R Programming - Best Practice

Mei Eisenbach

Introduction

- Why this workshop
- What’s covered
  - Base R and RStudio
  - Workflow
  - Formatting style
  - Programming patterns
- What’s not covered
  - Tidyverse

Workflow

Workflow

General Tips:

- Think about workflow first
- Use directories
- Use the features in your IDE (RStudio)

RStudio Workspace

- Default behavior
- Why saving the workspace is bad
- The code is what's important
- “Clear Workspace” command: under the Session menu

RStudio Projects

- Have their own workspace
- Working directory is set
- Can be created from existing directories

Saving objects

- As an alternative to saving the workspace, save objects to files.
- saveRDS(), readRDS()
- Go to tutorial…

Github

- Version Control System
- RStudio works well with Github
- How to setup it up: https://www.r-bloggers.com/rstudio-and-github/

Style

“Good coding style is like correct punctuation: you can manage without it, butitsuremakesthingseasiertoread” - Hadley Wickham

Naming schemes

Snake Case

- Use underlines to separate words.
- E.g. snake_case
- Preferred by Hadley Wickham's Style Guide.

Naming schemes (cont’d)

Period Separated

- E.g. period.separated
- Has historical precedence in the R community.
- Preferred by Google's R Style Guide.
- Confusing for people coming from other programming languages where the the period is an operator (e.g. Java)

Naming schemes (cont’d)

Camel Case

- Use capital letters to separate words. Note, the first character is not capitalized.
- E.g. camelCase
- Widely used in other programming languages

Naming tips

- Many workplaces will have their own standards
- Be consistent
- Strive for names that are concise and meaningful
- Variable names should be nouns and function names should be verbs

Comments

- Comment your code!
- Comments should be informative

Bad

```
# set a to 0.05
a <- 0.05
```

Good

```
# set alpha to a typical significance level
alpha <- 0.05
```

Spacing

Bad

```
average<-mean(x,na.rm=TRUE)
average <- mean(x ,na.rm = TRUE)
```

Good

```
average <- mean(x, na.rm = TRUE)
```

Code organization

- Place all dependencies at the beginning of the script.
- Use blank lines and comments to group sections of code.

```
library(randomForest)

data(iris)

# create train and test datasets
train_index <- sample(150, 0.80 * 150)
train <- iris[train_index,]
test <- iris[-train_index,]
```

Assignment Operator

- Use “->” for assignment
- “=” is used in function arguments
- For more information about assignment operators in R, http://stat.ethz.ch/R-manual/R-patched/library/base/html/assignOps.html

Styler

- Reformats selected code, entire files, or packages
- Written in 2017 by Lorenz Walthert with Kirill Müller and Yihui Xie
- Once installed can be accessed from RStudio Add-ins
- Default style is Tidyverse but is customizable: https://lorenzwalthert.netlify.com/posts/stylerpost/
- Go to tutorial…

Programming Patterns