



Overview



Contents

- Why R?
- What is R?
- General pros and cons
- RStudio IDE
- Graphics
- Markdown
- Shiny
- Conclusions

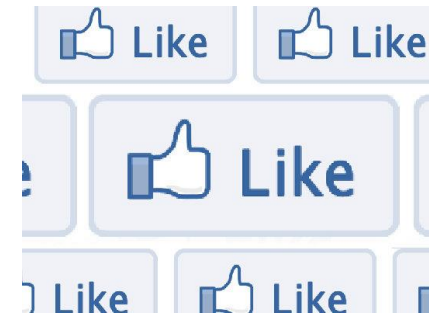
Quiz



- How old is R?
- What language(s) is R derived from?
- Who is Chief Data Scientist at Rstudio?
- What is the tidyverse?
- What is CRAN?

What is R

- Similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues
- R can be considered as a different implementation of S. There are some important differences though
- From its origins as a mainly scripting language for statistics its has develop in recent years to be much more
- This does mean that it has some quirks and idiosyncrasies!

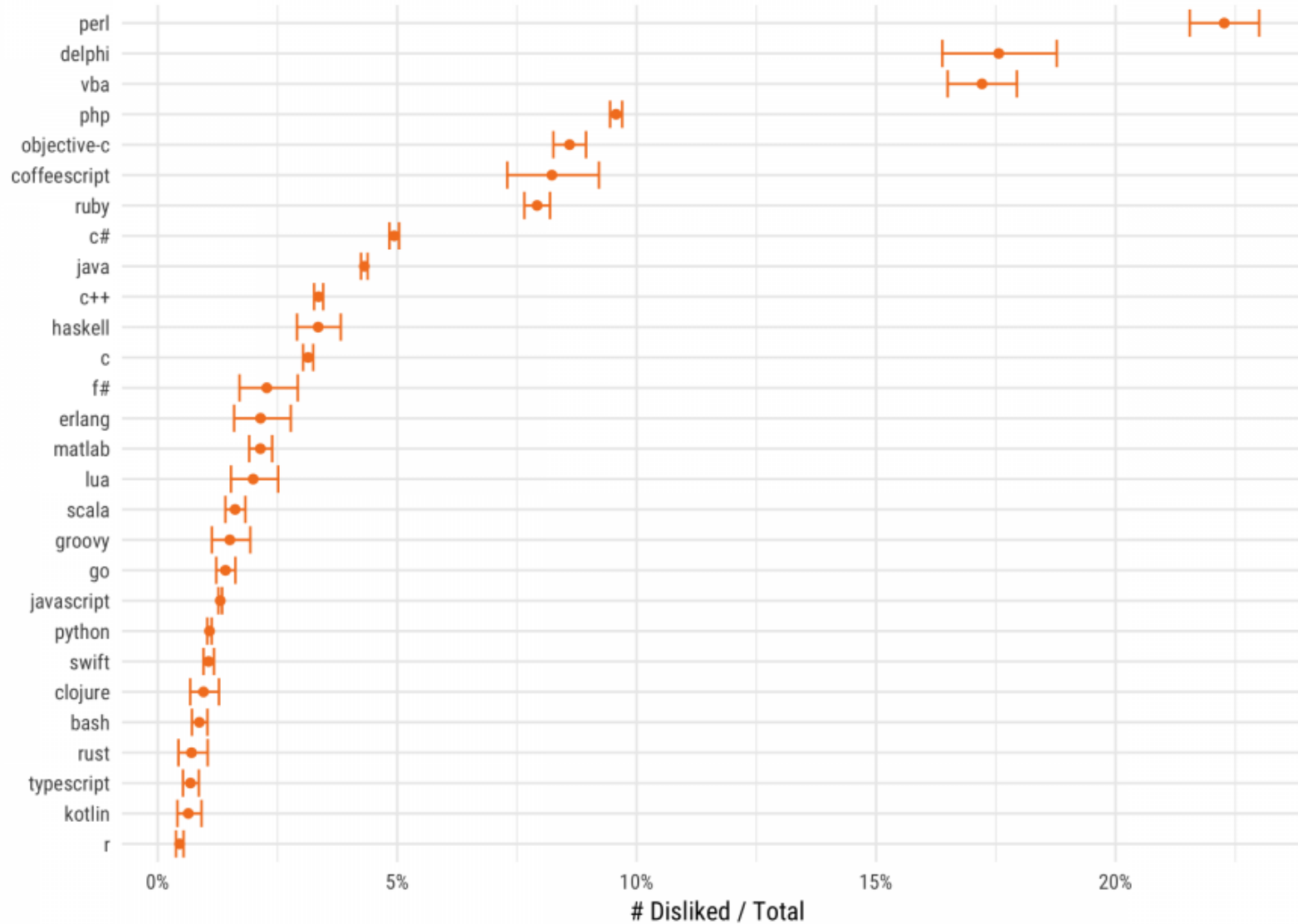


R's popularity

- R is a free, open source high-level software program
- R is arguably the go-to software for data science and statistics
- R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, ...) and graphical techniques
- Ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed

How disliked is each programming language?

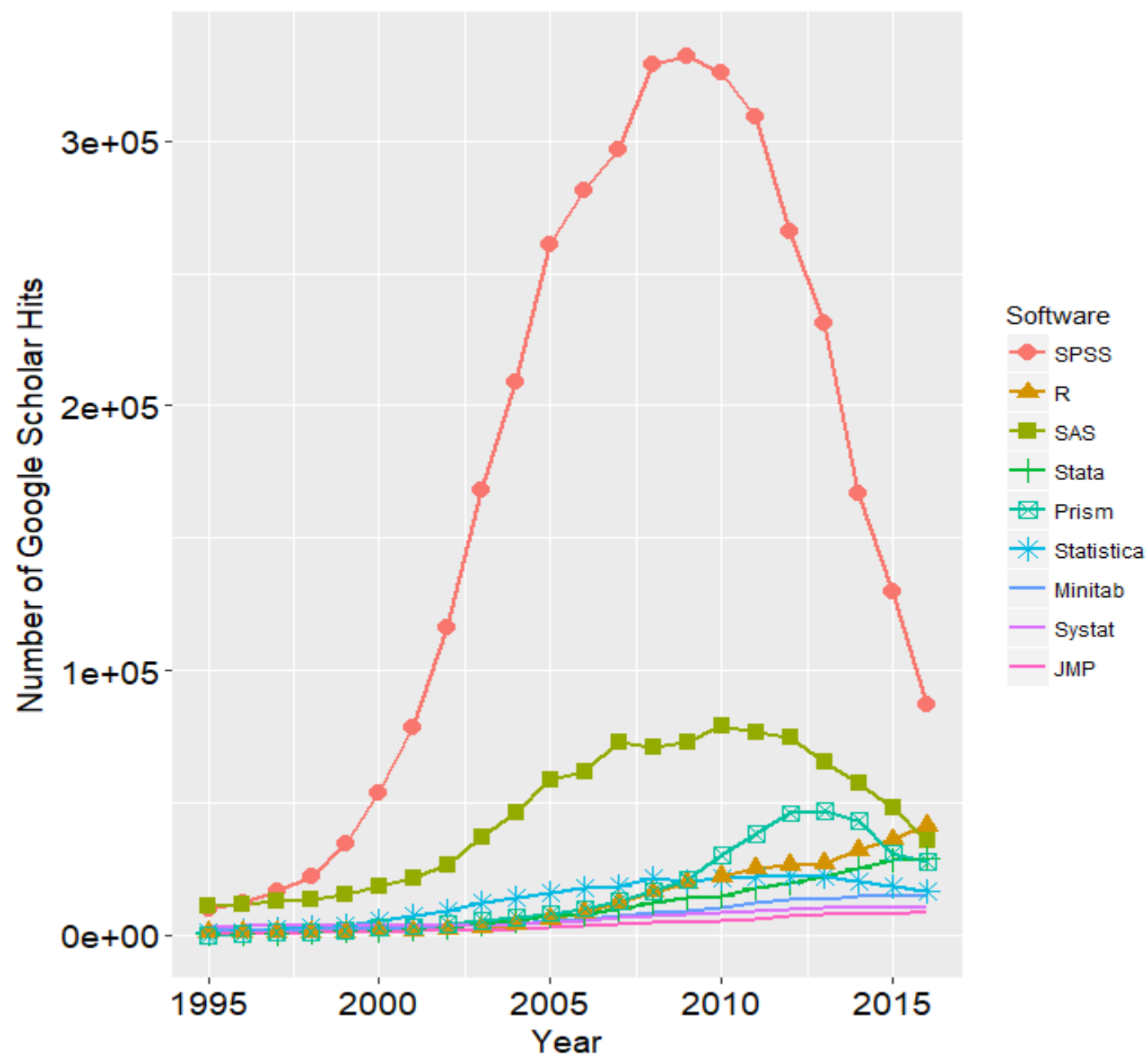
Based on "likes" and "dislikes" on Stack Overflow Developer Stories. Includes 95% credible intervals



