a not-so-short answer to the question...

...is

suitable enough for biostatisticians

involved in Clinical Research & Evidence-Based Medicine?

...and can it replace SAS in this area?

Adrian Olszewski
r.clin.res@gmail.com

WeRcome!

Welcome to my journey through the world of R!

✔ Become familiar with the *lingua franca* of statistics

✔ Discover applications of R in Evidence-Based Medicine

✔ Rock, squeeze and explore your data deeply - for free

✔ Find 13 reasons why you will love R! cannot wait? jump now!

✔ Enhance your skills...

...and start using R today!

...befoRe we start...

DISCLAIMER

All trademarks, logos of companies and names of products

used in this document

are the sole property of their respective owners

and are included here

for informational, illustrative purposes only *nominative fair use*

•••

This presentation is based exclusively on information

publicly available on the Internet under provided hyperlinks.

•••

If you think I violate your rights, please email me: r.clin.res@gmail.com

Agenda

- Brief introduction
- The R family
- 7 cool facts about R
- List of R users
- Is R a popular software?
- Demonstrative screenshots
- 13 reasons why you will love GNU R
- Further impoRtant issues
- Conclusion

Agenda: 13 reasons why you will love GNU R

I	R is (extremely) cheap. In fact - it's free :)
II	R has (extremely) wide range of capabilities
II 1/3 :)	R is easy to maintain!
II 2/3 :)	R is not resource consuming!
III	R is supported by the world of science
IV	R is supported by the community
IV ½ :)	Books
V	R is supported by the business
V ½ :)	R and SAS
VI	R is able to read data in many formats
VI 1/3 :)	R and relational databases
VI 2/3 :)	Advanced data manipulation
VII	Interoperability is easy to achieve
VIII	R is truly cross-platform
IX	R offers numerous ways of presenting data
IX ½ :)	Graphics – all have waited for this moment :)
X	There are many options to optimize the code
XI	R is able to handle large amount of data
XII	R has a set of fancy tools and IDEs
XIII	FDA: R is OK for drug trials! - it should start from this slide :)

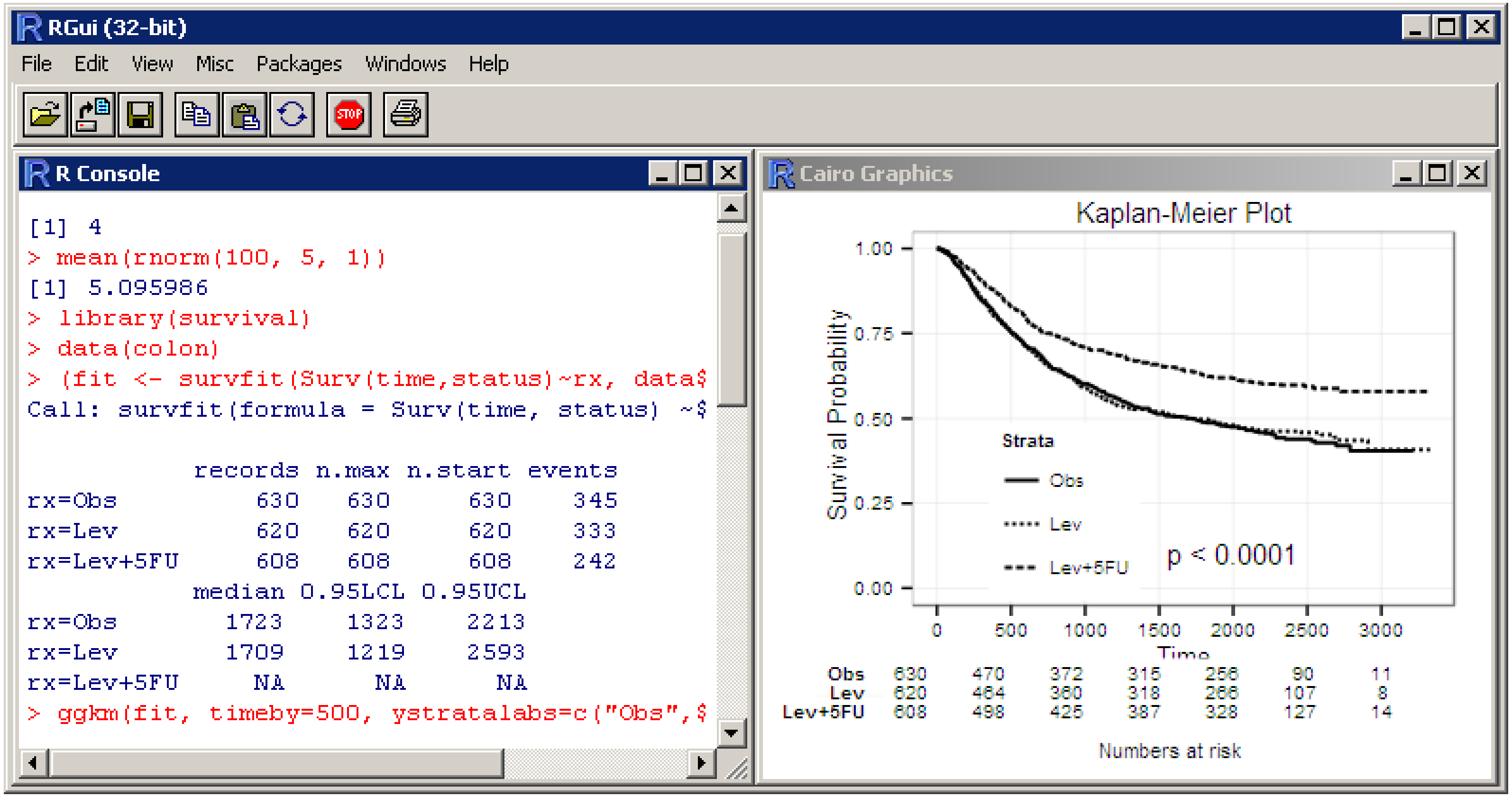
Agenda: Further impoRtant issues

IntRoduction

...so, what is “R”?

In simply words, R is a free software environment for statistical computing, data manipulation and charting, widely used in the scientific world. It is also the name of a high-level, interpreted programming language.

R (the successor of S) was created in 1992 by Ross Ihaka and Robert Gentleman at University of Auckland. Now it is developed by the R Development Core Team.



```
[1] 4
> mean(rnorm(100, 5, 1))
[1] 5.095986
> library(survival)
> data(colon)
> (fit <- survfit(Surv(time,status)~rx, data$
Call: survfit(formula = Surv(time, status) ~$

            records n.max n.start events
rx=Obs          630   630     630    345
rx=Lev          620   620     620    333
rx=Lev+5FU      608   608     608    242
            median 0.95LCL 0.95UCL
rx=Obs        1723    1323    2213
rx=Lev        1709    1219    2593
rx=Lev+5FU      NA      NA      NA
> ggkm(fit, timeby=500, ystratalabs=c("Obs",$
```

is it a popular softwaRe ?

[…] R is also the name of a popular programming language used by a growing number of data analysts inside corporations and academia. It is becoming their *lingua franca* partly because data mining has entered a golden age, whether being used to set ad prices, find new drugs more quickly or fine-tune financial models.

Companies as diverse as Google, Pfizer, Merck, Bank of America, the InterContinental Hotels Group and Shell use it.

Data Analysts Captivated by R's Power (The New York Times)

http://tinyurl.com/captivated-by-r

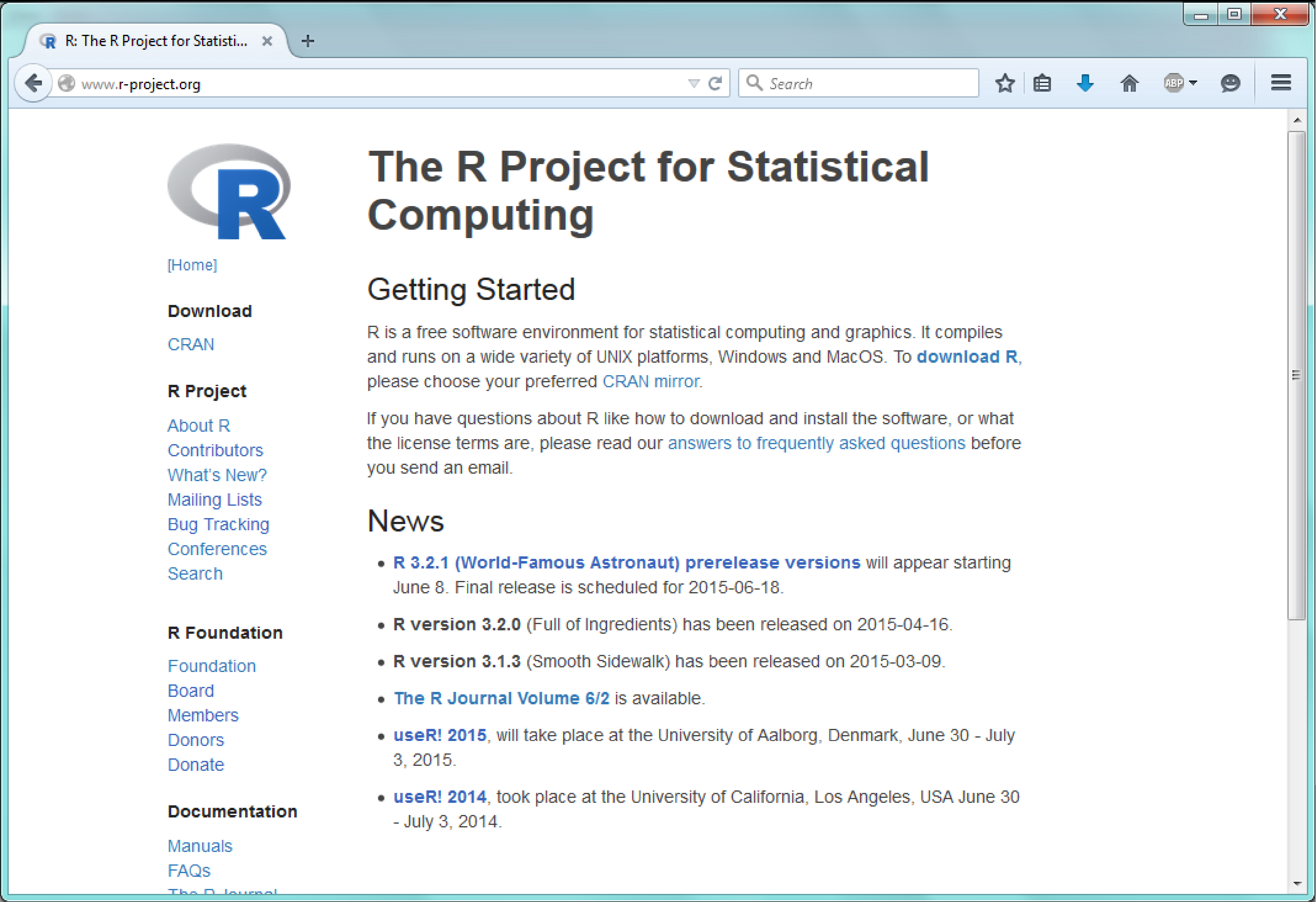
…so, what is “R”?

Short characteristics:

Description:	computational environment + programming language
Developer:	R Development Core Team
Operating systems:	cross-platform: Windows, Unix, Linux, Linux-based “mobile‟ OS: (Android, Maemo, Raspbian), Mac OS X
Form:	command line + third-party IDEs and editors RStudio, RCommander, etc.
Infrastructure:	R core library + shell + libraries (base and third-party)
Model of work:	1) standalone application, 2) standalone server, 3) server process
Programming lang.:	interpreted, high-level with dynamic typing; debugger onboard
Paradigm:	1) array, 2) object-oriented, 3) imperative, 4) functional, 5) procedural, 6) reflective
Source of libraries:	central, mirrored repository – CRAN; users' private repositories, third-party repositories (Github, Rforge), other sources
License of the core:	GNU General Public License ver. 2
License of libraries:	99.9% open-source, rest – licensed (free for non-commercial use)

http://www.r-project.org

…so, what is "R"?



The R Project for Statistical Computing

[Home]

Download

CRAN

R Project

About R
Contributors
What's New?
Mailing Lists
Bug Tracking
Conferences
Search

R Foundation

Foundation
Board
Members
Donors
Donate

Documentation

Manuals
FAQs
The R Journal

Getting Started

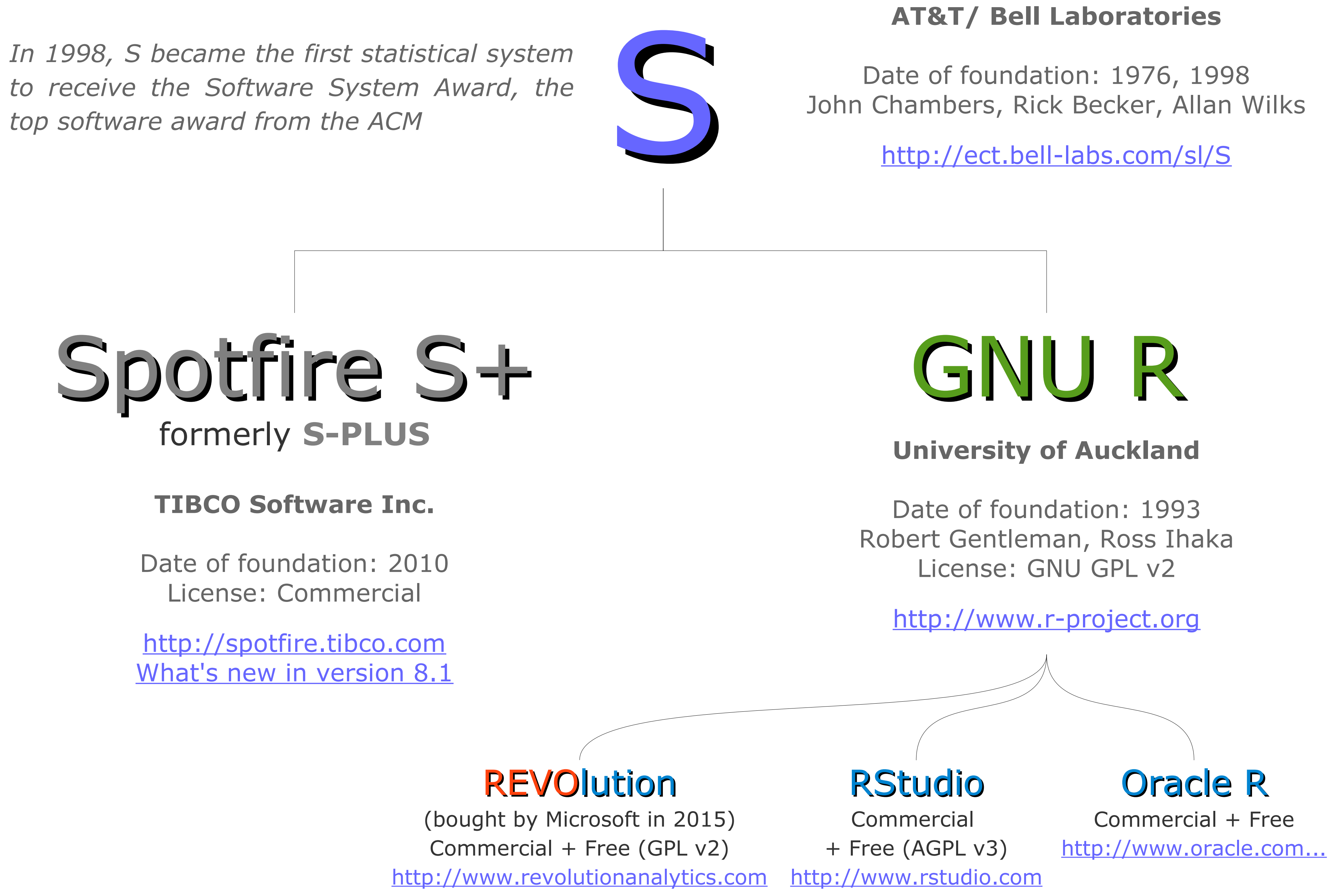
R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To **download R**, please choose your preferred CRAN mirror.

If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to frequently asked questions before you send an email.

News

- **R 3.2.1 (World-Famous Astronaut) prerelease versions** will appear starting June 8. Final release is scheduled for 2015-06-18.
- **R version 3.2.0** (Full of Ingredients) has been released on 2015-04-16.
- **R version 3.1.3** (Smooth Sidewalk) has been released on 2015-03-09.
- **The R Journal Volume 6/2** is available.
- **useR! 2015**, will take place at the University of Aalborg, Denmark, June 30 - July 3, 2015.
- **useR! 2014**, took place at the University of California, Los Angeles, USA June 30 - July 3, 2014.

The R family



In 1998, S became the first statistical system to receive the Software System Award, the top software award from the ACM

S

AT&T/ Bell Laboratories

Date of foundation: 1976, 1998
John Chambers, Rick Becker, Allan Wilks

http://ect.bell-labs.com/sl/S

Spotfire S+

formerly **S-PLUS**

TIBCO Software Inc.

Date of foundation: 2010
License: Commercial

http://spotfire.tibco.com
What's new in version 8.1

GNU R

University of Auckland

Date of foundation: 1993
Robert Gentleman, Ross Ihaka
License: GNU GPL v2

http://www.r-project.org

REVOlution

(bought by Microsoft in 2015)
Commercial + Free (GPL v2)
http://www.revolutionanalytics.com

RStudio

Commercial
+ Free (AGPL v3)
http://www.rstudio.com

Oracle R

Commercial + Free
http://www.oracle.com...

did you know, that...

Seven quick (and cool) facts about R April 2014

(1) R is the **highest paid** IT skill [1]

(2) R is **most-used** data science language **after SQL** [2]

(3) R is used by **70%** of data miners [3]

(4) R is **#15** of all programming languages [4]

(5) R is **growing faster** than any other data science language [5]

(6) R is **the #1** Google Search for Advanced Analytic software [6]

(7) R has **more than 2 million** users worldwide [7]

1. Dice.com survey, January 2014
2. O'Reilly survey, January 2014
3. Rexer survey, October 2013
4. RedMonk language rankings, January 2014
5. KDNuggets survey, August 2013
6. Google Trends, March 2014
7. Oracle estimate, February 2012

http://blog.revolutionanalytics.com/2014/04/seven-quick-facts-about-r.html

List of R users

They use(d) R

facebook

ebay

Bank of America

ANZ

Canada
Health Canada

mozilla
FOUNDATION

Google

MERCK

NOVARTIS

AstraZeneca

Microsoft

AMGEN

UBER

KCR

Pfizer

Allianz

DEPARTMENT OF HEALTH & HUMAN SERVICES • USA

FDA

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

BAYER

MOLINA HEALTHCARE

BIOSITE
NEW DIMENSIONS IN DIAGNOSIS

CARDIODX

Ford

Roche

benetech
TECHNOLOGY SERVING HUMANITY

European Environment Agency

KICKSTARTER

Pacific Northwest
NATIONAL LABORATORY

NewScientist

http://www.revolutionanalytics.com/companies-using-r
http://www.rstudio.com/

A word about the list of UseRs

The list is built based exclusively on publicly available information:

- lists of users provided by Revolution, RStudio and others
- articles (example, example) and interviews (example)
- published documents in which a name of a company is visible (example)
- job advertisements
- names of companies supporting / organizing events (conferences, courses)

That is to say, a logo of a company is included in the list only if there is a strong evidence that the company uses or supports (or used or supported) R, based on information shared on the Internet – and thus available for everyone.

Please note, that I am not aware if all listed companies are still using any version of R at the time the presentation is being viewed.

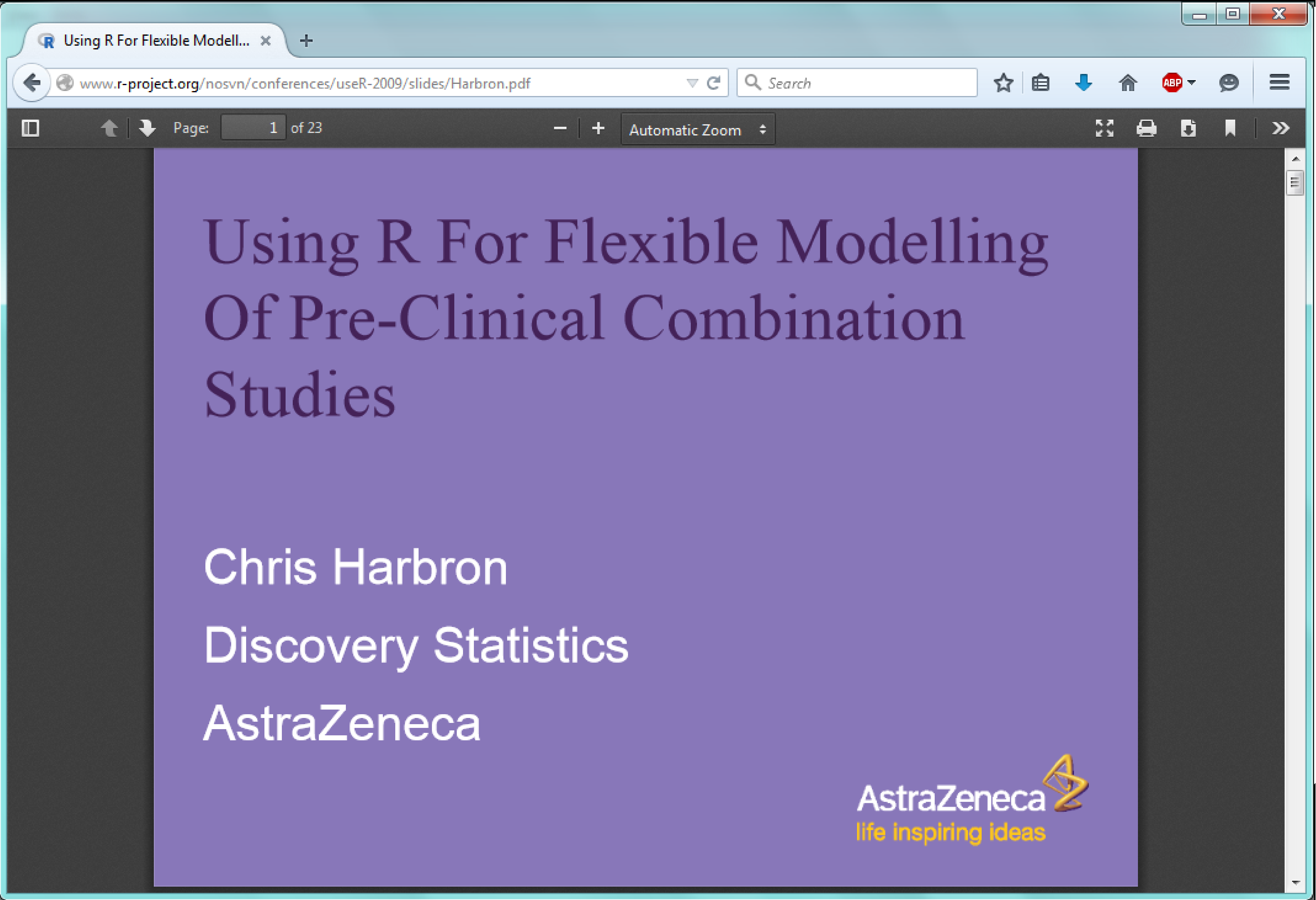
They use R

"We use R for adaptive designs frequently because it's the fastest tool to explore designs that interest us. Off-the-shelf software, gives you off-the-shelf options. Those are a good first order approximation, but if you really want to nail down a design, R is going to be the fastest way to do that."

Keaven Anderson
Executive Director Late Stage Biostatistics,
Merck

http://www.revolutionanalytics.com/content/merck-optimizes-clinical-drug-development...

They use R



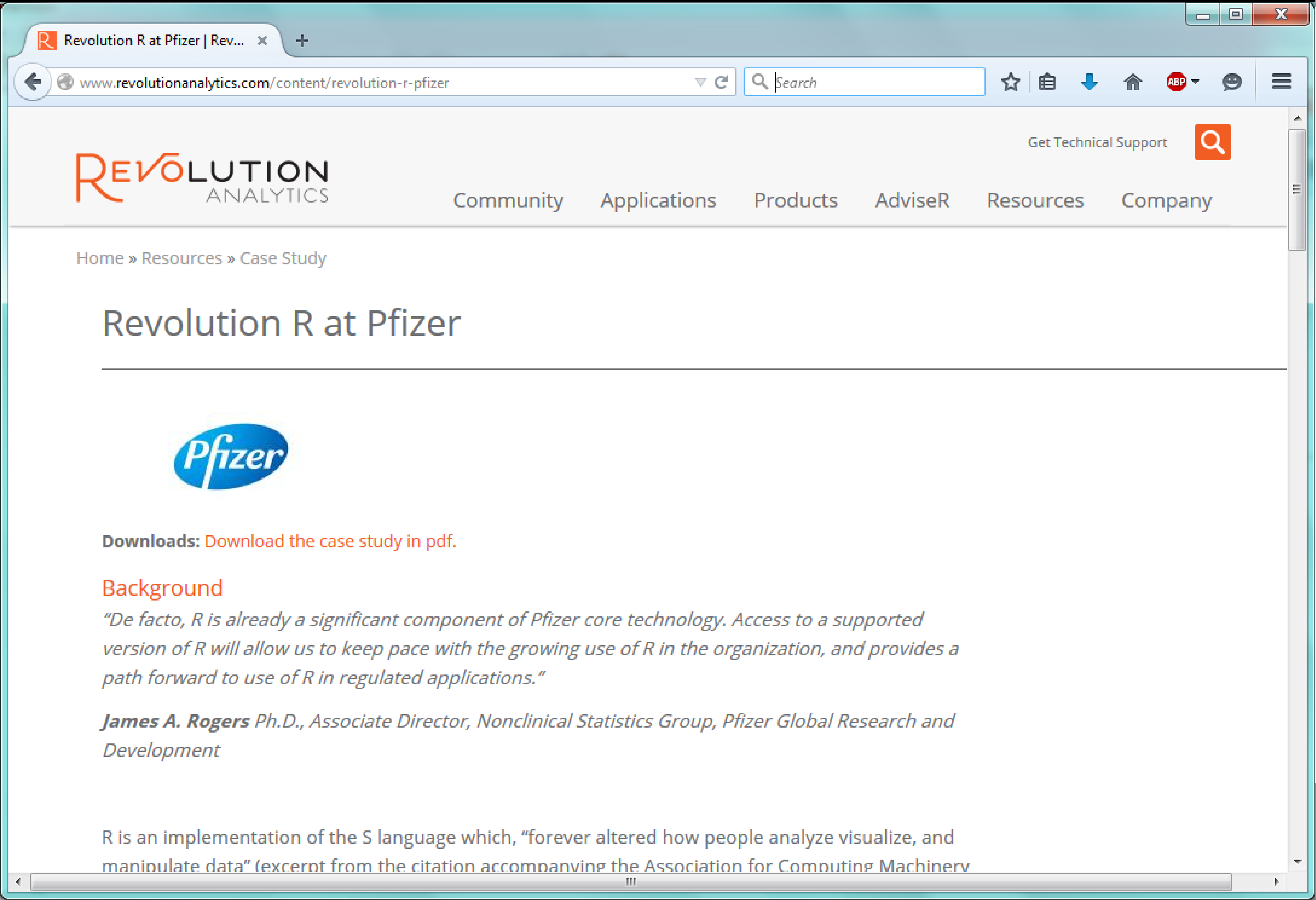
Using R For Flexible Modelling Of Pre-Clinical Combination Studies

Chris Harbron

Discovery Statistics

AstraZeneca

They use R



Home » Resources » Case Study

Revolution R at Pfizer

Downloads: Download the case study in pdf.

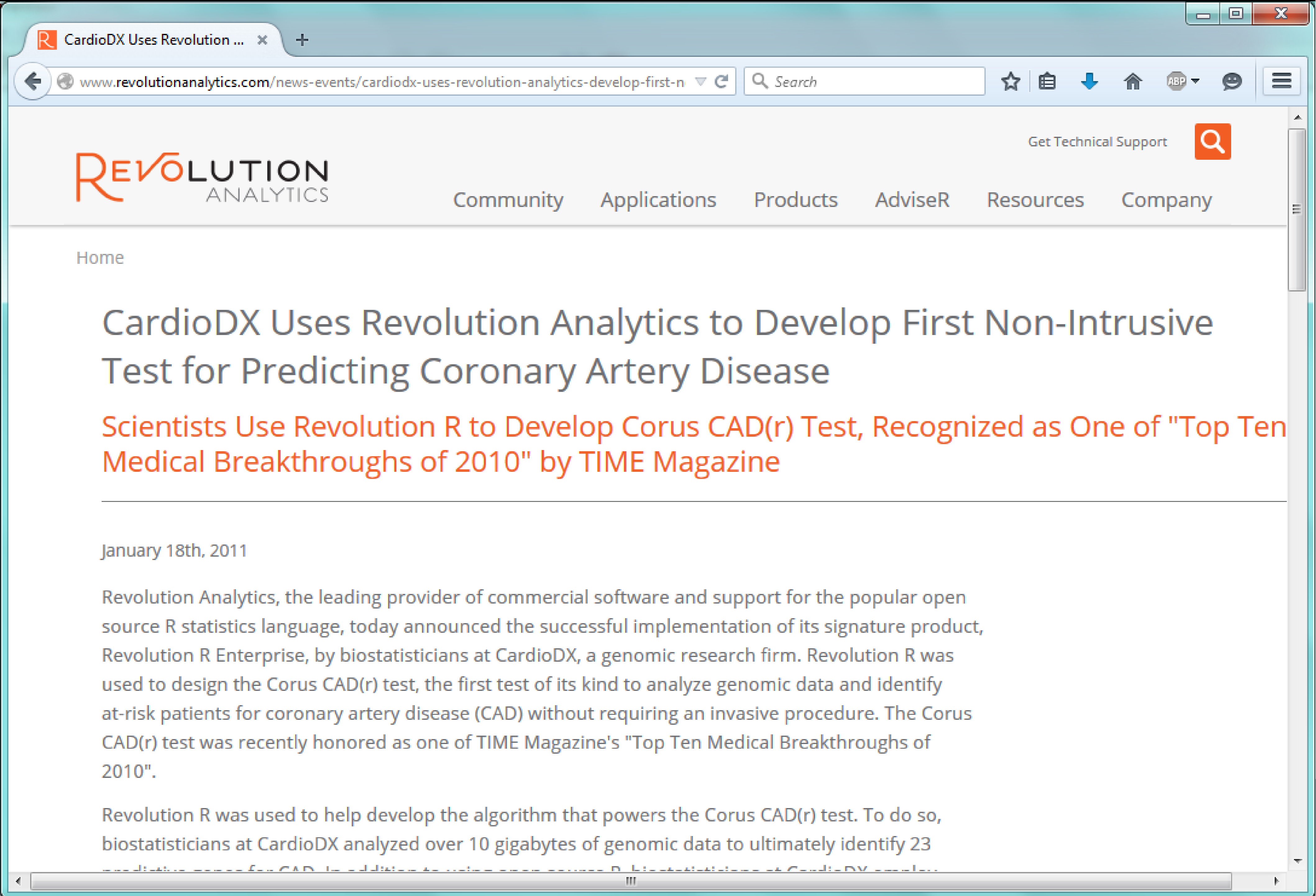
Background

"De facto, R is already a significant component of Pfizer core technology. Access to a supported version of R will allow us to keep pace with the growing use of R in the organization, and provides a path forward to use of R in regulated applications."

James A. Rogers *Ph.D., Associate Director, Nonclinical Statistics Group, Pfizer Global Research and Development*

R is an implementation of the S language which, "forever altered how people analyze visualize, and manipulate data" (excerpt from the citation accompanying the Association for Computing Machinery

They use R



CardioDX Uses Revolution Analytics to Develop First Non-Intrusive Test for Predicting Coronary Artery Disease

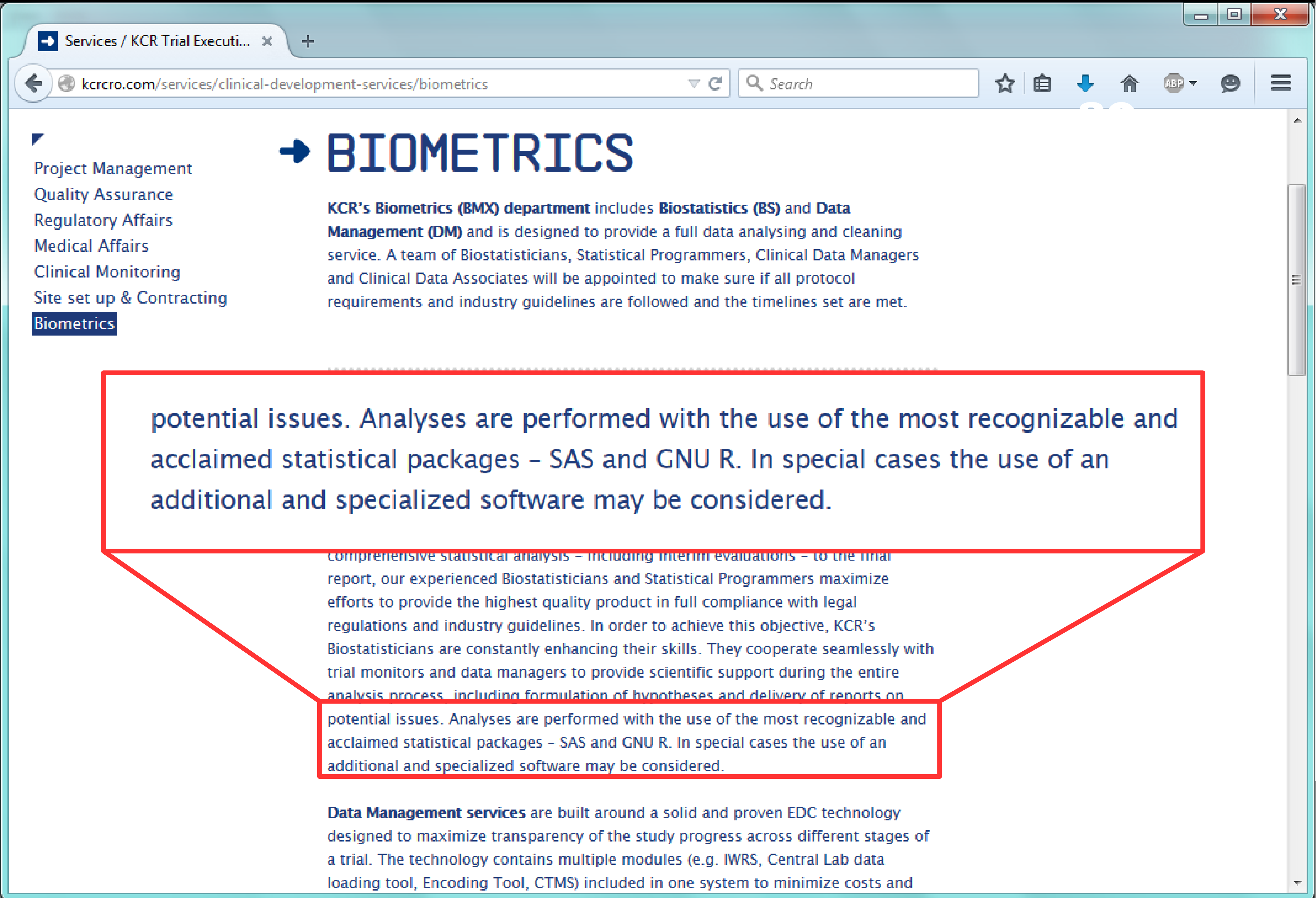
Scientists Use Revolution R to Develop Corus CAD(r) Test, Recognized as One of "Top Ten Medical Breakthroughs of 2010" by TIME Magazine

January 18th, 2011

Revolution Analytics, the leading provider of commercial software and support for the popular open source R statistics language, today announced the successful implementation of its signature product, Revolution R Enterprise, by biostatisticians at CardioDX, a genomic research firm. Revolution R was used to design the Corus CAD(r) test, the first test of its kind to analyze genomic data and identify at-risk patients for coronary artery disease (CAD) without requiring an invasive procedure. The Corus CAD(r) test was recently honored as one of TIME Magazine's "Top Ten Medical Breakthroughs of 2010".

Revolution R was used to help develop the algorithm that powers the Corus CAD(r) test. To do so, biostatisticians at CardioDX analyzed over 10 gigabytes of genomic data to ultimately identify 23

They use R



- Project Management
- Quality Assurance
- Regulatory Affairs
- Medical Affairs
- Clinical Monitoring
- Site set up & Contracting
- Biometrics

BIOMETRICS

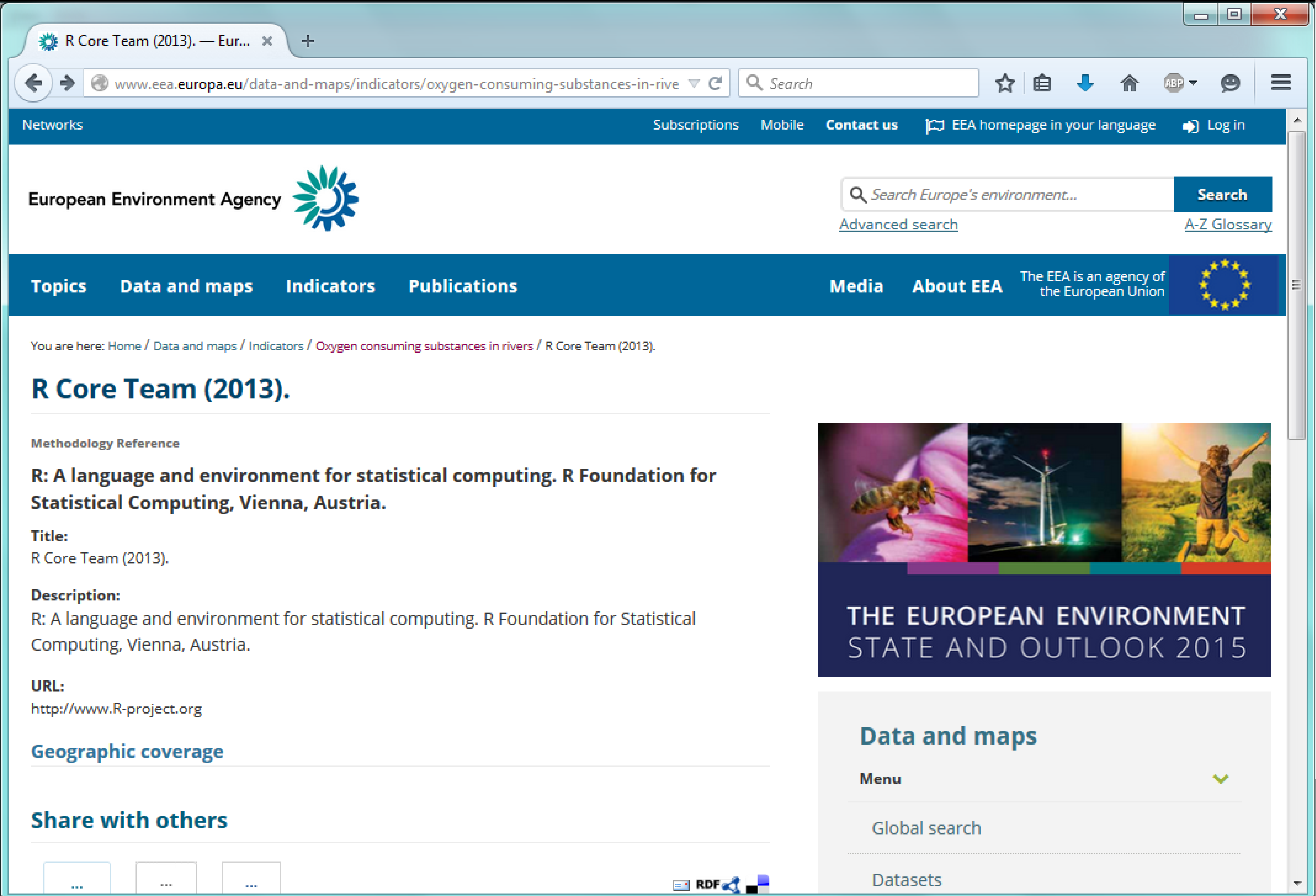
KCR's Biometrics (BMX) department includes **Biostatistics (BS)** and **Data Management (DM)** and is designed to provide a full data analysing and cleaning service. A team of Biostatisticians, Statistical Programmers, Clinical Data Managers and Clinical Data Associates will be appointed to make sure if all protocol requirements and industry guidelines are followed and the timelines set are met.

potential issues. Analyses are performed with the use of the most recognizable and acclaimed statistical packages – SAS and GNU R. In special cases the use of an additional and specialized software may be considered.

comprehensive statistical analysis – including interim evaluations – to the final report, our experienced Biostatisticians and Statistical Programmers maximize efforts to provide the highest quality product in full compliance with legal regulations and industry guidelines. In order to achieve this objective, KCR's Biostatisticians are constantly enhancing their skills. They cooperate seamlessly with trial monitors and data managers to provide scientific support during the entire analysis process, including formulation of hypotheses and delivery of reports on potential issues. Analyses are performed with the use of the most recognizable and acclaimed statistical packages – SAS and GNU R. In special cases the use of an additional and specialized software may be considered.

Data Management services are built around a solid and proven EDC technology designed to maximize transparency of the study progress across different stages of a trial. The technology contains multiple modules (e.g. IWRS, Central Lab data loading tool, Encoding Tool, CTMS) included in one system to minimize costs and

They use R



You are here: Home / Data and maps / Indicators / Oxygen consuming substances in rivers / R Core Team (2013).

R Core Team (2013).

Methodology Reference

R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Title:
R Core Team (2013).

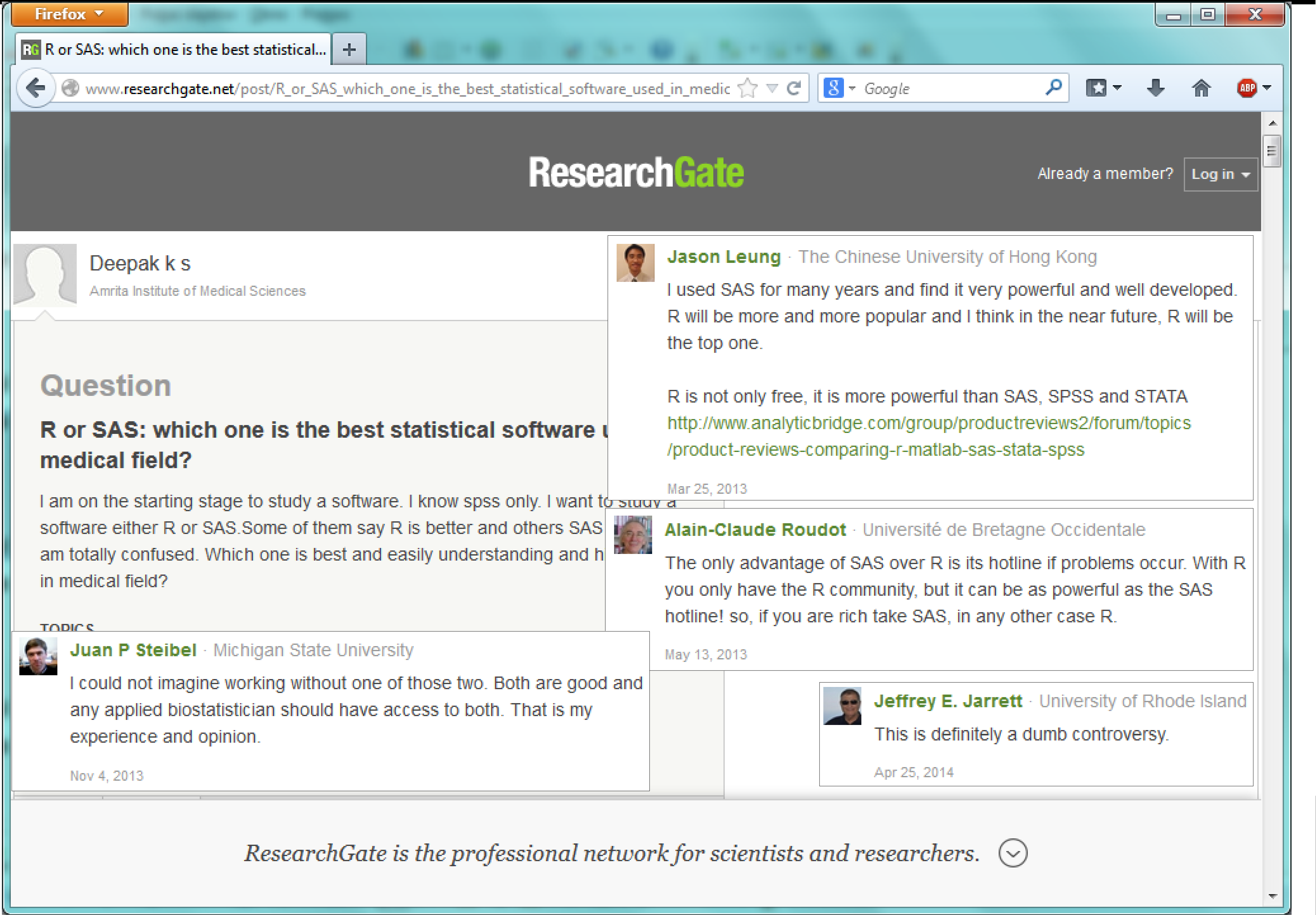
Description:
R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

URL:
http://www.R-project.org

Geographic coverage

Share with others

In many discussions some praise SAS, others STATA or SPSS. But most of them refer to R with respect.



Deepak k s
Amrita Institute of Medical Sciences

Question

R or SAS: which one is the best statistical software u medical field?

I am on the starting stage to study a software. I know spss only. I want to study a software either R or SAS.Some of them say R is better and others SAS am totally confused. Which one is best and easily understanding and h in medical field?

TOPICS

Jason Leung · The Chinese University of Hong Kong

I used SAS for many years and find it very powerful and well developed. R will be more and more popular and I think in the near future, R will be the top one.

R is not only free, it is more powerful than SAS, SPSS and STATA
http://www.analyticbridge.com/group/productreviews2/forum/topics/product-reviews-comparing-r-matlab-sas-stata-spss

Mar 25, 2013

Alain-Claude Roudot · Université de Bretagne Occidentale

The only advantage of SAS over R is its hotline if problems occur. With R you only have the R community, but it can be as powerful as the SAS hotline! so, if you are rich take SAS, in any other case R.

May 13, 2013

Juan P Steibel · Michigan State University

I could not imagine working without one of those two. Both are good and any applied biostatistician should have access to both. That is my experience and opinion.

Nov 4, 2013

Jeffrey E. Jarrett · University of Rhode Island

This is definitely a dumb controversy.

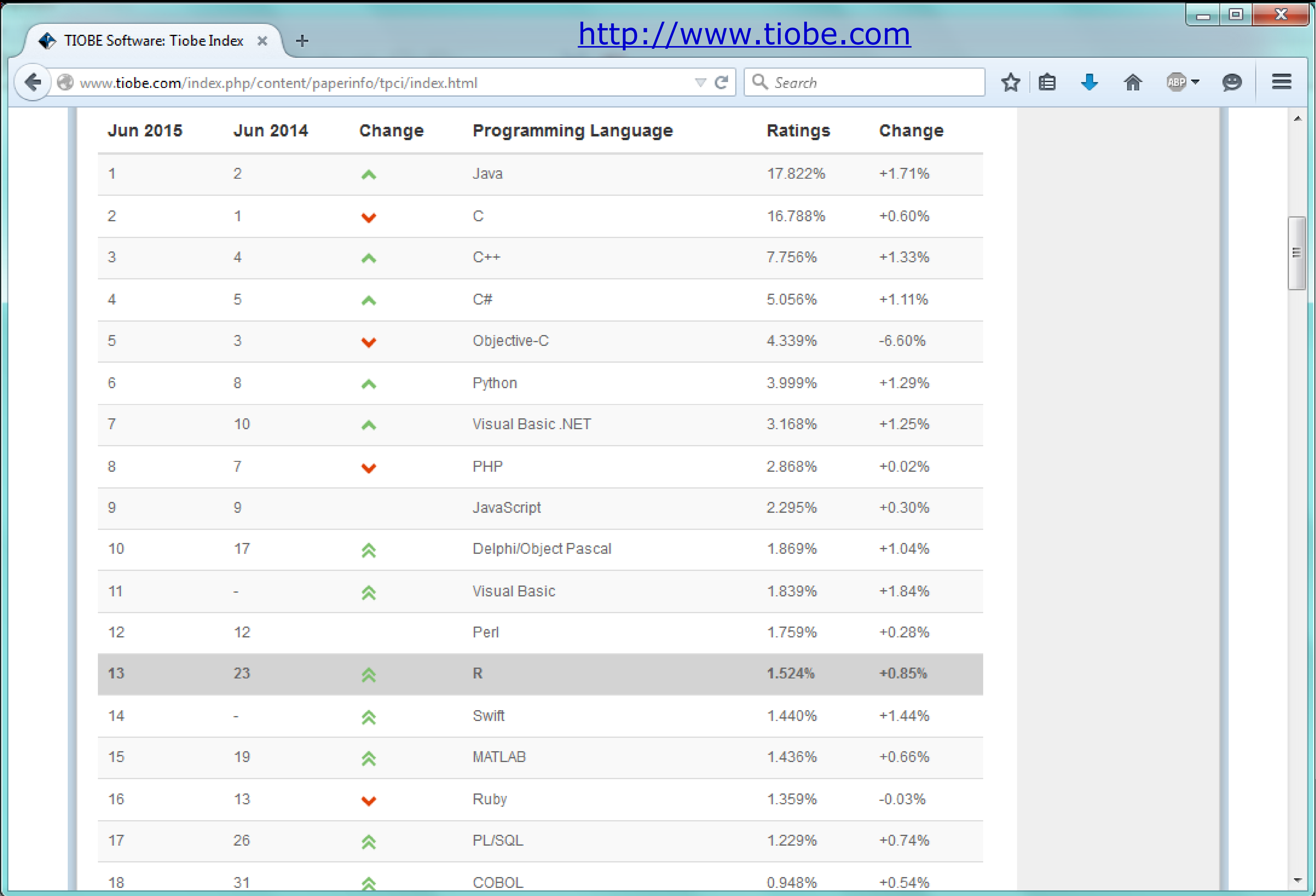
Apr 25, 2014

ResearchGate is the professional network for scientists and researchers.

Is R a popular software?

R is near the top 10 most popular languages

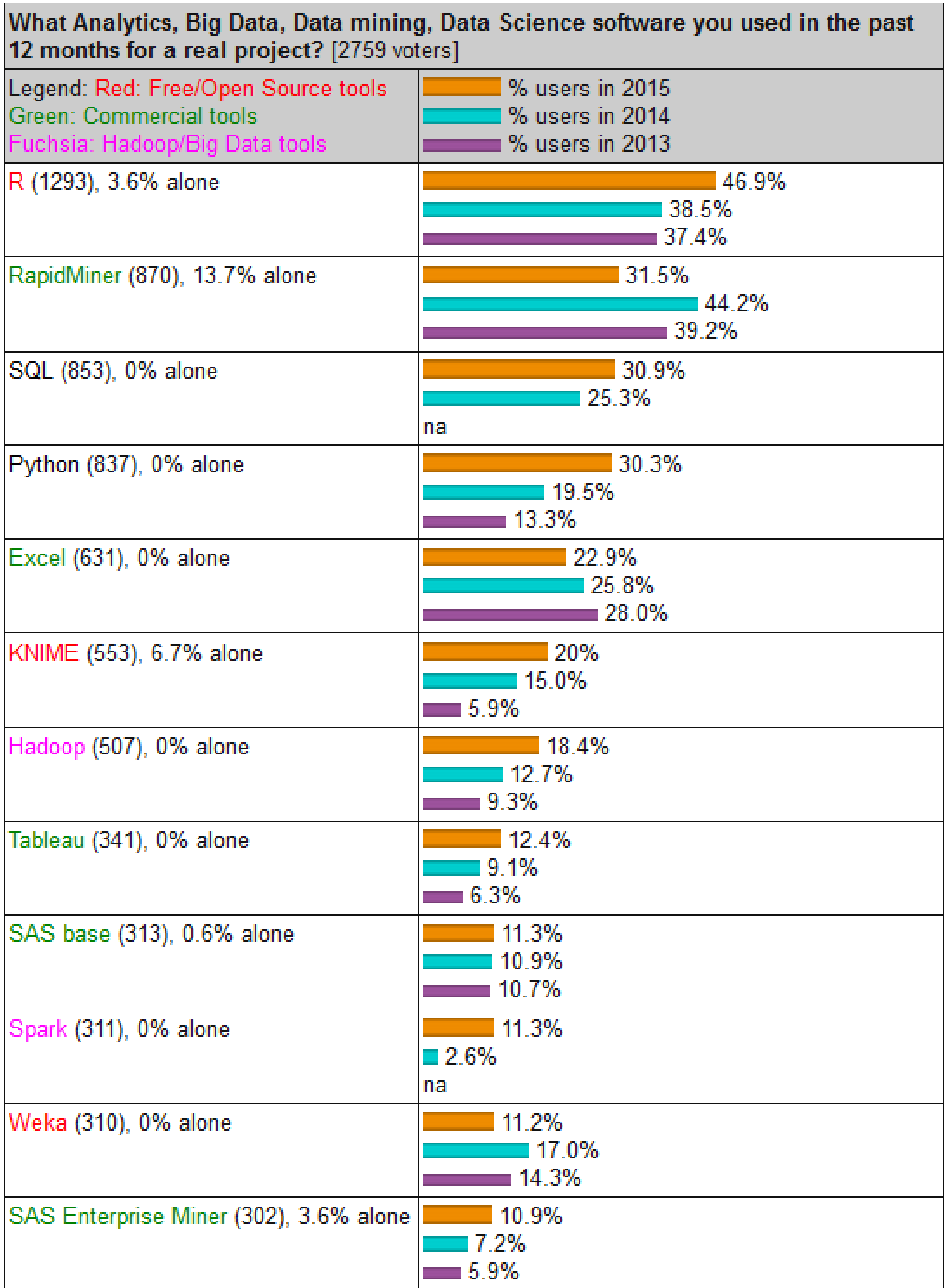
TIOBE, 2015



http://www.tiobe.com

Jun 2015	Jun 2014	Change	Programming Language	Ratings	Change
1	2	∧	Java	17.822%	+1.71%
2	1	∨	C	16.788%	+0.60%
3	4	∧	C++	7.756%	+1.33%
4	5	∧	C#	5.056%	+1.11%
5	3	∨	Objective-C	4.339%	-6.60%
6	8	∧	Python	3.999%	+1.29%
7	10	∧	Visual Basic .NET	3.168%	+1.25%
8	7	∨	PHP	2.868%	+0.02%
9	9		JavaScript	2.295%	+0.30%
10	17	⇈	Delphi/Object Pascal	1.869%	+1.04%
11	-	⇈	Visual Basic	1.839%	+1.84%
12	12		Perl	1.759%	+0.28%
13	23	⇈	R	1.524%	+0.85%
14	-	⇈	Swift	1.440%	+1.44%
15	19	⇈	MATLAB	1.436%	+0.66%
16	13	∨	Ruby	1.359%	-0.03%
17	26	⇈	PL/SQL	1.229%	+0.74%
18	31	⇈	COBOL	0.948%	+0.54%

The 16th annual KDnuggets Software Poll^{2015}

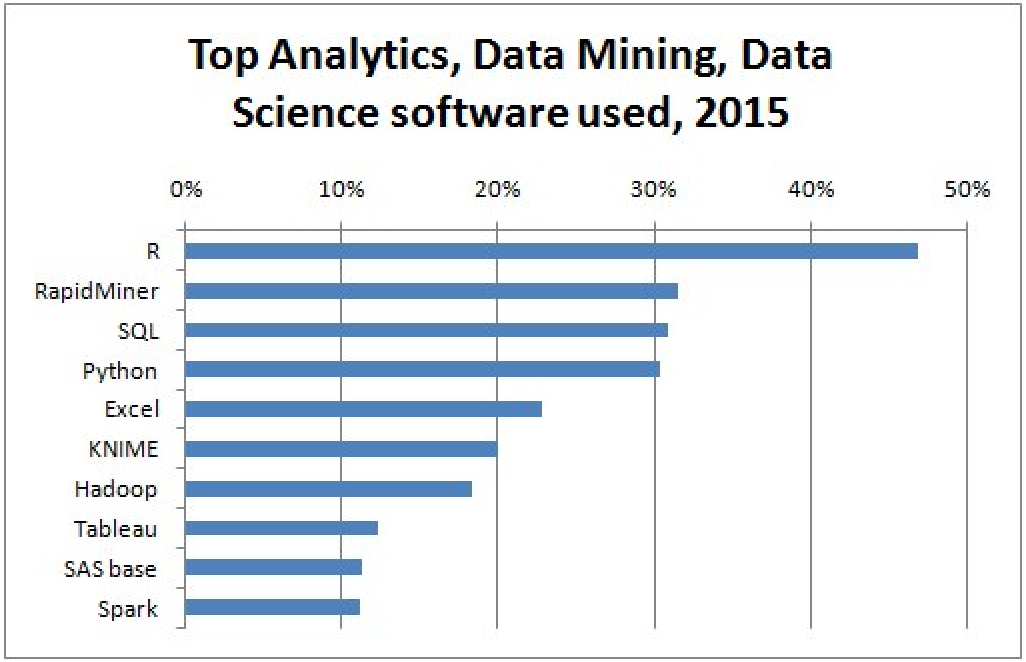


What Analytics, Big Data, Data mining, Data Science software you used in the past 12 months for a real project? [2759 voters]

Legend: Red: Free/Open Source tools
Green: Commercial tools
Fuchsia: Hadoop/Big Data tools

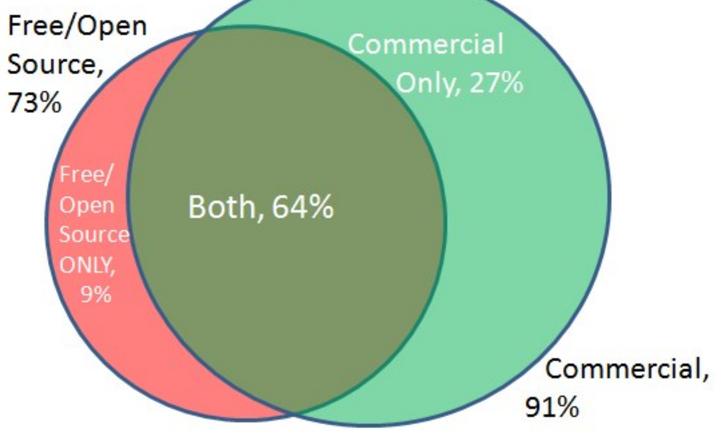
🟧 % users in 2015
🟦 % users in 2014
🟪 % users in 2013

Tool	% users in 2015	% users in 2014	% users in 2013
R (1293), 3.6% alone	46.9%	38.5%	37.4%
RapidMiner (870), 13.7% alone	31.5%	44.2%	39.2%
SQL (853), 0% alone	30.9%	25.3%	na
Python (837), 0% alone	30.3%	19.5%	13.3%
Excel (631), 0% alone	22.9%	25.8%	28.0%
KNIME (553), 6.7% alone	20%	15.0%	5.9%
Hadoop (507), 0% alone	18.4%	12.7%	9.3%
Tableau (341), 0% alone	12.4%	9.1%	6.3%
SAS base (313), 0.6% alone	11.3%	10.9%	10.7%
Spark (311), 0% alone	11.3%	2.6%	na
Weka (310), 0% alone	11.2%	17.0%	14.3%
SAS Enterprise Miner (302), 3.6% alone	10.9%	7.2%	5.9%



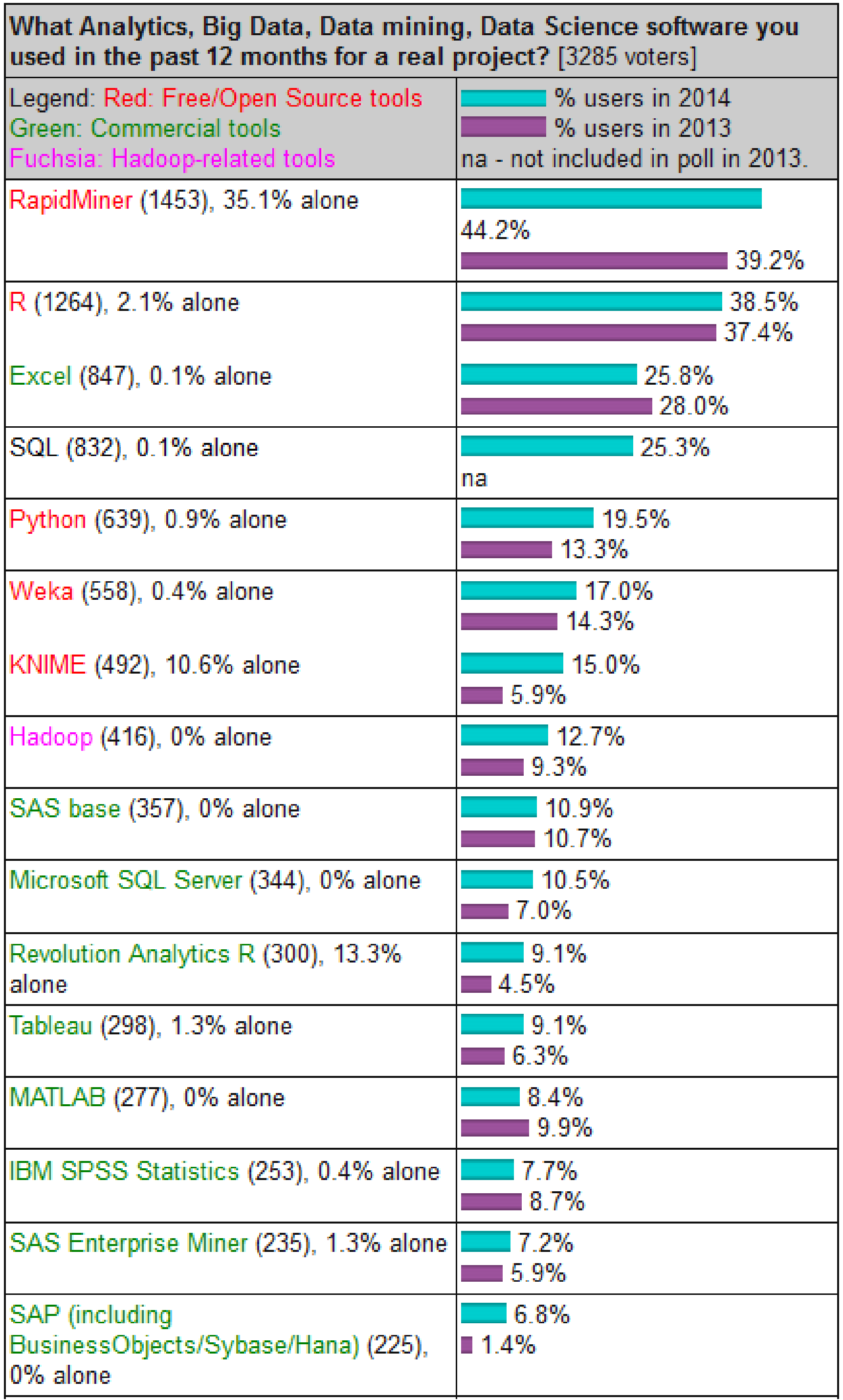
Top Analytics, Data Mining, Data Science software used, 2015

Analytics, Data Mining, Data Science Software Usage, 2015



http://www.kdnuggets.com/2015/05/poll...

The 15th annual KDnuggets Software Poll^{2014}

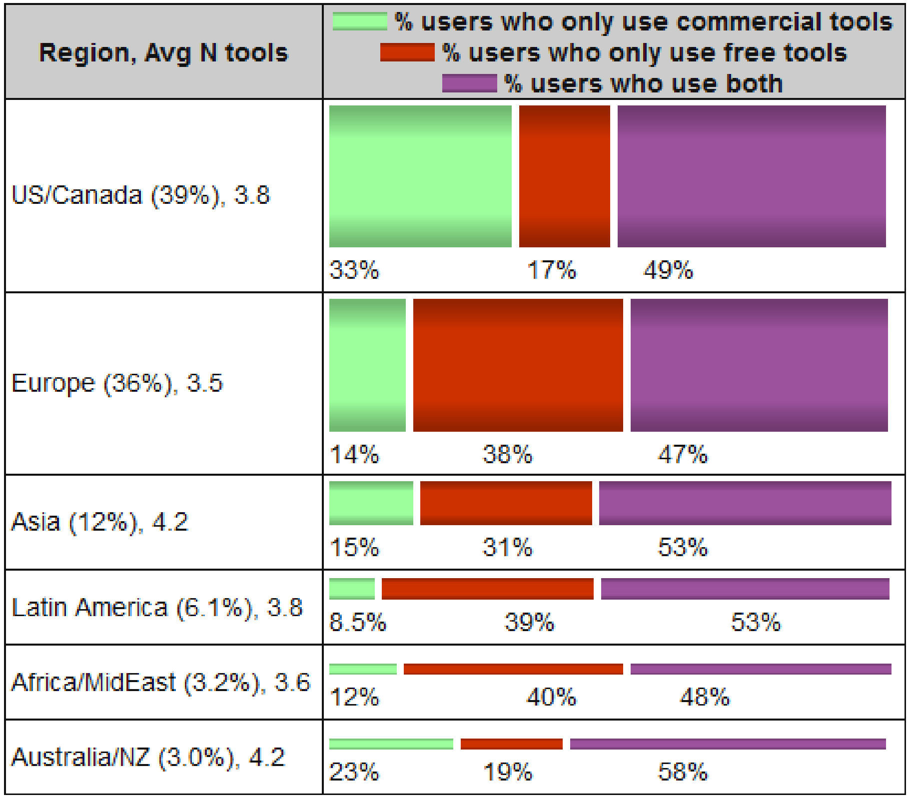


What Analytics, Big Data, Data mining, Data Science software you used in the past 12 months for a real project? [3285 voters]

Legend: Red: Free/Open Source tools
Green: Commercial tools
Fuchsia: Hadoop-related tools

🟦 % users in 2014
🟪 % users in 2013
na - not included in poll in 2013.

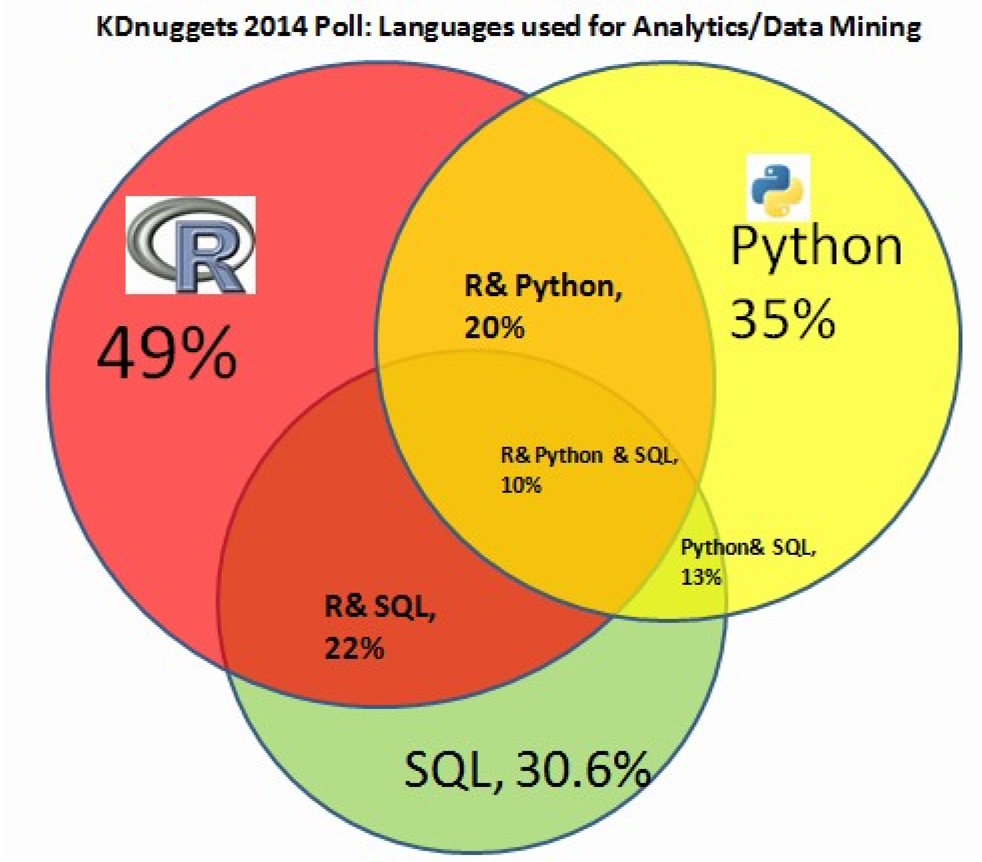
Tool	% users in 2014	% users in 2013
RapidMiner (1453), 35.1% alone	44.2%	39.2%
R (1264), 2.1% alone	38.5%	37.4%
Excel (847), 0.1% alone	25.8%	28.0%
SQL (832), 0.1% alone	25.3%	na
Python (639), 0.9% alone	19.5%	13.3%
Weka (558), 0.4% alone	17.0%	14.3%
KNIME (492), 10.6% alone	15.0%	5.9%
Hadoop (416), 0% alone	12.7%	9.3%
SAS base (357), 0% alone	10.9%	10.7%
Microsoft SQL Server (344), 0% alone	10.5%	7.0%
Revolution Analytics R (300), 13.3% alone	9.1%	4.5%
Tableau (298), 1.3% alone	9.1%	6.3%
MATLAB (277), 0% alone	8.4%	9.9%
IBM SPSS Statistics (253), 0.4% alone	7.7%	8.7%
SAS Enterprise Miner (235), 1.3% alone	7.2%	5.9%
SAP (including BusinessObjects/Sybase/Hana) (225), 0% alone	6.8%	1.4%

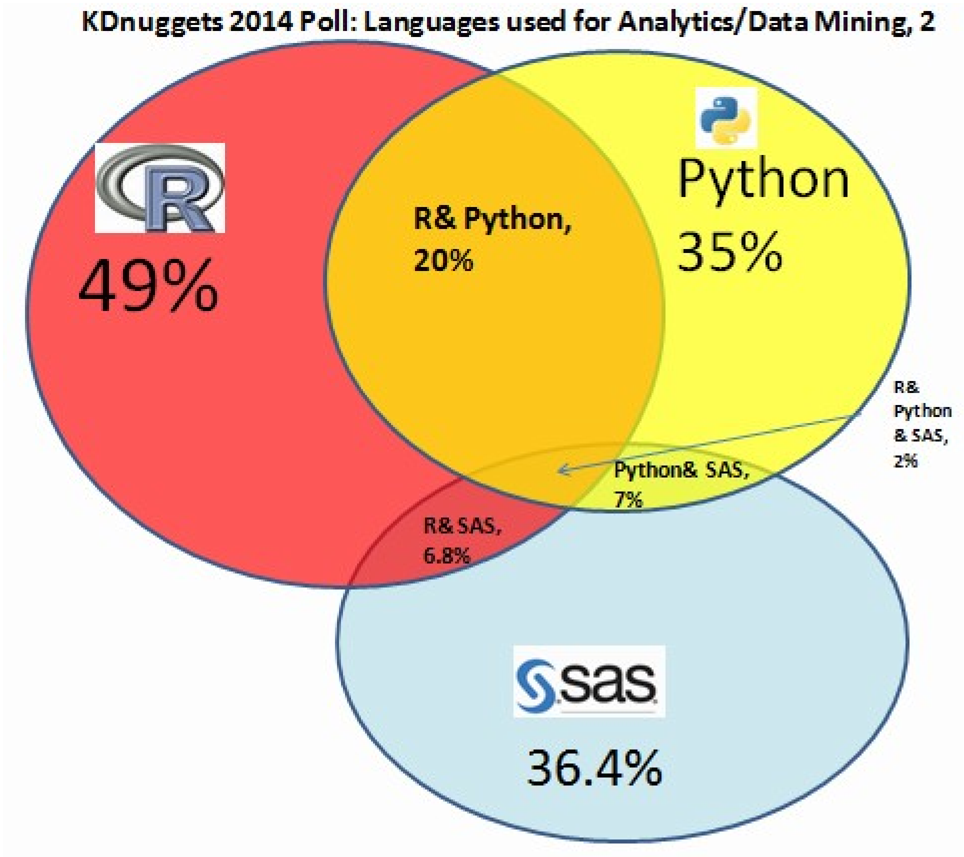


Region, Avg N tools	🟩 % users who only use commercial tools	🟥 % users who only use free tools	🟪 % users who use both
US/Canada (39%), 3.8	33%	17%	49%
Europe (36%), 3.5	14%	38%	47%
Asia (12%), 4.2	15%	31%	53%
Latin America (6.1%), 3.8	8.5%	39%	53%
Africa/MidEast (3.2%), 3.6	12%	40%	48%
Australia/NZ (3.0%), 4.2	23%	19%	58%

http://www.kdnuggets.com/2014/08...

KDnuggets^{2014}: R vs. Python vs. SAS vs. SQL

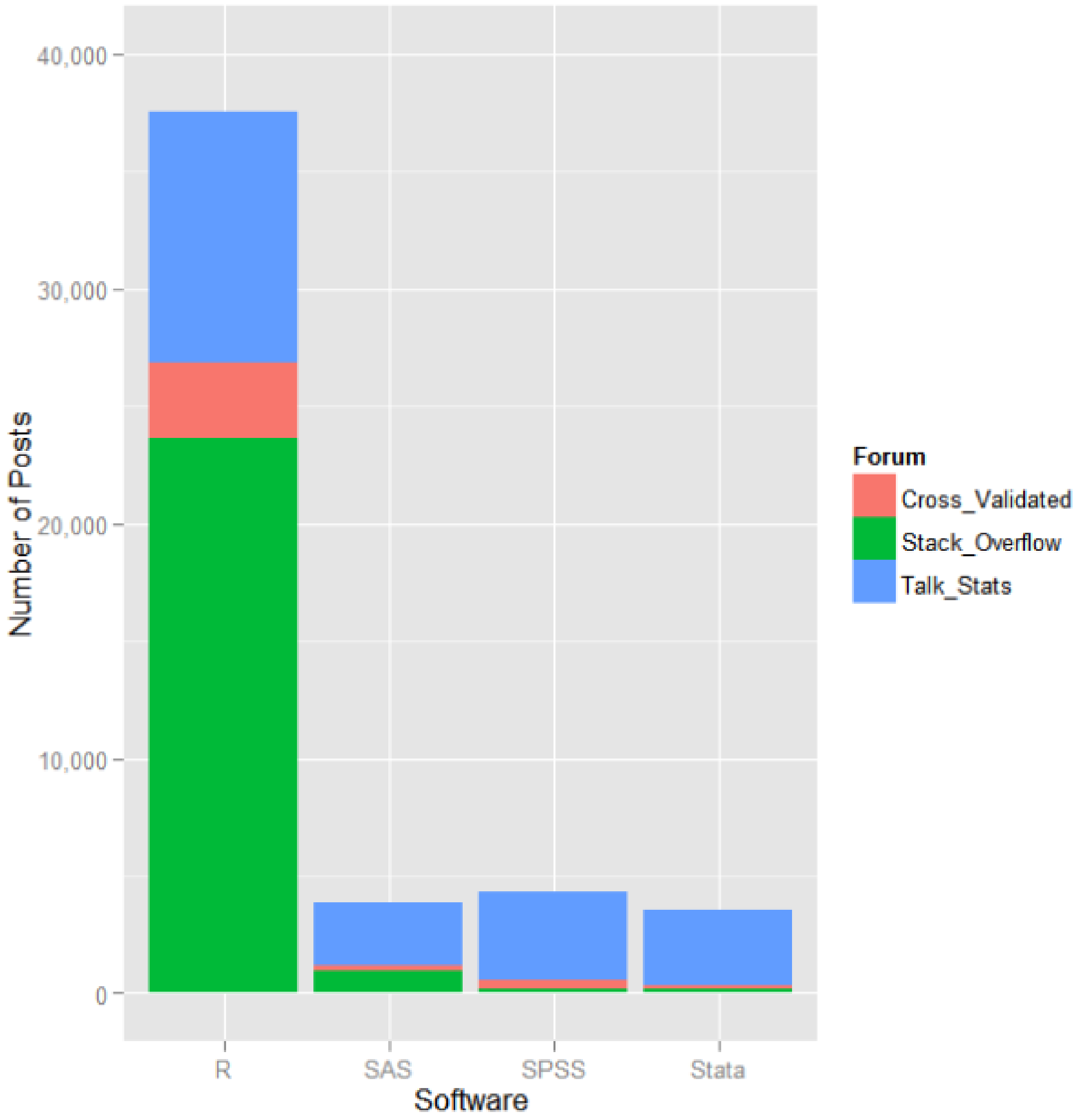

KDnuggets 2014 Poll: Languages used for Analytics/Data Mining


KDnuggets 2014 Poll: Languages used for Analytics/Data Mining, 2

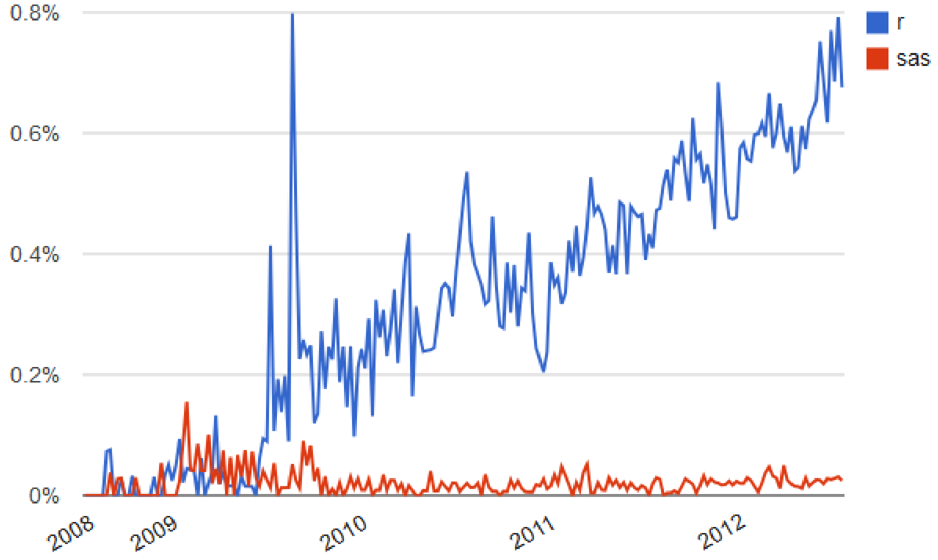
http://www.kdnuggets.com/2014/08...

R quickly gains **high position** in statistics 2011-2013

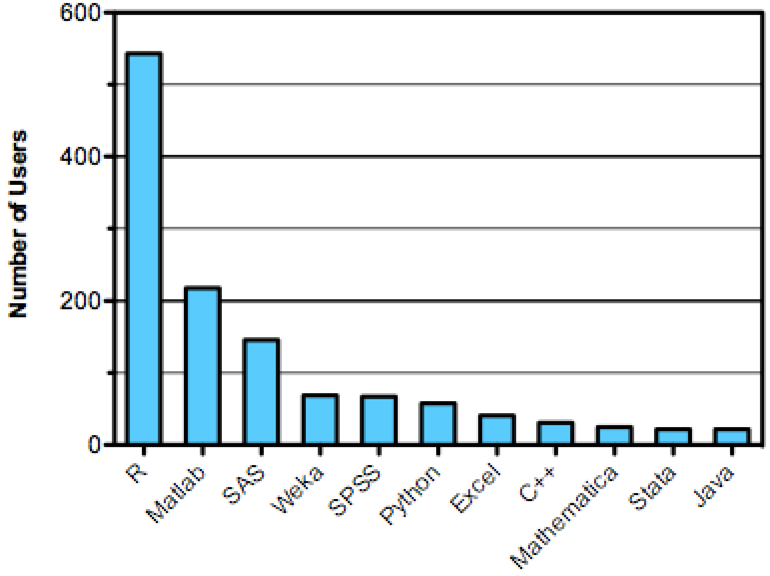
http://r4stats.com/articles/popularity/



Number of posts per software on each forum on 2/10/2013

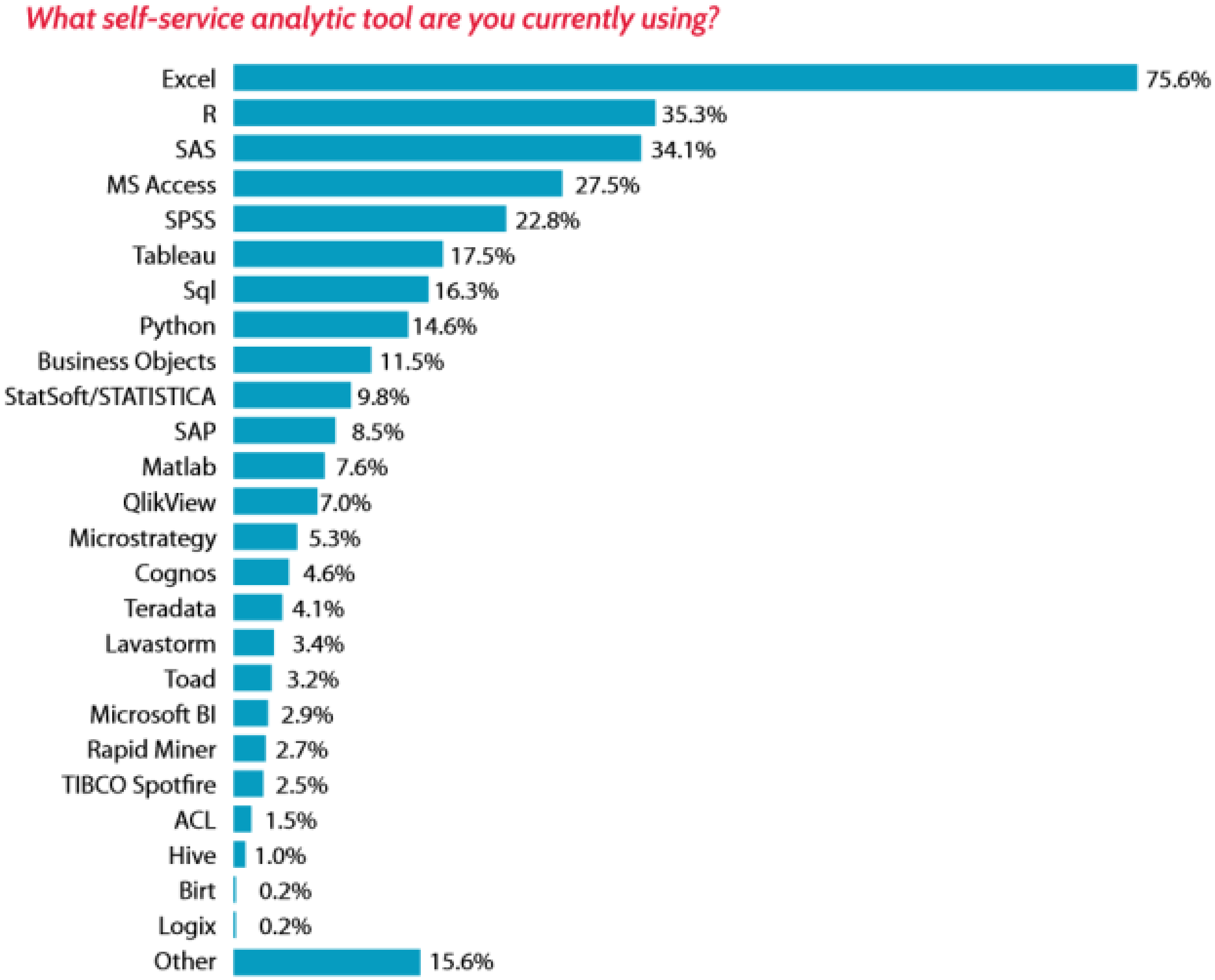


Number of R- or SAS-related posts to Stack Overflow by week



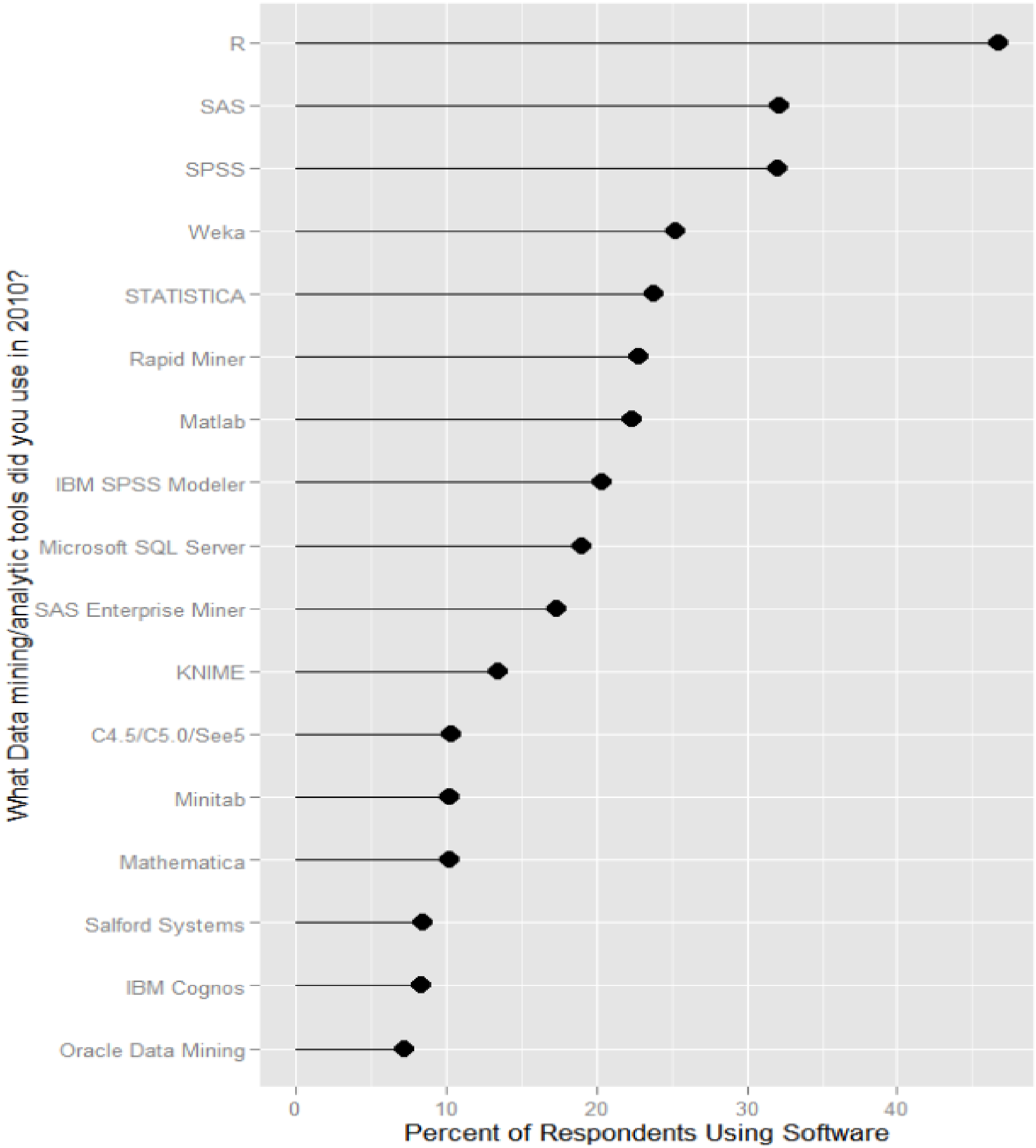
Software used in data analysis competitions in 2011.

R quickly gains high position in statistics^{2013}

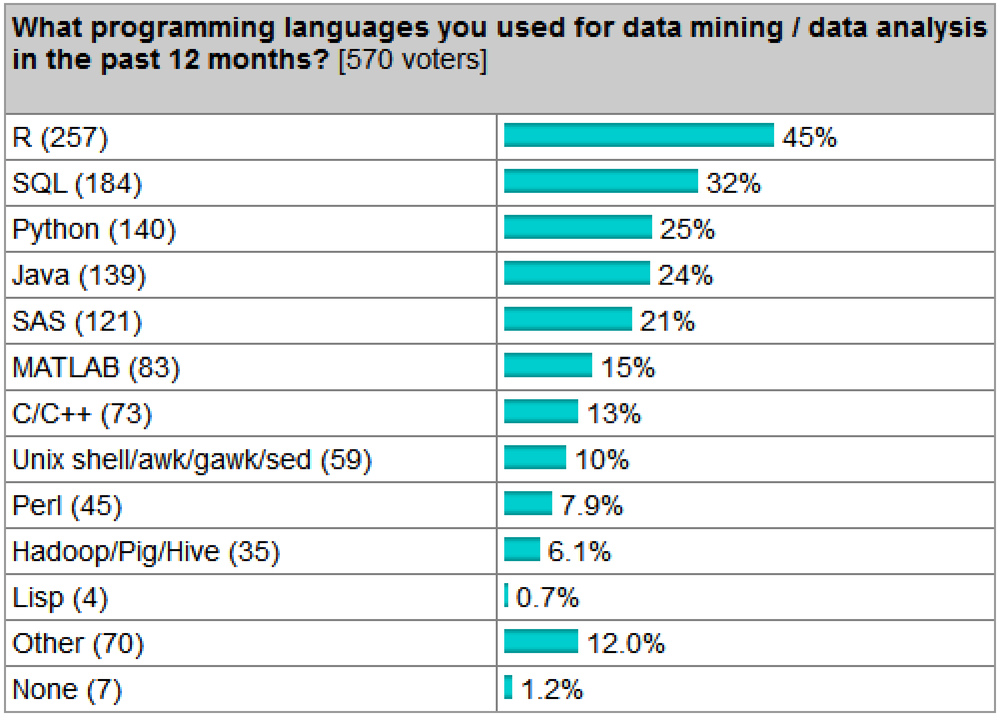


http://r4stats.com/articles/popularity/

Rexer Analytics Survey^{2010}



Rexer Analytics Survey 2010 results for data mining/analytic tools



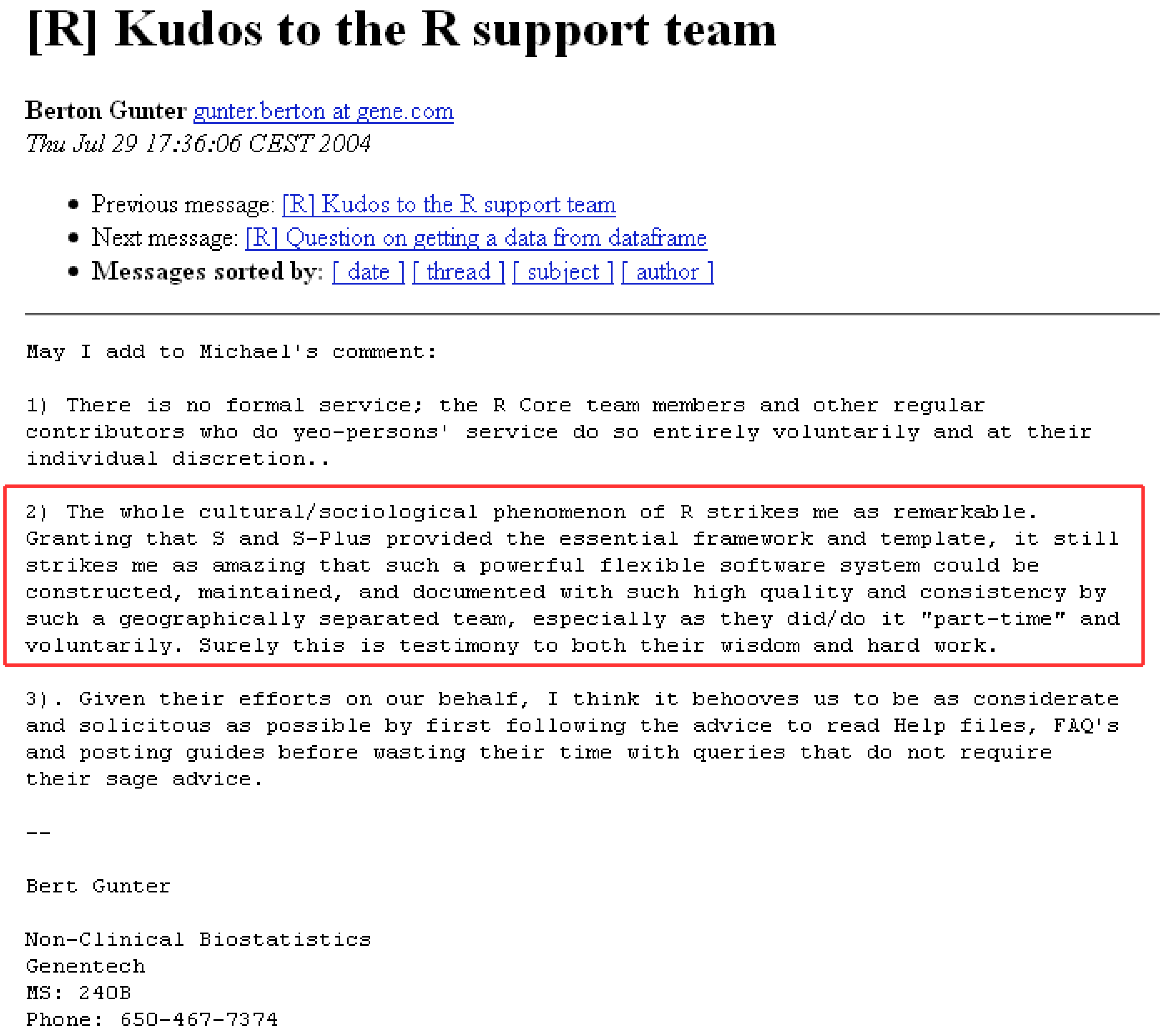
What programming languages you used for data mining / data analysis in the past 12 months? [570 voters]	
R (257)	45%
SQL (184)	32%
Python (140)	25%
Java (139)	24%
SAS (121)	21%
MATLAB (83)	15%
C/C++ (73)	13%
Unix shell/awk/gawk/sed (59)	10%
Perl (45)	7.9%
Hadoop/Pig/Hive (35)	6.1%
Lisp (4)	0.7%
Other (70)	12.0%
None (7)	1.2%

Soon, they'll be speaking R on the subway

Michael Rennie
(giving 'Kudos to the R support team')
Fortunes(68)

http://r4stats.com/articles/popularity/

Ancient history: R was already popular in 2004



[R] Kudos to the R support team

Berton Gunter gunter.berton at gene.com
Thu Jul 29 17:36:06 CEST 2004

- Previous message: [R] Kudos to the R support team
- Next message: [R] Question on getting a data from dataframe
- **Messages sorted by:** [date] [thread] [subject] [author]

May I add to Michael's comment:

1) There is no formal service; the R Core team members and other regular contributors who do yeo-persons' service do so entirely voluntarily and at their individual discretion..

2) The whole cultural/sociological phenomenon of R strikes me as remarkable. Granting that S and S-Plus provided the essential framework and template, it still strikes me as amazing that such a powerful flexible software system could be constructed, maintained, and documented with such high quality and consistency by such a geographically separated team, especially as they did/do it "part-time" and voluntarily. Surely this is testimony to both their wisdom and hard work.

3). Given their efforts on our behalf, I think it behooves us to be as considerate and solicitous as possible by first following the advice to read Help files, FAQ's and posting guides before wasting their time with queries that do not require their sage advice.

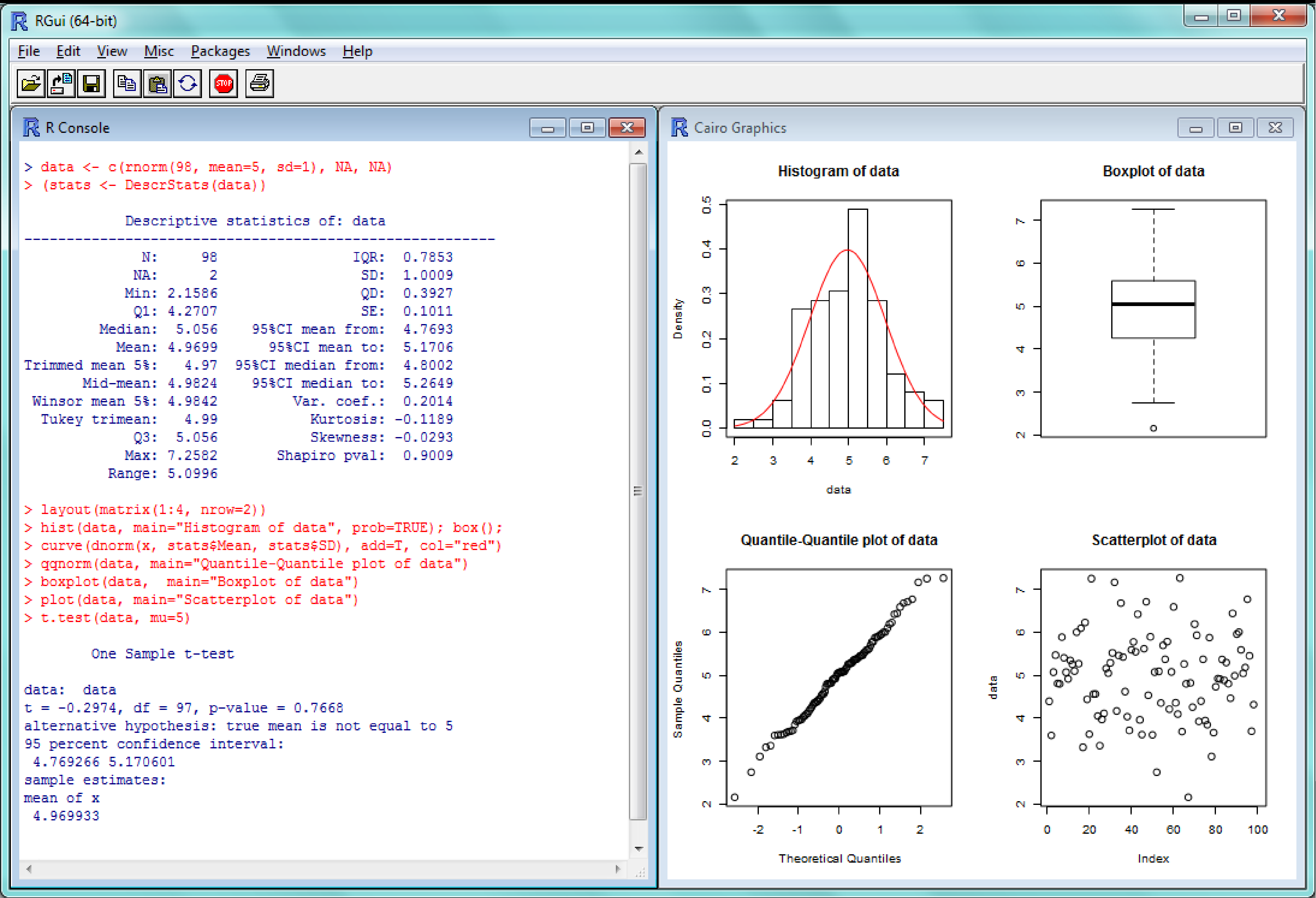
--

Bert Gunter

Non-Clinical Biostatistics
Genentech
MS: 240B
Phone: 650-467-7374

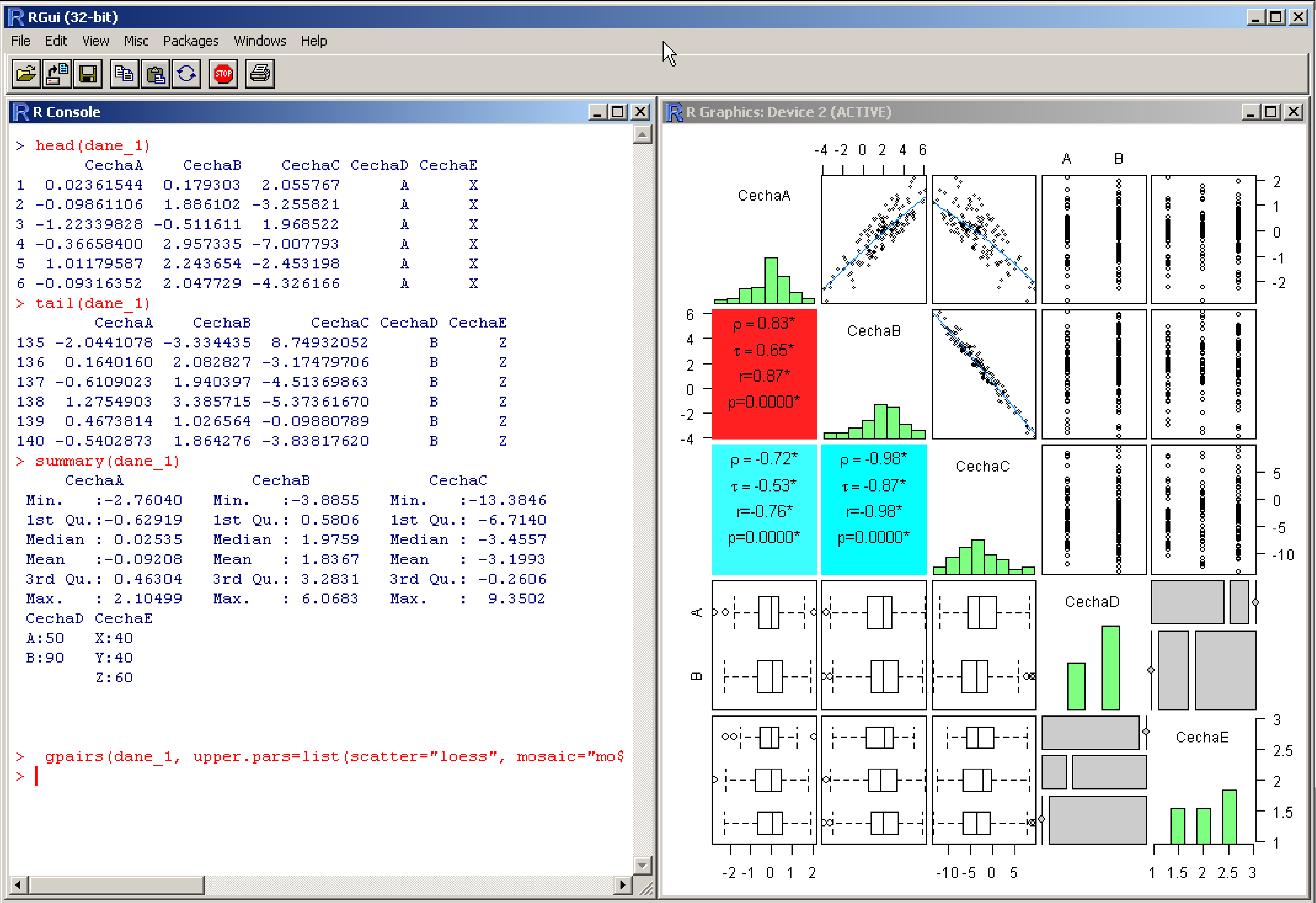
Demonstrative screenshots

fiRst insight: the main program window



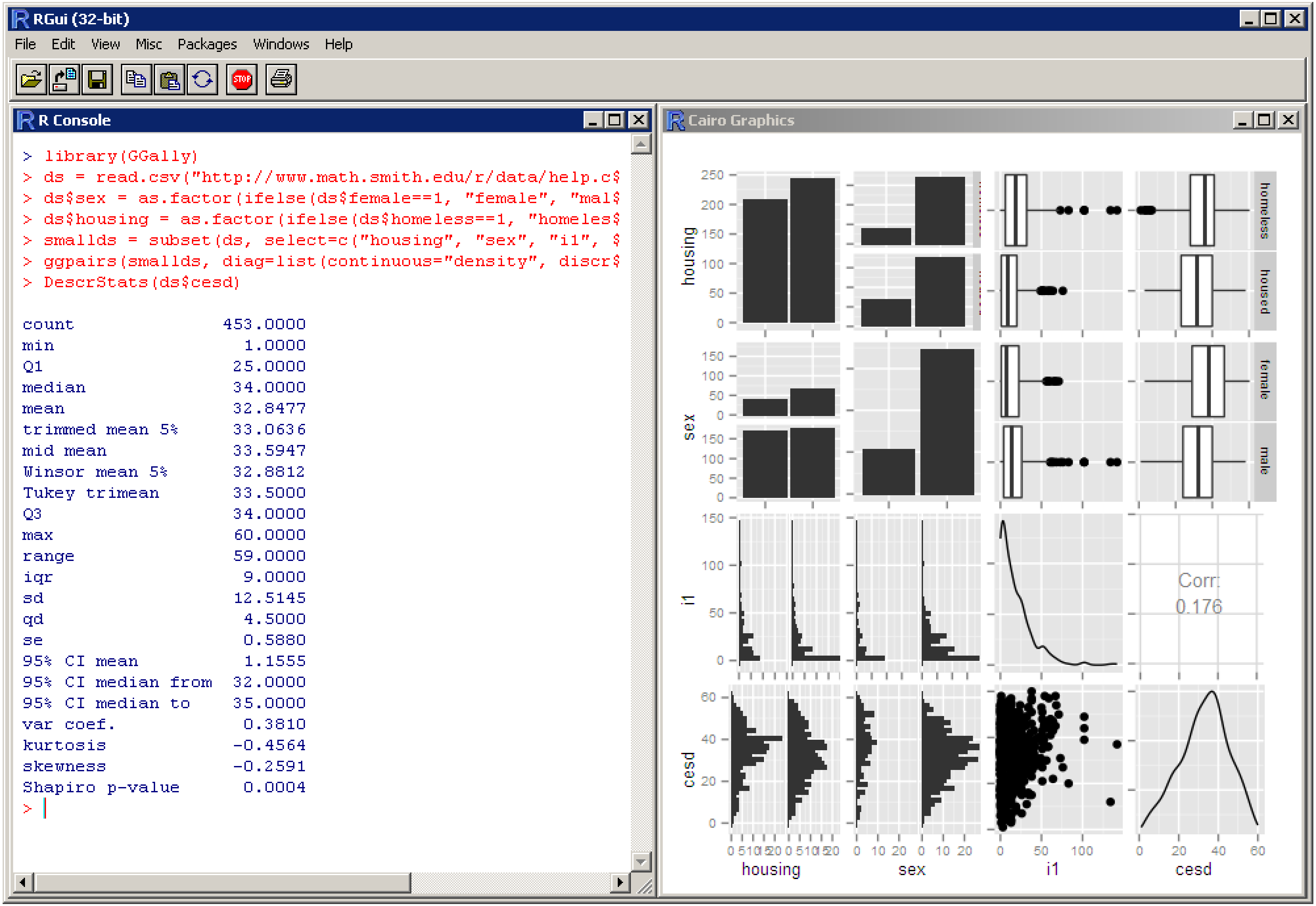
```
> data <- c(rnorm(98, mean=5, sd=1), NA, NA)
> (stats <- DescrStats(data))

            Descriptive statistics of: data
---------------------------------------------------------
             N:     98                 IQR:  0.7853
            NA:      2                  SD:  1.0009
           Min: 2.1586                  QD:  0.3927
            Q1: 4.2707                  SE:  0.1011
        Median:  5.056     95%CI mean from:  4.7693
          Mean: 4.9699       95%CI mean to:  5.1706
Trimmed mean 5%:   4.97   95%CI median from:  4.8002
      Mid-mean: 4.9824     95%CI median to:  5.2649
 Winsor mean 5%: 4.9842          Var. coef.:  0.2014
  Tukey trimean:   4.99            Kurtosis: -0.1189
            Q3:  5.056            Skewness: -0.0293
           Max: 7.2582        Shapiro pval:  0.9009
         Range: 5.0996

> layout(matrix(1:4, nrow=2))
> hist(data, main="Histogram of data", prob=TRUE); box();
> curve(dnorm(x, stats$Mean, stats$SD), add=T, col="red")
> qqnorm(data, main="Quantile-Quantile plot of data")
> boxplot(data,  main="Boxplot of data")
> plot(data, main="Scatterplot of data")
> t.test(data, mu=5)

        One Sample t-test

data:  data
t = -0.2974, df = 97, p-value = 0.7668
alternative hypothesis: true mean is not equal to 5
95 percent confidence interval:
 4.769266 5.170601
sample estimates:
mean of x
 4.969933
```

...some quick'n'dirty data inspection...



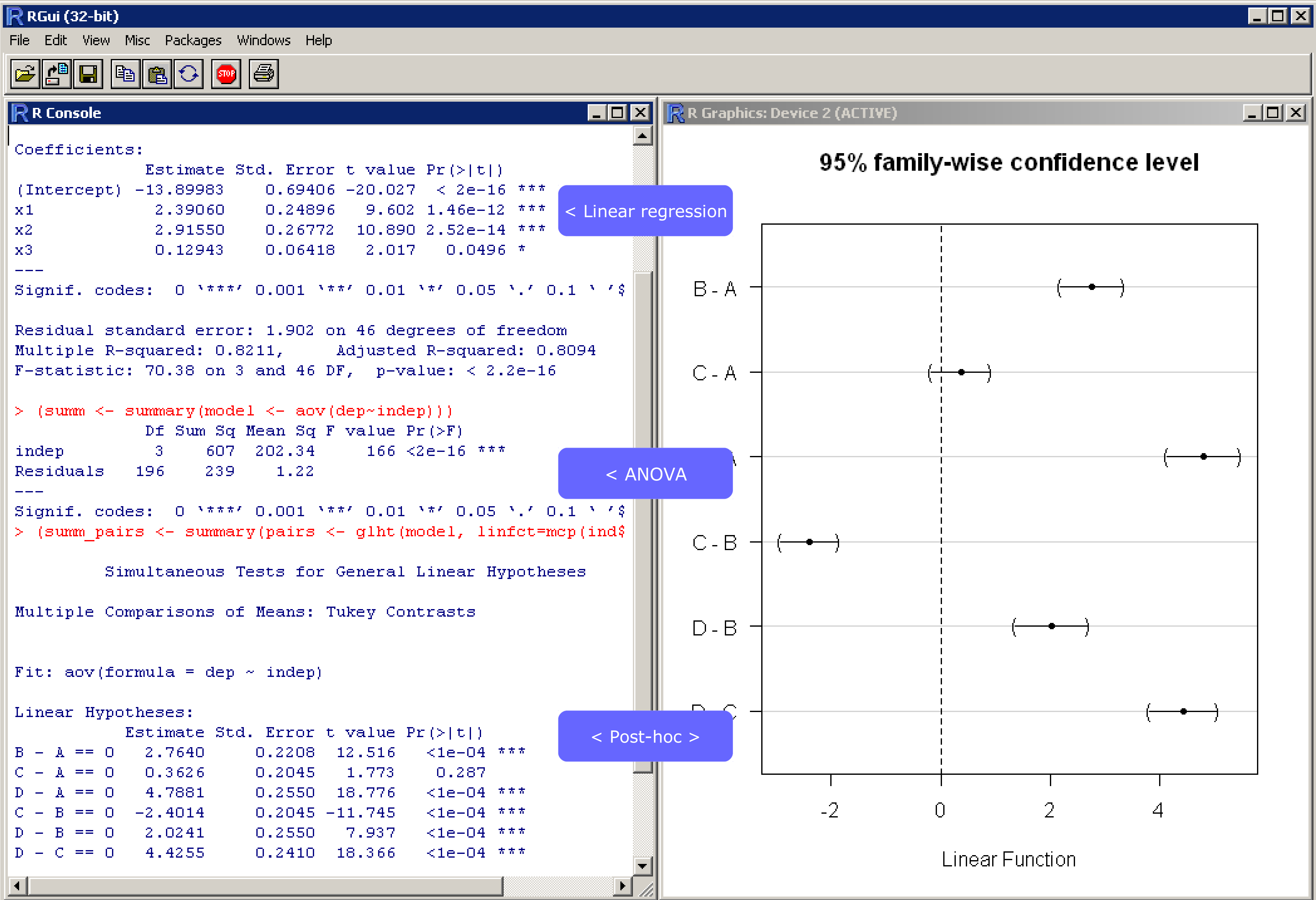
```
> head(dane_1)
       CechaA    CechaB    CechaC CechaD CechaE
1  0.02361544  0.179303  2.055767      A      X
2 -0.09861106  1.886102 -3.255821      A      X
3 -1.22339828 -0.511611  1.968522      A      X
4 -0.36658400  2.957335 -7.007793      A      X
5  1.01179587  2.243654 -2.453198      A      X
6 -0.09316352  2.047729 -4.326166      A      X
> tail(dane_1)
         CechaA    CechaB      CechaC CechaD CechaE
135 -2.0441078 -3.334435  8.74932052      B      Z
136  0.1640160  2.082827 -3.17479706      B      Z
137 -0.6109023  1.940397 -4.51369863      B      Z
138  1.2754903  3.385715 -5.37361670      B      Z
139  0.4673814  1.026564 -0.09880789      B      Z
140 -0.5402873  1.864276 -3.83817620      B      Z
> summary(dane_1)
     CechaA              CechaB             CechaC       
 Min.   :-2.76040   Min.   :-3.8855   Min.   :-13.3846  
 1st Qu.:-0.62919   1st Qu.: 0.5806   1st Qu.: -6.7140  
 Median : 0.02535   Median : 1.9759   Median : -3.4557  
 Mean   :-0.09208   Mean   : 1.8367   Mean   : -3.1993  
 3rd Qu.: 0.46304   3rd Qu.: 3.2831   3rd Qu.: -0.2606  
 Max.   : 2.10499   Max.   : 6.0683   Max.   :  9.3502  
 CechaD CechaE
 A:50   X:40  
 B:90   Y:40  
        Z:60  



>  gpairs(dane_1, upper.pars=list(scatter="loess", mosaic="mo$
> 
```

...quick'n'dirty data inspection other way...



```
> library(GGally)
> ds = read.csv("http://www.math.smith.edu/r/data/help.c$
> ds$sex = as.factor(ifelse(ds$female==1, "female", "mal$
> ds$housing = as.factor(ifelse(ds$homeless==1, "homeles$
> smallds = subset(ds, select=c("housing", "sex", "i1", $
> ggpairs(smallds, diag=list(continuous="density", discr$
> DescrStats(ds$cesd)

count                 453.0000
min                     1.0000
Q1                     25.0000
median                 34.0000
mean                   32.8477
trimmed mean 5%        33.0636
mid mean               33.5947
Winsor mean 5%         32.8812
Tukey trimean          33.5000
Q3                     34.0000
max                    60.0000
range                  59.0000
iqr                     9.0000
sd                     12.5145
qd                      4.5000
se                      0.5880
95% CI mean             1.1555
95% CI median from     32.0000
95% CI median to       35.0000
var coef.               0.3810
kurtosis               -0.4564
skewness               -0.2591
Shapiro p-value         0.0004
> 
```

...some linear modeling...



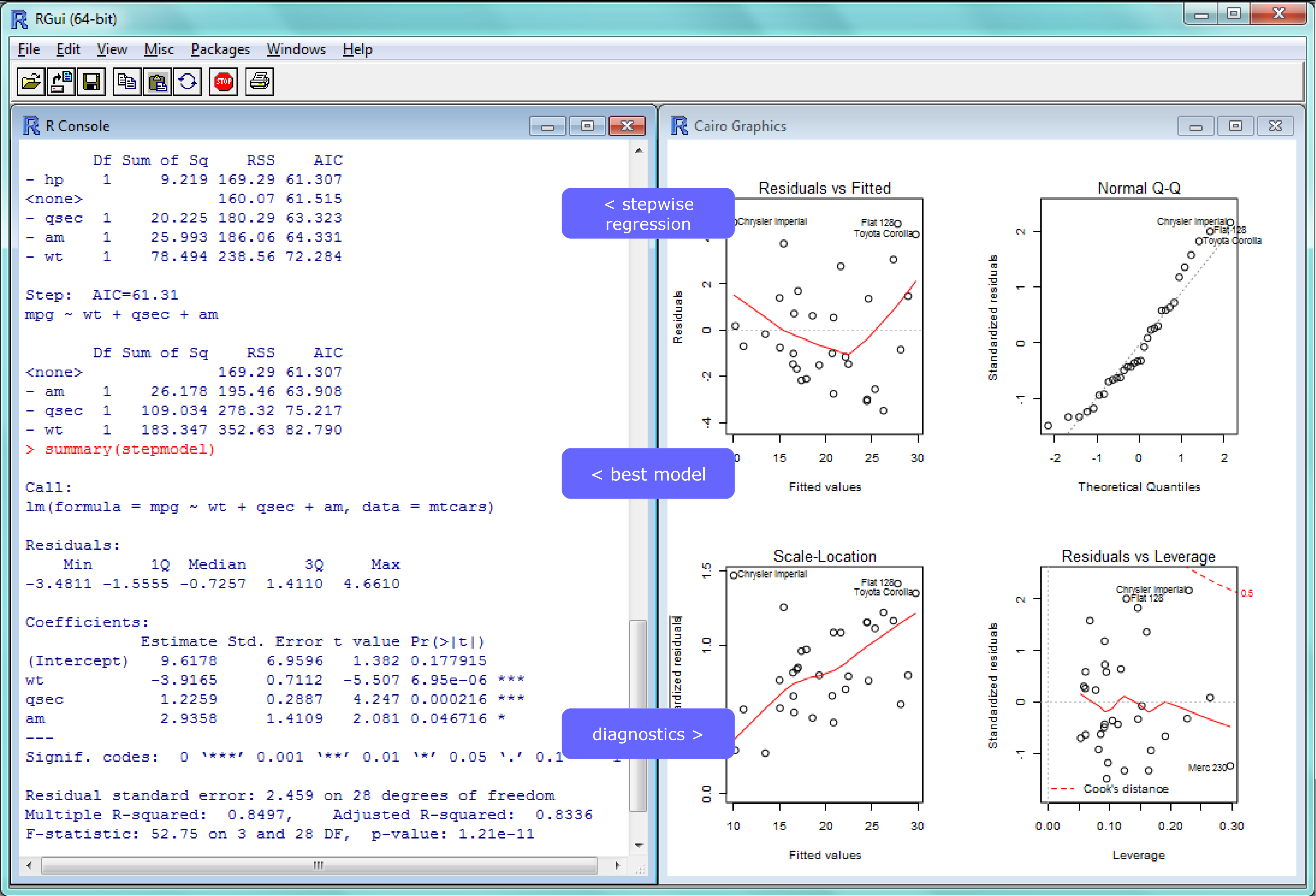
```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -13.89983    0.69406 -20.027  < 2e-16 ***
x1            2.39060    0.24896   9.602 1.46e-12 ***
x2            2.91550    0.26772  10.890 2.52e-14 ***
x3            0.12943    0.06418   2.017   0.0496 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’$

Residual standard error: 1.902 on 46 degrees of freedom
Multiple R-squared: 0.8211,     Adjusted R-squared: 0.8094
F-statistic: 70.38 on 3 and 46 DF,  p-value: < 2.2e-16

> (summ <- summary(model <- aov(dep~indep)))
             Df Sum Sq Mean Sq F value Pr(>F)
indep         3    607  202.34     166 <2e-16 ***
Residuals   196    239    1.22
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’$
> (summ_pairs <- summary(pairs <- glht(model, linfct=mcp(ind$

         Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: Tukey Contrasts


Fit: aov(formula = dep ~ indep)

Linear Hypotheses:
           Estimate Std. Error t value Pr(>|t|)
B - A == 0   2.7640     0.2208  12.516   <1e-04 ***
C - A == 0   0.3626     0.2045   1.773    0.287
D - A == 0   4.7881     0.2550  18.776   <1e-04 ***
C - B == 0  -2.4014     0.2045 -11.745   <1e-04 ***
D - B == 0   2.0241     0.2550   7.937   <1e-04 ***
D - C == 0   4.4255     0.2410  18.366   <1e-04 ***
```

< Linear regression

< ANOVA

< Post-hoc >

...some linear modeling...



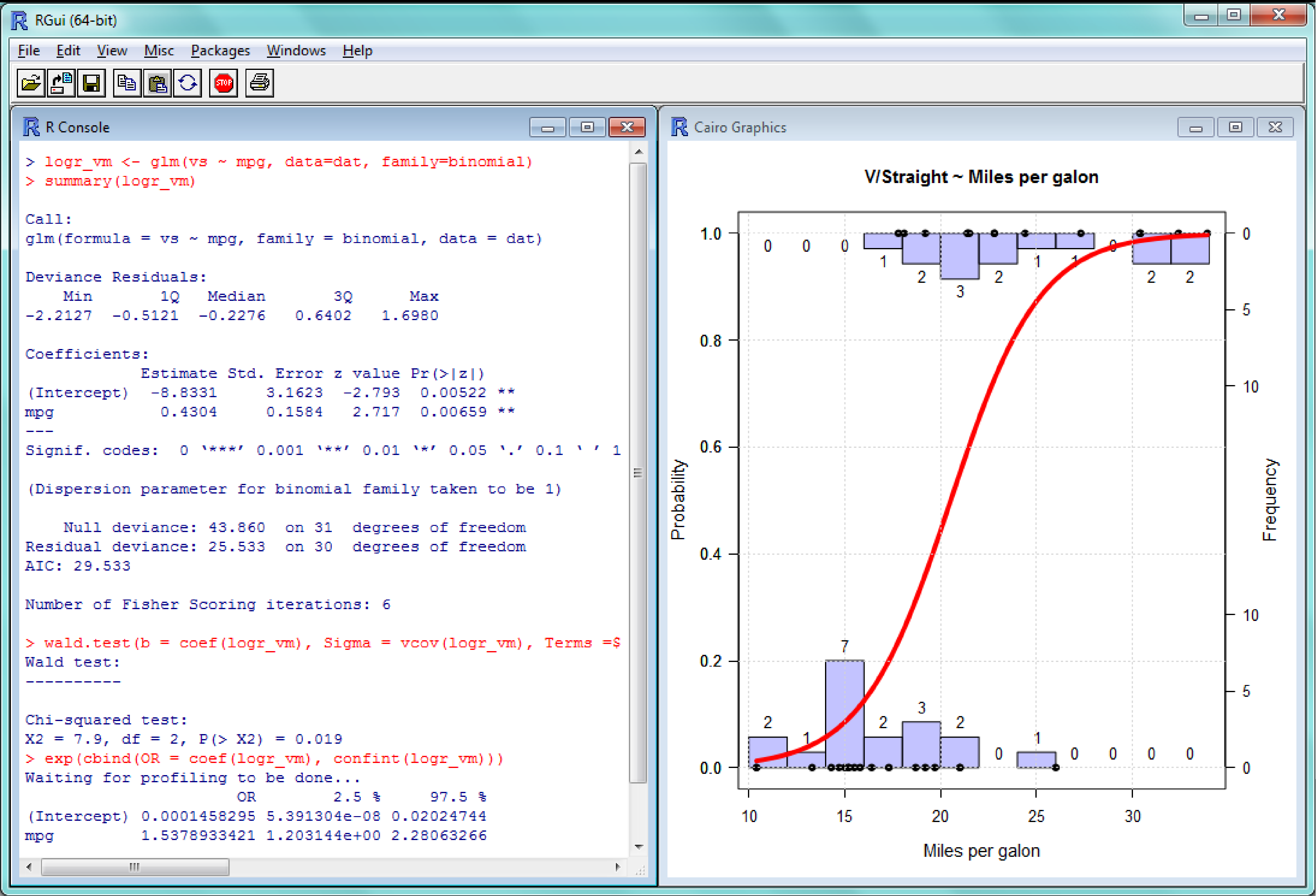
```
         Df Sum of Sq    RSS    AIC
- hp      1     9.219 169.29 61.307
<none>                160.07 61.515
- qsec    1    20.225 180.29 63.323
- am      1    25.993 186.06 64.331
- wt      1    78.494 238.56 72.284

Step:  AIC=61.31
mpg ~ wt + qsec + am

         Df Sum of Sq    RSS    AIC
<none>                169.29 61.307
- am      1    26.178 195.46 63.908
- qsec    1   109.034 278.32 75.217
- wt      1   183.347 352.63 82.790
> summary(stepmodel)

Call:
lm(formula = mpg ~ wt + qsec + am, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4811 -1.5555 -0.7257  1.4110  4.6610

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.6178     6.9596   1.382 0.177915
wt           -3.9165     0.7112  -5.507 6.95e-06 ***
qsec          1.2259     0.2887   4.247 0.000216 ***
am            2.9358     1.4109   2.081 0.046716 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1      1

Residual standard error: 2.459 on 28 degrees of freedom
Multiple R-squared:  0.8497,    Adjusted R-squared:  0.8336
F-statistic: 52.75 on 3 and 28 DF,  p-value: 1.21e-11
```

< stepwise regression

< best model

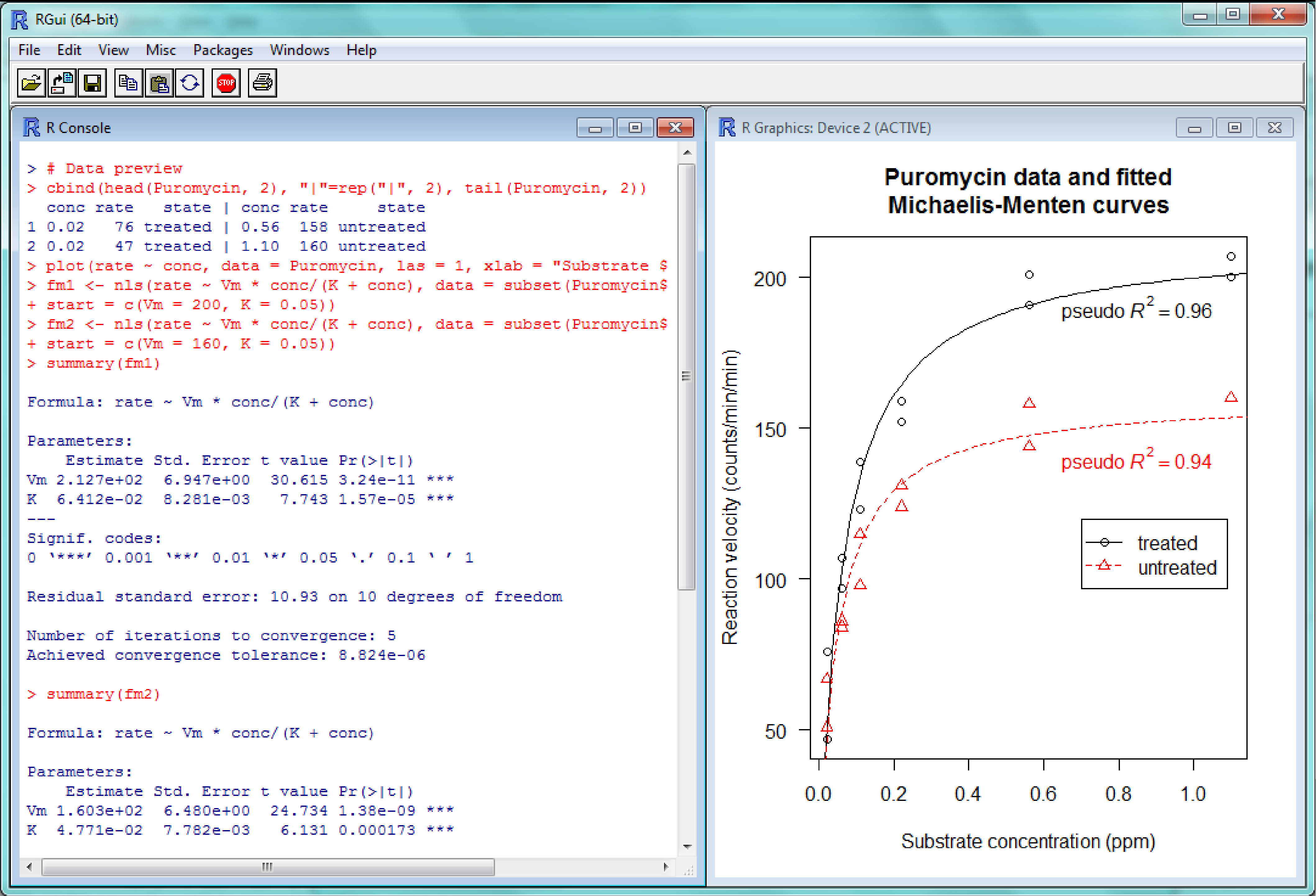
diagnostics >

...some GLM modeling...



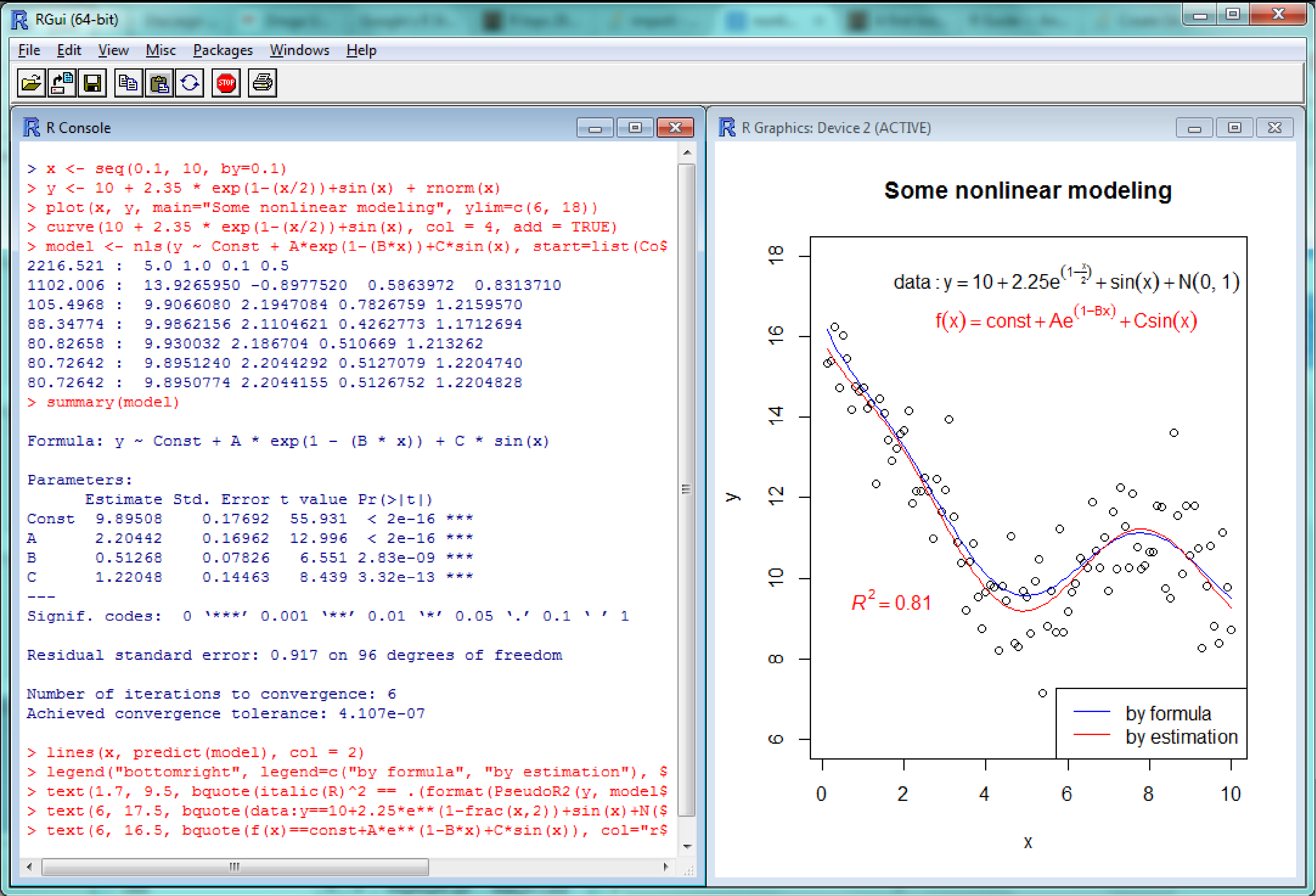
```
> logr_vm <- glm(vs ~ mpg, data=dat, family=binomial)
> summary(logr_vm)

Call:
glm(formula = vs ~ mpg, family = binomial, data = dat)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2127  -0.5121  -0.2276   0.6402   1.6980

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -8.8331     3.1623  -2.793  0.00522 **
mpg           0.4304     0.1584   2.717  0.00659 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 43.860  on 31  degrees of freedom
Residual deviance: 25.533  on 30  degrees of freedom
AIC: 29.533

Number of Fisher Scoring iterations: 6

> wald.test(b = coef(logr_vm), Sigma = vcov(logr_vm), Terms =$
Wald test:
----------

Chi-squared test:
X2 = 7.9, df = 2, P(> X2) = 0.019
> exp(cbind(OR = coef(logr_vm), confint(logr_vm)))
Waiting for profiling to be done...
                      OR        2.5 %     97.5 %
(Intercept) 0.0001458295 5.391304e-08 0.02024744
mpg         1.5378933421 1.203144e+00 2.28063266
```

...and nonlinear modeling...



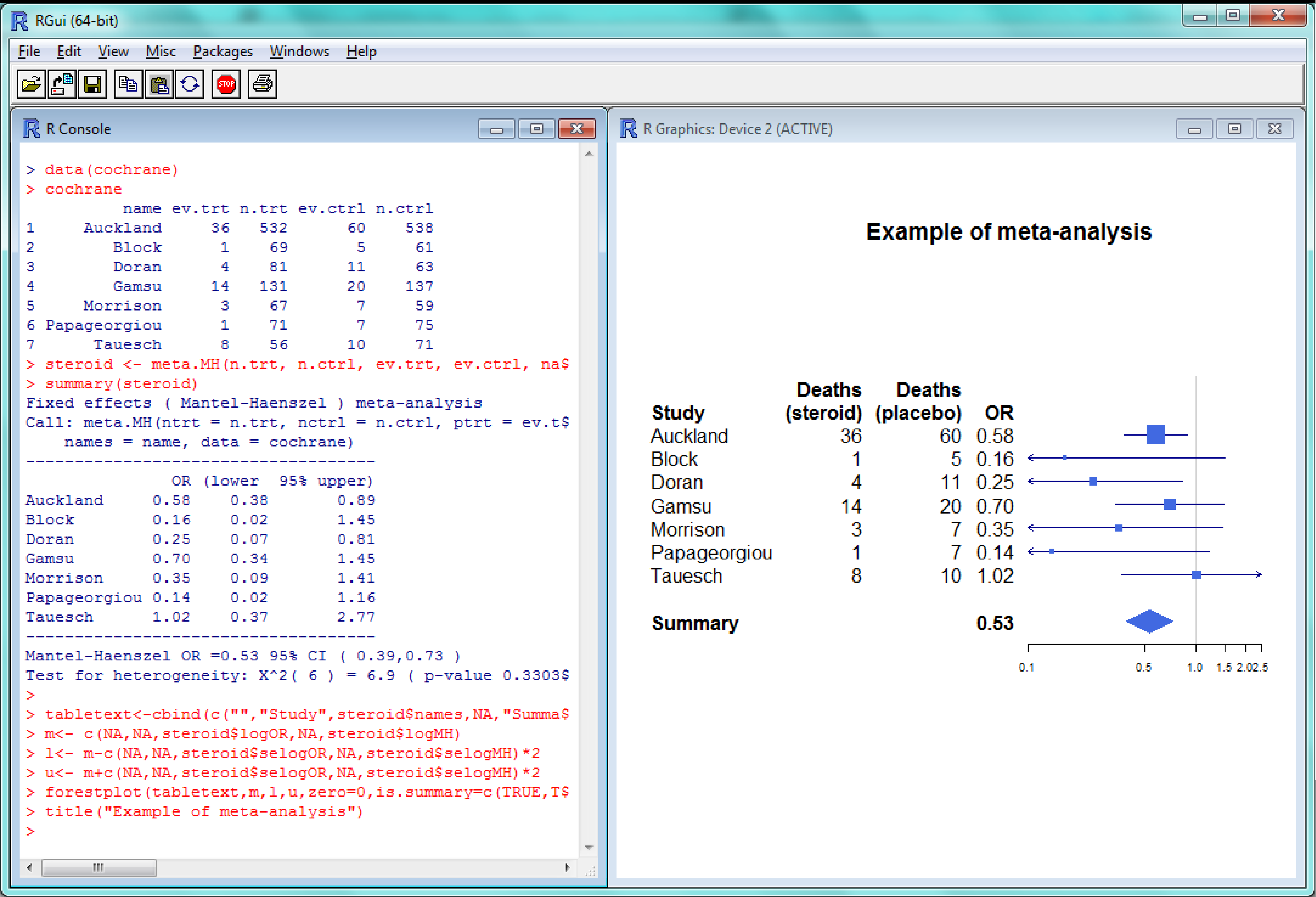
```
> # Data preview
> cbind(head(Puromycin, 2), "|"=rep("|", 2), tail(Puromycin, 2))
  conc rate   state | conc rate     state
1 0.02   76 treated | 0.56  158 untreated
2 0.02   47 treated | 1.10  160 untreated
> plot(rate ~ conc, data = Puromycin, las = 1, xlab = "Substrate $
> fm1 <- nls(rate ~ Vm * conc/(K + conc), data = subset(Puromycin$
+ start = c(Vm = 200, K = 0.05))
> fm2 <- nls(rate ~ Vm * conc/(K + conc), data = subset(Puromycin$
+ start = c(Vm = 160, K = 0.05))
> summary(fm1)

Formula: rate ~ Vm * conc/(K + conc)

Parameters:
    Estimate Std. Error t value Pr(>|t|)
Vm 2.127e+02  6.947e+00  30.615 3.24e-11 ***
K  6.412e-02  8.281e-03   7.743 1.57e-05 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.93 on 10 degrees of freedom

Number of iterations to convergence: 5
Achieved convergence tolerance: 8.824e-06

> summary(fm2)

Formula: rate ~ Vm * conc/(K + conc)

Parameters:
    Estimate Std. Error t value Pr(>|t|)
Vm 1.603e+02  6.480e+00  24.734 1.38e-09 ***
K  4.771e-02  7.782e-03   6.131 0.000173 ***
```

...and nonlinear modeling...



```
> x <- seq(0.1, 10, by=0.1)
> y <- 10 + 2.35 * exp(1-(x/2))+sin(x) + rnorm(x)
> plot(x, y, main="Some nonlinear modeling", ylim=c(6, 18))
> curve(10 + 2.35 * exp(1-(x/2))+sin(x), col = 4, add = TRUE)
> model <- nls(y ~ Const + A*exp(1-(B*x))+C*sin(x), start=list(Co$
2216.521 :  5.0 1.0 0.1 0.5
1102.006 :  13.9265950 -0.8977520  0.5863972  0.8313710
105.4968 :  9.9066080 2.1947084 0.7826759 1.2159570
88.34774 :  9.9862156 2.1104621 0.4262773 1.1712694
80.82658 :  9.930032 2.186704 0.510669 1.213262
80.72642 :  9.8951240 2.2044292 0.5127079 1.2204740
80.72642 :  9.8950774 2.2044155 0.5126752 1.2204828
> summary(model)

Formula: y ~ Const + A * exp(1 - (B * x)) + C * sin(x)

Parameters:
      Estimate Std. Error t value Pr(>|t|)
Const  9.89508    0.17692  55.931  < 2e-16 ***
A      2.20442    0.16962  12.996  < 2e-16 ***
B      0.51268    0.07826   6.551 2.83e-09 ***
C      1.22048    0.14463   8.439 3.32e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.917 on 96 degrees of freedom

Number of iterations to convergence: 6
Achieved convergence tolerance: 4.107e-07

> lines(x, predict(model), col = 2)
> legend("bottomright", legend=c("by formula", "by estimation"), $
> text(1.7, 9.5, bquote(italic(R)^2 == .(format(PseudoR2(y, model$
> text(6, 17.5, bquote(data:y==10+2.25*e**(1-frac(x,2))+sin(x)+N($
> text(6, 16.5, bquote(f(x)==const+A*e**(1-B*x)+C*sin(x)), col="r$
```

...and meta-analysis...

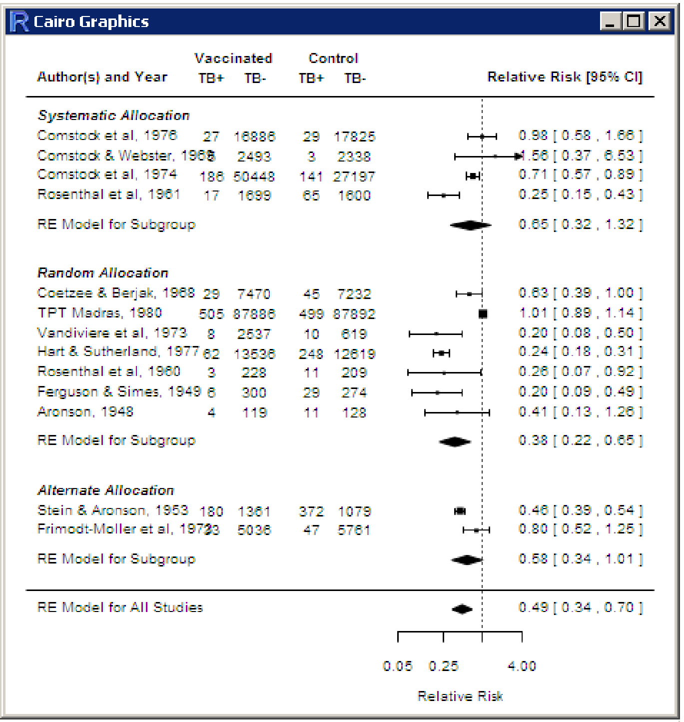


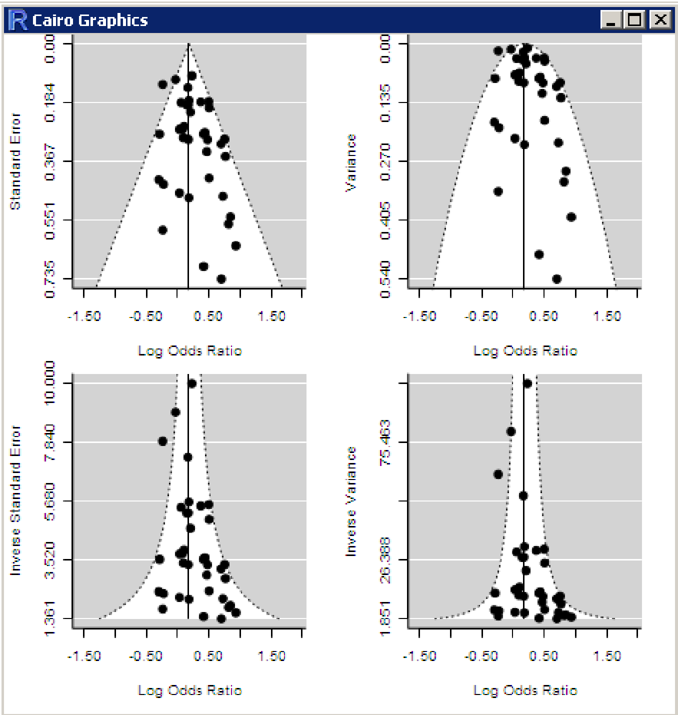
```
> data(cochrane)
> cochrane
          name ev.trt n.trt ev.ctrl n.ctrl
1     Auckland     36   532      60    538
2        Block      1    69       5     61
3        Doran      4    81      11     63
4        Gamsu     14   131      20    137
5     Morrison      3    67       7     59
6 Papageorgiou      1    71       7     75
7      Tauesch      8    56      10     71
> steroid <- meta.MH(n.trt, n.ctrl, ev.trt, ev.ctrl, na$
> summary(steroid)
Fixed effects ( Mantel-Haenszel ) meta-analysis
Call: meta.MH(ntrt = n.trt, nctrl = n.ctrl, ptrt = ev.t$
    names = name, data = cochrane)
------------------------------------
               OR (lower  95% upper)
Auckland     0.58    0.38       0.89
Block        0.16    0.02       1.45
Doran        0.25    0.07       0.81
Gamsu        0.70    0.34       1.45
Morrison     0.35    0.09       1.41
Papageorgiou 0.14    0.02       1.16
Tauesch      1.02    0.37       2.77
------------------------------------
Mantel-Haenszel OR =0.53 95% CI ( 0.39,0.73 )
Test for heterogeneity: X^2( 6 ) = 6.9 ( p-value 0.3303$
>
> tabletext<-cbind(c("","Study",steroid$names,NA,"Summa$
> m<- c(NA,NA,steroid$logOR,NA,steroid$logMH)
> l<- m-c(NA,NA,steroid$selogOR,NA,steroid$selogMH)*2
> u<- m+c(NA,NA,steroid$selogOR,NA,steroid$selogMH)*2
> forestplot(tabletext,m,l,u,zero=0,is.summary=c(TRUE,T$
> title("Example of meta-analysis")
>
```

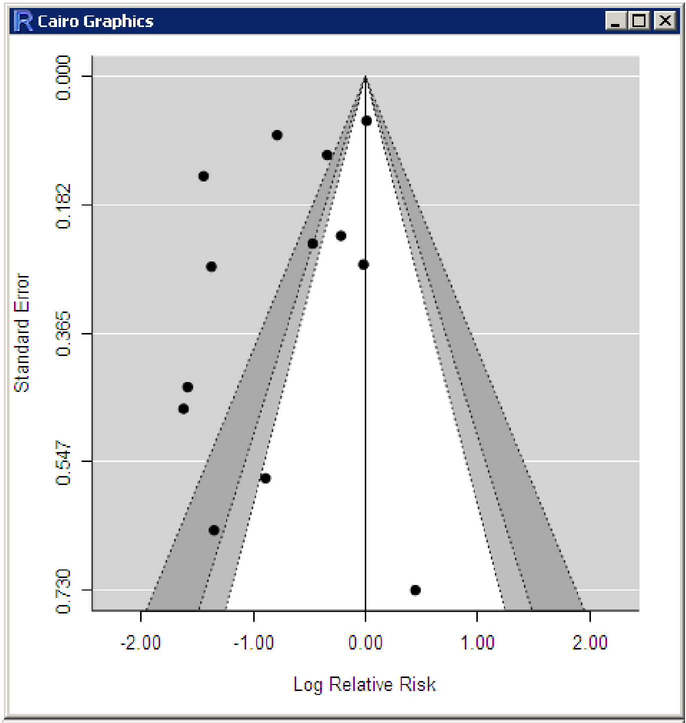
Example of meta-analysis

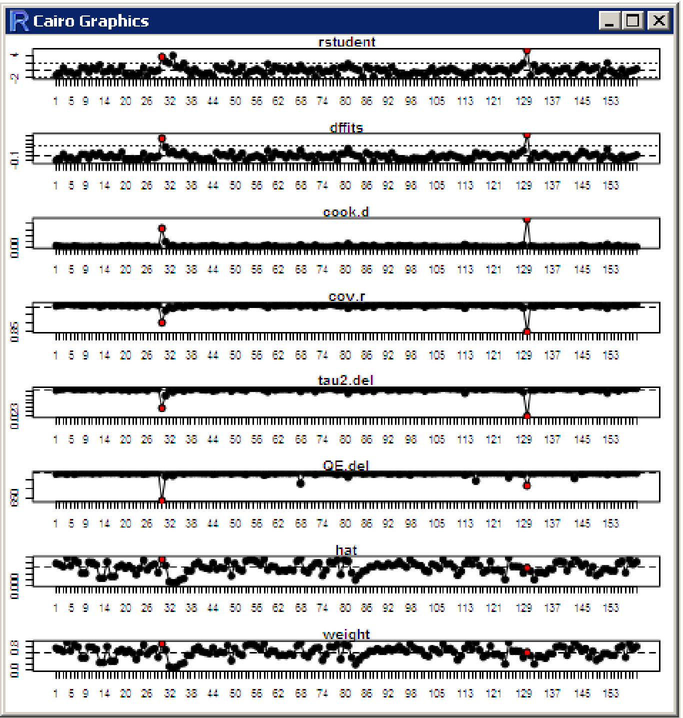
Study	**Deaths (steroid)**	**Deaths (placebo)**	**OR**
Auckland	36	60	0.58
Block	1	5	0.16
Doran	4	11	0.25
Gamsu	14	20	0.70
Morrison	3	7	0.35
Papageorgiou	1	7	0.14
Tauesch	8	10	1.02
Summary			**0.53**

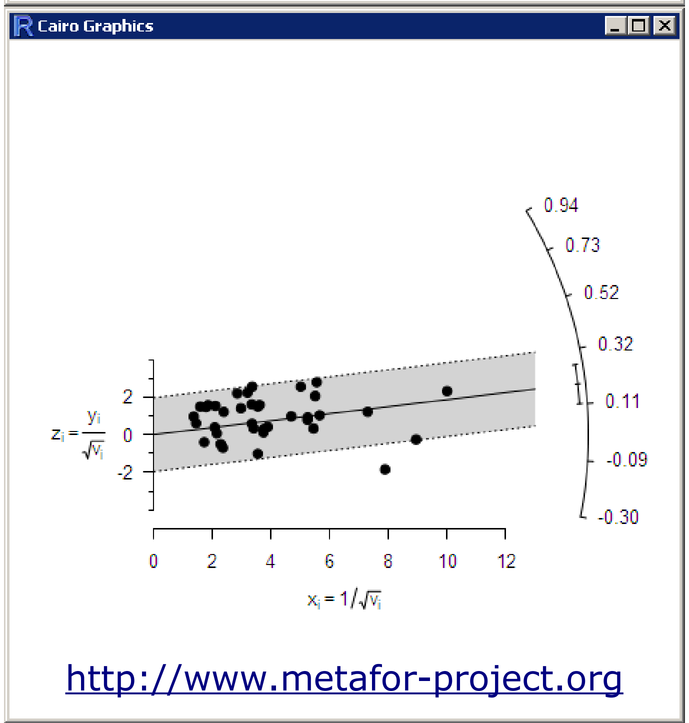
…a bit more detailed output…



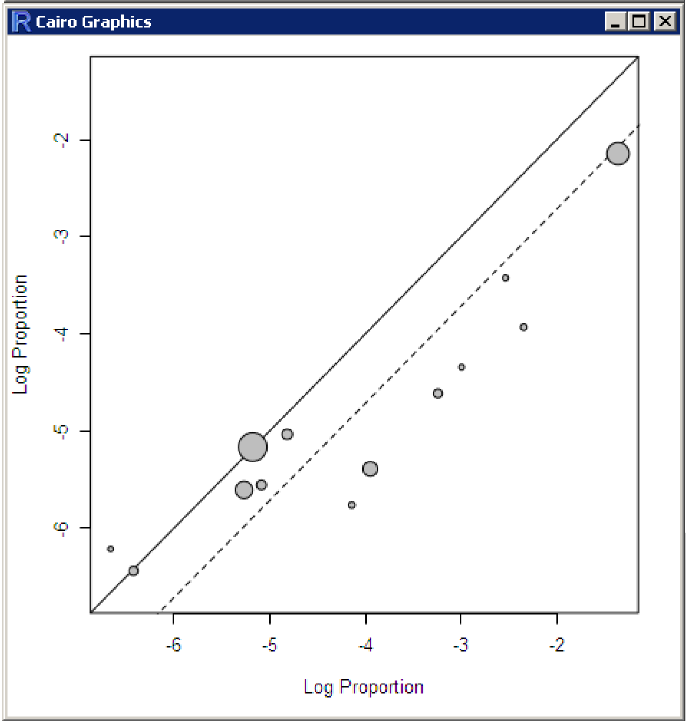




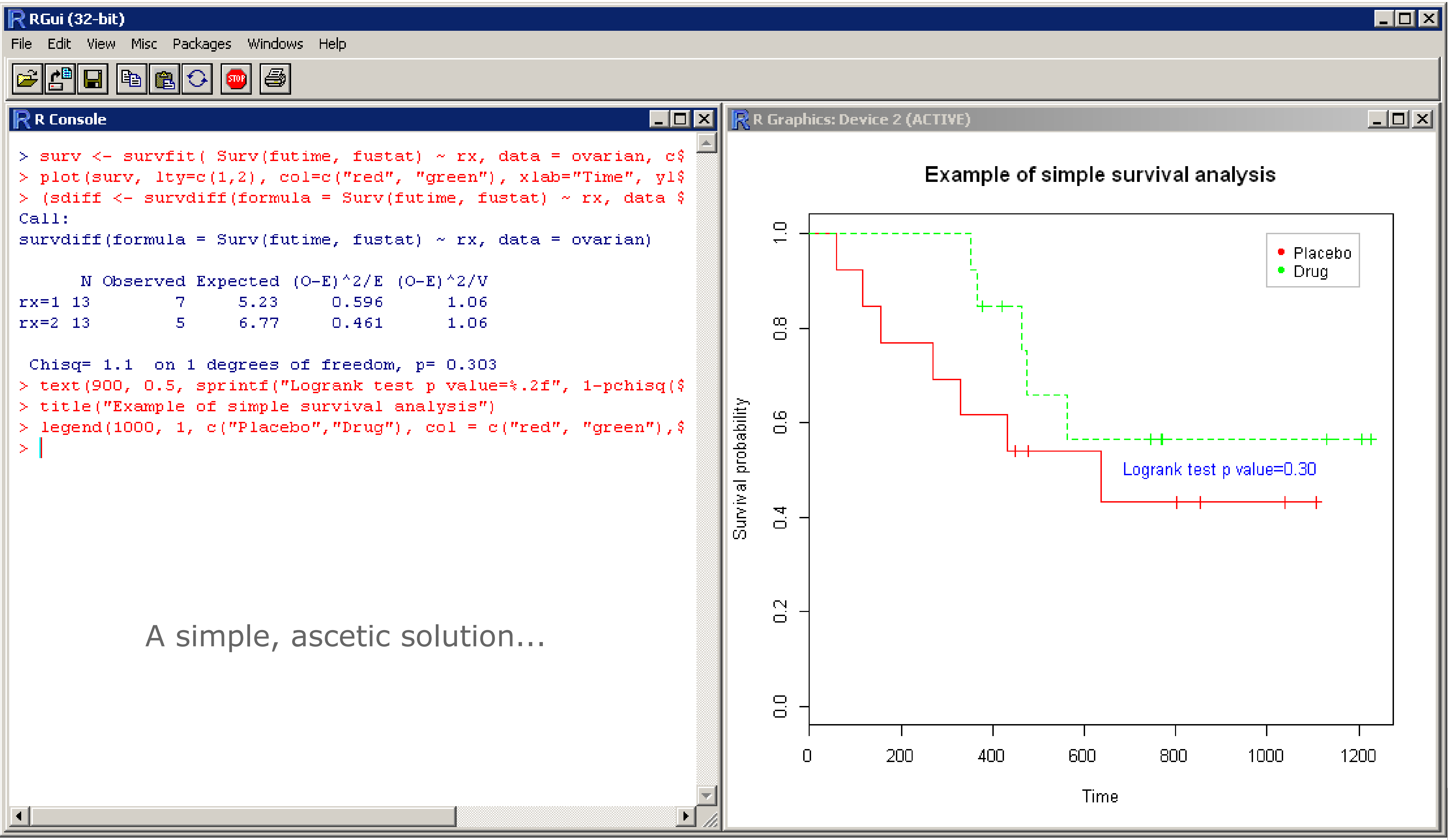




http://www.metafor-project.org



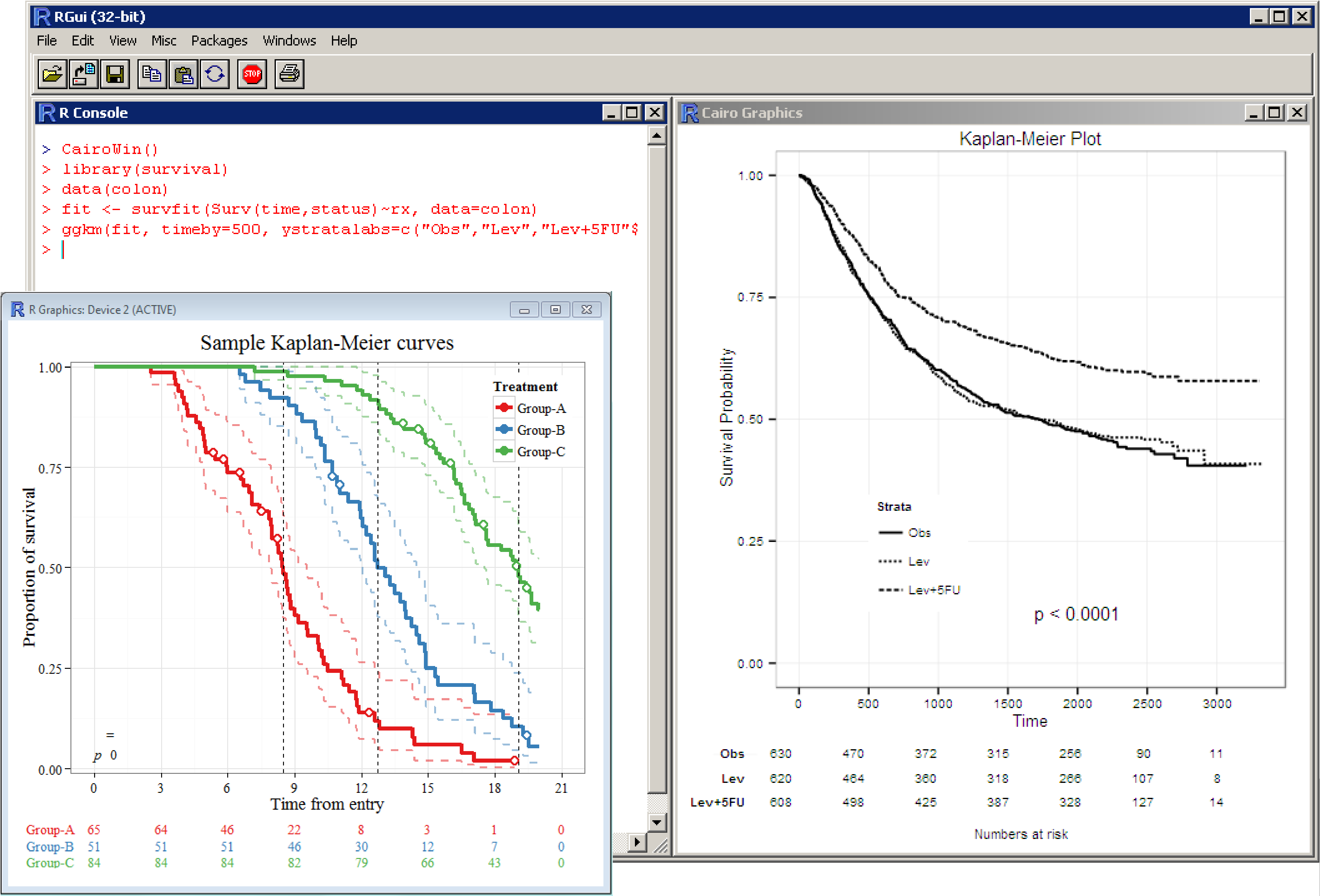
...and survival analysis...



```
> surv <- survfit( Surv(futime, fustat) ~ rx, data = ovarian, c$
> plot(surv, lty=c(1,2), col=c("red", "green"), xlab="Time", yl$
> (sdiff <- survdiff(formula = Surv(futime, fustat) ~ rx, data $
Call:
survdiff(formula = Surv(futime, fustat) ~ rx, data = ovarian)

      N Observed Expected (O-E)^2/E (O-E)^2/V
rx=1 13        7     5.23     0.596      1.06
rx=2 13        5     6.77     0.461      1.06

 Chisq= 1.1  on 1 degrees of freedom, p= 0.303
> text(900, 0.5, sprintf("Logrank test p value=%.2f", 1-pchisq($
> title("Example of simple survival analysis")
> legend(1000, 1, c("Placebo","Drug"), col = c("red", "green"),$
> 
```

A simple, ascetic solution...

...and survival analysis...

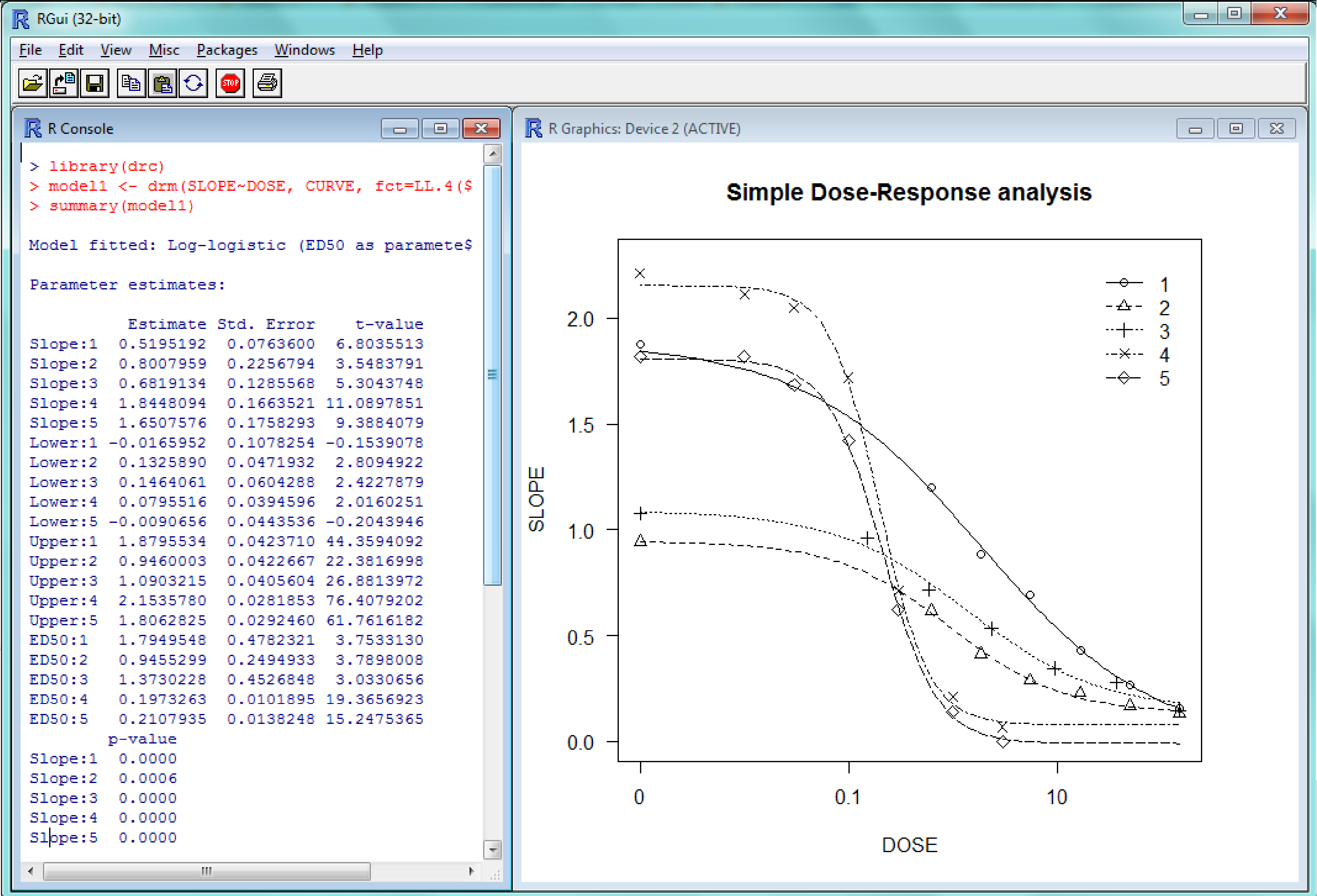


```
> CairoWin()
> library(survival)
> data(colon)
> fit <- survfit(Surv(time,status)~rx, data=colon)
> ggkm(fit, timeby=500, ystratalabs=c("Obs","Lev","Lev+5FU"$
> |
```

	0	3	6	9	12	15	18	21
Group-A	65	64	46	22	8	3	1	0
Group-B	51	51	51	46	30	12	7	0
Group-C	84	84	84	82	79	66	43	0

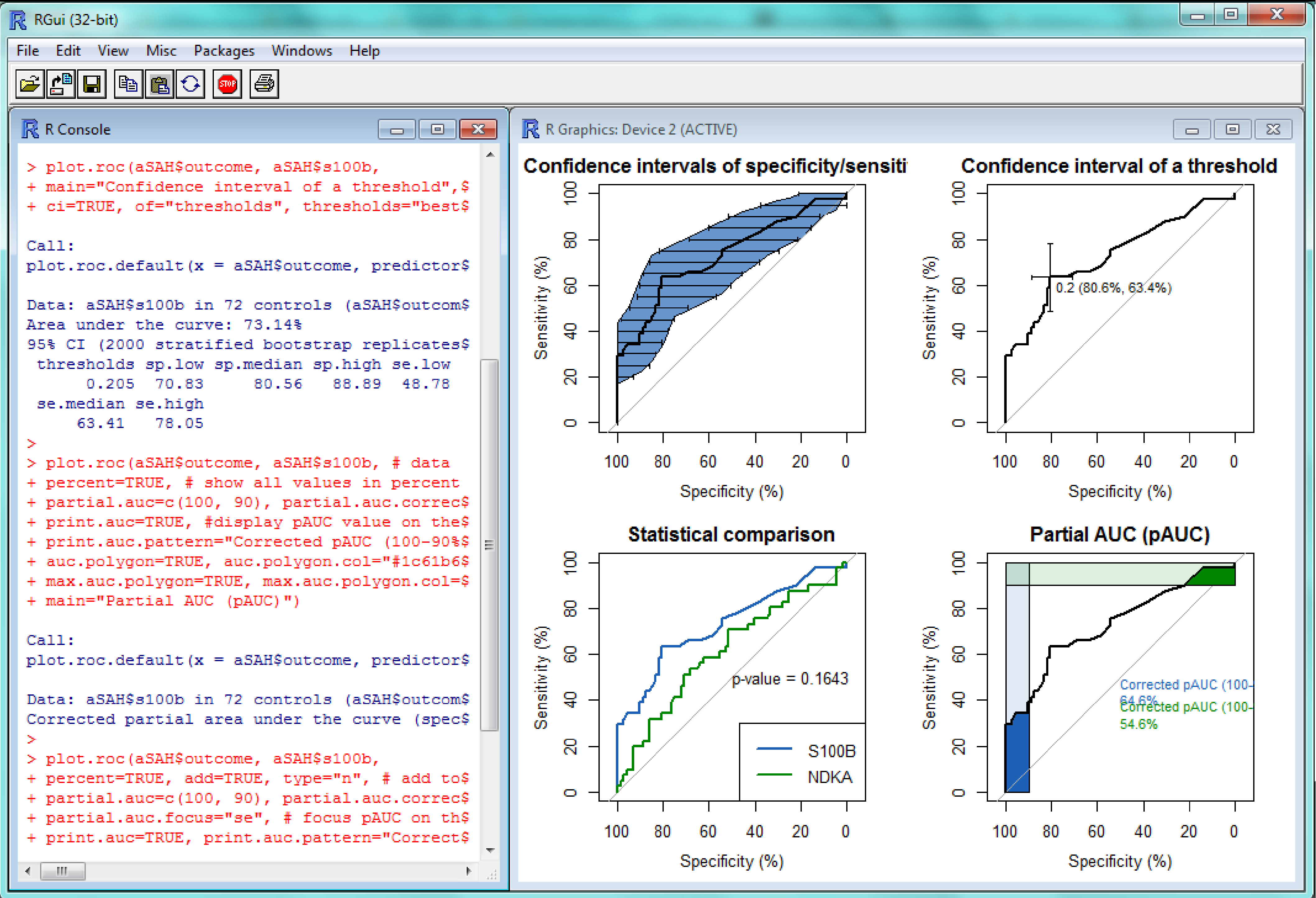
	0	500	1000	1500	2000	2500	3000
Obs	630	470	372	315	256	90	11
Lev	620	464	360	318	266	107	8
Lev+5FU	608	498	425	387	328	127	14

...and dose-response analysis...



```
> library(drc)
> model1 <- drm(SLOPE~DOSE, CURVE, fct=LL.4($
> summary(model1)

Model fitted: Log-logistic (ED50 as paramete$

Parameter estimates:

           Estimate Std. Error    t-value
Slope:1   0.5195192  0.0763600  6.8035513
Slope:2   0.8007959  0.2256794  3.5483791
Slope:3   0.6819134  0.1285568  5.3043748
Slope:4   1.8448094  0.1663521 11.0897851
Slope:5   1.6507576  0.1758293  9.3884079
Lower:1  -0.0165952  0.1078254 -0.1539078
Lower:2   0.1325890  0.0471932  2.8094922
Lower:3   0.1464061  0.0604288  2.4227879
Lower:4   0.0795516  0.0394596  2.0160251
Lower:5  -0.0090656  0.0443536 -0.2043946
Upper:1   1.8795534  0.0423710 44.3594092
Upper:2   0.9460003  0.0422667 22.3816998
Upper:3   1.0903215  0.0405604 26.8813972
Upper:4   2.1535780  0.0281853 76.4079202
Upper:5   1.8062825  0.0292460 61.7616182
ED50:1    1.7949548  0.4782321  3.7533130
ED50:2    0.9455299  0.2494933  3.7898008
ED50:3    1.3730228  0.4526848  3.0330656
ED50:4    0.1973263  0.0101895 19.3656923
ED50:5    0.2107935  0.0138248 15.2475365
          p-value
Slope:1   0.0000
Slope:2   0.0006
Slope:3   0.0000
Slope:4   0.0000
Slope:5   0.0000
```

...and ROC curves...



```
> plot.roc(aSAH$outcome, aSAH$s100b,
+ main="Confidence interval of a threshold",$
+ ci=TRUE, of="thresholds", thresholds="best$

Call:
plot.roc.default(x = aSAH$outcome, predictor$

Data: aSAH$s100b in 72 controls (aSAH$outcom$
Area under the curve: 73.14%
95% CI (2000 stratified bootstrap replicates$
 thresholds sp.low sp.median sp.high se.low
      0.205  70.83     80.56   88.89  48.78
 se.median se.high
     63.41   78.05
>
> plot.roc(aSAH$outcome, aSAH$s100b, # data
+ percent=TRUE, # show all values in percent
+ partial.auc=c(100, 90), partial.auc.correc$
+ print.auc=TRUE, #display pAUC value on the$
+ print.auc.pattern="Corrected pAUC (100-90%$
+ auc.polygon=TRUE, auc.polygon.col="#1c61b6$
+ max.auc.polygon=TRUE, max.auc.polygon.col=$
+ main="Partial AUC (pAUC)")

Call:
plot.roc.default(x = aSAH$outcome, predictor$

Data: aSAH$s100b in 72 controls (aSAH$outcom$
Corrected partial area under the curve (spec$
>
> plot.roc(aSAH$outcome, aSAH$s100b,
+ percent=TRUE, add=TRUE, type="n", # add to$
+ partial.auc=c(100, 90), partial.auc.correc$
+ partial.auc.focus="se", # focus pAUC on th$
+ print.auc=TRUE, print.auc.pattern="Correct$
```

13 reasons why you will love GNU R

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is supported by the world of science

IV R is supported by the community

V R is supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XIII FDA: R is OK for drug trials! !

the pRice

GNU R is a **free software**.

One can legally use it, even commercially, at no cost.

Some companies provide their own, both free and commercial, customized versions of R along with **support and integration services**. Well known providers are: Revolution (since 2015 part of Microsoft), RStudio and Oracle.

REVOLUTION ANALYTICS

http://buy.revolutionanalytics.com/

http://blogs.microsoft.com/blog/2015/01/23/...

