Introduction to R

a language and environment for statistical computing and graphics

Ben R. Fitzpatrick
PhD Candidate, Statistical Science, Mathematical Sciences School, Queensland University of Technology

0000-0003-1916-0939 github.com/brfitzpatrick/ @benrfitzpatrick

- Please sit in groups
- Anyone who has used R before please spread yourselves around 1 per group
- so I can help you all more efficiently could the people using MacOS please sit together and the people using Windows also please site together

What is R?

- R is a language and environment for statistical computing and graphics
- R similar to the S language and environment
- R is Free and Open Source Software
- R will compile and run on most popular operating systems e.g. MS Windows, MacOS, UNIX, FreeBSD & GNU+Linux

Why Use R?

R is Free in the Sense of Free Speach & Free Beer

- chance are if you have a 'modern' computer you will be able to install R on it
- unlike software with paid licensing models you can use R anywhere free of charge
- you are free to modify and extend R provided you acknowledge the contributions of those who have gone before you

Why Use R?

R is Popular with a Large & Steadily Growing User Base

Subsequently packages have been written for R that implement a wide range of statistical analyses. Furthermore, more packages and functionality are continually being added and active forums exist on which to seek and find help.

CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

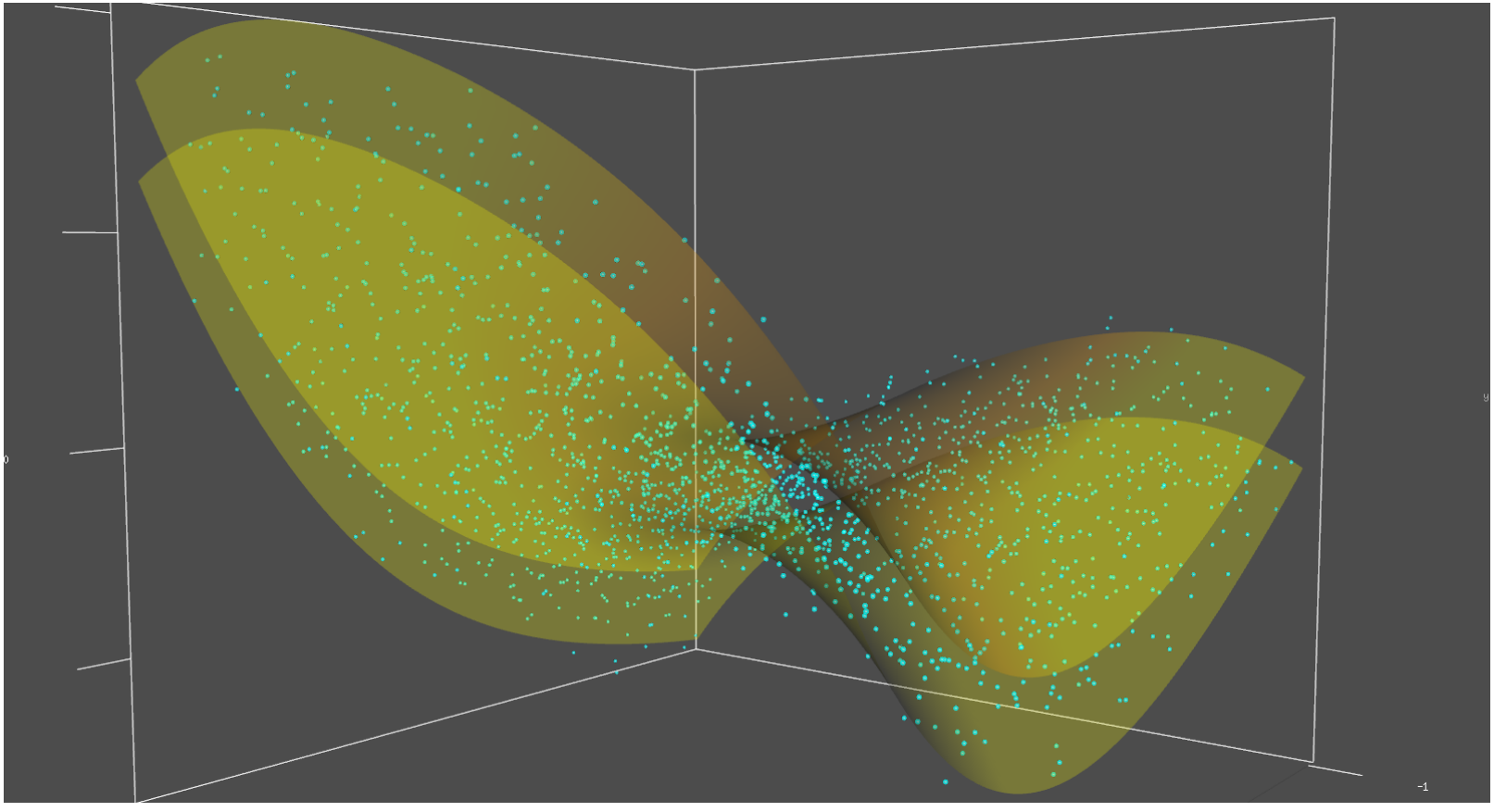
Documentation
Manuals
FAQs
Contributed

CRAN Task Views

Bayesian	Bayesian Inference