<https://stackoverflow.blog/2017/10/31/disliked-programming-languages/>

30/11/2018

Health Economics in R: scoping workshop



<https://r4stats.com/2014/08/20/r-passes-spss-in-scholarly-use-stata-growing-rapidly/>

30/11/2018

Health Economics in R: scoping workshop

General Pros

- Free and open source.
- Available for Windows, Macintosh, and Linux
- Publication-quality graphs
- Rivals (and in many cases, exceeds) SAS and Stata in terms of availability of advanced statistical methods and algorithms, through availability of user-created packages
- Packages for *literate statistical programming* - weaving written reports and analysis code in one document
- Simple syntax
- Interacts with other software, including Excel, C, Python, SQL, stan, WinBUGs and others

General pros

- R uses command-line scripting, which is ideal for storing numerous series of complex data-analysis and recycling that analysis' on similar sets of data
- Upgrades to the software are much more regular
 - This is extremely advantageous for statistical programming languages and environments
- R's large and active online community supply a myriad of documentation, tutorials and online query forums
 - It is now supplemented by more than 8000 community developed open- source packages available for download from The Comprehensive R Archive Network (CRAN)
 - Authors often supplement the package submission with a publication in the Journal of Statistical Software, with more rigorous documentation and relevant theoretical material

General pros

- Profiling tools examine program performance and aid speeding up run times
- Debugging tools enable faster, less stressful bug fixes
- Similar to MATLAB (expensive)

General Cons

- Programming is required!
- Is relatively slow, e.g. loops, to other lower level programming languages e.g. C
 - but can leverage this by linking with them
- Some of its structure/behaviour can be surprising for people coming from other programming languages
- People just may be more comfortable with analyses in something other than R

RStudio

- RStudio is a free and open-source integrated development environment (IDE) for R, a programming language for statistical computing and graphics
- RStudio is available in open source and commercial editions and runs on the desktop (Windows, macOS, and Linux)
- Its interface is organized so that the user can clearly view graphs, data tables, R code, and output all at the same time
- Also offers an Import-Wizard-like feature that allows users to import CSV, Excel, SAS (*.sas7bdat), SPSS (*.sav), and Stata (*.dta) files into R without having to write the code to do so.

