

Graduate School Class Reminders

- ▶ Maintain six feet of distancing
- ▶ Please sit in the same chair each class time
- ▶ Observe entry/exit doors as marked
- ▶ Use hand sanitizer when you enter/exit the classroom
- ▶ Use a disinfectant wipe/spray to wipe down your learning space before and after class
- ▶ Media Services: 414 955-4357 option 2

Outline for today

- ▶ Introductions
- ▶ Syllabus
- ▶ Documentation
- ▶ Computing on the Cheese Cluster
- ▶ Welcome to Emacs

Introductions and syllabus

- ▶ Introduce yourself
 - ▶ Pronounce your name
 - ▶ Where are you from?
 - ▶ What graduate program are you in?
 - ▶ What schools did you go to and your major?
 - ▶ What are your other interests?
- ▶ Do you have a laptop? If so, what OS? Bring it along
- ▶ Programming experience with R, SAS and Linux (the biostatistical “stack” and basis for this course)
- ▶ For example, here's my background
- ▶ A statistician AND a programmer: you have to be both
- ▶ I consider myself a data scientist, NOT a computer scientist
- ▶ From 1991, heavy use of SAS
- ▶ From 1991, heavy use of UNIX/Linux/Unix-like
- ▶ From 2009, heavy use of R

Documentation on the web

- ▶ CRAN: <http://cran.r-project.org>
- ▶ R manuals: <https://cran.r-project.org/manuals.html>
- ▶ SAS: <http://support.sas.com/documentation>
- ▶ Step-by-Step Programming with Base SAS 9.4 (SbS):
<https://documentation.sas.com/api/docsets/basess/9.4/content/basess.pdf>
- ▶ SAS 9.4 Programmer's Guide: Essentials (PGE):
<https://documentation.sas.com/api/docsets/lepg/9.4/content/lepg.pdf>
- ▶ Wiki: <https://wiki.biostat.mcw.edu> (MCW/VPN)

Wikipedia: a moment in my history

- ▶ for computer science and technology, it is an excellent free choice when others are unavailable
- ▶ most articles have citations for those who need to dig further
- ▶ it is very helpful as an introduction to a new topic
- ▶ but doesn't have much to offer for those learning R and SAS
- ▶ the follow-on to Nupedia which I was a part of in 2000
- ▶ Nupedia founded by Jimmy Wales who later founded Wikipedia in January 2001 which arose from it
- ▶ Nupedia was a dramatic failure and a footnote in history
- ▶ But it spawned Wikipedia a major unintended success
- ▶ wiki is simple hence its popularity: ironically I far prefer LaTeX
- ▶ <https://en.wikipedia.org/wiki/Wiki>

RTFM: Read the F****ing Manual

“If all else fails read the instructions.” Donald Knuth, a renowned computer scientist and the inventor of TeX (precursor to LaTeX) and many other achievements, might (or might not) have said this

- ▶ Unix means UNIX, Linux, macOS or similar Unix-like operating systems
- ▶ To identify Unix-like: `.Platform$OS.type=="unix"`
- ▶ A counter-example: it does NOT mean Microsoft Windows
- ▶ To identify MS Windows: `.Platform$OS.type=="windows"`
- ▶ `on gouda $ man Unix-command like $ man man`
- ▶ `on gouda $ info Unix-command like $ info info`
- ▶ Emacs has the capability to render both types as we'll see
- ▶ `M-x man`
- ▶ `M-x info`
- ▶ N.B. the Helvetica sans serif proportional font used for text whereas a fixed-width teletype/typewriter font with serifs is used for commands and URLs, etc., i.e., what you type

Typical biostatistical computing stack

(TIOBE Index of the Top 100 Programming Languages:
August 2022) <https://www.tiobe.com/tiobe-index>

- ▶ GNU/Linux kernel
- ▶ GNU Compiler Collection (GCC): C(2)/C++(4)/Fortran(19)
- ▶ GNU tools: bash(43), core utilities
- ▶ GNU Emacs: programmer' editor
mostly written in Emacs Lisp(30) and the rest in C
- ▶ GNU R(16): its free and is supported by other free software
Comprehensive R Archive Network (CRAN): 18558 packages
descendent of S and S-Plus
- ▶ SAS(24) is not free, but MCW has a site-wide license
(that I negotiated)
- ▶ X Window System and GNOME2/MATE
- ▶ Structured Query Language: SQL(9) for relational databases
- ▶ TeX Live bundle of TeX/LaTeX
- ▶ Python(1)/Java(3)/Perl(20) NOT used much in biostatistics
so they will not be discussed further in this course

