Workshop 1: *Introduction to Statistical Computing*
Training Course on Economic Assessment Methods for Policy Support of Climate Change Adaptation in the Agricultural Sector in Lao PDR

Drew Behnke

Food and Agriculture Organization of the United Nations
UC Berkeley and UC Santa Barbara

NAFRI

Vientiane, Lao PDR

August, 2014

What is R?

- R is an open source statistical programming language that is a free alternative to languages such as SPSS, Stata, and SAS.

- It is used largely by statisticians and others in the social sciences for statistical analysis and displaying data graphically.

- It is used by people all over the world, and consequently there are a wide variety of resources available online.

- You can download the R program from the training website or directly from CRAN:

  - Windows:
    - http://cran.r-project.org/bin/windows/base/R-2.13.1-win.exe

  - Mac:
    http://cran.r-project.org/bin/macosx/R-2.13.1.pkg

- R comes with a set of "base" packages already installed.

- Hundreds of additional packages have been created to form specific tasks.

- Installing packages requires two steps:

    - (1)   install.packages("package_name")
    - (2)   library("packagename")

    - (1) only has to be done the first time, (2) has to be done every time you have a new R session.

- POSITIVES:
  - Free

  - It is very flexible.
    - This is partly because of the many "packages" available online that people have written for a wide variety of problems.

  - It is transparent – easy to see what the commands do.

  - It has very good graphical capabilities including making pretty maps (free alternative to ArcGIS).

- NEGATIVES:

  - It has a command line interface so, depending on programming experience, there may be a sharp learning curve.

  - If you are familiar with other programs (like SPSS, SAS, or Stata) it may be frustrating because R uses largely its own, non-standard, terminology.

  - It is slow with very large data sets (SAS is the best for this).

- Running statistical models and forecasting.

- Displaying data graphically (especially on Maps).

- Managing data.

Now, install R!

- When R is open you can type commands, written in R syntax, directly into the console.

- Alternatively, R code can be stored in text files (ending with .R) which you can then "run" to communicate with R.

- "Workshop1.R" is an example of this type of file.
  - In order to run code from a file you can highlight the text that you want to run, then copy and paste it into the R console and press return.
  - There are also shortcuts for Windows and Mac machines to run entire files or just highlighted code.
    - (Ctrl + r) in Windows; (command + return) for Mac;

- You can type any valid command into the R console and it will print the result, however, it will not be saved.

- You can use "=" to create a new object which contains the output.

- R stores "objects" in its memory which can include numbers, imported data sets, and functions that operate on inputs, among other things.

- Almost every command you execute in R uses one or more functions.

- Functions are called by their name followed by parentheses ().

- For most functions, there are arguments that need to be specified inside the parentheses that tell the function more about what you want it to do. Even if there are no arguments, you still need to use parentheses.

  – For example: help() opens up a blank help window.

- Vectors – n x 1 or 1 x n collection of elements.

- Matrices – n x m collection of elements of the same type and same length.

- Data Frames – n x m collection of elements of the same length (not necessarily same type)

- Arrays – n x m x p collection of elements (generally a collection of data.frames)

- Lists –Can store anything. Elements do not have to be the same/ type or length/etc

- R has very powerful graphics packages that can be used to make of almost any type of graphic you can think of.

- Primary graphics functions:
  - The most basic function is plot().
  - Others include hist(), barplot(), image.plot()

- Secondary graphics functions:
  - lines(), points(), text(), abline()