RStudio

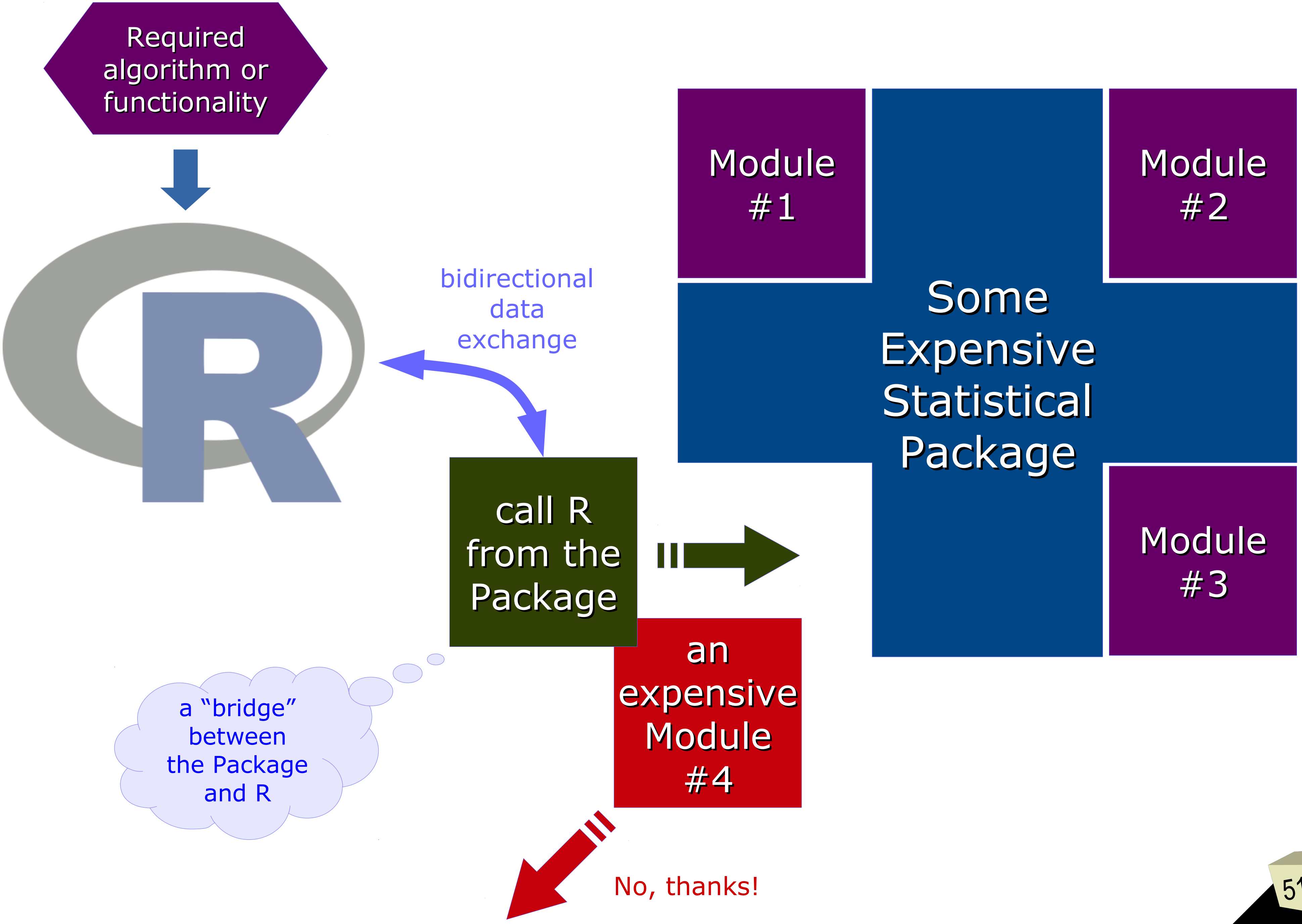
http://www.rstudio.com...

R Technologies from Oracle
Bringing the Power of R to the Enterprise

http://www.oracle.com/technetwork/...

http://www.rstudio.com/about/trademark/

save your money with R!



Licenses

The following licenses are in use for R and associated software:

- GNU **A**ffero **GPL** v.3
- **A**rtistic **L**icense" v.2.0
- **BSD** 2-clause
- **BSD** 3-clause
- GNU **GPL** v.2
- GNU **GPL** v.3
- GNU **L**ibrary **GPL** v.2
- GNU **L**esser **GPL** v.2.1
- GNU **L**esser **GPL** v.3
- **M**assachusetts **I**nstitute of **T**echnology (X11)

R as a package is licensed under **GPLv2**

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is supported by the world of science

IV R is supported by the community

V R is supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

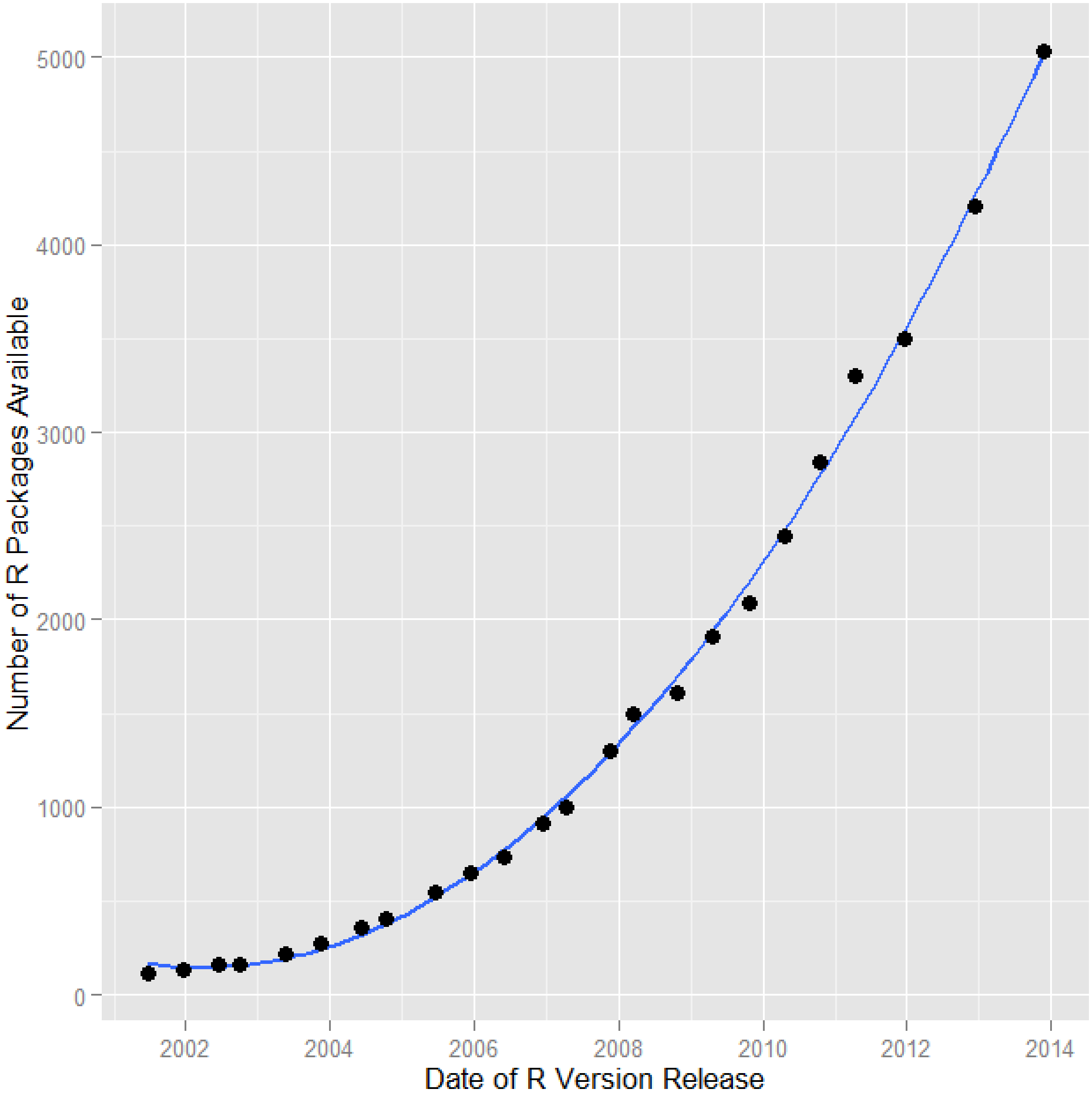
the Comprehensive R Archive Network

Sit down and hold tight! CRAN holds more than **6600** nrow(available.packages()) packages.

Just describe your problem or ask me for a statistical test or procedure and I will give you the right package(s): http://cran.r-project.org/web/views

- Linear models of ANY kind. (M)AN(C)OVA, regression (linear, logistic, hierarchical, etc.)
- Post-factum analysis and planned comparisons
- Nonlinear models (NLS, Generalized Additive Models with a rich set of smoothers), trees
- Robust methods: Regularized methods, M-estimators
- Models with fixed and random effects (mixed models). Variance components
- Monte Carlo methods (permutational, bootstrap). Exact methods.
- Survival • PK/PD • Superior / Non inferior. / Equiv. / Bioequiv. trials • Meta-analysis
- Design of experiments – including those applicable in clinical research
- Structural equations. Time series. Forecasting.
- Methods for analyzing multidimensional data: NN, SVM, LDA/QDA, PCA, FA, CC, MS, KLT, CA, MART, POLYMARS, PINDIS, PPR, ACE, AVAS, K-means, KN, and lots of more!
- Trees (CART, CHAID). Random forests. Aggregating (boosting, bagging)
- Reproducible research. Graphical User Interfaces. Widely understood interoperability.

Growth in the number of packages



A list of thematic sections covered by the CRAN repository:

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

Clinical Research

What kind of analyses common in clinical research can be done in R?

- Descriptive statistics, summaries (demographic, recruitment)
- Advanced, linear and nonlinear modeling (models of any type)
- Comparisons of treatments
- PK / PD analysis
- Analysis of bio-equivalence, non-inferiority, superiority
- Design of experiments
- Time-to-event analysis (survival analysis)
- Analysis of data from longitudinal trials
- Sample size determination and power analysis
- Meta-analysis
- Bayesian analyzes
- Analysis of Adverse Events
- Analysis of DNA micro-arrays
- ROC curves

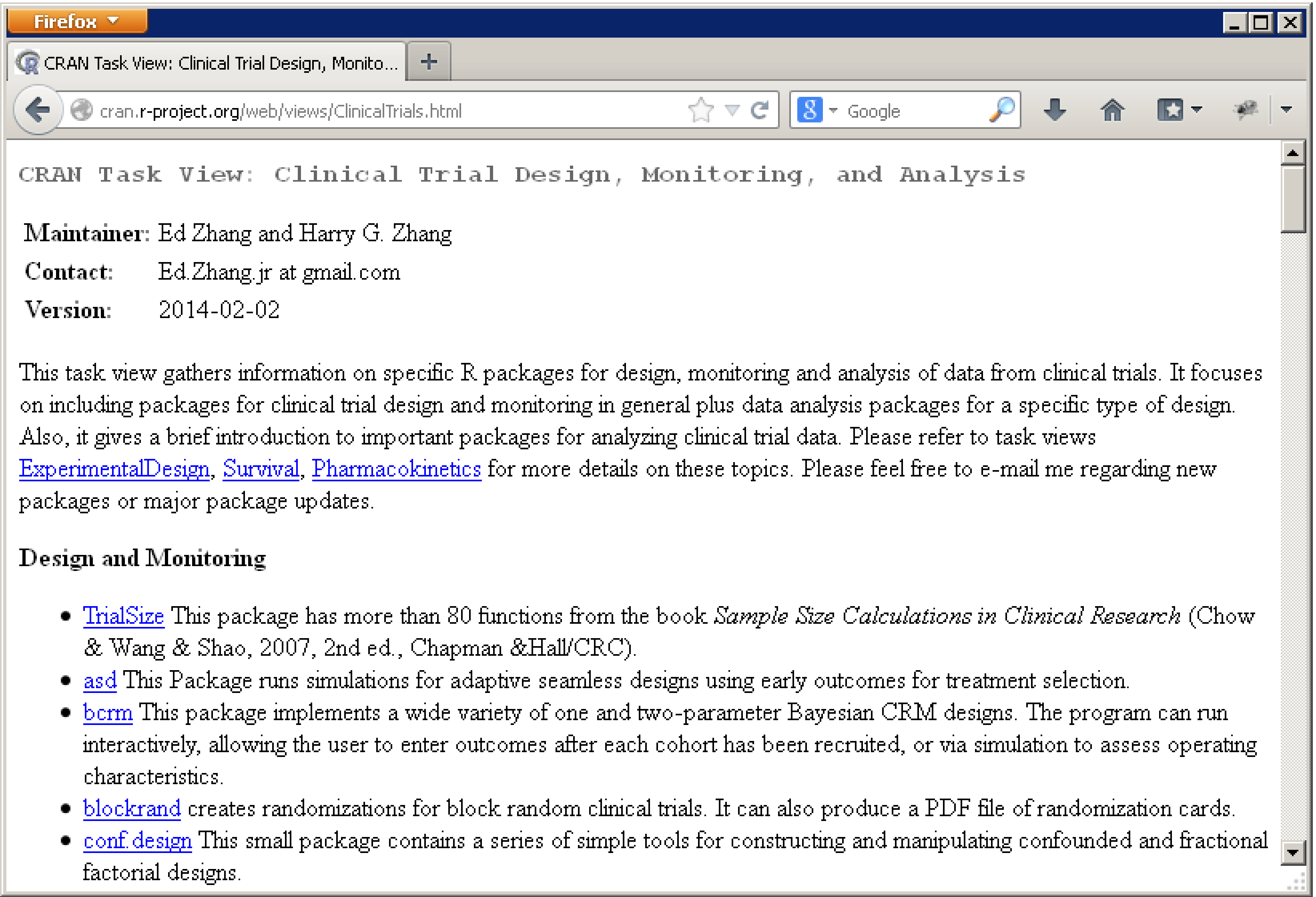
Clinical Research

http://cran.r-project.org/web/views/ClinicalTrials.html

http://cran.r-project.org/web/views/Survival.html

http://cran.r-project.org/web/views/ExperimentalDesign.html

http://cran.r-project.org/web/views/Pharmacokinetics.html



CRAN Task View: Clinical Trial Design, Monitoring, and Analysis

Maintainer: Ed Zhang and Harry G. Zhang
Contact: Ed.Zhang.jr at gmail.com
Version: 2014-02-02

This task view gathers information on specific R packages for design, monitoring and analysis of data from clinical trials. It focuses on including packages for clinical trial design and monitoring in general plus data analysis packages for a specific type of design. Also, it gives a brief introduction to important packages for analyzing clinical trial data. Please refer to task views ExperimentalDesign, Survival, Pharmacokinetics for more details on these topics. Please feel free to e-mail me regarding new packages or major package updates.

Design and Monitoring

- TrialSize This package has more than 80 functions from the book *Sample Size Calculations in Clinical Research* (Chow & Wang & Shao, 2007, 2nd ed., Chapman &Hall/CRC).
- asd This Package runs simulations for adaptive seamless designs using early outcomes for treatment selection.
- bcrm This package implements a wide variety of one and two-parameter Bayesian CRM designs. The program can run interactively, allowing the user to enter outcomes after each cohort has been recruited, or via simulation to assess operating characteristics.
- blockrand creates randomizations for block random clinical trials. It can also produce a PDF file of randomization cards.
- conf.design This small package contains a series of simple tools for constructing and manipulating confounded and fractional factorial designs.

CRAN packages:

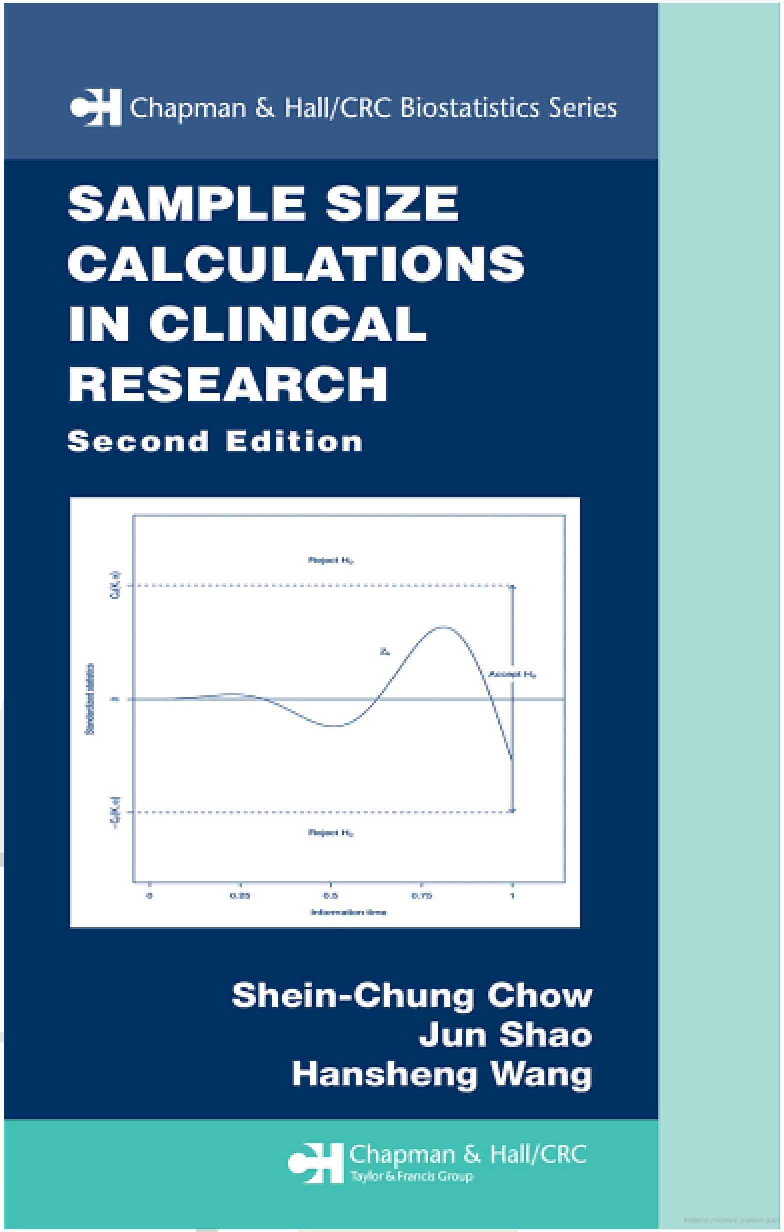
- adaptTest
- AGSDest
- asd
- asypow
- bcrm
- bifactorial
- blockrand
- clinfun
- clinsig
- coin
- conf.design
- copas
- CRM
- CRTSize
- dfcrm
- DoseFinding
- epibasix
- epicalc
- experiment
- FrF2
- GroupSeq
- gsDesign
- HH
- Hmisc
- ldbounds
- longpower
- MChtest
- MCPMod
- meta
- metafor
- metaLik
- multcomp
- nppbib
- PIPS
- PowerTOST
- pwr
- PwrGSD
- qtlDesign
- rmeta
- samplesize
- seqmon
- speff2trial
- ssanv
- survival
- tdm
- TEQR
- TrialSize

Clinical Research

Package: TrialSize

"More than 80 functions in this package are widely used to calculate sample size in clinical trial research studies.

This package covers the functions in Chapter 3,4,6,7,9,10,11,12,14,15 of the reference book"



Chapman & Hall/CRC Biostatistics Series

SAMPLE SIZE CALCULATIONS IN CLINICAL RESEARCH

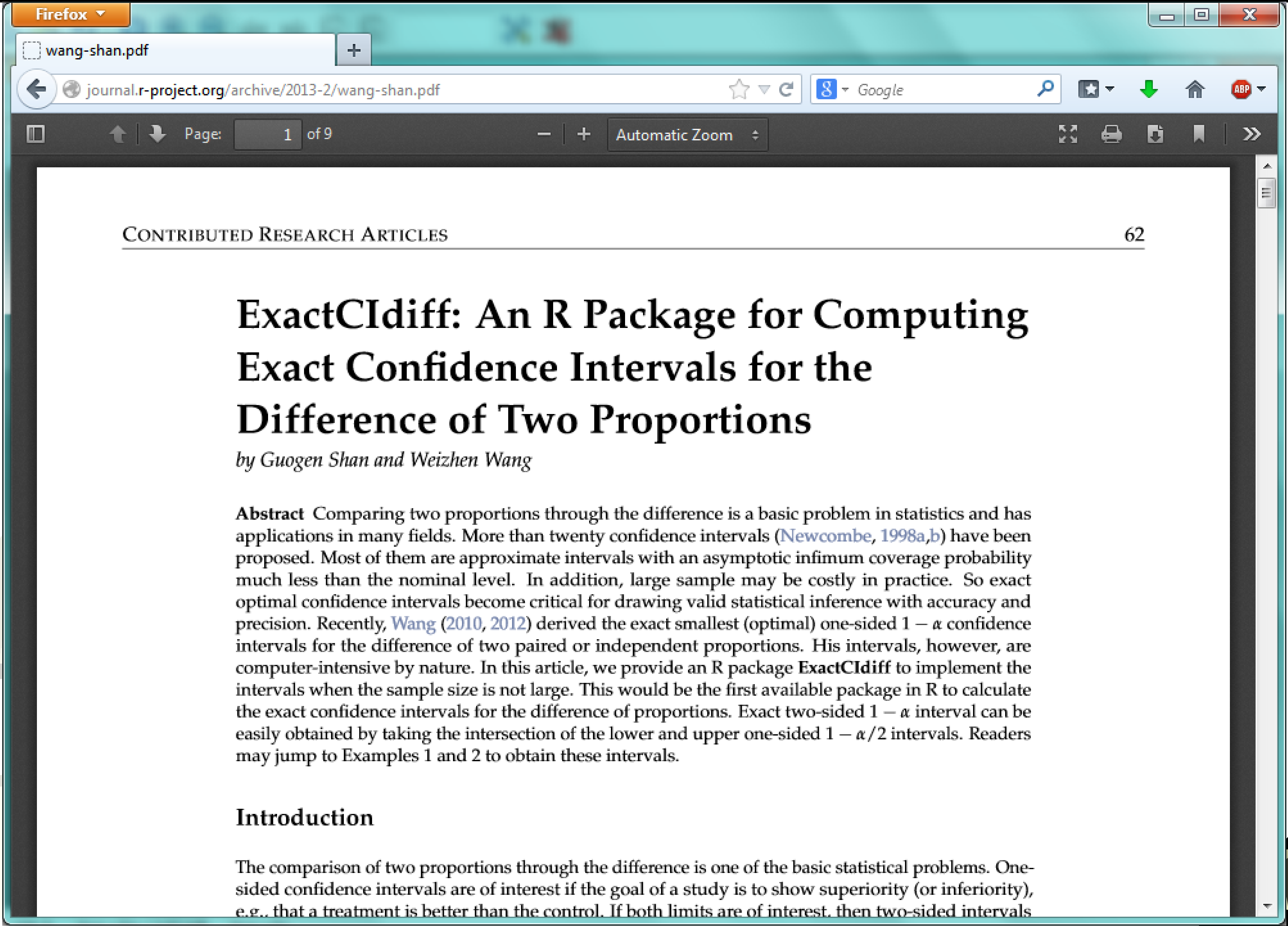
Second Edition

Shein-Chung Chow
Jun Shao
Hansheng Wang

Chapman & Hall/CRC
Taylor & Francis Group

AB.withDescalation
AB.withoutDescalation
ABE
ANOVA.Repeat.Measure
Carry.Over
Cochran.Armitage/Trend
Cox.Equality
Cox.Equivalence
Cox.NIS
CrossOver.ISV.Equality
CrossOver.ISV.Equivalence
CrossOver.ISV.NIS
Dose.MinE.ffect
Dose.Response.binary
Dose.Response.Linear
Dose.Response.time.to.event
gof.Pearson
gof.Pearson.twoway
IBE
InterSV.Equality
InterSV.NIS
ISCV.Equality
ISCV.Equivalence
ISCV.NIS
ISV.Equality
ISV.Equivalence
ISV.NIS
McNemar.Test
MeanWilliamsDesign.Equality
MeanWilliamsDesign.Equivalence
MeanWilliamsDesign.NIS
Multiple.Testing
Nonpara.Independ
Nonpara.One.Sample
Nonpara.Two.Sample
OneSampleMean.Equality
OneSampleMean.Equivalence
OneSampleMean.NIS
OneSampleProportion.Equality
OneSampleProportion.Equivalence
OneSampleProportion.NIS
OneSide.fixEffect
OneSide.varyEffect
OneWayANOVA.pairwise
OneWayANOVA.PairwiseComparison
PBE
Propensity.Score.nostrata
Propensity.Score.strata
QOL
QT.crossover
QT.parallel
QT.PK.crossover
QT.PK.parallel
RelativeRisk.Equality
RelativeRisk.Equivalence
RelativeRisk.NIS
RelativeRiskCrossOver.Equality
RelativeRiskCrossOver.Equivalence
RelativeRiskCrossOver.NIS
Sensitivity.Index
Stuart.Maxwell.Test
TwoSampleCrossOver.Equality
TwoSampleCrossOver.Equivalence
TwoSampleCrossOver.NIS
TwoSampleMean.Equality
TwoSampleMean.Equivalence
TwoSampleMean.NIS
TwoSampleProportion.Equality
TwoSampleProportion.Equivalence
TwoSampleProportion.NIS
TwoSampleSeqCrossOver.Equality
TwoSampleSeqCrossOver.Equivalence
TwoSampleSeqCrossOver.NIS
TwoSampleSurvival.Conditional
TwoSampleSurvival.Equality
TwoSampleSurvival.Equivalence
TwoSampleSurvival.NIS
TwoSide.fixEffect
TwoSide.varyEffect
Vaccine.CEM
Vaccine.ELDI
Vaccine.RDI
Vitro.BE
WilliamsDesign.Equality
WilliamsDesign.Equivalence
WilliamsDesign.NIS

Clinical Research



CONTRIBUTED RESEARCH ARTICLES 62

ExactCIdiff: An R Package for Computing Exact Confidence Intervals for the Difference of Two Proportions

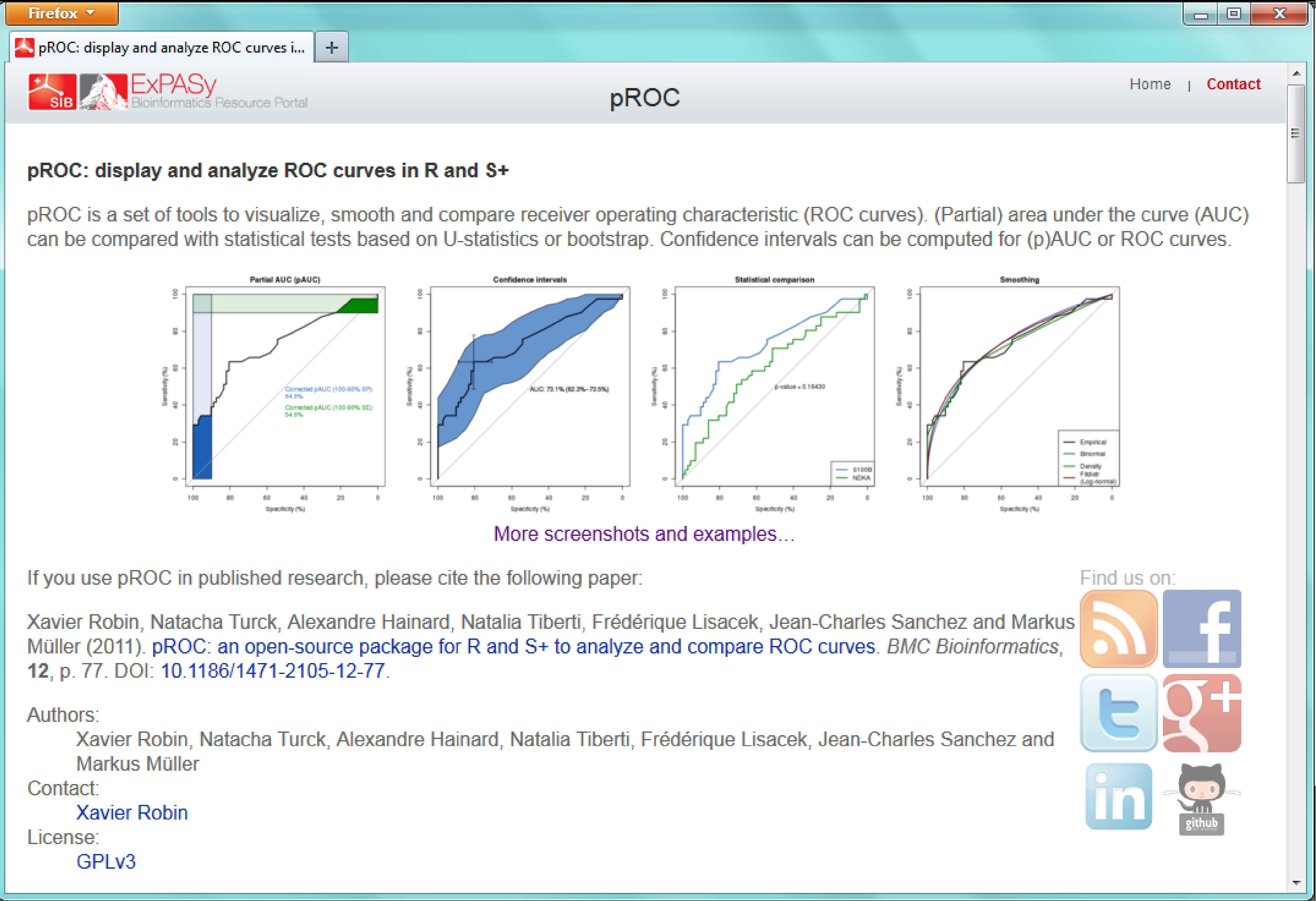
by Guogen Shan and Weizhen Wang

Abstract Comparing two proportions through the difference is a basic problem in statistics and has applications in many fields. More than twenty confidence intervals (Newcombe, 1998a,b) have been proposed. Most of them are approximate intervals with an asymptotic infimum coverage probability much less than the nominal level. In addition, large sample may be costly in practice. So exact optimal confidence intervals become critical for drawing valid statistical inference with accuracy and precision. Recently, Wang (2010, 2012) derived the exact smallest (optimal) one-sided $1-\alpha$ confidence intervals for the difference of two paired or independent proportions. His intervals, however, are computer-intensive by nature. In this article, we provide an R package **ExactCIdiff** to implement the intervals when the sample size is not large. This would be the first available package in R to calculate the exact confidence intervals for the difference of proportions. Exact two-sided $1-\alpha$ interval can be easily obtained by taking the intersection of the lower and upper one-sided $1-\alpha/2$ intervals. Readers may jump to Examples 1 and 2 to obtain these intervals.

Introduction

The comparison of two proportions through the difference is one of the basic statistical problems. One-sided confidence intervals are of interest if the goal of a study is to show superiority (or inferiority), e.g., that a treatment is better than the control. If both limits are of interest, then two-sided intervals

ROC I



pROC

pROC: display and analyze ROC curves in R and S+

pROC is a set of tools to visualize, smooth and compare receiver operating characteristic (ROC curves). (Partial) area under the curve (AUC) can be compared with statistical tests based on U-statistics or bootstrap. Confidence intervals can be computed for (p)AUC or ROC curves.

More screenshots and examples...

If you use pROC in published research, please cite the following paper:

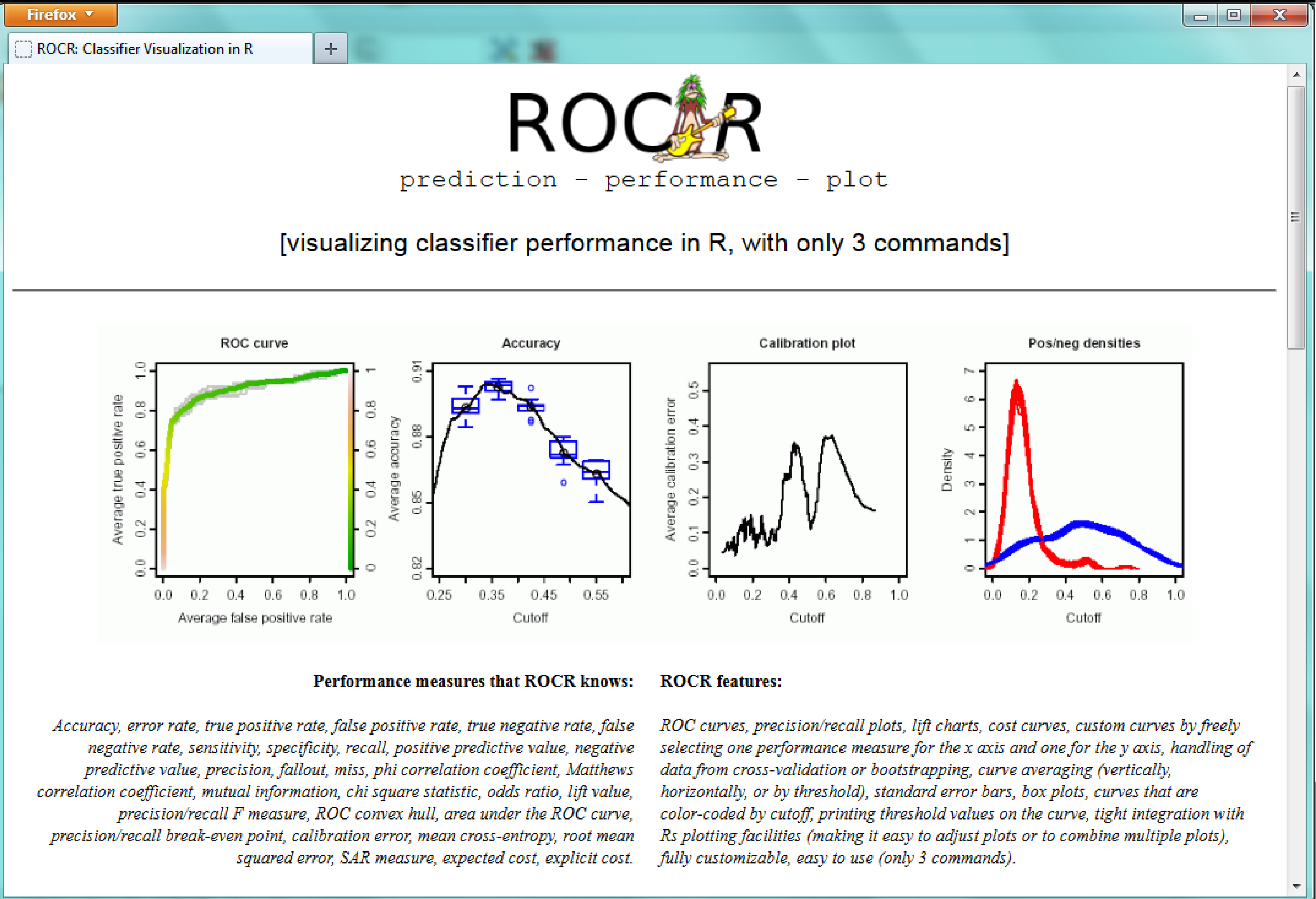
Xavier Robin, Natacha Turck, Alexandre Hainard, Natalia Tiberti, Frédérique Lisacek, Jean-Charles Sanchez and Markus Müller (2011). pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics*, **12**, p. 77. DOI: 10.1186/1471-2105-12-77.

Authors:
Xavier Robin, Natacha Turck, Alexandre Hainard, Natalia Tiberti, Frédérique Lisacek, Jean-Charles Sanchez and Markus Müller

Contact:
Xavier Robin

License:
GPLv3

ROC II



ROCR

prediction - performance - plot

[visualizing classifier performance in R, with only 3 commands]

Performance measures that ROCR knows:

Accuracy, error rate, true positive rate, false positive rate, true negative rate, false negative rate, sensitivity, specificity, recall, positive predictive value, negative predictive value, precision, fallout, miss, phi correlation coefficient, Matthews correlation coefficient, mutual information, chi square statistic, odds ratio, lift value, precision/recall F measure, ROC convex hull, area under the ROC curve, precision/recall break-even point, calibration error, mean cross-entropy, root mean squared error, SAR measure, expected cost, explicit cost.

ROCR features:

ROC curves, precision/recall plots, lift charts, cost curves, custom curves by freely selecting one performance measure for the x axis and one for the y axis, handling of data from cross-validation or bootstrapping, curve averaging (vertically, horizontally, or by threshold), standard error bars, box plots, curves that are color-coded by cutoff, printing threshold values on the curve, tight integration with Rs plotting facilities (making it easy to adjust plots or to combine multiple plots), fully customizable, easy to use (only 3 commands).

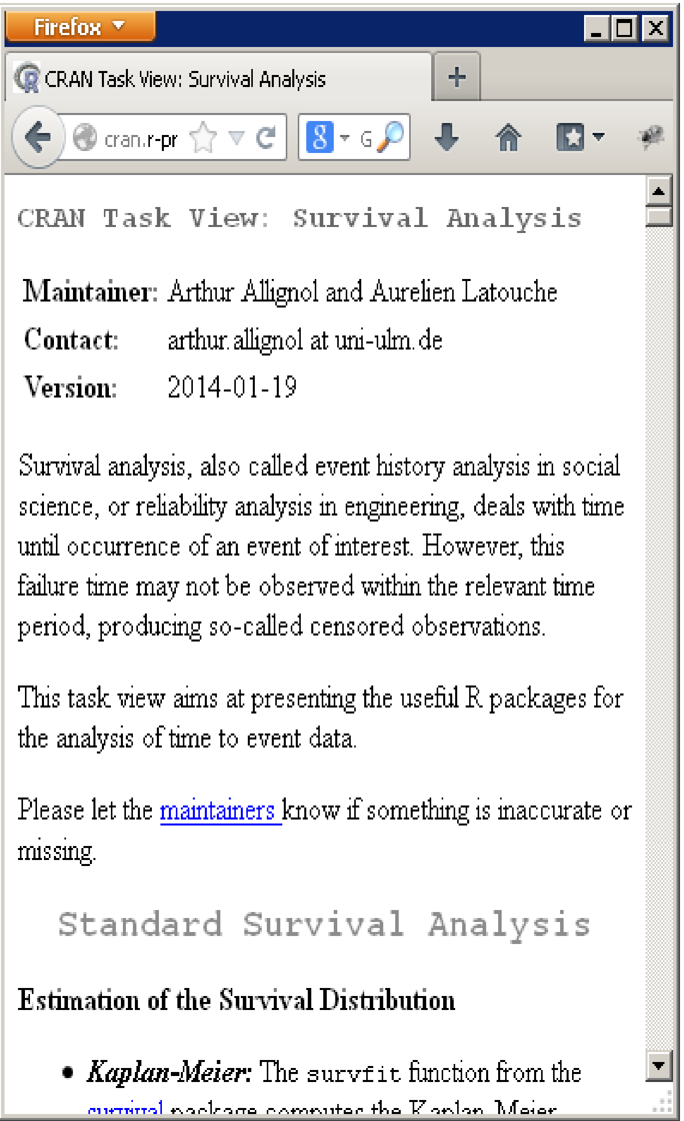
Clinical Research

http://cran.r-project.org/web/views/ClinicalTrials.html

http://cran.r-project.org/web/views/Survival.html

http://cran.r-project.org/web/views/ExperimentalDesign.html

http://cran.r-project.org/web/views/Pharmacokinetics.html



CRAN Task View: Survival Analysis

Maintainer: Arthur Allignol and Aurelien Latouche
Contact: arthur.allignol at uni-ulm.de
Version: 2014-01-19

Survival analysis, also called event history analysis in social science, or reliability analysis in engineering, deals with time until occurrence of an event of interest. However, this failure time may not be observed within the relevant time period, producing so-called censored observations.

This task view aims at presenting the useful R packages for the analysis of time to event data.

Please let the maintainers know if something is inaccurate or missing.

Standard Survival Analysis

Estimation of the Survival Distribution

- ***Kaplan-Meier:*** The `survfit` function from the survival package computes the Kaplan-Meier

CRAN packages:

- AdapEnetClass
- AER
- ahaz
- AIM
- asbio
- aster
- aster2
- BaSTA
- bayesSurv
- BayHaz
- BGPhazard
- Biograph
- BMA
- boot
- bpcp
- censReg
- changeLOS
- clinfun
- cmprsk
- coarseDataTools
- coin
- compeir
- complex.surv.dat.sim
- compound.Cox
- concreg
- condGEE
- CoxBoost
- coxme
- coxphf
- coxphw
- CoxRidge
- coxrobust
- CPE
- CPHshape
- CR
- crrSC
- crrstep
- currentSurvival
- DAAG
- dblcens
- DPpackage
- DTDA
- dynpred
- dynsurv
- eha
- emplik
- Epi
- epiR
- etm
- exactRankTests
- fastcox
- FHtest
- fitdistrplus
- flexsurv
- frailtyHL
- frailtypack
- gamlss.cens
- gbm
- gems
- genSurv
- glmpath
- globalboosttest
- glrt
- gof
- gss
- ICE
- intcox
- interval
- ipdmeta
- ipred
- ipw
- jackknifeKME
- JM
- JMbayes
- jmec
- JMLSD
- joineR
- JPSurv
- kaps
- km.ci
- kmc
- kmconfband
- kmi
- KMsurv
- lava.tobit
- lbiassurv
- LearnBayes
- lmec
- locfit
- logconcens
- LogicReg
- LogrankA
- logspline
- lpc
- lss
- MAMSE
- maxstat
- mboost
- MCMCglmm
- MCMCpack
- Mets
- mfp
- mixAK
- mixPHM
- MLEcens
- MRsurv
- msm
- msSurv
- mstate
- muhaz
- multcomp
- mvna
- mvpart
- NADA
- NestedCohort
- NPHMC
- NPMLEcmprsk
- OIsurv
- OrdFacReg
- OutlierDC
- p3state.msm
- pamr
- parfm
- party
- pec
- penalized
- peperr
- PermAlgo
- phmm
- plsRcox
- polspline
- powerSurvEpi
- prodlim
- psbcGroup
- pseudo
- quantreg
- randomForestSRC
- RandomSurvivalForest
- rankhazard
- relsurv
- rhosp
- riskRegression
- risksetROC
- rms
- RobustAFT
- ROCt
- rpart
- rsig
- saws
- SemiCompRisks
- SemiMarkov
- SGL
- simexaft
- simMSM
- simPH
- SMIR
- SmoothHazard
- smoothHR
- smoothSurv
- SMPracticals
- spatstat
- spatsurv
- superpc
- surv2sampleComp
- survAUC
- survBayes
- survC1
- survexp.fr
- survey
- Survgini
- survIDINRI
- survival
- survivalBIV
- survivalROC
- survJamda
- survMisc
- survPresmooth
- survrec
- SurvRegCensCov
- survsim
- survSNP
- SvyNom
- TBSSurvival
- TestSurvRec
- timereg
- timeROC
- tlmec
- TPmsm
- tpr
- TraMineR
- TSHRC
- uniCox
- VGAM
- wtcrsk
- YPmodel

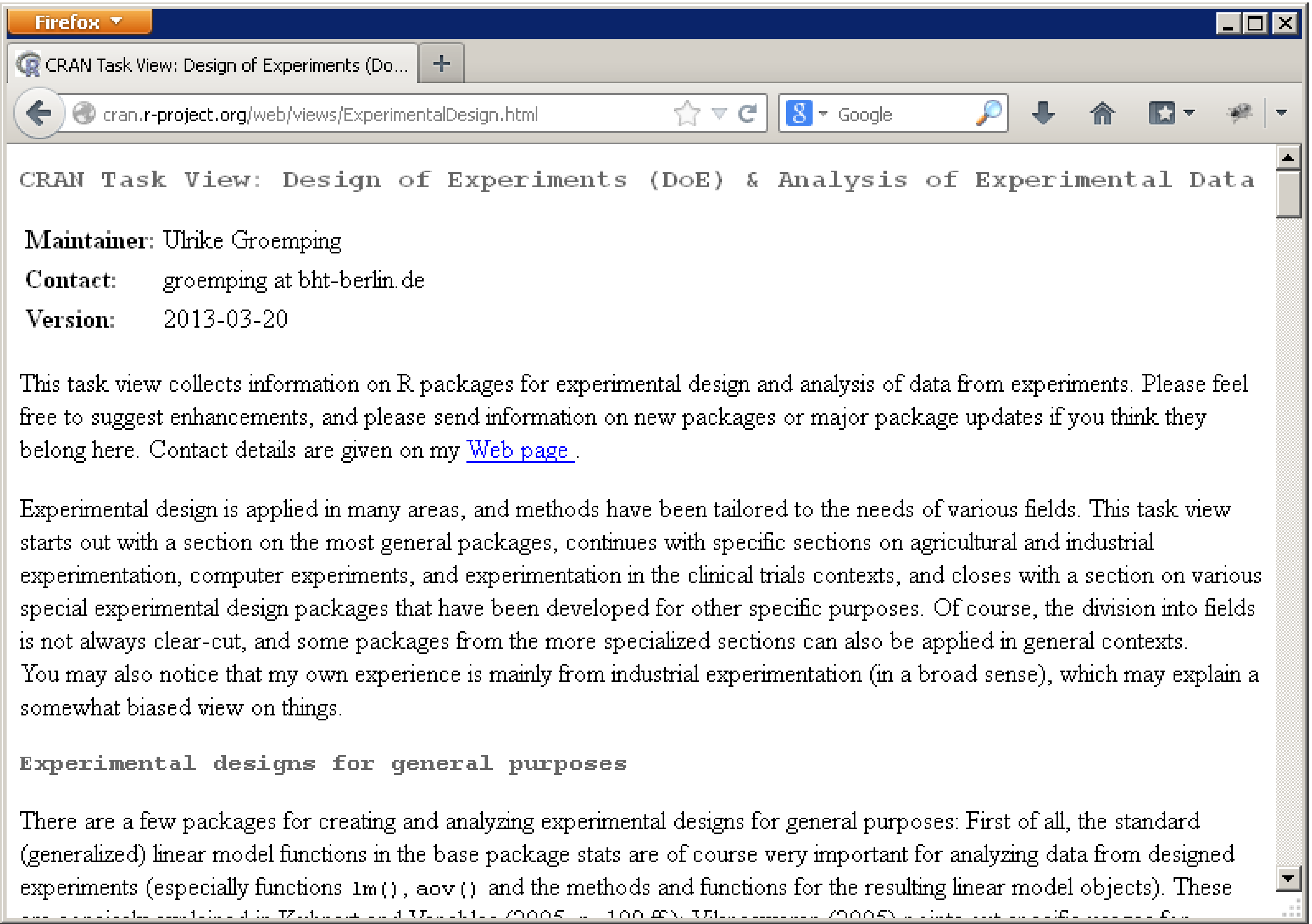
Clinical Research

http://cran.r-project.org/web/views/ClinicalTrials.html

http://cran.r-project.org/web/views/Survival.html

http://cran.r-project.org/web/views/ExperimentalDesign.html

http://cran.r-project.org/web/views/Pharmacokinetics.html



CRAN Task View: Design of Experiments (DoE) & Analysis of Experimental Data

Maintainer: Ulrike Groemping
Contact: groemping at bht-berlin.de
Version: 2013-03-20

This task view collects information on R packages for experimental design and analysis of data from experiments. Please feel free to suggest enhancements, and please send information on new packages or major package updates if you think they belong here. Contact details are given on my Web page .

Experimental design is applied in many areas, and methods have been tailored to the needs of various fields. This task view starts out with a section on the most general packages, continues with specific sections on agricultural and industrial experimentation, computer experiments, and experimentation in the clinical trials contexts, and closes with a section on various special experimental design packages that have been developed for other specific purposes. Of course, the division into fields is not always clear-cut, and some packages from the more specialized sections can also be applied in general contexts. You may also notice that my own experience is mainly from industrial experimentation (in a broad sense), which may explain a somewhat biased view on things.

Experimental designs for general purposes

There are a few packages for creating and analyzing experimental designs for general purposes: First of all, the standard (generalized) linear model functions in the base package stats are of course very important for analyzing data from designed experiments (especially functions `lm()`, `aov()` and the methods and functions for the resulting linear model objects). These

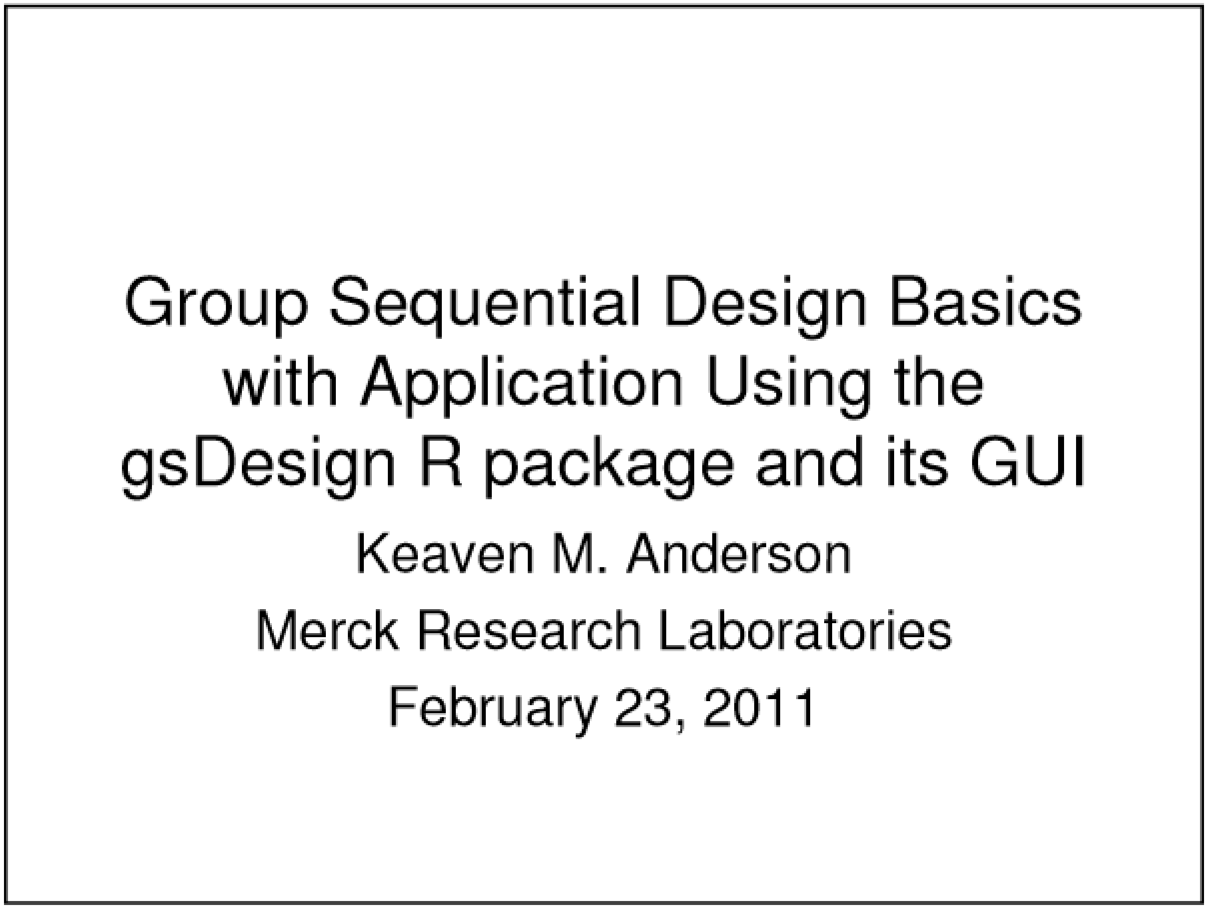
CRAN packages:

- agricolae
- AlgDesign
- asd
- BatchExperiments
- BHH2
- blockTools
- BsMD
- conf.design
- crossdes
- dae
- desirability
- DiceDesign
- DiceEval
- DiceKriging
- DiceView
- displayHTS
- DoE.base
- DoE.wrapper
- DoseFinding
- dynaTree
- experiment
- FrF2
- FrF2.catlg128
- GAD
- granova
- gsbDesign
- gsDesign
- ldDesign
- lhs
- mixexp
- mkssd
- mxkssd
- odprism
- osDesign
- planor
- plgp
- qtlDesign
- qualityTools
- RcmdrPlugin.DoE
- rsm
- SensoMineR
- support.CEs
- TEQR
- tgp
- Vdgraph

let's ^{gs}Design it!

Some R packages are real gems! Meet **gsDesign**, a package for deriving and describing group sequential designs, created by Keaven Anderson (Merck) and *REVOlution Computing* to optimize Merck's clinical trial process...

http://www.amstat.org/sections/sbiop/webinars/2011/AndersonWebinar2-23-2011.pdf



Group Sequential Design Basics with Application Using the gsDesign R package and its GUI

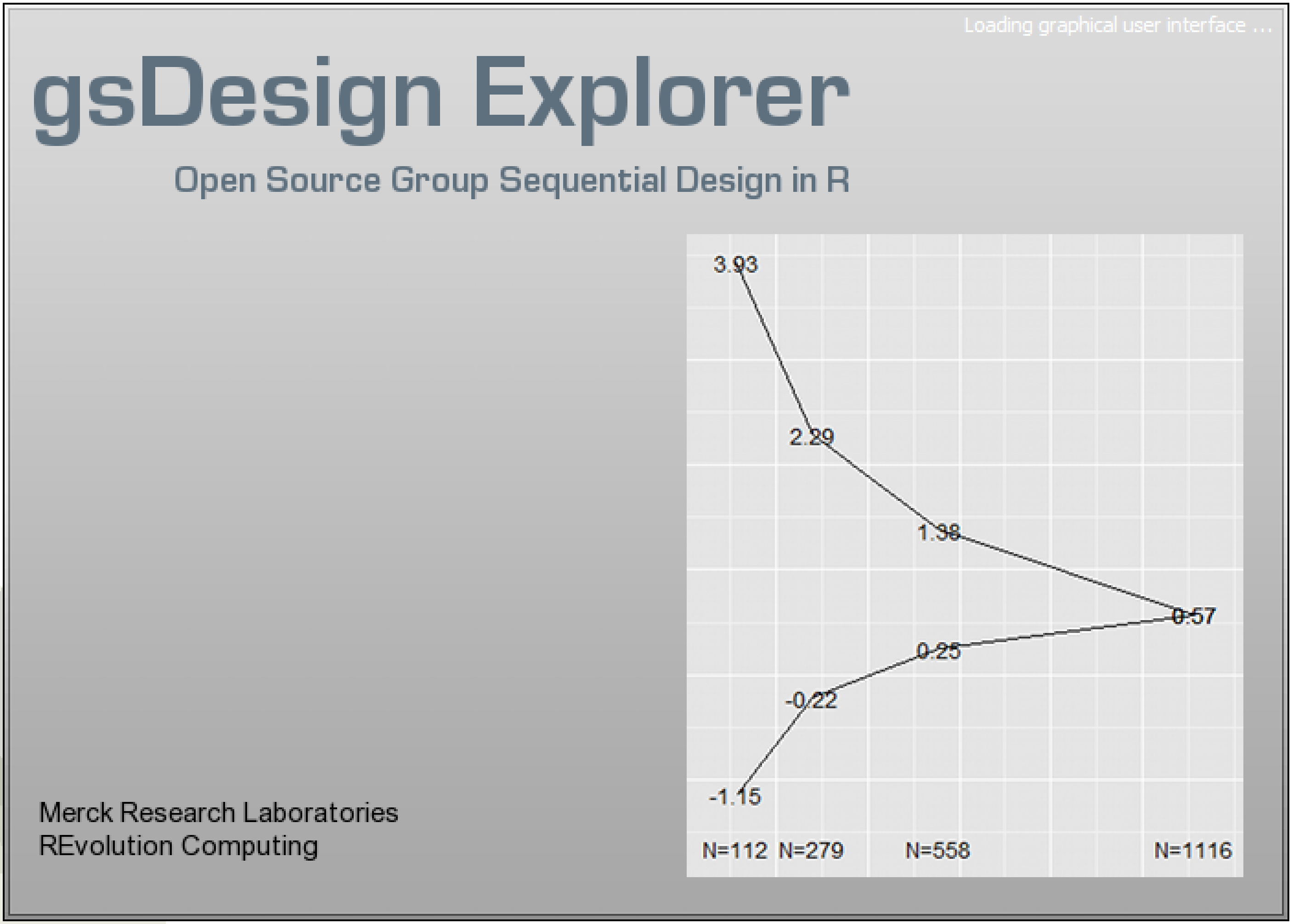
Keaven M. Anderson

Merck Research Laboratories

February 23, 2011

gsDesignExplorer

… and gsDesignExplorer - a nice GUI for gsDesign. Unfortunately it works only with 2.x version of R but, luckily, old "portable" versions of R are available.



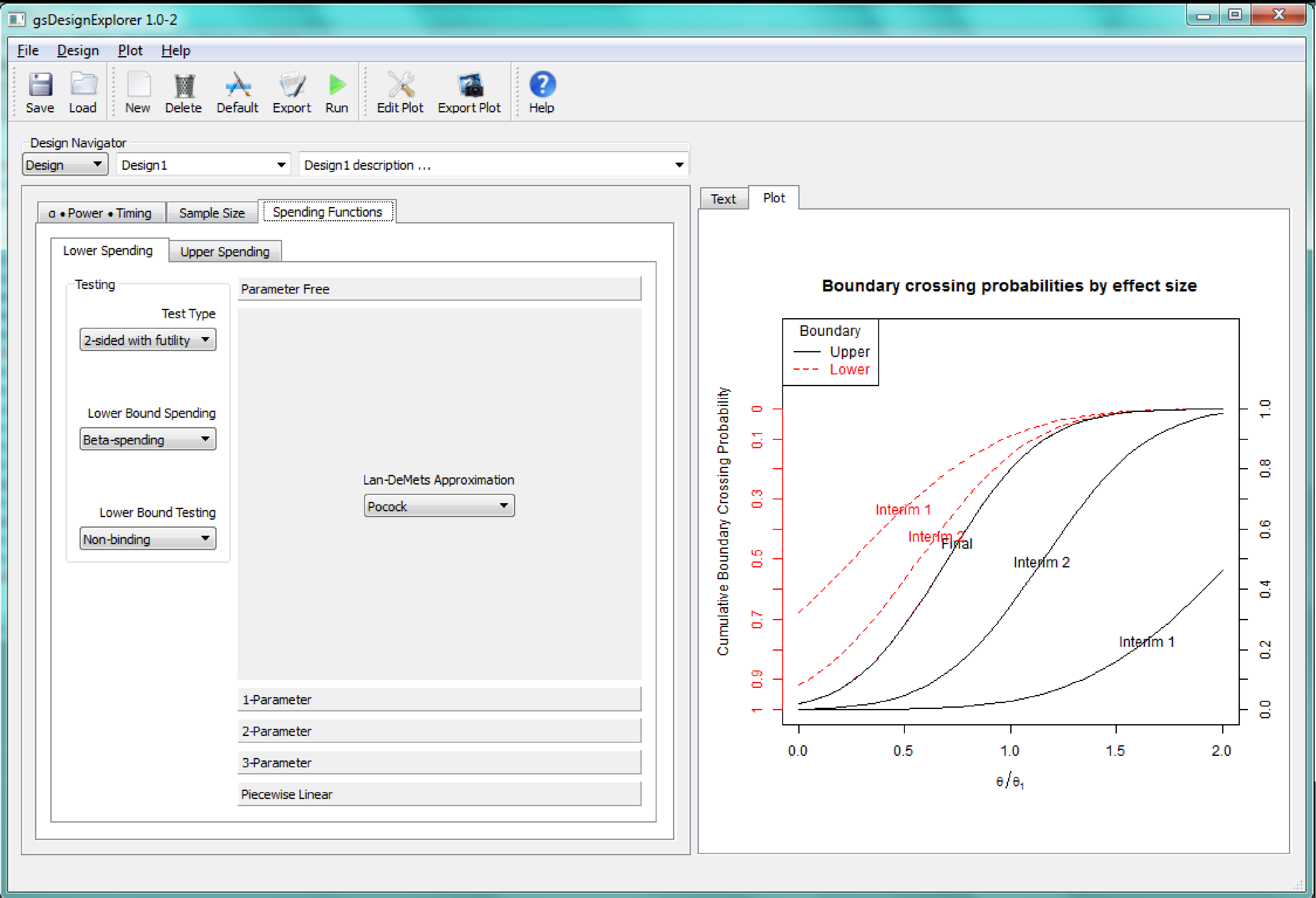
Loading graphical user interface …

gsDesign Explorer

Open Source Group Sequential Design in R

Merck Research Laboratories
REvolution Computing

gsDesignExplorer



Design Navigator: Design | Design1 | Design1 description ...

Test Type: 2-sided with futility

Lower Bound Spending: Beta-spending

Lower Bound Testing: Non-binding

Lan-DeMets Approximation: Pocock

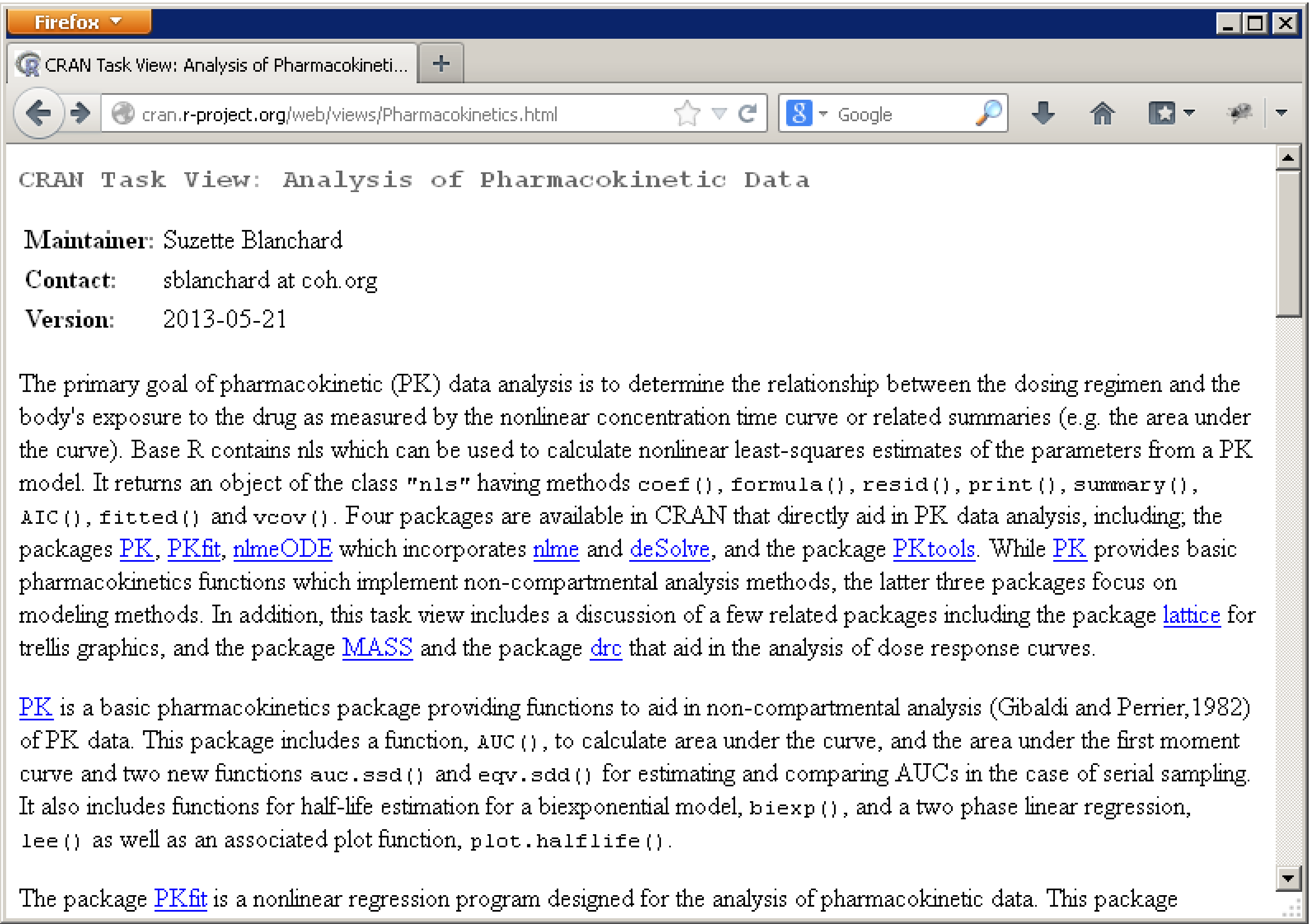
Clinical Research

http://cran.r-project.org/web/views/ClinicalTrials.html

http://cran.r-project.org/web/views/Survival.html

http://cran.r-project.org/web/views/ExperimentalDesign.html

http://cran.r-project.org/web/views/Pharmacokinetics.html



CRAN Task View: Analysis of Pharmacokinetic Data

Maintainer: Suzette Blanchard
Contact: sblanchard at coh.org
Version: 2013-05-21

The primary goal of pharmacokinetic (PK) data analysis is to determine the relationship between the dosing regimen and the body's exposure to the drug as measured by the nonlinear concentration time curve or related summaries (e.g. the area under the curve). Base R contains nls which can be used to calculate nonlinear least-squares estimates of the parameters from a PK model. It returns an object of the class `"nls"` having methods `coef()`, `formula()`, `resid()`, `print()`, `summary()`, `AIC()`, `fitted()` and `vcov()`. Four packages are available in CRAN that directly aid in PK data analysis, including; the packages PK, PKfit, nlmeODE which incorporates nlme and deSolve, and the package PKtools. While PK provides basic pharmacokinetics functions which implement non-compartmental analysis methods, the latter three packages focus on modeling methods. In addition, this task view includes a discussion of a few related packages including the package lattice for trellis graphics, and the package MASS and the package drc that aid in the analysis of dose response curves.

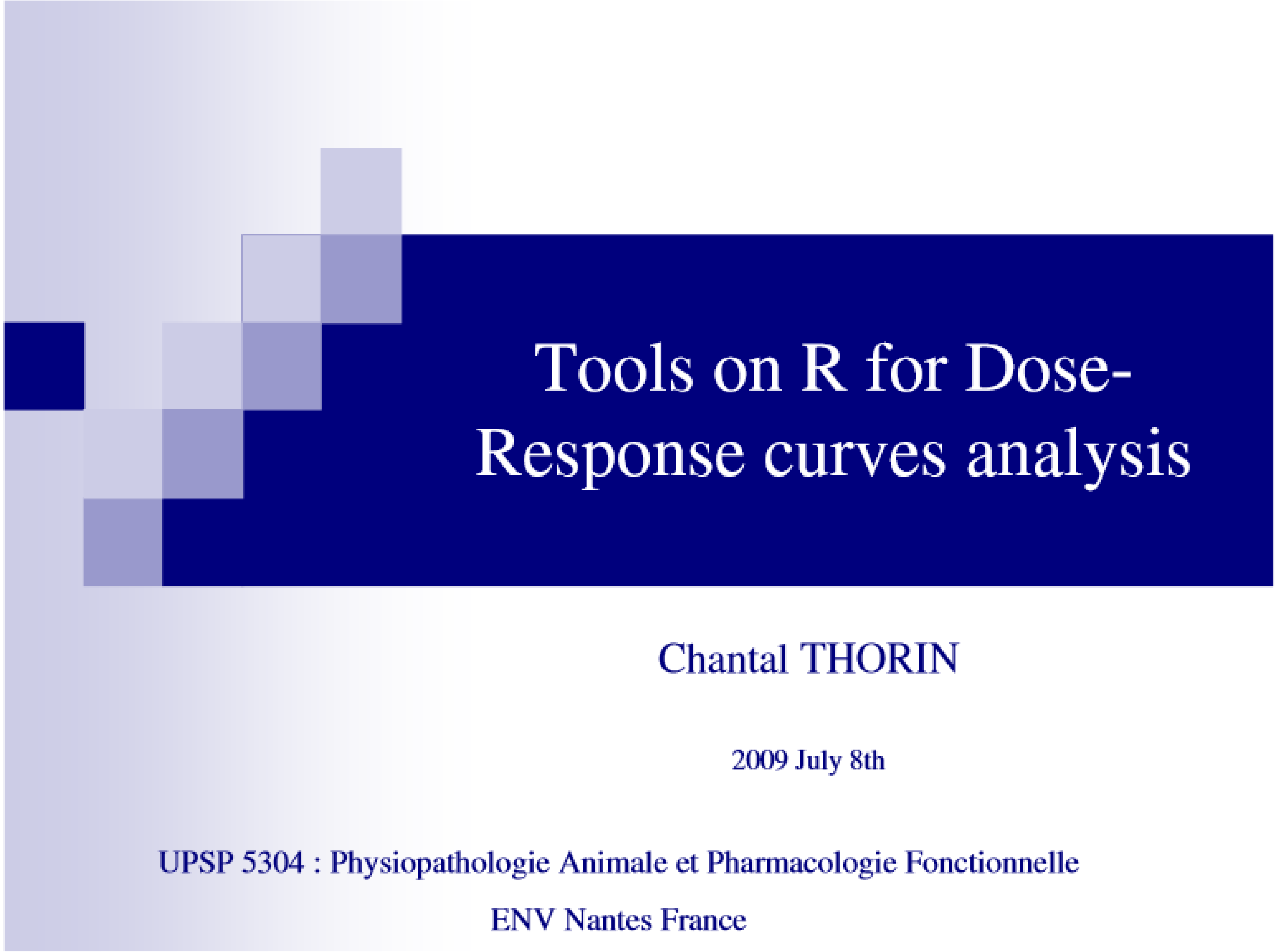
PK is a basic pharmacokinetics package providing functions to aid in non-compartmental analysis (Gibaldi and Perrier,1982) of PK data. This package includes a function, `AUC()`, to calculate area under the curve, and the area under the first moment curve and two new functions `auc.ssd()` and `eqv.sdd()` for estimating and comparing AUCs in the case of serial sampling. It also includes functions for half-life estimation for a biexponential model, `biexp()`, and a two phase linear regression, `lee()` as well as an associated plot function, `plot.halflife()`.

The package PKfit is a nonlinear regression program designed for the analysis of pharmacokinetic data. This package

CRAN packages:

- deSolve
- drc
- lattice
- MASS
- nlme
- nlmeODE
- PK
- PKfit
- PKPDmodels
- PKtools

PK/PD



Tools on R for Dose-Response curves analysis

Chantal THORIN

2009 July 8th

UPSP 5304 : Physiopathologie Animale et Pharmacologie Fonctionnelle

ENV Nantes France

http://www.r-project.org/conferences/useR-2009/slides/Thorin+Mallem+Noireaud+Desfontis.pdf

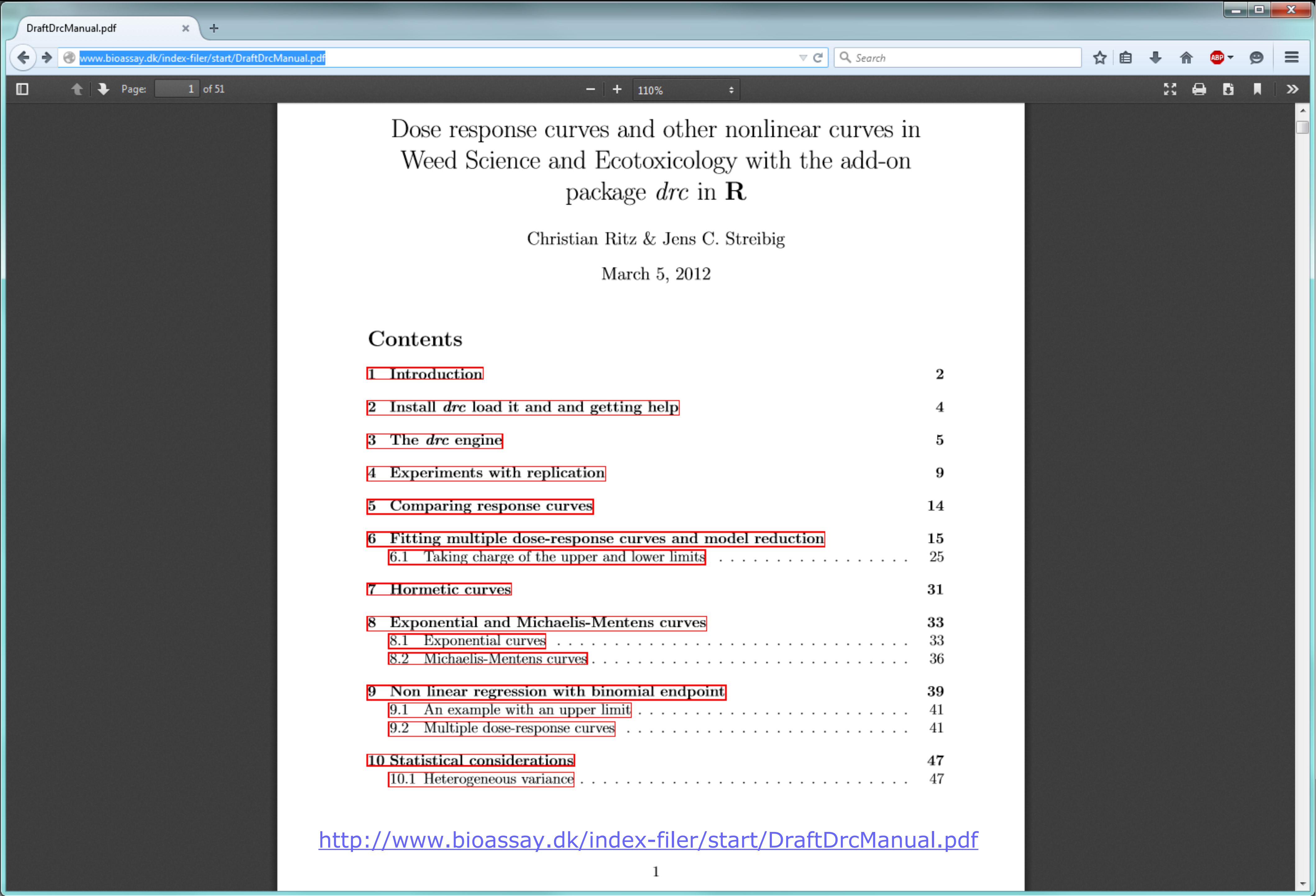
Analysis of dose-Response curves

actimL	Model function for the universal response surface approach (URSA) for the quantitative assessment of drug interaction
anova.drc	ANOVA for dose-response model fits
AR.2	Asymptotic regression model
AR.3	Asymptotic regression model
baro5	The modified baro5 function
BC.4	The Brain-Cousens hormesis models
BC.5	The Brain-Cousens hormesis models
bcl3	The Brain-Cousens hormesis models
bcl4	The Brain-Cousens hormesis models
boxcox.drc	Transform-both-sides Box-Cox transformation
braincousens	The Brain-Cousens hormesis models
cedergreen	The Cedergreen-Ritz-Streibig model
coef.drc	Extract Model Coefficients
comped	Comparison of effective dose values
compParm	Comparison of parameters
confint.drc	Confidence Intervals for model parameters
CRS.4a	The Cedergreen-Ritz-Streibig model
CRS.4b	The Cedergreen-Ritz-Streibig model
CRS.4c	The Cedergreen-Ritz-Streibig model
CRS.5a	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
CRS.5b	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
CRS.5c	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
CRS.6	The Cedergreen-Ritz-Streibig model
diagnostics	Information on estimation
drm	Fitting dose-response models
drmc	Sets control arguments
ED	Estimating effective doses
ED.drc	Estimating effective doses
ED.mrdrc	Estimating effective doses
estfun.drc	Bread and meat for the sandwich
EXD.2	Exponential decay model
EXD.3	Exponential decay model
fitted.drc	Extract fitted values from model
FPL.4	Fractional polynomial-logistic dose-response models
fplogistic	Fractional polynomial-logistic dose-response models
G.2	Mean function for the Gompertz dose-response or growth curve
G.3	Mean function for the Gompertz dose-response or growth curve
G.3u	Mean function for the Gompertz dose-response or growth curve
G.4	Mean function for the Gompertz dose-response or growth curve
genBliss	Model function for the universal response surface approach (URSA) for the quantitative assessment of drug interaction
genBliss2	Model function for the universal response surface approach (URSA) for the quantitative assessment of drug interaction
genLoewe	Model function for the universal response surface approach (URSA) for the quantitative assessment of drug interaction
genLoewe2	Model function for the universal response surface approach (URSA) for the quantitative assessment of drug interaction
genursa	Model function for the universal response surface approach (URSA) for the quantitative assessment of drug interaction
getInitial	Showing starting values used
getMeanFunctions	Display available dose-response models
gompertz	Mean function for the Gompertz dose-response or growth curve
gompertzd	The derivative of the Gompertz function
gompGrowth.1	Gompertz growth models
gompGrowth.2	Gompertz growth models
gompGrowth.3	Gompertz growth models
iceLoewe.1	Model function for the universal response surface approach (URSA) for the quantitative assessment of drug interaction
iceLoewe2.1	Model function for the universal response surface approach (URSA) for the quantitative assessment of drug interaction
isobole	Creating isobolograms
L.3	The logistic model
L.4	The logistic model
L.5	The logistic model
l2	The two-parameter log-logistic function
l3	The three-parameter log-logistic function
l3u	The three-parameter log-logistic function
l4	The four-parameter log-logistic function
l5	The five-parameter log-logistic function
lin.test	Lack-of-fit test for the mean structure based on cumulated residuals
LL.2	The two-parameter log-logistic function
LL.3	The three-parameter log-logistic function
LL.3u	The three-parameter log-logistic function
LL.4	The four-parameter log-logistic function
LL.5	The five-parameter log-logistic function
LL2.2	The two-parameter log-logistic function
LL2.3	The three-parameter log-logistic function
LL2.3u	The three-parameter log-logistic function
LL2.4	The four-parameter log-logistic function
LL2.5	The five-parameter log-logistic function
llogistic	The log-logistic function
llogistic2	The log-logistic function
LN.2	Log-normal dose-response model
LN.3	Log-normal dose-response model
LN.3u	Log-normal dose-response model
LN.4	Log-normal dose-response model
lnormal	Log-normal dose-response model
logistic	The logistic model
logLik.drc	Extracting the log likelihood
maED	Estimation of ED values using model-averaging
MAX	Maximum mean response
mixture	Fitting binary mixture models
ml3a	The Cedergreen-Ritz-Streibig model
ml3b	The Cedergreen-Ritz-Streibig model
ml3c	The Cedergreen-Ritz-Streibig model
ml4a	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
ml4b	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
ml4c	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
MM.2	Michaelis-Menten model
MM.3	Michaelis-Menten model
modelFit	Assessing the model fit

Analysis of dose-Response curves [cont'd]

mr.test	Mizon-Richard test for dose-response models
mrdrm	Model-robust dose-response modelling
mselect	Model selection
NEC	Dose-response model for estimation of no effect concentration (NEC).
NEC.2	Dose-response model for estimation of no effect concentration (NEC).
NEC.3	Dose-response model for estimation of no effect concentration (NEC).
NEC.4	Dose-response model for estimation of no effect concentration (NEC).
neill.test	Neill's lack-of-fit test for dose-response models
plot.drc	Plotting fitted curves for a 'drc' or 'mrdrc' object
plot.mrdrc	Plotting fitted curves for a 'drc' or 'mrdrc' object
PR	Expected or predicted response
predict.drc	Prediction
predict.mrdrc	Prediction
print.drc	Printing key features
print.mrdrc	Printing key features
print.summary.drc	Printing summary of non-linear model fits
rdrm	Simulating a dose-response curve
relpot	Comparing selectivity indices across curves
residuals.drc	Extracting residuals from model
ryegrass	Effect of ferulic acid on growth of ryegrass
S.capricornutum	Effect of cadmium on growth of green alga
searchdrc	Searching through a range of initial parameter values to obtain convergence
secalonic	Root length measurements
SI	Comparing selectivity indices across curves
simDR	Simulating ED values under various scenarios
summary.drc	Summarising non-linear model fits
twophase	Two-phase dose-response model
ucedergreen	The Cedergreen-Ritz-Streibig model
UCRS.4a	The Cedergreen-Ritz-Streibig model
UCRS.4b	The Cedergreen-Ritz-Streibig model
UCRS.4c	The Cedergreen-Ritz-Streibig model
UCRS.5a	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
UCRS.5b	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
UCRS.5c	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
uml3a	The Cedergreen-Ritz-Streibig model
uml3b	The Cedergreen-Ritz-Streibig model
uml3c	The Cedergreen-Ritz-Streibig model
uml4a	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
uml4b	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
uml4c	Cedergreen-Ritz-Streibig dose-reponse model for describing hormesis
update.drc	Updating and re-fitting a model
ursa	Model function for the universal response surface approach (URSA) for the quantitative assessment of drug interaction
vcov.drc	Calculating variance-covariance matrix for objects of class 'drc'
W1.2	The two-parameter Weibull functions
W1.3	The three-parameter Weibull functions
W1.3u	The three-parameter Weibull functions
W1.4	The four-parameter Weibull functions
w2	The two-parameter Weibull functions
W2.2	The two-parameter Weibull functions
W2.3	The three-parameter Weibull functions
W2.3u	The three-parameter Weibull functions
W2.4	The four-parameter Weibull functions
W2x.3	The three-parameter Weibull functions
W2x.4	The four-parameter Weibull functions
w3	The three-parameter Weibull functions
w4	The four-parameter Weibull functions
weibull1	Weibull model functions
weibull2	Weibull model functions
weibull2x	Weibull model functions
yieldLoss	Calculating yield loss parameters

Analysis of dose-Response curves [cont'd]



Dose response curves and other nonlinear curves in Weed Science and Ecotoxicology with the add-on package *drc* in R

Christian Ritz & Jens C. Streibig

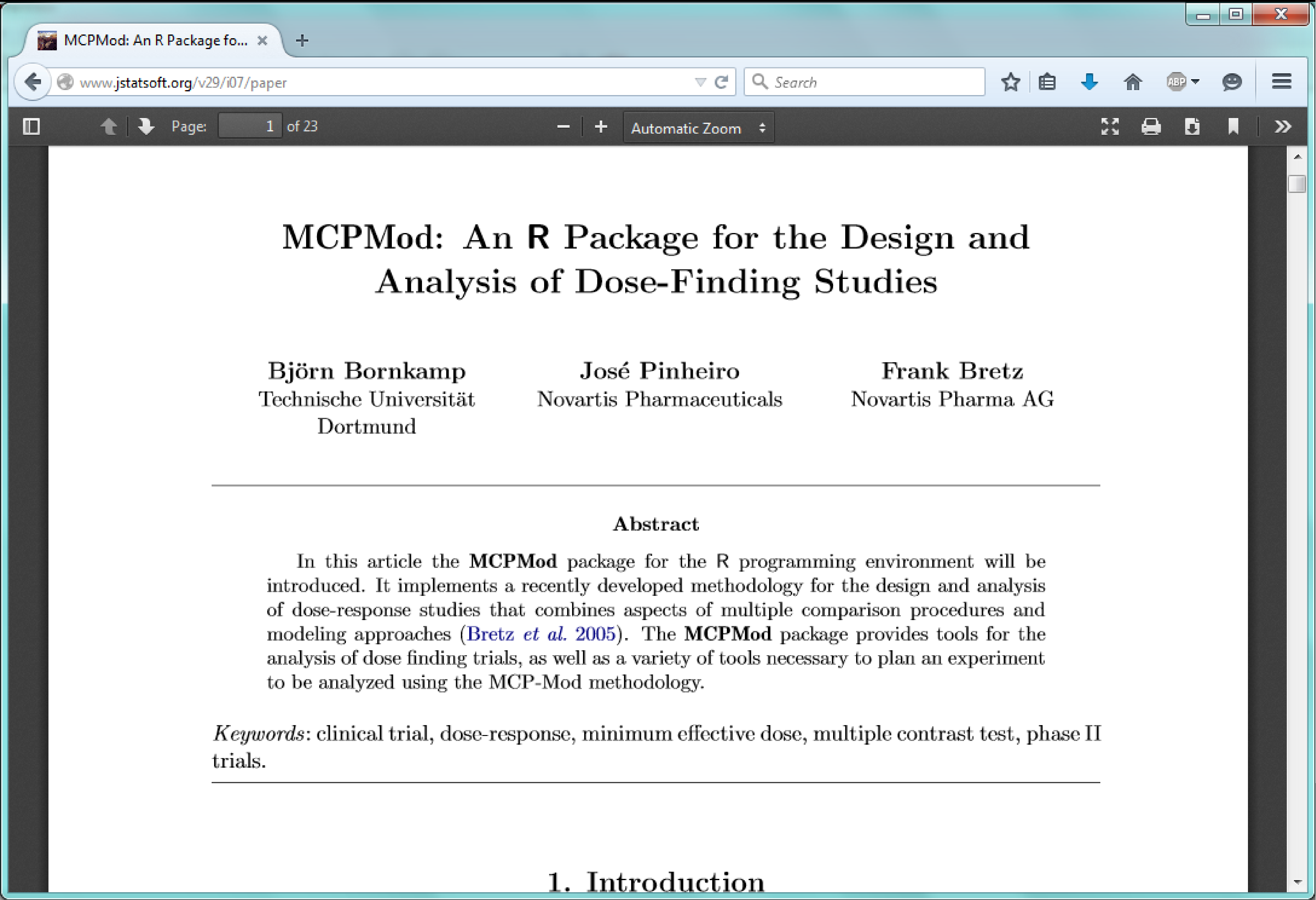
March 5, 2012

Contents

http://www.bioassay.dk/index-filer/start/DraftDrcManual.pdf

1

They use R



MCPMod: An R Package for the Design and Analysis of Dose-Finding Studies

Björn Bornkamp
Technische Universität Dortmund

José Pinheiro
Novartis Pharmaceuticals

Frank Bretz
Novartis Pharma AG

Abstract

In this article the **MCPMod** package for the R programming environment will be introduced. It implements a recently developed methodology for the design and analysis of dose-response studies that combines aspects of multiple comparison procedures and modeling approaches (Bretz *et al.* 2005). The **MCPMod** package provides tools for the analysis of dose finding trials, as well as a variety of tools necessary to plan an experiment to be analyzed using the MCP-Mod methodology.

Keywords: clinical trial, dose-response, minimum effective dose, multiple contrast test, phase II trials.

1. Introduction

Advanced PK modeling with mixed effects

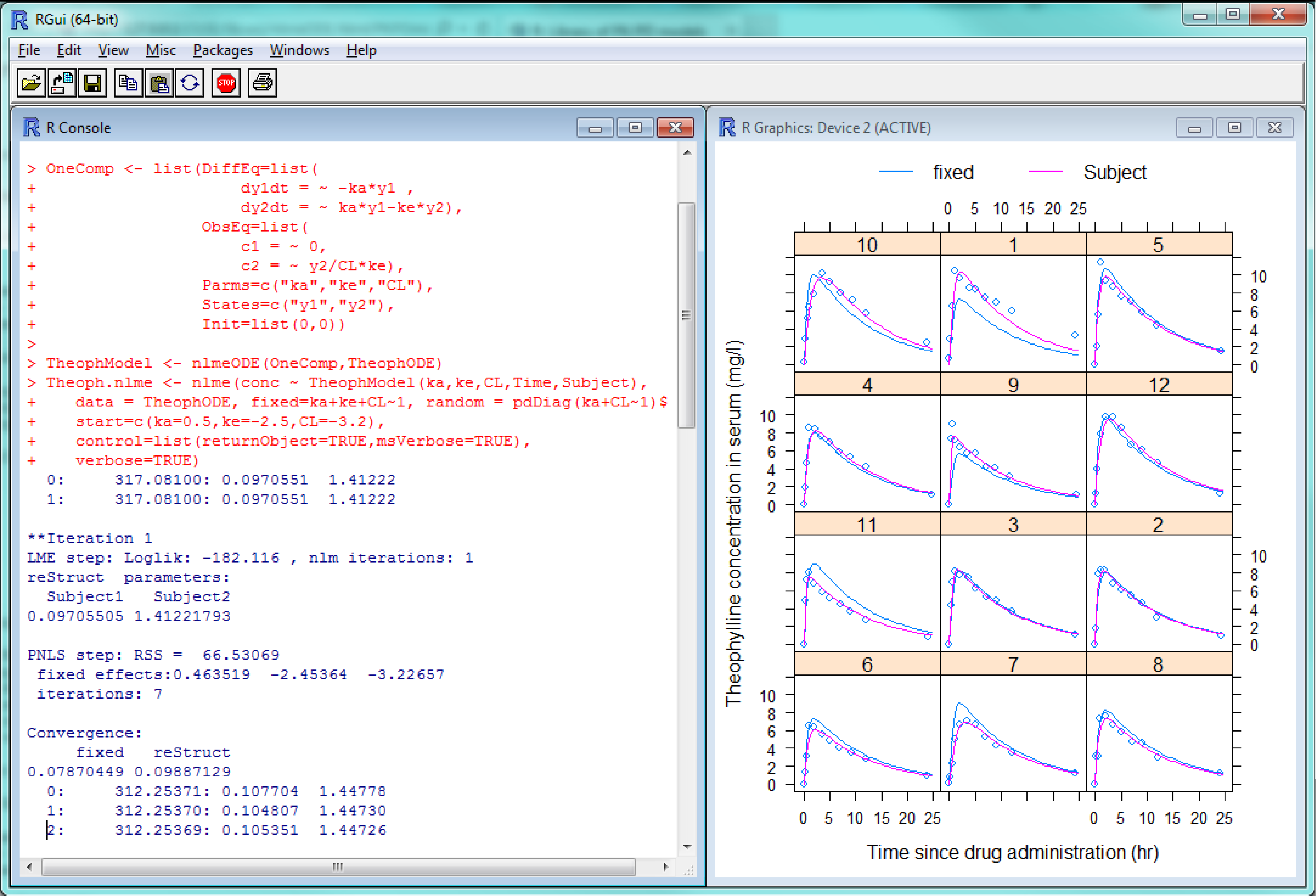
nlmeODE is a powerful combination of the famous nlme package with deSolve (an ordinary differential equations solver) for non-linear, mixed-effects modelling, designed especially for pharmaceutical research.

The package also includes several PK examples such as one- and two-compartment models with multiple doses and infusions.

- Pharmacokinetics of Theophylline
- Pharmacokinetics of Indomethacine
- Absorption model with estimation of time/rate of infusion
- Simulation and simultaneous estimation of PK/PD data
- Minimal Model of Glucose and Insulin
- Minimal Model of Glucose using observed insulin as forcing function

http://www.inside-r.org/packages/cran/nlmeODE/docs/PKPDmodels

Advanced PK modeling with mixed effects



```
> OneComp <- list(DiffEq=list(
+                     dy1dt = ~ -ka*y1 ,
+                     dy2dt = ~ ka*y1-ke*y2),
+                 ObsEq=list(
+                     c1 = ~ 0,
+                     c2 = ~ y2/CL*ke),
+                 Parms=c("ka","ke","CL"),
+                 States=c("y1","y2"),
+                 Init=list(0,0))
>
> TheophModel <- nlmeODE(OneComp,TheophODE)
> Theoph.nlme <- nlme(conc ~ TheophModel(ka,ke,CL,Time,Subject),
+    data = TheophODE, fixed=ka+ke+CL~1, random = pdDiag(ka+CL~1)$
+    start=c(ka=0.5,ke=-2.5,CL=-3.2),
+    control=list(returnObject=TRUE,msVerbose=TRUE),
+    verbose=TRUE)
  0:     317.08100: 0.0970551  1.41222
  1:     317.08100: 0.0970551  1.41222

**Iteration 1
LME step: Loglik: -182.116 , nlm iterations: 1
reStruct  parameters:
  Subject1   Subject2
0.09705505 1.41221793

PNLS step: RSS =  66.53069
 fixed effects:0.463519  -2.45364  -3.22657
 iterations: 7

Convergence:
     fixed   reStruct
0.07870449 0.09887129
  0:     312.25371: 0.107704   1.44778
  1:     312.25370: 0.104807   1.44730
  2:     312.25369: 0.105351   1.44726
```

R + ADMB (automatic diffeRentiation!)

You are here: Home

- FAQ
- Download
- Documentation
- **API-Function reference**
- User Community
- Users mailing lists
- Courses
- Repository
- Issue tracker
- Donate

Welcome

AD Model Builder, or ADMB, is a powerful software package for the development of state-of-the-art nonlinear statistical models. ADMB is built around the AUTODIF Library, a C++ language extension which implements reverse mode automatic differentiation. A closely related software package, ADMB-RE, implements random effects in nonlinear models.
ADMB was created by David Fournier and now continues to be developed by the ADMB Project, a creation of the non-profit ADMB Foundation. ADMB is free, open source, and available for Windows, Linux, MacOS, and Sun/SPARC.

How to cite ADMB:

Fournier, D.A., H.J. Skaug, J. Ancheta, J. Ianelli, A. Magnusson, M.N. Maunder, A. Nielsen, and J. Sibert. 2012. AD Model Builder: using automatic differentiation for statistical inference of highly parameterized complex nonlinear models. Optim. Methods Softw. 27:233-249.

ADMB project background

News

ADMB-11.4 released
May 29, 2015

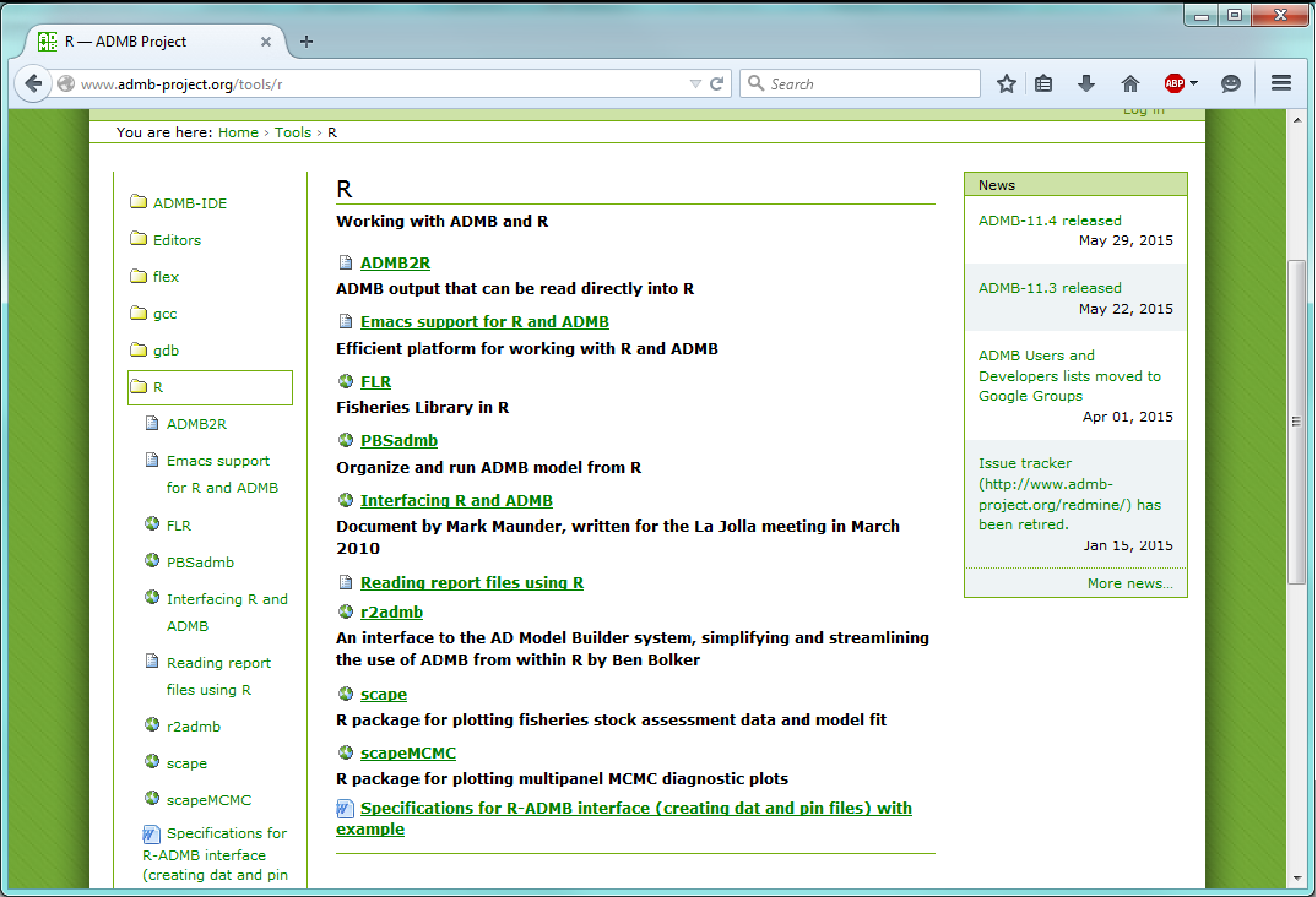
ADMB-11.3 released
May 22, 2015

ADMB Users and Developers lists moved to Google Groups
Apr 01, 2015

Issue tracker (http://www.admb-project.org/redmine/) has been retired.
Jan 15, 2015

More news…

R + ADMB (automatic diffeRentiation!)



You are here: Home › Tools › R

- ADMB-IDE
- Editors
- flex
- gcc
- gdb
- R
  - ADMB2R
  - Emacs support for R and ADMB
  - FLR
  - PBSadmb
  - Interfacing R and ADMB
  - Reading report files using R
  - r2admb
  - scape
  - scapeMCMC
  - Specifications for R-ADMB interface (creating dat and pin

R

Working with ADMB and R

ADMB2R

ADMB output that can be read directly into R

Emacs support for R and ADMB

Efficient platform for working with R and ADMB

FLR

Fisheries Library in R

PBSadmb

Organize and run ADMB model from R

Interfacing R and ADMB

Document by Mark Maunder, written for the La Jolla meeting in March 2010

Reading report files using R

r2admb

An interface to the AD Model Builder system, simplifying and streamlining the use of ADMB from within R by Ben Bolker

scape

R package for plotting fisheries stock assessment data and model fit

scapeMCMC

R package for plotting multipanel MCMC diagnostic plots

Specifications for R-ADMB interface (creating dat and pin files) with example

News

ADMB-11.4 released
May 29, 2015

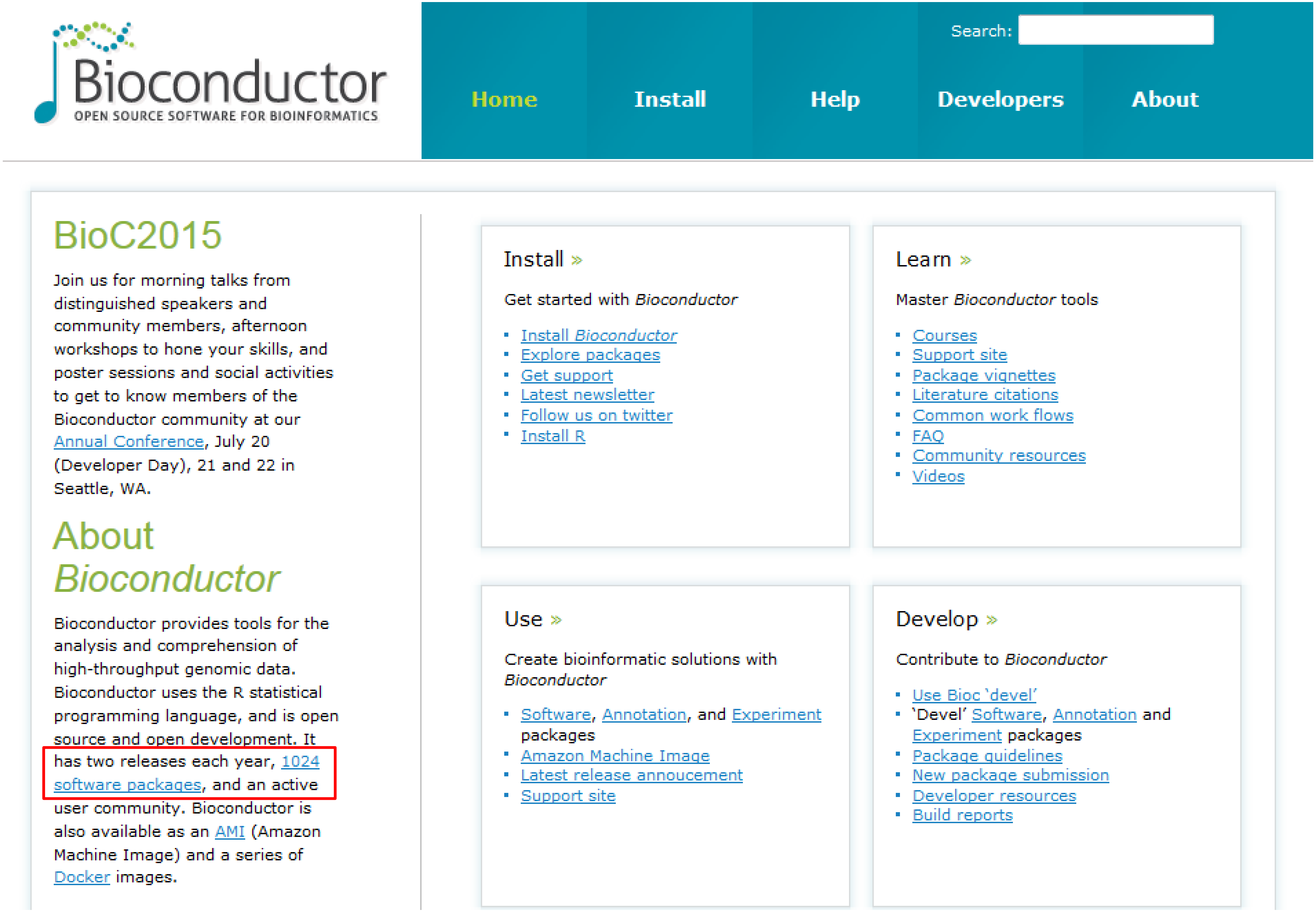
ADMB-11.3 released
May 22, 2015

ADMB Users and Developers lists moved to Google Groups
Apr 01, 2015

Issue tracker (http://www.admb-project.org/redmine/) has been retired.
Jan 15, 2015

More news...

Genomic data analysis



BioC2015

Join us for morning talks from distinguished speakers and community members, afternoon workshops to hone your skills, and poster sessions and social activities to get to know members of the Bioconductor community at our Annual Conference, July 20 (Developer Day), 21 and 22 in Seattle, WA.

About *Bioconductor*

Bioconductor provides tools for the analysis and comprehension of high-throughput genomic data. Bioconductor uses the R statistical programming language, and is open source and open development. It has two releases each year, 1024 software packages, and an active user community. Bioconductor is also available as an AMI (Amazon Machine Image) and a series of Docker images.

Install »

Get started with *Bioconductor*

- Install *Bioconductor*
- Explore packages
- Get support
- Latest newsletter
- Follow us on twitter
- Install R

Learn »

Master *Bioconductor* tools

- Courses
- Support site
- Package vignettes
- Literature citations
- Common work flows
- FAQ
- Community resources
- Videos

Use »

Create bioinformatic solutions with *Bioconductor*

- Software, Annotation, and Experiment packages
- Amazon Machine Image
- Latest release annoucement
- Support site

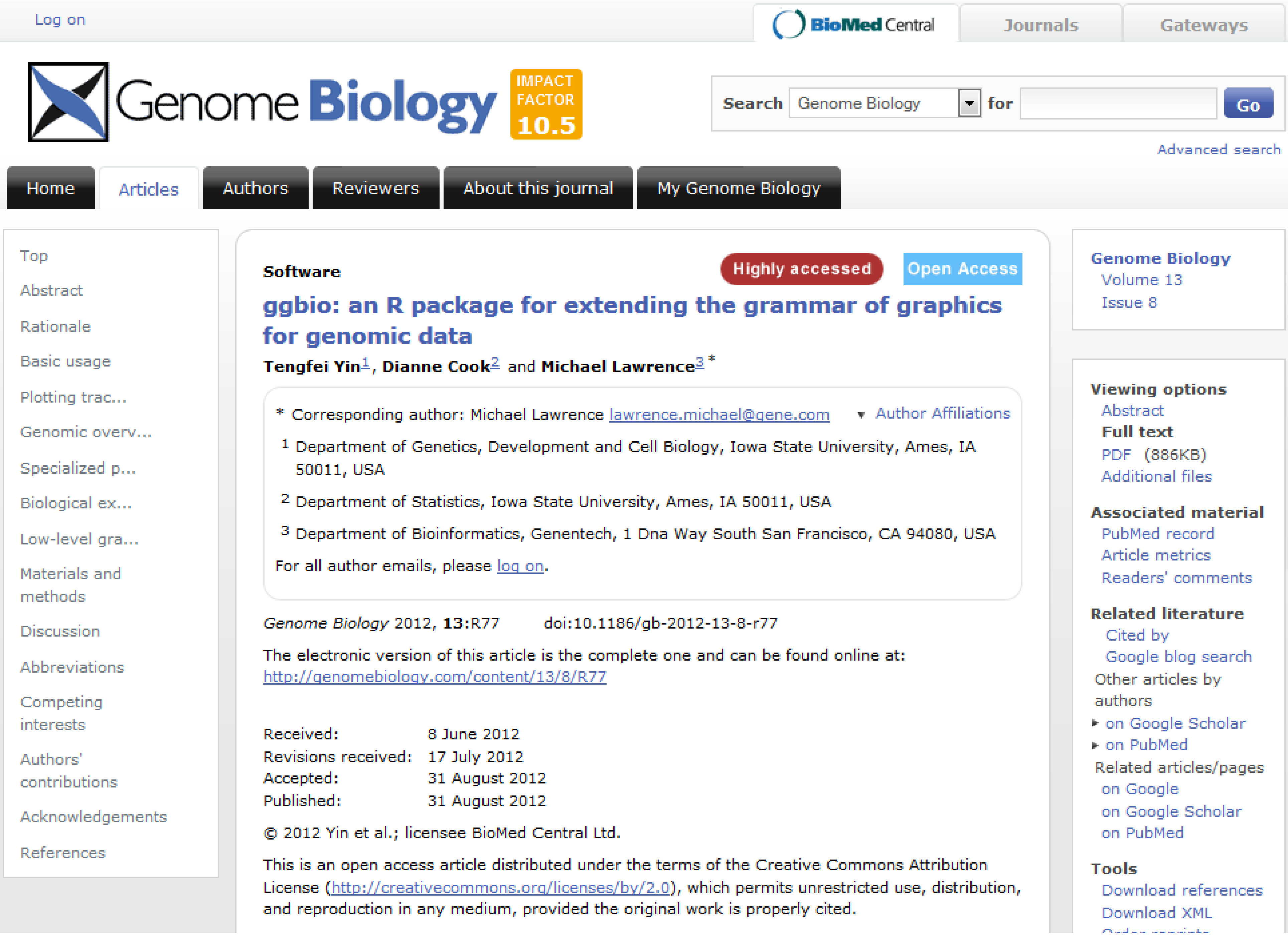
Develop »

Contribute to *Bioconductor*

- Use Bioc 'devel'
- 'Devel' Software, Annotation and Experiment packages
- Package guidelines
- New package submission
- Developer resources
- Build reports

http://tinyurl.com/bioconductor-packages

Genomic data analysis



Software

ggbio: an R package for extending the grammar of graphics for genomic data

Tengfei Yin[1], **Dianne Cook**[2] and **Michael Lawrence**[3] *

* Corresponding author: Michael Lawrence lawrence.michael@gene.com ▾ Author Affiliations

[1] Department of Genetics, Development and Cell Biology, Iowa State University, Ames, IA 50011, USA

[2] Department of Statistics, Iowa State University, Ames, IA 50011, USA

[3] Department of Bioinformatics, Genentech, 1 Dna Way South San Francisco, CA 94080, USA

For all author emails, please log on.

Genome Biology 2012, **13**:R77 doi:10.1186/gb-2012-13-8-r77

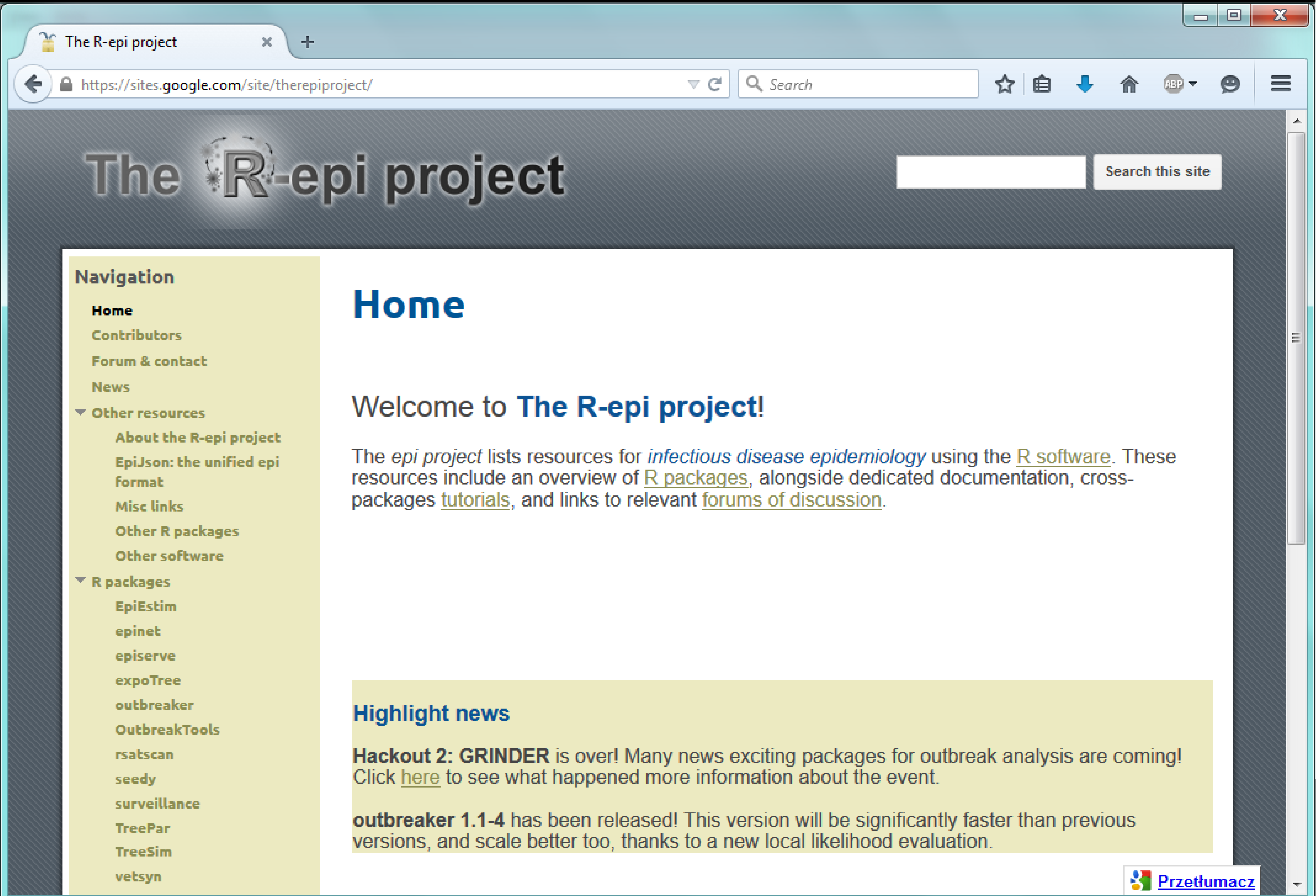
The electronic version of this article is the complete one and can be found online at: http://genomebiology.com/content/13/8/R77

Received:	8 June 2012
Revisions received:	17 July 2012
Accepted:	31 August 2012
Published:	31 August 2012

© 2012 Yin et al.; licensee BioMed Central Ltd.

This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/2.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

R for Epidemiology



Home

Welcome to **The R-epi project!**

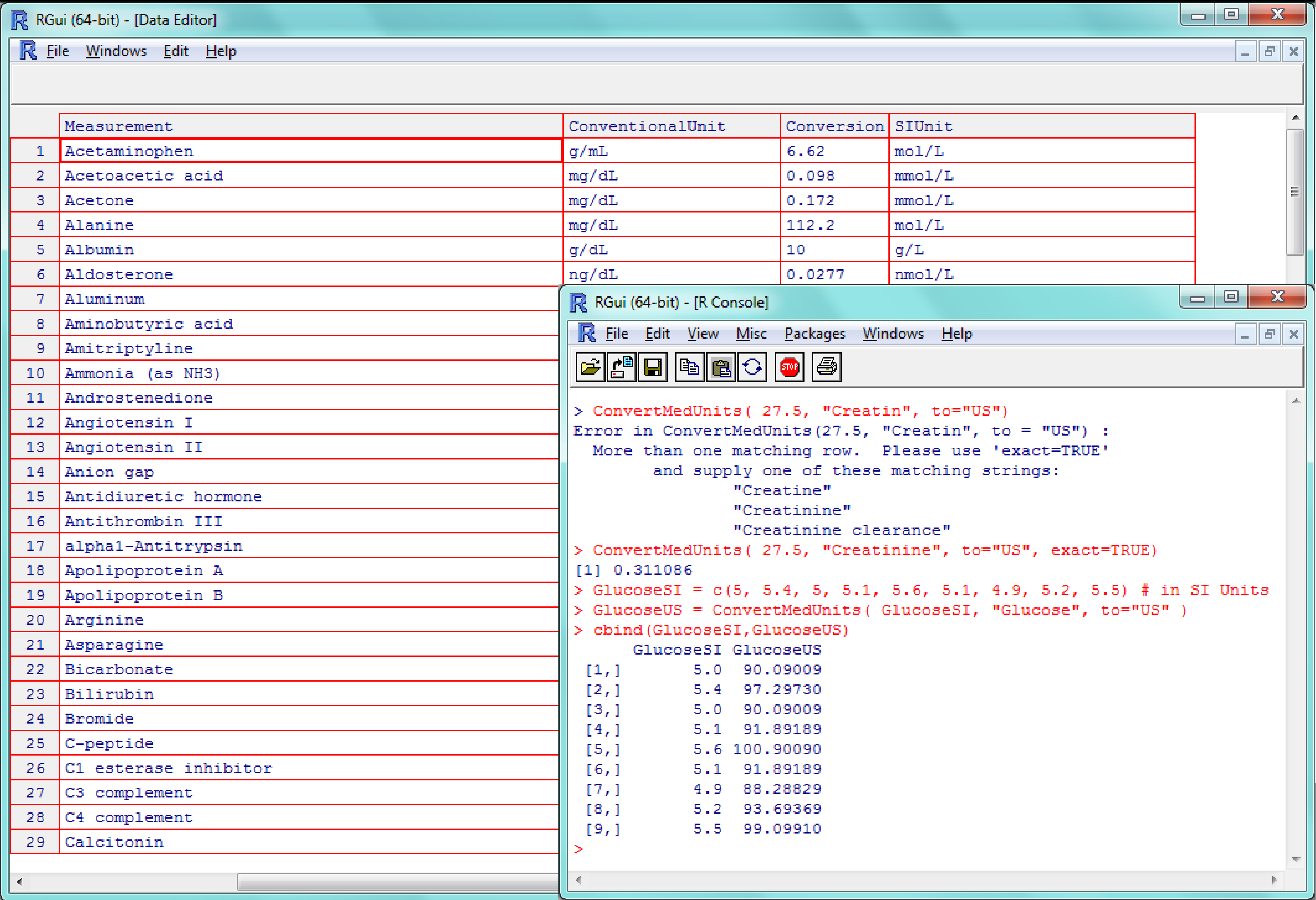
The *epi project* lists resources for *infectious disease epidemiology* using the R software. These resources include an overview of R packages, alongside dedicated documentation, cross-packages tutorials, and links to relevant forums of discussion.

Highlight news

Hackout 2: GRINDER is over! Many news exciting packages for outbreak analysis are coming! Click here to see what happened more information about the event.

outbreaker 1.1-4 has been released! This version will be significantly faster than previous versions, and scale better too, thanks to a new local likelihood evaluation.

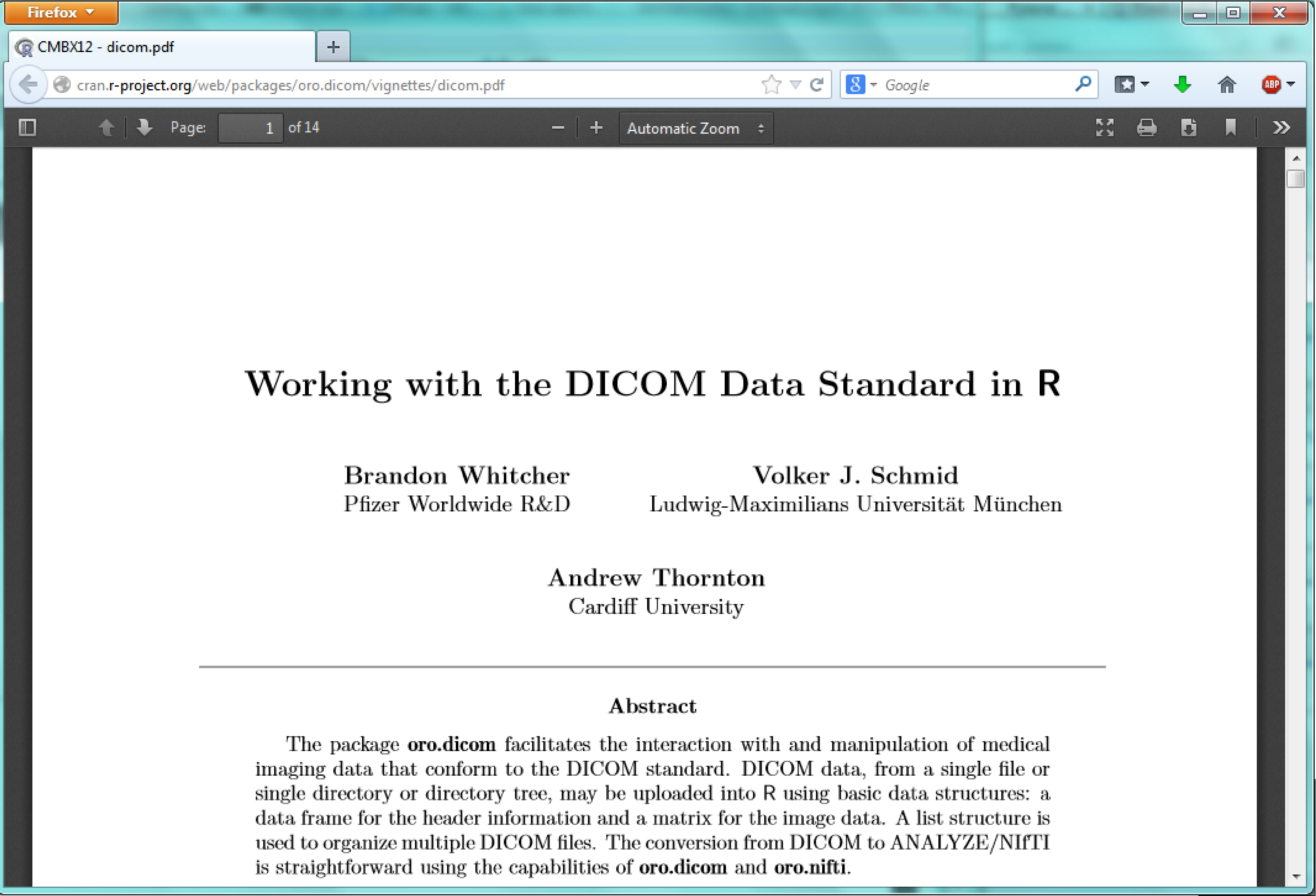
R for laboratory diagnostics



	Measurement	ConventionalUnit	Conversion	SIUnit
1	Acetaminophen	g/mL	6.62	mol/L
2	Acetoacetic acid	mg/dL	0.098	mmol/L
3	Acetone	mg/dL	0.172	mmol/L
4	Alanine	mg/dL	112.2	mol/L
5	Albumin	g/dL	10	g/L
6	Aldosterone	ng/dL	0.0277	nmol/L
7	Aluminum			
8	Aminobutyric acid			
9	Amitriptyline			
10	Ammonia (as NH3)			
11	Androstenedione			
12	Angiotensin I			
13	Angiotensin II			
14	Anion gap			
15	Antidiuretic hormone			
16	Antithrombin III			
17	alpha1-Antitrypsin			
18	Apolipoprotein A			
19	Apolipoprotein B			
20	Arginine			
21	Asparagine			
22	Bicarbonate			
23	Bilirubin			
24	Bromide			
25	C-peptide			
26	C1 esterase inhibitor			
27	C3 complement			
28	C4 complement			
29	Calcitonin			

```
> ConvertMedUnits( 27.5, "Creatin", to="US")
Error in ConvertMedUnits(27.5, "Creatin", to = "US") :
  More than one matching row.  Please use 'exact=TRUE'
        and supply one of these matching strings:
                "Creatine"
                "Creatinine"
                "Creatinine clearance"
> ConvertMedUnits( 27.5, "Creatinine", to="US", exact=TRUE)
[1] 0.311086
> GlucoseSI = c(5, 5.4, 5, 5.1, 5.6, 5.1, 4.9, 5.2, 5.5) # in SI Units
> GlucoseUS = ConvertMedUnits( GlucoseSI, "Glucose", to="US" )
> cbind(GlucoseSI,GlucoseUS)
      GlucoseSI GlucoseUS
 [1,]       5.0  90.09009
 [2,]       5.4  97.29730
 [3,]       5.0  90.09009
 [4,]       5.1  91.89189
 [5,]       5.6 100.90090
 [6,]       5.1  91.89189
 [7,]       4.9  88.28829
 [8,]       5.2  93.69369
 [9,]       5.5  99.09910
>
```

Process DICOM data with oro.dicom



Working with the DICOM Data Standard in R

Brandon Whitcher
Pfizer Worldwide R&D

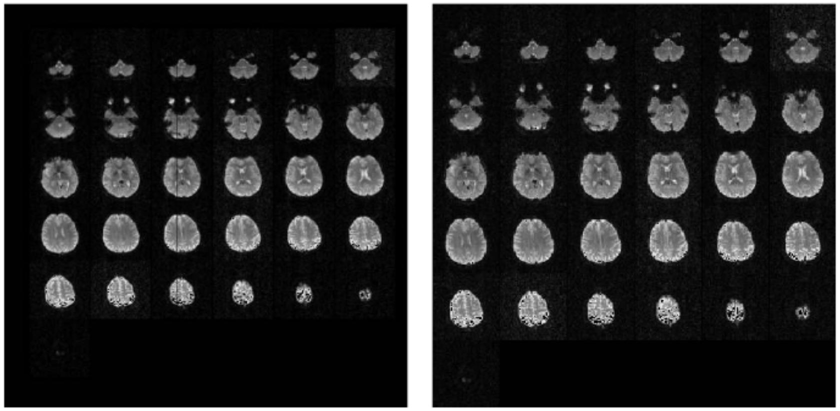
Volker J. Schmid
Ludwig-Maximilians Universität München

Andrew Thornton
Cardiff University

Abstract

The package **oro.dicom** facilitates the interaction with and manipulation of medical imaging data that conform to the DICOM standard. DICOM data, from a single file or single directory or directory tree, may be uploaded into R using basic data structures: a data frame for the header information and a matrix for the image data. A list structure is used to organize multiple DICOM files. The conversion from DICOM to ANALYZE/NIfTI is straightforward using the capabilities of **oro.dicom** and **oro.nifti**.

Process DICOM data with oro.dicom



(a) (b)

Figure 3: (a) Single MOSAIC image as read in from `readDICOMFile`. (b) Lightbox display of three-dimensional array of images after processing via `create3D`.

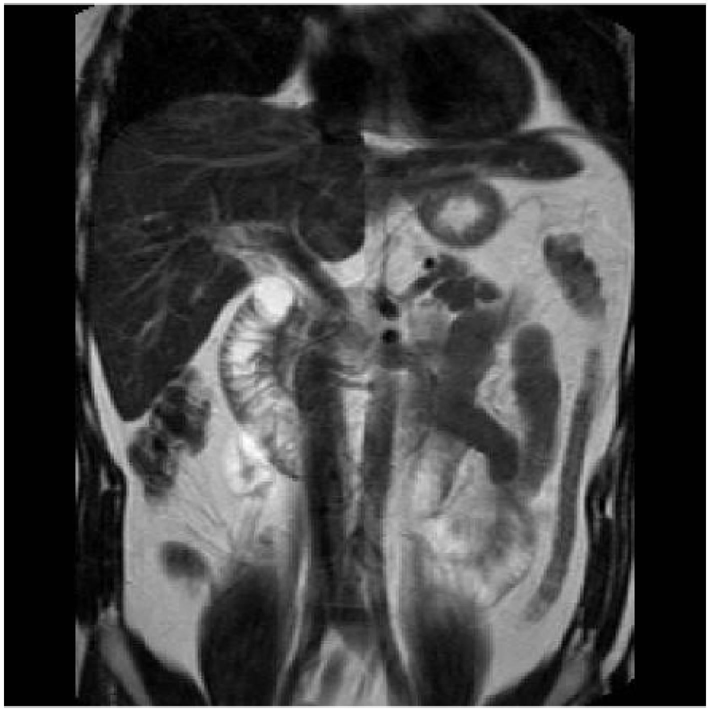
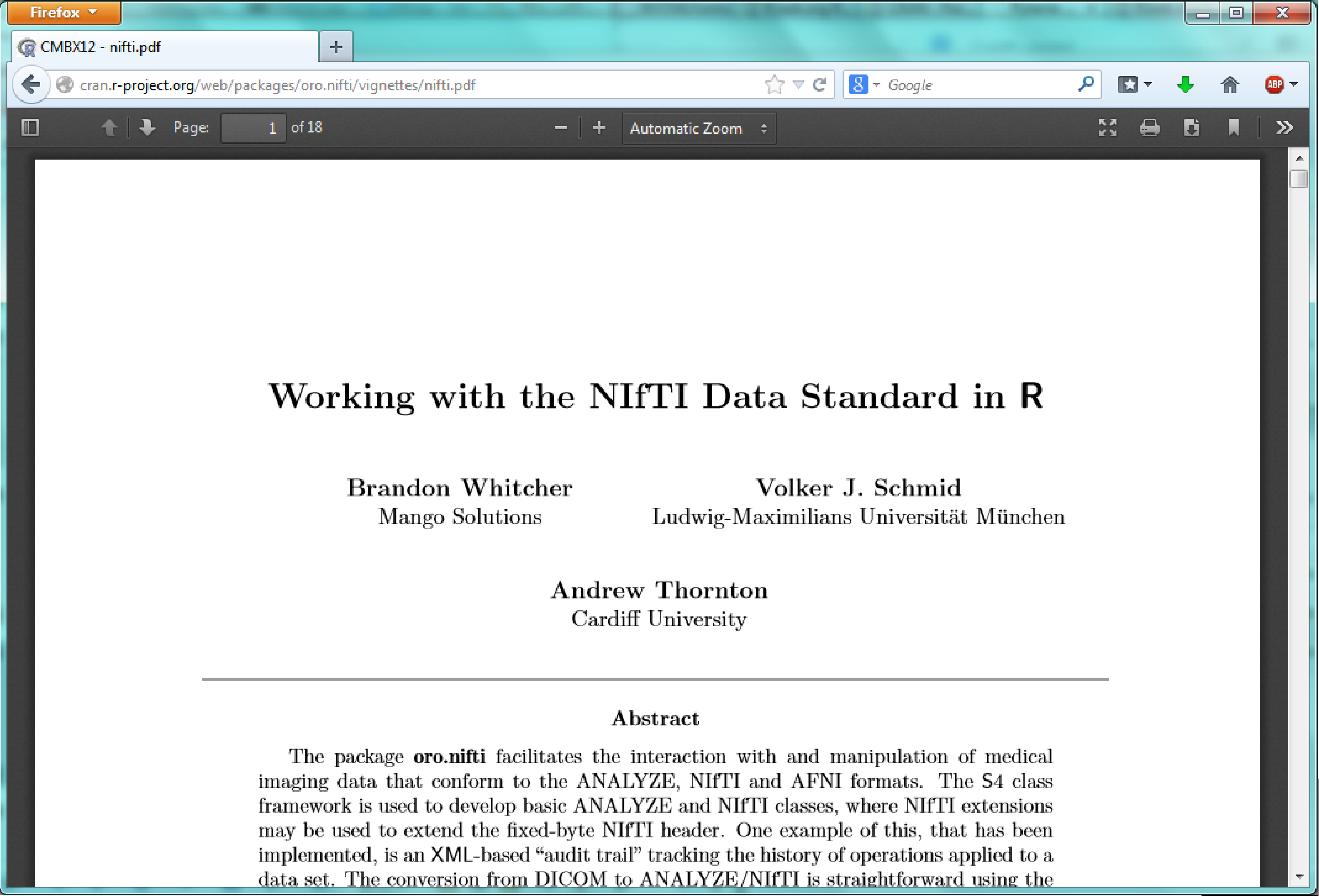


Figure 2: Coronal slice of the abdomen viewed in *neurological* convention (left is right and right is left).

```
R> tail(abdo$hdr)

   group element                name code length     value sequence
79  0028    0101          BitsStored   US      2        12
80  0028    0102             HighBit   US      2        11
81  0028    0103 PixelRepresentation   US      2         0
82  0028    1050        WindowCenter   DS      4       530
83  0028    1051         WindowWidth   DS      4      1052
84  7FE0    0010           PixelData   OB 131072 PixelData
```

http://cran.r-project.org/web/packages/oro.dicom/index.html

Process NIfTI data with oro.nifti



Working with the NIfTI Data Standard in R

Brandon Whitcher
Mango Solutions

Volker J. Schmid
Ludwig-Maximilians Universität München

Andrew Thornton
Cardiff University

Abstract

The package **oro.nifti** facilitates the interaction with and manipulation of medical imaging data that conform to the ANALYZE, NIfTI and AFNI formats. The S4 class framework is used to develop basic ANALYZE and NIfTI classes, where NIfTI extensions may be used to extend the fixed-byte NIfTI header. One example of this, that has been implemented, is an XML-based "audit trail" tracking the history of operations applied to a data set. The conversion from DICOM to ANALYZE/NIfTI is straightforward using the

The Origin Of Things - RForge

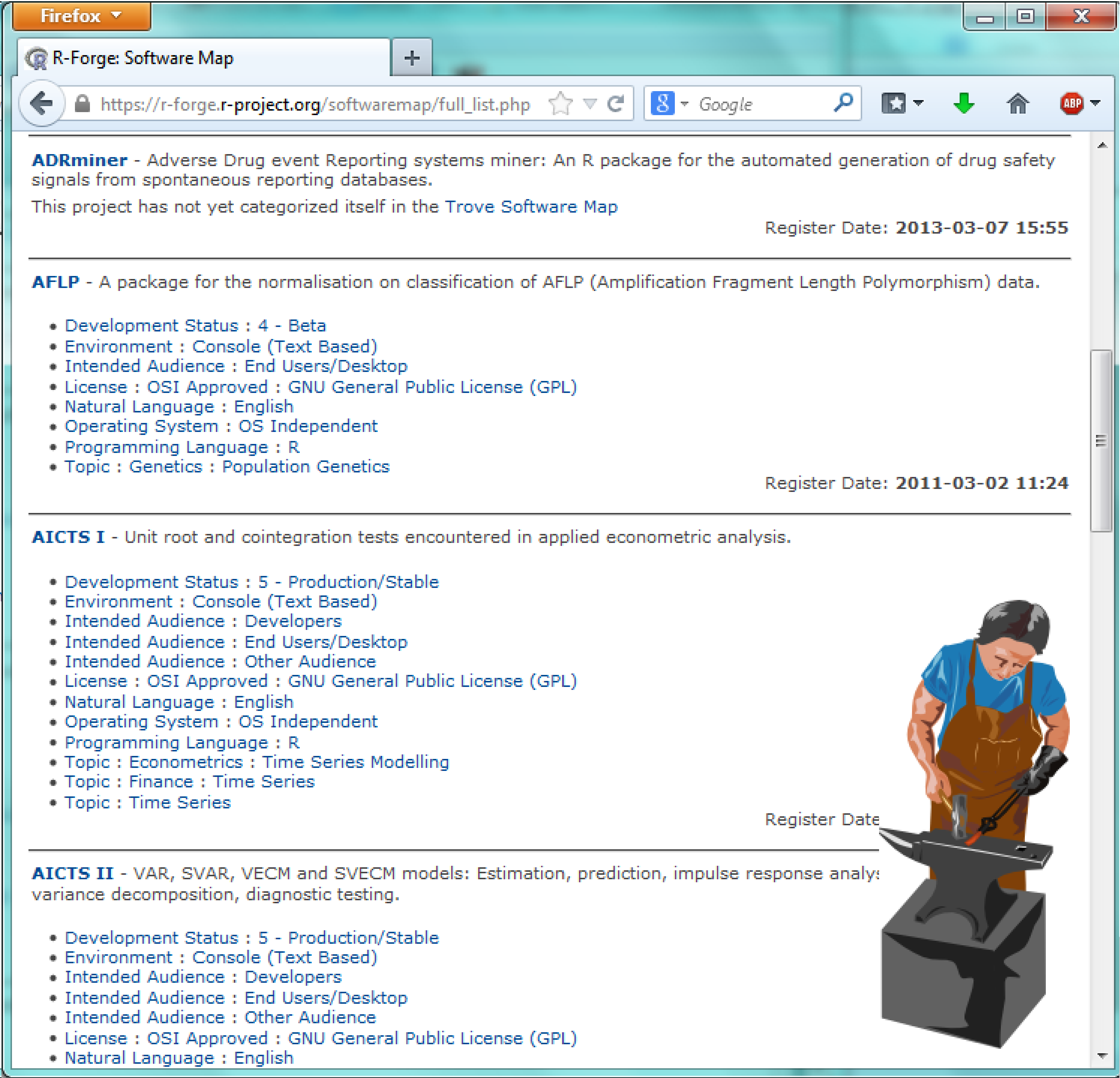
Software Map

Tag cloud | Project Tree | Project List

Project tree

Topic

- Bayesian Statistics *(53 projects)*
- Bioinformatics *(148 projects)*
- Biostatistics & Medical Statistics *(68 projects)*
- Chemoinformatics *(17 projects)*
- Cluster Analysis *(41 projects)*
- Computational Physics *(12 projects)*
- Connectivity *(20 projects)*
- Database *(27 projects)*
- Datasets *(28 projects)*
- Design of Experiments & Analysis of Experimen
- Econometrics *(63 projects)*
- Education *(27 projects)*
- Environmetrics *(44 projects)*
- Finance *(69 projects)*
- Genetics *(67 projects)*
- Graphical Models *(9 projects)*
- Graphical User Interface *(21 projects)*
- Graphics *(77 projects)*
- High Performance Computing *(28 projects)*
- Machine Learning *(71 projects)*
- Marketing & Business Analytics *(1 projects)*
- Multivariate Statistics *(138 projects)*
- Numerical Analysis *(18 projects)*
- Optimization *(56 projects)*
- Other/Nonlisted Topic *(75 projects)*
- Regression Models *(80 projects)*
- Robust Statistics *(22 projects)*
- Social Sciences *(76 projects)*
- Software Development *(54 projects)*
- Spatial Data & Statistics *(133 projects)*
- Text *(37 projects)*
- Time Series *(62 projects)*



ADRminer - Adverse Drug event Reporting systems miner: An R package for the automated generation of drug safety signals from spontaneous reporting databases.

This project has not yet categorized itself in the Trove Software Map

Register Date: **2013-03-07 15:55**

AFLP - A package for the normalisation on classification of AFLP (Amplification Fragment Length Polymorphism) data.

- Development Status : 4 - Beta
- Environment : Console (Text Based)
- Intended Audience : End Users/Desktop
- License : OSI Approved : GNU General Public License (GPL)
- Natural Language : English
- Operating System : OS Independent
- Programming Language : R
- Topic : Genetics : Population Genetics

Register Date: **2011-03-02 11:24**

AICTS I - Unit root and cointegration tests encountered in applied econometric analysis.

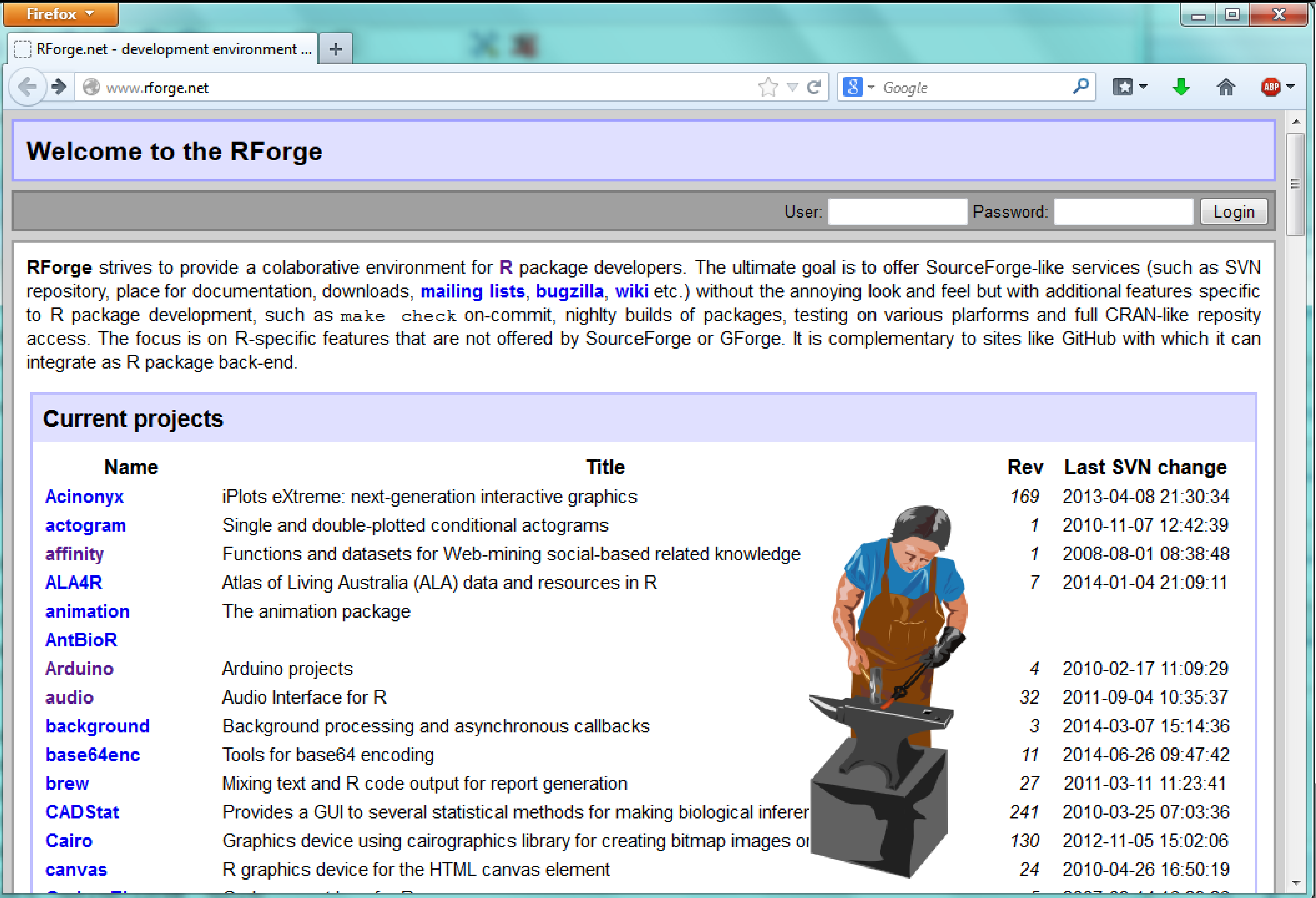
- Development Status : 5 - Production/Stable
- Environment : Console (Text Based)
- Intended Audience : Developers
- Intended Audience : End Users/Desktop
- Intended Audience : Other Audience
- License : OSI Approved : GNU General Public License (GPL)
- Natural Language : English
- Operating System : OS Independent
- Programming Language : R
- Topic : Econometrics : Time Series Modelling
- Topic : Finance : Time Series
- Topic : Time Series

Register Date

AICTS II - VAR, SVAR, VECM and SVECM models: Estimation, prediction, impulse response analy: variance decomposition, diagnostic testing.

- Development Status : 5 - Production/Stable
- Environment : Console (Text Based)
- Intended Audience : Developers
- Intended Audience : End Users/Desktop
- Intended Audience : Other Audience
- License : OSI Approved : GNU General Public License (GPL)
- Natural Language : English

The Origin Of Things - RForge



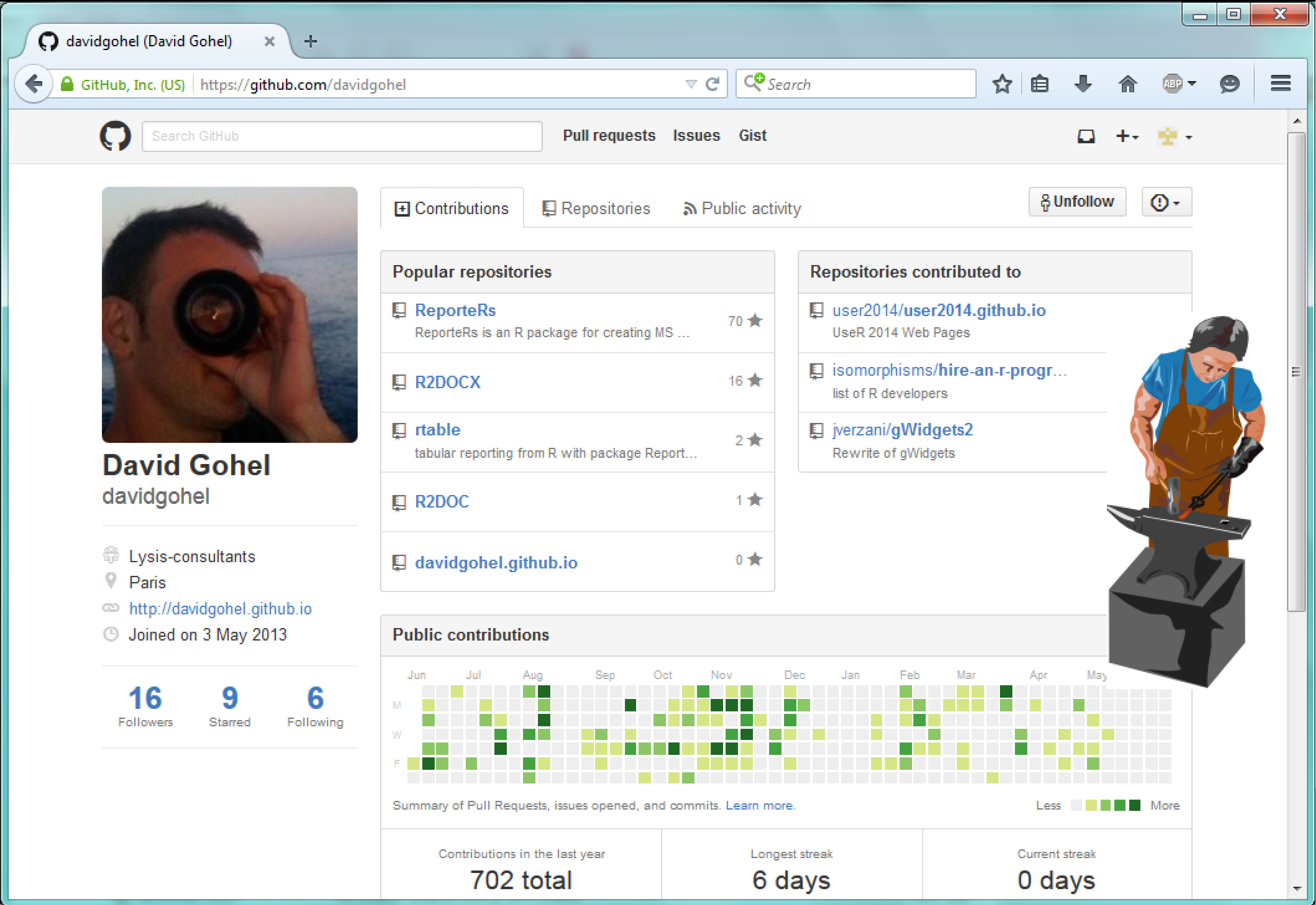
Welcome to the RForge

RForge strives to provide a colaborative environment for **R** package developers. The ultimate goal is to offer SourceForge-like services (such as SVN repository, place for documentation, downloads, **mailing lists**, **bugzilla**, **wiki** etc.) without the annoying look and feel but with additional features specific to R package development, such as `make check` on-commit, nighlty builds of packages, testing on various plarforms and full CRAN-like reposity access. The focus is on R-specific features that are not offered by SourceForge or GForge. It is complementary to sites like GitHub with which it can integrate as R package back-end.

Current projects

Name	Title	Rev	Last SVN change
Acinonyx	iPlots eXtreme: next-generation interactive graphics	*169*	2013-04-08 21:30:34
actogram	Single and double-plotted conditional actograms	*1*	2010-11-07 12:42:39
affinity	Functions and datasets for Web-mining social-based related knowledge	*1*	2008-08-01 08:38:48
ALA4R	Atlas of Living Australia (ALA) data and resources in R	*7*	2014-01-04 21:09:11
animation	The animation package		
AntBioR			
Arduino	Arduino projects	*4*	2010-02-17 11:09:29
audio	Audio Interface for R	*32*	2011-09-04 10:35:37
background	Background processing and asynchronous callbacks	*3*	2014-03-07 15:14:36
base64enc	Tools for base64 encoding	*11*	2014-06-26 09:47:42
brew	Mixing text and R code output for report generation	*27*	2011-03-11 11:23:41
CADStat	Provides a GUI to several statistical methods for making biological inferer	*241*	2010-03-25 07:03:36
Cairo	Graphics device using cairographics library for creating bitmap images or	*130*	2012-11-05 15:02:06
canvas	R graphics device for the HTML canvas element	*24*	2010-04-26 16:50:19

The Origin Of Things - GitHub



David Gohel
davidgohel

Lysis-consultants
Paris
http://davidgohel.github.io
Joined on 3 May 2013

16	9	6
Followers	Starred	Following

Popular repositories

Repository	Stars
ReporteRs ReporteRs is an R package for creating MS ...	70
R2DOCX	16
rtable tabular reporting from R with package Report...	2
R2DOC	1
davidgohel.github.io	0

Repositories contributed to

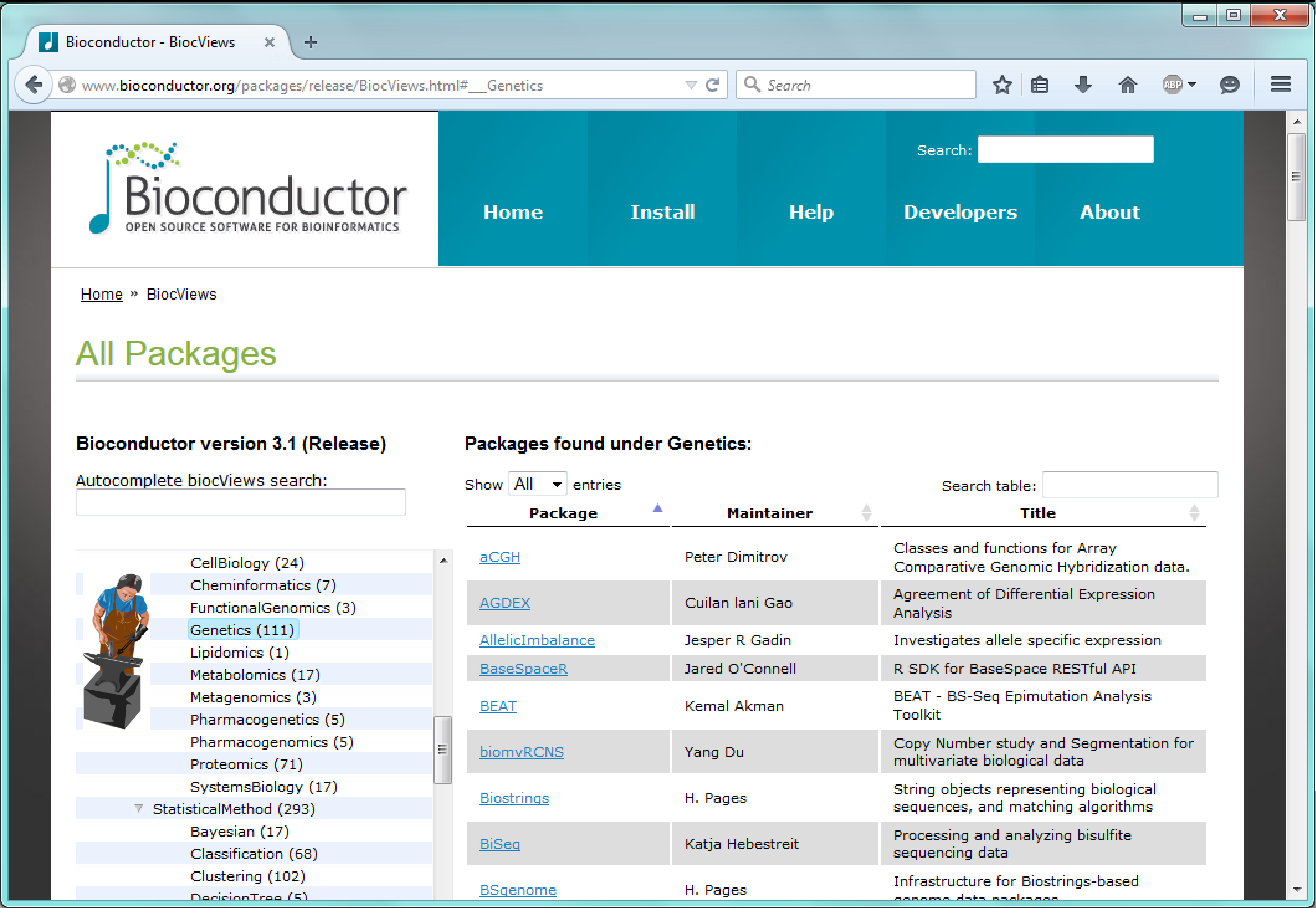
- user2014/user2014.github.io — UseR 2014 Web Pages
- isomorphisms/hire-an-r-progr... — list of R developers
- jverzani/gWidgets2 — Rewrite of gWidgets

Public contributions

Summary of Pull Requests, issues opened, and commits. Learn more.

Contributions in the last year	Longest streak	Current streak
702 total	6 days	0 days

The Origin Of Things - Bioconductor



Home » BiocViews

All Packages

Bioconductor version 3.1 (Release)

Autocomplete biocViews search:

- CellBiology (24)
- Cheminformatics (7)
- FunctionalGenomics (3)
- Genetics (111)
- Lipidomics (1)
- Metabolomics (17)
- Metagenomics (3)
- Pharmacogenetics (5)
- Pharmacogenomics (5)
- Proteomics (71)
- SystemsBiology (17)
- StatisticalMethod (293)
  - Bayesian (17)
  - Classification (68)
  - Clustering (102)
  - DecisionTree (5)

Packages found under Genetics:

Show All entries

Search table:

Package	Maintainer	Title
aCGH	Peter Dimitrov	Classes and functions for Array Comparative Genomic Hybridization data.
AGDEX	Cuilan lani Gao	Agreement of Differential Expression Analysis
AllelicImbalance	Jesper R Gadin	Investigates allele specific expression
BaseSpaceR	Jared O'Connell	R SDK for BaseSpace RESTful API
BEAT	Kemal Akman	BEAT - BS-Seq Epimutation Analysis Toolkit
biomvRCNS	Yang Du	Copy Number study and Segmentation for multivariate biological data
Biostrings	H. Pages	String objects representing biological sequences, and matching algorithms
BiSeq	Katja Hebestreit	Processing and analyzing bisulfite sequencing data
BSgenome	H. Pages	Infrastructure for Biostrings-based genome data packages

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

II 1/3 :) **R is easy to maintain!**

IV R is supported by the community

V R is supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

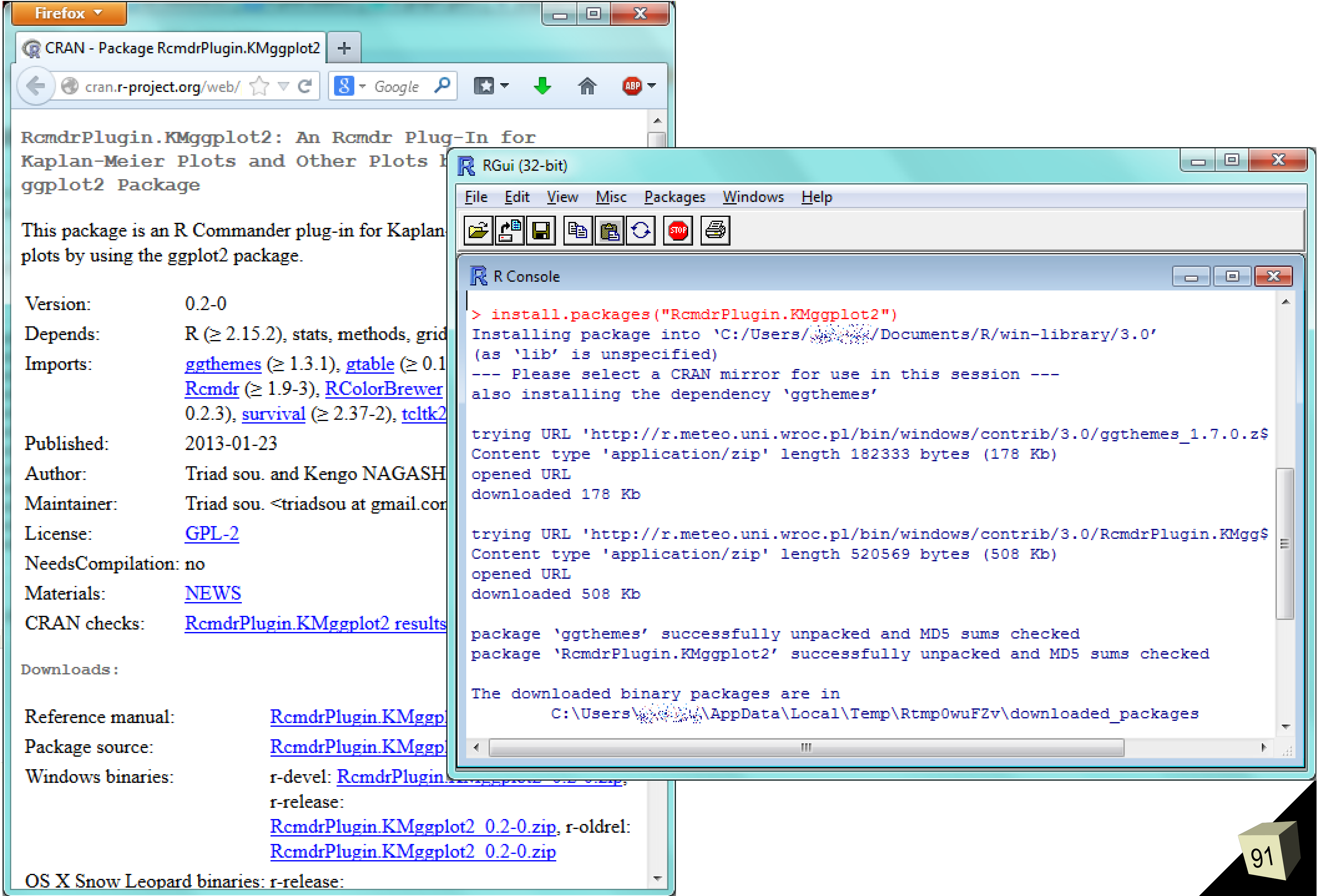
XII FDA accepted using R for drug trials!

R is easy to maintain!

R is easy to maintain. Managing installed packages and keeping them up-to-date becomes a piece of cake:

- Packages are available in CRAN, GitHub, BioConductor and Omegahat repositories. CRAN Repositories are mirrored in more than 50 countries.
- Dependencies between packages are resolved automatically. Just type install.packages(“package_name”) and R download necessary packages, validate MD5 and install them.
- Packages in a binary form can be installed from local .zip archives
- Packages can be also built from sources (common approach on Linux)
- Stay up-to-date with update.packages(). This could not be easier.
- By the use of installr package entire environment can be updated at once.
- One can have multiple libraries of packages in order to organize them

R is easy to maintain!



RcmdrPlugin.KMggplot2: An Rcmdr Plug-In for Kaplan-Meier Plots and Other Plots b
ggplot2 Package

This package is an R Commander plug-in for Kaplan
plots by using the ggplot2 package.

Version:	0.2-0
Depends:	R (≥ 2.15.2), stats, methods, grid
Imports:	ggthemes (≥ 1.3.1), gtable (≥ 0.1 Rcmdr (≥ 1.9-3), RColorBrewer 0.2.3), survival (≥ 2.37-2), tcltk2
Published:	2013-01-23
Author:	Triad sou. and Kengo NAGASH
Maintainer:	Triad sou. <triadsou at gmail.cor
License:	GPL-2
NeedsCompilation:	no
Materials:	NEWS
CRAN checks:	RcmdrPlugin.KMggplot2 results

Downloads:

Reference manual:	RcmdrPlugin.KMggp
Package source:	RcmdrPlugin.KMggp
Windows binaries:	r-devel: RcmdrPlugin.KMggplot2_0.2-0.zip, r-release: RcmdrPlugin.KMggplot2_0.2-0.zip, r-oldrel: RcmdrPlugin.KMggplot2_0.2-0.zip
OS X Snow Leopard binaries:	r-release:

```
> install.packages("RcmdrPlugin.KMggplot2")
Installing package into 'C:/Users/[redacted]/Documents/R/win-library/3.0'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
also installing the dependency 'ggthemes'

trying URL 'http://r.meteo.uni.wroc.pl/bin/windows/contrib/3.0/ggthemes_1.7.0.z$
Content type 'application/zip' length 182333 bytes (178 Kb)
opened URL
downloaded 178 Kb

trying URL 'http://r.meteo.uni.wroc.pl/bin/windows/contrib/3.0/RcmdrPlugin.KMgg$
Content type 'application/zip' length 520569 bytes (508 Kb)
opened URL
downloaded 508 Kb

package 'ggthemes' successfully unpacked and MD5 sums checked
package 'RcmdrPlugin.KMggplot2' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\[redacted]\AppData\Local\Temp\Rtmp0wuFZv\downloaded_packages
```

CRAN D.I.Y.

With the miniCRAN package one can build his own, private mini repository of R packages. This is a perfect solution for creating dedicated, in-house production environments for the following reasons:

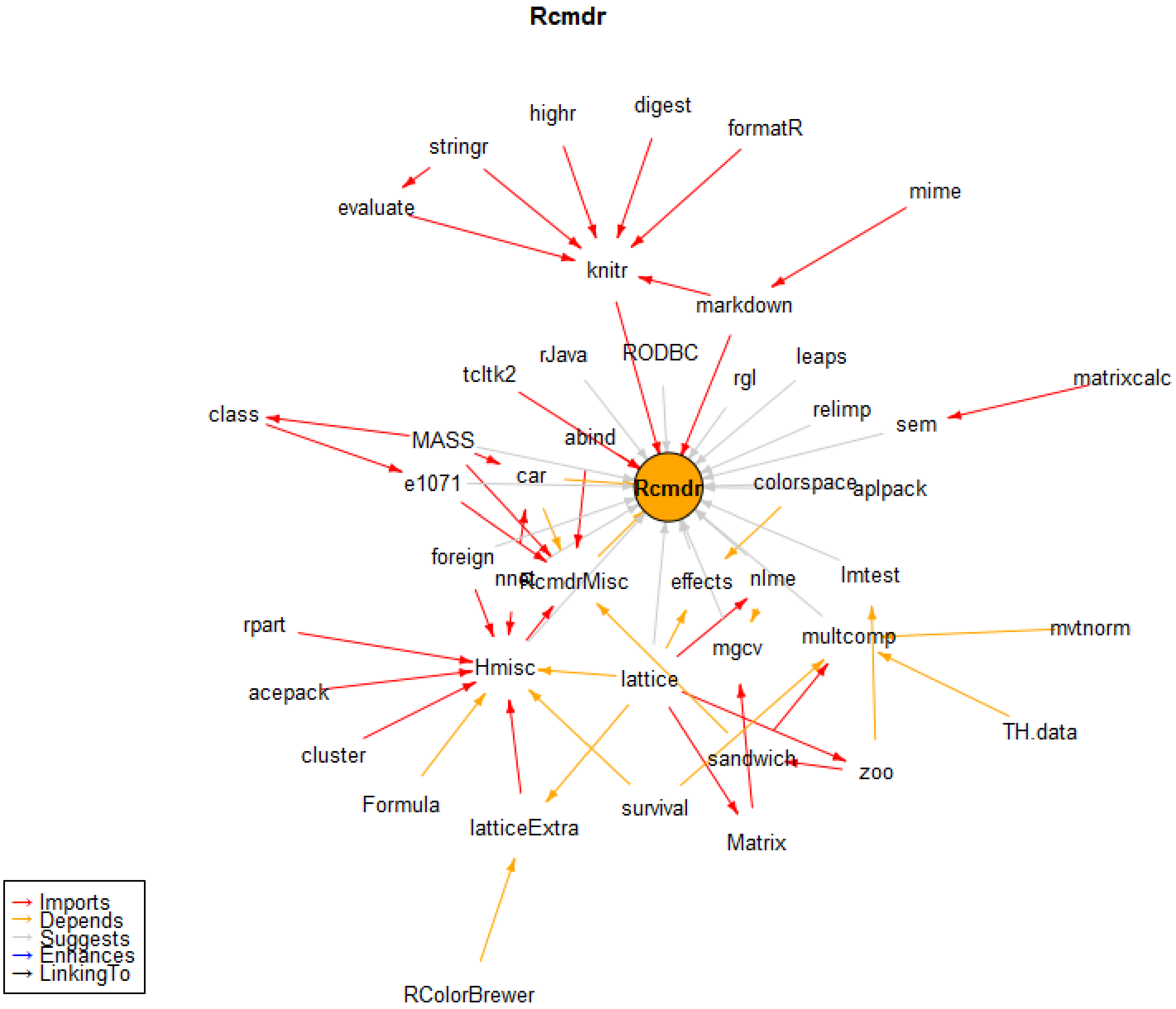
- *You may wish to mirror only a subset of CRAN, for security, legal compliance or any other in-house reason*
- *You may wish to restrict internal package use to a subset of public packages, to minimize package duplication, or other reasons of coding standards*
- *You may wish to make packages available from public repositories other than CRAN, e.g. BioConductor, r-forge, OmegaHat, etc.*
- *You may wish to add custom in-house packages to your repository*

www.r-bloggers.com/introducing-minicran-an-r-package-to-create-a-private-cran-repository/

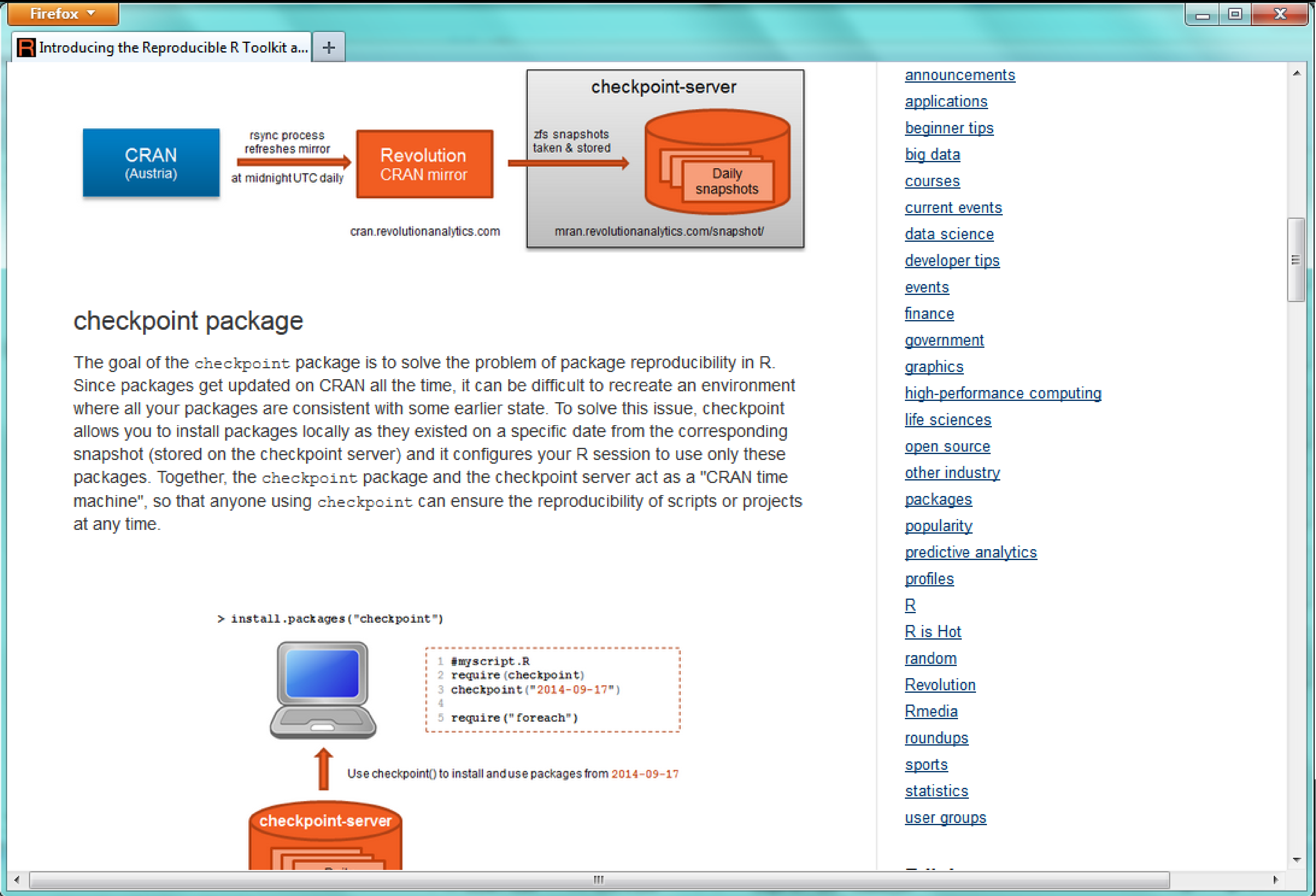
Do not forget to visit the quick introduction to miniCRAN.

CRAN D.I.Y.

MiniCRAN also helps you to track dependencies between packages.



Versioned CRAN – meet Revolution::*checkpoint*



checkpoint package

The goal of the `checkpoint` package is to solve the problem of package reproducibility in R. Since packages get updated on CRAN all the time, it can be difficult to recreate an environment where all your packages are consistent with some earlier state. To solve this issue, checkpoint allows you to install packages locally as they existed on a specific date from the corresponding snapshot (stored on the checkpoint server) and it configures your R session to use only these packages. Together, the `checkpoint` package and the checkpoint server act as a "CRAN time machine", so that anyone using `checkpoint` can ensure the reproducibility of scripts or projects at any time.

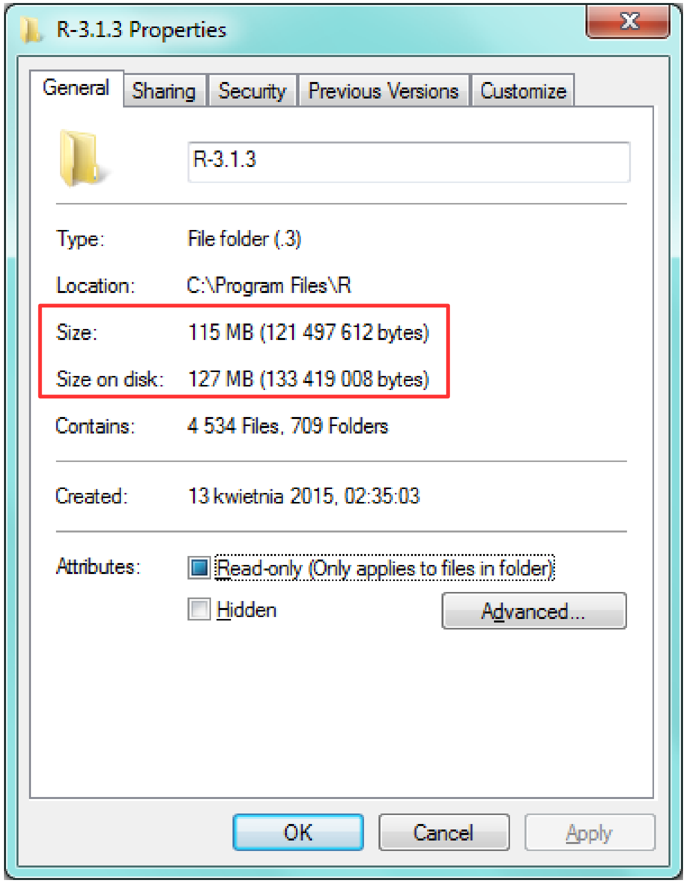
```
> install.packages("checkpoint")
```

```
#myscript.R
require(checkpoint)
checkpoint("2014-09-17")

require("foreach")
```

13 reasons why **you will** love GNU R

I	R is (extremely) cheap. In fact - it's free :)
II	R has (extremely) wide range of capabilities
II 2/3 :)	**R is not resource consuming!**
IV	R is supported by the community
V	R is supported by the business
VI	R is able to read data in many formats
VII	Interoperability is easy to achieve
VIII	R is truly cross-platform
IX	R offers numerous ways of presenting data
X	There are many options to optimize the code
XI	R is able to handle large amount of data
XII	R has a set of fancy tools and IDEs
XII	FDA accepted using R for drug trials!

R is not resouRce consuming!



