

Getting Help in R/Stats/Computing

Erika Mudrak

Cornell Statistical Consulting Unit

Cornell Statistical Consulting Unit

CSCU assists researchers on the Cornell Campus and elsewhere by providing the following services:

- **Consulting**
- **Contract Services**
- **Workshops**
- **Instructional Materials**


www.cscu.cornell.edu

So you need some help...

- Where to start with some data?
- Teach yourself
 - CRAN Task Views: <https://cran.r-project.org/web/views/>
 - Package Vignettes: `vignette()`

So you need some help...

Your code produced an error?


- Debug it yourself
 - Google error messages!
- Ask colleagues/local resources
 - Lab mates!
 - CSCU!
 - Other Cornell resources like CISER, RDMSG
- Ask strangers
 - Stackexchange 
 - Listservs (R-SIG-xxx) (<https://www.r-project.org/mail.html>)
 - Twitter (?)

Break it down

- Works for any of the avenues above.
- Break down big problem into small problems
- Isolate the error or issue
- Determine cases when the issue occurs, but also maybe cases when things go well.

So you need some help...

Your code produced an error?

- **Debug it yourself**
 - Google error messages!
- Ask colleagues/local resources
 - Lab mates!
 - CSCU!
 - Other Cornell resources like CISER, RDMSG
- Ask strangers
 - Stackexchange
 - Listservs (R-SIG-xxx) ( [stackoverflow](https://www.r-project.org/mail.html) <https://www.r-project.org/mail.html>)
 - Twitter (?)

Debugging

- Messages from R
 - Errors, warnings and messages
 - Could come from the function you called
 - Could come from a function called by a function you called.
 - Could come from a function called by a function called by a function you called.
 - ...

```
f <- function(a) g(a)
g <- function(b) h(b)
h <- function(c) i(c)
i <- function(d) "a" + d
f(10)
```

```
> f(10)
```

```
Error in "a" + d : non-numeric argument to binary operator
```

Debugging

- Messages from R

- If these are functions you wrote:

- take a step backwards in writing process until you solve it
 - or start over with defensive programming

- If these are functions from a package- good luck!

- Check that all objects passed to the function are as you intend
 - Google/forum/stackoverflow
 - traceback()
 - RStudio debug mode.

- <https://support.posit.co/hc/en-us/articles/205612627-Debugging-with-RStudio>
 - <https://adv-r.hadley.nz/debugging.html>

```
f <- function(a) g(a)
g <- function(b) h(b)
h <- function(c) i(c)
i <- function(d) "a" + d
f(10)
```

```
traceback()
#> 5: stop("`d` must be numeric", call. = FALSE) at debugging.R#6
#> 4: i(c) at debugging.R#3
#> 3: h(b) at debugging.R#2
#> 2: g(a) at debugging.R#1
#> 1: f("a")
```


Debugging

- Start with whole block of code, whittle down until you find the smallest piece that creates the error.
- Check intermediate steps as you go

Debugging

- Quick tip: name your variables unique things
 - i.e not “data”, “prob”, “mean”
 - These words may be used in deep functions and cause conflicts
 - Rename “gunnisondata”, “hatchprob”, “meanDON”

Defensive Coding

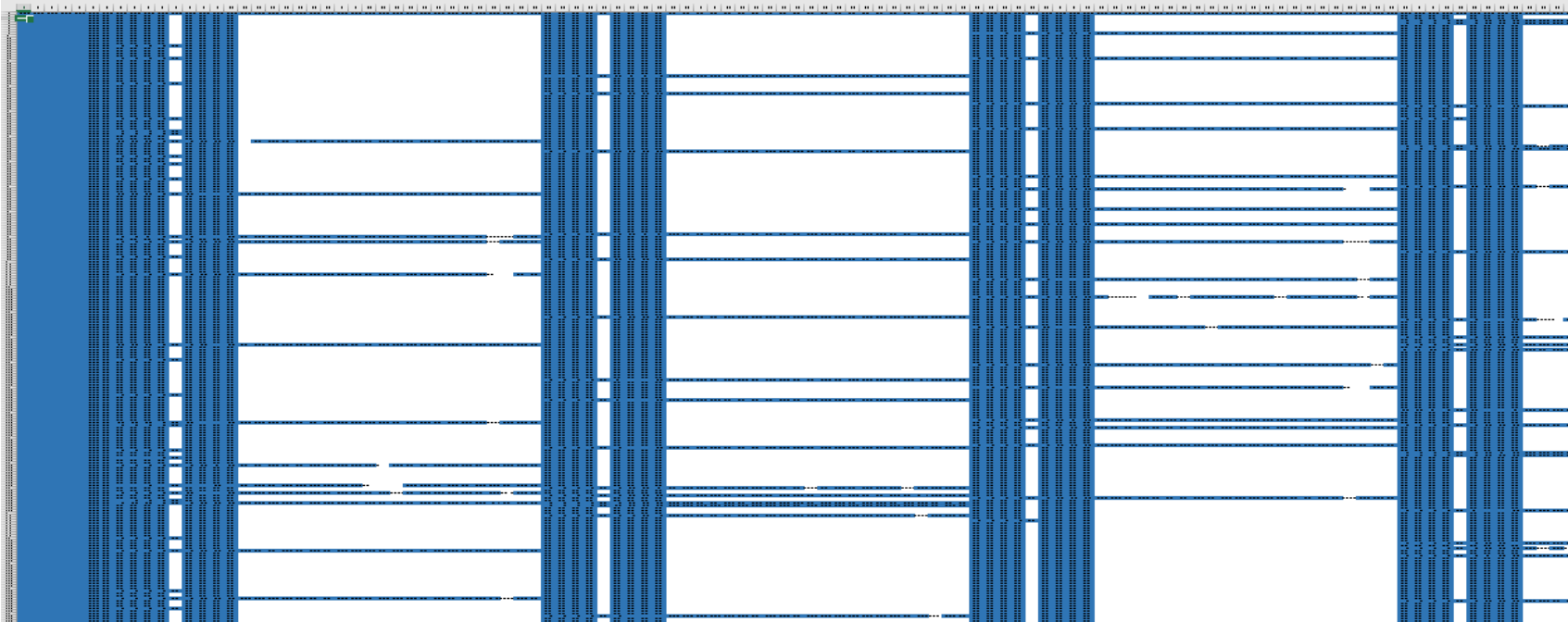
- Avoid difficult debugging situations
- Fail early and often
- Program in small steps, check each step of the way.
- Spot check your intuition each step of the way.

