USEFUL R FUNCTIONS

for Debugging Purposes

Kartikeya Shandilya
Credit Risk Analytics Associate
Enova

Overview

Problem

- Debugging an R-code can be daunting for beginners
- Compatibility issues between R data-types and functions

Solution

- Use native R functions to better understand the data-type
- Generate additional summary / descriptive stats of the data
- Utilize advanced debugging tools

Generic debugging strategy

1. Observe the program behavior, over different inputs / runs
2. Formulate possible explanations for the given behavior
3. Design experiments to support / refute these theories
4. Run experiments and evaluate results
5. Correct the bug and retest

Understanding the data-type

- class()
  - Describes the data-type
- length()
  - Vectors (integer, numeric, character, factor) : Number of elements
  - Data frame : Number of columns
  - Matrix / Array : Number of elements (# of columns * # of rows)
- dim()
  - Only works with multi-dimensional objects such as df, matrix, array
  - Typically returns: nrow, ncol

Understanding the data-type

- names() – for vector objects
  - Gives the names i.e. labels for vector data-types
  - Returns colnames for data frames
- dimnames() – for multi-dimensional objects
  - Returns list with two elements: rownames and colnames
- attributes()
  - Provides all useful attribs. such as class, names, dimnames, levels, etc.
- str()
  - Gives a compact description of an object's internal structure

Summarizing the data

- Native functions for summarizing / tabulating data
  - summary() : Displays a summary of the input object
  - table(), xtabs() : Used to produce contingency tables, cross-tabs, etc.
  - apply(), sapply() : Compute descriptive stats on subsets of data

- Non-native functions
  - Hmisc :: describe() : Similar to summary
  - doBy :: summaryBy : Quite similar to apply / sapply, easier syntax
  - reshape :: cast : Similar to summaryBy

Advanced Debugging tools

- Excellent debugging tool: RStudio IDE
- Official doc: http://www.rstudio.com/ide/docs/debugging/overview
- Key debugging features:
  - Error inspector and traceback()
  - Re-run with Debug and options(error = browser)
  - Breakpoints and browser()

THANK YOU !!