

Getting the most out of the workshop

- Ask questions!!!
- Don't sit next to someone you already know
- Work with someone with a different skillset and different experience level
- Use the workshop laptop
 - You will have access to your files after you leave
- Come to the social functions
- Ask questions!!!

A first look at the operating system and R-studio...



Sarah Medland



Morning sessions

- Optional
 - Feel free to wander in and out/check email etc
- Topics
 - Shift in response to feedback
 - Tomorrow: OpenMx code and concepts, assumption testing and univariate modeling

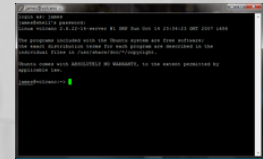
This year's OS

- Debian (linux)
 - Free
 - Many free software packages available
 - Open office
 - R
 - PSPP
 - Terminal
- Based on Unix
 - long and venerable history
 - <http://en.wikipedia.org/wiki/Unix>



Close but not the same...

- Most basic shortcuts will work
 - ctrl+C for copy ctrl+V for paste etc
- Supports folder based navigation
- \backslash BIG PROBLEM is \backslash vs $/$
- You will have used some version of unix previously



File hygiene is very important

- Files are stored in Unix format not DOS or Mac
 - Changes the line ending characters
 - Use `dos2unix`, `unix2dos`, `mac2unix`, `unix2mac` to change formats
 - Can use the `file` command to check format
- Unix systems are case sensitive!
- NO SPACES in your file/directory names!!
- Wildcards ie `dos2unix *.dat`

Working in the terminal

Input Output

- Input

- Most commands don't need input signifiers
- < can be used to specify

- Output

- Without specifying most output will print to the screen
- > can be used to direct

- type: echo 'this is a dummy file'

- echo 'this is a dummy file' > dummy.txt

- | (pipe) | more pauses the output after a screen worth of text has appeared hit the space bar to get the next screens worth

The manual

- The man command can be used in conjunction with other commands to put up some basic instructions
- type: man ls
 - ls is the list command it pulls up a list of the files in the directory

Also many many helpful webpages w examples

Permissions

the ability to read, write and execute files

- type: `ls -l`

```
Integlio@Lapis /cygdrive/c/wedtemp
$ ls -l
total 32
-rw-r--r-- 1 Integlio mkpasswd 21 Mar  4 13:25 dummy.txt
```



- These are the permissions
- 1st a directory flag (d or -)
- then 3 letters to define the owners permissions
- 3 letters to define the groups permissions
- 3 letters to define the everyone else's permissions

Permissions

the ability to read, write and execute files

- read access
- **w**rite access
- **x**ecute
 - to 'run' script or a program the file must be made executable

Permissions

the ability to **read**, **w**rite and **ex**ecute files

- To change the mode/permissions use `chmod`
 - a number of ways to do this
 - **type:** `echo "this is a test" > dummy.txt`
 - `ls -l`
 - `chmod +x dummy.txt`
 - `ls -l`
 - `chmod -x dummy.txt`
 - `ls -l`
 - **what happened?**

Useful 'one liners'

- cp copy
- mv move = rename
- rm remove
- ls list
- echo
- head looks at the top 10 lines
- tail looks at the last 10 lines
- wc counts number of lines, words, characters
- sed find and replace
- grep find and report
- awk restructure files

Grep

- search **g**lobally for lines matching the **r**egular **e**xpression, and **p**rint them
 - For association output for chromosome 2
 - To extract the result for snp rs59831
 - Type: `grep 'rs59831' output.txt > summary.txt`

Grep

- Useful flags
 - -v
 - reverse grep select line that does not have the pattern
 - -C x
 - To x rows before and after the target
 - -n
 - Print the line number before the line
 - Many more...