N.B. Python is interactive (slow) so we naturally use R instead

Data Advantages of R and/or SAS vs. everything else

- ▶ programming languages for data processing and analysis
- ▶ convenient environments that are naturally **vectorized**
- ▶ many years of refinement: SAS (1966) and S/R (1976)
- ▶ wide availability on many platforms
- ▶ particularly well-adapted to Linux/UNIX (and macOS for R)
neither SAS nor R adapt well to Windows as we'll see
- ▶ reasonably priced: R is free and SAS is paid for by employers
- ▶ at MCW, SAS is free for students and cheap/free for others
- ▶ large user community of statisticians and related professions
- ▶ interfaces for data input and data management systems
- ▶ many functions of interest such as probability distributions
- ▶ facilitate the creation of graphical visualizations
- ▶ provide access to advanced statistical analysis routines: NOT
the focus of this course so listed last, but immensely important

Advantages of the Client-Server Model: the Server

Modern GNU/Linux Server Technology Layers

- ▶ Linux kernel: always in memory facilitating hardware/software
- ▶ Software tools: GNU Project
- ▶ Dynamic shared libraries and package managers:
- ▶ Transmission Control Protocol/Internet Protocol (TCP/IP)
Secure Sockets Layer/Transport Level Security (SSL/TLS)
- ▶ Programming language: GNU Compiler Collection (GCC)
- ▶ Graphical User Interface: X.org X Window System
X version 11 (X11) protocol and GNOME2/MATE
- ▶ Text: American Standard Code for Information Interchange (ASCII) and Unicode Transformation Format (UTF)
- ▶ File system: Redundant Array of Independent Disks (RAID),
Journaling, Extents File System (XFS) and **tape backup**
- ▶ Central Processing Unit (CPU): AMD64/x86-64
Multi-threading: servers typically have 2 or 4 CPUs and each
has multiple cores capable of 1 or 2 threads
- ▶ General Purpose Computing on Graphics Processing Units (GPGPU): As a consequence, incapable of audio/video

Advantages of the Client-Server Model: the Client

Many layers are similar: just listing important differences

- ▶ Use your PC/Mac as the client rather than comparable server hardware/software which does not work as well for a variety of reasons: not up-to-date, lacking hardware/drivers, etc.
- ▶ Client software: email software with calendaring/filtering/etc., modern web browsers and Microsoft Office support
- ▶ Graphics Processing Units (GPU): Capable of processing audio/video
- ▶ File system: **NO TAPE BACKUP**
Store everything that you want to keep on the server
- ▶ Virtual Private Network (VPN): with VPN software the client can reach the server from ANYWHERE via the Internet (ANYWHERE except MCW because the VPN hardware is on the Internet side of our local connection so we can't get to it, but we don't need to when we are on campus)

A Brief History of UNIX (R)

- 1969: AT&T Bell Labs starts work on UNIX
- 1970: open source UNIX provided for small fee: Bell flavored UNIX
- 1972-3: Bell Labs develops C, re-writes UNIX in C
- 1973-8: DARPA invents TCP/IP network protocol
- 1978: University of California releases Berkeley Software Distribution BSD flavored UNIX
- 1981-3: ARPANET goes TCP/IP creating the Internet
- 1987: MIT/DEC release the X Window System
X protocol version 11 AKA X11
- 1988-94: Apple lawsuit vs. Microsoft and HP over
GUI “Look and Feel” copyright/patent infringement
- 1990: AT&T and BSD are merged into UNIX SVR4
- 1992-4: **UNIX Wars**: Free BSD release blocked by AT&T lawsuit
- 1993: CDE is released: standard UNIX GUI
- 2000: TLS (1999) and SSH (2000) are released for Internet security

A Brief History of GNU Linux

- 1984: Richard Stallman creates GNU (GNU is Not Unix) as “a complete UNIX-compatible software system” along with the GNU General Public License (GPL)
- 1991: Linux kernel by Linus Torvalds mimics UNIX
- 1992: Linux kernel GPLed and paired with GNU tools
- 1992-4: **UNIX Wars:** Free BSD release blocked by AT&T lawsuit gifts Linux mindshare with Debian and Red Hat releases
- 1997-8: KDE/GNOME GUIs head start (GNOME for Solaris not until 2000)
- 1999: Intelligent package installers
GNU Compiler Collection (GCC): C/C++/FORTRAN
- 2002: Red Hat Enterprise Linux (RHEL): Red Hat flavored Linux
- 2003: Fedora Project including Fedora Linux for lap-/desk-top PCs and Extra Packages for Enterprise Linux (EPEL)
- 2004: Lap-/desk-top friendly Ubuntu Linux: Debian flavored Linux shipped on 20-50% of new servers
- 2006: the free RHEL clone CentOS is announced
- 2014-20: CentOS: an “official” clone of Red Hat

