Course actually starts way-below. See

Section: HarvardX/ PH525.1x Statistics & R for Life Sciences

Inhalt:

Broken Arrows:

Pdf document in this folder: Functions () R programming

Presentations Made at 9-13-16 BARUG Meeting

- Ankur Gupta: Quirks of R: http://www.perfectlyrandom.org/r/talks/barug-sep-13-2016/#/9
- Aaron Hoffer: Extending Shiny: Building reactive drag & drop elements: https://ayayron.shinyapps.io/ShinyDND_BARUG_Lightning_Talk_Sept2016/
- Dex Groves: Winning 24-hour Predictive Modeling Competitions: http://dexgroves.github.io/talks/

From notes renegade R programming:

Achtung!

There are 2x Sections in this document.

- Immediately below is from multiple sources and still has to be cleaned-up and subsumed.
- Below this are notes from the book :Book: Beginning R and introduction to statistical programming

Last active section(s) dated:

MapR Converged Data Platform:

- http://info.mapr.com/WB_PredictingChurn_Global_DG_17.06.15_ConfirmationPage_CDP.html?aliId=36527615

Upcoming in-person courses (you're scheduled) in R at Stanford http://library.stanford.edu/events/mapping-and-spatial-analysis-r-part-1

ggplot2

- https://en.wikipedia.org/wiki/Ggplot2
  - http://ggplot2.org/

Data Visualization: https://www.springer.com/us/search?query=data+visualization&submit=Submit

R programming

- *https://en.wikipedia.org/wiki/R_(programming_language)#Packages*
- *installing packages in R (plus a whole series (video) on R)*https://www.youtube.com/watch?v=cX532N_XLIs&list=PLqzoL9-eJTNBDdKgJgJzaQcY6OXmsXAHU

- *The R manuals*: https://cran.r-project.org/manuals.html
- *ssearch*: Designing Regression using R: https://www.google.com/search?client=opera&q=designing+regressions+using+r&sourceid=opera&ie=UTF-8&oe=UTF-8

Les Doigts Cassé

- *Other export methods to and from R*
- *What is a vector in R and does agree with the ML and Physics vectors?*
- *How to list all user installed R pkgs and versions?*
- *How to merge from 2 different sheets, files, different formats and what if at least one is not on the computer?*

==========

Need to learn how to represent and graph equations for the calculus course to check answers and for insight

- How to install and implement ggplot2 for slope and line equations https://www.youtube.com/watch?v=1n3kiuxW8HU
  - *Question: What effect will this have on base plot in R? or is it a case where I have to type this every time I create a session such that it's available over or with base plot?*
  - *Question: Is it possible to switch between the two packages?*
  - *Question: Is it possible to have both outputs on the same graph…, from both packages?*
  - *Question: after I install… I typed library("ggplot2:). Seemed to work but I got this message:Use SuppressPackageStarupMessages() to eliminate package startup messages!*
  - *Question: How to see what packages are installed in r and a specific package? What's all that other utility stuff that goes with the command and how to use it?: https://www.google.com/search?client=opera&q=how+to+see+what+packages+are+installed+in+r&sourceid=opera&ie=UTF-8&oe=UTF-8*
  -
- How to switch between ggplot2 and base plot (Question re: accessibility of each and can you use output from both on the same plot

==========

R programming: Defining functions in R. (not quite the same as in Calculus but there are some parallels!)

- General search: "Using functions in R": https://www.google.com/search?client=opera&q=using+function+in+r&sourceid=opera&ie=UTF-8&oe=UTF-8

Specific: https://www.datacamp.com/community/tutorials/functions-in-r-a-tutorial#gs.hcHTttE

==========

Graphing equations in R:

- https://www.google.com/search?client=opera&q=graphing+equations+with+r&sourceid=opera&ie=UTF-8&oe=UTF-8
- http://stackoverflow.com/questions/26091323/how-to-plot-a-function-curve-in-r
- https://www.google.com/search?client=opera&q=graphing+equations+with+r&sourceid=opera&ie=UTF-8&oe=UTF-8

==========

R Documentation: https://www.google.com/search?client=opera&q=r+documentation&sourceid=opera&ie=UTF-8&oe=UTF-8
- How to get help with commands, functions and features?
- I don't understand the documentation syntax: *par example:* https://stat.ethz.ch/R-manual/R-devel/library/utils/html/installed.packages.html
    - Getting help in R (from the Introduction to R Doc): https://cran.r-project.org/doc/manuals/r-release/R-intro.html#Introduction-and-preliminaries

==========

Thursday, September 8, 2016

Microsoft: DAT204x Introduction to R (Using archived course) See notes in Edx folder

This course is lame. At least when they hand you off to the DataCamp. Perhaps the course selected is too basic. Anyway I plan on approaching R from
- The Wed September 7th approach (link online)
- This Edx course (lame). I'm still in the DataCamp Section!
- As I'm trying to directly use it in Calculus:
    - Plot graphs and slopes etc.

I also plan on getting a **Wolfram** Subscription and pursuing that in parallel along with any R features it may support!

DataCamp, R

R works with numerous data types. Some of the most basic types to get started are:
- Decimals values like 4.5 are called numerics.
- Natural numbers like 4 are called integers. Integers are also numerics.
    - Does this jive with having to input them as 4L? Otherwise it's a "double"
    - Note the error message you get if you type x <- 4.5L / Integer literal 4.5L contains a decimal; using numeric value!/
- Boolean values (TRUE or FALSE) are called logical.
- Text (or string) values are called characters.
- Note how the quotation marks on the right indicate that "some text" is a character.

Note what class() does. How does this compare with typeof(), str(), attributes(), is.vector….etc?

OK. FINISHED BEGINNER LEVEL AND NOW I'M AT THE INTERMEDIATE COURSE. STARTING TOMORROW!
Also take the course in French, Spanish etc. to get your language skills up to speed!

==========

Udemy: Udemy seems a waste of time and money!

==========

Monday, May 8, 2017

STATUS:
Review this doc, condense and where we left off
Issues:
1. understanding error messages
2. difference between library and packages

1. seeing the packages or libraries installed
    1. *R+How to check if a package is installed?*: https://www.google.com/search?q=r+%2B+how+to+check+if+a+package+is+installed&oq=r+%2B+how+to+check+if+a+package+is+installed&aqs=chrome..69i57j0l3j69i64.7517j0j7&sourceid=chrome&ie=UTF-8
2. selectively installing or removing either
3. stdin, stdout & stderr in R (like saving to a file) and comparables or cross-overs to bash, builtins, reed commands?
    1. **How to pip to a file in R:**
        1. https://www.google.com/search?q=how+to+pipe+to+a+file+in+r&oq=how+to+pipe+to+a+file+in+r&aqs=chrome..69i57.4638j0j4&sourceid=chrome&ie=UTF-8
        2. **R + how to save to a file?:** https://www.google.com/search?q=r+%2B+how+to+save+to+a+file&oq=r+%2B+how+to+save+to+a+file&aqs=chrome..69i57j0l5.18767j0j7&sourceid=chrome&ie=UTF-8
        3.
4. **knitr: what is it and is it installed?**
    1. **What is knit:** https://www.google.com/search?q=what+is+knitr&oq=what+is+knitr&aqs=chrome..69i57j0l2.1868j0j7&sourceid=chrome&ie=UTF-8

Flexdashboard

- **Sparklyr - R**
- sparklyr.Rstudio.com
    - **sparkr**
    - **sparklyr # is this on CRAN, yet?**

RStudio

- **What is it? especially in relation to other R groups?** https://en.wikipedia.org/wiki/RStudio

Introduction to Mapping and Spatial Analysis with R: https://cengel.github.io/rspatial/

Wednesday, September 7, 2016

Online R Tutorial (Hadley Wickam): http://adv-r.had.co.nz/Data-structures.html (Advanced R)

Chapter 1: Data Structures (Summary)

Chapter summarizes the most important data structures in base R. Not much in detail but in how they're interrelated. For detailed descriptions of the data structures pursue the R documentation.

