console, Terminal, and Background Jobs, showing the R 3.6.3 startup message. The main workspace consists of three panes: Environment (Global Environment is empty), Plots (empty), and Files (with an 'Upload' button highlighted by a red box). The bottom status bar indicates the session was restored from work on 2023-May-25 21:29:05 UTC (3 hours ago).

```
R version 3.6.3 (2020-02-29) -- "Holding the Windsock"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(ggthemes)
Session restored from your saved work on 2023-May-25 21:29:05 UTC (3 hours ago)
>
```

R Markdown: Open the markdown

The screenshot shows the RStudio interface running in a web browser. The title bar indicates the URL is `gsafcomp01.ccbb.utexas.edu/rstudio/`. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The top right corner shows a user profile for "student48" and a "Project: (None)" dropdown. The main workspace consists of several panes: a left pane with "Console", "Terminal", and "Background Jobs"; a top-right pane with tabs for "Environment", "History", "Connections", and "Tutorial"; and a bottom-right pane for file management. A central modal dialog box is open, titled "Upload Files". It has fields for "Target directory:" (set to "~") and "File to upload:" (with a "Choose File" button and a message "No file chosen"). Below these is a tip: "TIP: To upload multiple files or a directory, create a zip file. The zip file will be automatically expanded after upload." At the bottom of the dialog are "OK" and "Cancel" buttons. The console pane contains R startup messages, including the R version 3.6.3 welcome screen and a note about natural language support. The bottom-left pane shows a command-line session starting with `> library(ggthemes)`.

```
R version 3.6.3 (2020-02-29) -- "Holding the Windsock"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(ggthemes)
Session restored from your saved work on 2023-May-25 21:29:05 UTC (3 hours ago)
>
```

R Markdown: Open the markdown

The screenshot shows the RStudio interface running in a web browser window. The title bar indicates the URL is `gsafcomp01.ccbb.utexas.edu/rstudio/`. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The left sidebar has tabs for Console, Terminal, and Background Jobs, with the Console tab active. The main workspace shows R version 3.6.3 starting up. A modal dialog box titled "Upload Files" is open in the center. It has fields for "Target directory:" (set to "~") and "File to upload:" with a "Choose File" button highlighted by a red box. Below the file input field is a tip: "TIP: To upload multiple files or a directory, create a zip file. The zip file will be automatically expanded after upload." At the bottom of the dialog are "OK" and "Cancel" buttons. The right side of the screen shows the Environment, History, Connections, and Tutorial panes, and a Global Environment view.

R version 3.6.3 (2020-02-29) -- "Holding the Windsock"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

```
> library(ggthemes)
Session restored from your saved work on 2023-May-25 21:29:05 UTC (3 hours ago)
>
```

R Markdown: Open the markdown

Demonstration Time!!

The screenshot shows the RStudio interface with the following details:

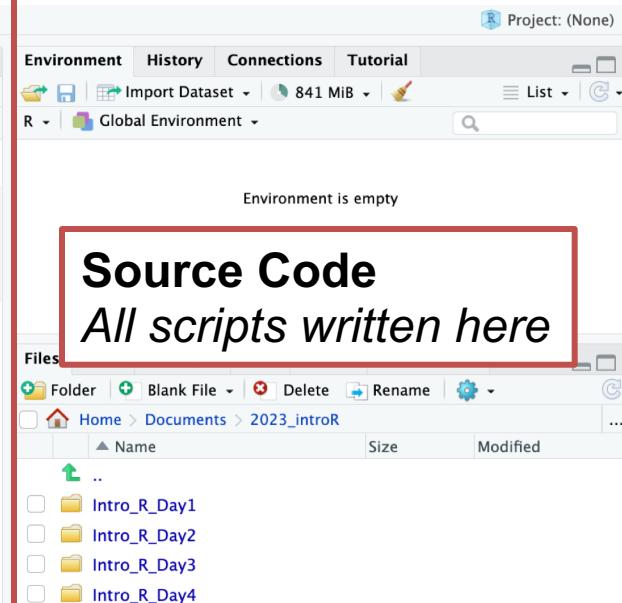
- Header Bar:** Shows the URL gsafcomp01.cccb.utexas.edu/rstudio/, a search bar, and various bookmarks.
- Toolbar:** Includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help, and Addins menus.
- Console Tab:** Displays R version 3.6.3 startup messages and a command history.
- Environment Tab:** Shows the Global Environment with an empty list.
- File Menu:** Contains options like New Blank File, Upload, Delete, Rename, More, and a dropdown for Size and Modified.
- Central Area:** An "Upload Files" dialog box is open, prompting for a target directory and a file to upload. The "Choose File" button is highlighted with a red box.
- Bottom:** A command line interface with the prompt > |.

R Markdown: Relevant Sections

```
## Developed by Rachael Cox
``{r global_options, include=FALSE}
library(knitr)
library(tidyverse)
opts_chunk$set(fig.align="center", fig.height=4, fig.width=4)
```
Day 1: Introduction to R
In-class worksheet
May 31st, 2024

Computational analyses require methods and notes to be recorded the same way you would for wet lab experiments. An excellent way to do this is via R Markdown documents. R Markdown documents are documents that combine text, R code, and R code output, and figures. They are a great way to produce self-contained and documented statistical analyses.

