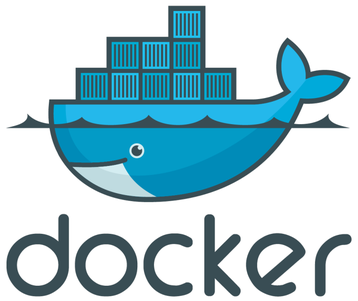
using





for introductory statistics teaching

slides & demo materials at https://github.com/mine-cetinkaya-rundel/useR-2015

mine çetinkaya-rundel

duke university

mine@stat.duke.edu

@minebocek

mine-cetinkaya-rundel

playing nice in the classroom

slides & demo materials at https://github.com/mine-cetinkaya-rundel/useR-2015

mine çetinkaya-rundel
duke university

mine@stat.duke.edu
@minebocek
mine-cetinkaya-rundel

context


fist course in stats for non-majors (sta 101)
not calculus based
mostly social science majors
possibly only quantitative course these students take in ugrad
weekly lab session + in class activities using R

why R?

unlike other software designed specifically for courses at this level

why R?

- free & open source
- powerful & flexible
- relevant beyond intro stat

why not R?

- challenge of teaching programming in addition to stats concepts
- command line more intimidating than GUI

don’t do any hands on data analysis

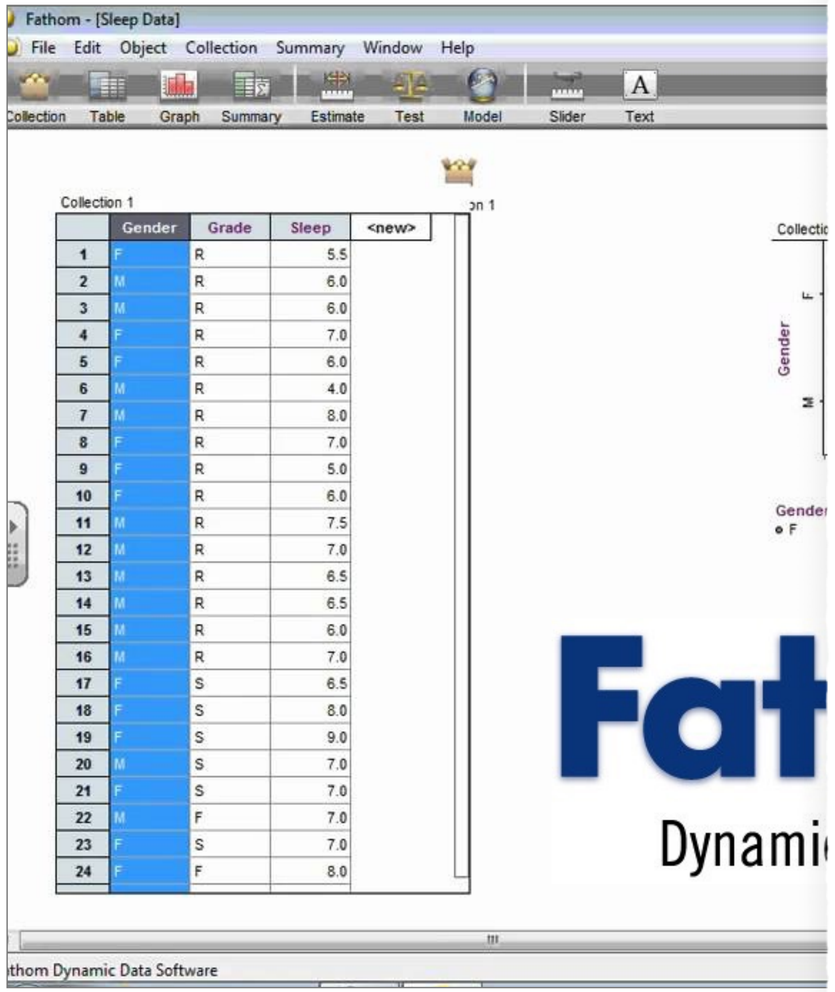
challenge of teaching programming in addition to stats concepts

don’t do any hands on data analysis

disservice to everyone involved

challenge of teaching programming in addition to stats concepts

use a drag-and-drop type tool



III. Adding Proportions to Summary Table

For categorical variables, you should see the counts of each possible outcome of that variable in the **Summary Table**. To see the breakdown of proportions or percentages, follow these steps:

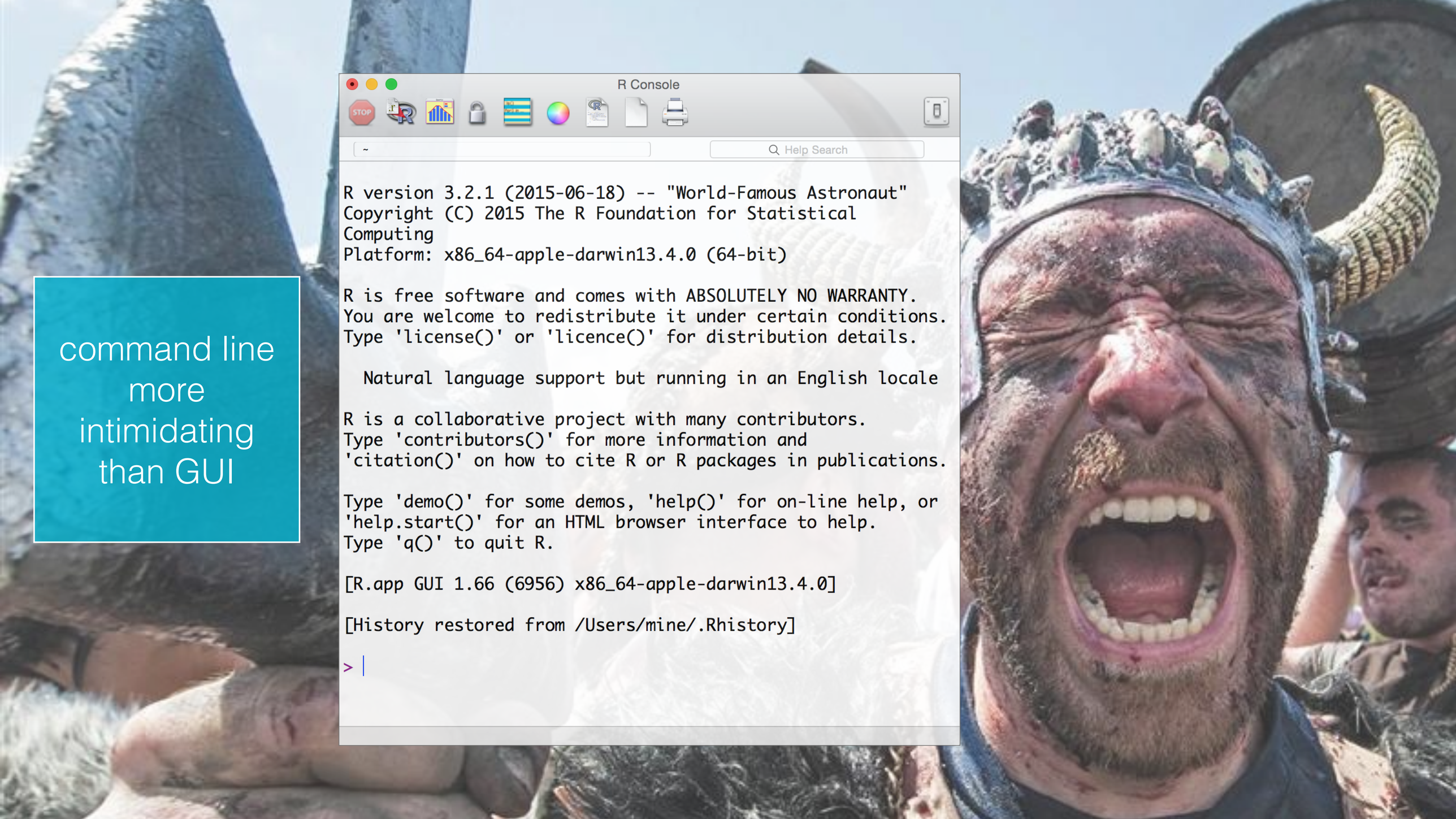
- Click on the **Summary Table** to highlight it, click on the "**Summary**" drop-down menu and select "**Add Formula**". In general, whenever you click and select a *Fathom* object (such as a **Table**, **Graph**, or **Summary**) the menu at the top of the screen will change to give you options for working on that object.