R-3.1.3

Type: File folder (.3)

Location: C:\Program Files\R

Size: 115 MB (121 497 612 bytes)

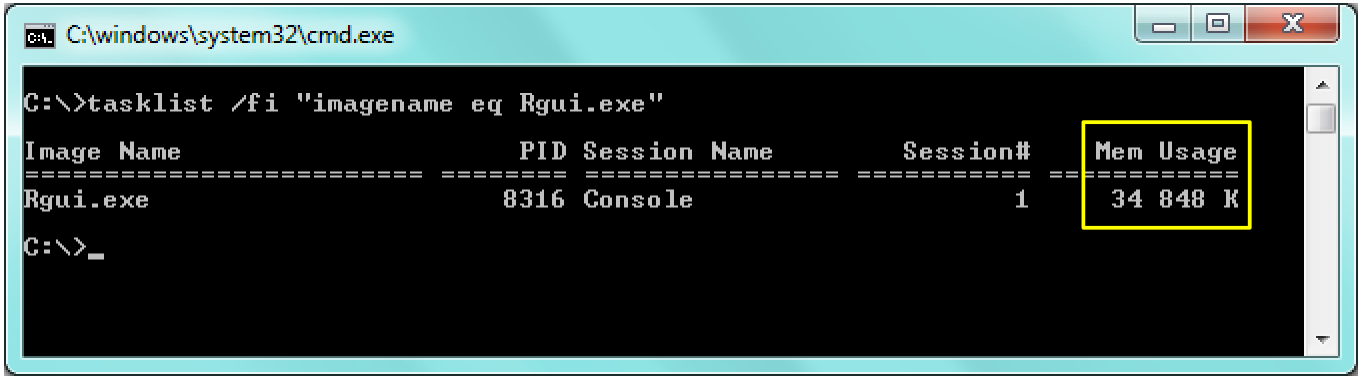
Size on disk: 127 MB (133 419 008 bytes)

Contains: 4 534 Files, 709 Folders

Created: 13 kwietnia 2015, 02:35:03

Attributes: ☑ Read-only (Only applies to files in folder)

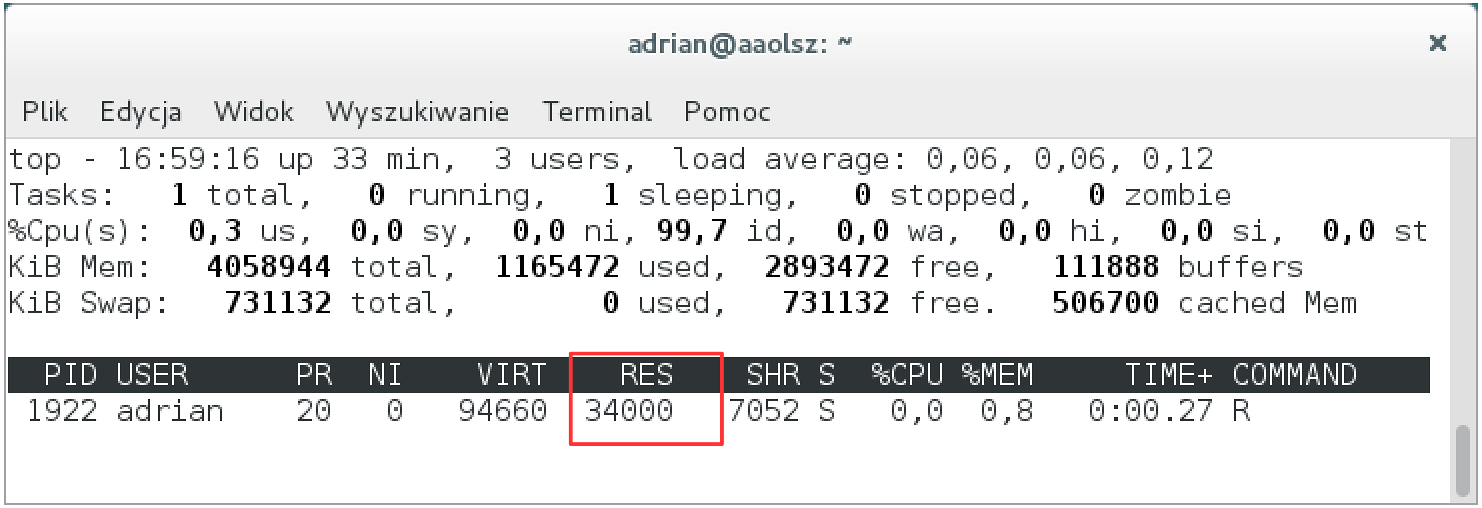
☐ Hidden



```
C:\>tasklist /fi "imagename eq Rgui.exe"

Image Name                     PID Session Name        Session#    Mem Usage
========================= ======== ================ =========== ============
Rgui.exe                      8316 Console                    1     34 848 K

C:\>_
```

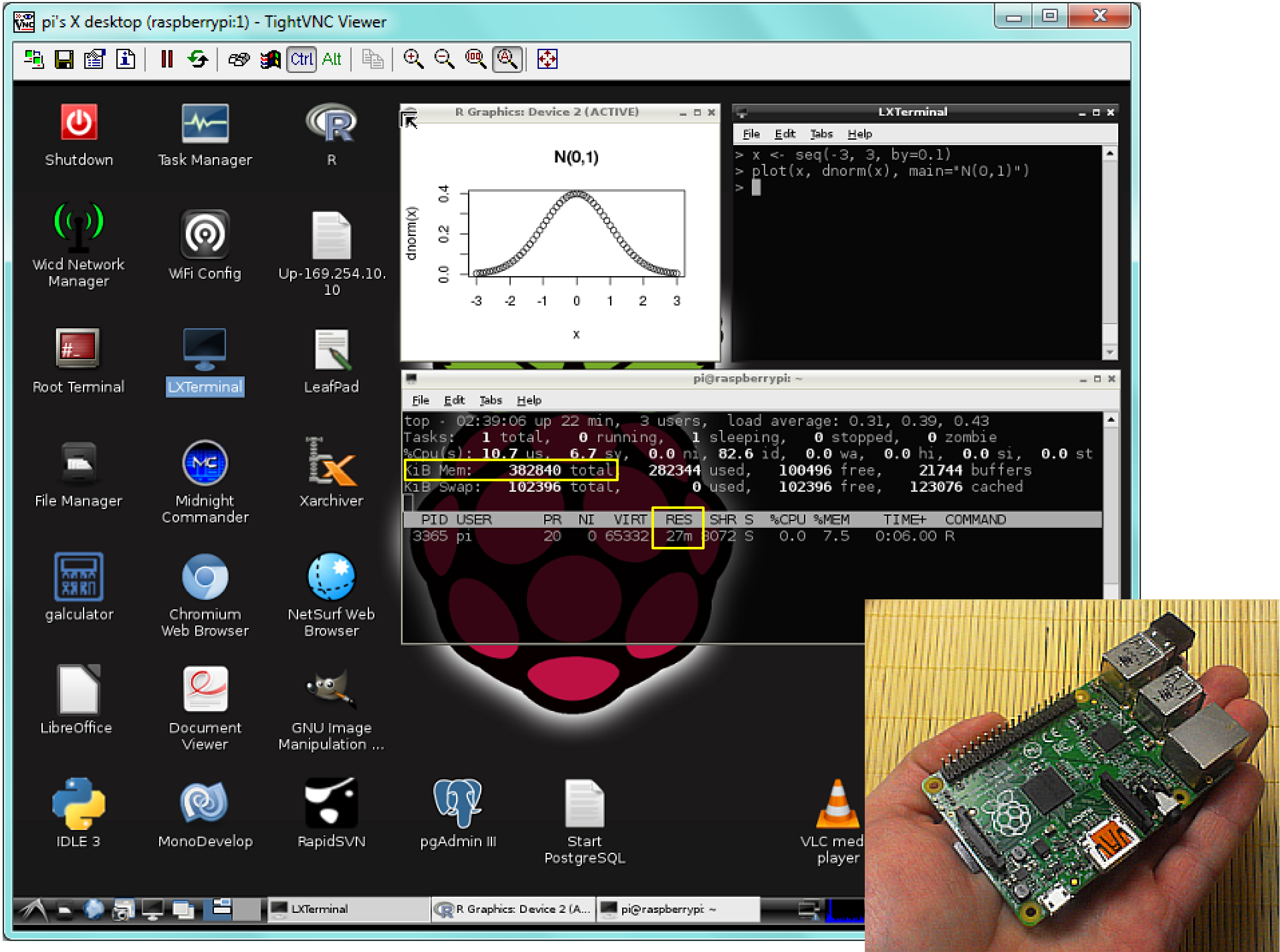
Fresh installation:

- Disk space: ~ 130MiB
- RAM: ~ 35MiB



```
top - 16:59:16 up 33 min,  3 users,  load average: 0,06, 0,06, 0,12
Tasks:   1 total,   0 running,   1 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0,3 us,  0,0 sy,  0,0 ni, 99,7 id,  0,0 wa,  0,0 hi,  0,0 si,  0,0 st
KiB Mem:   4058944 total,  1165472 used,  2893472 free,   111888 buffers
KiB Swap:   731132 total,        0 used,   731132 free.   506700 cached Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 1922 adrian    20   0   94660  34000   7052 S   0,0  0,8   0:00.27 R
```

R is not resouRce consuming!



```
> x <- seq(-3, 3, by=0.1)
> plot(x, dnorm(x), main="N(0,1)")
>
```

```
top - 02:39:06 up 22 min,  3 users,  load average: 0.31, 0.39, 0.43
Tasks:   1 total,   0 running,   1 sleeping,   0 stopped,   0 zombie
%Cpu(s): 10.7 us,  6.7 sy,  0.0 ni, 82.6 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem:    382840 total,   282344 used,   100496 free,    21744 buffers
KiB Swap:   102396 total,        0 used,   102396 free,   123076 cached

  PID USER      PR  NI  VIRT  RES  SHR S  %CPU %MEM    TIME+  COMMAND
 3365 pi        20   0 65332  27m 3072 S   0.0  7.5   0:06.00 R
```

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is supported by the world of science

IV R is supported by the community

V R is supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

...supported by the science

At the center of the R Community is the R Core Group of approximately 20 developers who maintain R and guide its evolution. They are experienced statisticians, well-known in the world of science, with significant achievements.

Each thematic section has its own academic supervisor.

Robert Gentleman's articles about R and Bioconductor are among the most cited in bioinformatics: over 5200 citations according to Google Scholar April 2014

There are numerous e-learning materials on the Internet which provide valuable assistance in data analysis with R. These are often provided by prestigious academic centers like Pricenton, Stanford or Hopkins.

The R project is closely linked with "the R Journal" which is a mine of scientific knowledge of using R for professional data analysis.

…supported by the science

The R Journal

RSS Feed
ISSN: 2073-4859

About The R Journal

The R Journal is the open access, refereed journal of the R project for statistical computing. It features short to medium length articles covering topics that might be of interest to users or developers of R, including

Add-on packages:	short introductions to R extension packages.
Programmer's Niche:	hints for programming in R.
Help Desk:	hints for newcomers explaining aspects of R that might not be so obvious from reading the manuals and FAQs.
Applications:	demonstrating how a new or existing technique can be applied in an area of current interest using R, providing a fresh view of such analyses in R that is of benefit beyond the specific application.

The R Journal intends to reach a wide audience and have a fast-track but thorough review process. Papers are expected to be reasonably short, clearly written, not too technical, and of course focused on R. Authors of refereed articles should take care to:

- put their contribution in context, in particular discuss related R functions or packages;
- explain the motivation for their contribution;
- provide code examples that are reproducible.

Continuing from *R News*, *The R Journal* also has a news section, including information on:

Changes in R:	new features of the latest release.

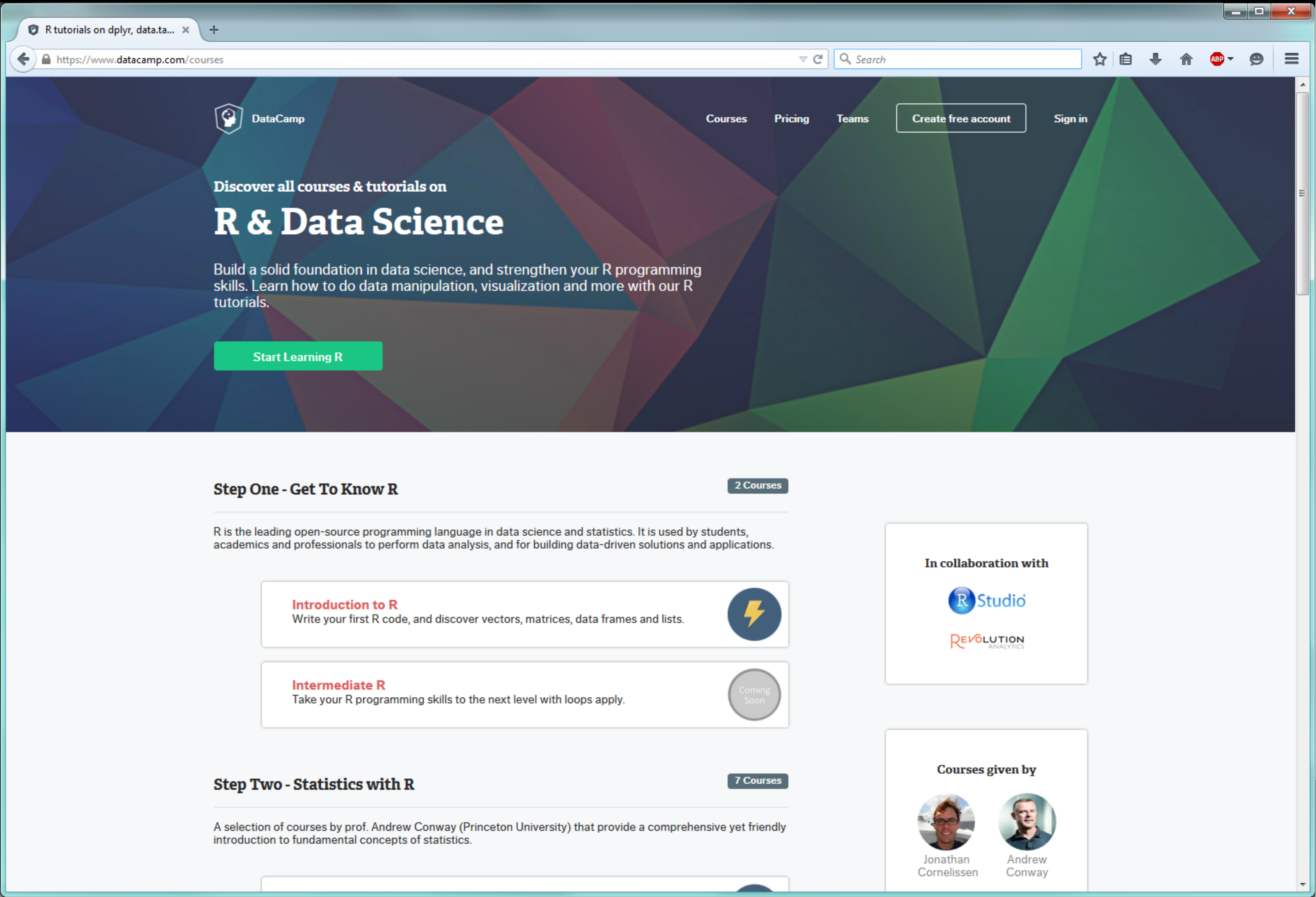
where to leaRn?

There are hundreds of places containing e-learning materials devoted to R.

Good places to start learning R:

1. http://cran.r-project.org/manuals.html
2. http://cran.r-project.org/other-docs.html
3. http://cran.r-project.org/doc/manuals/r-patched/R-intro.html
4. http://stats.stackexchange.com/questions/138/resources-for-learning-r
5. http://www.youtube.com/results?search_query=R+learn+statistics
6. http://www.statmethods.net - **Quick R**
7. http://adv-r.had.co.nz - **Advanced R** by Hadley Wickham
8. http://www.cookbook-r.com
9. http://rseek.org/?q=learn+R
10. http://www.google.pl/search?q=learn+R

Take a couRse!



Discover all courses & tutorials on

R & Data Science

Build a solid foundation in data science, and strengthen your R programming skills. Learn how to do data manipulation, visualization and more with our R tutorials.

Step One - Get To Know R

R is the leading open-source programming language in data science and statistics. It is used by students, academics and professionals to perform data analysis, and for building data-driven solutions and applications.

Introduction to R
Write your first R code, and discover vectors, matrices, data frames and lists.

Intermediate R
Take your R programming skills to the next level with loops apply.

Step Two - Statistics with R

A selection of courses by prof. Andrew Conway (Princeton University) that provide a comprehensive yet friendly introduction to fundamental concepts of statistics.

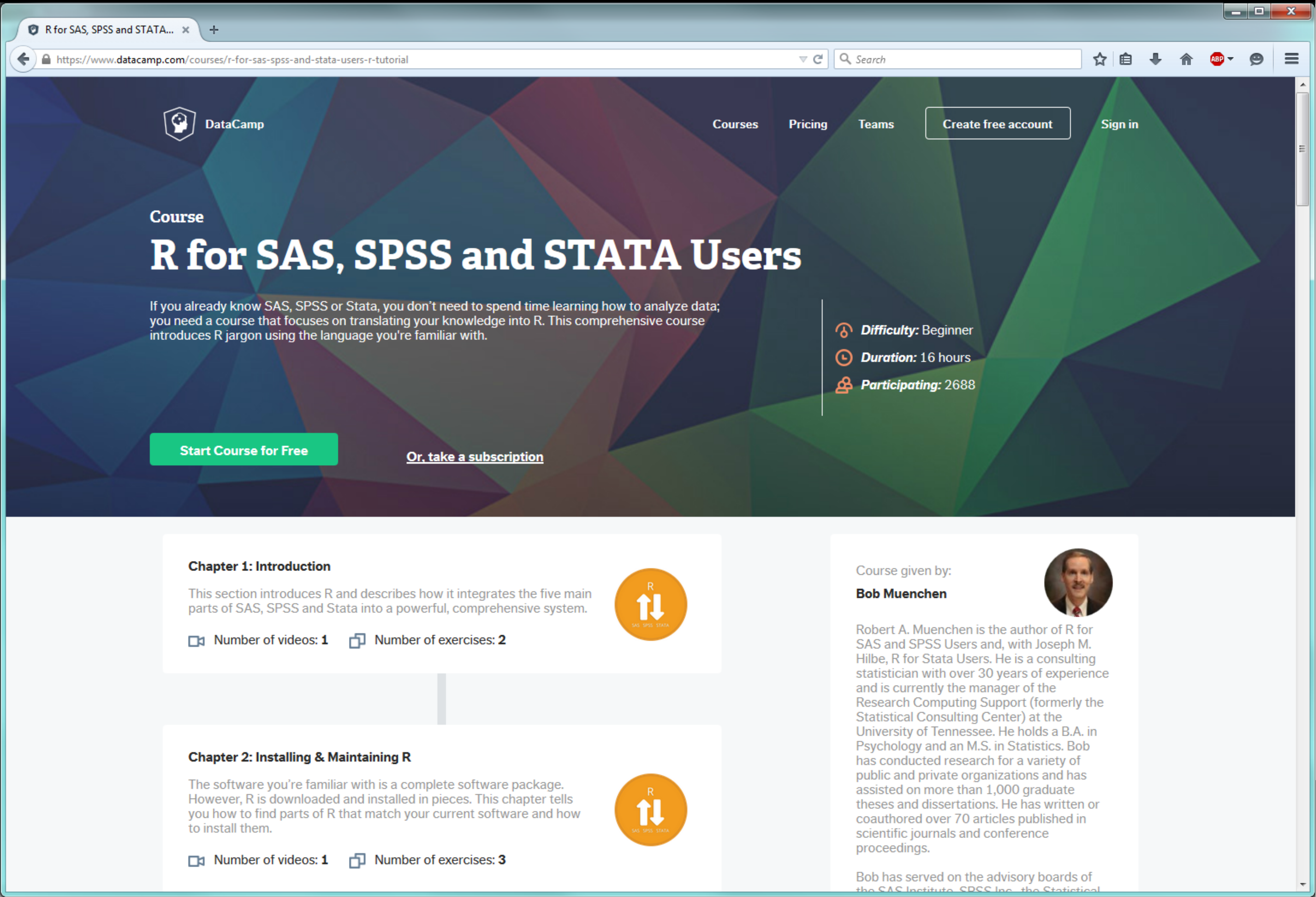
In collaboration with

Courses given by

Jonathan Cornelissen

Andrew Conway

Take a couRse!



Course

R for SAS, SPSS and STATA Users

If you already know SAS, SPSS or Stata, you don't need to spend time learning how to analyze data; you need a course that focuses on translating your knowledge into R. This comprehensive course introduces R jargon using the language you're familiar with.

Difficulty: Beginner
Duration: 16 hours
Participating: 2688

Chapter 1: Introduction

This section introduces R and describes how it integrates the five main parts of SAS, SPSS and Stata into a powerful, comprehensive system.

Number of videos: **1** Number of exercises: **2**

Chapter 2: Installing & Maintaining R

The software you're familiar with is a complete software package. However, R is downloaded and installed in pieces. This chapter tells you how to find parts of R that match your current software and how to install them.

Number of videos: **1** Number of exercises: **3**

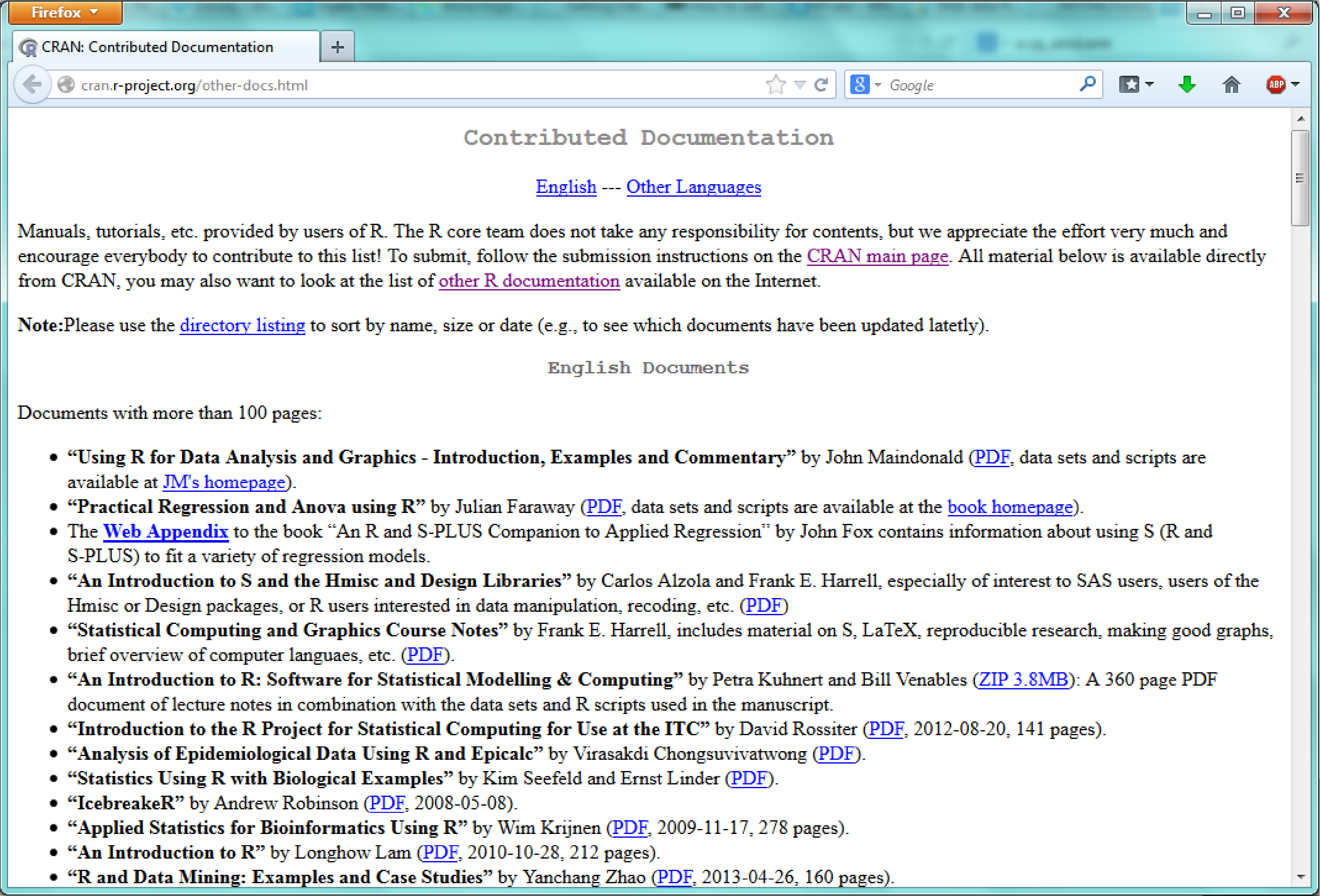
Course given by:

Bob Muenchen

Robert A. Muenchen is the author of R for SAS and SPSS Users and, with Joseph M. Hilbe, R for Stata Users. He is a consulting statistician with over 30 years of experience and is currently the manager of the Research Computing Support (formerly the Statistical Consulting Center) at the University of Tennessee. He holds a B.A. in Psychology and an M.S. in Statistics. Bob has conducted research for a variety of public and private organizations and has assisted on more than 1,000 graduate theses and dissertations. He has written or coauthored over 70 articles published in scientific journals and conference proceedings.

Bob has served on the advisory boards of

Look! This library contains pRiceless resources!



Contributed Documentation

English --- Other Languages

Manuals, tutorials, etc. provided by users of R. The R core team does not take any responsibility for contents, but we appreciate the effort very much and encourage everybody to contribute to this list! To submit, follow the submission instructions on the CRAN main page. All material below is available directly from CRAN, you may also want to look at the list of other R documentation available on the Internet.

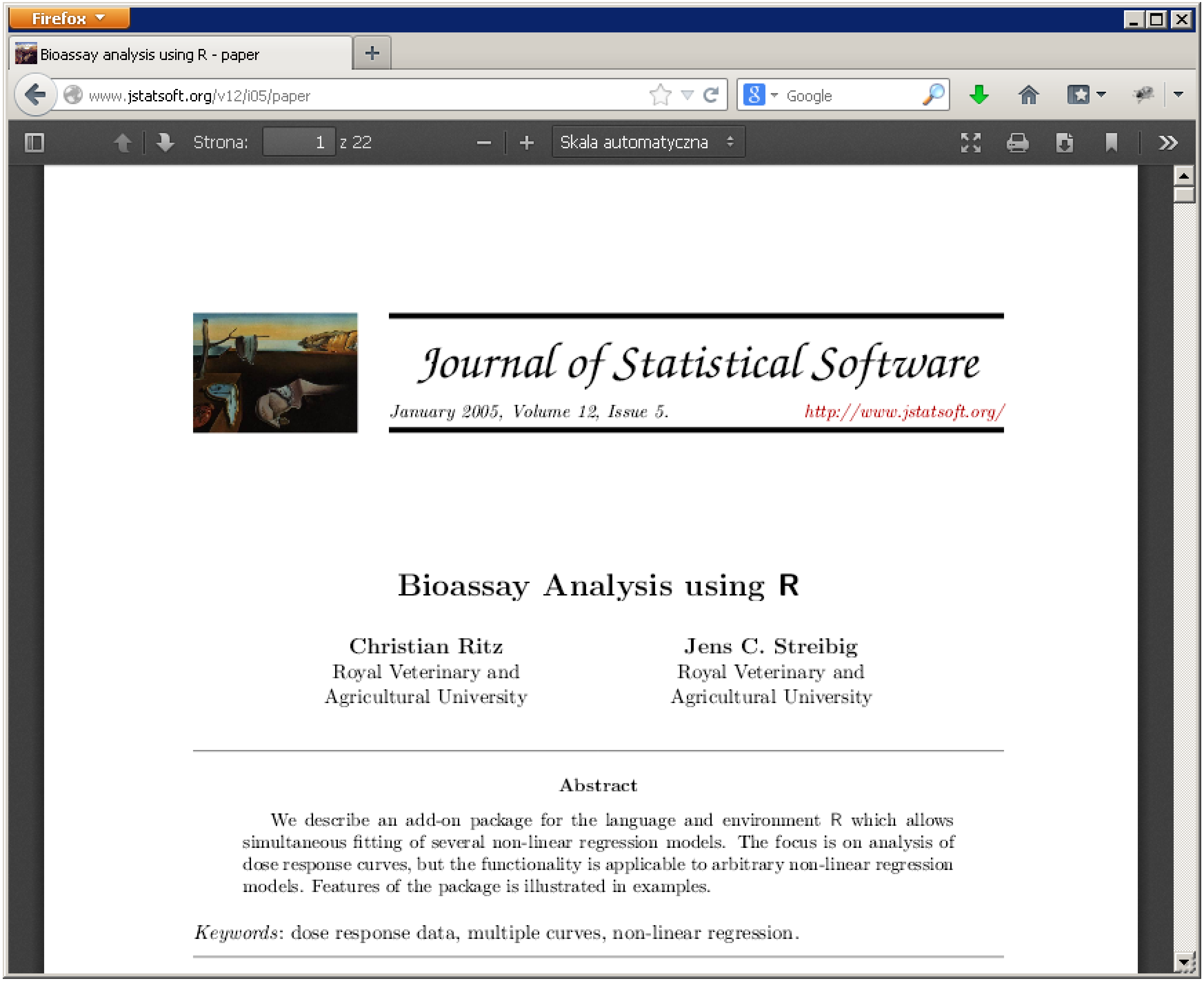
Note: Please use the directory listing to sort by name, size or date (e.g., to see which documents have been updated latetly).

English Documents

Documents with more than 100 pages:

- **“Using R for Data Analysis and Graphics - Introduction, Examples and Commentary”** by John Maindonald (PDF, data sets and scripts are available at JM's homepage).
- **“Practical Regression and Anova using R”** by Julian Faraway (PDF, data sets and scripts are available at the book homepage).
- The **Web Appendix** to the book “An R and S-PLUS Companion to Applied Regression” by John Fox contains information about using S (R and S-PLUS) to fit a variety of regression models.
- **“An Introduction to S and the Hmisc and Design Libraries”** by Carlos Alzola and Frank E. Harrell, especially of interest to SAS users, users of the Hmisc or Design packages, or R users interested in data manipulation, recoding, etc. (PDF)
- **“Statistical Computing and Graphics Course Notes”** by Frank E. Harrell, includes material on S, LaTeX, reproducible research, making good graphs, brief overview of computer languaes, etc. (PDF).
- **“An Introduction to R: Software for Statistical Modelling & Computing”** by Petra Kuhnert and Bill Venables (ZIP 3.8MB): A 360 page PDF document of lecture notes in combination with the data sets and R scripts used in the manuscript.
- **“Introduction to the R Project for Statistical Computing for Use at the ITC”** by David Rossiter (PDF, 2012-08-20, 141 pages).
- **“Analysis of Epidemiological Data Using R and Epicalc”** by Virasakdi Chongsuvivatwong (PDF).
- **“Statistics Using R with Biological Examples”** by Kim Seefeld and Ernst Linder (PDF).
- **“IcebreakeR”** by Andrew Robinson (PDF, 2008-05-08).
- **“Applied Statistics for Bioinformatics Using R”** by Wim Krijnen (PDF, 2009-11-17, 278 pages).
- **“An Introduction to R”** by Longhow Lam (PDF, 2010-10-28, 212 pages).
- **“R and Data Mining: Examples and Case Studies”** by Yanchang Zhao (PDF, 2013-04-26, 160 pages).

...thousands of valuable papeRs at our fingertips



Journal of Statistical Software

January 2005, Volume 12, Issue 5. http://www.jstatsoft.org/

Bioassay Analysis using R

Christian Ritz
Royal Veterinary and
Agricultural University

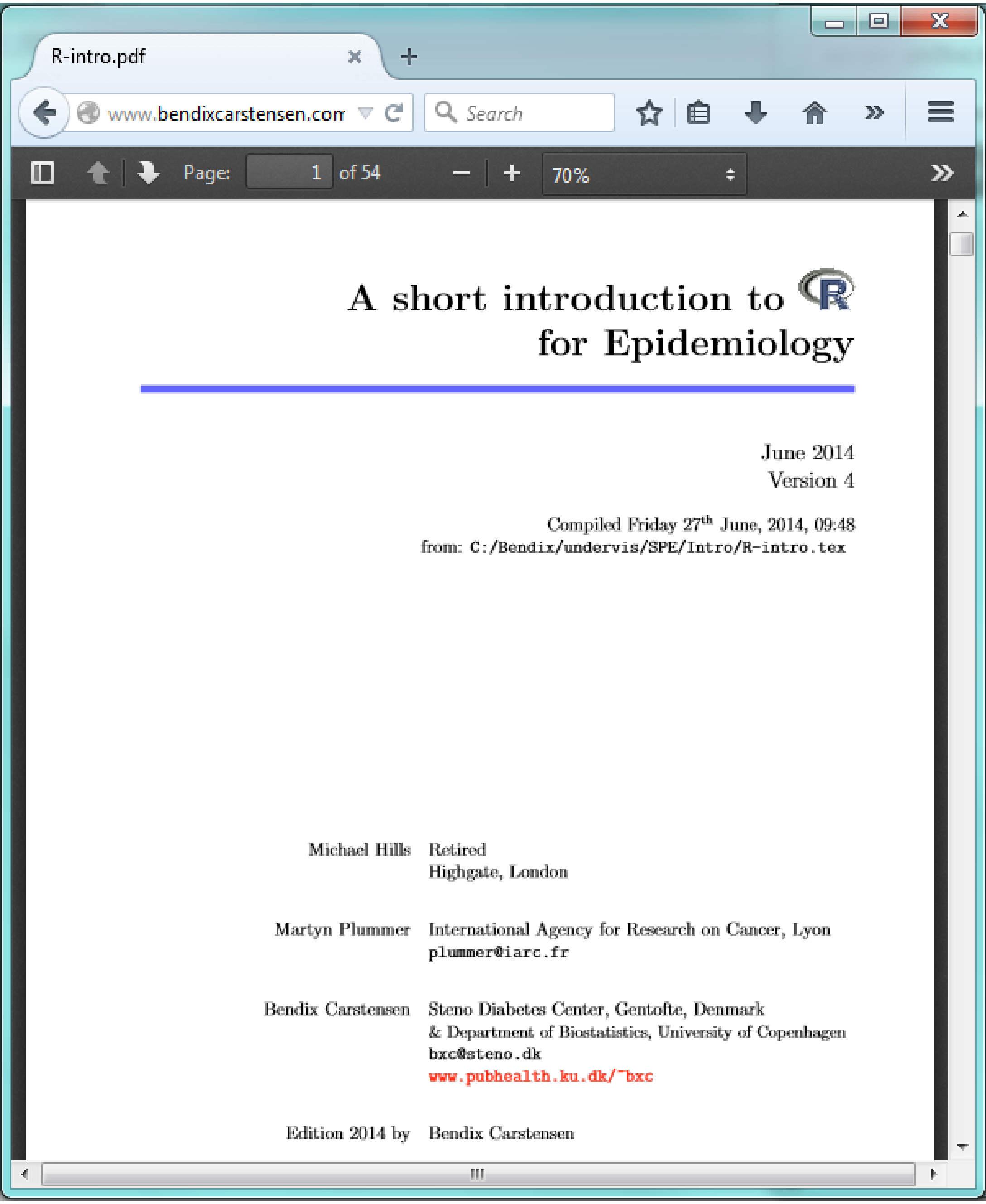
Jens C. Streibig
Royal Veterinary and
Agricultural University

Abstract

We describe an add-on package for the language and environment R which allows simultaneous fitting of several non-linear regression models. The focus is on analysis of dose response curves, but the functionality is applicable to arbitrary non-linear regression models. Features of the package is illustrated in examples.

Keywords: dose response data, multiple curves, non-linear regression.

...thousands of valuable papeRs at our fingertips



A short introduction to R
for Epidemiology

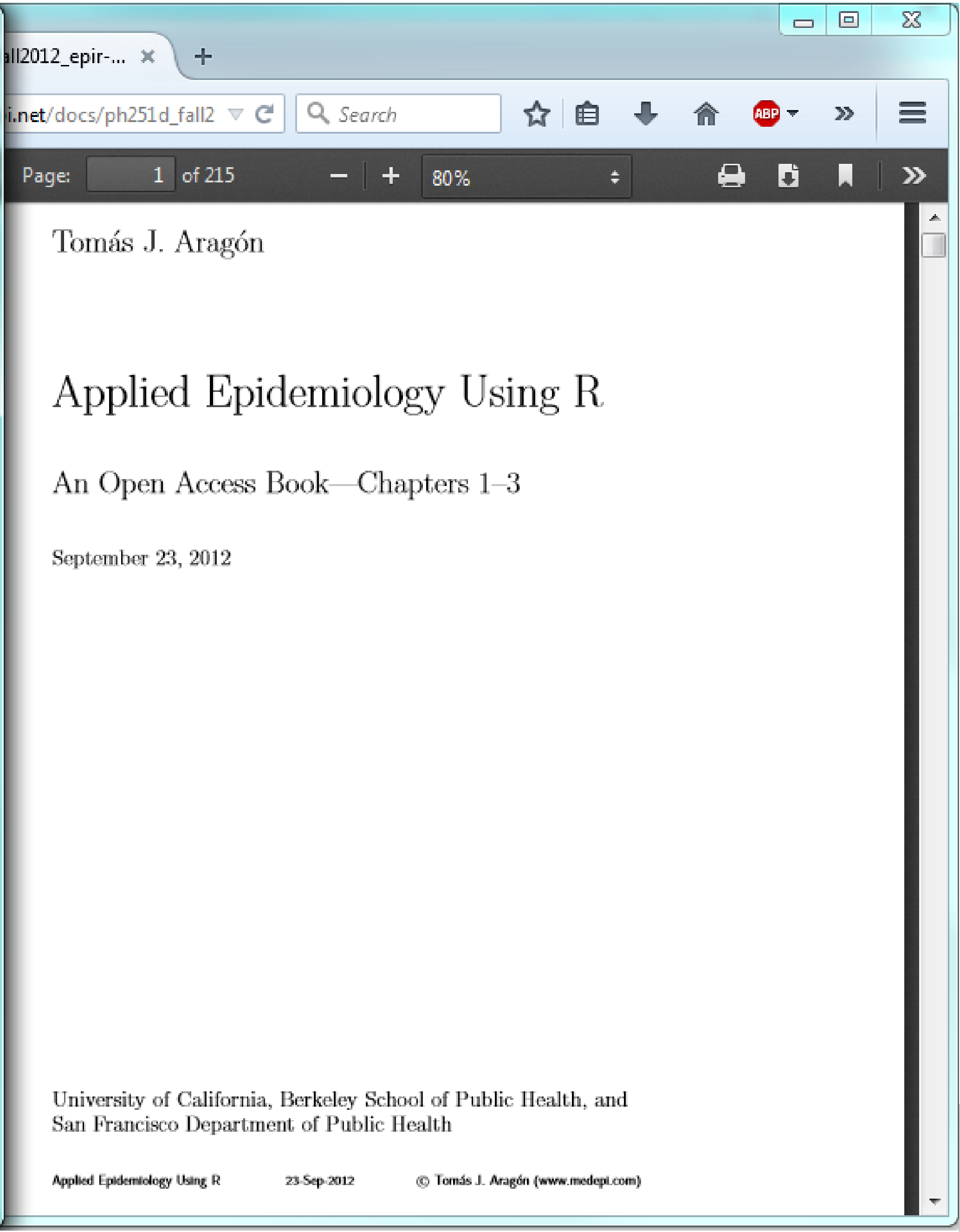
June 2014
Version 4

Compiled Friday 27th June, 2014, 09:48
from: `C:/Bendix/undervis/SPE/Intro/R-intro.tex`

Michael Hills	Retired Highgate, London
Martyn Plummer	International Agency for Research on Cancer, Lyon `plummer@iarc.fr`

Bendix Carstensen	Steno Diabetes Center, Gentofte, Denmark
	& Department of Biostatistics, University of Copenhagen
	`bxc@steno.dk`
	`www.pubhealth.ku.dk/~bxc`

Edition 2014 by	Bendix Carstensen



Tomás J. Aragón

Applied Epidemiology Using R

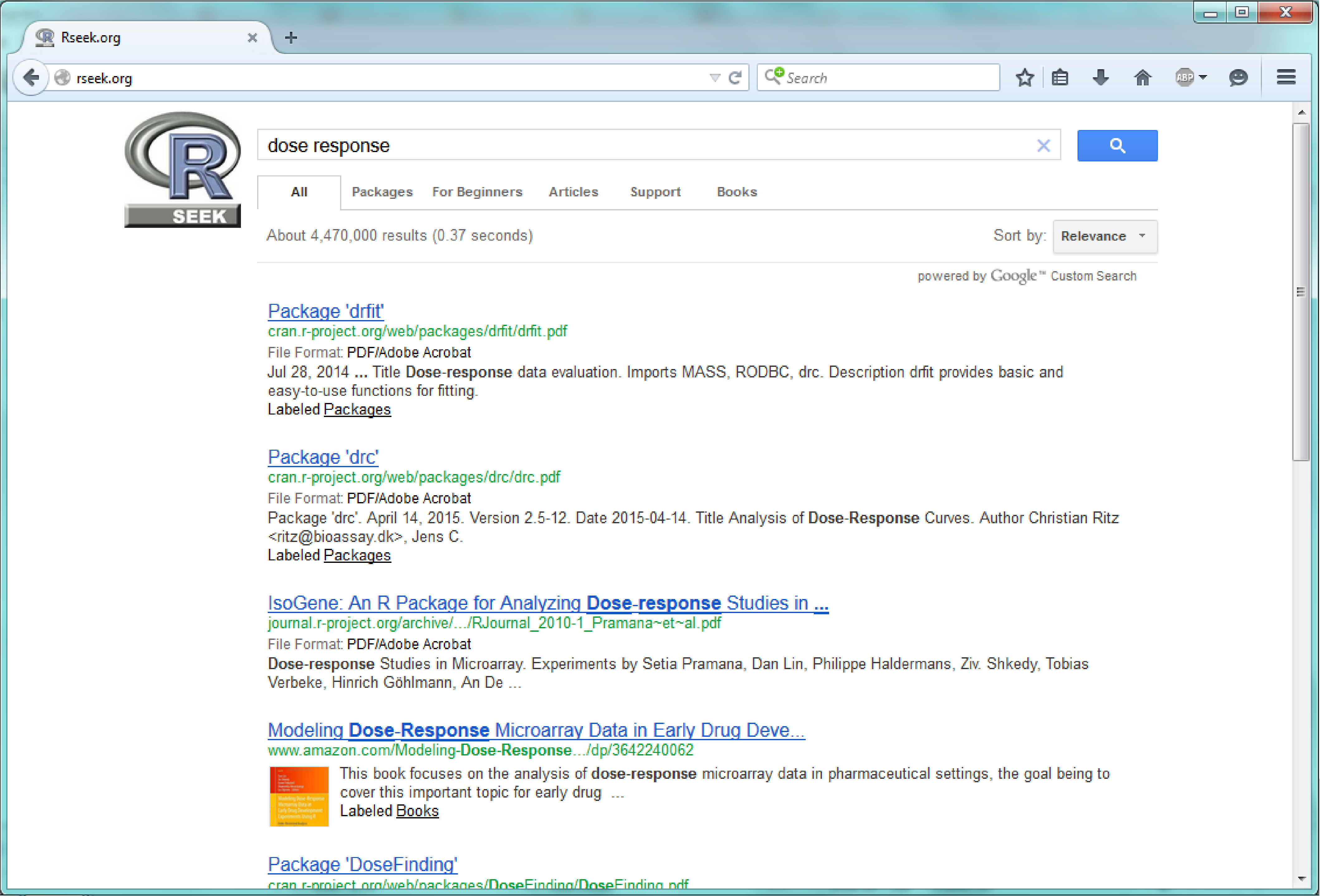
An Open Access Book—Chapters 1–3

September 23, 2012

University of California, Berkeley School of Public Health, and
San Francisco Department of Public Health

Applied Epidemiology Using R 23-Sep-2012 © Tomás J. Aragón (www.medepi.com)

Where to seaRch? → Rseek.org



dose response

About 4,470,000 results (0.37 seconds)

[Package 'drfit'](#)
cran.r-project.org/web/packages/drfit/drfit.pdf
File Format: PDF/Adobe Acrobat
Jul 28, 2014 **...** Title **Dose**-**response** data evaluation. Imports MASS, RODBC, drc. Description drfit provides basic and easy-to-use functions for fitting.
Labeled Packages

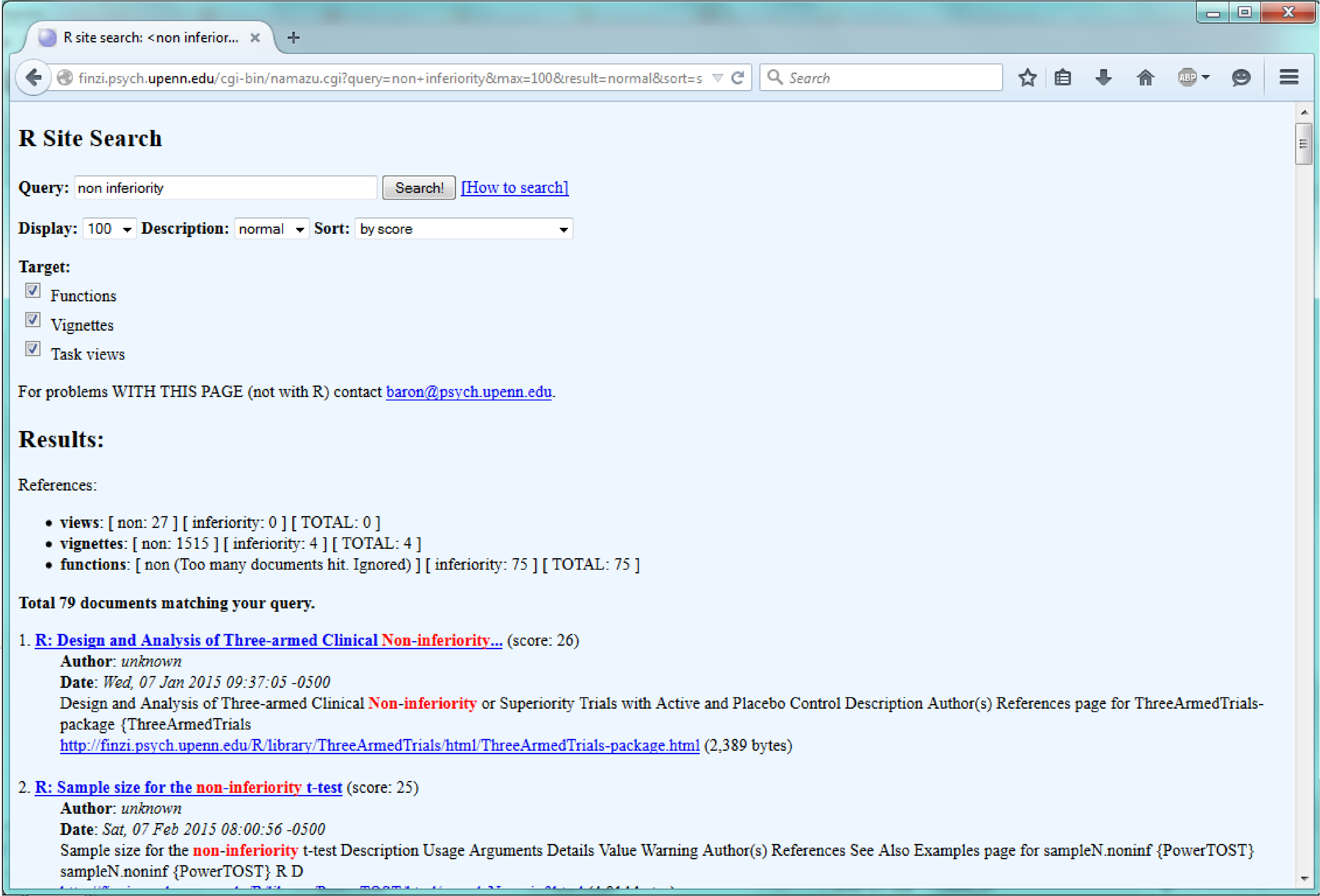
[Package 'drc'](#)
cran.r-project.org/web/packages/drc/drc.pdf
File Format: PDF/Adobe Acrobat
Package 'drc'. April 14, 2015. Version 2.5-12. Date 2015-04-14. Title Analysis of **Dose**-**Response** Curves. Author Christian Ritz <ritz@bioassay.dk>, Jens C.
Labeled Packages

[IsoGene: An R Package for Analyzing **Dose**-**response** Studies in **...**](#)
journal.r-project.org/archive/.../RJournal_2010-1_Pramana~et~al.pdf
File Format: PDF/Adobe Acrobat
Dose-**response** Studies in Microarray. Experiments by Setia Pramana, Dan Lin, Philippe Haldermans, Ziv. Shkedy, Tobias Verbeke, Hinrich Göhlmann, An De ...

[Modeling **Dose**-**Response** Microarray Data in Early Drug Deve...](#)
www.amazon.com/Modeling-**Dose**-**Response**.../dp/3642240062
This book focuses on the analysis of **dose**-**response** microarray data in pharmaceutical settings, the goal being to cover this important topic for early drug ...
Labeled Books

[Package 'DoseFinding'](#)
cran.r-project.org/web/packages/DoseFinding/DoseFinding.pdf

Where to seaRch? → R Site Search



R Site Search

Query: non inferiority Search! [How to search]

Display: 100 **Description:** normal **Sort:** by score

Target:

☑ Functions

☑ Vignettes

☑ Task views

For problems WITH THIS PAGE (not with R) contact baron@psych.upenn.edu.

Results:

References:

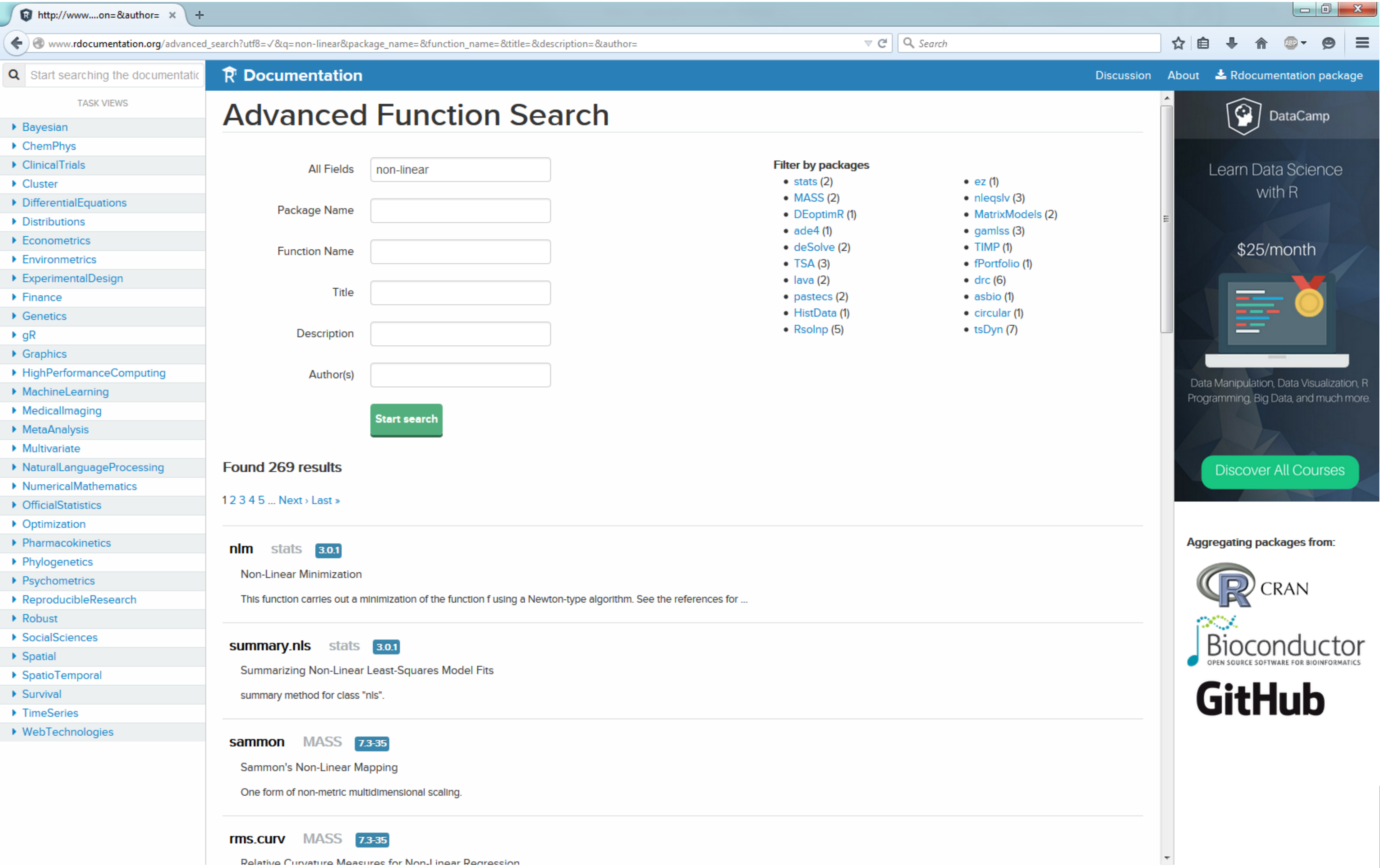
- **views**: [non: 27] [inferiority: 0] [TOTAL: 0]
- **vignettes**: [non: 1515] [inferiority: 4] [TOTAL: 4]
- **functions**: [non (Too many documents hit. Ignored)] [inferiority: 75] [TOTAL: 75]

Total 79 documents matching your query.

1. R: Design and Analysis of Three-armed Clinical **Non-inferiority...** (score: 26)
   Author: *unknown*
   Date: *Wed, 07 Jan 2015 09:37:05 -0500*
   Design and Analysis of Three-armed Clinical **Non-inferiority** or Superiority Trials with Active and Placebo Control Description Author(s) References page for ThreeArmedTrials-package {ThreeArmedTrials
   http://finzi.psych.upenn.edu/R/library/ThreeArmedTrials/html/ThreeArmedTrials-package.html (2,389 bytes)

2. R: Sample size for the **non-inferiority** t-test (score: 25)
   Author: *unknown*
   Date: *Sat, 07 Feb 2015 08:00:56 -0500*
   Sample size for the **non-inferiority** t-test Description Usage Arguments Details Value Warning Author(s) References See Also Examples page for sampleN.noninf {PowerTOST} sampleN.noninf {PowerTOST} R D

Where to seaRch? → R Documentation



TASK VIEWS

- Bayesian
- ChemPhys
- ClinicalTrials
- Cluster
- DifferentialEquations
- Distributions
- Econometrics
- Environmetrics
- ExperimentalDesign
- Finance
- Genetics
- gR
- Graphics
- HighPerformanceComputing
- MachineLearning
- MedicalImaging
- MetaAnalysis
- Multivariate
- NaturalLanguageProcessing
- NumericalMathematics
- OfficialStatistics
- Optimization
- Pharmacokinetics
- Phylogenetics
- Psychometrics
- ReproducibleResearch
- Robust
- SocialSciences
- Spatial
- SpatioTemporal
- Survival
- TimeSeries
- WebTechnologies

Advanced Function Search

All Fields: non-linear
Package Name:
Function Name:
Title:
Description:
Author(s):

Start search

Filter by packages

- stats (2)
- MASS (2)
- DEoptimR (1)
- ade4 (1)
- deSolve (2)
- TSA (3)
- lava (2)
- pastecs (2)
- HistData (1)
- Rsolnp (5)
- ez (1)
- nleqslv (3)
- MatrixModels (2)
- gamlss (3)
- TIMP (1)
- fPortfolio (1)
- drc (6)
- asbio (1)
- circular (1)
- tsDyn (7)

Found 269 results

1 2 3 4 5 ... Next › Last »

nlm stats 3.0.1

Non-Linear Minimization

This function carries out a minimization of the function f using a Newton-type algorithm. See the references for ...

summary.nls stats 3.0.1

Summarizing Non-Linear Least-Squares Model Fits

summary method for class "nls".

sammon MASS 7.3-35

Sammon's Non-Linear Mapping

One form of non-metric multidimensional scaling.

rms.curv MASS 7.3-35

Relative Curvature Measures for Non-Linear Regression

Aggregating packages from:

CRAN
Bioconductor
GitHub

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is supported by the world of science

IV R is supported by the community

V R is supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

...supported by the community

R is a global community of more than 2 million and still growing! users (2012, Oracle) and developers who voluntarily contribute their time and technical expertise to maintain, support, test and extend the R language and its environment, tools and infrastructure (e.g. CRAN). Among them are experienced statisticians, often employed in the well-known pharmaceutical companies like *Merck* or *Amgen*.

Hundreds of bloggers maintain their webpages, take active part in communities both online and offline through hundreds of forums and mailing lists, building knowledge base (visit rseek.org). UseRs also organize meetings and conferences.

There are dozens of valuable books written both by academics, researchers and "regular" R users. These books are issued by the prestigious publishing houses like Springer Verlag and Wiley & Sons.

...supported by the community

The size of the R user community (diffcult to define precisely, because there are no sales transactions, but conservatively estimated as being in the tens of thousands, with some independent estimates in the hundreds of thousands), provides for extensive review of source code and testing in "real world" settings outside the connes of the formalized testing performed by R Core.

This is a key distinction, related to product quality, between R and similar software that is only available to end users in a binary, executable format. In conjunction with detailed documentation and references provided to end users, the size of the R user community, all having full access to the source code, enables a superior ability to anticipate and verify R's performance and the results produced by R.

http://www.r-project.org/doc/R-FDA.pdf

…supported by the community

PAZUR

Poznański Akademicki Zlot Użytkowników R

Organizatorzy:

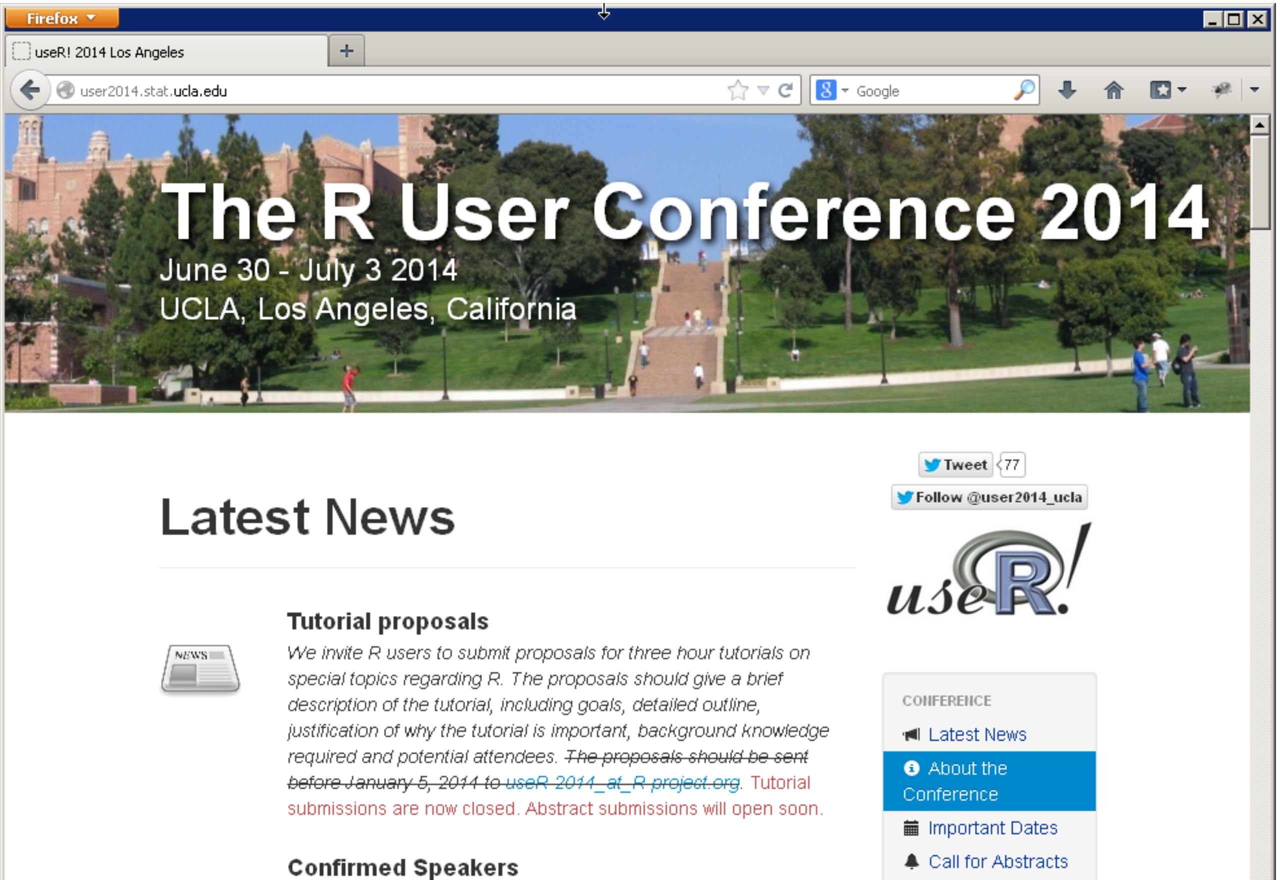
Uniwersytet Przyrodniczy w Poznaniu

ESTYMATOR
STUDENCKIE KOŁO NAUKOWE
Uniwersytet Ekonomiczny
w Poznaniu

Sponsor:

REVOLUTION ANALYTICS

13 grudnia (piątek) o 18:00
Zoo Coworking
Zwierzyniecka 20, II piętro
Szczegóły:
thinking-in-r.blogspot.com



The R User Conference 2014

June 30 - July 3 2014
UCLA, Los Angeles, California

Latest News

Tutorial proposals

We invite R users to submit proposals for three hour tutorials on special topics regarding R. The proposals should give a brief description of the tutorial, including goals, detailed outline, justification of why the tutorial is important, background knowledge required and potential attendees. ~~*The proposals should be sent before January 5, 2014 to useR-2014_at_R-project.org*~~. Tutorial submissions are now closed. Abstract submissions will open soon.

Confirmed Speakers

EARL 2015

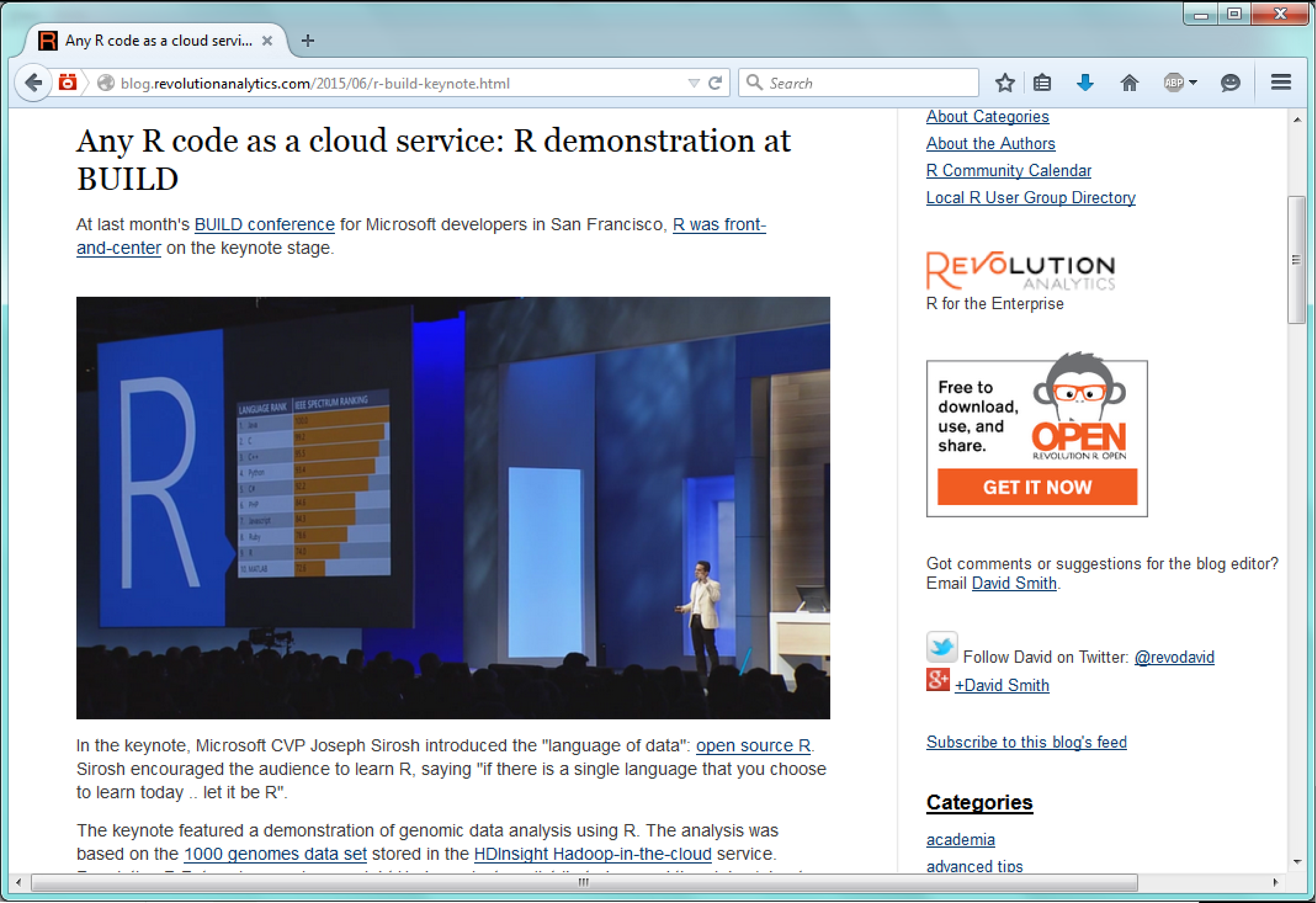
EFFECTIVE APPLICATIONS OF THE R LANGUAGE

LONDON 14 - 16 SEPTEMBER

@earlconf

R at Microsoft's BUILD 2015 conference

say "wow!"



Any R code as a cloud service: R demonstration at BUILD

At last month's BUILD conference for Microsoft developers in San Francisco, R was front-and-center on the keynote stage.

In the keynote, Microsoft CVP Joseph Sirosh introduced the "language of data": open source R. Sirosh encouraged the audience to learn R, saying "if there is a single language that you choose to learn today .. let it be R".

The keynote featured a demonstration of genomic data analysis using R. The analysis was based on the 1000 genomes data set stored in the HDInsight Hadoop-in-the-cloud service.

About Categories
About the Authors
R Community Calendar
Local R User Group Directory

REVOLUTION ANALYTICS
R for the Enterprise

Free to download, use, and share.
OPEN
REVOLUTION R OPEN
GET IT NOW

Got comments or suggestions for the blog editor? Email David Smith.

Follow David on Twitter: @revodavid
+David Smith

Subscribe to this blog's feed

Categories

academia
advanced tips

...supported by the community

R news and tutorials contributed by (552) R bloggers

WELCOME!

Here you will find daily **news and tutorials about R**, contributed by over 555 bloggers. You can subscribe for e-mail updates:

And get updates to your Facebook:

17,457 people like R

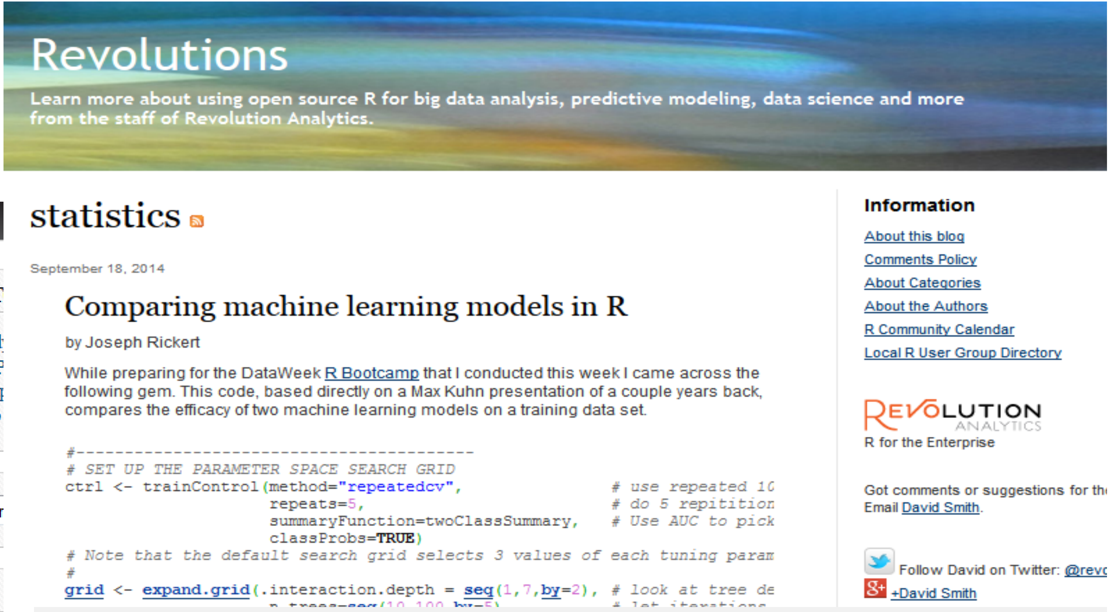
Fun with the Raspberry Pi

January 21, 2014

By Markus Gesmann

(This article was first published on **mages' blog**, and kindly contributed to R-bloggers)

Since Christmas I have been playing around with a Raspberry Pi. It is certainly not the fastest computer, but what a great little toy! Here are a few experiences and online resources that I found helpful.



Learn more about using open source R for big data analysis, predictive modeling, data science and more from the staff of Revolution Analytics.

statistics

September 18, 2014

Comparing machine learning models in R

by Joseph Rickert

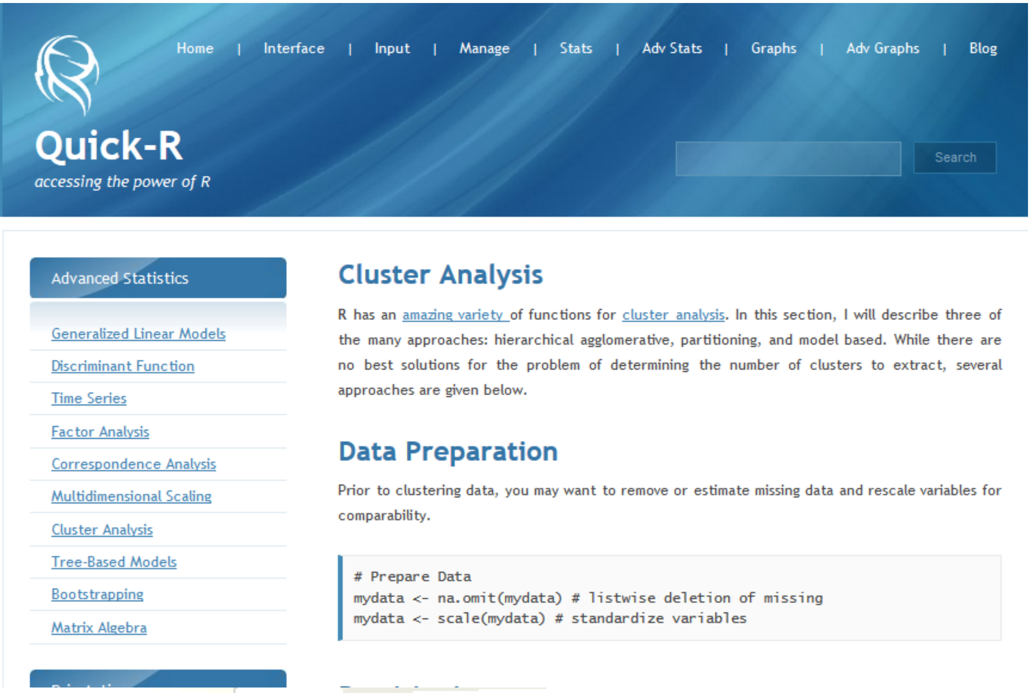
While preparing for the DataWeek R Bootcamp that I conducted this week I came across the following gem. This code, based directly on a Max Kuhn presentation of a couple years back, compares the efficacy of two machine learning models on a training data set.

```
#-----------------------------------------
# SET UP THE PARAMETER SPACE SEARCH GRID
ctrl <- trainControl(method="repeatedcv",          # use repeated 10
                     repeats=5,                    # do 5 repitition
                     summaryFunction=twoClassSummary,   # Use AUC to pick
                     classProbs=TRUE)
# Note that the default search grid selects 3 values of each tuning param
#
grid <- expand.grid(.interaction.depth = seq(1,7,by=2), # look at tree de
```

R for the Enterprise

Got comments or suggestions for th
Email David Smith.

Follow David on Twitter: @revo
+David Smith



accessing the power of R

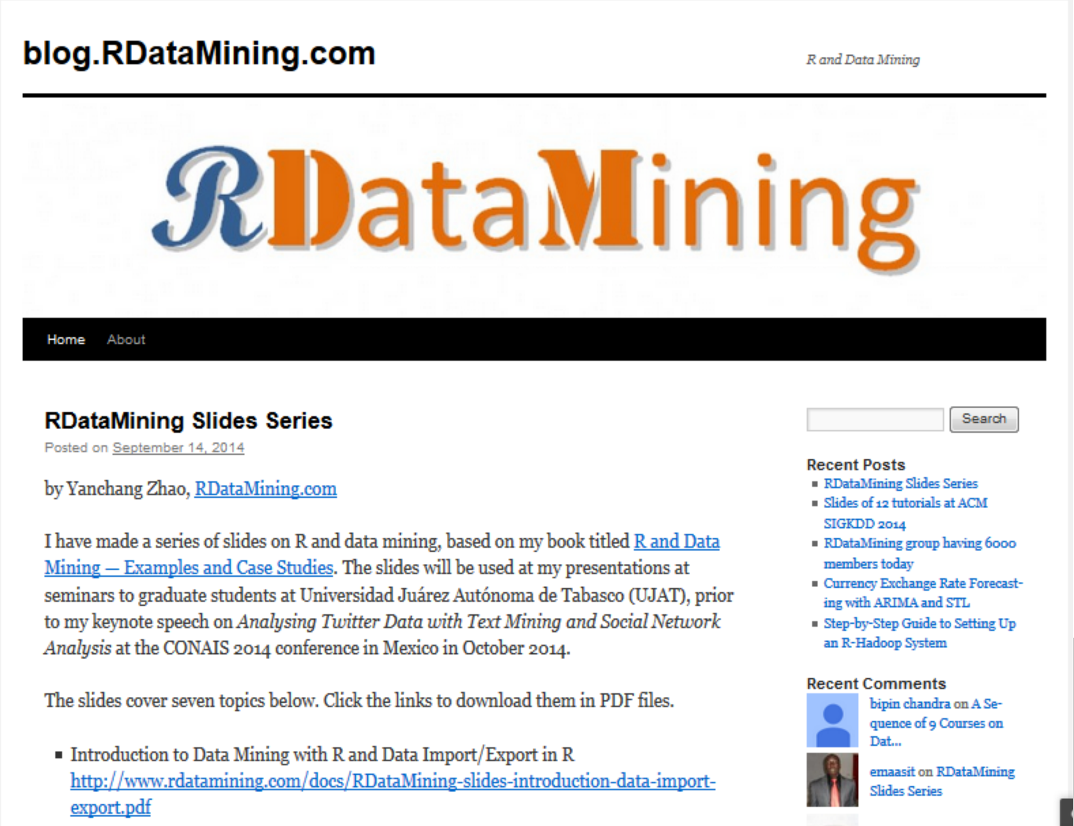
Cluster Analysis

R has an amazing variety of functions for cluster analysis. In this section, I will describe three of the many approaches: hierarchical agglomerative, partitioning, and model based. While there are no best solutions for the problem of determining the number of clusters to extract, several approaches are given below.

Data Preparation

Prior to clustering data, you may want to remove or estimate missing data and rescale variables for comparability.

```
# Prepare Data
mydata <- na.omit(mydata) # listwise deletion of missing
mydata <- scale(mydata) # standardize variables
```



RDataMining Slides Series

Posted on September 14, 2014

by Yanchang Zhao, RDataMining.com

I have made a series of slides on R and data mining, based on my book titled R and Data Mining – Examples and Case Studies. The slides will be used at my presentations at seminars to graduate students at Universidad Juárez Autónoma de Tabasco (UJAT), prior to my keynote speech on *Analysing Twitter Data with Text Mining and Social Network Analysis* at the CONAIS 2014 conference in Mexico in October 2014.

The slides cover seven topics below. Click the links to download them in PDF files.

- Introduction to Data Mining with R and Data Import/Export in R
  http://www.rdatamining.com/docs/RDataMining-slides-introduction-data-import-export.pdf

Recent Posts
- RDataMining Slides Series
- Slides of 12 tutorials at ACM SIGKDD 2014
- RDataMining group having 6000 members today
- Currency Exchange Rate Forecasting with ARIMA and STL
- Step-by-Step Guide to Setting Up an R-Hadoop System

Recent Comments

bipin chandra on A Sequence of 9 Courses on Dat...

emaasit on RDataMining Slides Series

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is supported by the world of science

IV ½ :) Books

V R is supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

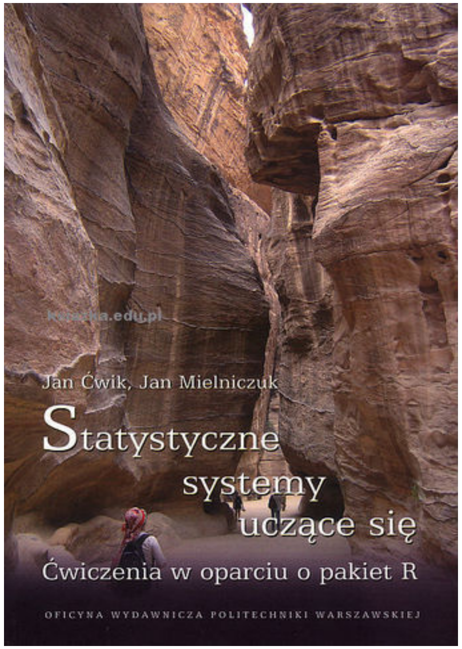
X There are many options to optimize the code

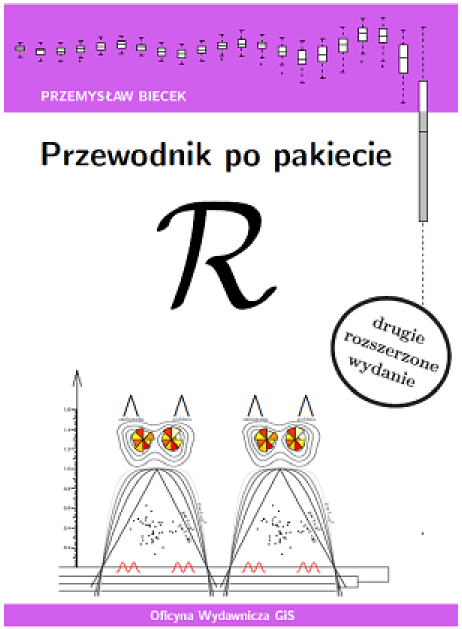
XI R is able to handle large amount of data

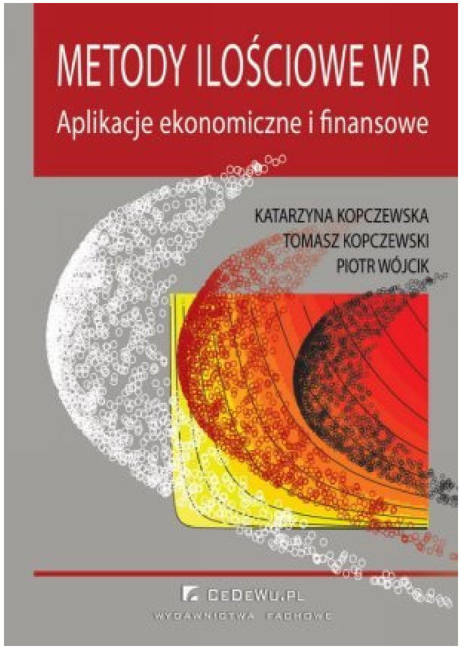
XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

...some polish books

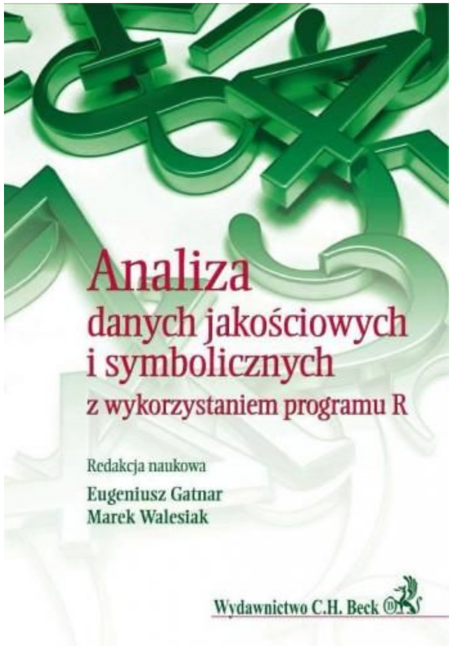

Jan Ćwik, Jan Mielniczuk
Statystyczne systemy uczące się
Ćwiczenia w oparciu o pakiet R
OFICYNA WYDAWNICZA POLITECHNIKI WARSZAWSKIEJ

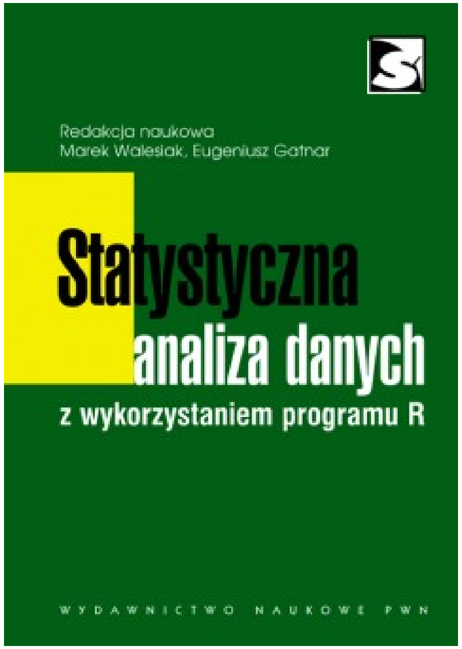

PRZEMYSŁAW BIECEK
Przewodnik po pakiecie R
drugie rozszerzone wydanie
Oficyna Wydawnicza GiS

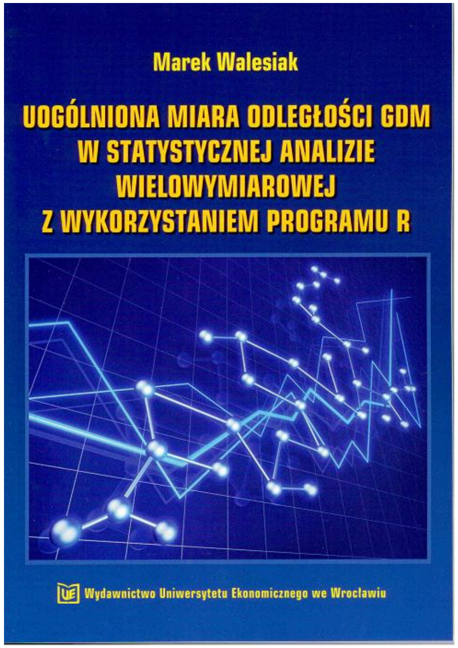

METODY ILOŚCIOWE W R
Aplikacje ekonomiczne i finansowe
KATARZYNA KOPCZEWSKA
TOMASZ KOPCZEWSKI
PIOTR WÓJCIK
CeDeWu.PL
WYDAWNICTWA FACHOWE


Technologia i rozwiązania
Analiza statystyczna w środowisku R dla początkujących
Dowiedz się, jak zwiększyć skuteczność analiz danych dzięki R!
Helion
John M. Quick
open source


Przemysław Biecek
Analiza danych z programem R
Modele liniowe z efektami stałymi, losowymi i mieszanymi
WYDAWNICTO NAUKOWE PWN


Analiza danych jakościowych i symbolicznych z wykorzystaniem programu R
Redakcja naukowa
Eugeniusz Gatnar
Marek Walesiak
Wydawnictwo C.H. Beck


Redakcja naukowa
Marek Walesiak, Eugeniusz Gatnar
Statystyczna analiza danych z wykorzystaniem programu R
WYDAWNICTWO NAUKOWE PWN


Marek Walesiak
UOGÓLNIONA MIARA ODLEGŁOŚCI GDM W STATYSTYCZNEJ ANALIZIE WIELOWYMIAROWEJ Z WYKORZYSTANIEM PROGRAMU R
Wydawnictwo Uniwersytetu Ekonomicznego we Wrocławiu

...some foReign books...

Biostatistical Design and Analysis Using R
A Practical Guide
Murray Logan

Chapman & Hall/CRC Biostatistics Series
Clinical Trial Data Analysis Using R
Ding-Geng (Din) Chen
Karl E. Peace
CRC Press
A CHAPMAN & HALL BOOK

Chapman & Hall/CRC Biostatistics Series
Applied Meta-Analysis with R
Ding-Geng (Din) Chen
Karl E. Peace
CRC
A CHAPMAN & HA

Springer Texts in Statistics
Robert H. Shumway
David S. Stoffer
Time Series Analysis and Its Applications
With R Examples
Third Edition
Springer

Making Everything Easier!™
R FOR DUMMIES®
Learn to:
- Use R for data analysis and processing
- Write functions and scripts for repeatable analysis
- Create high-quality charts and graphics
- Perform statistical analysis and build models

Andrie de Vries
Joris Meys

Copyrighted Material
R IN ACTION
Data analysis and graphics with R
Robert I. Kabacoff
MANNING
Copyrighted Material

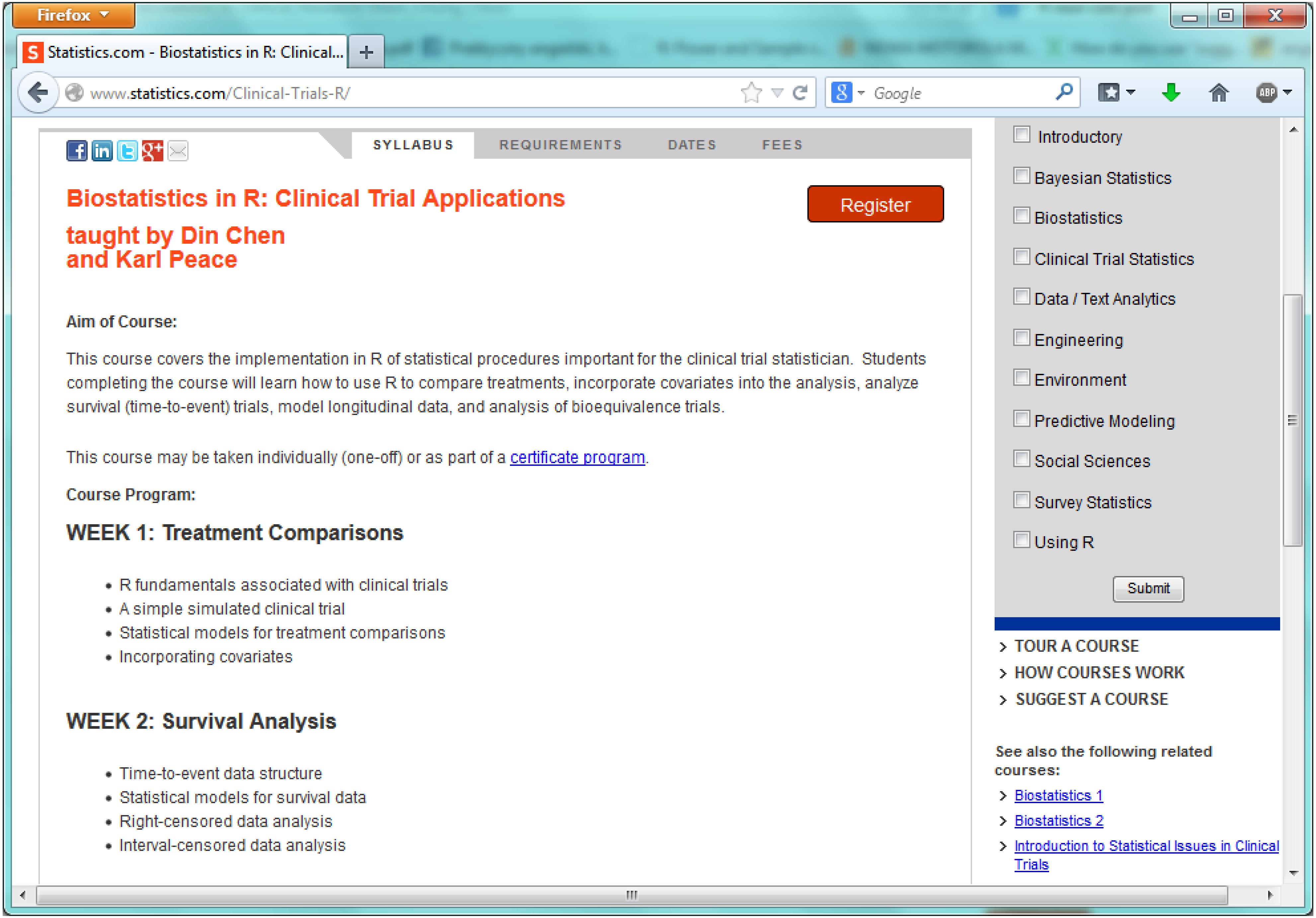
Jim Albert
Bayesian Computation with R
Springer

Texts in Statistical Science
xtending the Linear Model With R
Generalized Linear, Mixed Effects and Nonparametric Regression Models
Julian J. Faraway
Chapman & Hall/CRC

Use R!
Dan Lin
Ziv Shkedy
Daniel Yekutieli
Dhammika Amaratunga
Luc Bijnens Editors
Modeling Dose-Response Microarray Data in Early Drug Development Experiments Using R
Order-Restricted Analysis of Microarray Data
Springer

INSTANT R
An Introduction to R for Statistical Analysis
SARAH STOWELL

…and courses provided by their authors…



Biostatistics in R: Clinical Trial Applications

taught by Din Chen
and Karl Peace

Aim of Course:

This course covers the implementation in R of statistical procedures important for the clinical trial statistician. Students completing the course will learn how to use R to compare treatments, incorporate covariates into the analysis, analyze survival (time-to-event) trials, model longitudinal data, and analysis of bioequivalence trials.

This course may be taken individually (one-off) or as part of a certificate program.

Course Program:

WEEK 1: Treatment Comparisons

- R fundamentals associated with clinical trials
- A simple simulated clinical trial
- Statistical models for treatment comparisons
- Incorporating covariates

WEEK 2: Survival Analysis

- Time-to-event data structure
- Statistical models for survival data
- Right-censored data analysis
- Interval-censored data analysis

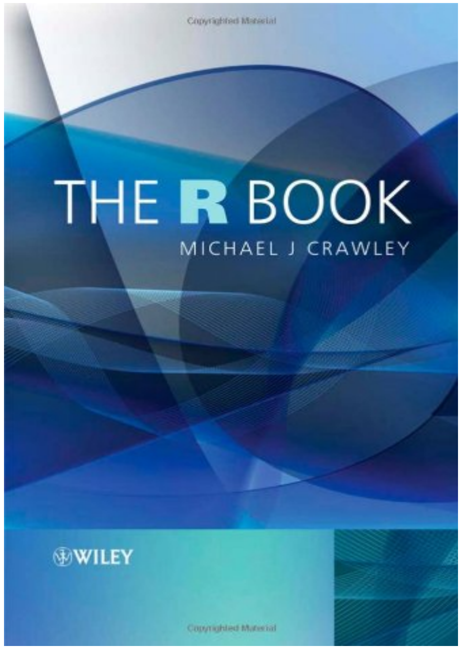
☐ Introductory
☐ Bayesian Statistics
☐ Biostatistics
☐ Clinical Trial Statistics
☐ Data / Text Analytics
☐ Engineering
☐ Environment
☐ Predictive Modeling
☐ Social Sciences
☐ Survey Statistics
☐ Using R

- TOUR A COURSE
- HOW COURSES WORK
- SUGGEST A COURSE

See also the following related courses:

- Biostatistics 1
- Biostatistics 2
- Introduction to Statistical Issues in Clinical Trials

...more foreign books...

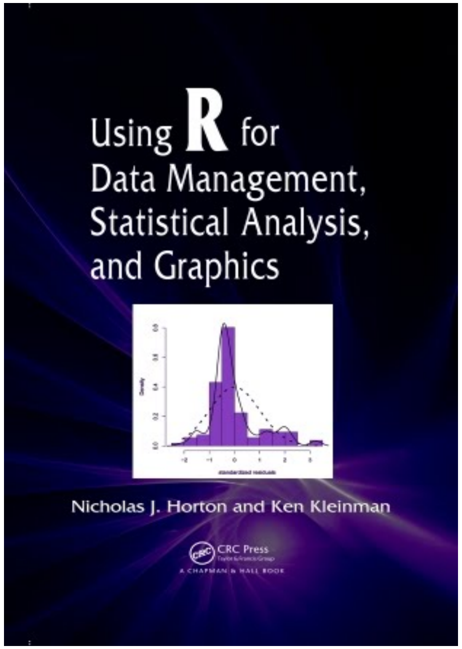

Copyrighted Material

THE R BOOK

MICHAEL J CRAWLEY

WILEY

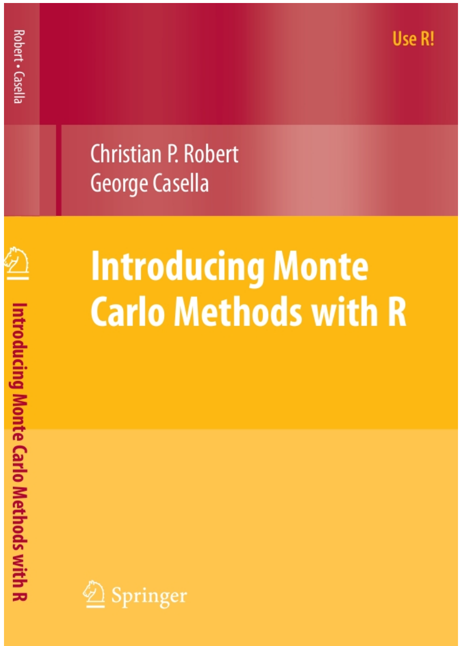
Copyrighted Material


Using R for Data Management, Statistical Analysis, and Graphics

Nicholas J. Horton and Ken Kleinman

CRC Press

A CHAPMAN & HALL BOOK


Use R!

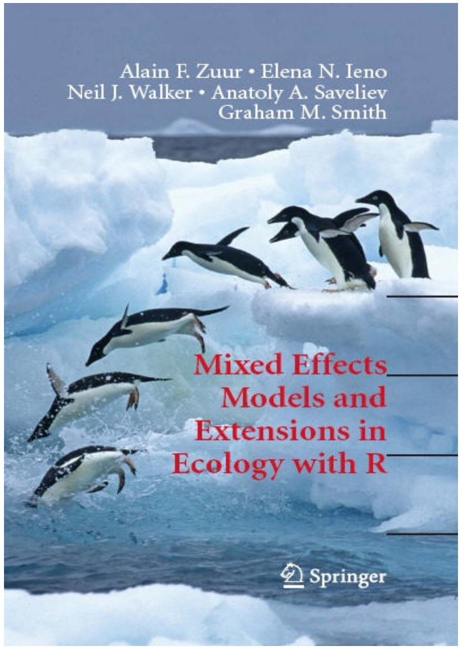
Christian P. Robert
George Casella

Introducing Monte Carlo Methods with R

Springer

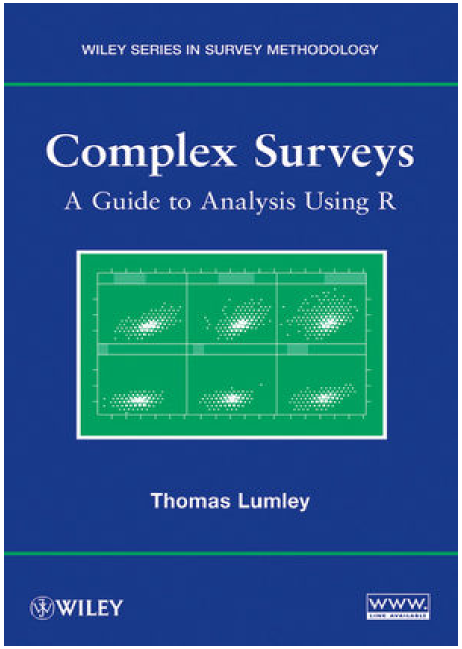
Robert · Casella

Introducing Monte Carlo Methods with R


Alain F. Zuur • Elena N. Ieno
Neil J. Walker • Anatoly A. Saveliev
Graham M. Smith

Mixed Effects Models and Extensions in Ecology with R

Springer


WILEY SERIES IN SURVEY METHODOLOGY

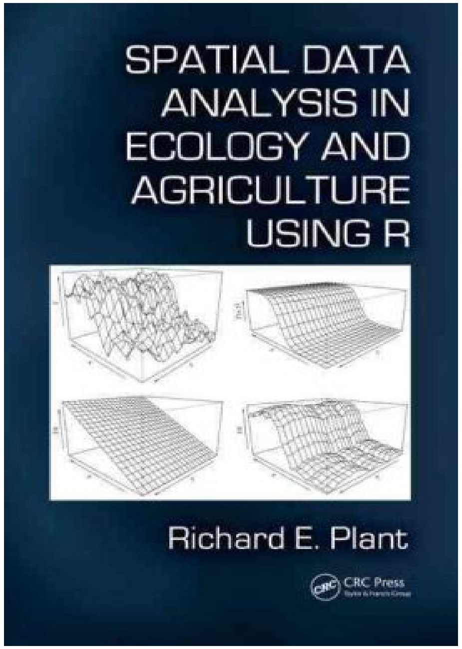
Complex Surveys

A Guide to Analysis Using R

Thomas Lumley

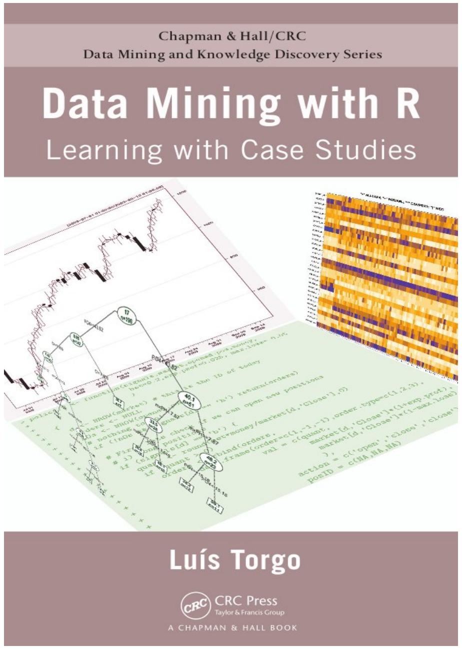
WILEY

WWW.


SPATIAL DATA ANALYSIS IN ECOLOGY AND AGRICULTURE USING R

Richard E. Plant

CRC Press


Chapman & Hall/CRC
Data Mining and Knowledge Discovery Series

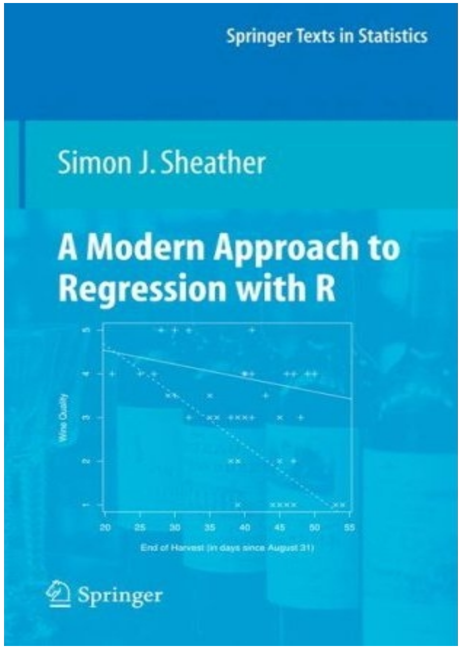
Data Mining with R

Learning with Case Studies

Luís Torgo

CRC Press

A CHAPMAN & HALL BOOK

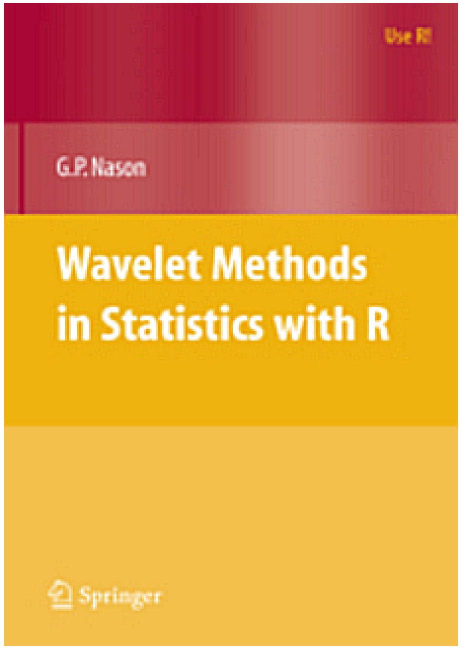

Springer Texts in Statistics

Simon J. Sheather

A Modern Approach to Regression with R

Springer

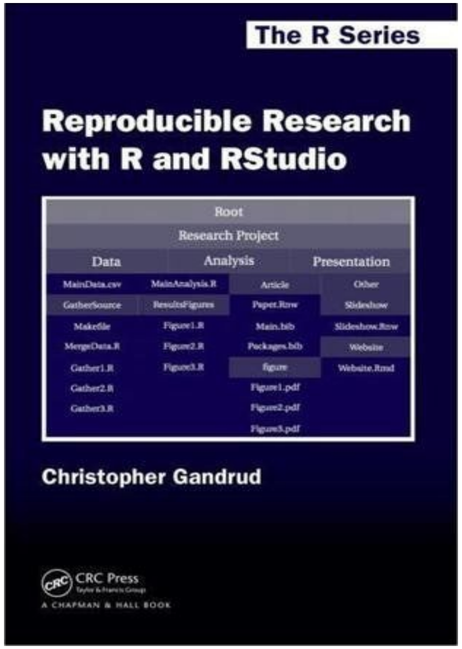
...and more foreign books...


Use R!

G.P. Nason

Wavelet Methods in Statistics with R

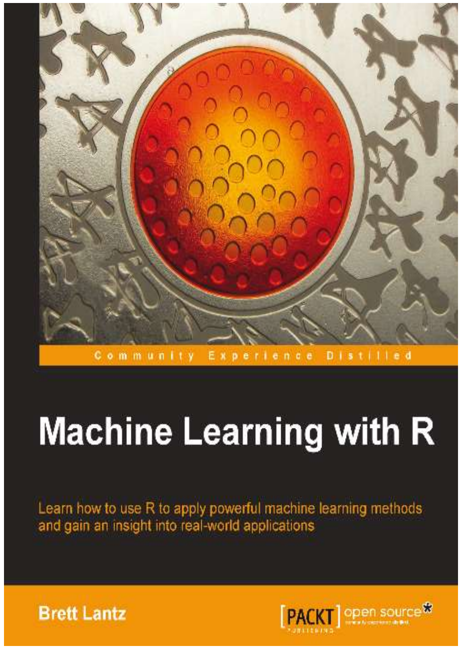
Springer


The R Series

Reproducible Research with R and RStudio

Christopher Gandrud

CRC Press
A CHAPMAN & HALL BOOK

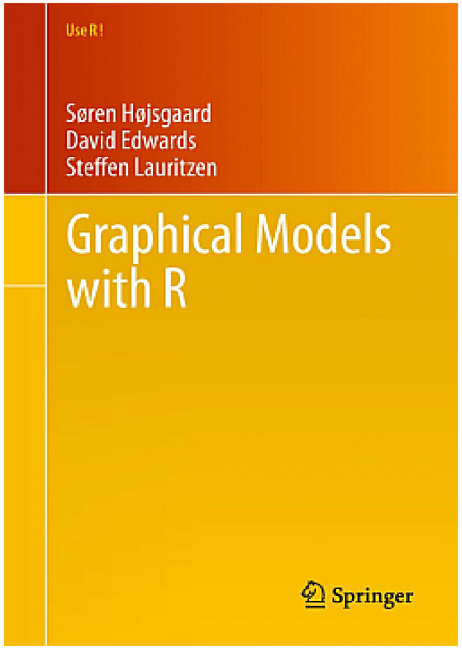

Community Experience Distilled

Machine Learning with R

Learn how to use R to apply powerful machine learning methods and gain an insight into real-world applications

Brett Lantz

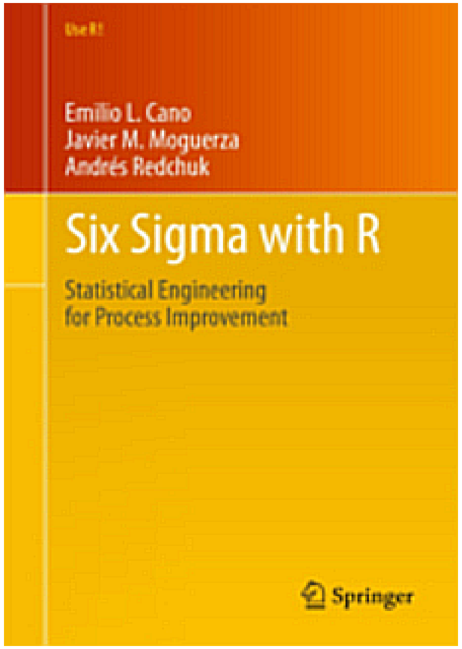
[PACKT] open source


Use R!

Søren Højsgaard
David Edwards
Steffen Lauritzen

Graphical Models with R

Springer


Use R!

Emilio L. Cano
Javier M. Moguerza
Andrés Redchuk

Six Sigma with R

Statistical Engineering for Process Improvement

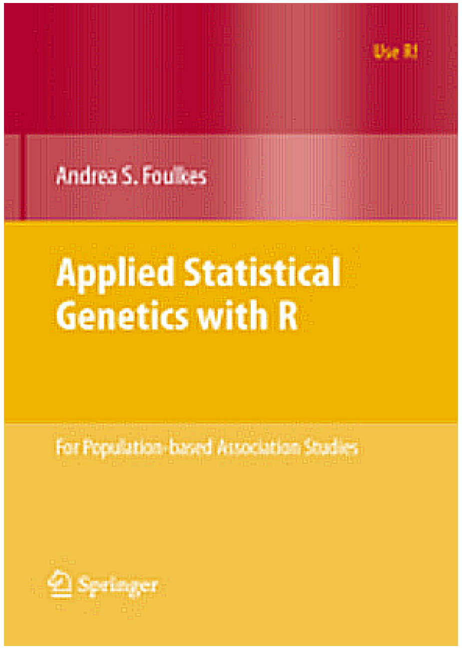
Springer


Use R!

Yuelin Li
Jonathan Baron

Behavioral Research Data Analysis with R

Springer

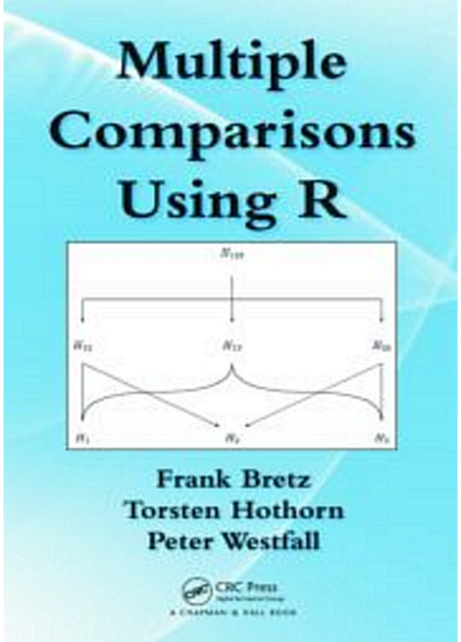

Use R!

Andrea S. Foulkes

Applied Statistical Genetics with R

For Population-based Association Studies

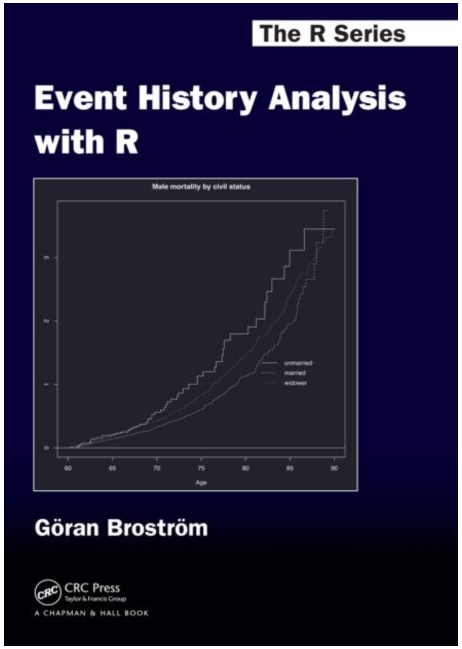
Springer


Multiple Comparisons Using R

Frank Bretz
Torsten Hothorn
Peter Westfall

CRC Press
A CHAPMAN & HALL BOOK

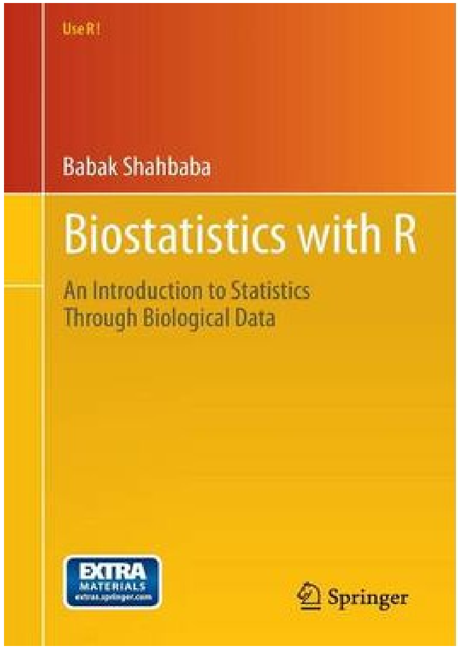
...and more foreign books...


The R Series

Event History Analysis with R

Göran Broström

CRC Press
A CHAPMAN & HALL BOOK


Use R!

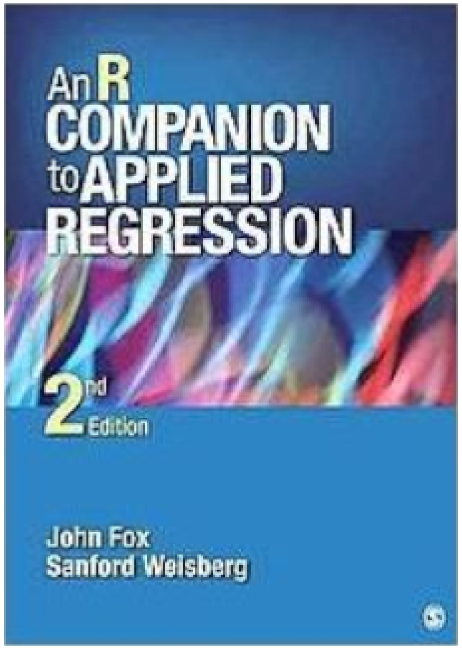
Babak Shahbaba

Biostatistics with R

An Introduction to Statistics Through Biological Data

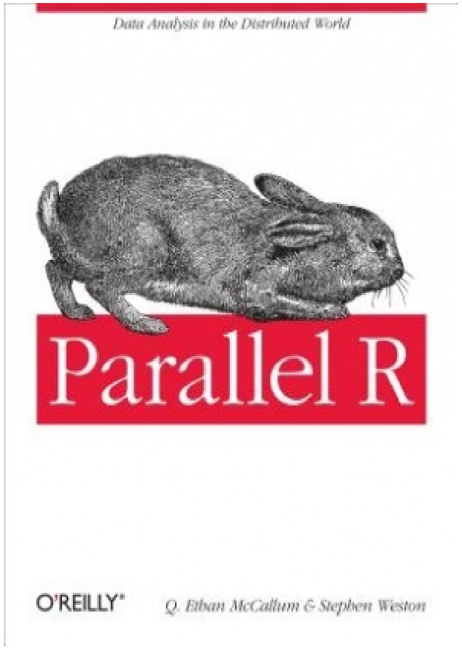
EXTRA MATERIALS

Springer


An R Companion to Applied Regression

2nd Edition

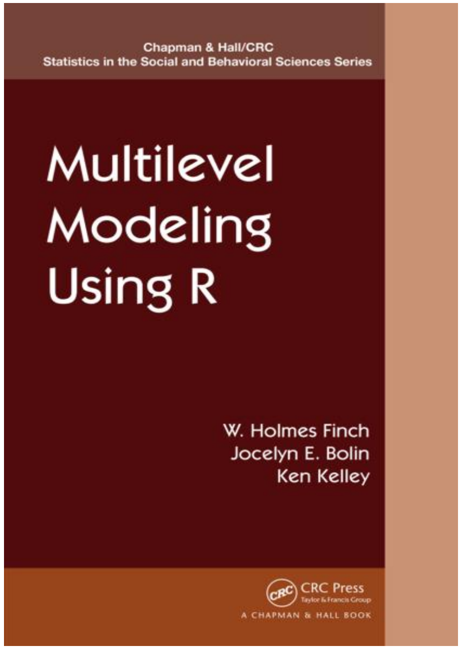
John Fox
Sanford Weisberg


Data Analysis in the Distributed World

Parallel R

O'REILLY

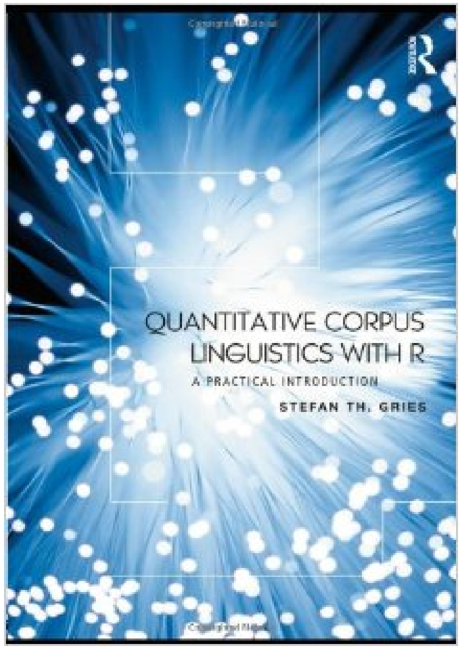
Q. Ethan McCallum & Stephen Weston


Chapman & Hall/CRC
Statistics in the Social and Behavioral Sciences Series

Multilevel Modeling Using R

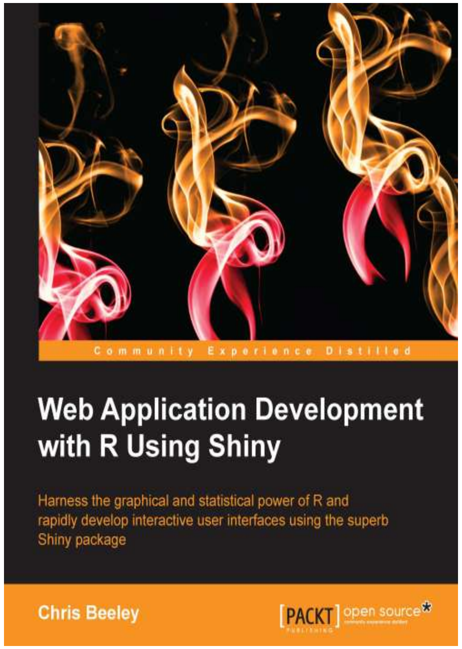
W. Holmes Finch
Jocelyn E. Bolin
Ken Kelley

CRC Press
A CHAPMAN & HALL BOOK


QUANTITATIVE CORPUS LINGUISTICS WITH R

A PRACTICAL INTRODUCTION

STEFAN TH. GRIES

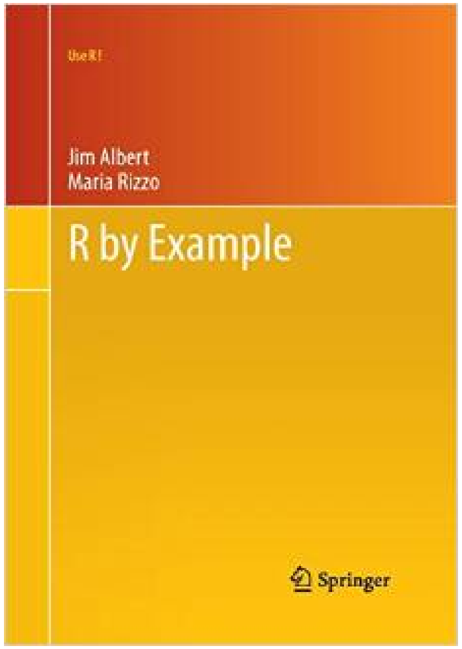

Community Experience Distilled

Web Application Development with R Using Shiny

Harness the graphical and statistical power of R and rapidly develop interactive user interfaces using the superb Shiny package

Chris Beeley

[PACKT] open source

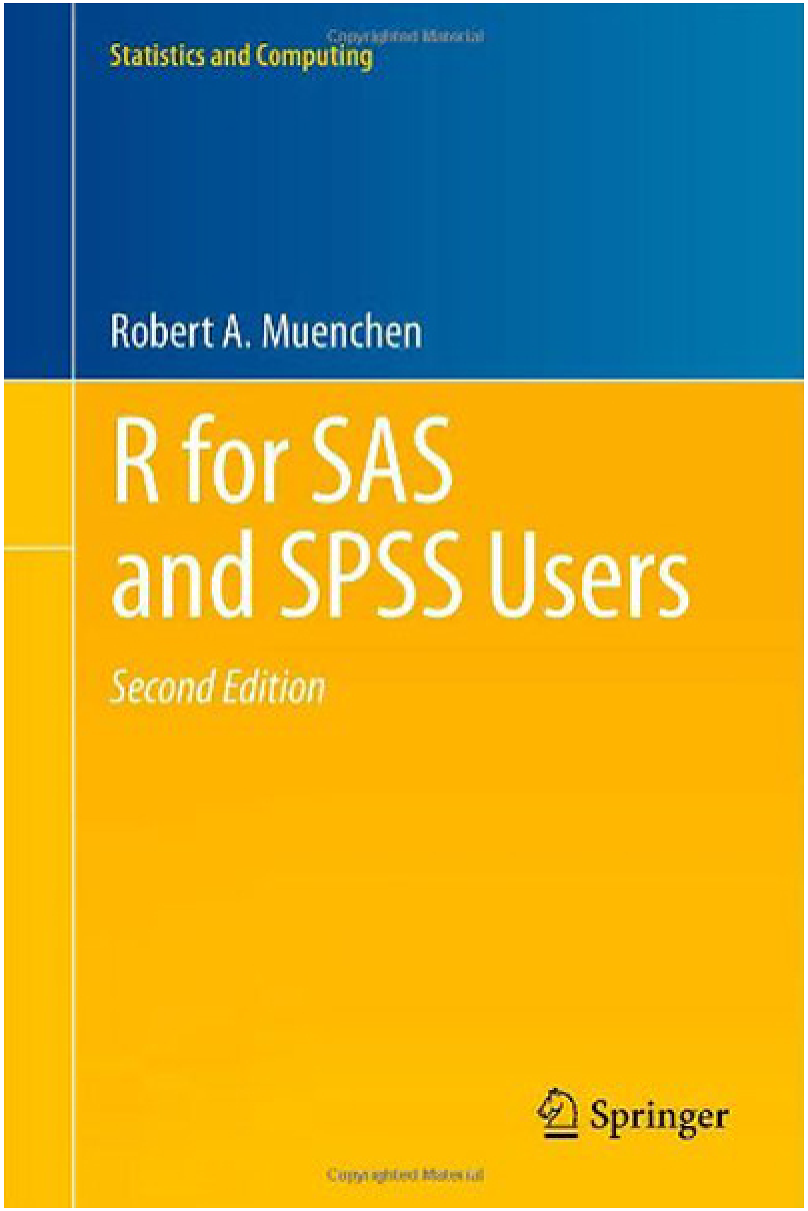

Use R!

Jim Albert
Maria Rizzo

R by Example

Springer

R for foReigners :-)



Statistics and Computing

Copyrighted Material

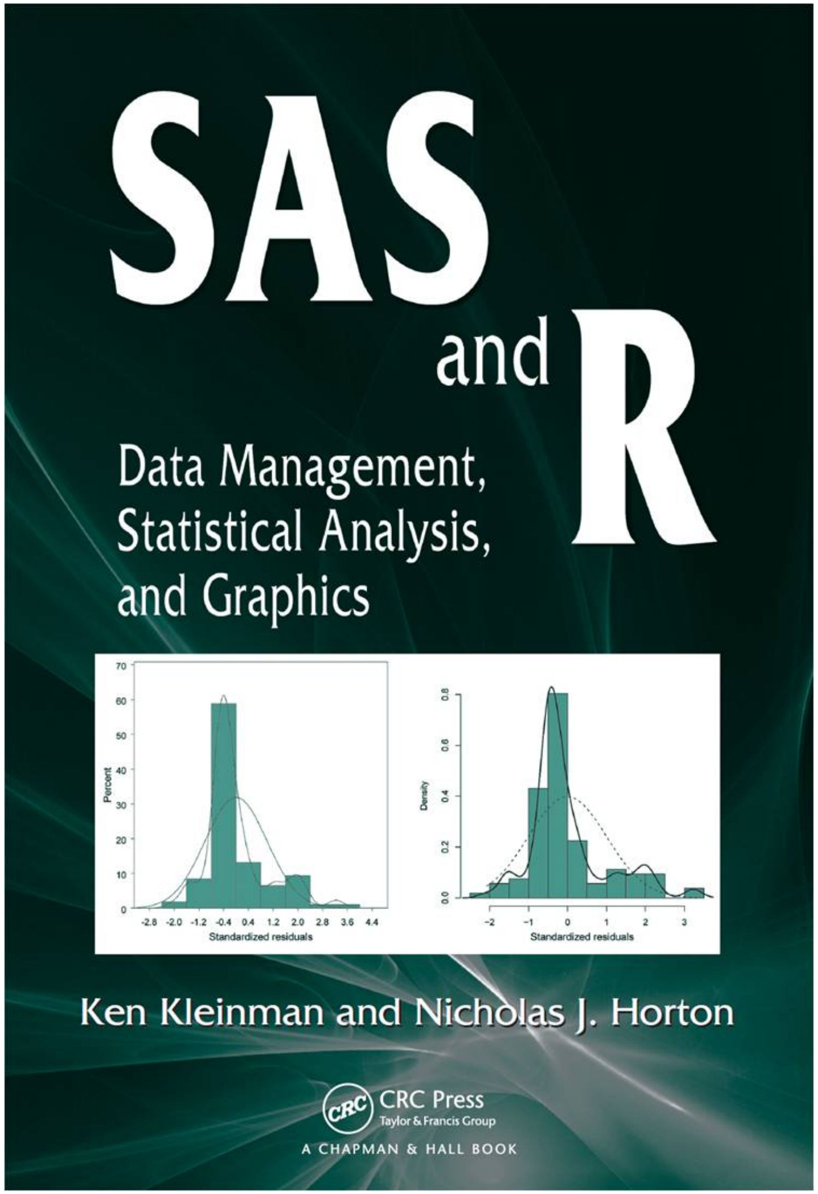
Robert A. Muenchen

R for SAS and SPSS Users

Second Edition

Springer

Copyrighted Material



SAS and R

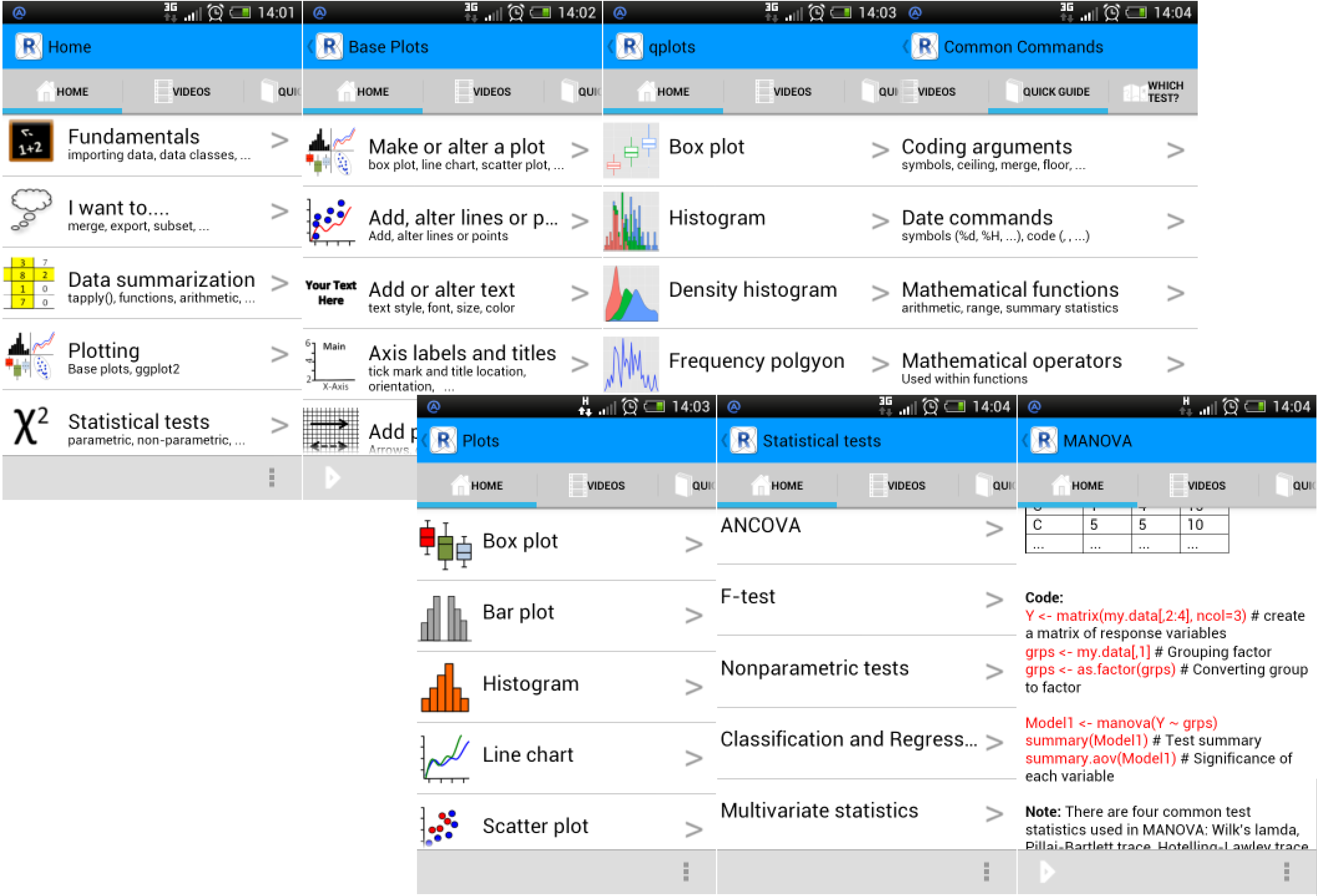
Data Management, Statistical Analysis, and Graphics

Ken Kleinman and Nicholas J. Horton

CRC Press
Taylor & Francis Group
A CHAPMAN & HALL BOOK

http://sas-and-r.blogspot.com

...and even mobile tutorials...



Fundamentals
importing data, data classes, ...

I want to....
merge, export, subset, ...

Data summarization
tapply(), functions, arithmetic, ...

Plotting
Base plots, ggplot2

Statistical tests
parametric, non-parametric, ...

Make or alter a plot
box plot, line chart, scatter plot, ...

Add, alter lines or p...
Add, alter lines or points

Add or alter text
text style, font, size, color

Axis labels and titles
tick mark and title location, orientation, ...

Add p
Arrows

Box plot

Histogram

Density histogram

Frequency polgyon

Coding arguments
symbols, ceiling, merge, floor, ...

Date commands
symbols (%d, %H, ...), code (, , ...)

Mathematical functions
arithmetic, range, summary statistics

Mathematical operators
Used within functions

Box plot

Bar plot

Histogram

Line chart

Scatter plot

ANCOVA

F-test

Nonparametric tests

Classification and Regress...

Multivariate statistics

C	5	5	10
...	...	...	...

Code:

```
Y <- matrix(my.data[,2:4], ncol=3) # create a matrix of response variables
grps <- my.data[,1] # Grouping factor
grps <- as.factor(grps) # Converting group to factor

Model1 <- manova(Y ~ grps)
summary(Model1) # Test summary
summary.aov(Model1) # Significance of each variable
```

Note: There are four common test statistics used in MANOVA: Wilk's lamda, Pillai-Bartlett trace, Hotelling-Lawley trace

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is (widely) supported by the world of science

IV R is supported by the community

V R is (increasingly) supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

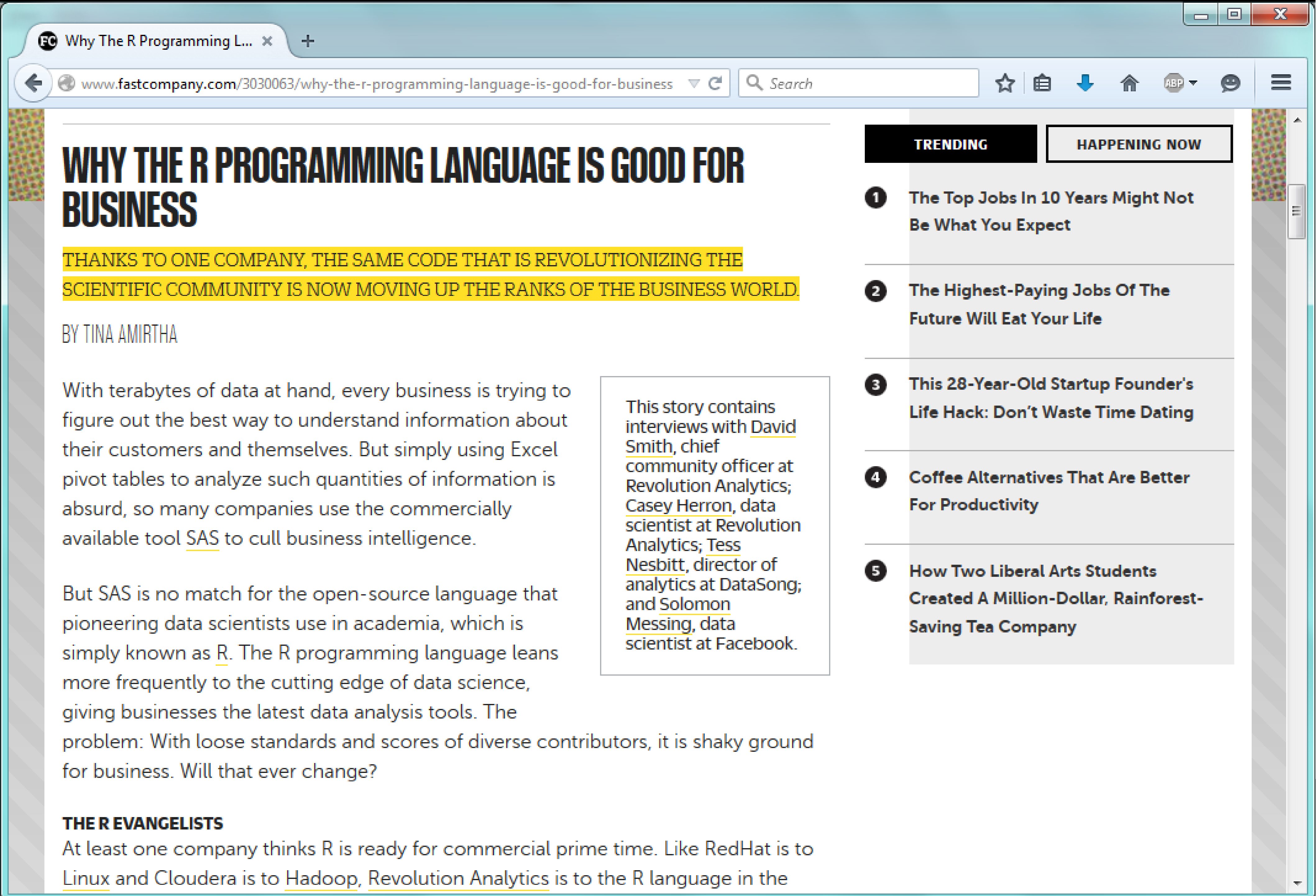
...supported by the **business**

The biggest tycoons in the market of statistical analysis and data mining recognize the potential of the R and develop specialistic packages as well as provide methods of easy integration between their products and R. These include companies such as Oracle, StatSoft, IBM (SPSS), Teradata, Merck, Tibco, Sybase, RapidMiner, SAS and others.

The situation is well described by the two following articles:

1. Adoption of R by large Enterprise Software Vendors
2. R integrated throughout the enterprise analytics stack

…supported by the **business**



WHY THE R PROGRAMMING LANGUAGE IS GOOD FOR BUSINESS

THANKS TO ONE COMPANY, THE SAME CODE THAT IS REVOLUTIONIZING THE SCIENTIFIC COMMUNITY IS NOW MOVING UP THE RANKS OF THE BUSINESS WORLD.

BY TINA AMIRTHA

With terabytes of data at hand, every business is trying to figure out the best way to understand information about their customers and themselves. But simply using Excel pivot tables to analyze such quantities of information is absurd, so many companies use the commercially available tool SAS to cull business intelligence.

But SAS is no match for the open-source language that pioneering data scientists use in academia, which is simply known as R. The R programming language leans more frequently to the cutting edge of data science, giving businesses the latest data analysis tools. The problem: With loose standards and scores of diverse contributors, it is shaky ground for business. Will that ever change?

This story contains interviews with David Smith, chief community officer at Revolution Analytics; Casey Herron, data scientist at Revolution Analytics; Tess Nesbitt, director of analytics at DataSong; and Solomon Messing, data scientist at Facebook.

THE R EVANGELISTS

At least one company thinks R is ready for commercial prime time. Like RedHat is to Linux and Cloudera is to Hadoop, Revolution Analytics is to the R language in the

TRENDING | **HAPPENING NOW**

1. The Top Jobs In 10 Years Might Not Be What You Expect
2. The Highest-Paying Jobs Of The Future Will Eat Your Life
3. This 28-Year-Old Startup Founder's Life Hack: Don't Waste Time Dating
4. Coffee Alternatives That Are Better For Productivity
5. How Two Liberal Arts Students Created A Million-Dollar, Rainforest-Saving Tea Company

R feat. Microsoft

May 15, 2015

In-database R coming to SQL Server 2016

R is coming to SQL Server. SQL Server 2016 (which will be in public preview this summer) will include new real-time analytics, automatic data encryption, and the ability to run R within the database itself:

> For deeper insights into data, SQL Server 2016 expands its scope beyond transaction processing, data warehousing and business intelligence to deliver advanced analytics as an additional workload in SQL Server with proven technology from Revolution Analytics. We want to make advanced analytics more accessible and increase performance for your advanced analytic workloads by bringing R processing closer to the data and building advanced analytic capabilities right into SQL Server. Additionally, we are building PolyBase into SQL Server, expanding the power to extract value from unstructured and structured data using your existing T-SQL skills. With this wave, you can then gain faster insights through rich visualizations on many devices including mobile applications on Windows, iOS and Android.

With this update, data scientists will no longer need to extract data from SQL server via ODBC to analyze it with R. Instead, you will be able to take your R code to the data, where is will be run inside a sandbox process within SQL Server itself. This eliminates the time and storage required to move the data, and gives you all the power of R and CRAN packages to apply to your database.

At last weeks' Microsoft Ignite conference in Chicago, SQL Server program managers Lindsey Allen and Borko Novakovic demonstrated a prototype of running R within SQL Server. (A description of the intergration begins at 57:00, and the demo at 1:05:00, in the video below.) In the demo, Lindsey applies a Naive Bayes classification model (from the e1071 R package) to the famous Iris data, using the same R code used in this Azure ML Studio experiment.

About this blog
Comments Policy
About Categories
About the Authors
R Community Calendar
Local R User Group Directory

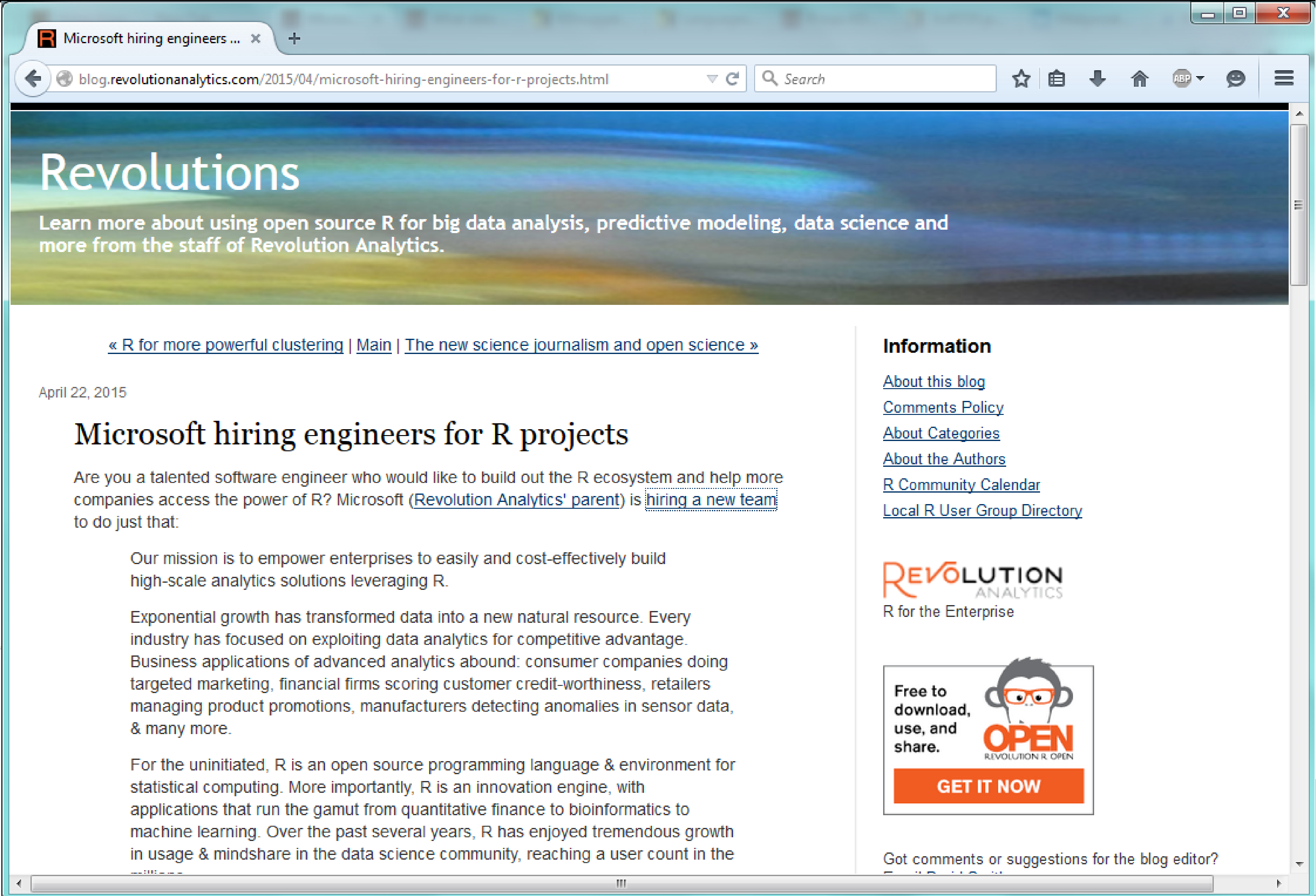
R for the Enterprise

Got comments or suggestions for the blog editor?
Email David Smith.

Follow David on Twitter: @revodavid
+David Smith

Subscribe to this blog's feed

R feat. Microsoft



Revolutions

Learn more about using open source R for big data analysis, predictive modeling, data science and more from the staff of Revolution Analytics.

« R for more powerful clustering | Main | The new science journalism and open science »

April 22, 2015

Microsoft hiring engineers for R projects

Are you a talented software engineer who would like to build out the R ecosystem and help more companies access the power of R? Microsoft (Revolution Analytics' parent) is hiring a new team to do just that:

> Our mission is to empower enterprises to easily and cost-effectively build high-scale analytics solutions leveraging R.
>
> Exponential growth has transformed data into a new natural resource. Every industry has focused on exploiting data analytics for competitive advantage. Business applications of advanced analytics abound: consumer companies doing targeted marketing, financial firms scoring customer credit-worthiness, retailers managing product promotions, manufacturers detecting anomalies in sensor data, & many more.
>
> For the uninitiated, R is an open source programming language & environment for statistical computing. More importantly, R is an innovation engine, with applications that run the gamut from quantitative finance to bioinformatics to machine learning. Over the past several years, R has enjoyed tremendous growth in usage & mindshare in the data science community, reaching a user count in the

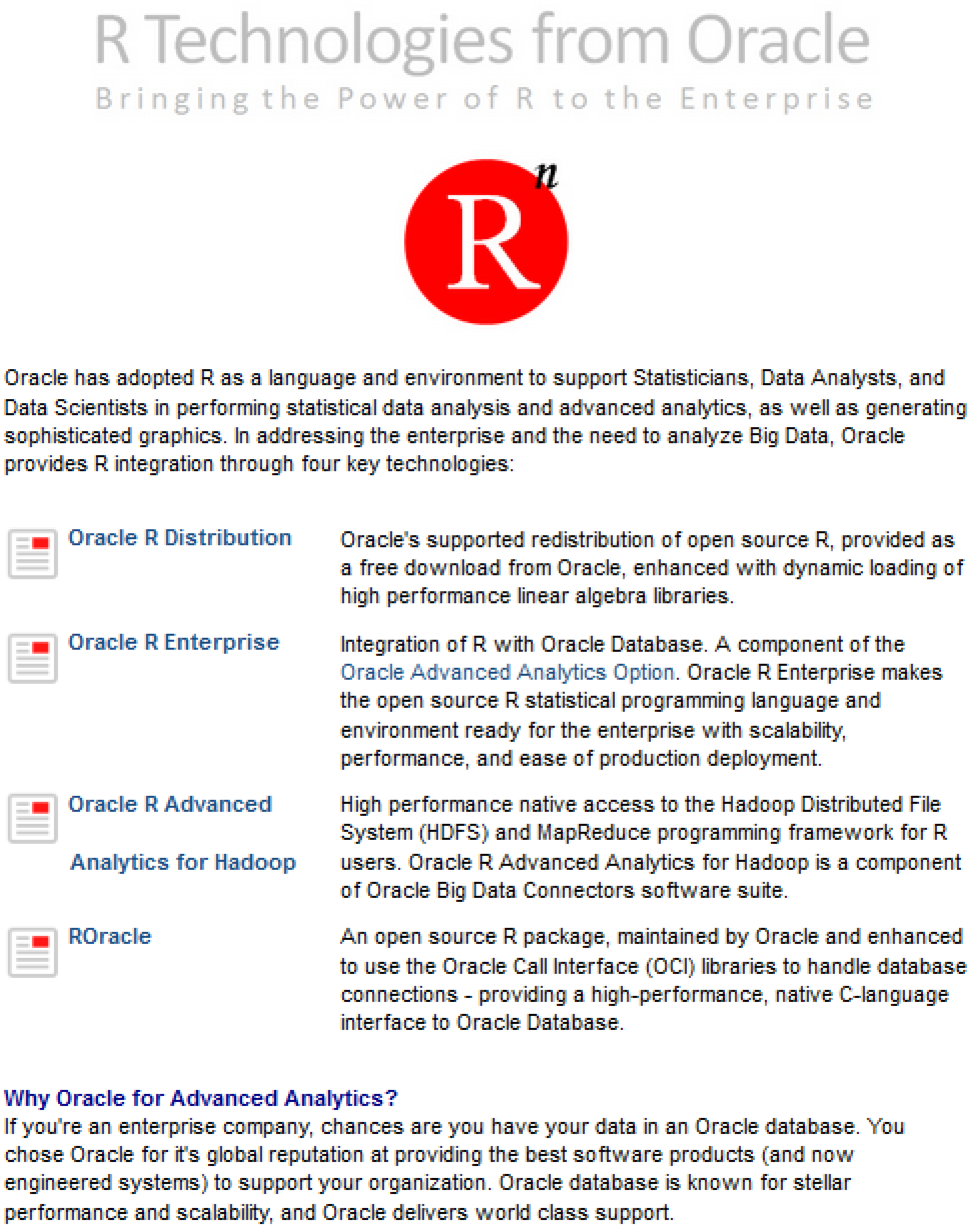
Information

- About this blog
- Comments Policy
- About Categories
- About the Authors
- R Community Calendar
- Local R User Group Directory

R for the Enterprise

Got comments or suggestions for the blog editor?

R feat. Oracle



R Technologies from Oracle

Bringing the Power of R to the Enterprise

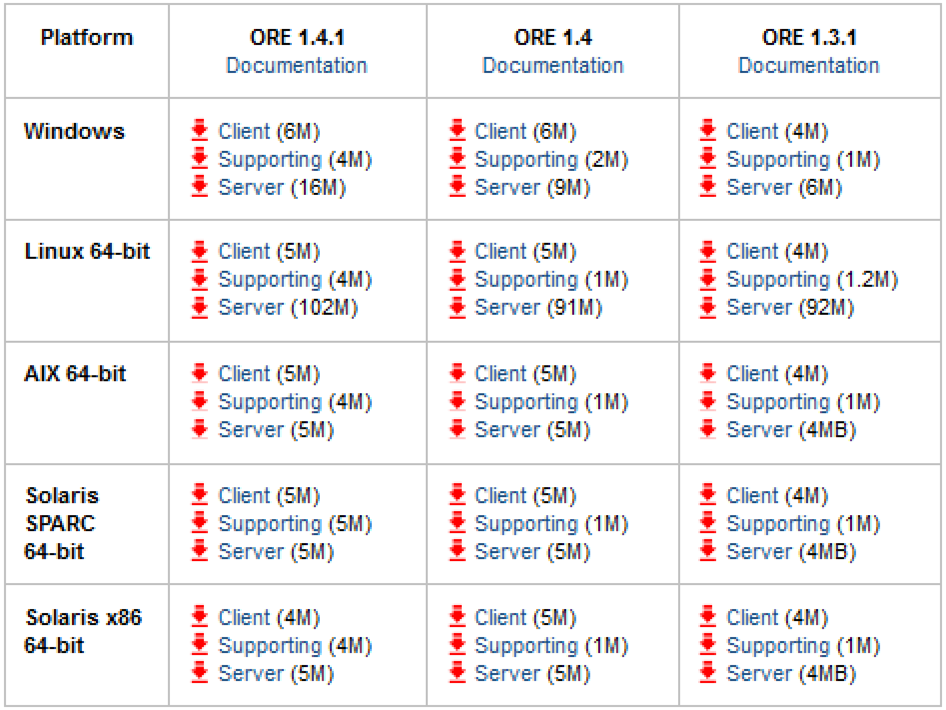
Oracle has adopted R as a language and environment to support Statisticians, Data Analysts, and Data Scientists in performing statistical data analysis and advanced analytics, as well as generating sophisticated graphics. In addressing the enterprise and the need to analyze Big Data, Oracle provides R integration through four key technologies:

Oracle R Distribution	Oracle's supported redistribution of open source R, provided as a free download from Oracle, enhanced with dynamic loading of high performance linear algebra libraries.
Oracle R Enterprise	Integration of R with Oracle Database. A component of the Oracle Advanced Analytics Option. Oracle R Enterprise makes the open source R statistical programming language and environment ready for the enterprise with scalability, performance, and ease of production deployment.
Oracle R Advanced Analytics for Hadoop	High performance native access to the Hadoop Distributed File System (HDFS) and MapReduce programming framework for R users. Oracle R Advanced Analytics for Hadoop is a component of Oracle Big Data Connectors software suite.
ROracle	An open source R package, maintained by Oracle and enhanced to use the Oracle Call Interface (OCI) libraries to handle database connections - providing a high-performance, native C-language interface to Oracle Database.

Why Oracle for Advanced Analytics?

If you're an enterprise company, chances are you have your data in an Oracle database. You chose Oracle for it's global reputation at providing the best software products (and now engineered systems) to support your organization. Oracle database is known for stellar performance and scalability, and Oracle delivers world class support.

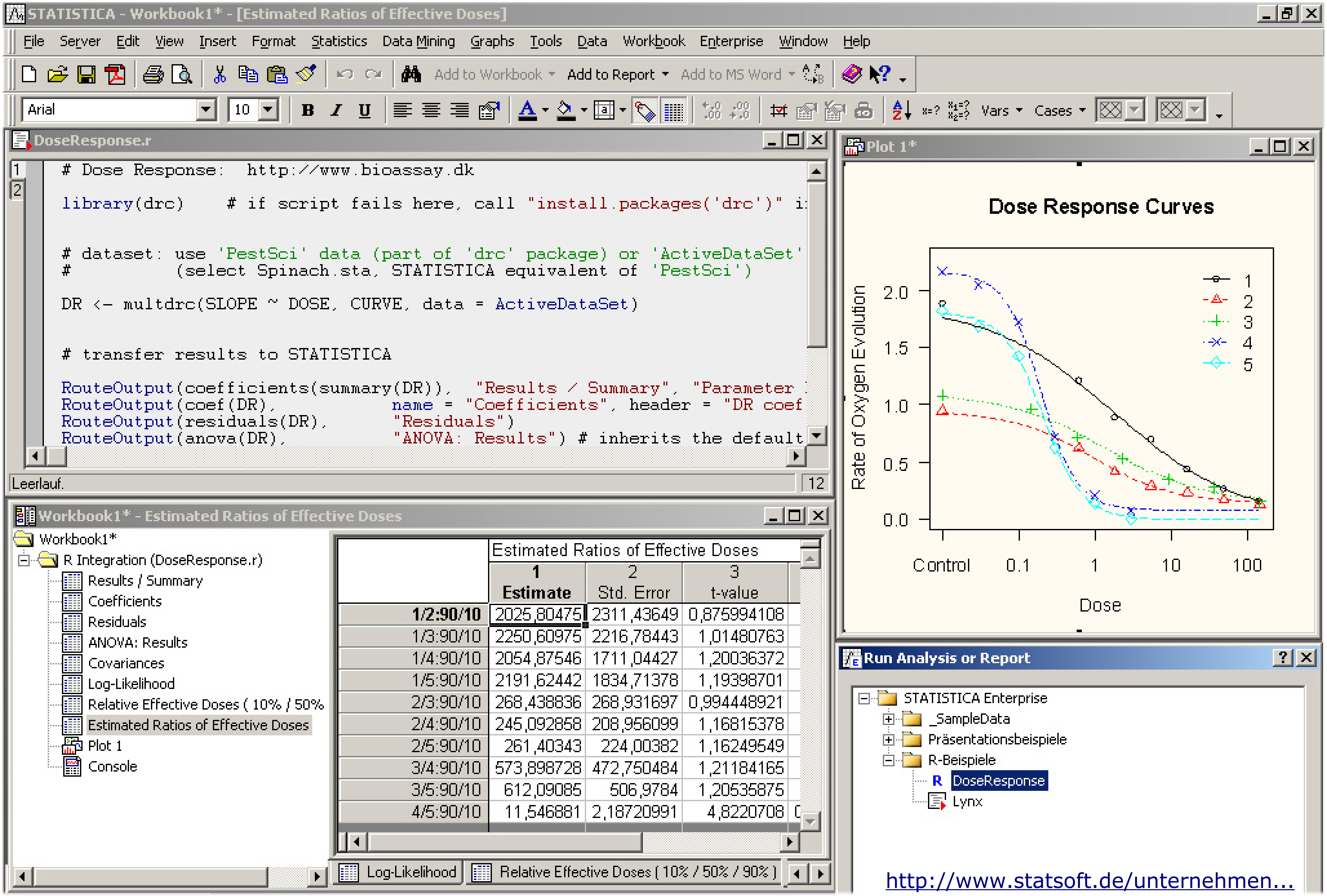
R in the world of **big UNIX machines** :)



Platform	ORE 1.4.1 Documentation	ORE 1.4 Documentation	ORE 1.3.1 Documentation
Windows	Client (6M) Supporting (4M) Server (16M)	Client (6M) Supporting (2M) Server (9M)	Client (4M) Supporting (1M) Server (6M)
Linux 64-bit	Client (5M) Supporting (4M) Server (102M)	Client (5M) Supporting (1M) Server (91M)	Client (4M) Supporting (1.2M) Server (92M)
AIX 64-bit	Client (5M) Supporting (4M) Server (5M)	Client (5M) Supporting (1M) Server (5M)	Client (4M) Supporting (1M) Server (4MB)
Solaris SPARC 64-bit	Client (5M) Supporting (5M) Server (5M)	Client (5M) Supporting (1M) Server (5M)	Client (4M) Supporting (1M) Server (4MB)
Solaris x86 64-bit	Client (4M) Supporting (4M) Server (5M)	Client (5M) Supporting (1M) Server (5M)	Client (4M) Supporting (1M) Server (4MB)

http://www.oracle.com/technetwork/database/database-technologies/r/r-technologies...

R feat. **Statistica**

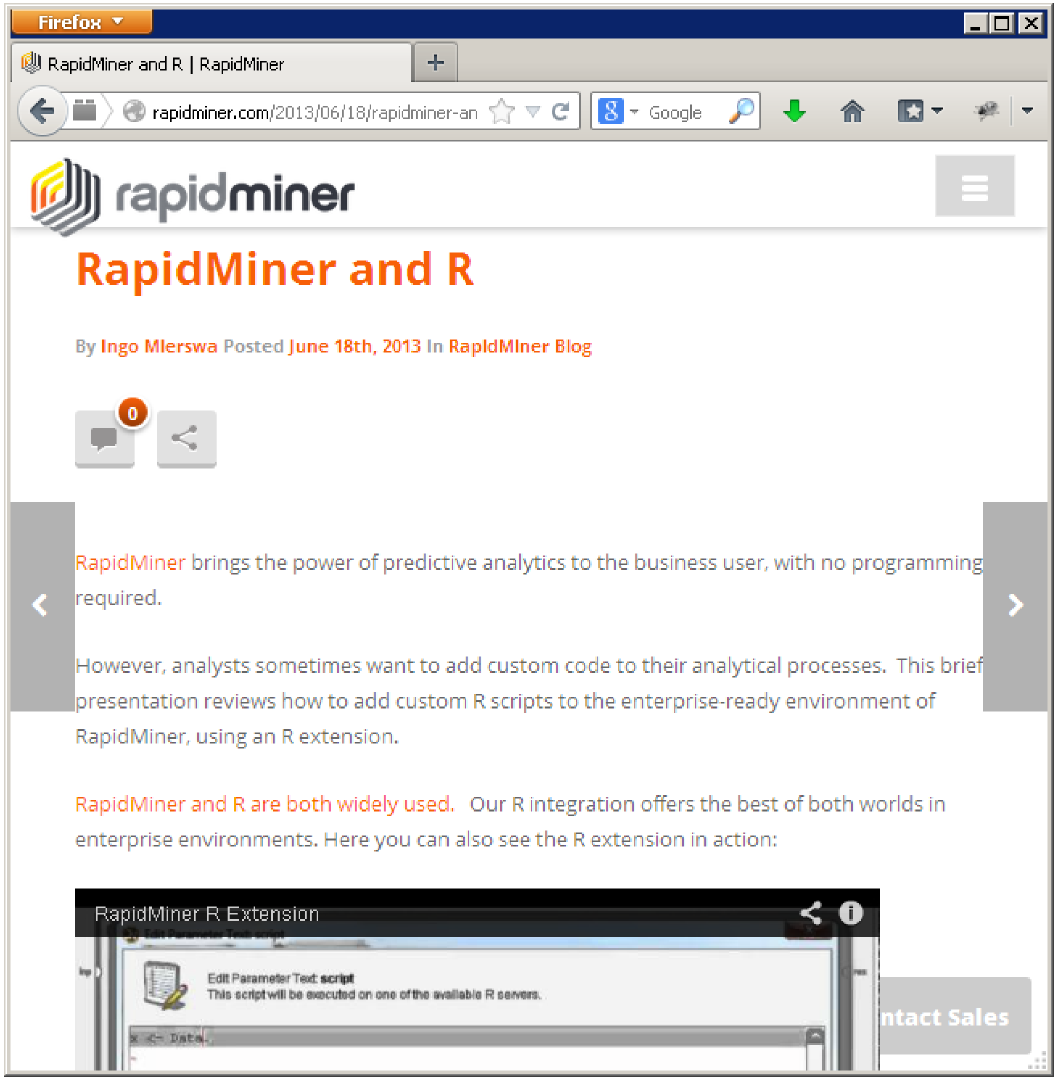


```
# Dose Response:  http://www.bioassay.dk

library(drc)    # if script fails here, call "install.packages('drc')" i


# dataset: use 'PestSci' data (part of 'drc' package) or 'ActiveDataSet'
#          (select Spinach.sta, STATISTICA equivalent of 'PestSci')

DR <- multdrc(SLOPE ~ DOSE, CURVE, data = ActiveDataSet)


# transfer results to STATISTICA

RouteOutput(coefficients(summary(DR)),  "Results / Summary", "Parameter
RouteOutput(coef(DR),              name = "Coefficients", header = "DR coef
RouteOutput(residuals(DR),         "Residuals")
RouteOutput(anova(DR),             "ANOVA: Results") # inherits the default
```

	Estimated Ratios of Effective Doses		
	1 Estimate	2 Std. Error	3 t-value
1/2:90/10	2025,80475	2311,43649	0,875994108
1/3:90/10	2250,60975	2216,78443	1,01480763
1/4:90/10	2054,87546	1711,04427	1,20036372
1/5:90/10	2191,62442	1834,71378	1,19398701
2/3:90/10	268,438836	268,931697	0,994448921
2/4:90/10	245,092858	208,956099	1,16815378
2/5:90/10	261,40343	224,00382	1,16249549
3/4:90/10	573,898728	472,750484	1,21184165
3/5:90/10	612,09085	506,9784	1,20535875
4/5:90/10	11,546881	2,18720991	4,8220708

http://www.statsoft.de/unternehmen...

R feat. RapidMiner



RapidMiner and R

By Ingo Mierswa Posted June 18th, 2013 In RapidMiner Blog

0

RapidMiner brings the power of predictive analytics to the business user, with no programming required.

However, analysts sometimes want to add custom code to their analytical processes. This brief presentation reviews how to add custom R scripts to the enterprise-ready environment of RapidMiner, using an R extension.

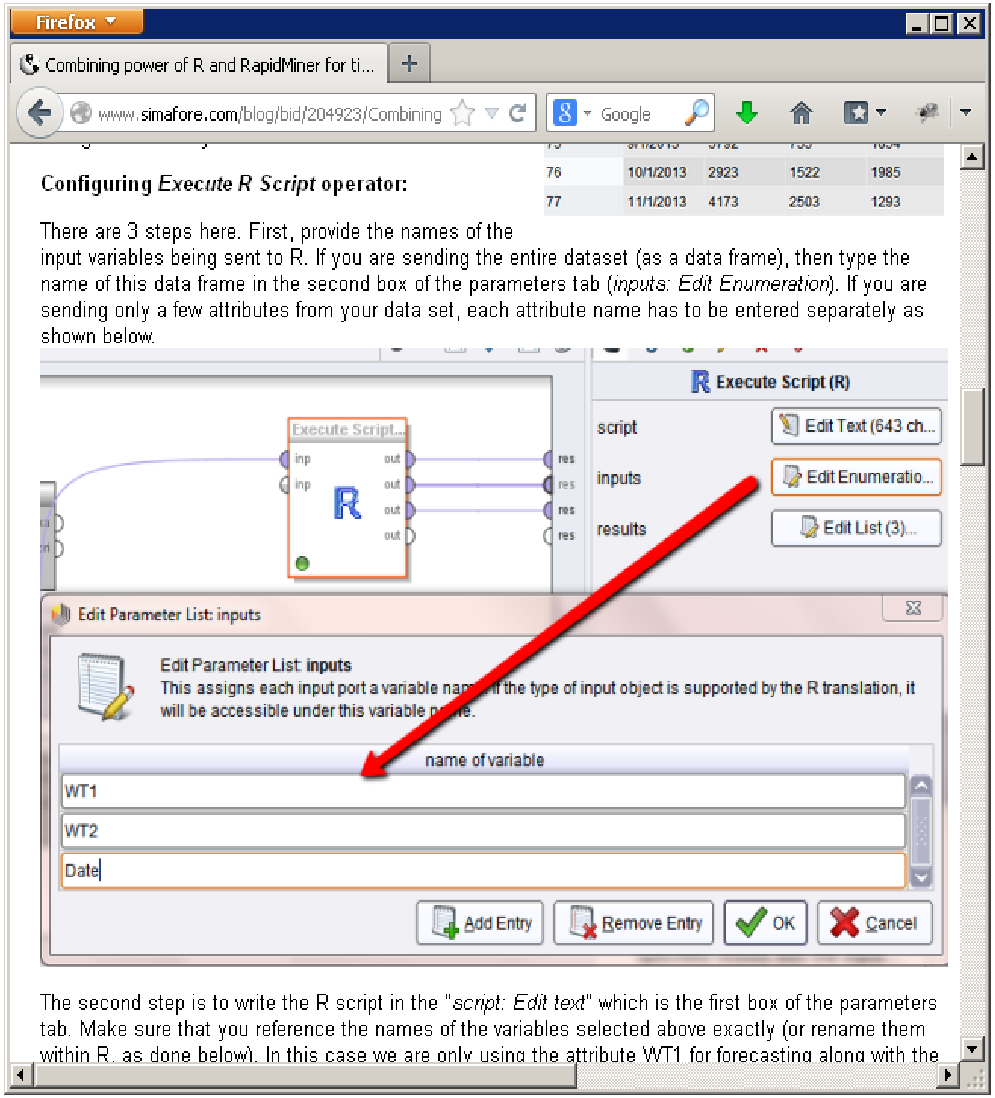
RapidMiner and R are both widely used. Our R integration offers the best of both worlds in enterprise environments. Here you can also see the R extension in action:

RapidMiner R Extension

Edit Parameter Text: script
This script will be executed on one of the available R servers.

http://rapidminer.com/2013/06/18/rapidminer-and-r/

http://www.simafore.com/blog/bid/204923/Combining-power-of-R



76	10/1/2013	2923	1522	1985
77	11/1/2013	4173	2503	1293

Configuring *Execute R Script* operator:

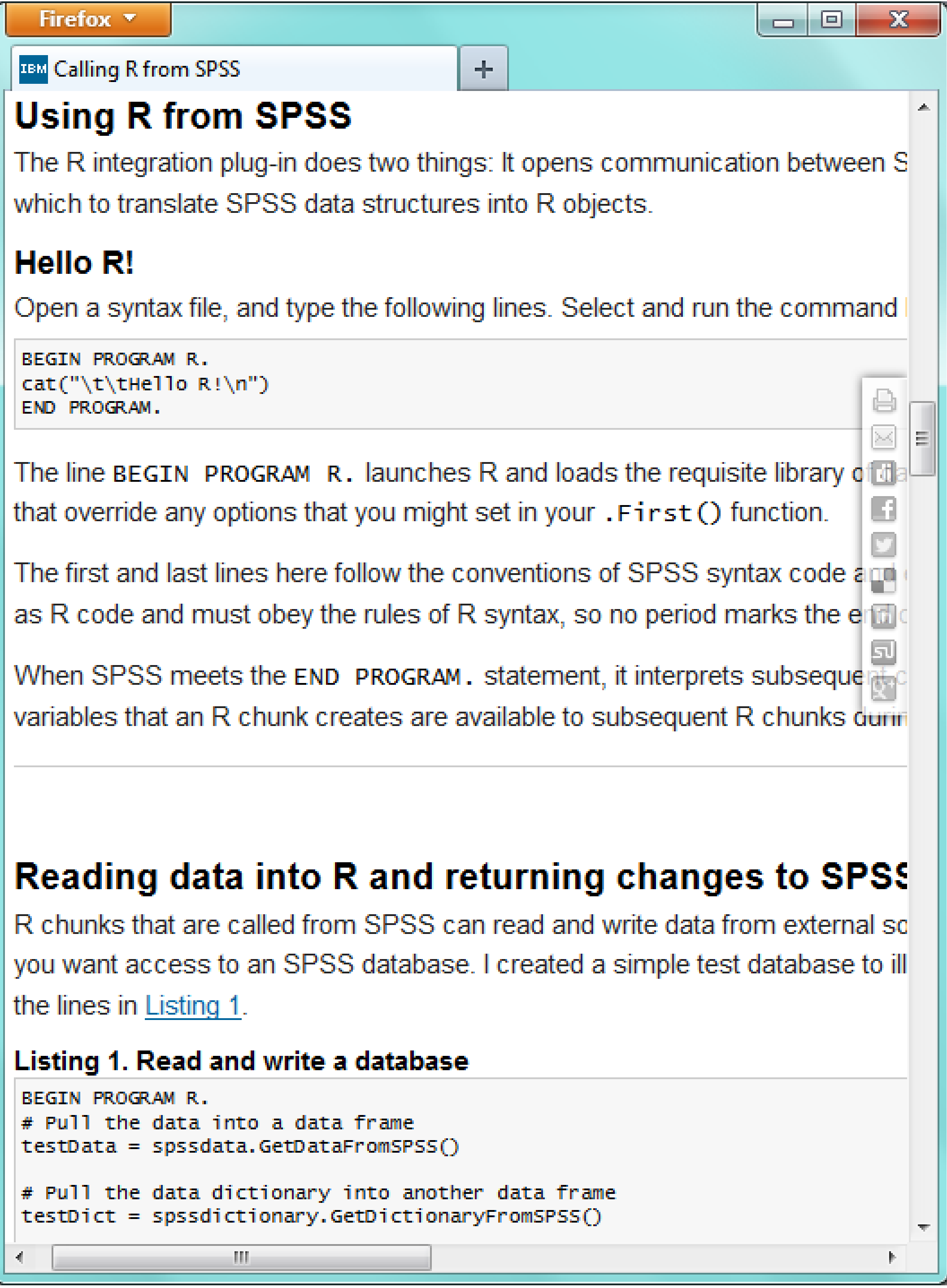
There are 3 steps here. First, provide the names of the input variables being sent to R. If you are sending the entire dataset (as a data frame), then type the name of this data frame in the second box of the parameters tab (*inputs: Edit Enumeration*). If you are sending only a few attributes from your data set, each attribute name has to be entered separately as shown below.

Edit Parameter List: inputs
This assigns each input port a variable name. If the type of input object is supported by the R translation, it will be accessible under this variable name.

name of variable
WT1
WT2
Date

The second step is to write the R script in the "*script: Edit text*" which is the first box of the parameters tab. Make sure that you reference the names of the variables selected above exactly (or rename them within R, as done below). In this case we are only using the attribute WT1 for forecasting along with the

R feat. SPSS



Using R from SPSS

The R integration plug-in does two things: It opens communication between S
which to translate SPSS data structures into R objects.

Hello R!

Open a syntax file, and type the following lines. Select and run the command

```
BEGIN PROGRAM R.
cat("\t\tHello R!\n")
END PROGRAM.
```

The line `BEGIN PROGRAM R.` launches R and loads the requisite library o
that override any options that you might set in your `.First()` function.

The first and last lines here follow the conventions of SPSS syntax code a
as R code and must obey the rules of R syntax, so no period marks the e

When SPSS meets the `END PROGRAM.` statement, it interprets subseque
variables that an R chunk creates are available to subsequent R chunks duri

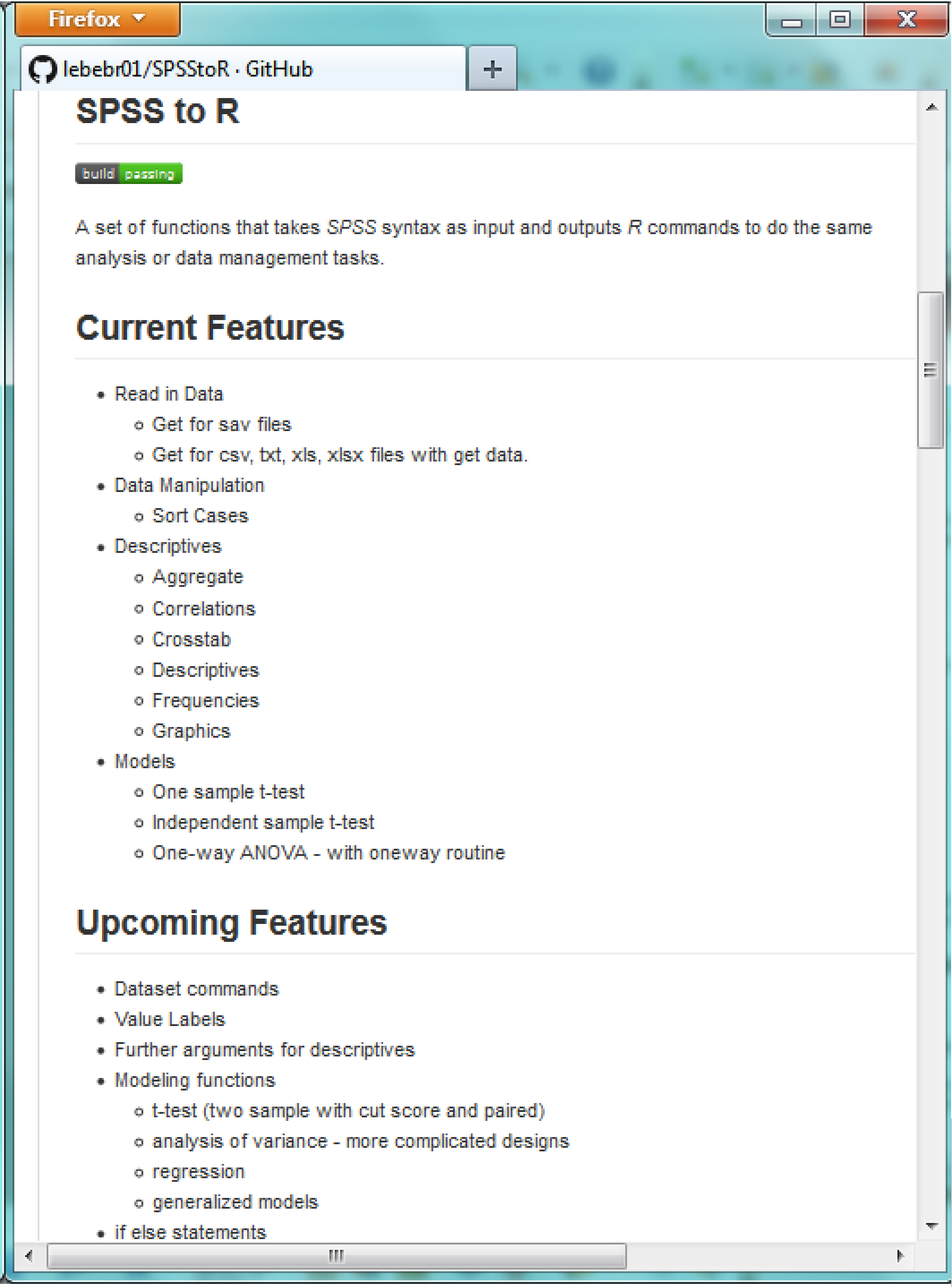
---

Reading data into R and returning changes to SPS

R chunks that are called from SPSS can read and write data from external so
you want access to an SPSS database. I created a simple test database to ill
the lines in Listing 1.

Listing 1. Read and write a database

```
BEGIN PROGRAM R.
# Pull the data into a data frame
testData = spssdata.GetDataFromSPSS()

# Pull the data dictionary into another data frame
testDict = spssdictionary.GetDictionaryFromSPSS()
```



SPSS to R

A set of functions that takes *SPSS* syntax as input and outputs *R* commands to do the same analysis or data management tasks.