Awk

- derived from the surnames of its authors — Alfred **A**ho, Peter **W**einberger, and Brian **K**ernighan
- Many functions
- Very useful for restructuring data

Awk

- Ozbmi2.rec

115	0	0.21	1	2	58	57	1.7	1.7	20.0692	19.7232	20.9943	20.8726
121	0	0.24	1	2	54	53	1.6299	1.6299	20.3244	19.9481	21.0828	20.9519
158	0	0.21	1	2	55	50	1.6499	1.6799	20.202	17.7154	21.0405	20.121
172	0	0.21	1	2	66	76	1.5698	1.6499	26.7759	27.9155	23.0125	23.3043
182	0	0.19	1	2	50	48	1.6099	1.6299	19.2894	18.0662	20.7169	20.2583
199	0	0.26	1	2	60	60	1.5999	1.5698	23.4375	24.3418	22.0804	22.3454
221	0	0.23	1	2	65	65	1.75	1.7698	21.2245	20.7476	21.3861	21.227
239	0	0.29	1	2	40	39	1.5598	1.5298	16.4366	16.6603	19.5966	19.6912
246	0	0.24	1	2	60	57	1.7598	1.7698	19.3698	18.194	20.746	20.3076

- awk '{ print \$1, \$10, \$11, \$4, \$5 }' ozbmi2.rec >
new.rec

115	20.0692	19.7232	1	2
121	20.3244	19.9481	1	2
158	20.202	17.7154	1	2
172	26.7759	27.9155	1	2
182	19.2894	18.0662	1	2
199	23.4375	24.3418	1	2
221	21.2245	20.7476	1	2
239	16.4366	16.6603	1	2
246	19.3698	18.194	1	2

Awk

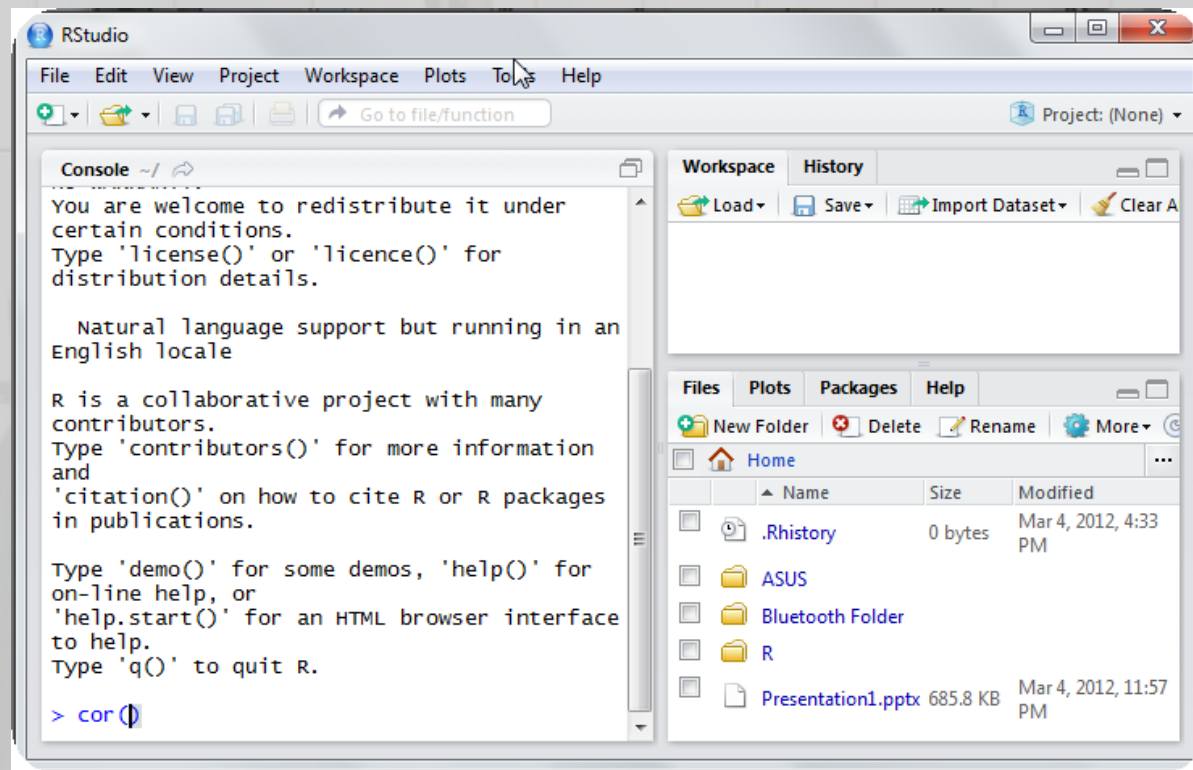
- `$1` = column 1
- Print `$0` = print whole line
- add subtract multiply etc
- change number of decimals
- Many functions