- In the formula editor that pops up, type "*rowproportion*" (without the quotes) to see the row proportions or "*columnproportion*" to see the column proportions. Be sure to spell the names of the formulas correctly or else *Fathom* will give you an error. (If you spell the names correctly, they should change to a purplish color in your editor.)

- You will see that each cell in the **Summary Table** now includes numbers for multiple statistics. To see which numbers correspond with which statistics, simply look at the bottom of your summary table to see the order of the statistics or formulas within each cell.

- To delete (or change) a particular statistic from the table, you can double click on its name at the bottom of the **Summary Table**. In the formula editor, press delete (or make your changes) and then click "**OK**".


don’t do any hands on data analysis
disservice to everyone involved
challenge of teaching programming in addition to stats concepts
use a drag-and-drop type tool
there’s still a learning curve


command line more intimidating than GUI
R Console
~
Help Search
R version 3.2.1 (2015-06-18) -- "World-Famous Astronaut"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin13.4.0 (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.
[R.app GUI 1.66 (6956) x86_64-apple-darwin13.4.0]
[History restored from /Users/mine/.Rhistory]
>

command line more intimidating than GUI

RStudio

example.Rmd

```
---
title: "example"
author: "Mine Cetinkaya-Rundel"
date: "June 27, 2015"
output: html_document
---

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```{r}
summary(cars)
```
```

```
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

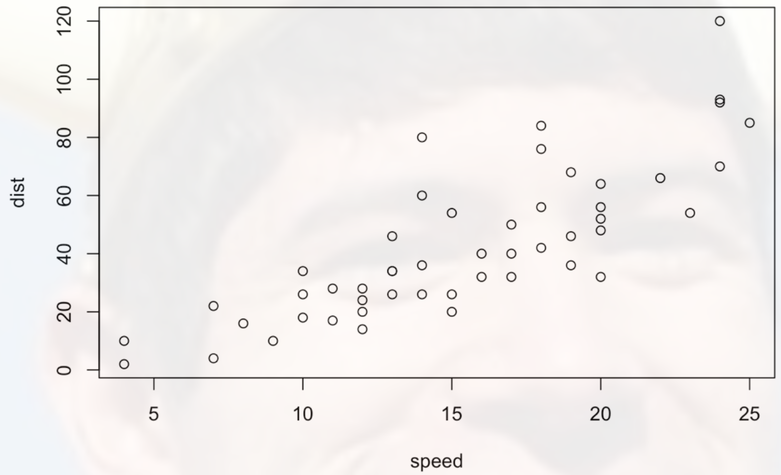
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

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.

>
```

```
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

how R?

technical

pedagogical

getting started:
“like a knife through butter”