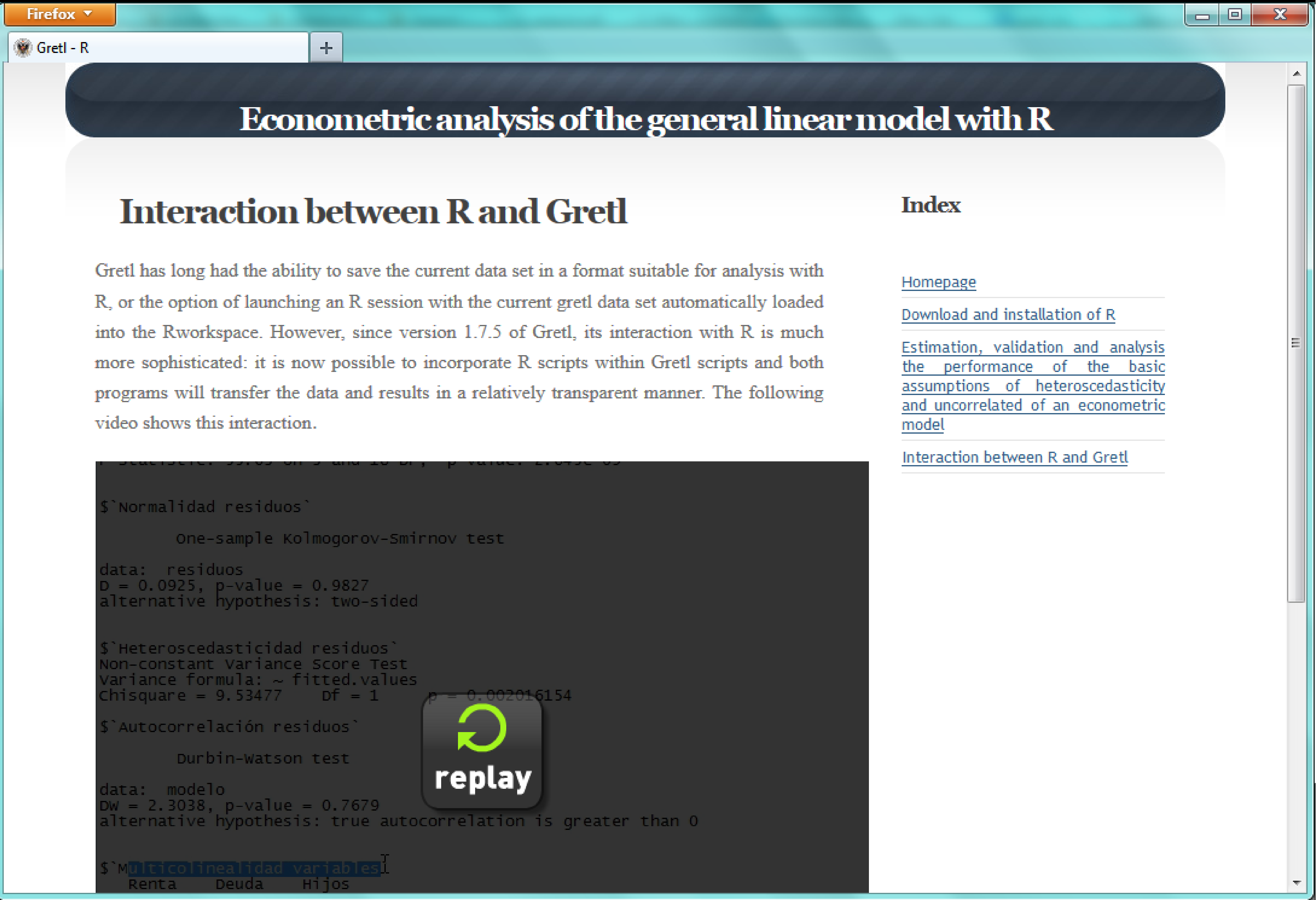
Current Features

- Read in Data
  - Get for sav files
  - Get for csv, txt, xls, xlsx files with get data.
- Data Manipulation
  - Sort Cases
- Descriptives
  - Aggregate
  - Correlations
  - Crosstab
  - Descriptives
  - Frequencies
  - Graphics
- Models
  - One sample t-test
  - Independent sample t-test
  - One-way ANOVA - with oneway routine

Upcoming Features

- Dataset commands
- Value Labels
- Further arguments for descriptives
- Modeling functions
  - t-test (two sample with cut score and paired)
  - analysis of variance - more complicated designs
  - regression
  - generalized models
- if else statements

R feat. Gretl



Econometric analysis of the general linear model with R

Interaction between R and Gretl

Gretl has long had the ability to save the current data set in a format suitable for analysis with R, or the option of launching an R session with the current gretl data set automatically loaded into the Rworkspace. However, since version 1.7.5 of Gretl, its interaction with R is much more sophisticated: it is now possible to incorporate R scripts within Gretl scripts and both programs will transfer the data and results in a relatively transparent manner. The following video shows this interaction.

```
$`Normalidad residuos`

        One-sample Kolmogorov-Smirnov test

data:  residuos
D = 0.0925, p-value = 0.9827
alternative hypothesis: two-sided


$`Heteroscedasticidad residuos`
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 9.53477    Df = 1     p = 0.002016154

$`Autocorrelación residuos`

        Durbin-Watson test

data:  modelo
DW = 2.3038, p-value = 0.7679
alternative hypothesis: true autocorrelation is greater than 0


$`Multicolinealidad variables`
   Renta     Deuda     Hijos
```

Index

- Homepage
- Download and installation of R
- Estimation, validation and analysis the performance of the basic assumptions of heteroscedasticity and uncorrelated of an econometric model
- Interaction between R and Gretl

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is (widely) supported by the world of science

IV R is supported by the community

V ½ :) R and SAS

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

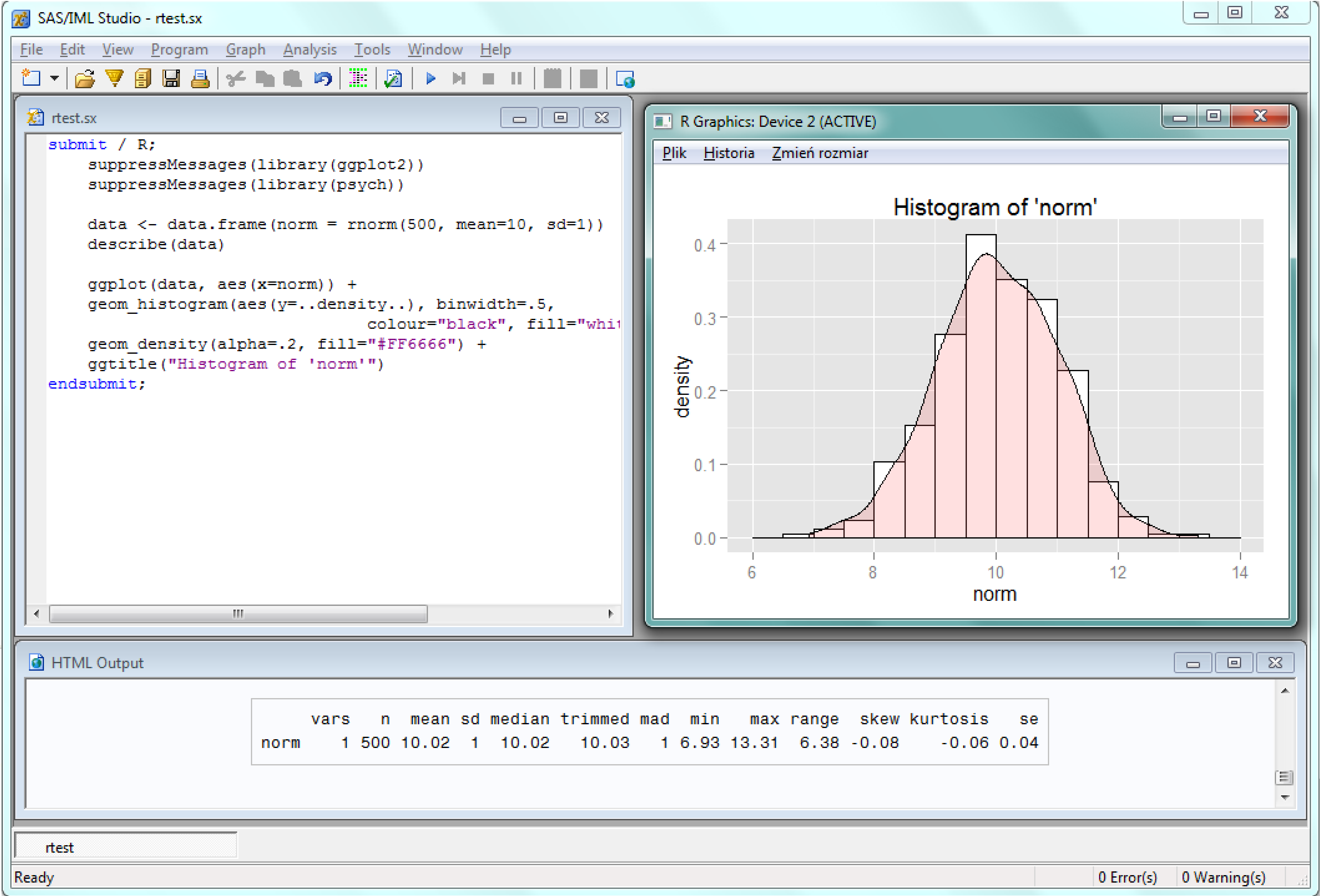
X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

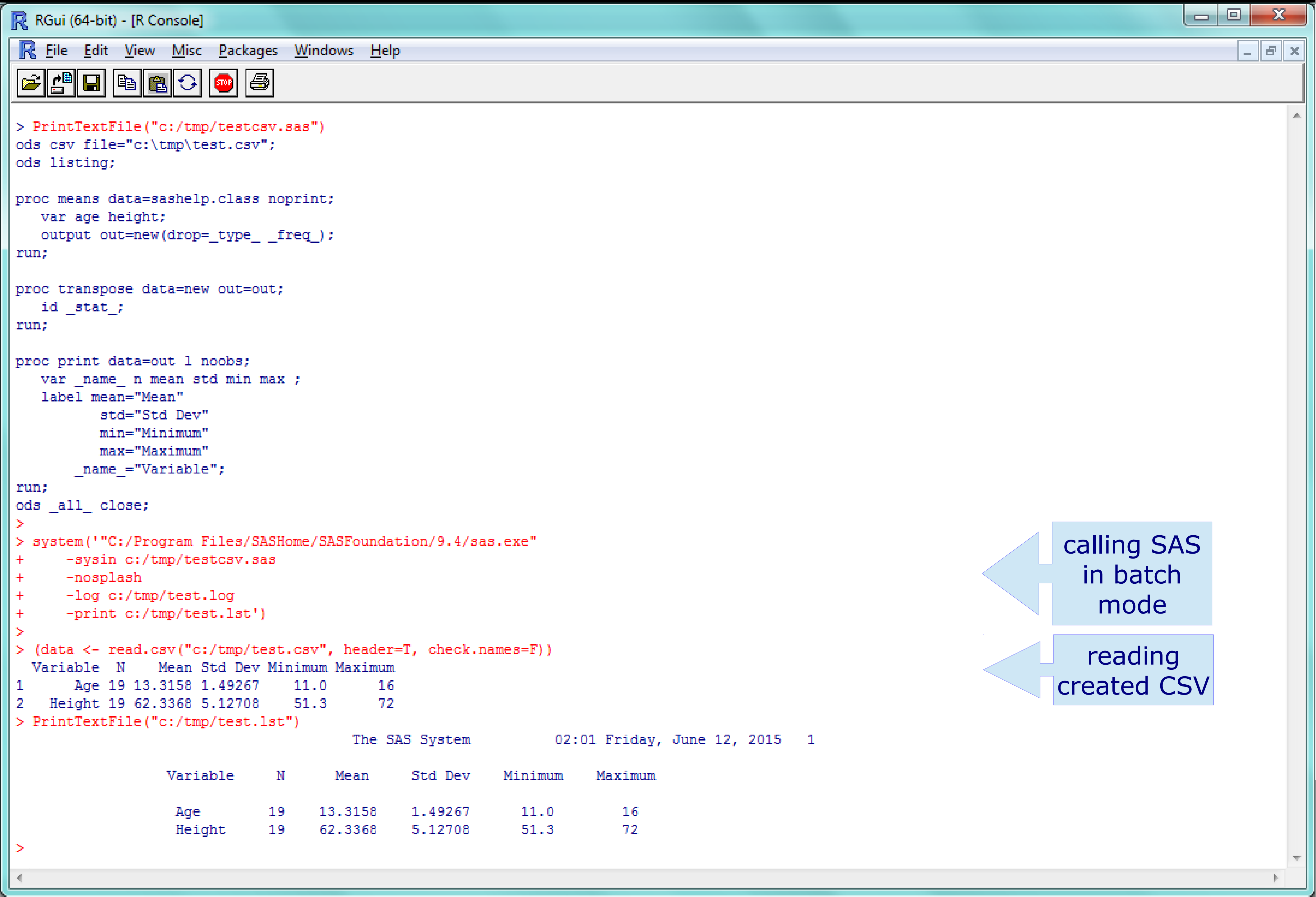
R feat. SAS



```
submit / R;
    suppressMessages(library(ggplot2))
    suppressMessages(library(psych))

    data <- data.frame(norm = rnorm(500, mean=10, sd=1))
    describe(data)

    ggplot(data, aes(x=norm)) +
    geom_histogram(aes(y=..density..), binwidth=.5,
                            colour="black", fill="whi
    geom_density(alpha=.2, fill="#FF6666") +
    ggtitle("Histogram of 'norm'")
endsubmit;
```

```
      vars   n  mean sd median trimmed mad  min   max range  skew kurtosis   se
norm     1 500 10.02  1  10.02   10.03   1 6.93 13.31  6.38 -0.08    -0.06 0.04
```

R feat. SAS

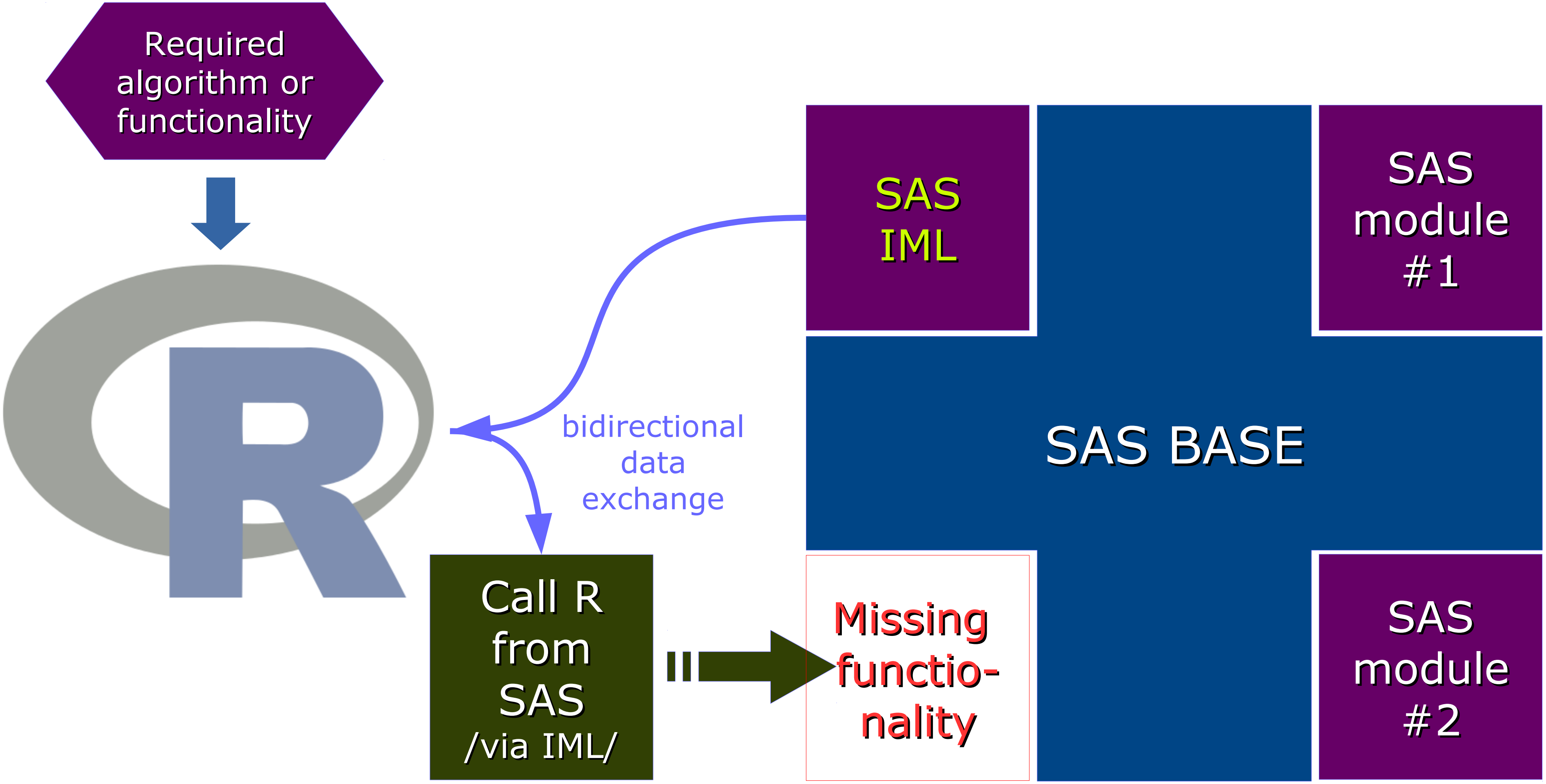


```
> PrintTextFile("c:/tmp/testcsv.sas")
ods csv file="c:\tmp\test.csv";
ods listing;

proc means data=sashelp.class noprint;
   var age height;
   output out=new(drop=_type_ _freq_);
run;

proc transpose data=new out=out;
   id _stat_;
run;

proc print data=out l noobs;
   var _name_ n mean std min max ;
   label mean="Mean"
          std="Std Dev"
          min="Minimum"
          max="Maximum"
       _name_="Variable";
run;
ods _all_ close;
>
> system('"C:/Program Files/SASHome/SASFoundation/9.4/sas.exe"
+     -sysin c:/tmp/testcsv.sas
+     -nosplash
+     -log c:/tmp/test.log
+     -print c:/tmp/test.lst')
>
> (data <- read.csv("c:/tmp/test.csv", header=T, check.names=F))
  Variable  N    Mean Std Dev Minimum Maximum
1      Age 19 13.3158 1.49267    11.0      16
2   Height 19 62.3368 5.12708    51.3      72
> PrintTextFile("c:/tmp/test.lst")
                                        The SAS System          02:01 Friday, June 12, 2015   1

                  Variable     N      Mean     Std Dev    Minimum    Maximum

                   Age        19    13.3158    1.49267      11.0        16
                   Height     19    62.3368    5.12708      51.3        72
>
```

calling SAS in batch mode

reading created CSV

Enhance SAS with R via IML



Required algorithm or functionality

bidirectional data exchange

SAS IML

SAS module #1

SAS BASE

Call R from SAS /via IML/

Missing functio-nality

SAS module #2

SAS – R companion

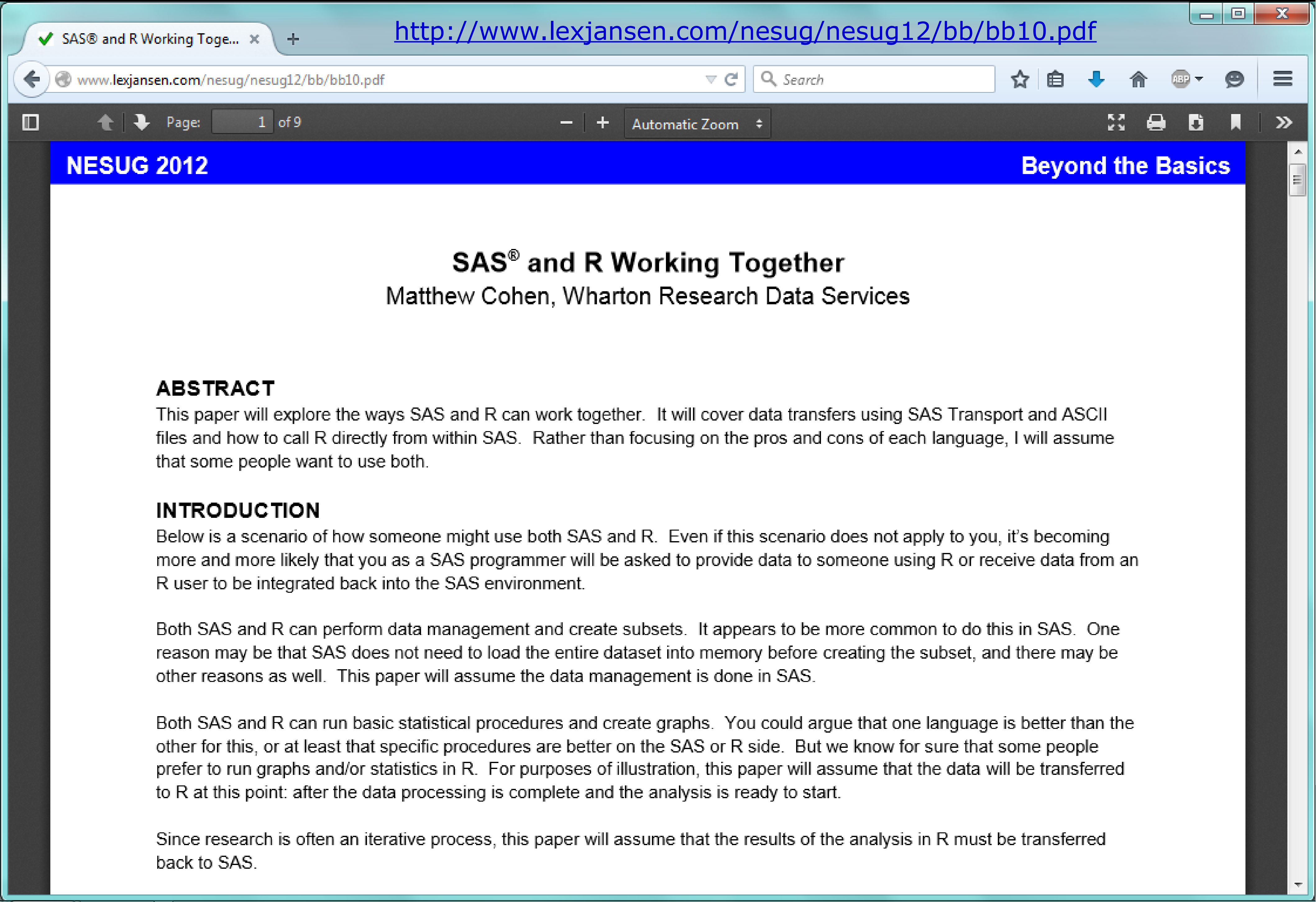
A set of factors makes the cooperation between SAS and R easier:

- Package sas7bdat enables R reading SAS SAS7BDAT datasets without having SAS installed
- Package SASxport enables R both reading and writing SAS Transport files (XPT) without having SAS installed.

  It is removed from CRAN yet still available in the CRAN Archive. I use this package extensively and have never been disappointed.
- Package foreign is another option for exchanging data with SAS
- For several statistical methods SAS-compliant variants are created:
  - Contrast: contr.SAS
  - Quantiles: type 3 → *nearest even order statistic*
  - Type-III Sum of Square is available in R
  - Both SAS and R share the same numbers rounding algorithm
- Both R and SAS can call each other in batch mode from a command line
- R can also be accessed conveniently from SAS via IML module

SAS – R companion

http://www.lexjansen.com/nesug/nesug12/bb/bb10.pdf



NESUG 2012 **Beyond the Basics**

SAS® and R Working Together

Matthew Cohen, Wharton Research Data Services

ABSTRACT

This paper will explore the ways SAS and R can work together. It will cover data transfers using SAS Transport and ASCII files and how to call R directly from within SAS. Rather than focusing on the pros and cons of each language, I will assume that some people want to use both.

INTRODUCTION

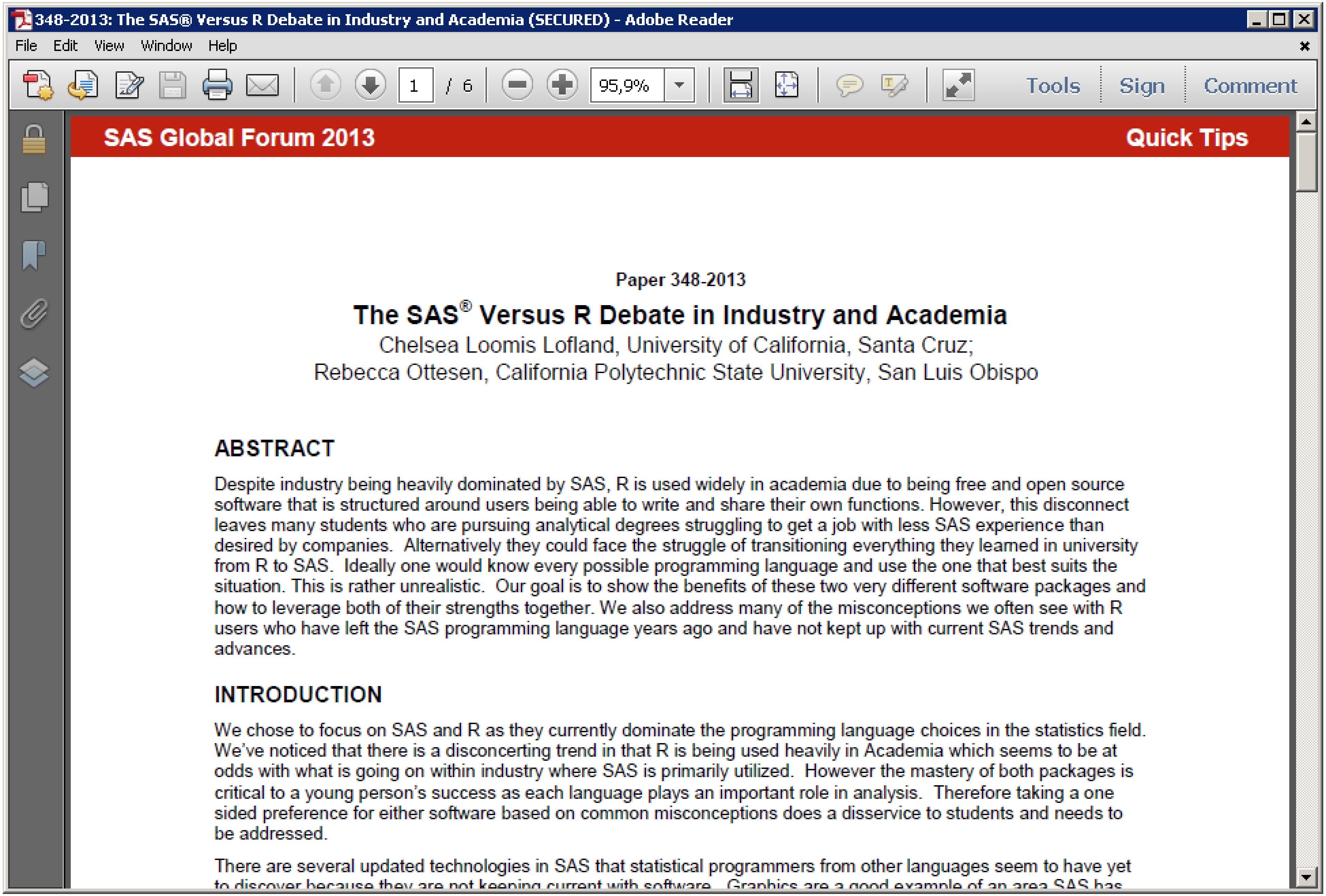
Below is a scenario of how someone might use both SAS and R. Even if this scenario does not apply to you, it's becoming more and more likely that you as a SAS programmer will be asked to provide data to someone using R or receive data from an R user to be integrated back into the SAS environment.

Both SAS and R can perform data management and create subsets. It appears to be more common to do this in SAS. One reason may be that SAS does not need to load the entire dataset into memory before creating the subset, and there may be other reasons as well. This paper will assume the data management is done in SAS.

Both SAS and R can run basic statistical procedures and create graphs. You could argue that one language is better than the other for this, or at least that specific procedures are better on the SAS or R side. But we know for sure that some people prefer to run graphs and/or statistics in R. For purposes of illustration, this paper will assume that the data will be transferred to R at this point: after the data processing is complete and the analysis is ready to start.

Since research is often an iterative process, this paper will assume that the results of the analysis in R must be transferred back to SAS.

Let's MoRtal Kombat begin!



SAS Global Forum 2013 Quick Tips

Paper 348-2013

The SAS® Versus R Debate in Industry and Academia

Chelsea Loomis Lofland, University of California, Santa Cruz;
Rebecca Ottesen, California Polytechnic State University, San Luis Obispo

ABSTRACT

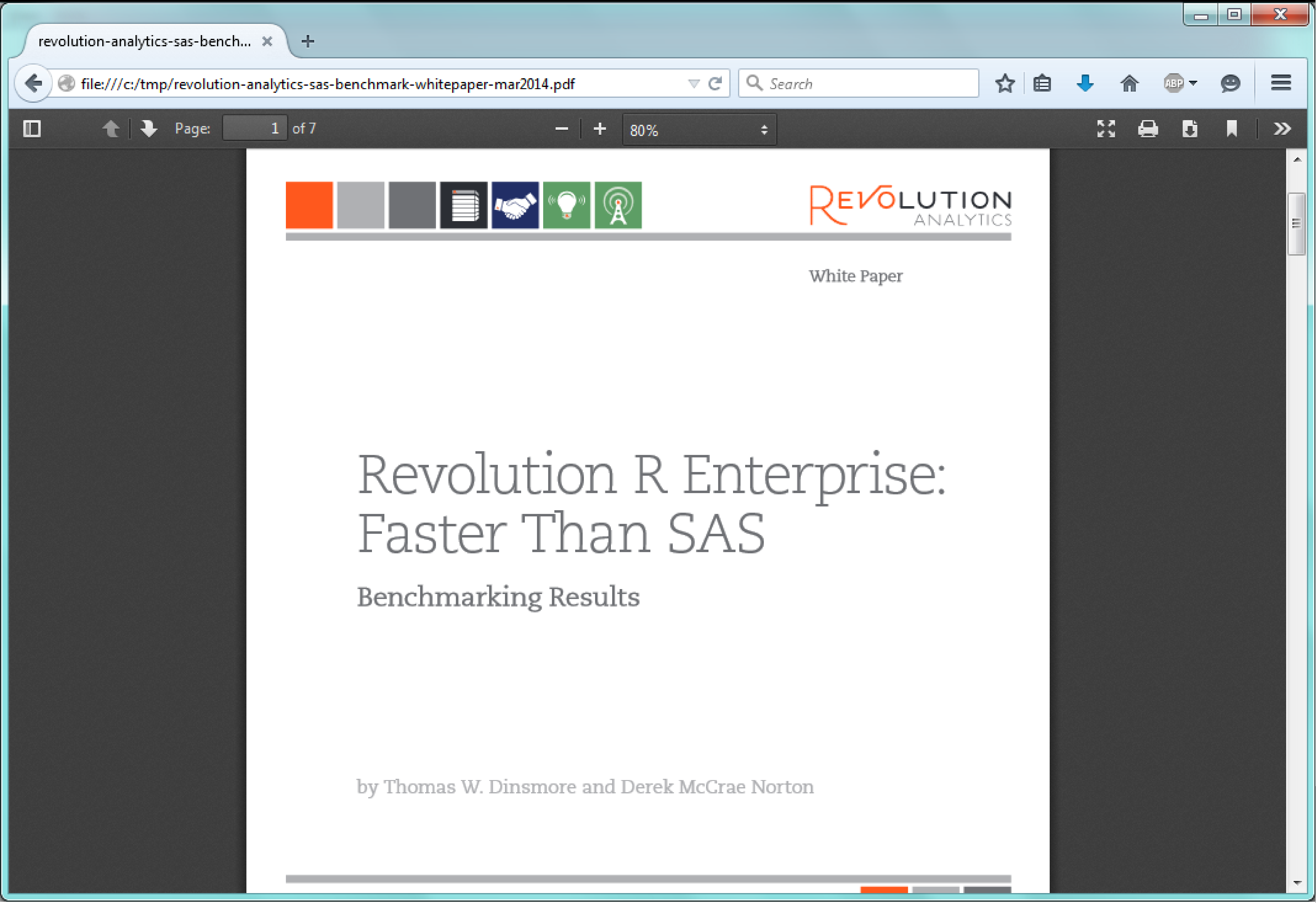
Despite industry being heavily dominated by SAS, R is used widely in academia due to being free and open source software that is structured around users being able to write and share their own functions. However, this disconnect leaves many students who are pursuing analytical degrees struggling to get a job with less SAS experience than desired by companies. Alternatively they could face the struggle of transitioning everything they learned in university from R to SAS. Ideally one would know every possible programming language and use the one that best suits the situation. This is rather unrealistic. Our goal is to show the benefits of these two very different software packages and how to leverage both of their strengths together. We also address many of the misconceptions we often see with R users who have left the SAS programming language years ago and have not kept up with current SAS trends and advances.

INTRODUCTION

We chose to focus on SAS and R as they currently dominate the programming language choices in the statistics field. We've noticed that there is a disconcerting trend in that R is being used heavily in Academia which seems to be at odds with what is going on within industry where SAS is primarily utilized. However the mastery of both packages is critical to a young person's success as each language plays an important role in analysis. Therefore taking a one sided preference for either software based on common misconceptions does a disservice to students and needs to be addressed.

There are several updated technologies in SAS that statistical programmers from other languages seem to have yet to discover because they are not keeping current with software. Graphics are a good example of an area SAS has

Let's MoRtal Kombat begin!



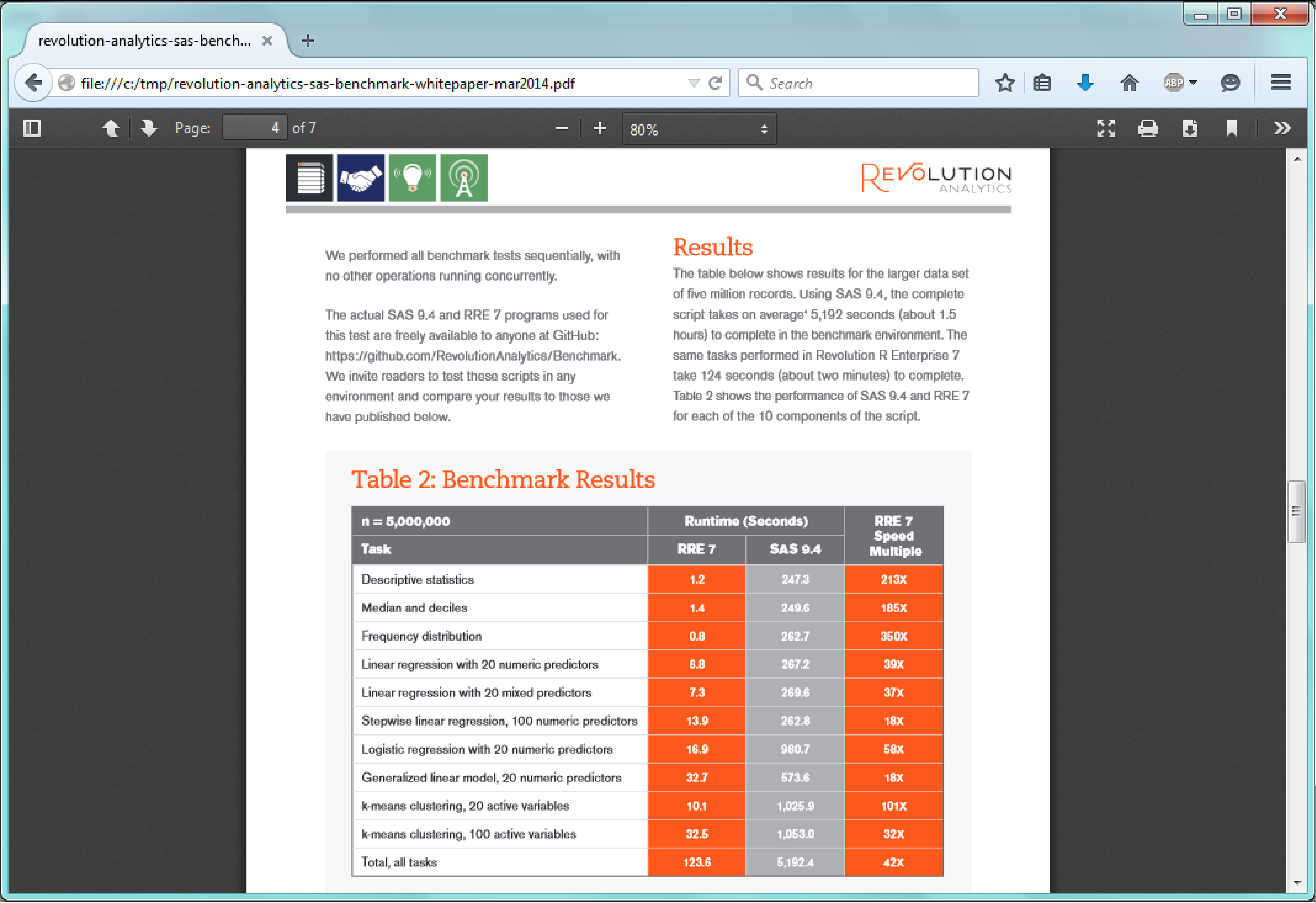
White Paper

Revolution R Enterprise: Faster Than SAS

Benchmarking Results

by Thomas W. Dinsmore and Derek McCrae Norton

Let's MoRtal Kombat begin!



We performed all benchmark tests sequentially, with no other operations running concurrently.

The actual SAS 9.4 and RRE 7 programs used for this test are freely available to anyone at GitHub: https://github.com/RevolutionAnalytics/Benchmark. We invite readers to test these scripts in any environment and compare your results to those we have published below.

Results

The table below shows results for the larger data set of five million records. Using SAS 9.4, the complete script takes on average[1] 5,192 seconds (about 1.5 hours) to complete in the benchmark environment. The same tasks performed in Revolution R Enterprise 7 take 124 seconds (about two minutes) to complete. Table 2 shows the performance of SAS 9.4 and RRE 7 for each of the 10 components of the script.

Table 2: Benchmark Results

n = 5,000,000	Runtime (Seconds)		RRE 7 Speed Multiple
Task	RRE 7	SAS 9.4	
Descriptive statistics	1.2	247.3	213X
Median and deciles	1.4	249.6	185X
Frequency distribution	0.8	262.7	350X
Linear regression with 20 numeric predictors	6.8	267.2	39X
Linear regression with 20 mixed predictors	7.3	269.6	37X
Stepwise linear regression, 100 numeric predictors	13.9	262.8	18X
Logistic regression with 20 numeric predictors	16.9	980.7	58X
Generalized linear model, 20 numeric predictors	32.7	573.6	18X
k-means clustering, 20 active variables	10.1	1,025.9	101X
k-means clustering, 100 active variables	32.5	1,053.0	32X
Total, all tasks	123.6	5,192.4	42X

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is (widely) supported by the world of science

IV R is supported by the community

V R is (increasingly) supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

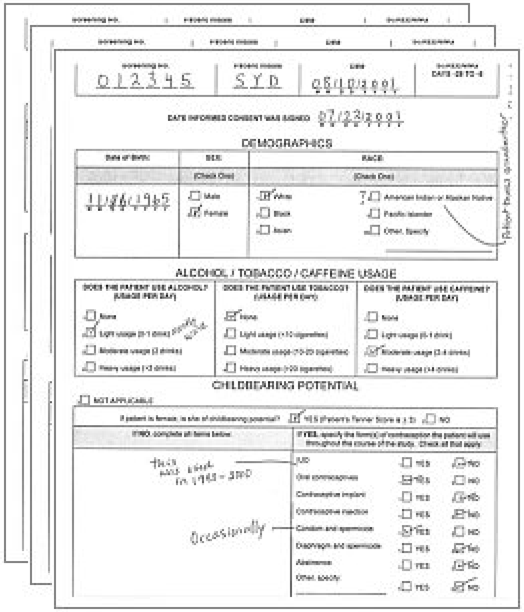
XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

Does R speak XYZ ?

R is able to read data in many formats: MS Excel, OpenOffice Calc, Gnumeric, DICOM, SPSS, Weka, Systat, Stata, EpiInfo, SAS datasets, SAS Transport, Systat, Minitab, Octave, Matlab, DBF (dBase, FoxPro), CSV/TSV, XML, HTML, JSON and...

One can access data stored in various databases via native drivers (MySQL, PostgreSQL, SQLite, MS Access) as well as ODBC and JDBC interfaces. This gives easy access to any database for which appropriate drivers are available: Oracle, SQL Server, MySQL, dBase, PostgreSQL, SQLite, DB/2, Informix, Firebird, etc.



Does R speak XYZ ?

```
> read.csv("c:/tmp/data.csv", header=TRUE, sep=";")
  Column.1 Column.2   Column.3
1        1    First 2011-01-03
2        2   Second 2011-01-04
3        3    Third 2011-01-05
4        4   Fourth 2011-01-06
5        5    Fifth 2011-01-07
```

CSV

```
> sqldf::read.csv.sql("c:/tmp/data.csv", sql="SELECT * FROM file f WHERE f.`Column.1` BETWEEN 2 AND 3",  sep=";")
  Column.1 Column.2   Column.3
1        2   Second 2011-01-04
2        3    Third 2011-01-05
```

CSV^{SQL}

```
> openxlsx::read.xlsx("c:/tmp/data.xlsx", sheet=1, colNames=TRUE, detectDates = TRUE)
  Column.1 Column.2   Column.3
1        1    First 2011-01-03
2        2   Second 2011-01-04
3        3    Third 2011-01-05
4        4   Fourth 2011-01-06
5        5    Fifth 2011-01-07
```

Excel

```
> data <- read.ods("c:/tmp/data.ods", sheet=1)
> colnames(data) <- data[1, ]
> data[-1, ]
  Column 1 Column 2   Column 3
2        1    First 2011-01-03
3        2   Second 2011-01-04
4        3    Third 2011-01-05
5        4   Fourth 2011-01-06
6        5    Fifth 2011-01-0
```

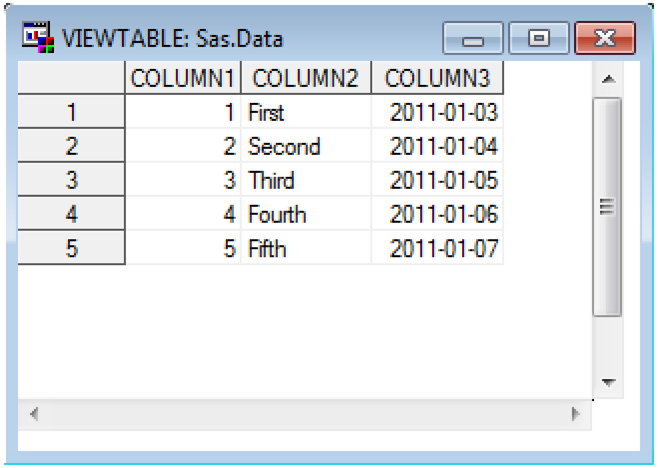
Calc

Does R speak XYZ ?

```
> read.xport("c:/tmp/data.xpt")
  COLUMN1 COLUMN2    COLUMN3
1       1   First 2011-01-03
2       2  Second 2011-01-04
3       3   Third 2011-01-05
4       4  Fourth 2011-01-06
5       5   Fifth 2011-01-07
```

SAS
Transport

```
> data=read.sas7bdat("c:/tmp/data.sas7bdat")
> data$COLUMN4 <- as.Date(data$COLUMN3, origin = "1960-01-01")
> data
  COLUMN1 COLUMN2    COLUMN3
1       1   First 2011-01-03
2       2  Second 2011-01-04
3       3   Third 2011-01-05
4       4  Fourth 2011-01-06
5       5   Fifth 2011-01-07
```

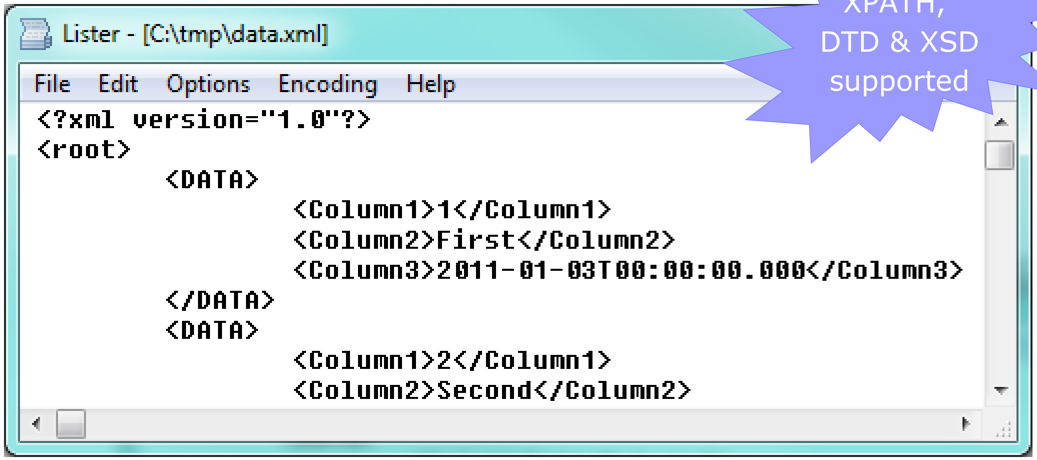


VIEWTABLE: Sas.Data

	COLUMN1	COLUMN2	COLUMN3
1	1	First	2011-01-03
2	2	Second	2011-01-04
3	3	Third	2011-01-05
4	4	Fourth	2011-01-06
5	5	Fifth	2011-01-07

SAS 7

```
> xml <- xmlParse("c:/tmp/data.xml")
> xmlToDataFrame(xml, homogeneous = TRUE, collectNames = TRUE, colClasses = c("integer", "character", "character"))
> data$Column3 <- as.Date(data$Column3)
> data
  Column1 Column2    Column3
1       1   First 2011-01-03
2       2  Second 2011-01-04
3       3   Third 2011-01-05
4       4  Fourth 2011-06-03
5       5   Fifth 2011-01-07
```



Lister - [C:\tmp\data.xml]

```
<?xml version="1.0"?>
<root>
        <DATA>
                <Column1>1</Column1>
                <Column2>First</Column2>
                <Column3>2011-01-03T00:00:00.000</Column3>
        </DATA>
        <DATA>
                <Column1>2</Column1>
                <Column2>Second</Column2>
```

XPATH,
DTD & XSD
supported

XML

Does R speak XYZ ?

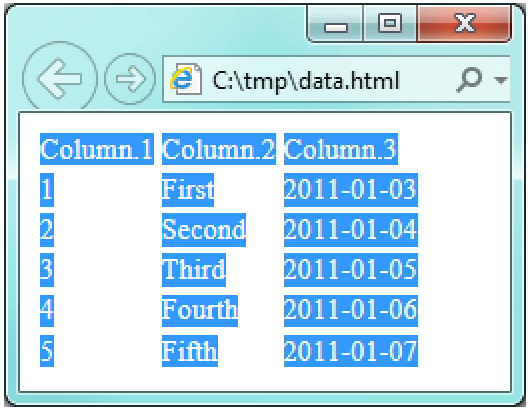
```
> readHTMLTable("c:/tmp/data.html", header =T, which = 1)
  Column 1 Column 2   Column 3
1        1    first 2011-01-03
2        2   second 2011-01-04
3        3    third 2011-01-05
4        4   fourth 2011-01-06
5        5    fifth 2011-01-07
```

HTML

```
> json
[1] "{\"Column 1\":[\"1\",\"2\",\"3\",\"4\",\"5\"],\"Column 2\":
[\"First\",\"Second\",\"Third\",\"Fourth\",\"Fifth\"],\"Column 3\":[\"2011-01-03\",\"2011-01-04\",\"2011-01-05\",\"2011-01-
06\",\"2011-01-07\"]}"
> as.data.frame(fromJSON(json))
  Column.1 Column.2   Column.3
1        1    First 2011-01-03
2        2   Second 2011-01-04
3        3    Third 2011-01-05
4        4   Fourth 2011-01-06
5        5    Fifth 2011-01-07
```

JSON

```
> read.table(file = "clipboard", sep = " ", header=TRUE)    (# with some artefacts)
  Column.1 Column.2   Column.3  X
1        1    First 2011-01-03 NA
2        2   Second 2011-01-04 NA
3        3    Third 2011-01-05 NA
4        4   Fourth 2011-01-06 NA
5        5    Fifth 2011-01-07 NA
```



Column.1	Column.2	Column.3
1	First	2011-01-03
2	Second	2011-01-04
3	Third	2011-01-05
4	Fourth	2011-01-06
5	Fifth	2011-01-07

Clipboard

PRocessing Excel spreadsheets

- RODBC - in SQL manner
- XLConnect - requires Java
- xlsx - requires Java
- openxlsx - Java-free, fast, requires ZIP
- gdata - portable, requires PERL
- RExcel - COM based, licensed
- By using external code
- From clipboard via read.table(“clipboard”)

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is (widely) supported by the world of science

IV R is supported by the community

V R is (increasingly) supported by the business

VI 1/3 :) R and relational databases

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

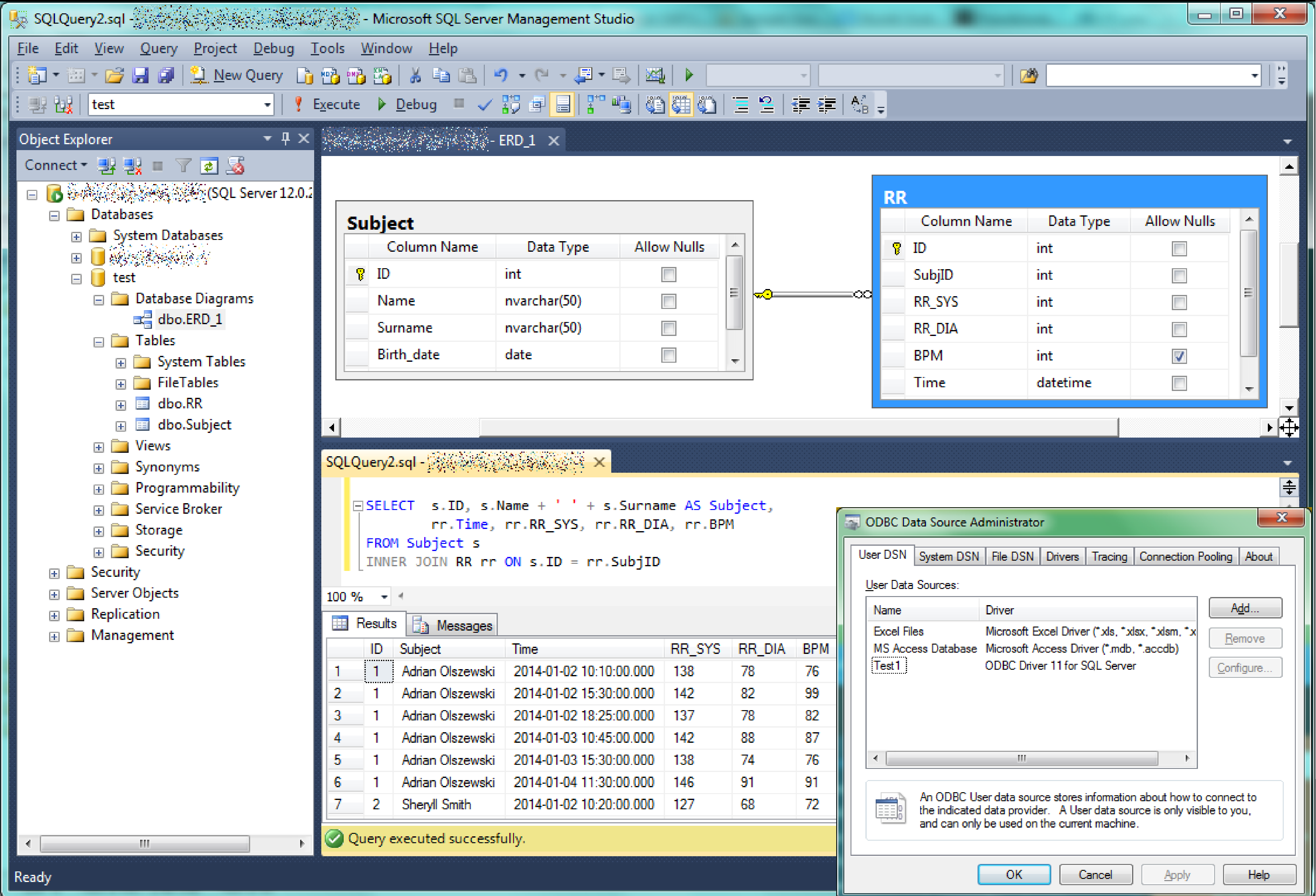
X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

The R and SQL companion



Subject

Column Name	Data Type	Allow Nulls
ID	int	☐
Name	nvarchar(50)	☐
Surname	nvarchar(50)	☐
Birth_date	date	☐

RR

Column Name	Data Type	Allow Nulls
ID	int	☐
SubjID	int	☐
RR_SYS	int	☐
RR_DIA	int	☐
BPM	int	☑
Time	datetime	☐

```
SELECT  s.ID, s.Name + ' ' + s.Surname AS Subject,
        rr.Time, rr.RR_SYS, rr.RR_DIA, rr.BPM
FROM Subject s
INNER JOIN RR rr ON s.ID = rr.SubjID
```

	ID	Subject	Time	RR_SYS	RR_DIA	BPM
1	1	Adrian Olszewski	2014-01-02 10:10:00.000	138	78	76
2	1	Adrian Olszewski	2014-01-02 15:30:00.000	142	82	99
3	1	Adrian Olszewski	2014-01-02 18:25:00.000	137	78	82
4	1	Adrian Olszewski	2014-01-03 10:45:00.000	142	88	87
5	1	Adrian Olszewski	2014-01-03 15:30:00.000	138	74	76
6	1	Adrian Olszewski	2014-01-04 11:30:00.000	146	91	91
7	2	Sheryll Smith	2014-01-02 10:20:00.000	127	68	72

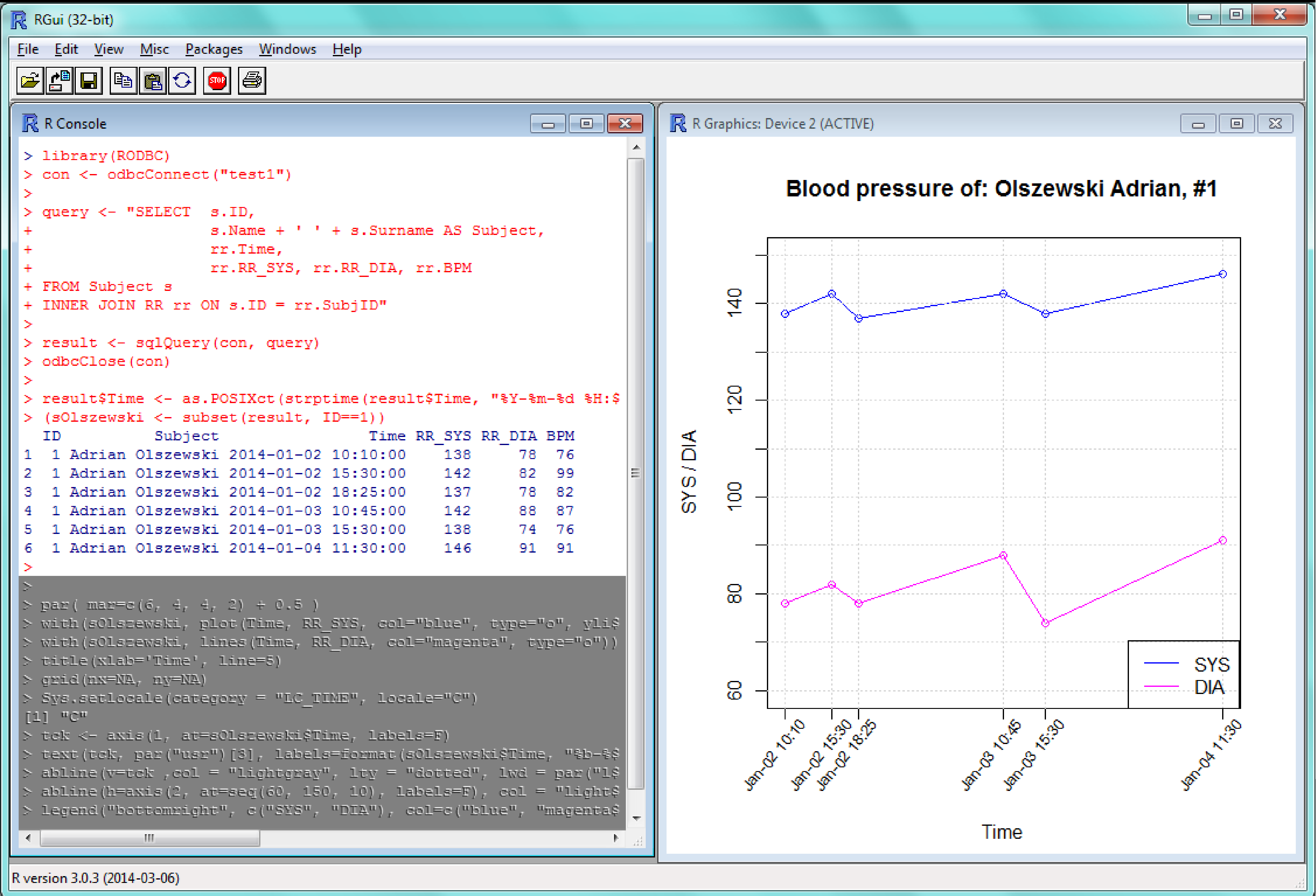
Query executed successfully.

User Data Sources:

Name	Driver
Excel Files	Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.x
MS Access Database	Microsoft Access Driver (*.mdb, *.accdb)
Test1	ODBC Driver 11 for SQL Server

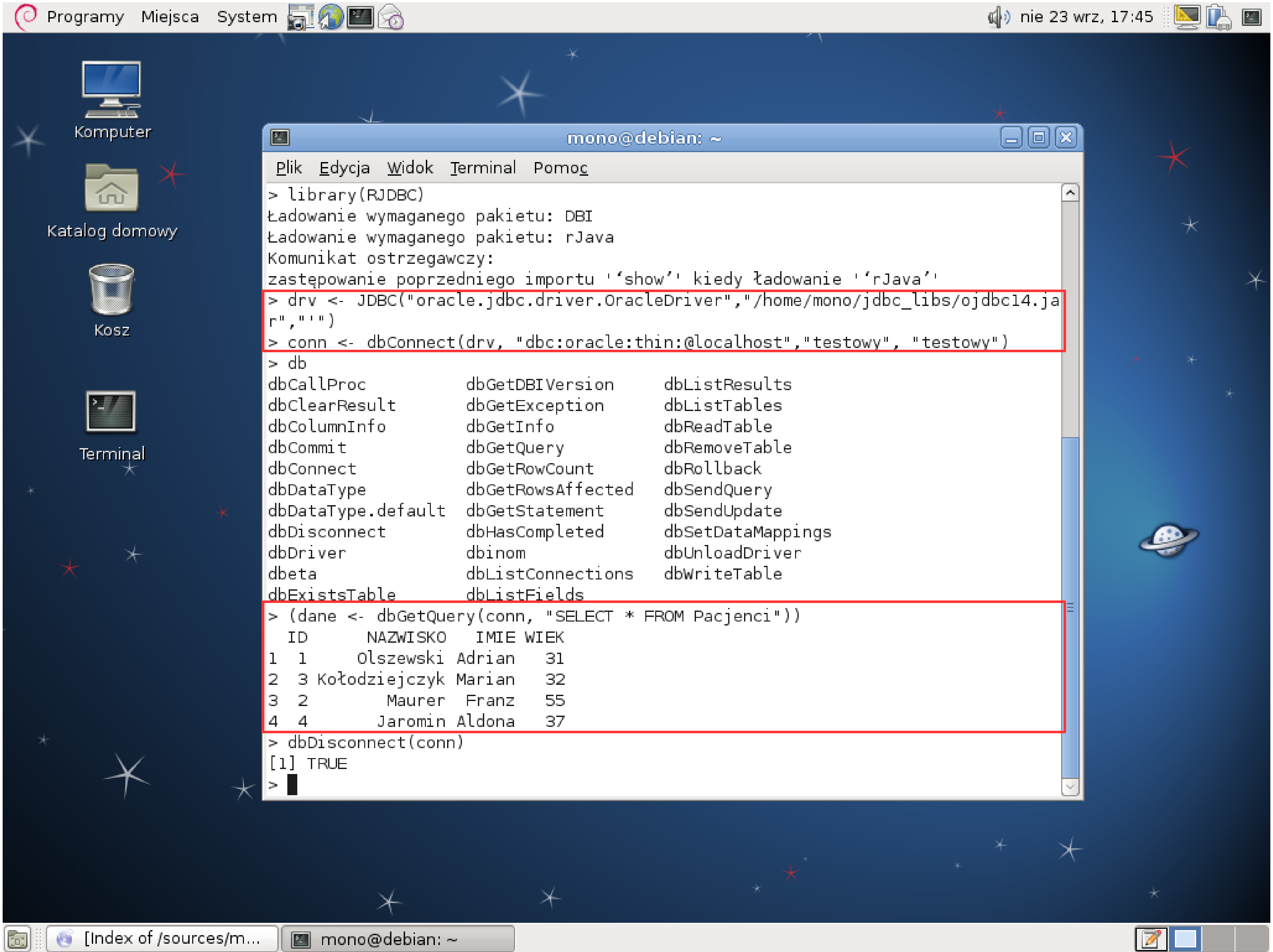
An ODBC User data source stores information about how to connect to the indicated data provider. A User data source is only visible to you, and can only be used on the current machine.

Querying data sources via ODBC ...



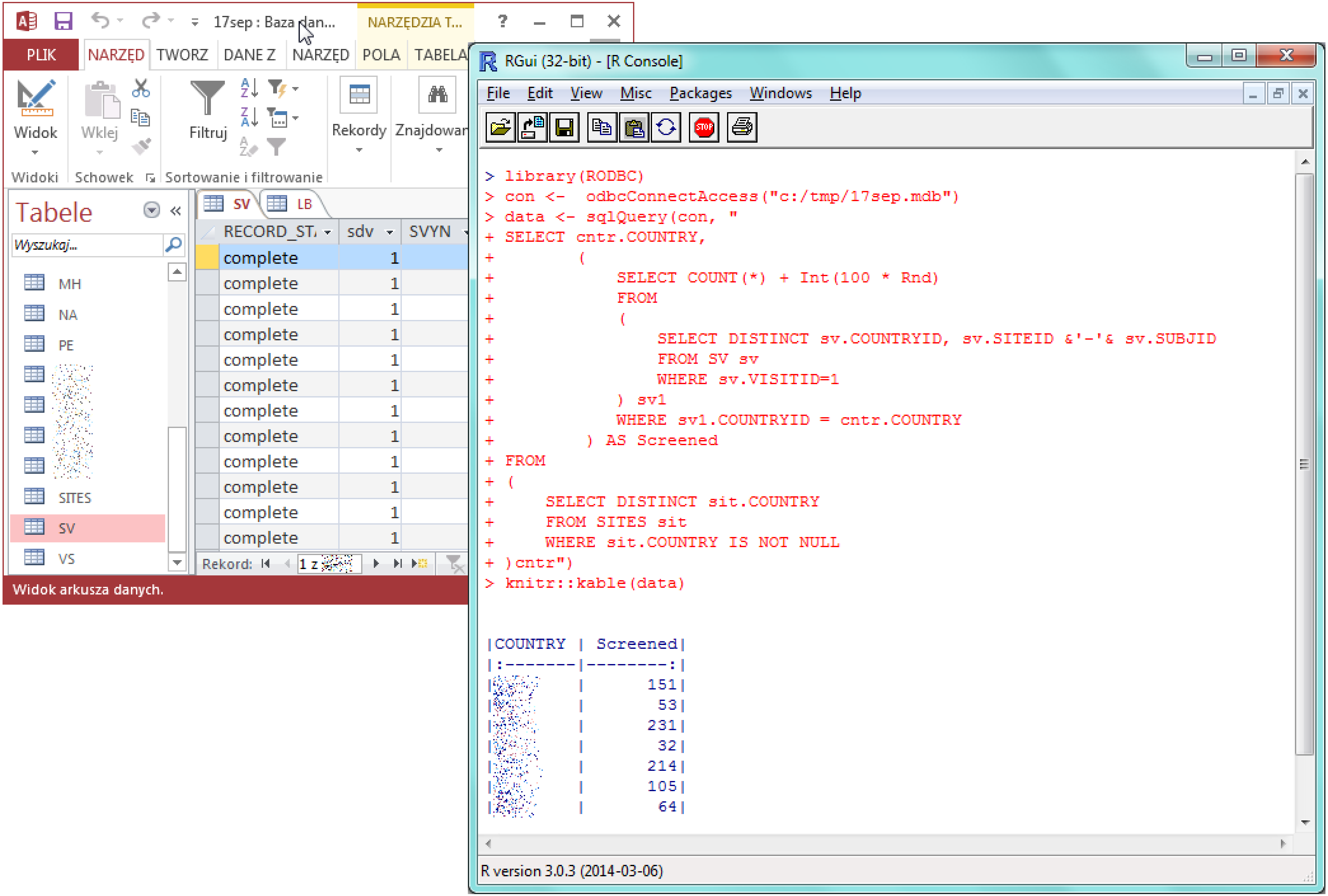
```
> library(RODBC)
> con <- odbcConnect("test1")
>
> query <- "SELECT  s.ID,
+                   s.Name + ' ' + s.Surname AS Subject,
+                   rr.Time,
+                   rr.RR_SYS, rr.RR_DIA, rr.BPM
+ FROM Subject s
+ INNER JOIN RR rr ON s.ID = rr.SubjID"
>
> result <- sqlQuery(con, query)
> odbcClose(con)
>
> result$Time <- as.POSIXct(strptime(result$Time, "%Y-%m-%d %H:$
> (sOlszewski <- subset(result, ID==1))
  ID          Subject                Time RR_SYS RR_DIA BPM
1  1 Adrian Olszewski 2014-01-02 10:10:00    138     78  76
2  1 Adrian Olszewski 2014-01-02 15:30:00    142     82  99
3  1 Adrian Olszewski 2014-01-02 18:25:00    137     78  82
4  1 Adrian Olszewski 2014-01-03 10:45:00    142     88  87
5  1 Adrian Olszewski 2014-01-03 15:30:00    138     74  76
6  1 Adrian Olszewski 2014-01-04 11:30:00    146     91  91
>
>
> par( mar=c(6, 4, 4, 2) + 0.5 )
> with(sOlszewski, plot(Time, RR_SYS, col="blue", type="o", yli$
> with(sOlszewski, lines(Time, RR_DIA, col="magenta", type="o"))
> title(xlab='Time', line=5)
> grid(nx=NA, ny=NA)
> Sys.setlocale(category = "LC_TIME", locale="C")
[1] "C"
> tck <- axis(1, at=sOlszewski$Time, labels=F)
> text(tck, par("usr")[3], labels=format(sOlszewski$Time, "%b-%$
> abline(v=tck ,col = "lightgray", lty = "dotted", lwd = par("l$
> abline(h=axis(2, at=seq(60, 150, 10), labels=F), col = "light$
> legend("bottomright", c("SYS", "DIA"), col=c("blue", "magenta$
```

… or JDBC …



```
> library(RJDBC)
Ładowanie wymaganego pakietu: DBI
Ładowanie wymaganego pakietu: rJava
Komunikat ostrzegawczy:
zastępowanie poprzedniego importu '‘show’' kiedy ładowanie '‘rJava’'
> drv <- JDBC("oracle.jdbc.driver.OracleDriver","/home/mono/jdbc_libs/ojdbc14.ja
r","'")
> conn <- dbConnect(drv, "dbc:oracle:thin:@localhost","testowy", "testowy")
> db
dbCallProc          dbGetDBIVersion     dbListResults
dbClearResult       dbGetException      dbListTables
dbColumnInfo        dbGetInfo           dbReadTable
dbCommit            dbGetQuery          dbRemoveTable
dbConnect           dbGetRowCount       dbRollback
dbDataType          dbGetRowsAffected   dbSendQuery
dbDataType.default  dbGetStatement      dbSendUpdate
dbDisconnect        dbHasCompleted      dbSetDataMappings
dbDriver            dbinom              dbUnloadDriver
dbeta               dbListConnections   dbWriteTable
dbExistsTable       dbListFields
> (dane <- dbGetQuery(conn, "SELECT * FROM Pacjenci"))
  ID      NAZWISKO  IMIE WIEK
1  1     Olszewski Adrian   31
2  3 Kołodziejczyk Marian   32
3  2        Maurer  Franz   55
4  4       Jaromin Aldona   37
> dbDisconnect(conn)
[1] TRUE
> 
```

… or diRect …

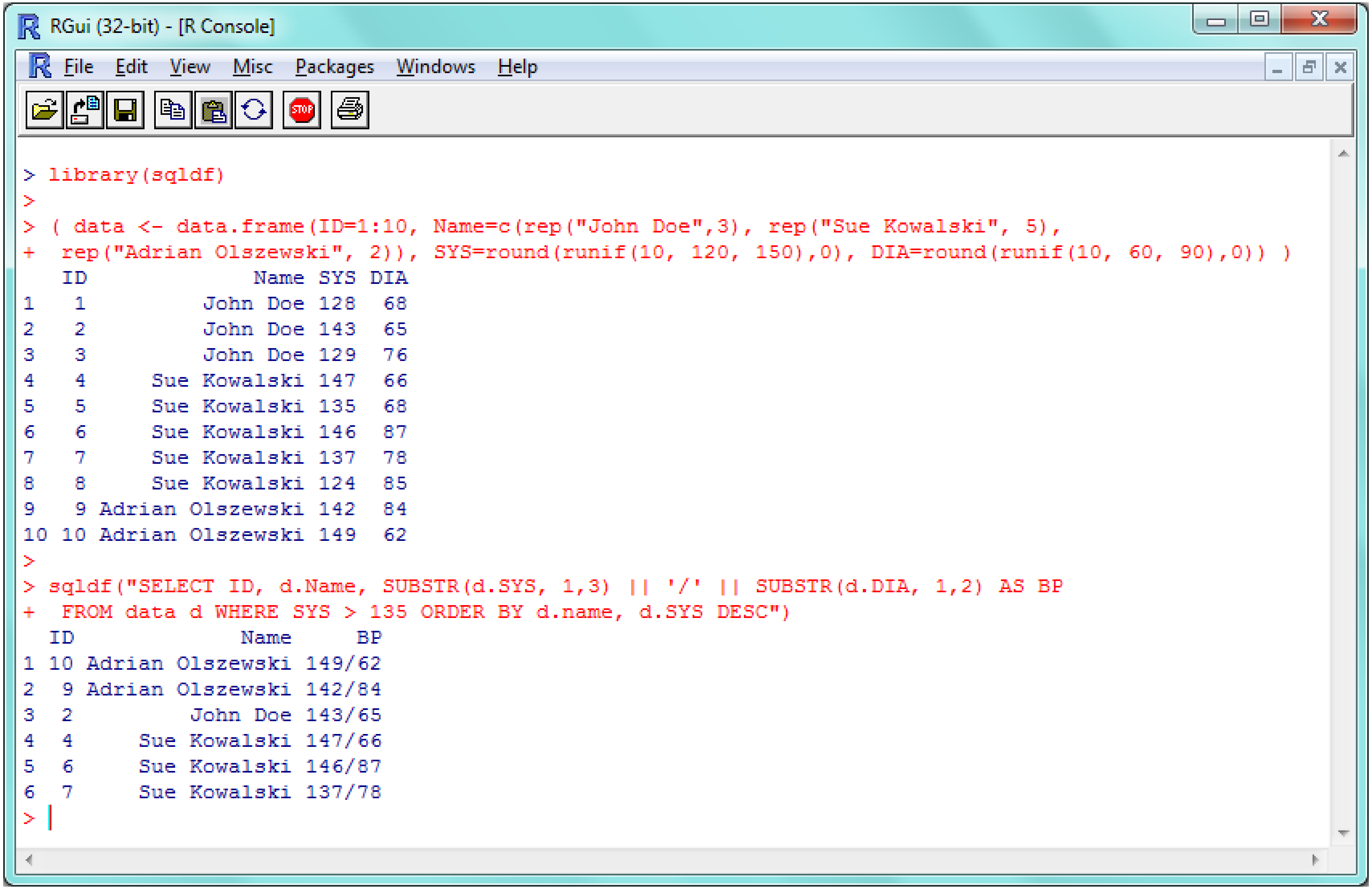


RECORD_ST	sdv	SVYN
complete	1	
complete	1	
complete	1	
complete	1	
complete	1	
complete	1	
complete	1	
complete	1	
complete	1	
complete	1	
complete	1	
complete	1	

```
> library(RODBC)
> con <-  odbcConnectAccess("c:/tmp/17sep.mdb")
> data <- sqlQuery(con, "
+ SELECT cntr.COUNTRY,
+       (
+           SELECT COUNT(*) + Int(100 * Rnd)
+           FROM
+           (
+               SELECT DISTINCT sv.COUNTRYID, sv.SITEID &'-'& sv.SUBJID
+               FROM SV sv
+               WHERE sv.VISITID=1
+           ) sv1
+           WHERE sv1.COUNTRYID = cntr.COUNTRY
+       ) AS Screened
+ FROM
+ (
+     SELECT DISTINCT sit.COUNTRY
+     FROM SITES sit
+     WHERE sit.COUNTRY IS NOT NULL
+ )cntr")
> knitr::kable(data)


|COUNTRY | Screened|
|:-------|--------:|
|[redacted]       |      151|
|[redacted]       |       53|
|[redacted]       |      231|
|[redacted]       |       32|
|[redacted]       |      214|
|[redacted]       |      105|
|[redacted]       |       64|
```

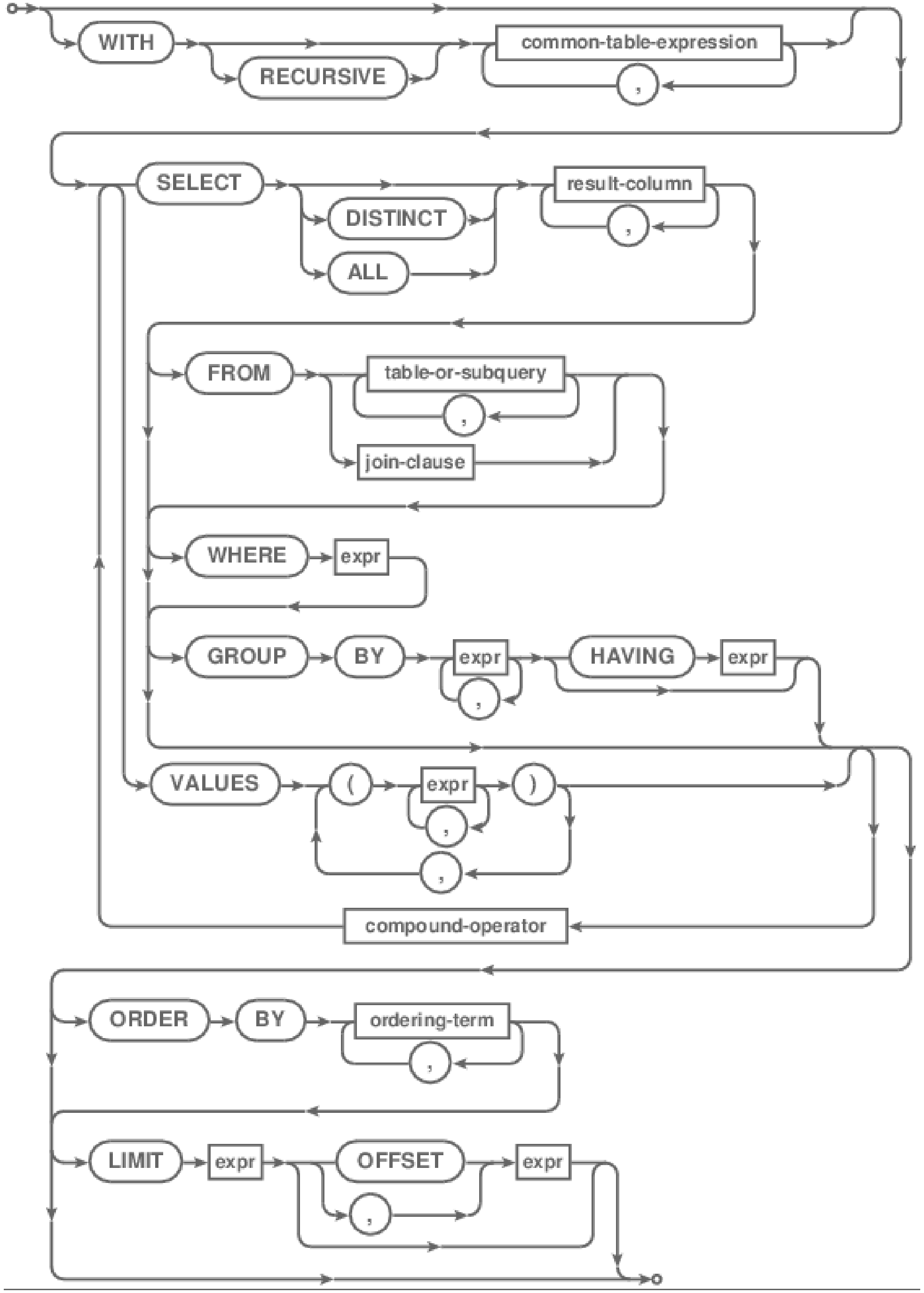
Querying R data.frames with sqldf

With the sqldf library data.frames can be queried with a full blown SQL like any regular database tables. Available syntax depends on the chosen engine: SQLite $^{\text{default}}$, H2, PostgreSQL or MySQL.



```
> library(sqldf)
>
> ( data <- data.frame(ID=1:10, Name=c(rep("John Doe",3), rep("Sue Kowalski", 5),
+  rep("Adrian Olszewski", 2)), SYS=round(runif(10, 120, 150),0), DIA=round(runif(10, 60, 90),0)) )
   ID             Name SYS DIA
1   1         John Doe 128  68
2   2         John Doe 143  65
3   3         John Doe 129  76
4   4     Sue Kowalski 147  66
5   5     Sue Kowalski 135  68
6   6     Sue Kowalski 146  87
7   7     Sue Kowalski 137  78
8   8     Sue Kowalski 124  85
9   9 Adrian Olszewski 142  84
10 10 Adrian Olszewski 149  62
>
> sqldf("SELECT ID, d.Name, SUBSTR(d.SYS, 1,3) || '/' || SUBSTR(d.DIA, 1,2) AS BP
+  FROM data d WHERE SYS > 135 ORDER BY d.name, d.SYS DESC")
  ID             Name     BP
1 10 Adrian Olszewski 149/62
2  9 Adrian Olszewski 142/84
3  2         John Doe 143/65
4  4     Sue Kowalski 147/66
5  6     Sue Kowalski 146/87
6  7     Sue Kowalski 137/78
> |
```

sqldf – a full blown SQL at your fingertips!



- ✔ Write any complex SQL queries using your favourite syntax and engine
- ✔ Use complex sub-queries in:
  - ✔ SELECT
  - ✔ FROM / JOIN
  - ✔ WHERE
- ✔ Nest up to 255 levels of sub-queries
- ✔ Use lots of engine-specific functions
- ✔ Limit results with LIMIT / OFFSET
- ✔ Query R data frames with spaces and dots in column names
- ✔ … and much more!

Do I really need SQL for querying data.frames?

Why would one bother using SQL when R is capable itself of sub-setting (filtering) data.frames, merging them, ordering, summarizing in sub-groups and reshaping? There are various packages that make it easy!

```
> attach(mydata)
> fit <- aov(y~A*B)
> summary(fit)
> layout(matrix(c(1:4), 2))
> plot(fit)
> TukeyHSD(fit)
> interaction.plot(A, B, y)
```

R is perfect for analysing data

SQL is ideal for querying data

```
SELECT  Grp, Sex, COUNT(*)
FROM Table1 t1
INNER JOIN Table2 t2
      ON t1.ID=t2.ID
WHERE t1.Age >=
      (SELECT AVG(t.Age)
       FROM Table1 t
       WHERE t.ID=t1.ID)
GROUP BY Grp, Sex
HAVING COUNT(*) > 1
```

Let everyone do what he does best!

SQLite – it's quite fast!

1 kB = 1000 B
1 kiB = 1024 B

The default, SQLite engine makes sqldf really fast.

Three simple benchmarks were run. Results are quite satisfying:

I. Reading a 1.7 GiB (4^{columns} x $40,000,000^{\text{rows}}$) CSV file into R data.frame

- via sqldf: **4m:20sec**
- via data.table::fread, the fastest method available in R: **3m:45sec**

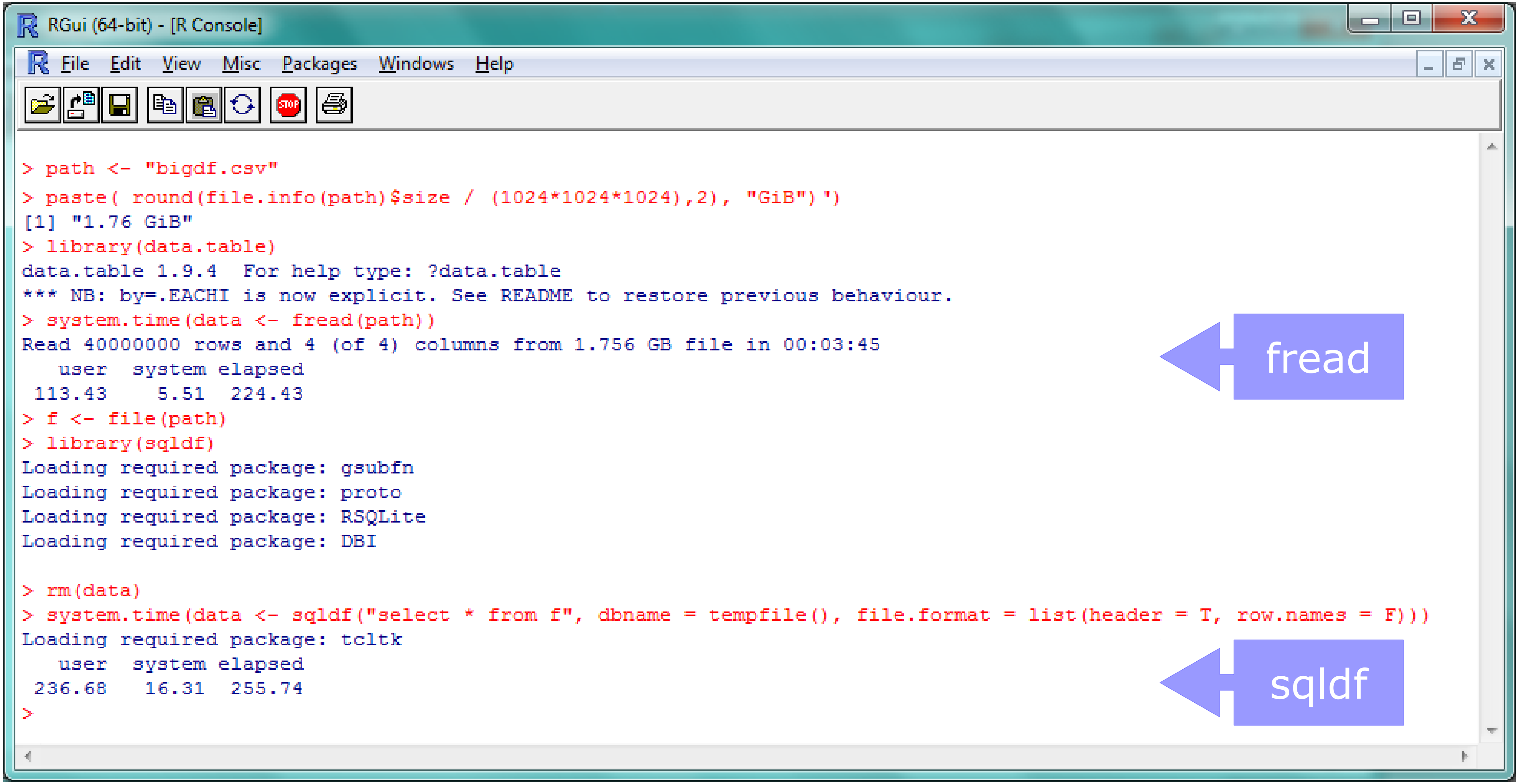
II. Querying this ~1.1 GiB dataset.

- via sqldf: **5.6 sec**
- via the fastest, native sub-setting method: **7.4 sec**

III. Querying a ~17 MiB (22^{columns} x $100,000^{\text{rows}}$) data.frame

- via sqldf: **0.7 sec** – one won't notice any lags in everyday practice.

SQLite – reading a huge CSV

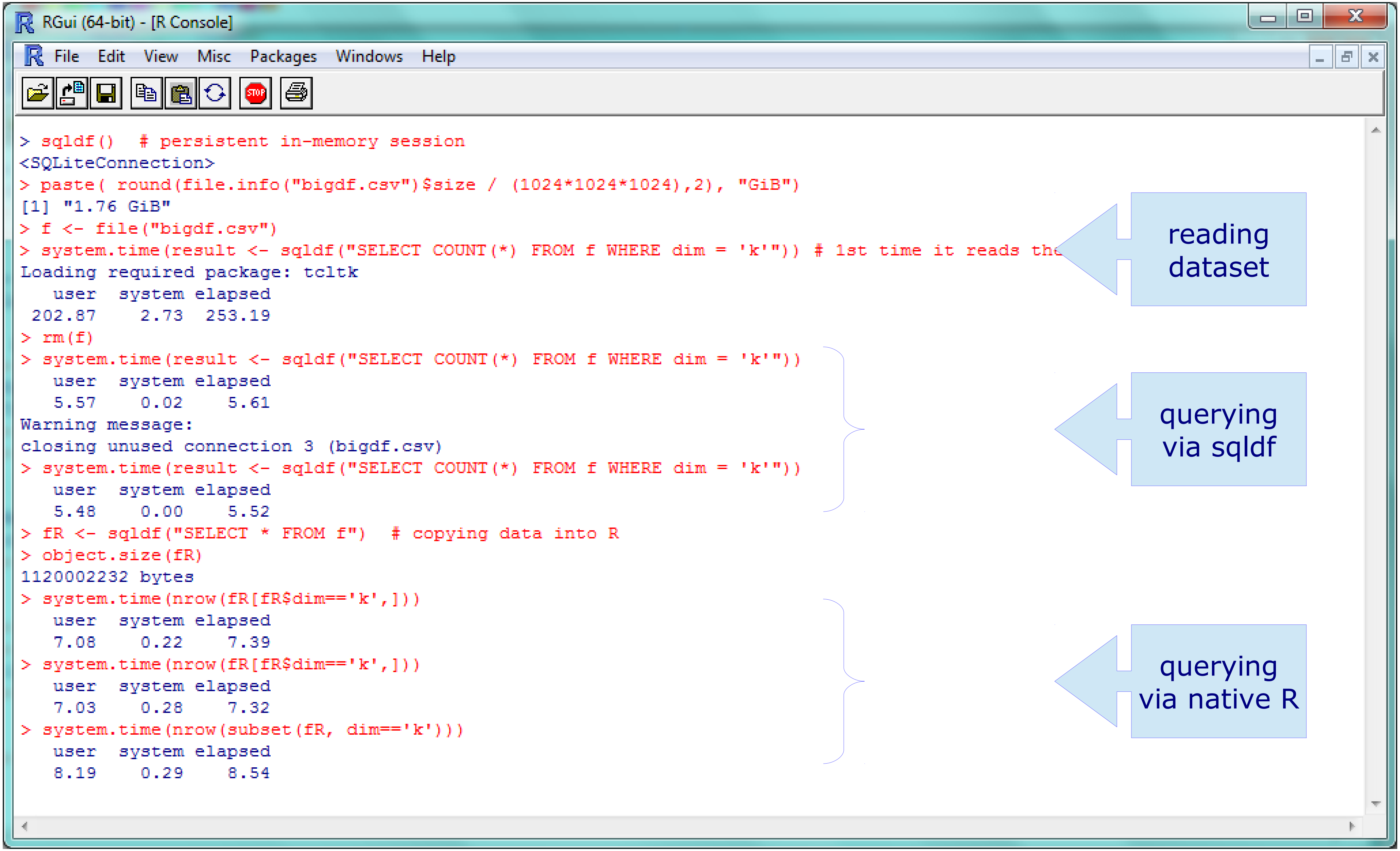


```
> path <- "bigdf.csv"
> paste( round(file.info(path)$size / (1024*1024*1024),2), "GiB")")
[1] "1.76 GiB"
> library(data.table)
data.table 1.9.4  For help type: ?data.table
*** NB: by=.EACHI is now explicit. See README to restore previous behaviour.
> system.time(data <- fread(path))
Read 40000000 rows and 4 (of 4) columns from 1.756 GB file in 00:03:45
   user  system elapsed 
 113.43    5.51  224.43 
> f <- file(path)
> library(sqldf)
Loading required package: gsubfn
Loading required package: proto
Loading required package: RSQLite
Loading required package: DBI

> rm(data)
> system.time(data <- sqldf("select * from f", dbname = tempfile(), file.format = list(header = T, row.names = F)))
Loading required package: tcltk
   user  system elapsed 
 236.68   16.31  255.74 
>
```

fread

sqldf

SQLite – performance of querying huge dataset



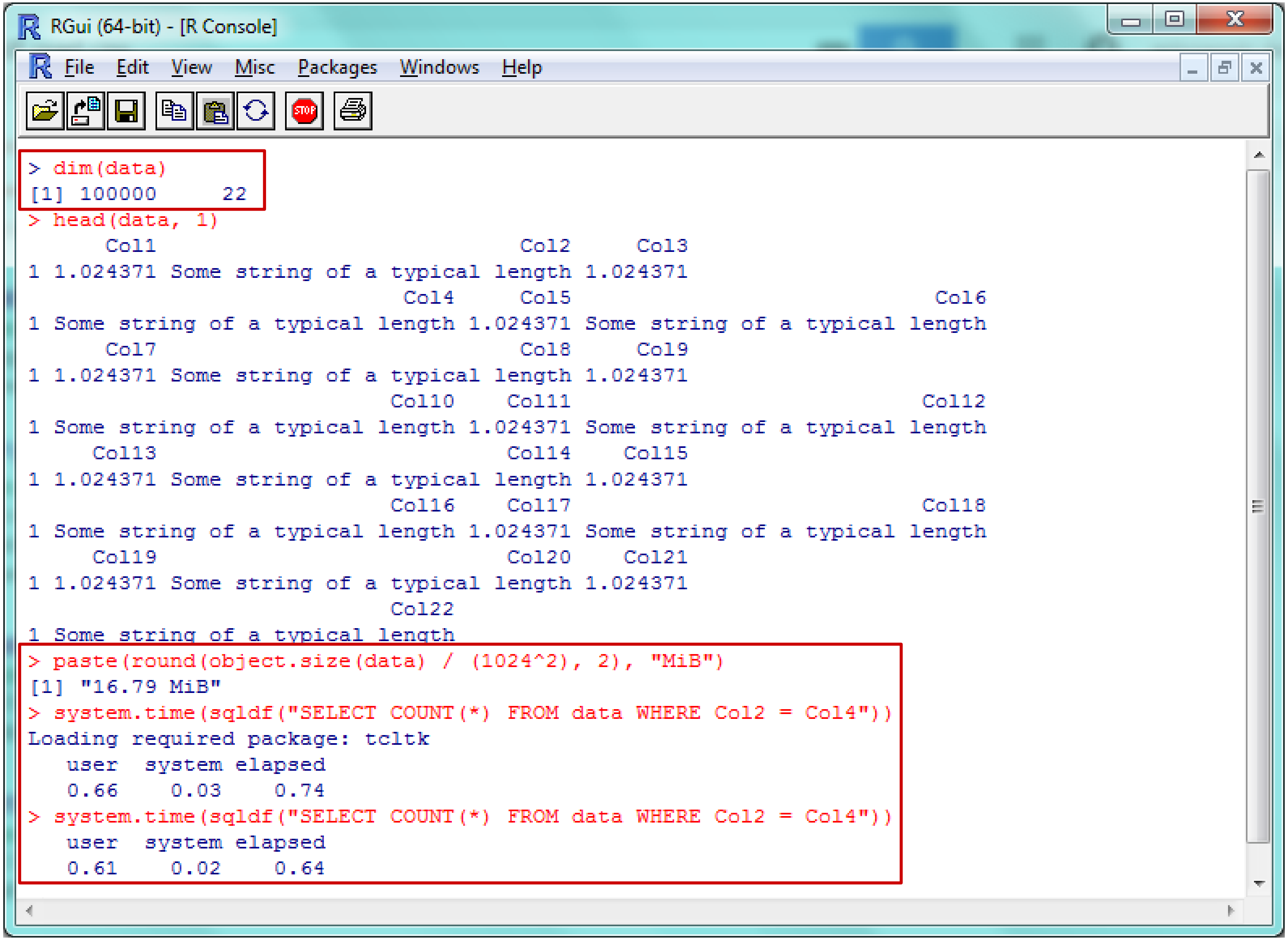
```
> sqldf()  # persistent in-memory session
<SQLiteConnection>
> paste( round(file.info("bigdf.csv")$size / (1024*1024*1024),2), "GiB")
[1] "1.76 GiB"
> f <- file("bigdf.csv")
> system.time(result <- sqldf("SELECT COUNT(*) FROM f WHERE dim = 'k'")) # 1st time it reads the
Loading required package: tcltk
   user  system elapsed
 202.87    2.73  253.19
> rm(f)
> system.time(result <- sqldf("SELECT COUNT(*) FROM f WHERE dim = 'k'"))
   user  system elapsed
   5.57    0.02    5.61
Warning message:
closing unused connection 3 (bigdf.csv)
> system.time(result <- sqldf("SELECT COUNT(*) FROM f WHERE dim = 'k'"))
   user  system elapsed
   5.48    0.00    5.52
> fR <- sqldf("SELECT * FROM f")  # copying data into R
> object.size(fR)
1120002232 bytes
> system.time(nrow(fR[fR$dim=='k',]))
   user  system elapsed
   7.08    0.22    7.39
> system.time(nrow(fR[fR$dim=='k',]))
   user  system elapsed
   7.03    0.28    7.32
> system.time(nrow(subset(fR, dim=='k')))
   user  system elapsed
   8.19    0.29    8.54
```

reading dataset

querying via sqldf

querying via native R

SQLite – performance of querying typical dataset



```
> dim(data)
[1] 100000     22
> head(data, 1)
      Col1                            Col2      Col3
1 1.024371 Some string of a typical length 1.024371
                             Col4      Col5                            Col6
1 Some string of a typical length 1.024371 Some string of a typical length
      Col7                            Col8      Col9
1 1.024371 Some string of a typical length 1.024371
                            Col10     Col11                           Col12
1 Some string of a typical length 1.024371 Some string of a typical length
     Col13                           Col14     Col15
1 1.024371 Some string of a typical length 1.024371
                            Col16     Col17                           Col18
1 Some string of a typical length 1.024371 Some string of a typical length
     Col19                           Col20     Col21
1 1.024371 Some string of a typical length 1.024371
                            Col22
1 Some string of a typical length
> paste(round(object.size(data) / (1024^2), 2), "MiB")
[1] "16.79 MiB"
> system.time(sqldf("SELECT COUNT(*) FROM data WHERE Col2 = Col4"))
Loading required package: tcltk
   user  system elapsed
   0.66    0.03    0.74
> system.time(sqldf("SELECT COUNT(*) FROM data WHERE Col2 = Col4"))
   user  system elapsed
   0.61    0.02    0.64
```

...much closer cooperation...

R can be invoked directly from within PostgreSQL and Microsoft SQL Server:

- In PostgreSQL one can use the PL/R language (example)
- In SQL Server there are three options:
  - write a CLR function which wraps invocations of a chosen R↔.NET “connector”: R.NET, RserveCLI, StatConnector, etc.
  - create and use DCOM objects directly (sp_OACreate)
  - NEW: use native T-SQL calls to run R directly from SQL Server

This enables the user to easily perform statistical analyses directly under the database engine and employ complex validation rules in triggers.

Easy data transfeR

With R and a rich set of database drivers (ODBC, JDBC, native) it is easy to transfer data between various sources with only few lines of code.

Actually, this is even easier than in C# :)

```
tables <- list()
tables[["tab1"]] <- read.csv("c:/tmp/table1.csv")              # source 1
tables[["tab2"]] <- read.xlsx("c:/tmp/mysheet.xlsx", sheet=2)  # source 2
tables[["tab3"]] <- sqlQuery(otherCon, "SELECT * FROM Table")  # source 3
…
myCon <- odbcConnectAccess("c:/tmp/my.mdb"))

for(table in names(tables)){
     sqlSave(myCon, tables[[table]], tablename = table)      # that's all!
}
OdbcClose(myCon)
```

Isn't this just beautiful?

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is (widely) supported by the world of science

IV R is supported by the community

V R is (increasingly) supported by the business

VI 2/3 :) Advanced data manipulation

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

Manipulating data

SQL is extremely useful for querying data but it is not the only option the user can choose. ddplyr, reshape2, tidyr and data.table libraries provide a rich set of functions for advanced querying and manipulating data structures, i.e.:

- Filtering (sub-setting) rows
- Selecting variables (columns)
- Adding new variables, e.g. computed (mutating and transmutating)
- Ordering results (sorting)
- Aggregating (grouping and summarizing)
- Reshaping data from wide to long format and vice-versa
- Combining (piping) the above operations by the *chain operator* %>%

Unquestionably, these packages are one of the top-most useful, “must-have” packages in the analyst's toolbox.

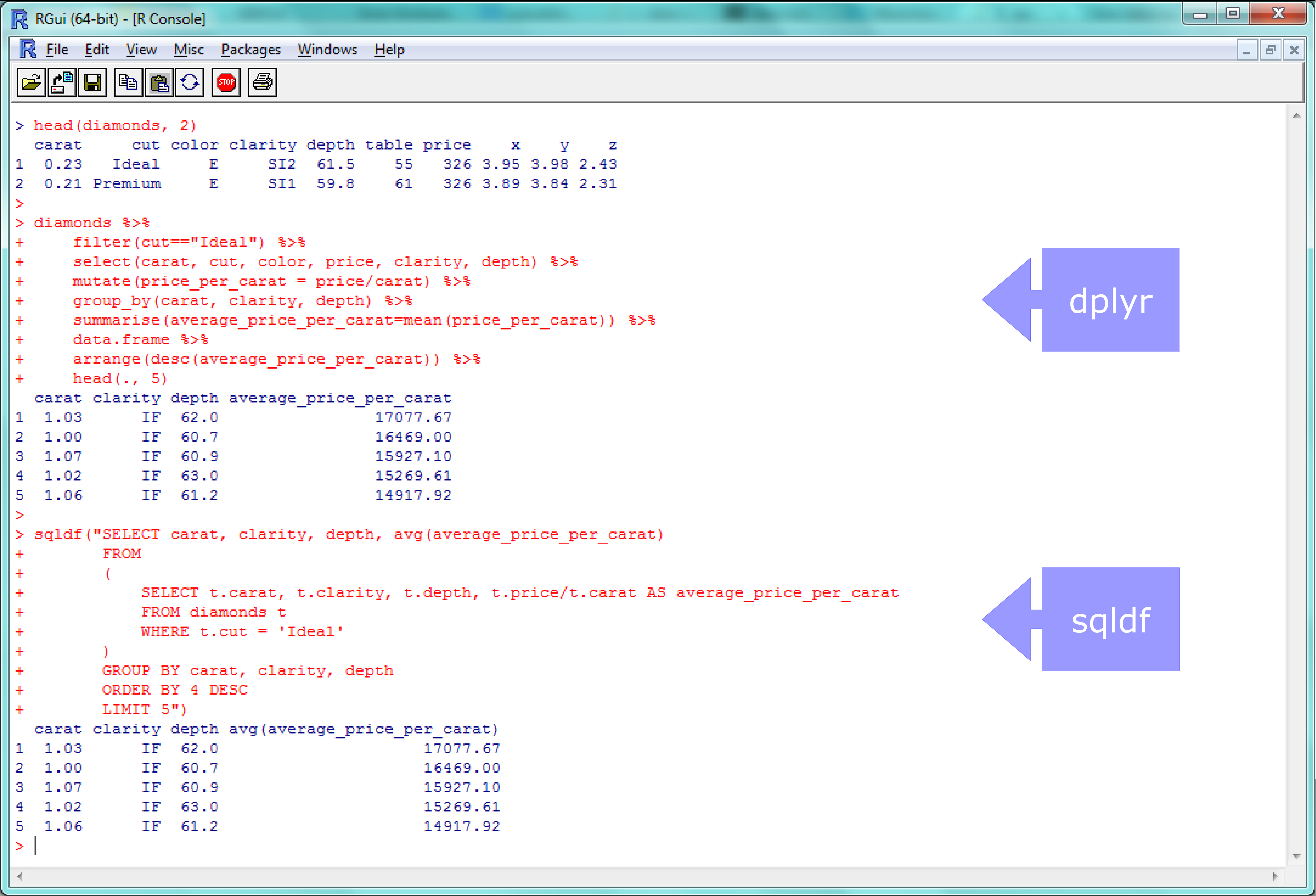
Manipulating data

The chain operator %>% is a one of the most useful operators in R.

It remarkably facilitates common data processing tasks placing them in a flow.

```
dataset %>%
    filter(column1 == 2 & column2 %in% c("A", "B", "C")) %>%
    select(column1, column2, colum3) %>%
    group_by(column1, column2, column3) %>%
        summarize(Avg = mean(column3), SD = sd(column3)) %>%
            mutate(CV = SD/Avg) %>%
            arrange(desc(CV)) %>%
                merge(., dictionary, by.x="column1", by.y="ID") %>%
                    head(., 10)
```

Querying data with dplyr compared to SQL

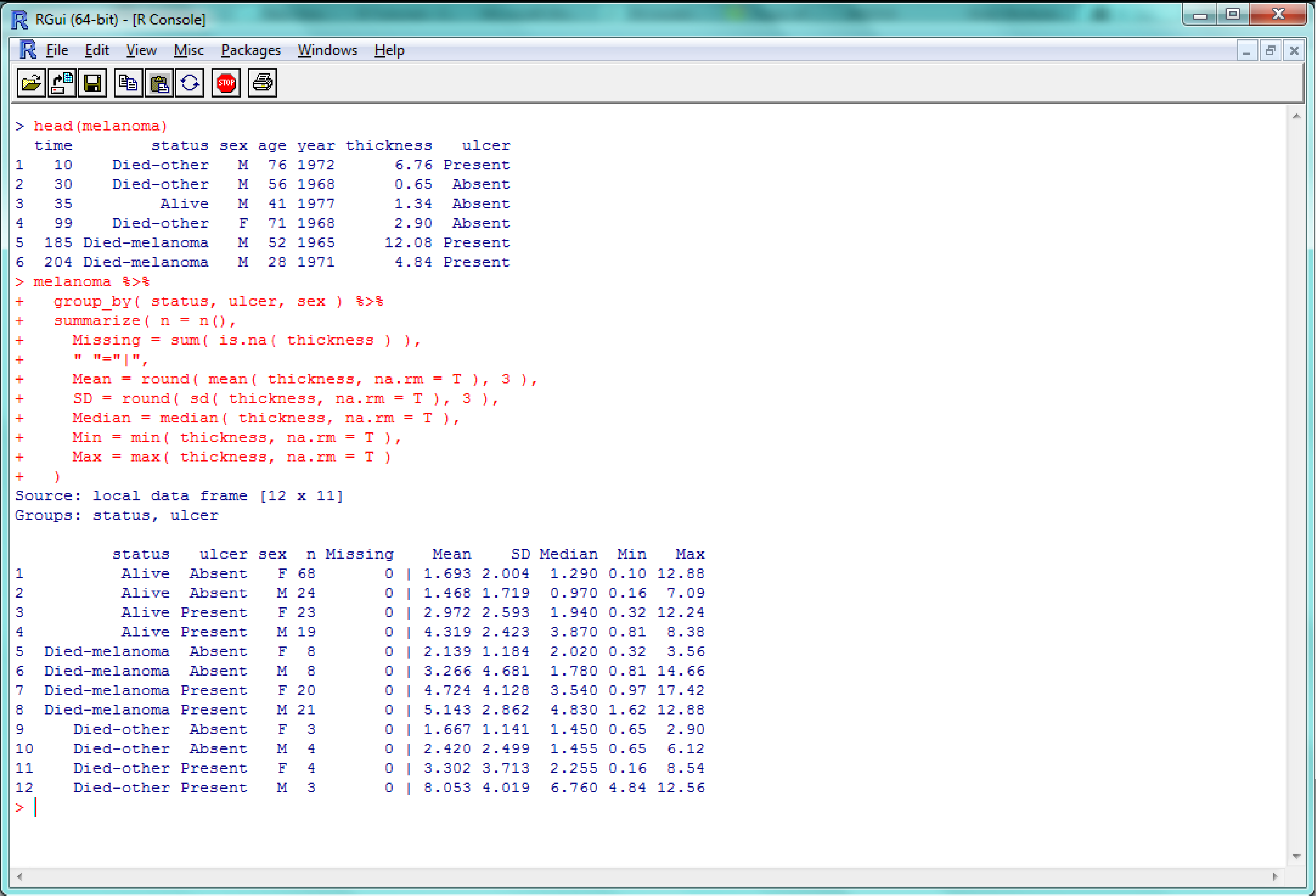


```
> head(diamonds, 2)
  carat     cut color clarity depth table price    x    y    z
1  0.23   Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21 Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
>
> diamonds %>%
+     filter(cut=="Ideal") %>%
+     select(carat, cut, color, price, clarity, depth) %>%
+     mutate(price_per_carat = price/carat) %>%
+     group_by(carat, clarity, depth) %>%
+     summarise(average_price_per_carat=mean(price_per_carat)) %>%
+     data.frame %>%
+     arrange(desc(average_price_per_carat)) %>%
+     head(., 5)
  carat clarity depth average_price_per_carat
1  1.03      IF  62.0                17077.67
2  1.00      IF  60.7                16469.00
3  1.07      IF  60.9                15927.10
4  1.02      IF  63.0                15269.61
5  1.06      IF  61.2                14917.92
>
> sqldf("SELECT carat, clarity, depth, avg(average_price_per_carat)
+        FROM
+        (
+            SELECT t.carat, t.clarity, t.depth, t.price/t.carat AS average_price_per_carat
+            FROM diamonds t
+            WHERE t.cut = 'Ideal'
+        )
+        GROUP BY carat, clarity, depth
+        ORDER BY 4 DESC
+        LIMIT 5")
  carat clarity depth avg(average_price_per_carat)
1  1.03      IF  62.0                     17077.67
2  1.00      IF  60.7                     16469.00
3  1.07      IF  60.9                     15927.10
4  1.02      IF  63.0                     15269.61
5  1.06      IF  61.2                     14917.92
> 
```

dplyr

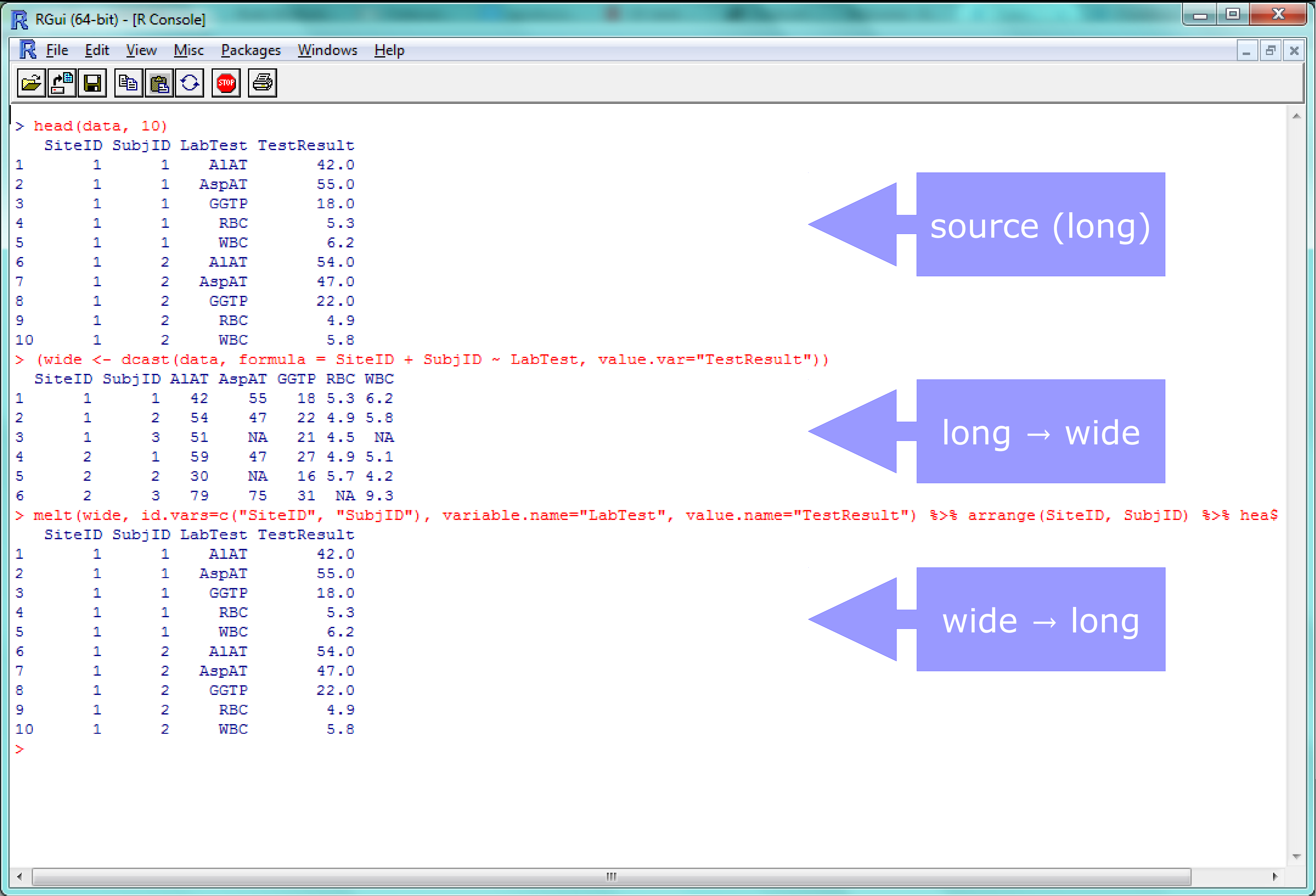
sqldf

With dplyr “group-by summaries” are easy!



```
> head(melanoma)
  time        status sex age year thickness   ulcer
1   10    Died-other   M  76 1972      6.76 Present
2   30    Died-other   M  56 1968      0.65  Absent
3   35         Alive   M  41 1977      1.34  Absent
4   99    Died-other   F  71 1968      2.90  Absent
5  185 Died-melanoma   M  52 1965     12.08 Present
6  204 Died-melanoma   M  28 1971      4.84 Present
> melanoma %>%
+   group_by( status, ulcer, sex ) %>%
+   summarize( n = n(),
+     Missing = sum( is.na( thickness ) ),
+     " "="|",
+     Mean = round( mean( thickness, na.rm = T ), 3 ),
+     SD = round( sd( thickness, na.rm = T ), 3 ),
+     Median = median( thickness, na.rm = T ),
+     Min = min( thickness, na.rm = T ),
+     Max = max( thickness, na.rm = T )
+   )
Source: local data frame [12 x 11]
Groups: status, ulcer

          status   ulcer sex  n Missing    Mean    SD Median  Min   Max
1          Alive  Absent   F 68       0 | 1.693 2.004  1.290 0.10 12.88
2          Alive  Absent   M 24       0 | 1.468 1.719  0.970 0.16  7.09
3          Alive Present   F 23       0 | 2.972 2.593  1.940 0.32 12.24
4          Alive Present   M 19       0 | 4.319 2.423  3.870 0.81  8.38
5  Died-melanoma  Absent   F  8       0 | 2.139 1.184  2.020 0.32  3.56
6  Died-melanoma  Absent   M  8       0 | 3.266 4.681  1.780 0.81 14.66
7  Died-melanoma Present   F 20       0 | 4.724 4.128  3.540 0.97 17.42
8  Died-melanoma Present   M 21       0 | 5.143 2.862  4.830 1.62 12.88
9     Died-other  Absent   F  3       0 | 1.667 1.141  1.450 0.65  2.90
10    Died-other  Absent   M  4       0 | 2.420 2.499  1.455 0.65  6.12
11    Died-other Present   F  4       0 | 3.302 3.713  2.255 0.16  8.54
12    Died-other Present   M  3       0 | 8.053 4.019  6.760 4.84 12.56
> |
```

Reshaping long ↔ wide format with reshape2



```
> head(data, 10)
   SiteID SubjID LabTest TestResult
1       1      1    AlAT       42.0
2       1      1   AspAT       55.0
3       1      1    GGTP       18.0
4       1      1     RBC        5.3
5       1      1     WBC        6.2
6       1      2    AlAT       54.0
7       1      2   AspAT       47.0
8       1      2    GGTP       22.0
9       1      2     RBC        4.9
10      1      2     WBC        5.8
> (wide <- dcast(data, formula = SiteID + SubjID ~ LabTest, value.var="TestResult"))
  SiteID SubjID AlAT AspAT GGTP RBC WBC
1      1      1   42    55   18 5.3 6.2
2      1      2   54    47   22 4.9 5.8
3      1      3   51    NA   21 4.5  NA
4      2      1   59    47   27 4.9 5.1
5      2      2   30    NA   16 5.7 4.2
6      2      3   79    75   31  NA 9.3
> melt(wide, id.vars=c("SiteID", "SubjID"), variable.name="LabTest", value.name="TestResult") %>% arrange(SiteID, SubjID) %>% hea$
   SiteID SubjID LabTest TestResult
1       1      1    AlAT       42.0
2       1      1   AspAT       55.0
3       1      1    GGTP       18.0
4       1      1     RBC        5.3
5       1      1     WBC        6.2
6       1      2    AlAT       54.0
7       1      2   AspAT       47.0
8       1      2    GGTP       22.0
9       1      2     RBC        4.9
10      1      2     WBC        5.8
>
```

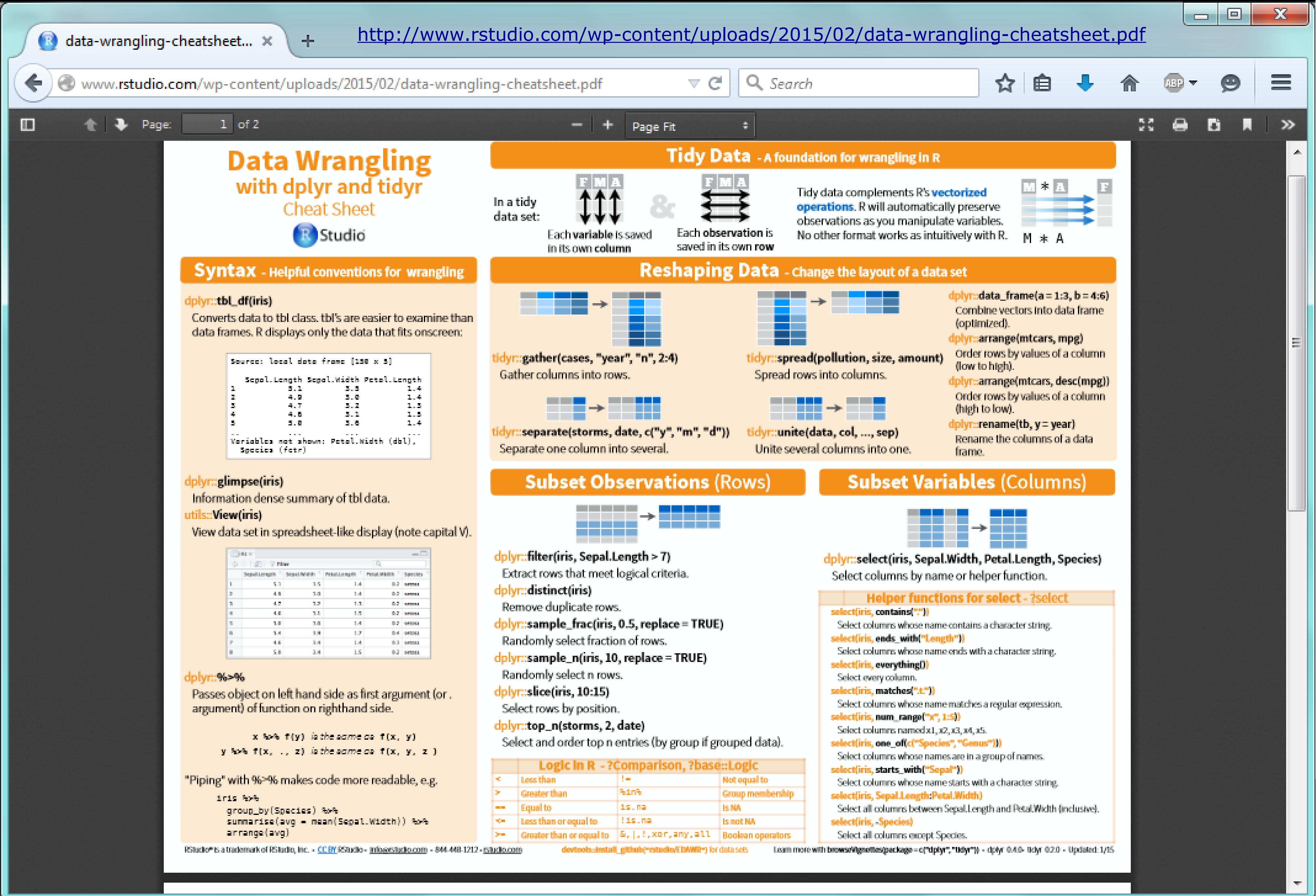
source (long)

long → wide

wide → long

Manipulating data

http://www.rstudio.com/wp-content/uploads/2015/02/data-wrangling-cheatsheet.pdf



Data Wrangling
with dplyr and tidyr
Cheat Sheet

RStudio

Syntax - Helpful conventions for wrangling

dplyr::tbl_df(iris)

Converts data to tbl class. tbl's are easier to examine than data frames. R displays only the data that fits onscreen:

```
Source: local data frame [150 x 5]

   Sepal.Length Sepal.Width Petal.Length
1           5.1         3.5          1.4
2           4.9         3.0          1.4
3           4.7         3.2          1.3
4           4.6         3.1          1.5
5           5.0         3.6          1.4
..          ...         ...          ...
Variables not shown: Petal.Width (dbl),
  Species (fctr)
```

dplyr::glimpse(iris)

Information dense summary of tbl data.

utils::View(iris)

View data set in spreadsheet-like display (note capital V).

dplyr::%>%

Passes object on left hand side as first argument (or . argument) of function on righthand side.

```
   x %>% f(y) is the same as f(x, y)
y %>% f(x, ., z) is the same as f(x, y, z )
```

"Piping" with %>% makes code more readable, e.g.

```
iris %>%
  group_by(Species) %>%
  summarise(avg = mean(Sepal.Width)) %>%
  arrange(avg)
```

Tidy Data - A foundation for wrangling in R

In a tidy data set:

Each **variable** is saved in its own **column**

Each **observation** is saved in its own **row**

Tidy data complements R's **vectorized operations**. R will automatically preserve observations as you manipulate variables. No other format works as intuitively with R.

Reshaping Data - Change the layout of a data set

tidyr::gather(cases, "year", "n", 2:4)

Gather columns into rows.

tidyr::spread(pollution, size, amount)

Spread rows into columns.

tidyr::separate(storms, date, c("y", "m", "d"))

Separate one column into several.

tidyr::unite(data, col, ..., sep)

Unite several columns into one.

dplyr::data_frame(a = 1:3, b = 4:6)

Combine vectors into data frame (optimized).

dplyr::arrange(mtcars, mpg)

Order rows by values of a column (low to high).

dplyr::arrange(mtcars, desc(mpg))

Order rows by values of a column (high to low).

dplyr::rename(tb, y = year)

Rename the columns of a data frame.

Subset Observations (Rows)

dplyr::filter(iris, Sepal.Length > 7)

Extract rows that meet logical criteria.

dplyr::distinct(iris)

Remove duplicate rows.

dplyr::sample_frac(iris, 0.5, replace = TRUE)

Randomly select fraction of rows.

dplyr::sample_n(iris, 10, replace = TRUE)

Randomly select n rows.

dplyr::slice(iris, 10:15)

Select rows by position.

dplyr::top_n(storms, 2, date)

Select and order top n entries (by group if grouped data).

Logic in R - ?Comparison, ?base::Logic			
<	Less than	!=	Not equal to
>	Greater than	%in%	Group membership
==	Equal to	is.na	Is NA
<=	Less than or equal to	!is.na	Is not NA
>=	Greater than or equal to	&,|,!,xor,any,all	Boolean operators

Subset Variables (Columns)

dplyr::select(iris, Sepal.Width, Petal.Length, Species)

Select columns by name or helper function.

Helper functions for select - ?select

select(iris, contains("."))
Select columns whose name contains a character string.

select(iris, ends_with("Length"))
Select columns whose name ends with a character string.

select(iris, everything())
Select every column.

select(iris, matches(".t."))
Select columns whose name matches a regular expression.

select(iris, num_range("x", 1:5))
Select columns named x1, x2, x3, x4, x5.

select(iris, one_of(c("Species", "Genus")))
Select columns whose names are in a group of names.

select(iris, starts_with("Sepal"))
Select columns whose name starts with a character string.

select(iris, Sepal.Length:Petal.Width)
Select all columns between Sepal.Length and Petal.Width (inclusive).

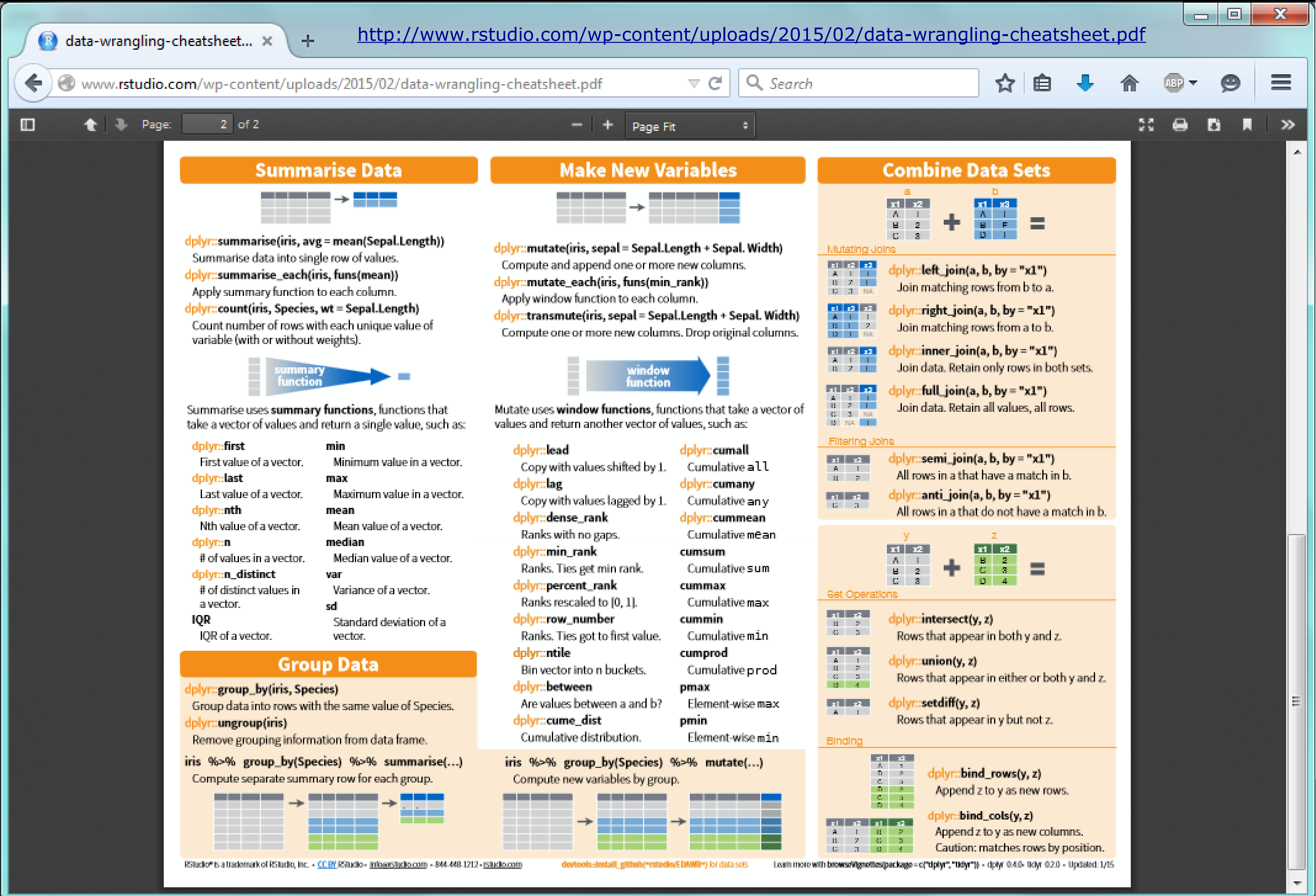
select(iris, -Species)
Select all columns except Species.

devtools::install_github("rstudio/EDAWR") for data sets

Learn more with browseVignettes(package = c("dplyr", "tidyr")) • dplyr 0.4.0• tidyr 0.2.0 • Updated: 1/15

Manipulating data

http://www.rstudio.com/wp-content/uploads/2015/02/data-wrangling-cheatsheet.pdf



Summarise Data

dplyr::**summarise(iris, avg = mean(Sepal.Length))**
Summarise data into single row of values.

dplyr::**summarise_each(iris, funs(mean))**
Apply summary function to each column.

dplyr::**count(iris, Species, wt = Sepal.Length)**
Count number of rows with each unique value of variable (with or without weights).

Summarise uses **summary functions**, functions that take a vector of values and return a single value, such as:

- dplyr::**first** — First value of a vector.
- dplyr::**last** — Last value of a vector.
- dplyr::**nth** — Nth value of a vector.
- dplyr::**n** — # of values in a vector.
- dplyr::**n_distinct** — # of distinct values in a vector.
- **IQR** — IQR of a vector.
- **min** — Minimum value in a vector.
- **max** — Maximum value in a vector.
- **mean** — Mean value of a vector.
- **median** — Median value of a vector.
- **var** — Variance of a vector.
- **sd** — Standard deviation of a vector.

Group Data

dplyr::**group_by(iris, Species)**
Group data into rows with the same value of Species.

dplyr::**ungroup(iris)**
Remove grouping information from data frame.

iris %>% group_by(Species) %>% summarise(...)
Compute separate summary row for each group.

Make New Variables

dplyr::**mutate(iris, sepal = Sepal.Length + Sepal. Width)**
Compute and append one or more new columns.

dplyr::**mutate_each(iris, funs(min_rank))**
Apply window function to each column.

dplyr::**transmute(iris, sepal = Sepal.Length + Sepal. Width)**
Compute one or more new columns. Drop original columns.

Mutate uses **window functions**, functions that take a vector of values and return another vector of values, such as:

- dplyr::**lead** — Copy with values shifted by 1.
- dplyr::**lag** — Copy with values lagged by 1.
- dplyr::**dense_rank** — Ranks with no gaps.
- dplyr::**min_rank** — Ranks. Ties get min rank.
- dplyr::**percent_rank** — Ranks rescaled to [0, 1].
- dplyr::**row_number** — Ranks. Ties got to first value.
- dplyr::**ntile** — Bin vector into n buckets.
- dplyr::**between** — Are values between a and b?
- dplyr::**cume_dist** — Cumulative distribution.
- dplyr::**cumall** — Cumulative `all`
- dplyr::**cumany** — Cumulative `any`
- dplyr::**cummean** — Cumulative `mean`
- **cumsum** — Cumulative `sum`
- **cummax** — Cumulative `max`
- **cummin** — Cumulative `min`
- **cumprod** — Cumulative `prod`
- **pmax** — Element-wise `max`
- **pmin** — Element-wise `min`

iris %>% group_by(Species) %>% mutate(...)
Compute new variables by group.

Combine Data Sets

Mutating Joins

dplyr::**left_join(a, b, by = "x1")**
Join matching rows from b to a.

dplyr::**right_join(a, b, by = "x1")**
Join matching rows from a to b.

dplyr::**inner_join(a, b, by = "x1")**
Join data. Retain only rows in both sets.

dplyr::**full_join(a, b, by = "x1")**
Join data. Retain all values, all rows.

Filtering Joins

dplyr::**semi_join(a, b, by = "x1")**
All rows in a that have a match in b.

dplyr::**anti_join(a, b, by = "x1")**
All rows in a that do not have a match in b.

Set Operations

dplyr::**intersect(y, z)**
Rows that appear in both y and z.

dplyr::**union(y, z)**
Rows that appear in either or both y and z.

dplyr::**setdiff(y, z)**
Rows that appear in y but not z.

Binding

dplyr::**bind_rows(y, z)**
Append z to y as new rows.

dplyr::**bind_cols(y, z)**
Append z to y as new columns.
Caution: matches rows by position.

RStudio® is a trademark of RStudio, Inc. • CC BY RStudio • info@rstudio.com • 844-448-1212 • rstudio.com
devtools::install_github("rstudio/EDAWR") for data sets
Learn more with browseVignettes(package = c("dplyr", "tidyr")) • dplyr 0.4.0 • tidyr 0.2.0 • Updated: 1/15

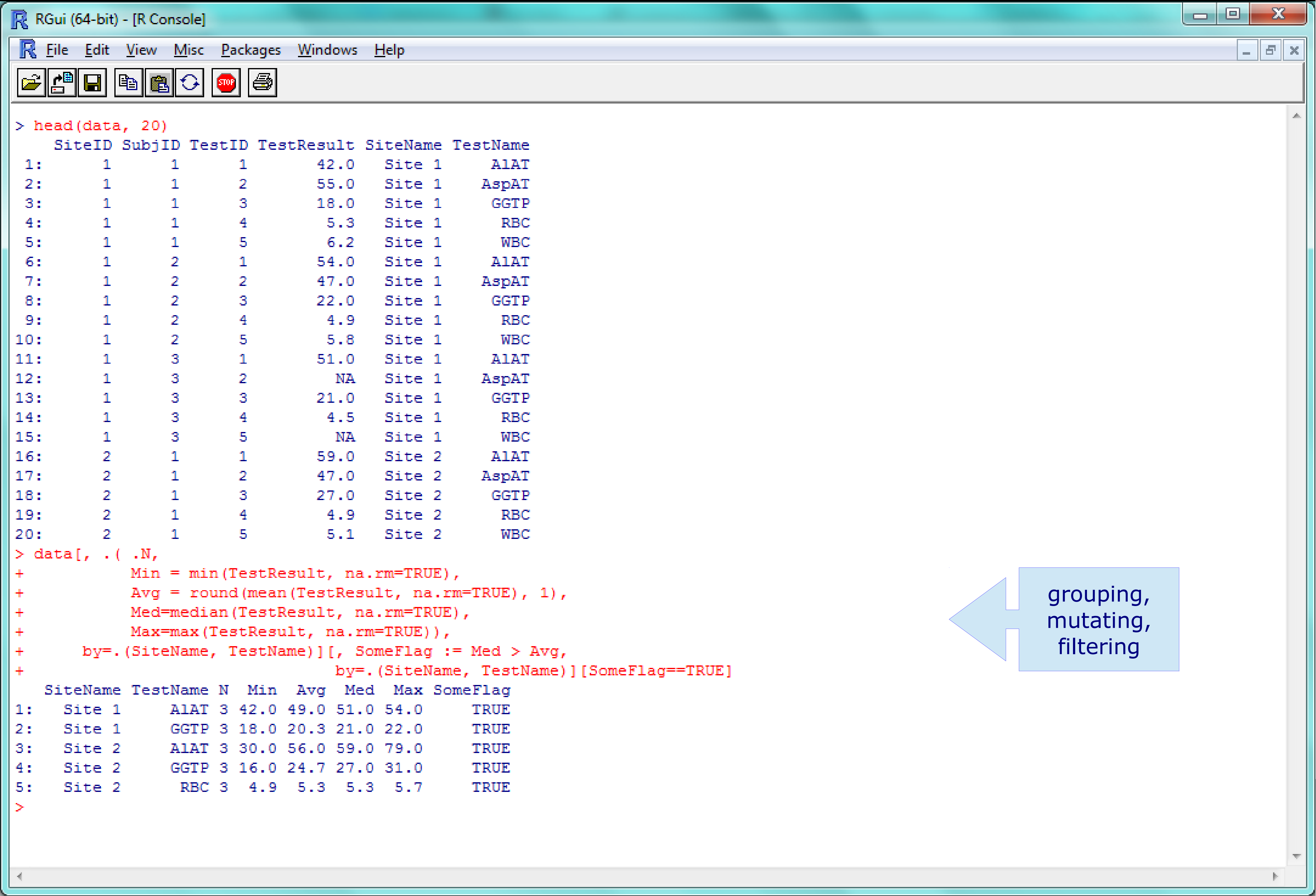
data.frame → data.table

data.table is a library providing a significant enhancement of the regular R data.frame. It allows the user to perform really fast (indexing) in-memory processing of huge volumes of data (>100 GB) with a relatively easy syntax.
It covers the following tasks:

- filtering rows and selecting columns
- adding and deleting (mutating) columns using no intermediate copies at all
- joining tables by indexes
- aggregating (calculations in sub-groups)
- reading huge CSV files – this is the fastest method currently available in R
- indexing selected content (set of columns) of a data.table

Data.tables are compatible with data.frames with some exceptions (FAQ 2.17).

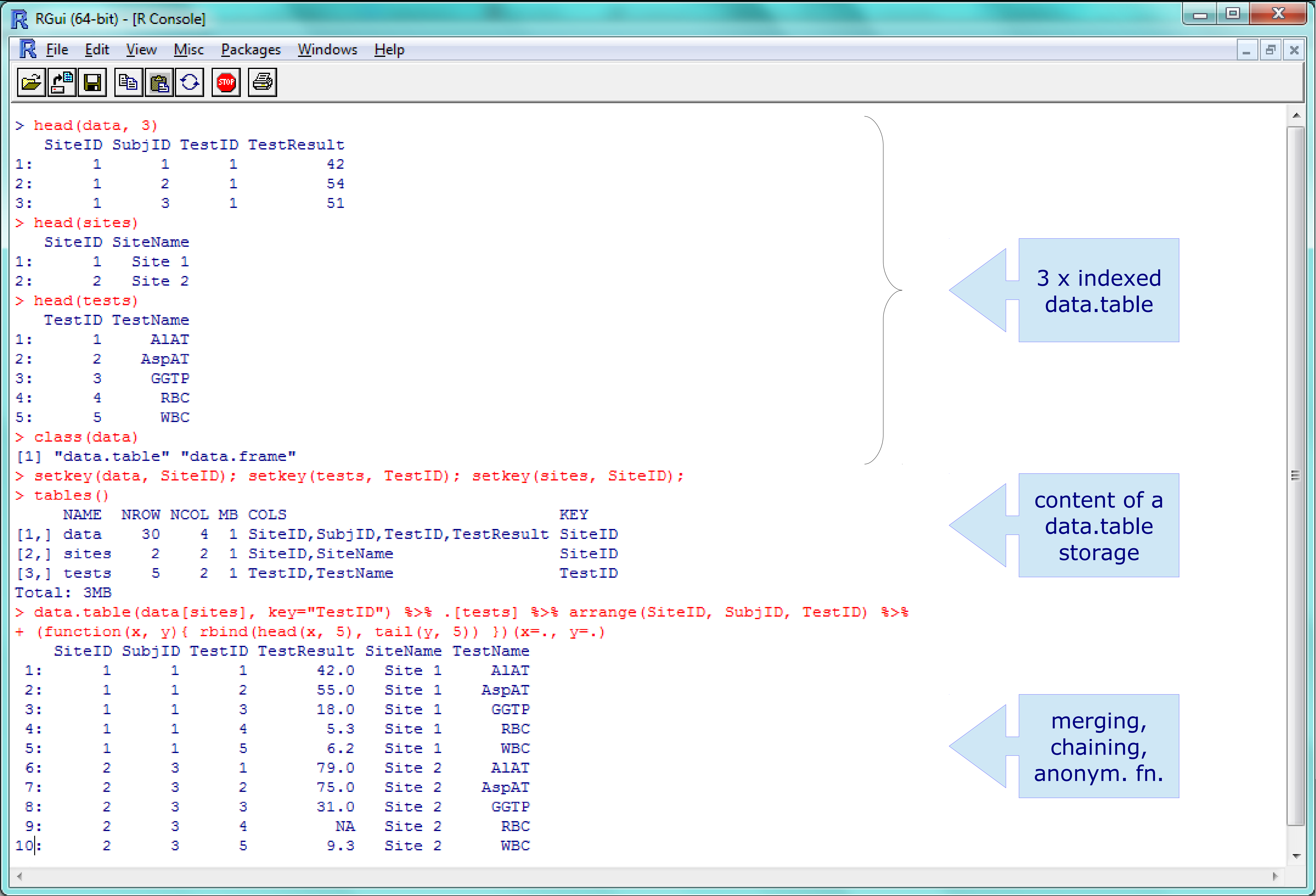
With data.table summaries are easy too!



```
> head(data, 20)
    SiteID SubjID TestID TestResult SiteName TestName
 1:      1      1      1       42.0   Site 1     AlAT
 2:      1      1      2       55.0   Site 1    AspAT
 3:      1      1      3       18.0   Site 1     GGTP
 4:      1      1      4        5.3   Site 1      RBC
 5:      1      1      5        6.2   Site 1      WBC
 6:      1      2      1       54.0   Site 1     AlAT
 7:      1      2      2       47.0   Site 1    AspAT
 8:      1      2      3       22.0   Site 1     GGTP
 9:      1      2      4        4.9   Site 1      RBC
10:      1      2      5        5.8   Site 1      WBC
11:      1      3      1       51.0   Site 1     AlAT
12:      1      3      2         NA   Site 1    AspAT
13:      1      3      3       21.0   Site 1     GGTP
14:      1      3      4        4.5   Site 1      RBC
15:      1      3      5         NA   Site 1      WBC
16:      2      1      1       59.0   Site 2     AlAT
17:      2      1      2       47.0   Site 2    AspAT
18:      2      1      3       27.0   Site 2     GGTP
19:      2      1      4        4.9   Site 2      RBC
20:      2      1      5        5.1   Site 2      WBC
> data[, .( .N,
+           Min = min(TestResult, na.rm=TRUE),
+           Avg = round(mean(TestResult, na.rm=TRUE), 1),
+           Med=median(TestResult, na.rm=TRUE),
+           Max=max(TestResult, na.rm=TRUE)),
+      by=.(SiteName, TestName)][, SomeFlag := Med > Avg,
+                                by=.(SiteName, TestName)][SomeFlag==TRUE]
   SiteName TestName N  Min  Avg  Med  Max SomeFlag
1:   Site 1     AlAT 3 42.0 49.0 51.0 54.0     TRUE
2:   Site 1     GGTP 3 18.0 20.3 21.0 22.0     TRUE
3:   Site 2     AlAT 3 30.0 56.0 59.0 79.0     TRUE
4:   Site 2     GGTP 3 16.0 24.7 27.0 31.0     TRUE
5:   Site 2      RBC 3  4.9  5.3  5.3  5.7     TRUE
>
```

grouping,
mutating,
filtering

data.table feat. dplyr



```
> head(data, 3)
   SiteID SubjID TestID TestResult
1:      1      1      1         42
2:      1      2      1         54
3:      1      3      1         51
> head(sites)
   SiteID SiteName
1:      1   Site 1
2:      2   Site 2
> head(tests)
   TestID TestName
1:      1     AlAT
2:      2    AspAT
3:      3     GGTP
4:      4      RBC
5:      5      WBC
> class(data)
[1] "data.table" "data.frame"
> setkey(data, SiteID); setkey(tests, TestID); setkey(sites, SiteID);
> tables()
     NAME  NROW NCOL MB COLS                            KEY
[1,] data    30    4  1 SiteID,SubjID,TestID,TestResult SiteID
[2,] sites    2    2  1 SiteID,SiteName                 SiteID
[3,] tests    5    2  1 TestID,TestName                 TestID
Total: 3MB
> data.table(data[sites], key="TestID") %>% .[tests] %>% arrange(SiteID, SubjID, TestID) %>%
+ (function(x, y){ rbind(head(x, 5), tail(y, 5)) })(x=., y=.)
    SiteID SubjID TestID TestResult SiteName TestName
 1:      1      1      1       42.0   Site 1     AlAT
 2:      1      1      2       55.0   Site 1    AspAT
 3:      1      1      3       18.0   Site 1     GGTP
 4:      1      1      4        5.3   Site 1      RBC
 5:      1      1      5        6.2   Site 1      WBC
 6:      2      3      1       79.0   Site 2     AlAT
 7:      2      3      2       75.0   Site 2    AspAT
 8:      2      3      3       31.0   Site 2     GGTP
 9:      2      3      4         NA   Site 2      RBC
10:      2      3      5        9.3   Site 2      WBC
```

3 x indexed data.table

content of a data.table storage

merging, chaining, anonym. fn.

Manipulating data

Valuable resources:

- http://seananderson.ca/2013/10/19/reshape.html
- http://seananderson.ca/2014/09/13/dplyr-intro.html
- http://www.sharpsightlabs.com/dplyr-intro-data-manipulation-with-r
- http://cran.rstudio.com/web/packages/dplyr/vignettes/introduction.html
- http://www.statsblogs.com/2014/02/10/how-dplyr-replaced-my-most...
- http://www.cookbookr.com/Manipulating_data/Converting_data_between...
- http://datatable.r-forge.r-project.org/datatable-faq.pdf
- http://s3.amazonaws.com/assets.datacamp.com/img/blog/data+table...
- http://github.com/Rdatatable/data.table/wiki/Benchmarks-%3A-Grouping

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)
II R has (extremely) wide range of capabilities
III R is (widely) supported by the world of science
IV R is supported by the community
V R is (increasingly) supported by the business
VI R is able to read data in many formats
VII Interoperability is easy to achieve
VIII R is truly cross-platform
IX R offers numerous ways of presenting data
X There are many possibilities to optimize the code
XI R is able to handle large amount of data
XII R has a set of fancy tools and IDEs
XII FDA accepted using R for drug trials!

"R" stands for inteRoperability

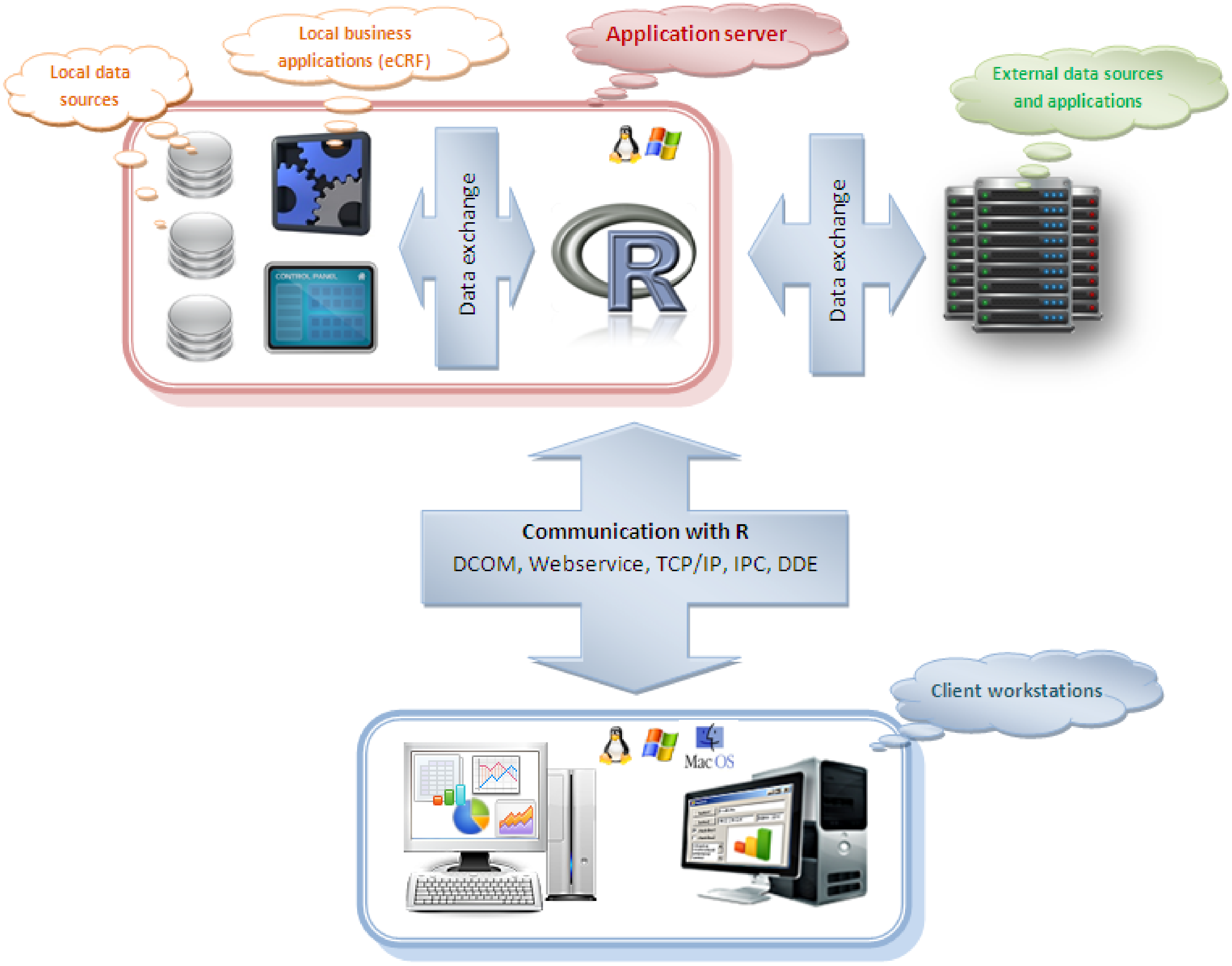
R is not only a great statistical package. It is often used as a part of more complex systems as a computing engine. It may also function as a standalone computational server through the Web.

There are number of libraries that allow to communicate with R in many ways (COM, TCP/IP, WebServices, DDE, direct linking) from many programming languages and platforms (C++, Java, .NET/Mono, Perl, Python, Scala, ...) or external applications able to act as a COM, DDE or Webservice client.

It's worth noting that R is able to call C/C++, Java, .NET, Perl and Python code.

In addition, R allows to create embedded graphical user interfaces (GUI) for easier entering data and performing analyzes by people not familiar with R. One can make use of the GTK, Tcl/Tk, wxWidgets and gWidgets toolkits or the R GUI Generator.

“R” stands for inteRoperability



Local data sources

Local business applications (eCRF)

Application server

External data sources and applications

Data exchange

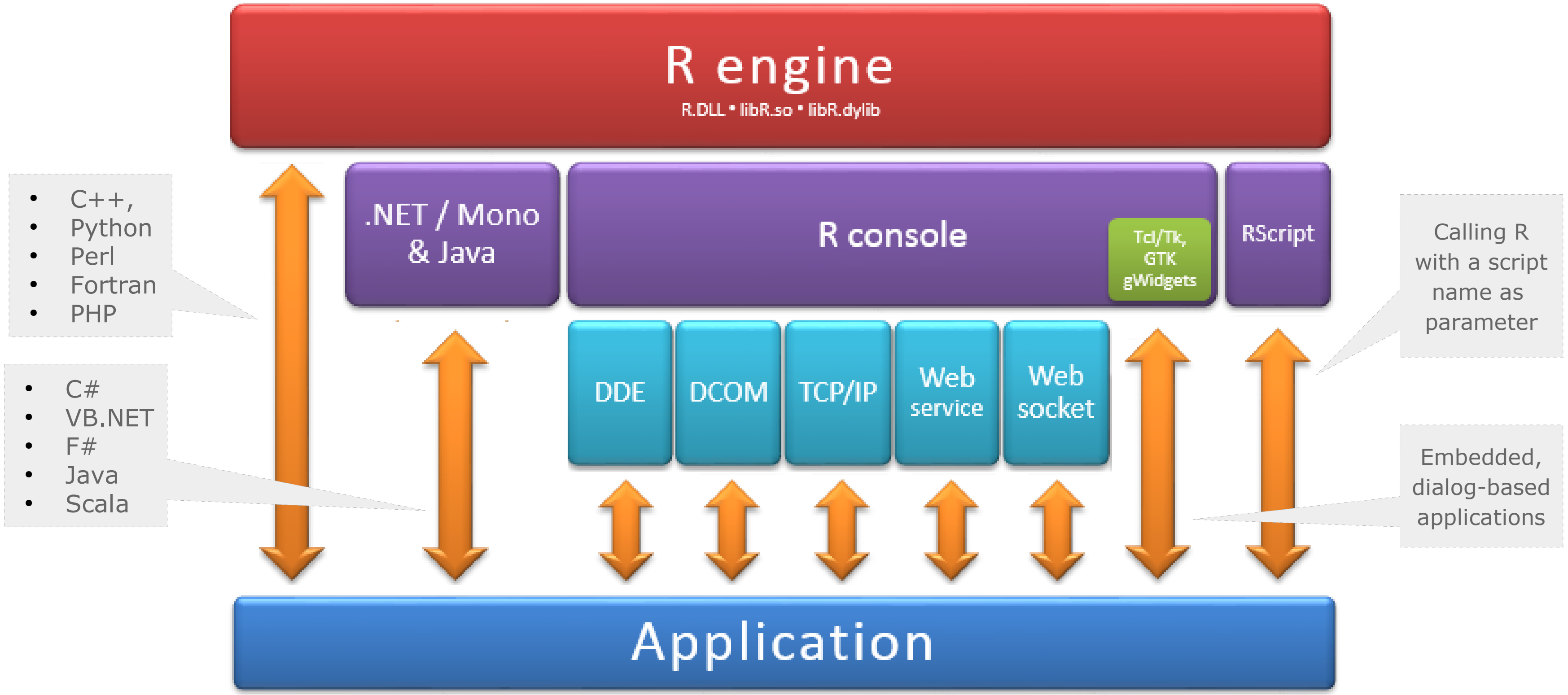
Data exchange

Communication with R

DCOM, Webservice, TCP/IP, IPC, DDE

Client workstations

…"there are so many ways of speaking"



R engine

R.DLL • libR.so • libR.dylib

- C++,
- Python
- Perl
- Fortran
- PHP

- C#
- VB.NET
- F#
- Java
- Scala

.NET / Mono & Java

R console

Tcl/Tk, GTK gWidgets

RScript

Calling R with a script name as parameter

DDE

DCOM

TCP/IP

Web service

Web socket

Embedded, dialog-based applications

Application

http://www.rforge.net/rJava http://rpy.sourceforge.net http://rclr.codeplex.com

http://rservecli.codeplex.com http://rcom.univie.ac.at http://ndde.codeplex.com

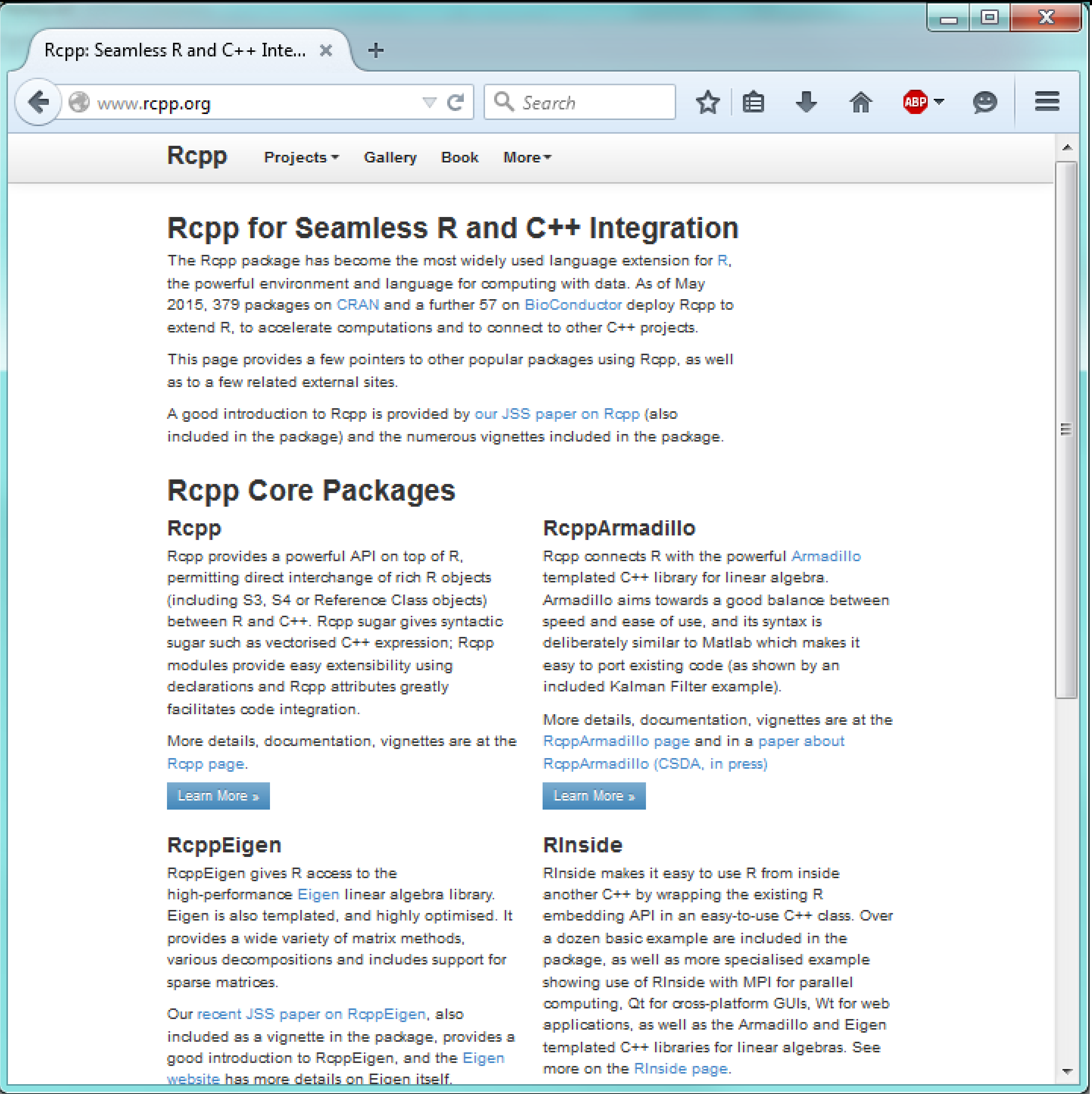
http://www.rforge.net/Rserve http://www.rcpp.org http://rdotnet.codeplex.com

http://dahl.byu.edu/software/jvmr http://www.rstudio.com/shiny

http://rpython.r-forge.r-project.org http://www.omegahat.org/RSPerl

http://www.goldenline.pl/forum/2478242/nowy-interfejs-r-net/ PL

Easy R – C++ integration



Rcpp for Seamless R and C++ Integration

The Rcpp package has become the most widely used language extension for R, the powerful environment and language for computing with data. As of May 2015, 379 packages on CRAN and a further 57 on BioConductor deploy Rcpp to extend R, to accelerate computations and to connect to other C++ projects.

This page provides a few pointers to other popular packages using Rcpp, as well as to a few related external sites.

A good introduction to Rcpp is provided by our JSS paper on Rcpp (also included in the package) and the numerous vignettes included in the package.

Rcpp Core Packages

Rcpp

Rcpp provides a powerful API on top of R, permitting direct interchange of rich R objects (including S3, S4 or Reference Class objects) between R and C++. Rcpp sugar gives syntactic sugar such as vectorised C++ expression; Rcpp modules provide easy extensibility using declarations and Rcpp attributes greatly facilitates code integration.

More details, documentation, vignettes are at the Rcpp page.

RcppArmadillo

Rcpp connects R with the powerful Armadillo templated C++ library for linear algebra. Armadillo aims towards a good balance between speed and ease of use, and its syntax is deliberately similar to Matlab which makes it easy to port existing code (as shown by an included Kalman Filter example).

More details, documentation, vignettes are at the RcppArmadillo page and in a paper about RcppArmadillo (CSDA, in press)

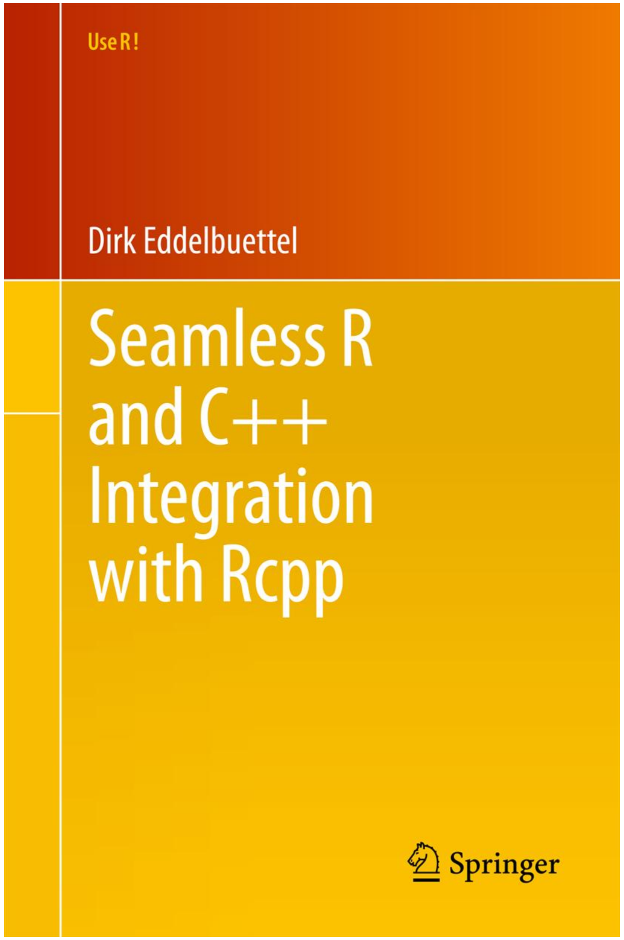
RcppEigen

RcppEigen gives R access to the high-performance Eigen linear algebra library. Eigen is also templated, and highly optimised. It provides a wide variety of matrix methods, various decompositions and includes support for sparse matrices.

Our recent JSS paper on RcppEigen, also included as a vignette in the package, provides a good introduction to RcppEigen, and the Eigen website has more details on Eigen itself.

RInside

RInside makes it easy to use R from inside another C++ by wrapping the existing R embedding API in an easy-to-use C++ class. Over a dozen basic example are included in the package, as well as more specialised example showing use of RInside with MPI for parallel computing, Qt for cross-platform GUIs, Wt for web applications, as well as the Armadillo and Eigen templated C++ libraries for linear algebras. See more on the RInside page.



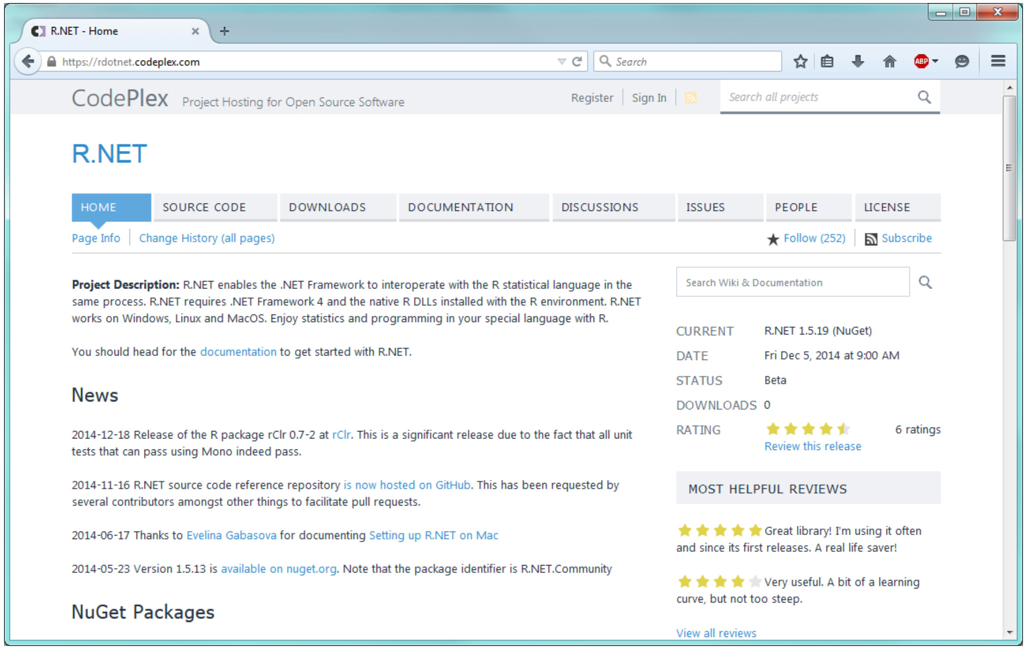
Use R!

Dirk Eddelbuettel

Seamless R and C++ Integration with Rcpp

Springer

Various R bindings



R.NET

Project Description: R.NET enables the .NET Framework to interoperate with the R statistical language in the same process. R.NET requires .NET Framework 4 and the native R DLLs installed with the R environment. R.NET works on Windows, Linux and MacOS. Enjoy statistics and programming in your special language with R.

You should head for the documentation to get started with R.NET.

News

2014-12-18 Release of the R package rClr 0.7-2 at rClr. This is a significant release due to the fact that all unit tests that can pass using Mono indeed pass.

2014-11-16 R.NET source code reference repository is now hosted on GitHub. This has been requested by several contributors amongst other things to facilitate pull requests.

2014-06-17 Thanks to Evelina Gabasova for documenting Setting up R.NET on Mac

2014-05-23 Version 1.5.13 is available on nuget.org. Note that the package identifier is R.NET.Community

NuGet Packages

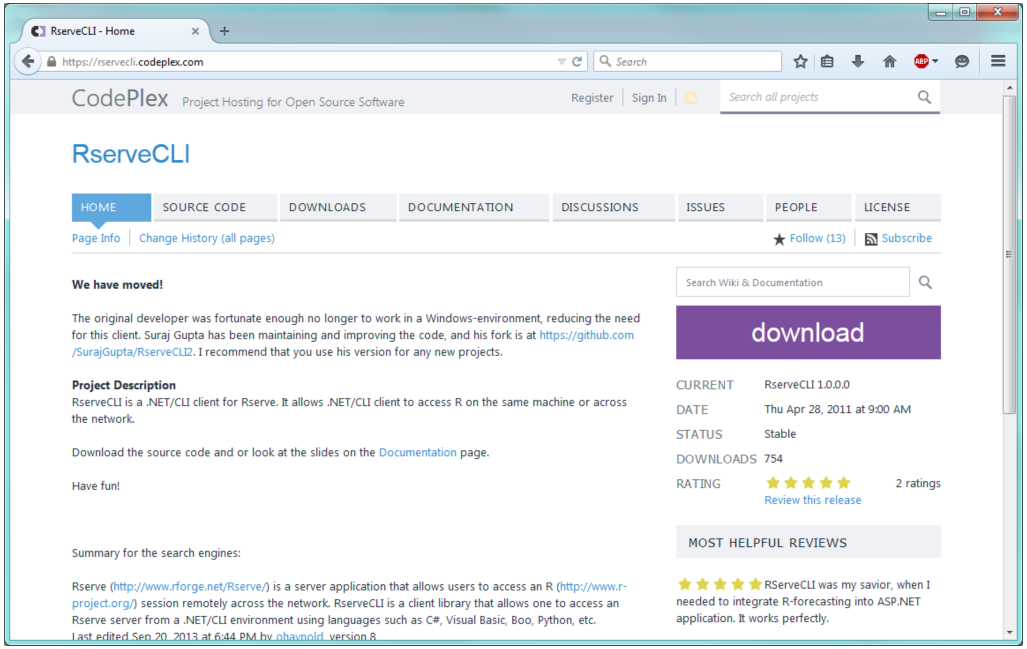
CURRENT R.NET 1.5.19 (NuGet)
DATE Fri Dec 5, 2014 at 9:00 AM
STATUS Beta
DOWNLOADS 0
RATING ★★★★½ 6 ratings
Review this release

MOST HELPFUL REVIEWS

★★★★★ Great library! I'm using it often and since its first releases. A real life saver!

★★★★☆ Very useful. A bit of a learning curve, but not too steep.

View all reviews



RserveCLI

We have moved!

The original developer was fortunate enough no longer to work in a Windows-environment, reducing the need for this client. Suraj Gupta has been maintaining and improving the code, and his fork is at https://github.com/SurajGupta/RserveCLI2. I recommend that you use his version for any new projects.

Project Description
RserveCLI is a .NET/CLI client for Rserve. It allows .NET/CLI client to access R on the same machine or across the network.

Download the source code and or look at the slides on the Documentation page.

Have fun!

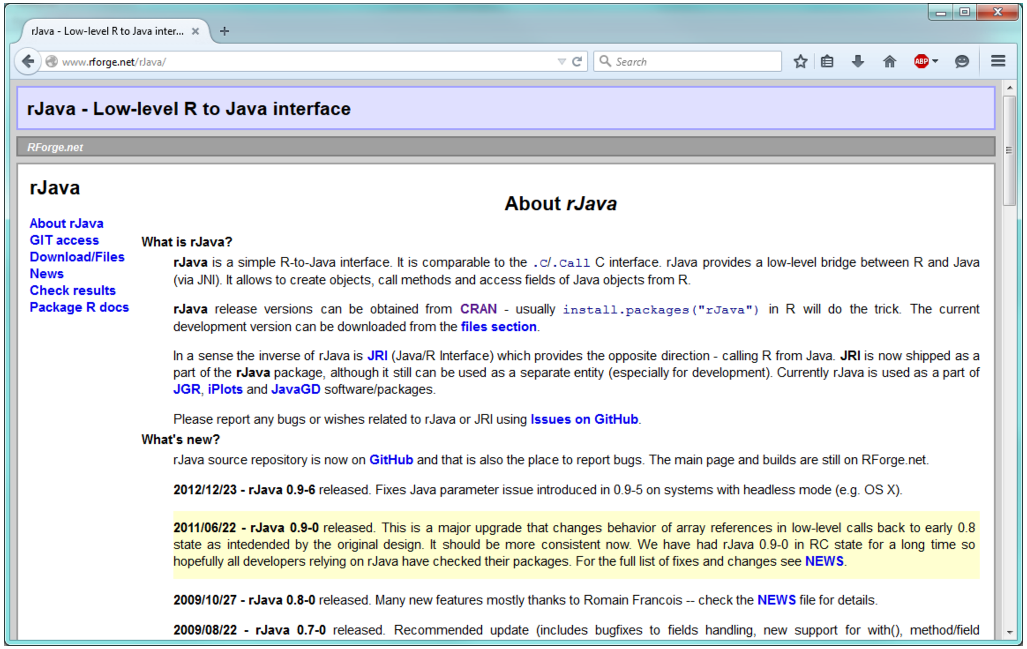
Summary for the search engines:

Rserve (http://www.rforge.net/Rserve/) is a server application that allows users to access an R (http://www.r-project.org/) session remotely across the network. RserveCLI is a client library that allows one to access an Rserve server from a .NET/CLI environment using languages such as C#, Visual Basic, Boo, Python, etc.
Last edited Sep 20, 2013 at 6:44 PM by ohaynold, version 8

CURRENT RserveCLI 1.0.0.0
DATE Thu Apr 28, 2011 at 9:00 AM
STATUS Stable
DOWNLOADS 754
RATING ★★★★★ 2 ratings
Review this release

MOST HELPFUL REVIEWS

★★★★★ RServeCLI was my savior, when I needed to integrate R-forecasting into ASP.NET application. It works perfectly.



rJava - Low-level R to Java interface

rJava

About *rJava*

What is rJava?

rJava is a simple R-to-Java interface. It is comparable to the `.C`/`.Call` C interface. rJava provides a low-level bridge between R and Java (via JNI). It allows to create objects, call methods and access fields of Java objects from R.

rJava release versions can be obtained from **CRAN** - usually `install.packages("rJava")` in R will do the trick. The current development version can be downloaded from the **files section**.

In a sense the inverse of rJava is **JRI** (Java/R Interface) which provides the opposite direction - calling R from Java. **JRI** is now shipped as a part of the **rJava** package, although it still can be used as a separate entity (especially for development). Currently rJava is used as a part of **JGR**, **iPlots** and **JavaGD** software/packages.

Please report any bugs or wishes related to rJava or JRI using **Issues on GitHub**.

What's new?

rJava source repository is now on **GitHub** and that is also the place to report bugs. The main page and builds are still on RForge.net.

2012/12/23 - rJava 0.9-6 released. Fixes Java parameter issue introduced in 0.9-5 on systems with headless mode (e.g. OS X).

2011/06/22 - rJava 0.9-0 released. This is a major upgrade that changes behavior of array references in low-level calls back to early 0.8 state as intededed by the original design. It should be more consistent now. We have had rJava 0.9-0 in RC state for a long time so hopefully all developers relying on rJava have checked their packages. For the full list of fixes and changes see **NEWS**.

2009/10/27 - rJava 0.8-0 released. Many new features mostly thanks to Romain Francois -- check the **NEWS** file for details.

2009/08/22 - rJava 0.7-0 released. Recommended update (includes bugfixes to fields handling, new support for with(), method/field

rpy2 - R in Python

About

rpy2 is a redesign and rewrite of **rpy**. It is providing a low-level interface to **R** from **Python**, a proposed high-level interface, including wrappers to graphical libraries, as well as R-like structures and functions.

Source and installation

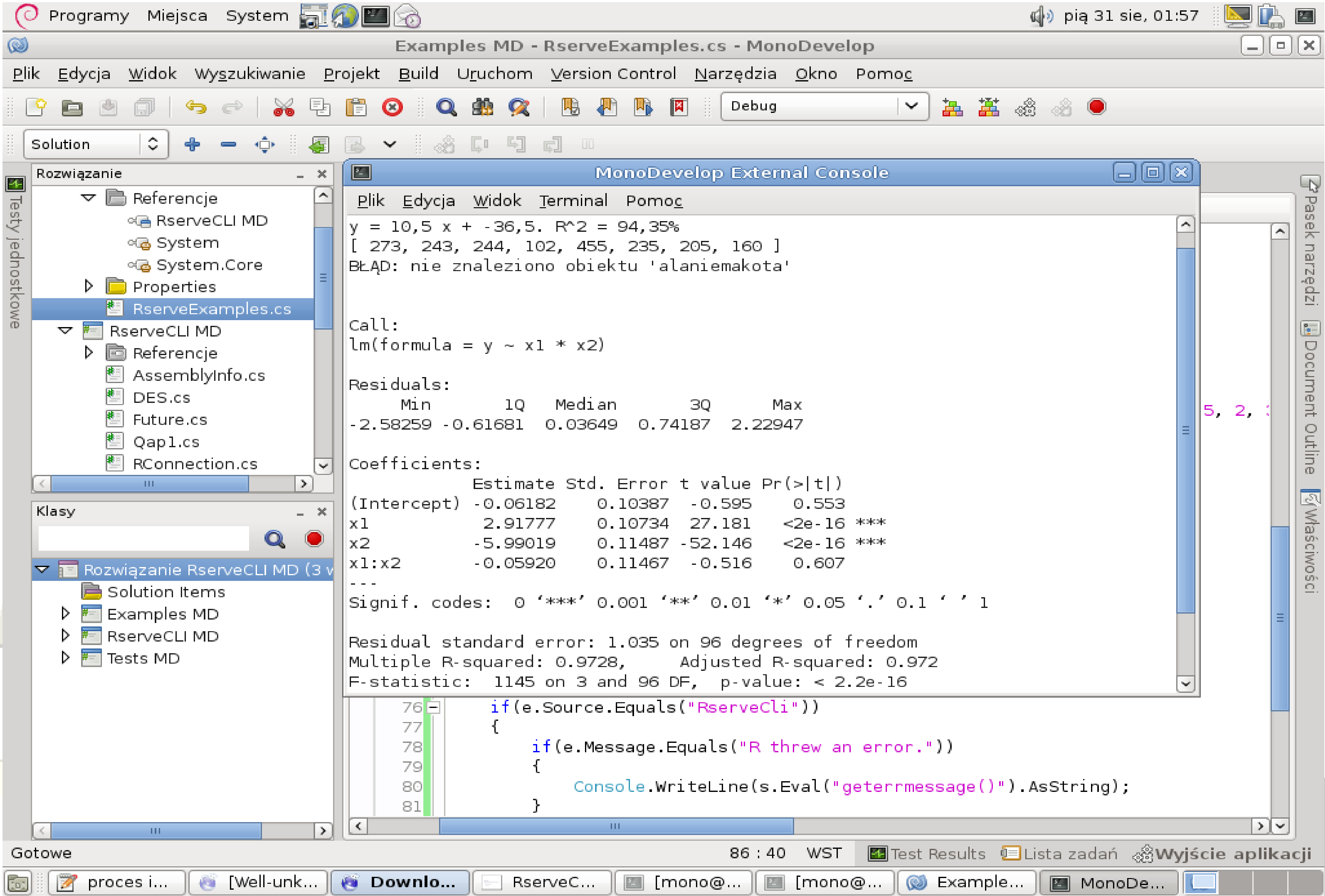
Questions and issues

Questions should preferably be asked on the rpy mailing-list on SourceForge, or on **StackOverflow**. Bugs, or wishes, can also go to the **bitbucket page** for the project.

