VIKING THURSDAYS

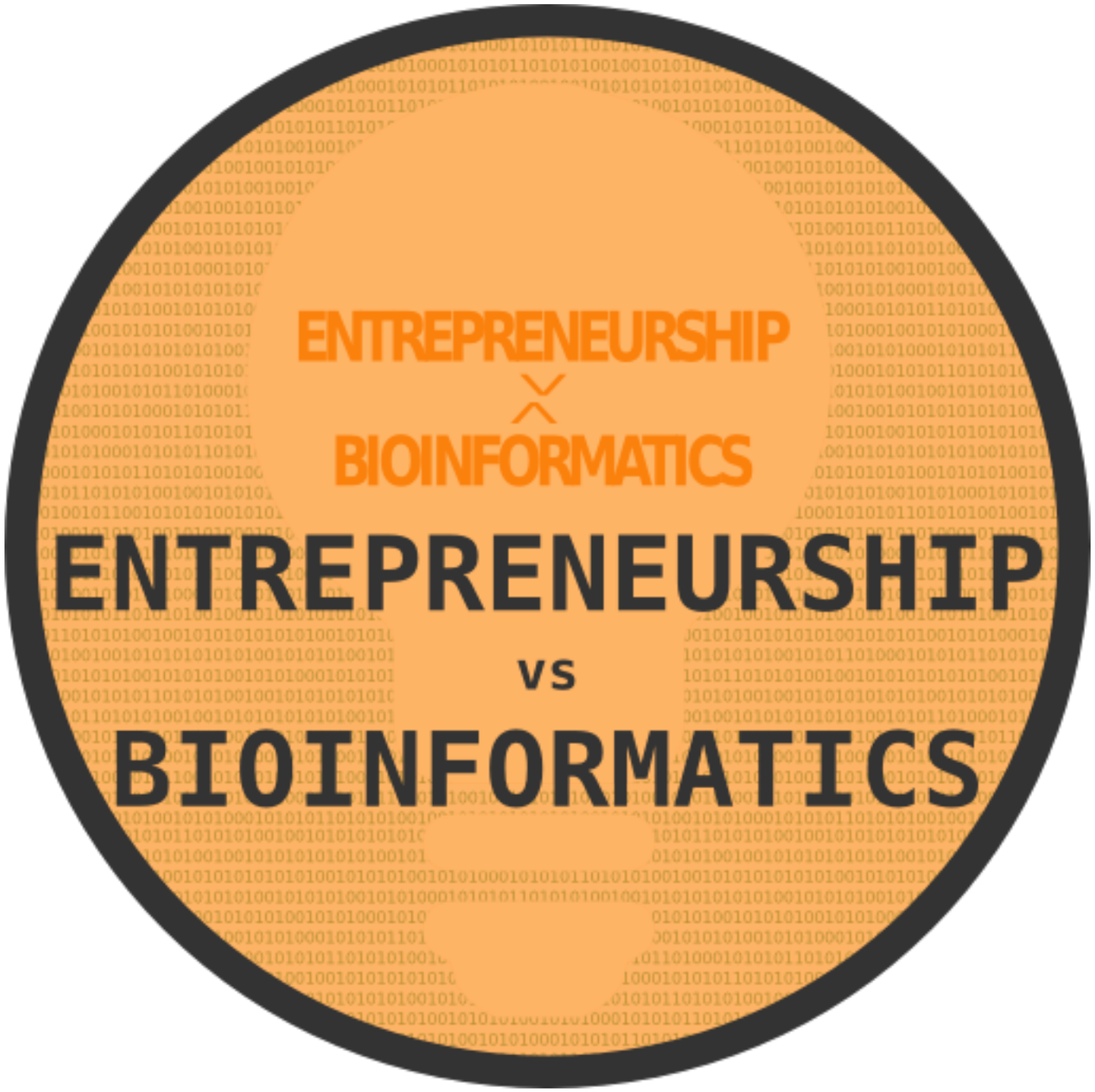
Type less and do more at the UNIX shell

Helen Cook
October, 2015

CBioVikings

Upcoming events

CBioVikings



CBioVikings

October 10

Sign up today at www.CBioVikings.org!

Open Source Event Planning

Tuesday Oct 13, 17:00
Studenterhuset

or email:
helen.cook@cbiovikings.org

CBioVikings

Next Viking Thursday

Nov 5, 2015

Studenterhuset

CBioVikings

Cancer Event November

email:
jessica.hu@cbiovikings.org

CBioVikings

Food Event December

email:
joe.blair@cbiovikings.org

CBioVikings

Elections

CBioVikings

Exercises, presentation and configuration files at:

http://github.com/bitmask/type-less

Install git

http://desktop.github.com/

or

http://git-scm.com/downloads

Mac OS X

homebrew
package manager

http://brew.sh/

Mac OS X

brew install

colordiff
coreutils
git
grep
moreutils
parallel
pymol
tmux
wget

install iterm2

http://iterm2.com

Mac OS X

```
open .
```

Linux

apt-get install …

Windows

Options:

1. ssh to a server
2. cygwin

Getting help

… years to master but can perform seeming miracles in seconds …

-- a Bellevue Linux Users Group member, 2005

command -h

command --help

man command

Google

stackoverflow

Biostars
BIOINFORMATICS EXPLAINED

CBioVikings

More Resources

Books
Data Science at the Command Line
Bioinformatics Data Skills

Guides
https://github.com/jlevy/the-art-of-command-line

Shell
http://www.shellcheck.net/
http://explainshell.com/

Specific tools
http://www.grymoire.com/Unix/Sed.html
https://www.gnu.org/software/parallel/parallel_tutorial.html

~$

Laziness
Impatience
Hubris

A choice of shells

```
Path: senator-bedfellow.mit.edu!bloom-beacon.mit.edu!news
From: Tom Christiansen <tchrist@mox.perl.com>
Newsgroups: comp.unix.shell,comp.unix.questions,comp.unix
Subject: Csh Programming Considered Harmful
Followup-To: comp.unix.shell
Date: 6 Oct 1996 14:03:18 GMT
Organization: Perl Training and Consulting
Lines: 558
Approved: news-answers-request@MIT.Edu
Expires: Sun, 1 Dec 1996 12:00:00 GMT
Message-ID: <538e76$8uq$1@csnews.cs.colorado.edu>
NNTP-Posting-Host: perl.com
Originator: tchrist@mox.perl.com
Xref: senator-bedfellow.mit.edu comp.unix.shell:42768 com
```

zsh

Ask if you have questions

Shout loudly if you have a better solution

Your shell should
make your life easy

Exercise

```
mv *.fa.txt *.fa
```

These FASTA files are viral orthology groups from http://eggnogdb.embl.de/#/app/viruses

```
mv 1.fa.txt 1.fa
mv 2.fa.txt 2.fa
...
mv 32578.fa.txt 32578.fa
```

mmv *.fa.txt *.fa

~/.zshrc

```
alias mmv='noglob zmv -W'
```

```
cp config/zshrc ~/.zshrc
zsh

cd exercises/mv
mmv *.fa.txt *.txt
```

Bring back what I typed

~/.history

History == Lab book

History config

```
setopt EXTENDED_HISTORY
export SAVEHIST=9999999
export HISTSIZE=$SAVEHIST
setopt HIST_IGNORE_ALL_DUPS
setopt INC_APPEND_HISTORY
setopt SHARE_HISTORY
```

How do you access the history?

up-arrow

ctrl-R

~/.zshrc

```
bindkey "^_" insert-last-word
```

Don’t type it
in the first place

<tab>

Programmable tab completion

git<tab>
git co<tab>
git commit <tab>
kill <tab>
ssh <tab>

ls **/EXERCISE

cd e/f<tab>

exercises/fasta

cat mt.u<tab>.fasta

cd ~

cd -

dirs

Exercise

Are two files the same?

exercises/same

cat
sort
wc -l
ls -l
md5

diff

A1 & A2

A1 & A2

A2 has an extra line

B1 & B2

B1 & B2

One letter change in id

C1 & C2

C1 & C2

C1 has DOS line endings

Ignore whitespace

diff -w -B -E

D1 & D2

exercises/same

D1 & D2

Line order is different

Streams

command < in > out

command < in > out

command < in > out

STDIN STDOUT

```
command > out 2> err
```

STDERR

Exercise

Add rows of A1 and A2

These are gene expression levels from pulldown experiments under two conditions

A1 line 1
A1 line 2
A2 line 1
A2 line 2

exercises/comb

```
cat A1 A2 > A
```

cat A1 >> A2

Exercise

Add cols of A and B

B contains the common names of the genes in A

A line 1 B line 1

A line 2 B line 2

exercises/comb

Hint: paste

paste A B > AB

Exercise

Combine ids of A and C

C contains the common names of all yeast genes

First column of A and C contain the id that we will join on

id1 A rest of line 1 C rest of line 2
id2 A rest of line 2 C rest of line 1

exercises/comb

Hint: join

join A C > AC

join input **MUST** be sorted on the join column

```
sort -k 1 A > A.sorted
sort -k 1 C > C.sorted
join A.sorted C.sorted > AC
```

exercises/comb

```
join -1 1 -2 1 -o 0,1.2,1.3,2.2 A C > AC
```

(A and C are sorted)

Pipes

command > file

command1 | command2

command1 | command2

STDOUT

STDIN

Exercise

Print unique lines

goterms contains the GO terms associated with all proteins in influenza virus

Hint: uniq

```
uniq goterms > out
```

uniq only uniqs
adjacent rows

exercises/uniq

```
sort goterms | uniq > out
```

exercises/uniq

```
cat goterms | sort | uniq > out
```

Useless use of cat

```
sort -u goterms > out
```

Useful switches to sort

-n numerically
-k on column k
-r reverse order
-u unique

Exercise

Most frequent GO term

exercises/uniq

```
sort goterms | uniq -c | sort -n
```

~/.zshrc

```
alias hist='sort | uniq -c | sort -n'
```

exercises/uniq

```
cat goterms | hist
```

Exercise

Sort a file in place

gcn contains the gene copy number of yeast tRNAs

sort gcn > gcn

```
cp gcn.backup gcn
```

exercises/sort

```
sort gcn > tmp && mv tmp gcn
```

sort gcn | tee gcn

Exercise

Sort on the 2nd column, numerically

sort -k 2 -n gcn

Cut data out of a file

Exercise

Print the 7th row

pep contains (a small part of) the output from a mass spec experiment in yeast

exercises/cut

```
head -7 pep | tail -1
```

```
awk 'NR == 7' pep
```

```
cat pep | nl | grep -w 7
```

