

Data Science

CS301

Intro to R

Week 2

Fall 2024

Oliver BONHAM-CARTER



Where To Now?

- Softwares exist to do analysis
- Created by developers for a specified purpose (i.e., web site traffic, economy trends, etc.).





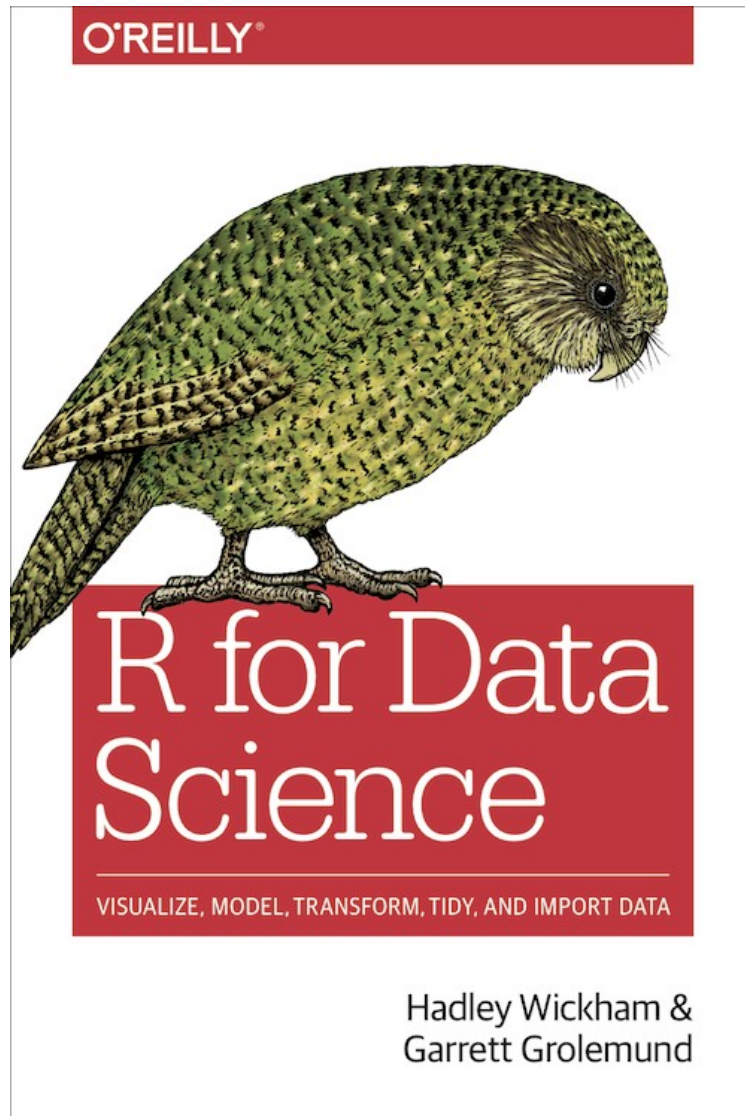
For Your Own Analysis?

- **BUT! What if you are working on a project and no tools currently exist?!**

Develop
Your
Own
Tools!!



We will be using the Book



- Note the chapters between the book and the website are not numbered identically!
- Book:
 - Chap 1: Data Visualization with ggplot
 - ~~Chap 2: Workflow, Basics~~
- On the web site:
 - <http://r4ds.had.co.nz/>
 - Chap 3: Data Visualization
 - ~~Chap 4: Workflow, Basics~~



Who uses R?

R applications are not enough until you don't know how people/companies are using the R programming language.

Facebook – Facebook uses R to update status and its social network graph. It is also used for predicting colleague interactions with R.

Ford Motor Company – Ford relies on Hadoop. It also relies on R for statistical analysis as well as carrying out data-driven support for decision making.

Google – Google uses R to calculate ROI on advertising campaigns and to predict economic activity and also to improve the efficiency of online advertising.

Foursquare – R is an important stack behind Foursquare's famed recommendation engine.

John Deere – Statisticians at John Deere use R for time series modeling and also geospatial analysis in a reliable and reproducible way. The results are then integrated with Excel and SAP.

Microsoft – Microsoft uses R for the Xbox matchmaking service and also as a statistical engine within the Azure ML framework.

Mozilla – It is the foundation behind the Firefox web browser and uses R to visualize web activity.

Ref: <https://data-flair.training/blogs/r-applications/>

Who uses R?