RStudio IDE : : CHEAT SHEET

Documents and Apps

Check spelling, Render output, Choose output format, Choose output location, Insert code chunk, Open Shiny, R Markdown, knitr, Sweave, LaTeX, Rd files and more in Source Pane

Jump to previous chunk, Jump to next chunk, Run selected chunk, Publish to server, Show file outline

Access markdown guide at **Help > Markdown Quick Reference**

Jump to chunk, Set knitr chunk options, Run this and all previous code chunks, Run this code chunk

RStudio recognizes that files named **app.R**, **server.R**, **ui.R**, and **global.R** belong to a shiny app

Run app, Choose location to view app, Publish to shinyapps.io or server, Manage publish accounts

Write Code

Navigate tabs, Open in new window, Save, Find and replace, Compile as notebook, Run selected code

Source with or without Echo, Show file outline

Multiple cursors/column selection with **Alt + mouse drag**

Code diagnostics that appear in the margin. Hover over diagnostic symbols for details.

Syntax highlighting based on your file's extension

Tab completion to finish function names, file paths, arguments, and more.

Multi-language code snippets to quickly use common blocks of code.

Jump to function in file, Change file type

Working Directory, Press **↑** to see command history, Maximize, minimize panes, Drag pane boundaries

R Support

Import data with wizard, History of past commands to run/copy, Display .Rpres slideshows **File > New File > R Presentation**

Load workspace, Save workspace, Delete all saved objects, Search inside environment

Choose environment to display from list of parent environments, Display objects as list or grid

Displays saved objects by type with short description, View in data viewer, View function source code

Create folder, Upload file, Delete file, Rename file, Change directory

Path to displayed directory, A File browser keyed to your working directory. Click on file or directory name to open.

Pro Features

Share Project with Collaborators, Active shared collaborators, Start new R Session in current project, Close R Session in project, Select R Version

PROJECT SYSTEM
File > New Project

RStudio saves the call history, workspace, and working directory associated with a project. It reloads each when you re-open a project.

RStudio opens plots in a dedicated Plots pane

Navigate recent plots, Open in window, Export plot, Delete plot, Delete all plots

GUI Package manager lists every installed package

Install Packages, Update Packages, Create reproducible package library for your project

Click to load package with **library()**. Unclick to detach package with **detach()**

Package version installed, Delete from library

Debug Mode

Open with **debug()**, **browser()**, or a breakpoint. RStudio will open the debugger mode when it encounters a breakpoint while executing code.

Launch debugger mode from origin of error, Open traceback to examine the functions that R called before the error occurred

Click next to line number to add/remove a breakpoint.

Highlighted line shows where execution has paused

Run commands in environment where execution has paused

Examine variables in executing environment

Select function in traceback to debug

Step through code one line at a time

Step into and out of functions to run

Resume execution mode

Quit debug

Version Control with Git or SVN

Turn on at **Tools > Project Options > Git/SVN**

Stage files, Show file diff, Commit staged files, Push/Pull to remote, View History

Added, Deleted, Modified, Renamed, Untracked

Open shell to type commands, current branch

Package Writing

File > New Project > New Directory > R Package

Turn project into package, Enable roxygen documentation with **Tools > Project Options > Build Tools**

Roxygen guide at **Help > Roxygen Quick Reference**

Help

RStudio opens documentation in a dedicated Help pane

Home page of helpful links, Search within help file, Search for help file

Viewer Pane displays HTML content, such as Shiny apps, RMarkdown reports, and interactive visualizations

Stop Shiny app, Publish to shinyapps.io, rpubs, RSConnect, ..., Refresh

View(<data>) opens spreadsheet like view of data set

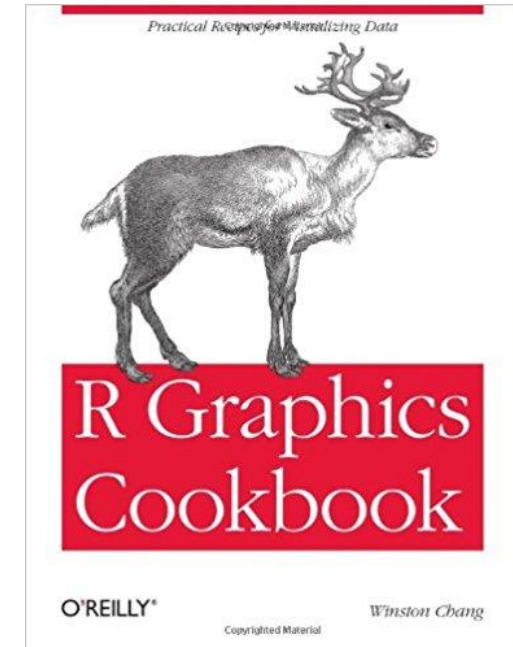
	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2					
3					
4					