A Brief History of GNU R

- 1976: John Chambers creates S at Bell labs for UNIX: free software
- 1988: S-Plus released for DOS: a commercial successor to S
- 1995-8: S-Plus released for Windows (1995), UNIX (1996) and Linux (1998)
- 1997: Ross Ihaka and Robert Gentleman release GNU R as a free successor to S that is integrated with MacOS and Windows with the R Development Core Team (including Chambers)
- 1999: Comprehensive R Archive Network (CRAN) debuts
- 2001: R supports Mac OS X (macOS) and enters the Unix-like era
- 2019: R v3.6 released
- 2020: R v4.0 released

A Brief History of SAS

- 1966-8: Anthony Barr and Jim Goodnight begin development at North Carolina State University on IBM mainframe computers
- 1972: First release of SAS, but they lose their NIH funding
- 1973: John Sall joins the project
- 1976: SAS Institute is incorporated in Cary, NC
- 1985: SAS re-written in C for OS portability: modern SAS era begins
- 1990: SAS v6 released with added support for UNIX and Windows
- 1999: SAS v8 released with added support for Linux
- 2013: SAS v9.4 Analytical Products 12.3 released
- 2018: SAS v9.4 Analytical Products 15.1 released
- 2020: SAS v9.4 Analytical Products 15.2 released

MCWCorp authentication and Solstice

- ▶ start here on day 2
- ▶ one username/password to rule them all?
- ▶ cross-platform MCWCORP ActiveDirectory authentication
- ▶ WiFi, email, Office365, D2L, VPN and the Cheese Cluster
- ▶ download and install the Solstice client
<https://www.mersive.com/download>
- ▶ hands-on: taking over the projector
Caveat: M1 Macs don't work well in my experience

The Cheese Cluster

- ▶ login to gouda
- ▶ Your WiFi gateway
- ▶ To run demanding code start on an interactive shell on colby
- ▶ The Cheese Cluster uses the Terascale Open-source Resource and QUEue Manager (TORQUE)
- ▶ For single threading

```
qsub -I -X
qsub -I -X -l nodes=1
```
- ▶ For multi-threading with P threads

```
qsub -I -X -l nodes=1:ppn=P
```
- ▶ See the wiki for more info on TORQUE

Secure shell, ssh, and Secure copy, scp

- ▶ The commands `ssh` and `scp` are included in Linux and macOS
- ▶ For Windows, see the wiki

`https:`

`//wiki.biostat.mcw.edu/Secure_shell_and_secure_copy`

`https://wiki.biostat.mcw.edu/Secure_copy`

Secure shell, ssh, and Secure copy, scp

- ▶ First, we need to create keys: a public key and a private key
- ▶ Let's look at the manual for ssh-keygen
- ▶ check the directory on client and/or server: `$ ls -la .ssh`
- ▶ for example, on Linux or macOS, we would do as follows
- ▶ create it if not there: `$ mkdir .ssh`
- ▶ and protect it! `$ chmod 700 .ssh`
- ▶ check it again: `$ ls -la .ssh`
- ▶ generate your keys on the server and copy them to your client
- ▶ `$ ssh-keygen -t rsa`
- ▶ `$ cp .ssh/id_rsa.pub .ssh/authorized_keys`

```
scp USER@gouda.biostat.mcw.edu:~/.ssh/id_rsa.pub ~/.ssh
```

```
scp USER@gouda.biostat.mcw.edu:~/.ssh/id_rsa ~/.ssh
```

login to gouda with ssh or x2go with your key (no password)

x2go clients

- ▶ download and install x2go client
- ▶ <https://wiki.x2go.org/doku.php/doc:installation:x2goclient>
- ▶ For Windows, this is the best version according to Chris
<https://code.x2go.org/releases/binary-win32/x2goclient/releases/4.1.2.0-2018.06.22>
- ▶ let's go over the x2go settings and test

A Brief History of Emacs and ESS

- 1975: Emacs created by Richard Stallman (AKA RMS) at MIT
- 1984: RMS re-writes GNU Emacs (GPL) in C
 - Apple Macintosh Human Interface Guidelines (HIG)
- 1986: emacs FORTRAN-mode: intelligent editing for FORTRAN
- 1987: IBM Common User Access (CUA) failed alternative to HIG
- 1988-94: Apple lawsuit vs. Microsoft and HP over GUI “Look and Feel” copyright/patent infringement
- 1990: John Sall adds some SAS support to GNU Emacs
- 1991: Multi-lingual XEmacs (GPL) for X11 released
- 1994: GNU Emacs (GPL) for X11 released
 - Tom Cook releases SAS-mode (GPL)
- 1994-7: Anthony Rossini creates ESS (GPL) containing the Emacs modes ESS[SAS], ESS[R] and ESS[Stata]
- 1999+: Rich Heiberger and I improve ESS[SAS] for batch processing
- 2000+: ESS and Emacs evolve together for more user-friendliness

