Running R on Linux Workstation

Chuck Song 4/12/2011

Overview

- Preparation
- Linux Basics
- Running R in Linux
- Advanced skills (optional)

Tools needed

- Windows Users
  - Web browser (IE, Firefox, safari, NOT chrome)
  - SSH client (e.g. PuTTY)
  - FTP client (e.g. FileZilla)
  - X server (optional, e.g. Xming)
- Linux or Mac Users
  - Web browser (Firefox)

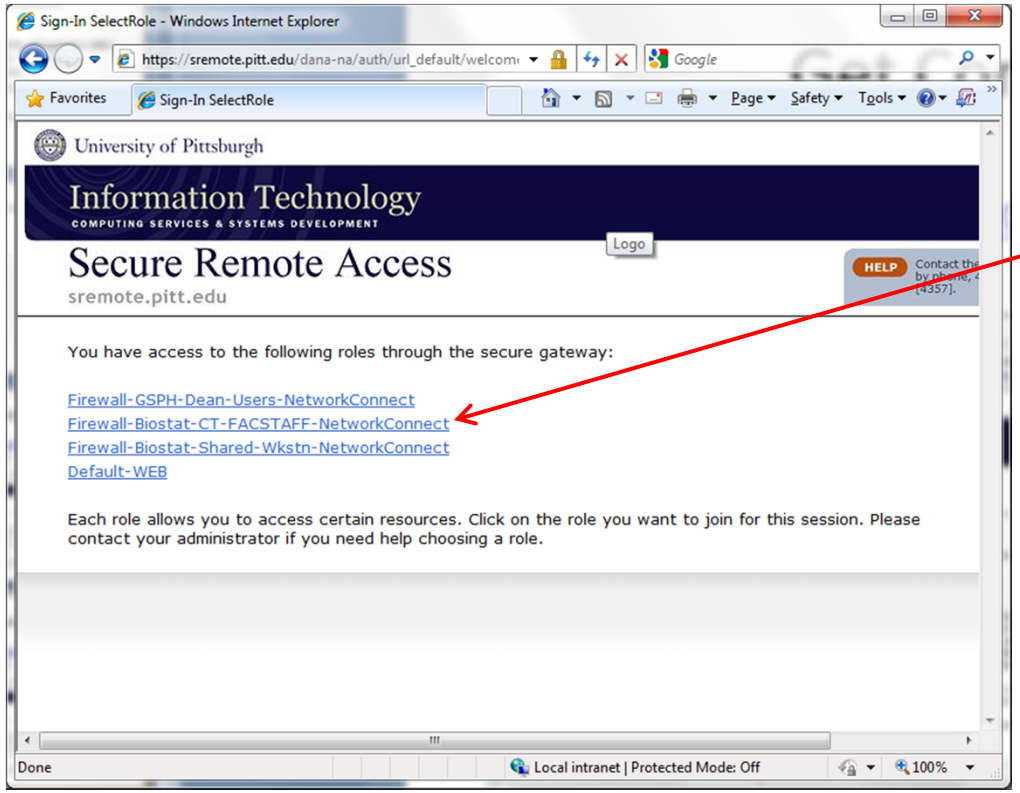
Connect Firewall

- Login on https://sremote.pitt.edu using your Pitt ID and password.