- R's base data structures: *Can be organized, how?*
    - *Compare with matricies;… vectors?*

- *In consequence to this organizational perspective what are the 5 base data types?*
  - *(almost) All other objects are built from these basics*
    - *analogy to fundamental and derived quantities in physics?*
  - *The OO field guide show how(q.v.)*

Ok, so now you have to be thoroughly comprehending of the following:

- *Vectors introduces you to atomic vectors and lists, R's 1d data structures.*
  - *The members of a vector are called components*
  - *Note there are no 0 dimensional or (scalars) types. Individual numbers or strings you might think of as scalars are actually vectors of dimension one.*
  - *How to create & combine vectors:*
    - *if you create an atomic vector with both integer and decimal it doesn't complain but treats everyone (on output) as decimal. Same if you add 2x vectors: one integer and one decimal (should I say floating point?). Note the original vector doesn't get converted.*
      - *> joey <- c(3, -2, .6, 99)*
      - *> joey*
      - *[1] 3.0 -2.0 0.6 99.0 /# All converted to decimal*
      - *integer and one decimal (should I say floating point?). Note the original does not get converted.*
    - *One reference suggested using x<-c(1l, 3L, 5L) for making integer atomic vectors and that works. But if you mix types…, same result as above*

      > xxxx <- c(1L, 2.2L)
      warning message: integer literal 2.2L contains decimal; using numeric value
      > xxxx
      [1] 1.0 2.2

- *Attributes takes a small detour to discuss attributes, R's flexible metadata specification. Here you'll learn about factors, an important data structure created by setting attributes of an atomic vector.*
- *Matrices and arrays introduces matrices and arrays, data structures for storing 2d and higher dimensional data.*
- *Data frames teaches you about the data frame, the most important data structure for storing data in R. Data frames combine the behaviour of lists and matrices to make a structure ideally suited for the needs of statistical data.*

autrefois…

- *Explain R and scalar types. Given this definition,what is a vector in R?*
- *What are the 2 flavors of vectors? Explain relative to the data structure organization*
  - *What are their 3 properties?*

- *explain the following commands:*
  - *is.vector()*
  - *is.list || is.atomicci*
  - *stru()*
  - *typeof()*
  - *attributes()*
  - *how is.vector differs from (is.list || is.atomic)?*

Status II

Correlations between python basics and R (among other languages)

- **Does R have it's own notebook or can I just use Jupiter-Notebooks?**
- **If Jupiter-Notebooks is *multilingual*, do I have to load something to use R,…m etc. with it?**

Adding book "R for beginners" Get through Chapter 1.

==========

Saturday, September 30, 2017

Broken arrows(1):

Actuarial Analytics using R

- https://www.google.com/search?q=actuarial+analytics+using+r&oq=actuarial+analytics+using+r&aqs=chrome..69i57.9115j0j7&sourceid=chrome&ie=UTF-8
  - http://freakonometrics.hypotheses.org/1173
    - figure-out the code on this!

R for Actuarial Science:

- http://freakonometrics.hypotheses.org/1173
  - …a vector based language?
  - https://www.google.com/search?q=what+is+a+vector+based+language&oq=what+is+a+vector+based+language&aqs=chrome..69i57.9787j0j7&sourceid=chrome&ie=UTF-8
    - https://www.quantshare.com/sa-63-a-vector-based-language

Saturday, October 7, 2017
Source - Book: Beginning R and introduction to statistical programming

Clearing the R-Console:

R version 3.3.0 (2016-05-03) -- *"Supposedly Educational"* *# How to print out only the version #?*
Copyright (C) 2016 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.
[Previously saved workspace restored]

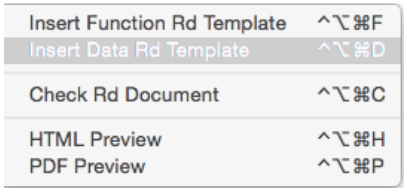
Note

I've got the RGui and R via the command line running. I haven't got RStudio running but it's installed. Why so many R's for one platform? I'll investigate the difference of RStudio, later this week! I suspect the book *Mac OS X for Unix Geeks has a good* clue!

Text indicates, to clear the console type Ctrl + L. This works if you start R from the command line. If using the GUI application, (both on Mac Os X) it's Option-Command-L

Also note q() closes and R session!

In creating a new document there are actually 2x options on the Mac. One saying "create a new document. the 2nd…"create a new Rd document. It s distinctives pictured below:



R uses the basic operators, but I'm noticing there are 2x for exponentiation:

> 2 * 2

[1] 4

> 2 **3

[1] 8

> 2 ^ 3

[1] 8

>

as well, it labels output with [numbers] for some unknown reason. Text says there are no scalar quantities in R. So are these bracketed numbers indicative of vectors?

Les Doigts Cassé

- I believed this article described R as a *Vectorized Language:*
- https://www.google.com/search?q=r+is+a+vectorized+language&oq=r+is+a+vectro&aqs=chrome.1.69i57j0l5.6038j0j7&sourceid=chrome&ie=UTF-8
- Vectorization in R, Why?: http://www.noamross.net/blog/2014/4/16/vectorization-in-r--why.html

Text indicated that [1] is the index of the 1st number of the vector. So how would I reach the others in the following case?

> x <- c(2,3,1,5)

> 3*x

[1] 6 9 3 15

x[2]

[1] 3

> [1] 3

> 3*x[3]

[1] 3

Ok. Now I get it!

==========

In the console window, you'll find some interesting tools: The 4th from left opens a panel with a history of your commands.

The power of R is not in basic mathematical calculations (though it does them flawlessly), but in the ability to assign values to objects and use functions to manipulate or analyze those objects. R

allows three different assignment operators, but we will use only the traditional <- for assignment.

You can use the equal sign = to assign a value to an object, but this does not always work and is easy to confuse with the test for equality, which is ==.

You can also use a right-pointing assignment operator ->, but that is not something we will do in this book. *When you assign an object in R, there is no need to declare or type it*. Just assign and start using it. We can use x as the label for a single value, a vector, a matrix, a list, or a data frame.

We will discuss each data type in more detail, but for now, just open a new script window and type, and then execute the following code. We will assign several different objects to x, and check the *mode* (storage class) of each object. We create a single-element vector, a numeric vector, a matrix (which is actually a kind of vector to R), a character vector, a logical vector, and a list. The three main types or modes of data in R are *numeric*, *character*, and *logical*. Vectors must be *homogeneous* (use the same data type), but lists, matrices, and data frames can all be *heterogeneous*. I do not recommend the use of heterogeneous matrices, but lists and data frames are commonly composed of different data types. Here is our code and the results of its execution. Note that the code in the R Editor does not have the prompt > in front of each line.