Defensive Coding - Example

- Qualtrics survey
 - 30 possible scenarios
 - Each participant was assigned three scenarios
 - Answer a set of 22 questions for assigned scenarios

Defensive Coding - Example

- Qualtrics survey
 - Answer a set of 22 questions for assigned scenarios




Defensive Coding - Example

```
mturk%>%  
gather(key, value, `Q14.1_First Click`:Q26.26) %>%  
# make tall data out of only columns between Q14_First Click  
# and Q26.26- repeat all the rest  
separate(key, c("organism", "question"), sep="\\.") %>%  
# separate into two columns by period  
mutate(question=factor(question, levels=unique(question))) %>%  
# make question a factor with levels ordered according to original  
# arrangement- this will make the following spread function  
# organize it correctly  
spread(question, value) %>%  
# spread back into wide format, but this time only on question-  
# leave organism on repeating rows  
filter(!is.na(`1_Click Count`)) %>%  
# filter out na values (usign 1_Click count as proxy for the whole set)  
arrange(MTurkCode)  
# order by MTurkCode to group the three responses together for a person
```

So you need some help...

Your code produced an error?

- Debug it yourself
 - Google error messages!
- **Ask colleagues/local resources**
 - Lab mates!
 - CSCU!
 - Other Cornell resources like CISER, RDMSG
- Ask strangers
 - Stack Overflow 
 - Listservs (R-SIG-xxx) (<https://www.r-project.org/mail.html>)
 - Twitter (?)

Sending code to colleagues

- Small questions (“help me debug...”)
- Big questions (“how should I approach this data set?”)

Sending code to colleagues - debug

- Save current version of the data so we don't have to re-process it all
- Supply only needed code

Sending code to colleagues - debug

- Load packages at top of script
 - (report package versions with `sessionInfo()` – copy results to script)
- Save current version of the data so we don't have to re-process it all
 - `write.csv(mydataframe, "mydataframe.csv")`
 - `save(currentdata, file="currentdata.R")`
- Remove everything unrelated to problem
 - Long codes get ignored until the perceived amount of free time
- Use nice formatting, commenting
- Well-named variables (concise and informative)
- Use commenting to indicate where the issue is

Sending code to colleagues

- Small questions (“help me debug...”)
- Big questions (“how should I approach this data set?”)

Sending code to colleagues – big questions

- The raw data.
- A tidy data set and code book
- An explicit and exact recipe you used to go from 1 -> 2,3

<https://github.com/jtleek/datasharing>

Sending code to colleagues – big questions

- The raw data.
 - Ran no software on the data
 - Did not modify any of the data values
 - You did not remove any data from the data set
 - You did not summarize the data in any way

<https://github.com/jtleek/datasharing>

Sending code to colleagues – big questions

- A tidy data set and code book
 - Each variable you measure should be in one column
 - Each different observation of that variable should be in a different row
 - There should be one table for each "kind" or "level" of observation
 - If you have multiple tables, they should include a column in the table that allows them to be joined or merged

<https://github.com/jtleek/datasharing>

Sending code to colleagues – big questions

- A tidy data set and code book
 - Information about the experimental study design or sampling method
 - Information about the variables
 - Data type
 - units
 - Information about the summary choices you made

<https://github.com/jtleek/datasharing>


Sending code to colleagues – big questions

- An explicit and exact recipe you used to go from 1 -> 2,3
 - Pseudocode
 - Readme file showing how all files are related.