In this first worksheet, you will learn how to do some basic markdown editing in addition to the basic use of variables and functions in R. After you have made a change to the document, press "Knit HTML" in R Studio and see what kind of a result
```



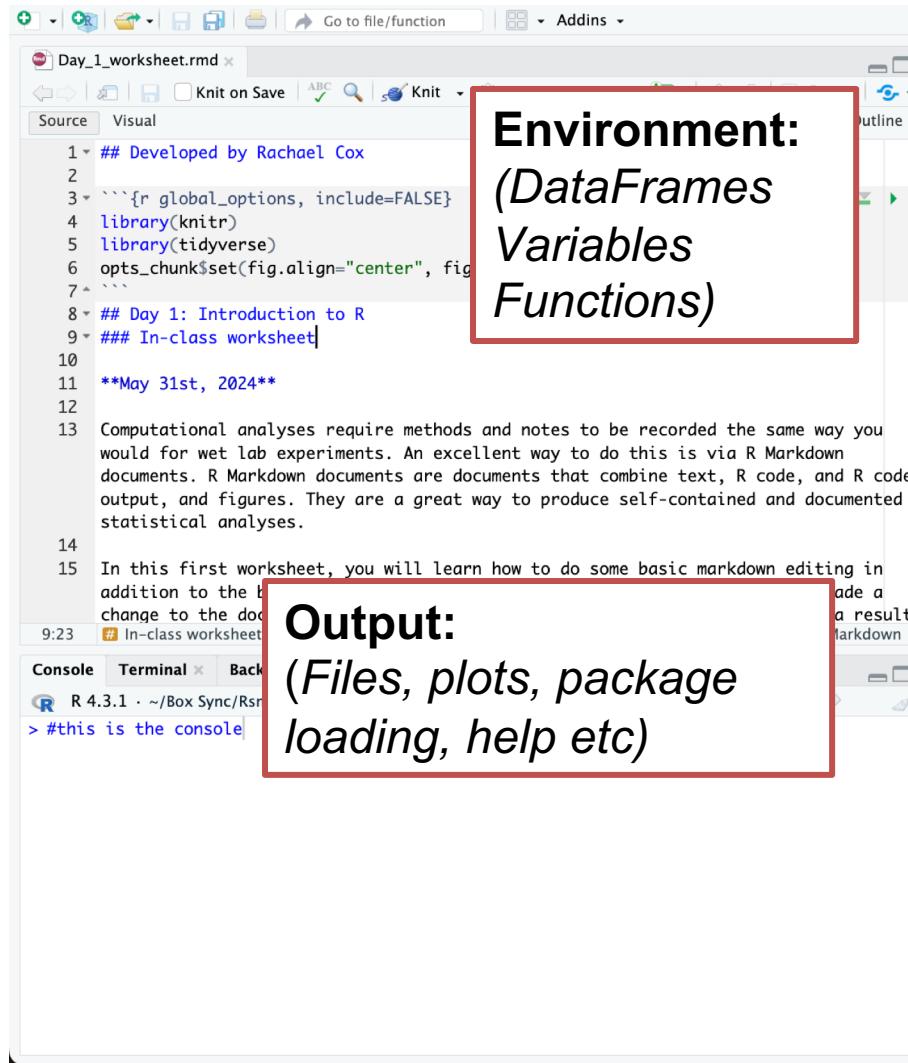
Console Terminal Background Jobs

R 4.3.1 · ~/Box Sync/RsmA biophysical model manuscript and documents/Other Pathogen Modeling/

```
> #this is the console
```

**Console**  
*Output and history  
of executed code*

# R Markdown: Relevant Sections



```
Developed by Rachael Cox
```{r global_options, include=FALSE}
library(knitr)
library(tidyverse)
opts_chunk$set(fig.align="center", fig.width=5, fig.height=5)
```
Day 1: Introduction to R
In-class worksheet

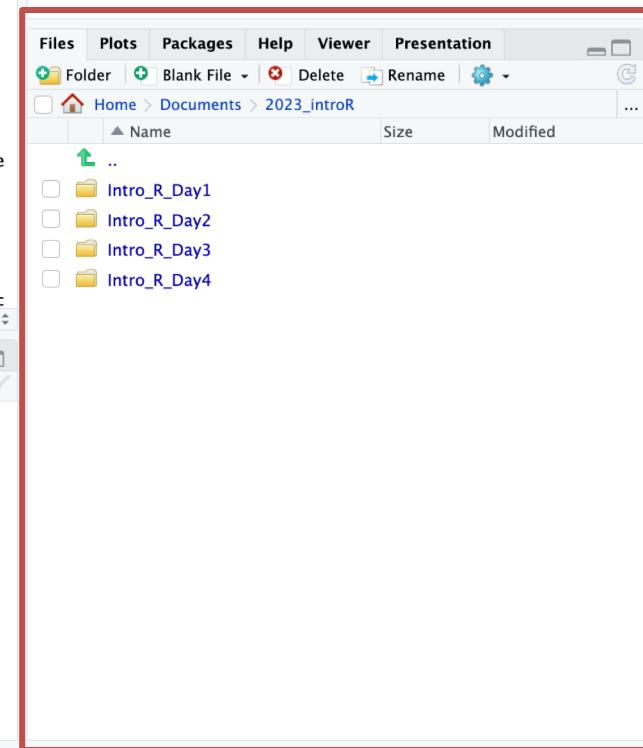
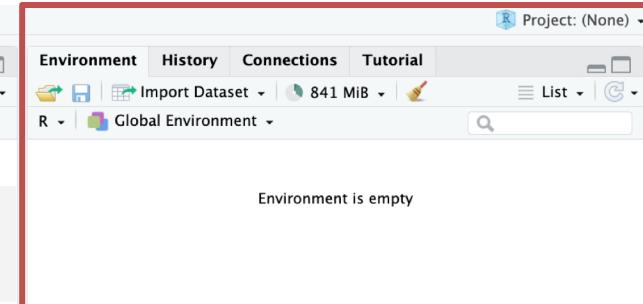
May 31st, 2024

Computational analyses require methods and notes to be recorded the same way you would for wet lab experiments. An excellent way to do this is via R Markdown documents. R Markdown documents are documents that combine text, R code, and R code output, and figures. They are a great way to produce self-contained and documented statistical analyses.

In this first worksheet, you will learn how to do some basic markdown editing in addition to the basic R code editing. You can change to the document tab to see how the changes appear in the document.

Environment:
(DataFrames
Variables
Functions)
```

Output:  
(Files, plots, package loading, help etc)



# R Markdown: Writing documents with embedded R code

The screenshot shows the RStudio interface with the file 'class1.Rmd' open. The code editor displays the following R Markdown content:

```
16 -----
17 ## 1. Basic Markdown editing
18 Try out basic R Markdown features, as described
[here.](http://rmarkdown.rstudio.com/authoring_basics.html) Write some text
that is bold, and some that is in italics. Make a numbered list and a bulleted
list. Make a nested list. Try the block-quote feature.
19
20 ## 2. Embedding R code
21
22 R code embedded in R chunks will be executed and the output will be shown.
23 ```{r}
24 # R code goes here
25 x <- 5
26 y <- 7
27 z <- x * y
28 z
29 ...
30
```

# R Markdown: Writing documents with embedded R code

## 1. Basic Markdown editing

Try out basic R Markdown features, as described [here](#). Write some text that is bold, and some that is *italics*. Make a numbered list and a bulleted list. Make a nested list. Try the block-quote feature.

## 2. Embedding R code

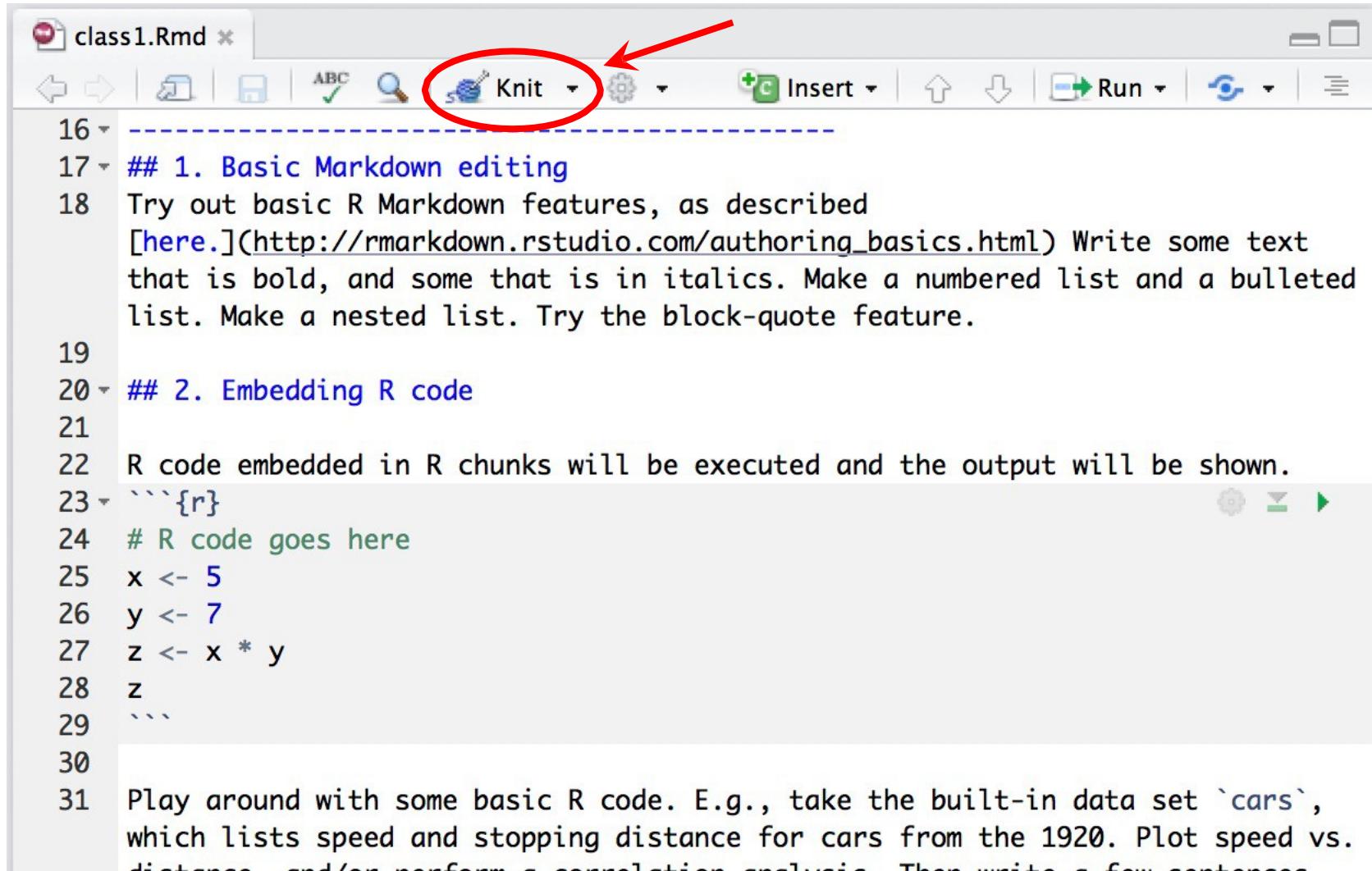
R code embedded in R chunks will be executed and the output will be shown.

```
R code goes here
x <- 5
y <- 7
z <- x * y
z
```

```
[1] 35
```

Play around with some basic R code. E.g., take the built-in data set `cars`, which lists speed and stopping distance for cars from the 1920. Plot speed vs. distance, and/or perform a correlation analysis. Then write a few sentences describing what you see.

# We convert R Markdown to HTML by “knitting” the Markdown file



A screenshot of the RStudio interface showing an R Markdown file named "class1.Rmd". The file content includes sections on basic Markdown editing, embedding R code, and playing around with R code. The RStudio toolbar at the top has several icons, and the "Knit" icon (represented by a blue gear and a white document) is circled in red with a large arrow pointing to it from the left.

```
16 -----
17 ## 1. Basic Markdown editing
18 Try out basic R Markdown features, as described
[here.](http://rmarkdown.rstudio.com/authoring_basics.html) Write some text
that is bold, and some that is in italics. Make a numbered list and a bulleted
list. Make a nested list. Try the block-quote feature.
19
20 ## 2. Embedding R code
21
22 R code embedded in R chunks will be executed and the output will be shown.
23 `r`
24 # R code goes here
25 x <- 5
26 y <- 7
27 z <- x * y
28 z
29 `r`
30
31 Play around with some basic R code. E.g., take the built-in data set `cars`,
which lists speed and stopping distance for cars from the 1920. Plot speed vs.
distance and/or perform a correlation analysis. Then write a few sentences
```

# Markdown basics

[http://rmarkdown.rstudio.com/authoring\\_basics.html](http://rmarkdown.rstudio.com/authoring_basics.html)

<https://www.rstudio.com/wp-content/uploads/2015/02/rmarkdown-cheatsheet.pdf>

normal text

\*italics\*

\*\*bold\*\*