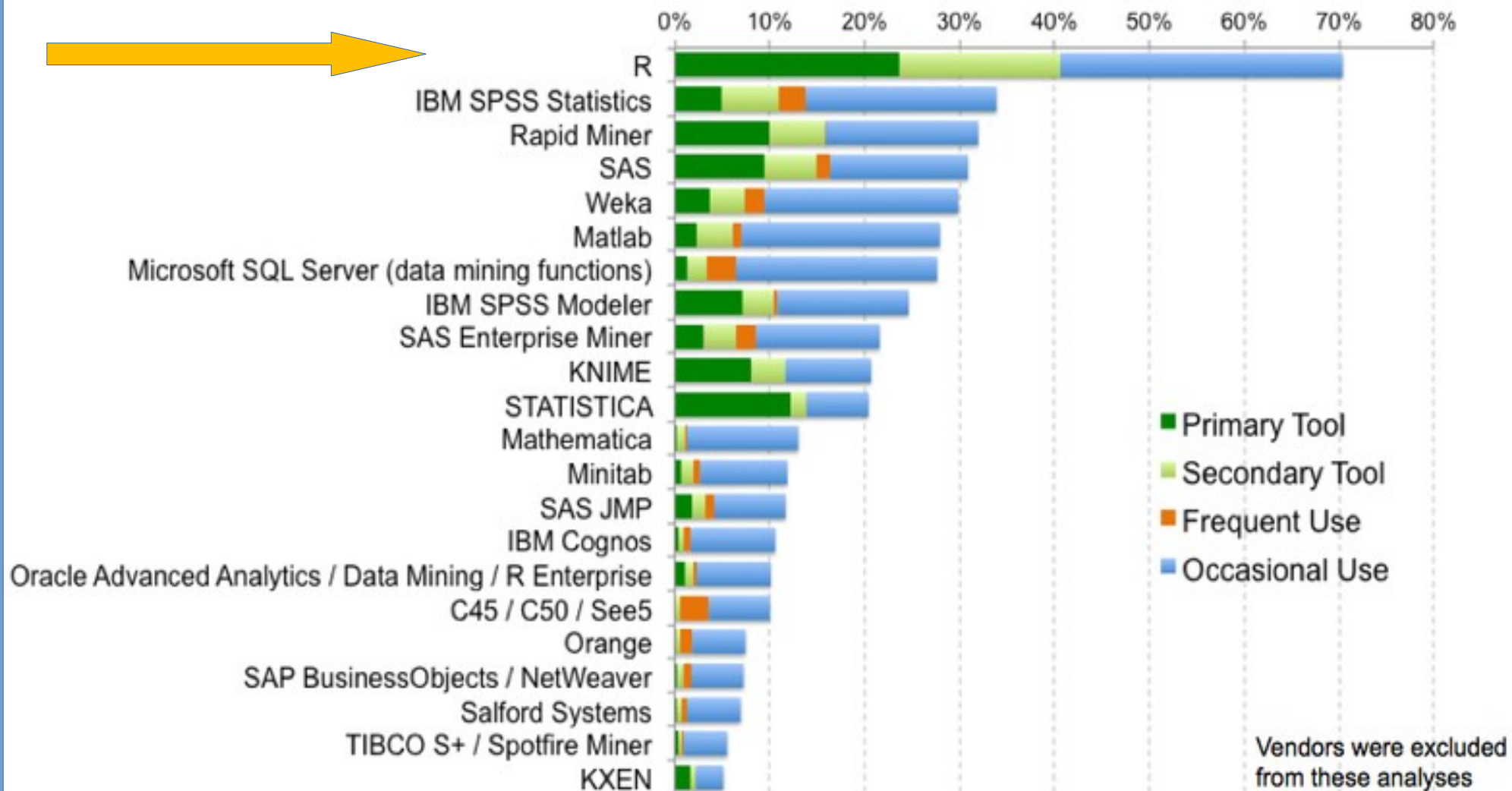


Companies that use R for Analytics





R: The Most Popular Data Mining Tool



NEWS

COMPUTING

Top Programming Languages 2022 >

Python's still No. 1, but employers love to see SQL skills

BY STEPHEN CASS | 23 AUG 2022 | 4 MIN READ |

ALLEGHENY
COLLEGE

R is in
the top
Ten!