I don't understand this error:

2017-10-07 21:45:14.312 R[1371:118446] Communications error: <OS_xpc_error: <error: 0x7fff72fffb60> { count = 1, contents =
"XPCErrorDescription" => <string: 0x7fff72ffffa8> { length = 22, contents = "Connection interrupted" }
}>

Notes: from R programming Udemy

Introduction to R ($35.00 Udemy)

+programming_b_44895771829_c&utm_source=adwords&gclid=CO3Q9o_FzMYCFdccgQod22wPyw&matchtype=b&utm_content=44895771829&couponCode=43049d1b-ad3e-41ed-8f11-08aab8bf8840

search: Designing Regression using R: https://www.google.com/search?client=opera&q=designing+regressions+using+r&sourceid=opera&ie=UTF-8&oe=UTF-8

Meetup: July 17th
Bayesian analysis in R via stan + a bonus talk
Group: R-Ladies San Francisco
https://www.meetup.com/rladies-san-francisco/events/262662040/?refund_policy=true&rv=ea1_v2&_xtd=gatlbWFpbF9jbGlja9oAJGU1MmRjNmY5LTQ0ZWYtNDliYi04OTY5LWQ4NWI5ZmU2NjA0YQ&_af=event&_af_eid=262662040&response=3&action=rsvp

R Programming+ system commands:

https://www.google.com/search?q=r+programming+%2B+system+command&oq=r+programming+%2B+system+command&aqs=chrome..69i57j69i60l3j69i59l2.4278j0j7&sourceid=chrome&ie=UTF-8

https://stat.ethz.ch/R-manual/R-devel/library/base/html/system.html

R programming + for loops:
https://www.google.com/search?q=r+programming+%2B+for+loops&oq=r+programming+%2B+for+loops&aqs=chrome..69i57j69i64l3.4037j0j7&sourceid=chrome&ie=UTF-8

R programming + Range function:
- https://www.google.com/search?q=r+programming+%2B+range&oq=r+programming+%2B+range&aqs=chrome..69i57j69i64l3.6988j0j7&sourceid=chrome&ie=UTF-8
- http://www.datasciencemadesimple.com/range-function-in-r/

Literal (Computer Programming):
https://en.wikipedia.org/wiki/Literal_(computer_programming)

R programming + cor function:
https://www.google.com/search?q=r+programming+cor+function&oq=r+programming+cor+function&aqs=chrome..69i57j69i64l3.5924j0j7&sourceid=chrome&ie=UTF-8

R programming + r vs Object R:
https://www.google.com/search?q=r+programming+%2B+r+vs+object+r&oq=r+programming+%2B+r+vs+object+r&aqs=chrome..69i57j69i64l3.13468j0j7&sourceid=chrome&ie=UTF-8

R lists - Learn all you can do with lists in R: Data Flair ***(*** *et encore plus sur R. Voir les tables de matières* ***)***
- https://data-flair.training/blogs/r-list-tutorial/
- https://www.google.com/search?q=r+programming+does+r+have+lists&oq=r+programming+does+r+have+lists&aqs=chrome..69i57j69i64l3.10008j0j7&sourceid=chrome&ie=UTF-8

Programming with R: Data types and structures:
https://swcarpentry.github.io/r-novice-inflammation/13-supp-data-structures/

R programming: typeof() vs class():
https://www.google.com/search?q=r+programming+%2B+typeof()+vs+class()&oq=r+programming+%2B+typeof()+vs+class()&aqs=chrome..69i57j69i64l3.16394j0j7&sourceid=chrome&ie=UTF-8

R programming only zeros may be mixed with negative subscripts: *(subsetting problem)*
https://www.google.com/search?q=r+programming+only+zeros+may+be+mixed+with+negative+subscripts&oq=r+programming+only+zeros+may+be+mixed+with+negative+subscripts&aqs=chrome..69i57j69i64l3.13170j0j7&sourceid=chrome&ie=UTF-8

R programming and MapR:
- https://www.google.com/search?ei=xOxHXL2KM47B0PEPlbipwAU&q=r+programming+and+mapr&oq=r+programming+and+mapr&gs_l=psy-ab.3..33i22i29i30.5373.5681..6041...0.0..0.81.144.2.....0....1..gws-wiz.....0i71j35i39j0i22i30j0i22i10i30.uFHh0uumWFI
- https://mapr.com/
- https://en.wikipedia.org/wiki/MapR

R programming + what is mutually exclusive:
https://www.google.com/search?q=r+programming+%2B+what+is+a+mutually+exclusive&oq=r+programming+%2B+what+is+a+mutually+exclusive&aqs=chrome..69i57j69i64l3.23191j0j7&sourceid=chrome&ie=UTF-8

R programming + output:
https://www.google.com/search?ei=Z_7YXPqwI6na0gLl6ZuQCw&q=r+programming+%2B++output&oq=r+programming+%2B++output&gs_l=psy-ab.3..0i7i30l2j0i30l2j0i8i30.18730.20661..21906...0.0..0.161.1097.10j2.....0....1..gws-wiz.....0i71j35i304i39.SGVY5usRWv4

Programming a Stream:
https://www.google.com/search?ei=yoIZXYGQEY73-gSX7bnIDA&q=programming+a+stream&oq=r+programming+%2B+stream&gs_l=psy-ab.1.0.0i22i30l2.45940.50014..53732...6.0..0.99.973.12.....0....1..gws-wiz.....0i71j33i21j33i22i29i30.Uw6IfRsllrI

R programming a stream:
- https://www.google.com/search?ei=AYMZXZyMB9P5-gSckJfADA&q=r+programming+a+stream&oq=r+programming+a+stream&gs_l=psy-ab.3...3293.3535..4392...0.0..0.81.140.2.....0....1..gws-wiz.....0i71j35i304i39j0i13i30.2pH_7_l-9Og
- https://www.google.com/search?ei=I4MZXfjtKYrm-gSXgpeQBg&q=r+programming+and+streaming+data&oq=r+programming+and+streaming+data&gs_l=psy-ab.3..33i22i29i30.44647.48834..50094...0.0..0.97.799.10.....0....1..gws-wiz.....0i71j33i160.hBW87nTtvsI

R programming and streaming data into:
https://www.google.com/search?ei=VoMZXcyAG9T6-gT39JPgBQ&q=r+programming+and+streaming+data+into&oq=r+programming+and+streaming+data+&gs_l=psy-ab.1.0.33i22i29i30.5805.11796..14639...4.0..0.74.437.7.....0....1..gws-wiz.....35i39.wGWk6KJWh4s

Stream processing solutions: (Not necessarily R)
https://www.google.com/search?ei=ZYMZXcvoNJXV-gS63YXoAw&q=stream+processing+solution&oq=r+programming+and+streaming+data+into&gs_l=psy-ab.1.0.0i71l2.0.0..78296...0.0..0.0.0.....0.....gws-wiz.SQLw9LpZPSM

R programming + Why numbered output?:
https://www.google.com/search?q=r+programming+%2B+why+numbered+output&oq=r+programming+%2B+why+numbered+output&aqs=chrome..69i57j69i64l3.7104j0j7&sourceid=chrome&ie=UTF-8

https://www.google.com/search?ei=AIUZXZ2jA5et-gSYuZawAQ&q=r+programming+%2B+numbered+output&oq=r+programming+%2B+numbered+output&gs_l=psy-ab.3..33i22i29i30l2.58146.62375..62551...1.0..0.112.1489.17j2.....0....1..gws-wiz.....0i71j0i22i30j0i22i10i30j0i13i30j0i8i13i30j33i160j33i10.gj5ipR3649w

https://www.google.com/search?ei=FfjYXPDiNoep8AP6m6KgAQ&q=r+programming+%2B+why+do+bracketed+output&oq=r+programming+%2B+why+do+bracketed+output&gs_l=psy-ab.3...19426.19426..19827...0.0..0.178.178.0j1.....0....1..gws-wiz.H1WwwRqvmvI

…about that output style of R: [1] etc. *(answer I was looking for. Above misses*)
https://kingaa.github.io/R_Tutorial/#interactive-calculations

R programming + How to see what functions are available:?
https://www.google.com/search?q=r+programming+%2B+how+to+see+what+functions+are+available&oq=r+programming+%2B+how+to+see+what+functions+are+available&aqs=chrome..69i57j69i64l3.13762j0j7&sourceid=chrome&ie=UTF-8

A Tutorial introduction to R: (massive)
https://kingaa.github.io/R_Tutorial/
…about that output style of R: [1] etc.
https://kingaa.github.io/R_Tutorial/#interactive-calculations

r programming + warning numerical expression has elements only first used:

- https://www.google.com/search?q=r+programming+%2B+what+is+a+vector&oq=r+programming+%2B+what+is+a+vector&aqs=chrome..69i57j69i64l3.5311j0j7&sourceid=chrome&ie=UTF-8
- https://www.google.com/search?q=r+programming+%2B+warning+numerical+expression+has+elements+only+first+used&oq=r+programming+%2B+warning+numerical+expression+has++elements+only+first+used&aqs=chrome..69i57j69i64l3.40780j0j7&sourceid=chrome&ie=UTF-8

What is a vector in R/How to create vectors in R:
https://www.dummies.com/programming/r/how-to-create-vectors-in-r/
https://www.datamentor.io/r-programming/vector/

Adding an element to a vector:
https://www.google.com/search?q=r+programming+%2B+adding+an+element+to+a+vector&oq=r+programming+%2B+adding+an+element+to+a+vector&aqs=chrome..69i57j69i64l3.9205j0j7&sourceid=chrome&ie=UTF-8

Physics vectors vs r programming vectors:
https://www.google.com/search?q=physics+vectors+vs+R+programming+vectors&oq=physics+vectors+vs+R+programming+vectors&aqs=chrome..69i57.9766j0j7&sourceid=chrome&ie=UTF-8

Physics vectors in R programming:
https://www.google.com/search?ei=Rv4GXcOPDquN0wK1ibCwBg&q=physics+vectors+in+r+programming&oq=physics+vectors+in+r+programming&gs_l=psy-ab.3..33i22i29i30.7182.8627..8922...0.0..1.104.995.10j2.....0....1..gws-wiz.....0i22i30j33i10.-3gJodlGGgU

R cross product of vectors:

https://stackoverflow.com/questions/36798301/r-compute-cross-product-of-vectors-physics

What is the c function in R programming?:
https://www.google.com/search?q=what+is+the+c+function+in+r+programming&oq=what+is+the+c+function+in+r+programming&aqs=chrome..69i57.8023j0j7&sourceid=chrome&ie=UTF-8

R programming methods associated with functions:
https://www.google.com/search?q=r+programming+methods+associated+with+funtions&oq=r+programming+methods+associated+with+funtions&aqs=chrome..69i57j69i64l3.12973j0j7&sourceid=chrome&ie=UTF-8

R programming lab:
https://www.google.com/search?q=r+programmingd+lab&oq=r+programmingd+lab&aqs=chrome..69i57j69i64l3.254509j0j7&sourceid=chrome&ie=UTF-8

Functional Data Structures in R Advanced Statistical Programming in R
Downloaded & Tagged (D&T)

R programming + what is a data frame:?
https://www.google.com/search?q=r+programming+%2B+what+is+a+data+frame

---

Section: HarvardX/ PH525.1x Statistics & R (for Life Sciences) or is that a different course?
Source: https://courses.edx.org/courses/course-v1:HarvardX+PH525.1x+3T2019/course/

Introduction & Resources:

Inhalt()

COURSE INTRO TO STATISTICS FOR R AND THE LIFE SCIENCE

Text to above: Course Intro to statistics for r and the life science

My name is Rafael Irizarry, and I will be teaching statistics and R for the life sciences.

If you're performing research in the life sciences, you'll probably need to understand what a p-value is, what a confidence interval is. You'll probably need to analyze data, make figures. In this class, we're going to teach you how to do all that.

Now, we're not going to use the traditional way of teaching statistics. We're not going to show you formula like this.

Instead, we're going to analyze data. We're going to use the R programming language to do this. We're going to teach you some R.

You're going to analyze data. You're going to use data to understand these concepts. We're going to cover inference. We're going to demonstrate with examples what inference is and what it means. We're going to explain what random variables are. We're going to explain what null distributions are. Again, all with real data.

We're going to use exploratory data analysis to study your data to make sure that it's usable and it's correct to use the tests that we show you how to use.

We're going to learn what the central limit theorem is and how useful it is in practice. We're going to learn some details about statistical inference such as t-test, association test, and permutation test.

Finally, we'll learn about robust statistics and power calculations. When you're finished with this class, you'll be able to perform your own analysis and understand the output of your data analysis.

Course Discussions:

Discussion Resources

Discussion board support is not provided for this course. Instead, we recommend that learners join one of the many active discussions around the internet for R programmers and statistics students. Here are some sites that we recommend:

Stack Overflow External link, for question-and-answer discussion of programming tasks. Remember to search for existing answers before asking your question.

The Statistics Stack Exchange External link, for question-and-answer discussion of statistics. Again, remember to search for existing asnswers before asking your question.

The R Project mailing lists External link, for slower-paced discussions on particular topics and for official R announcements.

1. Author
August 10, 2020 at 14:06:43
What is R Markdown: https://www.google.com/search?q=what+is+r+markdown&rlz=1C5CHFA_enUS907US907&oq=what+is+r+markdown&aqs=chrome..69i57.5535j0j7&sourceid=chrome&ie=UTF-8

~ introduction: https://rmarkdown.rstudio.com/articles_intro.html

What is shiny?
~ https://www.google.com/search?q=what+is+shiny&rlz=1C5CHFA_enUS907US907&oq=what+is+shiny&aqs=chrome..69i57.6047j0j7&sourceid=chrome&ie=UTF-8

2. Author
August 10, 2020 at 14:06:43
What is R Markdown: https://www.google.com/search?q=what+is+r+markdown&rlz=1C5CHFA_enUS907US907&oq=what+is+r+markdown&aqs=chrome..69i57.5535j0j7&sourceid=chrome&ie=UTF-8

~ introduction: https://rmarkdown.rstudio.com/articles_intro.html

What is shiny?
~ https://www.google.com/search?q=what+is+shiny&rlz=1C5CHFA_enUS907US907&oq=what+is+shiny&aqs=chrome..69i57.6047j0j7&sourceid=chrome&ie=UTF-8

3. Author
August 21, 2020 at 20:24:39
https://en.wikipedia.org/wiki/Knitr

The RStudio Community site External link, for a wide variety of discussions.

TalkStats.com External link, for discussions more oriented toward statistics than programming (though you can still find some R help there). Some of these sites require you to create an account to post, but most do not require an account if you only want to read.

Troubleshooting

If you run into bugs or errors within this course, you should not report those to the lists above. Instead, please report those directly to edX, using the *Contact Us* link here or at the bottom of any page.

Course Materials

Textbook

We will be using the textbook Data Analysis for the Life Science which is freely available on leanpub.com External link. The book is written in *R markdown* which includes the R code used for several of the analysis shown in the course. The book includes links to the specific R markdown documents but you can access all of these in this *GitHub repository* described below. We also provide an *html version of the book*.
I downloaded it from the Russian/Czech site!

R Code / Scripts

All the R code used in the labs and to make the plots in the lectures is made publicly available on a website called Github, which is popular for hosting open source software. You can navigate all the Rmd (R markdown) scripts for the labs, divided into folders by course. Follow this link to the course *GitHub repository* External link.

You can download the individual Rmd scripts from Github by clicking on the appropriate folder, then the file-name, and then the 'Raw' button. Save this file to your computer, and then open it in RStudio. We will also put links to the Rmd scripts under the videos.

Running Lab Code

All software used for the class is free and open source:

- R can be downloaded and installed from *CRAN* External link (Comprehensive R Archive Network). If possible download the latest release. As of this course's launch date, that is R-3.6.1 ("Action of the Toes"), released 05 July 2019.
- We recommend using *RStudio* External link, a slick visual interface for R

Les Doigts Cassé

- Wolfram and creating interactive documents: https://www.google.com/search?q=wolfram+and+creating+interactive+documents&rlz=1C5CHFA_enUS907US907&oq=wolfram+and+creating+interactive+documents&aqs=chrome..69i57.6990j0j7&sourceid=chrome&ie=UTF-8
  - See also, The Wolfram Affair, folder!
- Wolfram vs Rmd
- Wolfram, Rmd, Tableau
- R, Python and SQL are languages you can write your code in.
- RMD and notebook interface

En plus…

R markdown Notes from the video!