Filter rows by value or value range, Sort by values, Search for value



Graphics

- R comes with great abilities in data visualization, should the visualization be static, interactive and even far more complicated
- ggplot2: “the grammar of graphics”

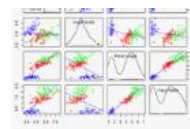




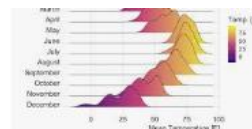
r graphics demo - YouTube
youtube.com



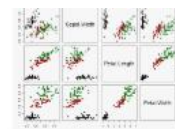
The R Graph Gallery – Inspiration and ...
r-graph-gallery.com



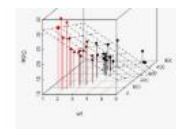
Quick-R: Advanced Graphs
statmethods.net



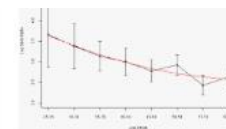
R Graphics Essentials - Articles - STHDA
sthda.com



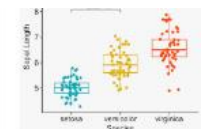
R Graphics | R-bloggers
r-bloggers.com



Anti-aliasing in R graphics u...
stackoverflow.com



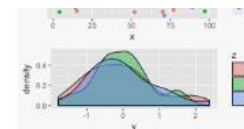
10 tips for making your R graphics I...
blog.revolutionanalytics.com



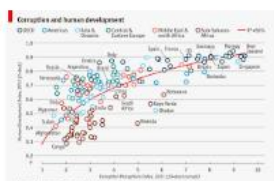
R Graphics Essentials - Artic...
sthda.com



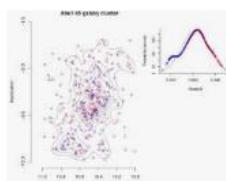
R Graphics window presenting th...
researchgate.net



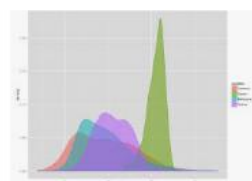
Save base graphics as pseudo-objects...
andrewheiss.com



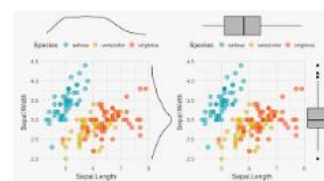
R graphics with ggplot2 workshop notes
tutorials.iq.harvard.edu



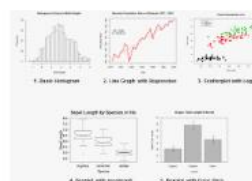
R graphics plot gallery - plots, charts...
sr.bham.ac.uk



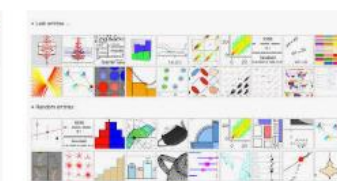
The R Graph Gallery – Inspiration and ...
r-graph-gallery.com



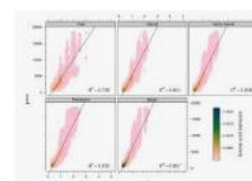
Plot Two Continuous Variables: Scatter ...
sthda.com



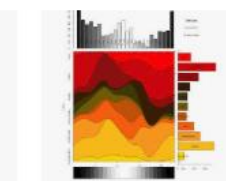
RPubs - R Base Graphics: An Idiot'...
rpubs.com