Time travel

If looking for rpy or the old website, this

R under Mono on Debian via RserveCLI

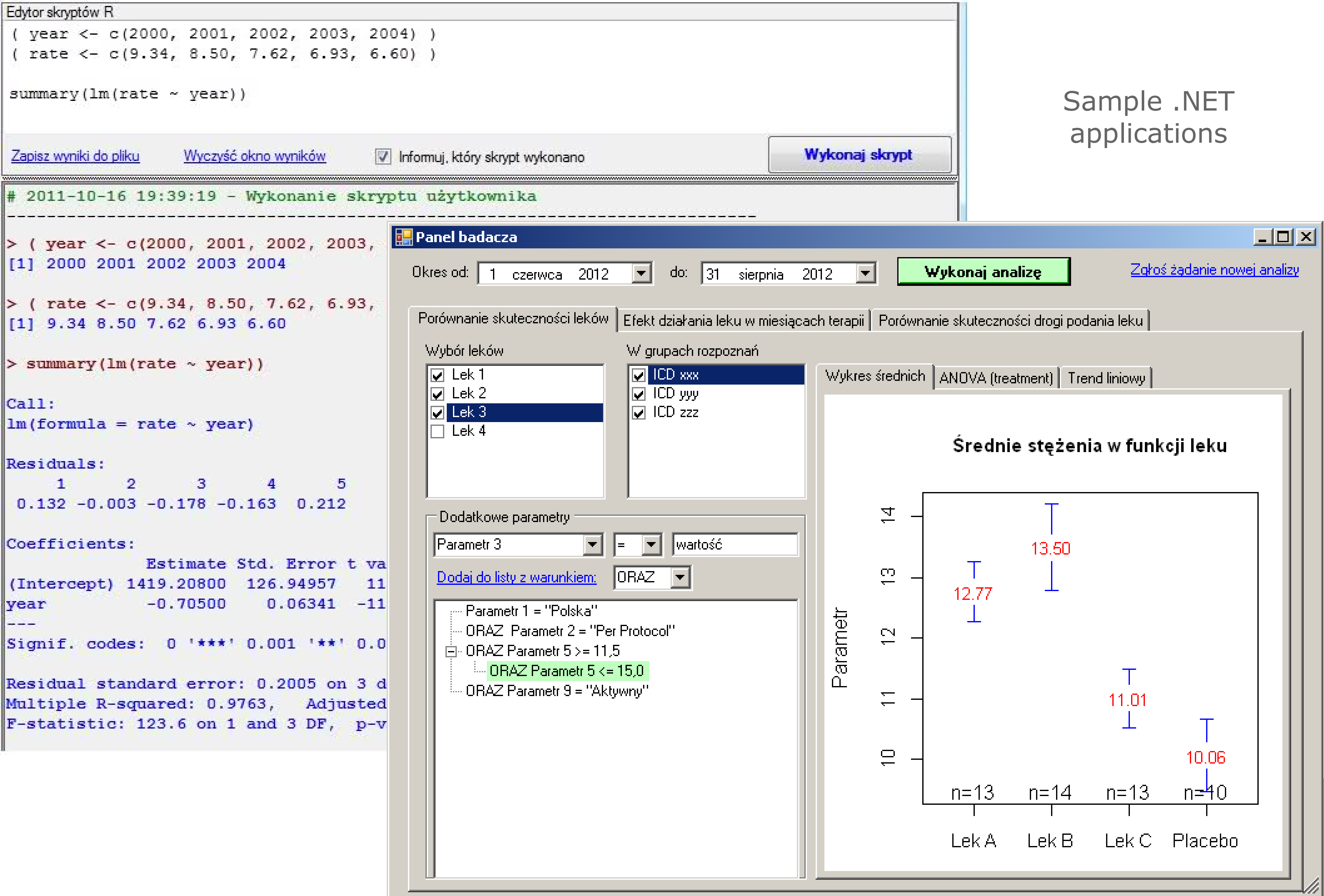


```
y = 10,5 x + -36,5. R^2 = 94,35%
[ 273, 243, 244, 102, 455, 235, 205, 160 ]
BŁĄD: nie znaleziono obiektu 'alaniemakota'


Call:
lm(formula = y ~ x1 * x2)

Residuals:
     Min       1Q   Median       3Q      Max
-2.58259 -0.61681  0.03649  0.74187  2.22947

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.06182    0.10387  -0.595    0.553
x1           2.91777    0.10734  27.181   <2e-16 ***
x2          -5.99019    0.11487 -52.146   <2e-16 ***
x1:x2       -0.05920    0.11467  -0.516    0.607
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.035 on 96 degrees of freedom
Multiple R-squared: 0.9728,     Adjusted R-squared: 0.972
F-statistic:  1145 on 3 and 96 DF,  p-value: < 2.2e-16
```

```
        if(e.Source.Equals("RserveCli"))
        {
            if(e.Message.Equals("R threw an error."))
            {
                Console.WriteLine(s.Eval("geterrmessage()").AsString);
            }
```

R.NET interface

Sample .NET applications



Edytor skryptów R

```
( year <- c(2000, 2001, 2002, 2003, 2004) )
( rate <- c(9.34, 8.50, 7.62, 6.93, 6.60) )

summary(lm(rate ~ year))
```

```
# 2011-10-16 19:39:19 - Wykonanie skryptu użytkownika
----------------------------------------------------------------------------

> ( year <- c(2000, 2001, 2002, 2003,
[1] 2000 2001 2002 2003 2004

> ( rate <- c(9.34, 8.50, 7.62, 6.93,
[1] 9.34 8.50 7.62 6.93 6.60

> summary(lm(rate ~ year))

Call:
lm(formula = rate ~ year)

Residuals:
     1      2      3      4      5
 0.132 -0.003 -0.178 -0.163  0.212

Coefficients:
              Estimate Std. Error t va
(Intercept) 1419.20800  126.94957   11
year          -0.70500    0.06341  -11
---
Signif. codes:  0 '***' 0.001 '**' 0.0

Residual standard error: 0.2005 on 3 d
Multiple R-squared: 0.9763,   Adjusted
F-statistic: 123.6 on 1 and 3 DF,  p-v
```

Panel badacza

Okres od: 1 czerwca 2012 do: 31 sierpnia 2012

Wybór leków
- ☑ Lek 1
- ☑ Lek 2
- ☑ Lek 3
- ☐ Lek 4

W grupach rozpoznań
- ☑ ICD xxx
- ☑ ICD yyy
- ☑ ICD zzz

Dodatkowe parametry

Parametr 3 = wartość

- Parametr 1 = "Polska"
- ORAZ Parametr 2 = "Per Protocol"
- ORAZ Parametr 5 >= 11,5
  - ORAZ Parametr 5 <= 15,0
- ORAZ Parametr 9 = "Aktywny"

shining web applications with R and Shiny

Shiny

by RStudio

A web application framework for R

Turn your analyses into interactive web applications

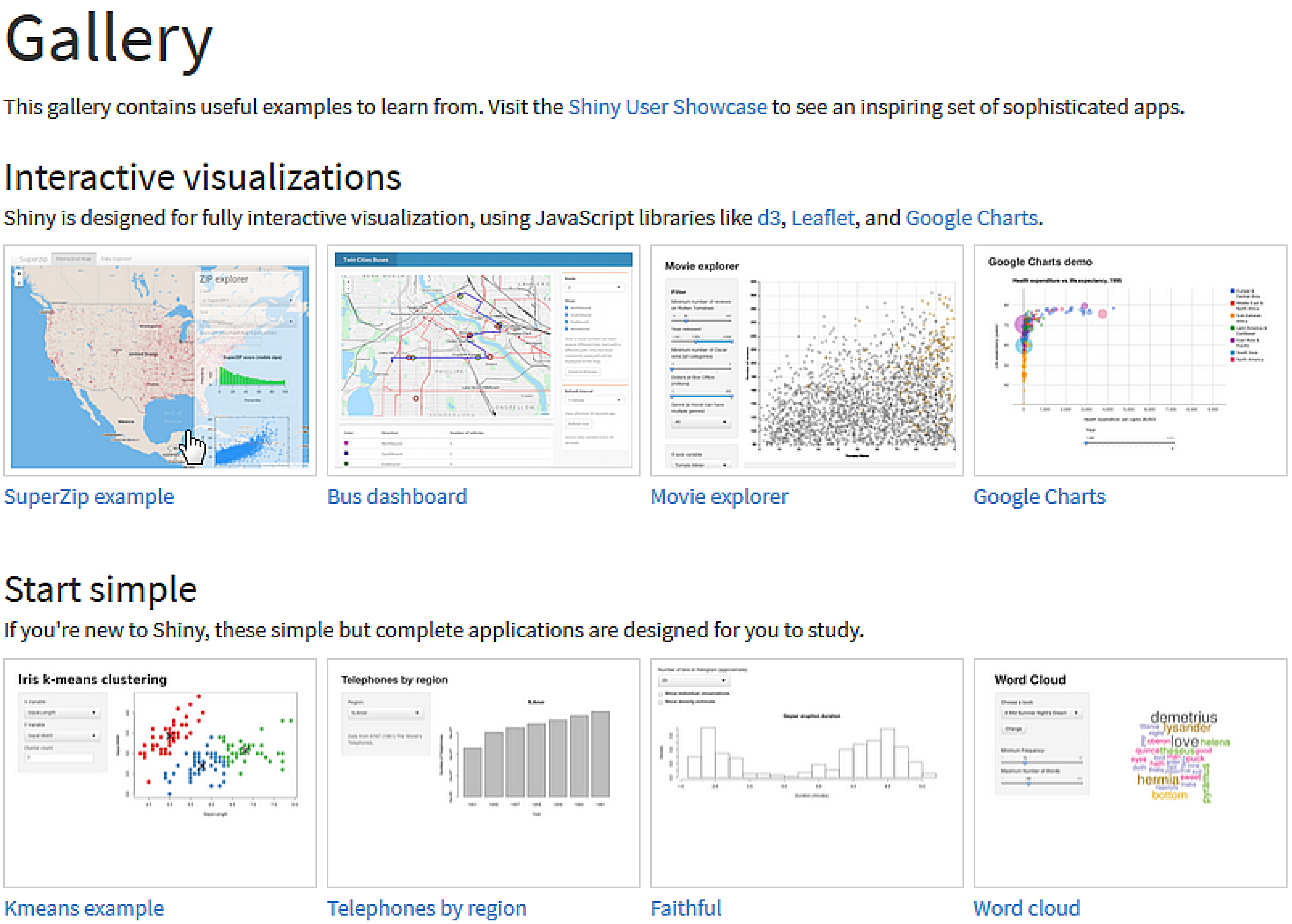
No HTML, CSS, or JavaScript knowledge required

Get inspired
(gallery)

Get started
(tutorial)

Go deeper
(articles)

shining web applications with R and Shiny



Gallery

This gallery contains useful examples to learn from. Visit the Shiny User Showcase to see an inspiring set of sophisticated apps.

Interactive visualizations

Shiny is designed for fully interactive visualization, using JavaScript libraries like d3, Leaflet, and Google Charts.

SuperZip example

Bus dashboard

Movie explorer

Google Charts

Start simple

If you're new to Shiny, these simple but complete applications are designed for you to study.

Kmeans example

Telephones by region

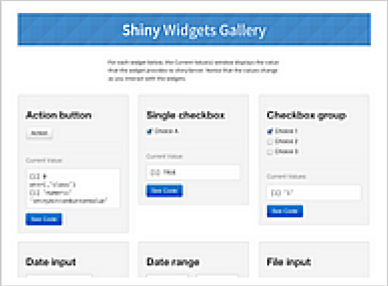
Faithful

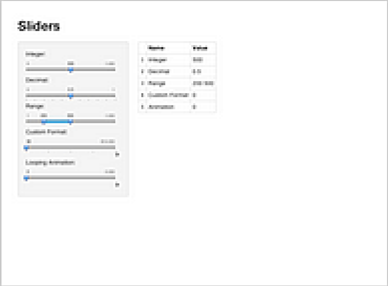
Word cloud

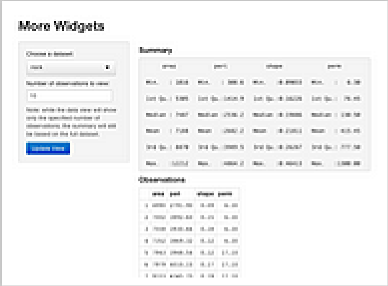
shining web applications with R and Shiny

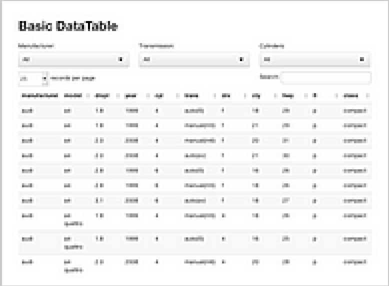
Widgets

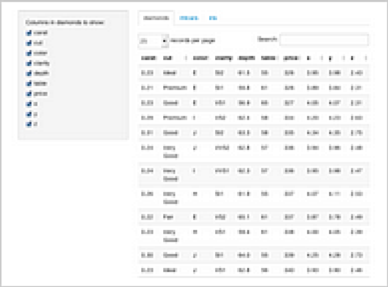
Get to know many of the input and output widgets that are available in Shiny with these examples.

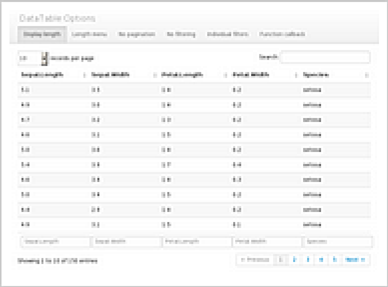

Widget Gallery


Sliders

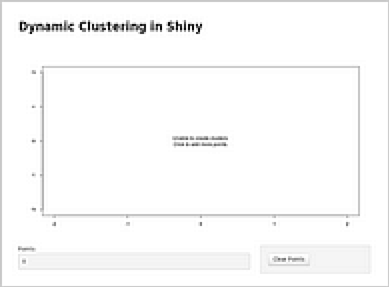

Widgets

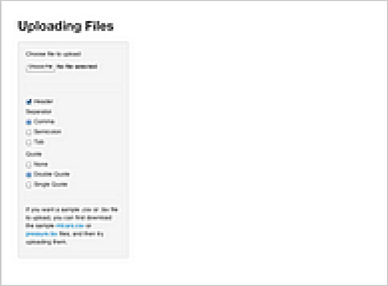

Basic DataTable

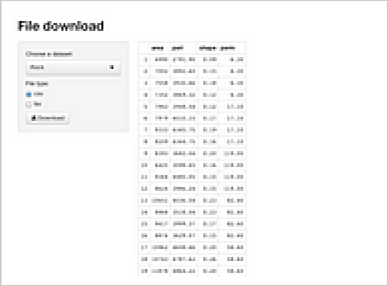

DataTables Demo

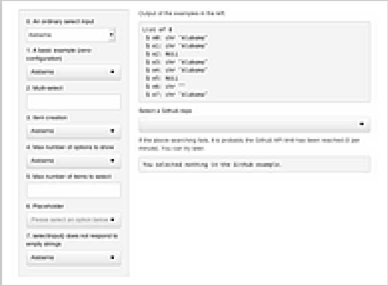

DataTables Options

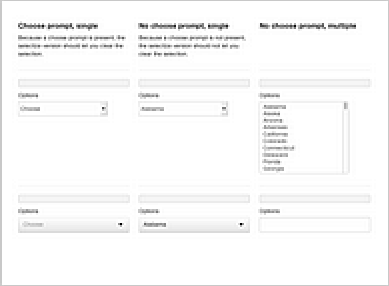

Date and date range

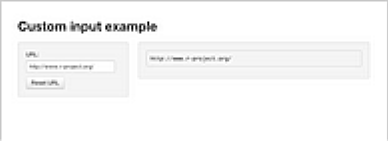

Dynamic Clustering


File Upload


File Download

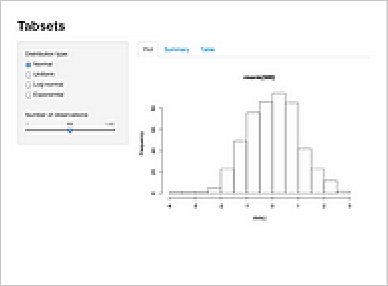

Selectize Examples


Selectize vs. Select



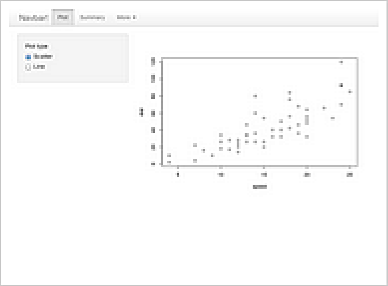
Application layout

Each example in this category demonstrates one or more of the functions you can use to organize app UI.

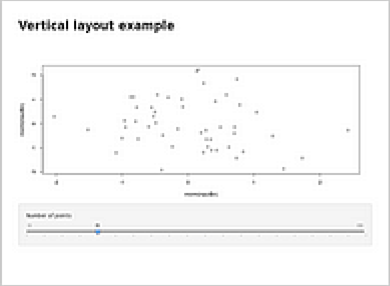


Tabsets

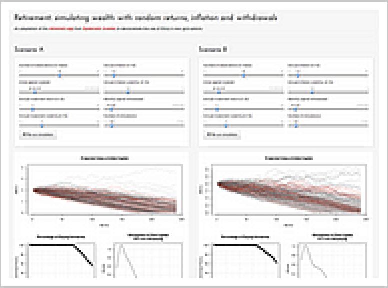
Plot plus three columns



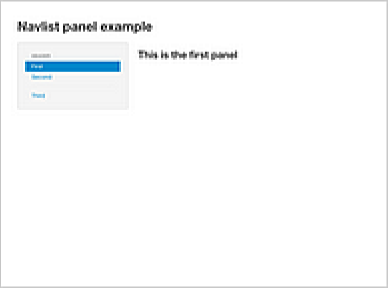
Navbar Example



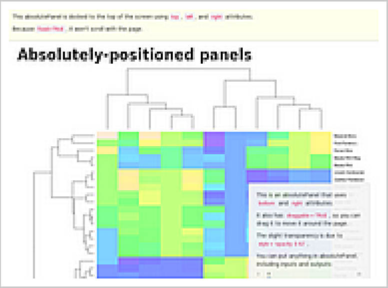
Vertical Layout



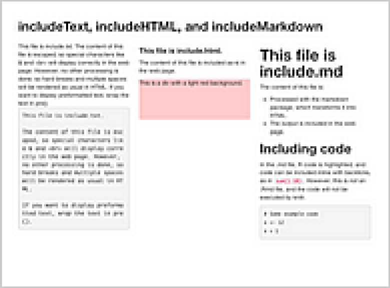
Retirement simulation



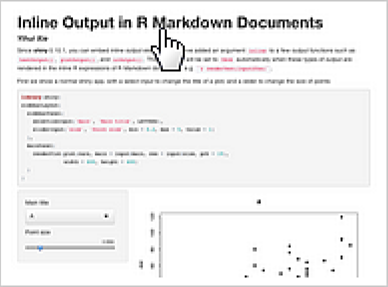
navlistPanel example



Absolutely-positioned panels

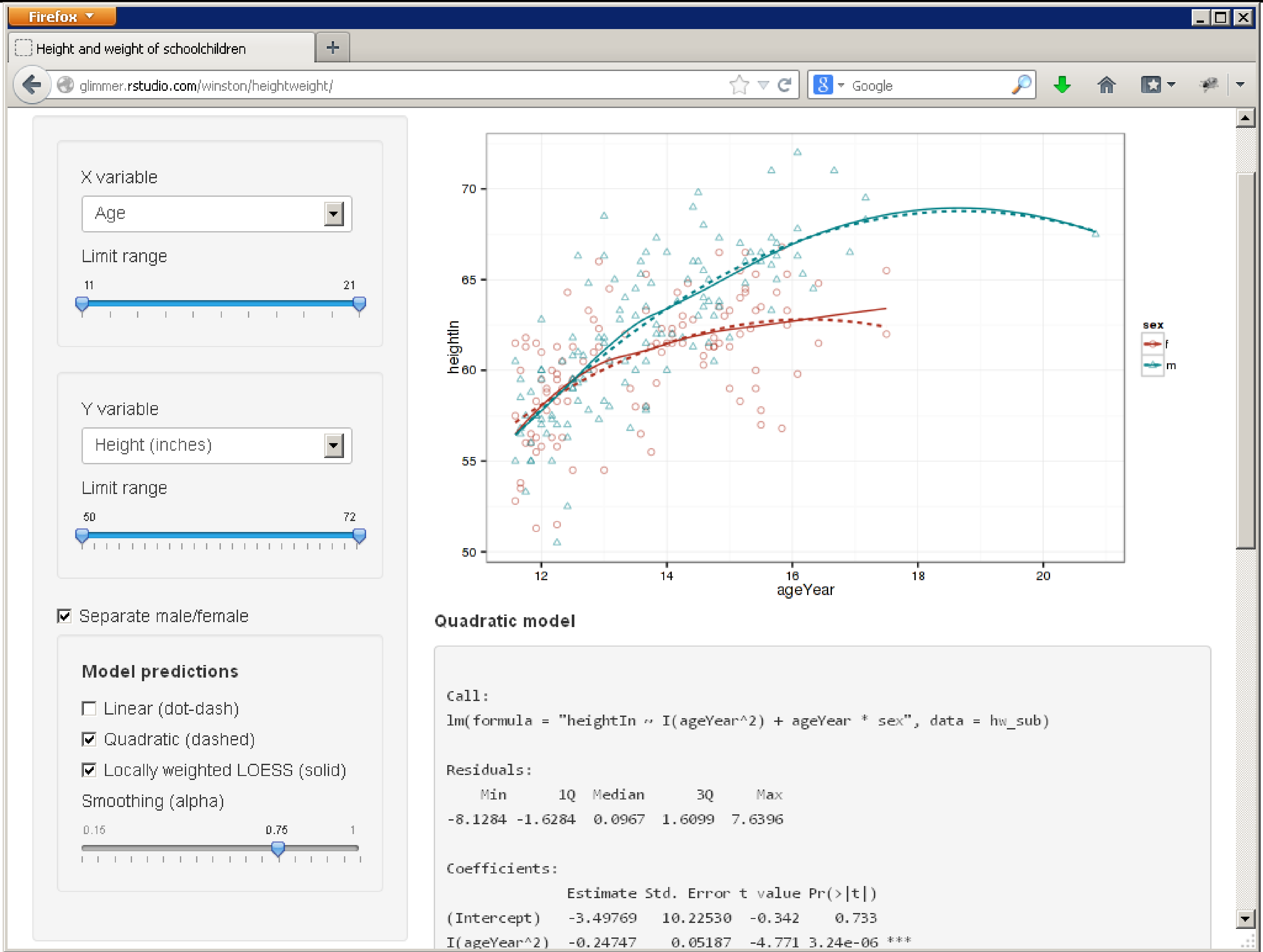


Including HTML, text, and Markdown files



Inline Output

easy (but shining) web applications with R and Shiny



X variable

Age

Limit range

11 21

Y variable

Height (inches)

Limit range

50 72

☑ Separate male/female

Model predictions

☐ Linear (dot-dash)

☑ Quadratic (dashed)

☑ Locally weighted LOESS (solid)

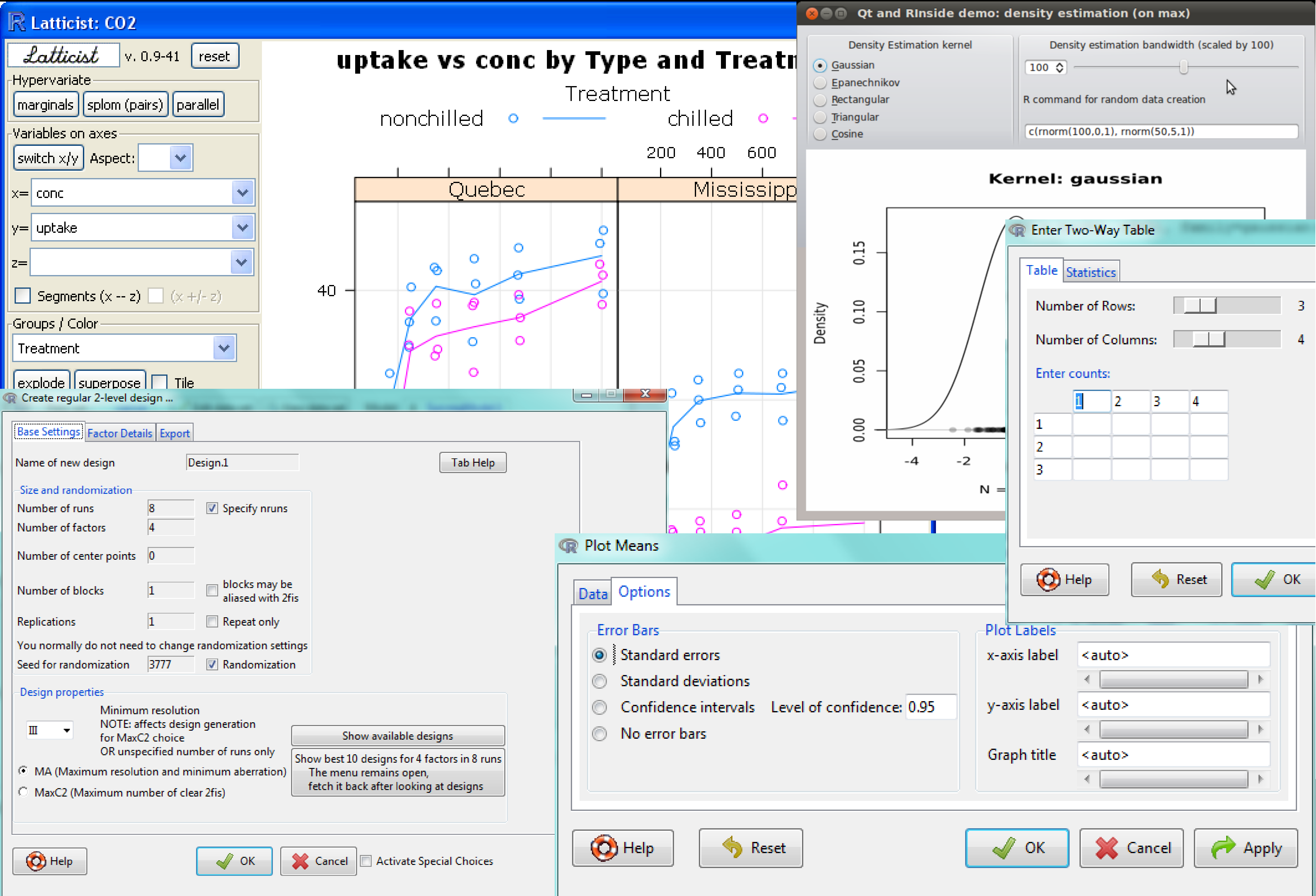
Smoothing (alpha)

0.15 0.75 1

Quadratic model

```
Call:
lm(formula = "heightIn ~ I(ageYear^2) + ageYear * sex", data = hw_sub)

Residuals:
    Min      1Q  Median      3Q     Max
-8.1284 -1.6284  0.0967  1.6099  7.6396

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3.49769   10.22530  -0.342    0.733
I(ageYear^2) -0.24747    0.05187  -4.771 3.24e-06 ***
```

some GUI hosted by R



Latticist: CO2

Latticist v. 0.9-41 reset

Hypervariate: marginals | splom (pairs) | parallel

Variables on axes: switch x/y Aspect:

x= conc

y= uptake

z=

☐ Segments (x -- z) ☐ (x +/- z)

Groups / Color: Treatment

explode superpose ☐ Tile

Qt and RInside demo: density estimation (on max)

Density Estimation kernel

- ● Gaussian
- ○ Epanechnikov
- ○ Rectangular
- ○ Triangular
- ○ Cosine

Density estimation bandwidth (scaled by 100): 100

R command for random data creation: c(rnorm(100,0,1), rnorm(50,5,1))

Enter Two-Way Table

Table | Statistics

Number of Rows: 3

Number of Columns: 4

Enter counts:

	1	2	3	4
1				
2				
3				

Help Reset OK

Create regular 2-level design ...

Base Settings | Factor Details | Export

Name of new design: Design.1 Tab Help

Size and randomization

Number of runs: 8 ☑ Specify nruns

Number of factors: 4

Number of center points: 0

Number of blocks: 1 ☐ blocks may be aliased with 2fis

Replications: 1 ☐ Repeat only

You normally do not need to change randomization settings

Seed for randomization: 3777 ☑ Randomization

Design properties

III — Minimum resolution
NOTE: affects design generation
for MaxC2 choice
OR unspecified number of runs only

Show available designs

Show best 10 designs for 4 factors in 8 runs
The menu remains open,
fetch it back after looking at designs

● MA (Maximum resolution and minimum aberration)

○ MaxC2 (Maximum number of clear 2fis)

Help OK Cancel ☐ Activate Special Choices

Plot Means

Data | Options

Error Bars

- ● Standard errors
- ○ Standard deviations
- ○ Confidence intervals Level of confidence: 0.95
- ○ No error bars

Plot Labels

x-axis label: <auto>

y-axis label: <auto>

Graph title: <auto>

Help Reset OK Cancel Apply

13 reasons why **you will** love GNU R

I	R is (extremely) cheap. In fact - it's free :)
II	R has (extremely) wide range of capabilities
III	R is (widely) supported by the world of science
IV	R is supported by the community
V	R is (increasingly) supported by the business
VI	R is able to read data in many formats
VII	Interoperability is easy to achieve
VIII	**R is truly cross-platform**
IX	R offers numerous ways of presenting data
X	There are many possibilities to optimize the code
XI	R is able to handle large amount of data
XII	R has a set of fancy tools and IDEs
XII	FDA accepted using R for drug trials!

R is truly cRoss-platform

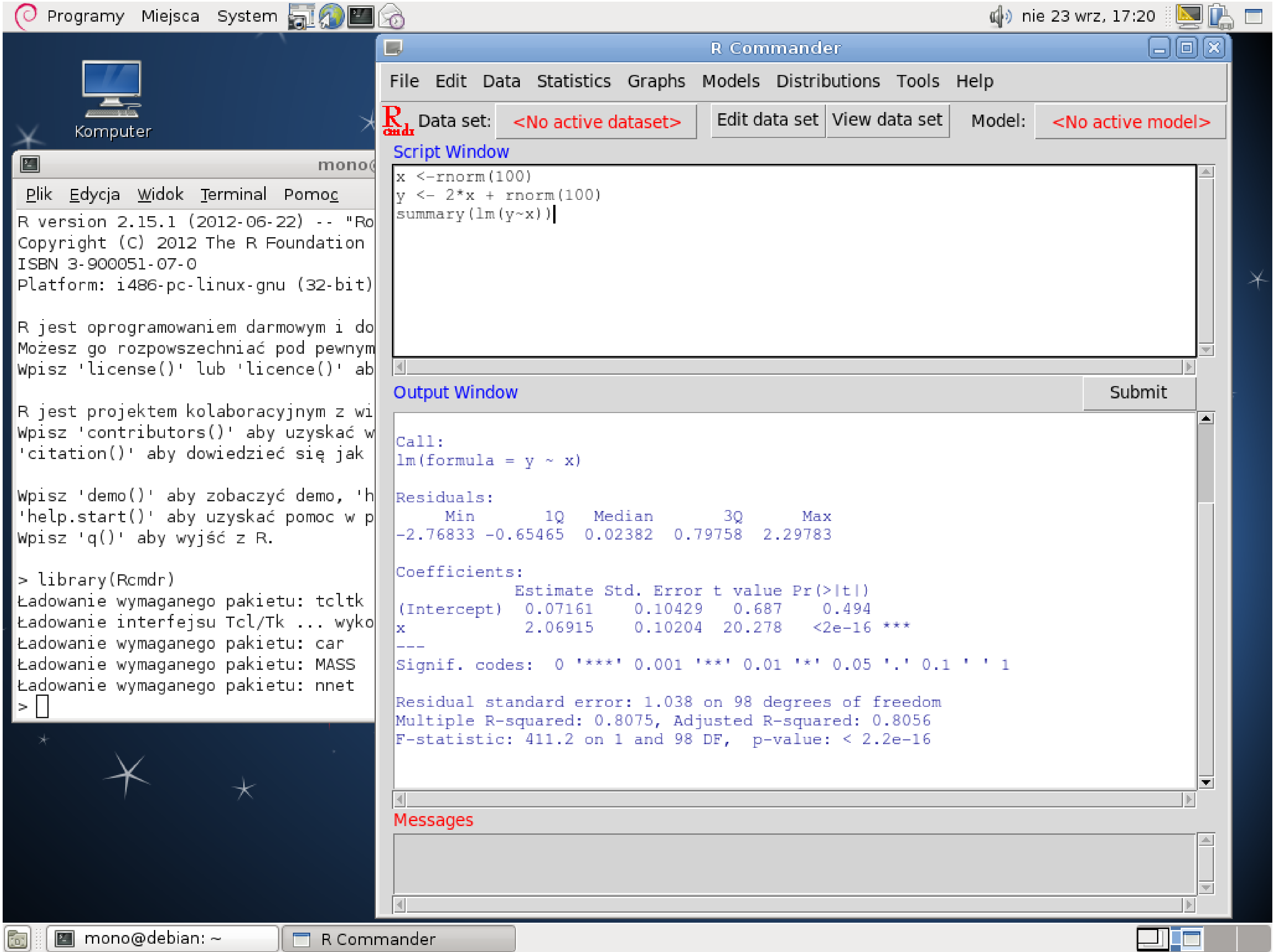
R can be run:

- on many operating systems: Windows (95-8.1), Unix (Solaris, AIX, HP-UX), Unix-like (GNU/Linux, FreeBSD), OSX and mobile: iOS, Android, Maemo
- on minicomputers like RaspberryPi (Raspbian OS)
- without installation, from any storage (USB device, CD/DVD, SD card)[portable] – just copy/paste the directory with R. You can have many versions of it.

Those properties make R an ideal base for creating:

- Handy, self-contained tools used by CRO's *Data Management* and *Biometrics* departments for easy monitoring the quality of collected data.
- Cheap, easy to deliver and backup (copy/paste) computing environments /free Linux + R + RStudio + PostgreSQL/MySQL + SVN + TrueCrypt + OpenVPN + VNC + rsync + Apache /
- Independent computing units based on simplified minicomputers.

GNU/Debian “Wheezy”



```
R version 2.15.1 (2012-06-22) -- "Ro
Copyright (C) 2012 The R Foundation
ISBN 3-900051-07-0
Platform: i486-pc-linux-gnu (32-bit)

R jest oprogramowaniem darmowym i do
Możesz go rozpowszechniać pod pewnym
Wpisz 'license()' lub 'licence()' ab

R jest projektem kolaboracyjnym z wi
Wpisz 'contributors()' aby uzyskać w
'citation()' aby dowiedzieć się jak

Wpisz 'demo()' aby zobaczyć demo, 'h
'help.start()' aby uzyskać pomoc w p
Wpisz 'q()' aby wyjść z R.

> library(Rcmdr)
Ładowanie wymaganego pakietu: tcltk
Ładowanie interfejsu Tcl/Tk ... wyko
Ładowanie wymaganego pakietu: car
Ładowanie wymaganego pakietu: MASS
Ładowanie wymaganego pakietu: nnet
> 
```

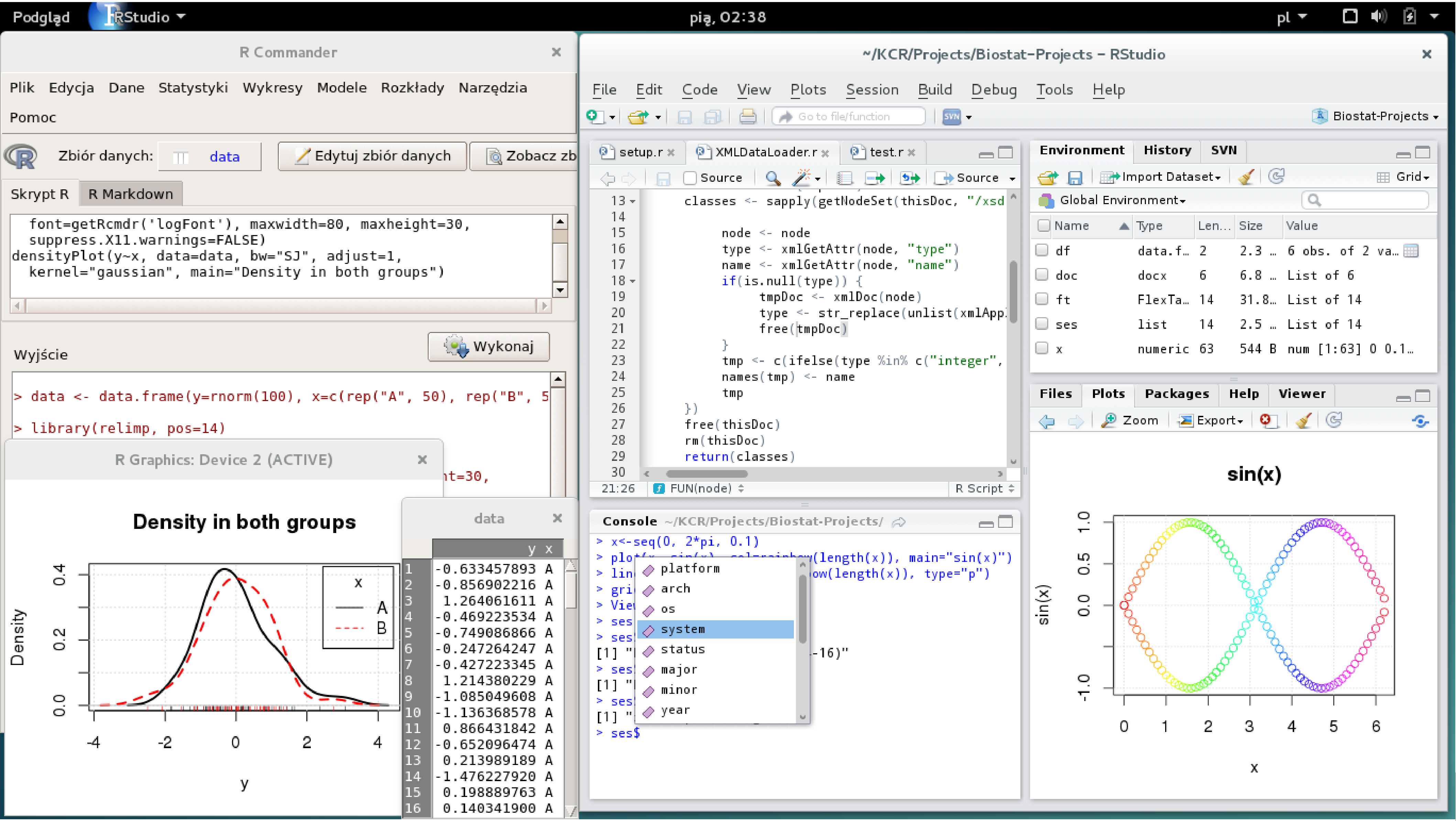
Data set: <No active dataset>

Model: <No active model>

```
x <-rnorm(100)
y <- 2*x + rnorm(100)
summary(lm(y~x))
```

```
Call:
lm(formula = y ~ x)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.76833 -0.65465  0.02382  0.79758  2.29783 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.07161    0.10429   0.687    0.494    
x            2.06915    0.10204  20.278   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.038 on 98 degrees of freedom
Multiple R-squared: 0.8075, Adjusted R-squared: 0.8056 
F-statistic: 411.2 on 1 and 98 DF,  p-value: < 2.2e-16
```

R and fRiends on Debian “Stretch”



```
  font=getRcmdr('logFont'), maxwidth=80, maxheight=30,
  suppress.X11.warnings=FALSE)
densityPlot(y~x, data=data, bw="SJ", adjust=1,
  kernel="gaussian", main="Density in both groups")
```

```
> data <- data.frame(y=rnorm(100), x=c(rep("A", 50), rep("B", 5

> library(relimp, pos=14)

nt=30,
```

	y	x
1	-0.633457893	A
2	-0.856902216	A
3	1.264061611	A
4	-0.469223534	A
5	-0.749086866	A
6	-0.247264247	A
7	-0.427223345	A
8	1.214380229	A
9	-1.085049608	A
10	-1.136368578	A
11	0.866431842	A
12	-0.652096474	A
13	0.213989189	A
14	-1.476227920	A
15	0.198889763	A
16	0.140341900	A

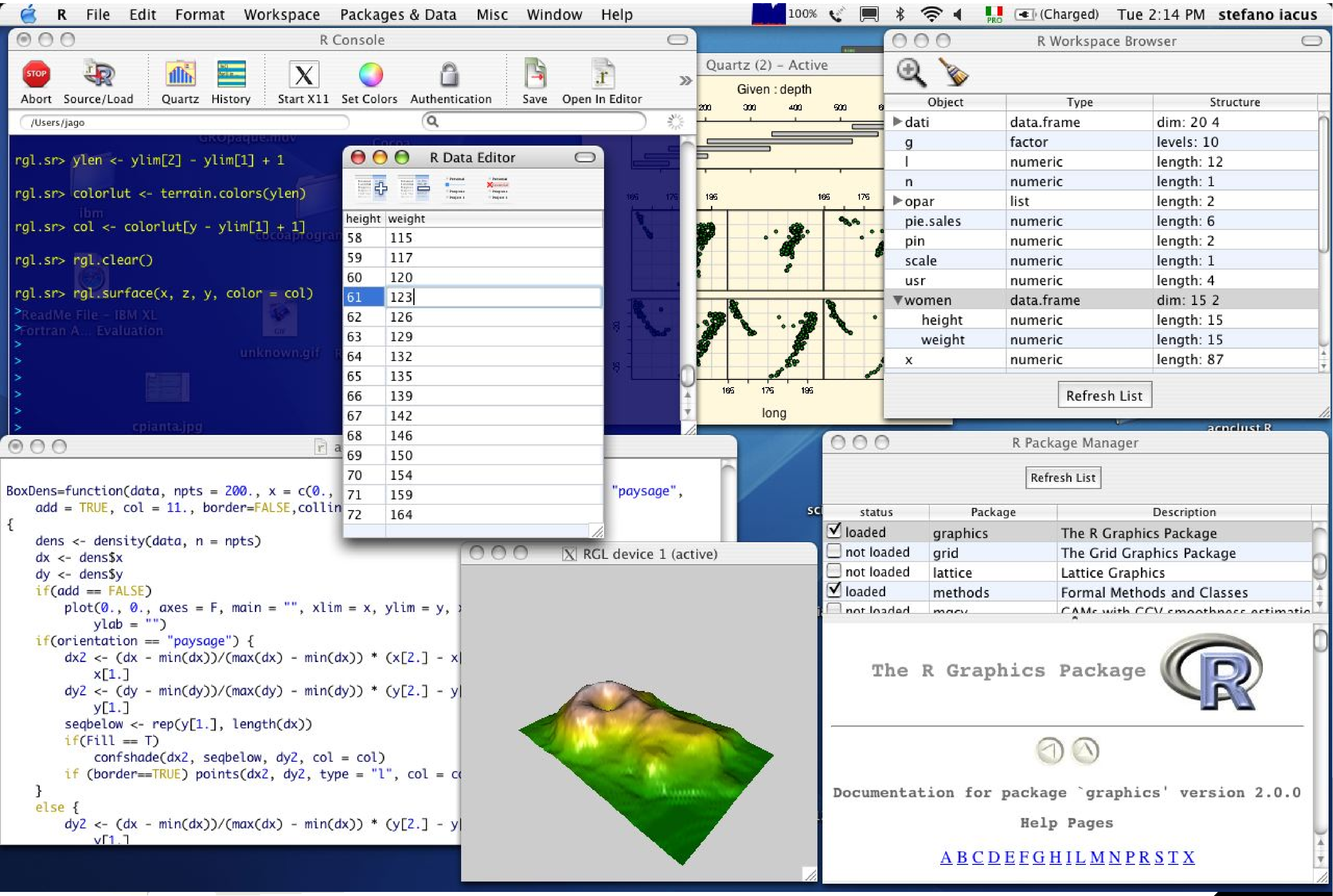
```
    classes <- sapply(getNodeSet(thisDoc, "/xsd

        node <- node
        type <- xmlGetAttr(node, "type")
        name <- xmlGetAttr(node, "name")
        if(is.null(type)) {
            tmpDoc <- xmlDoc(node)
            type <- str_replace(unlist(xmlApp
            free(tmpDoc)
        }
        tmp <- c(ifelse(type %in% c("integer",
        names(tmp) <- name
        tmp
    })
    free(thisDoc)
    rm(thisDoc)
    return(classes)
```

☐ Name	Type	Len…	Size	Value
☐ df	data.f…	2	2.3 …	6 obs. of 2 va…
☐ doc	docx	6	6.8 …	List of 6
☐ ft	FlexTa…	14	31.8…	List of 14
☐ ses	list	14	2.5 …	List of 14
☐ x	numeric	63	544 B	num [1:63] 0 0.1…

```
> x<-seq(0, 2*pi, 0.1)
> plo                                 (length(x)), main="sin(x)")
> lin                                 ow(length(x)), type="p")
> gri
> Vie
> ses
> ses
[1] "                                 -16)"
> ses
[1] "
> ses
[1] "
> ses$
```

platform
arch
os
system
status
major
minor
year

MacOS / OS X



```
rgl.sr> ylen <- ylim[2] - ylim[1] + 1

rgl.sr> colorlut <- terrain.colors(ylen)

rgl.sr> col <- colorlut[y - ylim[1] + 1]

rgl.sr> rgl.clear()

rgl.sr> rgl.surface(x, z, y, color = col)
>
>
>
>
>
>
>
>
```

height	weight
58	115
59	117
60	120
61	123
62	126
63	129
64	132
65	135
66	139
67	142
68	146
69	150
70	154
71	159
72	164

Object	Type	Structure
▶ dati	data.frame	dim: 20 4
g	factor	levels: 10
l	numeric	length: 12
n	numeric	length: 1
▶ opar	list	length: 2
pie.sales	numeric	length: 6
pin	numeric	length: 2
scale	numeric	length: 1
usr	numeric	length: 4
▼ women	data.frame	dim: 15 2
height	numeric	length: 15
weight	numeric	length: 15
x	numeric	length: 87

status	Package	Description
☑ loaded	graphics	The R Graphics Package
☐ not loaded	grid	The Grid Graphics Package
☐ not loaded	lattice	Lattice Graphics
☑ loaded	methods	Formal Methods and Classes

```
BoxDens=function(data, npts = 200., x = c(0.,                "paysage",
    add = TRUE, col = 11., border=FALSE,collin
{
    dens <- density(data, n = npts)
    dx <- dens$x
    dy <- dens$y
    if(add == FALSE)
        plot(0., 0., axes = F, main = "", xlim = x, ylim = y,
            ylab = "")
    if(orientation == "paysage") {
        dx2 <- (dx - min(dx))/(max(dx) - min(dx)) * (x[2.] - x
            x[1.]
        dy2 <- (dy - min(dy))/(max(dy) - min(dy)) * (y[2.] - y
            y[1.]
        seqbelow <- rep(y[1.], length(dx))
        if(Fill == T)
            confshade(dx2, seqbelow, dy2, col = col)
        if (border==TRUE) points(dx2, dy2, type = "l", col = c
    }
    else {
        dy2 <- (dx - min(dx))/(max(dx) - min(dx)) * (y[2.] - y
```

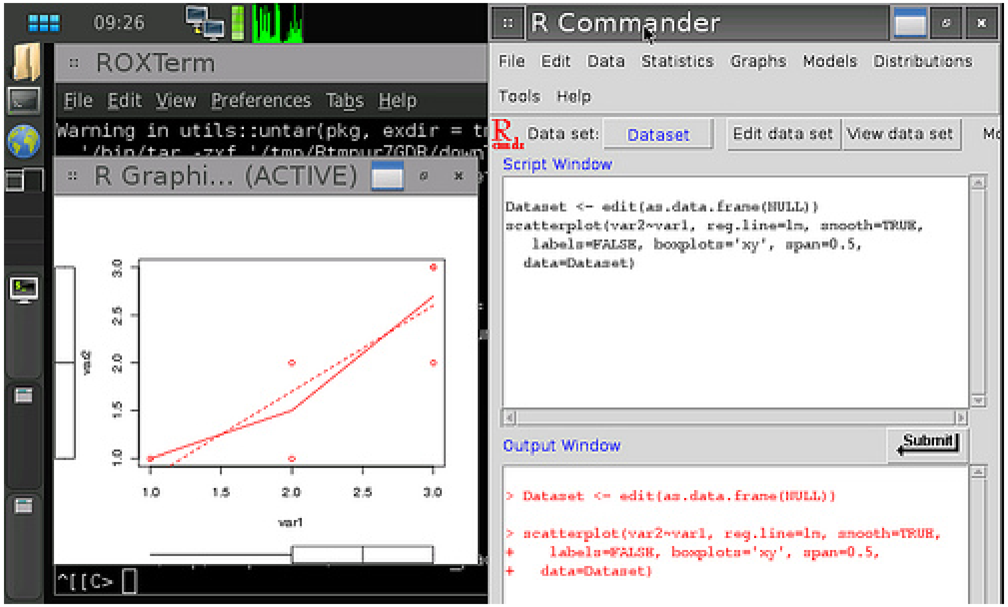
The R Graphics Package

Documentation for package `graphics' version 2.0.0

Help Pages

A B C D E F G H I L M N P R S T X

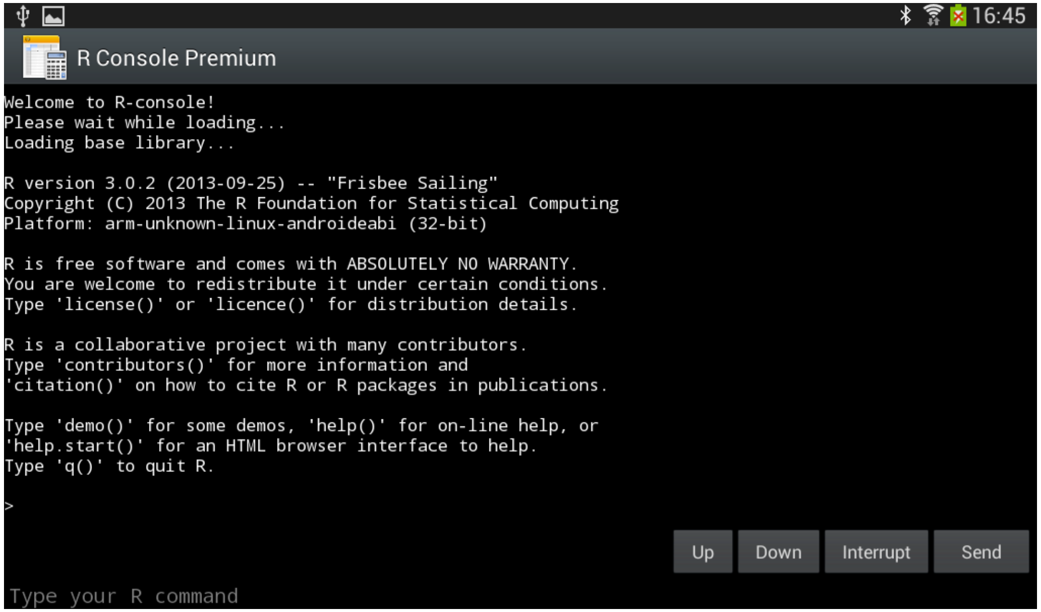
R on mobile platforms



```
Warning in utils::untar(pkg, exdir = tr
  '/bin/tar -zxf '/tmp/RtmpurZGDR/down
^[[C>
```

```
Dataset <- edit(as.data.frame(NULL))
scatterplot(var2~var1, reg.line=lm, smooth=TRUE,
   labels=FALSE, boxplots='xy', span=0.5,
  data=Dataset)
```

```
> Dataset <- edit(as.data.frame(NULL))

> scatterplot(var2~var1, reg.line=lm, smooth=TRUE,
+    labels=FALSE, boxplots='xy', span=0.5,
+   data=Dataset)
```



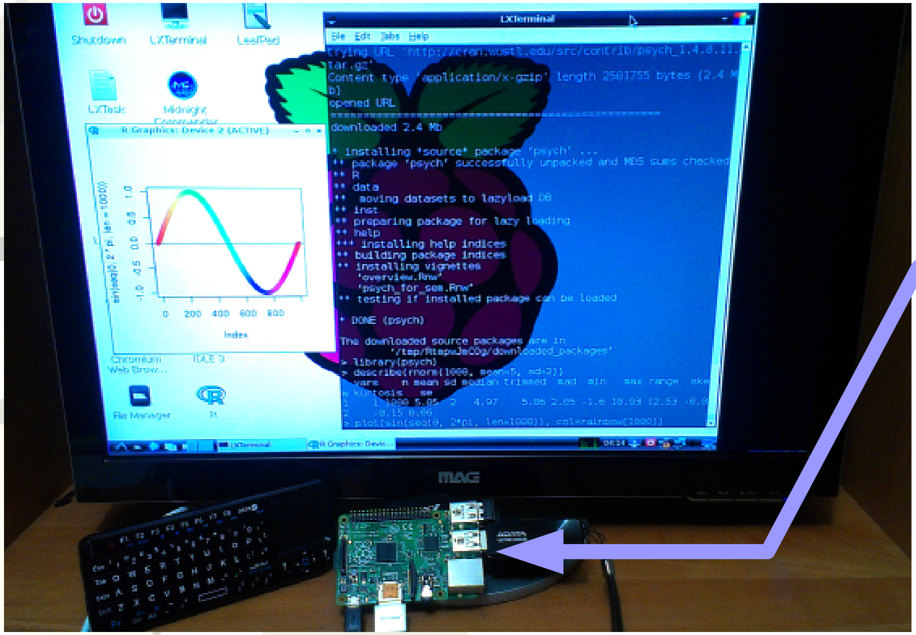
```
Welcome to R-console!
Please wait while loading...
Loading base library...

R version 3.0.2 (2013-09-25) -- "Frisbee Sailing"
Copyright (C) 2013 The R Foundation for Statistical Computing
Platform: arm-unknown-linux-androideabi (32-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>
```

Maemo (Nokia N900)

https://www.flickr.com/photos/n900user/

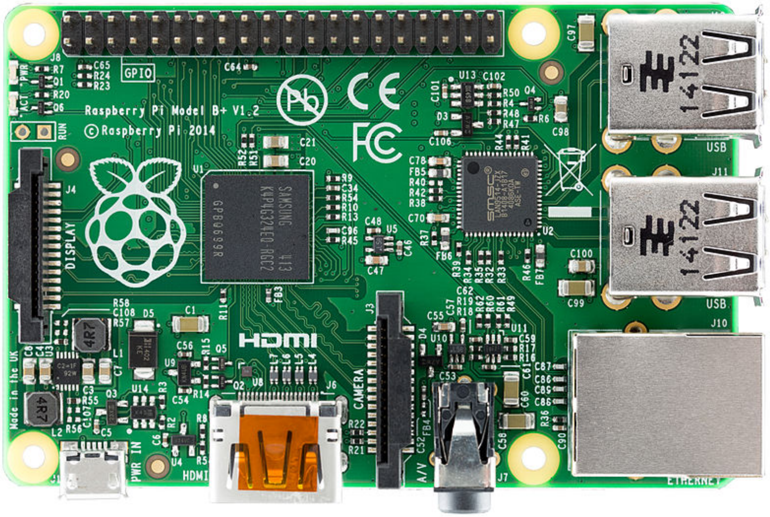
Android

R Console Premium (Google Play)



```
trying URL 'http://cran.wustl.edu/src/contrib/psych_1.4.8.11.
tar.gz'
Content type 'application/x-gzip' length 2501755 bytes (2.4 M
b)
opened URL
==================================================
downloaded 2.4 Mb

* installing *source* package 'psych' ...
** package 'psych' successfully unpacked and MD5 sums checked
** R
** data
**  moving datasets to lazyload DB
** inst
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
   'overview.Rnw'
   'psych_for_sem.Rnw'
** testing if installed package can be loaded

* DONE (psych)

The downloaded source packages are in
        '/tmp/RtmpwJmCOg/downloaded_packages'
> library(psych)
> describe(rnorm(1000, mean=5, sd=2))
```

```
> plot(sin(seq(0, 2*pi, len=1000)), col=rainbow(1000))
```

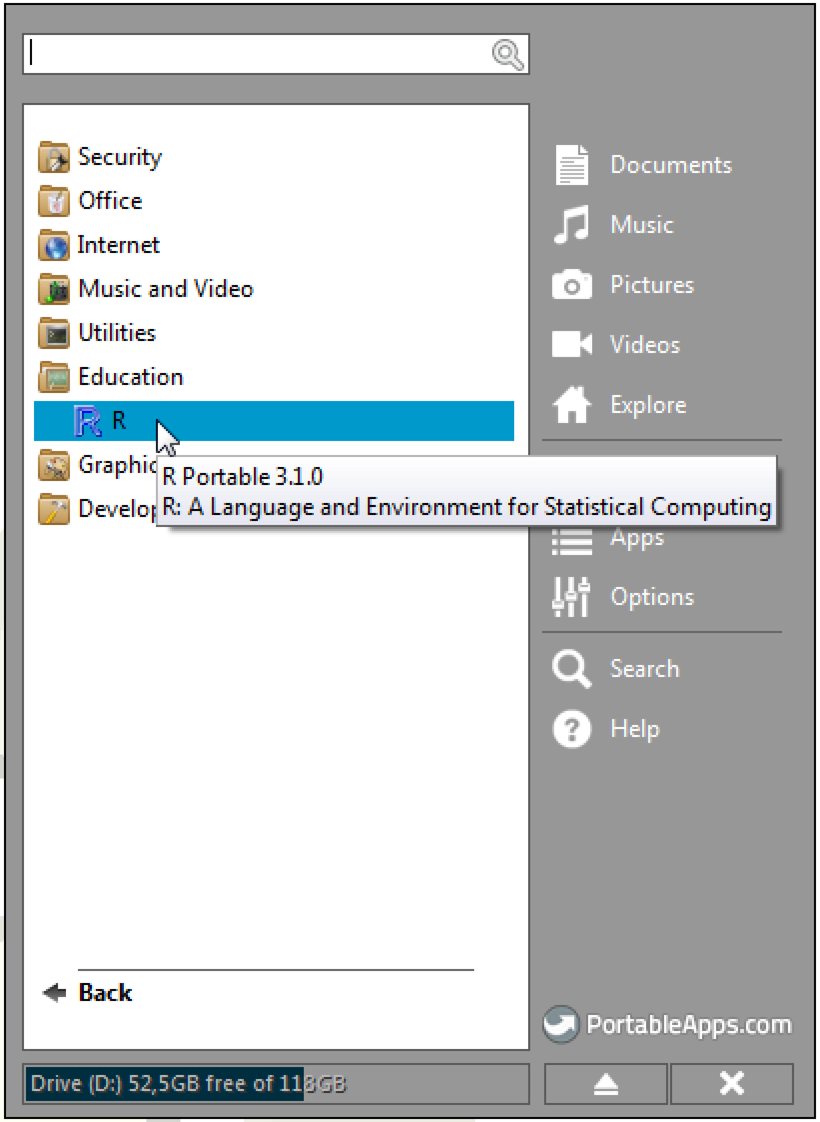


Raspberry Pi B+

PoRtable R

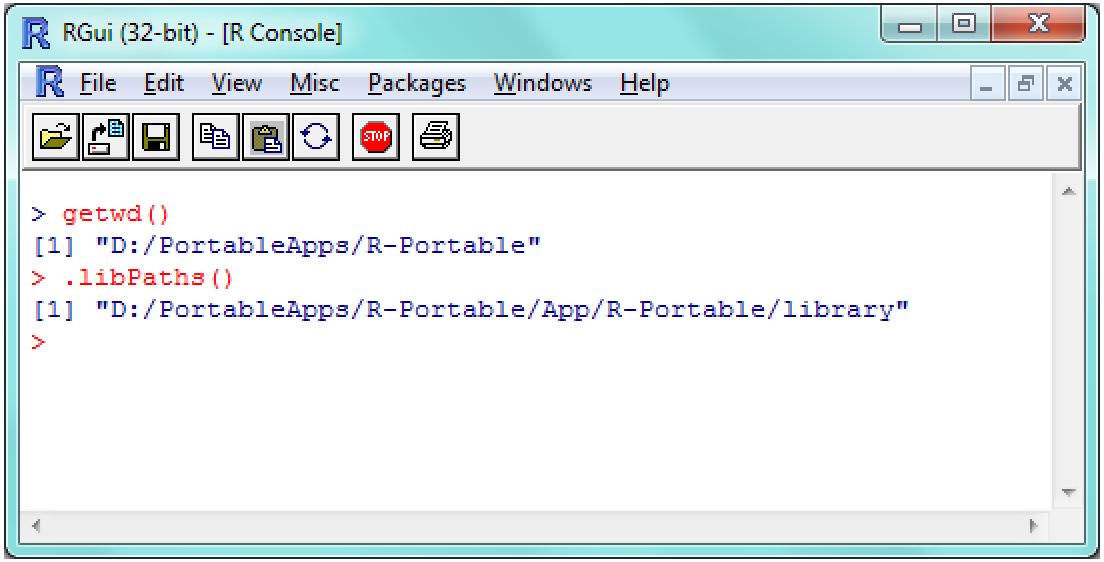
http://sourceforge.net/projects/rportable/

(also older, 2.11 – 2.15 versions are available!)

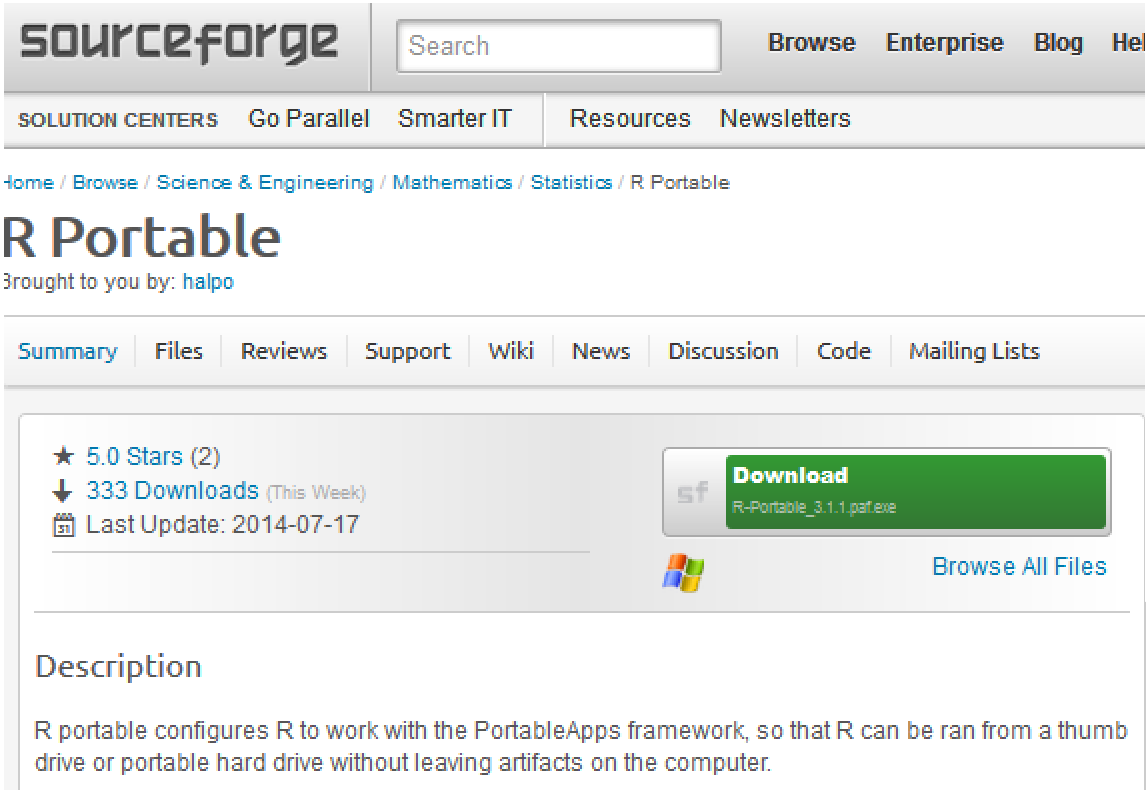


R Portable 3.1.0
R: A Language and Environment for Statistical Computing

Drive (D:) 52,5GB free of 118GB



```
> getwd()
[1] "D:/PortableApps/R-Portable"
> .libPaths()
[1] "D:/PortableApps/R-Portable/App/R-Portable/library"
>
```



Home / Browse / Science & Engineering / Mathematics / Statistics / R Portable

R Portable

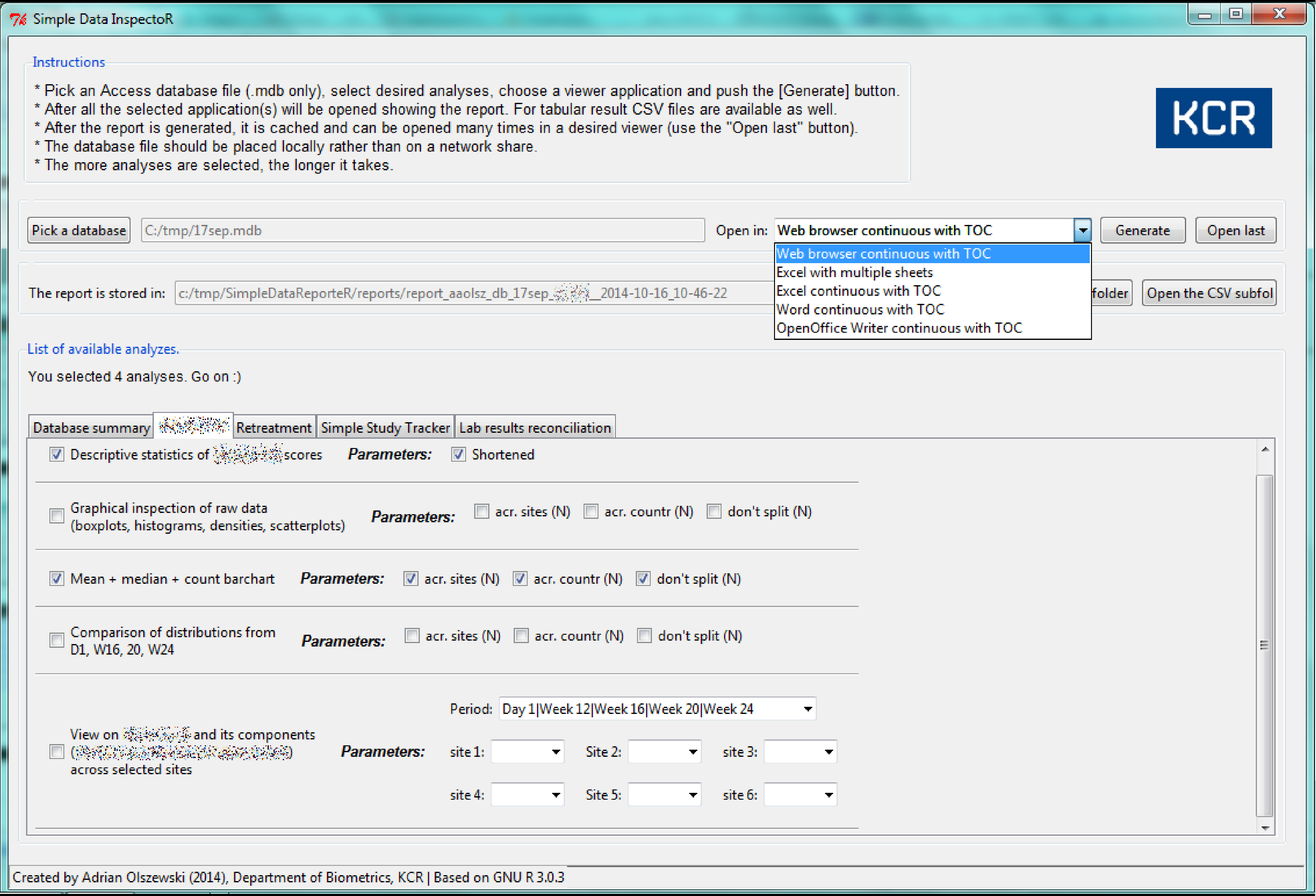
Brought to you by: halpo

★ 5.0 Stars (2)
333 Downloads (This Week)
Last Update: 2014-07-17

Description

R portable configures R to work with the PortableApps framework, so that R can be ran from a thumb drive or portable hard drive without leaving artifacts on the computer.

PoRtable in use – Simple Data InspectoR



Instructions

* Pick an Access database file (.mdb only), select desired analyses, choose a viewer application and push the [Generate] button.
* After all the selected application(s) will be opened showing the report. For tabular result CSV files are available as well.
* After the report is generated, it is cached and can be opened many times in a desired viewer (use the "Open last" button).
* The database file should be placed locally rather than on a network share.
* The more analyses are selected, the longer it takes.

C:/tmp/17sep.mdb

Open in: Web browser continuous with TOC

The report is stored in: c:/tmp/SimpleDataReporteR/reports/report_aaolsz_db_17sep_[redacted]_2014-10-16_10-46-22

List of available analyzes.

You selected 4 analyses. Go on :)

☑ Descriptive statistics of [redacted] scores — ***Parameters:*** ☑ Shortened

☐ Graphical inspection of raw data (boxplots, histograms, densities, scatterplots) — ***Parameters:*** ☐ acr. sites (N) ☐ acr. countr (N) ☐ don't split (N)

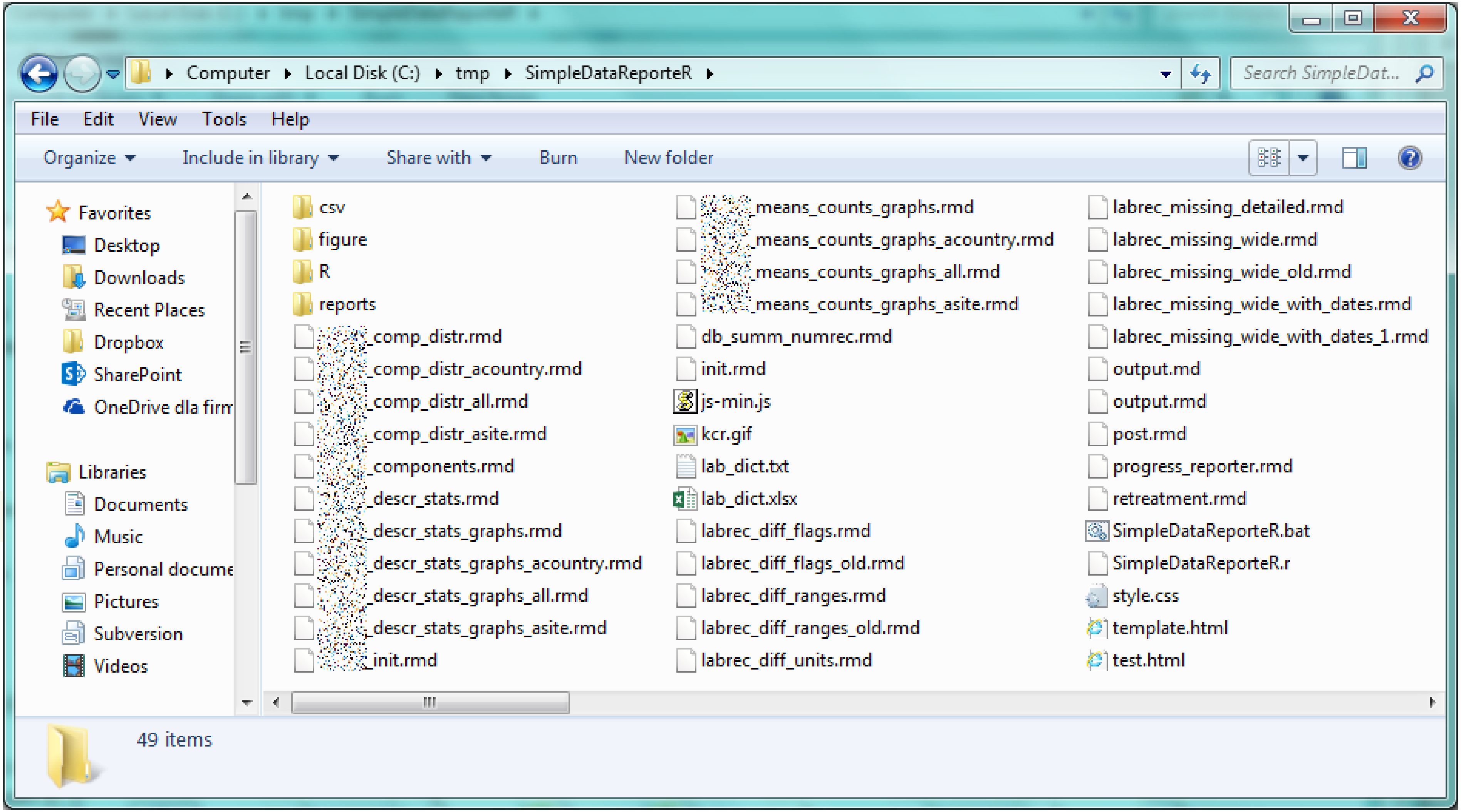
☑ Mean + median + count barchart — ***Parameters:*** ☑ acr. sites (N) ☑ acr. countr (N) ☑ don't split (N)

☐ Comparison of distributions from D1, W16, 20, W24 — ***Parameters:*** ☐ acr. sites (N) ☐ acr. countr (N) ☐ don't split (N)

☐ View on [redacted] and its components ([redacted]) across selected sites — ***Parameters:***
Period: Day 1|Week 12|Week 16|Week 20|Week 24
site 1: Site 2: site 3:
site 4: Site 5: site 6:

Created by Adrian Olszewski (2014), Department of Biometrics, KCR | Based on GNU R 3.0.3

♫ ...all you need is R ♫



- csv
- figure
- R
- reports
- [redacted]_comp_distr.rmd
- [redacted]_comp_distr_acountry.rmd
- [redacted]_comp_distr_all.rmd
- [redacted]_comp_distr_asite.rmd
- [redacted]_components.rmd
- [redacted]_descr_stats.rmd
- [redacted]_descr_stats_graphs.rmd
- [redacted]_descr_stats_graphs_acountry.rmd
- [redacted]_descr_stats_graphs_all.rmd
- [redacted]_descr_stats_graphs_asite.rmd
- [redacted]_init.rmd
- [redacted]_means_counts_graphs.rmd
- [redacted]_means_counts_graphs_acountry.rmd
- [redacted]_means_counts_graphs_all.rmd
- [redacted]_means_counts_graphs_asite.rmd
- db_summ_numrec.rmd
- init.rmd
- js-min.js
- kcr.gif
- lab_dict.txt
- lab_dict.xlsx
- labrec_diff_flags.rmd
- labrec_diff_flags_old.rmd
- labrec_diff_ranges.rmd
- labrec_diff_ranges_old.rmd
- labrec_diff_units.rmd
- labrec_missing_detailed.rmd
- labrec_missing_wide.rmd
- labrec_missing_wide_old.rmd
- labrec_missing_wide_with_dates.rmd
- labrec_missing_wide_with_dates_1.rmd
- output.md
- output.rmd
- post.rmd
- progress_reporter.rmd
- retreatment.rmd
- SimpleDataReporteR.bat
- SimpleDataReporteR.r
- style.css
- template.html
- test.html

Complete application: ~240MB raw,

~140MB after compression

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is (widely) supported by the world of science

IV R is supported by the community

V R is (increasingly) supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many possibilities to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

It's showtime!

R can produce output in a rich set of formats, such as:

- **pictures:** JPG, PNG, BMP, GIF, TIFF, SVG, EMF
- **animations:** GIF $^{\text{directly or via ImageMagic}}$, AVI $^{\text{via Ffmpeg}}$
- **plain text files**: CSV, TSV, XML, XPT (SAS transport files), ASCII
- **rich format docs**: RTF, PS, PDF, MS Word (*.doc; *.docx $^{\text{OpenXML}}$), Open/Libre Office Writer (*.odt $^{\text{OpenDocument}}$)
- **spreadsheets**: MS Excel (*.xls; *.xlsx), OpenOffice Calc (*.ods)
- **HTML pages**: simple pages, complex reports or presentations
  - HTML files can be easily read by Word and Excel. Tables, paragraphs, and styles are preserved
  - HTML is convertible to many other formats via pandoc
  - There is a way to obtain multi-sheet Excel workbooks
- R can also send the results to an external application via WebService $^{\text{JSON}}$

Reproducible Research

Definitions:

- *The goal of reproducible research is to tie specific instructions to data analysis and experimental data so that scholarship can be recreated, better understood and verified*.

  http://cran.r-project.org/web/views/ReproducibleResearch.html

- *The term reproducible research refers to the idea that the ultimate product of academic research is the paper along with the full computational environment used to produce the results in the paper such as the code, data, etc. that can be used to reproduce the results and create new work based on the research.*

  http://en.wikipedia.org/wiki/Reproducibility#Reproducible_research

Reproducible Research

In simply words, documents are created from templates containing presentation code (Markdown, HTML, LaTeX) or formatted paragraphs (MS Word, Open Document, RTF) mixed with chunks of R code.

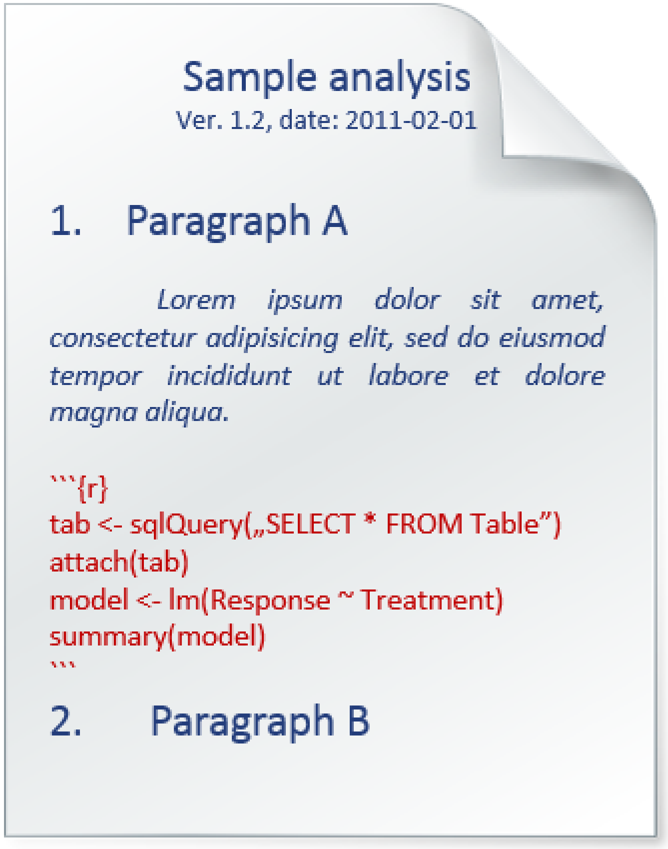
After the template is processed, chunks of R code are replaced with the result they produce.

With this approach one can forget about continuous copying and pasting objects (charts and tables) from the statistical package into a word processor.

Such documents are self-contained in that, they are capable to perform alone all the tasks required to complete rendering:

- setup the environment (install required packages, apply configuration),
- read data from the indicated sources or generate artificial datasets
- analyze the retrieved data and produce output objects (tables, graphics)

Reproducible Research



Sample analysis

Ver. 1.2, date: 2011-02-01

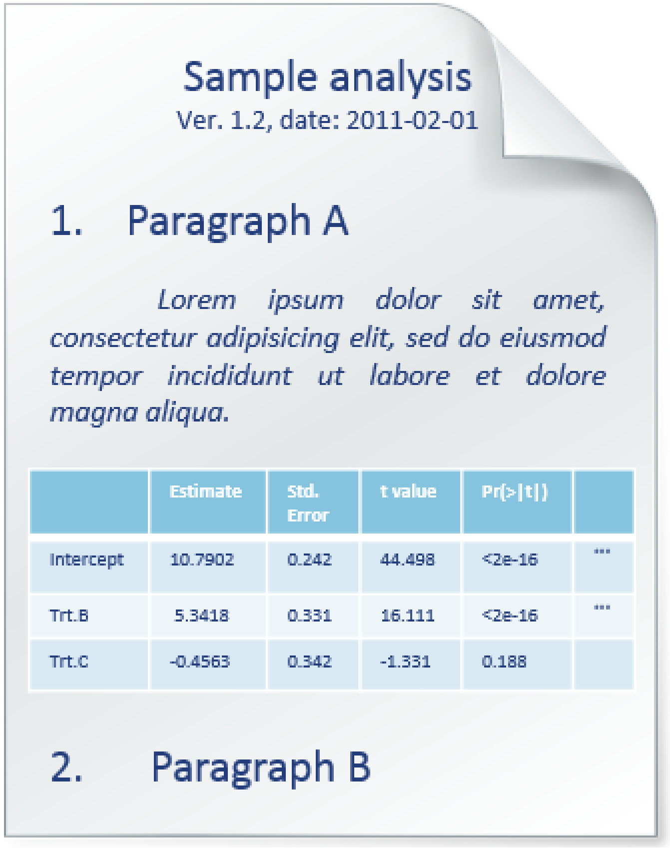
1. Paragraph A

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

~~~
```{r}
tab <- sqlQuery(„SELECT * FROM Table”)
attach(tab)
model <- lm(Response ~ Treatment)
summary(model)
```
~~~

2. Paragraph B

- Knitr + pandoc
- Sweave
- OdfWeave
- ReporteRs



Sample analysis

Ver. 1.2, date: 2011-02-01

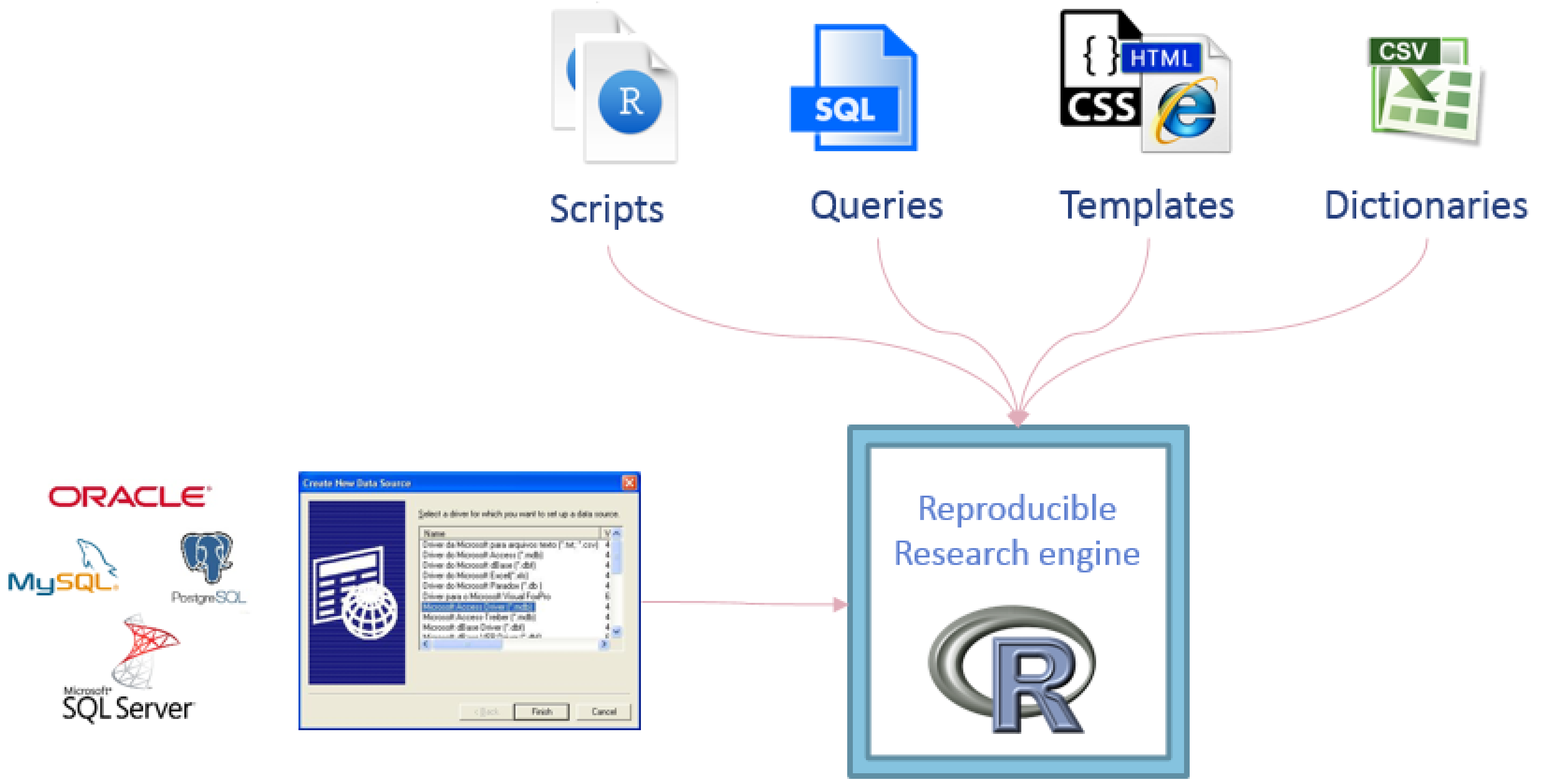
1. Paragraph A

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

	Estimate	Std. Error	t value	Pr(>|t|)	
Intercept	10.7902	0.242	44.498	<2e-16	***
Trt.B	5.3418	0.331	16.111	<2e-16	***
Trt.C	-0.4563	0.342	-1.331	0.188	

2. Paragraph B

Reproducible Research



Create New Data Source

Select a driver for which you want to set up a data source.

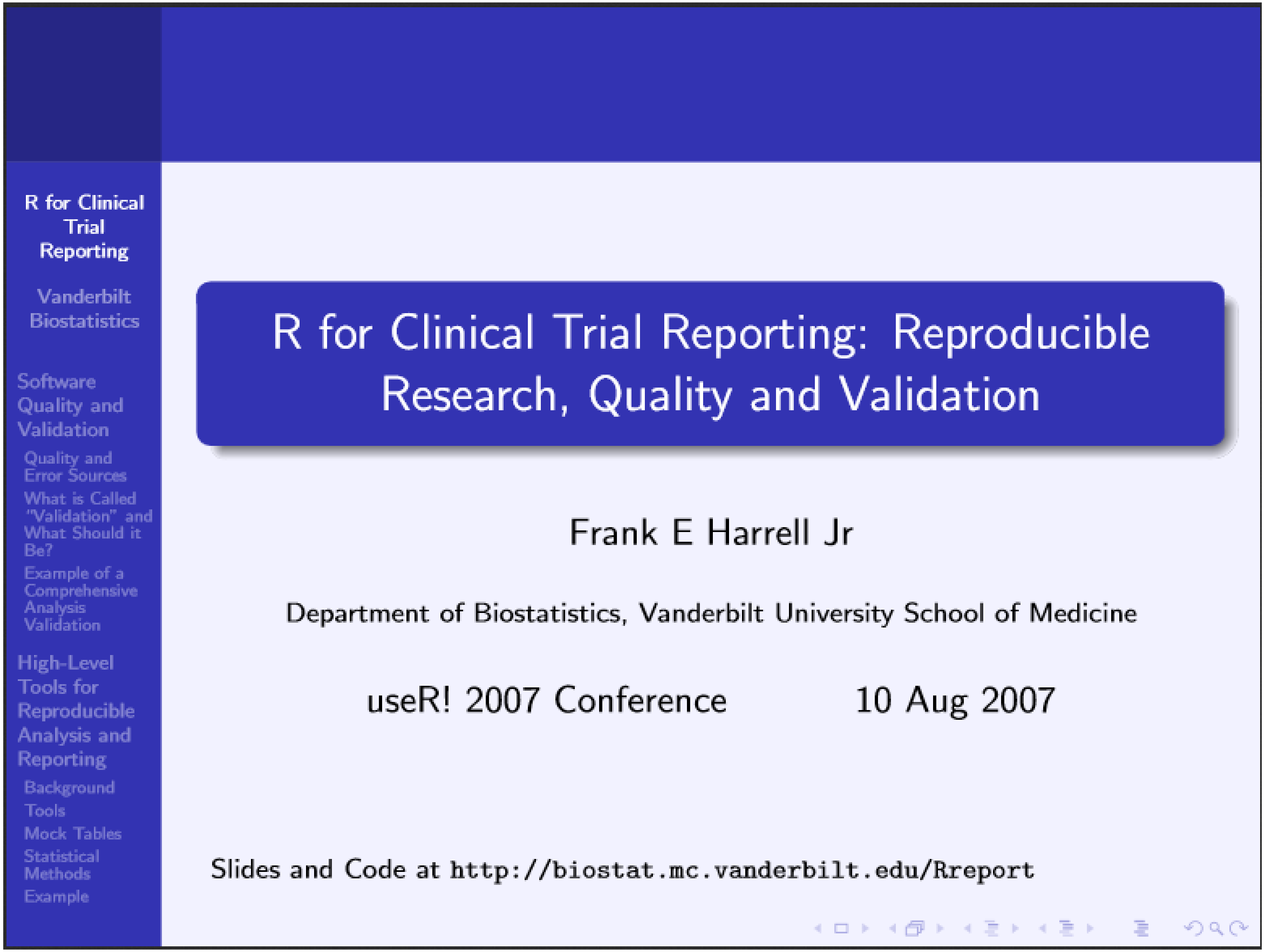
Name
Driver da Microsoft para arquivos texto (*.txt; *.csv)
Driver do Microsoft Access (*.mdb)
Driver do Microsoft dBase (*.dbf)
Driver do Microsoft Excel(*.xls)
Driver do Microsoft Paradox (*.db)
Driver para o Microsoft Visual FoxPro
Microsoft Access Driver (*.mdb)
Microsoft Access-Treiber (*.mdb)
Microsoft dBase Driver (*.dbf)

Reproducible Research engine

SiteID	SubjID	Lab Test	Screening	Day 1	Day 2
1	3	RBC	OK	OK	OK
1	3	WBC	OK	MISSING	N/A
1	4	ESR	OK	MISSING	MISSING
1	5	Hb	OK	OK	N/A
2	5	B-HCG	N/A	OK	MISSING

HTML

Reproducible Research



R for Clinical Trial Reporting

Vanderbilt Biostatistics

Software Quality and Validation

Quality and Error Sources

What is Called "Validation" and What Should it Be?

Example of a Comprehensive Analysis Validation

High-Level Tools for Reproducible Analysis and Reporting

Background

Tools

Mock Tables

Statistical Methods

Example

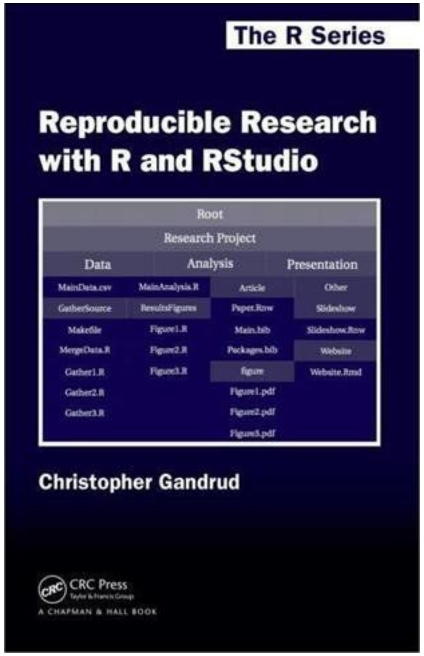
R for Clinical Trial Reporting: Reproducible Research, Quality and Validation

Frank E Harrell Jr

Department of Biostatistics, Vanderbilt University School of Medicine

useR! 2007 Conference 10 Aug 2007

Slides and Code at `http://biostat.mc.vanderbilt.edu/Rreport`



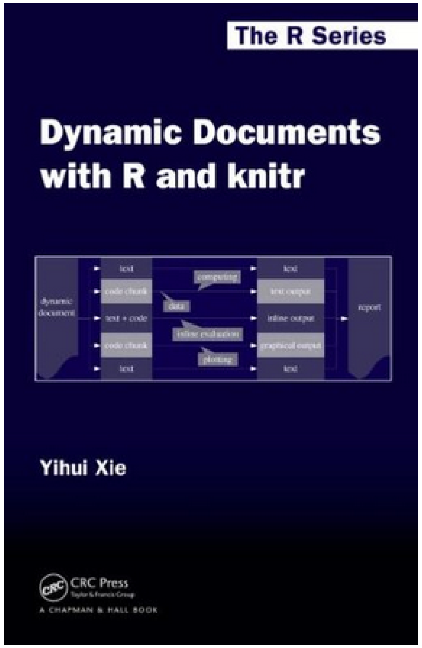
The R Series

Reproducible Research with R and RStudio

Christopher Gandrud

CRC Press

A CHAPMAN & HALL BOOK



The R Series

Dynamic Documents with R and knitr

Yihui Xie

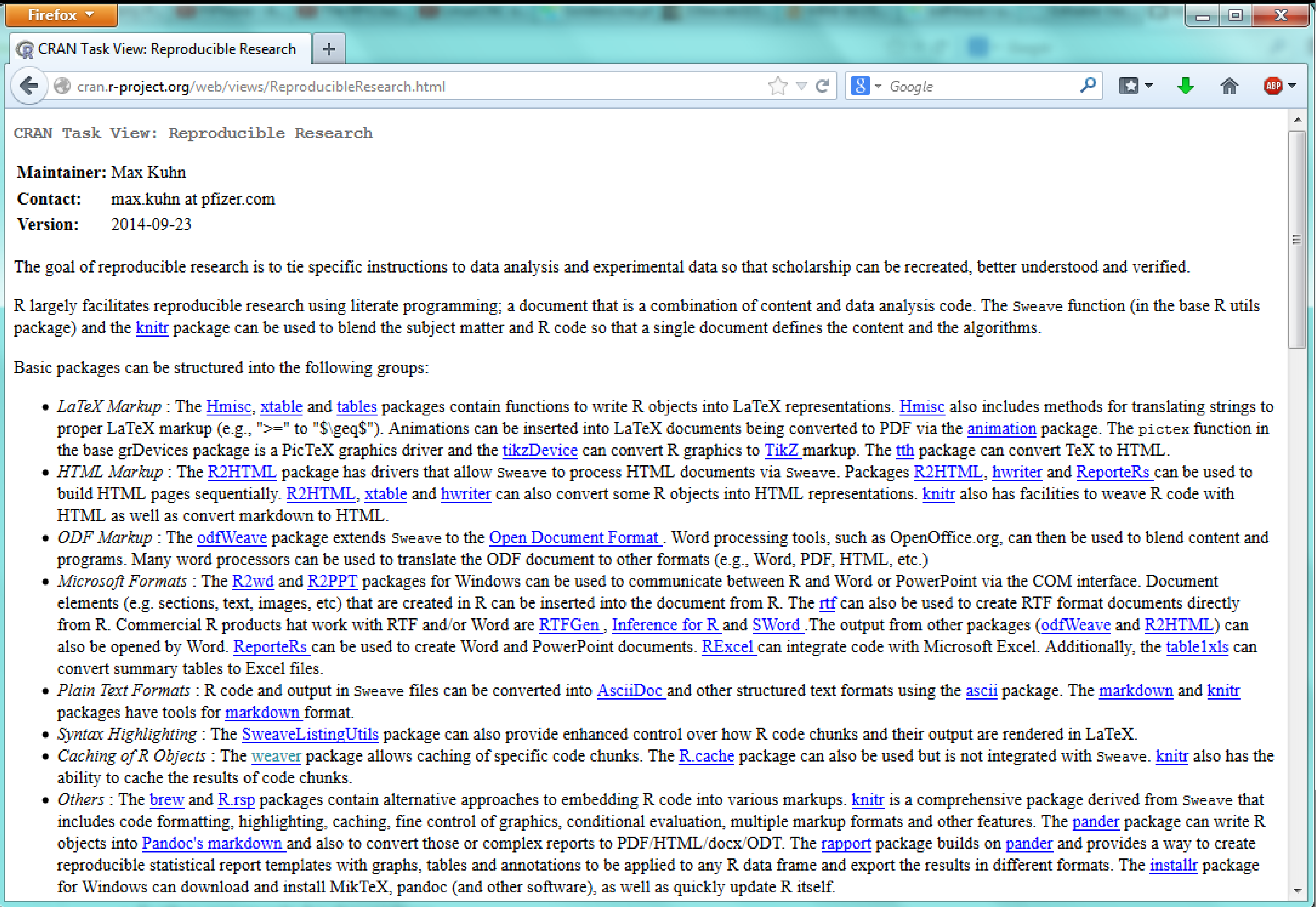
CRC Press

A CHAPMAN & HALL BOOK

http://biostat.mc.vanderbilt.edu/wiki/pub/Main/FHHandouts/dmcreport.pdf

http://handsondatascience.com/KnitRO.pdf

Reproducible Research



CRAN Task View: Reproducible Research

Maintainer: Max Kuhn
Contact: max.kuhn at pfizer.com
Version: 2014-09-23

The goal of reproducible research is to tie specific instructions to data analysis and experimental data so that scholarship can be recreated, better understood and verified.

R largely facilitates reproducible research using literate programming; a document that is a combination of content and data analysis code. The `Sweave` function (in the base R utils package) and the knitr package can be used to blend the subject matter and R code so that a single document defines the content and the algorithms.

Basic packages can be structured into the following groups:

- *LaTeX Markup* : The Hmisc, xtable and tables packages contain functions to write R objects into LaTeX representations. Hmisc also includes methods for translating strings to proper LaTeX markup (e.g., ">=" to "$\geq$"). Animations can be inserted into LaTeX documents being converted to PDF via the animation package. The `pictex` function in the base grDevices package is a PicTeX graphics driver and the tikzDevice can convert R graphics to TikZ markup. The tth package can convert TeX to HTML.
- *HTML Markup* : The R2HTML package has drivers that allow `Sweave` to process HTML documents via `Sweave`. Packages R2HTML, hwriter and ReporteRs can be used to build HTML pages sequentially. R2HTML, xtable and hwriter can also convert some R objects into HTML representations. knitr also has facilities to weave R code with HTML as well as convert markdown to HTML.
- *ODF Markup* : The odfWeave package extends `Sweave` to the Open Document Format . Word processing tools, such as OpenOffice.org, can then be used to blend content and programs. Many word processors can be used to translate the ODF document to other formats (e.g., Word, PDF, HTML, etc.)
- *Microsoft Formats* : The R2wd and R2PPT packages for Windows can be used to communicate between R and Word or PowerPoint via the COM interface. Document elements (e.g. sections, text, images, etc) that are created in R can be inserted into the document from R. The rtf can also be used to create RTF format documents directly from R. Commercial R products hat work with RTF and/or Word are RTFGen , Inference for R and SWord .The output from other packages (odfWeave and R2HTML) can also be opened by Word. ReporteRs can be used to create Word and PowerPoint documents. RExcel can integrate code with Microsoft Excel. Additionally, the table1xls can convert summary tables to Excel files.
- *Plain Text Formats* : R code and output in `Sweave` files can be converted into AsciiDoc and other structured text formats using the ascii package. The markdown and knitr packages have tools for markdown format.
- *Syntax Highlighting* : The SweaveListingUtils package can also provide enhanced control over how R code chunks and their output are rendered in LaTeX.
- *Caching of R Objects* : The weaver package allows caching of specific code chunks. The R.cache package can also be used but is not integrated with `Sweave`. knitr also has the ability to cache the results of code chunks.
- *Others* : The brew and R.rsp packages contain alternative approaches to embedding R code into various markups. knitr is a comprehensive package derived from `Sweave` that includes code formatting, highlighting, caching, fine control of graphics, conditional evaluation, multiple markup formats and other features. The pander package can write R objects into Pandoc's markdown and also to convert those or complex reports to PDF/HTML/docx/ODT. The rapport package builds on pander and provides a way to create reproducible statistical report templates with graphs, tables and annotations to be applied to any R data frame and export the results in different formats. The installr package for Windows can download and install MikTeX, pandoc (and other software), as well as quickly update R itself.

Reproducible Research – knitr and Sweave

With HTML, Latex, R Markdown, xtable and CSS processed by Sweave or knitr and (optionaly) pandoc, one can produce complex, professionally looking and reproducible reports in many formats, e.g. DOCX, ODT, HTML, RTF and PDF.

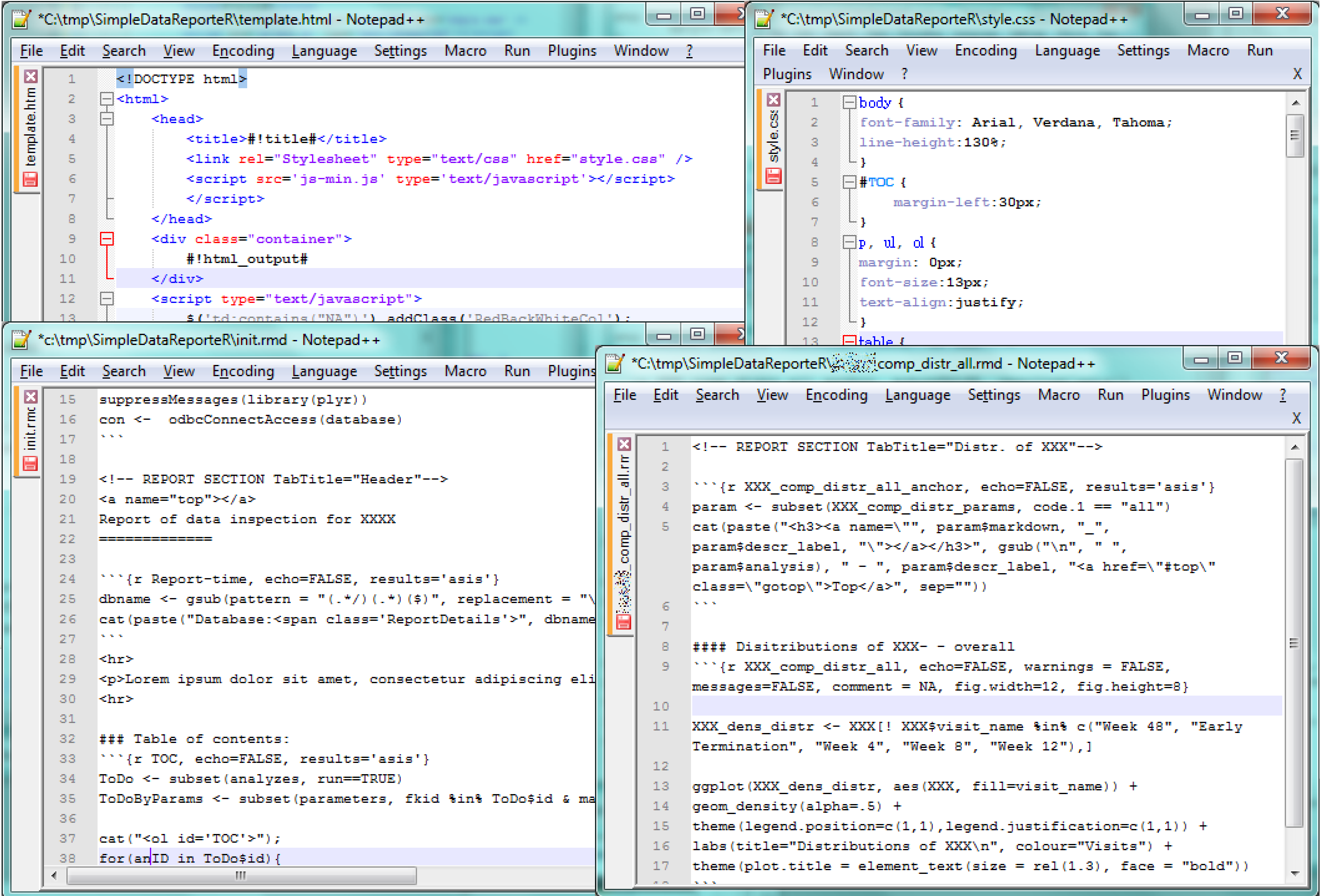
This R-scripts-driven approach might seem a bit complicated, but it gives the user maximum flexibility and control over how the final product will look like. HTML and CSS WYSIWYG editors are useful here.

Chunks of R code may be saved into named sub-reports for later reuse in many places in the template. They can be turned on/off depending on the result of a conditional expression.

Produced HTML file can be displayed in a web browser, opened directly in a word processor or spreadsheet or converted to another format (DOCX, PDF).

This is a perfect tool for automated generation of reports or record the flow of analyses (= R code along with results).

knitr + R Markdown + HTML + CSS



```
<!DOCTYPE html>
<html>
    <head>
        <title>#!title#</title>
        <link rel="Stylesheet" type="text/css" href="style.css" />
        <script src='js-min.js' type='text/javascript'></script>
        </script>
    </head>
    <div class="container">
        #!html_output#
    </div>
    <script type="text/javascript">
        $('td:contains("NA")').addClass('RedBackWhiteCol');
```

```
body {
font-family: Arial, Verdana, Tahoma;
line-height:130%;
}
#TOC {
    margin-left:30px;
}
p, ul, ol {
margin: 0px;
font-size:13px;
text-align:justify;
}
table {
```

```
suppressMessages(library(plyr))
con <-  odbcConnectAccess(database)
```

<!-- REPORT SECTION TabTitle="Header"-->

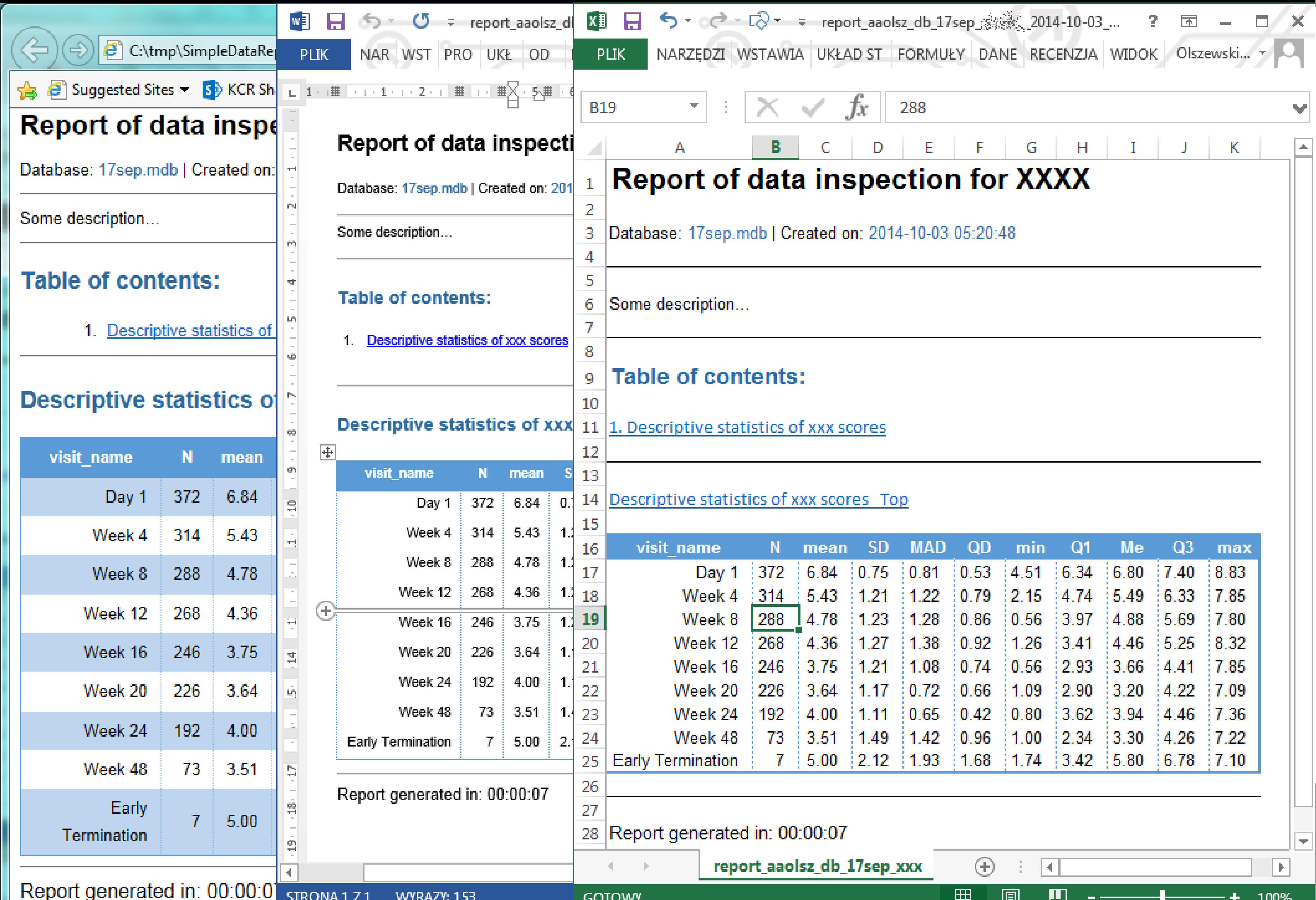
Report of data inspection for XXXX
=============

```{r Report-time, echo=FALSE, results='asis'}
dbname <- gsub(pattern = "(.*/)(.*)($)", replacement = "\
cat(paste("Database:<span class='ReportDetails'>", dbname
```
<hr>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing eli
<hr>

Table of contents:
```{r TOC, echo=FALSE, results='asis'}
ToDo <- subset(analyzes, run==TRUE)
ToDoByParams <- subset(parameters, fkid %in% ToDo$id & ma

cat("<ol id='TOC'>");
for(anID in ToDo$id){
```

~~~
<!-- REPORT SECTION TabTitle="Distr. of XXX"-->

```{r XXX_comp_distr_all_anchor, echo=FALSE, results='asis'}
param <- subset(XXX_comp_distr_params, code.1 == "all")
cat(paste("<h3><a name=\"", param$markdown, "_",
param$descr_label, "\"></a></h3>", gsub("\n", " ",
param$analysis), " - ", param$descr_label, "<a href=\"#top\"
class=\"gotop\">Top</a>", sep=""))
```


#### Disitributions of XXX- - overall
```{r XXX_comp_distr_all, echo=FALSE, warnings = FALSE,
messages=FALSE, comment = NA, fig.width=12, fig.height=8}

XXX_dens_distr <- XXX[! XXX$visit_name %in% c("Week 48", "Early
Termination", "Week 4", "Week 8", "Week 12"),]

ggplot(XXX_dens_distr, aes(XXX, fill=visit_name)) +
geom_density(alpha=.5) +
theme(legend.position=c(1,1),legend.justification=c(1,1)) +
labs(title="Distributions of XXX\n", colour="Visits") +
theme(plot.title = element_text(size = rel(1.3), face = "bold"))
```
~~~

HTML report opened in Word and Excel



Report of data inspection for XXXX

Database: 17sep.mdb | Created on: 2014-10-03 05:20:48

Some description…

Table of contents:

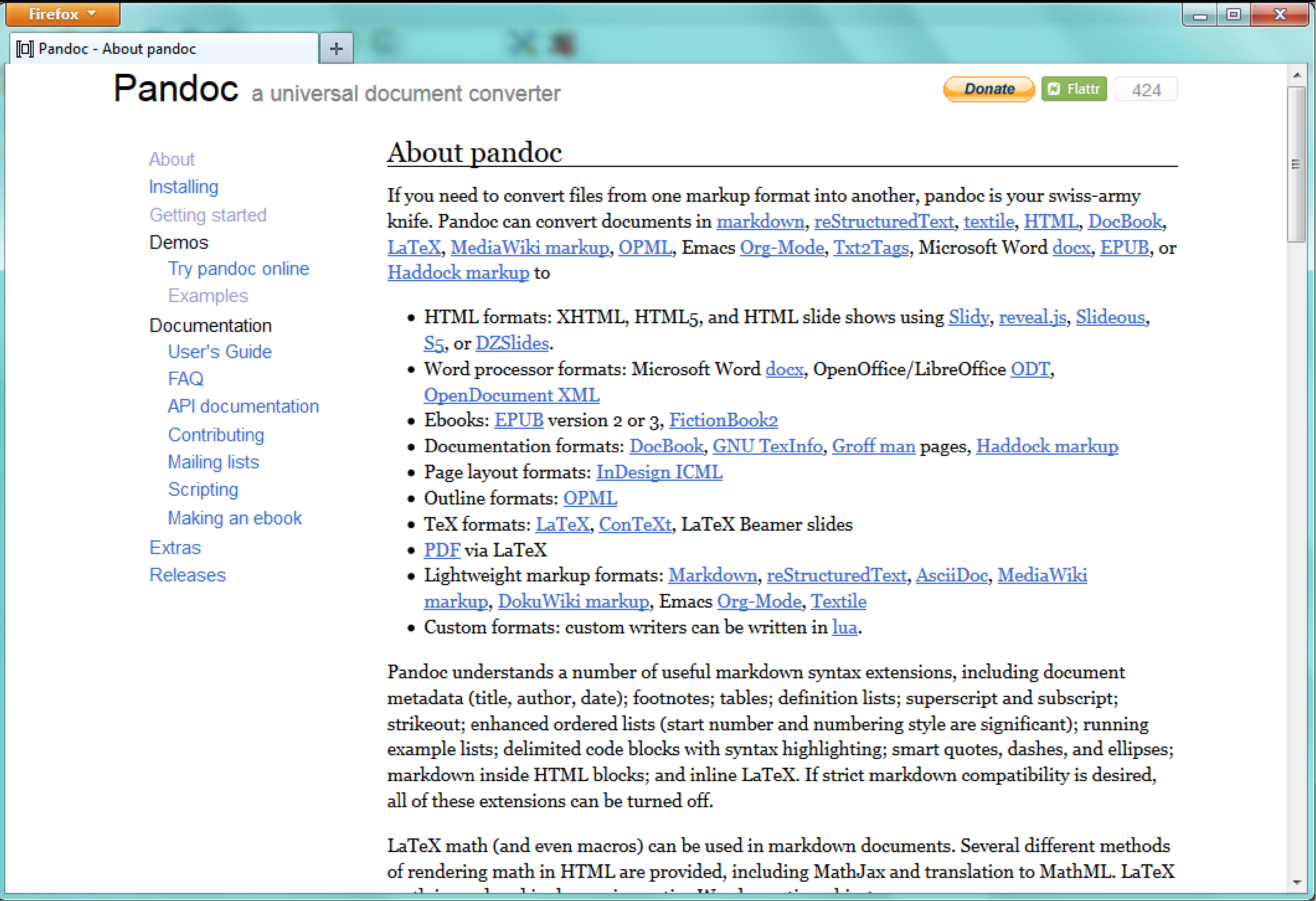
1. Descriptive statistics of xxx scores

Descriptive statistics of xxx scores Top

visit_name	N	mean	SD	MAD	QD	min	Q1	Me	Q3	max
Day 1	372	6.84	0.75	0.81	0.53	4.51	6.34	6.80	7.40	8.83
Week 4	314	5.43	1.21	1.22	0.79	2.15	4.74	5.49	6.33	7.85
Week 8	288	4.78	1.23	1.28	0.86	0.56	3.97	4.88	5.69	7.80
Week 12	268	4.36	1.27	1.38	0.92	1.26	3.41	4.46	5.25	8.32
Week 16	246	3.75	1.21	1.08	0.74	0.56	2.93	3.66	4.41	7.85
Week 20	226	3.64	1.17	0.72	0.66	1.09	2.90	3.20	4.22	7.09
Week 24	192	4.00	1.11	0.65	0.42	0.80	3.62	3.94	4.46	7.36
Week 48	73	3.51	1.49	1.42	0.96	1.00	2.34	3.30	4.26	7.22
Early Termination	7	5.00	2.12	1.93	1.68	1.74	3.42	5.80	6.78	7.10

Report generated in: 00:00:07

pandoc – convert to real DOC / OXML / PDF / ODF



Pandoc a universal document converter

About pandoc

If you need to convert files from one markup format into another, pandoc is your swiss-army knife. Pandoc can convert documents in markdown, reStructuredText, textile, HTML, DocBook, LaTeX, MediaWiki markup, OPML, Emacs Org-Mode, Txt2Tags, Microsoft Word docx, EPUB, or Haddock markup to

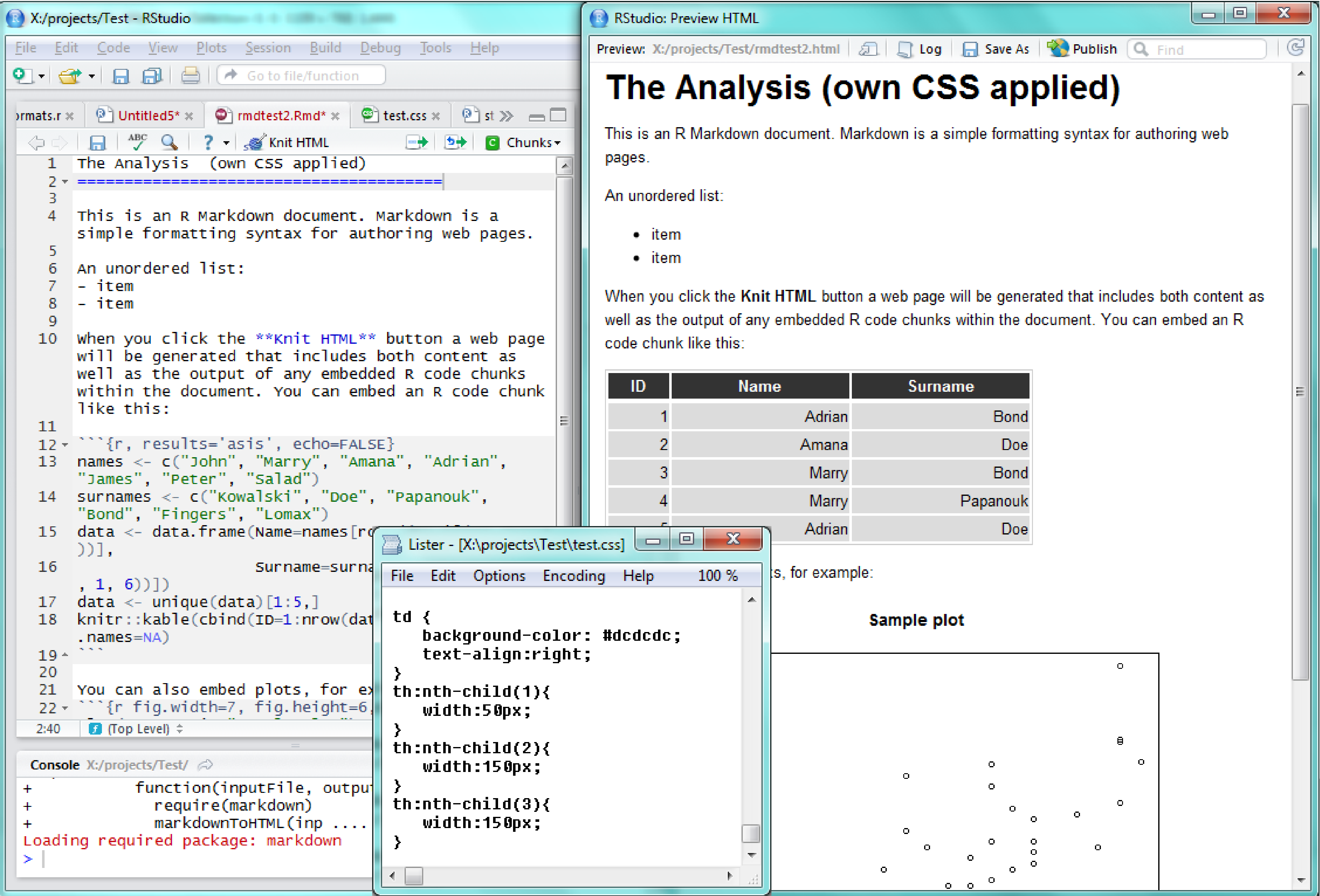
- HTML formats: XHTML, HTML5, and HTML slide shows using Slidy, reveal.js, Slideous, S5, or DZSlides.
- Word processor formats: Microsoft Word docx, OpenOffice/LibreOffice ODT, OpenDocument XML
- Ebooks: EPUB version 2 or 3, FictionBook2
- Documentation formats: DocBook, GNU TexInfo, Groff man pages, Haddock markup
- Page layout formats: InDesign ICML
- Outline formats: OPML
- TeX formats: LaTeX, ConTeXt, LaTeX Beamer slides
- PDF via LaTeX
- Lightweight markup formats: Markdown, reStructuredText, AsciiDoc, MediaWiki markup, DokuWiki markup, Emacs Org-Mode, Textile
- Custom formats: custom writers can be written in lua.

Pandoc understands a number of useful markdown syntax extensions, including document metadata (title, author, date); footnotes; tables; definition lists; superscript and subscript; strikeout; enhanced ordered lists (start number and numbering style are significant); running example lists; delimited code blocks with syntax highlighting; smart quotes, dashes, and ellipses; markdown inside HTML blocks; and inline LaTeX. If strict markdown compatibility is desired, all of these extensions can be turned off.

LaTeX math (and even macros) can be used in markdown documents. Several different methods of rendering math in HTML are provided, including MathJax and translation to MathML. LaTeX

Reproducible Research with RMarkdown

RStudio



~~~
The Analysis  (own CSS applied)
========================================

This is an R Markdown document. Markdown is a
simple formatting syntax for authoring web pages.

An unordered list:
- item
- item

When you click the **Knit HTML** button a web page
will be generated that includes both content as
well as the output of any embedded R code chunks
within the document. You can embed an R code chunk
like this:

```{r, results='asis', echo=FALSE}
names <- c("John", "Marry", "Amana", "Adrian",
"James", "Peter", "Salad")
surnames <- c("Kowalski", "Doe", "Papanouk",
"Bond", "Fingers", "Lomax")
data <- data.frame(Name=names[r
))],
                   Surname=surn
, 1, 6))])
data <- unique(data)[1:5,]
knitr::kable(cbind(ID=1:nrow(dat
.names=NA)
~~~

You can also embed plots, for ex
```{r fig.width=7, fig.height=6,
```

```
+        function(inputFile, outpu
+          require(markdown)
+          markdownToHTML(inp ....
Loading required package: markdown
>
```

The Analysis (own CSS applied)

This is an R Markdown document. Markdown is a simple formatting syntax for authoring web pages.

An unordered list:

- item
- item

When you click the **Knit HTML** button a web page will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

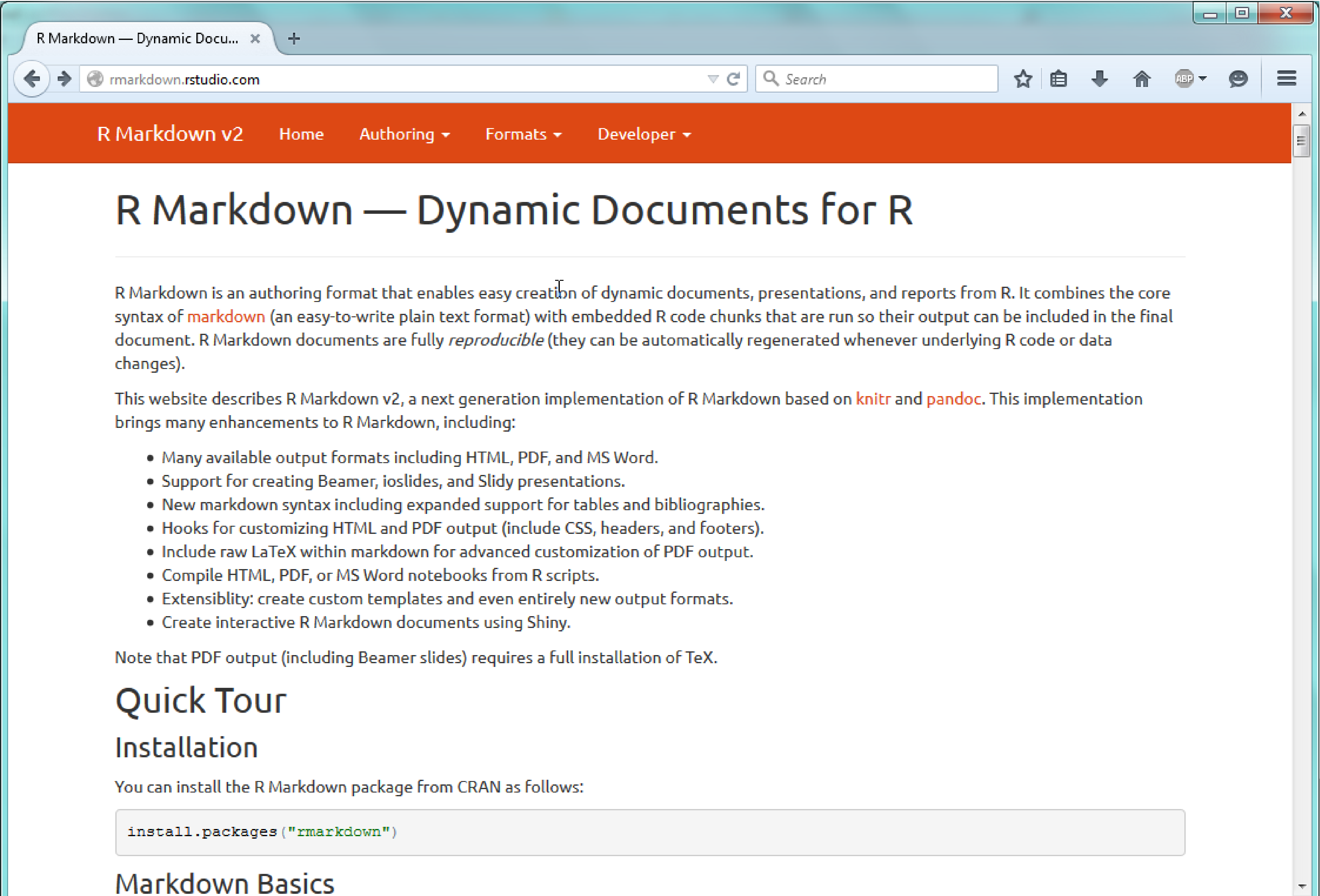
ID	Name	Surname
1	Adrian	Bond
2	Amana	Doe
3	Marry	Bond
4	Marry	Papanouk
	Adrian	Doe

s, for example:

Sample plot

```
td {
   background-color: #dcdcdc;
   text-align:right;
}
th:nth-child(1){
   width:50px;
}
th:nth-child(2){
   width:150px;
}
th:nth-child(3){
   width:150px;
}
```

Take a moment and discover R Markdown!



R Markdown — Dynamic Documents for R

R Markdown is an authoring format that enables easy creation of dynamic documents, presentations, and reports from R. It combines the core syntax of markdown (an easy-to-write plain text format) with embedded R code chunks that are run so their output can be included in the final document. R Markdown documents are fully *reproducible* (they can be automatically regenerated whenever underlying R code or data changes).

This website describes R Markdown v2, a next generation implementation of R Markdown based on knitr and pandoc. This implementation brings many enhancements to R Markdown, including:

- Many available output formats including HTML, PDF, and MS Word.
- Support for creating Beamer, ioslides, and Slidy presentations.
- New markdown syntax including expanded support for tables and bibliographies.
- Hooks for customizing HTML and PDF output (include CSS, headers, and footers).
- Include raw LaTeX within markdown for advanced customization of PDF output.
- Compile HTML, PDF, or MS Word notebooks from R scripts.
- Extensiblity: create custom templates and even entirely new output formats.
- Create interactive R Markdown documents using Shiny.

Note that PDF output (including Beamer slides) requires a full installation of TeX.

Quick Tour

Installation

You can install the R Markdown package from CRAN as follows:

```
install.packages("rmarkdown")
```

Markdown Basics

Reproducible Research with ODF – odfWeave

OdfWeave is an adaptation of Sweave for OpenDocument files. Such files serve as a template of the report with R code embedded in. MS Office, Open/Libre Office and Calligra Suite (formerly KOffice) work with the OpenDocument format smoothly.

OdfWeave is flexible and simple to use. It allows to create templates directly in a word processor. This is the most convenient way to accomplish the Reproducible Research goals. Styling can be applied (in a limited extent) from the R code and, mostly – in the editor.

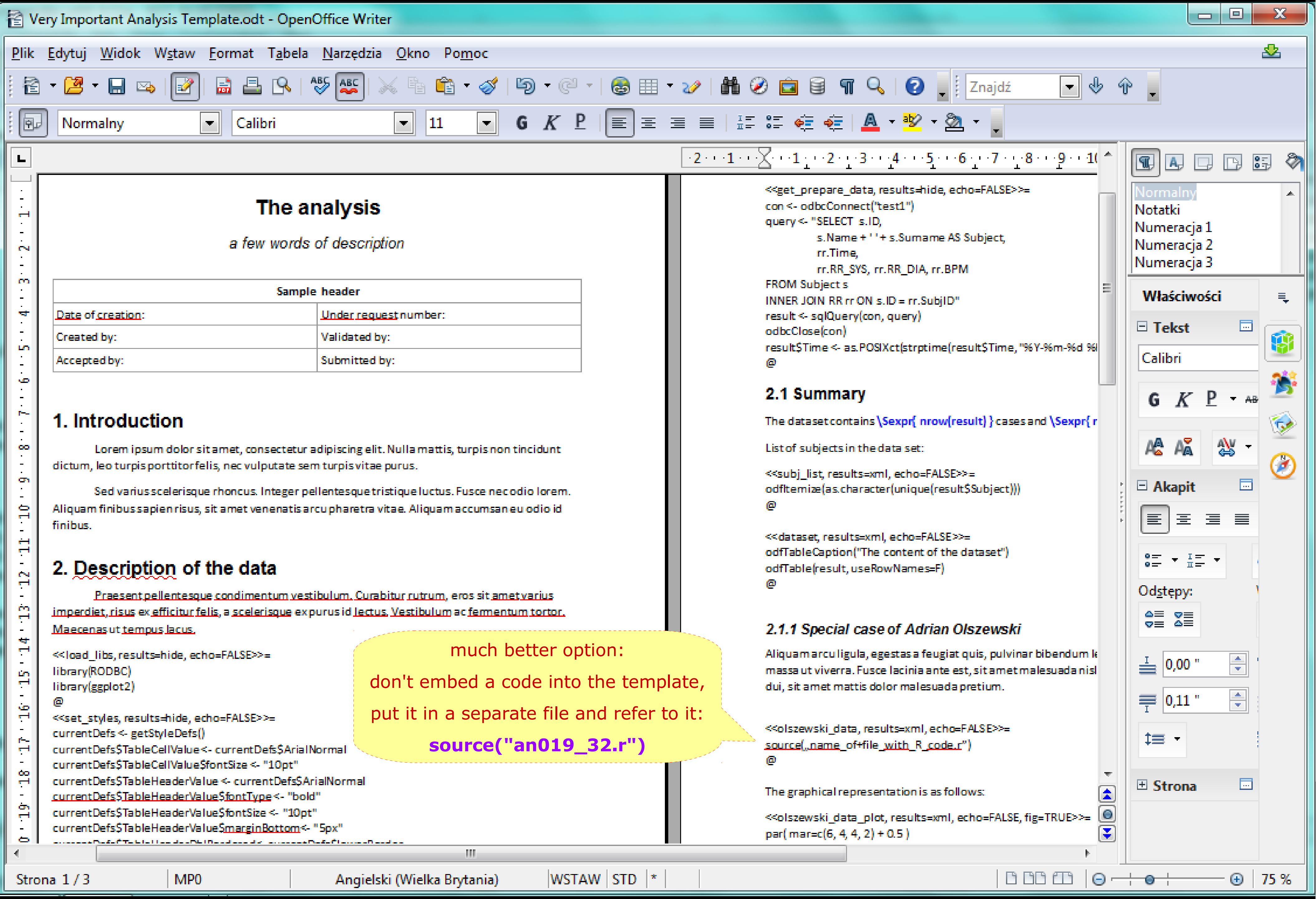
The R code may be embedded into a template document or stored in a separate file, which is then referrenced in the template. Both approaches have their pros and cons. The second approach is particularly useful, when the same code is shared between (reused in) many documents.

Reproducible Research with ODF – odfWeave

Unfortunately, some serious caveats and limitations must be noted:

- MS Word incorrectly saves a definition of autonumbered captions which causes all captions begin from: “Table: 999”
- MS Windows does not honor UTF-8 locale. Therefore odfWeave cannot handle mixed characters from different alphabets (óęłéüš яфы βθΣΩ), especially ASCII (English) and non-ASCII (Russian, Greek). There are four options to solve this issue:
    - process document under Linux or MacOS, which are Unicode-enabled
    - use old good non-Unicode fonts, one for each language, e.g. Symbol for Greek characters and math symbols.
    - replace all non-ASCII characters with their ASCII equivalents in the fetched data by using **iconv(text, "UTF-8", "ASCII//TRANSLIT")**
        - Ä, Á, Ą --> A | Ë, É, Ę --> E | Õ, Ó --> O (U)
        - Ň, Ń --> N | Ü, Ó --> U | Š, Ş, Ś --> S etc.

From an ODF template...



The analysis

a few words of description

Sample header	
Date of creation:	Under request number:
Created by:	Validated by:
Accepted by:	Submitted by:

1. Introduction

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla mattis, turpis non tincidunt dictum, leo turpis porttitor felis, nec vulputate sem turpis vitae purus.

Sed varius scelerisque rhoncus. Integer pellentesque tristique luctus. Fusce nec odio lorem. Aliquam finibus sapien risus, sit amet venenatis arcu pharetra vitae. Aliquam accumsan eu odio id finibus.

2. Description of the data

Praesent pellentesque condimentum vestibulum. Curabitur rutrum, eros sit amet varius imperdiet, risus ex efficitur felis, a scelerisque ex purus id lectus. Vestibulum ac fermentum tortor. Maecenas ut tempus lacus.

```
<<load_libs, results=hide, echo=FALSE>>=
library(RODBC)
library(ggplot2)
@
<<set_styles, results=hide, echo=FALSE>>=
currentDefs <- getStyleDefs()
currentDefs$TableCellValue <- currentDefs$ArialNormal
currentDefs$TableCellValue$fontSize <- "10pt"
currentDefs$TableHeaderValue <- currentDefs$ArialNormal
currentDefs$TableHeaderValue$fontType <- "bold"
currentDefs$TableHeaderValue$fontSize <- "10pt"
currentDefs$TableHeaderValue$marginBottom <- "5px"
```

```
<<get_prepare_data, results=hide, echo=FALSE>>=
con <- odbcConnect("test1")
query <- "SELECT  s.ID,
            s.Name + ' ' + s.Surname AS Subject,
            rr.Time,
            rr.RR_SYS, rr.RR_DIA, rr.BPM
FROM Subject s
INNER JOIN RR rr ON s.ID = rr.SubjID"
result <- sqlQuery(con, query)
odbcClose(con)
result$Time <- as.POSIXct(strptime(result$Time, "%Y-%m-%d %
@
```

2.1 Summary

The dataset contains \Sexpr{ nrow(result) } cases and \Sexpr{ r

List of subjects in the data set:

```
<<subj_list, results=xml, echo=FALSE>>=
odfItemize(as.character(unique(result$Subject)))
@

<<dataset, results=xml, echo=FALSE>>=
odfTableCaption("The content of the dataset")
odfTable(result, useRowNames=F)
@
```

2.1.1 Special case of Adrian Olszewski

Aliquam arcu ligula, egestas a feugiat quis, pulvinar bibendum l
massa ut viverra. Fusce lacinia ante est, sit amet malesuada nisl
dui, sit amet mattis dolor malesuada pretium.

```
<<olszewski_data, results=xml, echo=FALSE>>=
source(„name_of+file_with_R_code.r")
@
```

The graphical representation is as follows:

```
<<olszewski_data_plot, results=xml, echo=FALSE, fig=TRUE>>=
par( mar=c(6, 4, 4, 2) + 0.5 )
```

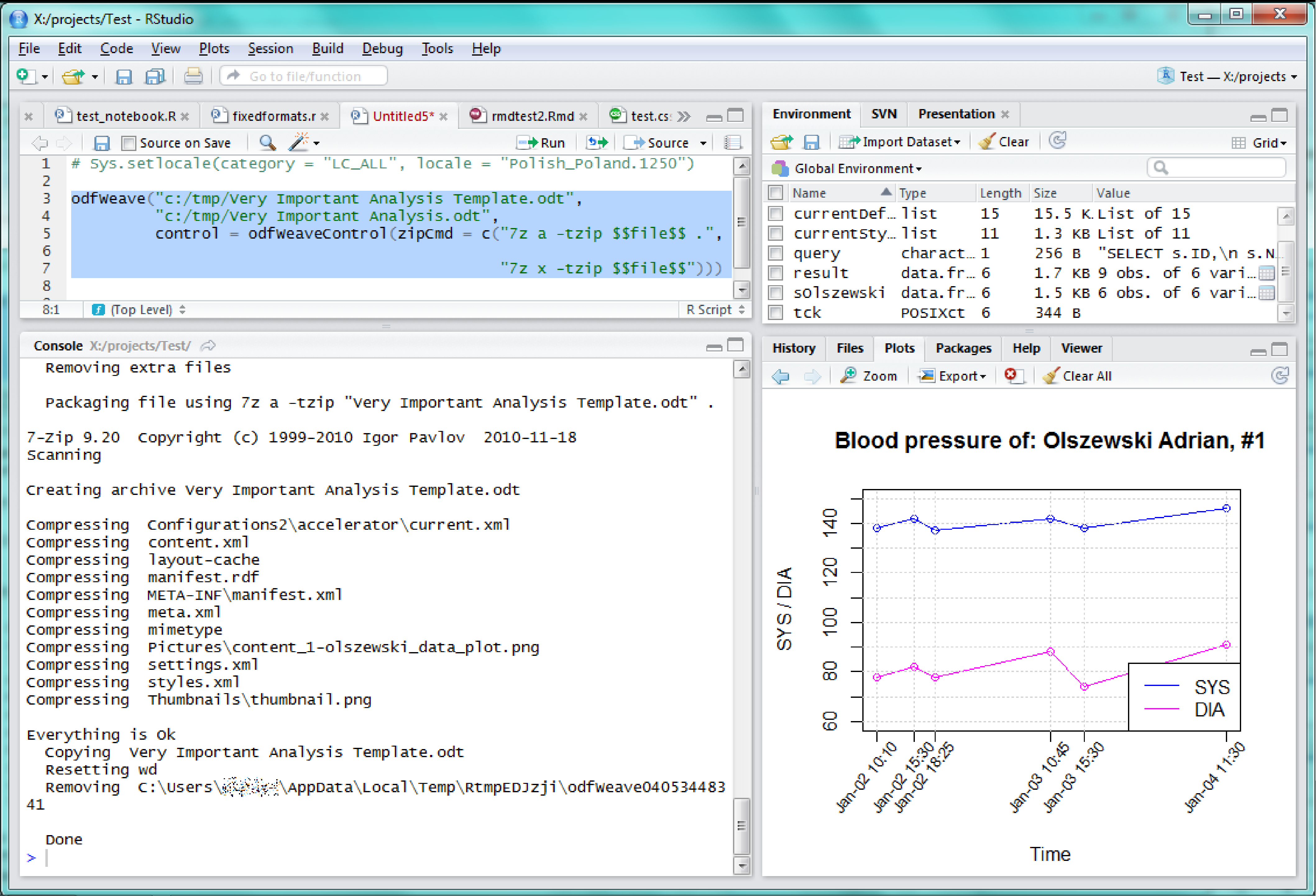
much better option:

don't embed a code into the template,

put it in a separate file and refer to it:

source("an019_32.r")

…via the odfWeave engine…

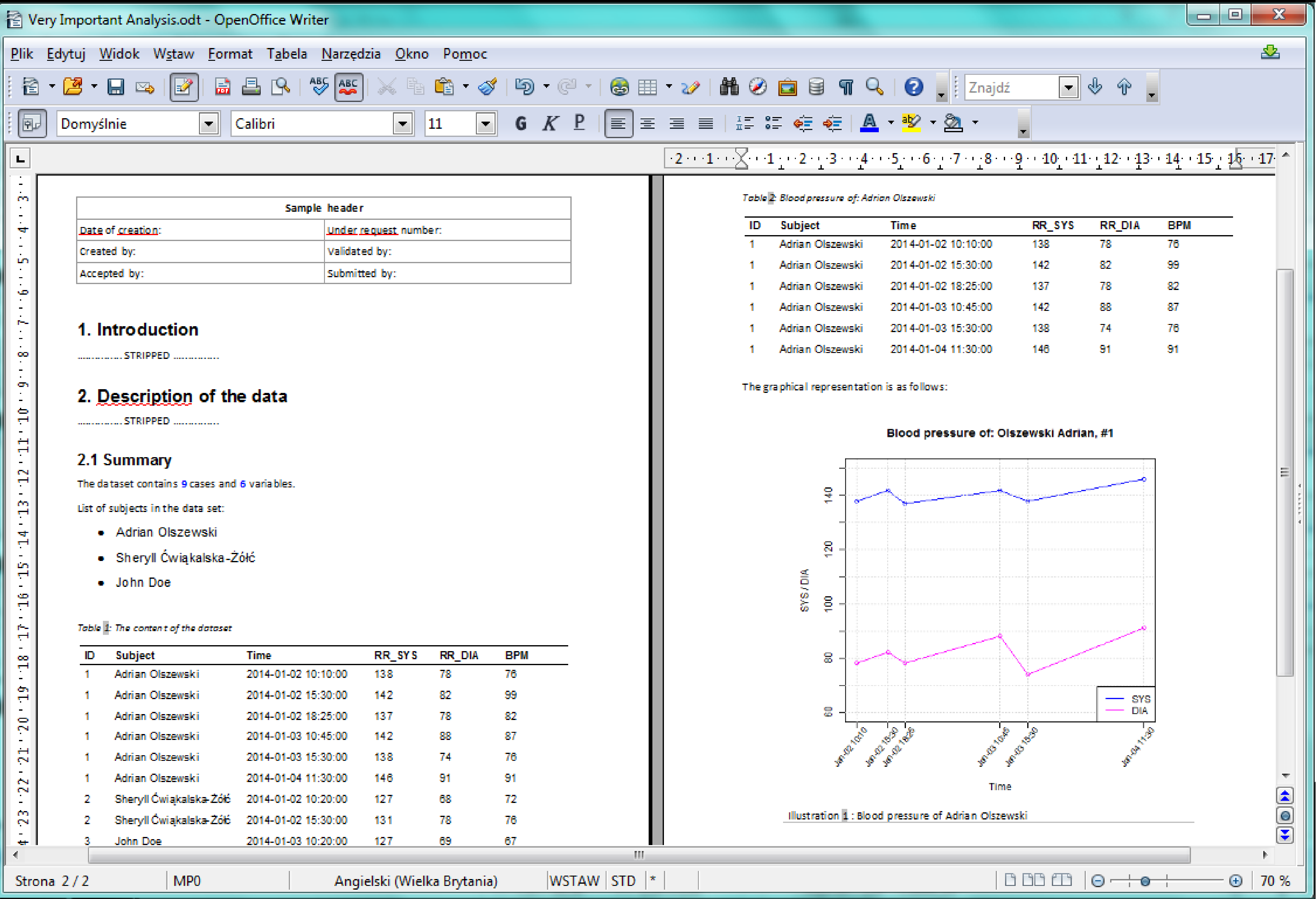


```r
# Sys.setlocale(category = "LC_ALL", locale = "Polish_Poland.1250")

odfWeave("c:/tmp/Very Important Analysis Template.odt",
         "c:/tmp/Very Important Analysis.odt",
         control = odfWeaveControl(zipCmd = c("7z a -tzip $$file$$ .",
                                             "7z x -tzip $$file$$")))
```

☐	Name	Type	Length	Size	Value
☐	currentDef…	list	15	15.5 K.	List of 15
☐	currentSty…	list	11	1.3 KB	List of 11
☐	query	charact…	1	256 B	"SELECT s.ID,\n s.N…
☐	result	data.fr…	6	1.7 KB	9 obs. of 6 vari…
☐	solszewski	data.fr…	6	1.5 KB	6 obs. of 6 vari…
☐	tck	POSIXct	6	344 B	

```
  Removing extra files

  Packaging file using 7z a -tzip "Very Important Analysis Template.odt" .

7-Zip 9.20  Copyright (c) 1999-2010 Igor Pavlov  2010-11-18
Scanning

Creating archive Very Important Analysis Template.odt

Compressing  Configurations2\accelerator\current.xml
Compressing  content.xml
Compressing  layout-cache
Compressing  manifest.rdf
Compressing  META-INF\manifest.xml
Compressing  meta.xml
Compressing  mimetype
Compressing  Pictures\content_1-olszewski_data_plot.png
Compressing  settings.xml
Compressing  styles.xml
Compressing  Thumbnails\thumbnail.png

Everything is Ok
  Copying  Very Important Analysis Template.odt
  Resetting wd
  Removing  C:\Users\[redacted]\AppData\Local\Temp\RtmpEDJzji\odfWeave040534483
41

  Done
> 
```

...to the final document



Sample header	
Date of creation:	Under request number:
Created by:	Validated by:
Accepted by:	Submitted by:

1. Introduction

…………. STRIPPED ………….

2. Description of the data

…………. STRIPPED ………….

2.1 Summary

The dataset contains 9 cases and 6 variables.

List of subjects in the data set:

- Adrian Olszewski
- Sheryll Ćwiąkalska-Żółć
- John Doe

Table 1: The content of the dataset

ID	Subject	Time	RR_SYS	RR_DIA	BPM
1	Adrian Olszewski	2014-01-02 10:10:00	138	78	76
1	Adrian Olszewski	2014-01-02 15:30:00	142	82	99
1	Adrian Olszewski	2014-01-02 18:25:00	137	78	82
1	Adrian Olszewski	2014-01-03 10:45:00	142	88	87
1	Adrian Olszewski	2014-01-03 15:30:00	138	74	76
1	Adrian Olszewski	2014-01-04 11:30:00	146	91	91
2	Sheryll Ćwiąkalska-Żółć	2014-01-02 10:20:00	127	68	72
2	Sheryll Ćwiąkalska-Żółć	2014-01-02 15:30:00	131	78	76
3	John Doe	2014-01-03 10:20:00	127	69	67

Table 2: Blood pressure of: Adrian Olszewski

ID	Subject	Time	RR_SYS	RR_DIA	BPM
1	Adrian Olszewski	2014-01-02 10:10:00	138	78	76
1	Adrian Olszewski	2014-01-02 15:30:00	142	82	99
1	Adrian Olszewski	2014-01-02 18:25:00	137	78	82
1	Adrian Olszewski	2014-01-03 10:45:00	142	88	87
1	Adrian Olszewski	2014-01-03 15:30:00	138	74	76
1	Adrian Olszewski	2014-01-04 11:30:00	146	91	91

The graphical representation is as follows:

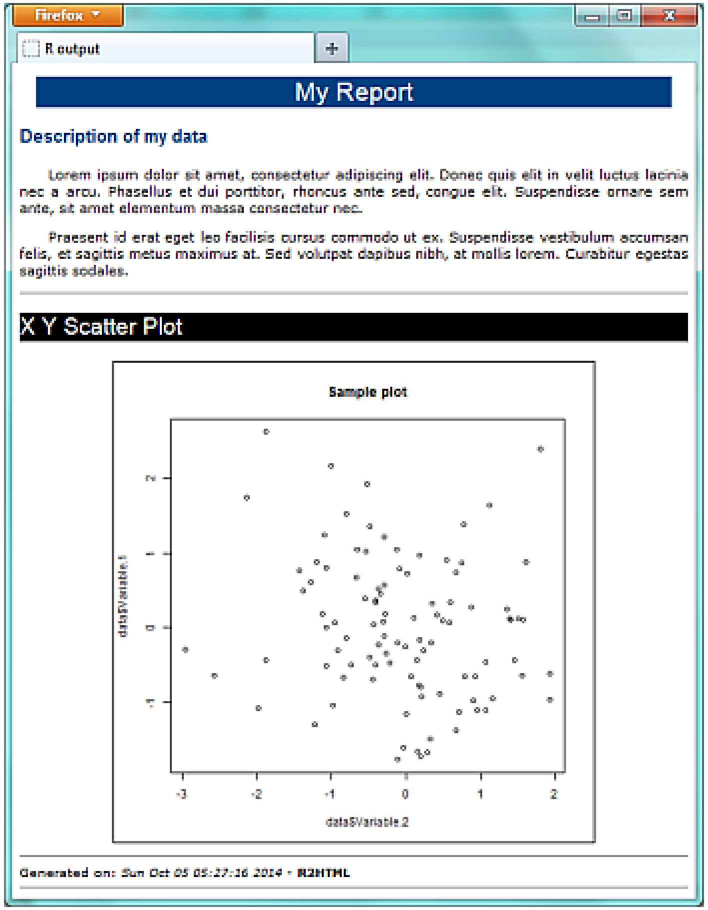
Illustration 1 : Blood pressure of Adrian Olszewski

Let's do the same otheR way

R allows the user to create documents directly from code. rtf, R2HTML, R2PPT and ReporteRs packages do the job. This is similar to how it is done in SAS.

The philosphy of document creation here is different from the approach taken in odfWeave or knitr. There is no mixing formatted paragraphs with R code. Every single object (literal, paragraph, table, graphics) is created by a specific function, directly in R.

```
data <- data.frame("Var 1" = rnorm(100),
                   "Var 2" = rnorm(100))
HTMLStart(outdir="c:/mydir", file="myreport",
extension="html", echo=FALSE...)
HTML.title("My Report", HR=1)
HTML.title("Description of my data", HR=3)
HTML("Lorem ipsum dolor sit amet,...")
HTML("Praesent id erat eget leo facilisis...")
summary(data)
HTMLhr()
HTML.title("X Y Scatter Plot", HR=2)
plot(data$Var.1~data$Var.2, main="Sample plot")
HTMLplot()
HTMLStop()
```



My Report

Description of my data

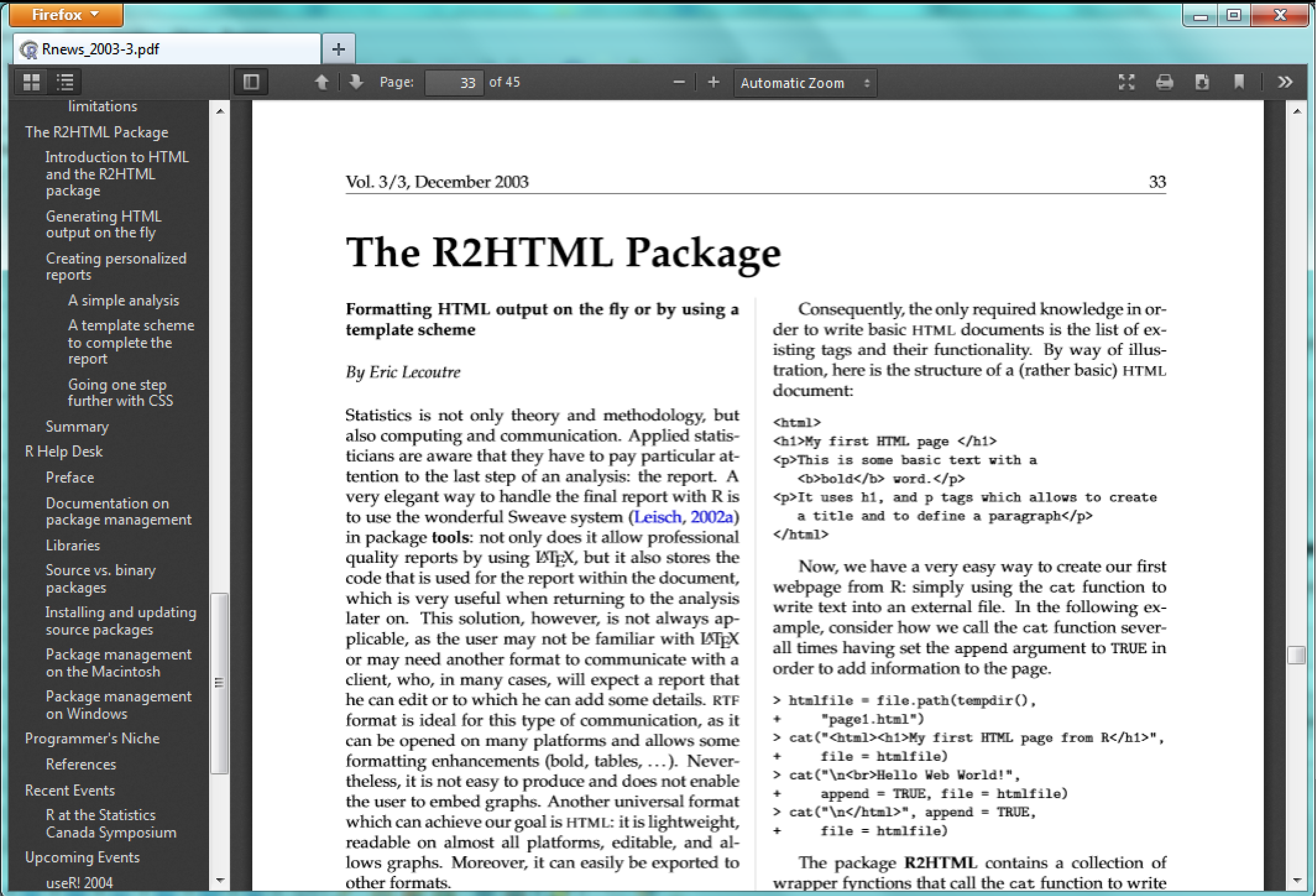
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec quis elit in velit luctus lacinia nec a arcu. Phasellus et dui porttitor, rhoncus ante sed, congue elit. Suspendisse ornare sem ante, sit amet elementum massa consectetur nec.

Praesent id erat eget leo facilisis cursus commodo ut ex. Suspendisse vestibulum accumsan felis, et sagittis metus maximus at. Sed volutpat dapibus nibh, at mollis lorem. Curabitur egestas sagittis sodales.

X Y Scatter Plot

Generated on: Sun Oct 05 05:27:16 2014 - R2HTML

R2HTML is a well known yet a bit outdated tool…



Vol. 3/3, December 2003 33

The R2HTML Package

Formatting HTML output on the fly or by using a template scheme

By Eric Lecoutre

Statistics is not only theory and methodology, but also computing and communication. Applied statisticians are aware that they have to pay particular attention to the last step of an analysis: the report. A very elegant way to handle the final report with R is to use the wonderful Sweave system (Leisch, 2002a) in package **tools**: not only does it allow professional quality reports by using LaTeX, but it also stores the code that is used for the report within the document, which is very useful when returning to the analysis later on. This solution, however, is not always applicable, as the user may not be familiar with LaTeX or may need another format to communicate with a client, who, in many cases, will expect a report that he can edit or to which he can add some details. RTF format is ideal for this type of communication, as it can be opened on many platforms and allows some formatting enhancements (bold, tables, …). Nevertheless, it is not easy to produce and does not enable the user to embed graphs. Another universal format which can achieve our goal is HTML: it is lightweight, readable on almost all platforms, editable, and allows graphs. Moreover, it can easily be exported to other formats.

Consequently, the only required knowledge in order to write basic HTML documents is the list of existing tags and their functionality. By way of illustration, here is the structure of a (rather basic) HTML document:

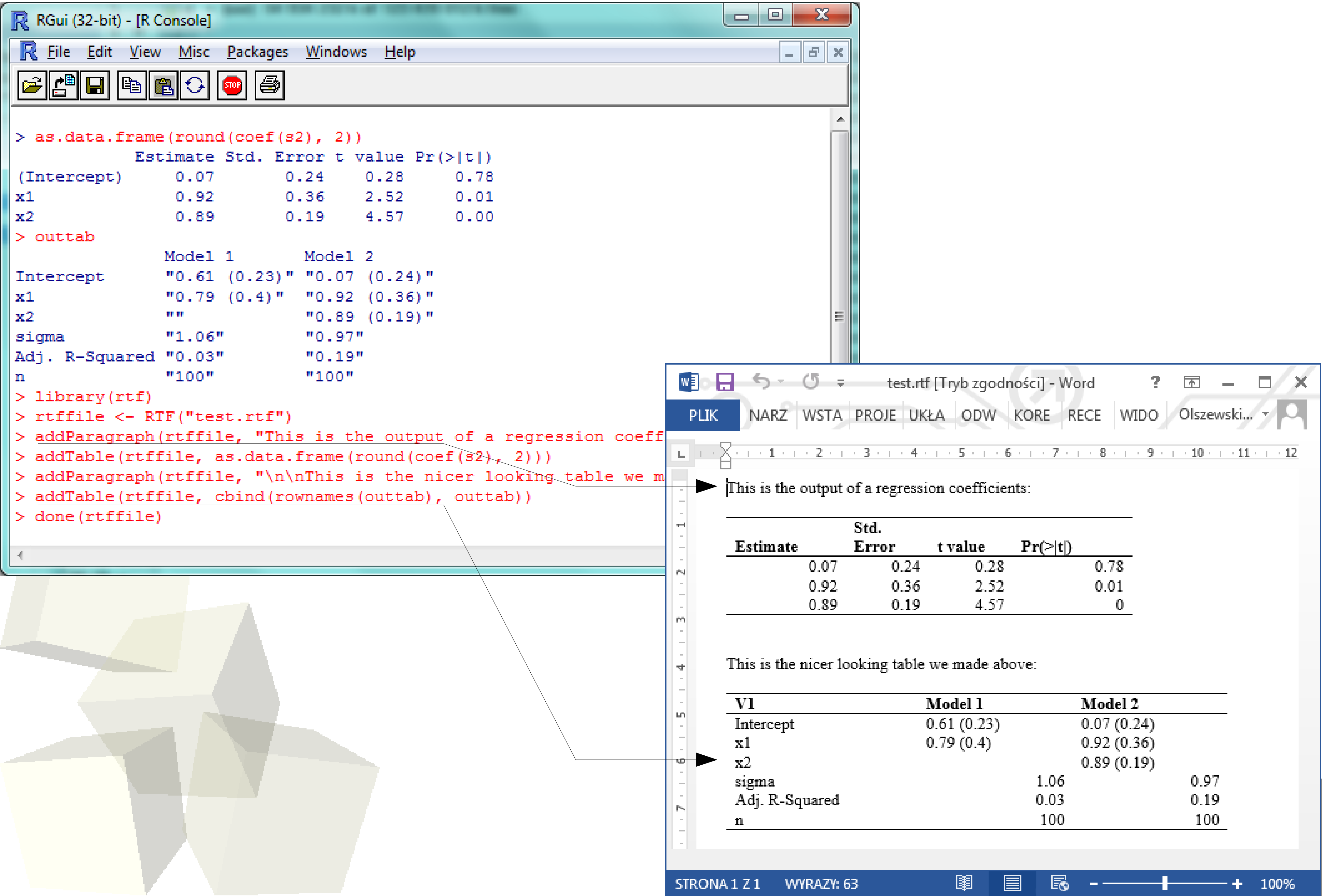
```
<html>
<h1>My first HTML page </h1>
<p>This is some basic text with a
  <b>bold</b> word.</p>
<p>It uses h1, and p tags which allows to create
  a title and to define a paragraph</p>
</html>
```

Now, we have a very easy way to create our first webpage from R: simply using the cat function to write text into an external file. In the following example, consider how we call the cat function severall times having set the append argument to TRUE in order to add information to the page.

```
> htmlfile = file.path(tempdir(),
+     "page1.html")
> cat("<html><h1>My first HTML page from R</h1>",
+     file = htmlfile)
> cat("\n<br>Hello Web World!",
+     append = TRUE, file = htmlfile)
> cat("\n</html>", append = TRUE,
+     file = htmlfile)
```

The package **R2HTML** contains a collection of wrapper fynctions that call the cat function to write

...as well as rtf package...



