Proposed course Curriculum: ECON 314 FRB+Howard

While in the past we have not used a text book, beginning in Spring 2018 I suggest we introduce R for Data Science by Hadley Wicham and Garret Grolemund as the textbook. <http://r4ds.had.co.nz/index.html>. This book is freely available online and provides an excellent overview and backbone for the course we are teaching.

For the first 4 weeks of the course we will not use the textbook and will instead focus on familiarizing the students with Base R. After week four we will focus on teaching the students packages in the Tidyverse and transition to using the textbook as a resource.

Learning Objectives

Week 1:

* Differentiate between console and R script
* Save, close, open, and re-run R script files
* Basic arithmetic in R
* Comments in R
* Data types: identify and convert (character and numeric)
* Vectors
* Matrices
* How to find help
* Access and clone the Github
* Submit homework 0.5 in class

Week 2:

* Understand how to comment and format code
* Understand basic logical operators in R
* Understand R data frames
  + How do data frames relate to vectors and matrices
  + How values are indexed and referenced
  + Basic summary functions
* Understand operations on data frame
  + Creating new columns and rows
  + Subsetting/filtering

Week 3:

* Understand R packages, how to install and how to load
* How to find appropriate packages for data ingestion
* Read and write .csv and .txt files
  + stringsAsFactors
* Merging data in base R
  + Merge
  + rbind
  + cbind
* Sequences – seq() and rep()
* How to create an Rstudio project and a Github Account
* Understand good project file organization
  + The importance of good file/variable names

Week 4:

* Understanding control flow (if/else, while, for loops)
* How to write a function

Now that students have a basic understanding and familiarity with R we will turn to the textbook more closely:

Week 5:

* Understand what a tibble is (Tibble)
* Reading in data as tibbles (Readr)
* Manipulating data using Dplyr (Dplyr)
  + Select, filter, arrange, mutate
* Readr (chapter 11)
* Dplyr (chapter 5)
* Tibbles (chapter 12)

Week 6: Dplyr continued – piping, joining, data cleaning

* Financial Literacy Lecture (1 hour)
* Compute summary statistics for data subset using Dplyr
  + Group\_by and summarise
* Simplify code using pipes
* Joining data with Dplyr
  + Left\_join, right\_join, etc
* Understand the difference and uses of long vs. wide data and how to convert between the two using the Reshape2 package
* Piping (chapter 18)
* Relational Data (chapter 13)
* Reshape2

Week 7: Data Cleaning

* Understand how to process data
  + Changing formats
    - From character to numeric, vice versa
    - From factor to character etc
  + Extracting useful fragments of strings
  + Splitting strings
  + Converting upper and lower case
* Stringr (chapter 14)
* Understand how to reformat different types of character strings into R date values
  + How to extract month or year values from a date
  + How to convert series frequency
    - Round date to nearest “x” and average value by “x”
  + Create sequences of dates of varying frequencies
  + How to construct lag or leading variables
* Lubridate (chapter 16)
* Understand what a factor is, how to use them, when to use them, and when to avoid them
  + Binning variables using the cut() function
* Factors (chapter 15)

Week 8: Visualization 1

* Understand what makes an informative graphic
  + How to avoid pitfalls of creating false/misleading graphics
* How to create a ggplot object
  + Understand the ggplot underlying grammar of mapping data to aesthetics via the aes() function
  + How to layer ggplot objects
    - Creating basic line, scatter, and bar charts
* GGplot2 part 1 (chapter 3)

Week 9: Visualization 2

* Understand how to customize a ggplot object
  + How to set custom scales
    - Changing the colors and line types and axis labels
  + Customizing the legend
    - Combining legends (when scaling on the same data)
  + Controlling the plot’s color scheme via the theme() functions
* GGplot2 part 2 (chapter 28)

Week 10: midterm presentations

* Midterms include data cleaning, summary stats

Week 11: Regression Analysis

* Understand the difference between economic modeling and data science broadly
* Understand when it is appropriate to use regression analysis
* Understand how to run basic regressions in R
  + Ordinary Least Squares
  + Logit
* Understand basic error analysis
* Learn how to use the Broom package to facilitate working with regression results
* Understand how to plot regression results using ggplot
  + Lines of best fit and error bars