The R Graph Gallery goes social ...
blog.revolutionanalytics.com



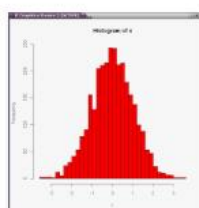
publication quality graphics using R ...
metvurst.wordpress.com



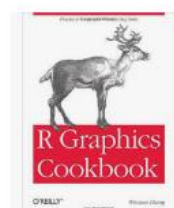
R Graphics| Beautiful graphics...
talkstats.com



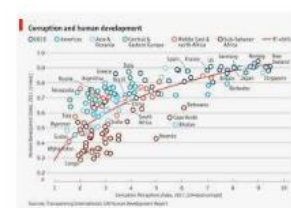
R Graphics Essentials - Articles - S...
sthda.com



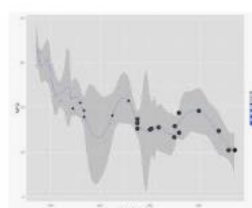
File:R Graphics-histogram.png ...
commons.wikimedia.org



R Graphics Cookbook: Prac...
amazon.com



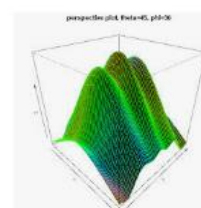
R : Graphics Tutorial Series (Part 6 ...
groups.com



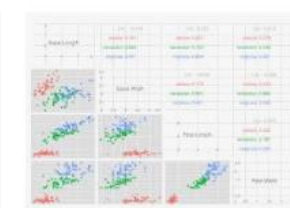
Graphics in R - The Analysis Institute
theanalysisinstitute.com



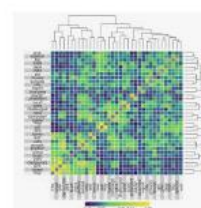
Calling Functions in the R Language ...
support.sas.com



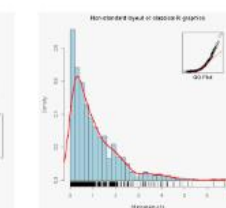
Customizing graphics
zoonek2.free.fr



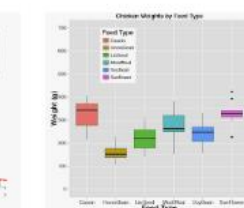
R graphics with 'ggplot2' and 'rgl ...
bragout.wordpress.com



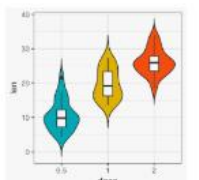
Revolutions: graphics
blog.revolutionanalytics.com



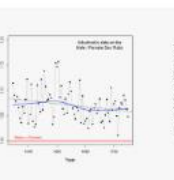
Vincent Zoonekynd's Blog
zoonek.free.fr



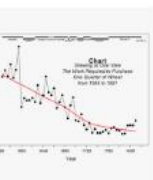
Graphics with R: an introduction t...
bmecblog.wordpress.com



R Graphics Essentials - Arti...
sthda.com



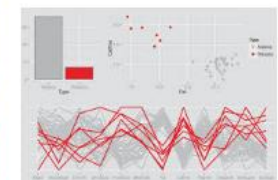
Data Visualization in R
datavis.ca



A Survey Of R Graphics
slideshare.net



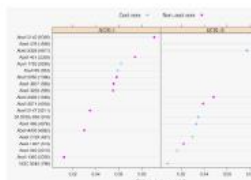
Ch12 Ensemble Graphics and Case Studi...
gradaanr.net



Power BI custom visuals, based on R ...
blog.revolutionanalytics.com



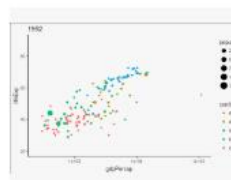
R graphics plot gallery - plots, charts ...
sr.bham.ac.uk



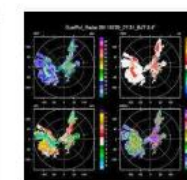
Circular barplot – The R Gr...
r-graph-gallery.com