```
Header 1
Header 2
```

List:

1. Item 1
2. Item 2
3. Item 3

normal text

*italics*

**bold**

## Header 1

## Header 2

List:

1. Item 1
2. Item 2
3. Item 3



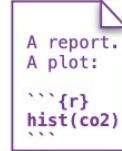
i. **Open** - Open a file that uses the .Rmd extension.



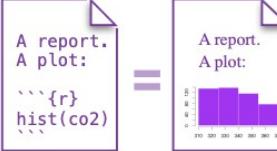
ii. **Write** - Write content with the easy to use R Markdown syntax



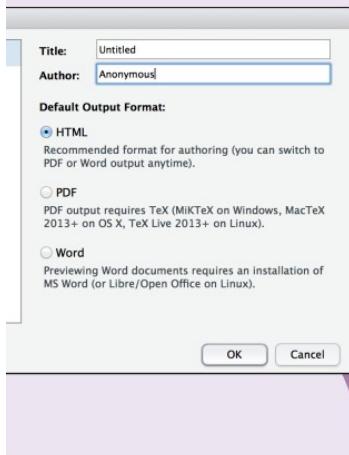
iii. **Embed** - Embed R code that creates output to include in the report



iv. **Render** - Replace R code with its output and transform the report into a slideshow, pdf, html or ms Word file.



extension .Rmd, or open



explains what type of

The RStudio template writes the YAML header for you

om your .Rmd file (in Step 6)



web page)

ent

### 3. Markdown

Next, write your report in plain text. Use markdown syntax to describe how to format text in the final report.

#### syntax

Plain text

End a line with two spaces to start a new paragraph.

\*italics\* and \_italics\_  
\*\*bold\*\* and \_\_bold\_\_  
superscript^2^  
~~strikethrough~~  
[link] (www.rstudio.com)

# Header 1

## Header 2

### Header 3

#### Header 4

##### Header 5

###### Header 6

endash: --

emdash: ---

ellipsis: ...

inline equation: \$A = \pi r^2\$  
image: 

horizontal rule (or slide break):

\*\*\*

> block quote

\* unordered list  
\* item 2  
  + sub-item 1  
  + sub-item 2

1. ordered list

#### becomes

Plain text

End a line with two spaces to start a new paragraph.

italics and italics

bold and bold

superscript<sup>2</sup>

strikethrough

link

## Header 1

## Header 2

## Header 3

### Header 4

### Header 5

### Header 6

endash: –

emdash: —

ellipsis: ...

inline equation:  $A = \pi * r^2$



horizontal rule (or slide break):

block quote

- unordered list
- item 2
  - sub-item 1
  - sub-item 2

1. ordered list

# Markdown basics

Embedded R code will be evaluated and printed

```
```{r}
head(cars)
plot(cars$speed, cars$dist)
```
```

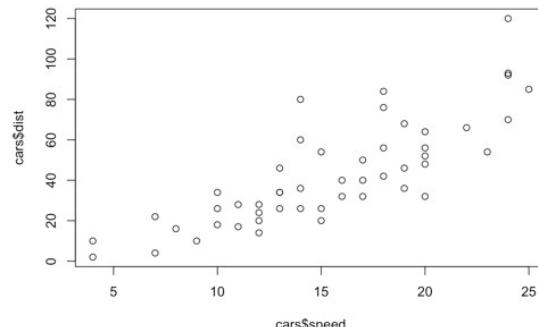


Embedded R code will be evaluated and printed

```
head(cars)
```

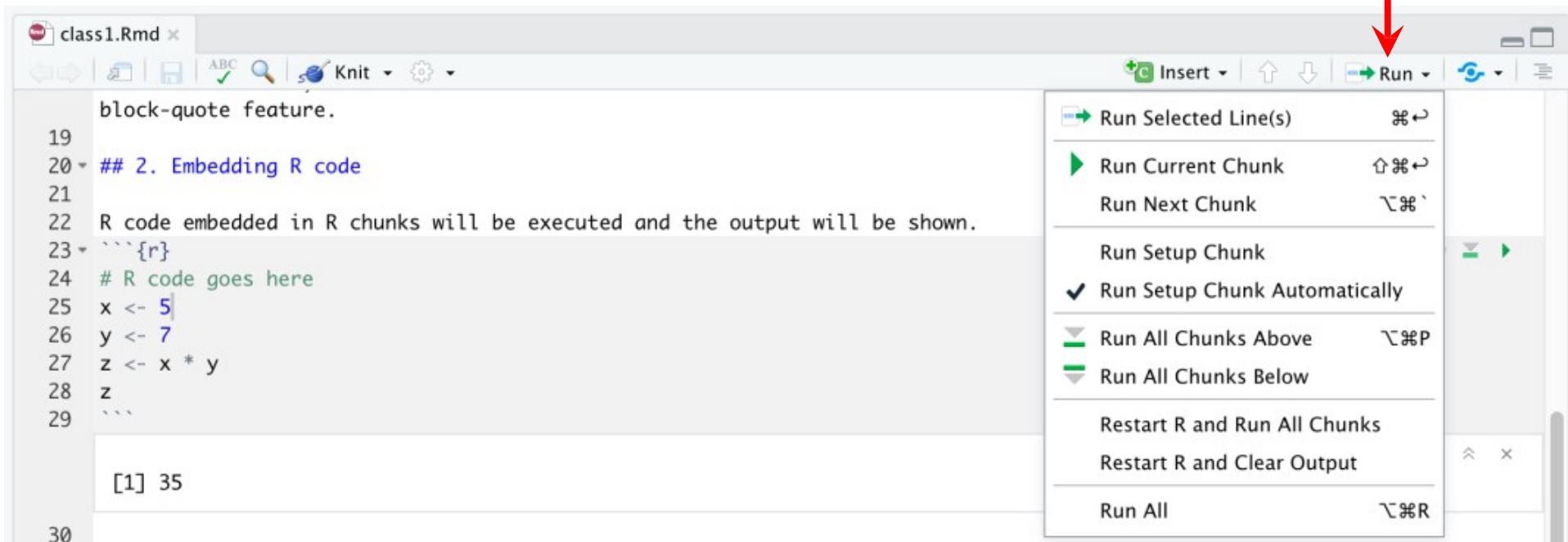
```
speed dist
1 4 2
2 4 10
3 7 4
4 7 22
5 8 16
6 9 10
```

```
plot(cars$speed, cars$dist)
```



# Different ways to execute code in RStudio

# Option 1: Press the “Run” button



## Option 2: Highlight code you want to execute and press ctrl+Enter (cmd+Enter on Macs)

R code embedded in R chunks will be executed and the output will be shown.