Week 12: Communication – R Markdown

* Financial Literacy lecture (1 hour)
* Understanding when it is appropriate to use Rmarkdown
* How to create documents with inline code, graphics, and tables
  + Understanding the built-in Latex math mode usage
* How to communicate R code effectively
* R Markdown (chapter 27)
* Graphics for Communication (chapter 28)
* Additional Background (chapters 29/30)

Week 13: Final Presentations

Detailed Course Schedule

Lecture 1 – Introduction

* Hand out materials
* Downloading R and Rstudio
* Console, editor, help windows of Rstudio
* How to access the github, downloading from github
* Functions:
  + Different types: character, numeric, logical
    - integer vs double
  + Paste(), paste0()
  + c()
  + vectors
  + <-
    - How to name your variables correctly
  + Basic mathematics (+, -, \*, /, \*\*)
    - Vector math – adding two vectors etc
    - Vector recycling
  + sum(), length(), diff(), mean(), range(), max(), min()
  + Matrix() – a collection of vectors of the same type and length
  + Help(), ?
* Assign Homework 0.5
  + Overview of homework template file
  + How to save Rscript files
  + Closing a script file, reopening it and running it
  + How to submit homework
* Assign Homework 1

Lecture 2 – Intro to data.frames

* Homework 1 due
* Review homework 1 submissions, (formatting, commenting, sourcing etc)
* Hotkeys for Rstudio to insert comments and for Rstudio to automatically format your code
  + Ctrl-shift-c
  + Ctrl-shift-A
* Indexing vectors in R with [] and [[]]
* Overview of data.frames
  + What is a data.frame? – a collection of equal length vectors of the same or different types
  + Data.frame(), as.data.frame()
  + Subsetting with $
  + Creating new columns with $
  + Subsetting rows using []
  + Subsetting columns using []
  + Mean(), min(), max(), range(), sum(), sd() – data.frame context
    - Mean(Data.frame$column)
    - Mean(data.frame[, c(2,3,4)])
* Logical Operators
  + %in%, &, |
* StackOverflow, R-bloggers, google, package docs, etc…
* Homework 2 assigned

Lecture 3: File Input/Output, R libraries, Merging

* Homework 2 due
* Intro to the final project (15 minutes)
* Read.csv()
  + stringsAsFactors = F
* write.csv()
* Install.packages()
* Update.packages()
* Library()
* Readr package
  + Read\_csv() (aka read\_delim())
* merge()
* rbind(), cbind()
* Sequences: seq(), `:` operator, rep(), pretty()
* Homework 3 assigned

Lecture 4: Data Cleaning

* homework 3 due
* Working with Dates
  + as.Date()
  + format()
  + lubridate
    - floor\_date(), ceiling\_date(), ymd(), dmy(), mdy()
* working with strings – stringr package
  + str\_replace(), str\_sort(), str\_split(), str\_length(), str\_match(), str\_join()
  + str\_sub(), str\_subset(), str\_to\_lower(), str\_to\_upper()
* factors
  + what are factors
  + why are factors useful, why are they not useful, how to create/avoid them
    - stringsAsFactors = FALSE
* Constructing lag or lead variables
  + Diff()
* homework 4 assigned

Lecture 5: Control flow and basic function writing

* Homework 4 due
* Control flow (if/else/else if)
* For loops
  + Indexing using vectors, using seq, using colon, using names(), nrow(), etc
* While loops
  + Infinite loops and how to avoid/break out of them
* Nested loops
* Functions in R
  + What are arguments? Preset vs. non-preset arguments
  + How to write a function in R
  + Return()
* Homework 5 assigned

Lecture 6 – Tibbles and Dplyr

* Homework 5 due
* Tibbles (10 minutes)
  + What is tidy data
    - How to organize your data frames
  + How is a tbl different from a data.frame
    - As\_tibble()
    - Tbl\_df()
* Dplyr (2.5 hours)
  + Select(), mutate(), filter(), summarise(), arrange()
  + %>% operator, group\_by()
* Homework 6 assigned
* Take home test assigned, due Monday at 11:59 pm on piazza
  + Students who score C or lower should be encouraged to drop the course

Lecture 7: Financial Literacy & Merging/reshaping Data

* Homework 6 due
* Financial literacy lecture 1 (1 hour)
* Data reshaping – Reshape2 - (0.5 hours)
  + Melt(), cast()
* Joins
  + Left\_join(), full\_join(), right\_join(), anti\_join()
* Dplyr review (1 hour)
  + Review of using the %>% to string together chains of commands
* homework 7 assigned