<https://github.com/jtleek/datasharing>

So you need some help...

Your code produced an error?

- Debug it yourself
 - Google error messages!
- Ask colleagues/local resources
 - Lab mates!
 - CSCU!
 - Other Cornell resources like CISER, RDMSG
- **Ask strangers**
 - Stackexchange
 - Listservs (R-SIG-xxx) ( [stackoverflow](https://www.r-project.org/mail.html) <https://www.r-project.org/mail.html>)
 - Twitter (?)

Asking strangers

- Minimum reproducible example (“reprex”)
 - a minimal dataset, necessary to reproduce the error
 - the minimal runnable code necessary to reproduce the error, which can be run on the given dataset.
 - the necessary information on the used packages, R version, and system it is run on.
 - in the case of random processes, a seed (set by `set.seed()`) for reproducibility

<https://stackoverflow.com/questions/5963269/how-to-make-a-great-r-reproducible-example>

Asking strangers

- Minimum reproducible example (reprex)
 - Use existing data included in packages
 - `iris`, `mtcars`, `OrchardSprays`, `chickwts`, `npk`
 - Generate fake data that illustrates your question/issue/problem
 - `dput()` to get your data in an easily loadable form
 - No attachments/links to websites...

<https://stackoverflow.com/questions/5963269/how-to-make-a-great-r-reproducible-example>

Asking strangers

- Minimum reproducible example (reprex)
 - Generate fake data that illustrates your question/issue/problem

```
my.df <- data.frame(  
  col1 = sample(c(1,2), 10, replace = TRUE),  
  col2 = as.factor(sample(10)), col3 = letters[1:10],  
  col4 = sample(c(TRUE, FALSE), 10, replace = TRUE))
```

	col1	col2	col3	col4
1	2	10	a	FALSE
2	2	3	b	FALSE
3	2	2	c	TRUE
4	1	7	d	FALSE
5	1	5	e	FALSE
6	2	6	f	FALSE
7	1	1	g	FALSE
8	1	8	h	TRUE
9	2	4	i	TRUE
10	2	9	j	TRUE

<https://stackoverflow.com/questions/5963269/how-to-make-a-great-r-reproducible-example>

Asking strangers

- Minimum reproducible example (reprex)
 - `dput()` to get your data in an easily loadable form

`dput(chickwts)`

```
structure(list(weight = c(179, 160, 136, 227, 217, 168, 108, 124, 143, 140, 309,
229, 181, 141, 260, 203, 148, 169, 213, 257, 244, 271, 243, 230, 248, 327, 329, 250,
193, 271, 316, 267, 199, 171, 158, 248, 423, 340, 392, 339, 341, 226, 320, 295, 334,
322, 297, 318, 325, 257, 303, 315, 380, 153, 263, 242, 206, 344, 258, 368, 390, 379,
260, 404, 318, 352, 359, 216, 222, 283, 332), feed = structure(c(2L, 2L, 2L, 2L, 2L,
2L, 2L, 2L, 2L, 2L, 3L, 3L, 3L, 3L, 3L, 3L, 3L, 3L, 3L, 3L, 3L, 5L, 5L, 5L, 5L,
5L, 5L, 5L, 5L, 5L, 5L, 5L, 5L, 5L, 6L, 6L, 6L, 6L, 6L, 6L, 6L, 6L, 6L, 6L,
6L, 4L, 4L, 4L, 4L, 4L, 4L, 4L, 4L, 4L, 4L, 4L, 1L, 1L, 1L, 1L, 1L, 1L, 1L, 1L, 1L,
1L, 1L, 1L), .Label = c("casein", "horsebean", "linseed", "meatmeal", "soybean",
"sunflower"), class = "factor")), class = "data.frame", row.names = c(NA, -71L))
```

Asking strangers


- Minimum reproducible example (reprex)
 - Make sure users have all the info they need to run your reprex
 - reprex package in R (<https://reprex.tidyverse.org/>)
 - Will generate error if your code is not self-contained
 - Creates markdown for use on different venues (Stack Overflow, Github, HTML...)

Asking strangers

- Minimum reproducible example (reprex)
 - Often the act of making the reprex will help you solve the problem!
 - In this case, consider posting both the question and the answer on stack overflow so others can benefit..

So you need some help...

Your code produced an error?

- Debug it yourself
 - Google error messages!
- Ask colleagues/local resources
 - Lab mates!
 - CSCU!
 - Other Cornell resources like CISER, RDMSG
- Ask strangers
 - Stackexchange  stackoverflow
 - Listservs (R-SIG-xxx) (<https://www.r-project.org/mail.html>)
 - Twitter (?)

To receive assistance from CSCU:

Visit cscu.cornell.edu

email: cscu@cornell.edu

- Schedule an appointment
 - Remote appointments available (Zoom, Skype, phone)
- Zoom Drop-in consulting hours:
 - M-F, 11am, 12pm, 1pm, 2pm
 - <https://cscu.cornell.edu/consulting/drop-ins/>