Allow and Install the program

Connect Firewall

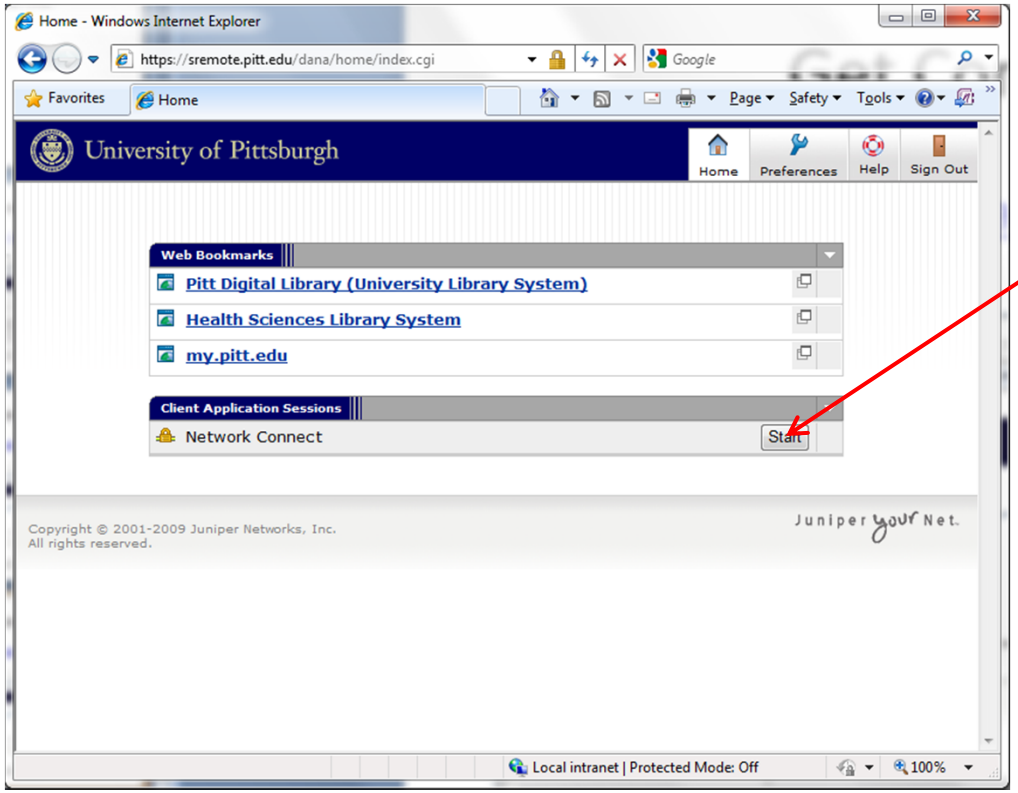
- Use the Role of Firewall-Biostat-CT-FACSTAFF-NetworkConnect



Click here

Connect Firewall

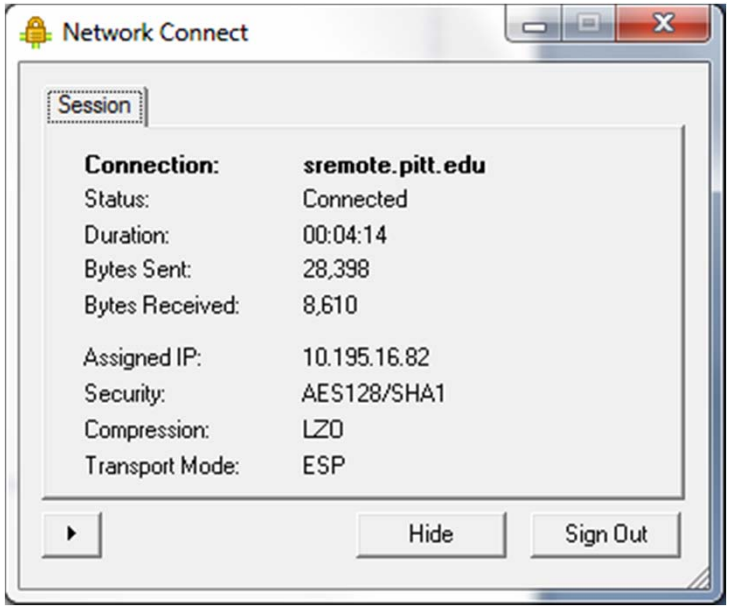
- Click start and allow the download and installation of “Network Connect 6.5.0”