```
> as.data.frame(round(coef(s2), 2))
            Estimate Std. Error t value Pr(>|t|)
(Intercept)     0.07       0.24    0.28     0.78
x1              0.92       0.36    2.52     0.01
x2              0.89       0.19    4.57     0.00
> outtab
               Model 1        Model 2
Intercept      "0.61 (0.23)" "0.07 (0.24)"
x1             "0.79 (0.4)"  "0.92 (0.36)"
x2             ""            "0.89 (0.19)"
sigma          "1.06"        "0.97"
Adj. R-Squared "0.03"        "0.19"
n              "100"         "100"
> library(rtf)
> rtffile <- RTF("test.rtf")
> addParagraph(rtffile, "This is the output of a regression coeff
> addTable(rtffile, as.data.frame(round(coef(s2), 2)))
> addParagraph(rtffile, "\n\nThis is the nicer looking table we m
> addTable(rtffile, cbind(rownames(outtab), outtab))
> done(rtffile)
```

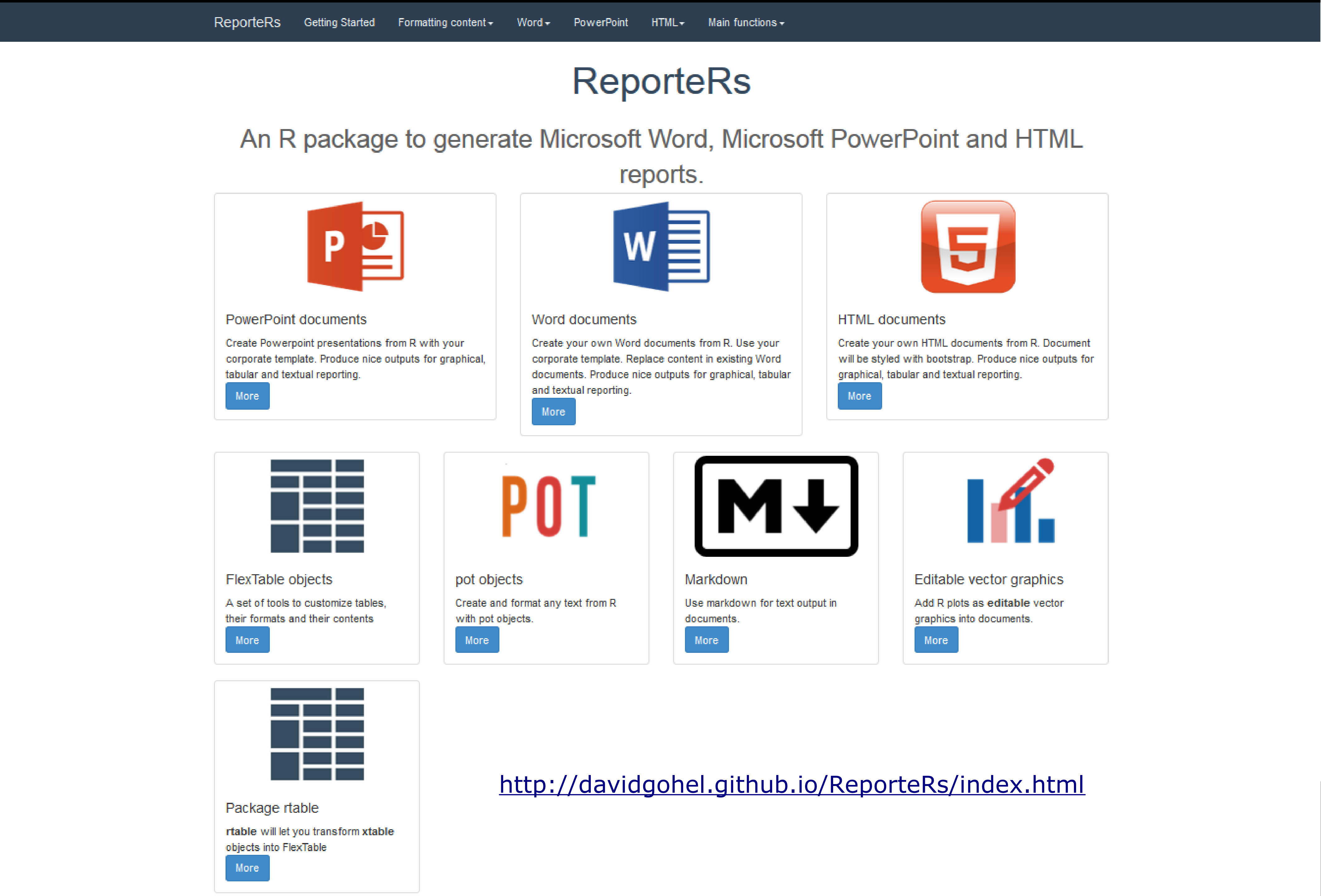
This is the output of a regression coefficients:

Estimate	Std. Error	t value	Pr(>|t|)
0.07	0.24	0.28	0.78
0.92	0.36	2.52	0.01
0.89	0.19	4.57	0

This is the nicer looking table we made above:

V1	Model 1	Model 2
Intercept	0.61 (0.23)	0.07 (0.24)
x1	0.79 (0.4)	0.92 (0.36)
x2		0.89 (0.19)
sigma	1.06	0.97
Adj. R-Squared	0.03	0.19
n	100	100

...but the new wave comes! The ReporteRs



ReporteRs

An R package to generate Microsoft Word, Microsoft PowerPoint and HTML reports.

PowerPoint documents

Create Powerpoint presentations from R with your corporate template. Produce nice outputs for graphical, tabular and textual reporting.

Word documents

Create your own Word documents from R. Use your corporate template. Replace content in existing Word documents. Produce nice outputs for graphical, tabular and textual reporting.

HTML documents

Create your own HTML documents from R. Document will be styled with bootstrap. Produce nice outputs for graphical, tabular and textual reporting.

FlexTable objects

A set of tools to customize tables, their formats and their contents

pot objects

Create and format any text from R with pot objects.

Markdown

Use markdown for text output in documents.

Editable vector graphics

Add R plots as **editable** vector graphics into documents.

Package rtable

rtable will let you transform **xtable** objects into FlexTable

http://davidgohel.github.io/ReporteRs/index.html

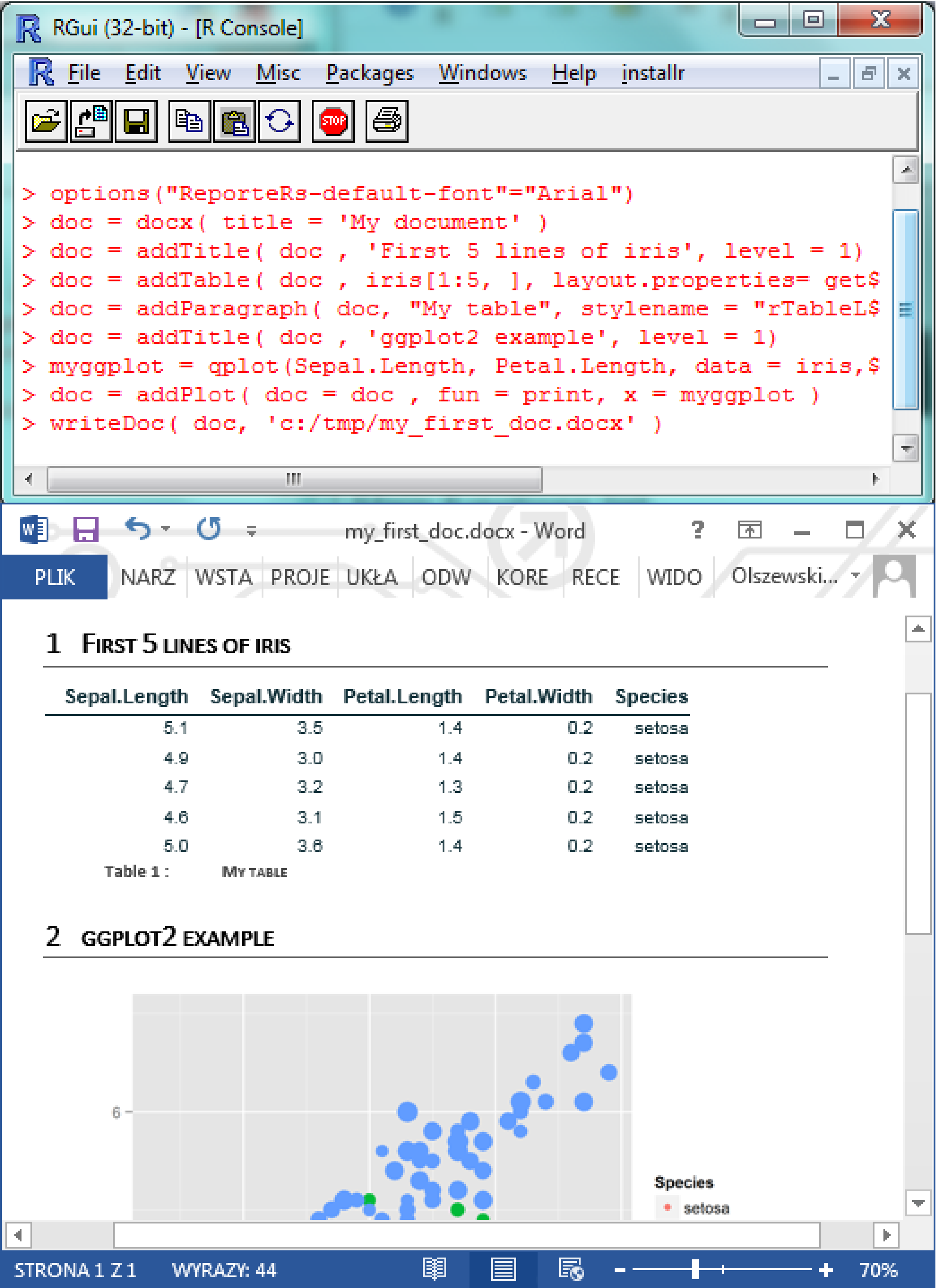
ReporteRs the King

ReporteRs is just outstanding package which allows to create Word, PowerPoint (OpenXML) and HTML5 presentations without need to have MS Office installed.

It is hard to enumerate all capabilities offered by the package:

- Styles – use the power of styles to easy control format of each object, and
  - add auto-numbered captions to tables and graphics. These captions are recognizable by TOC generator, so one can easily add a list of tables.
  - add nested, numbered or bulleted lists (also from code)
- Templates – use predefined styles, prepare overall layout of a document and fill it with a content, replacing “bookmarks” with tables and graphics
- Editable, scalable graphics (requires MS Office > 2007) – must see this!
- Flexible tables, with cells spanning multiple rows and columns, advanced, conditional formatting, “zebra style”. FlexTables are compatible with knitr.
- Multilevel ordered lists with advanced, level-dependent, numbering
- Footnotes, “pot objects” and many more!

ReporteRs in action



```
> options("ReporteRs-default-font"="Arial")
> doc = docx( title = 'My document' )
> doc = addTitle( doc , 'First 5 lines of iris', level = 1)
> doc = addTable( doc , iris[1:5, ], layout.properties= get$
> doc = addParagraph( doc, "My table", stylename = "rTableL$
> doc = addTitle( doc , 'ggplot2 example', level = 1)
> myggplot = qplot(Sepal.Length, Petal.Length, data = iris,$
> doc = addPlot( doc = doc , fun = print, x = myggplot )
> writeDoc( doc, 'c:/tmp/my_first_doc.docx' )
```

1 FIRST 5 LINES OF IRIS

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa

Table 1 : MY TABLE

2 GGPLOT2 EXAMPLE

Main functions list

All theses functions are documented in the package.

The following functions can be used whatever the output format is (docx, pptx, html):

- *addTitle*: Add a title
- *addFlexTable*: Add a FlexTable
- *addPlot*: Add plots
- *addImage*: Add external images
- *addRScript*: Add syntax highlighted R code
- *addParagraph*: Add paragraphs of text
- *addTable*: Add a table
- *writeDoc*: Write the document into a file

The following functions can only be used when the output format is docx.

- *styles*: Get available styles
- *addTOC*: Add a table of contents
- *addPageBreak* : Add a page break
- *dim* : Get page dimensions

The following functions can only be used when the output format is pptx.

- *slide.layouts*: Get available layout names
- *addSlide*: Add a slide
- *addDate*: Add a date
- *addPageNumber*: Add a page number
- *addFooter*: Add a comment in the footer
- *addSubtitle*: Add a sub title
- *dim* : Get shape dimensions

The following functions can only be used when the output format is html.

- *addPage*: Add a page

ReporteRs - tables

cyl	gear	carb	disp	mpg	wt
4	3	1	120.1	21.50	2.465
	4	1	84.2	29.10	2.072
	4	2	121.0	24.75	2.684
	5	2	107.7	28.20	1.827
6	3	1	241.5	19.75	3.337
	4	4	163.8	19.75	3.094
	5	6	145.0	19.70	2.770
8	3	2	345.5	17.15	3.560
	3	3	275.8	16.30	3.860
	3	4	416.4	12.62	4.686
	5	4	351.0	15.80	3.170
	5	8	301.0	15.00	3.570

	Estimate	Std. Error	t value	Pr(>|t|)	Signif
(Intercept)	33.991	1.888	18.006	< 0.001	***
wt	-3.206	0.754	-4.252	< 0.001	***
cyl6	-4.256	1.386	-3.070	0.00472	**
cyl8	-6.071	1.652	-3.674	< 0.001	***

Status	Ulceration	n	Mean	SD	Median	Min	Max	Missing
Alive	Absent	92	1.635	1.927	1.13	0.10	12.88	0
Alive	Present	42	3.581	2.578	3.06	0.32	12.24	0
Melanoma	Absent	16	2.703	3.350	1.94	0.32	14.66	0
Melanoma	Present	41	4.939	3.498	4.04	0.97	17.42	0
Non-melanoma	Absent	7	2.097	1.928	1.45	0.65	6.12	0
Non-melanoma	Present	7	5.339	4.328	4.84	0.16	12.56	0

	Golden Rain		Marvellous		Victory	
nitro	**est**	**df**	**est**	**df**	**est**	**df**
0	4.35 (0.08)	11.8	4.41 (0.08)	11.8	4.28 (0.08)	11.8
0.2	4.58 (0.07)	10.3	4.64 (0.07)	10.3	4.50 (0.07)	10.3
0.4	4.73 (0.07)	10.3	4.79 (0.07)	10.3	4.65 (0.07)	10.3
0.6	4.81 (0.08)	11.8	4.86 (0.08)	11.8	4.73 (0.08)	11.8

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sex	1	75.37	75.37	0.38	0.5417
ethnicty	*3*	*2572.15*	*857.38*	*4.27*	*0.0072*
grade	1	36.31	36.31	0.18	0.6717
disadvg	1	59.30	59.30	0.30	0.5882
Residuals	93	18682.87	200.89		

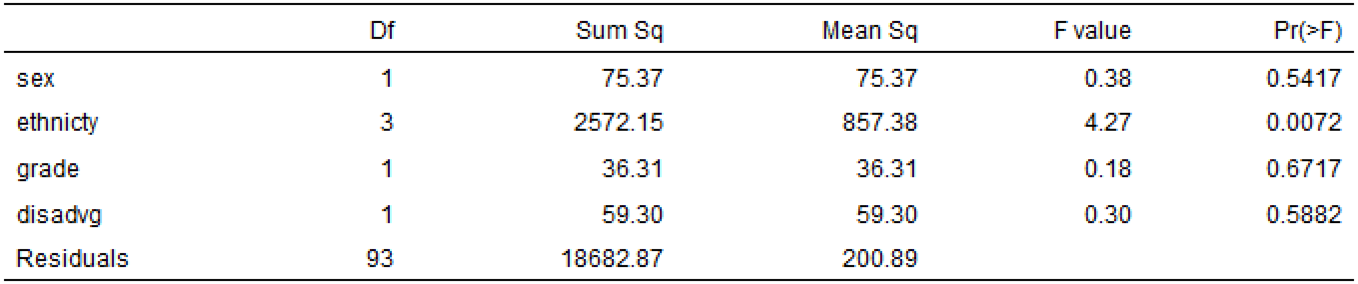
			Summary statistics for thickness	
Status	**Gender**	**Ulceration**	**Count**	**Mean (± sd)***
Alive	Female	Absent	68	1.693 (±2.004)
		Present	23	2.972 (±2.593)
	Male	Absent	24	1.468 (±1.719)
		Present	19	4.319 (±2.423)
Melanoma	Female	Absent	8	2.139 (±1.184)
		Present	20	4.724 (±4.128)
	Male	Absent	8	3.266 (±4.681)
		Present	21	5.143 (±2.862)
Non-melanoma	Female	Absent	3	1.667 (±1.141)
		Present	4	3.302 (±3.713)
	Male	Absent	4	2.420 (±2.499)
		Present	3	8.053 (±4.019)

*** This is a footer note**

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
mpg	1.000	-0.852	-0.848	-0.776	0.681	-0.868	0.419	0.664	0.600	0.480	-0.551
cyl	-0.852	1.000	0.902	0.832	-0.700	0.782	-0.591	-0.811	-0.523	-0.493	0.527
disp	-0.848	0.902	1.000	0.791	-0.710	0.888	-0.434	-0.710	-0.591	-0.556	0.395
hp	-0.776	0.832	0.791	1.000	-0.449	0.659	-0.708	-0.723	-0.243	-0.126	0.750
drat	0.681	-0.700	-0.710	-0.449	1.000	-0.712	0.091	0.440	0.713	0.700	-0.091
wt	-0.868	0.782	0.888	0.659	-0.712	1.000	-0.175	-0.555	-0.692	-0.583	0.428
qsec	0.419	-0.591	-0.434	-0.708	0.091	-0.175	1.000	0.745	-0.230	-0.213	-0.656
vs	0.664	-0.811	-0.710	-0.723	0.440	-0.555	0.745	1.000	0.168	0.206	-0.570
am	0.600	-0.523	-0.591	-0.243	0.713	-0.692	-0.230	0.168	1.000	0.794	0.058
gear	0.480	-0.493	-0.556	-0.126	0.700	-0.583	-0.213	0.206	0.794	1.000	0.274
carb	-0.551	0.527	0.395	0.750	-0.091	0.428	-0.656	-0.570	0.058	0.274	1.000

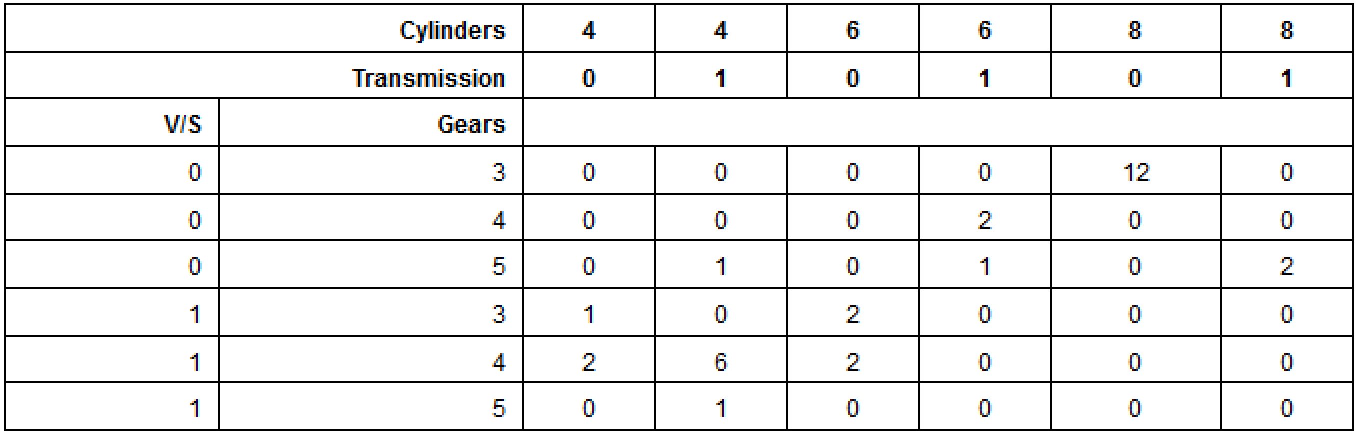
rtable

rtable is a set of functions to make easy tabular reporting from R with the ReporteRs package. It easily integrates with knitr and shiny.



	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sex	1	75.37	75.37	0.38	0.5417
ethnicty	3	2572.15	857.38	4.27	0.0072
grade	1	36.31	36.31	0.18	0.6717
disadvg	1	59.30	59.30	0.30	0.5882
Residuals	93	18682.87	200.89		

xtable → FlexTable



	Cylinders	**4**	**4**	**6**	**6**	**8**	**8**
	Transmission	**0**	**1**	**0**	**1**	**0**	**1**
V/S	**Gears**						
0	3	0	0	0	0	12	0
0	4	0	0	0	2	0	0
0	5	0	1	0	1	0	2
1	3	1	0	2	0	0	0
1	4	2	6	2	0	0	0
1	5	0	1	0	0	0	0

ftable → FlexTable

rtable

		High	Low	Middle	Sum
current	#	51	43	22	116
	%	14.33	12.08	6.18	32.58
	row %	43.97	37.07	18.97	
	col %	24.17	46.24	42.31	
former	#	92	28	21	141
	%	25.84	7.87	5.90	39.61
	row %	65.25	19.86	14.89	
	col %	43.60	30.11	40.38	
never	#	68	22	9	99
	%	19.10	6.18	2.53	27.81
	row %	68.69	22.22	9.09	
	col %	32.23	23.66	17.31	
Sum	#	211	93	52	356
	%	59.27	26.12	14.61	100.00

freqtable

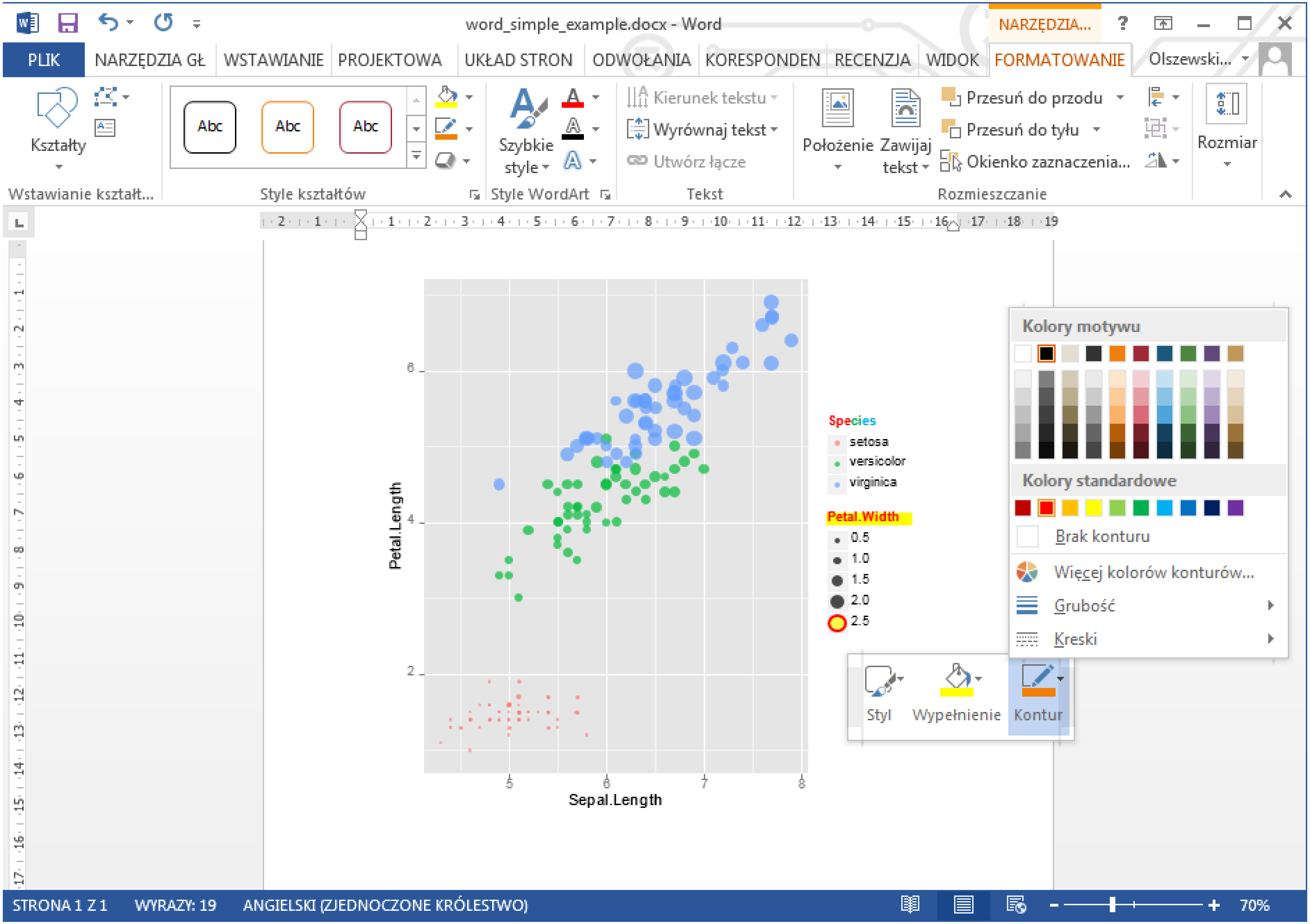
	Golden Rain		**Marvellous**		**Victory**	
nitro	**est**	**df**	**est**	**df**	**est**	**df**
0	4.35 (0.08)	11.8	4.41 (0.08)	11.8	4.28 (0.08)	11.8
0.2	4.58 (0.07)	10.3	4.64 (0.07)	10.3	4.50 (0.07)	10.3
0.4	4.73 (0.07)	10.3	4.79 (0.07)	10.3	4.65 (0.07)	10.3
0.6	4.81 (0.08)	11.8	4.86 (0.08)	11.8	4.73 (0.08)	11.8

pivot table

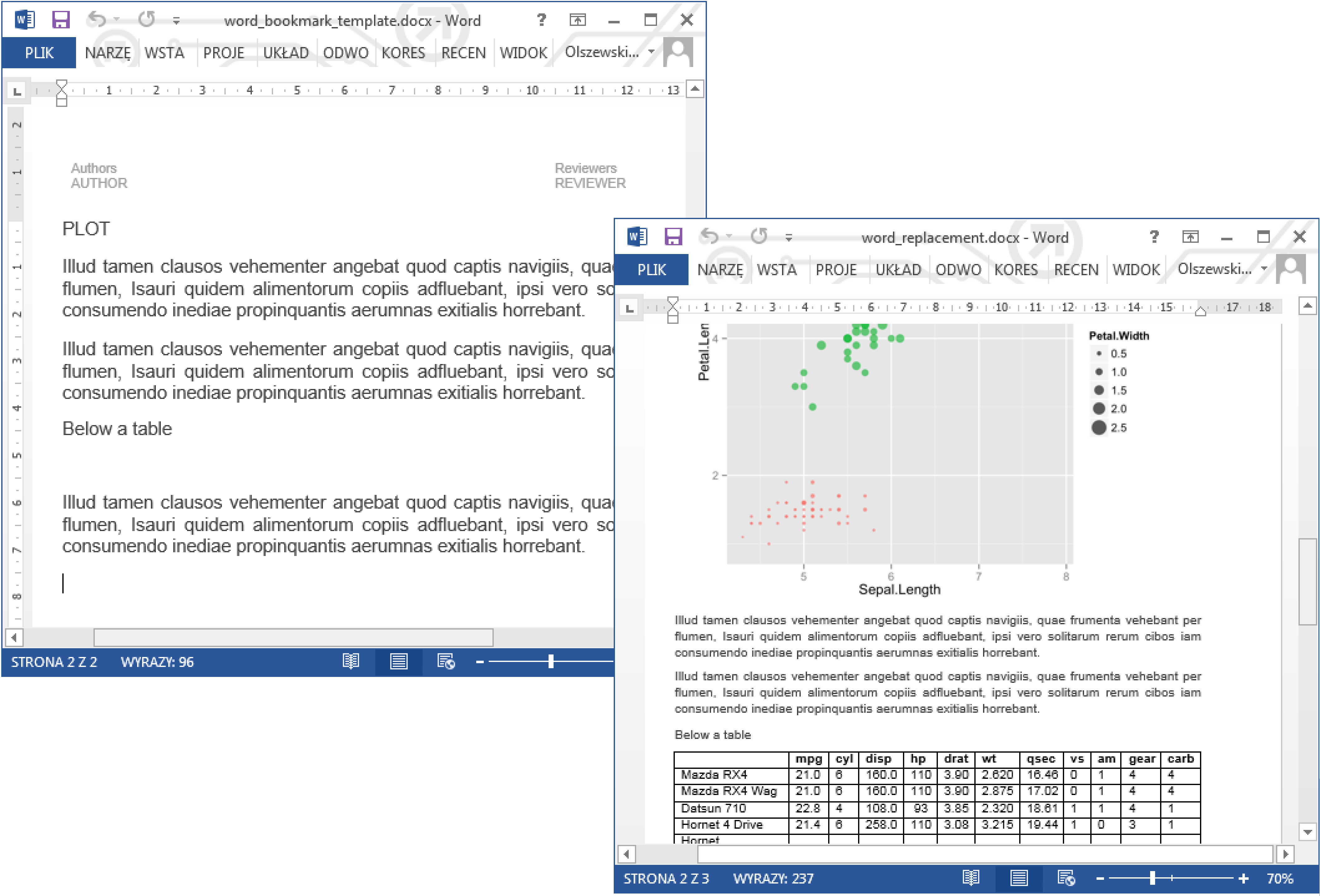
	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sex	1	75.37	75.37	0.38	0.5417
ethnicty	*3*	*2572.15*	*857.38*	*4.27*	*0.0072*
grade	1	36.31	36.31	0.18	0.6717
disadvg	1	59.30	59.30	0.30	0.5882
Residuals	93	18682.87	200.89		

custom forms

ReporteRs – editable graphics



ReporteRs – replacing content in a template



Authors
AUTHOR

Reviewers
REVIEWER

PLOT

Illud tamen clausos vehementer angebat quod captis navigiis, qua
flumen, Isauri quidem alimentorum copiis adfluebant, ipsi vero so
consumendo inediae propinquantis aerumnas exitialis horrebant.

Illud tamen clausos vehementer angebat quod captis navigiis, qua
flumen, Isauri quidem alimentorum copiis adfluebant, ipsi vero so
consumendo inediae propinquantis aerumnas exitialis horrebant.

Below a table

Illud tamen clausos vehementer angebat quod captis navigiis, qua
flumen, Isauri quidem alimentorum copiis adfluebant, ipsi vero so
consumendo inediae propinquantis aerumnas exitialis horrebant.

Illud tamen clausos vehementer angebat quod captis navigiis, quae frumenta vehebant per flumen, Isauri quidem alimentorum copiis adfluebant, ipsi vero solitarum rerum cibos iam consumendo inediae propinquantis aerumnas exitialis horrebant.

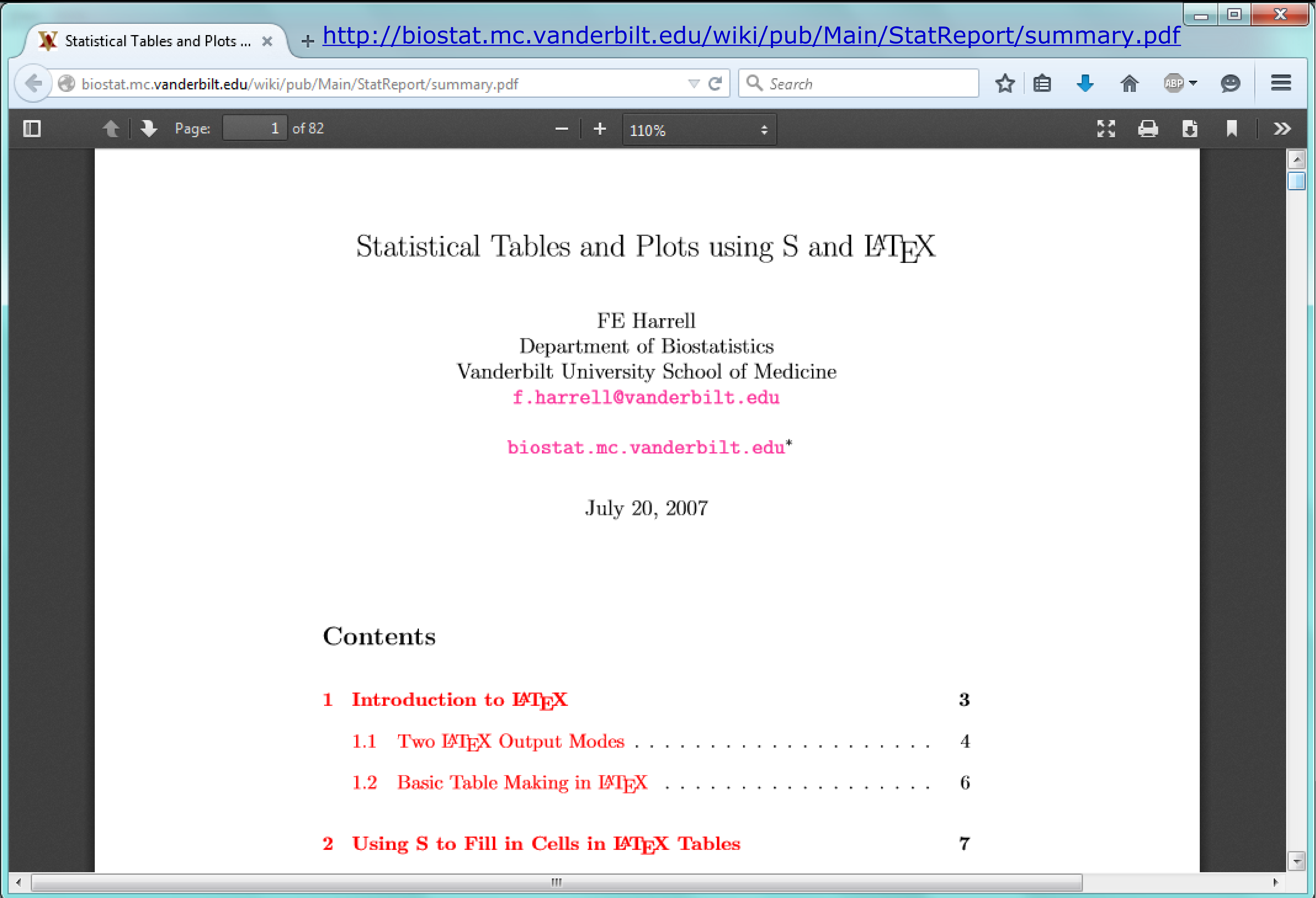
Illud tamen clausos vehementer angebat quod captis navigiis, quae frumenta vehebant per flumen, Isauri quidem alimentorum copiis adfluebant, ipsi vero solitarum rerum cibos iam consumendo inediae propinquantis aerumnas exitialis horrebant.

Below a table

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet											

What about those fancy, publication-Ready, tables?

http://biostat.mc.vanderbilt.edu/wiki/pub/Main/StatReport/summary.pdf



Statistical Tables and Plots using S and LaTeX

FE Harrell
Department of Biostatistics
Vanderbilt University School of Medicine
f.harrell@vanderbilt.edu

biostat.mc.vanderbilt.edu*

July 20, 2007

Contents

What about those fancy, publication-Ready, tables?

stargazer

beautiful LATEX, HTML and ASCII tables from R statistical output

Marek Hlavac
Harvard University

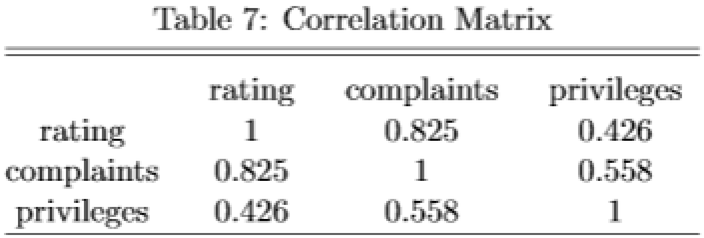


Table 7: Correlation Matrix

	rating	complaints	privileges
rating	1	0.825	0.426
complaints	0.825	1	0.558
privileges	0.426	0.558	1

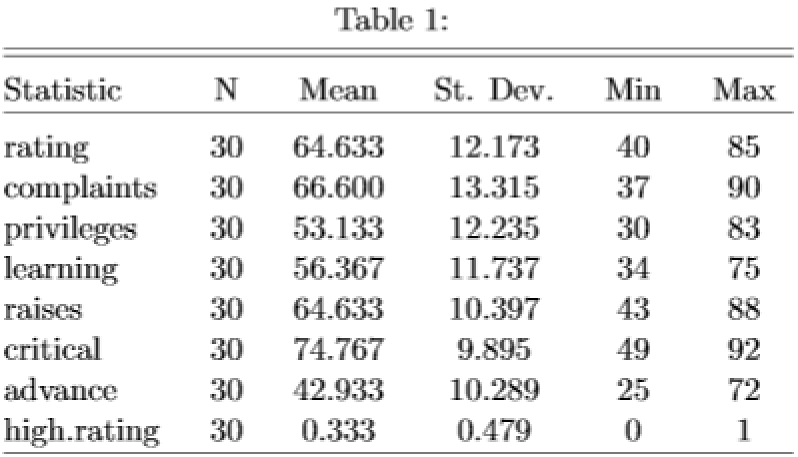


Table 1:

Statistic	N	Mean	St. Dev.	Min	Max
rating	30	64.633	12.173	40	85
complaints	30	66.600	13.315	37	90
privileges	30	53.133	12.235	30	83
learning	30	56.367	11.737	34	75
raises	30	64.633	10.397	43	88
critical	30	74.767	9.895	49	92
advance	30	42.933	10.289	25	72
high.rating	30	0.333	0.479	0	1

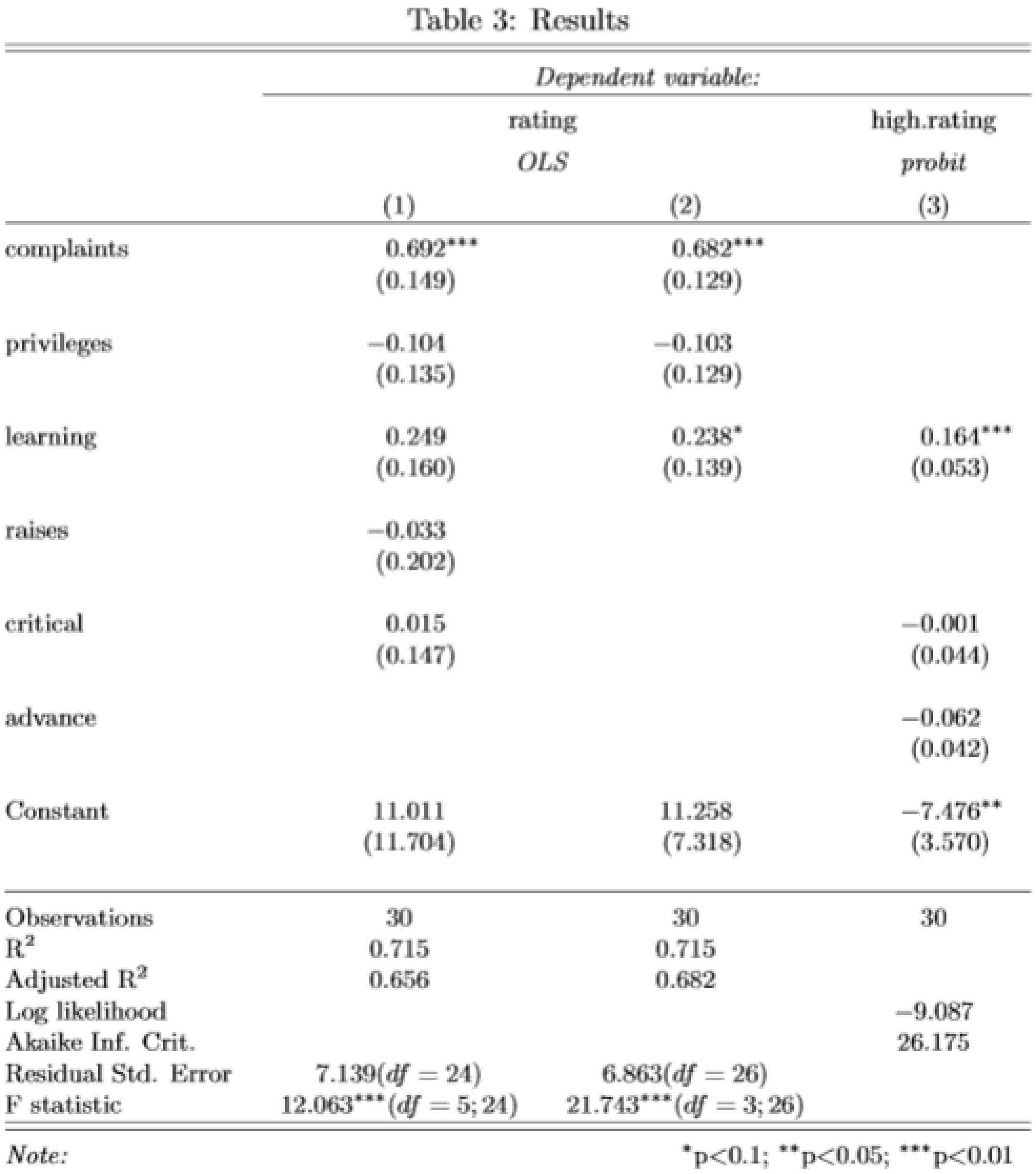


Table 3: Results

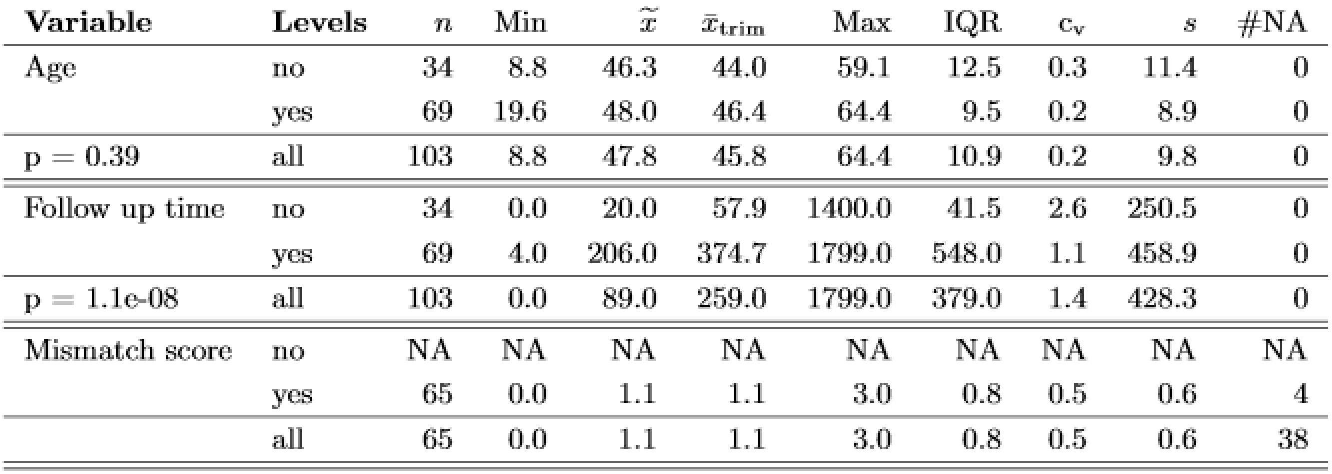
	Dependent variable:		
	rating		high.rating
	OLS		*probit*
	(1)	(2)	(3)
complaints	0.692^{***}	0.682^{***}	
	(0.149)	(0.129)	
privileges	−0.104	−0.103	
	(0.135)	(0.129)	
learning	0.249	0.238^{*}	0.164^{***}
	(0.160)	(0.139)	(0.053)
raises	−0.033		
	(0.202)		
critical	0.015		−0.001
	(0.147)		(0.044)
advance			−0.062
			(0.042)
Constant	11.011	11.258	-7.476^{**}
	(11.704)	(7.318)	(3.570)
Observations	30	30	30
R^2	0.715	0.715	
Adjusted R^2	0.656	0.682	
Log likelihood			−9.087
Akaike Inf. Crit.			26.175
Residual Std. Error	7.139(*df* = 24)	6.863(*df* = 26)	
F statistic	12.063^{***}(*df* = 5; 24)	21.743^{***}(*df* = 3; 26)	
Note:			*p<0.1; **p<0.05; ***p<0.01

What about those fancy, publication-ready, tables?

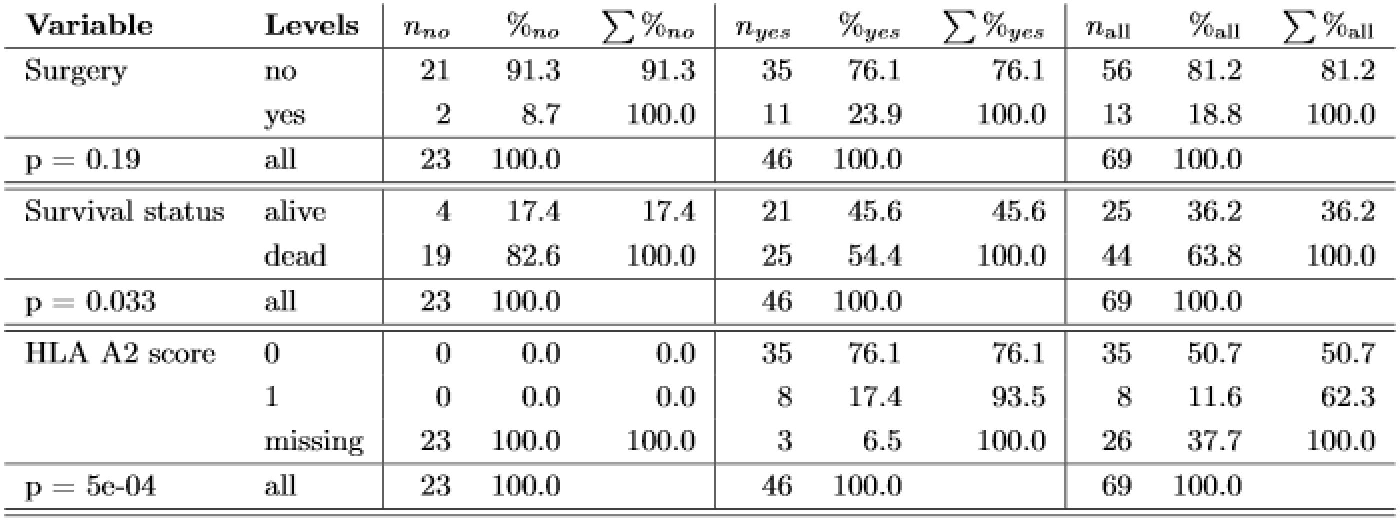
reporttools

R Functions to Generate
LATEX Tables of
Descriptive Statistics

Kaspar Rubach
University of Zurich



Variable	Levels	n	Min	$\widetilde{x}$	$\bar{x}_{\text{trim}}$	Max	IQR	c_v	s	#NA
Age	no	34	8.8	46.3	44.0	59.1	12.5	0.3	11.4	0
	yes	69	19.6	48.0	46.4	64.4	9.5	0.2	8.9	0
$p = 0.39$	all	103	8.8	47.8	45.8	64.4	10.9	0.2	9.8	0
Follow up time	no	34	0.0	20.0	57.9	1400.0	41.5	2.6	250.5	0
	yes	69	4.0	206.0	374.7	1799.0	548.0	1.1	458.9	0
$p = 1.1e{-}08$	all	103	0.0	89.0	259.0	1799.0	379.0	1.4	428.3	0
Mismatch score	no	NA	NA	NA	NA	NA	NA	NA	NA	NA
	yes	65	0.0	1.1	1.1	3.0	0.8	0.5	0.6	4
	all	65	0.0	1.1	1.1	3.0	0.8	0.5	0.6	38



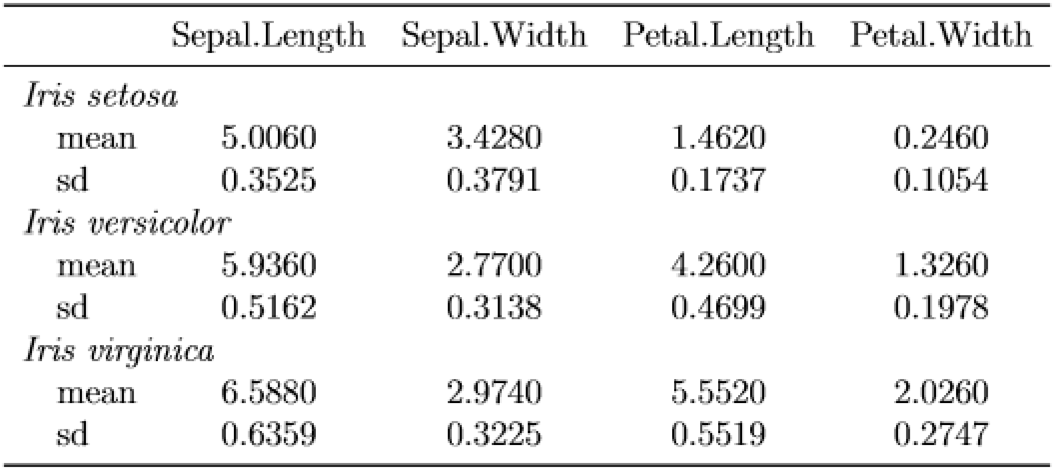
Variable	Levels	n_{no}	$\%_{no}$	$\sum \%_{no}$	n_{yes}	$\%_{yes}$	$\sum \%_{yes}$	n_{all}	$\%_{\text{all}}$	$\sum \%_{\text{all}}$
Surgery	no	21	91.3	91.3	35	76.1	76.1	56	81.2	81.2
	yes	2	8.7	100.0	11	23.9	100.0	13	18.8	100.0
$p = 0.19$	all	23	100.0		46	100.0		69	100.0	
Survival status	alive	4	17.4	17.4	21	45.6	45.6	25	36.2	36.2
	dead	19	82.6	100.0	25	54.4	100.0	44	63.8	100.0
$p = 0.033$	all	23	100.0		46	100.0		69	100.0	
HLA A2 score	0	0	0.0	0.0	35	76.1	76.1	35	50.7	50.7
	1	0	0.0	0.0	8	17.4	93.5	8	11.6	62.3
	missing	23	100.0	100.0	3	6.5	100.0	26	37.7	100.0
$p = 5e{-}04$	all	23	100.0		46	100.0		69	100.0	

What about those fancy, publication-ready, tables?

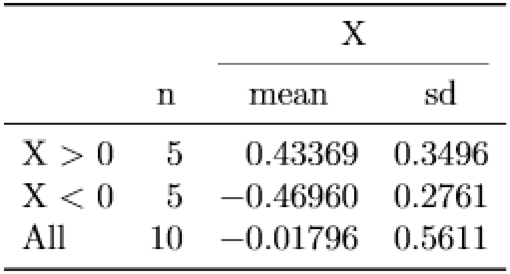
tables

Computes and displays complex tables of summary statistics

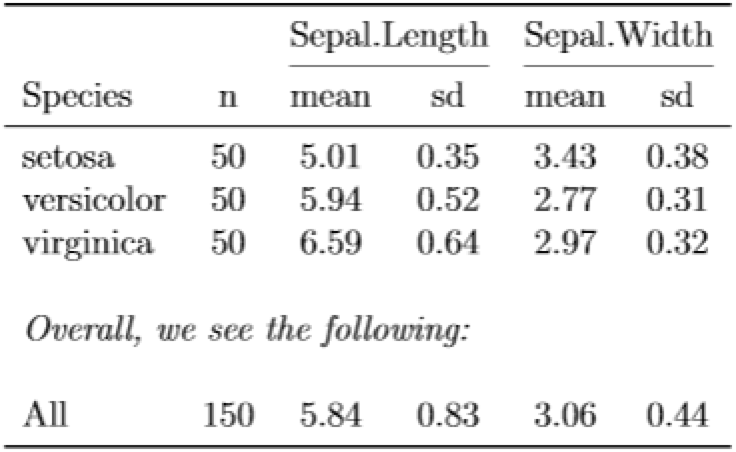
Duncan Murdoch
University of Western Ontario



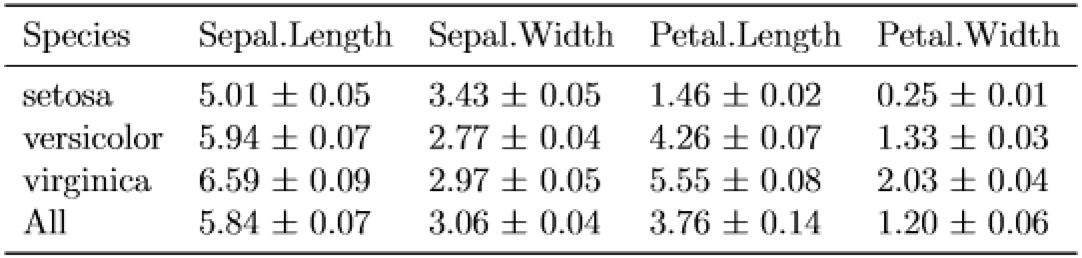
	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
Iris setosa				
mean	5.0060	3.4280	1.4620	0.2460
sd	0.3525	0.3791	0.1737	0.1054
Iris versicolor				
mean	5.9360	2.7700	4.2600	1.3260
sd	0.5162	0.3138	0.4699	0.1978
Iris virginica				
mean	6.5880	2.9740	5.5520	2.0260
sd	0.6359	0.3225	0.5519	0.2747



		X	
	n	mean	sd
$X > 0$	5	0.43369	0.3496
$X < 0$	5	−0.46960	0.2761
All	10	−0.01796	0.5611



		Sepal.Length		Sepal.Width	
Species	n	mean	sd	mean	sd
setosa	50	5.01	0.35	3.43	0.38
versicolor	50	5.94	0.52	2.77	0.31
virginica	50	6.59	0.64	2.97	0.32
Overall, we see the following:					
All	150	5.84	0.83	3.06	0.44



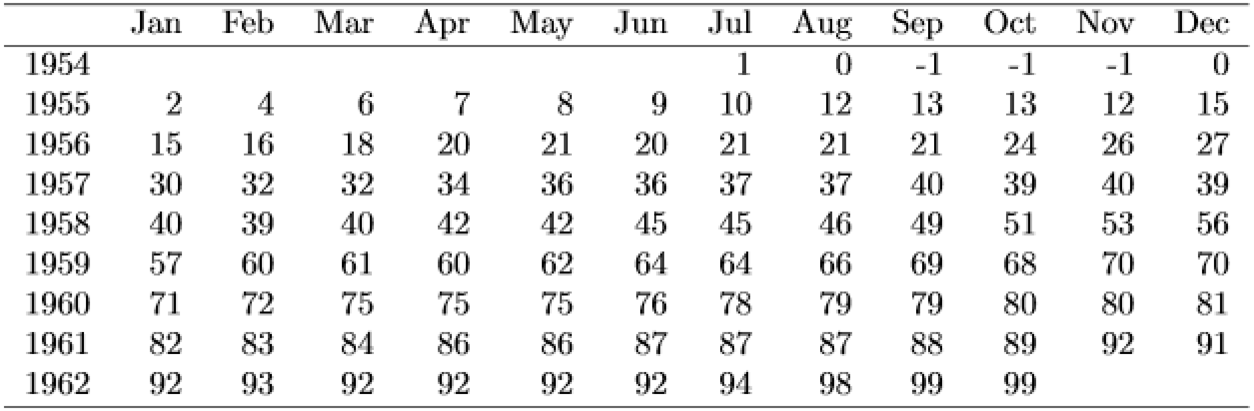
Species	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
setosa	5.01 ± 0.05	3.43 ± 0.05	1.46 ± 0.02	0.25 ± 0.01
versicolor	5.94 ± 0.07	2.77 ± 0.04	4.26 ± 0.07	1.33 ± 0.03
virginica	6.59 ± 0.09	2.97 ± 0.05	5.55 ± 0.08	2.03 ± 0.04
All	5.84 ± 0.07	3.06 ± 0.04	3.76 ± 0.14	1.20 ± 0.06

What about those fancy, publication-ready, tables?

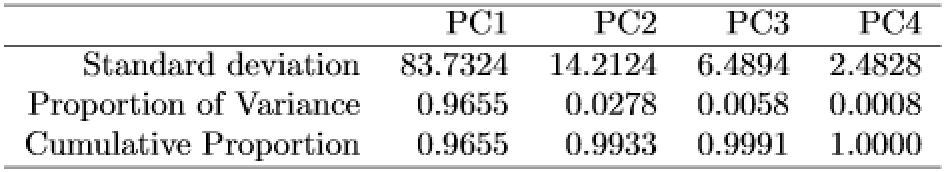
xtables

Coerce data to LaTeX and HTML tables

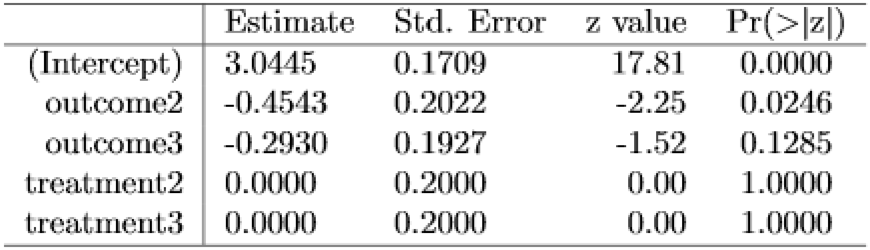
David B. Dahl
Brigham Young University



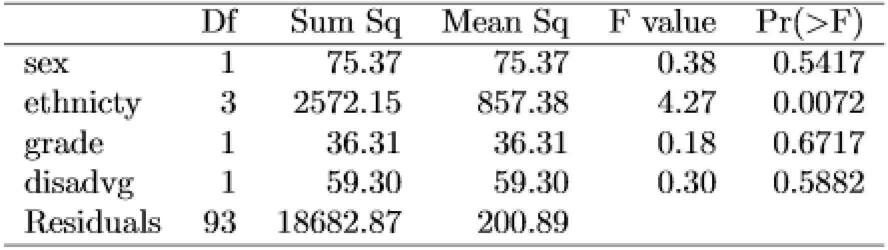
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1954							1	0	-1	-1	-1	0
1955	2	4	6	7	8	9	10	12	13	13	12	15
1956	15	16	18	20	21	20	21	21	21	24	26	27
1957	30	32	32	34	36	36	37	37	40	39	40	39
1958	40	39	40	42	42	45	45	46	49	51	53	56
1959	57	60	61	60	62	64	64	66	69	68	70	70
1960	71	72	75	75	75	76	78	79	79	80	80	81
1961	82	83	84	86	86	87	87	87	88	89	92	91
1962	92	93	92	92	92	92	94	98	99	99		



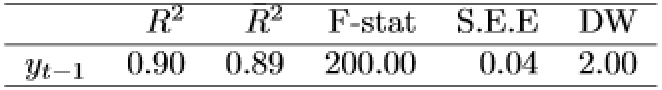
	PC1	PC2	PC3	PC4
Standard deviation	83.7324	14.2124	6.4894	2.4828
Proportion of Variance	0.9655	0.0278	0.0058	0.0008
Cumulative Proportion	0.9655	0.9933	0.9991	1.0000



	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	3.0445	0.1709	17.81	0.0000
outcome2	-0.4543	0.2022	-2.25	0.0246
outcome3	-0.2930	0.1927	-1.52	0.1285
treatment2	0.0000	0.2000	0.00	1.0000
treatment3	0.0000	0.2000	0.00	1.0000



	Df	Sum Sq	Mean Sq	F value	Pr(>F)
sex	1	75.37	75.37	0.38	0.5417
ethnicty	3	2572.15	857.38	4.27	0.0072
grade	1	36.31	36.31	0.18	0.6717
disadvg	1	59.30	59.30	0.30	0.5882
Residuals	93	18682.87	200.89		



	R^2	$\bar{R}^2$	F-stat	S.E.E	DW
y_{t-1}	0.90	0.89	200.00	0.04	2.00

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is (widely) supported by the world of science

IV R is supported by the community

V R is (increasingly) supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX ½ :) Graphics – all have waited for this moment :)

X There are many possibilities to optimize the code

XI R is able to handle large amount of data

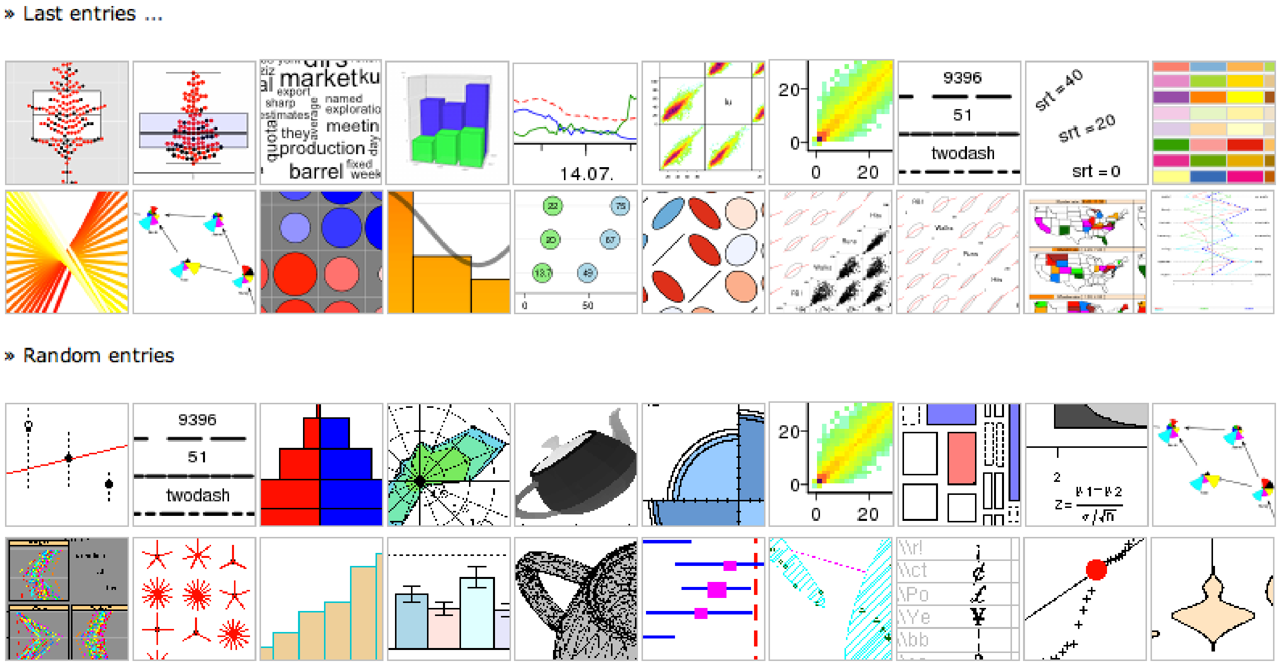
XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

a picture is woRth a thousand words

It is really hard to describe all the potential of R in creating graphics. Internet is full of of examples and tutorials of how to create complex and sophisticated charts in R. There are thousands of examples.

The user has at his disposal a number of libraries allowing creation of almost every chart or graphics he needs. R is able to produce clean, ascetic charts suitable for journals and books, and super fancy graphics, perfect for presentations.



Imagination

is the only limit here

http://rgraphgallery.blogspot.com/

11th commandment

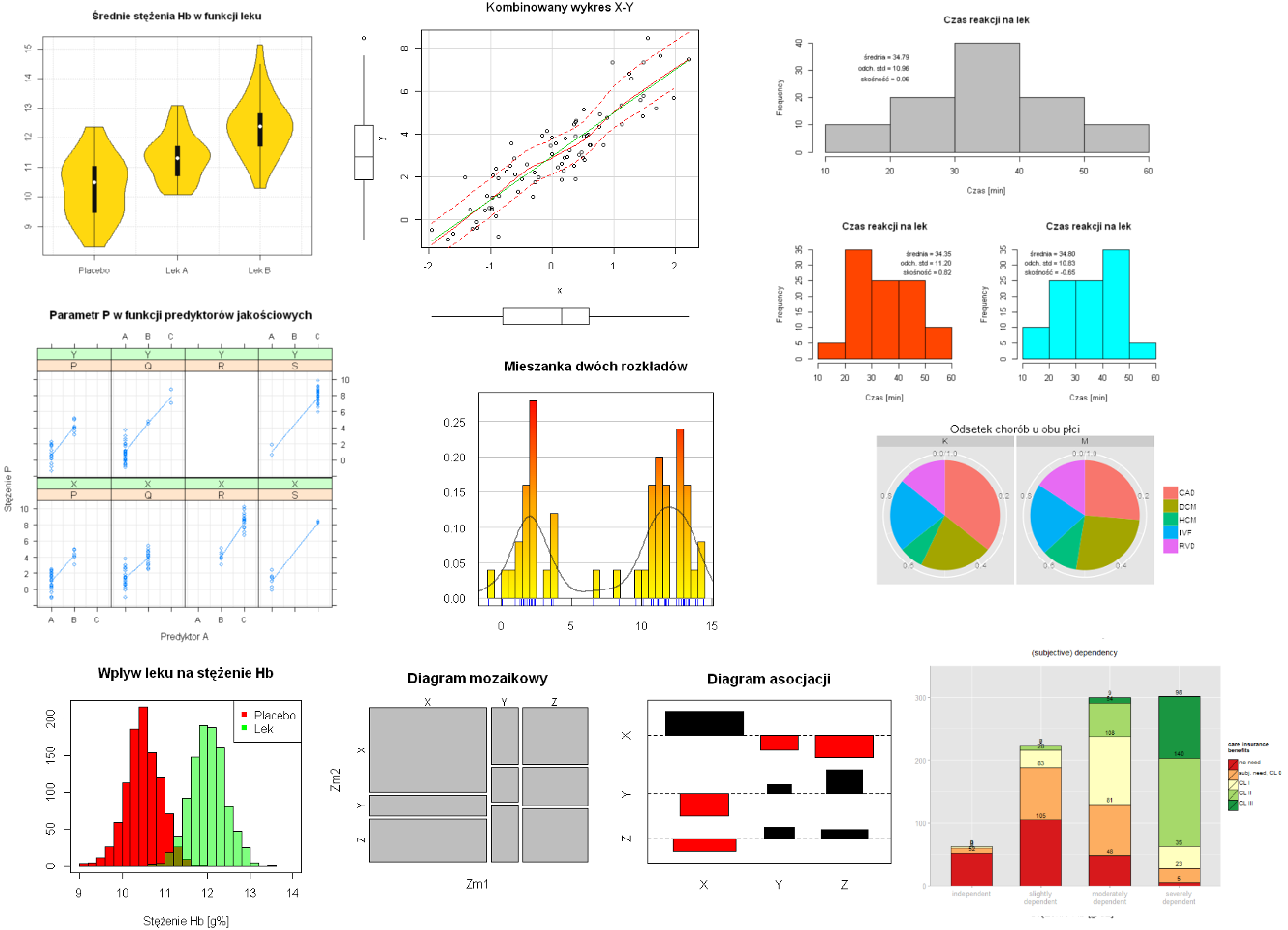
If anything, there should be a Law:

*Thou Shalt **Not Even Think** Of Producing A Graph*

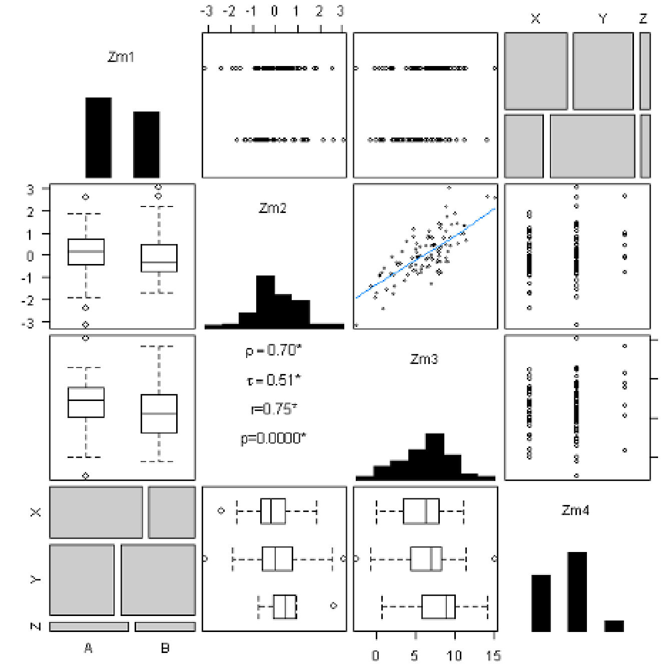
*That Looks Like **Anything From A Spreadsheet***

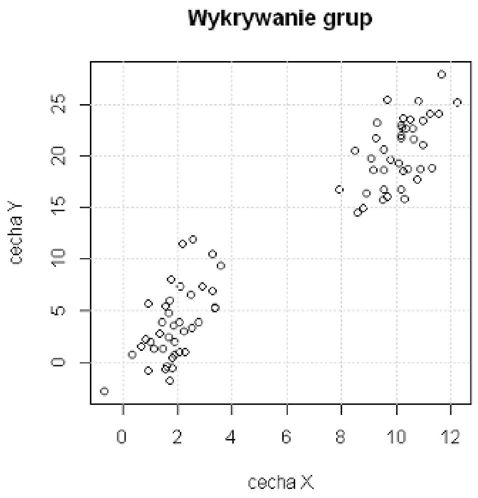
Ted Harding

R has great charting capabilities



…of any kind


Zm1
Zm2
Zm3
Zm4
ρ = 0.70*
τ = 0.51*
r=0.75*
p=0.0000*
X
Y
Z
A
B
-3 -2 -1 0 1 2 3
0 5 10 15


Wykrywanie grup
cecha Y
cecha X
0 5 10 15 20 25
0 2 4 6 8 10 12

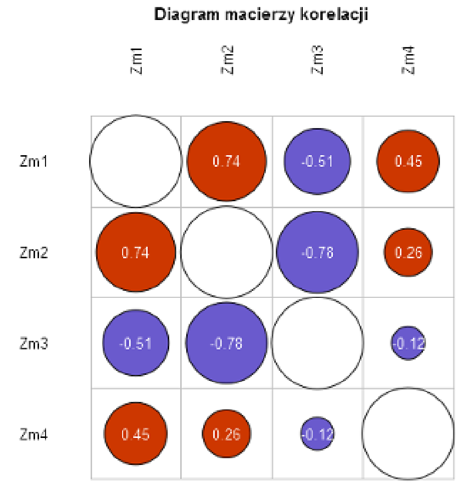
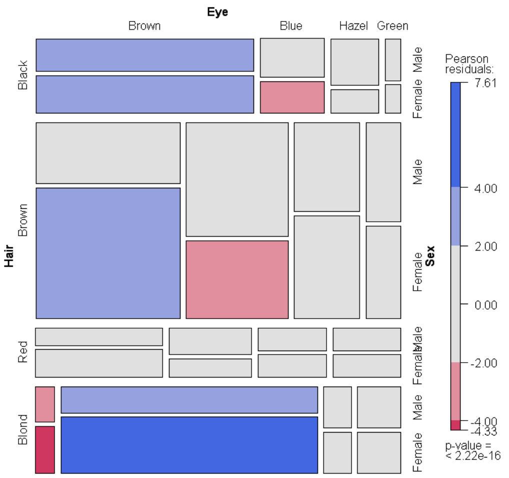
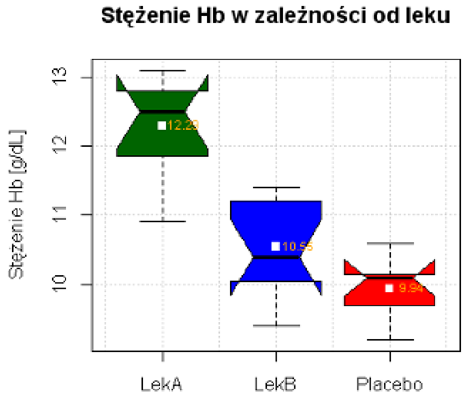
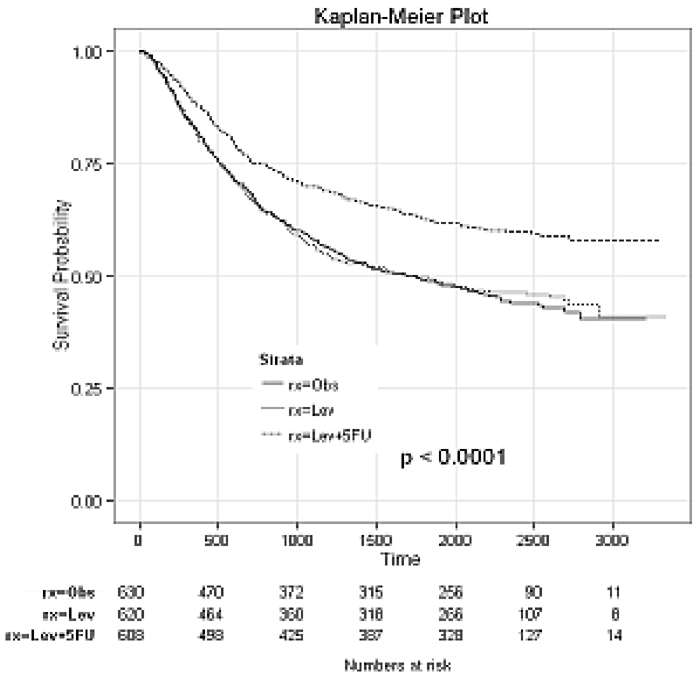
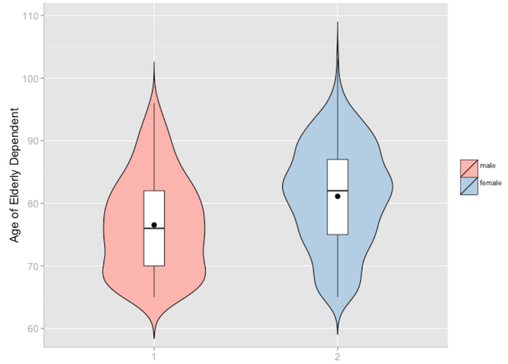


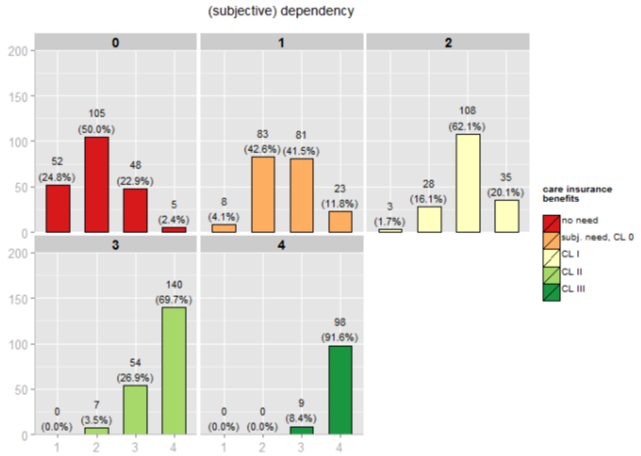
Diagram macierzy korelacji
Zm1
Zm2
Zm3
Zm4
0.74
-0.51
0.45
-0.78
0.26
-0.12


Eye
Brown
Blue
Hazel
Green
Hair
Black
Brown
Red
Blond
Sex
Male
Female
Pearson residuals:
7.61
4.00
2.00
0.00
-2.00
-4.00
-4.33
p-value = < 2.22e-16

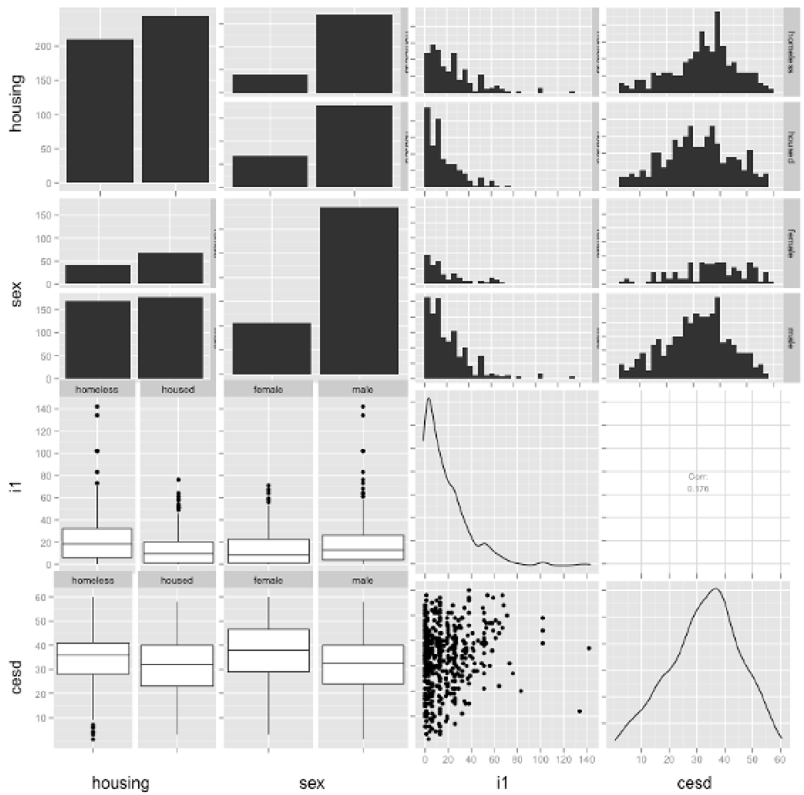

Stężenie Hb w zależności od leku
Stężenie Hb [g/dL]
10
11
12
13
12.23
10.55
9.94
LekA
LekB
Placebo

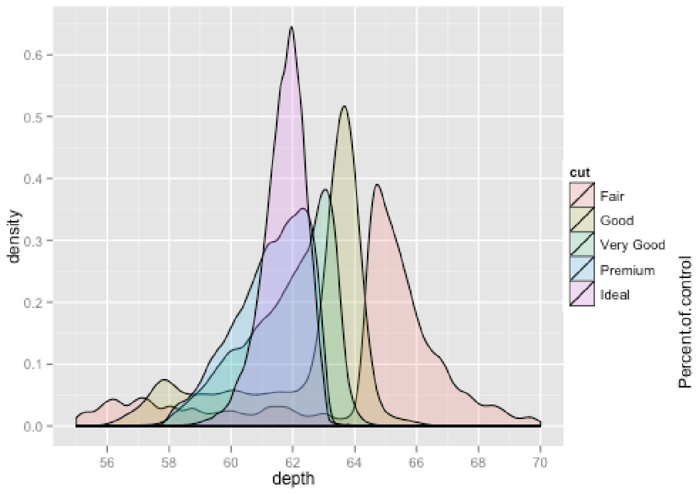

Kaplan-Meier Plot
Survival Probability
1.00
0.75
0.50
0.25
0.00
Strata
rx=Obs
rx=Lev
rx=Lev+5FU
p < 0.0001
0 500 1000 1500 2000 2500 3000
Time
rx=Obs 630 470 372 315 256 90 11
rx=Lev 620 464 360 318 266 107 8
rx=Lev+5FU 608 498 425 387 328 127 14
Numbers at risk

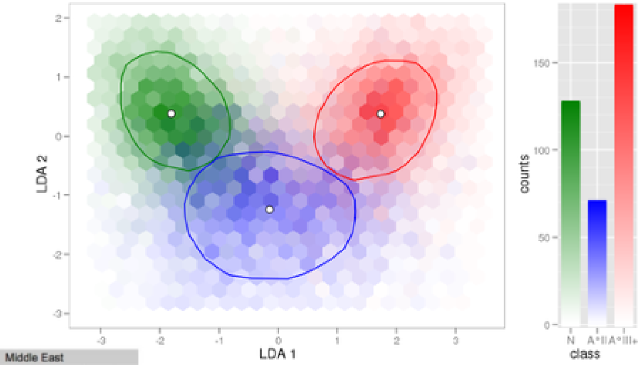

Age of Elderly Dependent
60
70
80
90
100
110
1
2
male
female

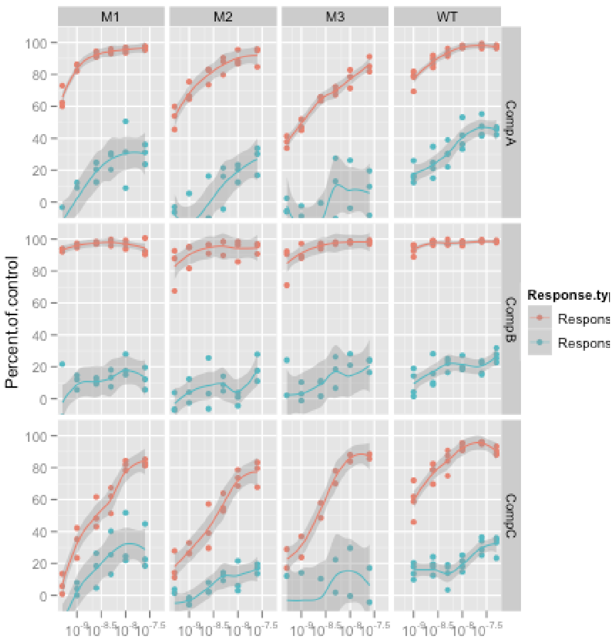

(subjective) dependency
0
1
2
3
4
52 (24.8%)
105 (50.0%)
48 (22.9%)
5 (2.4%)
8 (4.1%)
83 (42.6%)
81 (41.5%)
23 (11.8%)
3 (1.7%)
28 (16.1%)
108 (62.1%)
35 (20.1%)
0 (0.0%)
7 (3.5%)
54 (26.9%)
140 (69.7%)
0 (0.0%)
0 (0.0%)
9 (8.4%)
98 (91.6%)
care insurance benefits
no need
subj. need, CL 0
CL I
CL II
CL III

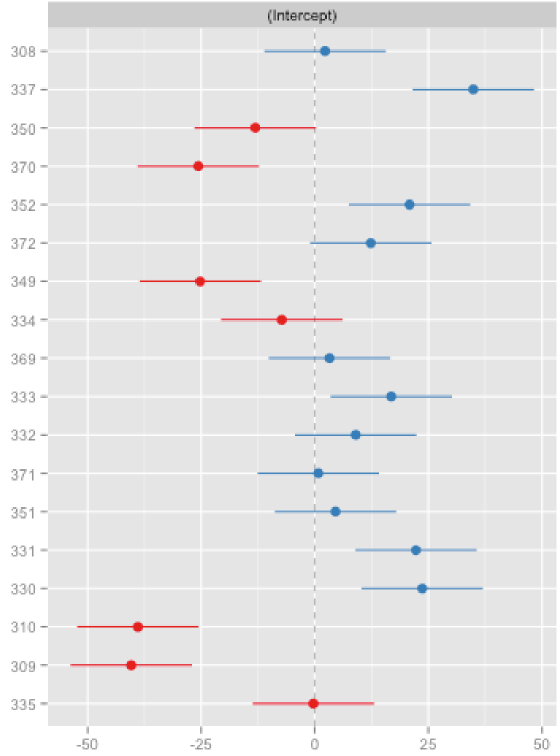
...and yes, these charts can be very elegant :)

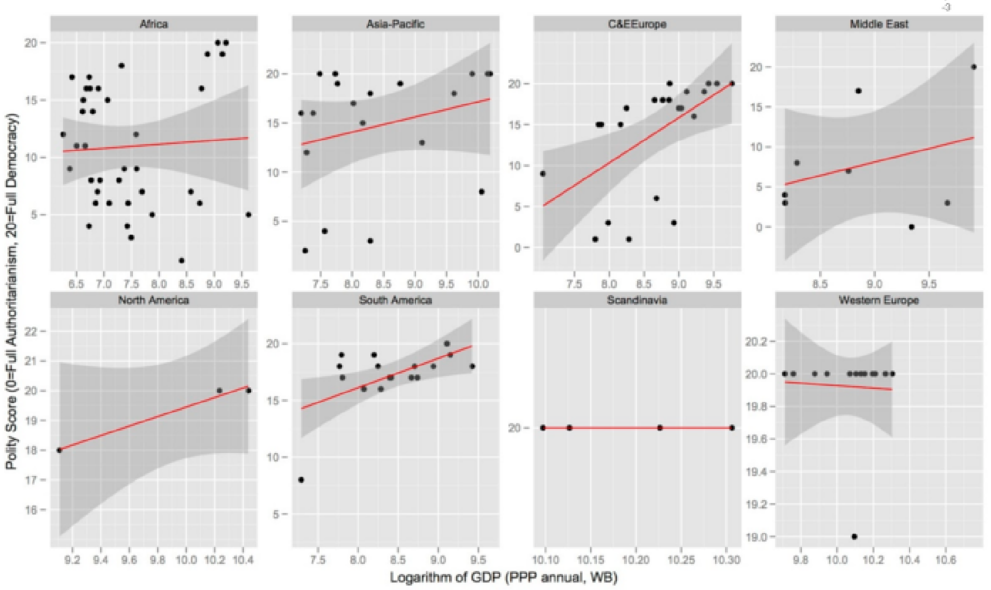


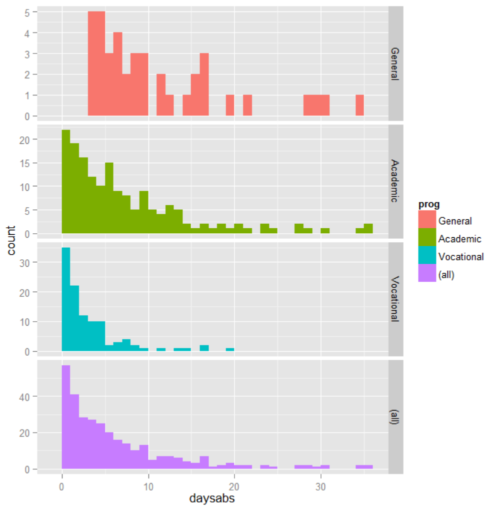




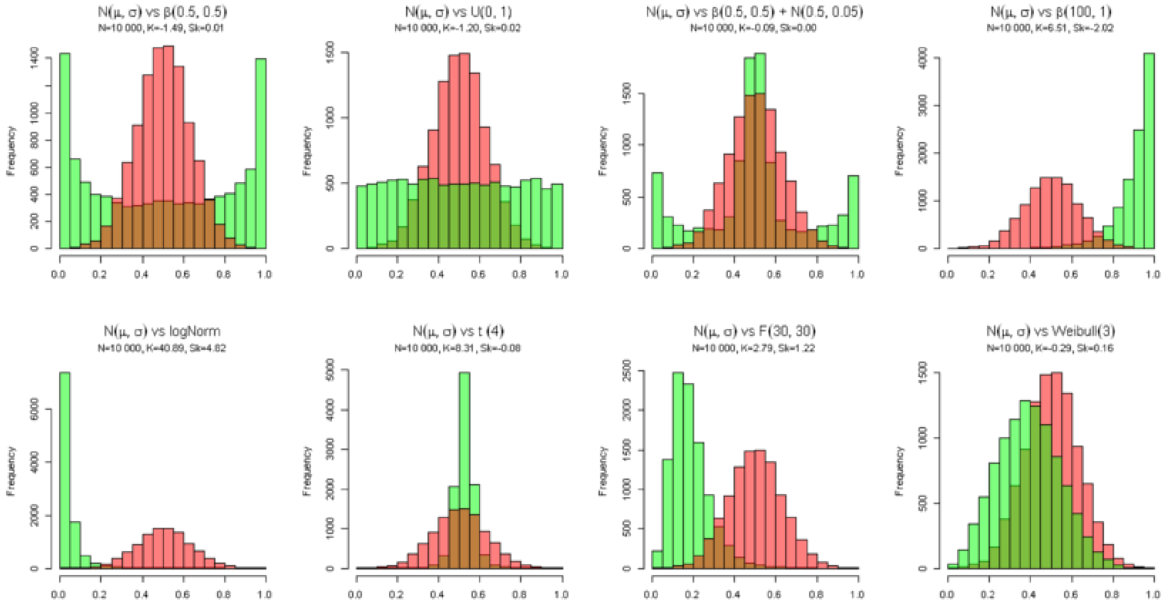




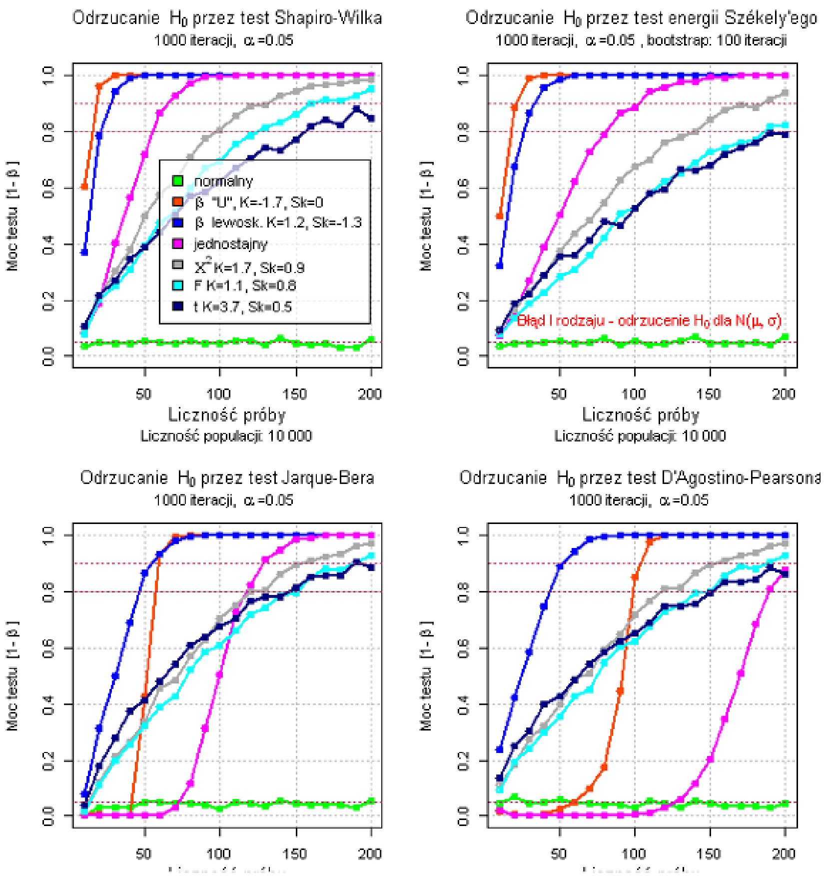




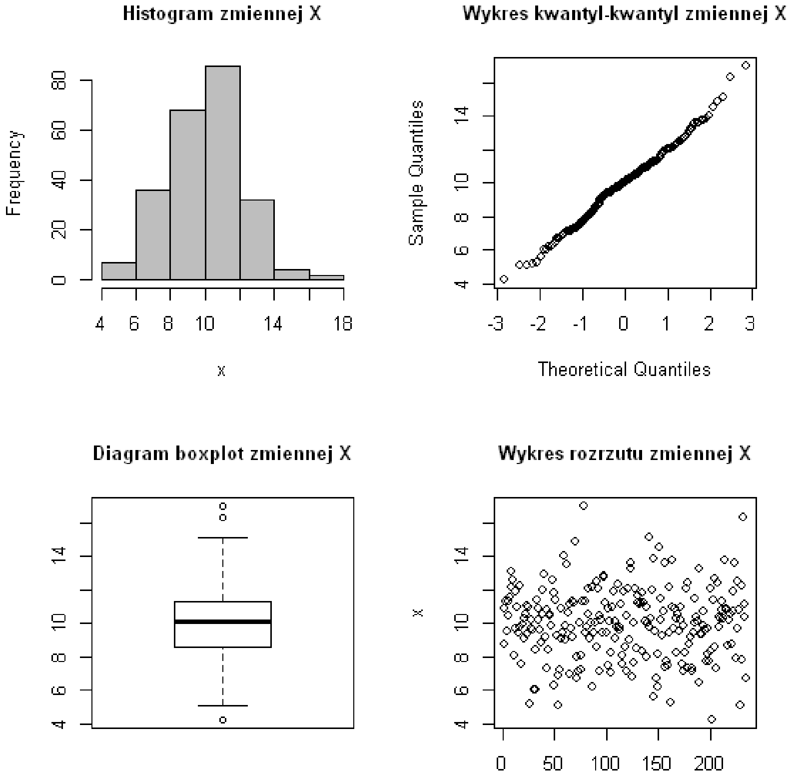
...and varied...



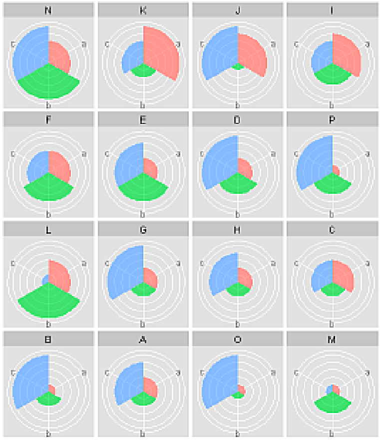
N(μ, σ) vs β(0.5, 0.5)
N=10 000, K=-1.49, Sk=0.01
N(μ, σ) vs U(0, 1)
N=10 000, K=-1.20, Sk=0.02
N(μ, σ) vs β(0.5, 0.5) + N(0.5, 0.05)
N=10 000, K=-0.09, Sk=0.00
N(μ, σ) vs β(100, 1)
N=10 000, K=6.51, Sk=-2.02
N(μ, σ) vs logNorm
N=10 000, K=40.89, Sk=4.82
N(μ, σ) vs t (4)
N=10 000, K=8.31, Sk=-0.08
N(μ, σ) vs F(30, 30)
N=10 000, K=2.79, Sk=1.22
N(μ, σ) vs Weibull(3)
N=10 000, K=-0.29, Sk=0.16
Frequency
0.0 0.2 0.4 0.6 0.8 1.0



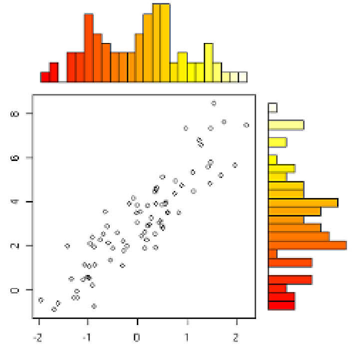
Odrzucanie H_0 przez test Shapiro-Wilka
1000 iteracji, α =0.05
Odrzucanie H_0 przez test energii Székely'ego
1000 iteracji, α =0.05 , bootstrap: 100 iteracji
Odrzucanie H_0 przez test Jarque-Bera
1000 iteracji, α =0.05
Odrzucanie H_0 przez test D'Agostino-Pearsona
1000 iteracji, α =0.05
Moc testu [1 - β]
Liczność próby
Liczność populacji: 10 000
normalny
β "U", K=-1.7, Sk=0
β lewosk. K=1.2, Sk=-1.3
jednostajny
X^2 K=1.7, Sk=0.9
F K=1.1, Sk=0.8
t K=3.7, Sk=0.5
Błąd I rodzaju - odrzucenie H_0 dla N(μ, σ)
0.0 0.2 0.4 0.6 0.8 1.0
50 100 150 200

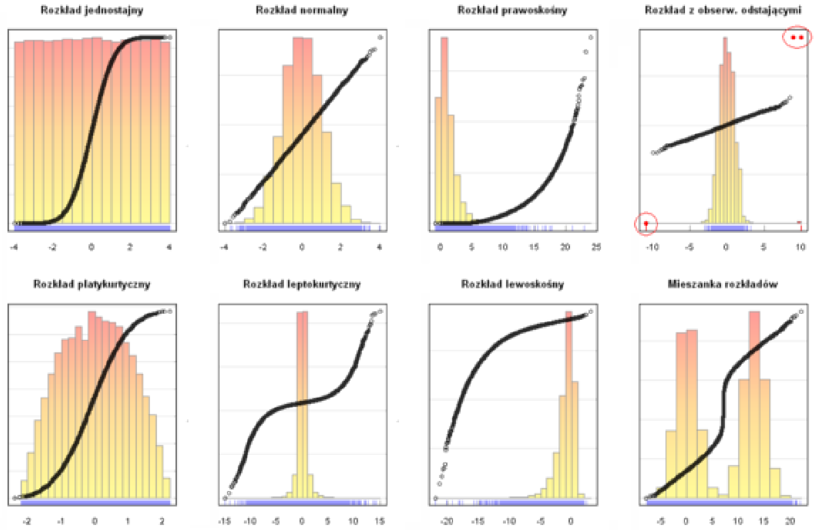


Histogram zmiennej X
Frequency
x
Wykres kwantyl-kwantyl zmiennej X
Sample Quantiles
Theoretical Quantiles
Diagram boxplot zmiennej X
Wykres rozrzutu zmiennej X
x



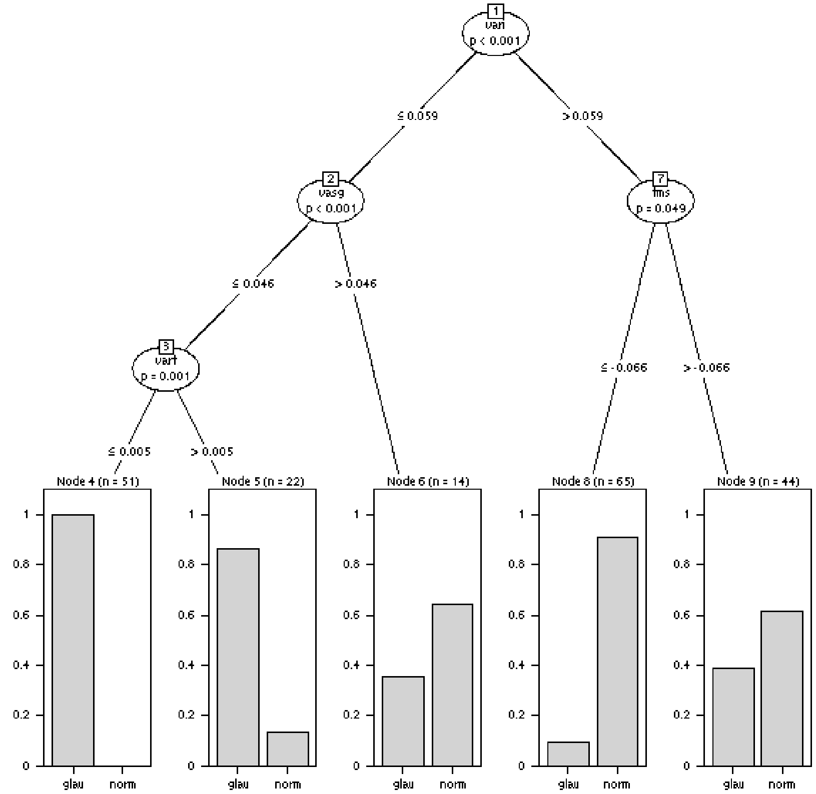
N K J I
F E D P
L G H C
B A O M

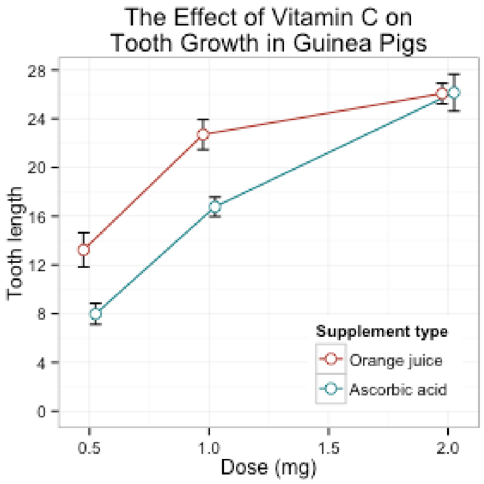


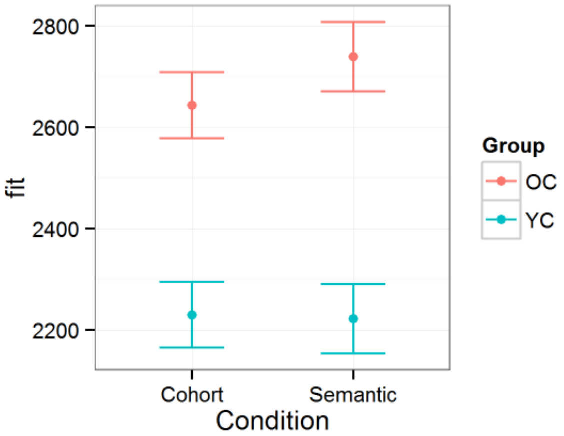


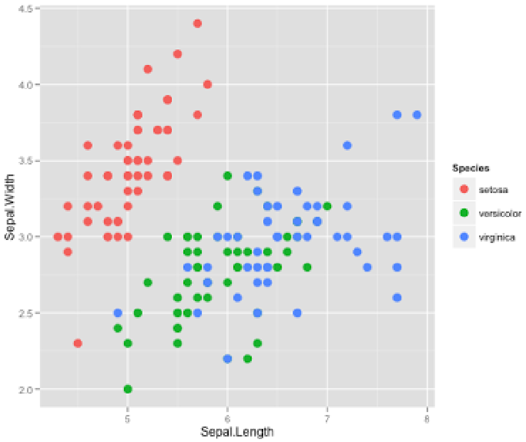
Rozkład jednostajny
Rozkład normalny
Rozkład prawoskośny
Rozkład z obserw. odstającymi
Rozkład platykurtyczny
Rozkład leptokurtyczny
Rozkład lewoskośny
Mieszanka rozkładów

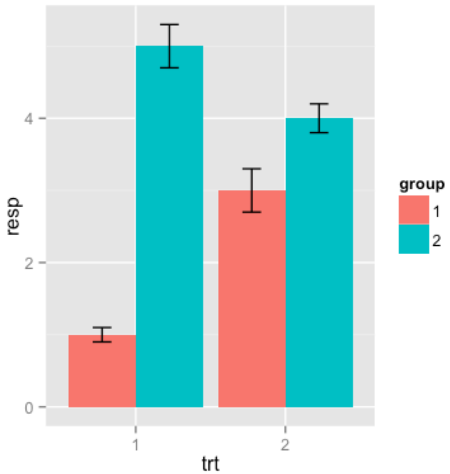
...and varied...

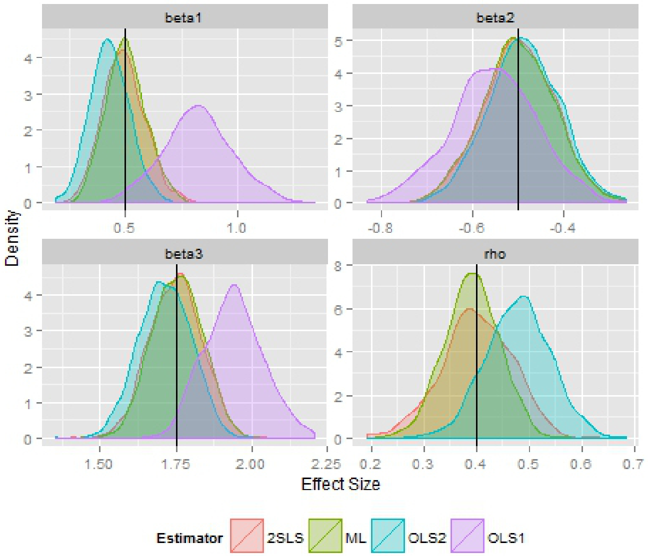

1
vari
p < 0.001
≤ 0.059
> 0.059
2
vasg
p < 0.001
7
tms
p = 0.049
≤ 0.046
> 0.046
3
vart
p = 0.001
≤ 0.005
> 0.005
≤ -0.066
> -0.066
Node 4 (n = 51)
Node 5 (n = 22)
Node 6 (n = 14)
Node 8 (n = 65)
Node 9 (n = 44)
1
0.8
0.6
0.4
0.2
0
glau
norm

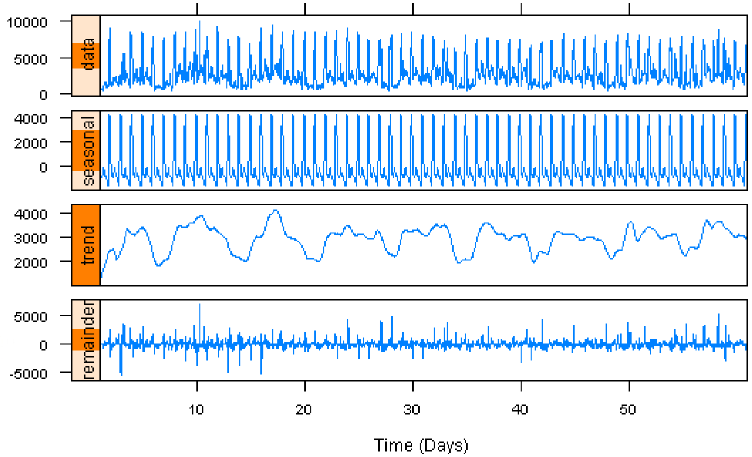

The Effect of Vitamin C on Tooth Growth in Guinea Pigs
Tooth length
Dose (mg)
0
4
8
12
16
20
24
28
0.5
1.0
1.5
2.0
Supplement type
Orange juice
Ascorbic acid

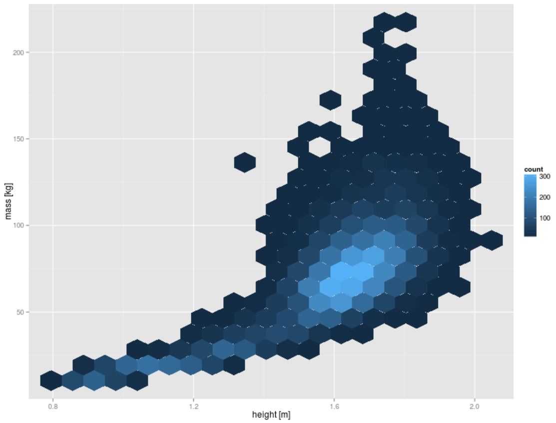

fit
2200
2400
2600
2800
Cohort
Semantic
Condition
Group
OC
YC


Sepal.Width
Sepal.Length
2.0
2.5
3.0
3.5
4.0
4.5
5
6
7
8
Species
setosa
versicolor
virginica

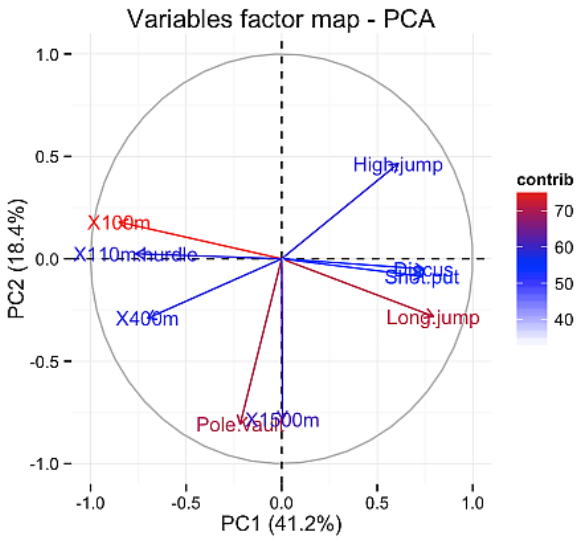

resp
trt
0
2
4
1
2
group
1
2


beta1
beta2
beta3
rho
Density
Effect Size
0.5
1.0
-0.8
-0.6
-0.4
1.50
1.75
2.00
2.25
0.2
0.3
0.4
0.5
0.6
0.7
Estimator
2SLS
ML
OLS2
OLS1

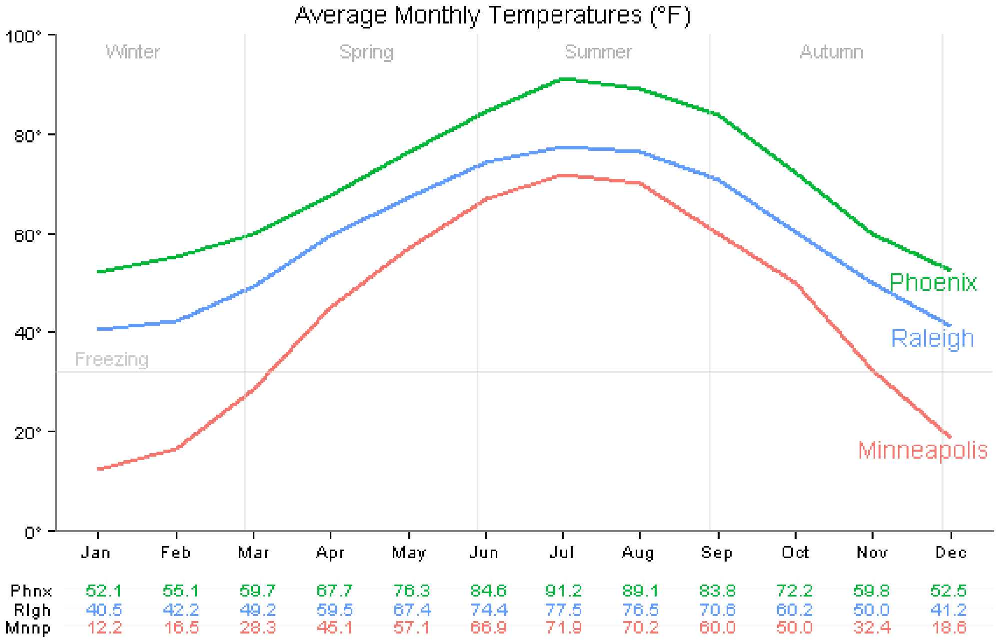

data
seasonal
trend
remainder
10000
5000
0
4000
2000
0
4000
3000
2000
5000
0
-5000
10
20
30
40
50
Time (Days)


mass [kg]
height [m]
count
300
200
100
50
100
150
200
0.8
1.2
1.6
2.0

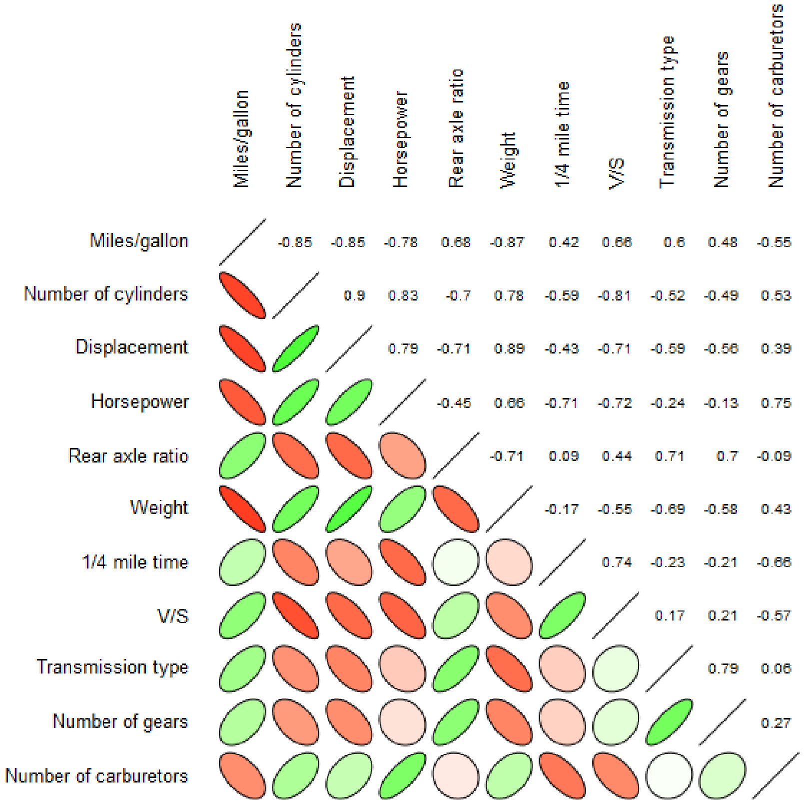
...and varied...



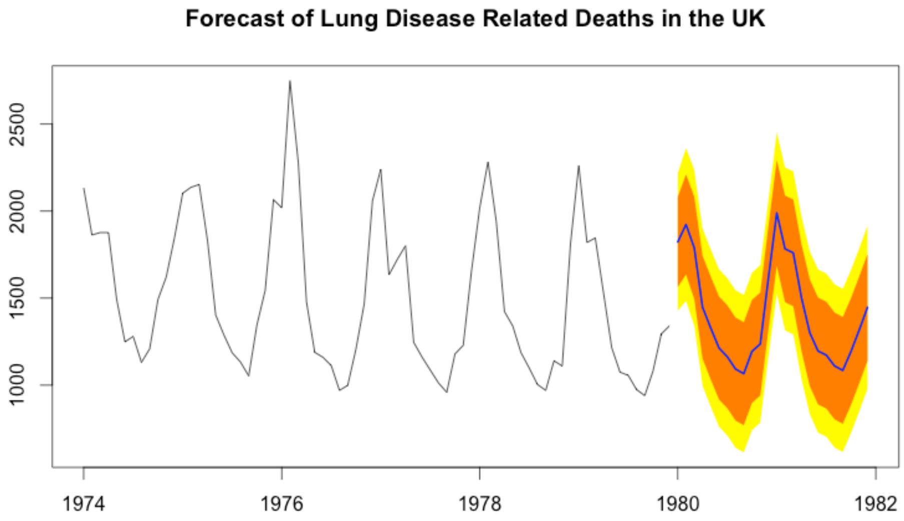
Variables factor map - PCA
1.0
0.5
0.0
-0.5
-1.0
PC2 (18.4%)
-1.0
-0.5
0.0
0.5
1.0
PC1 (41.2%)
High.jump
X100m
X110m.hurdle
Discus
Shot.put
X400m
Long.jump
Pole.vault
X1500m
contrib
70
60
50
40



Average Monthly Temperatures (°F)
100°
80°
60°
40°
20°
0°
Winter
Spring
Summer
Autumn
Freezing
Phoenix
Raleigh
Minneapolis
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Phnx 52.1 55.1 59.7 67.7 76.3 84.6 91.2 89.1 83.8 72.2 59.8 52.5
Rlgh 40.5 42.2 49.2 59.5 67.4 74.4 77.5 76.5 70.6 60.2 50.0 41.2
Mnnp 12.2 16.5 28.3 45.1 57.1 66.9 71.9 70.2 60.0 50.0 32.4 18.6



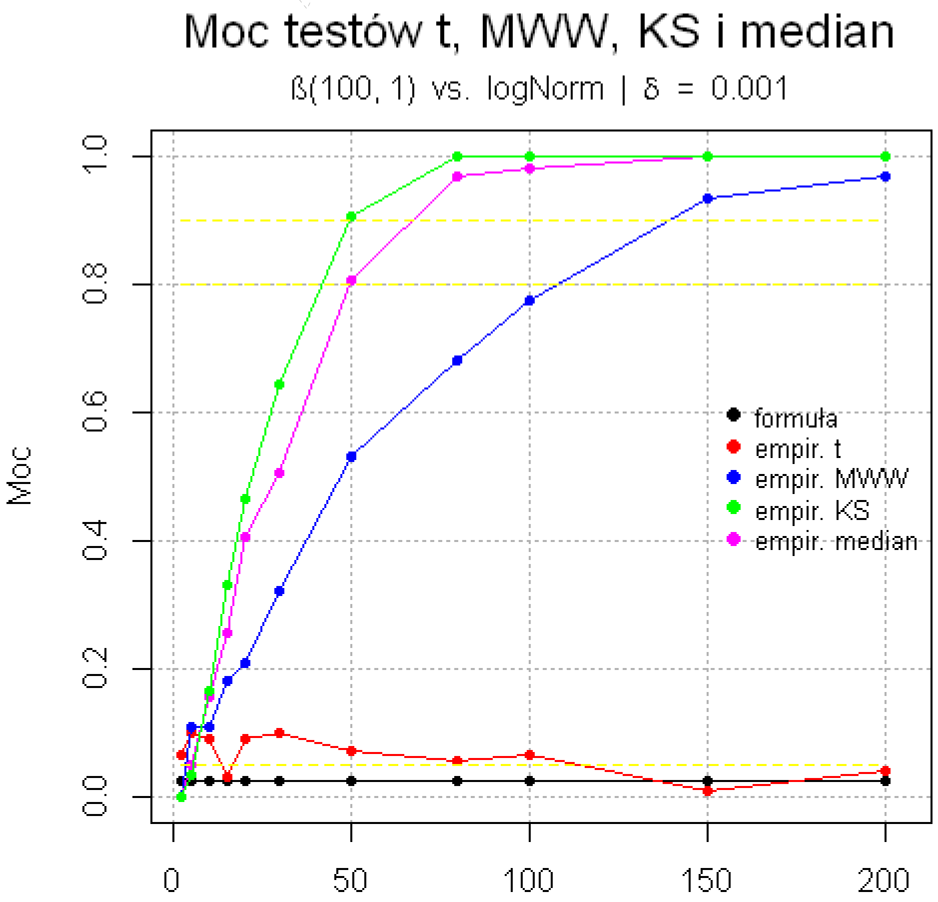
Miles/gallon
Number of cylinders
Displacement
Horsepower
Rear axle ratio
Weight
1/4 mile time
V/S
Transmission type
Number of gears
Number of carburetors
Miles/gallon -0.85 -0.85 -0.78 0.68 -0.87 0.42 0.66 0.6 0.48 -0.55
Number of cylinders 0.9 0.83 -0.7 0.78 -0.59 -0.81 -0.52 -0.49 0.53
Displacement 0.79 -0.71 0.89 -0.43 -0.71 -0.59 -0.56 0.39
Horsepower -0.45 0.66 -0.71 -0.72 -0.24 -0.13 0.75
Rear axle ratio -0.71 0.09 0.44 0.71 0.7 -0.09
Weight -0.17 -0.55 -0.69 -0.58 0.43
1/4 mile time 0.74 -0.23 -0.21 -0.66
V/S 0.17 0.21 -0.57
Transmission type 0.79 0.06
Number of gears 0.27
Number of carburetors



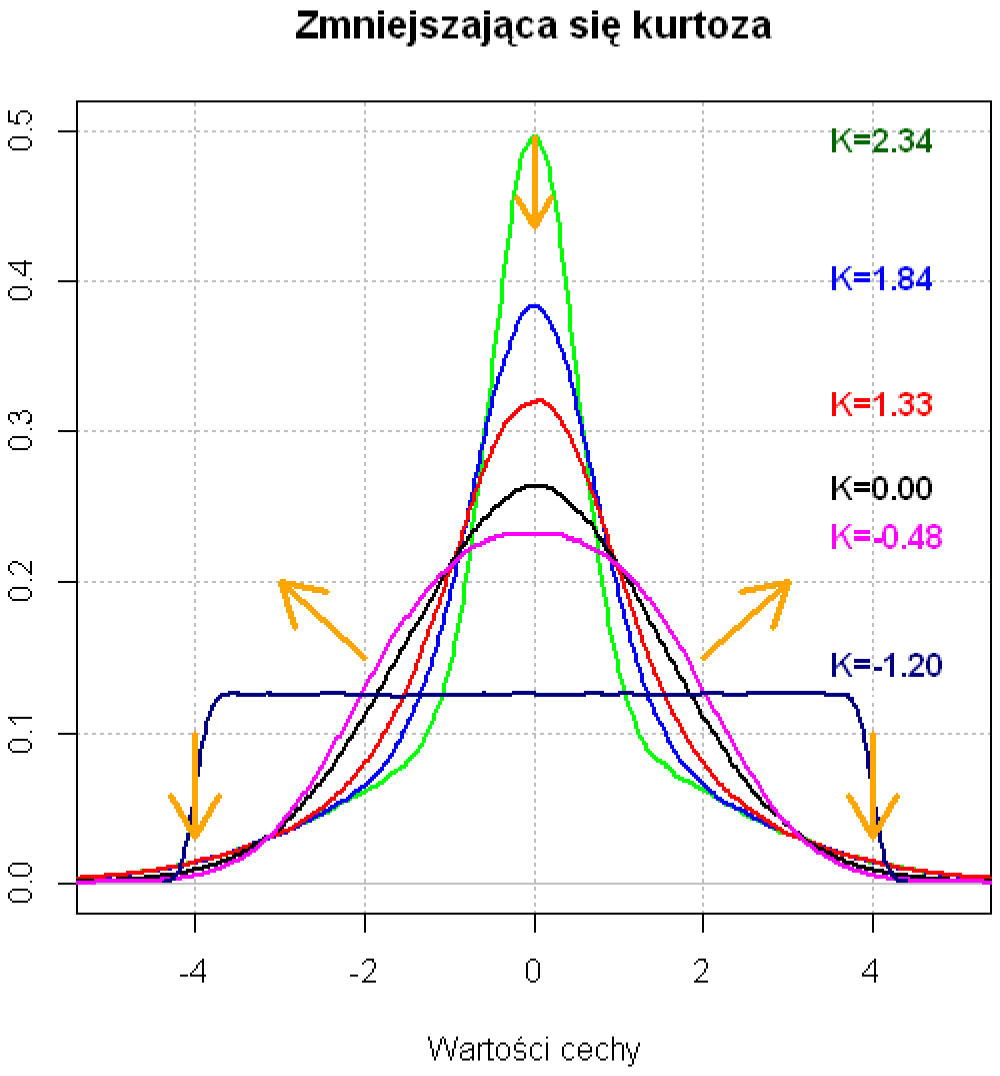
Forecast of Lung Disease Related Deaths in the UK
2500
2000
1500
1000
1974
1976
1978
1980
1982

...animated...

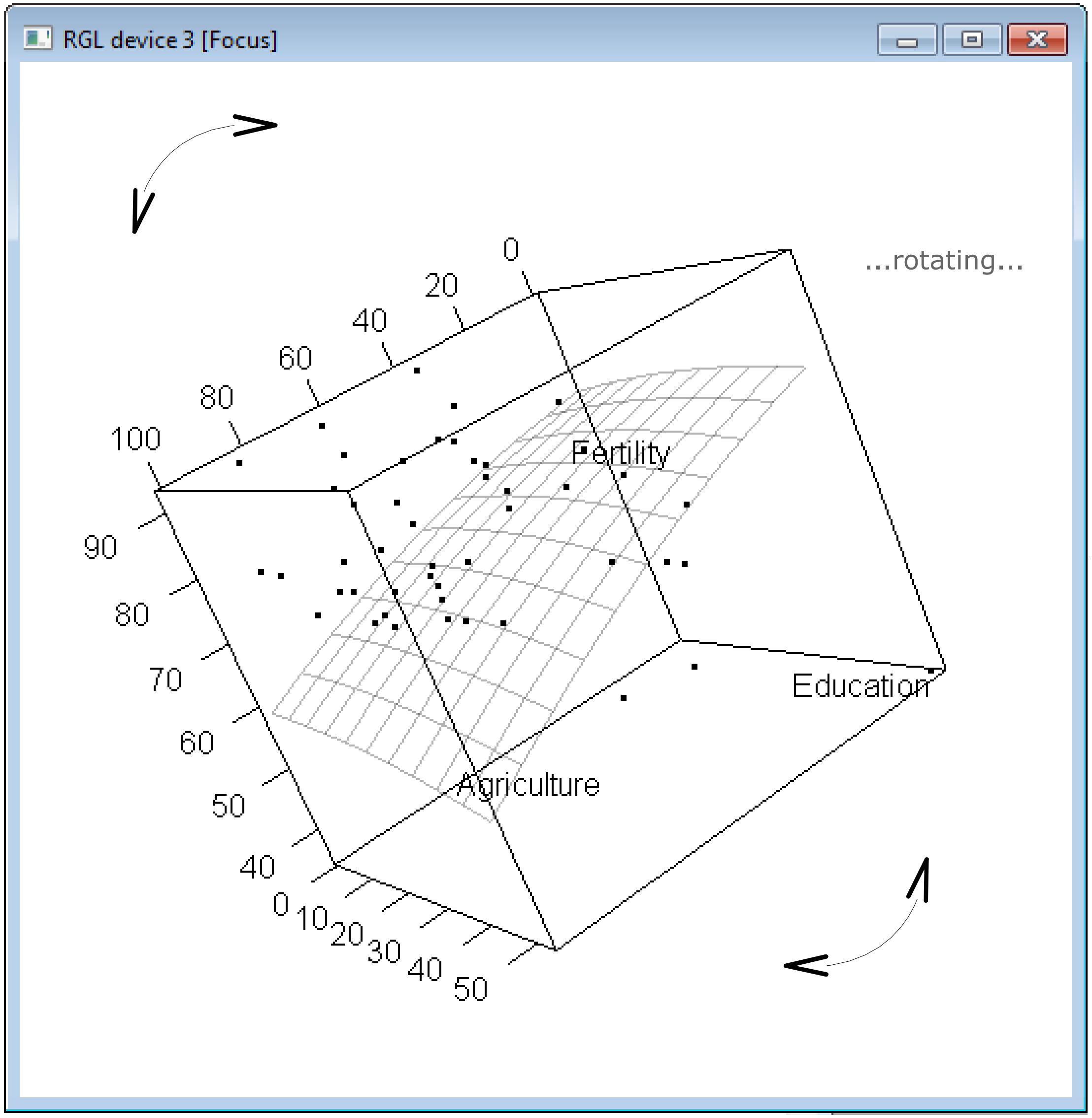
Power of tests



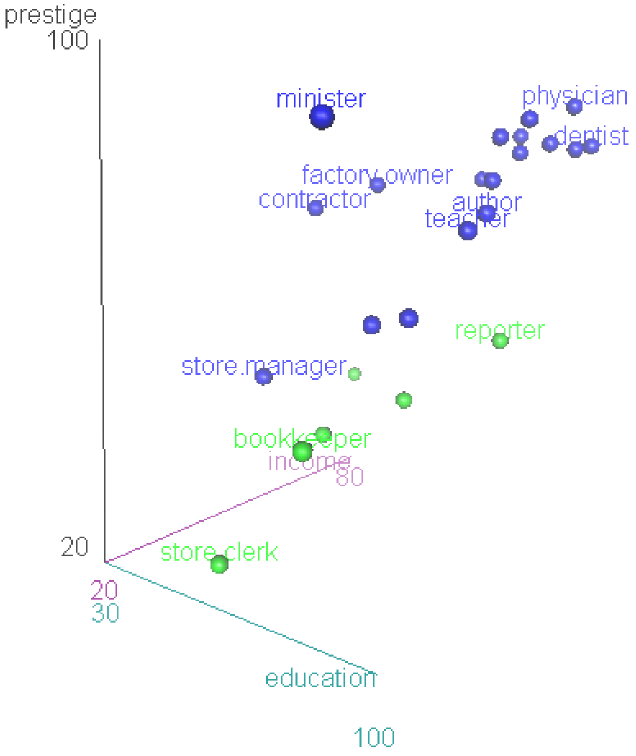
Decreasing kurtosis

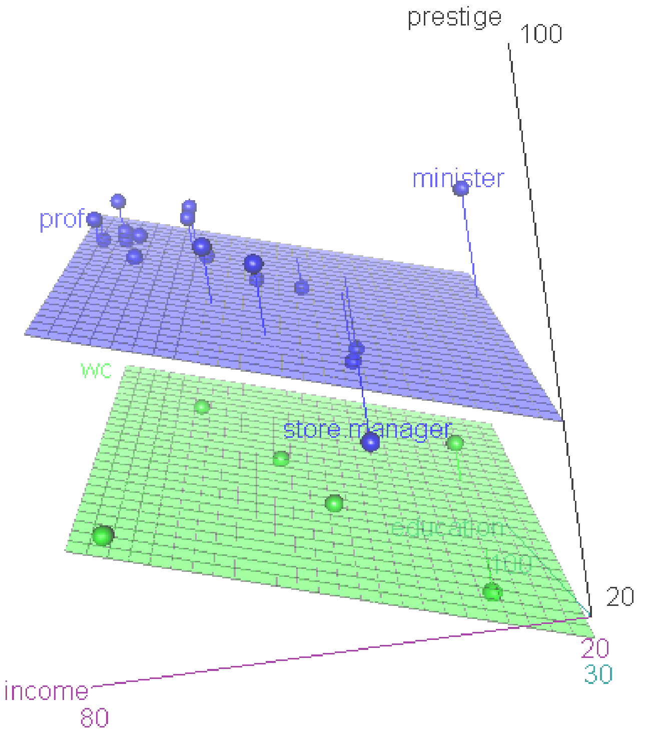


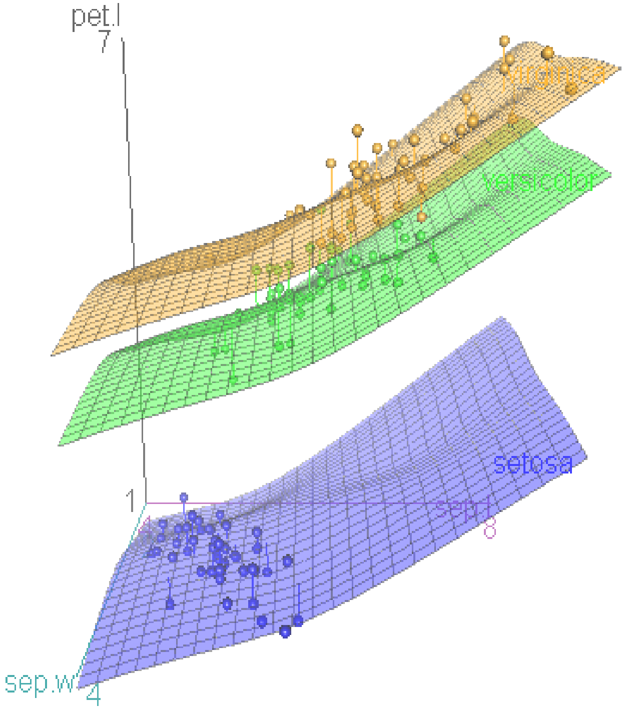
...and 3D, interactive and animated (rgl)



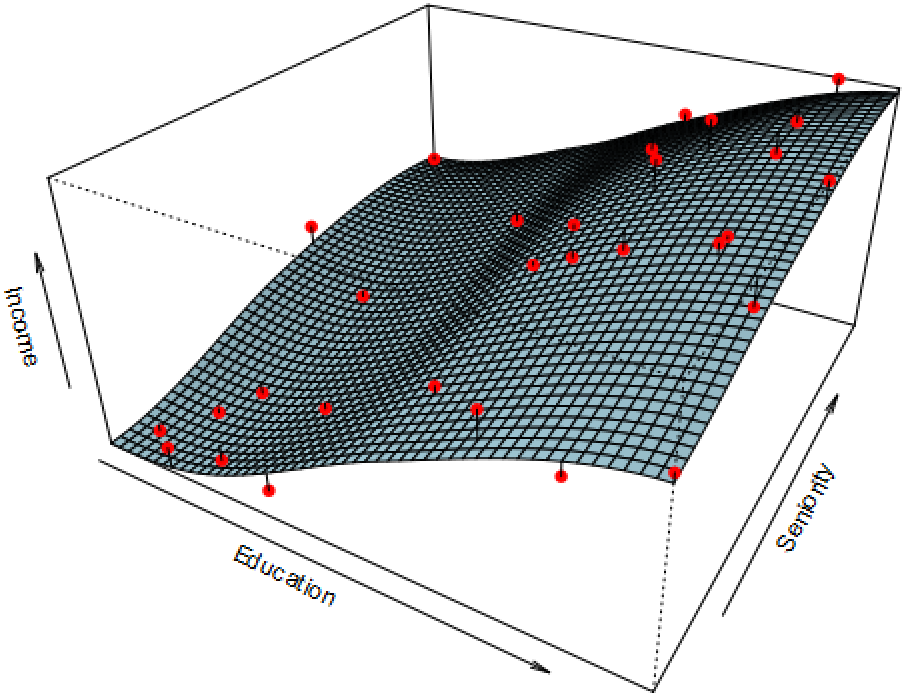
...and 3D, interactive and animated (rgl)

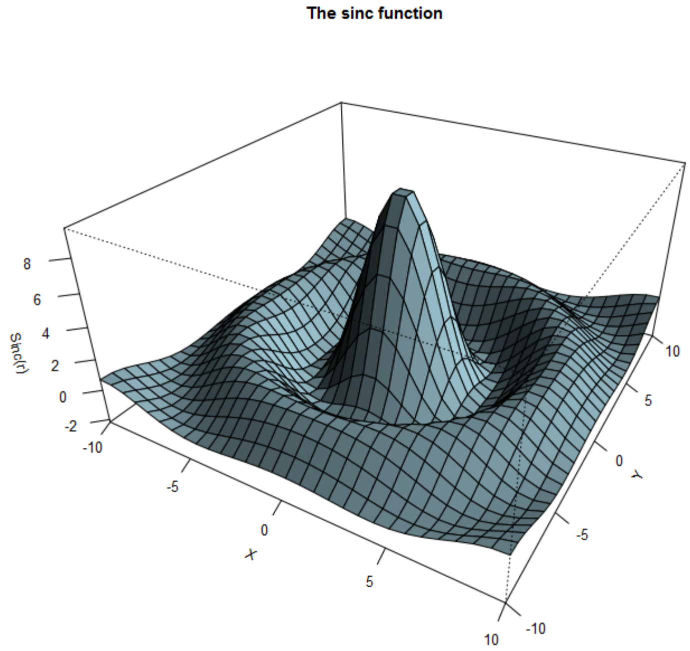


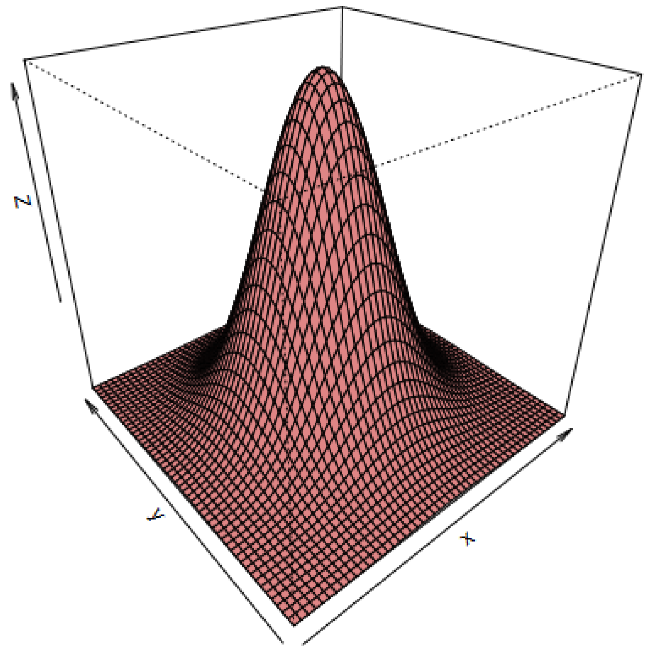


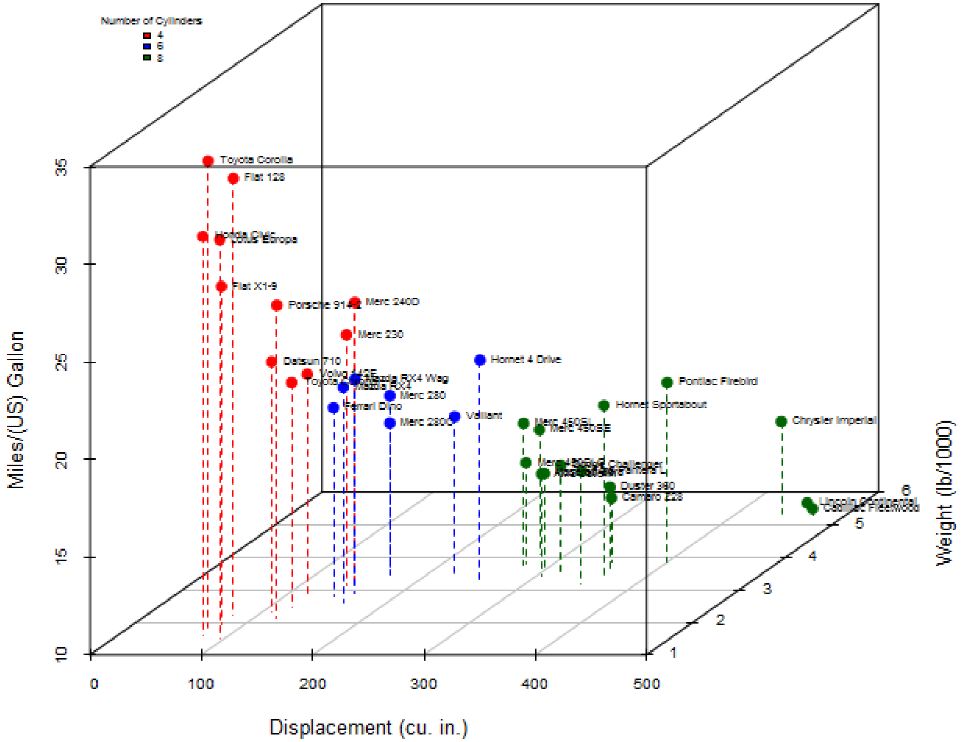


…and 3D interactive (scatterplot3d)

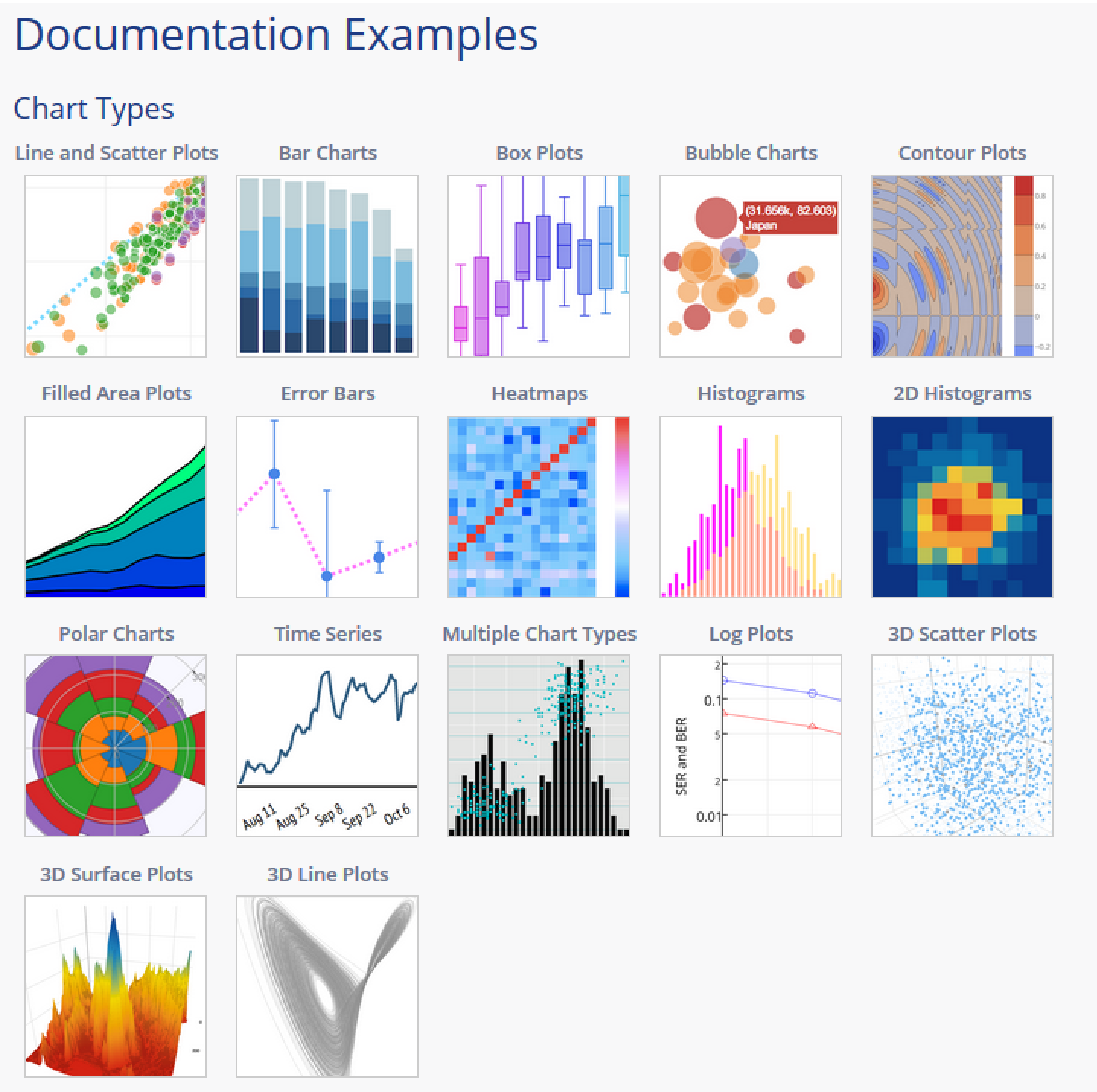








...and interactive (JScript)

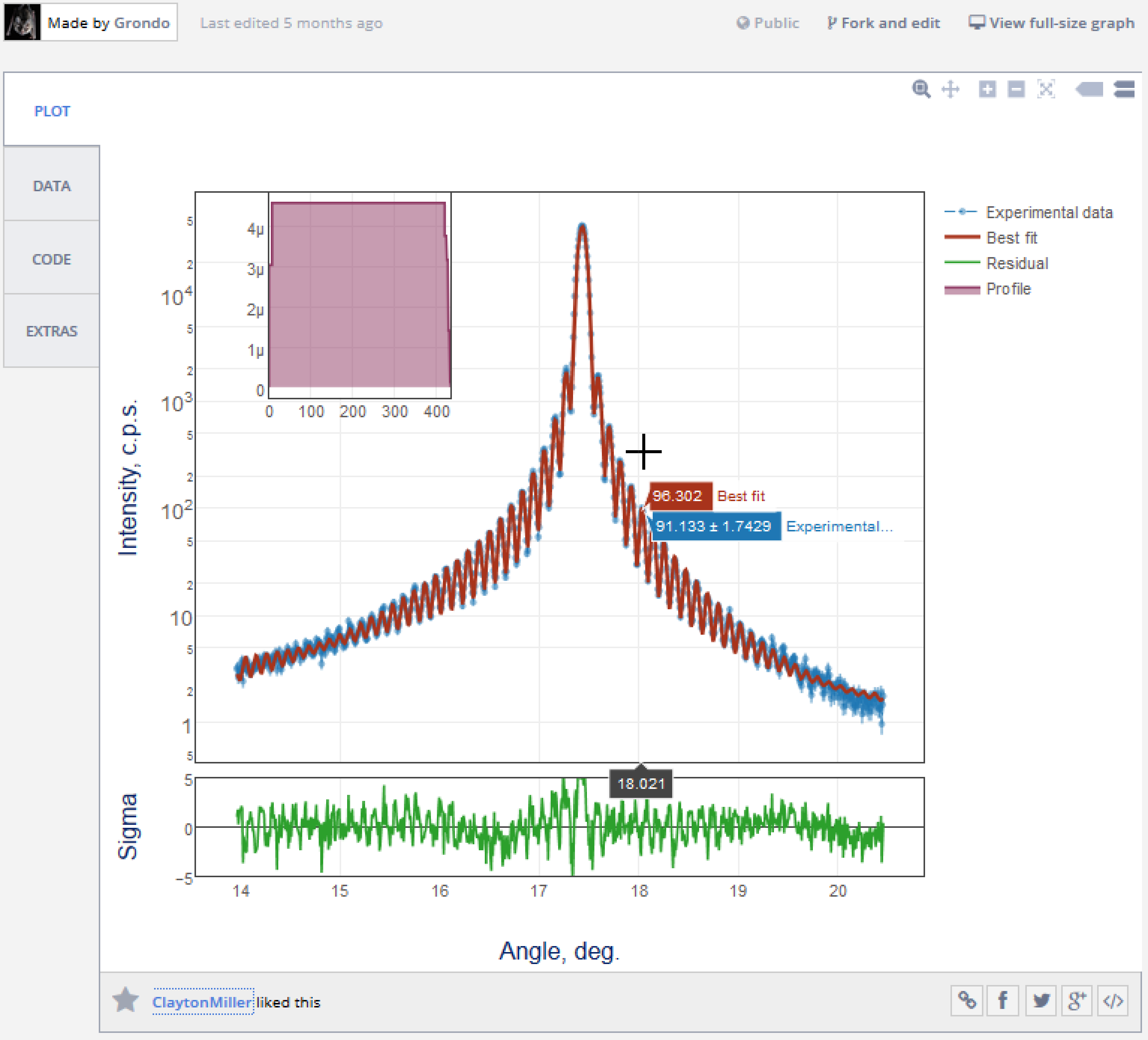


Documentation Examples

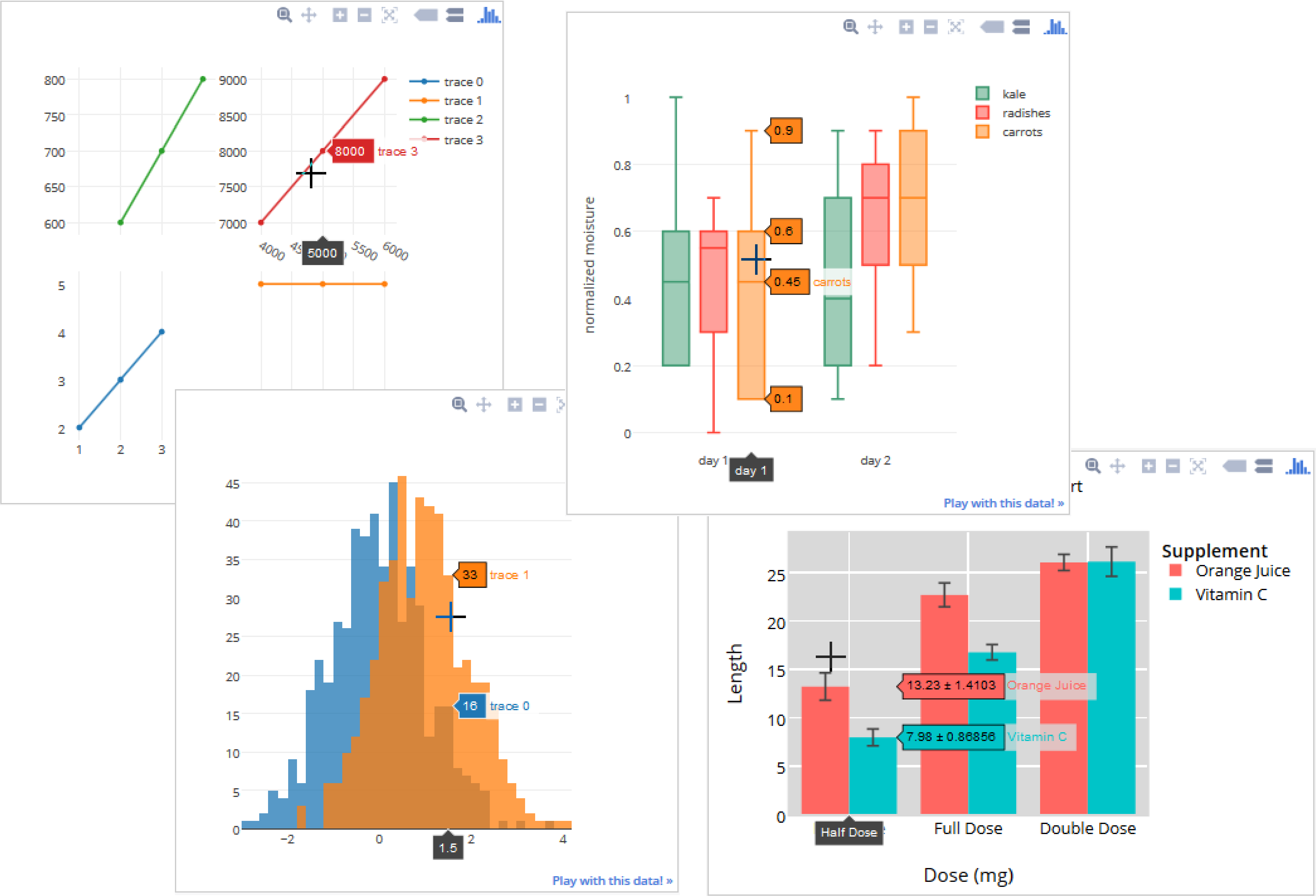
Chart Types

Line and Scatter Plots
Bar Charts
Box Plots
Bubble Charts
Contour Plots
Filled Area Plots
Error Bars
Heatmaps
Histograms
2D Histograms
Polar Charts
Time Series
Multiple Chart Types
Log Plots
3D Scatter Plots
3D Surface Plots
3D Line Plots

...and interactive (JScript)

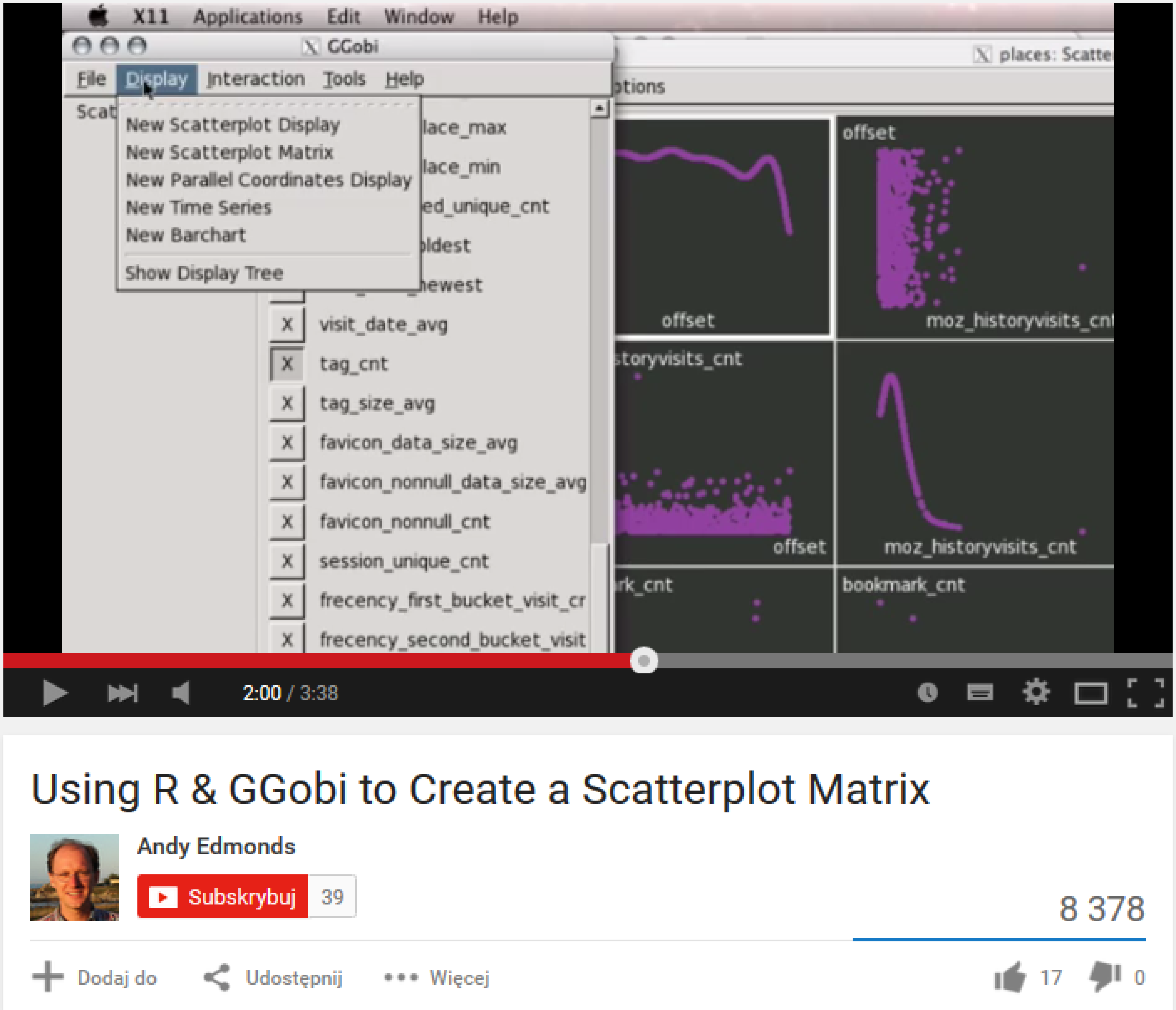


...and interactive (JScript)



R &GGobi in action

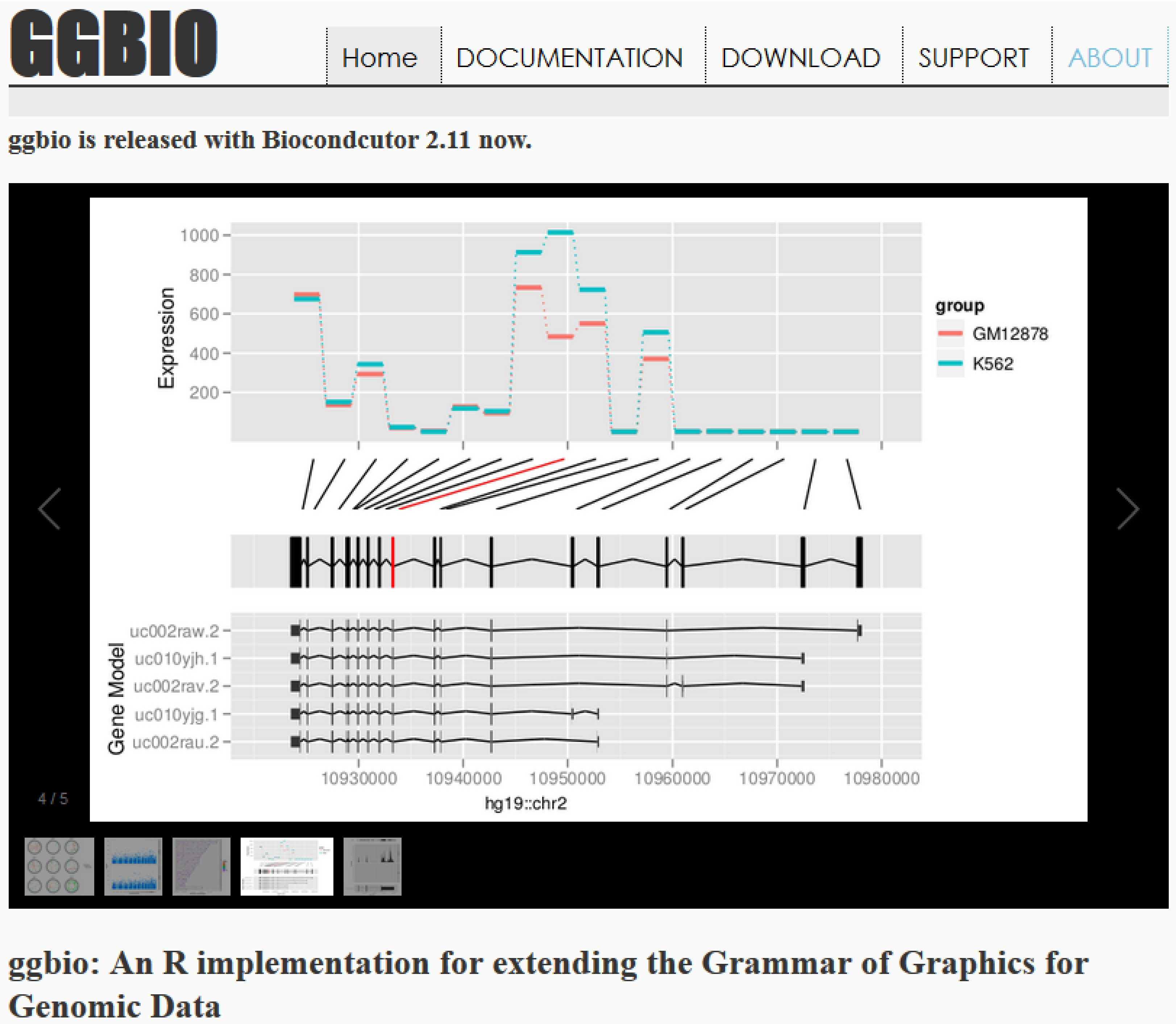
[YouTube: Using R & GGobi to Create a Scatterplot Matrix]



Using R & GGobi to Create a Scatterplot Matrix

Andy Edmonds

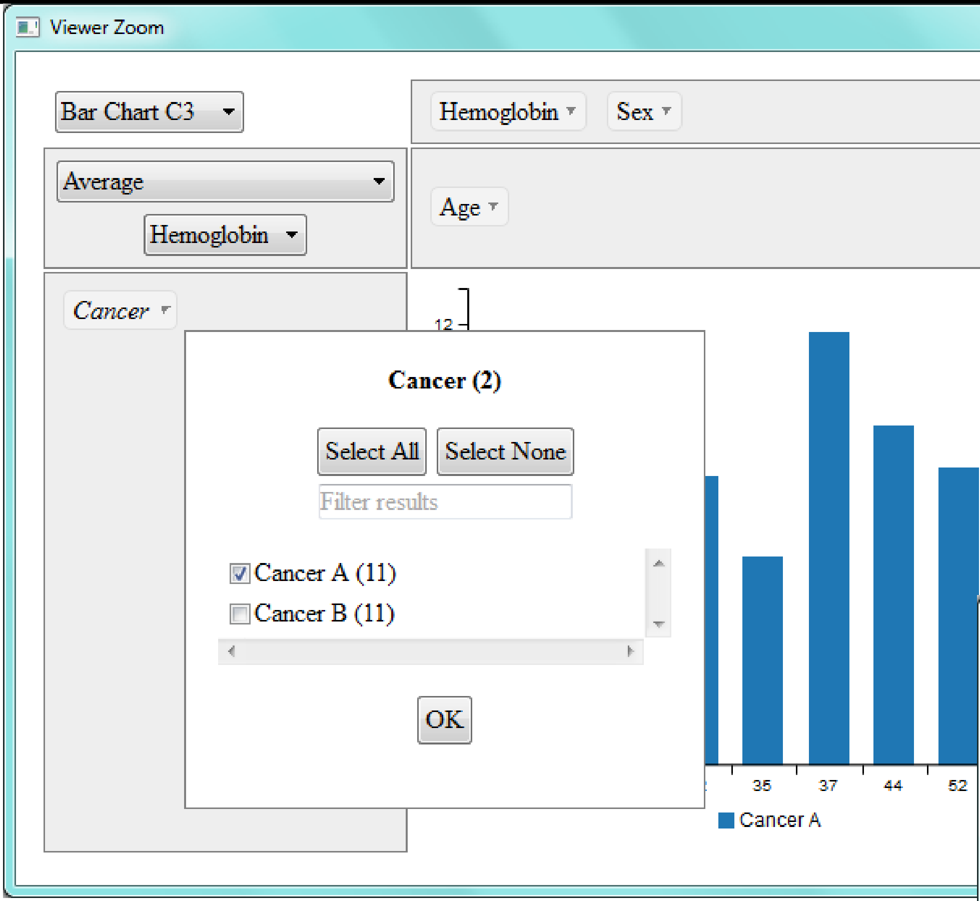
GGBio for geneticists



ggbio is released with Biocondcutor 2.11 now.

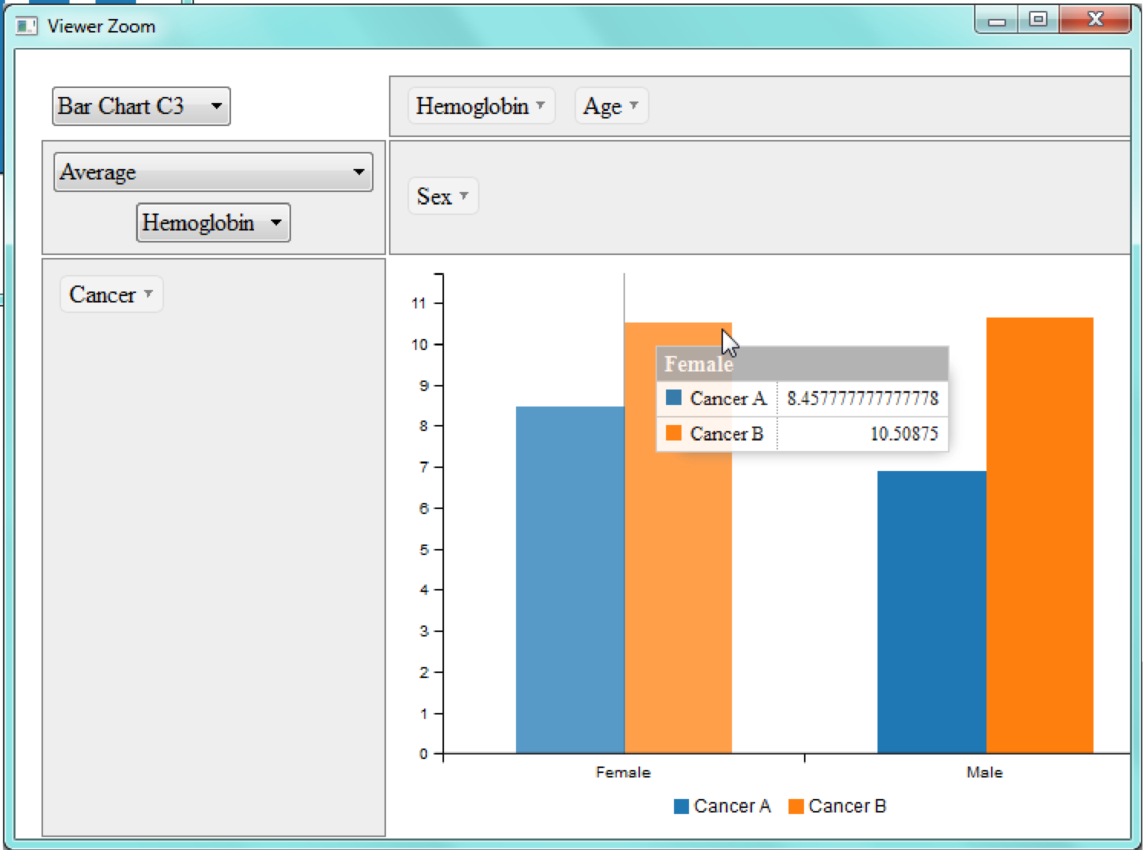
ggbio: An R implementation for extending the Grammar of Graphics for Genomic Data

rpivotTable – inteRactive pivot tables



Based on PivotTable.js by
[Nicolas Kruchten](http://nicolas.kruchten.com/pivottable)

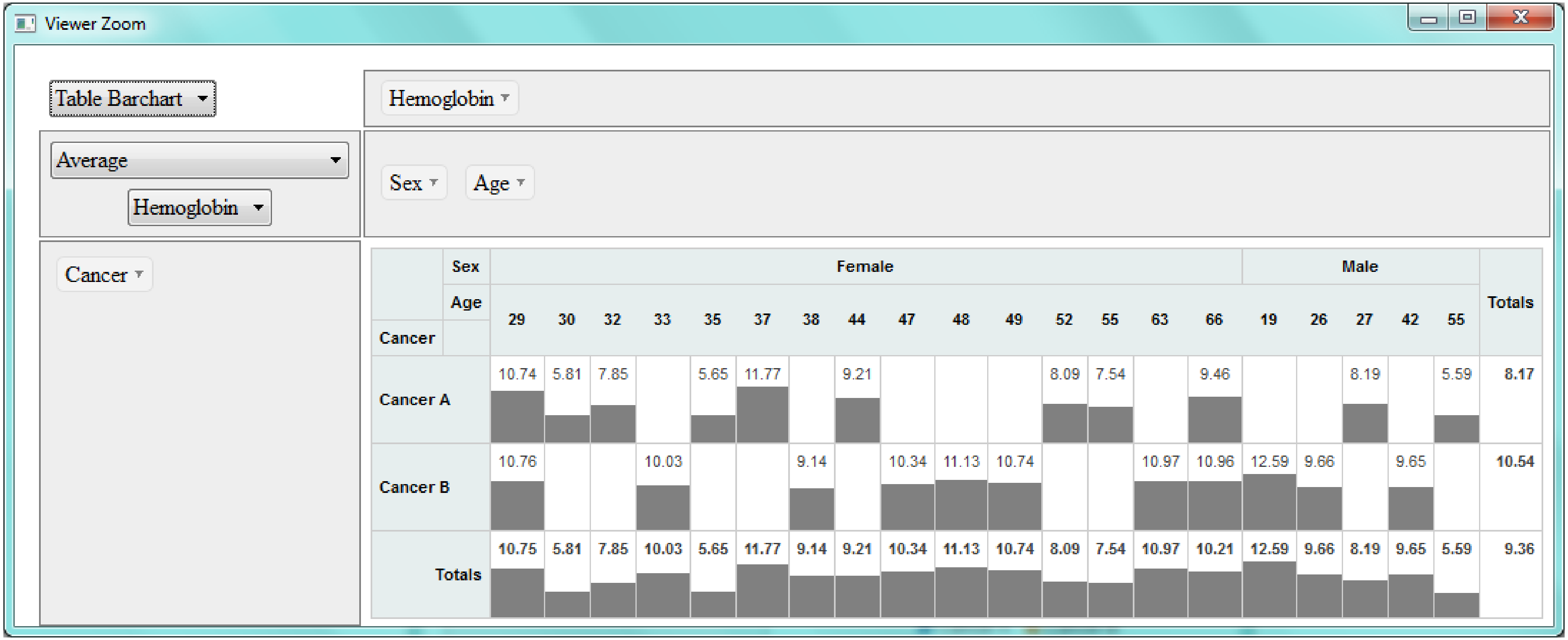
http://nicolas.kruchten.com/pivottable



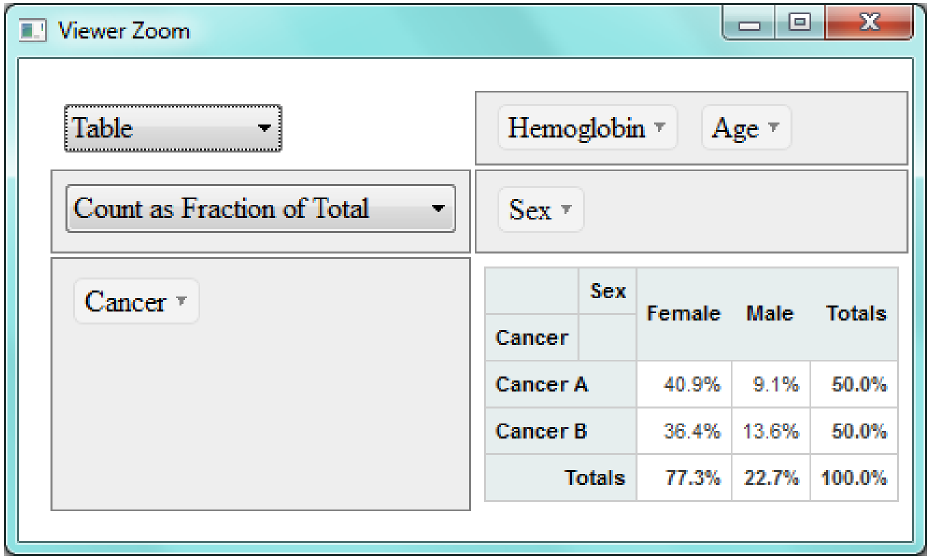
Female	
Cancer A	8.457777777777778
Cancer B	10.50875

http://www.magesblog.com/2015...

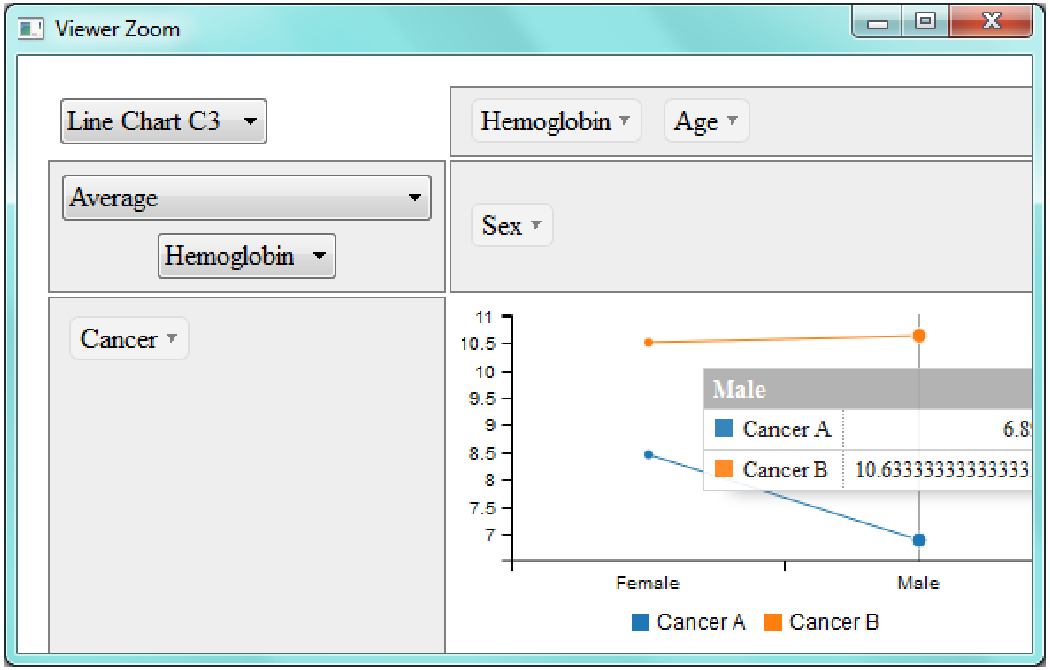
rpivotTable



Sex		Female															Male					
Cancer	Age	29	30	32	33	35	37	38	44	47	48	49	52	55	63	66	19	26	27	42	55	Totals
Cancer A		10.74	5.81	7.85		5.65	11.77		9.21				8.09	7.54		9.46			8.19		5.59	8.17
Cancer B		10.76			10.03			9.14		10.34	11.13	10.74			10.97	10.96	12.59	9.66		9.65		10.54
Totals		10.75	5.81	7.85	10.03	5.65	11.77	9.14	9.21	10.34	11.13	10.74	8.09	7.54	10.97	10.21	12.59	9.66	8.19	9.65	5.59	9.36



Sex	Female	Male	Totals
Cancer			
Cancer A	40.9%	9.1%	50.0%
Cancer B	36.4%	13.6%	50.0%
Totals	77.3%	22.7%	100.0%



Charting subsystems

Base, default library

by Ross Ihaka
University of Auckland

- low-level – graph. primitives
- easy to learn*
- most powerful – no limits
- well readable, ascetic
- may involve a lot of coding to get fancy results
- supports multiple plots
- interactive locator of points
- No anti-aliasing but it can draw on Cairo devices

Trellis

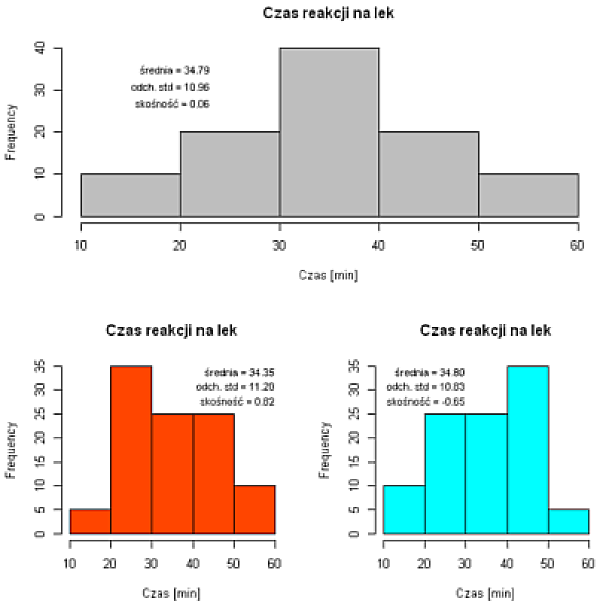
by Deepayan Sarka
University of Wisconsin

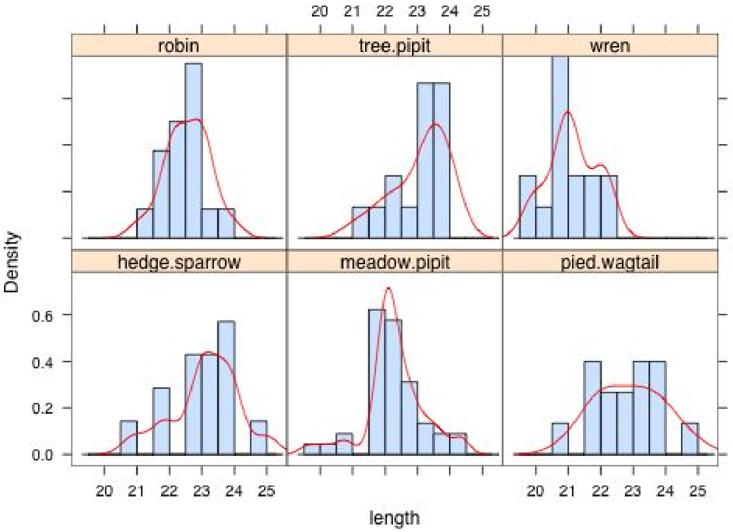
- designed for viewing multivariable datasets (grid of conditioned plots)
- well readable
- closed set of available diagrams: Barplot, Dotplot, Box and Whiskers, Histogram, Density, QQ, Scatterplot
- incompatible with other systems

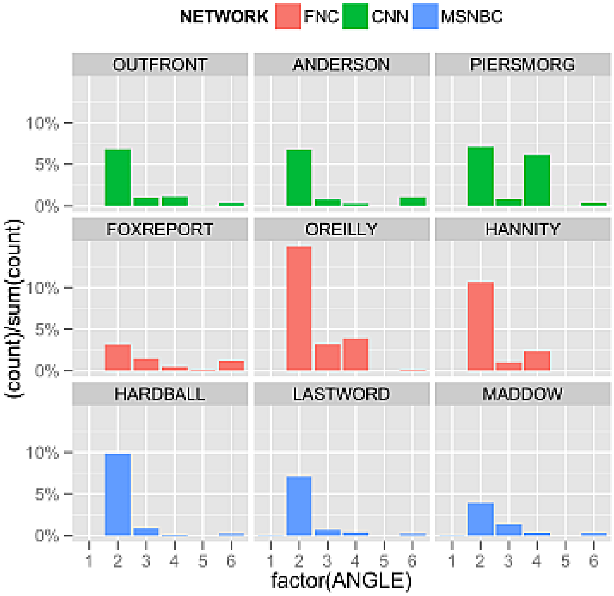
ggplot2

by Hadley Wickham
Rice University

- high-level and well organized
- implementation of *Grammar of Graphics*
- powerful, higly customizable
- well readable, polished output
- anti-aliased by design
- closed set of diagrams but easily expandable
- supports multiple plots (grid)
- incompatible with others







Charting subsystems

plotly
Teamwork

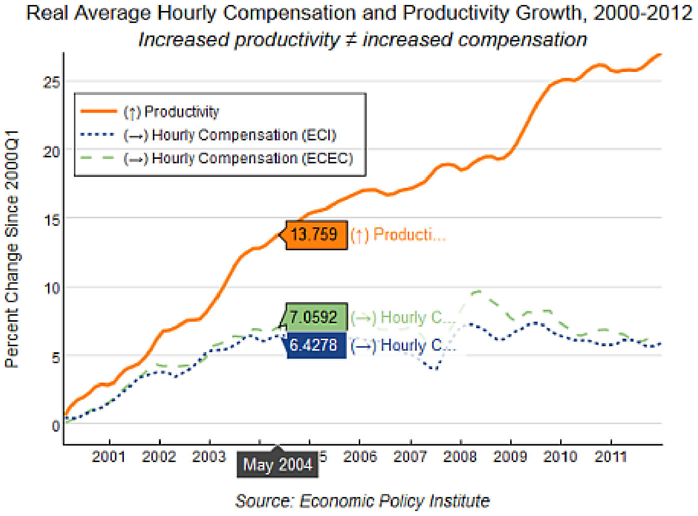
- based on ggplot2
- interactive (JScript)
- WWW enabled (HTML/JSON/JS)
- breathtaking output –must see!
- well readable, clean
- rich library of examples
- incompatible with other systems

ggvis
by RStudio team

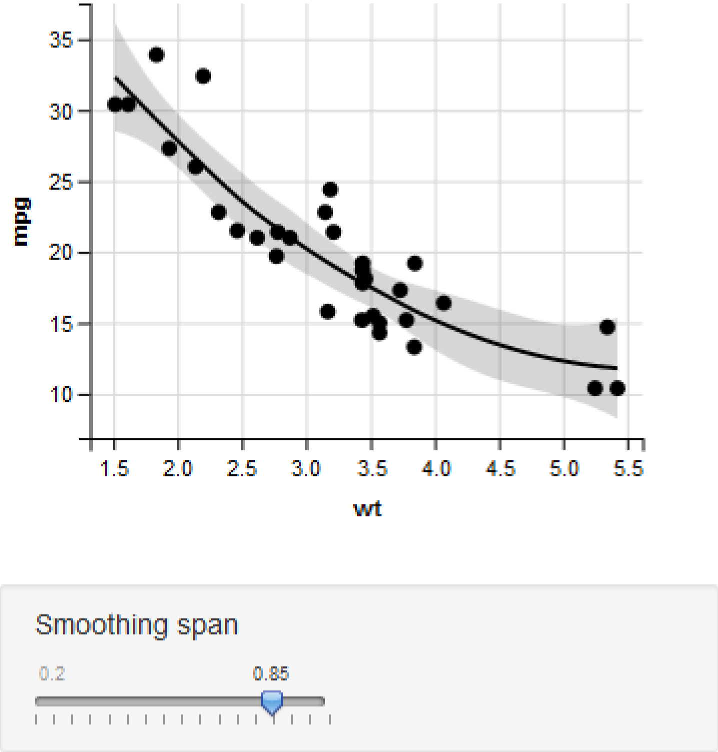
- another implementation of *Grammar of graphics*, similar in spirit to ggplot2
- Interactive (JScript)
- WWW enabled (HTML, JS)
- well readable
- designed for Shiny

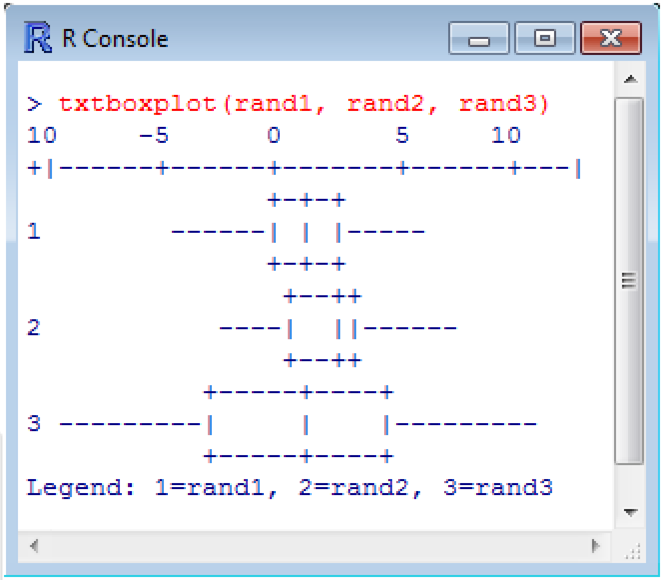
txtplot
by Bjoern Bornkamp

- produces graphs in pure ASCII
- rudimentary output
- closed set of plots (boxplot, lineplot, barplot, density, ACF)
- really useful when resources are limited (mobile devices, simple LCD displays, etc.) or output must be textual
- incompatible with others


Real Average Hourly Compensation and Productivity Growth, 2000-2012
Increased productivity ≠ increased compensation

Source: Economic Policy Institute

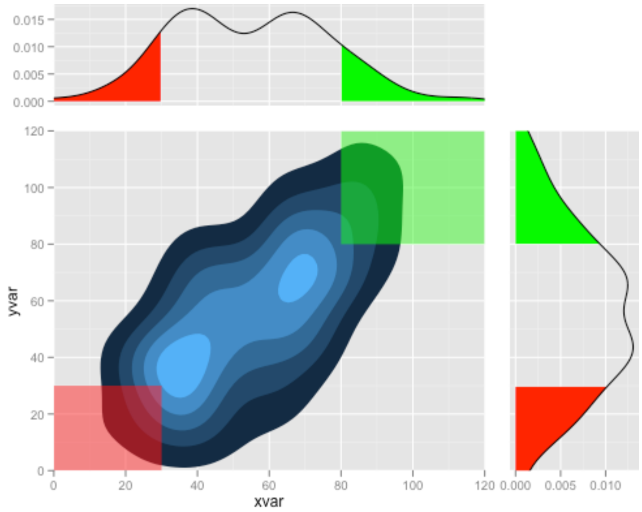


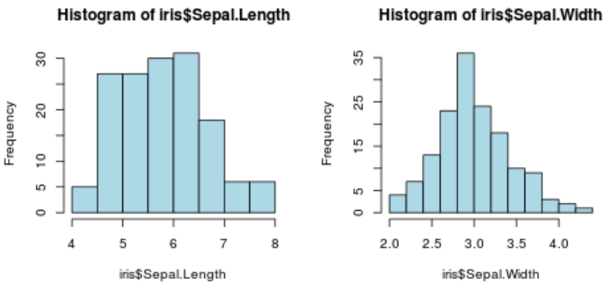