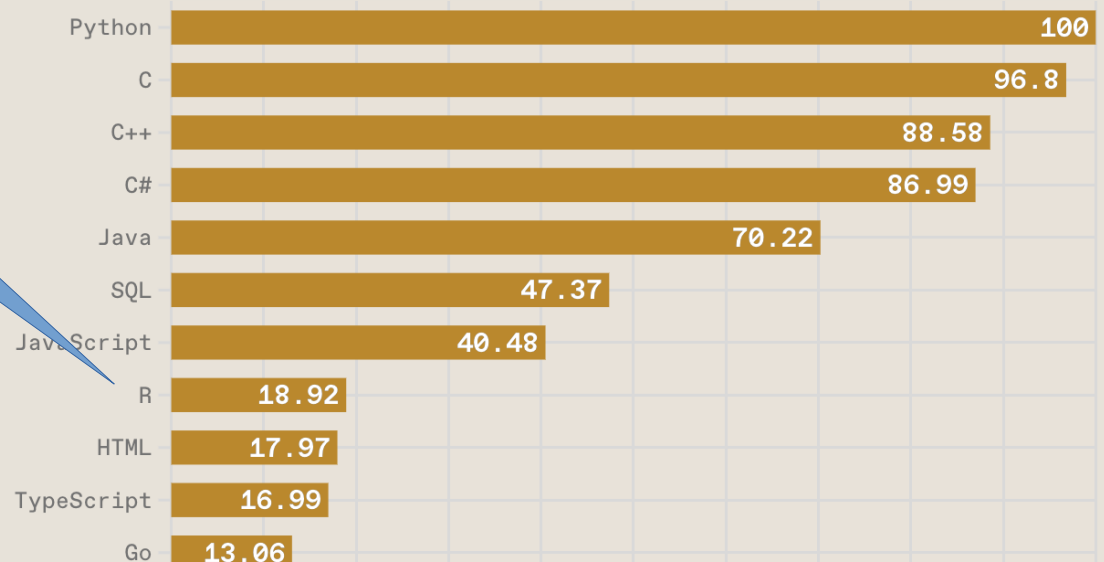
Top Programming Languages 2022

Click a button to see a differently weighted ranking

Spectrum

Jobs

Trending

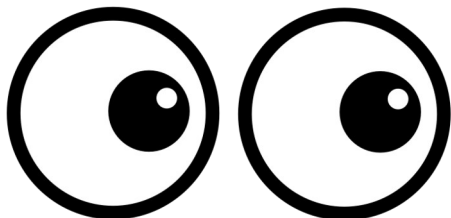


<https://spectrum.ieee.org/top-programming-languages-2022>



Let's look at R!

Click this
image to
go to
resources
!!



[Home](#) / [Resources](#) / [Data Science Resources](#)

Data Science Resources

Welcome to a resources page for Data Science research. Here you will find a list of links for data, tools, tutorials and related resources that may be very helpful to your work.

```
# You can run any R code...
print("Hello, world!")

# Use plots...
plot(cars)

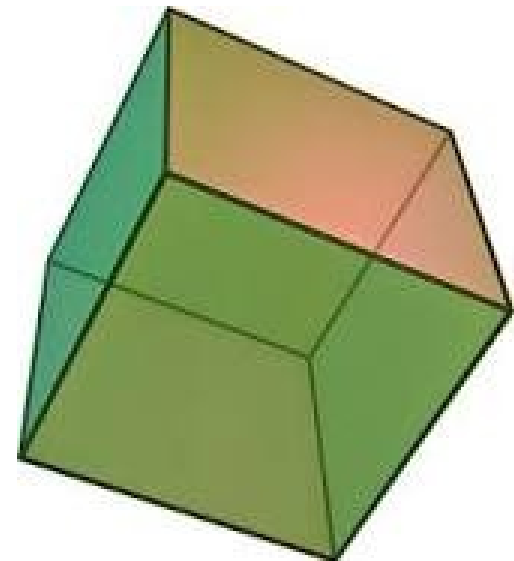
# Even packages like ggplot!
library(ggplot2)
qplot(wt, mpg, data = mtcars, colour = factor(cyl))
```

Run (Cmd-Enter)



Variable Names

- Variable Names:
 - Begin with a letter, and can only include letters, numbers, periods and underscore chars.
 - Underscores: “_”
 - Periods: “.”
 - Different character cases
- SnakeCase (recommended by book)
 - `val_of_height`,
 - `val_of_length`,
 - `val_of_width`





Basic Math

- Mathematics
 - Addition: $1 + 1$
 - Subtraction: $1 - 1$
 - Multiplication: $3 * 7$
 - Division: $1 / 4$
- More complicated math, var assignments:
 - $4*(7+3)/10+1$ **Note: watch the order of operations!**
 - Parameter of circle ($C = 2 * \text{pi} * r$)
 - $R <- 4$, Note the “<-” means *equal* in R.
 - $C <- 2 * \text{pi} * R = 2 * 3.1415 * 4$
 - C is 25.13274