3x components of Rmd files are:
- Code chunks to run (Python, R, SQL)
- Text to display
- Yaml metadata file..to guide the Rmd build process

Render function builds a report from your file.
- can't find the knit button in RStudio for this. *Maybe have to install knitr*?
  - Render function in RStudio: https://www.google.com/search?q=render+function+in+rstudio&rlz=1C5CHFA_enUS907US907&oq=render+function+in+rstudio&aqs=chrome..69i57.7764j0j7&sourceid=chrome&ie=UTF-8

What is shiny?: https://www.google.com/search?q=what+is+a+shiny+app&rlz=1C5CHFA_enUS907US907&oq=what+is+a+shiny+app&aqs=chrome..69i57.3414j0j7&sourceid=chrome&ie=UTF-8

R Resources

- *swirl*: learn R interactively from within the R console
- *R reference card* (PDF) by Tom Short. More can be found under *Short Documents and Reference Cards*.
- *Quick-R*: quick online reference for data input, basic statistics and plots
- Thomas Girke's *R & Bioconductor manuals*
- An *R programming* class on Coursera, taught by Roger Peng, Jeff Leek and Brian Caffo
- The free "try R" class from Code School is also a good place to start:https://www.pluralsight.com/search?q=R
- *Data structures summary* by Hadley Whickham

R books

- Software for Data Analysis: Programming with R (Statistics and Computing), by John M. Chambers (Springer)
- S Programming (Statistics and Computing), by Brian D. Ripley and William N. Venables (Springer)
- Programming with Data: A Guide to the S Language, by John M. Chambers (Springer)
- R for Everyone, by Jared Lander
- R Cookbook, by Paul Teetor, O'Reilly Media
- R in Action, by Robert Kabacoff

Pre-course Survey:

https://courses.edx.org/courses/course-v1:HarvardX+PH525.1x+3T2019/courseware/dcf8031210054672a6bd2a63d6f9d9ac/20a1d6c198534560ba840f714c264550/?child=first

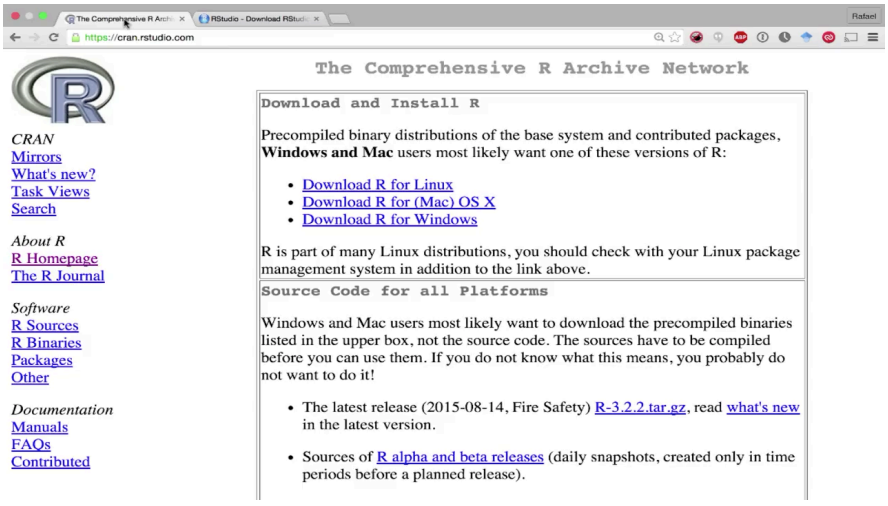
Course Week 1 > Getting Started > **Getting Started with R**

Getting Started with R: *https://courses.edx.org/courses/course-v1:HarvardX+PH525.1x+3T2019/courseware/2273065cc0f649b69c1240a58f7ab080/77c6a41ee38544c28d2d95ef7889cdb8/?child=first*

inhalt:

- *Getting Started with R*
- *First Assessment: Exercises*
- *Github*
- *RStudio*
- *Using the Textbook*

Video lecture ***Getting Started with R***



4. Author
January 19, 2020 at 18:49:55
Raw?
What does raw mean in R: https://www.google.com/search?q=r+programming+%2B+what+does+raw+mean&oq=r+programming+%2B+what+does+raw+mean&aqs=chrome..69i57j69i64l3.6159j0j4&sourceid=chrome&ie=UTF-8

https://www.google.com/search?q=r+%2B+raw+download&oq=r+%2B+raw+download&aqs=chrome..69i57j69i64l3.3503j0j7&sourceid=chrome&ie=UTF-8

- *RStudio for Organization*
- *Getting Started: Exercises*
- *Intro to DPLYR*
- *DPLYR: Exercises*

GETTING STARTED WITH R

The R code which Rafa runs in this video is available on the course GitHub repository External link

4 *You can download the individual Rmd scripts from GitHub by clicking on the filename, and then the 'Raw' button. Save this file to your computer, and then open it in RStudio.*

Shiny and RStudio: https://rstudio.com/products/shiny/

~~What is the raw button in RStudio?~~: https://www.google.com/search?q=rstudio+%2B+what+is+the+raw+button&oq=rstudio+%2B+what+is+the+raw+button&aqs=chrome..69i57.11031j0j7&sourceid=chrome&ie=UTF-8

- *Mis-understanding. The button is in Github.* ***Better start dissecting GitHub!!***

As well start reading the docs on Anaconda/~ studio. It behaves like a separate partition under Mac OS X How to and should I connect it with Mac OS X ? In particular terminal?

Friday, August 21, 2020

Stopped here!

I*mporting data with RStudio: https://support.rstudio.com/hc/en-us/articles/218611977-Importing-Data-with-RStudio*

Issue 1*: In order to install the latest R… you have to update your OS. I don't want to do that, just yet so I'm working with 3.30 in RStudio and at Terminal and using repl.it instead.*

Resolution 1*: New Mac is MacBook Pro 2012 running 10.12.6 (Sierra). R installed but I get the following message upon launching R:*

Wednesday, June 24, 2020: *No longer get that message from Anaconda terminal! Running Catalina 10.15.5 Xtools and SVN commands*

- *SVN command line on Mac OS X:*
- *SVN and Mac OS X:*

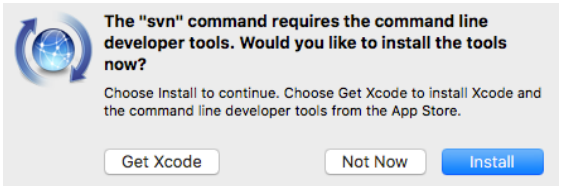
  12 subversion apps for OS X: https://gigaom.com/2009/02/23/12-subversion-apps-for-os-x/#:~:text=Mac%2DOnly%20SVN%20Clients&text=If%20you%20have%20a%20need,well%20as%20CVS%20and%20Perforce).