avoid local installation

preinstalled & preloaded packages

implementation:
phase 1

external (RStudio) solution

RStudio beta server

keep the experience

Gmail authentication a pain

university login

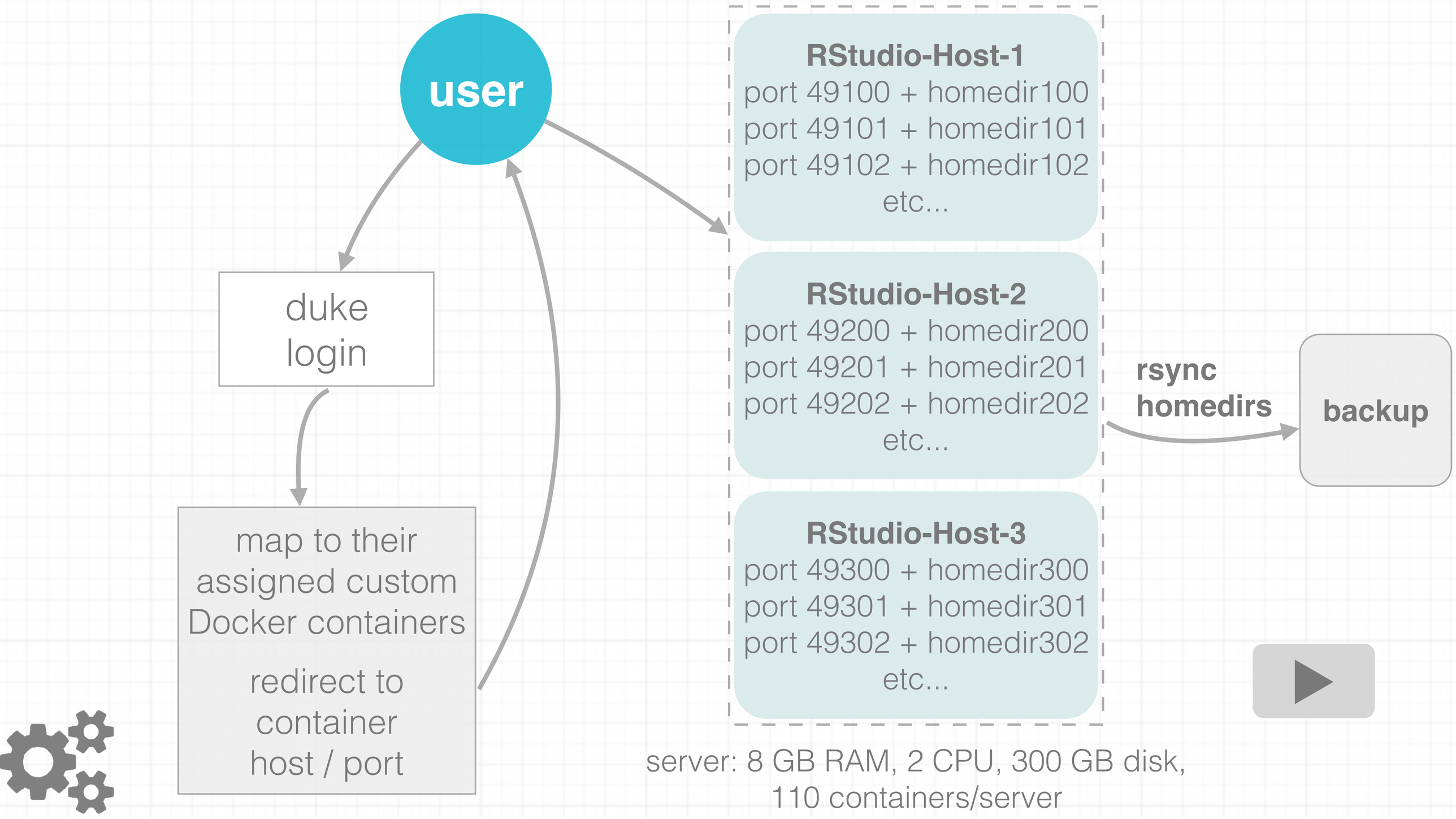
Control over version / packages limited

full control

implementation:
phase 2

in-house solution

option 1: monolithic RStudio server instance	scaling issues	load prediction	security consideration (large # of non-dept students)
option 2: personal VMs	resource intensive	duplication	
option 3: docker 	lighweight (with many virtues of individual VMs)	sandbox individual students	spin up new servers on the fly as needed


user
duke
login
map to their
assigned custom
Docker containers
redirect to
container
host / port
RStudio-Host-1
port 49100 + homedir100
port 49101 + homedir101
port 49102 + homedir102
etc...
RStudio-Host-2
port 49200 + homedir200
port 49201 + homedir201
port 49202 + homedir202
etc...
RStudio-Host-3
port 49300 + homedir300
port 49301 + homedir301
port 49302 + homedir302
etc...
rsync
homedirs
backup
server: 8 GB RAM, 2 CPU, 300 GB disk,
110 containers/server