```
> txtboxplot(rand1, rand2, rand3)
10      -5       0       5      10
+|------+-------+--------+-------+---|
                +-+-+
1        -------| | |-----
                +-+-+
                 +--++
2            ----|  ||------
                 +--++
            +-----+----+
3 ----------|     |    |----------
            +-----+----+
Legend: 1=rand1, 2=rand2, 3=rand3
```

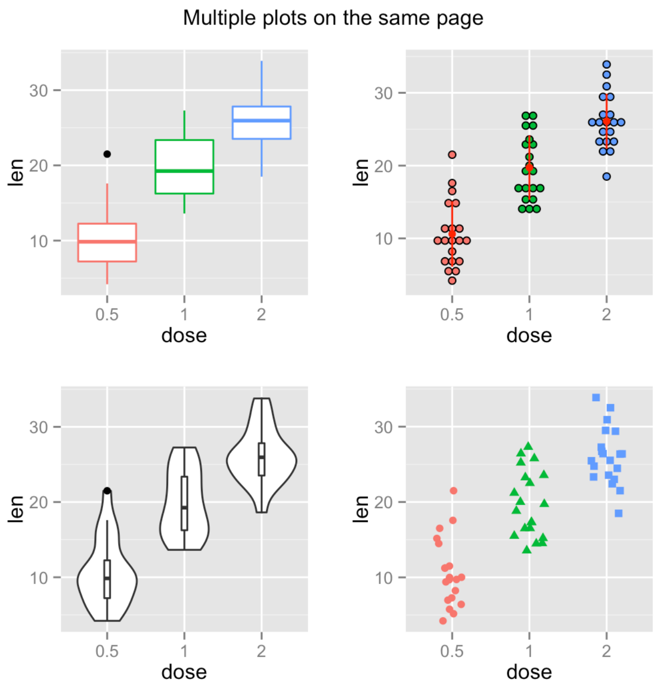
Additional helpers for ggplot2: gridExtra

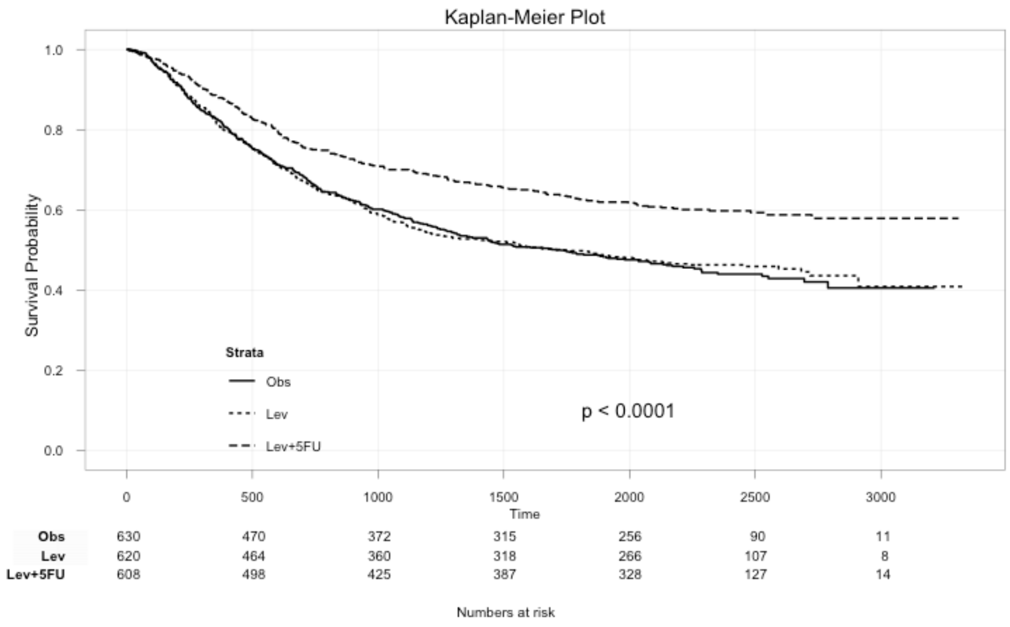
Helps to arrange multiple ggplot2 objects on the same page

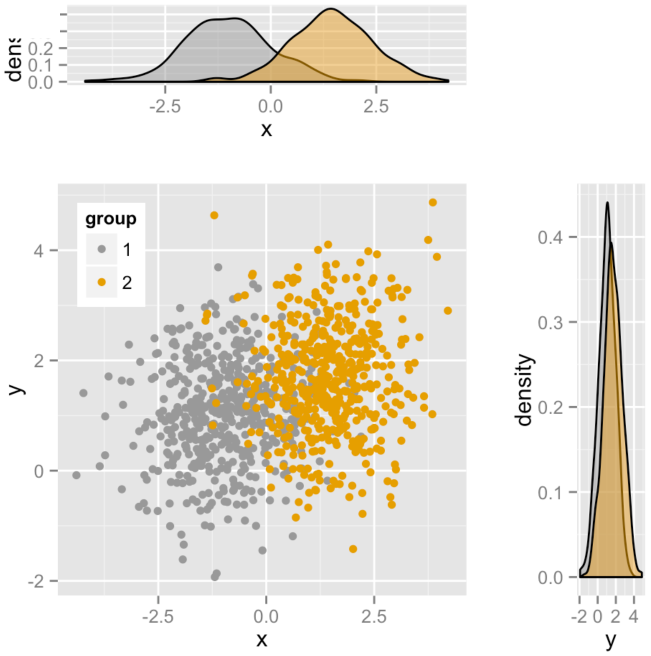




	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa

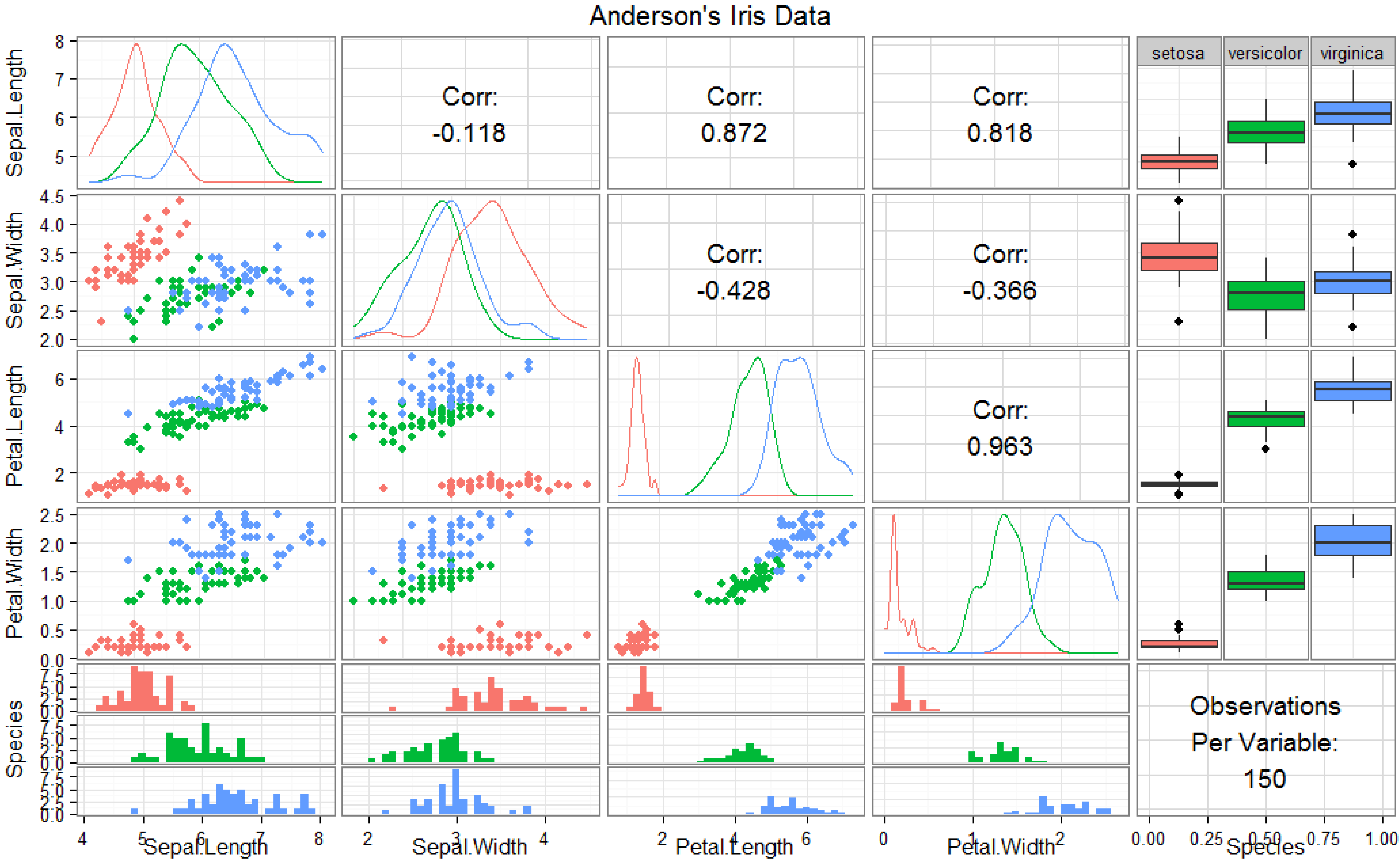






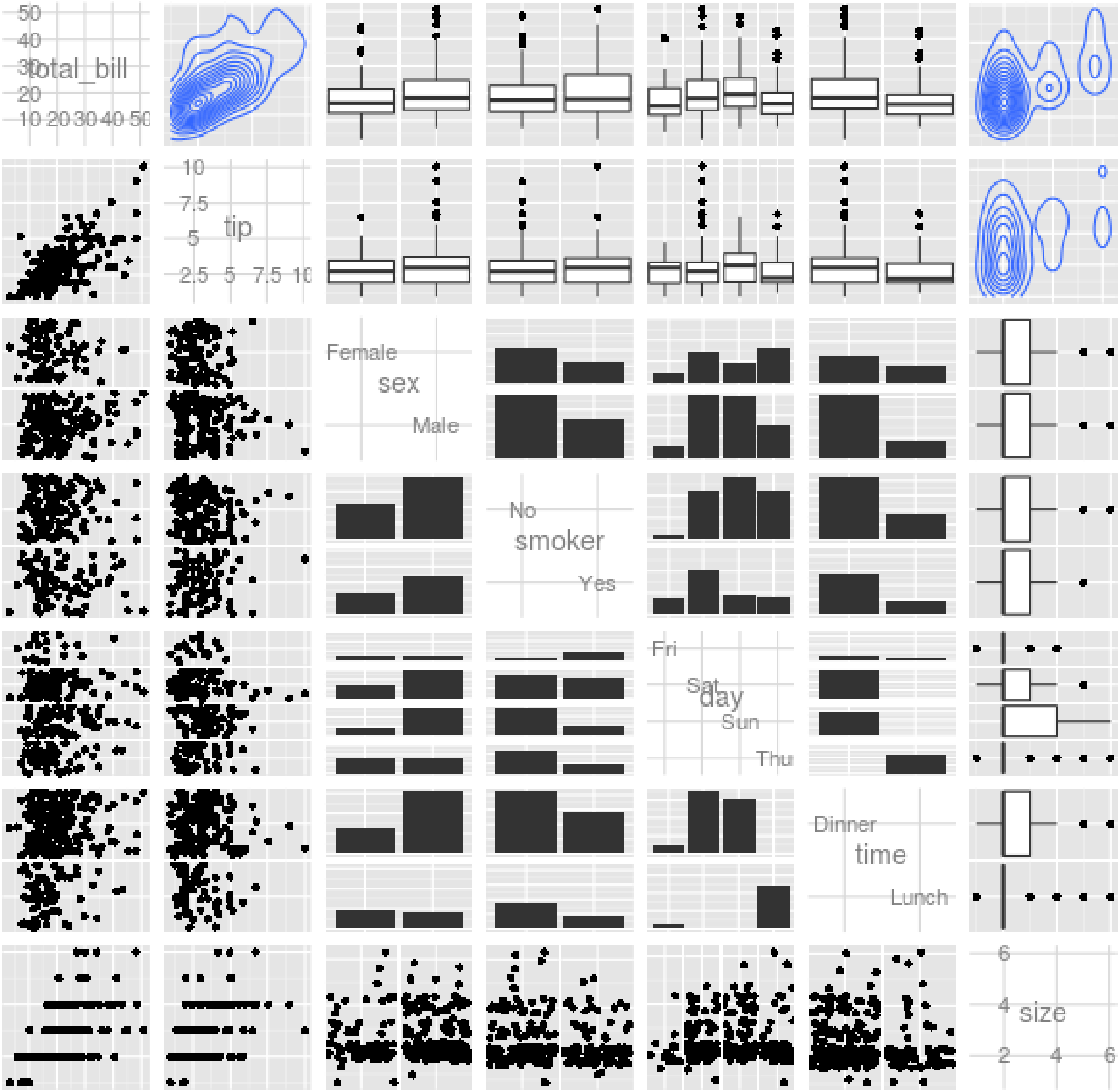
Additional helpers for ggplot2: GGally

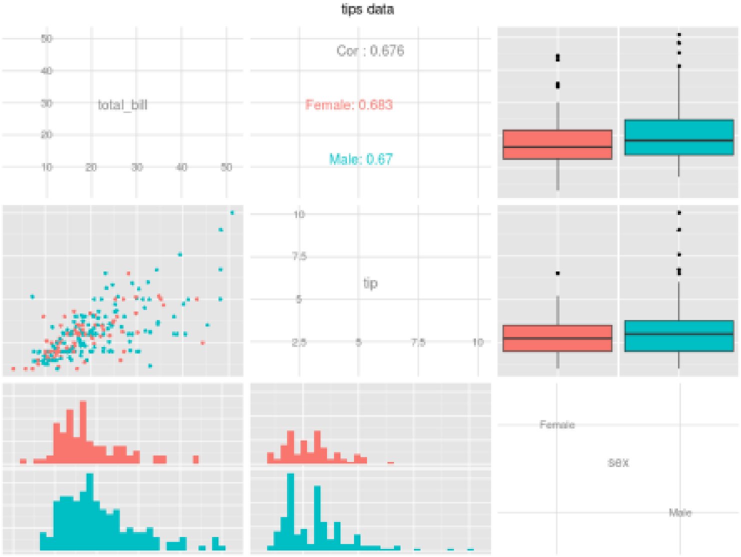
Creates a matrix of ggplot2 graphs for data exploration purposes.

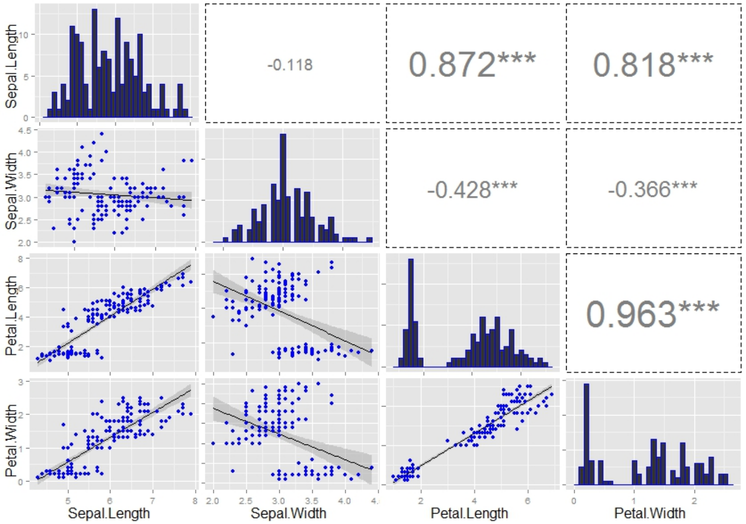


Additional helpers for ggplot2: GGally

Creates a matrix of ggplot2 graphs for data exploration purposes.

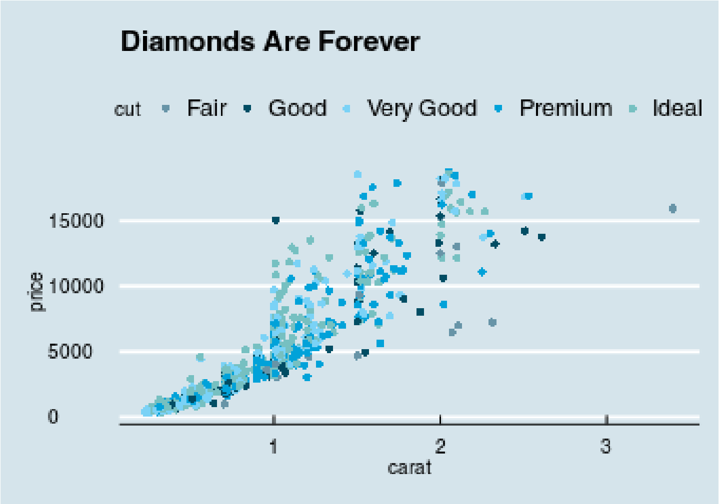




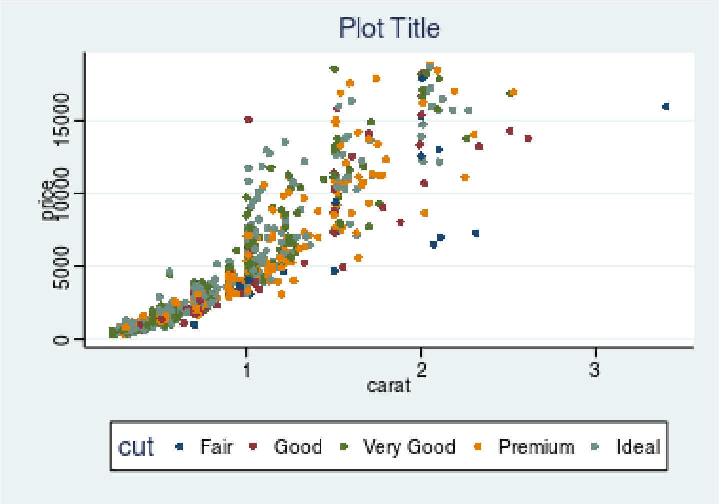


Additional helpers for ggplot2: ggthemes

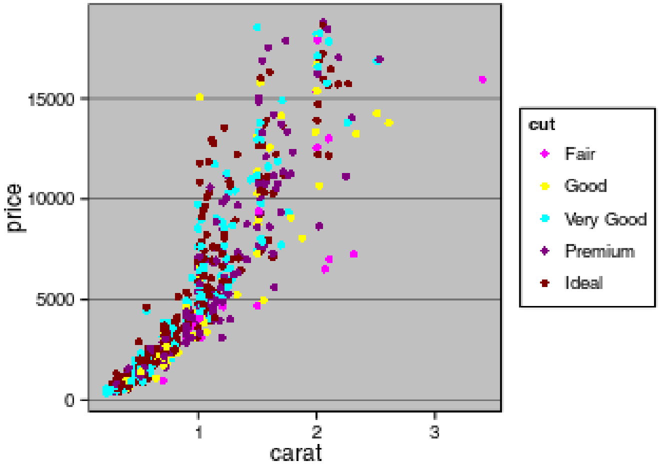
A set of themes (>15) for ggplot2: http://github.com/jrnold/ggthemes



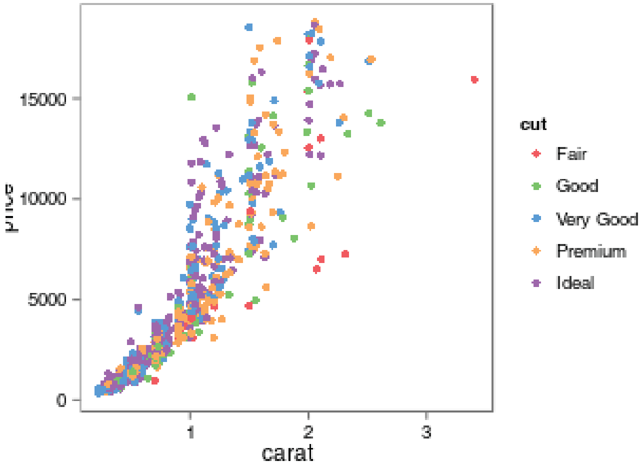
The Economist magazine theme



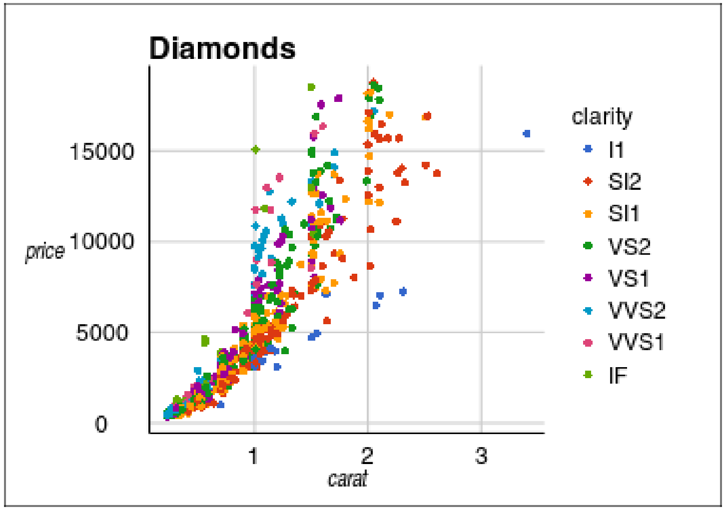
Stata theme



Excel theme

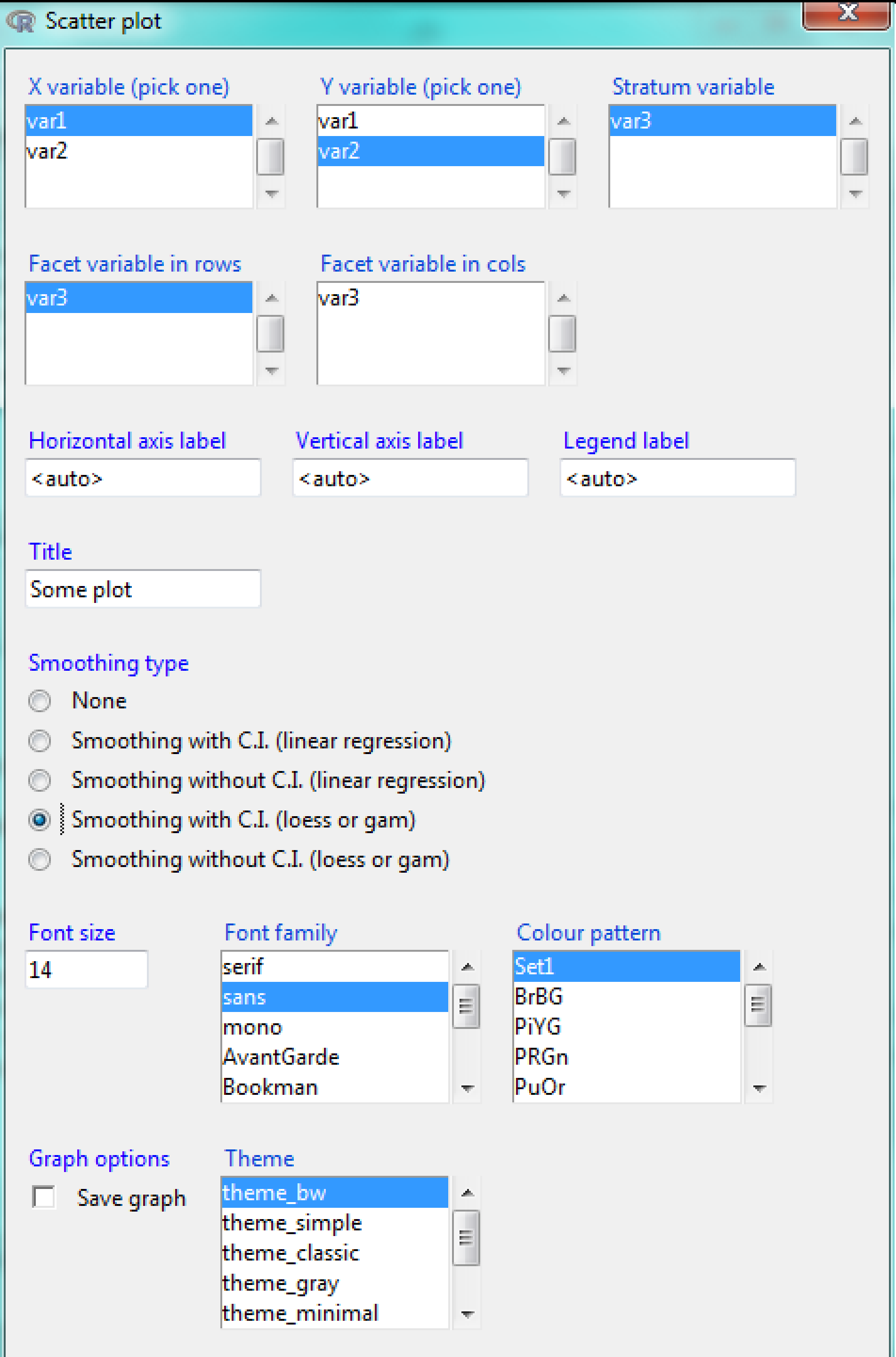


Stephen Few's rules



Google Docs theme

KMggplot2 plugin for RCommander



Scatter plot

X variable (pick one)
- var1
- var2

Y variable (pick one)
- var1
- var2

Stratum variable
- var3

Facet variable in rows
- var3

Facet variable in cols
- var3

Horizontal axis label
<auto>

Vertical axis label
<auto>

Legend label
<auto>

Title
Some plot

Smoothing type
- ○ None
- ○ Smoothing with C.I. (linear regression)
- ○ Smoothing without C.I. (linear regression)
- ● Smoothing with C.I. (loess or gam)
- ○ Smoothing without C.I. (loess or gam)

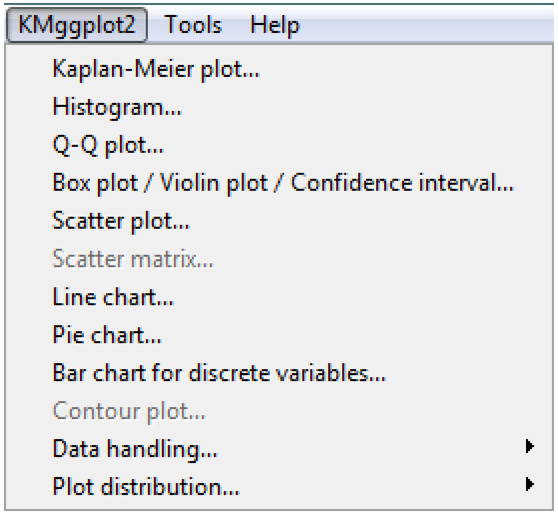
Font size
14

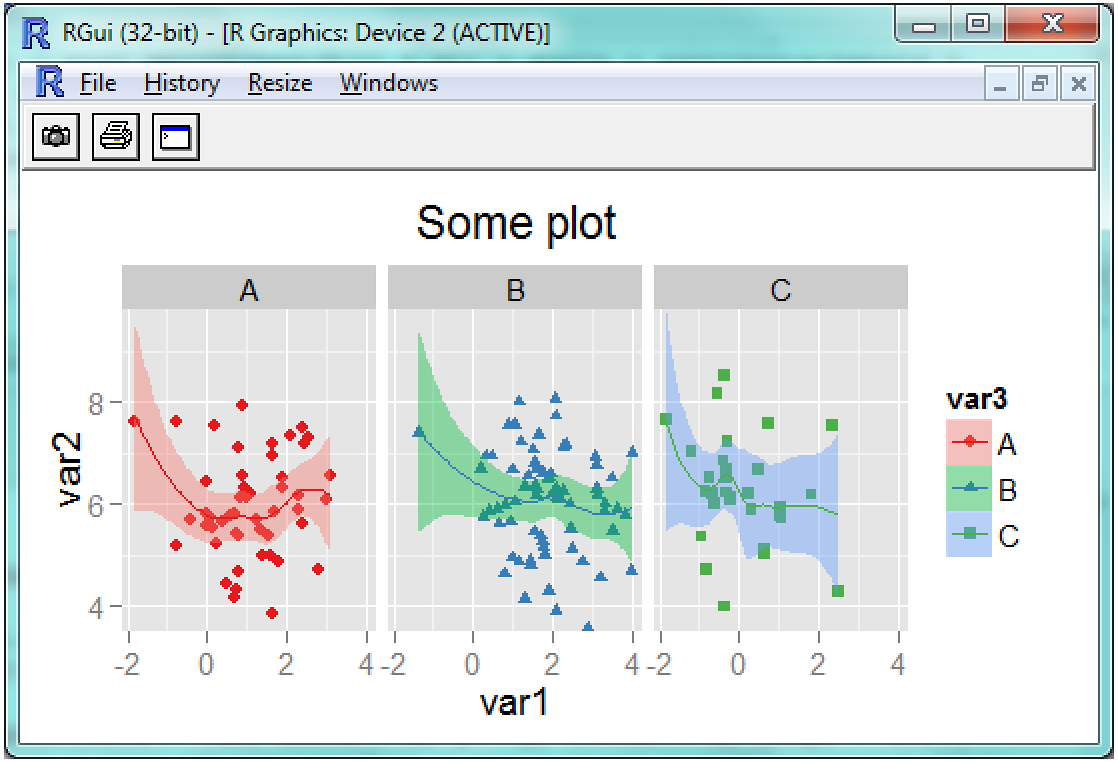
Font family
- serif
- sans
- mono
- AvantGarde
- Bookman

Colour pattern
- Set1
- BrBG
- PiYG
- PRGn
- PuOr

Graph options
- ☐ Save graph

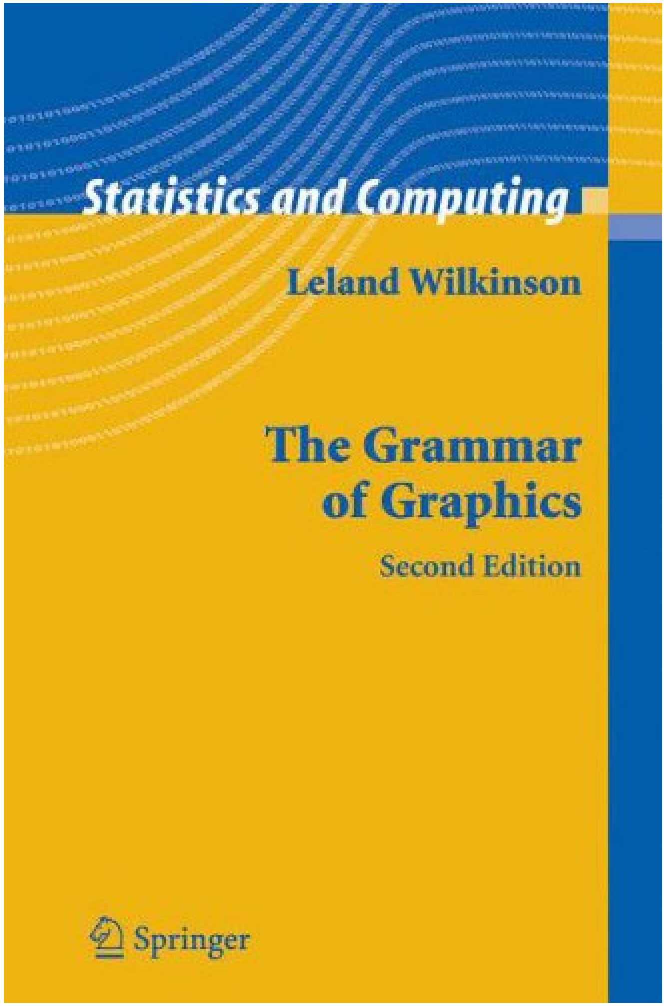
Theme
- theme_bw
- theme_simple
- theme_classic
- theme_gray
- theme_minimal





Deducer

Since ggplot2 is an implementation of Grammar of Graphics, which defines any graphics as a set of objects and layers and properties, it is possible to create graphical editor working on the principle “point and click”.



Statistics and Computing

Leland Wilkinson

The Grammar of Graphics

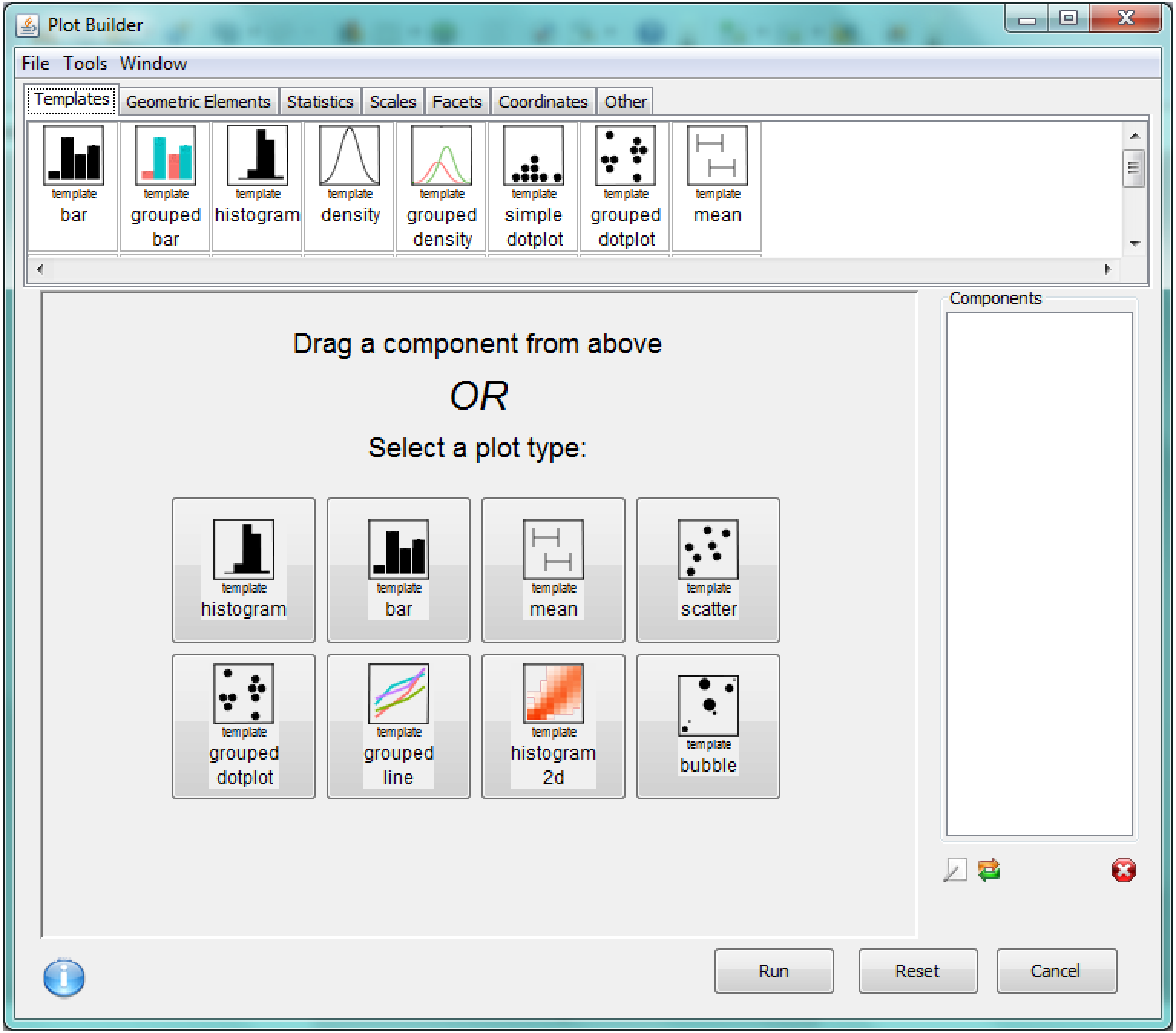
Second Edition

Springer

The Deducer package is an attempt to achieve this goal drawing from the power of ggplot2.

It allows the user to define complex panel of charts using only mouse.

This is, in my opinion, one of the most advanced, graphical, free chart creator available in the Internet.

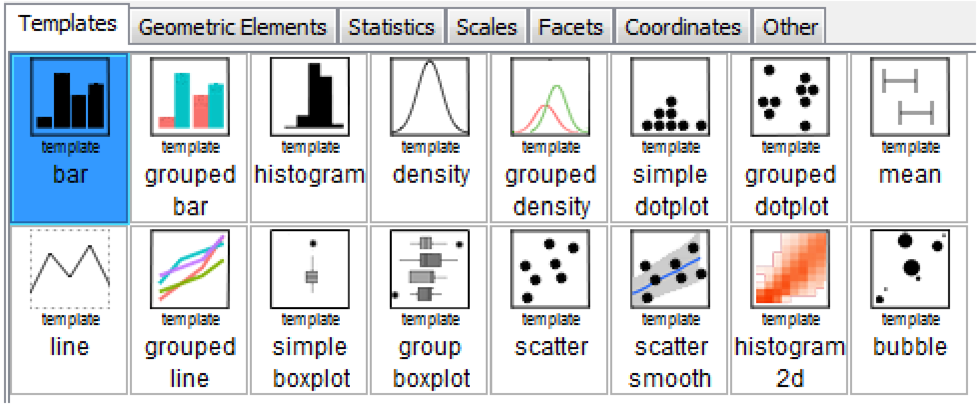


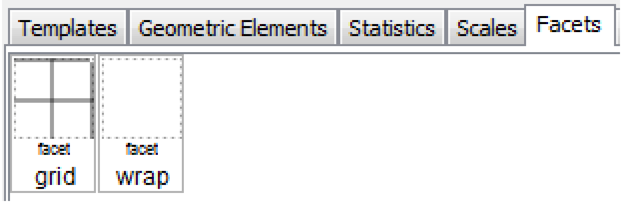
Drag a component from above

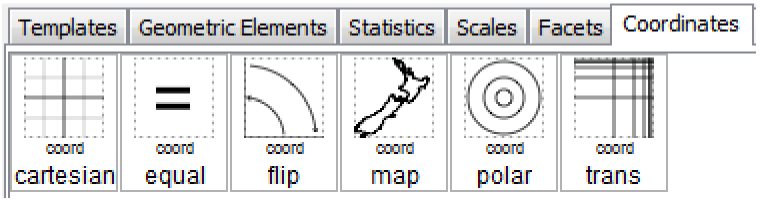
OR

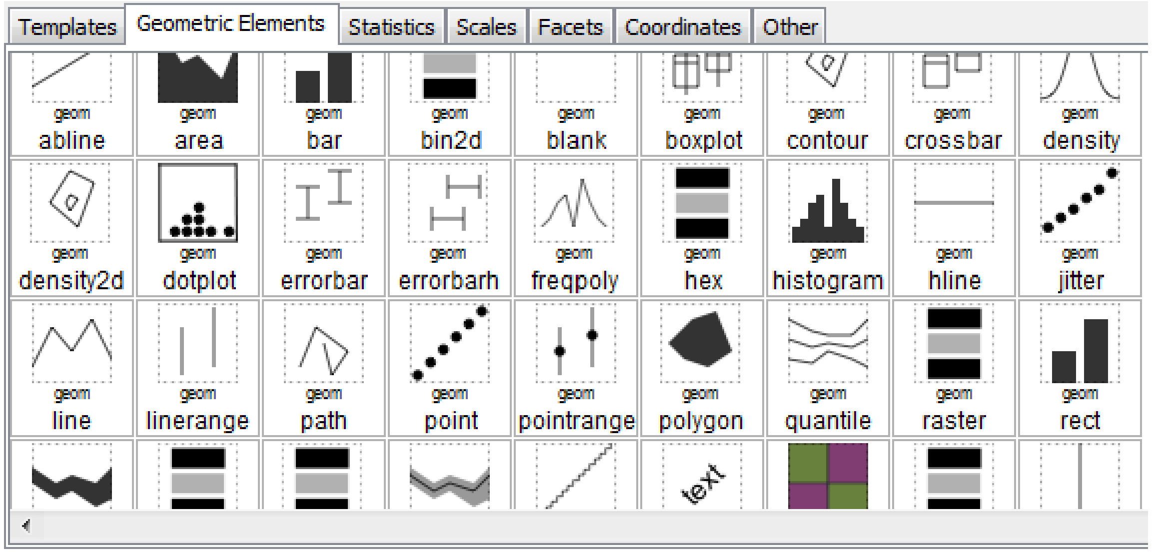
Select a plot type:

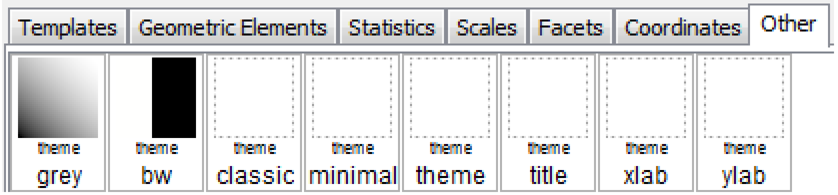
Deducer

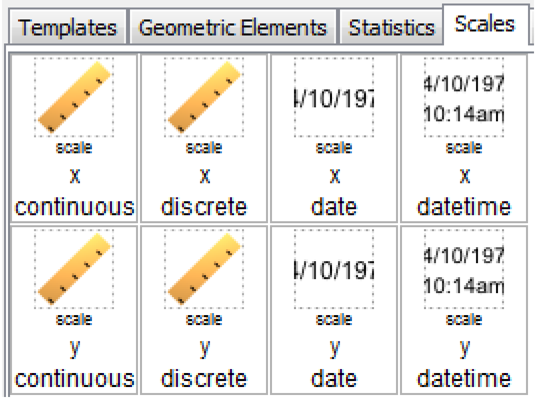

Templates
Geometric Elements
Statistics
Scales
Facets
Coordinates
Other
template bar
template grouped bar
template histogram
template density
template grouped density
template simple dotplot
template grouped dotplot
template mean
template line
template grouped line
template simple boxplot
template group boxplot
template scatter
template scatter smooth
template histogram 2d
template bubble

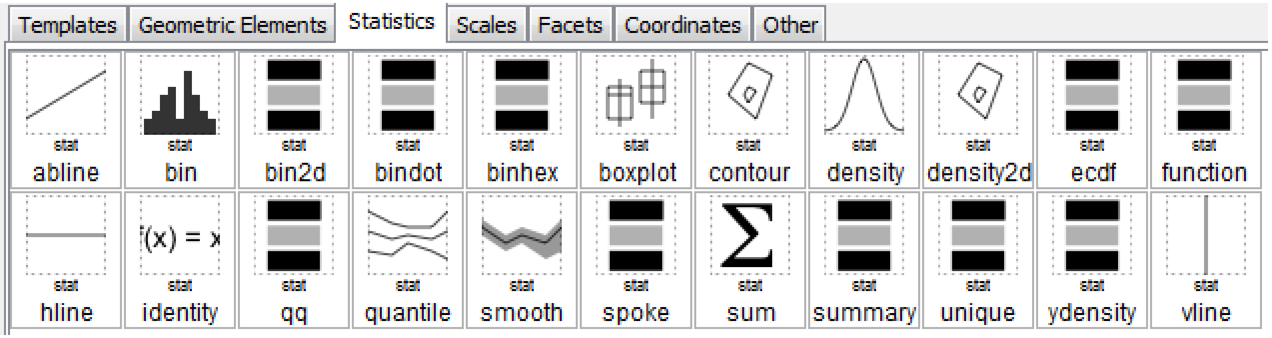

Templates
Geometric Elements
Statistics
Scales
Facets
facet grid
facet wrap


Templates
Geometric Elements
Statistics
Scales
Facets
Coordinates
coord cartesian
coord equal
coord flip
coord map
coord polar
coord trans

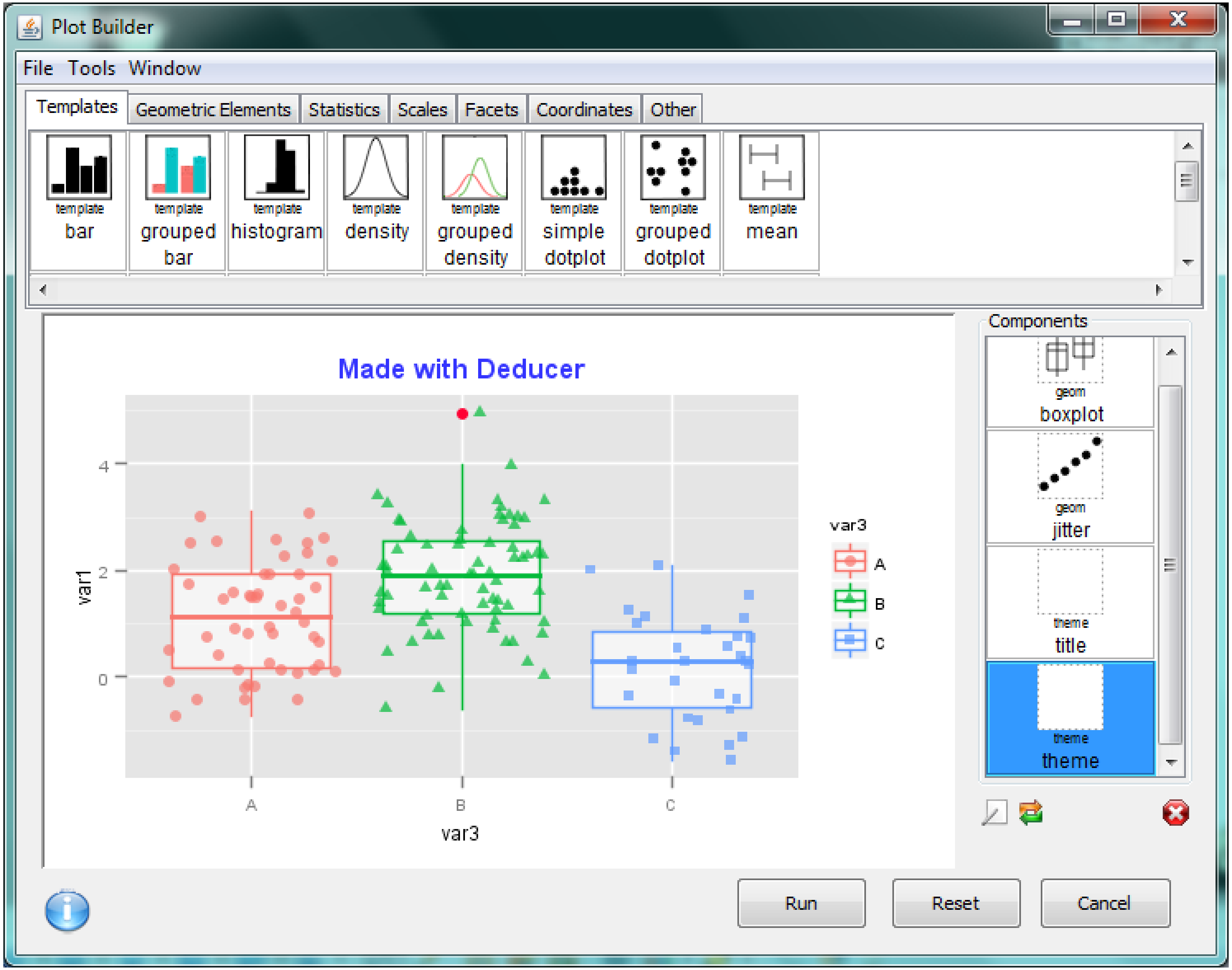

Templates
Geometric Elements
Statistics
Scales
Facets
Coordinates
Other
geom abline
geom area
geom bar
geom bin2d
geom blank
geom boxplot
geom contour
geom crossbar
geom density
geom density2d
geom dotplot
geom errorbar
geom errorbarh
geom freqpoly
geom hex
geom histogram
geom hline
geom jitter
geom line
geom linerange
geom path
geom point
geom pointrange
geom polygon
geom quantile
geom raster
geom rect
text


Templates
Geometric Elements
Statistics
Scales
Facets
Coordinates
Other
theme grey
theme bw
theme classic
theme minimal
theme theme
theme title
theme xlab
theme ylab


Templates
Geometric Elements
Statistics
Scales
1/10/197
4/10/197
10:14am
scale x continuous
scale x discrete
scale x date
scale x datetime
scale y continuous
scale y discrete
scale y date
scale y datetime

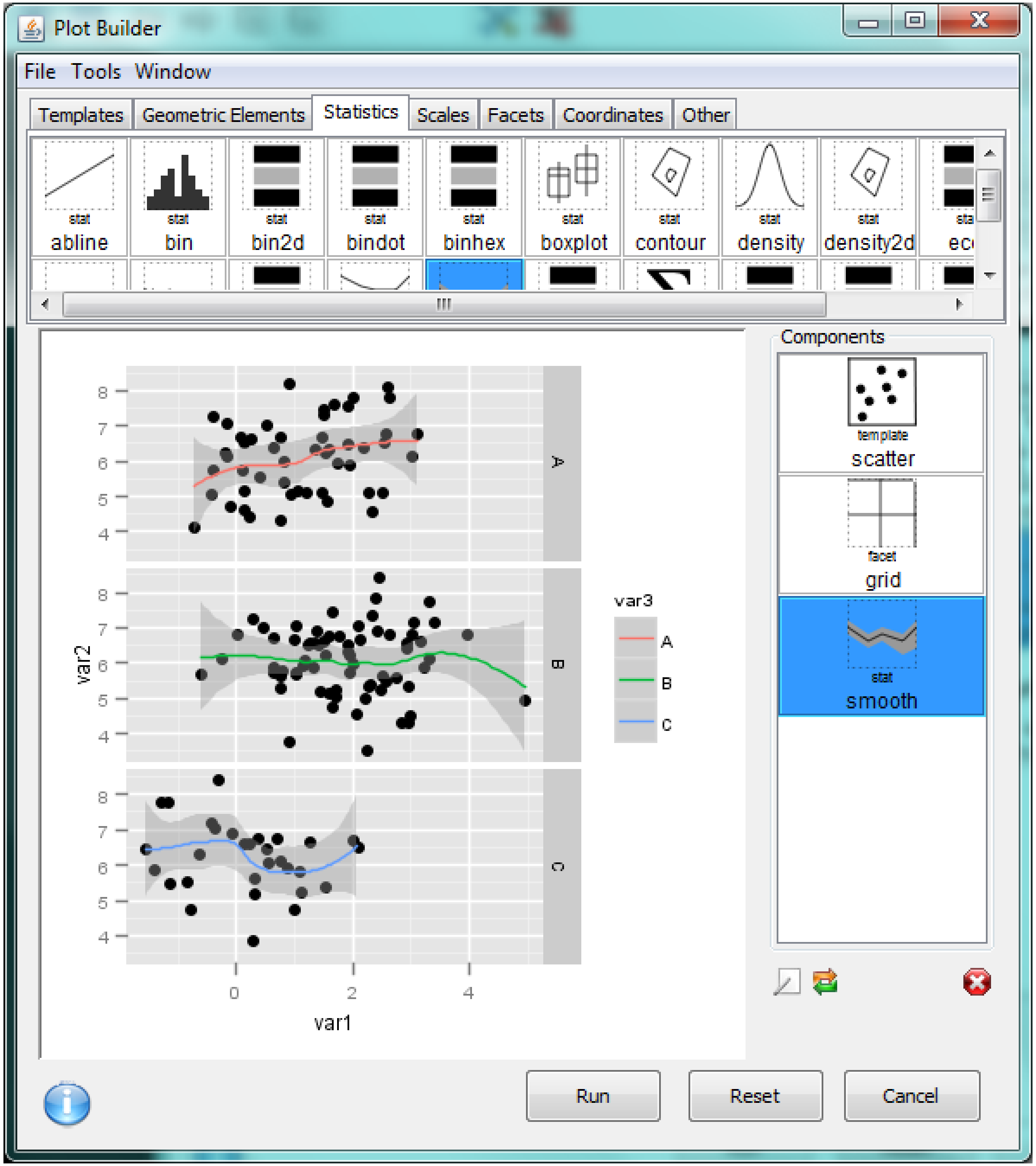

Templates
Geometric Elements
Statistics
Scales
Facets
Coordinates
Other
stat abline
stat bin
stat bin2d
stat bindot
stat binhex
stat boxplot
stat contour
stat density
stat density2d
stat ecdf
stat function
stat hline
f(x) = x
stat identity
stat qq
stat quantile
stat smooth
stat spoke
Σ
stat sum
stat summary
stat unique
stat ydensity
stat vline

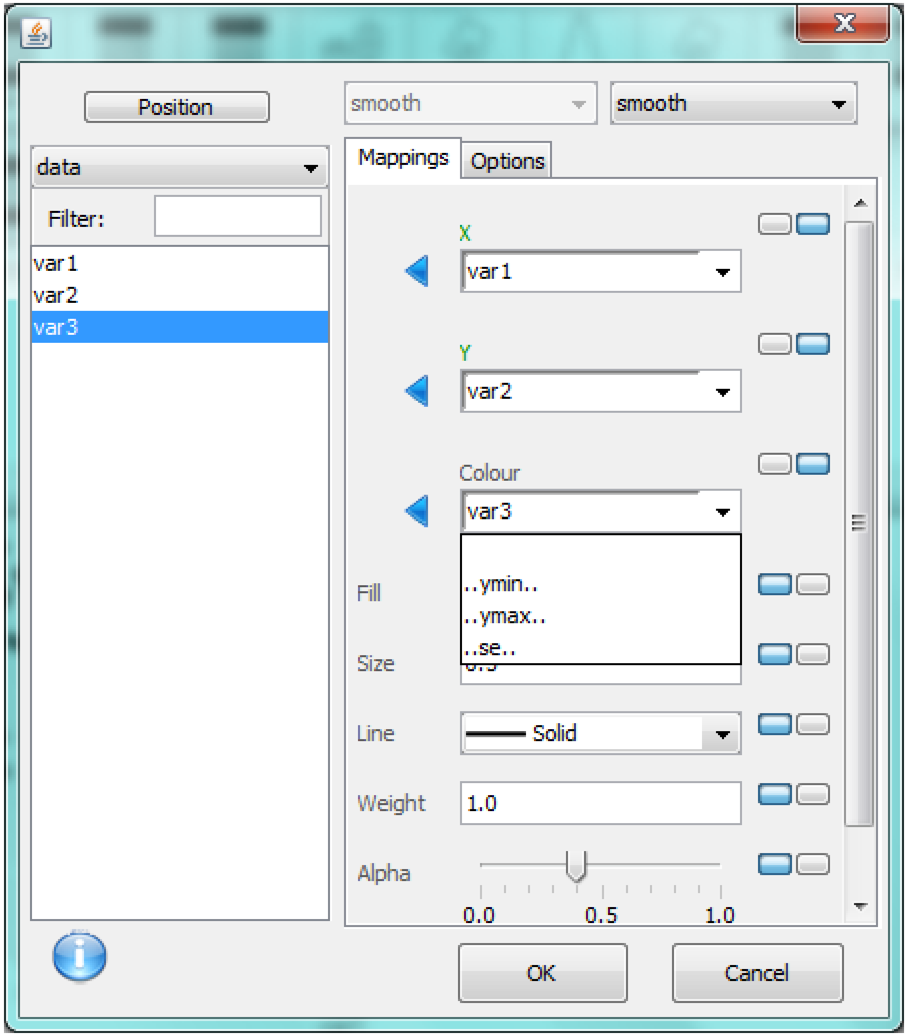
Deducer



```
ggplot() + geom_boxplot(aes(y = var1,x = var3,colour = var3),data=data,alpha = 0.6,outlier.colour = '#ff0033') +
          geom_jitter(aes(x = var3,y = var1,shape = var3,colour = var3),data=data,alpha = 0.7) +
          ggtitle(label = 'Made with Deducer') +
          theme(plot.title = element_text(family = 'Helvetica',face = 'bold',colour = '#3333ff',size = 17.0,vjust =
1.0),panel.border = element_line())
```

Deducer





X: var1

Y: var2

Colour: var3

Fill

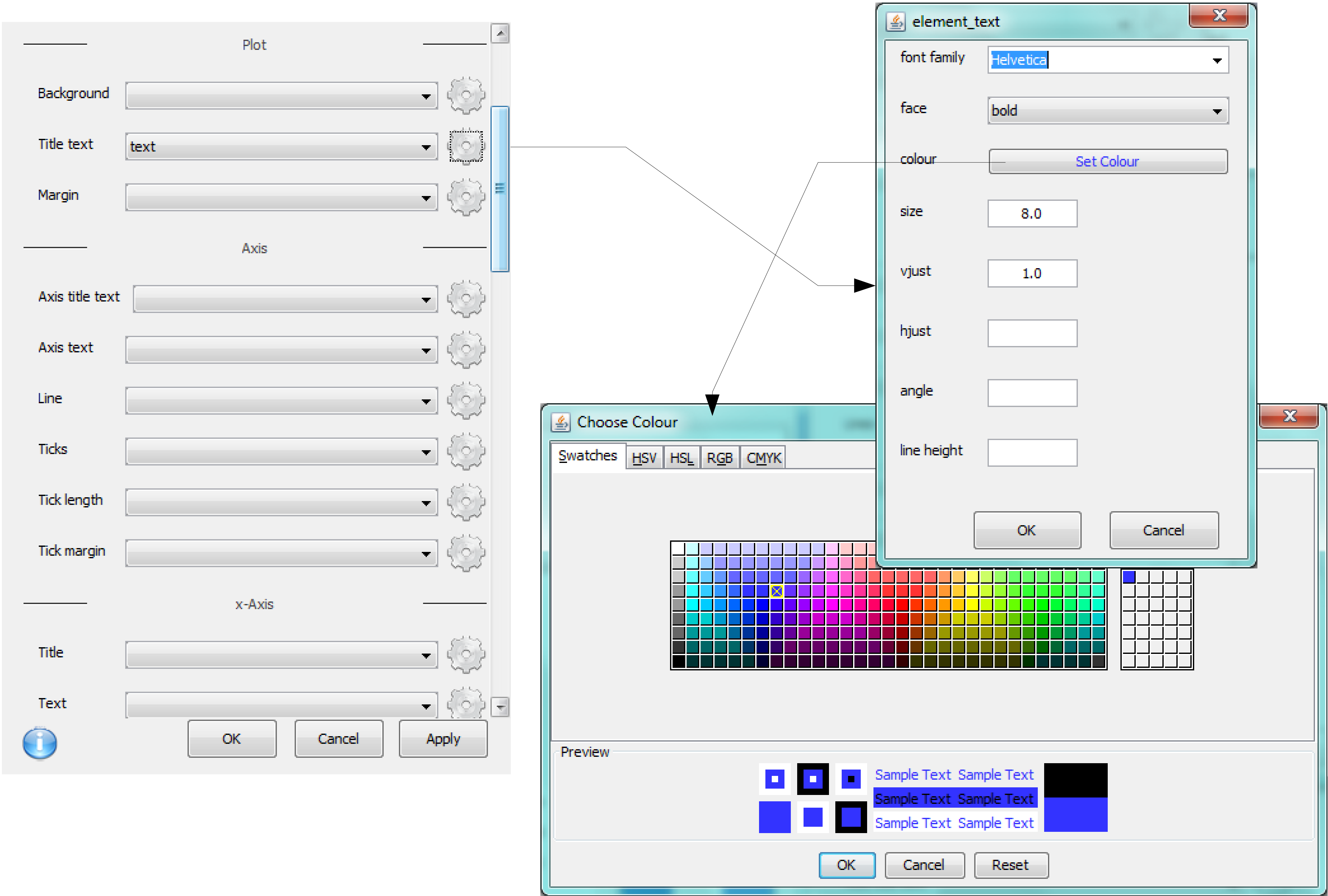
Size

Line: Solid

Weight: 1.0

Alpha

Deducer – control dozens of graph properties



Title text: text

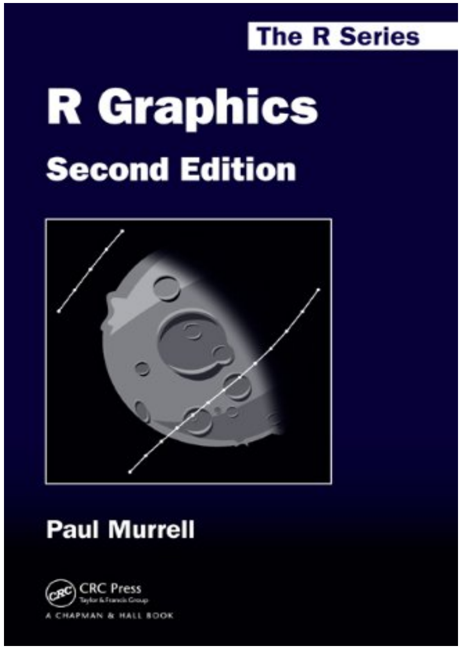
font family: Helvetica

face: bold

size: 8.0

vjust: 1.0

Some books about data visualization in R


The R Series

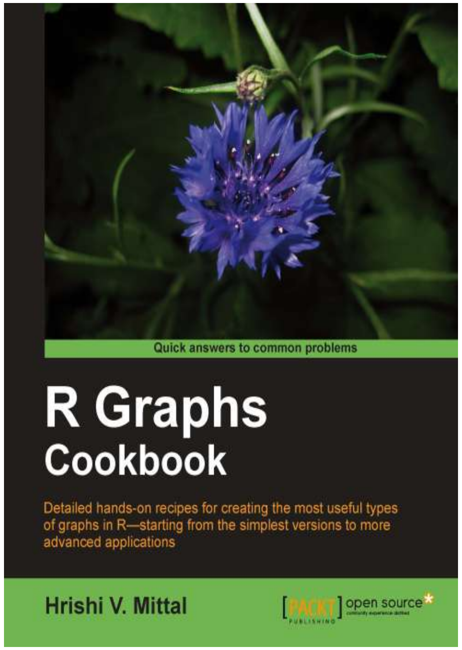
R Graphics

Second Edition

Paul Murrell

CRC Press

A CHAPMAN & HALL BOOK

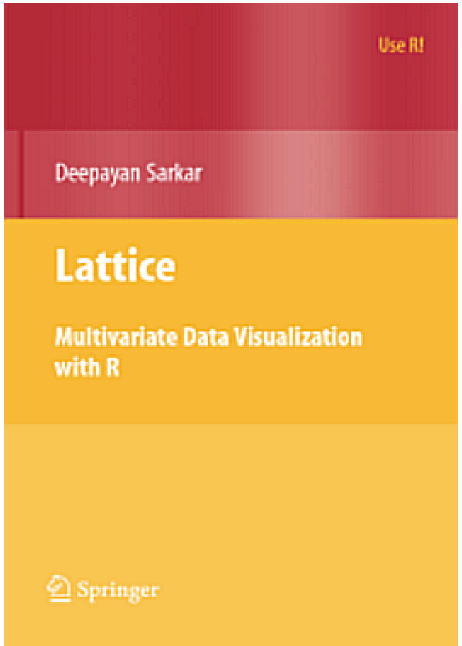

Quick answers to common problems

R Graphs Cookbook

Detailed hands-on recipes for creating the most useful types of graphs in R—starting from the simplest versions to more advanced applications

Hrishi V. Mittal

[PACKT] PUBLISHING open source

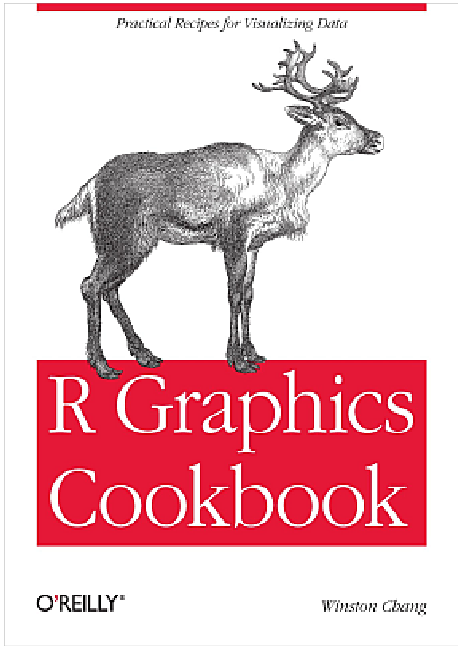

Use R!

Deepayan Sarkar

Lattice

Multivariate Data Visualization with R

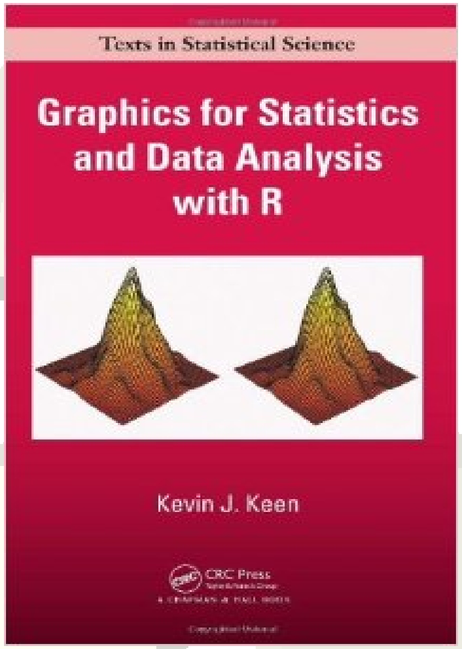
Springer


Practical Recipes for Visualizing Data

R Graphics Cookbook

O'REILLY

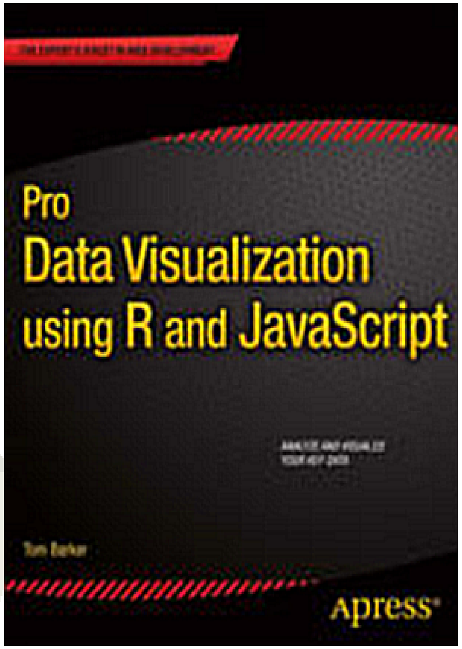
Winston Chang


Texts in Statistical Science

Graphics for Statistics and Data Analysis with R

Kevin J. Keen

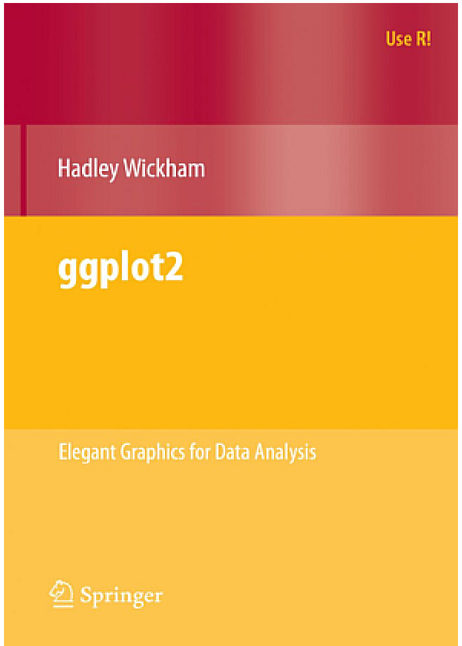
CRC Press


Pro Data Visualization using R and JavaScript

ANALYZE AND VISUALIZE YOUR KEY DATA

Tom Barker

Apress

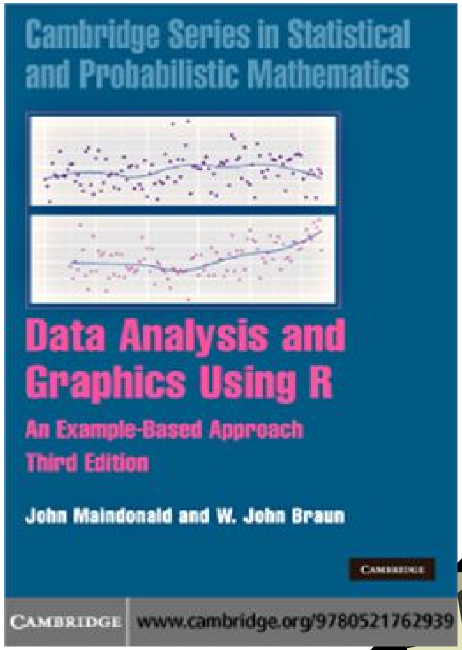

Use R!

Hadley Wickham

ggplot2

Elegant Graphics for Data Analysis

Springer


Cambridge Series in Statistical and Probabilistic Mathematics

Data Analysis and Graphics Using R

An Example-Based Approach

Third Edition

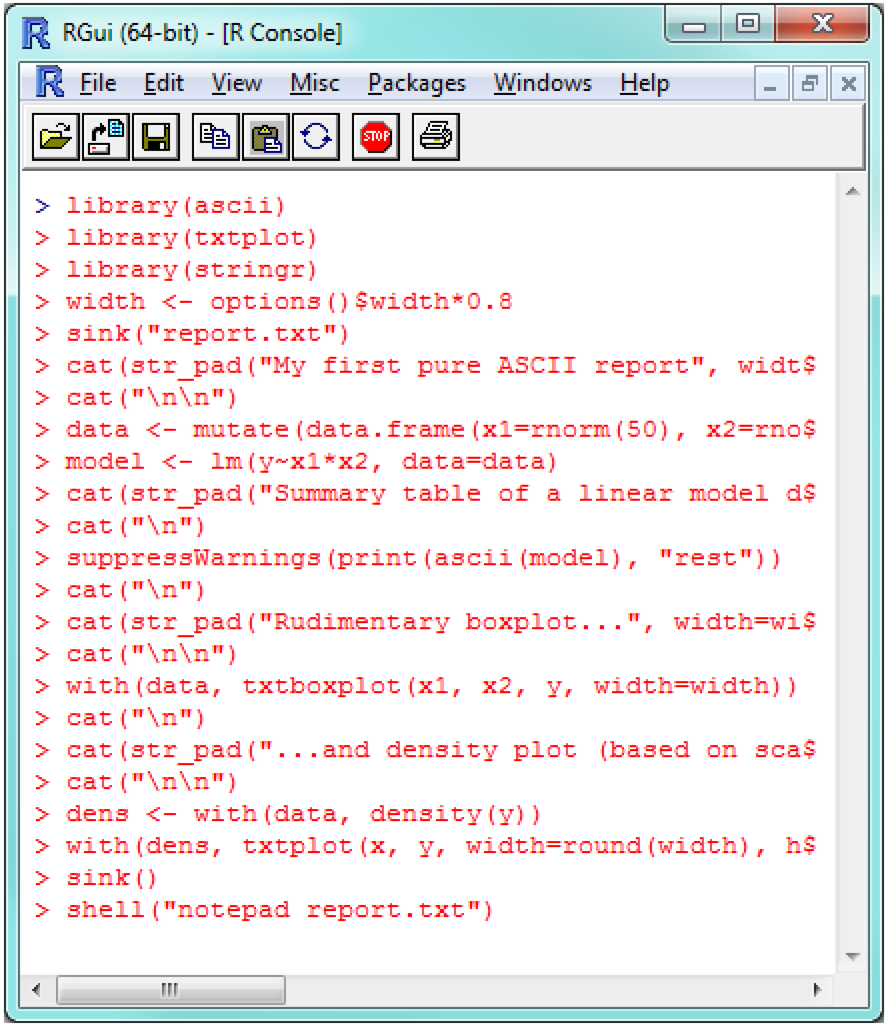
John Maindonald and W. John Braun

CAMBRIDGE

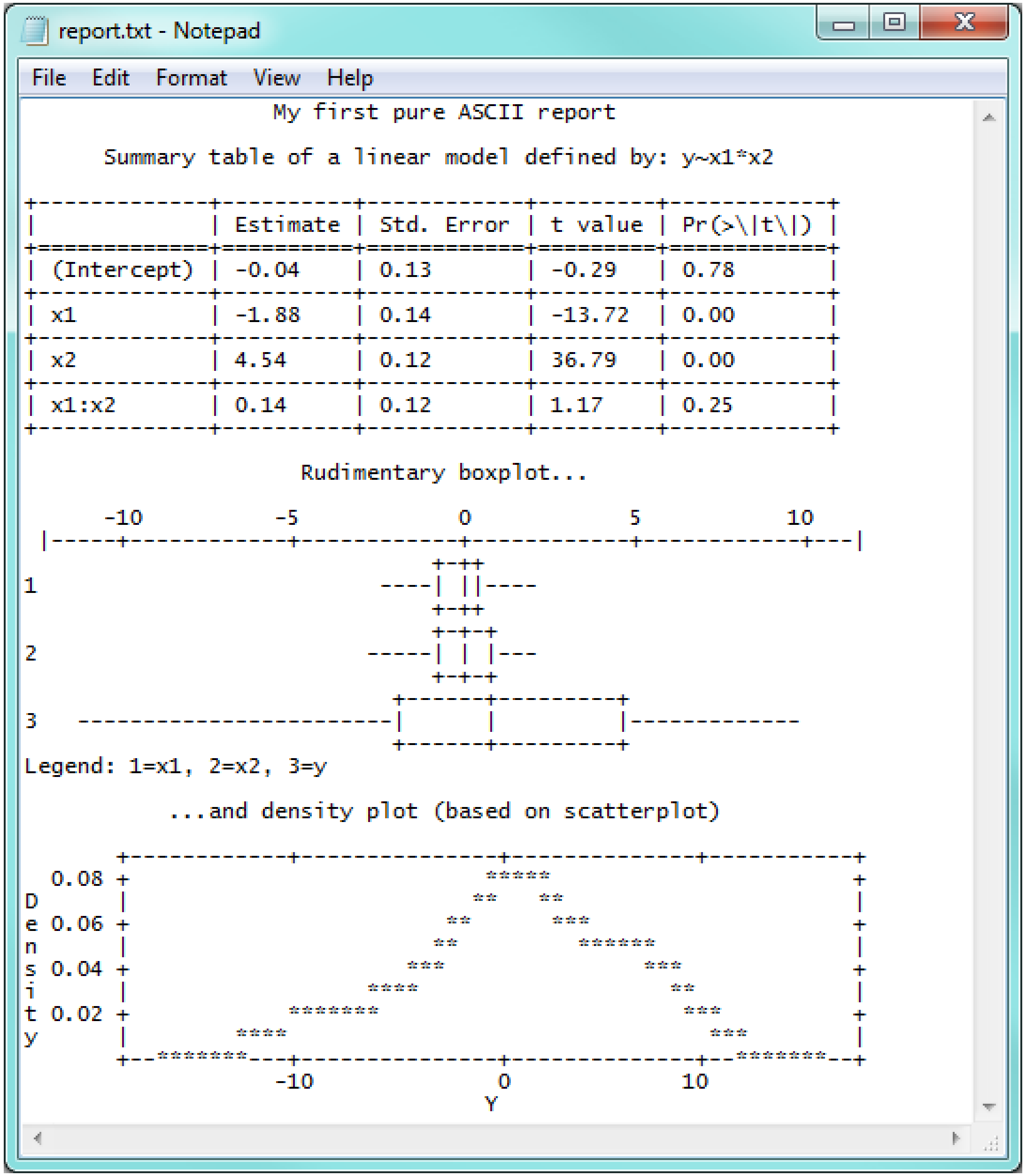
CAMBRIDGE www.cambridge.org/9780521762939

Plain ASCII output

With ascii and txtplot packages one can create text logs (listings) like SAS does.



```
> library(ascii)
> library(txtplot)
> library(stringr)
> width <- options()$width*0.8
> sink("report.txt")
> cat(str_pad("My first pure ASCII report", widt$
> cat("\n\n")
> data <- mutate(data.frame(x1=rnorm(50), x2=rno$
> model <- lm(y~x1*x2, data=data)
> cat(str_pad("Summary table of a linear model d$
> cat("\n")
> suppressWarnings(print(ascii(model), "rest"))
> cat("\n")
> cat(str_pad("Rudimentary boxplot...", width=wi$
> cat("\n\n")
> with(data, txtboxplot(x1, x2, y, width=width))
> cat("\n")
> cat(str_pad("...and density plot (based on sca$
> cat("\n\n")
> dens <- with(data, density(y))
> with(dens, txtplot(x, y, width=round(width), h$
> sink()
> shell("notepad report.txt")
```



```
                   My first pure ASCII report

      Summary table of a linear model defined by: y~x1*x2

+-------------+----------+------------+---------+------------+
|             | Estimate | Std. Error | t value | Pr(>\|t\|) |
+=============+==========+============+=========+============+
| (Intercept) | -0.04    | 0.13       | -0.29   | 0.78       |
+-------------+----------+------------+---------+------------+
| x1          | -1.88    | 0.14       | -13.72  | 0.00       |
+-------------+----------+------------+---------+------------+
| x2          | 4.54     | 0.12       | 36.79   | 0.00       |
+-------------+----------+------------+---------+------------+
| x1:x2       | 0.14     | 0.12       | 1.17    | 0.25       |
+-------------+----------+------------+---------+------------+

                     Rudimentary boxplot...

      -10          -5            0            5           10
 |-----+------------+------------+------------+------------+---|
                               +-++
1                          ----| ||----
                               +-++
                               +-+-+
2                         -----| | |---
                               +-+-+
                            +------+---------+
3   ------------------------|      |         |-------------
                            +------+---------+
Legend: 1=x1, 2=x2, 3=y

           ...and density plot (based on scatterplot)

       +------------+---------------+--------------+-----------+
  0.08 +                           *****                       +
D      |                          **   **                      |
e 0.06 +                        **      ***                    +
n      |                       **         ******               |
s 0.04 +                     ***               ***             +
i      |                  ****                   **            |
t 0.02 +            *******                       ***          +
y      |        ****                                ***        |
       +--*******---+---------------+--------------+--*******--+
                   -10              0             10
                                   Y
```

13 reasons why **you will** love GNU R

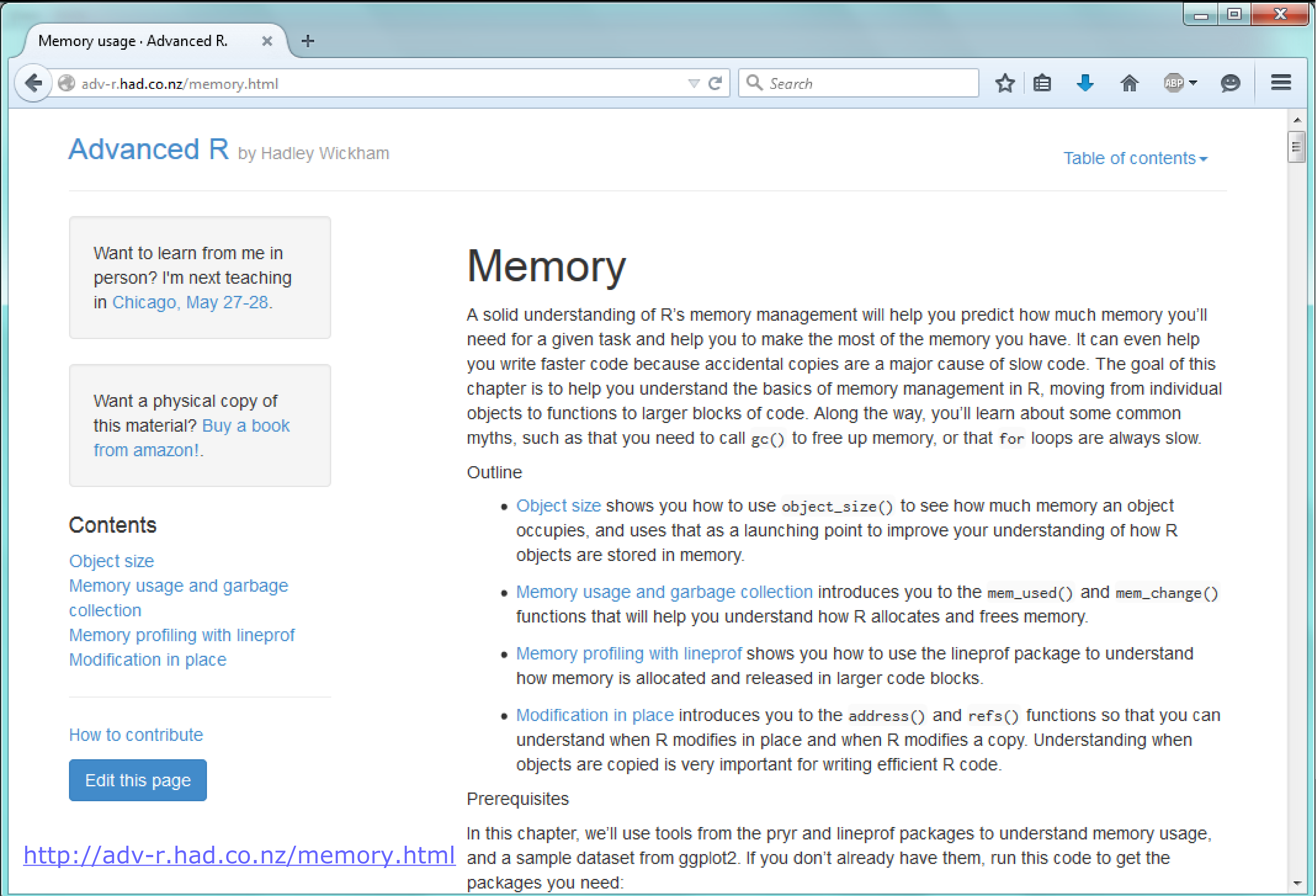
I R is (extremely) cheap. In fact - it's free :)
II R has (extremely) wide range of capabilities
III R is (widely) supported by the world of science
IV R is supported by the community
V R is (increasingly) supported by the business
VI R is able to read data in many formats
VII Interoperability is easy to achieve
VIII R is truly cross-platform
IX R offers numerous ways of presenting data
X There are many options to optimize the code
XI R is able to handle large amount of data
XII R has a set of fancy tools and IDEs
XII FDA accepted using R for drug trials!

Tune it up!

R offers a bunch of methods that can remarkably **increase the performance** of algorithms in the case of performing complex calculations:

- **Profiling** the code, which enables us to determine the computationally intensive portions of the program
- Turning on the "**Just In Time**" compilation of the code
- **Vectorizing** calculations – which means avoiding explicit loops. It can speed up computations 5-10 times
- Performing all **algebraic** computations with the use of libraries **tuned** for our hardware (e.g. BLAS)
- Become familiar with methods of **algorithmic differentiation** (ADMB)
- Executing **parallel** computations in a **cluster** environment
- Using the power of a **graphic card processor** (CUDA, OpenCL)
- **Implement key parts of algorithm in C++** and call them using RCPP

Profiling the memoRy usage



Advanced R by Hadley Wickham

Want to learn from me in person? I'm next teaching in Chicago, May 27-28.

Want a physical copy of this material? Buy a book from amazon!.

Contents

How to contribute

Memory

A solid understanding of R's memory management will help you predict how much memory you'll need for a given task and help you to make the most of the memory you have. It can even help you write faster code because accidental copies are a major cause of slow code. The goal of this chapter is to help you understand the basics of memory management in R, moving from individual objects to functions to larger blocks of code. Along the way, you'll learn about some common myths, such as that you need to call `gc()` to free up memory, or that `for` loops are always slow.

Outline

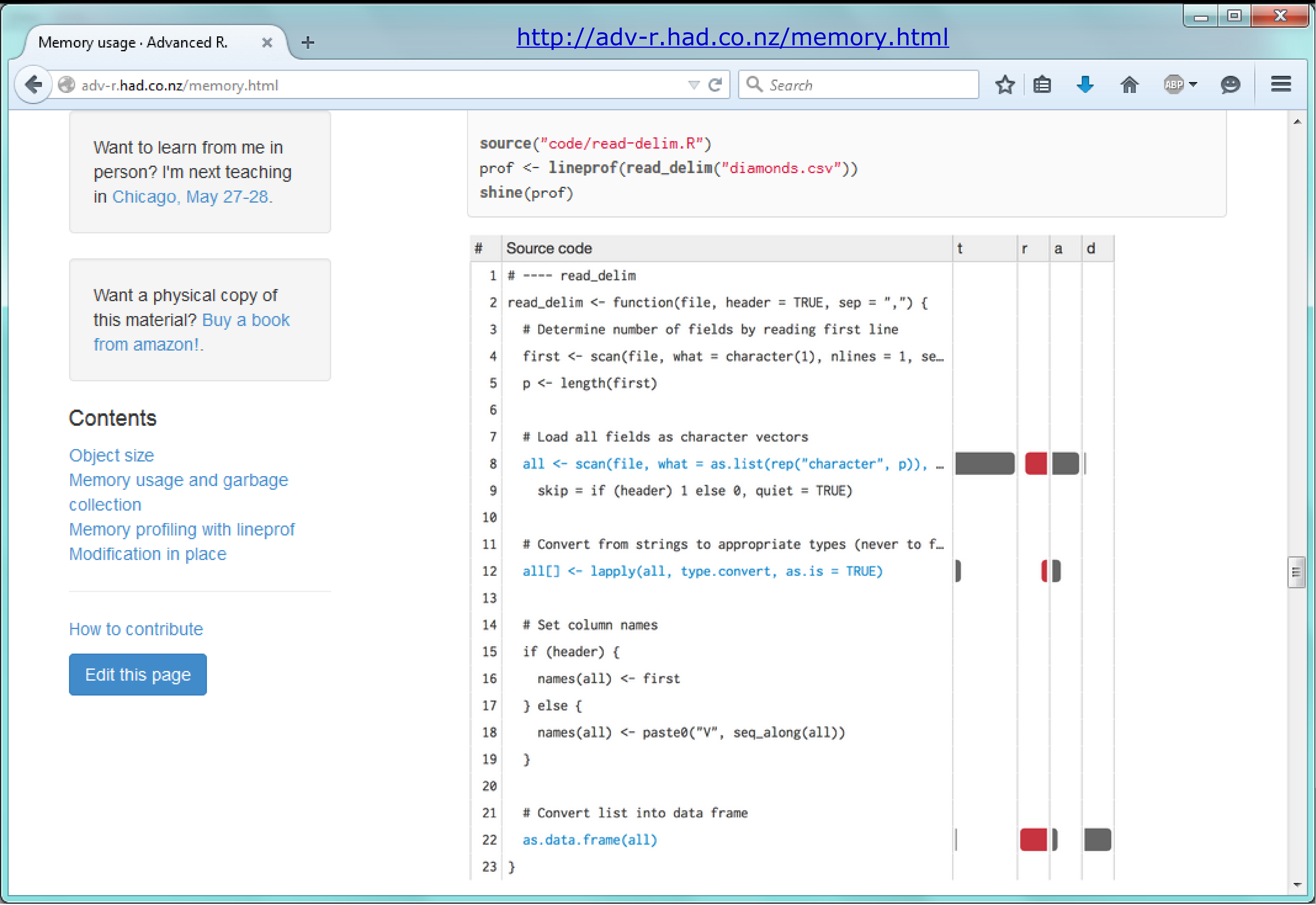
- Object size shows you how to use `object_size()` to see how much memory an object occupies, and uses that as a launching point to improve your understanding of how R objects are stored in memory.
- Memory usage and garbage collection introduces you to the `mem_used()` and `mem_change()` functions that will help you understand how R allocates and frees memory.
- Memory profiling with lineprof shows you how to use the lineprof package to understand how memory is allocated and released in larger code blocks.
- Modification in place introduces you to the `address()` and `refs()` functions so that you can understand when R modifies in place and when R modifies a copy. Understanding when objects are copied is very important for writing efficient R code.

Prerequisites

In this chapter, we'll use tools from the pryr and lineprof packages to understand memory usage, and a sample dataset from ggplot2. If you don't already have them, run this code to get the packages you need:

http://adv-r.had.co.nz/memory.html

Profiling the memoRy usage



Memory usage · Advanced R.

http://adv-r.had.co.nz/memory.html

adv-r.had.co.nz/memory.html

Want to learn from me in person? I'm next teaching in Chicago, May 27-28.

Want a physical copy of this material? Buy a book from amazon!.

Contents

- Object size
- Memory usage and garbage collection
- Memory profiling with lineprof
- Modification in place

How to contribute

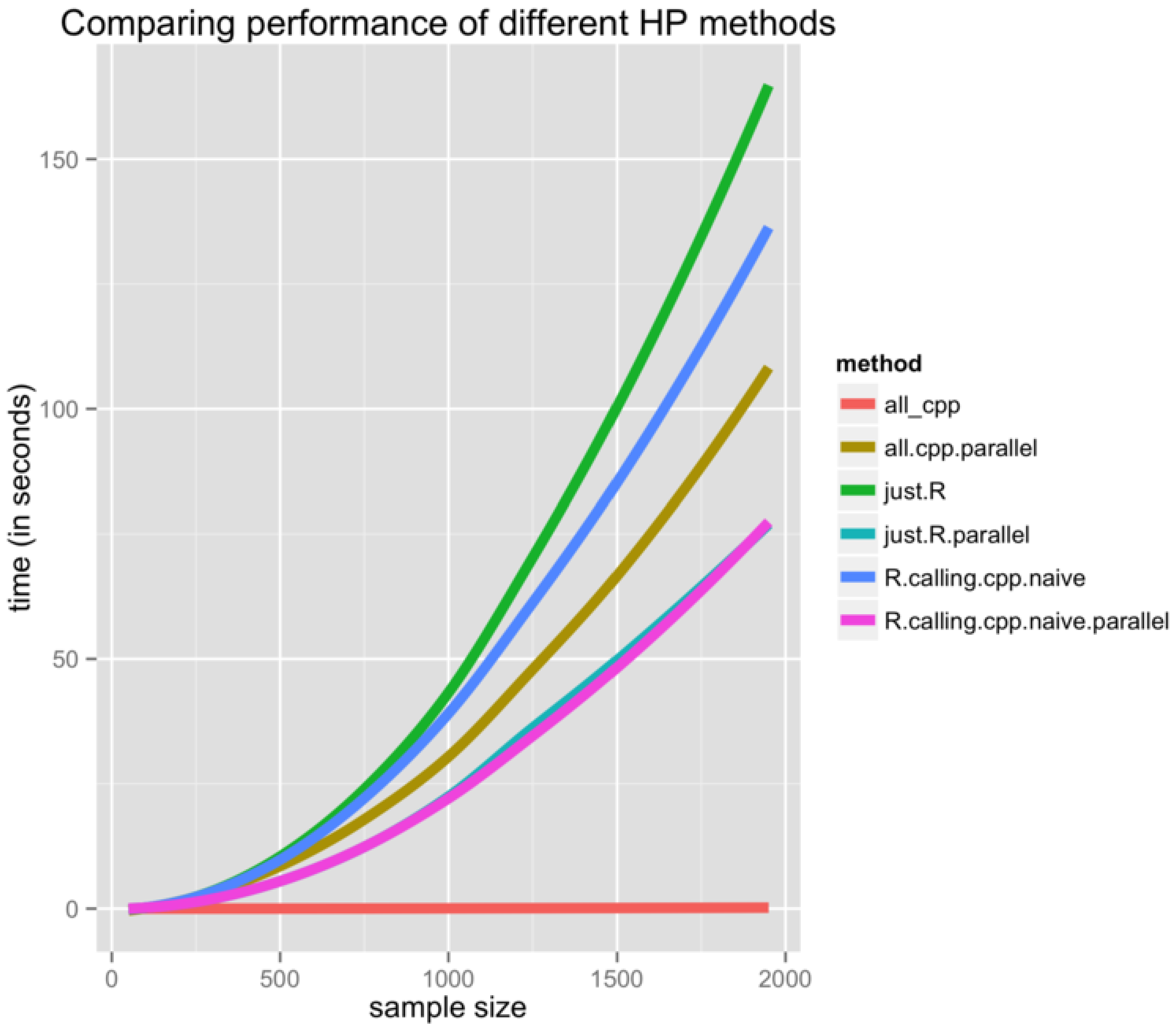
Edit this page

```r
source("code/read-delim.R")
prof <- lineprof(read_delim("diamonds.csv"))
shine(prof)
```

\# Source code t r a d

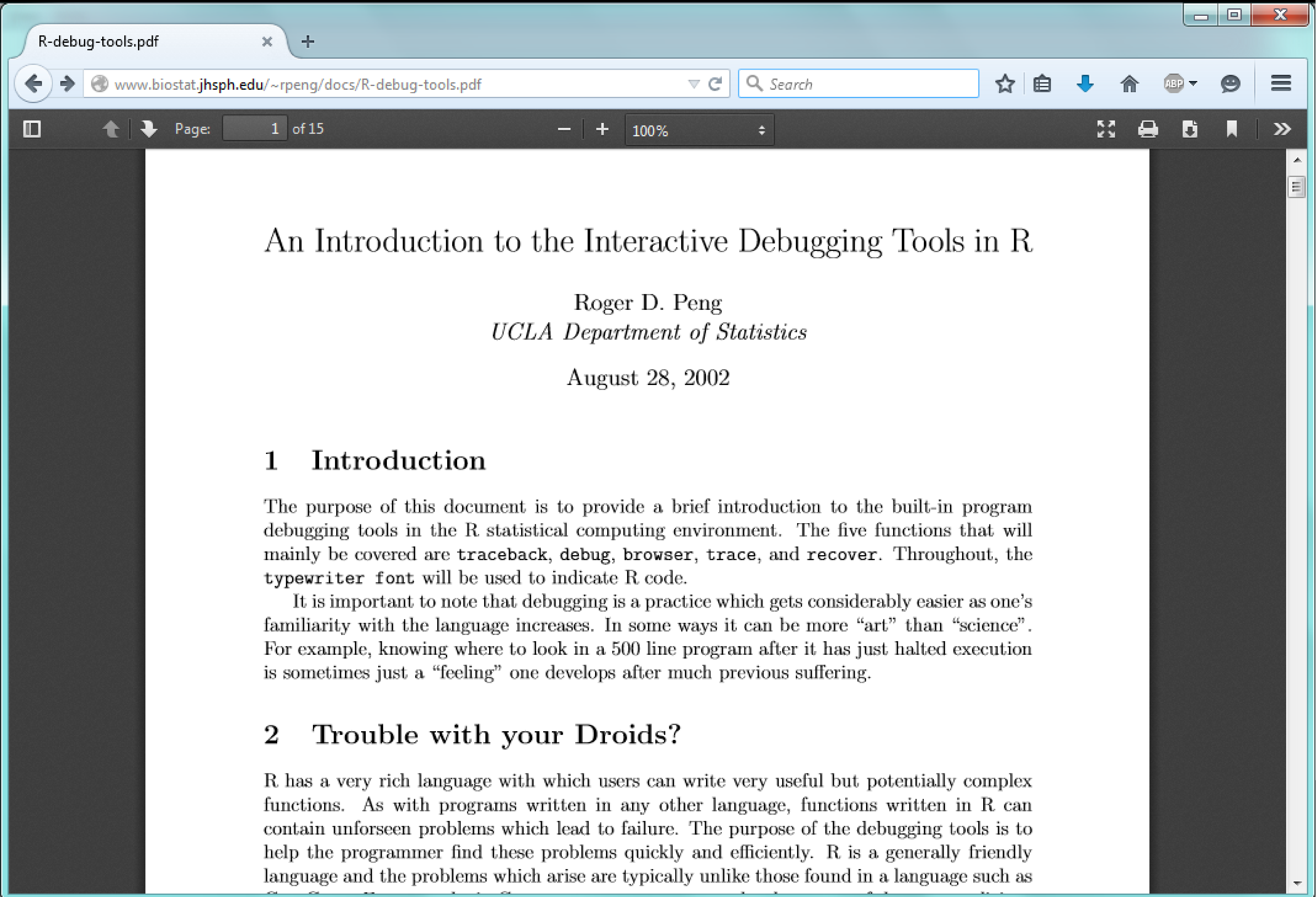
```r
# ---- read_delim
read_delim <- function(file, header = TRUE, sep = ",") {
  # Determine number of fields by reading first line
  first <- scan(file, what = character(1), nlines = 1, se…
  p <- length(first)

  # Load all fields as character vectors
  all <- scan(file, what = as.list(rep("character", p)), …
    skip = if (header) 1 else 0, quiet = TRUE)

  # Convert from strings to appropriate types (never to f…
  all[] <- lapply(all, type.convert, as.is = TRUE)

  # Set column names
  if (header) {
    names(all) <- first
  } else {
    names(all) <- paste0("V", seq_along(all))
  }

  # Convert list into data frame
  as.data.frame(all)
}
```

Tune it up!



http://blog.revolutionanalytics....comparison-of-high-performance-computing-techniques-in-r.html

Debug your code!



An Introduction to the Interactive Debugging Tools in R

Roger D. Peng
UCLA Department of Statistics

August 28, 2002

1 Introduction

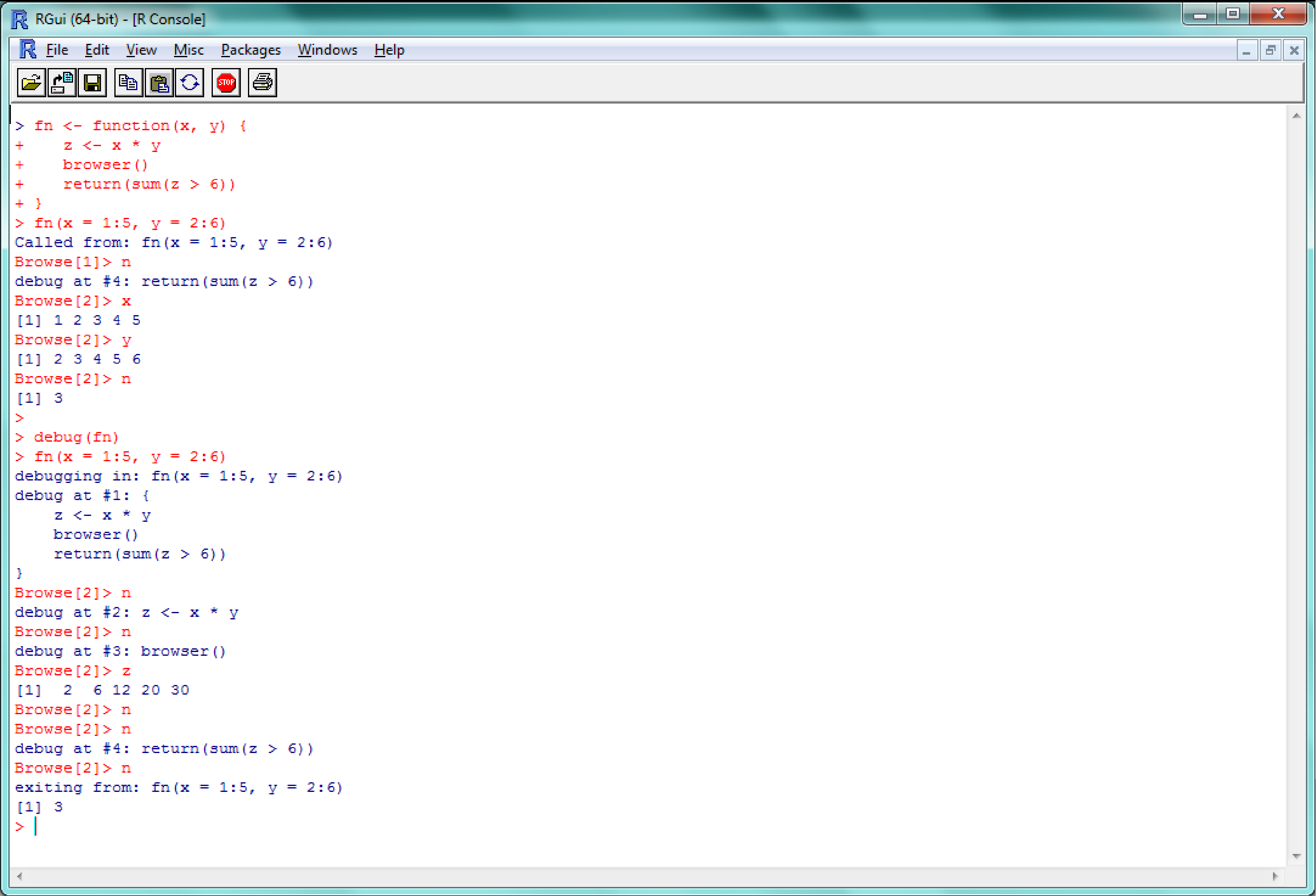
The purpose of this document is to provide a brief introduction to the built-in program debugging tools in the R statistical computing environment. The five functions that will mainly be covered are `traceback`, `debug`, `browser`, `trace`, and `recover`. Throughout, the `typewriter font` will be used to indicate R code.

It is important to note that debugging is a practice which gets considerably easier as one's familiarity with the language increases. In some ways it can be more "art" than "science". For example, knowing where to look in a 500 line program after it has just halted execution is sometimes just a "feeling" one develops after much previous suffering.

2 Trouble with your Droids?

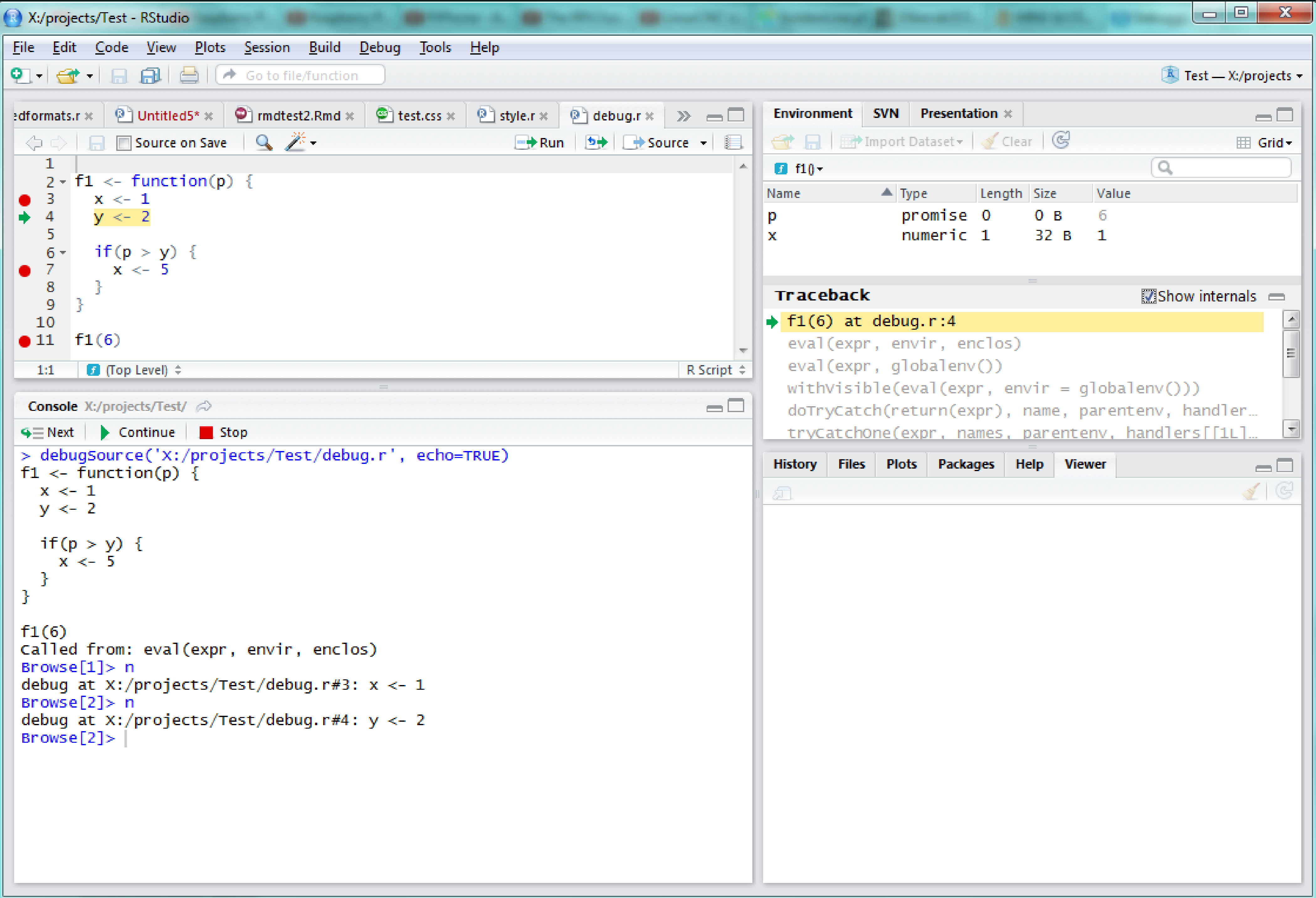
R has a very rich language with which users can write very useful but potentially complex functions. As with programs written in any other language, functions written in R can contain unforseen problems which lead to failure. The purpose of the debugging tools is to help the programmer find these problems quickly and efficiently. R is a generally friendly language and the problems which arise are typically unlike those found in a language such as

Debug your code!



```
> fn <- function(x, y) {
+     z <- x * y
+     browser()
+     return(sum(z > 6))
+ }
> fn(x = 1:5, y = 2:6)
Called from: fn(x = 1:5, y = 2:6)
Browse[1]> n
debug at #4: return(sum(z > 6))
Browse[2]> x
[1] 1 2 3 4 5
Browse[2]> y
[1] 2 3 4 5 6
Browse[2]> n
[1] 3
>
> debug(fn)
> fn(x = 1:5, y = 2:6)
debugging in: fn(x = 1:5, y = 2:6)
debug at #1: {
    z <- x * y
    browser()
    return(sum(z > 6))
}
Browse[2]> n
debug at #2: z <- x * y
Browse[2]> n
debug at #3: browser()
Browse[2]> z
[1]  2  6 12 20 30
Browse[2]> n
Browse[2]> n
debug at #4: return(sum(z > 6))
Browse[2]> n
exiting from: fn(x = 1:5, y = 2:6)
[1] 3
> 
```

Debug your code!



```r

f1 <- function(p) {
  x <- 1
  y <- 2

  if(p > y) {
    x <- 5
  }
}

f1(6)
```

Name	Type	Length	Size	Value
p	promise	0	0 B	6
x	numeric	1	32 B	1

Traceback

```
f1(6) at debug.r:4
eval(expr, envir, enclos)
eval(expr, globalenv())
withVisible(eval(expr, envir = globalenv()))
doTryCatch(return(expr), name, parentenv, handler…
tryCatchOne(expr, names, parentenv, handlers[[1L]…
```

```
> debugSource('X:/projects/Test/debug.r', echo=TRUE)
f1 <- function(p) {
  x <- 1
  y <- 2

  if(p > y) {
    x <- 5
  }
}

f1(6)
Called from: eval(expr, envir, enclos)
Browse[1]> n
debug at X:/projects/Test/debug.r#3: x <- 1
Browse[2]> n
debug at X:/projects/Test/debug.r#4: y <- 2
Browse[2]>
```

13 reasons why **you will** love GNU R

I	R is (extremely) cheap. In fact - it's free :)
II	R has (extremely) wide range of capabilities
III	R is (widely) supported by the world of science
IV	R is supported by the community
V	R is (increasingly) supported by the business
VI	R is able to read data in many formats
VII	Interoperability is easy to achieve
VIII	R is truly cross-platform
IX	R offers numerous ways of presenting data
X	There are many options to optimize the code
XI	**R is able to handle large amount of data**
XII	R has a set of fancy tools and IDEs
XII	FDA accepted using R for drug trials!

“Big data” doesn't mean a “big trouble”

Sometimes there is a need for processing large datasets that don't fit in the RAM memory. Size of the data cannot exceed 2GB per process in a 32-bit machine and 8GB per process in a 64-bit machine.

In such cases R gives a couple of methods for handling massive data:

- Building models on chunks of the data. Only linear models are supported
- Mapping physical memory on a set of files (disk virtual memory)
- Storing parts of datasets outside of the current process memory

pbdR

Programming with Big Data in R

http://www.r-pbd.org/

“Big data” doesn't mean a “big trouble”

REVOLUTION ANALYTICS

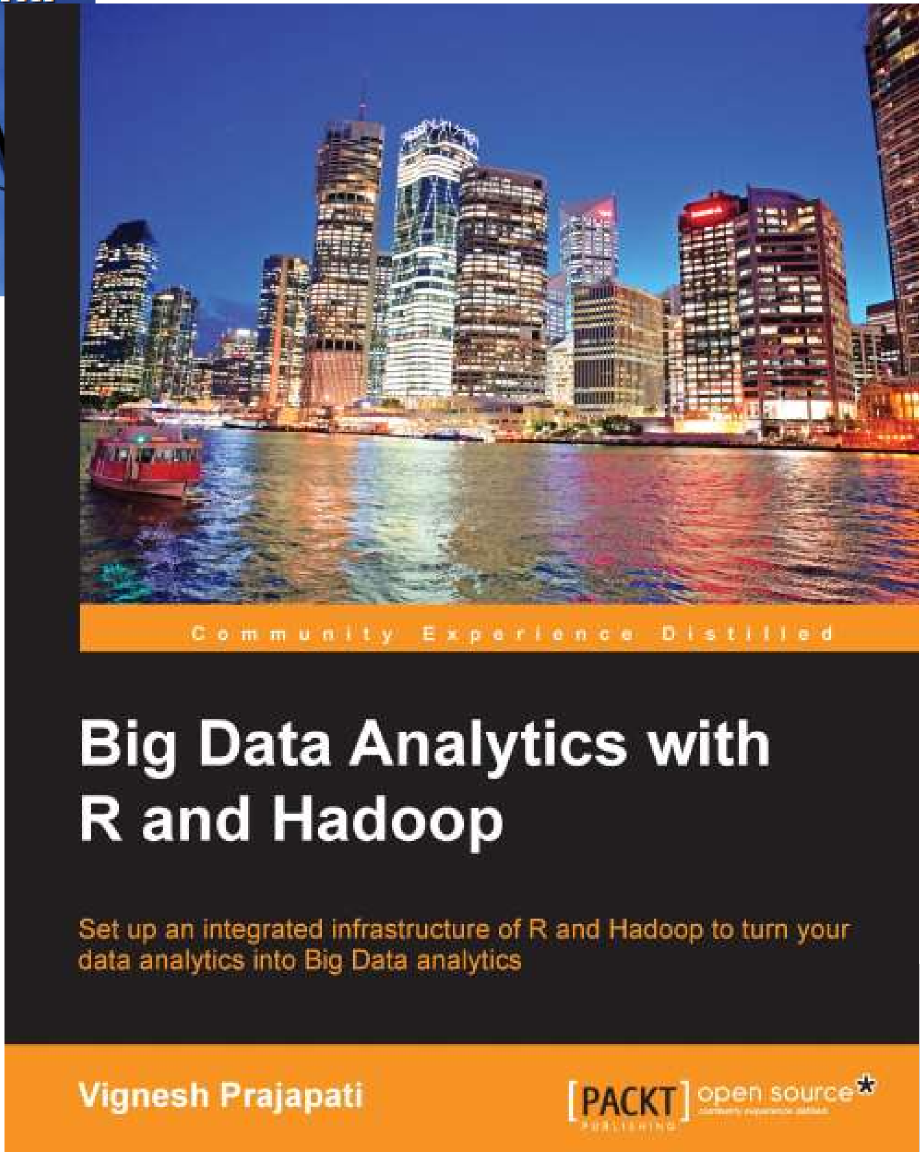
Get Technical Support

Big Data Big Analytics Software

Innovation, performance and value for today's data architecture.

High-Performance and Parallel Computing with R:

- Parallel computing:
  - Explicit and implicit parallelism
  - Grid computing
  - Hadoop
  - Random numbers
  - Resource managers and batch schedulers
  - Applications
  - GPUs
- Large memory and out-of-memory data
- Easier interfaces for Compiled code
- Profiling tools



Community Experience Distilled

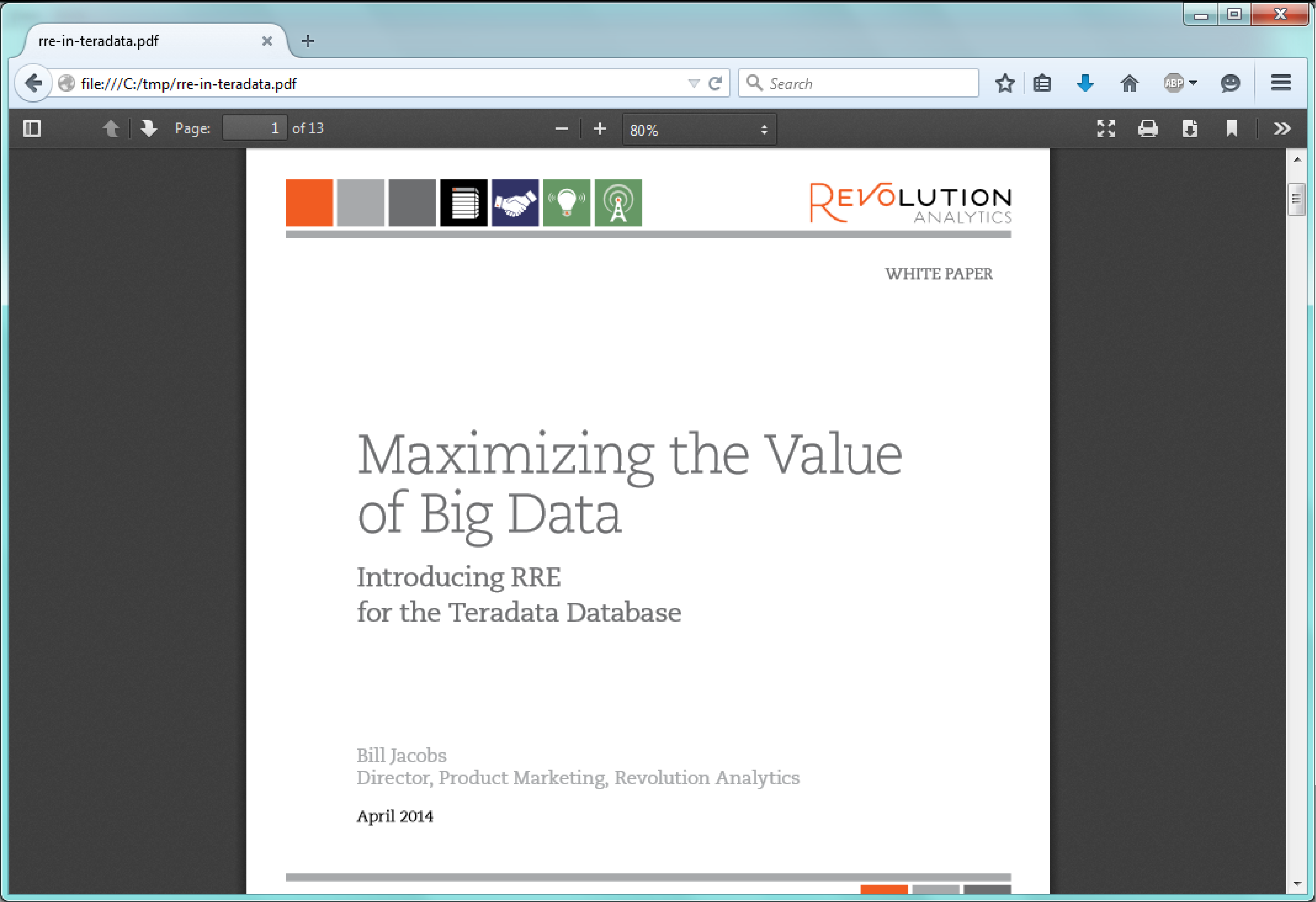
Big Data Analytics with R and Hadoop

Set up an integrated infrastructure of R and Hadoop to turn your data analytics into Big Data analytics

Vignesh Prajapati

[PACKT] open source

“Big data” doesn't mean a “big trouble”



WHITE PAPER

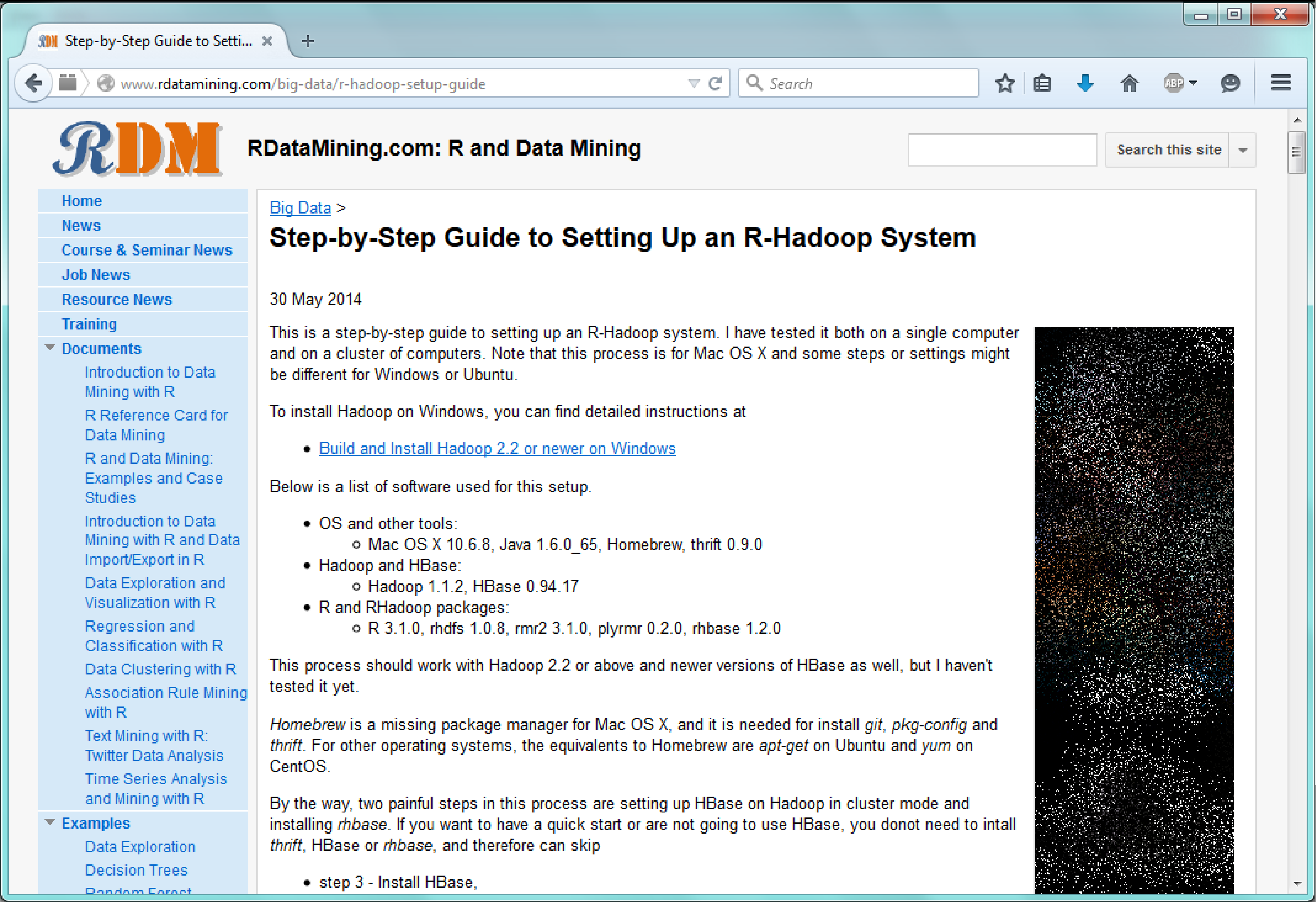
Maximizing the Value of Big Data

Introducing RRE
for the Teradata Database

Bill Jacobs
Director, Product Marketing, Revolution Analytics

April 2014

“Big data” doesn't mean a “big trouble”



RDataMining.com: R and Data Mining

Big Data >

Step-by-Step Guide to Setting Up an R-Hadoop System

30 May 2014

This is a step-by-step guide to setting up an R-Hadoop system. I have tested it both on a single computer and on a cluster of computers. Note that this process is for Mac OS X and some steps or settings might be different for Windows or Ubuntu.

To install Hadoop on Windows, you can find detailed instructions at

- Build and Install Hadoop 2.2 or newer on Windows

Below is a list of software used for this setup.

- OS and other tools:
  - Mac OS X 10.6.8, Java 1.6.0_65, Homebrew, thrift 0.9.0
- Hadoop and HBase:
  - Hadoop 1.1.2, HBase 0.94.17
- R and RHadoop packages:
  - R 3.1.0, rhdfs 1.0.8, rmr2 3.1.0, plyrmr 0.2.0, rhbase 1.2.0

This process should work with Hadoop 2.2 or above and newer versions of HBase as well, but I haven't tested it yet.

Homebrew is a missing package manager for Mac OS X, and it is needed for install *git*, *pkg-config* and *thrift*. For other operating systems, the equivalents to Homebrew are *apt-get* on Ubuntu and *yum* on CentOS.

By the way, two painful steps in this process are setting up HBase on Hadoop in cluster mode and installing *rhbase*. If you want to have a quick start or are not going to use HBase, you donot need to intall *thrift*, HBase or *rhbase*, and therefore can skip

- step 3 - Install HBase,

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is (widely) supported by the world of science

IV R is supported by the community

V R is (increasingly) supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

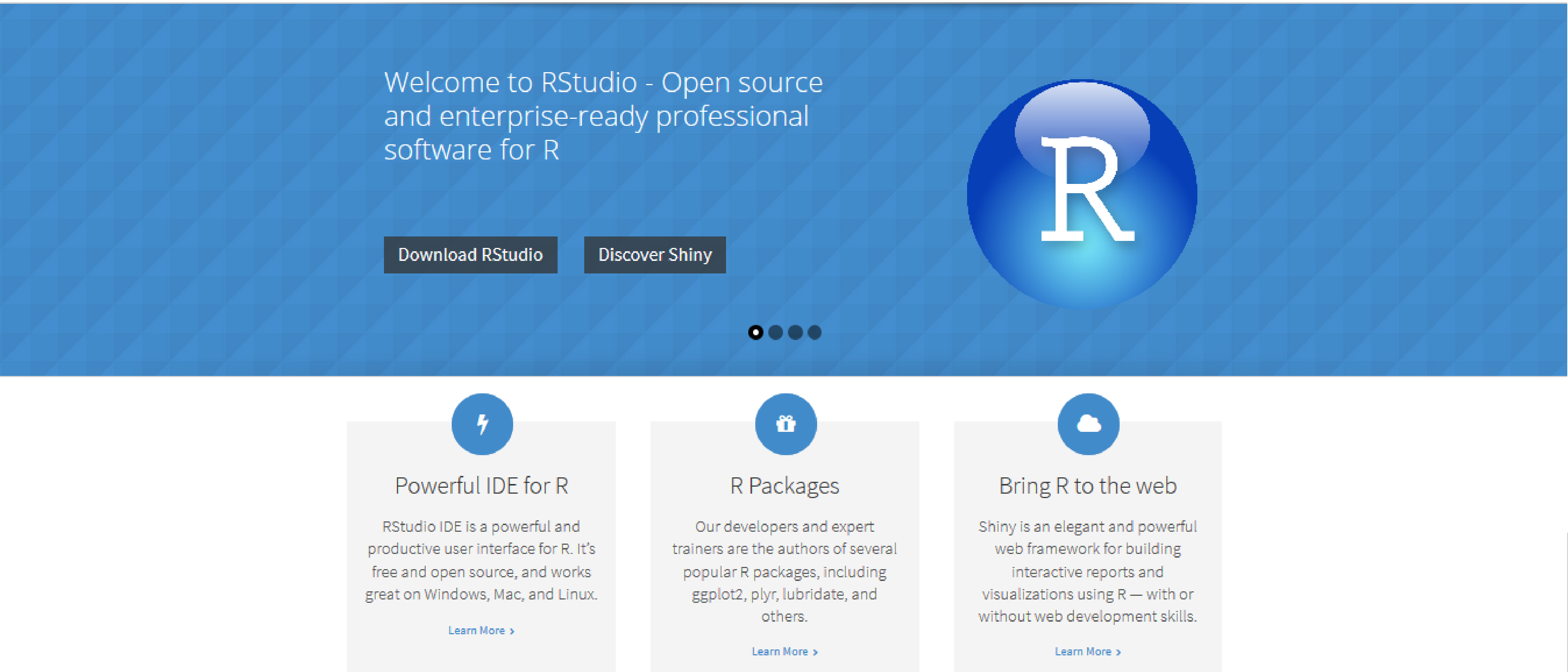
XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

Integrated Development Environments

1. RStudio

http://www.rstudio.com/



Welcome to RStudio - Open source and enterprise-ready professional software for R

Powerful IDE for R

RStudio IDE is a powerful and productive user interface for R. It's free and open source, and works great on Windows, Mac, and Linux.

Learn More ›

R Packages

Our developers and expert trainers are the authors of several popular R packages, including ggplot2, plyr, lubridate, and others.

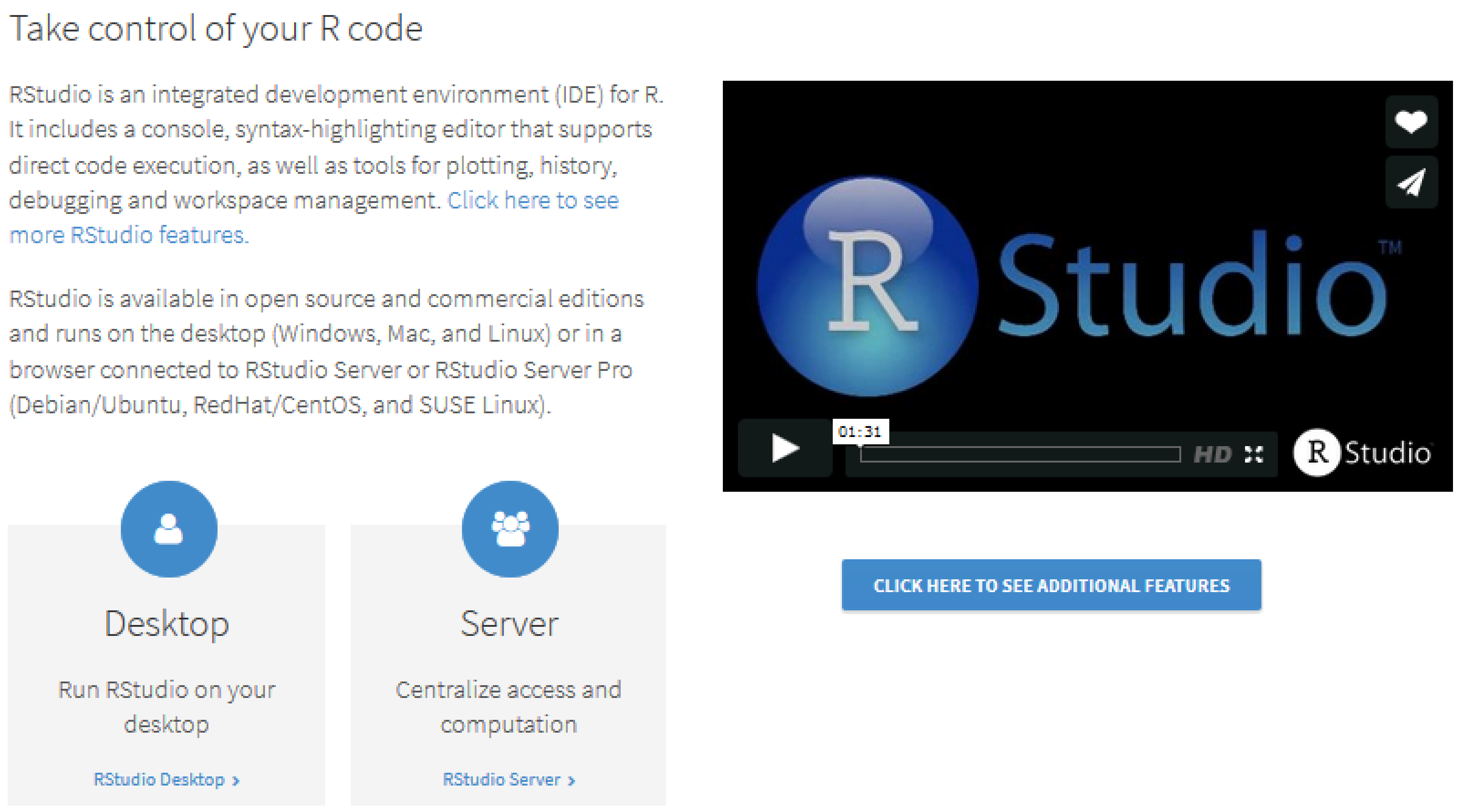
Learn More ›

Bring R to the web

Shiny is an elegant and powerful web framework for building interactive reports and visualizations using R — with or without web development skills.

Learn More ›

RStudio



Take control of your R code

RStudio is an integrated development environment (IDE) for R. It includes a console, syntax-highlighting editor that supports direct code execution, as well as tools for plotting, history, debugging and workspace management. Click here to see more RStudio features.

RStudio is available in open source and commercial editions and runs on the desktop (Windows, Mac, and Linux) or in a browser connected to RStudio Server or RStudio Server Pro (Debian/Ubuntu, RedHat/CentOS, and SUSE Linux).

Desktop

Run RStudio on your desktop

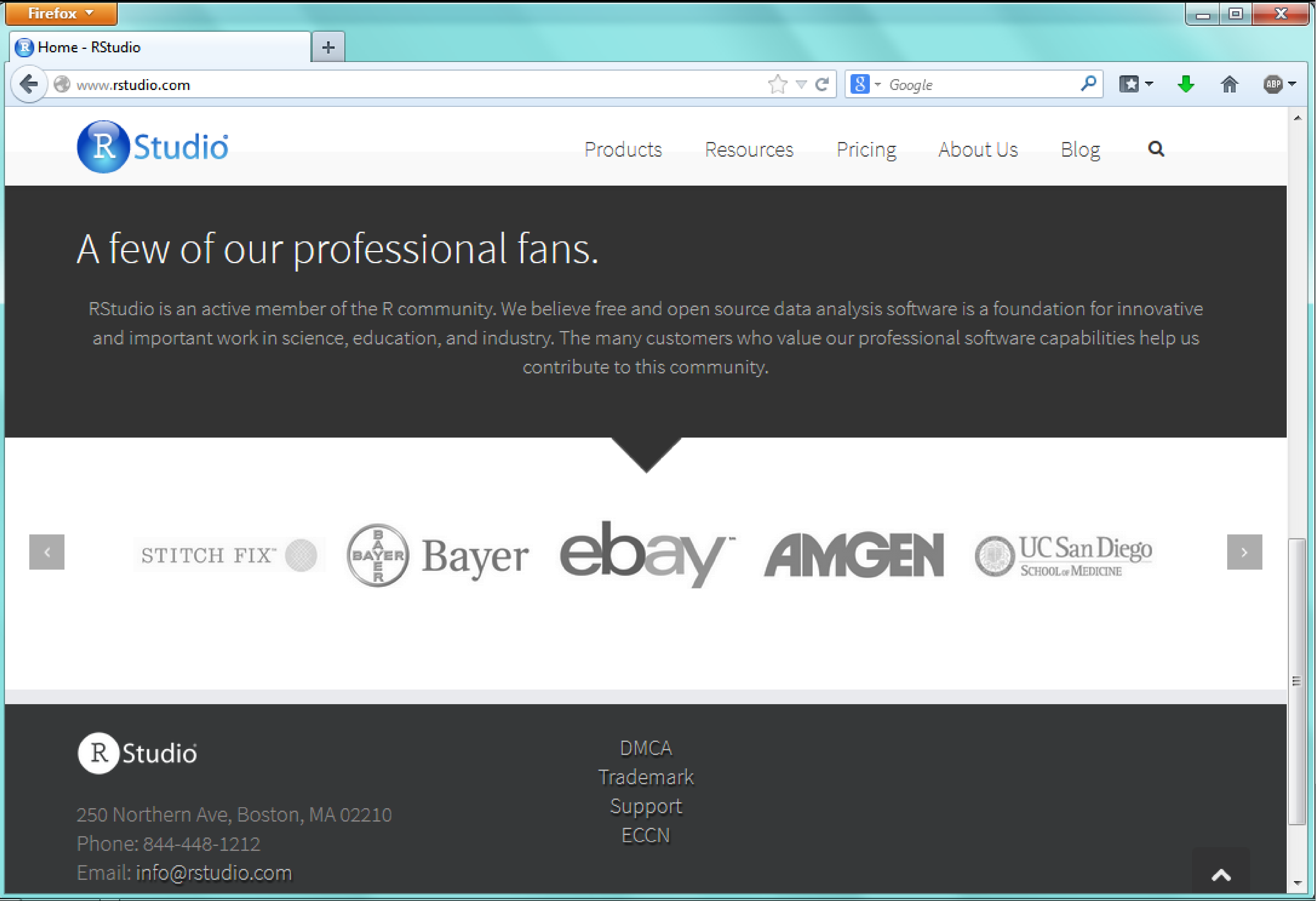
RStudio Desktop ›

Server

Centralize access and computation

RStudio Server ›

RStudio



A few of our professional fans.

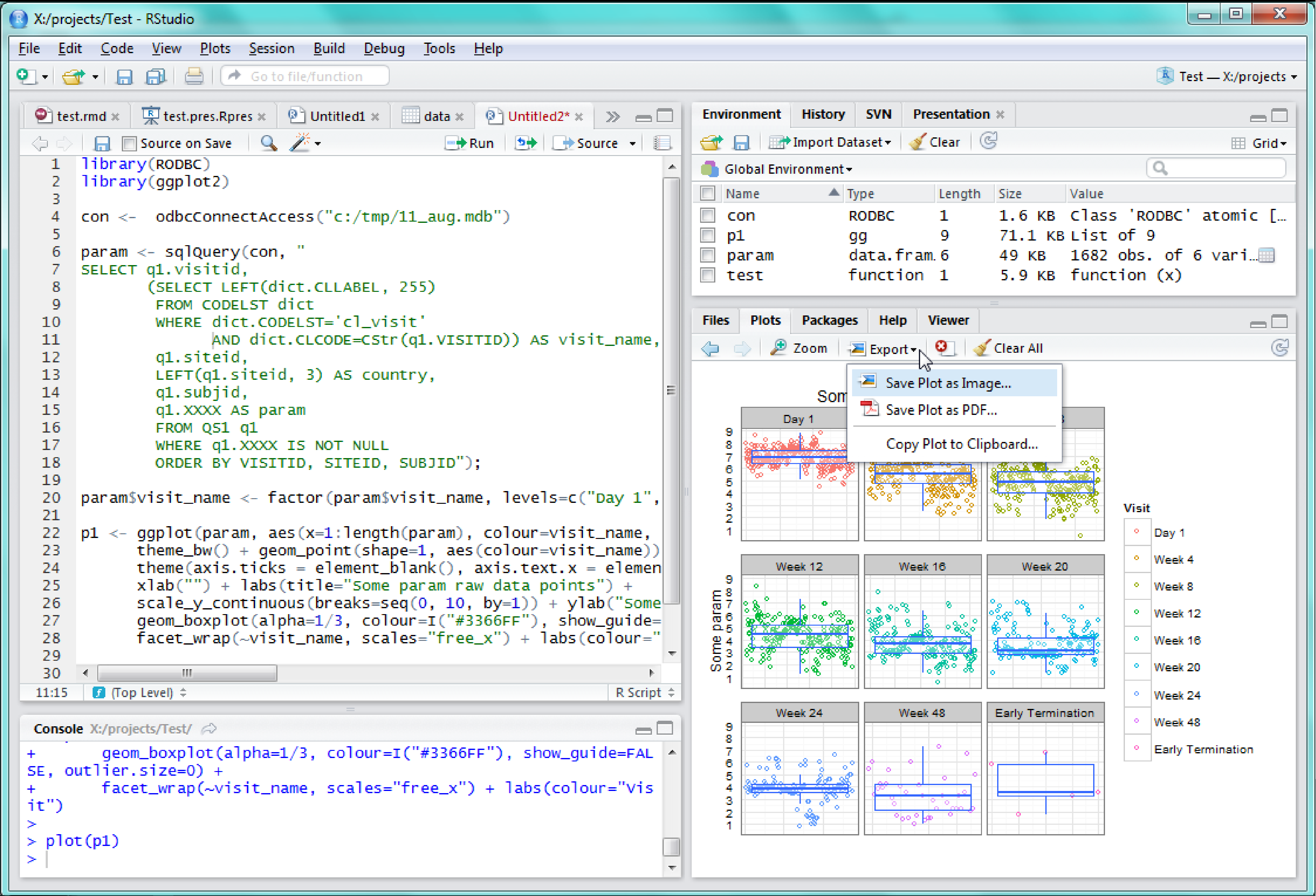
RStudio is an active member of the R community. We believe free and open source data analysis software is a foundation for innovative and important work in science, education, and industry. The many customers who value our professional software capabilities help us contribute to this community.

RStudio

250 Northern Ave, Boston, MA 02210
Phone: 844-448-1212
Email: info@rstudio.com

DMCA
Trademark
Support
ECCN

RStudio

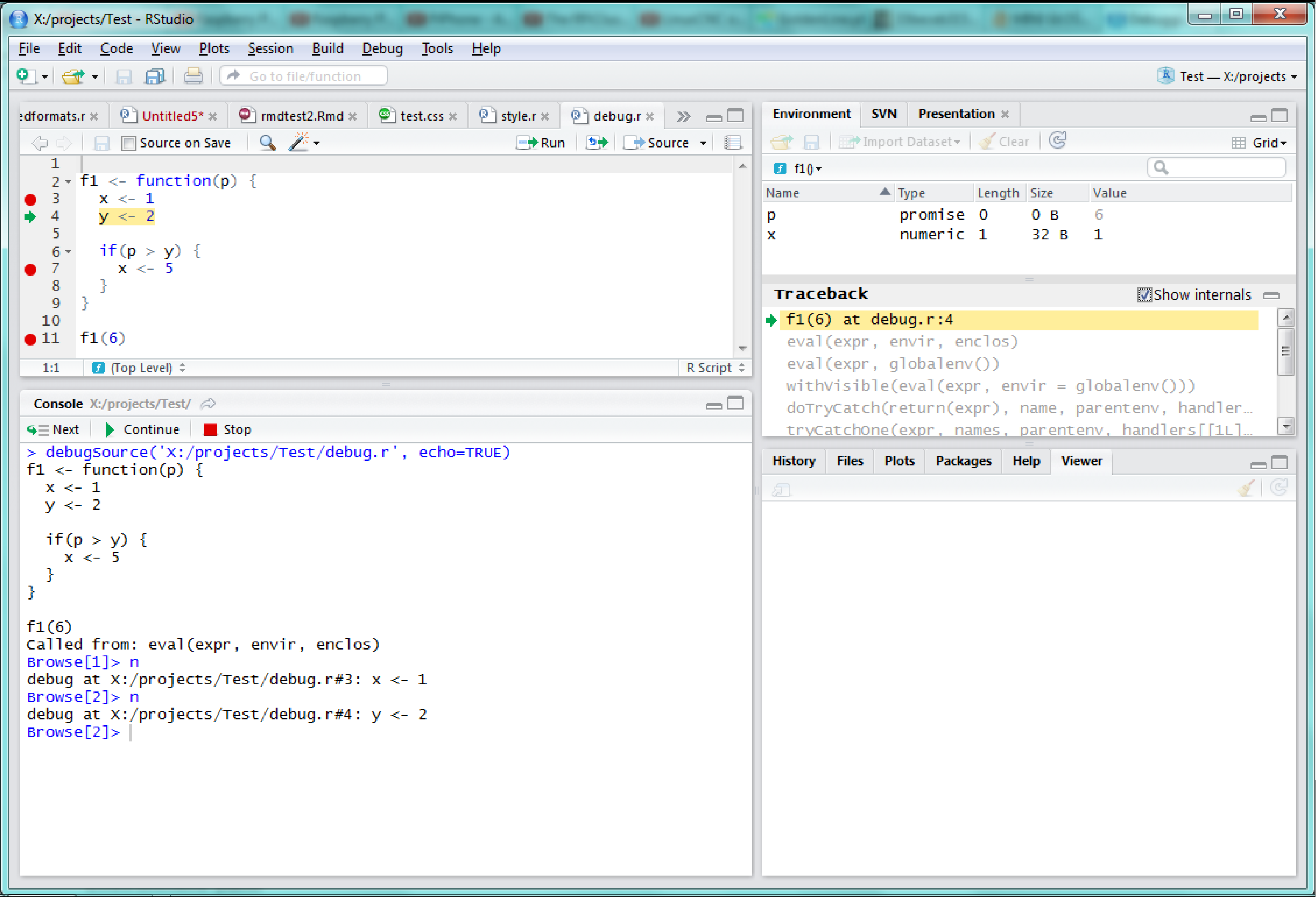


```
library(RODBC)
library(ggplot2)

con <-  odbcConnectAccess("c:/tmp/11_aug.mdb")

param <- sqlQuery(con, "
SELECT q1.visitid,
       (SELECT LEFT(dict.CLLABEL, 255)
        FROM CODELST dict
        WHERE dict.CODELST='cl_visit'
              AND dict.CLCODE=CStr(q1.VISITID)) AS visit_name,
        q1.siteid,
        LEFT(q1.siteid, 3) AS country,
        q1.subjid,
        q1.XXXX AS param
        FROM QS1 q1
        WHERE q1.XXXX IS NOT NULL
        ORDER BY VISITID, SITEID, SUBJID");

param$visit_name <- factor(param$visit_name, levels=c("Day 1",

p1 <- ggplot(param, aes(x=1:length(param), colour=visit_name,
      theme_bw() + geom_point(shape=1, aes(colour=visit_name))
      theme(axis.ticks = element_blank(), axis.text.x = elemen
      xlab("") + labs(title="Some param raw data points") +
      scale_y_continuous(breaks=seq(0, 10, by=1)) + ylab("Some
      geom_boxplot(alpha=1/3, colour=I("#3366FF"), show_guide=
      facet_wrap(~visit_name, scales="free_x") + labs(colour="


```

☐	Name	Type	Length	Size	Value
☐	con	RODBC	1	1.6 KB	Class 'RODBC' atomic [...
☐	p1	gg	9	71.1 KB	List of 9
☐	param	data.fram.	6	49 KB	1682 obs. of 6 vari...
☐	test	function	1	5.9 KB	function (x)

```
+       geom_boxplot(alpha=1/3, colour=I("#3366FF"), show_guide=FAL
SE, outlier.size=0) +
+       facet_wrap(~visit_name, scales="free_x") + labs(colour="vis
it")
>
> plot(p1)
>
```

RStudio – debugging



```
f1 <- function(p) {
  x <- 1
  y <- 2

  if(p > y) {
    x <- 5
  }
}

f1(6)
```

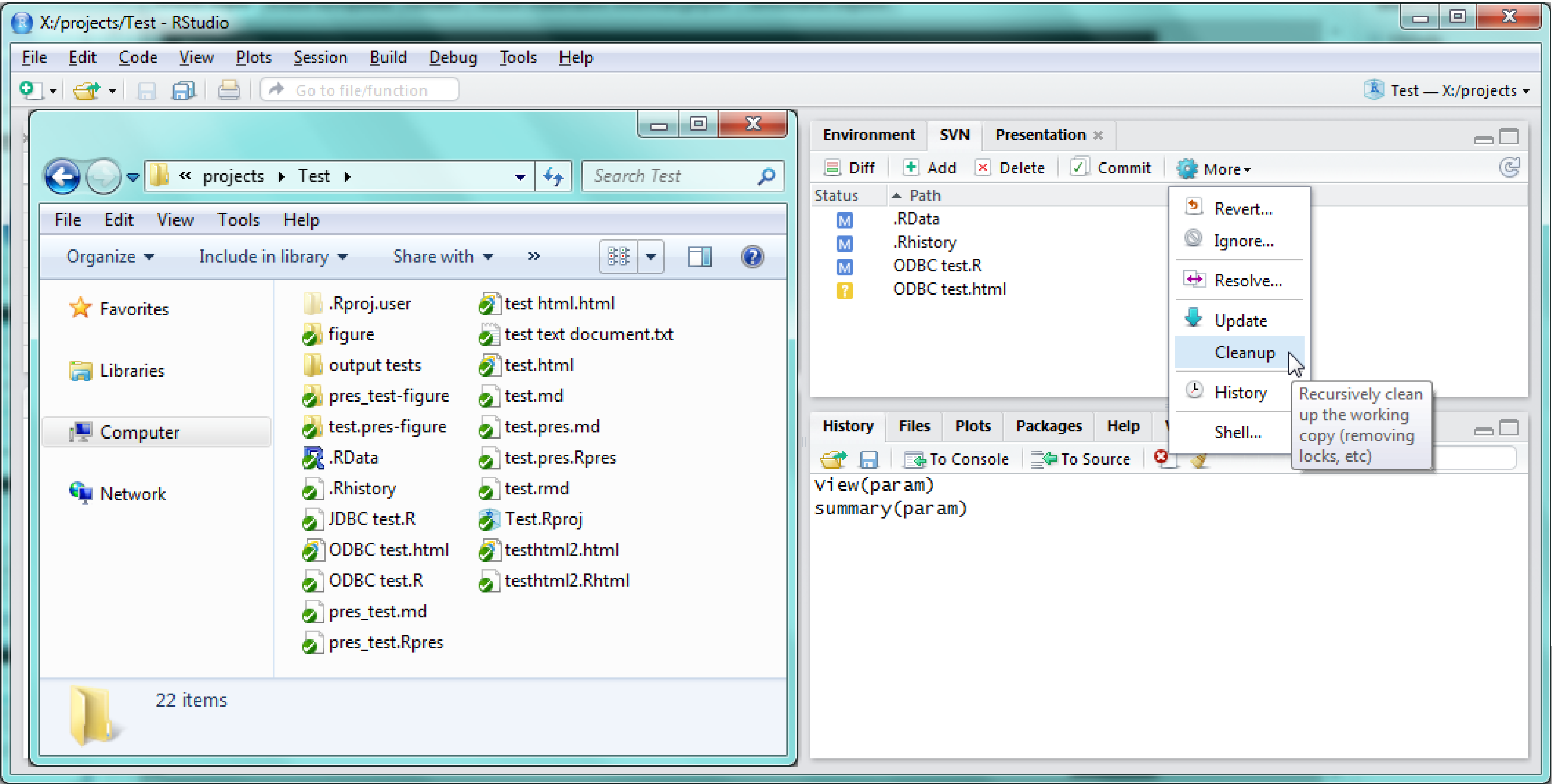
Name	Type	Length	Size	Value
p	promise	0	0 B	6
x	numeric	1	32 B	1

```
f1(6) at debug.r:4
eval(expr, envir, enclos)
eval(expr, globalenv())
withVisible(eval(expr, envir = globalenv()))
doTryCatch(return(expr), name, parentenv, handler…
tryCatchOne(expr, names, parentenv, handlers[[1L]…
```

```
> debugSource('X:/projects/Test/debug.r', echo=TRUE)
f1 <- function(p) {
  x <- 1
  y <- 2

  if(p > y) {
    x <- 5
  }
}

f1(6)
Called from: eval(expr, envir, enclos)
Browse[1]> n
debug at X:/projects/Test/debug.r#3: x <- 1
Browse[2]> n
debug at X:/projects/Test/debug.r#4: y <- 2
Browse[2]>
```

RStudio – version control

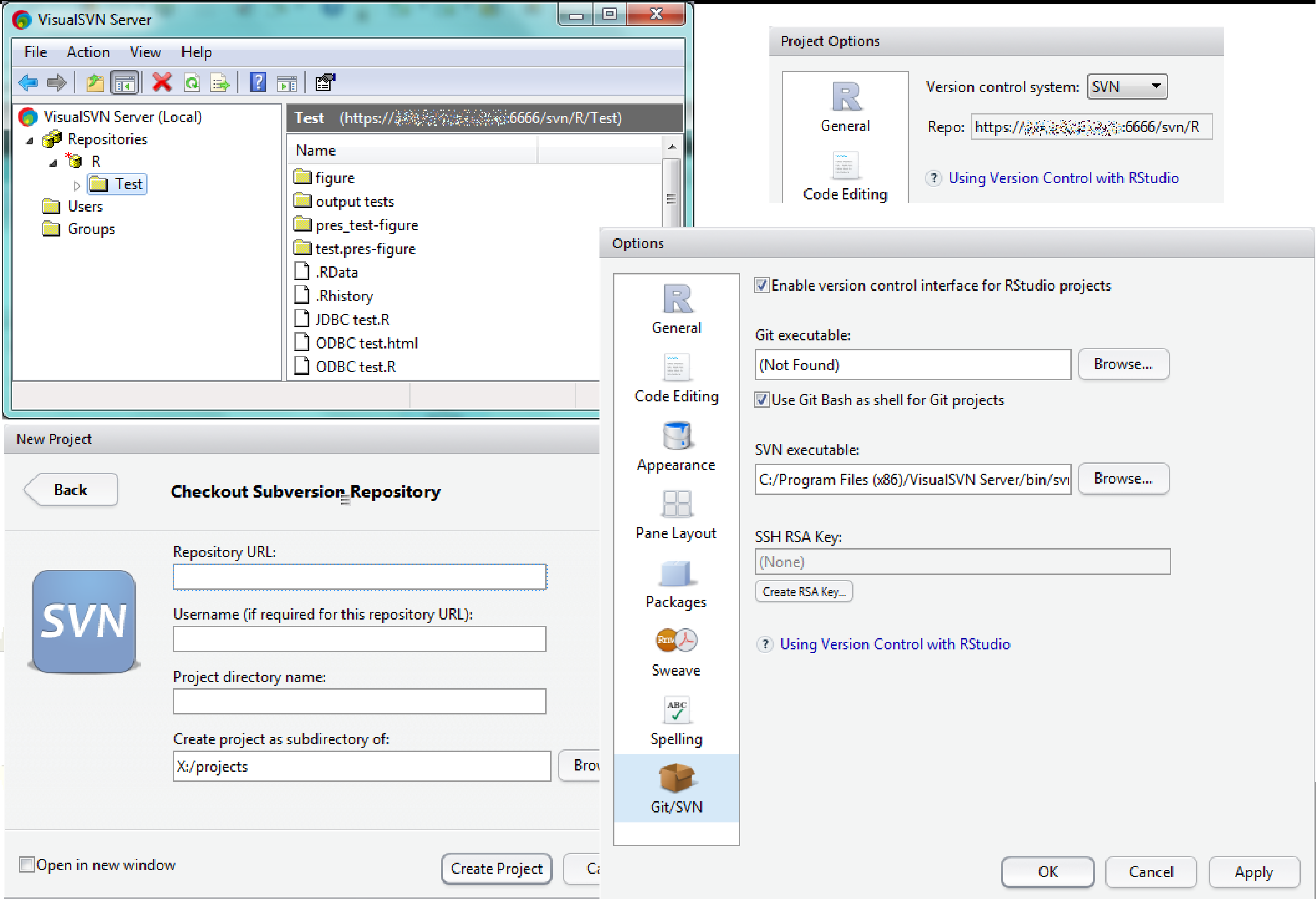
RStudio integrates seamlessly with version control systems like Git or Subversion (SVN). **Version control** is one of the major requirements for documentation management in clinical research.



Recursively clean up the working copy (removing locks, etc)

```
view(param)
summary(param)
```

RStudio – version control



Test (https://[redacted]:6666/svn/R/Test)

Name
- figure
- output tests
- pres_test-figure
- test.pres-figure
- .RData
- .Rhistory
- JDBC test.R
- ODBC test.html
- ODBC test.R

Project Options

Version control system: SVN

Repo: https://[redacted]:6666/svn/R

New Project

Checkout Subversion Repository

Repository URL:

Username (if required for this repository URL):

Project directory name:

Create project as subdirectory of: X:/projects

Options

☑ Enable version control interface for RStudio projects

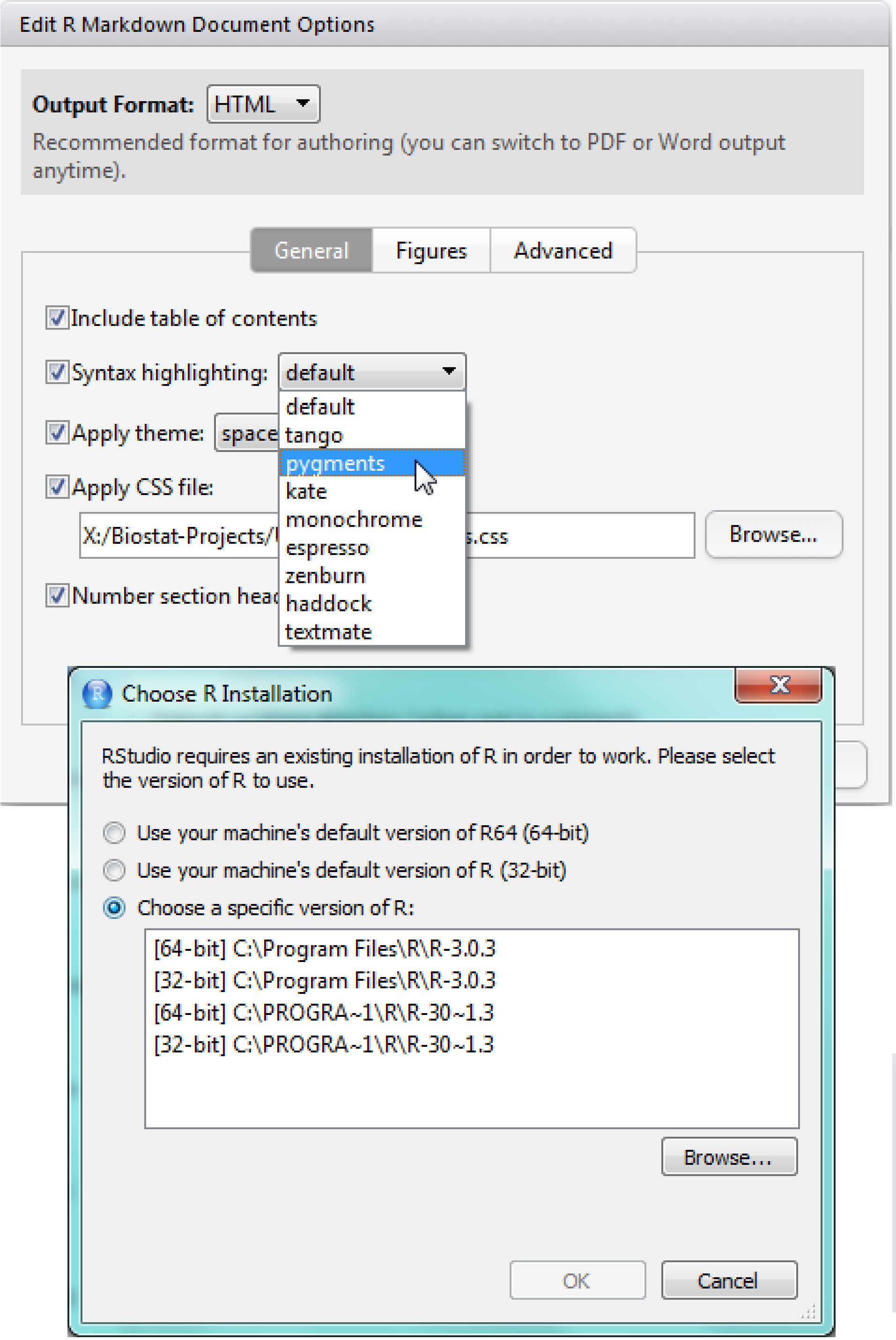
Git executable: (Not Found)

☑ Use Git Bash as shell for Git projects

SVN executable: C:/Program Files (x86)/VisualSVN Server/bin/svı

SSH RSA Key: (None)

RStudio – some taste…


Edit R Markdown Document Options

Output Format: HTML

Recommended format for authoring (you can switch to PDF or Word output anytime).

☑ Include table of contents

☑ Syntax highlighting: default

☑ Apply theme: space

☑ Apply CSS file: X:/Biostat-Projects/ .css

☑ Number section hea

Choose R Installation

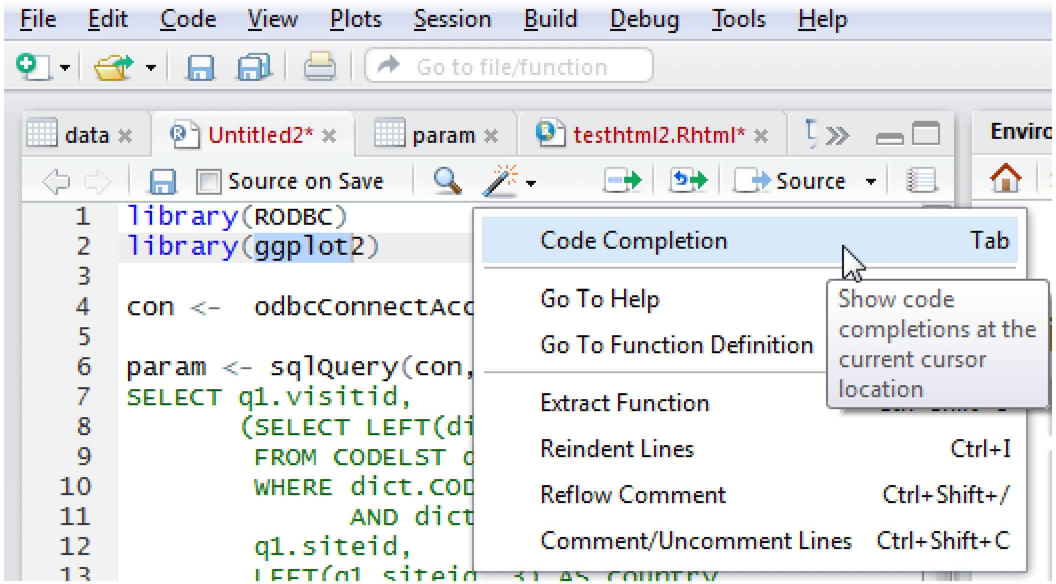
RStudio requires an existing installation of R in order to work. Please select the version of R to use.

○ Use your machine's default version of R64 (64-bit)

○ Use your machine's default version of R (32-bit)

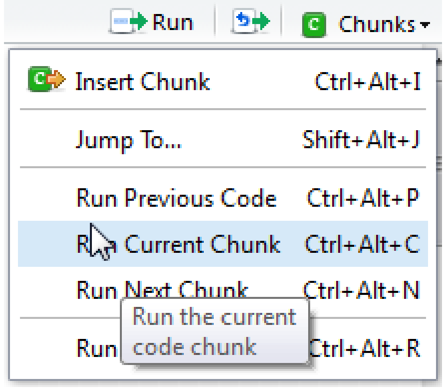
● Choose a specific version of R:

```
[64-bit] C:\Program Files\R\R-3.0.3
[32-bit] C:\Program Files\R\R-3.0.3
[64-bit] C:\PROGRA~1\R\R-30~1.3
[32-bit] C:\PROGRA~1\R\R-30~1.3
```

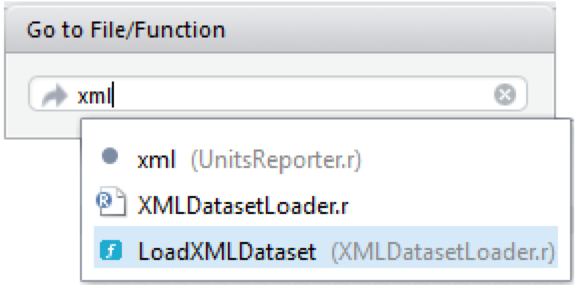


Show code completions at the current cursor location

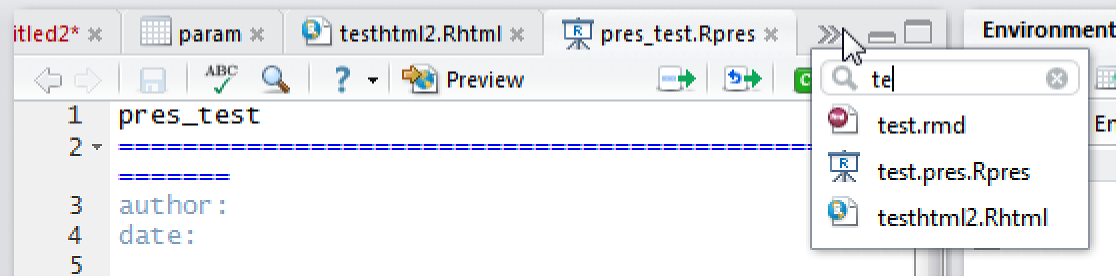
```
library(RODBC)
library(ggplot2)

con <-  odbcConnectAcc

param <- sqlQuery(con,
SELECT q1.visitid,
       (SELECT LEFT(di
        FROM CODELST c
        WHERE dict.COD
           AND dict
        q1.siteid,
        LEFT(q1.siteid, 3) AS country
```



Run the current code chunk

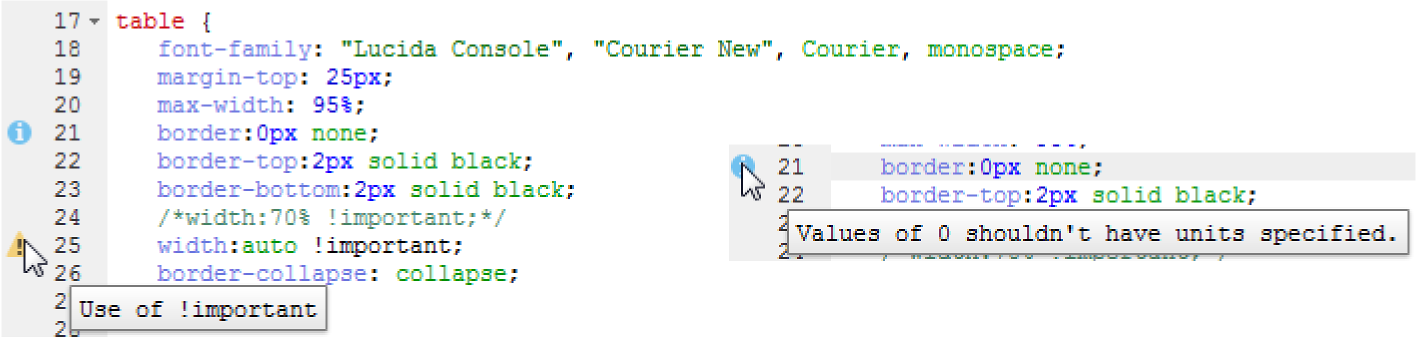

Go to File/Function

xml



```
pres_test
========================================
=======
author:
date:
```

RStudio – syntax and validation

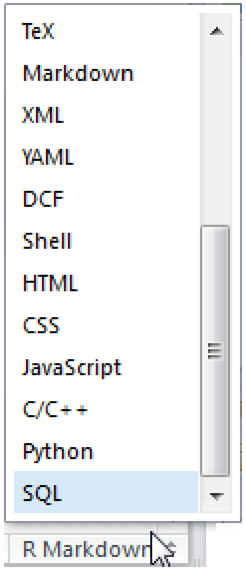


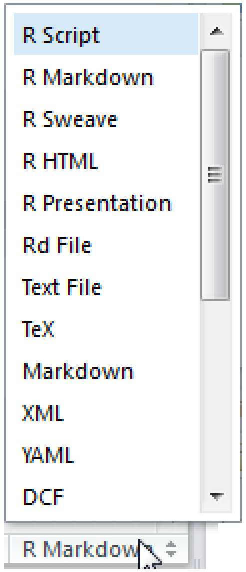
```
table {
   font-family: "Lucida Console", "Courier New", Courier, monospace;
   margin-top: 25px;
   max-width: 95%;
   border:0px none;
   border-top:2px solid black;
   border-bottom:2px solid black;
   /*width:70% !important;*/
   width:auto !important;
   border-collapse: collapse;
```

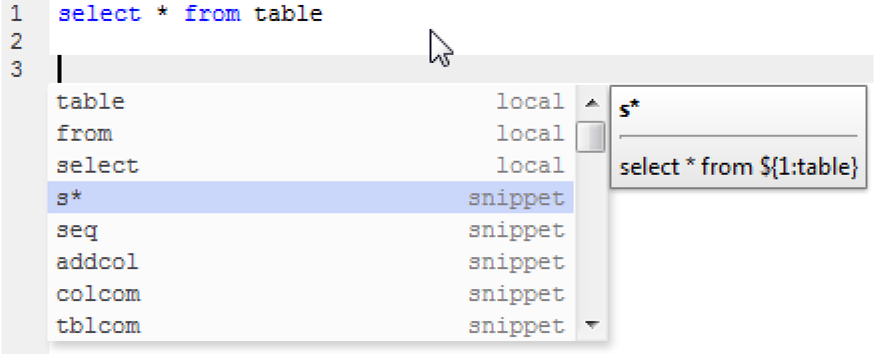
Use of !important

```
   border:0px none;
   border-top:2px solid black;
```

Values of 0 shouldn't have units specified.



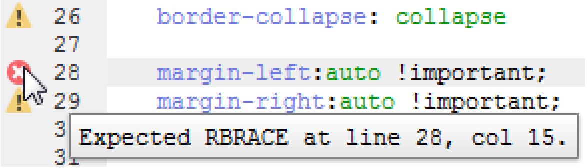




```
select * from table
```

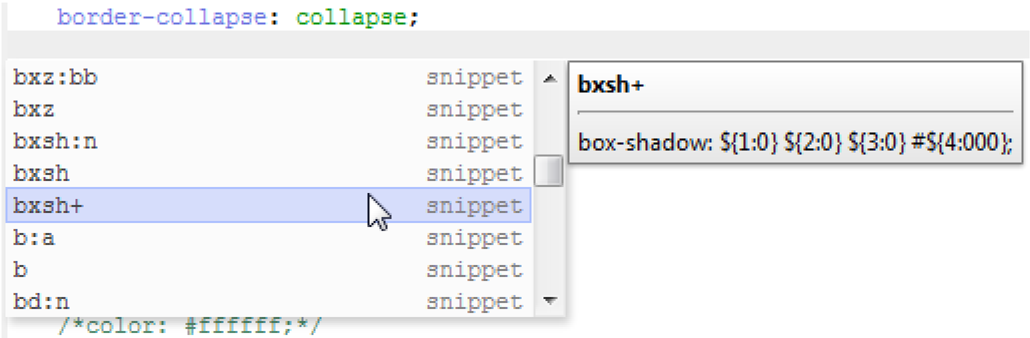
s*

select * from ${1:table}



```
   border-collapse: collapse

   margin-left:auto !important;
   margin-right:auto !important;
```

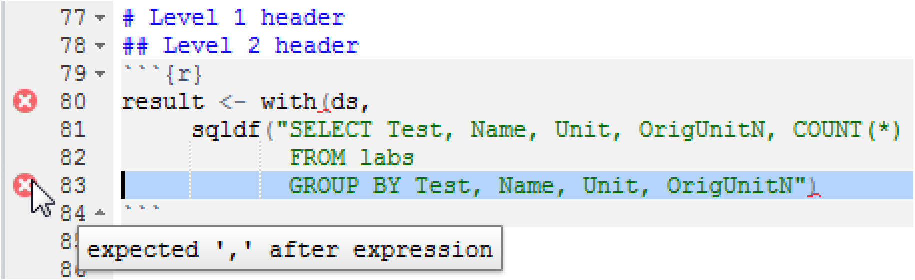
Expected RBRACE at line 28, col 15.



