Building a command-driven shell-script to plot your maps

- I like to be able to make nice maps of my study areas quickly without having to edit my scripts.
- To achieve that, I use the same script for every map I draw, but call it with command-line options.
- For instance, to plot a map of Western North America with seismicity ranging from \(4 < \text{magnitude} < 9\) and \(0 < \text{depth} < 100\text{km}\) I use:
  
  \[
  \text{run.noam } -a 7 \ -s 1 \ 4 \ 9 \ 0 \ 100 \n  \]

- To plot a map of Central and Easter US with seismicity ranging from \(6 < \text{magnitude} < 9\) and \(0 < \text{depth} < 100\text{km}\) and with topography I use:
  
  \[
  \text{run.noam } -a 1 \ -s 1 \ 6 \ 9 \ 0 \ 100 \ -t \ y \n  \]
Passing arguments to csh scripts

#!/bin/csh -f

# DEFINE DEFAULTS
set plot_topo = n
set area = 1

# USAGE:
if ($#argv < 1) then
    echo "Usage: run.map -a area"
    exit
endif

# READ USER INPUT
foreach a ($argv)
    switch ($a)
        case -a:
            set area = $argv[2]
            breaksw
endsw
shift
end

echo PLOTTING AREA $area
Passing arguments to csh scripts

Now pass arguments to define display of seismicity:
type magmin magmax depthmin depthmax

```
# READ USER INPUT
foreach a ($argv)
switch ($a)
case -a:
    set area = $argv[2]
    breaksw
case -s:
    set plot_sismi = $argv[2]
    set mm = $argv[3]; set mM = $argv[4]
    set dm = $argv[5]; set dM = $argv[6]
    breaksw
endsw
shift
end

Make sure that you are passing the correct arguments to your script!
```
Passing arguments to csh scripts

- Download script `run.map` from the class web site
- Make sure it is executable: `chmod +x run.map`
- Try it, you get:

  Usage: run.noam -a area [-s plot_sismi mm mM dm dM] [-t plot_topo]
  -a geographic area to plot:
    1 --> Central + Eastern US
    2 --> Western US
  -s plot seismicity: 1 => size = f(mag)
    2 => equal size [n]
    then give limits: magmin, magmax, depthmin, depthmax
  -p plot topo, y/n [n]

- Now let’s look at what’s inside...