Variable Names

- CamelCase:
 - valOfHeight,
 - valOfLength,
 - valOfWidth
- Period.Case
 - Val.of.height,
 - Val.of.length,
 - Val.of.width
- What-EVER.Case
 - Val.ofHEIGHT,
 - Val.Of_Length,
 - Val.oF.Width





Working On a Case (I)



```
firstName <- "Sherlock"  
print(firstName)  
[1] "Holmes"
```

Camal Case



Working On a Case (I)

```
Last.name <- "Holmes"  
print>Last.name)  
[1] "Holmes"
```

Period Case

```
firstClue <- "stain"  
second.clue <- "scarf"
```

**Whatever
Case (a mix)**



Working with Variables

```
x = 1
y = 3
# or
x <- 1
y <- 3
# Run:
x + y
#output
[1] 4
```

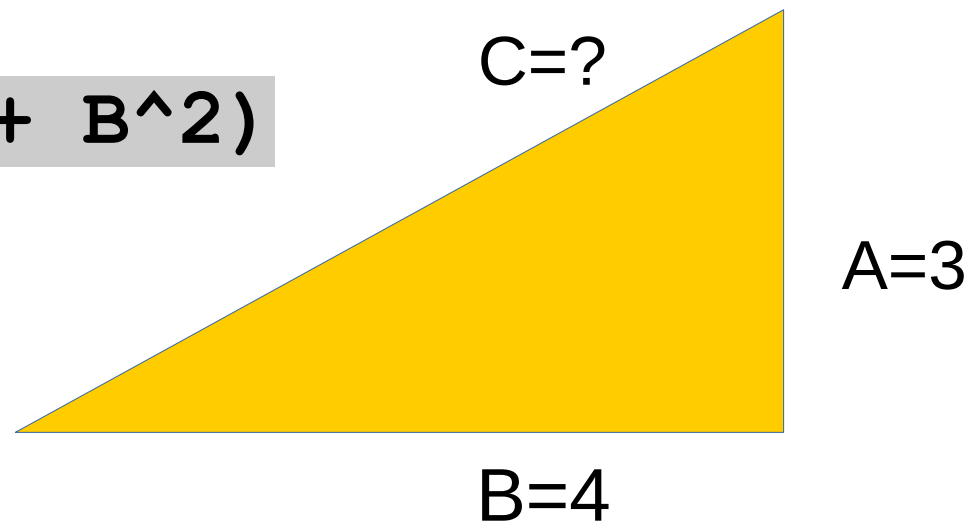
```
> x <- 1
> y <- 3
> x + y
[1] 4
```

```
> myNum <- -2
> myOtherNum <- -4
> myNum + myOtherNum
[1] -6
```



Variables and Assignments

- `A <- 3`
- You could also use “`A=3`” (but this is not traditional programming in R)
- Hypotenuse (C) defined by $\text{sqrt}(A^2 + B^2)$
- `A <- 3`
- `B <- 4`
- `C <- sqrt(A^2 + B^2)`
- `C is ??`



Logical Operations

- Booleans: Returning True or False:

`3 > 4, 3 < 4,`

`2 + 4 == 6,`

`2 + 3 == 4 + 1`

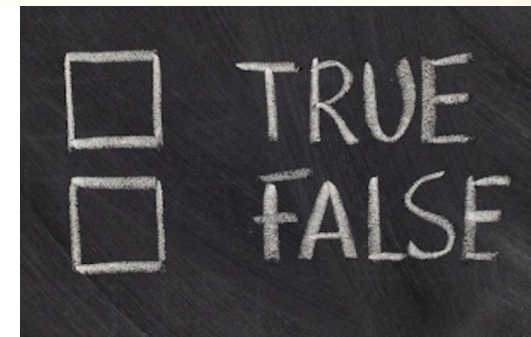
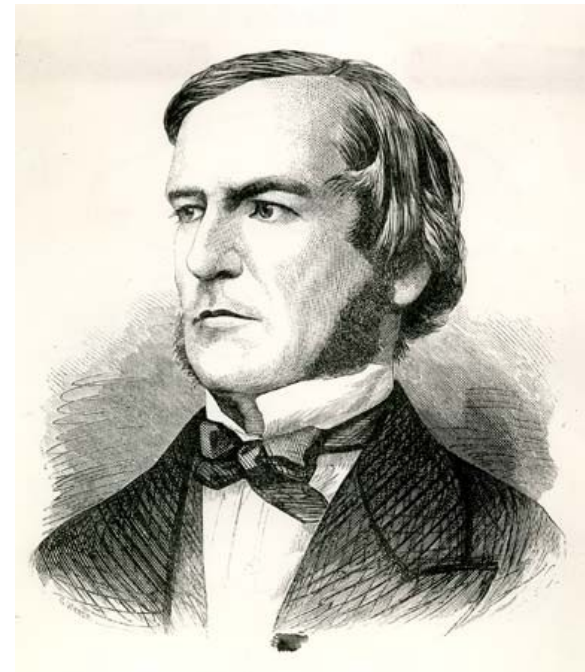
`T == TRUE`