ChemPhys	Chemometrics and Computational Physics
ClinicalTrials	Clinical Trial Design, Monitoring, and Analysis
Cluster	Cluster Analysis & Finite Mixture Models
DifferentialEquations	Differential Equations
Distributions	Probability Distributions
Econometrics	Econometrics
Environmetrics	Analysis of Ecological and Environmental Data
ExperimentalDesign	Design of Experiments (DoE) & Analysis of Experimental Data
Finance	Empirical Finance
Genetics	Statistical Genetics
Graphics	Graphic Displays & Dynamic Graphics & Graphic Devices & Visualization
HighPerformanceComputing	High-Performance and Parallel Computing with R
MachineLearning	Machine Learning & Statistical Learning
MedicalImaging	Medical Image Analysis
MetaAnalysis	Meta-Analysis
Multivariate	Multivariate Statistics
NaturalLanguageProcessing	Natural Language Processing
NumericalMathematics	Numerical Mathematics
OfficialStatistics	Official Statistics & Survey Methodology
Optimization	Optimization and Mathematical Programming
Pharmacokinetics	Analysis of Pharmacokinetic Data
Phylogenetics	Phylogenetics, Especially Comparative Methods
Psychometrics	Psychometric Models and Methods
ReproducibleResearch	Reproducible Research
Robust	Robust Statistical Methods
SocialSciences	Statistics for the Social Sciences
Spatial	Analysis of Spatial Data
SpatioTemporal	Handling and Analyzing Spatio-Temporal Data
Survival	Survival Analysis
TimeSeries	Time Series Analysis
WebTechnologies	Web Technologies and Services
gR	gRaphical Models in R

Why Use R?

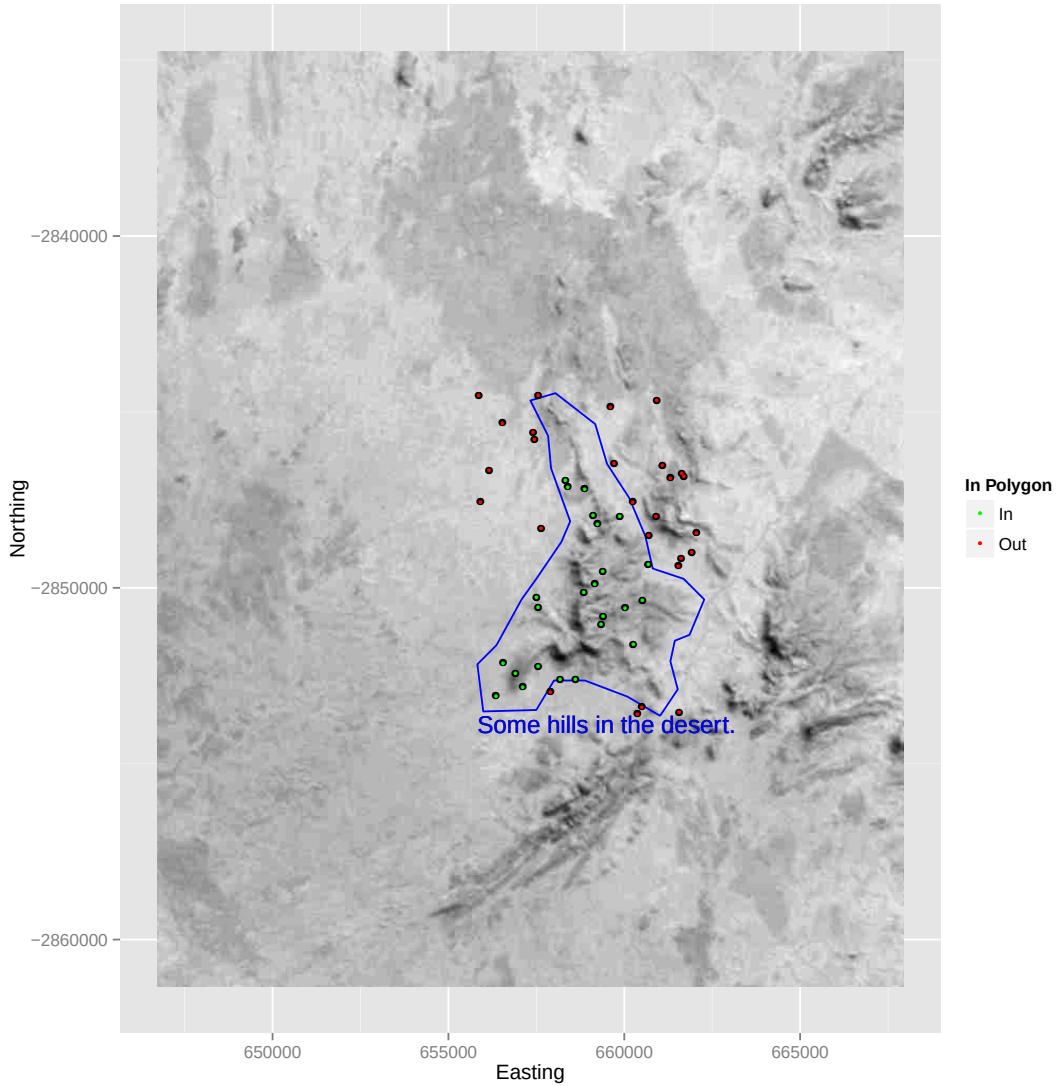
R has powerful graphics authoring capabilities



3D visualisation produced with the ‘rgl’ R package

Why Use R?