  SVN and Mac OS X: https://www.google.com/search?rlz=1C5CHFA_enUS907US907&sxsrf=ALeKk03IcRBMoa1i0YgFvZkJSZJkhbhmtA%3A1598067834977&ei=epRAX8ihO9DU-gSjvJjwDQ&q=mac+os+x+and+svn&oq=mac+os+x+and+svn&gs_lcp=CgZwc3ktYWIQAzIGCAAQFhAeMgYIABAWEB4yBggAEBYQHjIGCAAQFhAeMgYIABAWEB4yBggAEBYQHjoECAAQRzoECCMQJzoGCCMQJxATOgQIABBDOgIILjoCCAA6BwgAEBQQhwI6BwgjEOoCECc6BwguEOoCECc6CQgjEOoCECcQEzoECC4QQzoECC4QJzoFCAAQywE6BQguEMsBUNjoAViukQJgn5MCaAJwA3gAgAGWAogBwA6SAQYxMy40LjGYAQCgAQGqAQdnd3Mtd2l6sAEKwAEB&sclient=psy-ab&ved=0ahUKEwjIxoXN8q3rAhVQqp4KHSMeBt4Q4dUDCAw&uact=5

  The svn command requires….: https://www.google.com/search?q=the+svn+command+requires&rlz=1C5CHFA_enUS907US907&oq=the+svn+command+requires&aqs=chrome..69i57j69i64.5390j0j7&sourceid=chrome&ie=UTF-8

  https://stackoverflow.com/questions/60869347/mac-command-line-tools-11-4-no-longer-has-svn

- *Apple XTools: https://www.google.com/search?sxsrf=ALeKk01ZYE2cEsps3i0dlSCklLPKGtLSAg:1593031198856&q=apple+%22xtools%22&sa=X&ved=2ahUKEwi27-TVp5vqAhV0HjQIHQmpCB8Q5t4CMAF6BAgCEAs&biw=1280&bih=630*
- *Apple Developer Tools: https://en.wikipedia.org/wiki/Apple_Developer_Tools*



Stopped here
Friday, August 21, 2020

Issue 2.
Something I was doing in Anaconda python suggested not to do either. I have to go back to the documentation to find out which.
Other question is when you upgrade R do you have to delete the previous or is that handled by brew or other software management?
Issue 2_Update:
I get this message, now only with command open -a rstudio*, from the terminal or from desktop search launch of rstudio. Haven't tested with Anaconda navigator, yet. Don't get this message otherwise, do to re-install of R and automatic changes to bash profile (not sure which one) However, here is direct search for solution:*

https://www.google.com/search?q=the+svn+requires+the+command+line+developer+tools+would+you+like+to+install+them&oq=the+svn+requires+the+command+line+developer+tools+would+you+like+to+install+them&aqs=chrome..69i57j69i64.32583j0j7&sourceid=chrome&ie=UTF-8

Obverse: "stub perspective':

https://www.google.com/search?sxsrf=ACYBGNSnRjq6qrBdBSc5RzUooCU_IGviAQ%3A1580167583958&ei=n3EvXu2EOvi-0PEPy4aLyAk&q=+stub+%2B+the+svn+requires+the+command+line+developer+tools+would+you+like+to+install+them&oq=+stub+%2B+the+svn+requires+the+command+line+developer+tools+would+you+like+to+install+them&gs_l=psy-ab.3...1617.3095..3336...0.2..1.534.2578.2-2j3j1j1.....0....1..gws-wiz.....0i71.OP8gpM-qAX4&ved=0ahUKEwit0Jv89qTnAhV4HzQIHUvDApkQ4dUDCAs&uact=5

Note:
I downloaded Studio before installing R so when I ran RStudio, It couldn't find it! Then I installed Desktop: R-Studio Desktop, which I assume is the same for RStudio. The difference between RStudio Desktop vs RStudio Server is described in this wikipedia page: https://en.wikipedia.org/wiki/RStudio .

…and:

https://support.rstudio.com/hc/en-us/articles/217799198-What-is-the-difference-between-RStudio-Desktop-and-RStudio-Server-

He (Rafa) just says RStudio, so I assume It's the desktop version. Details indicate you have to compile RStudio Server for your particular platform but it's targeted toward linux and web.

RStudio Server:
https://www.google.com/search?q=rstudio+server&oq=rstudio+server&aqs=chrome..69i57j69i60.3328j0j7&sourceid=chrome&ie=UTF-8

- https://rstudio.com/products/rstudio/download-server/

Accessing RStudio Server (Open Source): https://support.rstudio.com/hc/en-us/articles/200552306-Getting-Started

range of unix/linux ids: https://www.google.com/search?q=range+of+unix+or+linux+user+ids&oq=range+of+unix+or+linux+user+ids&aqs=chrome..69i57.6404j0j7&sourceid=chrome&ie=UTF-8

User identifier: https://en.wikipedia.org/wiki/User_identifier

Most of this lecture, I already know. However. exploring further, the commands

- *install.packages("rafalib") # Note the autocompletion!*
- *library(rafalib) # Note: NO QUOTES NEEDED!*

…understand, despite your installing a certain package. You have to load it by typing library() *and the package within the parentheses!*

> library(swirl)

5. Author
October 11, 2019 at 20:36:43
Package vs library:
> library(jones)
Error in library(jones) : there is no package called ‘jones’
>

6. Author
January 19, 2020 at 19:32:42
Wonder how you’d do this from the command line?

| Hi! Type swirl() when you are ready to begin.

This is the feedback I got from typing the first command:

> install.packages("rafalib")
trying URL 'https://cran.rstudio.com/bin/macosx/mavericks/contrib/3.3/rafalib_1.0.0.tgz'
Content type 'application/x-gzip' length 40984 bytes (40 KB)
==
downloaded 40 KB

The downloaded binary packages are in

/var/folders/v1/vsprsysx0j14p84dkgml9xnw0000gn/T//RtmpYZAw3T/downloaded_packages
>
>
>
> library(rafalib)
> # Note: THERE is no confirmation after the the prompt; unlike what I thought I saw on the video. Is there some way to verify this or what libraries are loaded? Note: I'm sure somewhere in the RStudio consoles it shows you… but how would you do this from the command line?
> library(jones)
Error in library(jones) : there is no package called ‘jones’ # There is no package by that name; just testing.
5 *Note it calls it a package, not a library!*

 # Note: I’ve got a snapshot of the console. Notably different from the professors’ Get familiar with the console!
Stanford University Library and R programming events*:* *https://library.stanford.edu/projects/r*
The ‘Gear-Up’ Website: https://library.stanford.edu/projects/gear-research

- *He refers to RStudio as an ‘interface software’*

Issue 3. 2nd attempt
Some days later. Couldn’t load rafalib. Said it wasn’t there So did procedure again: Output below:
The downloaded binary packages are in
/var/folders/c3/zz14c4n57v53kvhr5sphl6lw0000gn/T//RtmpZeDJV6/downloaded_packages

Do I have to do this every time? Though packages were stored an just a matter of loading? is that “tmp” designation a clue as to why?
Issue 3. Resolution*: No! you typed load(rafalib) instead of library(rafalib)!*

OK continuing with the script from the websight

Running Lab Code

All software used for the class is free and open source:
R can be downloaded and installed from CRAN External link
(Comprehensive R Archive Network). If possible download the latest release.
We recommend using a slick visual interface for R.

Trans Script*:*
RAFAEL IRIZARRY: In this video, we're going to give a brief description of how to get started with the R programming language.
First thing you need to do is install R. To do that, go to your favorite browser, do an internet search for R, or for download R, and you should arrive pretty quickly to the Comprehensive R Archive Network. You will then pick the appropriate link here, and then a pretty self-explanatory process begins that takes you to installing R on your
6 computer.

7. Author
October 8, 2019 at 21:18:59
Is that what this is/these are, on the mac? Is that why they're located separately?

See book Mac OS X for Unix Geeks!

7

Once R is installed, we will install an *interface software* called RStudio. You can find RStudio similarly. You do an internet search on a browser, and you should arrive to their download page pretty quickly. Then you select a link that has your platform, that installs RStudio on your computer.

So in this course, we will be using R Studio to interact with R. You don't need to do that, you can run R directly. But we like to use RStudio, and we use it throughout the course. So that's what we recommend you do.

…but it's probably needed that you understand the straight from the terminal format, as well!

What's also needed is understanding the use of Anaconda vs using Mac terminal and how that's changed since installing Catalina!

Suggest Mac OS X for Unix Geeks!

So once you have installed RStudio, going forward, most videos you will be running RStudio as a first step. That's where we will be writing our code, running our code, analyzing data.

So once you start RStudio--and every system is different, but you just get that program started--it should look something like this. You can see this window here has R running in it. There's R Version 3.2.0. We can actually execute commands in the command line, and that's where R is running.