```
```{r}
# R code goes here
x <- 5
y <- 7
z <- x * y
z
...``
```



Console

Terminal ×

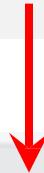
Jobs ×

~/Desktop/projects/ ↗

```
> x <- 5
> y <- 7
> z <- x * y
> z
[1] 35
> |
```

Option 3: Place pointer on line of code you want to execute, press ctrl+Enter (cmd+Enter on Macs)

```
R code embedded in R chunks will be executed and the output will be shown.  
```{r}  
R code goes here
x <- 5
y <- 7
z <- x * y
z
```
```



The screenshot shows an R console interface. At the top, there are tabs for "Console", "Terminal ×", and "Jobs ×". Below the tabs, the current working directory is indicated as "~/Desktop/projects/". In the main console area, the command "z <- x * y" is entered, followed by a new line indicator. A red arrow from the previous image points to this new line indicator, indicating where the user should place the cursor to execute the line of code.

```
Console Terminal × Jobs ×  
~/Desktop/projects/ ↵  
> z <- x * y  
> |
```

Use **ctrl+Shift+Enter** (**cmd+Shift+Enter** on Macs) to execute an entire code chunk

```
R code embedded in R chunks will be executed and the output will be shown.  
```{r}  
R code goes here
x <- 5
y <- 7
z <- x * y
z
```
```



```
Console Terminal × Jobs ×  
~/Desktop/projects/ ↵  
> x <- 5  
> y <- 7  
> z <- x * y  
> z  
[1] 35  
> |
```

Shortcuts for coding

- **Ctrl+Shift+C** (Cmd+Shift+C on Macs) will comment/uncomment a line or multiple lines
- **Tab** and **Shift+Tab** will indent and un-indent lines, respectively
- **Ctrl+Shift+M** (Cmd+Shift+M on Macs) produces a pipe operator `%>%` (will be used within tidyverse)

Your turn!

Try Example #1 and #2

Stuck? Raise your hand and I will come around

R Programming Basics

Variable assignments and objects

```
> x <- 5
```

Assign **number** 5 to **variable** x

```
> x
```

```
[1] 5
```

```
> 5*x^2+7
```

Calculate $5 \cdot x^2 + 7$

```
[1] 132
```

```
> y <- c(1, 2, 3, 4, 5)
```

Create object (**a vector**),
assign to **variable** y

```
> y
```

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

```
> x*y
```

Multiply each element
in **vector** y with the **number** in x

```
[1] 5 10 15 20 25
```

Strings

A **string** contains text:

```
> name <- "Alex L"  
> name  
[1] "Alex L"
```

A **vector of strings**:

```
> animals <- c("cat", "mouse", "mouse",  
"cat", "rabbit")  
> animals  
[1] "cat"      "mouse"    "mouse"    "cat"  
"rabbit"
```

Factors

Factors keep track of distinct categories (levels) in a vector:

```
> animals  
[1] "cat"      "mouse"    "mouse"    "cat"  
"rabbit"  
  
> factor(animals)  
[1] cat       mouse     mouse     cat       rabbit  
Levels: cat mouse rabbit
```

Data frames

We use **data frames** to store data sets with multiple variables:

```
> pets <- data.frame(  
  family = c(1, 2, 3, 4, 5),  
  pet = animals  
)  
  
> pets  
family      pet  
1       1     cat  
2       2   mouse  
3       3   mouse  
4       4     cat  
5       5 rabbit
```

Data frames

We access individual columns in a data frame with \$ + the column name:

```
> pets$family  
[1] 1 2 3 4 5
```

```
> pets$pet  
[1] cat      mouse    mouse    cat      rabbit  
Levels: cat mouse rabbit
```

Demonstration Time!

Work on Section #3

Data frames

R has many built-in data frames:

```
> cars
```

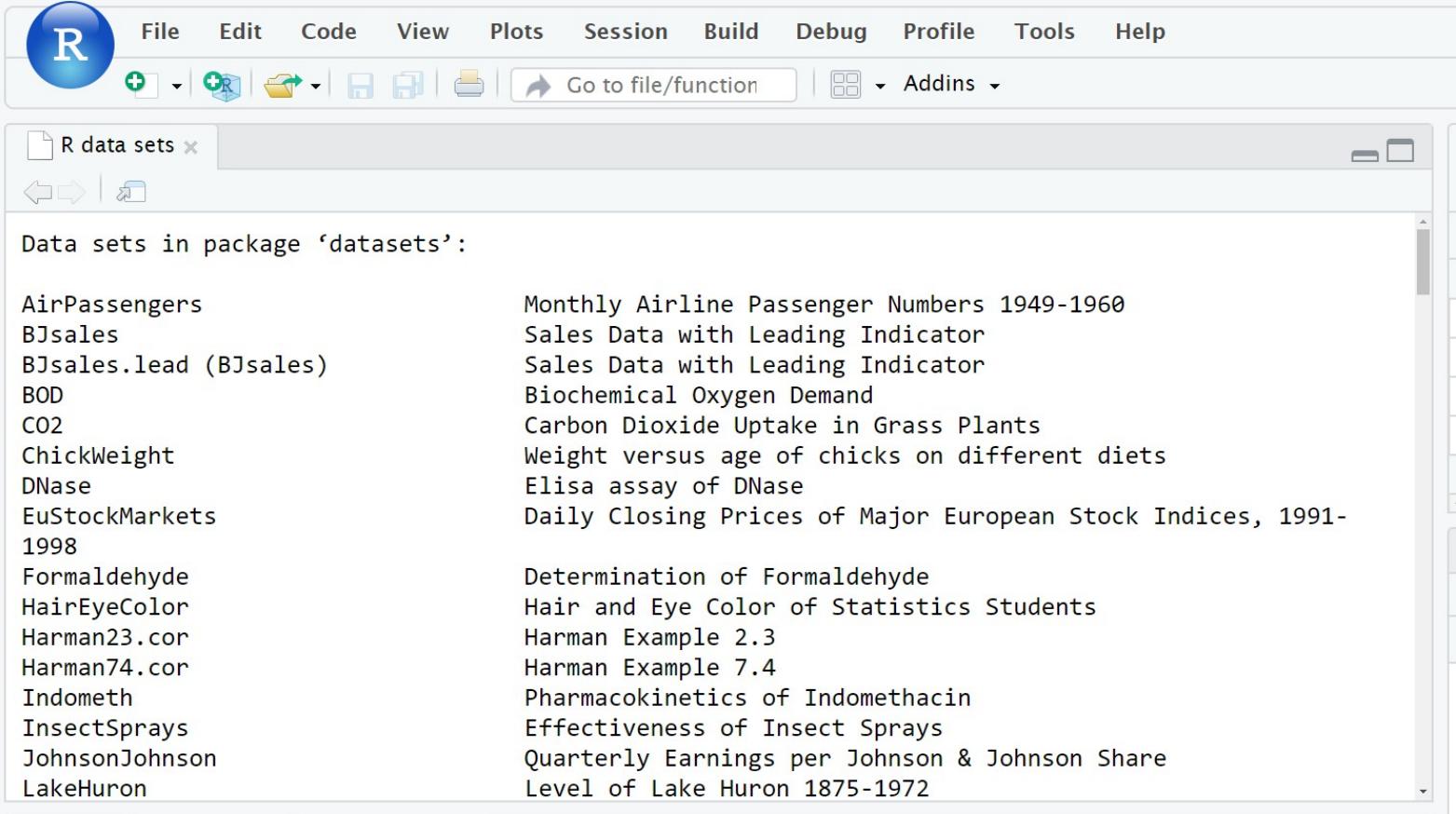
| | speed | dist |
|---|-------|------|
| 1 | 4 | 2 |
| 2 | 4 | 10 |
| 3 | 7 | 4 |
| 4 | 7 | 22 |
| 5 | 8 | 16 |
| 6 | 9 | 10 |
| 7 | 10 | 18 |
| 8 | 10 | 26 |
| 9 | 10 | 34 |

```
> chickwts
```

