



Using the right tools for doing right research

Yvan Richard

Dragonfly Science
Level 5, 158 Victoria St, Wellington

yvan@dragonfly.co.nz

26th April 2012



New data to be incorporated, or mistake in source data

- ➊ Add data to spreadsheet or correct mistake
- ➋ Re-run analyses, involving thousands of clicks
- ➌ Update figures, with a lot of frustration about formatting
- ➍ Update tables, endless copy & paste operations
- ➎ Update numbers in report – Have you missed one?

Waste of time, stressful moment, ending up with a vague feeling of having forgotten to update something...



Typical case #2

Accidental deletion or bad direction

- You realise at some stage that you deleted some data or some text a while ago and that you kept working in a wrong direction.
- You wish you could go back in time to a previous version of your work
- The deleted part might be lost forever, or at least requires to re-do everything since the mistake.

Waste of time and frustration! But you learn from your mistake, and since then, you keep a different version of everything every time you change something.



Typical case #3

Coming back to an old project

- The review of a submitted paper comes back and requires you to do additional analyses
- You open your project folder only to discover 20-odd different versions of the data
- Which did you use?
- After being fairly confident of the correct one to use, you re-do the analyses, and find different results!
- Eventually, you manage by trial and error to find the same results and keep going.

Waste of time, ending up unsatisfied and not very confident about the results.



Typical case #4

Changing computer

- After spending a lot of time perfecting the formatting of your thesis, presentation, or article, you realise your document looks quite different on another computer

Panic and frustration...



Typical case #5

Changing institution

- After finally getting into grip with a given software, you change institution and you realise they use a different software. Unfortunately, you cannot get your favourite one because the license is too expensive.

Waste of time re-learning a new software, never becoming a master at a given one



Typical case #6

Auditing

- You work on a sensitive issue, e.g. an endangered species at risk of a proposed development project.
- The “bad guys” do not really like your results, preventing their project to be approved.
- In the environmental court, it is agreed that your project gets audited, requiring you to show that you get the same results from the same data.

... Can you?



Typical case #7

Teaching

- A workmate asks you to show him/her how you do a certain analysis.
- You sit at a computer with him/her, and start explaining: “You go to this menu, and click there, then there, then you go there and type in that, and then you click on this, etc.”
- It takes quite a while, as your workmate writes down the whole complicated process.
- He/she gets back to you later, because his/her version of the program is slightly different and he/she cannot find a certain item in the menus.

... (sigh) ...



Let's dream that...

- all these problems could be solved,
- you could focus on content and process rather than formatting and eye-candy,
- all the tools to solve these problems and do proper research are free,
- on the way of solving these problems, you acquire great skills that would make you find a job very easily.

Well, yes, it is possible!



Research

Research should:

- be reproducible
- be transparent
- have a functional work flow

Sounds trivial, but it is rarely the case!




Open-source software

- Free!
- Generally quite portable between operating systems
- Huge community for support, bug checking and fixing, for new developments
- Transparent with a non-restrictive license, allowing easy communications between programs.



Main functions

- Data preparation, exploration, analysis, and plotting
- Reporting
- Bibliography
- Work flow
- Version control
- Distribution



Data preparation, exploration, analysis and plotting



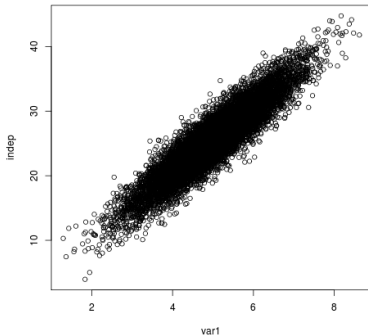
- NZ product!
- Software environment for statistical computing and graphics
- Programming language, but easy to learn
- Works on all systems (Windows, Mac, Linux, ...)
- Increasing popularity, real threat to commercial products (e.g. SAS, SPSS)
- Evolves fast, expandable with thousands of available packages

Example

```
## Load data
dat <- read.csv("file-with-data.csv")

## Data manipulation
dat$var3 <- dat$var1 + dat$var2

## Plot
plot(var1, var2)
```



Fitting a linear model

```
mod <- lm(dep ~ indep1 + indep2)
summary(mod)
```

Call:

```
lm(formula = indep ~ var1 + var2)
```

Residuals:

Min	1Q	Median	3Q	Max
-7.8060	-1.3795	-0.0133	1.3950	8.2858

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.21056	0.23224	0.907	0.3646
var1	4.95934	0.02074	239.112	<2e-16 ***
var2	0.04883	0.02062	2.368	0.0179 *

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.083 on 9997 degrees of freedom

Multiple R-squared: 0.8512, Adjusted R-squared: 0.8511

F-statistic: 2.859e+04 on 2 and 9997 DF, p-value: < 2.2e-16

R: advanced graphics

London Cycle Hire Journeys

Thicker, yellow lines mean more journeys





Scripting power

- Everything is written, no lost clicks
- Reproducible
- Easily changed
- Code is re-usable
- Repetitive tasks are done using loops
- Generally quicker than clicks and navigating menus