Correlate the terminal version vs RStudio in terms of what info is immediately available.

Can you Docker R and RStudio?

Now in this, throughout the course--and it's something that we highly recommend--if you're doing data analysis, we don't type commands directly in the R window. We open up a file--so here I'm going to File, New File, R Script—and you save commands in that file for later. So you can save that file--let's give it a name, let's call it code.R--and then you can save that file, *and look at it later*.

…but how interactive is this?

So what we're going to do today is describe how to install packages. This is one of the key first steps in the R programming language. R comes with a fixed, small number of functions of not too much functionality, but it's very easy to add to it. And the reason we do that is because there are so many applications of statistics and data analysis that instead of having it all in one place, we instead have different packages for doing different tasks. So today, we're going to show you how to do that.

So one of the packages we're going to be using throughout the course is the rafalib package, that was created specifically for this class. It has several functions that we use for several different things in the course. You will see that as you go along in the videos. And then today, I'm going to show you how to install it.

So there's a very nice command in R called install.packages. And one of the nice things about RStudio is that it tries to read your mind, and it does it fairly well. Notice that as soon as I type insta, it autocompleted the package that I wanted. So once you see it in front of you like this, you just hit the Tab key, and then it *autocompletes.* I don't actually have to type the whole thing. And I'm going to install the rafalib package. So what this is going to do, what this function does is it goes to CRAN, it finds the rafalib package, downloads it to your computer, and installs it. So there it goes.

You can see it's thinking, and it will download it there, download it, and installed it. Once it's installed, then I can load it into R. So it's installed on our computer. To get it into R, we type library, and the name of the package. Again, RStudio is autocompleting for us. Do that, and now I have several functions included here which I can run that are included in the rafalib package. And we'll see many of those throughout the course. OK, so that's how you install packages.

Note 3.

```
> library(rafalib)  # You don't type any quotes!
```

What is a library in R?: https://www.google.com/search?q=what+is+a+library+in+r&oq=what+is+a+library+in+r&aqs=chrome..69i57.3799j0j7&sourceid=chrome&ie=UTF-8

Packages vs Library: https://www.google.com/search?q=r+programming+%2B+packages+vs+library&oq=r+programming+%2B+packages+vs+library&aqs=chrome..69i57j69i64l3.7973j0j7&sourceid=chrome&ie=UTF-8

How to browse packages or libraries in R (What is a package/library?): https://www.google.com/search?sxsrf=ACYBGNSuTZb2G_JpoQSj8ZXb6WT3DXGT_A%3A1579407014831&ei=ptYjXp6pMof5-gSLu72gCA&q=how+to+browse+packages+or+libraries+in+r&oq=how+to+browse+packages+or+libraries+in+r&gs_l=psy-ab.3...67626.81003..81197...4.4..0.115.3665.37j7.....0....1..gws-

wiz.....0i71j35i39j0j35i39i19j0i67j0i203j0i22i30j33i22i29i30j33i21j33i160.Odqri3XwJ80&ved=0ahUKEwielMzQ5Y7nAhWHvJ4KHYtdD4QQ4dUDCAs&uact=5

Find installed packages: https://stat.ethz.ch/R-manual/R-devel/library/utils/html/installed.packages.html

So how do you find packages? Well, there are several ways to do that. And one way is to use internet searches. Another way is to take a course like this, where you learn the names of the key packages that are needed for different types of data analysis. So your first homework in this course is to learn the R syntax, if you don't know it already. This could take a little bit of time. One way we recommend you do that is to use the swirl package, which teaches you R inside R. And to do that, to install that,we would type install.packages("swirl"). Type S-W-I-R-L. And now that should install it. And now if I load up that package-- there, it's autocomplete-- you can see that it gives you a message, and it gives you an instruction on what to do first.

Note 4.

How to install packages in R (seems to favor the Gui)
https://www.google.com/search?q=how+to+install+packages+in+r&oq=how+to+install+packages+in+r&aqs=chrome..69i57j69i64.5447j0j7&sourceid=chrome&ie=UTF-8

Command to install packages in R:
https://www.google.com/search?sxsrf=ACYBGNQpFc0eYyLOTGLbcOXM8Pw52cLXDg%3A1579402502697&ei=BsUjXuGVKsnY-wTG96iICQ&q=command++to+install+packages+in+r&oq=command++to+install+packages+in+r&gs_l=psy-ab.3..0i203j0i8i30l5.8425.10925..11624...2.3..0.79.887.12.....0....1..gws-wiz.....0i71j0i8i7i30j0i7i30j0i7i30i19j0i8i7i30i19.gJ9LWmIRte4&ved=0ahUKEwjhsIXp1I7nAhVJ7J4KHcY7CpEQ4dUDCAs&uact=5
https://www.r-bloggers.com/installing-r-packages/

OK, so one important aspect of RStudio that you should know right away is that the way that you get these lines of code to run in the window below is there is a keystroke that lets you do that. And it's different for the different platforms. On a Mac, it's Command+Return. But you definitely want to find out what that is early on, so that you don't actually have to copy and paste it, you just— once you are on that line, if I hit Command+Return, it automatically runs that command.

Funny! Mac OS X...., just hit the return key. In Jupyter, you do all that shit!

So notice that when I loaded the swirl library, it gave me these instructions, says, type swirl, and we'll be ready to begin. So we can actually type that into R, command directly in the R command line directly this time, to get swirl going. So you're going to see, it starts asking you a question, welcome to swirl, et cetera, et cetera, what should I call you? Type your name, hit Enter, and you will get started with the R tutorial.

Note 5.: Actual Output. Performed at RStudio command line, on MacBook Pro 2012.

```
> install.packages('rafalib') # Single or double quotes are inconsequential!
also installing the dependency 'RColorBrewer'

trying URL 'https://cran.rstudio.com/bin/macosx/el-capitan/contrib/3.6/RColorBrewer_1.1-2.tgz'
Content type 'application/x-gzip' length 53161 bytes (51 KB)
==================================================
downloaded 51 KB

trying URL 'https://cran.rstudio.com/bin/macosx/el-capitan/contrib/3.6/rafalib_1.0.0.tgz'
Content type 'application/x-gzip' length 56180 bytes (54 KB)
==================================================
downloaded 54 KB


The downloaded binary packages are in
        /var/folders/c3/zz14c4n57v53kvhr5sphl6lw0000gn/T//RtmpmCwfzZ/downloaded_packages
————End Installation Transcript————————————————————————————————————————-
```

stopped here Monday, January 27, 2020

FIRST ASSESSMENT:EXERCISES

To download RStudio and R, go to: https://courses.edx.org/courses/course-v1:HarvardX+PH525.1x+3T2019/courseware/dcf8031210054672a6bd2a63d6f9d9ac/599b44cd22814a4795fda31f02c3719f/1?activate_block_id=block-v1%3AHarvardX%2BPH525.1x%2B3T2019%2Btype%40vertical%2Bblock%4031836d1b280e49c3b22ae8abd9fee332

If you have not done so already, download, install and load the swirl package.

```
install.packages("swirl")
library(swirl)# But, how to see what packages are already installed?:
```

Actual console interaction below:

Assignment:

Go through the R Programming Basic Building Blocks tutorial and then use the skills you have just learned to answer the following questions.

Issue 4.

...at this point I'm spending more time on the Google searches to the answers and testing than the swirl. It's boring & underscore doesn't function on this Japanese MacBook Pro (or I've yet to find the keystroke!). That I'll do later. Then get back to the professor's teachings.

Interesting observations re: swirl:

```
Do you want to install from sources the packages which need compilation?
y/n: n
```

When we learn how to compile sources, we'll pursue this!!!

```
 Hi! I see that you have some variables saved in your workspace. To keep
| things running smoothly, I recommend you clean up before starting swirl.

| Type ls() to see a list of the variables in your workspace. Then, type
| rm(list=ls()) to clear your workspace.
```

What's odd about this is the next day, upon starting swirl via RStudio not the terminal, I got the same message. I tried deleting them at the terminal but it says they're not there! Is RStudio senile? ..seeing something that got restored via the iDrive? ..hmmm!

