# S-Plus/R with ESS

helpdesk@stat.rice.edu

16th July 2003

## What is ESS? Why is it important?

ESS (Emacs Speaks Statistics `http://software.biostat.washington.edu/statsoft/ess/`) is an Emacs extension. It is desinged to make S-Plus/R easier to use, especially on Unix platforms. This guide will lead you through some of the most useful ESS features.

## How to install ESS?

`ESS` is installed under helpdesk. To make use of `ESS`, you need to add the following line to your `.emacs` file

```
(load "/home/helpdesk/trial/ess/ess-site")
```

ESS provides 50+ functions which are binded to keys like `C-c C-f`. Since these key-bindings are not easy enough to remember, I bind some of the most important functions to `F1, F9, F10, F11, F12`. If you would like to use these bindings, copy the ESS part of the sample `.emacs` file on helpdesk do tfile repository `http://www.stat.rice.edu/~helpdesk/dotfiles` to your `.emacs` file. Throughout this guide, I will use these bindings, instead of the standard ones.

## A Step by step Tutorial:

### Starting ESS

1. Start emacs with command: `emacs &`
2. Enter command `M-x R`

`M-x` means hold down the Meta key and press x. The meta key is usually located near the space bar on a SUN keyboard. If you are using a windows 102/105 keyboard with X-win32, the right Alt key is usually used as the meta key. If there is no meta key at all, press `Esc`, release it, then press `x`. You can also use `M-x S` to start S-Plus.

If ESS is properly installed, emacs will open a buffer named `*R*` and start an instance of R in it. If emacs returns an error, check your `.emacs` file. The easiest way to correct your .emacs problem is to use the sample `.emacs` file. You can copy the sample `.emacs` file directly from the helpdesk if you are on stat007, just use command:

```
cp /www/helpdesk/public_html/dotfiles/.emacs ~/.emacs
```

## Interact with the S-Plus/R process

Enter

```
vector1 <- c(1,2,3)
```

and press enter. Now, you would like to enter another command:

```
vector2 <- c(3,3,3)
```

You have several ways to input the second command. Instead of inputing the whole command again, you can use one of the following features of ESS:

1. **Edit the previous command directly**
   Click on the previous command, make appropriate changes, press ¡CR¿. ESS will copy the edited command to the last command line and execute it. The drawback of this method is that since you changed the first command, you would not be able to get a clean transcript of the command sequence.

2. **Navigate the command history and call back the previous commands**
   Use `Ctrl-Uparrow/Downarrow` to navigate the previous commands. In our case, use `Ctrl-Uparrow` to call back the previous command and edit it. This method is preferred if the output of the previous commands are long.

**Note:**

If you use gnome, `Ctrl-Arrow` keys are intercepted by gnome to do something else. In this case, try `Alt-Up/Down`.

Then, to add the two vectors together, use the command:

```
vector <- vector1 + vector2
```

The shortest key sequence is:

```
ve<tab>_ve<tab>1 + ve<tab>2
```

Here we used two features of ESS:

1. **Automatic translation of _ to <-.**
   Although many user prefers _, ¡- is surely easier to follow. ESS let you input ¡- with less key strokes by automatically changing _ to <- in output.
2. **Completion**
   ¡TAB¿ key lets ESS complete your object name as much as it can. Since there are only two objects with names starting with ve, ESS will complete ve to vector. If object vector2 does not exist, ESS will complete ve to vector1. This is the same as the command line completion in tcsh.

**Note:**

Unfortunately, S-Plus 6.0 for Sun Solaris (the version we are using) has a bug which prevents ESS from getting the right completion list. I have sent a bug report to ESS and newer version of ESS (or S-Plus) is supposed to solve or work around this problem.

Now, try to input more commands... Here are more tricks:

1. **Help on command: F1**
   Press F1 if you would like to look up the manual of a function. Input the function name after 'Help on:' in the mini-buffer (the last line of the emacs window.)
2. **Go to the last line of the command window: F12**
   Whenever you are in the command windows, press F12 to go to the last line to input new command.

## Open or create a script file

Files with extension .r .ssc or .s will be opened in ESS/S mode. You can tell this be the ESS[S] string on the status line. Press

```
C-x C-f test.r
```

to open file test.r. If this file does not exist, a new file will be created. You can use emacs command C-x 2 to open two windows so that you can display the ESS command window and the script window at the same time.

## Edit a script file

Input the following commands in the script buffer

```
vector1 <- c(2,3,4)
vector2 <- c(3,3,3)
vector <- vector1 + vector2
```

You can not use completion here but you have all the editing capacity of emacs at hand. The best part of the `ESS/S` mode is the syntax coloring. With the help of colors, the program is easier to read and it is easy to pick out minor grammar mistakes like unbalanced 's .

### Submit commands

Put the cursor on the first line, make sure you have a `R` process running. press

```
F9
```

ESS will ask you which process to load, if you only have one S-Plus/R process, press enter. ESS will then send **the current line** to the R process and execute. The actual ESS command is `ess-eval-line-and-step`. It is possible to use other commands like `ess-eval-line-and-go, ess-eval-line`. The first command evaluate the current line and go to the command window, the second command just execute the current line. If you would like to see all ess comand, type

```
M-x ess-<tab>
```

Similarly, use `F10` to send **selected text** and `F11` to send the **whole buffer** to the S-Plus/R process. So you have your choice of stepping through, evaluating part of the script or `source` the whole buffer.

`F1` and `F12` keep their meaning as they do in command window.

## ESS for window

The windows version of S-Plus is fairly easy to use but the windows version of R is not. As a result, you might want to install the windows version of R, Emacs and ESS. An outline of the steps is:

1. Download and install `Emacs` and `R`
2. Download `ESS` and decompress.
3. set `$HOME` and
4. add the path of `R-term` to `$PATH` from control panel.
5. change `ess-site.el` and optionally compile it with emacs command.
6. put an `.emacs` file under `$HOME.`

helpdesk

$Date: 2003/07/17 02:49:21 $