L^AT_EX

- Compiled documents, not WYSIWYG
- Very common
- Most scientific journals provide their own template
- Beautiful typesetting
- Takes care of formatting automatically
- Maths formulae are easy to write
- Easy PDF creation with pdf_latex
- Creation of presentations using Beamer (like this one)

the stochastic population growth rate (λ_s) may be found via simulation using the following formula \citep{caswell_matrix_2001}

$$\lambda_s = \exp \left(\frac{1}{T} \left(\ln(N_T) - \ln(N_0) \right) \right)$$

with N_i being the population size at time i , and T the number of time steps in the model. This formula is simply the geometric mean of the population growth rate at each time step ($\lambda_{t \rightarrow t+1} = N_{t+1}/N_t$). Similarly, the number of survivors after T time

the stochastic population growth rate (λ_s) may be found via simulation using the following formula (Caswell, 2001)

$$\lambda_s = \exp \left(\frac{1}{T} (\ln(N_T) - \ln(N_0)) \right)$$

with N_i being the population size at time i , and T the number of time steps in the model. This formula is simply the geometric mean of the population growth rate at each time step ($\lambda_{t \rightarrow t+1} = N_{t+1}/N_t$). Similarly, the



Calling R into LaTeX using Sweave

- Sweave lets data and tables from R to be included in LaTeX documents
- Why copying and pasting data manually when you can call them directly?
- Enormous time saver
- Chances of mistakes are minimal
- Initial data can be changed, the changes will be automatically reflected in the report




Calling R into LaTeX using Sweave

Example

```
\SweaveOpts{echo=FALSE, results=tex, prefix.string=sweave/fig}
```

```
<<load>>=  
dat <- read.csv("file-with-data.csv")  
minsize <- min(dat$popsize)  
maxsize <- max(dat$popsize)  
@
```

The population size varied between `\Sexpr{minsize}` and `\Sexpr{maxsize}`.



Bibliography in LaTeX

Including references is easy with BibTeX! References are stored in a text file (e.g.: refs.bib):

```
@article{richard_cost_2010,  
  title = "Cost distance modelling of landscape connectivity  
          and gap-crossing ability using radio-tracking data",  
  volume = "47",  
  number = "3",  
  journal = "Journal of Applied Ecology",  
  author = "Richard, Yvan and Armstrong, Doug P",  
  year = "2010",  
  pages = "603--610",  
  },
```

Then each reference is called in the LaTeX document by its tag:

```
... is a powerful tool to analyse movements \cite{richard_cost_2010}.
```



Bibliography in LaTeX

- BibTex format is very common
- References in this format can be downloaded from Google Scholar, imported from Zotero, and from journals web site
- Templates exist for all journals
- No more corrupted EndNote databases...



Workflow management



- GNU make
- Centralise jobs to be run
- Jobs are run in order, and only if necessary
- Jobs can be run in parallel in order to use several computer processors
- Can be used to document the whole workflow.

Workflow management

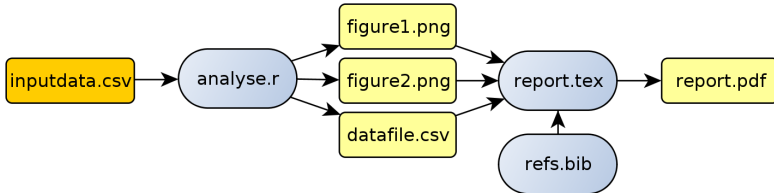
The jobs are written in a text file (makefile):


```
all:  report.pdf
```

```
report.pdf:  report.tex datafile.csv refs.bib  
             bibtex report  
             pdflatex report
```

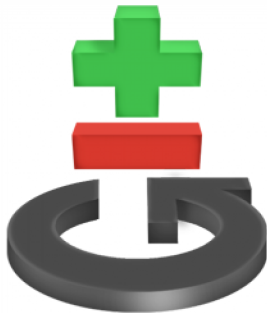
```
datafile.csv: analyse.r inputdata.csv  
              Rscript analyse.r
```

You run the whole process by only typing “make” in a terminal, it’s that easy.





Version control



- Saves all gradual changes of files
- Allows to safely keep only one version of each file
- Do not be afraid to delete stuff! You can always come back to previous versions
- Provides an easy outlook of all modifications
- Utilities to compare versions
- Great also for cooperative work



Workflow management

Easy commands:

`git status`: to get list of all modified files

`git add .`: to inform git to save all modifications

`git commit -m "Finished intro of chapter 3"`: to save locally the current state of modifications, with a comment to describe the changes

`git push`: to save the commits to the server

github
SOCIAL CODING



- GitHub is a web interface and service to store your git project
- Makes it easy to access your project from anywhere and to share it with others
- Free for open-source projects
- Great for issue tracking (to-do list)



Conclusions

- Great suite of tools for doing proper research, and they are all free!
- Risk of mistakes minimised
- Transparent and reproducible
- Fun! Just like playing Lego
- Adopting only one of these tools even is a great improvement over the traditional bad habits
- This workflow allows tackling some large projects comfortably that would be impossible otherwise
- These skills will help you all your life and make your life easier, and are great to get a job

But...

- It looks scary at first
- Big learning curve

But still worth it 100%!