Click here

Connect Firewall

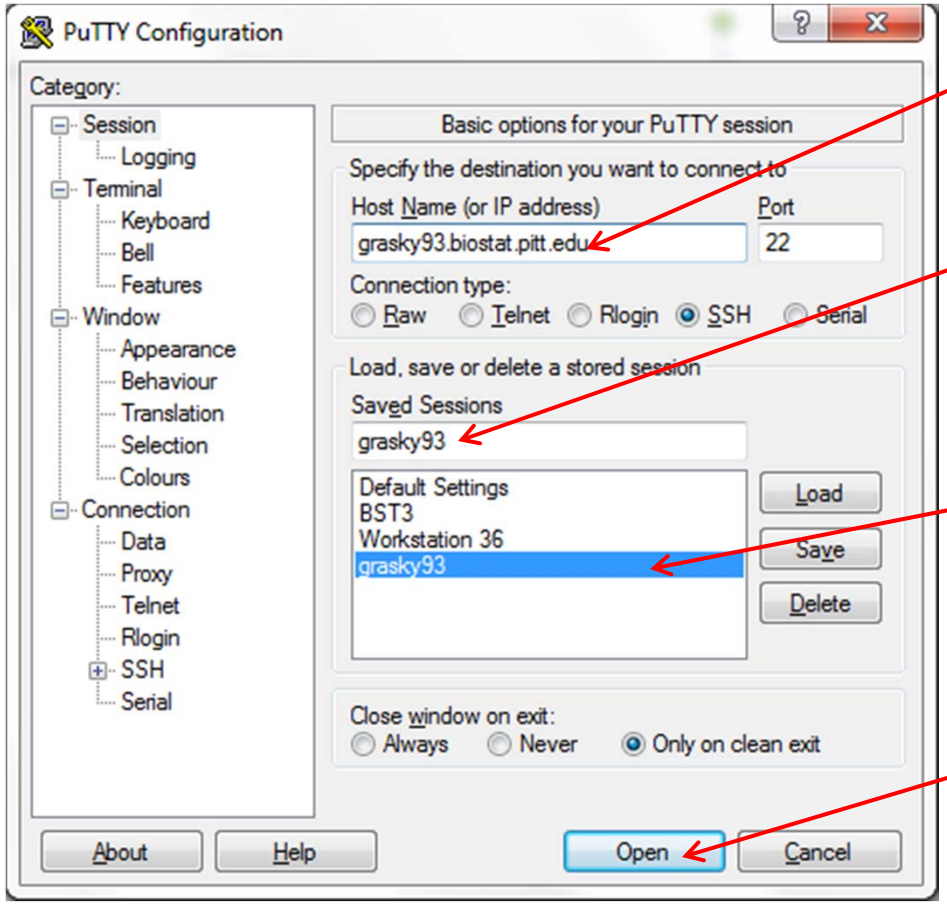
- The “Network Connect” program
- You can find it in the system tray (right bottom corner)