`F == FALSE`

`3 + 4 != 5`

`3 + 4 == 7`

`5 * 2 != 11`





Try some of These in R!

- Logical **AND**

- (&&)

F && F is F

F && T is F

T && F is F

T && T is T

- Logical **OR**

- (||)

F || F is F

F || T is T

T || F is T

T || T is T

- Logical **NOT**

- (!)

!F is T

!T is F

TRUE

FALSE

Truth Tables:

https://en.wikipedia.org/wiki/Truth_table

De Morgan's Laws:

https://en.wikipedia.org/wiki/De_Morgan%27s_laws



Concatenating Strings

- Strings have quotation marks
 - “Hello World”

```
H <- "Hello"
```

```
W <- "world"
```

```
paste(H,W, sep = " ")
```

> Hello, world!_

> Hello, world!_

> Hello, world!_

> Hello, world!_

> Hello, world!_

What is the result here??



You try:
Print your full name!



```
first <- "Sherlock"  
last <- "Holmes"  
paste(first,last, sep =" ")
```



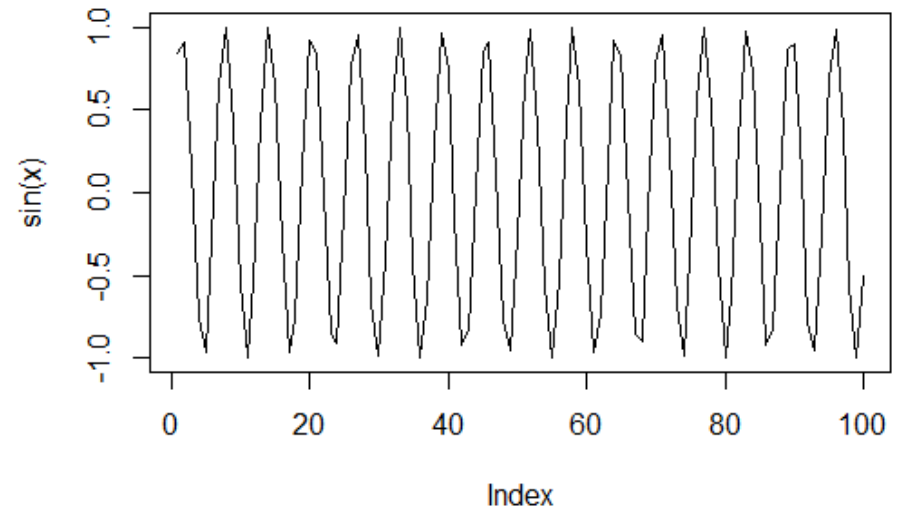
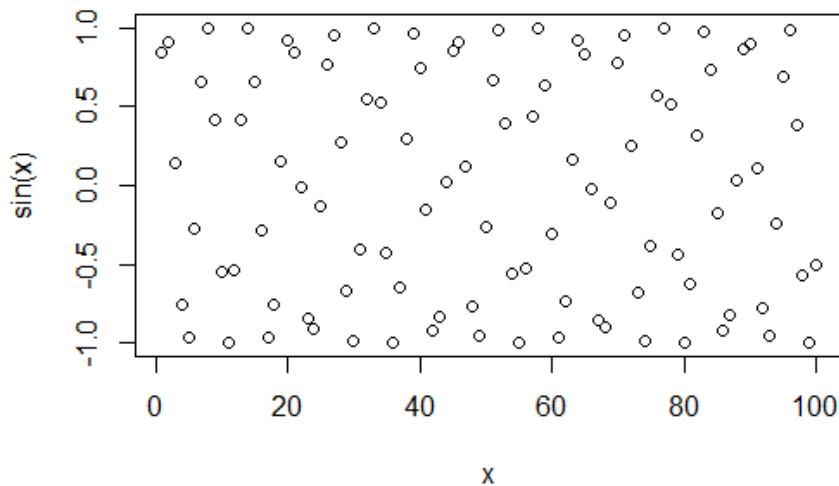