Anyhow, I've now assembled and removed those file in RSTudio, as well!

```
 > rm(list=ls())
 >
 > ls()
 character(0)
 >
| Type swirl() when you are ready to begin.
| You can exit swirl and return to the R prompt (>) at any time by
pressing the
| Esc key. If you are already at the prompt, type bye() to exit and save
your
| progress. When you exit properly, you'll see a short message letting you
know you've done so.

| When you are at the R prompt (>):
| -- Typing skip() allows you to skip the current question.
```

```
| -- Typing play() lets you experiment with R on your own; swirl will
ignore
| what you do...
| -- UNTIL you type nxt() which will regain swirl's attention.
| -- Typing bye() causes swirl to exit. Your progress will be saved.
| -- Typing main() returns you to swirl's main menu.
| -- Typing info() displays these options again.

| Let's get started!

 when you see '...',
| that means you should press Enter when you are done reading and ready to
| continue.

...  <-- That's your cue to press Enter to continue
```

This is what happened several days later when I restarted:

Note: I've logged back in . Do I have to continually install.package("swirl") or just load it, from now on with the library(swirl) command?

Answer: just the library(swirl) command

```
> install.packages("swirl")
--- Please select a CRAN mirror for use in this session ---

  There is a binary version available but the source version is later:
      binary source needs_compilation
swirl  2.4.3  2.4.4             FALSE

installing the source package 'swirl'

trying URL 'https://cran.cnr.berkeley.edu/src/contrib/swirl_2.4.4.tar.gz'
Content type 'application/x-gzip' length 108533 bytes (105 KB)
==================================================
downloaded 105 KB

* installing *source* package 'swirl' ...
** package 'swirl' successfully unpacked and MD5 sums checked
** R
** inst
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** testing if installed package can be loaded
* DONE (swirl)

The downloaded source packages are in
        '/private/var/folders/v1/vsprsysx0j14p84dkgml9xnw0000gn/T/
RtmpaRK3c0/downloaded_packages'
```

First Assessment: Actual Exercises (x)

As I said I'm googling answers and trying to get a composite of what will be needed to know, then doing the swirl. Most of what follows is for Palimpset!

1st question:

I can't use the most recent version w/o upgrading my OS and I've old HW
Options are:

- repl.it
- Cloud
  (*not sure how to do this, yet!*) I would think the best move would be to create an instance and install the proper version of R. This may seem extensive, but is actually part of another skill-set I need to employ on a regular basis.

2nd…Create a numeric vector containing the numbers 2.23, 3.45, 1.87, 2.11, 7.33, 18.34, 19.23. What is the average of these numbers?

How to create a vector in R: https://www.google.com/search?q=creating+a+vector+in+r&oq=creating+a+vector+in+r&aqs=chrome..69i57.6792j0j7&sourceid=chrome&ie=UTF-8

What I used was the following:

```
> x = c(2.23,3.45,1.87,2.11,7.33,18.34,19.23)
> length(x)
[1] 7
> x/length(x)
[1] 0.3185714 0.4928571 0.2671429 0.3014286 1.0471429 2.6200000 2.7471429
> x.sum
Error: object 'x.sum' not found
> ?sum()
> sum(x)
[1] 54.56
> sum(x)/length(x)
[1] 7.794286

 # Now the easier way!

> x <- c(2.23,3.45,1.87,2.11,7.33,18.34,19.23)
> sum(x)/length(x)
[1] 7.794286
```

What is a vector?

R programming + how to create a vector: https://www.google.com/search?q=r+programming+%2B+how+to+create+a+vector&oq=r+programming+%2B+how+to+create+a+vector&aqs=chrome..69i57j69i64l3.6479j0j7&sourceid=chrome&ie=UTF-8

Vectors in R: http://www.r-tutor.com/r-introduction/vector **STOPPED HERE! CREATE PALIMPSEST**

Questions this link (above answers):
What is a vector in R?
c.f. enumeration
Ways of creating them
c() function
coercion?
The : operator
~~Note unlike in python indexes, we can cross zero!~~ See correction-below!

```
> x <-  -2:4
> x
[1] -2 -1  0  1  2  3  4
```

seq() function, for more complex sequences

Correction: *We can dictate ranges that cross zero in python and R. It's within slices that we're limited (in python, anyway). To dictate slices in python you must proceed in the positive direction, when using the x coordinate axis as a model. But you can only go so far as the list has or zero.*

```
>>> x
[4, 12, 66, -7, 2]
>>> x[-5:-1]
[4, 12, 66, -7]
>>> x[-5:-0]
[]
>>> x[0:3]
[4, 12, 66]
>>> x
[4, 12, 66, -7, 2]
>>> x[3]
-7
>>> x[-5:1]
[4]
>>> x
[4, 12, 66, -7, 2]
>>> x
[4, 12, 66, -7, 2]
>>> x[-3:3]
[66]
>>> x[-3:5]
[66, -7, 2]
```

Accessing elements within a vector

Using integer vector as index

```
> x <- -2:4
> x
[1] -2 -1  0  1  2  3  4
> x[1]
[1] -2
> x[7]
[1] 4
> x[8]
[1] NA
```

My experimentations:

```
> x = 2:20
> x
 [1]  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
> x = seq(2:20)
> x
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
> x = seq(2:20, by=0.2)
Error in seq.default(2:20, by = 0.2) : 'from' must be of length 1
> x = seq(1:20, by=0.2)
Error in seq.default(1:20, by = 0.2) : 'from' must be of length 1
> x = seq(1:20, by=0.2)
Error in seq.default(1:20, by = 0.2) : 'from' must be of length 1
> seq(2:22, by=3_
Error: unexpected input in "seq(2:22, by=3_"
> seq(2:22, by=3)
```

```
Error in seq.default(2:22, by = 3) : 'from' must be of length 1
> seq(1:22, by=3)
Error in seq.default(1:22, by = 3) : 'from' must be of length 1
> seq(2,:22, by=3)
Error: unexpected ':' in "seq(2,:"
> seq(2,22, by=3)
[1]  2  5  8 11 14 17 20
```

Using logicals

3rd…

My answer was to do the following but I'm still shy of understanding Loops granted they're not a preferred method in R.

```
Vectorized answer
x <-  1:25
sum(x*x)
```

Loops in R programming: https://www.google.com/search?q=r+programming+%2B+for+loops&oq=r+programming+%2B+for+loops&aqs=chrome..69i57j69i64l3.3845j0j7&sourceid=chrome&ie=UTF-8

A Tutorial on loops in R Usage and Alternatives:

Vectorized alternatives of Loops: https://www.google.com/search?q=vectorized+alternatives+of+loops&oq=vectorized+alternatives+of+loops&aqs=chrome..69i57.6055j0j7&sourceid=chrome&ie=UTF-8

Exponentiation:

No, not immediately related to the material but I thought ^ could be used… That may be in Wolfram alpha

Meanwhile this is an interesting note contained in these posts.

Exponentiation in python: https://www.google.com/search?q=exponentiation+in+python&oq=exponentiation+in+python&aqs=chrome..69i57.4895j0j7&sourceid=chrome&ie=UTF-8

How do I perform exponentiation in python: https://stackoverflow.com/questions/30148740/how-do-i-do-exponentiation-in-python/30149434

Fonction OU exclusif: https://www.google.com/search?q=what+is+xor&oq=what+is+xor&aqs=chrome..69i57.3503j0j7&sourceid=chrome&ie=UTF-8

fonction OU exclusif: https://fr.wikipedia.org/wiki/Fonction_OU_exclusif