Unit Testing in R

Will Kaufman

July 1, 2016

What is Unit Testing?

- "Unit testing is a software testing method by which individual units of source code... are tested to determine whether they are fit for use." 1
- ▶ Allows for more efficient development of code and packages.
- Tests for each basic principle of the code, multiple situations and incorrect invocations.
- Newly discovered bugs and errors are added as additional test cases.

Options for Unit Testing in R

- RUnit by Matthias Burger, Klaus Juenemann, and Thomas Koenig.²
- testthat by Hadley Wickham.³
- RUnit implements the same syntax as other unit testing implementations in other languages (JUnit, CppUnit, PerlUnit). This tutorial will focus on RUnit (though both are acceptable solutions).

²https://cran.r-project.org/web/packages/RUnit/index.html.

³https://cran.r-project.org/web/packages/testthat/index.html.

RUnit Documentation and Resources

- ► For more detailed explanations and examples, see the reference manual and the package vignette.
- ► For a short walkthrough of RUnit basics, see this tutorial by John White.
- For other questions, Google is your friend.

Creating Unit Tests

Given a package

```
checkEquals(target, current, msg,
           tolerance = . Machine$double.eps^0.5,
3
          checkNames = TRUE, ...)
  checkEqualsNumeric(target, current, msg,
           tolerance = . Machine$double.eps^0.5, ...)
5
  checkIdentical(target, current, msg)
6
  checkTrue(expr, msg)
  checkException(expr, msg,
8
9
           silent = getOption("RUnit")$silent)
  DEACTIVATED (msg)
```

Tests Involving Random Numbers

 Specify the type of random number generator in defineTestSuite, and specify set.seed in the test function