Built-in Functions

- R has a large collection of built-in functions:
 - `function_name(arg1 = val1, arg2 = val2, ...)`
 - `seq(from, to)`, ex: `seq(0, 10)`
 - Gives a sequence, $S = \{0, \dots, 10\}$
 - What happens when you press TAB after typing, “seq”?
- Use the `sum()` function to add two numbers
 - `sum(1,10)`
 - Adds 1 and 10
- Add all elements in a vector, `v`
 - `v <- 0:10`
 - `sum(v)`
 - Adds: $0 + 1 + \dots + 9 + 10 = 55$



Simple Plots



























```
1,100) # assign x to the sequence 1 to 100  
# plot this sequence  
(x)) or plot(x,sin(x)) # see left plot below  
(x)) or plot(x,sin(x), type = "l") # see right plot :
```



Simple Plots

Plotting symbols

```
x<- seq(1,100)
# Try these!
plot(sin(x), pch = 1)
plot(sin(x), pch = 2)
...
plot(sin(x), pch = 25)
```

0 	1 	2 	3 	4 	
5 	6 	7 	8 	9 	
10 	11 	12 	13 	14 	
15 	16 	17 	18 	19 	
20 	21 	22 	23 	24 	25 

See more at reference:

<http://www.sthda.com/english/wiki/r-plot-pch-symbols-the-different-point-shapes-available-in-r>



Now, You Try

- Use R to write a command that...
 - Finds the **sum** of all numbers, 0 through 100
 - Finds the **sum** of all numbers, 0 through 10000
(now, set a variable equal to the sequence first)
- Using the plot function, `plot(x,y,type = "l")` to plot a line of the function, $f(x) = \sin(x)$ for x in $\{0, \dots, 30\}$
 - `x <- 0:10`
 - `plot(x, sin(x), type = "l")`