reproducible: literate programming

train new researchers whose only workflow is a reproducible one

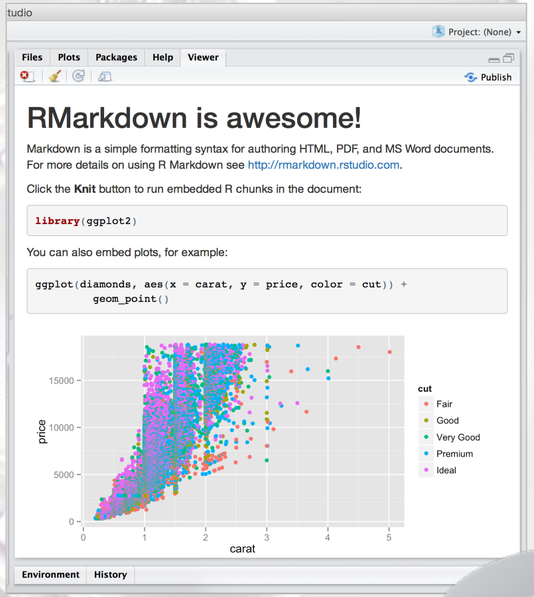
don't touch the raw data

keep track of all analysis steps

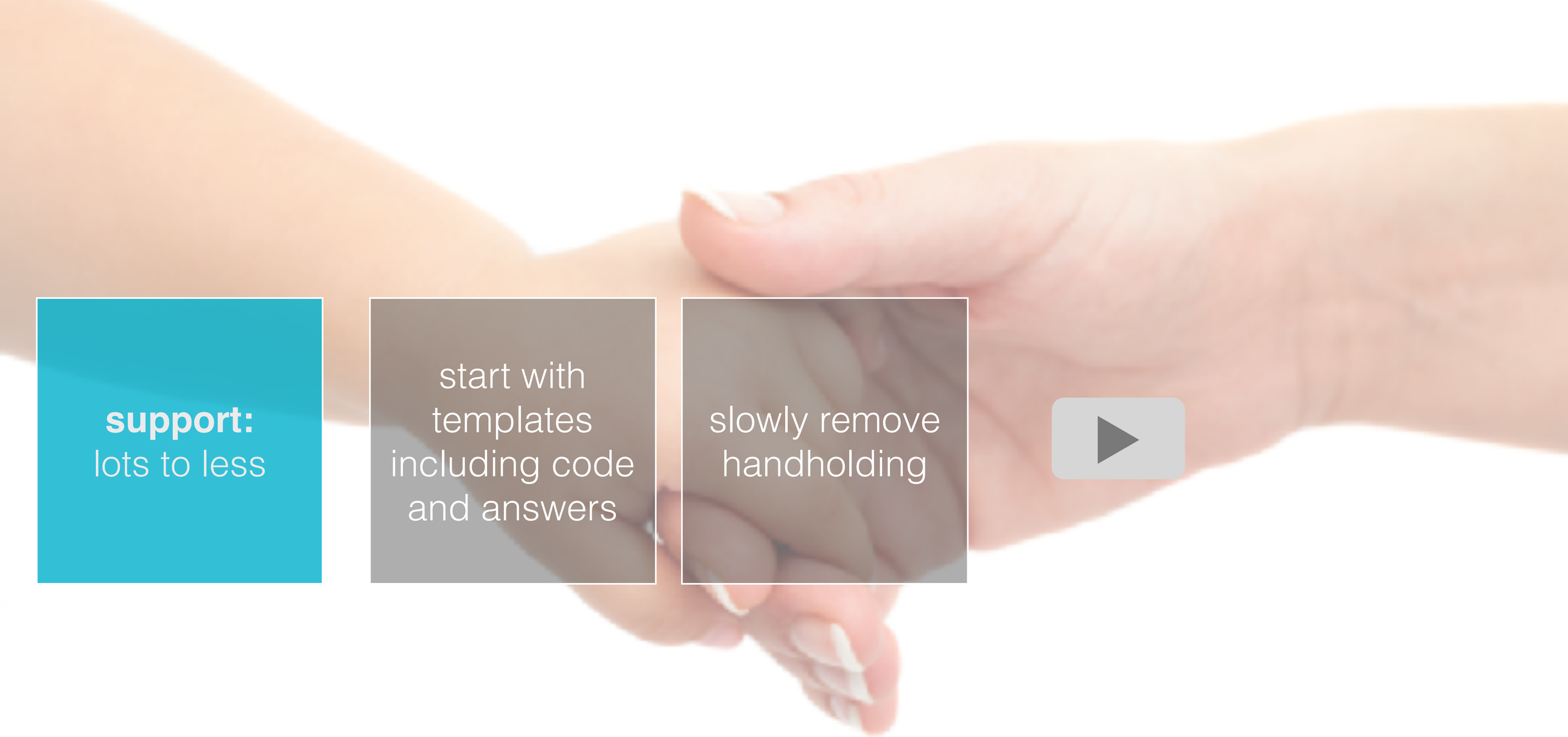
avoid copy-paste

toolkit

RStudio +



= Literate programming in



support:
lots to less

start with templates including code and answers

slowly remove handholding

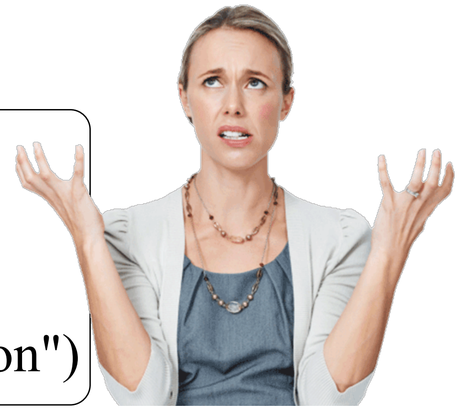
learn R

avoid the messy / frustrating console

built-in and consistent syntax highlighting

R Markdown learning outcomes (beyond reproducibility)

```
n <- 1000
p <- seq(0, 1, 0.01)
me <- 2 * sqrt(p * (1 - p)/n)
plot(me ~ p, ylab = "Margin of Error", xlab = "Population Proportion")
```



```r
n <- 1000
p <- seq(0, 1, 0.01)
me <- 2 * sqrt(p * (1 - p)/n)
plot(me ~ p, ylab = "Margin of Error", xlab = "Population Proportion")
```

R Markdown learning outcomes (beyond reproducibility)

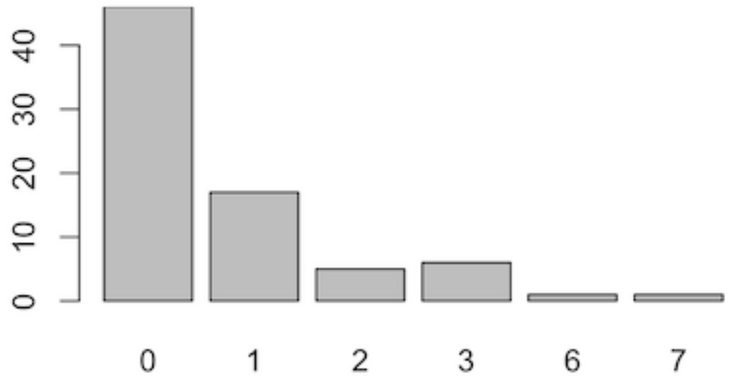
learn R

avoid the messy / frustrating console

built-in and consistent syntax highlighting

code and output always together

```
sim_streak <- calc_streak(sim_basket)