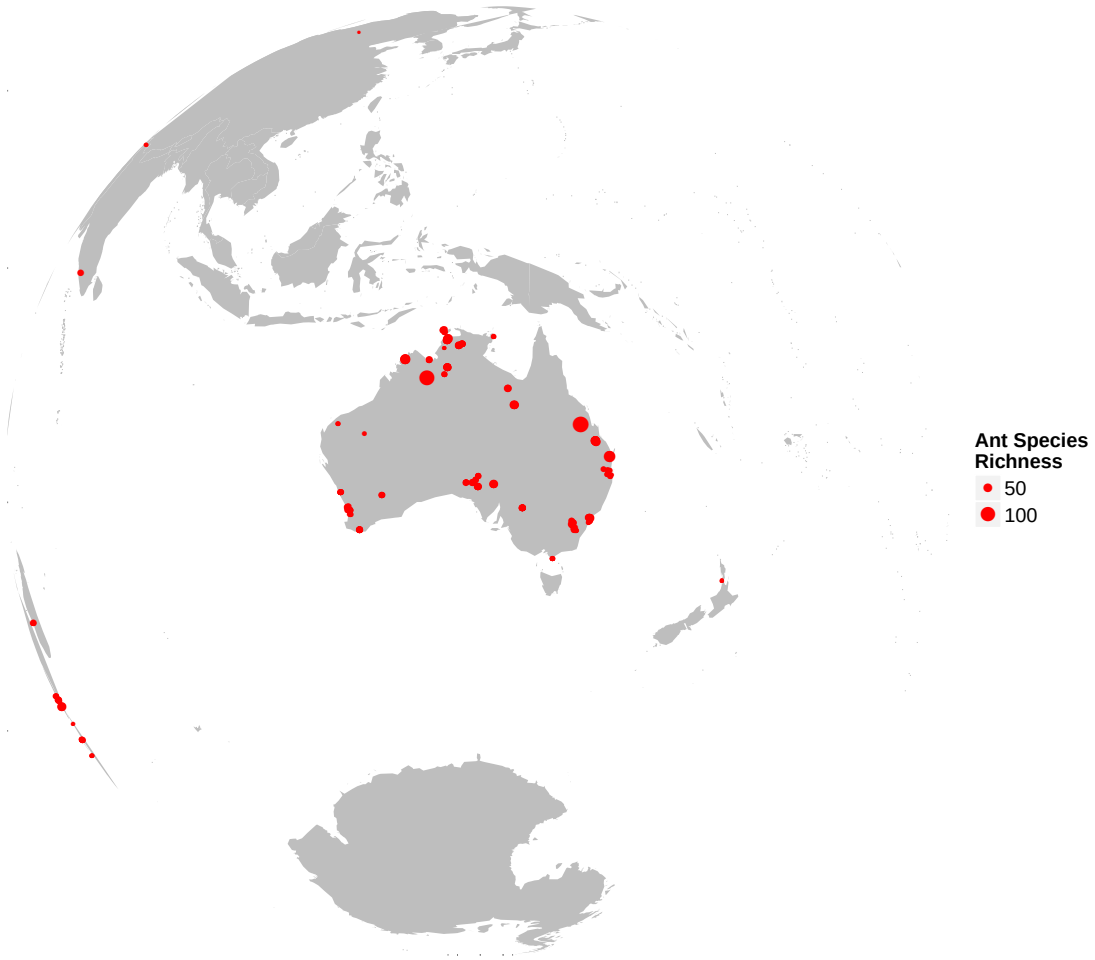
R has powerful graphics authoring capabilities



Geospatial Visualisation produced with the R packages ‘raster’ & ‘ggplot2’

Why Use R?