Sort

- Useful flags
 - -f ignore case
 - -n numeric sort
 - -r reverse
 - -c check if a file is sorted
 - -u prints only unique lines
 - -k2 sort starting at column2

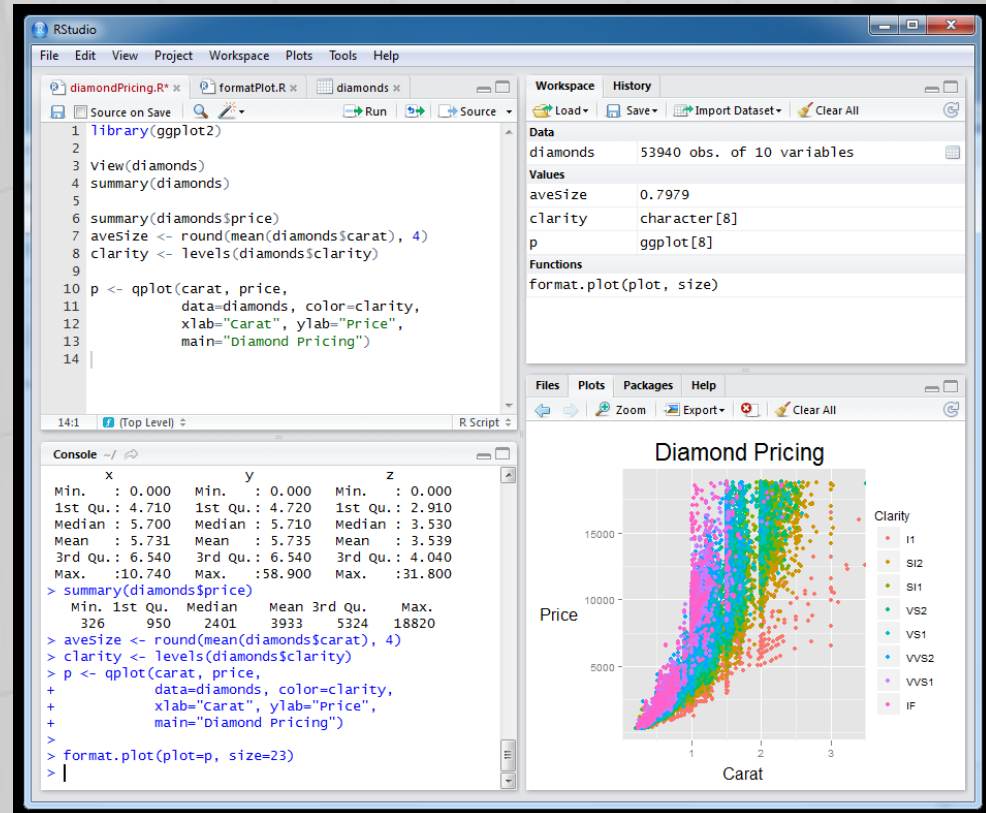
Using R this week


- R-studio <http://rstudio.org/>



Setting this up at home

- Install R first
- Install R studio
- Install packages



A photograph of a modern building's curved glass facade. The glass reflects the surrounding cityscape, including several tall, slender skyscrapers. The building's structure is composed of a grid of dark metal frames holding the glass panels. The overall scene is in grayscale, giving it a classic, architectural feel. In the upper right corner, there is a white rounded rectangular box containing the word "Vienna".

Vienna