Connect Server

- Server Information
  - URL: grasky93.biostat.pitt.edu
  - IP: 208.68.139.38

Connect Server



URL here

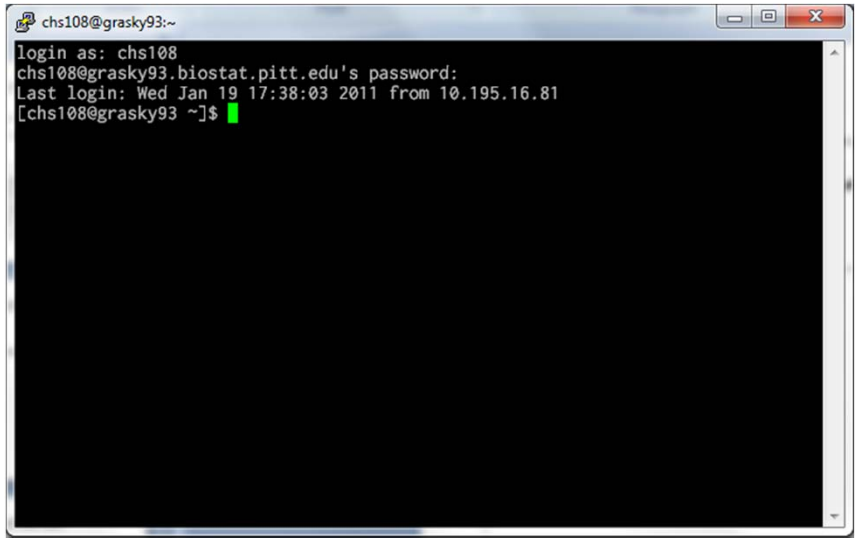
Name your session

Manage your sessions here

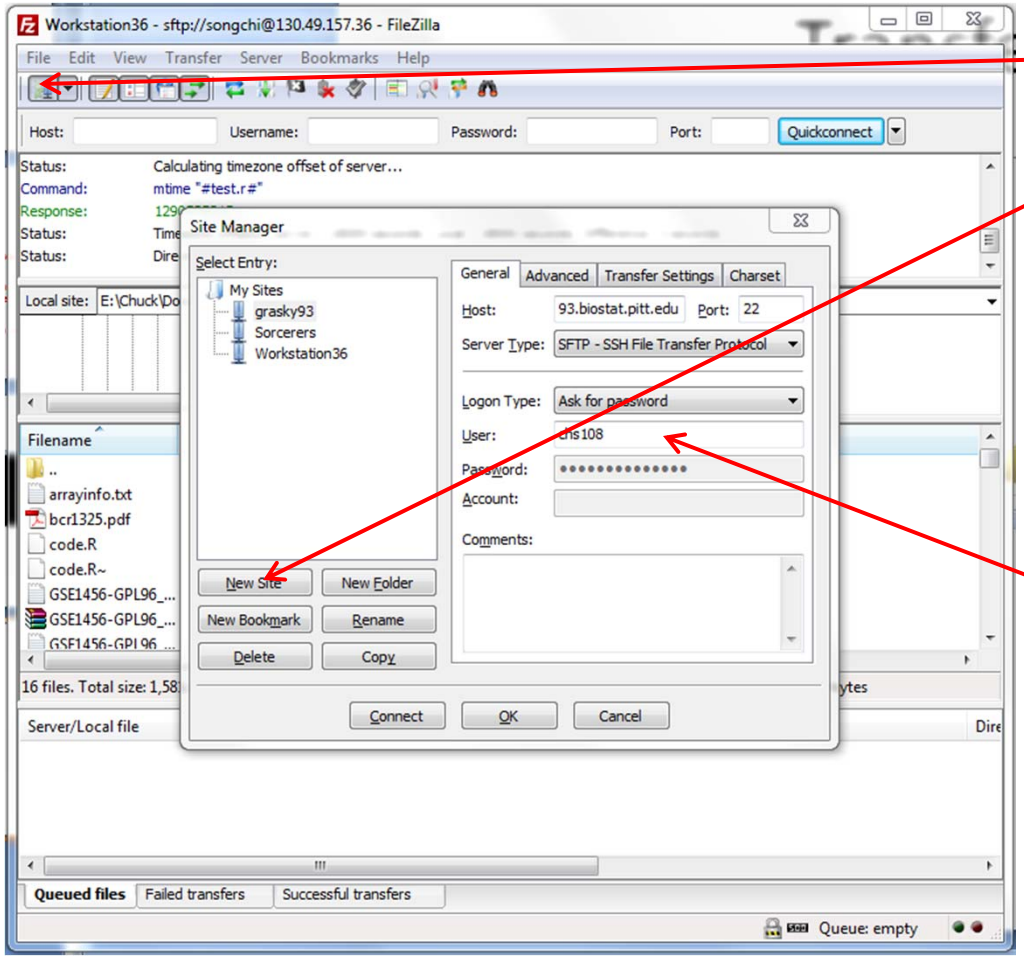
Click here to start

Connect Server

- Enter your username and password. Then you are ready to go!
- Contact system manager (Charles W. Alcorn, cwa3@pitt.edu) for your account information.



Transfer Files



Launch Site Manager

Create a new site

Input site info:

Server type - SFTP

Logon type - ask for password

Enter your username

Click OK or Connect

A Working Example

- helloworld.r

```
# our first R program in Linux
sink("helloword_routput.txt")
cat("Hello world!\n")
# generate 100 x's ~ N(0,1)
x <- rnorm(100)
print(x)
sink()
png("helloworld_rplot.png")
hist(x)
dev.off()
```

Histories (Wikipedia)

- Unix
  - A multitasking, multi-user computer operating system originally developed in 1969 by a group of AT&T employees at Bell Labs
  - BSD, IBM AIX, Sun Solaris, HP-UX
- GNU
  - GNU is a Unix-like computer operating system developed by the GNU project, ultimately aiming to be a "complete Unix-compatible software system" composed wholly of free software.
  - Development of GNU was initiated by Richard Stallman in 1983 and was the original focus of the Free Software Foundation (FSF)
- Linux
  - Linux refers to the family of Unix-like computer operating systems using the Linux kernel
  - The name "Linux" comes from the Linux kernel, originally written in 1991 by Linus Torvalds

Linux ABC

- Linux is case sensitive, most of its programs are also case sensitive
- File deleted cannot be easily recovered (no recycle bin)
- You are forced to use its text command lines (Yes, they are powerful!)