Exercise

Print 10 random lines from a file

Hint: shuf

(brew install coreutils)

exercises/cut

shuf pep | head

```
sort -R pep | head
```

Mac OS X tools are based on BSD, not GNU

Free BSD

~=

Darwin
(Mac OS X)

!=

GNU

brew install coreutils

```
alias sort='gsort'
alias ls='gls'
alias grep='ggrep'
alias shuf='gshuf'
...
```

Exercise

Print 10 random lines from a file in order

exercises/cut

Hint: nl

exercises/cut

```
cat pep | nl | shuf | head | sort -n | cut -f2-
```

exercises/cut

```
alias sam='nl | shuf | head | sort -n | cut -f2-'
```

cat pep | sam

Exercise

Print the gene names in the 2nd column

cut -f2 pep

```
awk '{print $2}' pep
```

Exercise

Print the 1st column if the 2nd column starts with Q

exercises/cut

```
grep '\sQ' pep | cut -f1
```

Regular expressions

^ beginning of line
$ end of line
\s white space

http://www.grymoire.com/Unix/Regular.html

exercises/cut

```
awk '$2 ~ /^Q/ {print $1}' pep
```

Exercise

Find sum of 3rd column

exercises/cut

```
awk '{c+=$3} END {print c}' pep
```

Exercise

How many entries are there in a FASTA file?

mt.fasta contains the yeast mitochondrial genes

```
grep ">" mt.fasta | wc -l
```

Exercise

Generate a FASTA file with only the verified ids

```
grep -f verified mt.fasta
```

```
grep -f verified -A 1
--no-group-separator
mt.fasta > verified.fasta
```

Exercise

Find the log2 fold change of column 2 to column 3

Hints:

1. Get rid of the header

2. Filter out rows that div by 0

Get rid of the header:

```
grep -v gene file
```

Get rid of the header:

```
sed 1d file
```

Filter div by 0:

```
awk '$3 > 0'
```

```
cat file | sed 1d |
awk '$3 > 0 {print log($2/$3)/log(2)}'
```

Controlling processes

How do you stop a running process?

ctrl-C

… pause a running process?

ctrl-Z

… start it again?

fg

… start it, but in the background?

bg

… see the processes
that are running?

ps

top

grep pattern
ctrl-Z
bg
ps
fg
ctrl-Z
top
fg
ctrl-C
ps

… kill a process if it’s not in the fg?

```
PID   TTY          TIME     CMD
10269 ttys007      0:00.00 ggrep --color=auto pattern
```

```
kill <PID>
```

```
ps a | grep grep |
awk '{print $1}' | xargs kill
```

pgrep grep

pkill grep

grep pattern
ctrl-Z
bg
ps
kill <pid>
ps

grep pattern
ctrl-Z
bg
pgrep grep
pkill grep

… wait for it to finish,
then run another cmd?

ctrl-Z
bg
wait && cmd

Working with many files at once

How do you create an archive of files?

exercises/mv

```
tar cf files.tar *.fa
```

Useful switches to tar

-c create

-x extract

-z gzip

Argument list too long

```
find . -name '*.fa' |
xargs tar cf files.tar
```

```
for i in *; do
echo $i;
...
done
```

parallel

https://www.gnu.org/software/parallel/parallel_tutorial.html

```
find . | parallel cmd {}
```

Working remotely

How do you ssh without typing your password?

ssh keys

```
ssh-keygen -t rsa -b 4096
scp ~/.ssh/id_rsa.pub you@server:

cat ~/id_rsa.pub >> ~/.ssh/authorized_keys
```

https://wiki.archlinux.org/index.php/SSH_keys

How do you work with files like they are local?

sshfs

Linux: http://fuse.sourceforge.net/sshfs.html

OSX: https://github.com/osxfuse/osxfuse/wiki/SSHFS

How do you keep context when connections drop?

screen

or

tmux

Other niceties

Set up your prompt

~/.zshrc

$PS1 =

Summary

>	diff	mmv	shuf
<	echo	nl	sort
>>	fg	parallel	ssh
|	for	paste	tail
awk	grep	pgrep	tar
bg	head	pkill	tmux
cat	join	ps	uniq
ctrl-C	kill	scp	wait
ctrl-V	ls	screen	wc
cut	md5	sed	zsh

Your shell should make your life easy…

…but it will take
some configuration…

…and knowledge of the available tools.

There is more than one way to do it

Get involved

CBioVikings

The end