| | weight | feed |
|-----|--------|-----------|
| 1 | 179 | horsebean |
| 2 | 160 | horsebean |
| 3 | 136 | horsebean |
| 4 | 227 | horsebean |
| ... | ... | ... |
| 11 | 309 | linseed |
| 12 | 229 | linseed |
| 13 | 181 | linseed |
| 14 | 141 | linseed |

Data frames

Available built-in datasets can be accessed with `data()`



The screenshot shows the RStudio interface with the 'R data sets' browser panel open. The panel lists various built-in datasets from the 'datasets' package, each with a brief description. The 'Console' tab at the bottom shows the command `> data()` being run.

| Data set | Description |
|------------------------|---|
| AirPassengers | Monthly Airline Passenger Numbers 1949-1960 |
| BJSales | Sales Data with Leading Indicator |
| BJSales.lead (BJSales) | Sales Data with Leading Indicator |
| BOD | Biochemical Oxygen Demand |
| CO2 | Carbon Dioxide Uptake in Grass Plants |
| ChickWeight | Weight versus age of chicks on different diets |
| DNase | Elisa assay of DNase |
| EuStockMarkets | Daily Closing Prices of Major European Stock Indices, 1991- |
| 1998 | |
| Formaldehyde | Determination of Formaldehyde |
| HairEyeColor | Hair and Eye Color of Statistics Students |
| Harman23.cor | Harman Example 2.3 |
| Harman74.cor | Harman Example 7.4 |
| Indometh | Pharmacokinetics of Indomethacin |
| InsectSprays | Effectiveness of Insect Sprays |
| JohnsonJohnson | Quarterly Earnings per Johnson & Johnson Share |
| LakeHuron | Level of Lake Huron 1875-1972 |

```
Console Terminal × R Markdown ×
```

```
> data()
```

```
>
```

Data frames

Data set information can be accessed with `?dataset`

The screenshot shows the RStudio interface. On the left, the Environment pane lists various datasets: Quarterly Time Series of the Number of Australian Residents, beaver1 (beavers), Body Temperature Series of Two Beavers, beaver2 (beavers), Body Temperature Series of Two Beavers, cars, chickwts, co2, Mauna Loa Atmospheric CO2 Concentration, and crimtab. The 'cars' dataset is highlighted with a red circle and has a red arrow pointing from it towards the Help pane. In the Help pane, the 'Packages' tab is selected (also circled in red). The search bar shows 'R: Speed and Stopping Distances of Cars'. The main content area displays the documentation for the 'cars' dataset, which includes:

Speed and Stopping Distances of Cars

Description

The data give the speed of cars and the distances taken to stop. Note that the data were recorded in the 1920s.

Usage

`cars`

Format

A data frame with 50 observations on 2 variables.

[,1] speed numeric Speed (mph)
[,2] dist numeric Stopping distance (ft)

Source

Ezekiel, M. (1930) *Methods of Correlation Analysis*. Wiley.

```
> data()
> ?data
> ?cars
> |
```

Data frames

The `head()` function shows the first few lines of a data frame:

```
> head(cars)
  speed dist
1     4     2
2     4    10
3     7     4
4     7    22
5     8    16
6     9    10
>
```

Functions

Functions are called in the format `function(argument)`

```
> head(cars)
```

Function name

First argument



Functions

Functions can have any number of required arguments or optional arguments

```
> head(cars, 8)
```

Function name First argument
(required) Second argument
(optional; default = 6)

Functions

`head(cars, 8)` will show the first 8 lines of the data frame instead of the default 6:

```
> head(cars, 8)
```

| | speed | dist |
|---|-------|------|
| 1 | 4 | 2 |
| 2 | 4 | 10 |
| 3 | 7 | 4 |
| 4 | 7 | 22 |
| 5 | 8 | 16 |
| 6 | 9 | 10 |
| 7 | 10 | 18 |
| 8 | 10 | 26 |

Functions

More information about what a function does and/or requires can be accessed with `?function`

The screenshot shows the RStudio interface with the following details:

- Environment Tab:** Shows a data frame with columns `speed` and `dist`. The data is as follows:

| | speed | dist |
|---|-------|------|
| 1 | 4 | 2 |
| 2 | 4 | 10 |
| 3 | 7 | 4 |
| 4 | 7 | 22 |
| 5 | 8 | 16 |
| 6 | 9 | 10 |
| 7 | 10 | 18 |
| 8 | 10 | 23 |

- Help Tab:** The `head` function is selected. The title is **Return the First or Last Part of an Object**.
- Description:** Returns the first or last parts of a vector, matrix, table, data frame or function. Since `head()` and `tail()` are generic functions.
- Usage:** The code for the `head` function is displayed.

```
head(x, ...)  
## Default S3 method:  
head(x, n = 6L, ...)  
## S3 method for class 'data.frame'  
head(x, n = 6L, ...)  
## S3 method for class 'matrix'  
head(x, n = 6L, ...)  
## S3 method for class 'ftable'  
head(x, n = 6L, ...)  
## S3 method for class 'table'  
head(x, n = 6L, ...)  
## S3 method for class 'function'  
head(x, n = 6L, ...)  
  
tail(x, ...)  
## Default S3 method:  
tail(x, n = 6L, ...)
```

- Console:** The command `> ?head` is highlighted with a red circle and has a red arrow pointing to the help title.