Linux ABC

- Knowing the prompt
  - [chs108@grasky93 ~]$_
  - [*username@hostname directory*]$_
- Knowing the commands
  - *command [option] [source file(s)] [target file]*
  - Options often have the “-x” or “--xxx” format
  - Advanced skills:
    - “<“, “>”, “>>”, “2>” for input/output/error redirection
    - “|” for pipe

Useful commands

- whatis – short description of a command
- man – display the manual of a command
- Try:
  - whatis whatis
  - whatis man
  - man man
  - man ls

Useful commands

- ls – list directory contents
  - options:
    - “-l” – long listing format
    - “-a”, “--all” – listing all files (hidden files starts with “.”)
    - “--help” – display the help for ls command
    - “-R” – recursively list subdirectories and their contents
- Try:
  - ls
  - ls -l
  - ls -al
  - ls -al ~
  - ls -aR

Useful commands

- Understanding “ls -l” results

```
[chs108@grasky93 ~]$ ls -l
total 48
drwxrwxr-x 2 chs108 chs108 4096 Feb 14 00:17 test
-rw-rw-r-- 1 chs108 chs108   57 Jan 13 15:34 test.r
-rw-rw-r-- 1 chs108 chs108   57 Jan 13 15:29 test.r~
-rw-rw-r-- 1 chs108 chs108    2 Jan 13 17:06 #test.R#
-rwxrwxr-x 1 chs108 chs108   64 Jan 19 17:54 test.R
-rw-rw-r-- 1 chs108 chs108 1475 Jan 13 15:35 test.r.Rout
```

Useful commands

- cd – change the working directory
  - Note:
    - “.” – current directory
    - “..” – parent directory
    - “~” – home directory
    - “/” – root directory
- Try:
  - cd ..
  - cd ~
  - cd /
  - cd home
  - cd /usr/local

Useful commands

- echo – display the content given
- touch – make a new file
- cat – display the content of a file
- more – display the content of a file in different pages
- head/tail – display the start/end of a file
- Try:
  - touch test.txt
  - echo “this is a test file” >test.txt
  - cat test.txt
  - ls -l /usr/bin|more
  - ls -l /usr/bin|head

Useful commands

- cp – copy files
- mv – move/rename files
- rm – remove files
- Try:
  - cp test.txt test.123
  - mv test.123 test.456
  - rm test.456

Useful commands

- pwd: display your current directory
- mkdir: make a new directory
- rmdir: remove an empty directory
  - "rm -r" to remove directory with files. **Be careful!**
- Try:
  - pwd
  - mkdir test
  - mouch test/test
  - rmdir test
  - rm -r test

Useful commands

- chmod – change file access permissions
- Try:
  - chmod a-r test.txt
  - cat test.txt
  - chmod u+r test.txt
  - ls -l test.txt
  - cat test.txt
  - chmod go+r test.txt
  - ls -l test.txt

Useful commands

- grep – print lines matching a pattern
- find – find files in a directory hierarchy
- Try:
  - grep -R “this is” .
  - ls -al|grep test
  - find . -name ‘test’

Useful commands

- top/htop – process viewer
- ps -ef – list all processes running
- Try:
  - top
  - htop
  - ps -ef|grep chs108

Useful commands

- screen – screen manager
  - Use “screen [-S name]” to make a screen session
  - Press Ctrl+A D to detach the current screen
  - Use “screen -ls” to list all available sessions
  - Use “screen -r session” to restore/attach a session
  - Use “exit” in an attached session to end it
- Try:
  - screen -S test
  - top / press Ctrl+A D
  - screen -ls
  - screen -r test
  - exit

Running R in Linux

- Simply using “R” command
- Try:
  - R
  - Type your R commands
  - Use q() to exit
- But, how can I run large R simulations?
  - Paste your commands in R and wait like in Windows? No! Your session will timeout and you will lose everything!

Run large simulations

- Write your R code in a source file and run it from command line.
- Use “screen”
- Use Rscript

Run from command line

- *[nohup]* R CMD BATCH [source] &
  - '&' – force the program to run in background
  - bg/fg can resume the stopped program in background or foreground
  - Ctrl+Z can temporarily stop a foreground program
  - Ctrl+C can terminate a foreground program
  - Text output will be directed to [source].Rout
  - Plots will be directed to Rplots.pdf

Using screen

- Start a screen session using “screen” command
- Run your R simulation as you do in windows
- Use Ctrl+A D to detach and feel free to logout your ssh session
- Next time, login and use “screen -r” to check your result.