Graphics and Image Processing in R ...
r-bloggers.com



NCL Graphics: Radar (r.th...
ncl.ucar.edu



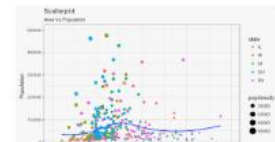
A Review of the R Graphic...
blog.revolutionanalytics.com



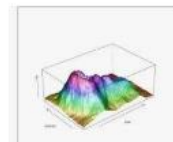
Holtz - The R Graph Gallery



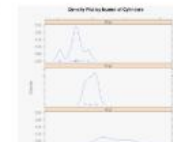
The igraph library for compl...



Top 50 ggplot2 Visualizations - The ...



Graphics — rpy2 v2.2.2 do...



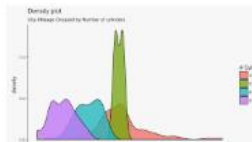
Quick-R: Lattice Graphs



Data Visualization in R



Top 50 ggplot2 Visualizations - The ...



Graphics - GoogleVis and Maps ...



R graphics on the Web with SVG ...

Markdown



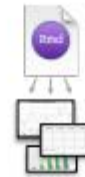
R Markdown Cheat Sheet

learn more at rmarkdown.rstudio.com



.Rmd files

An R Markdown (.Rmd) file is a record of your research. It contains the code that a scientist needs to reproduce your work along with the narration that a reader needs to understand your work.



Reproducible Research

At the click of a button, or the type of a command, you can rerun the code in an R Markdown file to reproduce your work and export the results as a finished report.



Dynamic Documents

You can choose to export the finished report as a html, pdf, MS Word, ODT, RTF, or markdown document; or as a html or pdf based slide show.

Workflow

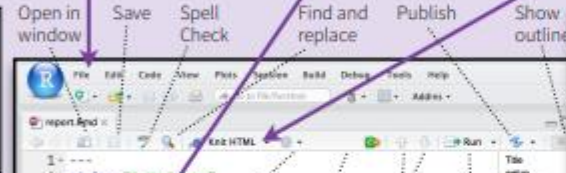
1 Open a new .Rmd file at File ► New File ► R Markdown. Use the wizard that opens to pre-populate the file with a template

2 Write document by editing template

3 Knit document to create report Use knit button or `render()` to knit

4 Preview Output in IDE window

5 Publish (optional) to web or server



Synch publish button to accounts at:

- rpubs.com
- shinyapps.io
- RStudio Connect

Reload document

Find in document

File path to output document

6 Examine build log in R Markdown console

7 Use output file that is saved alongside .Rmd

.Rmd structure

YAML Header

Optional section of render (e.g. pandoc) options written as key:value pairs (YAML).

- At start of file
- Between lines of ---

Text

Narration formatted with markdown, mixed with:

Code chunks

Chunks of embedded code. Each chunk:

- Begins with `{r}`
- ends with `}`

R Markdown will run the code and append the results to the doc.

It will use the location of the .Rmd file as the **working directory**

render()

Use `rmarkdown::render()` to render/knit at cmd line. Important args:

input - file to render

output_format

output_options - List of render options (as in YAML)

output_file

output_dir

params - list of params to use

envir - environment to evaluate code chunks in

encoding - of input file

Interactive Documents

Turn your report into an interactive Shiny document in 4 steps



- 1 Add runtime:** shiny to the YAML header.
- 2 Call Shiny input** functions to embed input objects.
- 3 Call Shiny render** functions to embed reactive output.
- 4 Render with `rmarkdown::run`** or click **Run Document** in RStudio IDE

```
---
output: html_document
runtime: shiny
---
```

```
{r, echo = FALSE}
numericInput("n",
  "How many cars?", 5)

renderTable({
  head(cars, input$n)
})
```