*Now try
`cos()` and `tan()`!*

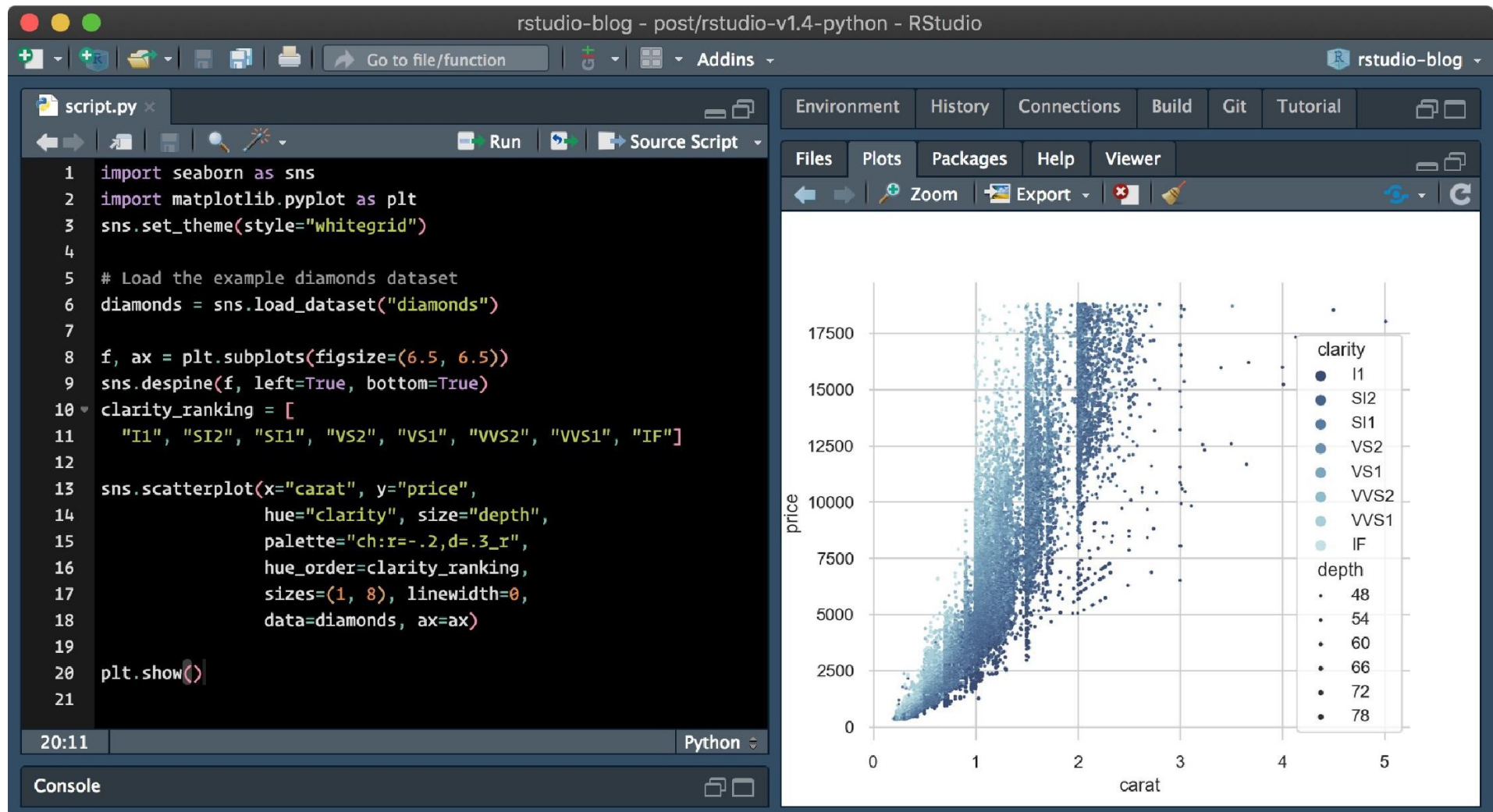
Exiting R:
`q()`

THINK

The R Programming Language

- <https://www.r-project.org/>
- What is the R language?
 - An open source, well-developed programming platform for work in statistics, mathematics and data analytics
 - Cross platform; runs on major OSs
 - Popular programming skill among Big Data analysts, and data scientists
- Community Blogs:
 - <https://www.r-bloggers.com/>
 - <https://twitter.com/rstudiotips/>
 - <https://towardsdatascience.com/>





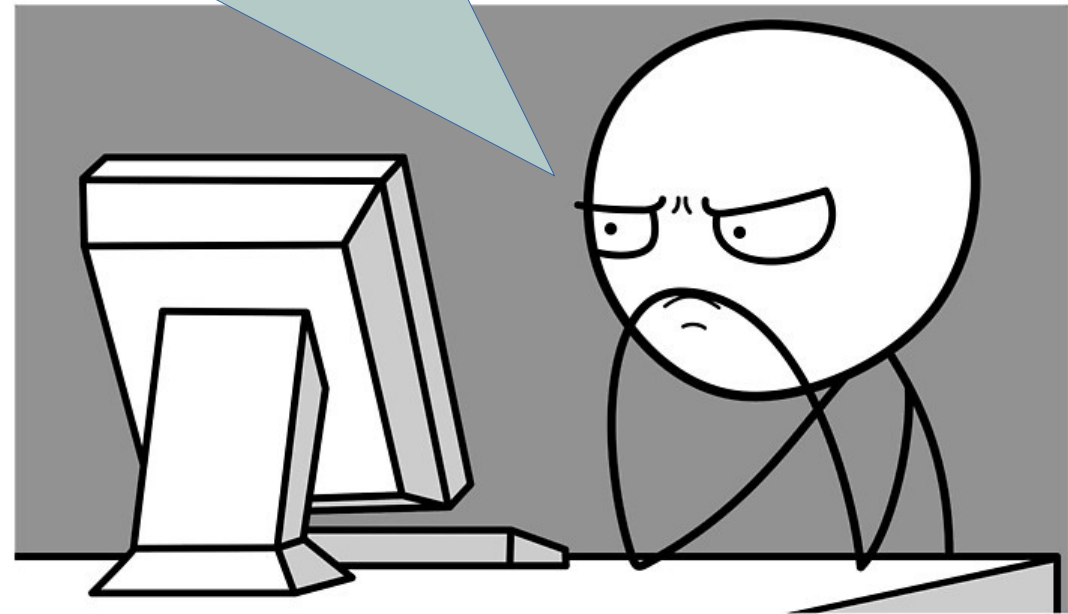


ALLEGHENY
COLLEGE

Let's take a moment
to install these
software packages!

 Language

 Studio®





ALLEGHENY
COLLEGE

**Install
links**



RStudio download

<https://posit.co/downloads/>



R Programming Language

<https://cran.rstudio.com/>

Verify Your Installation!

- Wait! R or RStudio? Same language!