Functions

?function has argument information

The screenshot shows the RStudio interface with the following details:

- Environment Tab:** Shows a data frame with columns "speed" and "dist". The first seven rows of the "speed" column are highlighted.
- Help Tab:** The title is "R: Return the First or Last Part of an Object".
- Arguments Section:** A red circle highlights the "Arguments" section.
- Console Tab:** The command "> ?head" is circled in red.
- Help Text:**
 - x**: an object
 - n**: a single integer. If positive, size for the resulting object: number of elements for a vector (including lists), rows for a matrix or data frame or lines for a function. If negative, all but the n last/first number of elements of x.
 - addrownums**: if there are no row names, create them from the row numbers.
 - ...**: arguments to be passed to or from other methods.

Functions

We can implicitly or explicitly pass arguments

```
> head(cars, 8)
```

| | speed | dist |
|--|-------|------|
|--|-------|------|

| | | |
|---|---|---|
| 1 | 4 | 2 |
|---|---|---|

| | | |
|---|---|----|
| 2 | 4 | 10 |
|---|---|----|

| | | |
|---|---|---|
| 3 | 7 | 4 |
|---|---|---|

| | | |
|---|---|----|
| 4 | 7 | 22 |
|---|---|----|

| | | |
|---|---|----|
| 5 | 8 | 16 |
|---|---|----|

| | | |
|---|---|----|
| 6 | 9 | 10 |
|---|---|----|

| | | |
|---|----|----|
| 7 | 10 | 18 |
|---|----|----|

| | | |
|---|----|----|
| 8 | 10 | 26 |
|---|----|----|

```
>
```

```
> head(x=cars, n=8)
```

| | speed | dist |
|--|-------|------|
|--|-------|------|

| | | |
|---|---|---|
| 1 | 4 | 2 |
|---|---|---|

| | | |
|---|---|----|
| 2 | 4 | 10 |
|---|---|----|

| | | |
|---|---|---|
| 3 | 7 | 4 |
|---|---|---|

| | | |
|---|---|----|
| 4 | 7 | 22 |
|---|---|----|

| | | |
|---|---|----|
| 5 | 8 | 16 |
|---|---|----|

| | | |
|---|---|----|
| 6 | 9 | 10 |
|---|---|----|

| | | |
|---|----|----|
| 7 | 10 | 18 |
|---|----|----|

| | | |
|---|----|----|
| 8 | 10 | 26 |
|---|----|----|

```
>
```

Reading and writing data

`write_csv(variable, "filename.csv")` is a package-specific function that allows you write R data frames to a comma-separated table

`read_csv("filename.csv")` is a package-specific function that allows you to import a file in your local environment:

```
> read_csv("mushrooms_tiny.csv")
```

*note: if you are running the workbook locally and have not installed tidyverse use:
`write.csv()` and `read.csv()` with the same arguments

Demonstration Time!

Work on Section #4

Troubleshooting

Ask RStudio for help

Type `?function` into console

The screenshot shows the RStudio interface with the 'Console' tab selected. In the console, the command `> ?t.test` is typed. A red arrow points from the console input to the search results in the 'Viewer' pane.

The 'Viewer' pane displays the help documentation for the `t.test` function. The title is "Student's t-Test". The "Description" section states: "Performs one and two sample t-tests on vectors of data." The "Usage" section shows the function call `t.test(x, ...)`. Below it, the default S3 method is defined with parameters: `x`, `y = NULL`, `alternative = c("two.sided", "less", "greater")`, `mu = 0`, `paired = FALSE`, `var.equal = FALSE`, and `conf.level = 0.95`. At the bottom, the S3 method for class 'formula' is shown as `t.test(formula, data, subset, na.action, ...)`.

```
Console Terminal × Jobs ×
~/Desktop/projects/
> ?t.test

R: Student's t-Test ✓ Find in Topic
t.test {stats} R Documentation

Student's t-Test

Description
Performs one and two sample t-tests on vectors of data.

Usage
t.test(x, ...)

## Default S3 method:
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)

## S3 method for class 'formula'
t.test(formula, data, subset, na.action, ...)
```

Ask Google for help

how do i run a t test in r

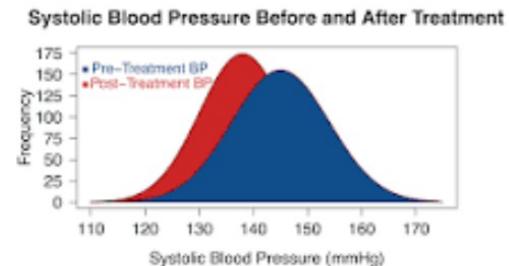
X |

All Videos Images News Shopping More Settings Tools

About 2,780,000,000 results (0.64 seconds)

How to Perform T-tests in R. To conduct a one-sample t-test in R, we use the syntax `t.test(y, mu = 0)` where x is the name of our variable of interest and mu is set equal to the mean specified by the null hypothesis.

Aug 17, 2015



[datascienceplus.com › t-tests](http://datascienceplus.com/t-tests/)

[How to Perform T-tests in R | DataScience+](https://www.datascienceplus.com/how-to-perform-t-tests-in-r/)

G Error in `t.test.default(x, y)` : not enough 'x' observations

Q Error in `t.test.default(x, y)` : not enough 'x' observations - Google Search

Ask StackOverflow for help

stack**overflow** Products  Log in Sign up

Home PUBLIC  Stack Overflow Tags Users Jobs TEAMS What's this?