Welcome to Emacs

<https://wiki.biostat.mcw.edu/Emacs>

- ▶ Modifier Keys: Emacs documentation looks like this
- ▶ C-KEY means hold down the Control key while pressing KEY
- ▶ For example, C-x means hold down Control while pressing x
- ▶ M-KEY means hold down the Meta key while pressing KEY
- ▶ On PC, the Meta key is usually the Alt key
- ▶ On Mac, the Meta key should be the Option key: see last slide
- ▶ Or, you can press Esc, release, and then press KEY
- ▶ Execute an emacs command: M-x COMMAND which is followed by pressing Enter
- ▶ For example, M-x man to bring up a man page or M-x info the directory of info pages
- ▶ S-KEY means hold down the Shift key while pressing KEY

Hands-on running Emacs

- ▶ To run more recent versions of GCC and the GNU Debugger (these are needed to compile R packages)
- ▶ `$ module load gcc/9.2 gdb/9.2`
- ▶ Let's create a bash shell script to avoid typing that again
- ▶ Open the file to edit: `C-x C-f emacs-26.3`
- ▶ `#!/bin/bash`
`(module load gcc/9.2 gdb/9.2; emacs "$@")`
Encasing commands in parentheses is a **subshell**
- ▶ Try to quit emacs: `C-x C-c`
- ▶ Interrupt command: `C-g`
- ▶ Save the file: `C-x C-s`
- ▶ Really quit emacs: `C-x C-c`
- ▶ Or just copy it:
`$ cp /data/shared/04224/emacs/emacs-26.3 ~`
- ▶ Check the files permissions: `$ ls -l emacs-26.3`
- ▶ Add the execute permission: `$ chmod +x emacs-26.3`
- ▶ Check the files permissions: `$ ls -l emacs-26.3`
- ▶ Run the script in the background: `$ emacs-26.3 &`

Common Emacs Shortcuts

- ▶ C-h is the help key and F1 is its alias
- ▶ But you have to get your laptop to generate an F1
- ▶ For example, C-h k describes the next key pressed
- ▶ Try C-h k F1 k
- ▶ M-F1 opens a new frame (MCW)
- ▶ C-F10 make the font smaller (MCW)
- ▶ C-F11 make the font bigger (MCW)
- ▶ C-x C-f is open a file or a directory
- ▶ F2 is refresh (ESS)
- ▶ F8 is go to *shell* buffer (ESS)
- ▶ M-F8 is go to *shell* buffer in the current directory (MCW)
- ▶ M-w is **copy**
- ▶ C-y is **paste**
- ▶ C-w and Delete are cut
- ▶ C-delete or S-delete is cut (MCW)

Common Emacs Shortcuts

- ▶ `C-c` comments a region (an area of text selected)
- ▶ `C-u` is the prefix command so `C-u C-c` uncomments a region
- ▶ `M-up` go to the beginning of the file (MCW)
- ▶ `M-down` go to the end of the file (MCW)
- ▶ `C-x 2` splits the buffer top over bottom
- ▶ `C-x 1` unsplit the buffer
- ▶ `C-x 3` splits the buffer left and right
- ▶ `C-s` starts a forward search
- ▶ Repeating `C-s` searches for the same string again
- ▶ `C-r` starts a reverse search
- ▶ `C-u C-s` starts a forward regular expression search
- ▶ See Search:Regexps entry of emacs manual : `M-x info`
- ▶ `M-x man re_format` man page on regular expressions

R and Emacs/ESS for your client

- ▶ <https://cran.r-project.org/bin>
- ▶ Vincent Goulet has developed installable binaries for Windows and macOS
- ▶ For Windows:
`https://vigou3.gitlab.io/emacs-modified-windows`
- ▶ Check ESS is working with `M-x ess-version`
- ▶ But, for macOS, Vincent's binaries cause frequent crashes

Emacs/ESS on macOS

- ▶ So I created my own binary based on homebrew from <https://github.com/railwaycat/homebrew-emacsmacport/releases>
- ▶ Copy this file (newer versions of emacs also have issues):
`/data/shared/04224/emacs/emacs-26.3.tar.gz`
- ▶ And extract: `$ tar xzf emacs-26.3.tar.gz -C /`
- ▶ **Mac Command key settings for ~/.emacs**
`/data/shared/04224/emacs/emacs-macOS.el`
- ▶ You also have to install XQuartz
<https://www.xquartz.org>
- ▶ And set its Preferences accordingly on the Input tab:
check “Option keys send Alt_L and Alt_R”
- ▶ Also, in the Keyboard System Preferences
- ▶ On the Keyboard tab
check “Use F1, F2, etc., keys as standard function keys”
- ▶ On the Shortcuts tab you will see ^UpArrow and ^DownArrow
uncheck C-up and C-down
- ▶ Check ESS is working with `M-x ess-version`