barplot(table(sim_streak))
```




```
median(sim_streak)
```

```
## [1] 0
```

```
IQR(sim_streak)
```

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

R Markdown learning outcomes (beyond reproducibility)

learn R

avoid the messy / frustrating console

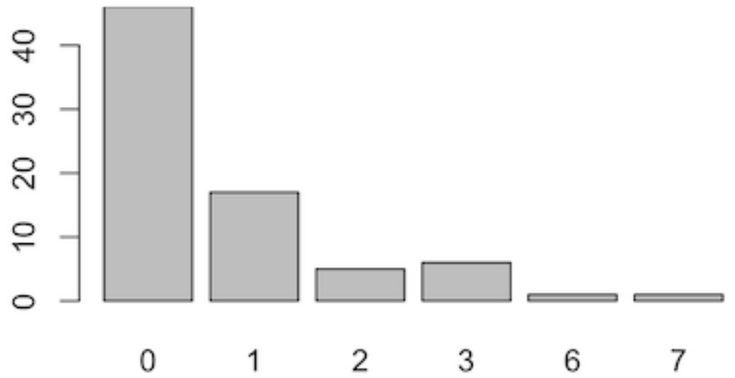
built-in and consistent syntax highlighting

code and output always together

feedback + grading

ambiguity removed

```
sim_streak <- calc_streak(sim_basket)
barplot(table(sim_streak))
```




```
median(sim_streak)
```

```
## [1] 0
```

```
IQR(sim_streak)
```

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

R Markdown learning outcomes (beyond reproducibility)

learn R	avoid the messy / frustrating console	built-in and consistent syntax highlighting	code and output always together
feedback + grading	ambiguity removed		
collaboration	just share the Rmd		

resources

designed to be adopted / adapted

OpenIntro

openintro.org

specific to my course

stat.duke.edu/~mc301

mine-cetinkaya-rundel

acknowledgements

mark mccahill, duke OIT

thank you!

comments / questions?

mine@stat.duke.edu
@minebocek
mine-cetinkaya-rundel