Using Rscript

- Similar to command line
- Can put arguments (simulation parameters) in the command line
- More flexible usages
- Example: “test.R” (chmod +x test.R)

```
[chs108@grasky93 ~]$ cat test.R
#!/usr/bin/env Rscript
args <- commandArgs(TRUE)
print(args)
[chs108@grasky93 ~]$ ./test.R 1 2 3
[1] "1" "2" "3"
[chs108@grasky93 ~]$
```

General Suggestions

- Always save your results (text and plots) to file explicitly in your program
- Always save/keep your important intermediate results (to “xxx.RData” files)
- Use try/tryCatch to handle possible errors
- Check if you program is running using command “ps -ef|grep xxx” or “top”
- Learning efficient programming skills

Advanced skills

- Edit text files
- Using ESS to talk to R in Emacs
- Launch X programs (GUI)

Edit text files

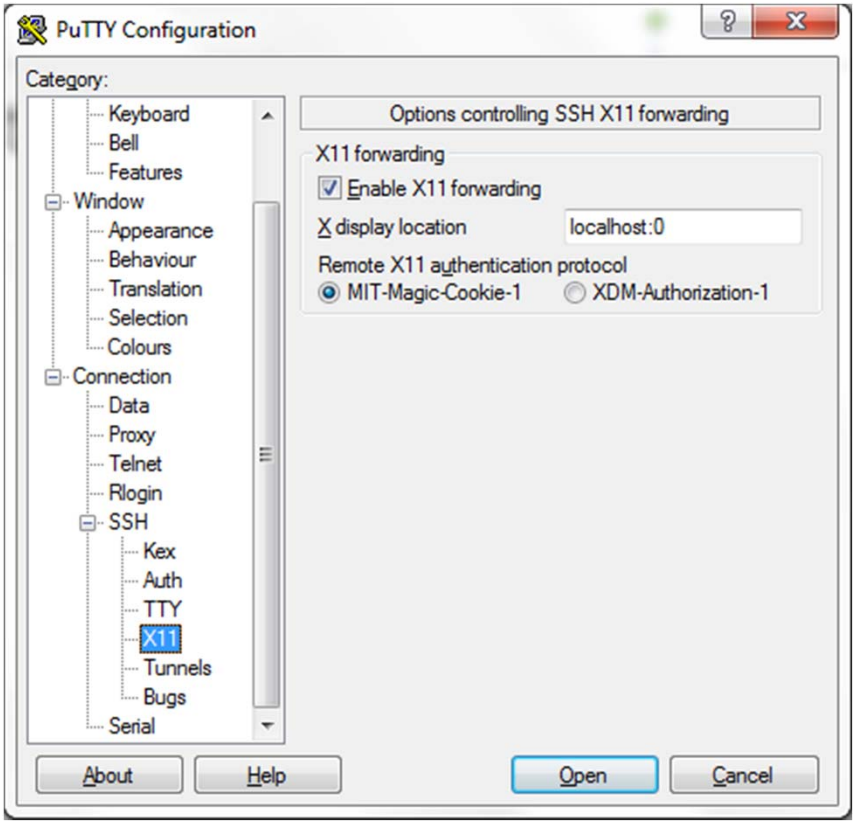
- nano
  - Easy to learn
  - Weakest function
- vim (http://www.vim.org)
  - Slightly tougher to learn (try: vimtutor)
  - Stronger function
- emacs (http://www.gnu.org/software/emacs/)
  - Toughest to learn (http://www.gnu.org/software/emacs/tour/)
  - Strongest function (cross talk to R, http://ess.r-project.org)

Using ESS

- Emacs Speaks Statistics
- Some commands
  - C-x c: exit
  - C-x s: save file
  - C-x 2 or 3: split screen
  - C-x 0: close screen
  - C-x o: move cursor to next screen
  - M-x R: start R session
  - C-c C-n: evaluate R code and go to next line

Using X server (GUI)

- Install X server (Xming)
- Set session properties in PuTTY
- Enable X11 forwarding
- X display location: localhost:0
- Try: firefox &
- Not recommended (only for special needs)



References

- http://www.comptechdoc.org/os/linux/
- http://www.r-project.org/
- http://www.rackaid.com/resources/linux-screen-tutorial-and-how-to/
- http://stat.ethz.ch/R-manual/R-patched/library/utils/html/Rscript.html