R Packages to Know and Love

Data Wrangling

When dealing with large data sets, changing the shape and format of the data can be cumbersome.

- **Dplyr** - Wickedly fast (relative to base R) tool for data manipulation. Successor to the popular plyr package
  https://cran.rstudio.com/web/packages/dplyr/vignettes/introduction.html
- **Tidyr** - Tool for data cleaning and reshaping. Successor to the reshape* packages
  http://blog.rstudio.org/2014/07/22/introducing-tidyr/
- **Readxl** - Read excel files gracefully, no more exporting individual sheets to csv.
  https://github.com/hadley/readxl/blob/master/README.md

Code Clarity

Not everyone uses them, but much new R code makes use of pipes "%>%" . Elementary piping can be done with dplyr, however there are more elaborate uses of them.

- **Magrittr** - Piping tools, named for Renee Magrittr and his infamous "The Treachery of Images"
  https://github.com/smbache/magrittr/blob/master/README.md

Note-Keeping and Reproducibility

A central topic in Science is the reproducibility of research. This is not limited to wet-lab/field experiments but code and analyses as well. To keep you code organized and easily presentable try:

- **Rmarkdown** - R specific markdown dialect, very easy, and reasonably powerful.
  http://rmarkdown.rstudio.com/
  http://cfhammill.github.io/posts/startRMarkdown.html (shameless personal plug)
- **Knitr** - Dynamic document generation for R. Essentially an R implementation of Donald Knuth's Literate Programming.
  http://yihui.name/knitr/

Interactivity and Appification

Historically R was really bad at making interactive programs with GUIs. That's changed recently

- **Shiny** - Interactive javascript apps for R. Write apps and analyses in native R code and all R to create a pretty web-app for you.
  http://shiny.rstudio.com/
- **DT** - Create sortable interactive tables on the fly. R wrapping for the DataTable jQuery plugin can save a trip to your spreadsheet.
  http://blog.rstudio.org/2015/06/24/dt-an-r-interface-to-the-datatables-library/

Machine Learning

R is reasonably good on the machine learning front, but gets less love than Scikit-Learn, Torch, etc. these days.

- **Caret** - Aims to be a unified interface for hyper-parameter tuning, classification, and predictive modelling, can substantially reduce startup time for using a new technique by providing consistent model specification. I'm uncertain when it's advantageous not to use Caret.
  http://topepo.github.io/caret/index.html
- **Kernlab** - A rich assortment of kernel based methods for machine learning. I can't find a good into page for this, so the official page will have to stand-in
  https://cran.r-project.org/web/packages/kernlab/vignettes/kernlab.pdf
- **RandomForest** - The canonical implementation of Breiman's RandomForest algorithms, ported from the original fortran. There are many updates/extensions to the algorithm but this one is a natural choice.
  https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm
- **Xgboost** - Do some extreme gradient boosting
  https://cran.r-project.org/web/packages/xgboost/index.html

Development

Writing short code bits is pretty straight-forward in R, but serious development is a different ball game

- **Devtools** - The essentially must have set of developer tools for R. Allows installation from github and much, much more.
  https://www.rstudio.com/products/rpackages/devtools/

- **Testthat** - Nice code testing framework to make sure your code does what you think it does
  https://github.com/hadley/testthat/blob/master/README.md

- **Argparse** - Graceful argument handling for your scripts. R port of python's argparse package
  https://github.com/trevorld/argparse/blob/master/README.rst