5		
	speed	dist
1	4.00	2.00
2	4.00	10.00
3	7.00	4.00
4	7.00	22.00
5	8.00	16.00

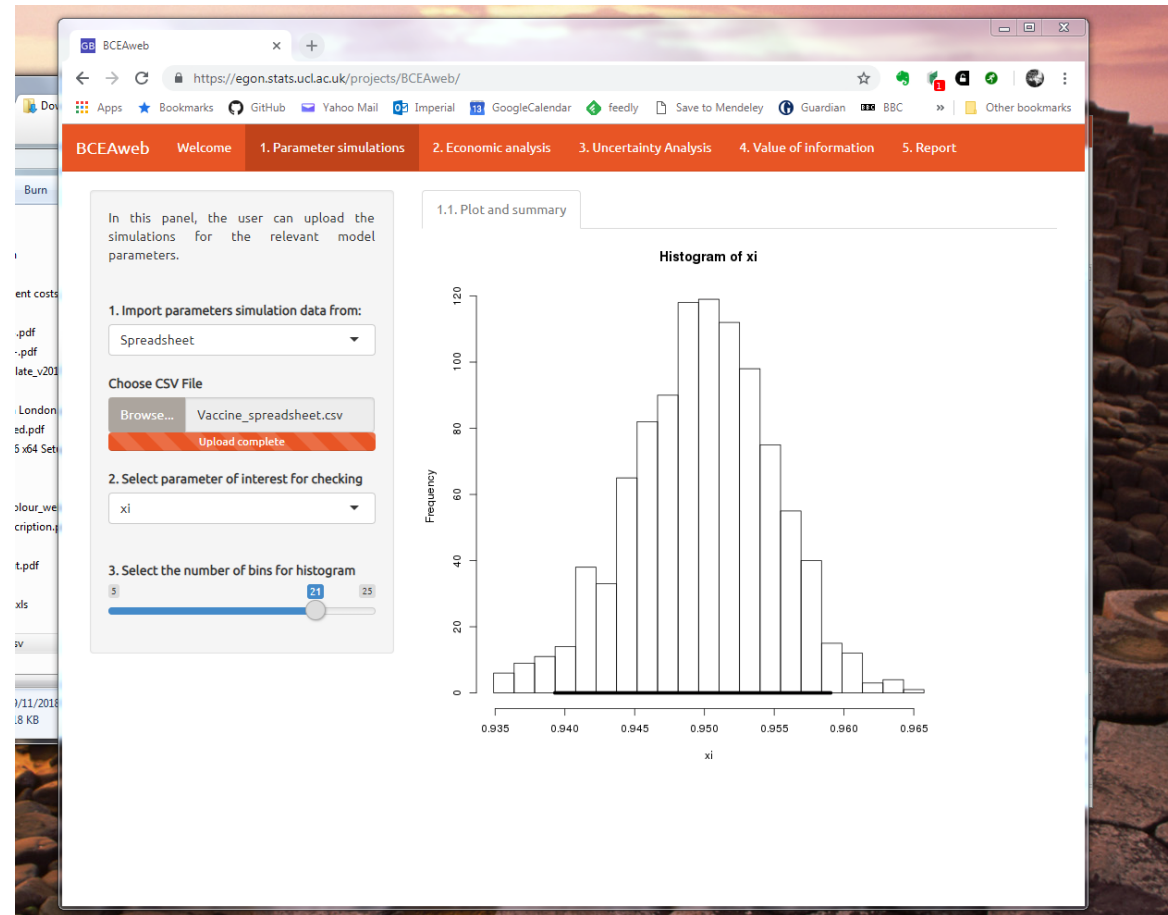
Embed a complete app into your document with `shiny::shinyAppDir()`

* Your report will be rendered as a Shiny app, which means you must choose an html output format, like **html_document**, and serve it with an active R Session.



Shiny

- Shiny is an R package that makes it easy to build interactive web apps straight from R
- You can
 - host standalone apps on a webpage
 - embed them in R Markdown documents
 - build dashboards
- You can also extend your Shiny apps with CSS themes, htmlwidgets, and JavaScript actions.



BCEAweb