To run:

Find its icon or type *rstudio* at terminal

```
R version 3.2.2 (2015-08-14) -- "Fire Safety"
Copyright (C) 2015 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
```

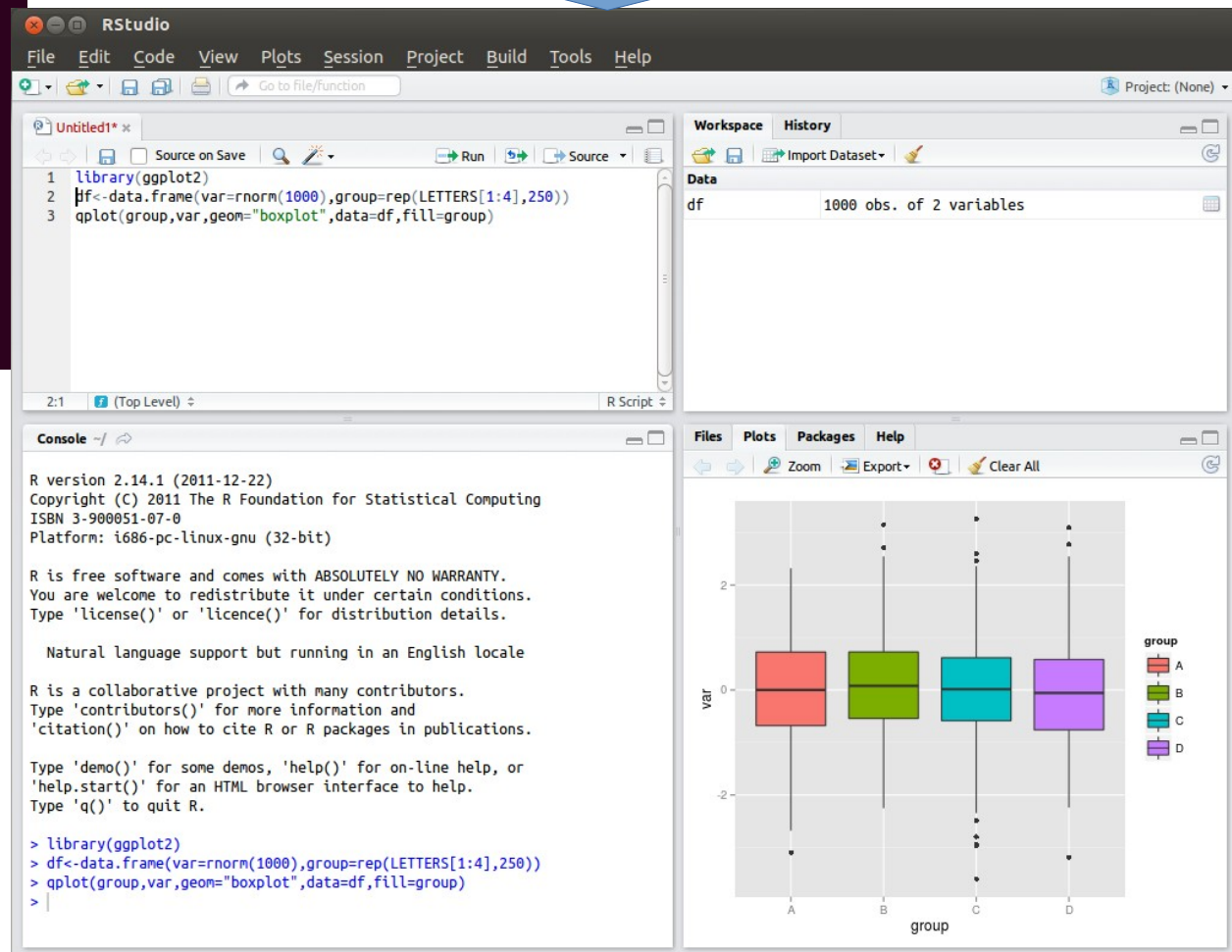
```
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
```

```
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.
```

```
> |
```



To run:
Type "R" at terminal





Getting Help in R

- Online help: place a “?” in front of a keyword
 - Ex: ?print

The screenshot displays the R Studio interface. The console on the left shows the R version (3.4.0) and startup messages. The environment pane on the top right shows a variable 'x' of type 'int' with values 1 through 10. The help viewer on the bottom right, titled 'R: Concatenate Strings', is highlighted with a red border and contains the following information:

Files **Plots** **Packages** **Help** **Viewer**

R: Concatenate Strings Find in Topic

paste {base} R Documentation

Concatenate Strings

Description

Concatenate vectors after converting to character.

Usage

```
paste(..., sep = " ", collapse = NULL)
paste0(..., collapse = NULL)
```

Arguments

...	one or more R objects, to be converted to character vectors.
sep	a character string to separate the terms. Not NA_character_ .
collapse	an optional character string to separate the results. Not NA_character_ .

Please take notes!!
We will be coding together.