R has powerful graphics authoring capabilities

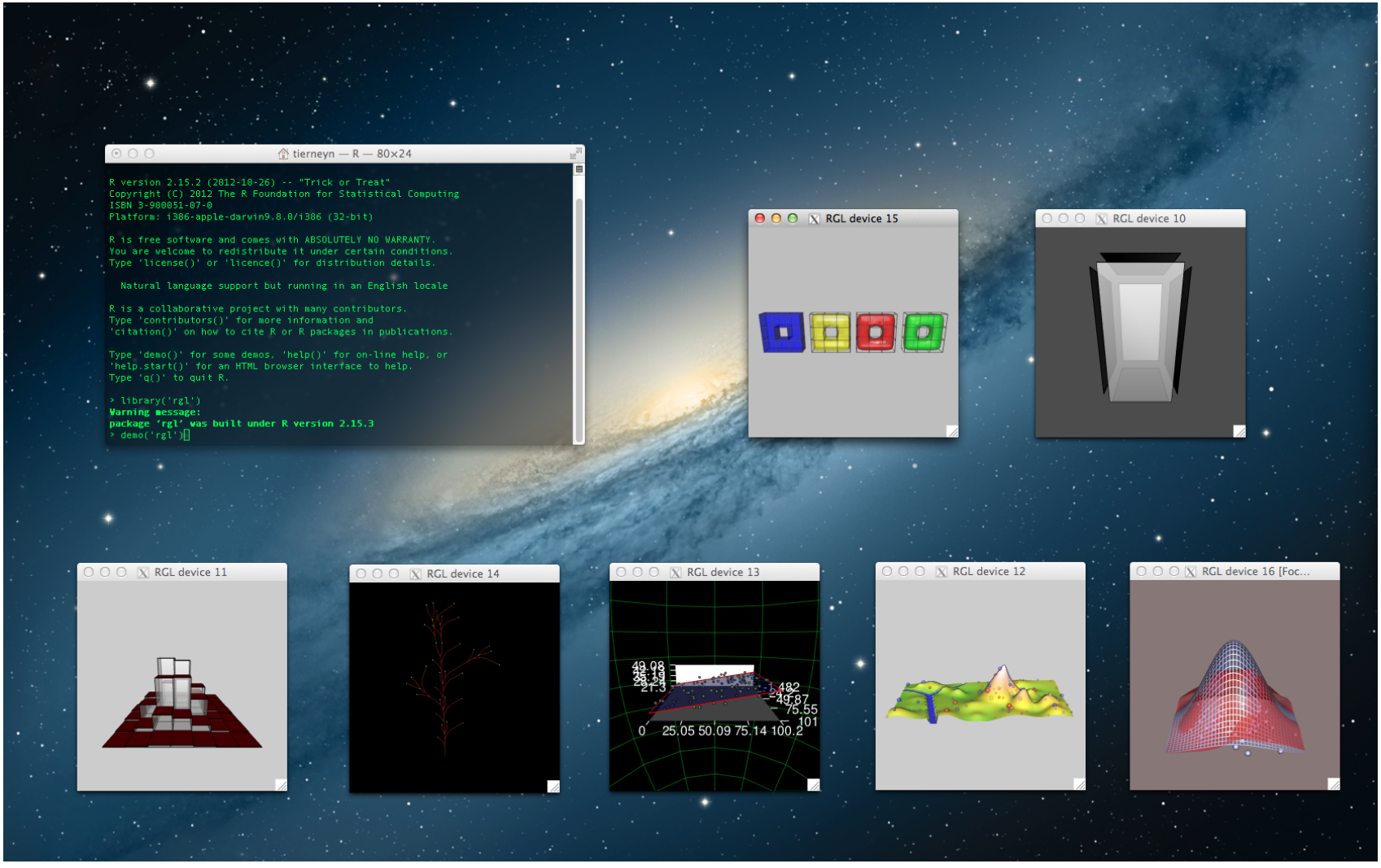


Geospatial Visualisation produced with the R packages ‘maps & ‘ggplot2’

Ways to Use R

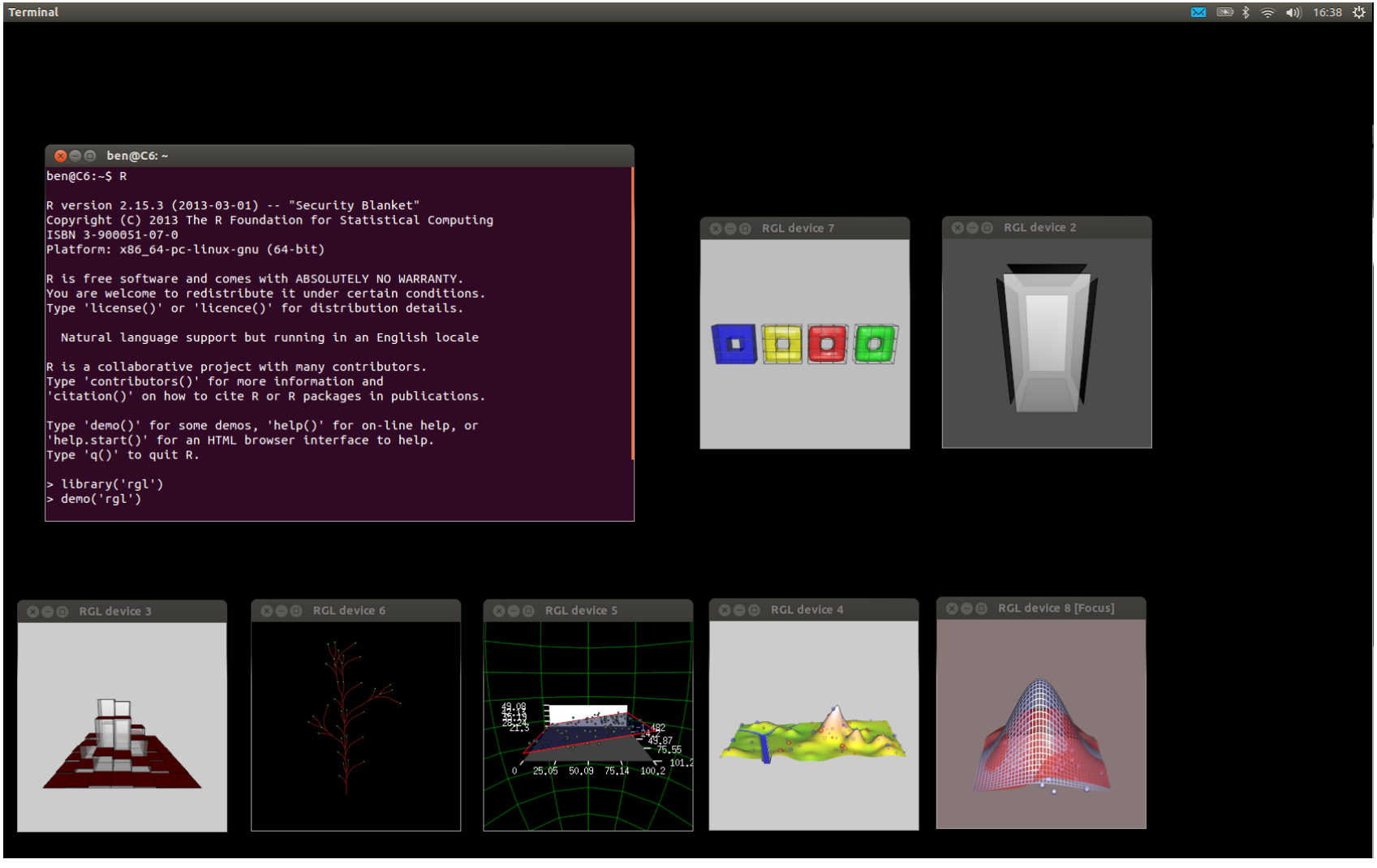
- via a command line interface e.g. PowerShell or Terminal
- via the default GUI clients for MS Windows & Mac OS
- via one of many Integrated Development Environments that either have been exclusively written for R or have R language modes e.g.
  - RStudio
  - Tinn-R
  - Sublime Text
  - Atom
  - Emacs Speaks Statistics
  - ...
- remotely i.e. submitting R scripts to a sever (e.g. HPC facility) to execute