Rotating and spacing axis labels in ggplot2

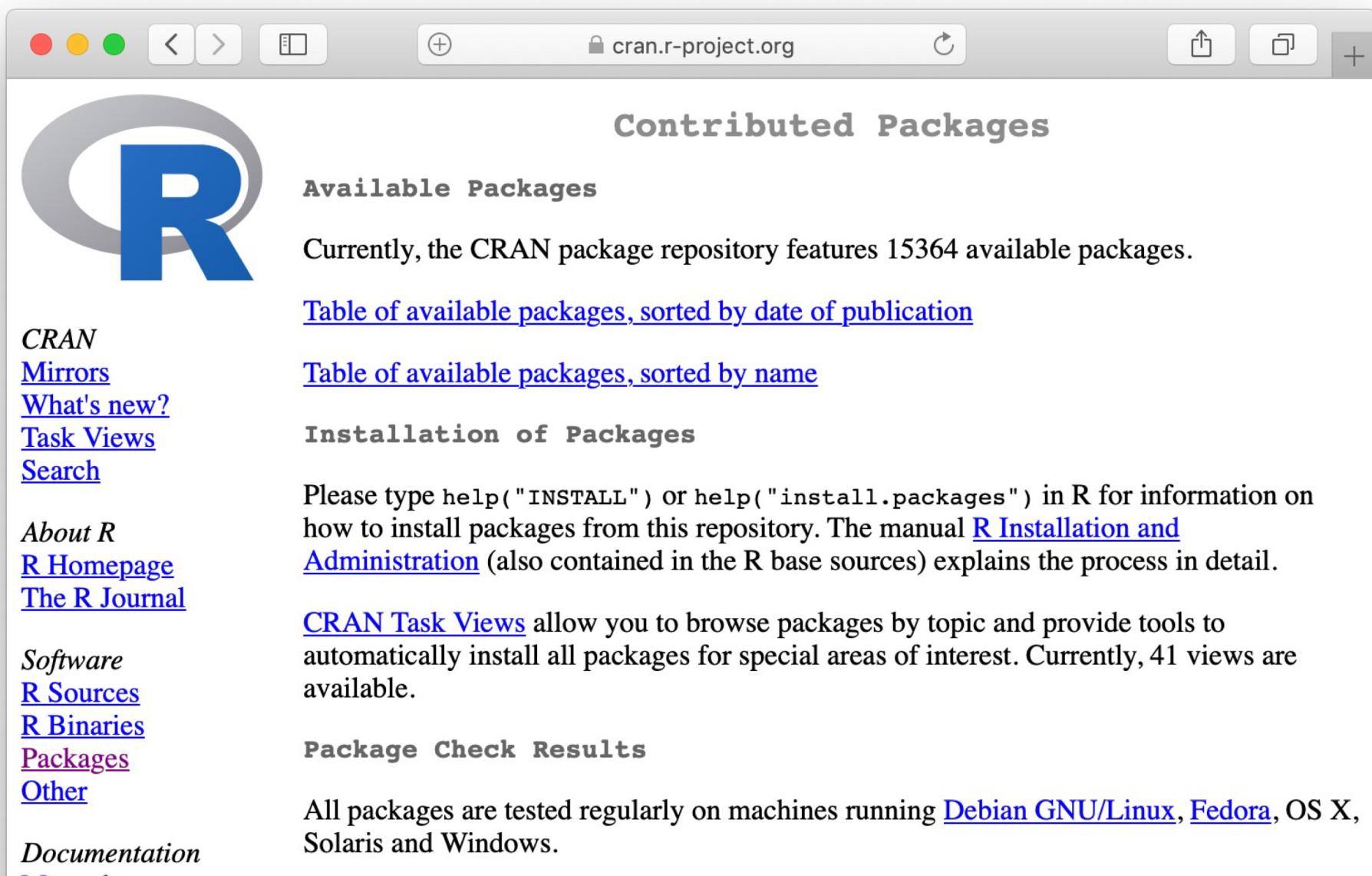
Asked 10 years, 10 months ago Active 18 days ago Viewed 768k times

 680 

I have a plot where the x-axis is a factor whose labels are long. While probably not an ideal visualization, for now I'd like to simply rotate these labels to be vertical. I've figured this part out with the code below, but as you can see, the labels aren't totally visible.

Extending R through packages:
There's a package for everything

R packages are available on CRAN (Comprehensive R Archive Network)



The screenshot shows a web browser window with the URL `cran.r-project.org` in the address bar. The page content is as follows:

Contributed Packages

Available Packages

Currently, the CRAN package repository features 15364 available packages.

[Table of available packages, sorted by date of publication](#)

[Table of available packages, sorted by name](#)

Installation of Packages

Please type `help("INSTALL")` or `help("install.packages")` in R for information on how to install packages from this repository. The manual [R Installation and Administration](#) (also contained in the R base sources) explains the process in detail.

CRAN Task Views allow you to browse packages by topic and provide tools to automatically install all packages for special areas of interest. Currently, 41 views are available.

Package Check Results

All packages are tested regularly on machines running [Debian GNU/Linux](#), [Fedora](#), OS X, Solaris and Windows.

CRAN
[Mirrors](#)
[What's new?](#)
[Task Views](#)
[Search](#)

About R
[R Homepage](#)
[The R Journal](#)

Software
[R Sources](#)
[R Binaries](#)
[Packages](#)
[Other](#)

Documentation

Bio-specific R packages are available on Bioconductor



Search:

[Home](#) [Install](#) [Help](#) [Developers](#) [About](#)

About Bioconductor

Bioconductor provides tools for the analysis and comprehension of high-throughput genomic data.

Bioconductor uses the R statistical programming language, and is open source and open development. It has two releases each year, and an active user community. *Bioconductor* is also available as an [AMI](#) (Amazon Machine Image) and [Docker](#) images.

News

- See our [google calendar](#) for events, conferences, meetings, forums, etc. Add your event with email to events at [bioconductor.org](#).
- *Bioconductor 3.11* is available.
- Nominate an outstanding community member for a *Bioconductor Award*! See the [support site](#) for more information.
- Registration open for [BioC2020](#).
- Core team **job opportunities** available, contact Martin.Morgan at RoswellPark.org
- *Bioconductor F1000 Research Channel* is

BioC 2020

Get the latest updates on the [BioC 2020 Conference](#)!

- BioC 2020 is going virtual July 27 - July 31. Please see the [Registration Page](#) for more information.
- Nominate an outstanding *Bioconductor* community member for a *Bioconductor Award*! See [posting](#) for more information.
- Call for birds-of-feather, hack-a-thon, and how-to sections. Please see [posting](#) for more information.
- Registration is now open. [Register today](#).

Install »

- Discover [1903 software packages](#) available in *Bioconductor* release 3.11.

Get started with *Bioconductor*

- [Install *Bioconductor*](#)
- [Get support](#)
- [Latest newsletter](#)
- [Follow us on twitter](#)
- [Install R](#)

Learn »

Master *Bioconductor* tools

- [Courses](#)
- [Support site](#)
- [Package vignettes](#)
- [Literature citations](#)
- [Common work flows](#)
- [FAQ](#)
- [Community resources](#)
- [Videos](#)