Lecture 8 Visualization with ggplot2 Part 1

* homework 7 due
* Good bad and ugly plots (1 hour)
  + “how to not mislead with statistics”
    - Putting two plots side by side
    - Time series data
  + How to make a chart easy to read
    - Consistent formatting
* Ggplot part 1:
  + What is a ggplot?
    - Ggplot()
  + Building plots up by layers
    - ‘+’ operator
  + Different geometries
    - Geom\_point(), geom\_line(), geom\_bar(), geom\_hist()
  + Letting ggplot automate good habits:
    - Facet\_wrap()
* Homework 8 assigned

Lecture 9: Visualization with ggplot2 Part 2

* Scales
  + Continuous vs. discrete scales
  + Scale\_x\_, scale\_y\_, scale\_color\_, scale\_fill\_, scale\_linetype\_, scale\_size\_
* Titles, labels, and guides
  + Ggittle(), labs()
* Plotting from multiple datasets
  + Data = argument of geom calls
* Controlling the theme
  + Theme()
* How make good plots:
  + How to show data changing over time
    - Percent growth, chained data, rolling means in zoo::
* Making maps in ggplot
* No homework assigned – students should be working on homework 8
  + Students will be working on midterm presentation materials

Lecture 10: Midterm Presentations

* Presentations (3 hours)

Lecture 11: Financial Literacy & Regression Analysis

* Financial Literacy lecture 2 (1 hour)
* Midterm resubmissions assigned (50% of corrected points added to score)
* Homework 7 due
* The lm() function
* Broom package
  + Tidy(), augment(), glance()
* Graphing/displaying outputs from models
* Homework assigned

Lecture 12: Communication with R markdown

* Financial Literacy Lecture (3)
* Midterm resubmissions due
* Opening an Rmarkdown document
* Text formatting
  + Bold, underline, italics, superscript, subscript, strikethrough
  + Inserting hyperlinks and pictures
  + Math mode
* Code blocks
  + Inserting code chunks, knitr code chunk options
  + Inserting graphs
  + Inserting tables
* making a document vs slides
  + pdf document
  + pdf beamer slides
* no homework assigned
  + students should now be spending their time working on final project materials

Lecture 13: Financial Literacy & overflow

* Financial Literacy Lecture (4)
* Overview of Final Project materials due
* Review of Regressions
* no homework assigned
  + students should now be spending their time working on final project materials

December 4: All presentation materials due

* slides, write up & code

Lecture 14: Final Project Presentations part 1

December 15 (?): Final Project Presentations part 2

I am assuming that the grading/structure for the course looks highly similar to the current semester. (i.e. semester-long final project with the check-in materials due at different check points – I did not include all the different check points in this writeup).

I agree with Jeff Naber that the final project has taken on too much importance in the course, it is currently over 50% of the total grade for the course and I think that we should scale it back so that all-in the final project sits somewhere around 30% of the overall grade. This gives us the ability to do any of the following:

* assign a second take home test, or a take home week-long project
  + The data and assignment that Andrew sent around for his lecture could easily be made to fit this idea
  + We could assign this November 17
* Assign the textbook readings as homework
  + Each chapter in the book has lots of comprehension questions throughout, we could assign those questions as pre-work, due before the start of lecture on the given topic
  + If we did this I am imaging homework jumping up to 40% of the overall grade
    - 25% for the homework assignments we create/assign
    - 15% for the textbook questions
  + The pro of this is that we would essentially force students to keep up on the readings
  + Con is that we would be asking for a big time investment from students on the homework since the homework load would essentially double
* Assign roughly the same amount of homework as spring 2017 but weight the homework more in the overall grade.

Sadly we no longer teach data.table :-(

Additionally, for the final project: going forward we must stress that personal github repositories be central to the project itself. Putting all data and code in a public centralized area for the project has many advantages:

* Students learn how to use github
  + Rstudio projects can be tied in with github repositories
* Students are able to see each other’s projects for help/inspiration
* Instructors are able to easily reproduce student work in order to pinpoint errors
* Projects become much more reproducible, a key idea we are looking to impart on students
* By making the projects available in public repositories students can show (instead of just tell) admissions committees and employers their work.