Ways to Use R:
In a termial e.g. on Mac OS

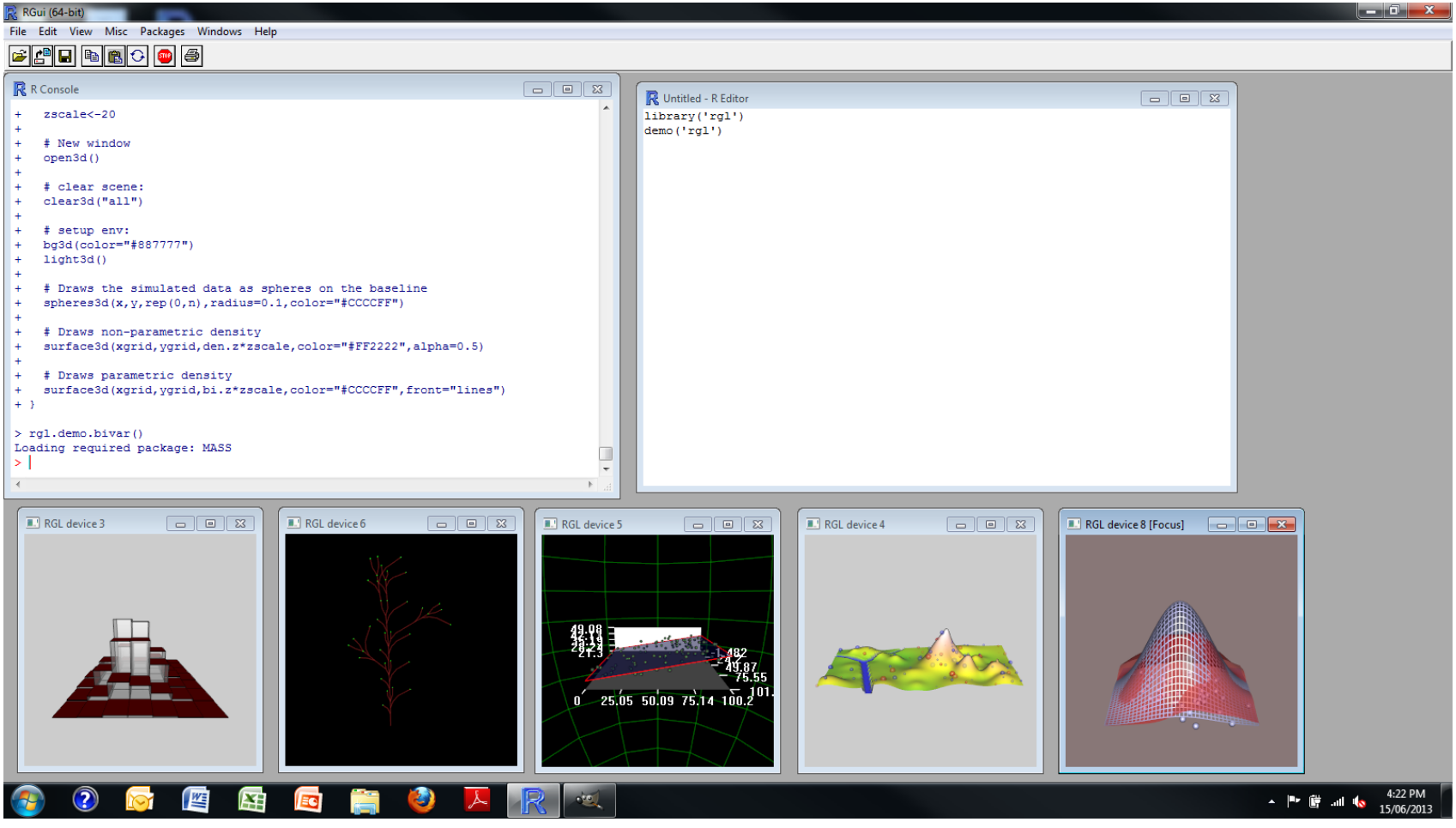


Ways to Use R:

In a termial e.g. on GNU+Linux

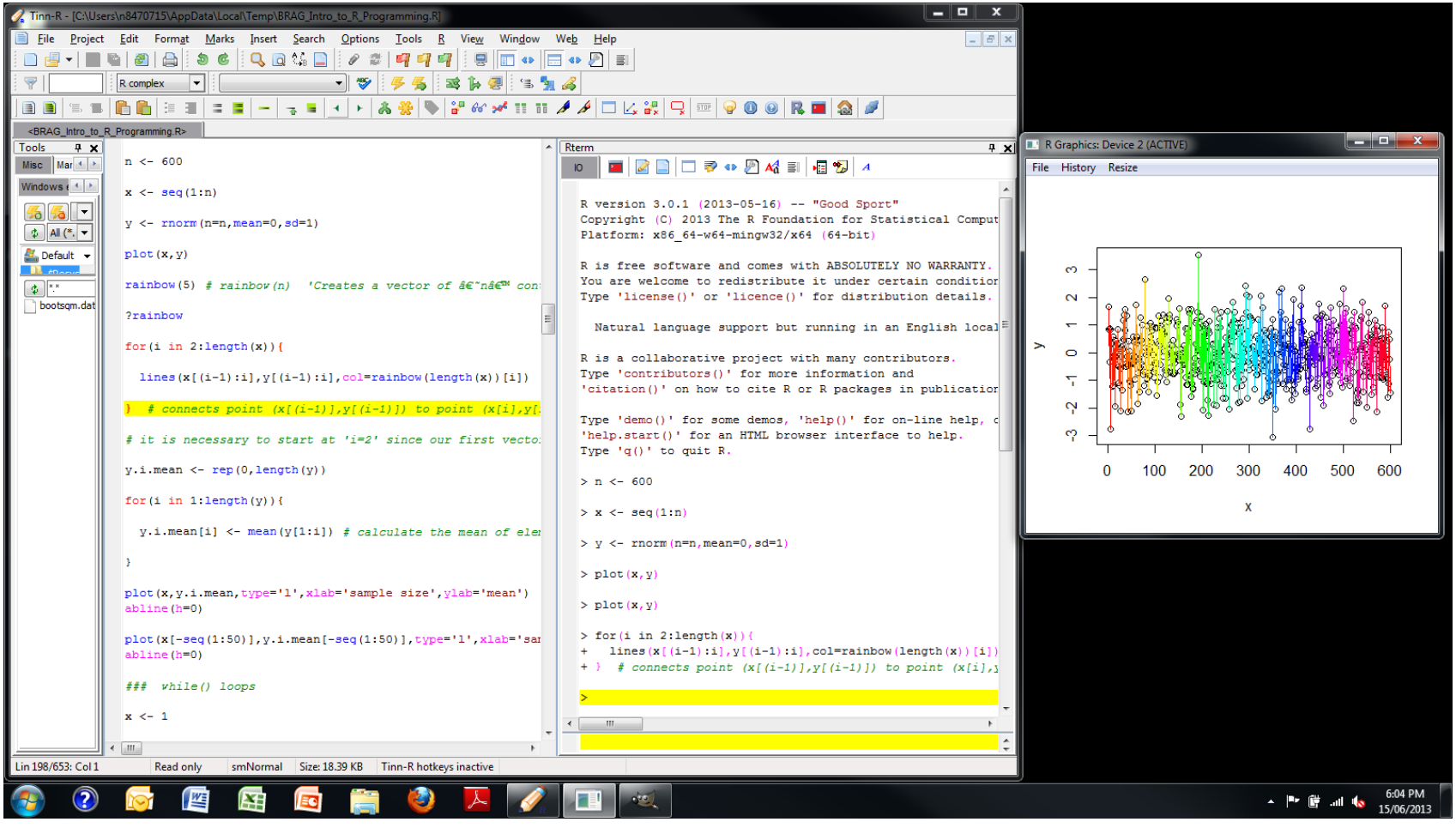


Ways to Use R:
Default Windows Client



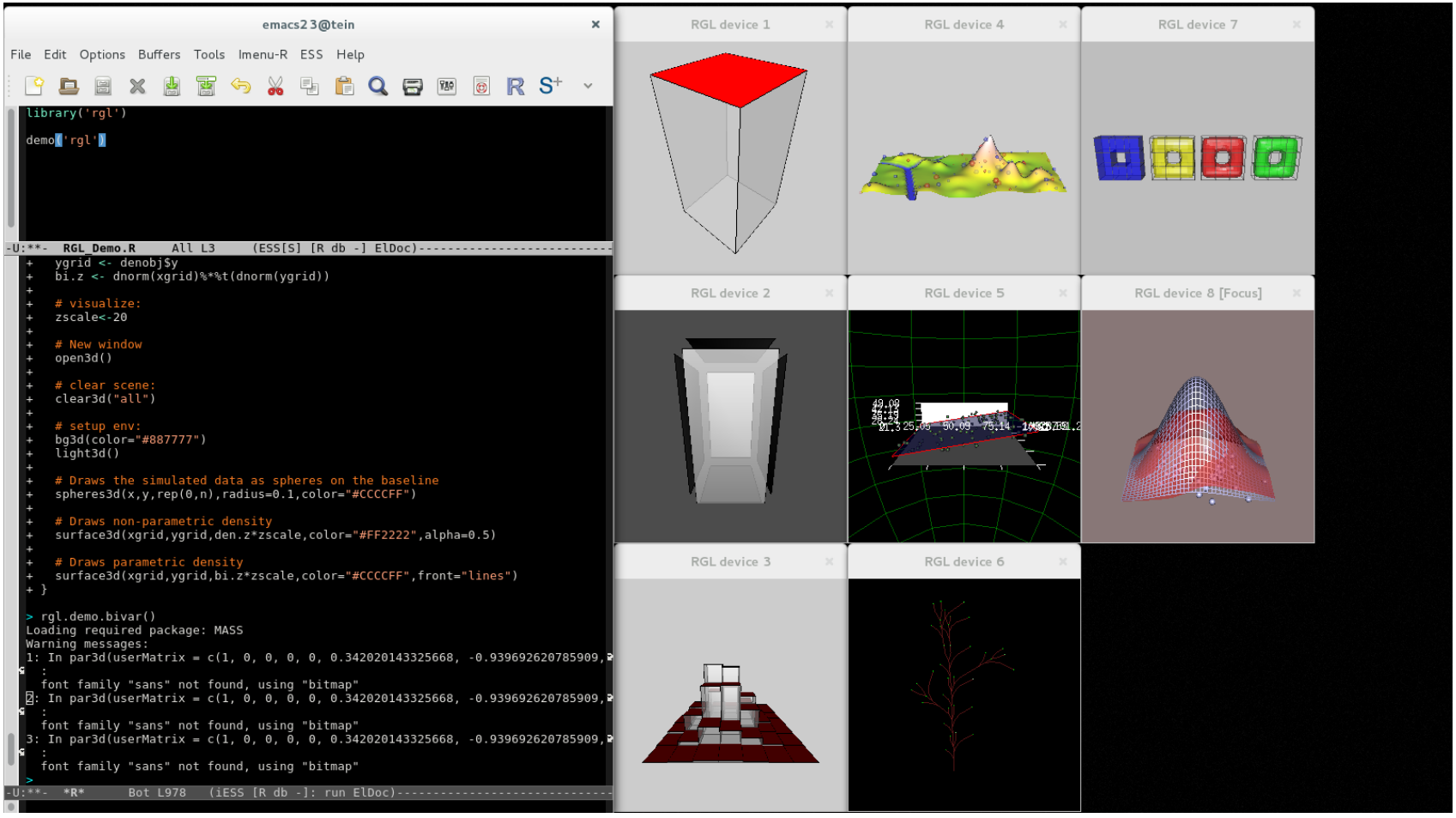
Ways to Use R:

Tinn-R Integrated Development Environement



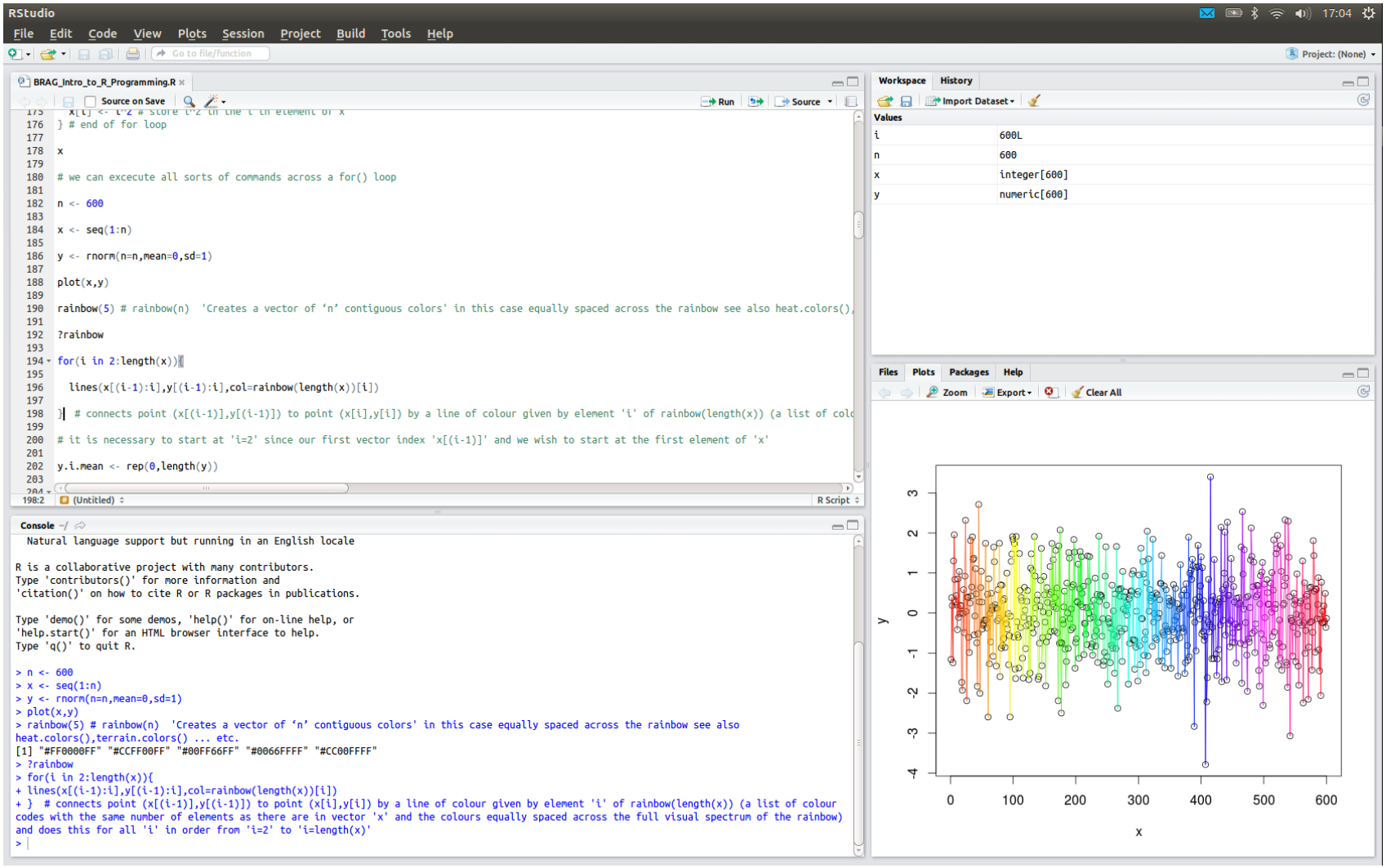
Ways to Use R:

Emacs Speaks Statistics Integrated Development Environement



Ways to Use R:

RStudio Integrated Development Environement



For this course I encourage you to use the RStudio IDE

Because:...

- it's comparatively intuitive and easy to learn
- feature rich
- available for most major operating systems (MS Windows, Mac OS, various flavours of GNU+Linux)

However, if you have already begun your journey learning R using a different IDE and wish to continue to use it please feel free to do so, provided you feel confident to open and execute .R files with this IDE.

The Plan

Feel free to use this time to pursue something that interests you

Course organised into 5 instructory modules and one extended, collaborative exercise.

Module:

1. Introduction to R & RStudio
2. Graphics with the R package 'ggplot2'
3. Linear Modelling in R
4. Programming in R
5. Version Control for solo & collaborative source code management with Git & GitHub
6. Capstone Collaborative Exercise

Key Learning Outcomes

Familiarisation with

- Command Line Computing
- RStudio Integrated Development Environment
- Commands and arguments
- Common Object Classes in R
- Assigning values to Objects
- Saving & Loading R Workspaces
- R Base Graphics
- Data Input

Key Learning Outcomes

The key concepts & mechanics of the plotting with the Grammar of Graphics[1] inspired 'ggplot2':

- the mechanics of the

  `ggplot( )`

  command
- the concept of aesthetic mapping
- plotting geometries
- scales
- faceting
- saving plots

[1] Leland Wilkinson, *The Grammar of Graphics*, Statistics and Computing. Springer, 2nd edition, 2005.

Key Learning Outcomes

- read data into R from an external file
- fit linear regression models
- produce & examine model diagnostics
- plot data along with predictions of model and associated uncertainty
- conduct stepwise variable selection
- produce summary statistics for model

Key Learning Outcomes

Writing:

- conditional statements
- loops
- functions

Solving problems by writing programs

Key Learning Outcomes

Understand:

- motivations for managaing a coding project via a version control system
- fundamentals of Git & GitHub:
  - local and remote repositories
  - developing multiple versions of the same file
  - combining disparate versions of the same file
  - returning to previous version of a file without loosing the current version
  - collaboratively editing files

Module 6
Collaborative Exercise

The Plan

Form small groups and collaboratively explore and analyse some data on ant species richness around the globe. Please re-use as much of the code from the preceeding exercises as you would like to.

Key Learning Outcomes

Practise and in doing so consolidate the skills you have learned over this course

If you're already familiar with R

Feel free to use this time to pursue something that interests you

You could:

- Visit the GitHub directory for this course and pick a code file you like to start working through `https://github.com/brfitzpatrick/Intro_to_R`
- See how far you can get through the incrementally harder maths/programming problems at `https://projecteuler.net/`
- pursue your own project work

tuning in occasionally for the sections that interest you.

I'll need to focus on delivering the course but I'll try to check in with you peridocially throughout the next 2.5 days.

Let's begin

Please open RStudio

Image Credits

R Foundation, from http://www.r-project.org - Originally from http://developer.r-project.org/Logo/Rlogo.svg, modified to simpler SVG format.