```
   border-collapse: collapse;
   /*color: #ffffff;*/
```

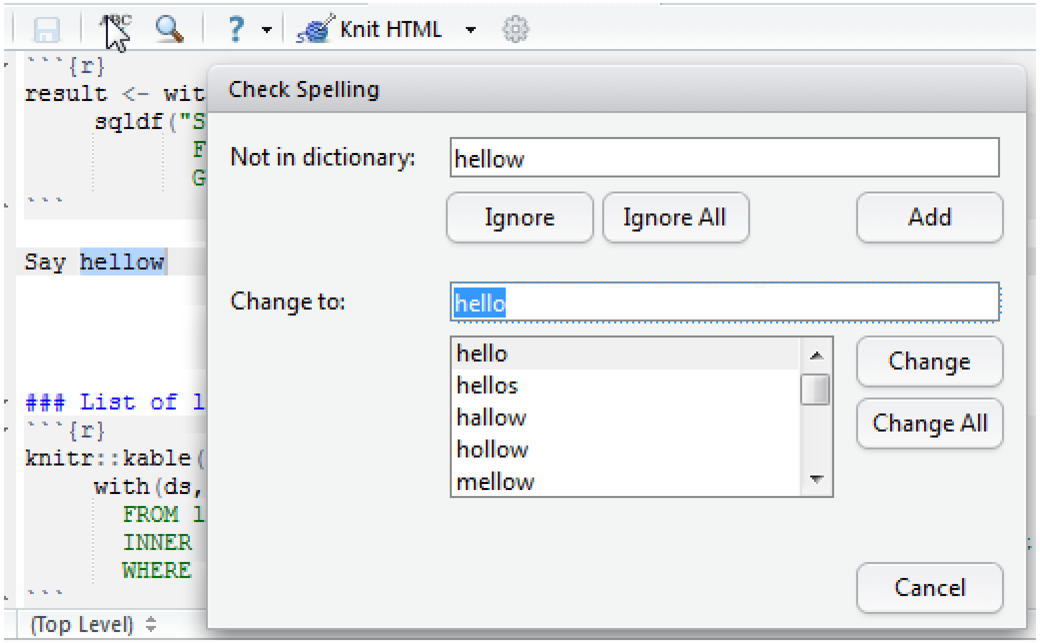
bxsh+

box-shadow: ${1:0} ${2:0} ${3:0} #${4:000};



~~~
# Level 1 header
## Level 2 header
```{r}
result <- with(ds,
     sqldf("SELECT Test, Name, Unit, OrigUnitN, COUNT(*)
            FROM labs
            GROUP BY Test, Name, Unit, OrigUnitN")
```
~~~

expected ',' after expression



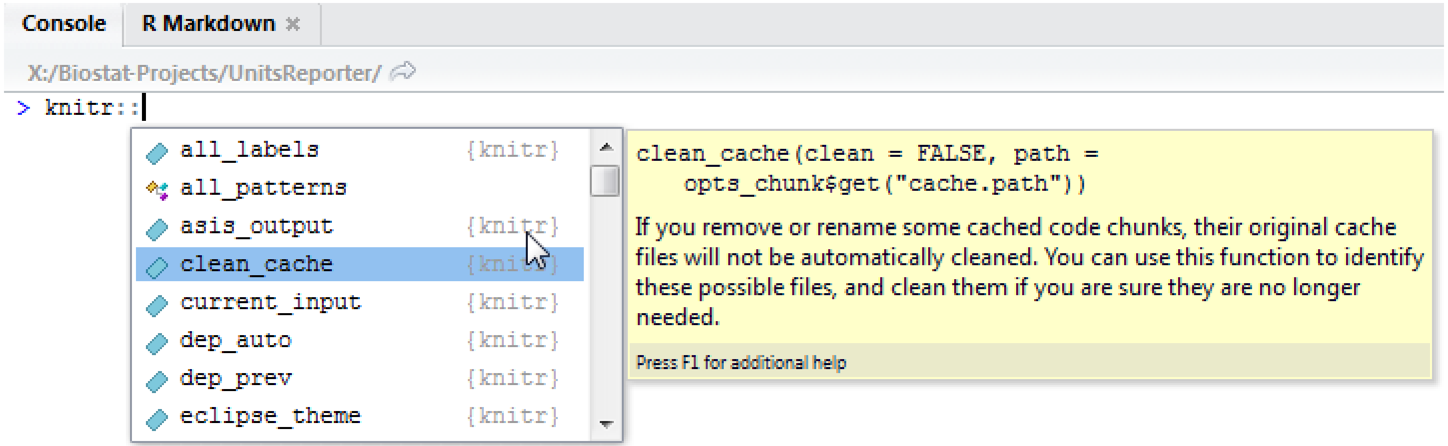
~~~
```{r}
result <- wit
     sqldf("S
            F
            G
```

Say hellow

### List of l
```{r}
knitr::kable(
     with(ds,
       FROM l
       INNER
       WHERE
```
~~~

Check Spelling

Not in dictionary: hellow

Change to: hello

RStudio - autocompletion

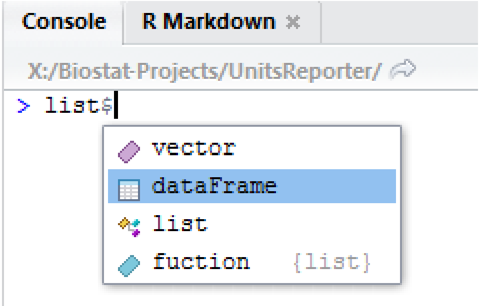


```
> knitr::
```

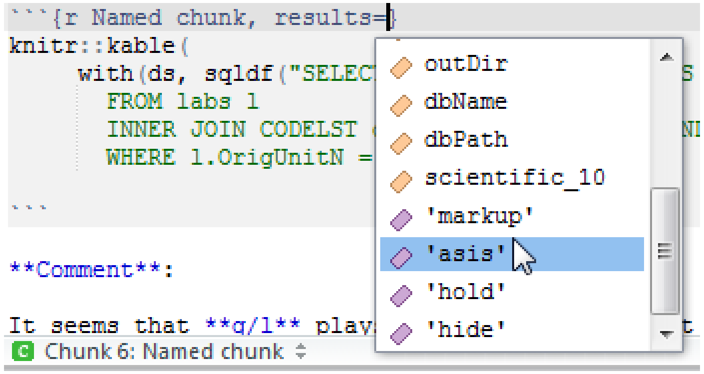
```
clean_cache(clean = FALSE, path =
    opts_chunk$get("cache.path"))
```

If you remove or rename some cached code chunks, their original cache files will not be automatically cleaned. You can use this function to identify these possible files, and clean them if you are sure they are no longer needed.

Press F1 for additional help

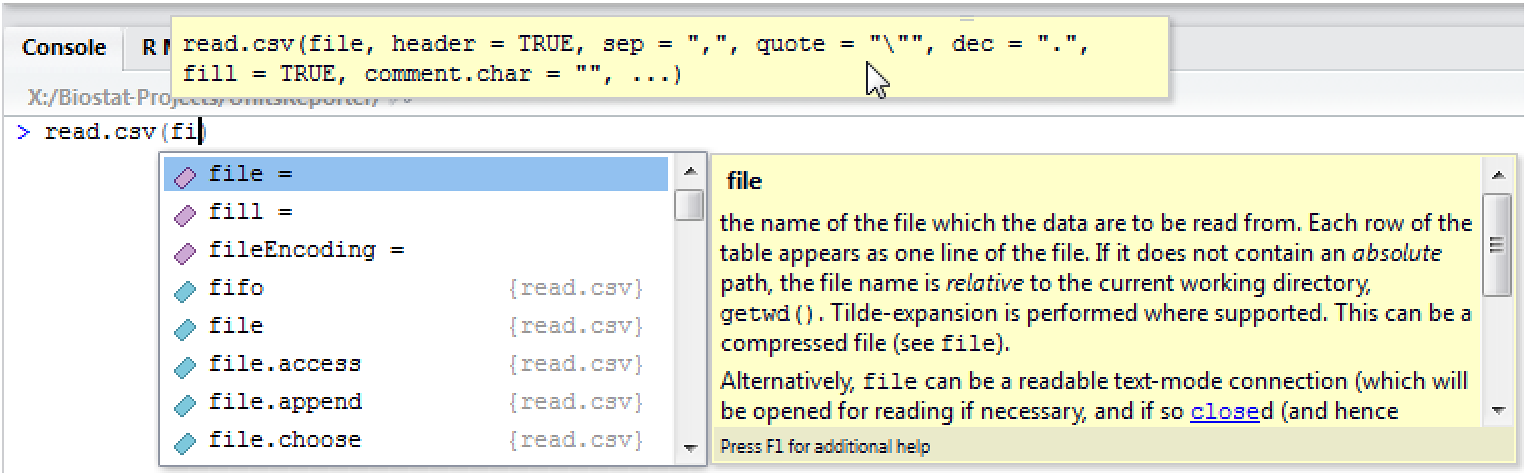


```
> list$
```

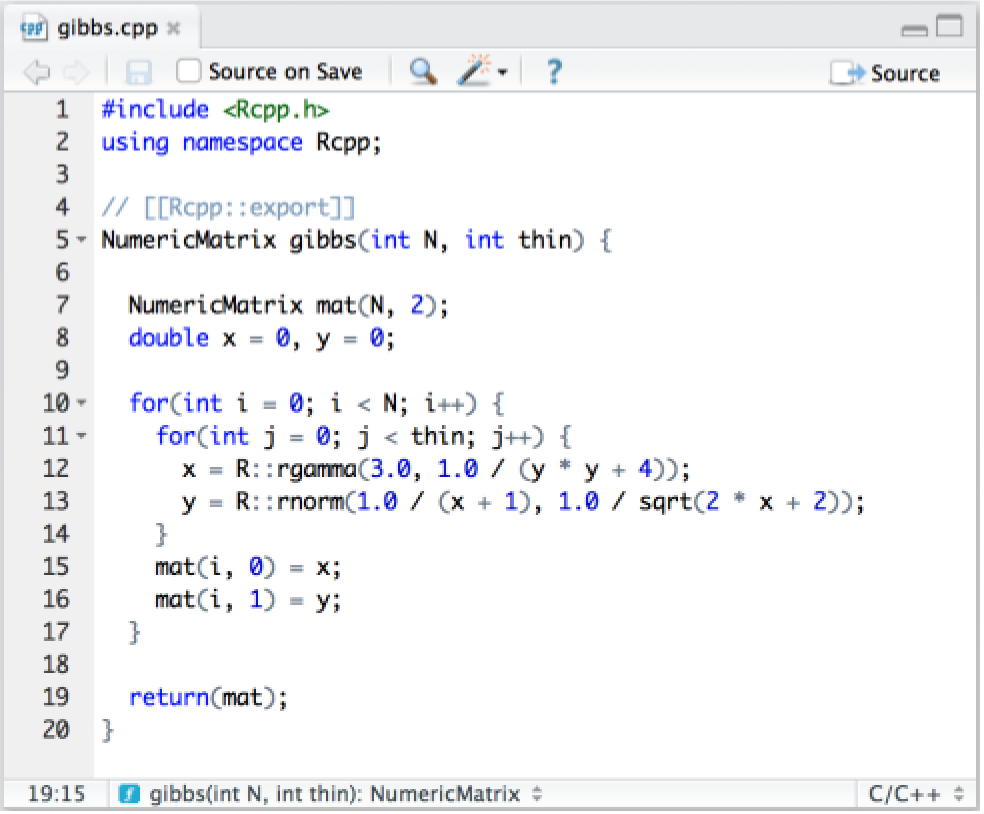


~~~
```{r Named chunk, results=}
knitr::kable(
     with(ds, sqldf("SELEC
       FROM labs l
       INNER JOIN CODELST
       WHERE l.OrigUnitN =

~~~

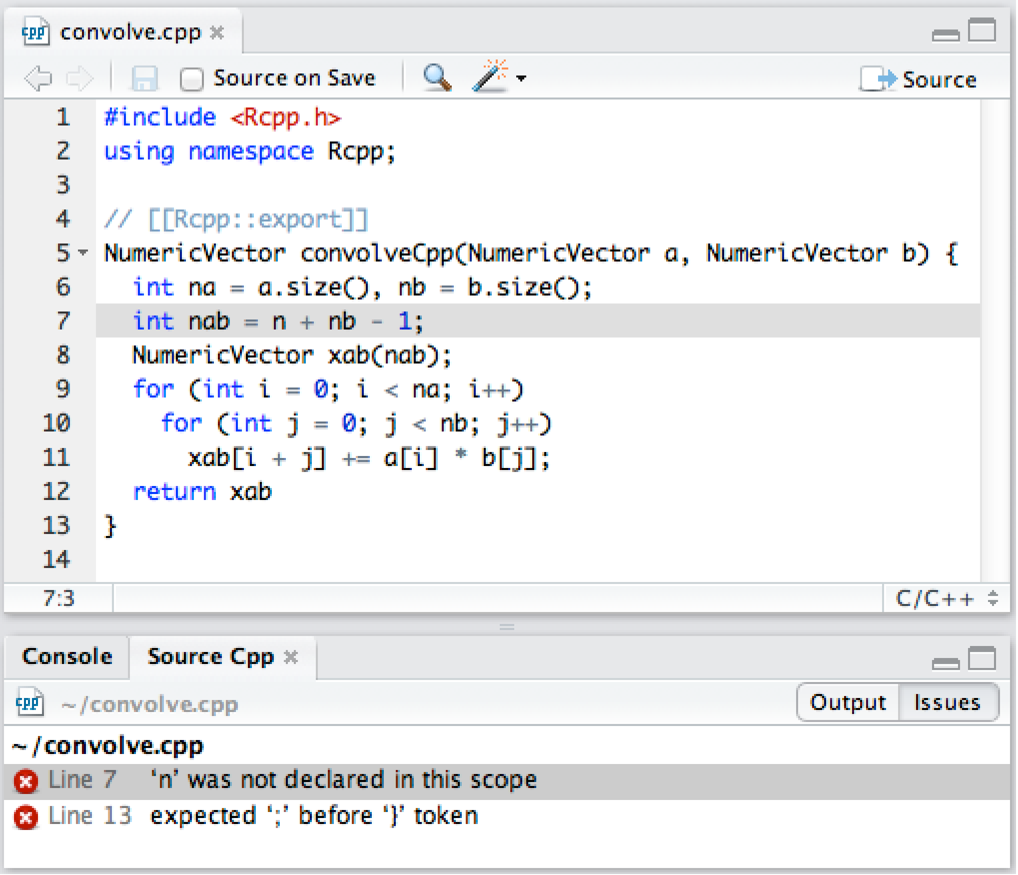
Comment:

It seems that **g/l** play
```


Console
R
X:/Biostat-Pro
file =
fill =
fileEncoding =
fifo {read.csv}
file {read.csv}
file.access {read.csv}
file.append {read.csv}
file.choose {read.csv}

```
read.csv(file, header = TRUE, sep = ",", quote = "\"", dec = ".",
fill = TRUE, comment.char = "", ...)
```

```
> read.csv(fi)
```

file

the name of the file which the data are to be read from. Each row of the table appears as one line of the file. If it does not contain an *absolute* path, the file name is *relative* to the current working directory, `getwd()`. Tilde-expansion is performed where supported. This can be a compressed file (see `file`).

Alternatively, `file` can be a readable text-mode connection (which will be opened for reading if necessary, and if so closed (and hence

Press F1 for additional help
```

RStudio – support for Rcpp



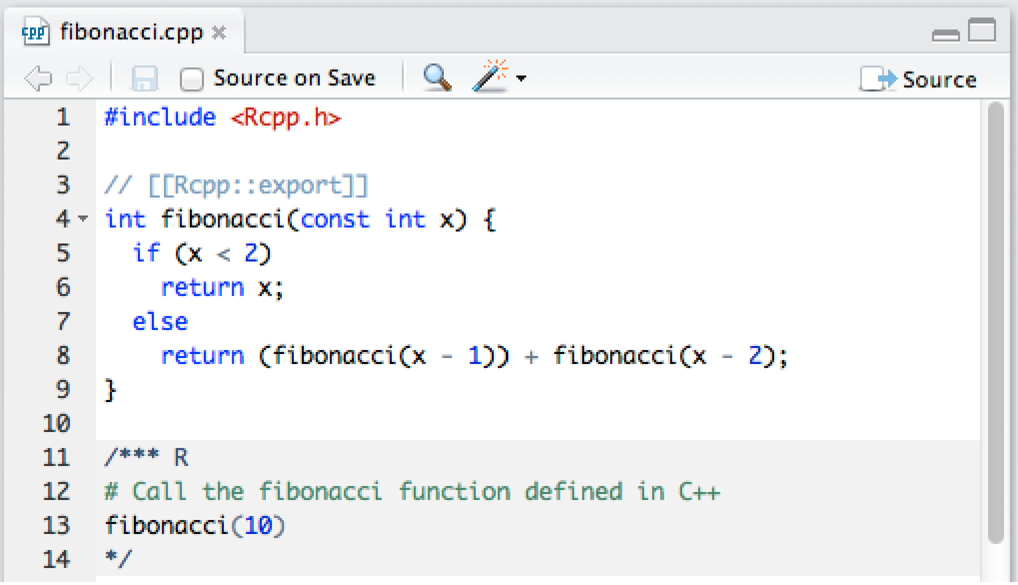
```cpp
#include <Rcpp.h>
using namespace Rcpp;

// [[Rcpp::export]]
NumericMatrix gibbs(int N, int thin) {

  NumericMatrix mat(N, 2);
  double x = 0, y = 0;

  for(int i = 0; i < N; i++) {
    for(int j = 0; j < thin; j++) {
      x = R::rgamma(3.0, 1.0 / (y * y + 4));
      y = R::rnorm(1.0 / (x + 1), 1.0 / sqrt(2 * x + 2));
    }
    mat(i, 0) = x;
    mat(i, 1) = y;
  }

  return(mat);
}
```



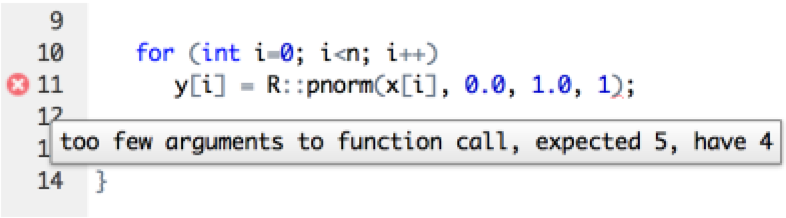
```cpp
 1  #include <Rcpp.h>
 2  using namespace Rcpp;
 3
 4  // [[Rcpp::export]]
 5  NumericVector convolveCpp(NumericVector a, NumericVector b) {
 6    int na = a.size(), nb = b.size();
 7    int nab = n + nb - 1;
 8    NumericVector xab(nab);
 9    for (int i = 0; i < na; i++)
10      for (int j = 0; j < nb; j++)
11        xab[i + j] += a[i] * b[j];
12    return xab
13  }
14
```

~/convolve.cpp

- ☒ Line 7 'n' was not declared in this scope
- ☒ Line 13 expected ';' before '}' token

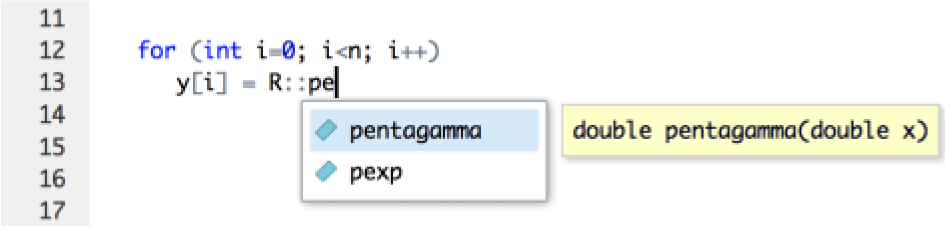


```cpp
#include <Rcpp.h>

// [[Rcpp::export]]
int fibonacci(const int x) {
  if (x < 2)
    return x;
  else
    return (fibonacci(x - 1)) + fibonacci(x - 2);
}

/*** R
# Call the fibonacci function defined in C++
fibonacci(10)
*/
```



```cpp
    for (int i=0; i<n; i++)
      y[i] = R::pnorm(x[i], 0.0, 1.0, 1);
}
```

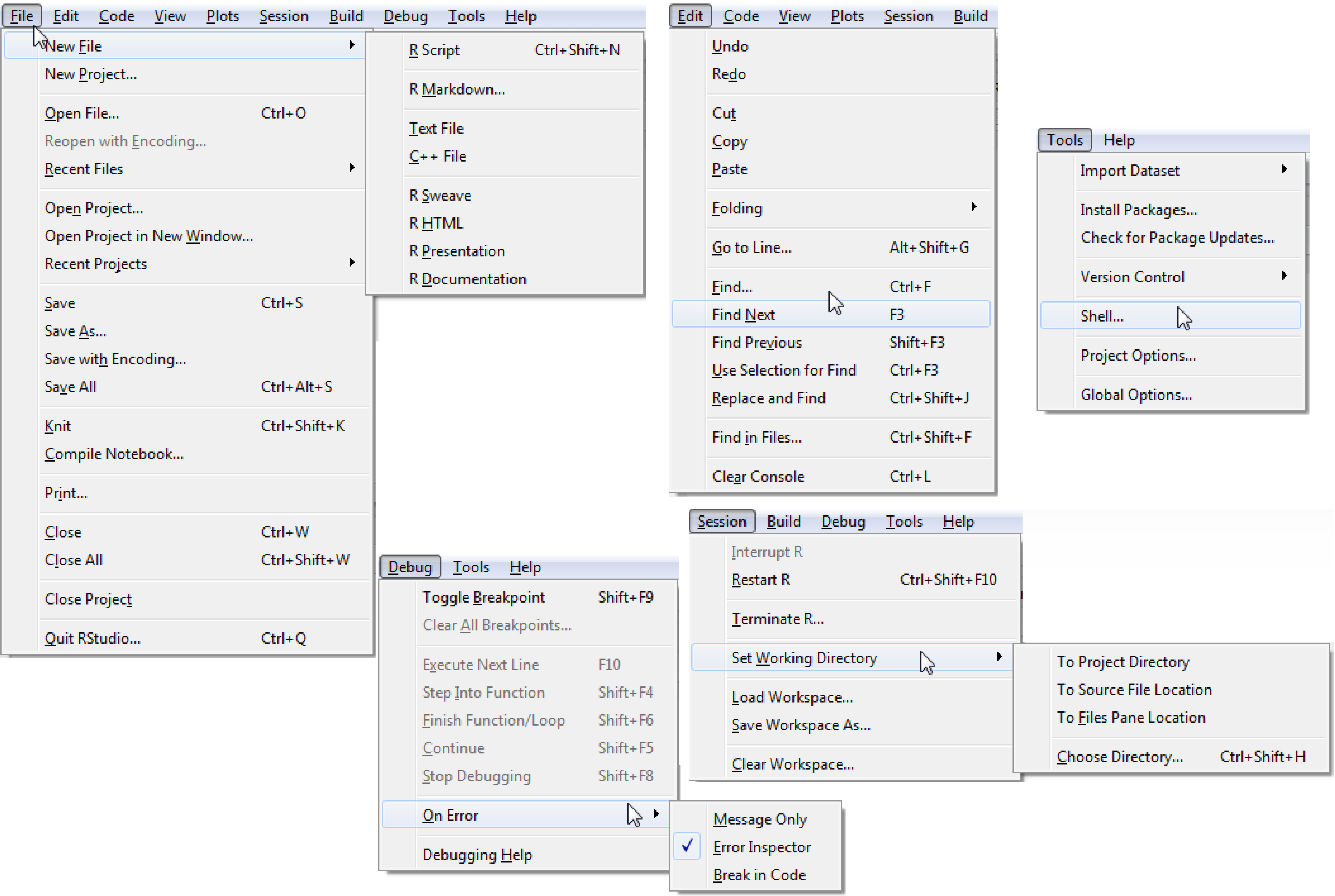
too few arguments to function call, expected 5, have 4



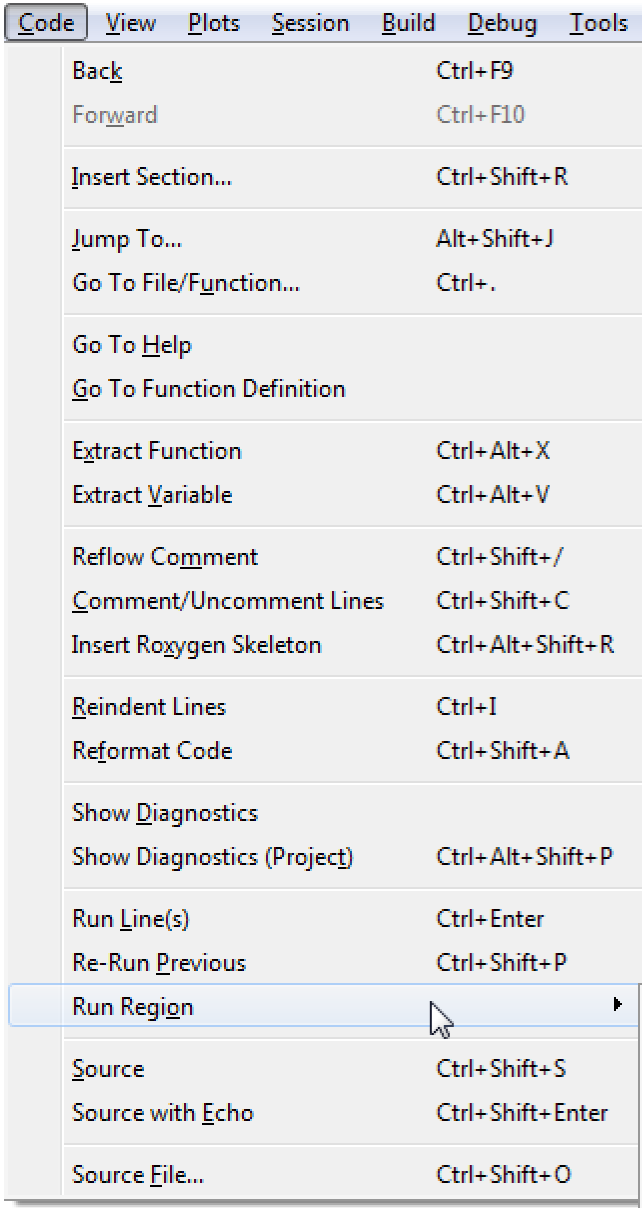
```cpp
    for (int i=0; i<n; i++)
      y[i] = R::pe
```

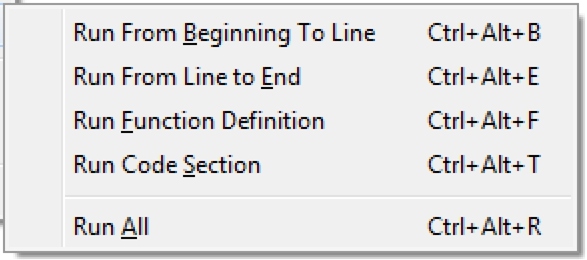
double pentagamma(double x)

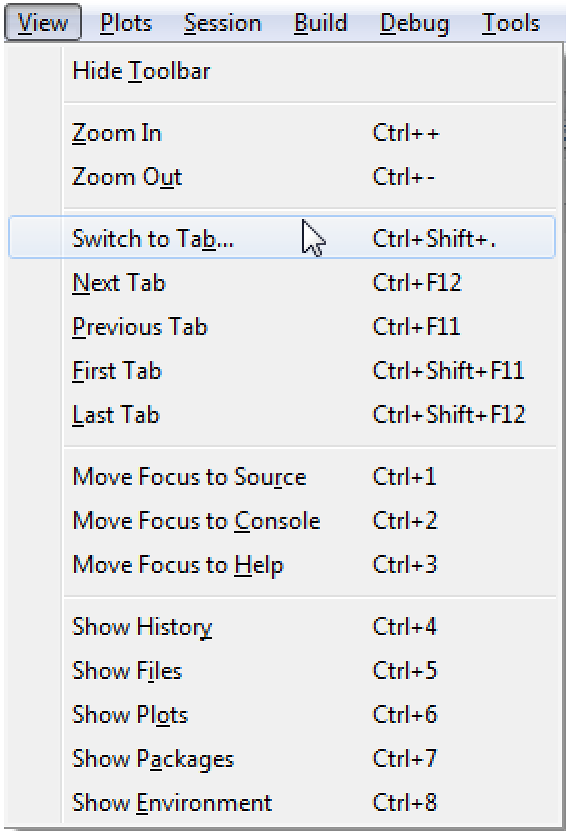
RStudio - menu

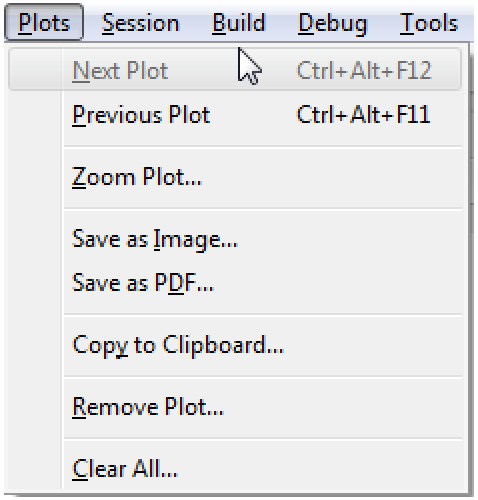


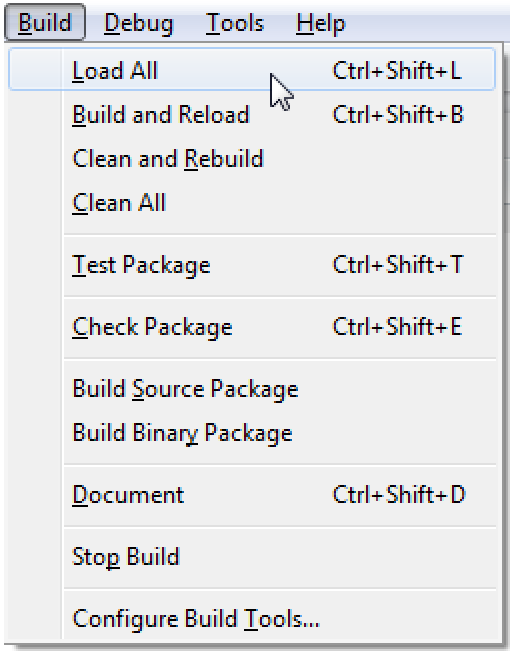
RStudio - menu



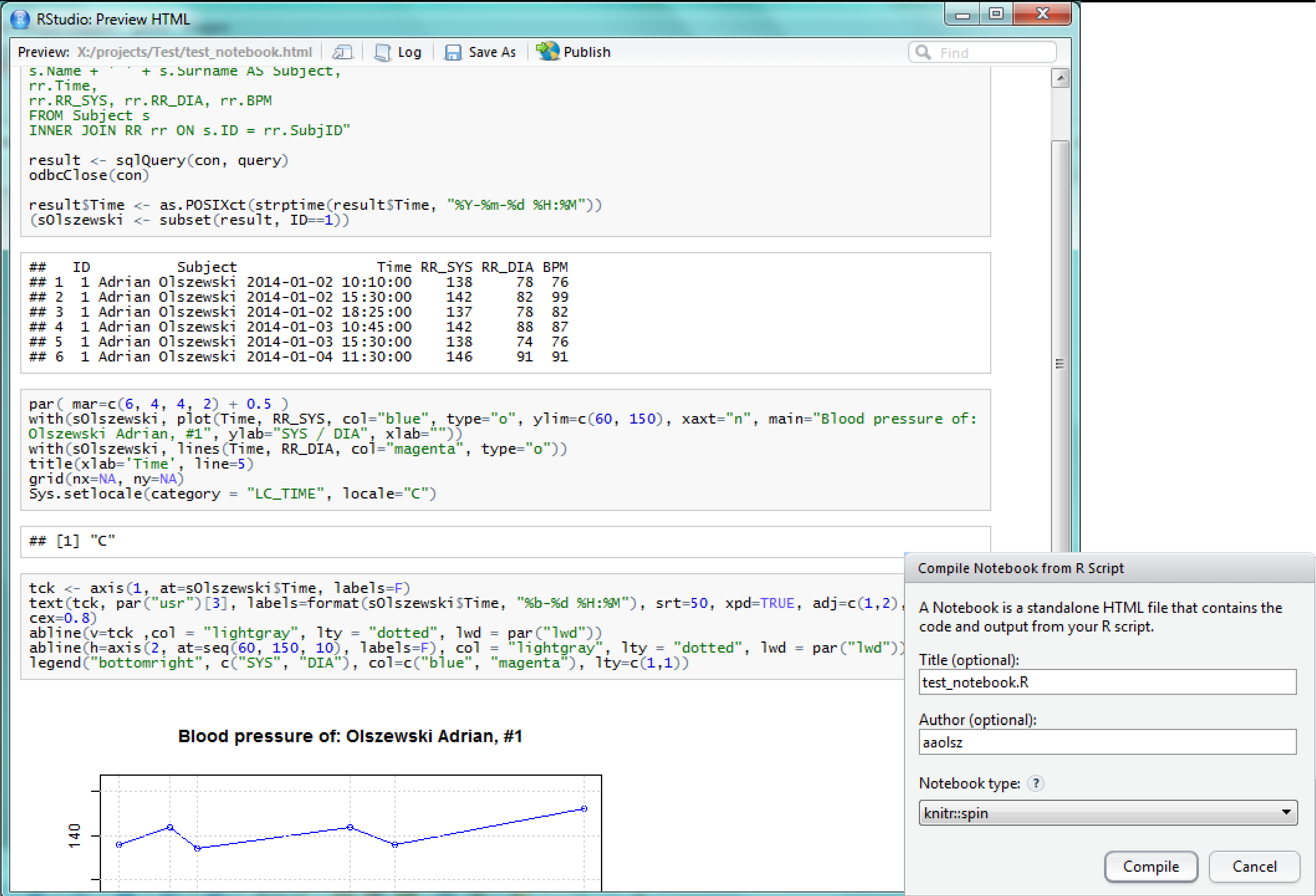








RStudio – the notebook (RMarkdown)



```
s.Name + ' ' + s.Surname AS Subject,
rr.Time,
rr.RR_SYS, rr.RR_DIA, rr.BPM
FROM Subject s
INNER JOIN RR rr ON s.ID = rr.SubjID"

result <- sqlQuery(con, query)
odbcClose(con)

result$Time <- as.POSIXct(strptime(result$Time, "%Y-%m-%d %H:%M"))
(sOlszewski <- subset(result, ID==1))
```

```
##   ID          Subject                Time RR_SYS RR_DIA BPM
## 1  1 Adrian Olszewski 2014-01-02 10:10:00    138     78  76
## 2  1 Adrian Olszewski 2014-01-02 15:30:00    142     82  99
## 3  1 Adrian Olszewski 2014-01-02 18:25:00    137     78  82
## 4  1 Adrian Olszewski 2014-01-03 10:45:00    142     88  87
## 5  1 Adrian Olszewski 2014-01-03 15:30:00    138     74  76
## 6  1 Adrian Olszewski 2014-01-04 11:30:00    146     91  91
```

```
par( mar=c(6, 4, 4, 2) + 0.5 )
with(sOlszewski, plot(Time, RR_SYS, col="blue", type="o", ylim=c(60, 150), xaxt="n", main="Blood pressure of:
Olszewski Adrian, #1", ylab="SYS / DIA", xlab=""))
with(sOlszewski, lines(Time, RR_DIA, col="magenta", type="o"))
title(xlab='Time', line=5)
grid(nx=NA, ny=NA)
Sys.setlocale(category = "LC_TIME", locale="C")
```

```
## [1] "C"
```

```
tck <- axis(1, at=sOlszewski$Time, labels=F)
text(tck, par("usr")[3], labels=format(sOlszewski$Time, "%b-%d %H:%M"), srt=50, xpd=TRUE, adj=c(1,2),
cex=0.8)
abline(v=tck ,col = "lightgray", lty = "dotted", lwd = par("lwd"))
abline(h=axis(2, at=seq(60, 150, 10), labels=F), col = "lightgray", lty = "dotted", lwd = par("lwd"))
legend("bottomright", c("SYS", "DIA"), col=c("blue", "magenta"), lty=c(1,1))
```

Compile Notebook from R Script

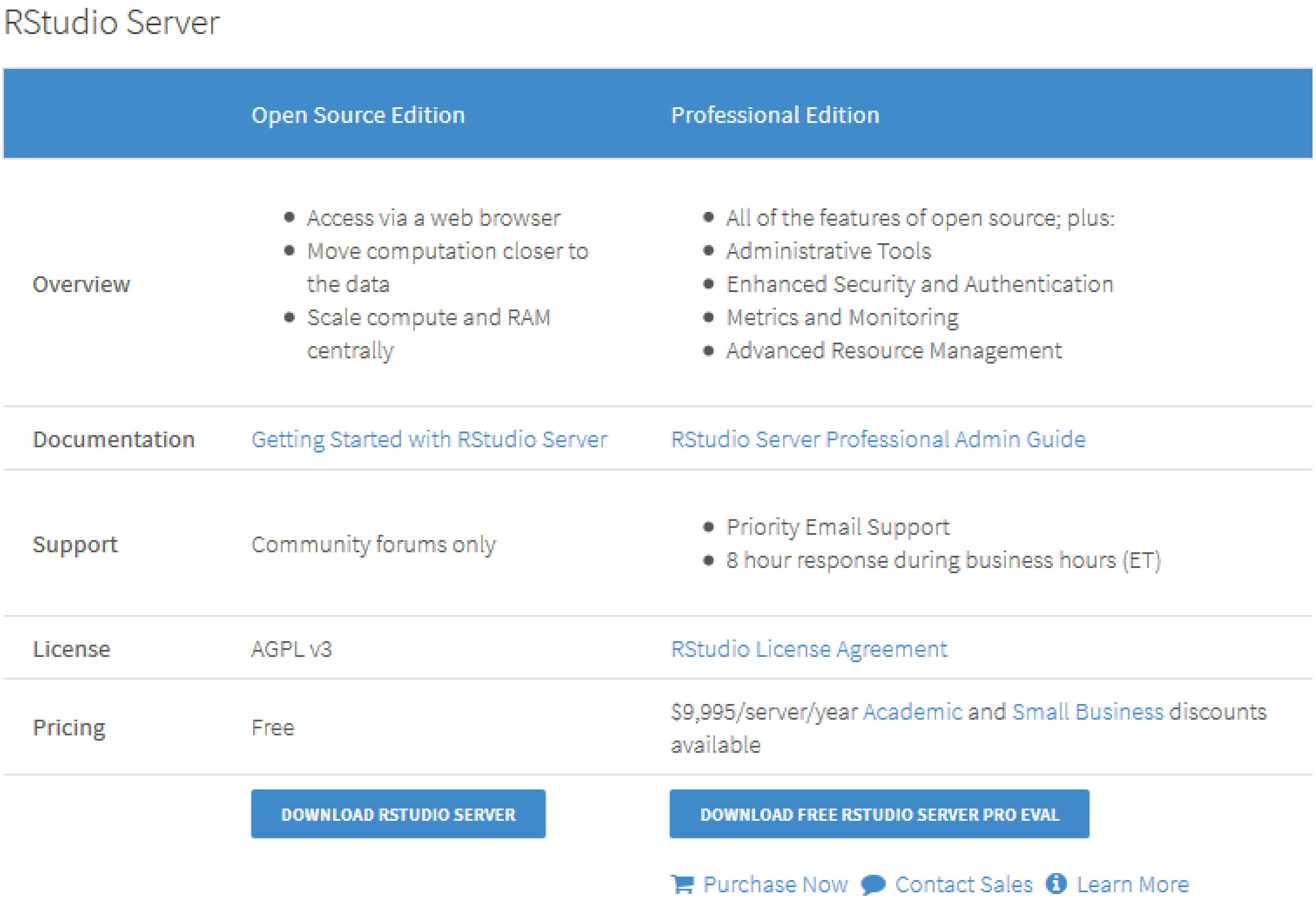
A Notebook is a standalone HTML file that contains the code and output from your R script.

Title (optional): test_notebook.R

Author (optional): aaolsz

Notebook type: ? knitr::spin

RStudio Server

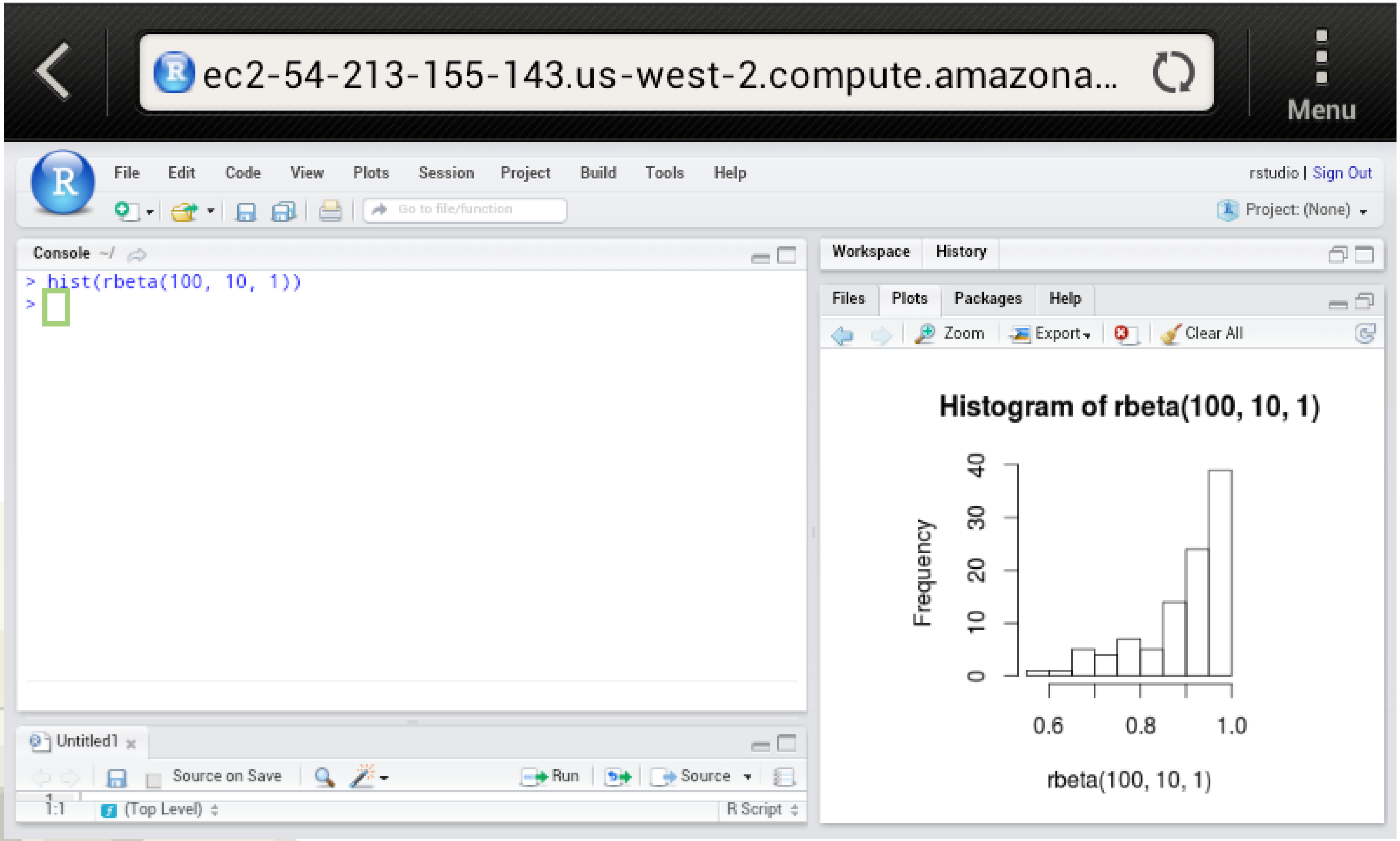


RStudio Server

	Open Source Edition	Professional Edition
Overview	• Access via a web browser • Move computation closer to the data • Scale compute and RAM centrally	• All of the features of open source; plus: • Administrative Tools • Enhanced Security and Authentication • Metrics and Monitoring • Advanced Resource Management
Documentation	Getting Started with RStudio Server	RStudio Server Professional Admin Guide
Support	Community forums only	• Priority Email Support • 8 hour response during business hours (ET)
License	AGPL v3	RStudio License Agreement
Pricing	Free	$9,995/server/year Academic and Small Business discounts available

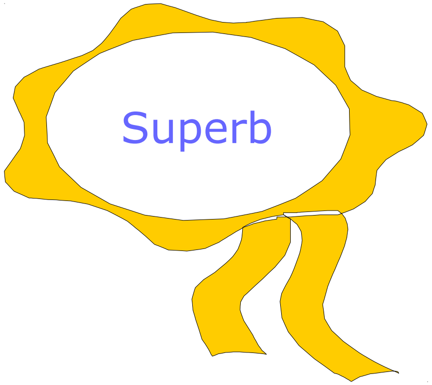
RStudio Server

On my... pocket HTC :)



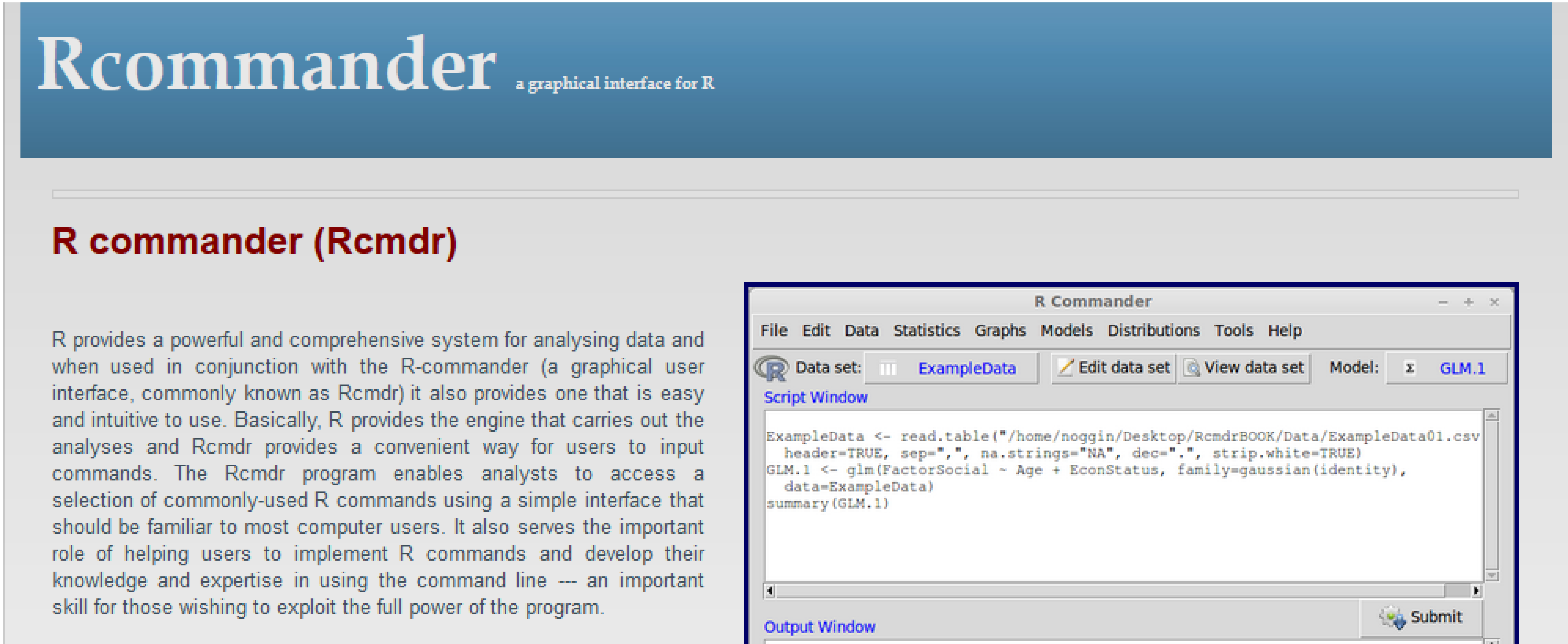
```
> hist(rbeta(100, 10, 1))
>
```

R is somewhat ascetic, isn't it?



2. Rcommander

http://www.rcommander.com



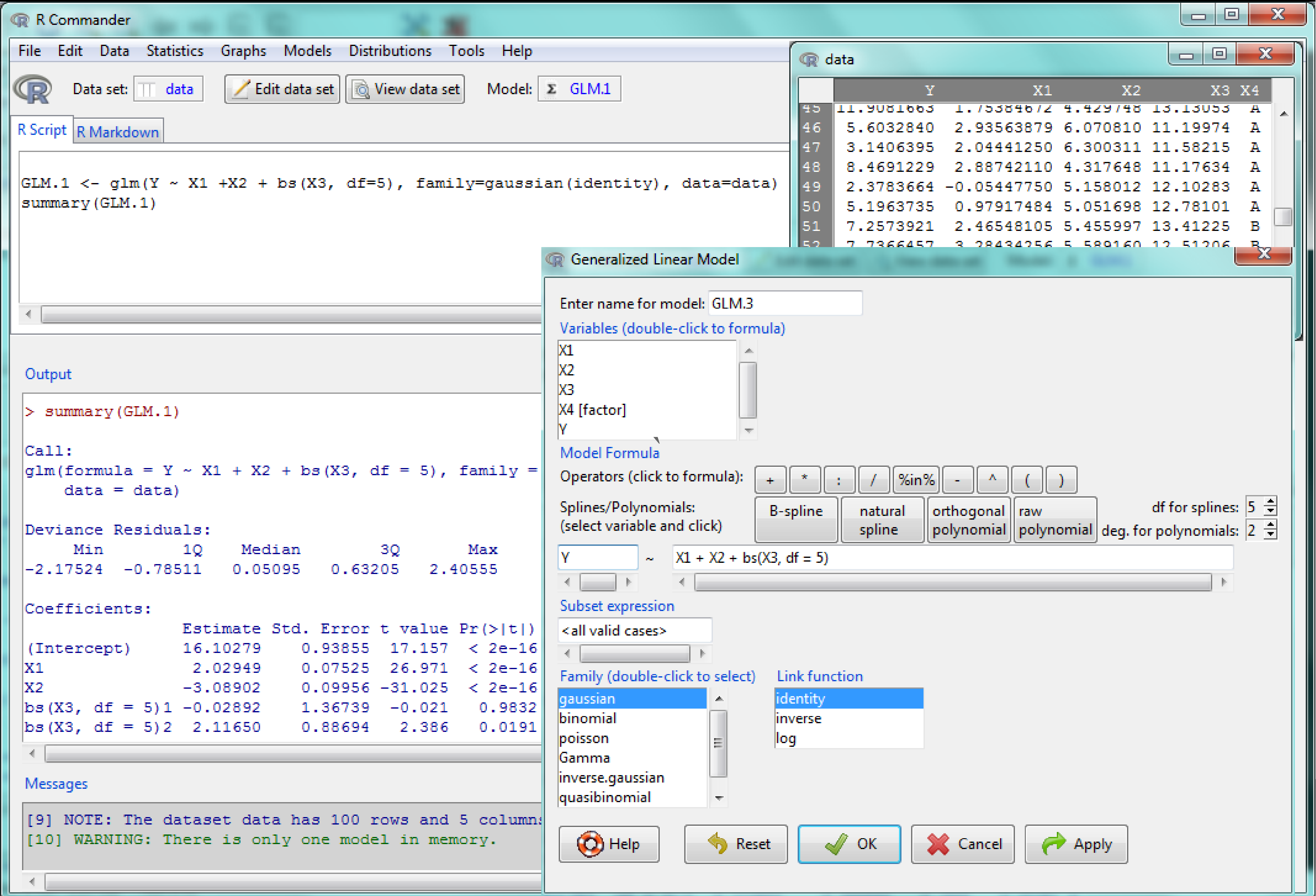
Rcommander a graphical interface for R

R commander (Rcmdr)

R provides a powerful and comprehensive system for analysing data and when used in conjunction with the R-commander (a graphical user interface, commonly known as Rcmdr) it also provides one that is easy and intuitive to use. Basically, R provides the engine that carries out the analyses and Rcmdr provides a convenient way for users to input commands. The Rcmdr program enables analysts to access a selection of commonly-used R commands using a simple interface that should be familiar to most computer users. It also serves the important role of helping users to implement R commands and develop their knowledge and expertise in using the command line --- an important skill for those wishing to exploit the full power of the program.

```
ExampleData <- read.table("/home/noggin/Desktop/RcmdrBOOK/Data/ExampleData01.csv
  header=TRUE, sep=",", na.strings="NA", dec=".", strip.white=TRUE)
GLM.1 <- glm(FactorSocial ~ Age + EconStatus, family=gaussian(identity),
  data=ExampleData)
summary(GLM.1)
```

RCommander



```
GLM.1 <- glm(Y ~ X1 +X2 + bs(X3, df=5), family=gaussian(identity), data=data)
summary(GLM.1)
```

```
> summary(GLM.1)

Call:
glm(formula = Y ~ X1 + X2 + bs(X3, df = 5), family =
    data = data)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.17524  -0.78511   0.05095   0.63205   2.40555

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     16.10279    0.93855  17.157  < 2e-16
X1               2.02949    0.07525  26.971  < 2e-16
X2              -3.08902    0.09956 -31.025  < 2e-16
bs(X3, df = 5)1 -0.02892    1.36739  -0.021   0.9832
bs(X3, df = 5)2  2.11650    0.88694   2.386   0.0191
```

```
[9] NOTE: The dataset data has 100 rows and 5 column
[10] WARNING: There is only one model in memory.
```

```
           Y          X1       X2       X3 X4
46 5.6032840  2.93563879 6.070810 11.19974  A
47 3.1406395  2.04441250 6.300311 11.58215  A
48 8.4691229  2.88742110 4.317648 11.17634  A
49 2.3783664 -0.05447750 5.158012 12.10283  A
50 5.1963735  0.97917484 5.051698 12.78101  A
51 7.2573921  2.46548105 5.455997 13.41225  B
```

Enter name for model: GLM.3

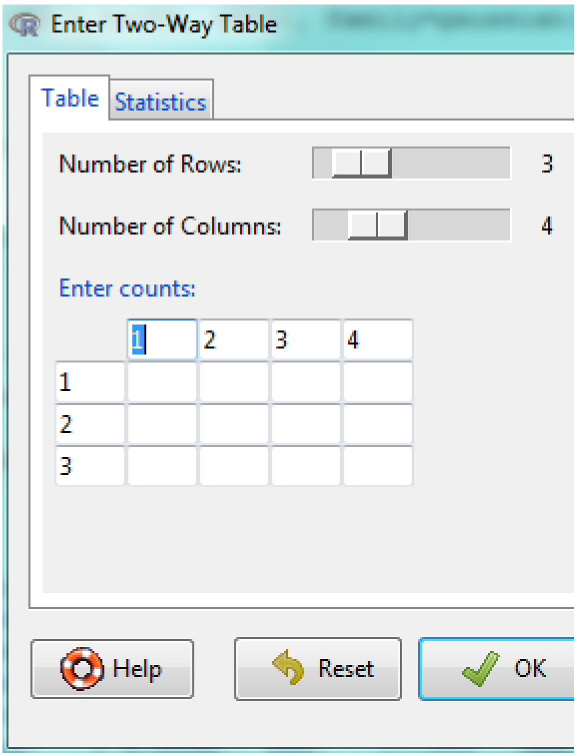
Y ~ X1 + X2 + bs(X3, df = 5)

df for splines: 5

deg. for polynomials: 2

<all valid cases>

RCommander



Enter Two-Way Table

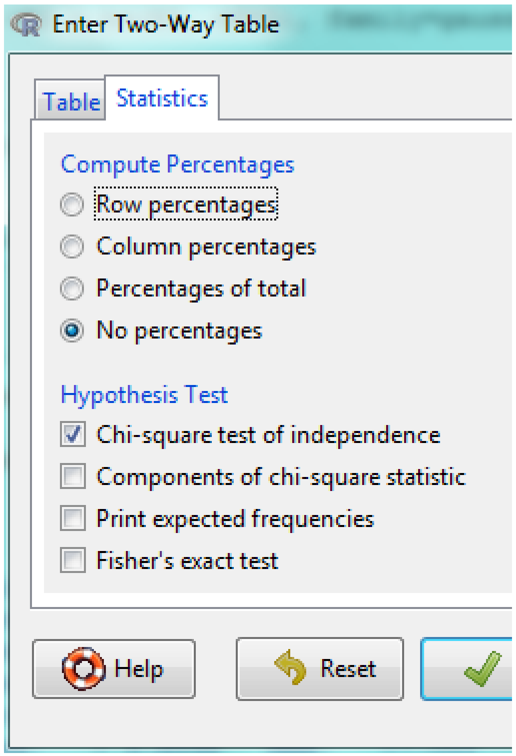
Table | Statistics

Number of Rows: 3

Number of Columns: 4

Enter counts:

	1	2	3	4
1				
2				
3				



Enter Two-Way Table

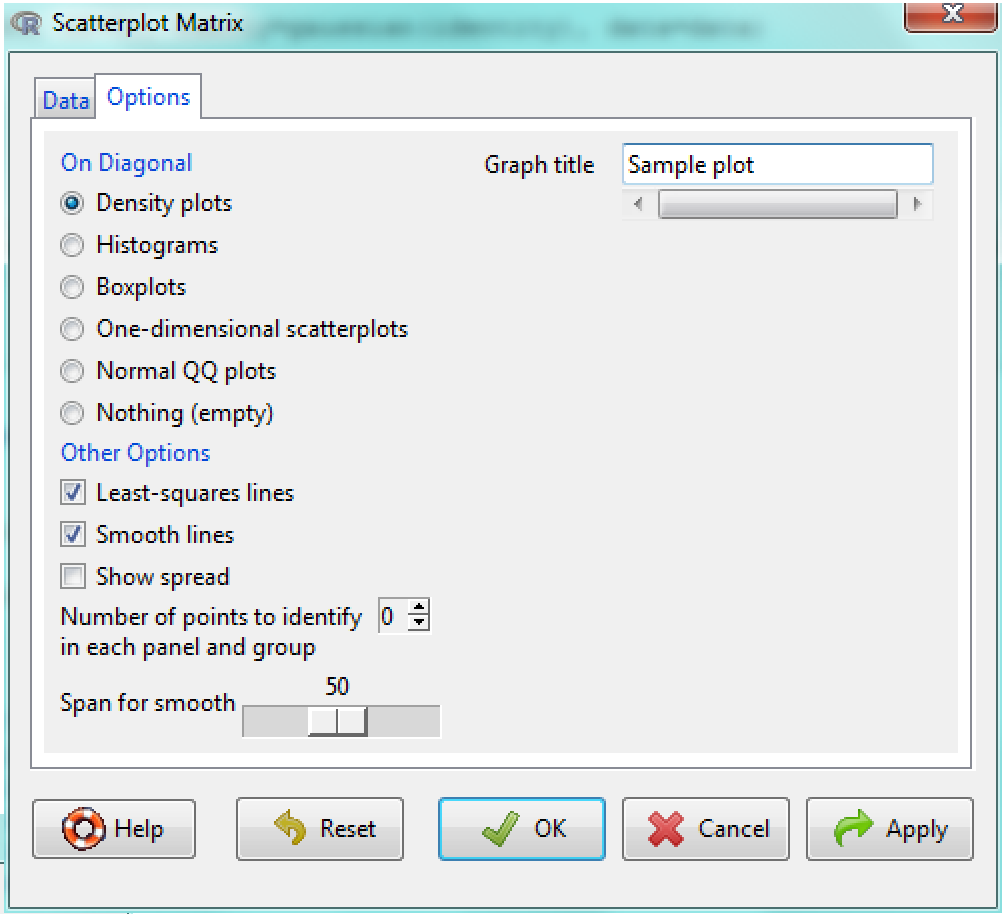
Table | Statistics

Compute Percentages

- ○ Row percentages
- ○ Column percentages
- ○ Percentages of total
- ● No percentages

Hypothesis Test

- ☑ Chi-square test of independence
- ☐ Components of chi-square statistic
- ☐ Print expected frequencies
- ☐ Fisher's exact test



Scatterplot Matrix

Data | Options

On Diagonal

- ● Density plots
- ○ Histograms
- ○ Boxplots
- ○ One-dimensional scatterplots
- ○ Normal QQ plots
- ○ Nothing (empty)

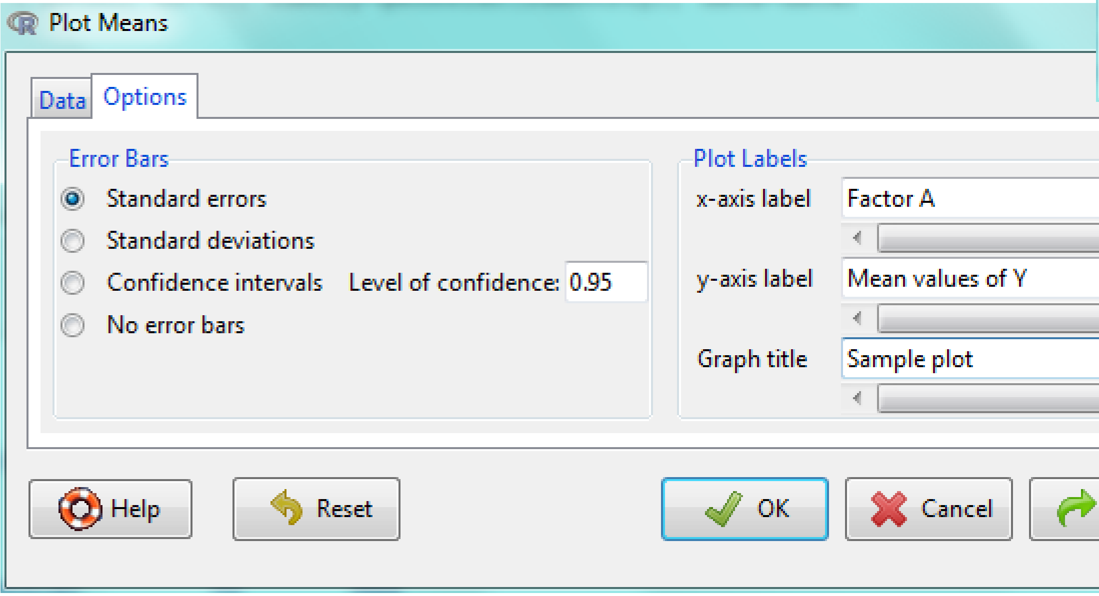
Graph title: Sample plot

Other Options

- ☑ Least-squares lines
- ☑ Smooth lines
- ☐ Show spread

Number of points to identify in each panel and group: 0

Span for smooth: 50



Plot Means

Data | Options

Error Bars

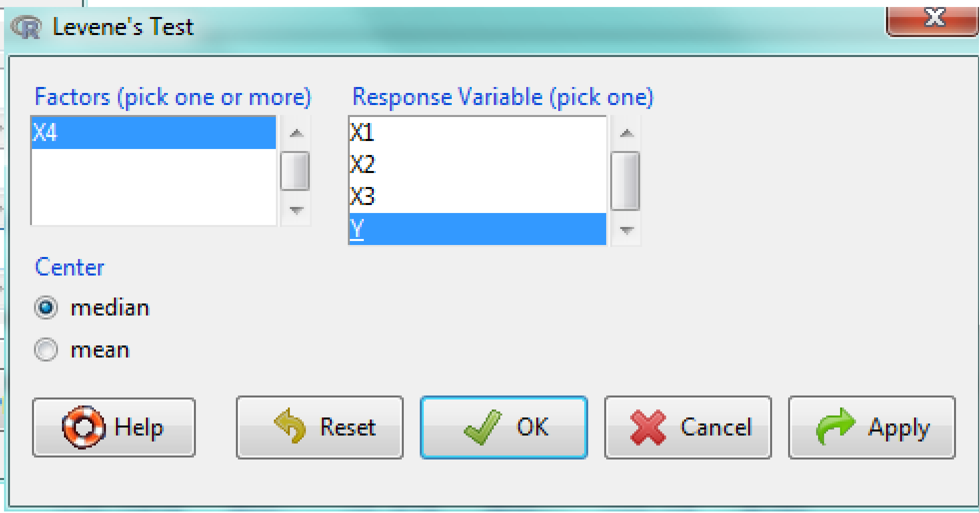
- ● Standard errors
- ○ Standard deviations
- ○ Confidence intervals   Level of confidence: 0.95
- ○ No error bars

Plot Labels

x-axis label: Factor A

y-axis label: Mean values of Y

Graph title: Sample plot



Levene's Test

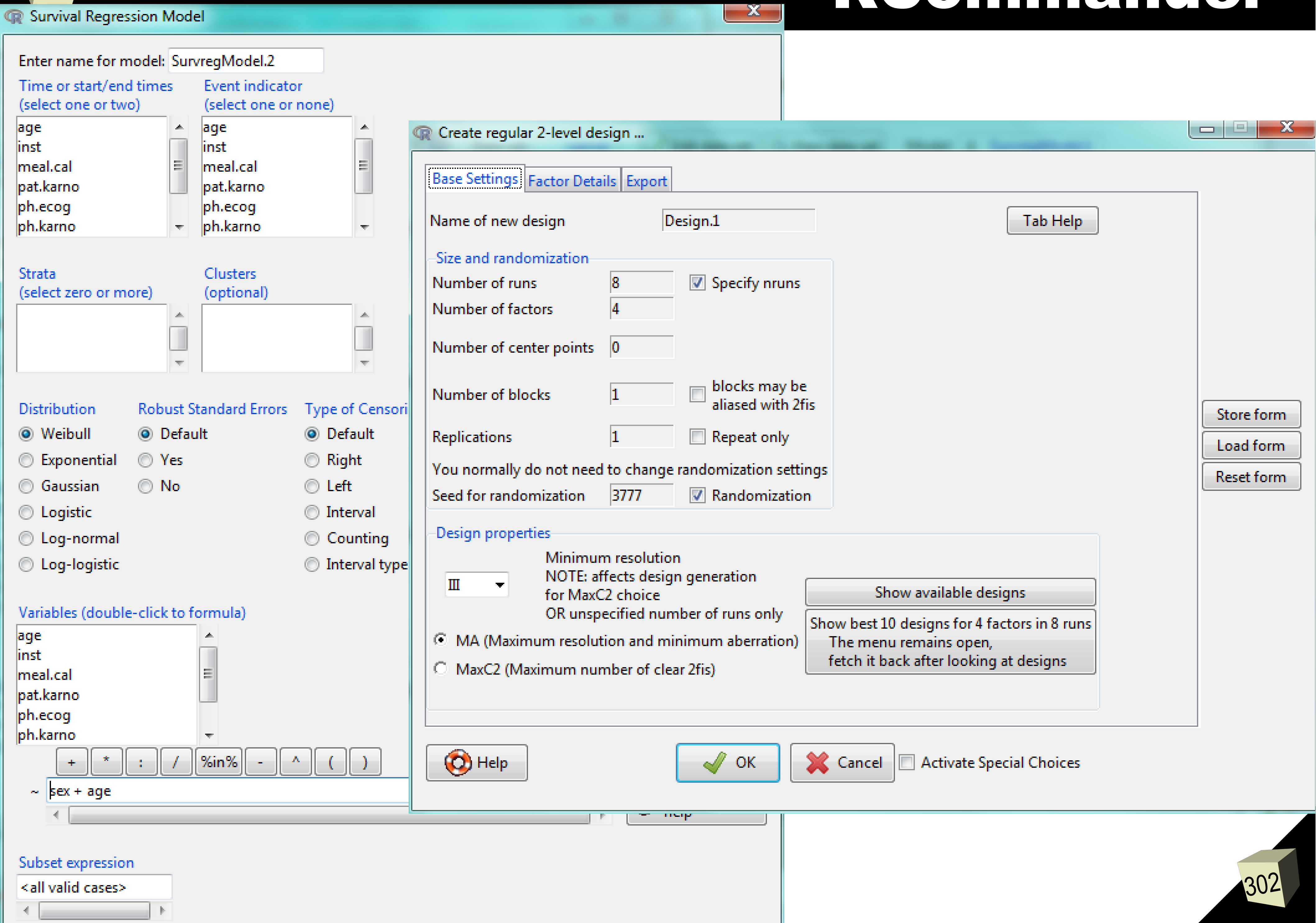
Factors (pick one or more): X4

Response Variable (pick one): X1, X2, X3, Y

Center

- ● median
- ○ mean

RCommander



Survival Regression Model

Enter name for model: SurvregModel.2

Time or start/end times (select one or two)
- age
- inst
- meal.cal
- pat.karno
- ph.ecog
- ph.karno

Event indicator (select one or none)
- age
- inst
- meal.cal
- pat.karno
- ph.ecog
- ph.karno

Strata (select zero or more)

Clusters (optional)

Distribution
- ● Weibull
- ○ Exponential
- ○ Gaussian
- ○ Logistic
- ○ Log-normal
- ○ Log-logistic

Robust Standard Errors
- ● Default
- ○ Yes
- ○ No

Type of Censori
- ● Default
- ○ Right
- ○ Left
- ○ Interval
- ○ Counting
- ○ Interval type

Variables (double-click to formula)
- age
- inst
- meal.cal
- pat.karno
- ph.ecog
- ph.karno

~ sex + age

Subset expression: <all valid cases>

Create regular 2-level design ...

Name of new design: Design.1

Size and randomization

Number of runs: 8 ☑ Specify nruns

Number of factors: 4

Number of center points: 0

Number of blocks: 1 ☐ blocks may be aliased with 2fis

Replications: 1 ☐ Repeat only

You normally do not need to change randomization settings

Seed for randomization: 3777 ☑ Randomization

Design properties

III

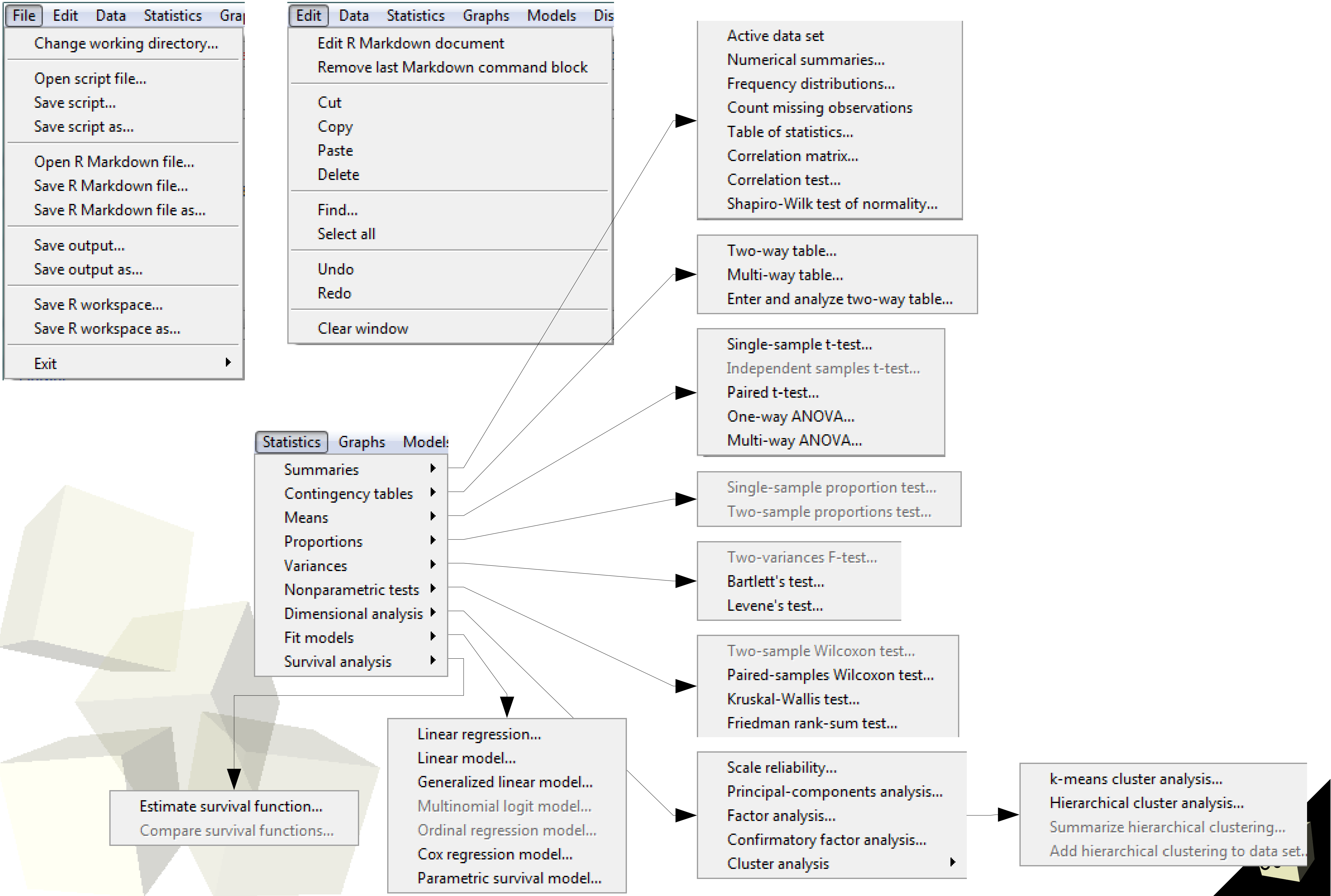
Minimum resolution
NOTE: affects design generation
for MaxC2 choice
OR unspecified number of runs only

● MA (Maximum resolution and minimum aberration)

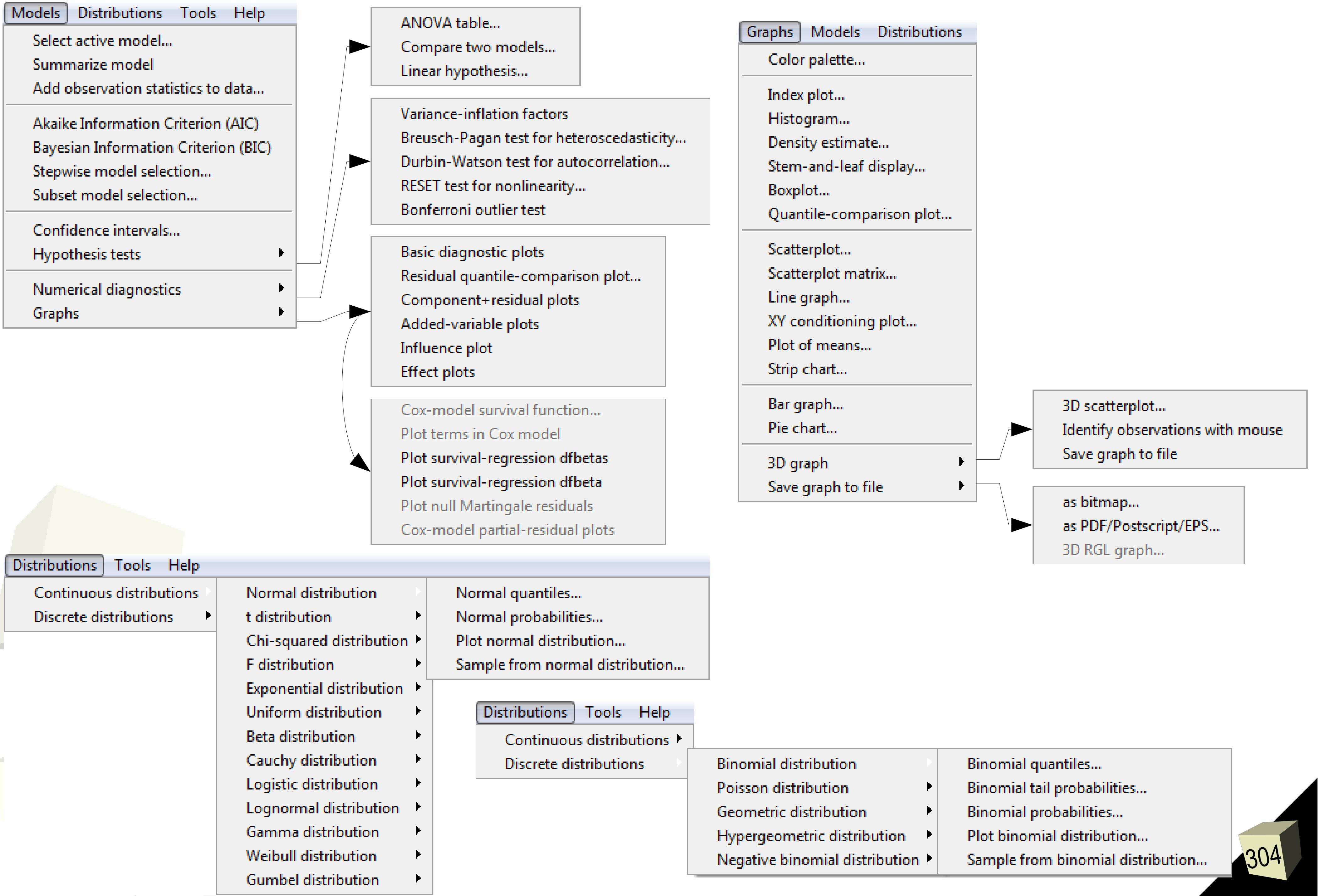
○ MaxC2 (Maximum number of clear 2fis)

☐ Activate Special Choices

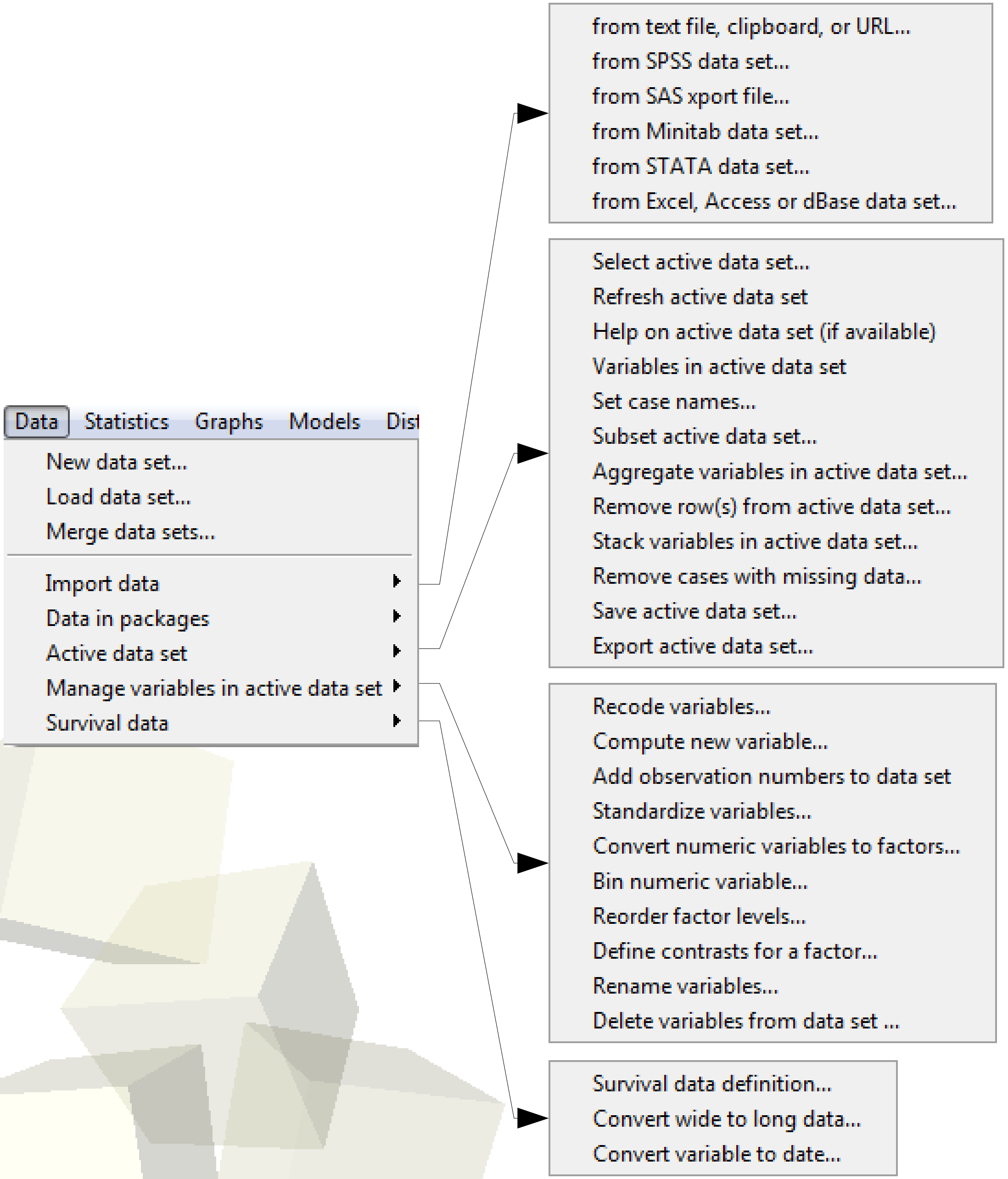
RCommander


File
Edit
Data
Statistics
Gra
Change working directory...
Open script file...
Save script...
Save script as...
Open R Markdown file...
Save R Markdown file...
Save R Markdown file as...
Save output...
Save output as...
Save R workspace...
Save R workspace as...
Exit
Edit
Data
Statistics
Graphs
Models
Dis
Edit R Markdown document
Remove last Markdown command block
Cut
Copy
Paste
Delete
Find...
Select all
Undo
Redo
Clear window
Statistics
Graphs
Models
Summaries
Contingency tables
Means
Proportions
Variances
Nonparametric tests
Dimensional analysis
Fit models
Survival analysis
Active data set
Numerical summaries...
Frequency distributions...
Count missing observations
Table of statistics...
Correlation matrix...
Correlation test...
Shapiro-Wilk test of normality...
Two-way table...
Multi-way table...
Enter and analyze two-way table...
Single-sample t-test...
Independent samples t-test...
Paired t-test...
One-way ANOVA...
Multi-way ANOVA...
Single-sample proportion test...
Two-sample proportions test...
Two-variances F-test...
Bartlett's test...
Levene's test...
Two-sample Wilcoxon test...
Paired-samples Wilcoxon test...
Kruskal-Wallis test...
Friedman rank-sum test...
Scale reliability...
Principal-components analysis...
Factor analysis...
Confirmatory factor analysis...
Cluster analysis
k-means cluster analysis...
Hierarchical cluster analysis...
Summarize hierarchical clustering...
Add hierarchical clustering to data set...
Linear regression...
Linear model...
Generalized linear model...
Multinomial logit model...
Ordinal regression model...
Cox regression model...
Parametric survival model...
Estimate survival function...
Compare survival functions...

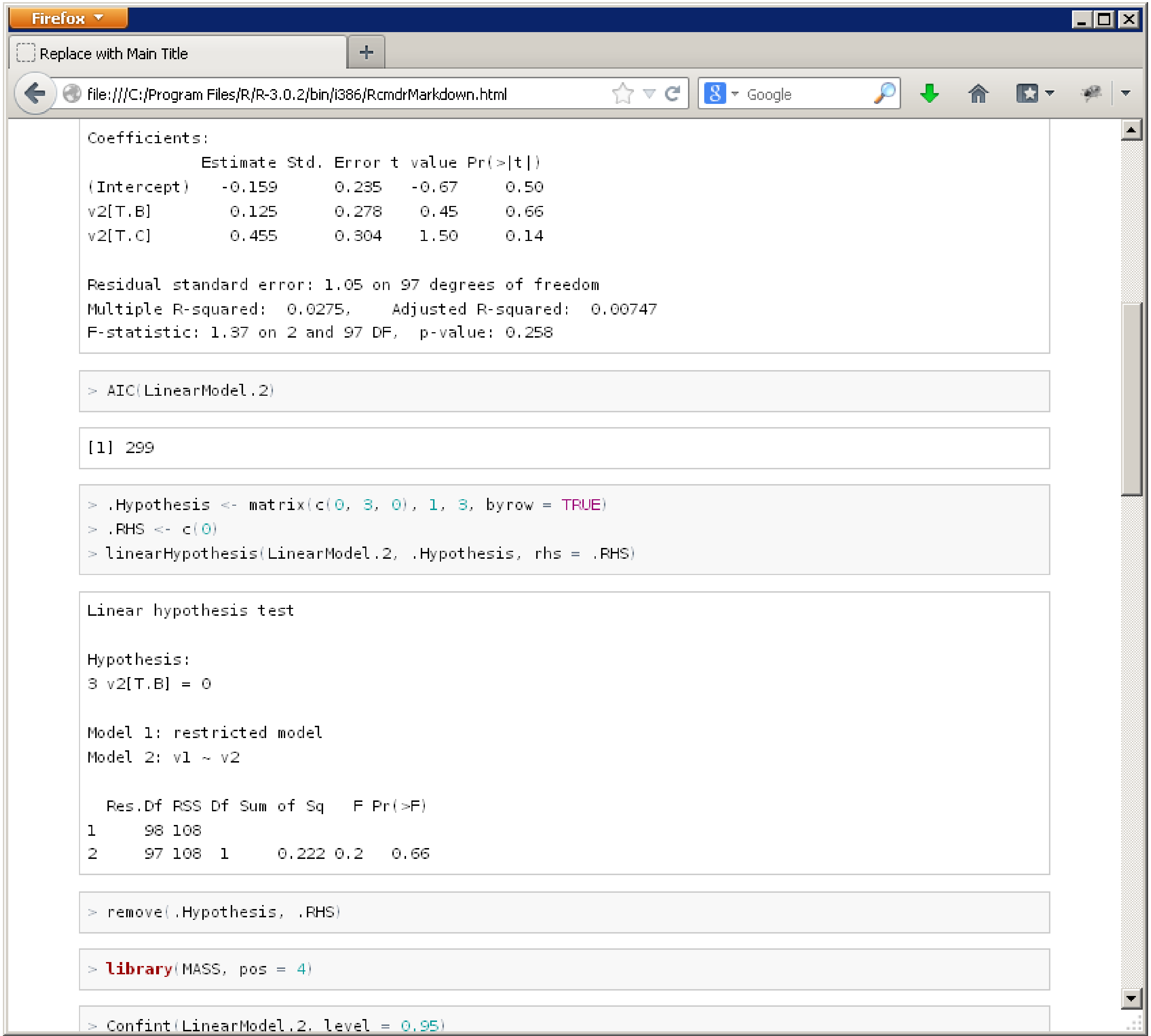
RCommander


Models
Distributions
Tools
Help
Select active model...
Summarize model
Add observation statistics to data...
Akaike Information Criterion (AIC)
Bayesian Information Criterion (BIC)
Stepwise model selection...
Subset model selection...
Confidence intervals...
Hypothesis tests
Numerical diagnostics
Graphs
ANOVA table...
Compare two models...
Linear hypothesis...
Variance-inflation factors
Breusch-Pagan test for heteroscedasticity...
Durbin-Watson test for autocorrelation...
RESET test for nonlinearity...
Bonferroni outlier test
Basic diagnostic plots
Residual quantile-comparison plot...
Component+residual plots
Added-variable plots
Influence plot
Effect plots
Cox-model survival function...
Plot terms in Cox model
Plot survival-regression dfbetas
Plot survival-regression dfbeta
Plot null Martingale residuals
Cox-model partial-residual plots
Graphs
Models
Distributions
Color palette...
Index plot...
Histogram...
Density estimate...
Stem-and-leaf display...
Boxplot...
Quantile-comparison plot...
Scatterplot...
Scatterplot matrix...
Line graph...
XY conditioning plot...
Plot of means...
Strip chart...
Bar graph...
Pie chart...
3D graph
Save graph to file
3D scatterplot...
Identify observations with mouse
Save graph to file
as bitmap...
as PDF/Postscript/EPS...
3D RGL graph...
Distributions
Tools
Help
Continuous distributions
Discrete distributions
Normal distribution
t distribution
Chi-squared distribution
F distribution
Exponential distribution
Uniform distribution
Beta distribution
Cauchy distribution
Logistic distribution
Lognormal distribution
Gamma distribution
Weibull distribution
Gumbel distribution
Normal quantiles...
Normal probabilities...
Plot normal distribution...
Sample from normal distribution...
Distributions
Tools
Help
Continuous distributions
Discrete distributions
Binomial distribution
Poisson distribution
Geometric distribution
Hypergeometric distribution
Negative binomial distribution
Binomial quantiles...
Binomial tail probabilities...
Binomial probabilities...
Plot binomial distribution...
Sample from binomial distribution...

RCommander


Data
Statistics
Graphs
Models
Dist
New data set...
Load data set...
Merge data sets...
Import data
Data in packages
Active data set
Manage variables in active data set
Survival data
from text file, clipboard, or URL...
from SPSS data set...
from SAS xport file...
from Minitab data set...
from STATA data set...
from Excel, Access or dBase data set...
Select active data set...
Refresh active data set
Help on active data set (if available)
Variables in active data set
Set case names...
Subset active data set...
Aggregate variables in active data set...
Remove row(s) from active data set...
Stack variables in active data set...
Remove cases with missing data...
Save active data set...
Export active data set...
Recode variables...
Compute new variable...
Add observation numbers to data set
Standardize variables...
Convert numeric variables to factors...
Bin numeric variable...
Reorder factor levels...
Define contrasts for a factor...
Rename variables...
Delete variables from data set ...
Survival data definition...
Convert wide to long data...
Convert variable to date...

RCommander – the notebook (RMarkdown)



```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   -0.159      0.235   -0.67     0.50
v2[T.B]        0.125      0.278    0.45     0.66
v2[T.C]        0.455      0.304    1.50     0.14

Residual standard error: 1.05 on 97 degrees of freedom
Multiple R-squared:  0.0275,    Adjusted R-squared:  0.00747
F-statistic: 1.37 on 2 and 97 DF,  p-value: 0.258
```

```
> AIC(LinearModel.2)
```

```
[1] 299
```

```
> .Hypothesis <- matrix(c(0, 3, 0), 1, 3, byrow = TRUE)
> .RHS <- c(0)
> linearHypothesis(LinearModel.2, .Hypothesis, rhs = .RHS)
```

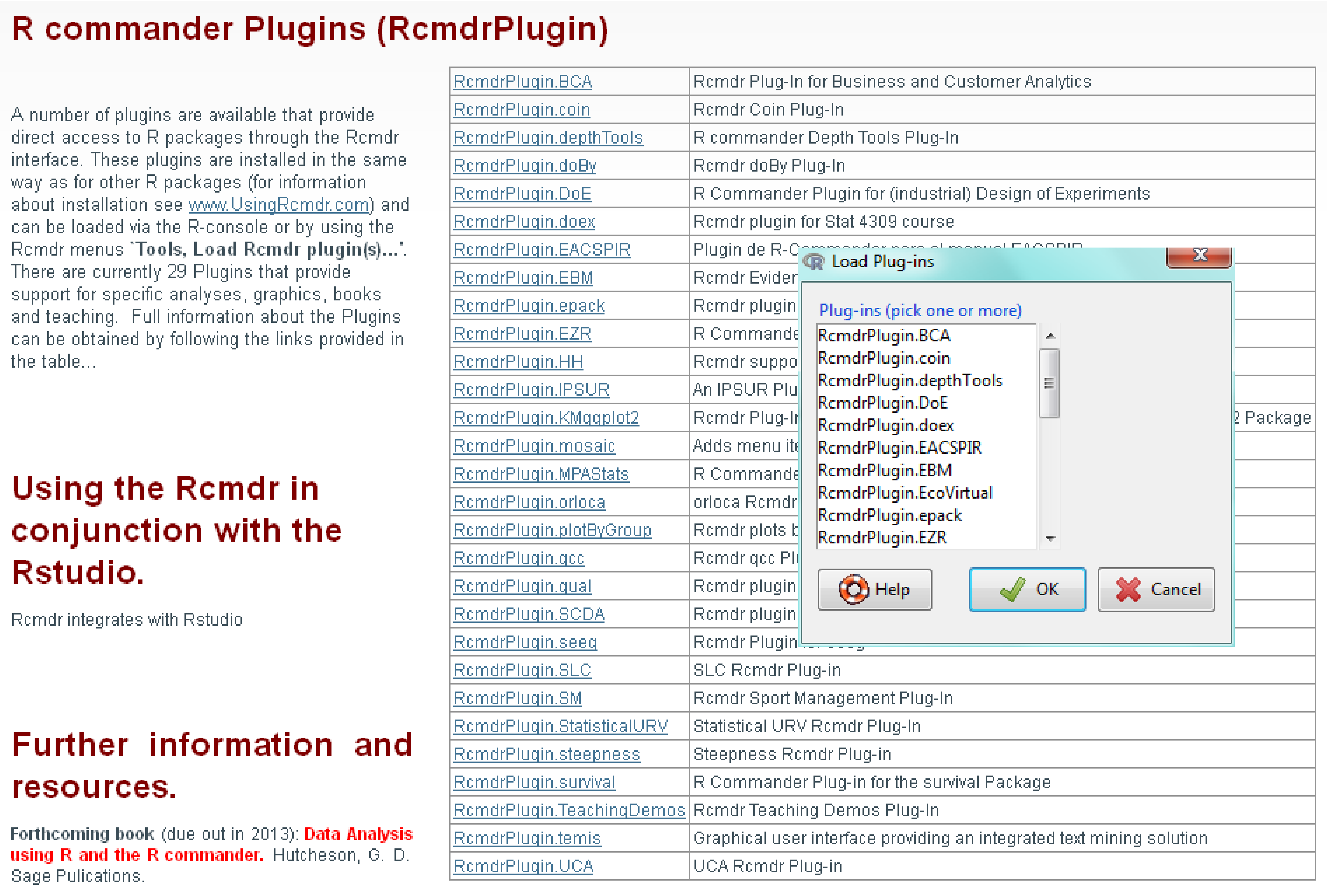
```
Linear hypothesis test

Hypothesis:
3 v2[T.B] = 0

Model 1: restricted model
Model 2: v1 ~ v2

  Res.Df RSS Df Sum of Sq   F Pr(>F)
1     98 108
2     97 108  1     0.222 0.2   0.66
```

```
> remove(.Hypothesis, .RHS)
```

```
> library(MASS, pos = 4)
```

```
> Confint(LinearModel.2, level = 0.95)
```

RCommander - plugins



R commander Plugins (RcmdrPlugin)

A number of plugins are available that provide direct access to R packages through the Rcmdr interface. These plugins are installed in the same way as for other R packages (for information about installation see www.UsingRcmdr.com) and can be loaded via the R-console or by using the Rcmdr menus `**Tools, Load Rcmdr plugin(s)...**'. There are currently 29 Plugins that provide support for specific analyses, graphics, books and teaching. Full information about the Plugins can be obtained by following the links provided in the table...

RcmdrPlugin.BCA	Rcmdr Plug-In for Business and Customer Analytics
RcmdrPlugin.coin	Rcmdr Coin Plug-In
RcmdrPlugin.depthTools	R commander Depth Tools Plug-In
RcmdrPlugin.doBy	Rcmdr doBy Plug-In
RcmdrPlugin.DoE	R Commander Plugin for (industrial) Design of Experiments
RcmdrPlugin.doex	Rcmdr plugin for Stat 4309 course
RcmdrPlugin.EACSPIR	Plugin de R-C
RcmdrPlugin.EBM	Rcmdr Evider
RcmdrPlugin.epack	Rcmdr plugin
RcmdrPlugin.EZR	R Commande
RcmdrPlugin.HH	Rcmdr suppo
RcmdrPlugin.IPSUR	An IPSUR Plu
RcmdrPlugin.KMggplot2	Rcmdr Plug-Ir 2 Package
RcmdrPlugin.mosaic	Adds menu ite
RcmdrPlugin.MPAStats	R Commande
RcmdrPlugin.orloca	orloca Rcmdr
RcmdrPlugin.plotByGroup	Rcmdr plots b
RcmdrPlugin.qcc	Rcmdr qcc Pl
RcmdrPlugin.qual	Rcmdr plugin
RcmdrPlugin.SCDA	Rcmdr plugin
RcmdrPlugin.seeg	Rcmdr Plugin
RcmdrPlugin.SLC	SLC Rcmdr Plug-in
RcmdrPlugin.SM	Rcmdr Sport Management Plug-In
RcmdrPlugin.StatisticalURV	Statistical URV Rcmdr Plug-In
RcmdrPlugin.steepness	Steepness Rcmdr Plug-in
RcmdrPlugin.survival	R Commander Plug-in for the survival Package
RcmdrPlugin.TeachingDemos	Rcmdr Teaching Demos Plug-In
RcmdrPlugin.temis	Graphical user interface providing an integrated text mining solution
RcmdrPlugin.UCA	UCA Rcmdr Plug-in

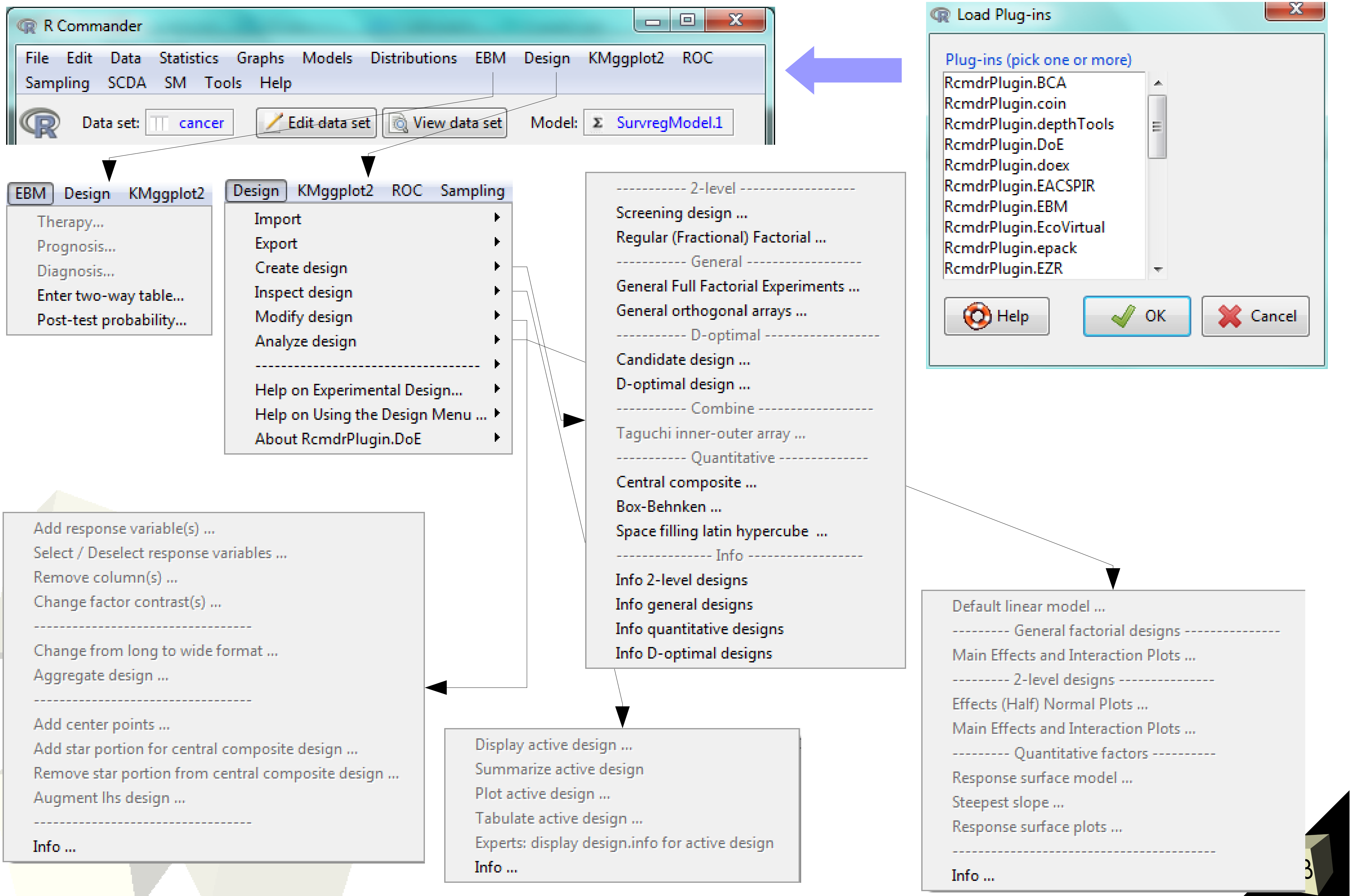
Using the Rcmdr in conjunction with the Rstudio.

Rcmdr integrates with Rstudio

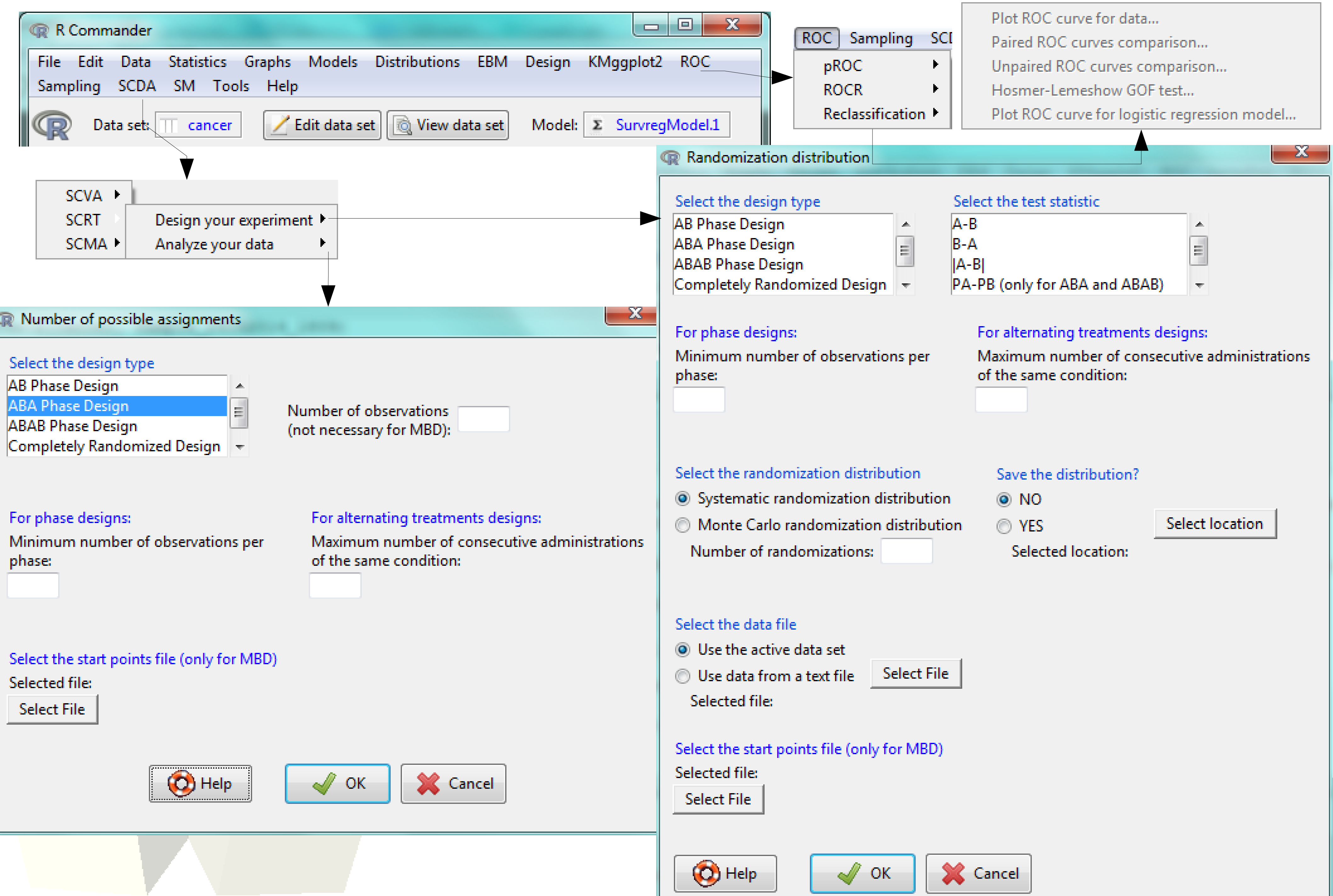
Further information and resources.

Forthcoming book (due out in 2013): **Data Analysis using R and the R commander.** Hutcheson, G. D. Sage Pulications.

bio RCommander



bio RCommander



Number of possible assignments

Select the design type

- AB Phase Design
- ABA Phase Design
- ABAB Phase Design
- Completely Randomized Design

Number of observations (not necessary for MBD):

For phase designs:

Minimum number of observations per phase:

For alternating treatments designs:

Maximum number of consecutive administrations of the same condition:

Select the start points file (only for MBD)

Selected file:

Randomization distribution

Select the design type

- AB Phase Design
- ABA Phase Design
- ABAB Phase Design
- Completely Randomized Design

Select the test statistic

- A-B
- B-A
- |A-B|
- PA-PB (only for ABA and ABAB)

For phase designs:

Minimum number of observations per phase:

For alternating treatments designs:

Maximum number of consecutive administrations of the same condition:

Select the randomization distribution

● Systematic randomization distribution

○ Monte Carlo randomization distribution

Number of randomizations:

Save the distribution?

● NO

○ YES

Selected location:

Select the data file

● Use the active data set

○ Use data from a text file

Selected file:

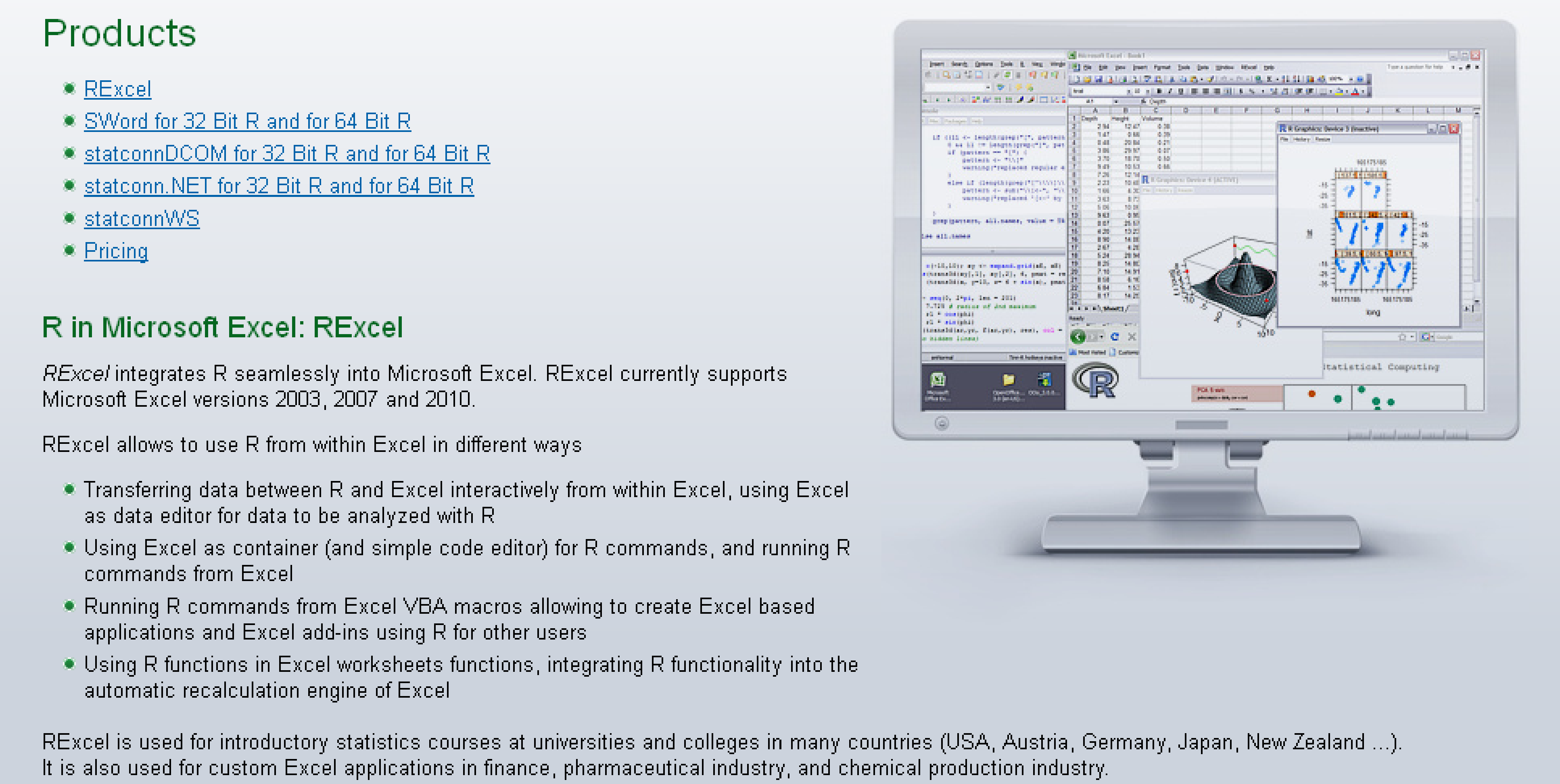
Select the start points file (only for MBD)

Selected file:

R is somewhat ascetic, isn't it?

3. RExcel

http://www.statconn.com/products.html



Products

- RExcel
- SWord for 32 Bit R and for 64 Bit R
- statconnDCOM for 32 Bit R and for 64 Bit R
- statconn.NET for 32 Bit R and for 64 Bit R
- statconnWS
- Pricing

R in Microsoft Excel: RExcel

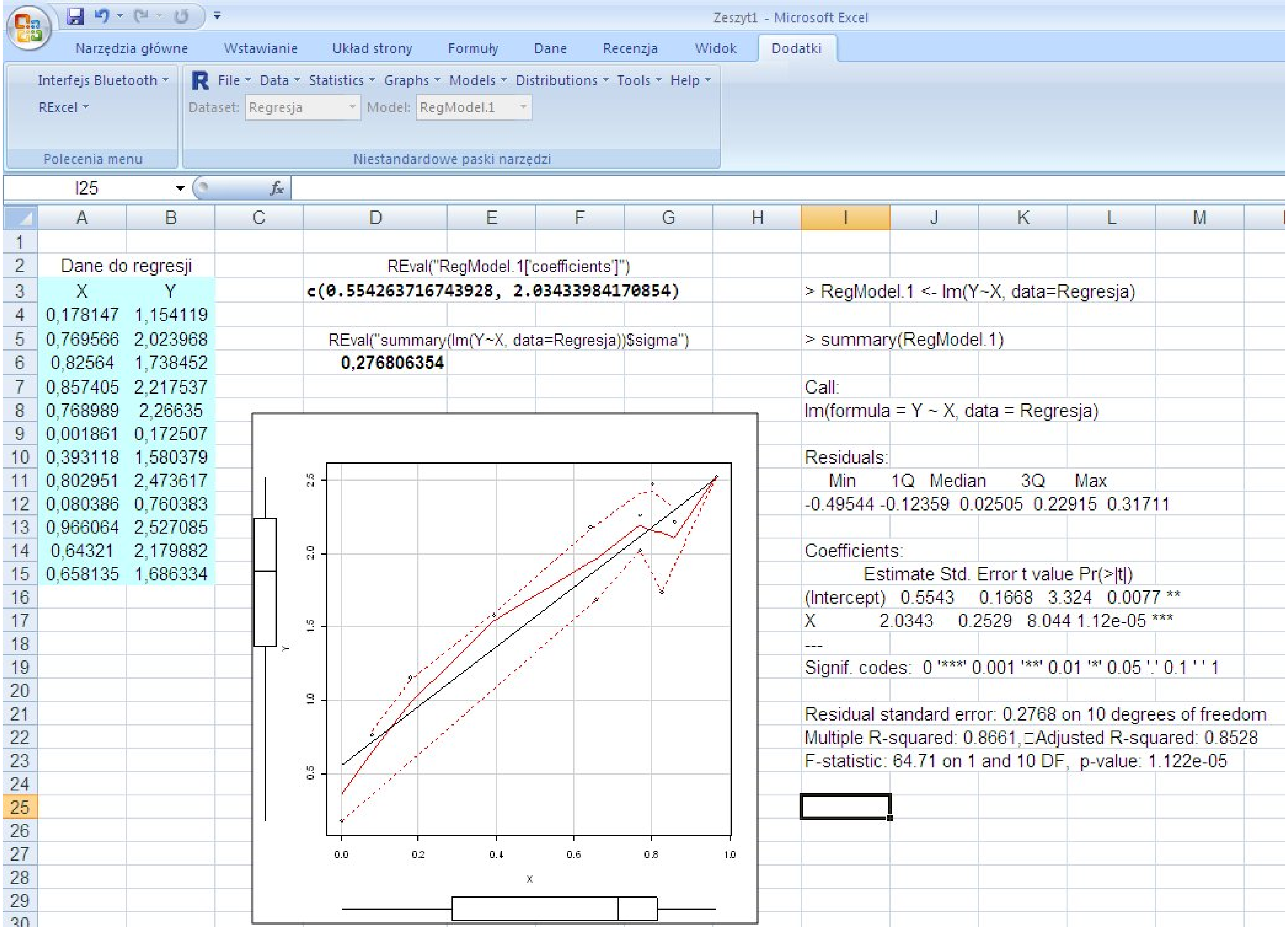
RExcel integrates R seamlessly into Microsoft Excel. RExcel currently supports Microsoft Excel versions 2003, 2007 and 2010.

RExcel allows to use R from within Excel in different ways

- Transferring data between R and Excel interactively from within Excel, using Excel as data editor for data to be analyzed with R
- Using Excel as container (and simple code editor) for R commands, and running R commands from Excel
- Running R commands from Excel VBA macros allowing to create Excel based applications and Excel add-ins using R for other users
- Using R functions in Excel worksheets functions, integrating R functionality into the automatic recalculation engine of Excel

RExcel is used for introductory statistics courses at universities and colleges in many countries (USA, Austria, Germany, Japan, New Zealand ...). It is also used for custom Excel applications in finance, pharmaceutical industry, and chemical production industry.

Excel + RExcel + RCommander



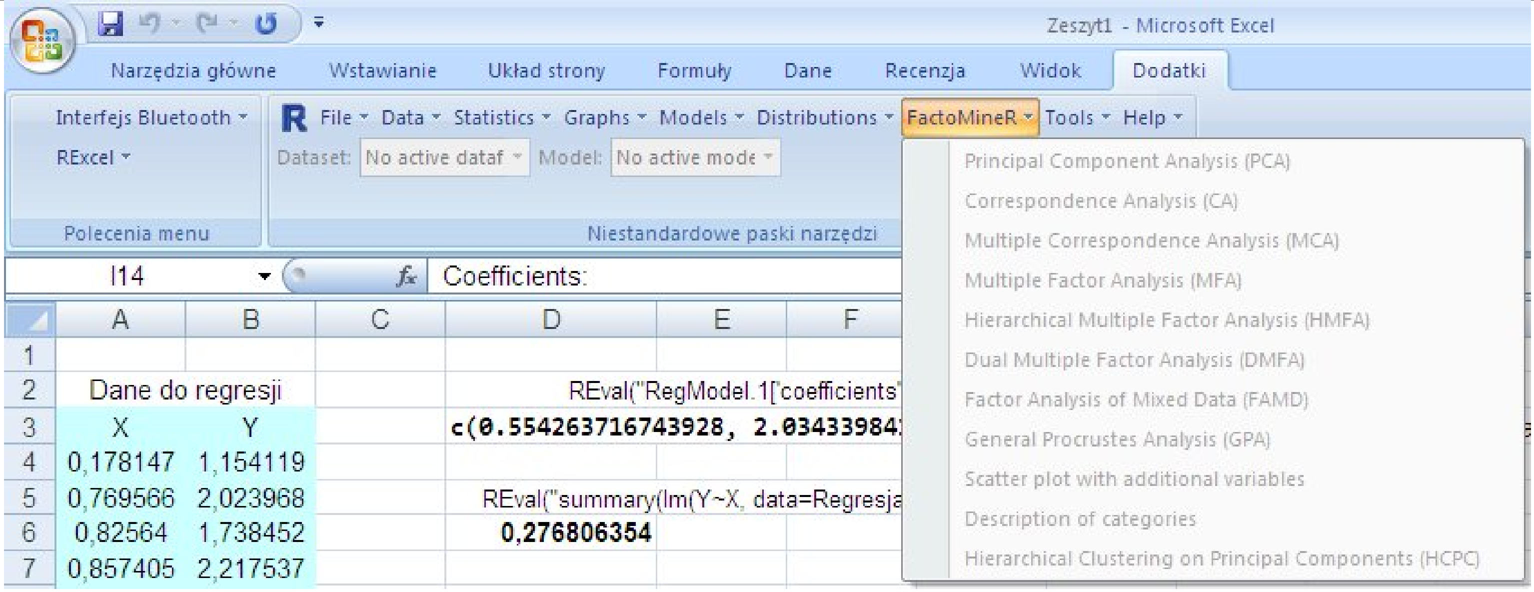
Dane do regresji

X	Y
0,178147	1,154119
0,769566	2,023968
0,82564	1,738452
0,857405	2,217537
0,768989	2,26635
0,001861	0,172507
0,393118	1,580379
0,802951	2,473617
0,080386	0,760383
0,966064	2,527085
0,64321	2,179882
0,658135	1,686334

```
REval("RegModel.1['coefficients']")
c(0.554263716743928, 2.03433984170854)

REval("summary(lm(Y~X, data=Regresja))$sigma")
0,276806354
```

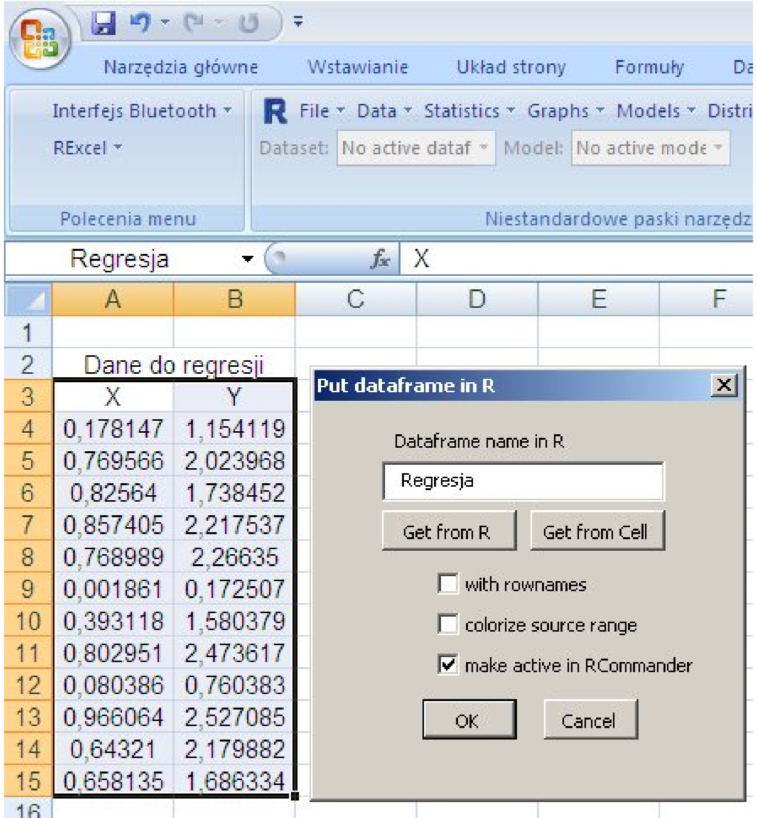
```
> RegModel.1 <- lm(Y~X, data=Regresja)

> summary(RegModel.1)

Call:
lm(formula = Y ~ X, data = Regresja)

Residuals:
    Min      1Q  Median      3Q     Max
-0.49544 -0.12359  0.02505  0.22915  0.31711

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.5543     0.1668   3.324   0.0077 **
X             2.0343     0.2529   8.044 1.12e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2768 on 10 degrees of freedom
Multiple R-squared: 0.8661,□Adjusted R-squared: 0.8528
F-statistic: 64.71 on 1 and 10 DF,  p-value: 1.122e-05
```

Excel + RExcel + RCommander



I14 — Coefficients:

	A	B	C	D	E	F
1						
2	Dane do regresji			REval("RegModel.1['coefficients'		
3	X	Y		c(0.554263716743928, 2.03433984		
4	0,178147	1,154119				
5	0,769566	2,023968		REval("summary(lm(Y~X, data=Regresja		
6	0,82564	1,738452		0,276806354		
7	0,857405	2,217537				



Regresja — X

	A	B	C	D	E	F
1						
2	Dane do regresji					
3	X	Y				
4	0,178147	1,154119				
5	0,769566	2,023968				
6	0,82564	1,738452				
7	0,857405	2,217537				
8	0,768989	2,26635				
9	0,001861	0,172507				
10	0,393118	1,580379				
11	0,802951	2,473617				
12	0,080386	0,760383				
13	0,966064	2,527085				
14	0,64321	2,179882				
15	0,658135	1,686334				

Put dataframe in R

Dataframe name in R: Regresja

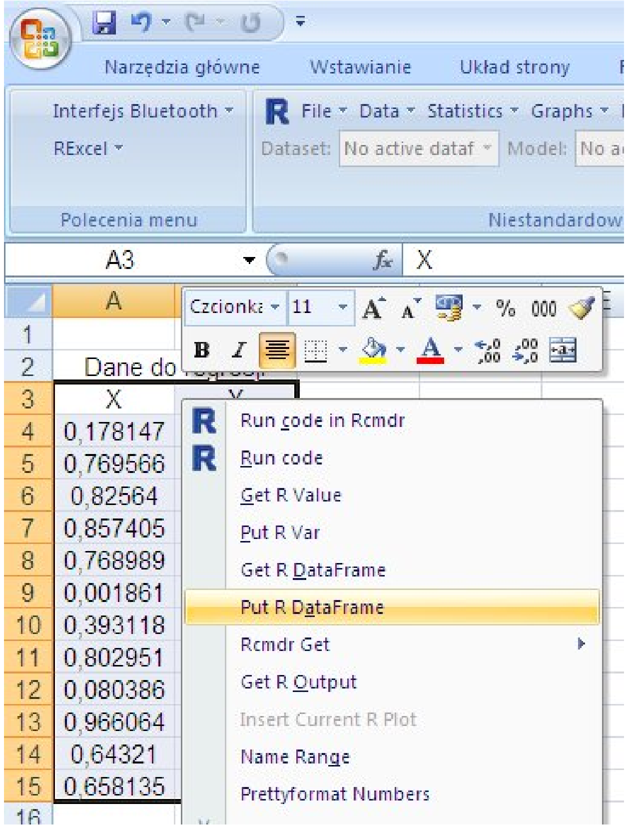
Get from R | Get from Cell

☐ with rownames

☐ colorize source range

☑ make active in RCommander

OK | Cancel



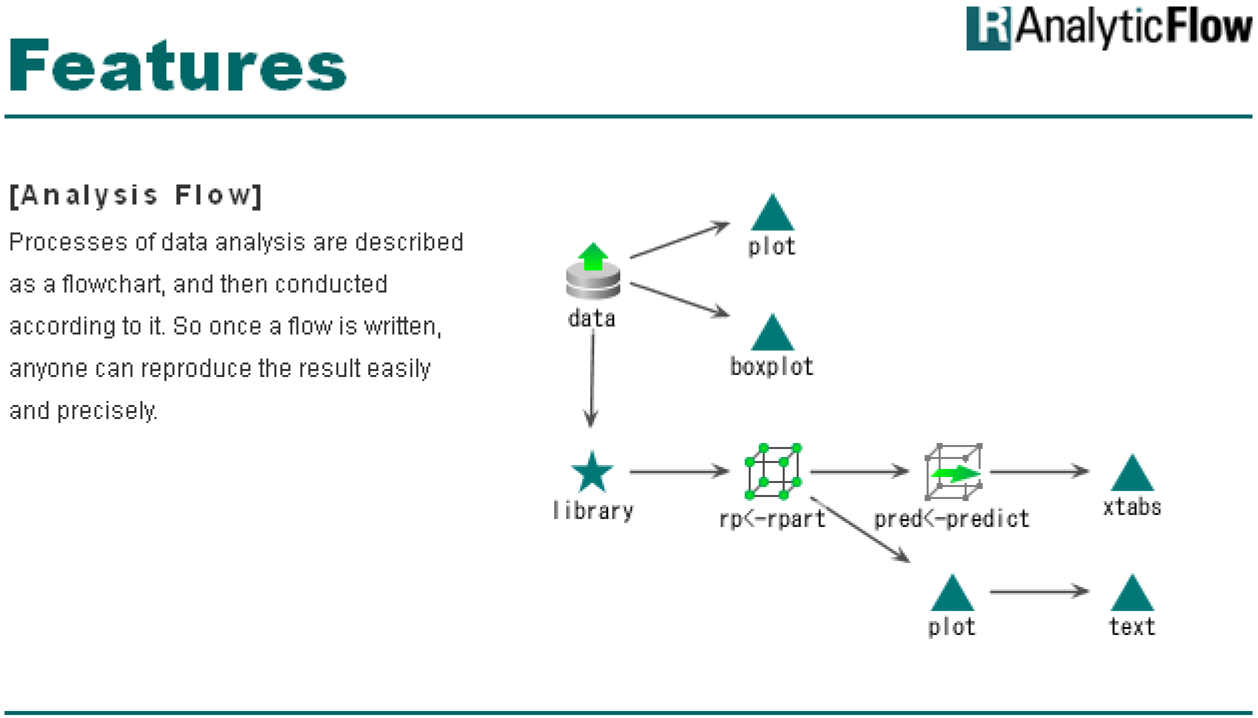
A3 — X

	A
1	
2	Dane do
3	X
4	0,178147
5	0,769566
6	0,82564
7	0,857405
8	0,768989
9	0,001861
10	0,393118
11	0,802951
12	0,080386
13	0,966064
14	0,64321
15	0,658135

R is somewhat ascetic, isn't it?

4. R Analytic FLow

http://www.ef-prime.com/products/ranalyticflow_en/

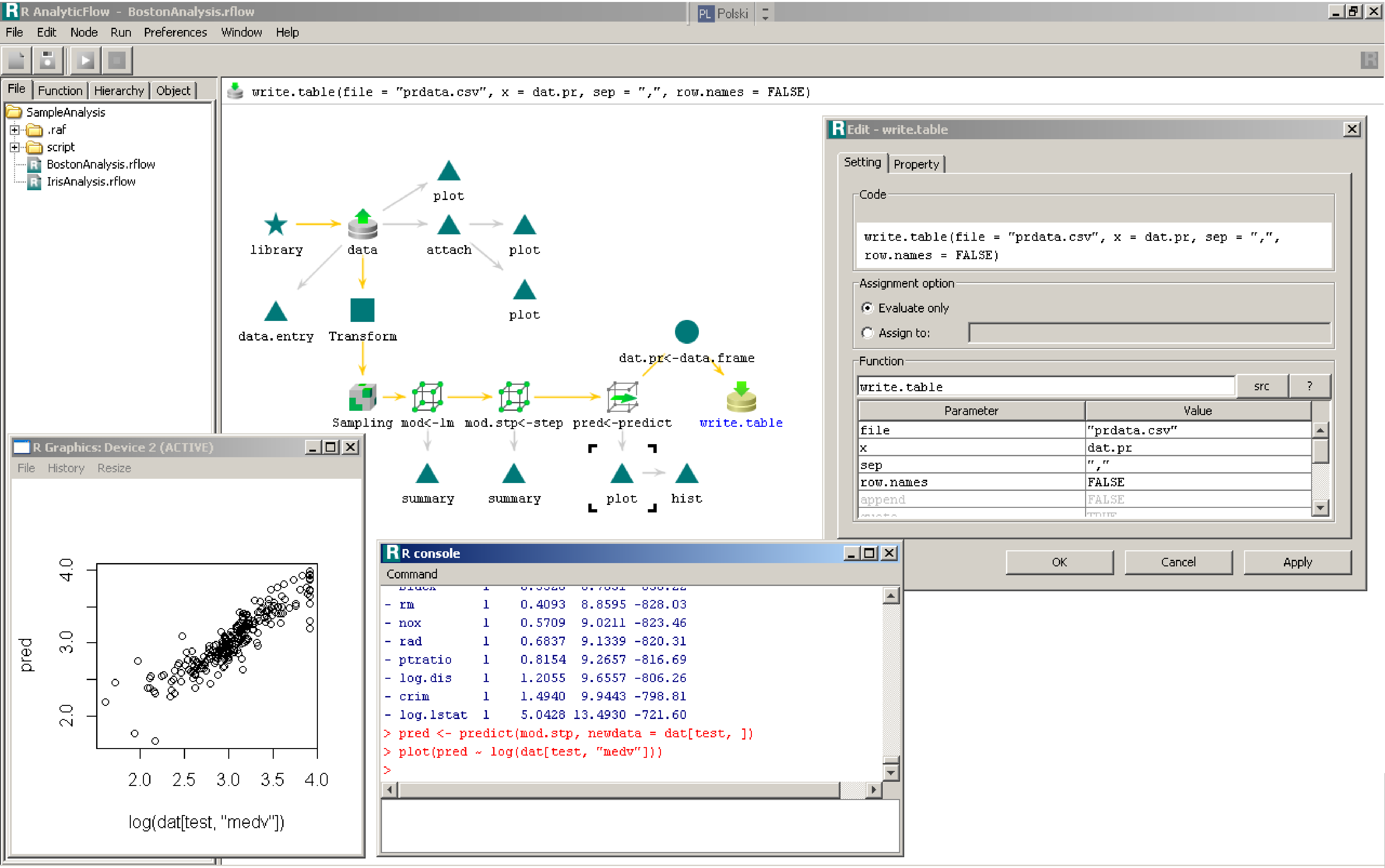


Features

[Analysis Flow]

Processes of data analysis are described as a flowchart, and then conducted according to it. So once a flow is written, anyone can reproduce the result easily and precisely.

R Analaytic Flow



```
write.table(file = "prdata.csv", x = dat.pr, sep = ",", row.names = FALSE)
```

Code

```
write.table(file = "prdata.csv", x = dat.pr, sep = ",",
row.names = FALSE)
```

Assignment option

● Evaluate only

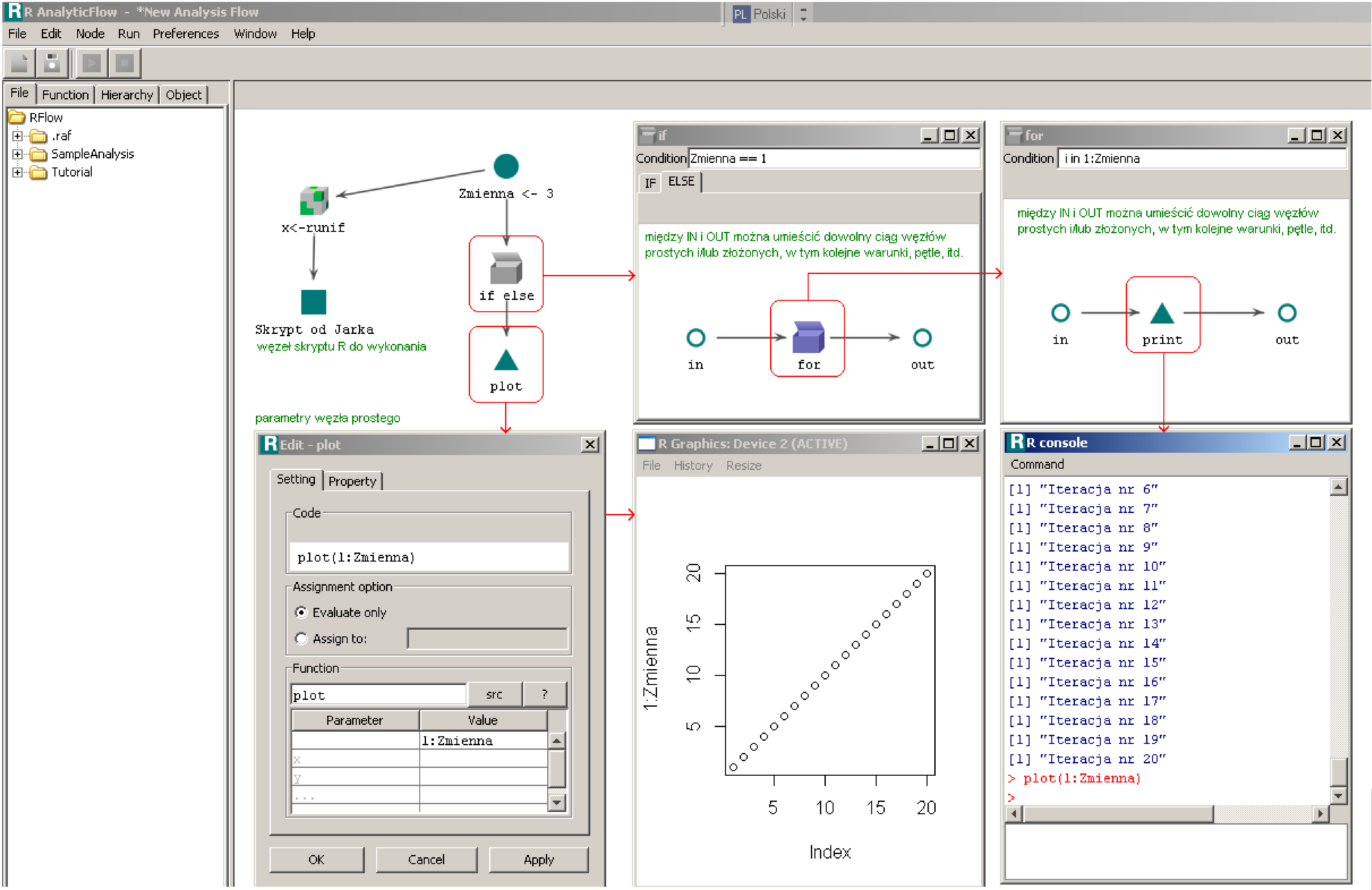
○ Assign to:

Function: write.table

Parameter	Value
file	"prdata.csv"
x	dat.pr
sep	","
row.names	FALSE
append	FALSE

```
- rm         1    0.4093  8.8595 -828.03
- nox        1    0.5709  9.0211 -823.46
- rad        1    0.6837  9.1339 -820.31
- ptratio    1    0.8154  9.2657 -816.69
- log.dis    1    1.2055  9.6557 -806.26
- crim       1    1.4940  9.9443 -798.81
- log.lstat  1    5.0428 13.4930 -721.60
> pred <- predict(mod.stp, newdata = dat[test, ])
> plot(pred ~ log(dat[test, "medv"]))
>
```

R Analaytic Flow



węzeł skryptu R do wykonania

parametry węzła prostego

między IN i OUT można umieścić dowolny ciąg węzłów prostych i/lub złożonych, w tym kolejne warunki, pętle, itd.

if – Condition: Zmienna == 1

for – Condition: i in 1:Zmienna

Edit - plot:

Code: `plot(1:Zmienna)`

Assignment option: ● Evaluate only ○ Assign to:

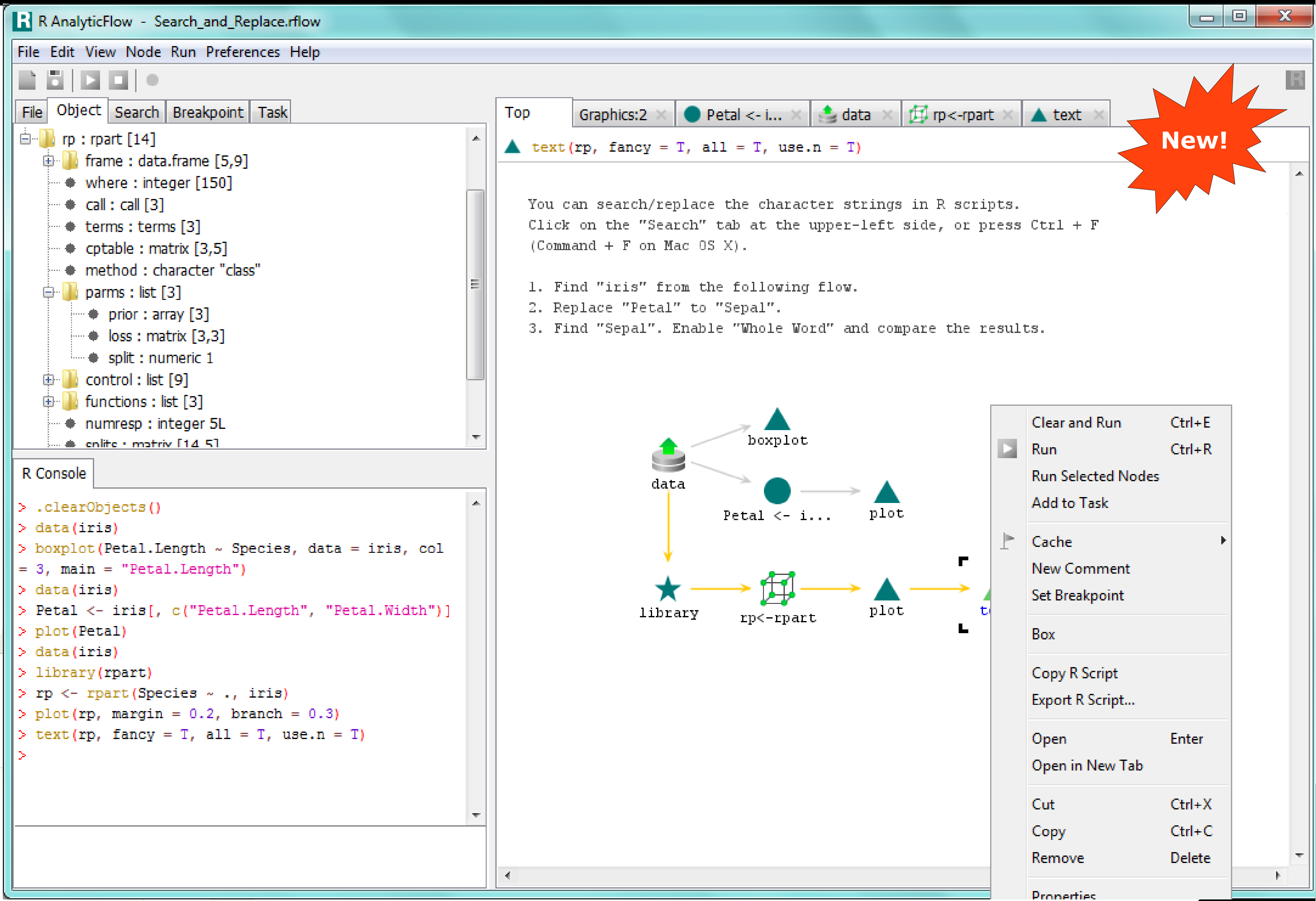
Function: plot

Parameter	Value
	1:Zmienna
x	
y	
...	

R console:

```
[1] "Iteracja nr 6"
[1] "Iteracja nr 7"
[1] "Iteracja nr 8"
[1] "Iteracja nr 9"
[1] "Iteracja nr 10"
[1] "Iteracja nr 11"
[1] "Iteracja nr 12"
[1] "Iteracja nr 13"
[1] "Iteracja nr 14"
[1] "Iteracja nr 15"
[1] "Iteracja nr 16"
[1] "Iteracja nr 17"
[1] "Iteracja nr 18"
[1] "Iteracja nr 19"
[1] "Iteracja nr 20"
> plot(1:Zmienna)
>
```

R Analaytic Flow



New!

```
rp : rpart [14]
  frame : data.frame [5,9]
  where : integer [150]
  call : call [3]
  terms : terms [3]
  cptable : matrix [3,5]
  method : character "class"
  parms : list [3]
    prior : array [3]
    loss : matrix [3,3]
    split : numeric 1
  control : list [9]
  functions : list [3]
  numresp : integer 5L
  splits : matrix [14,5]
```

```
text(rp, fancy = T, all = T, use.n = T)
```

You can search/replace the character strings in R scripts. Click on the "Search" tab at the upper-left side, or press Ctrl + F (Command + F on Mac OS X).

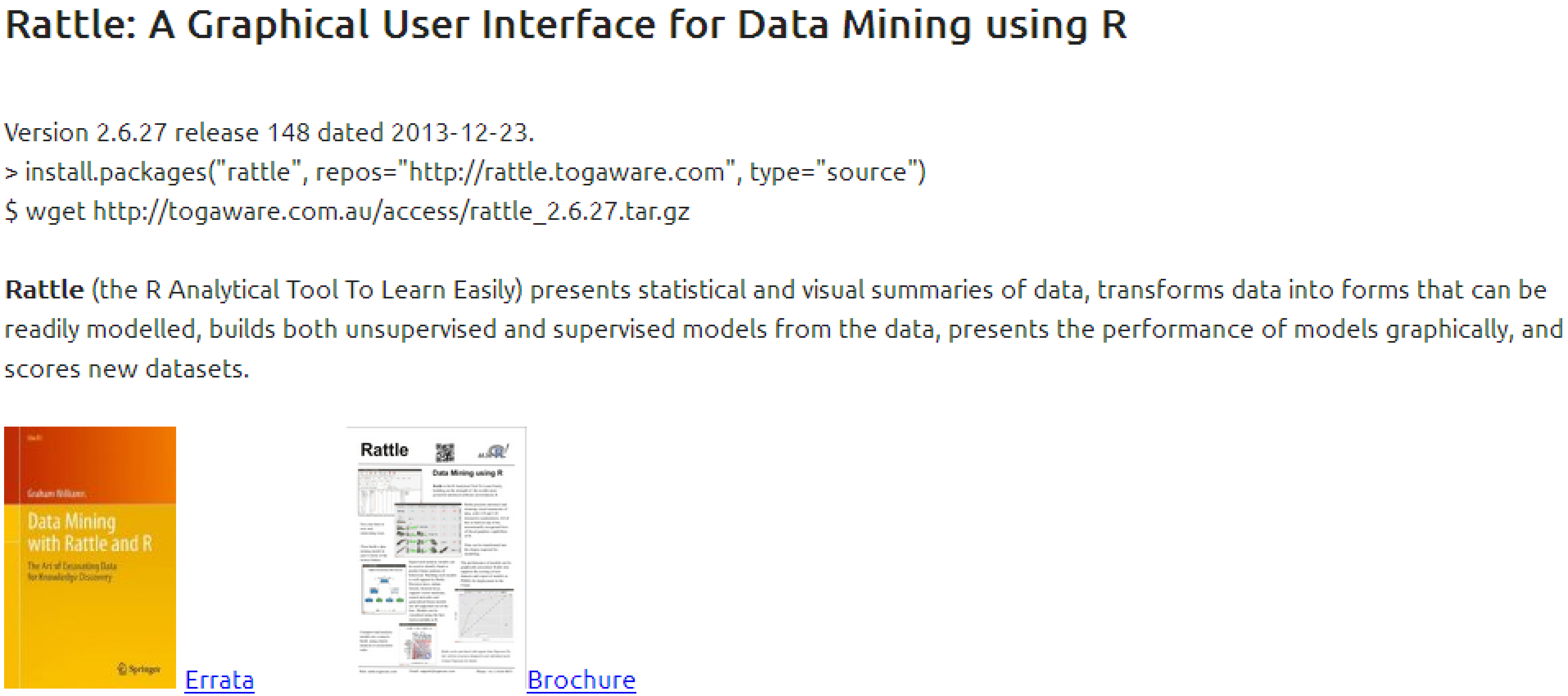
1. Find "iris" from the following flow.
2. Replace "Petal" to "Sepal".
3. Find "Sepal". Enable "Whole Word" and compare the results.

```
> .clearObjects()
> data(iris)
> boxplot(Petal.Length ~ Species, data = iris, col
= 3, main = "Petal.Length")
> data(iris)
> Petal <- iris[, c("Petal.Length", "Petal.Width")]
> plot(Petal)
> data(iris)
> library(rpart)
> rp <- rpart(Species ~ ., iris)
> plot(rp, margin = 0.2, branch = 0.3)
> text(rp, fancy = T, all = T, use.n = T)
>
```

R is somewhat ascetic, isn't it?

5. Rattle

http://rattle.togaware.com/



Rattle: A Graphical User Interface for Data Mining using R

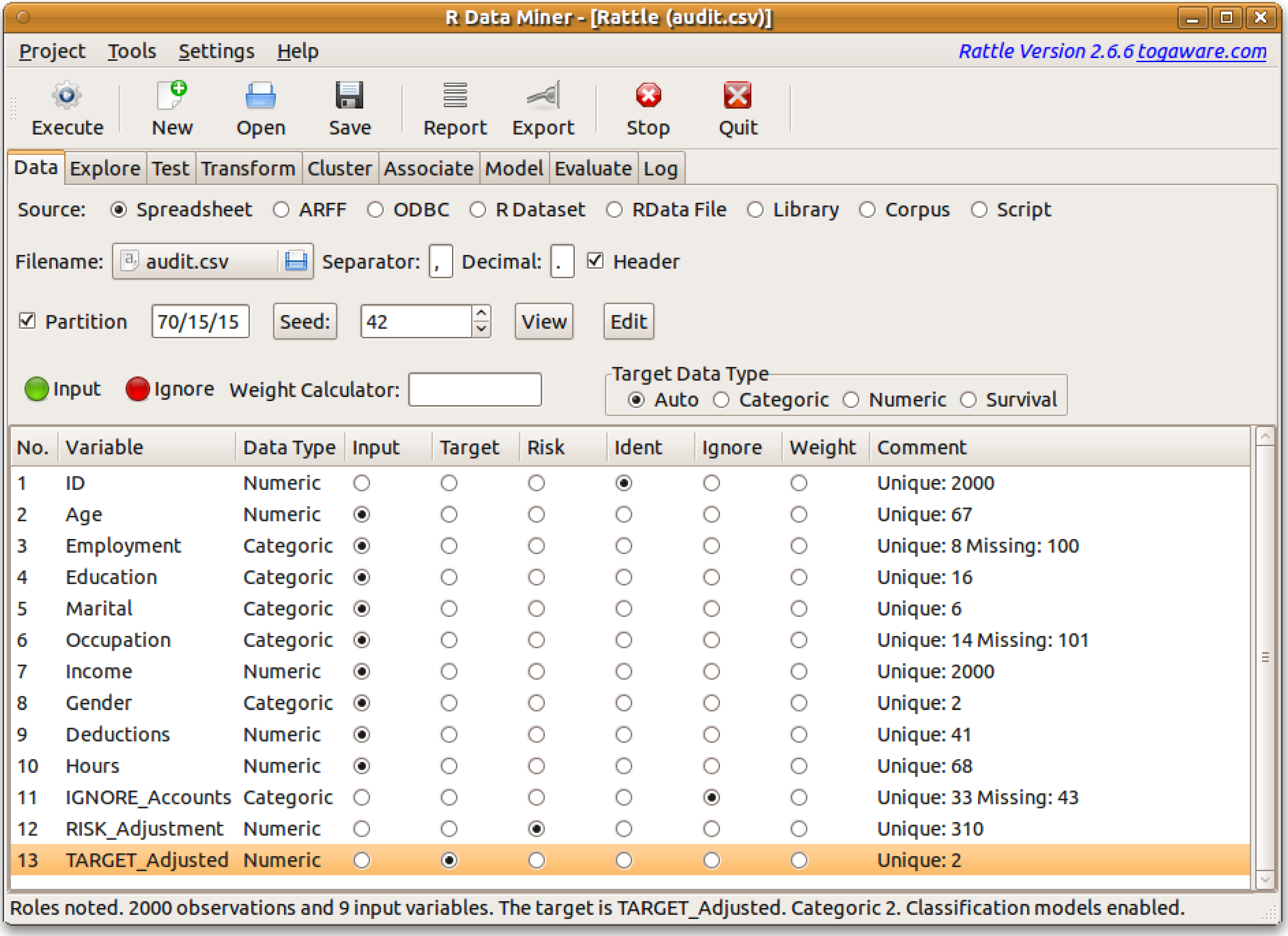
Version 2.6.27 release 148 dated 2013-12-23.

```
> install.packages("rattle", repos="http://rattle.togaware.com", type="source")
$ wget http://togaware.com.au/access/rattle_2.6.27.tar.gz
```

Rattle (the R Analytical Tool To Learn Easily) presents statistical and visual summaries of data, transforms data into forms that can be readily modelled, builds both unsupervised and supervised models from the data, presents the performance of models graphically, and scores new datasets.

[Errata]() [Brochure]()

Rattle your data



Source: ● Spreadsheet ○ ARFF ○ ODBC ○ R Dataset ○ RData File ○ Library ○ Corpus ○ Script

Filename: audit.csv Separator: , Decimal: . ☑ Header

☑ Partition 70/15/15 Seed: 42 View Edit

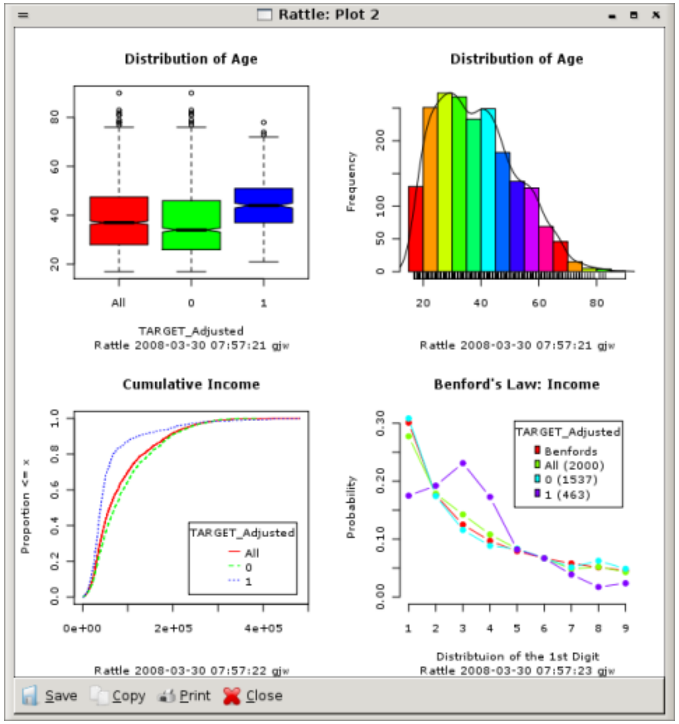
🟢 Input 🔴 Ignore Weight Calculator:

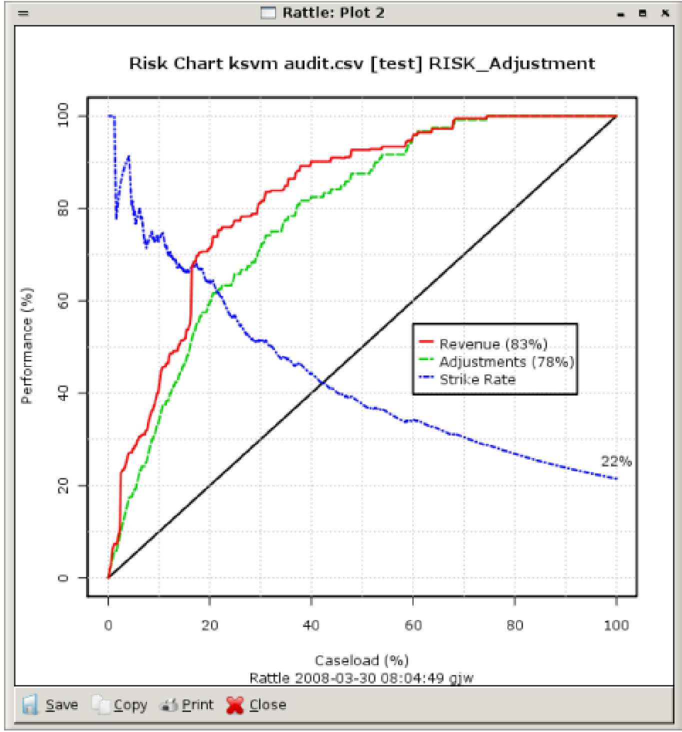
Target Data Type ● Auto ○ Categoric ○ Numeric ○ Survival

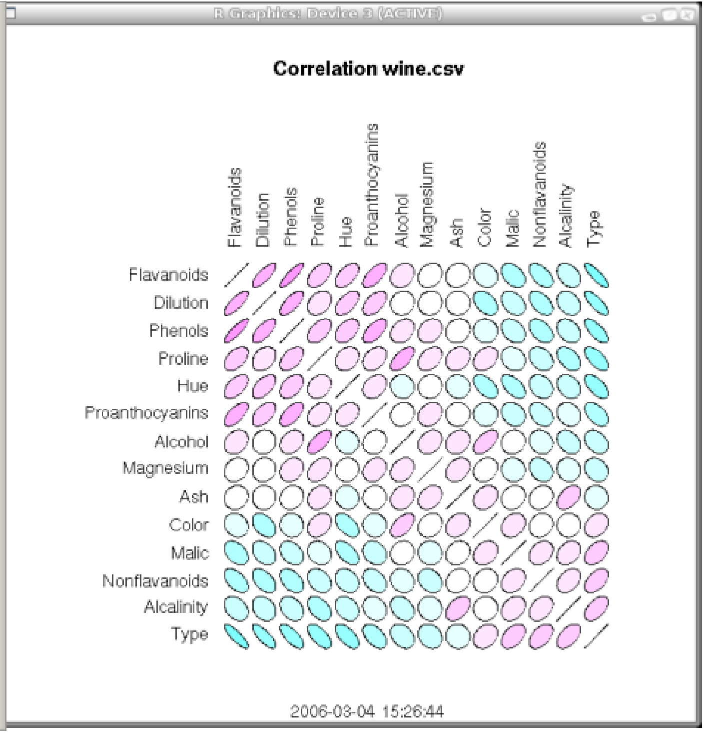
No.	Variable	Data Type	Input	Target	Risk	Ident	Ignore	Weight	Comment
1	ID	Numeric	○	○	○	●	○	○	Unique: 2000
2	Age	Numeric	●	○	○	○	○	○	Unique: 67
3	Employment	Categoric	●	○	○	○	○	○	Unique: 8 Missing: 100
4	Education	Categoric	●	○	○	○	○	○	Unique: 16
5	Marital	Categoric	●	○	○	○	○	○	Unique: 6
6	Occupation	Categoric	●	○	○	○	○	○	Unique: 14 Missing: 101
7	Income	Numeric	●	○	○	○	○	○	Unique: 2000
8	Gender	Categoric	●	○	○	○	○	○	Unique: 2
9	Deductions	Numeric	●	○	○	○	○	○	Unique: 41
10	Hours	Numeric	●	○	○	○	○	○	Unique: 68
11	IGNORE_Accounts	Categoric	○	○	○	○	●	○	Unique: 33 Missing: 43
12	RISK_Adjustment	Numeric	○	○	●	○	○	○	Unique: 310
13	TARGET_Adjusted	Numeric	○	●	○	○	○	○	Unique: 2

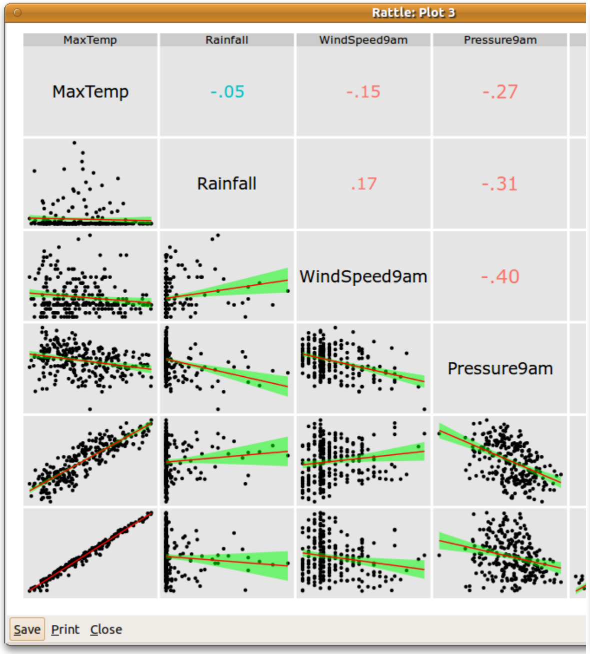
Roles noted. 2000 observations and 9 input variables. The target is TARGET_Adjusted. Categoric 2. Classification models enabled.

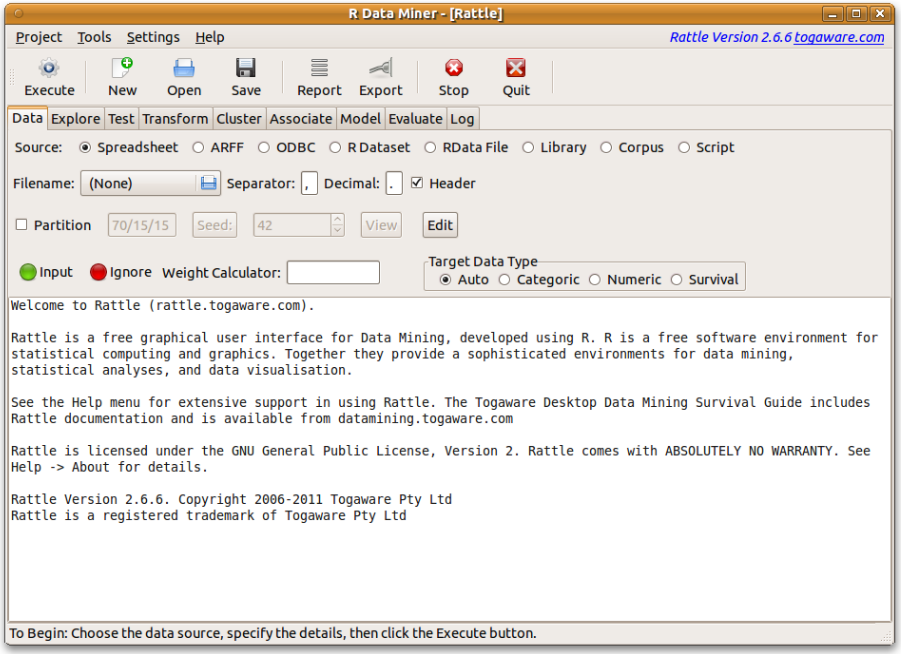
Rattle your data











Source: ◉ Spreadsheet ○ ARFF ○ ODBC ○ R Dataset ○ RData File ○ Library ○ Corpus ○ Script

Filename: (None) Separator: , Decimal: . ☑ Header

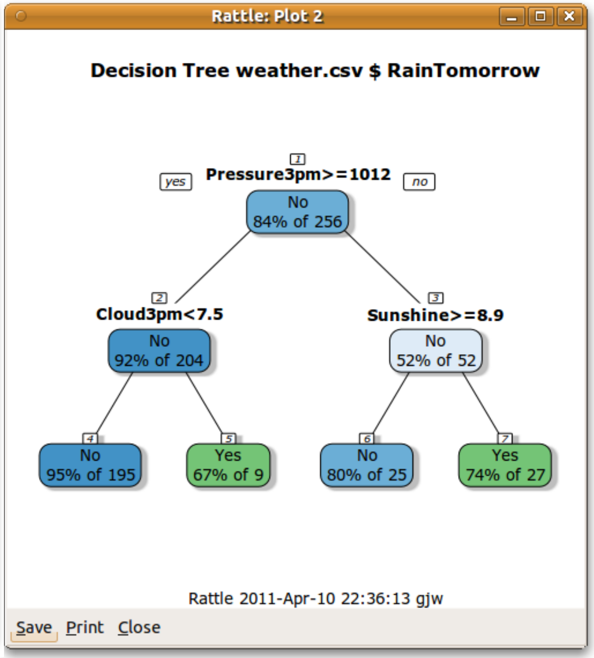
☐ Partition 70/15/15 Seed: 42 View Edit

● Input ● Ignore Weight Calculator:

Target Data Type ◉ Auto ○ Categoric ○ Numeric ○ Survival

```
Welcome to Rattle (rattle.togaware.com).

Rattle is a free graphical user interface for Data Mining, developed using R. R is a free software environment for
statistical computing and graphics. Together they provide a sophisticated environments for data mining,
statistical analyses, and data visualisation.

See the Help menu for extensive support in using Rattle. The Togaware Desktop Data Mining Survival Guide includes
Rattle documentation and is available from datamining.togaware.com

Rattle is licensed under the GNU General Public License, Version 2. Rattle comes with ABSOLUTELY NO WARRANTY. See
Help -> About for details.

Rattle Version 2.6.6. Copyright 2006-2011 Togaware Pty Ltd
Rattle is a registered trademark of Togaware Pty Ltd
```

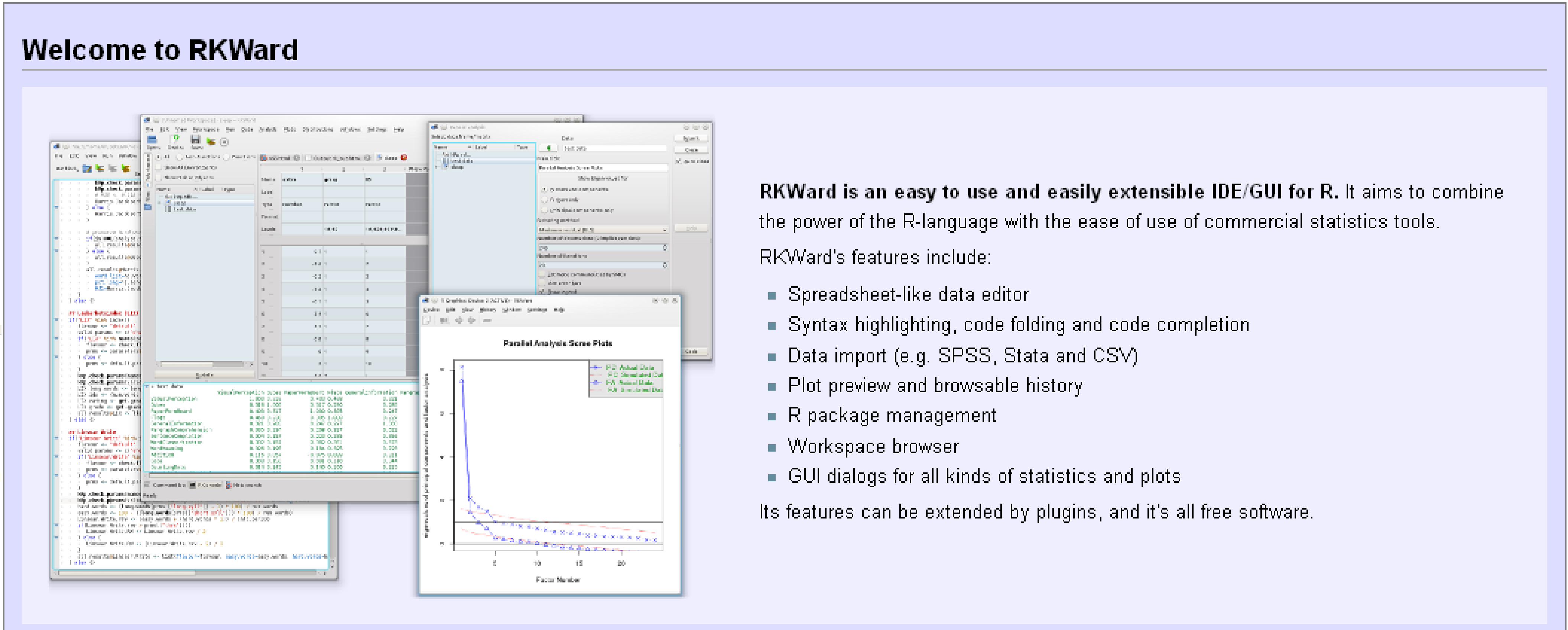
To Begin: Choose the data source, specify the details, then click the Execute button.



R is somewhat ascetic, isn't it?

6. RKWard (Linux only)

http://sourceforge.net/projects/rkward/



Welcome to RKWard

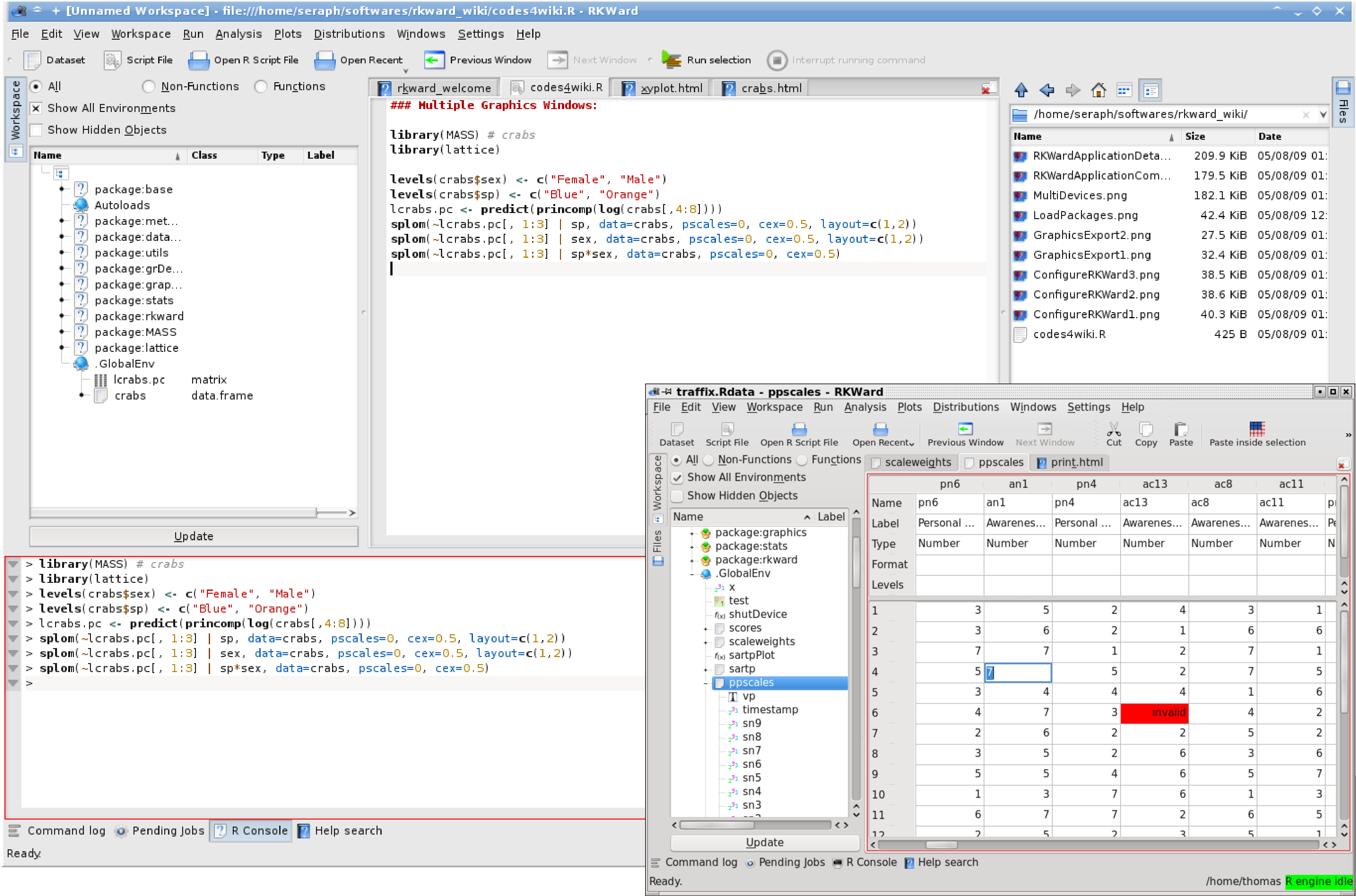
RKWard is an easy to use and easily extensible IDE/GUI for R. It aims to combine the power of the R-language with the ease of use of commercial statistics tools.

RKWard's features include:

- Spreadsheet-like data editor
- Syntax highlighting, code folding and code completion
- Data import (e.g. SPSS, Stata and CSV)
- Plot preview and browsable history
- R package management
- Workspace browser
- GUI dialogs for all kinds of statistics and plots

Its features can be extended by plugins, and it's all free software.

RKward



```
### Multiple Graphics Windows:

library(MASS) # crabs
library(lattice)

levels(crabs$sex) <- c("Female", "Male")
levels(crabs$sp) <- c("Blue", "Orange")
lcrabs.pc <- predict(princomp(log(crabs[,4:8])))
splom(~lcrabs.pc[, 1:3] | sp, data=crabs, pscales=0, cex=0.5, layout=c(1,2))
splom(~lcrabs.pc[, 1:3] | sex, data=crabs, pscales=0, cex=0.5, layout=c(1,2))
splom(~lcrabs.pc[, 1:3] | sp*sex, data=crabs, pscales=0, cex=0.5)
```

Name	Size	Date
RKWardApplicationDeta...	209.9 KiB	05/08/09 01:
RKWardApplicationCom...	179.5 KiB	05/08/09 01:
MultiDevices.png	182.1 KiB	05/08/09 01:
LoadPackages.png	42.4 KiB	05/08/09 12:
GraphicsExport2.png	27.5 KiB	05/08/09 01:
GraphicsExport1.png	32.4 KiB	05/08/09 01:
ConfigureRKWard3.png	38.5 KiB	05/08/09 01:
ConfigureRKWard2.png	38.6 KiB	05/08/09 01:
ConfigureRKWard1.png	40.3 KiB	05/08/09 01:
codes4wiki.R	425 B	05/08/09 01:

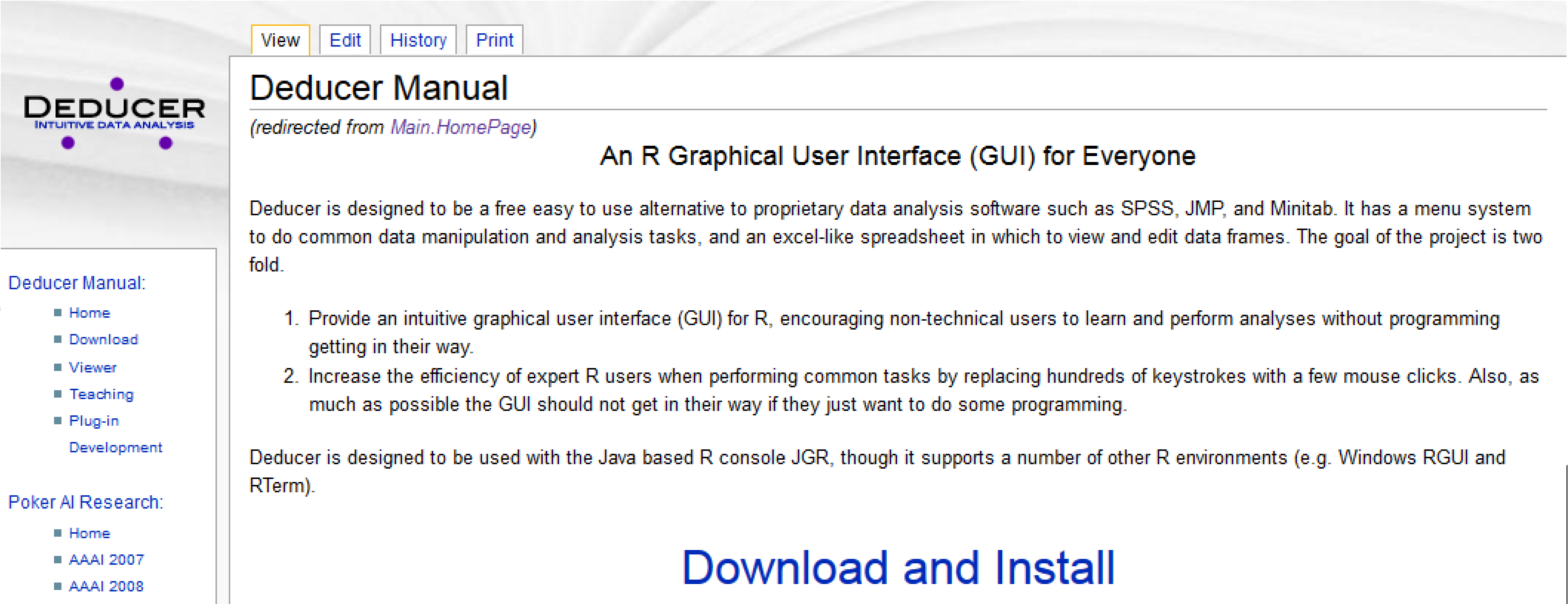
```
> library(MASS) # crabs
> library(lattice)
> levels(crabs$sex) <- c("Female", "Male")
> levels(crabs$sp) <- c("Blue", "Orange")
> lcrabs.pc <- predict(princomp(log(crabs[,4:8])))
> splom(~lcrabs.pc[, 1:3] | sp, data=crabs, pscales=0, cex=0.5, layout=c(1,2))
> splom(~lcrabs.pc[, 1:3] | sex, data=crabs, pscales=0, cex=0.5, layout=c(1,2))
> splom(~lcrabs.pc[, 1:3] | sp*sex, data=crabs, pscales=0, cex=0.5)
>
```

	pn6	an1	pn4	ac13	ac8	ac11
Name	pn6	an1	pn4	ac13	ac8	ac11
Label	Personal ...	Awarenes...	Personal ...	Awarenes...	Awarenes...	Awarenes...
Type	Number	Number	Number	Number	Number	Number
Format						
Levels						
1	3	5	2	4	3	1
2	3	6	2	1	6	6
3	7	7	1	2	7	1
4	5	7	5	2	7	5
5	3	4	4	4	1	6
6	4	7	3	invalid	4	2
7	2	6	2	2	5	2
8	3	5	2	6	3	6
9	5	5	4	6	5	7
10	1	3	7	6	1	3
11	6	7	7	2	6	5

R is somewhat ascetic, isn't it?

7. Deducer

http://www.deducer.org



Deducer Manual

(redirected from Main.HomePage)

An R Graphical User Interface (GUI) for Everyone

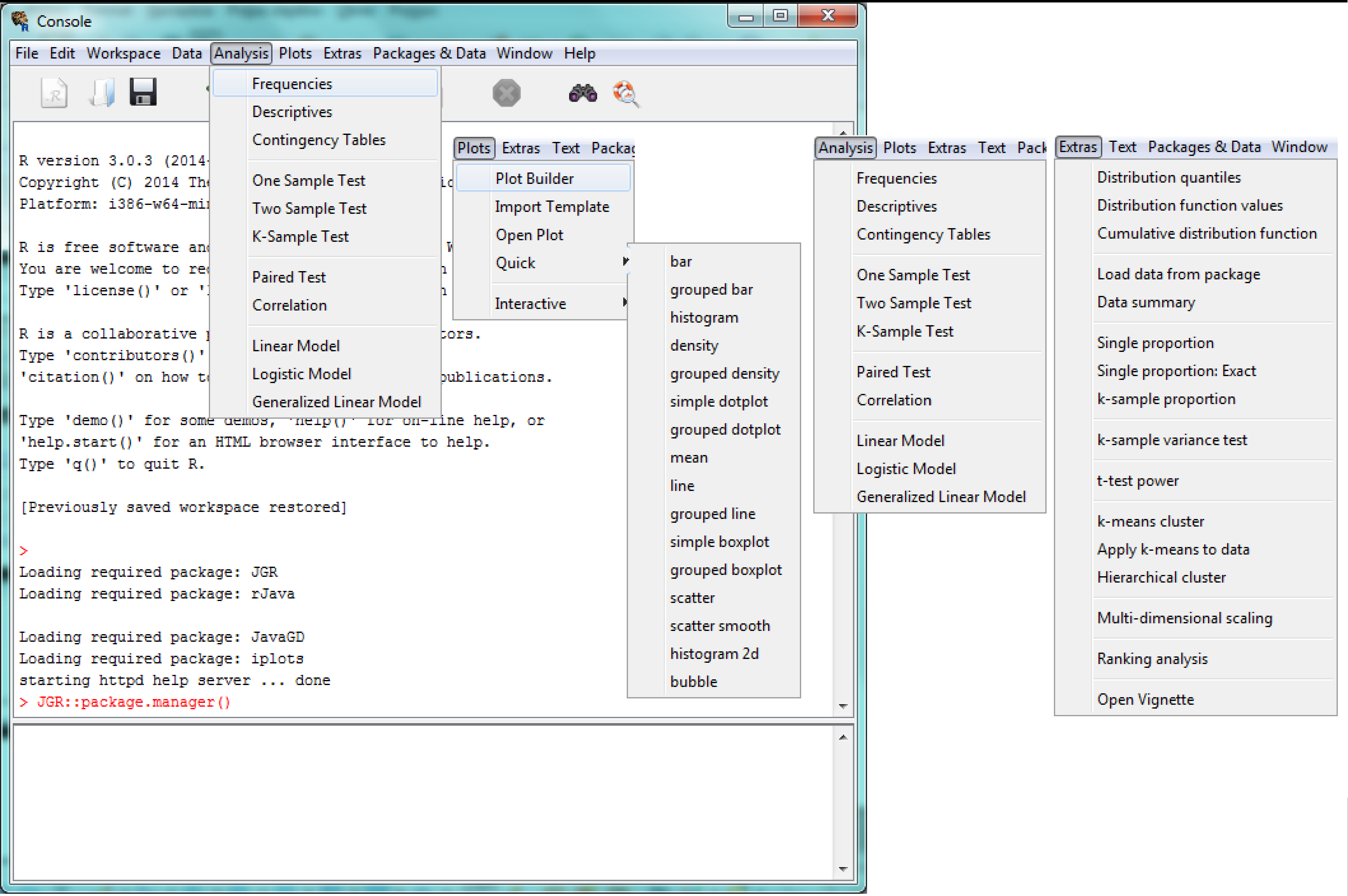
Deducer is designed to be a free easy to use alternative to proprietary data analysis software such as SPSS, JMP, and Minitab. It has a menu system to do common data manipulation and analysis tasks, and an excel-like spreadsheet in which to view and edit data frames. The goal of the project is two fold.

1. Provide an intuitive graphical user interface (GUI) for R, encouraging non-technical users to learn and perform analyses without programming getting in their way.
2. Increase the efficiency of expert R users when performing common tasks by replacing hundreds of keystrokes with a few mouse clicks. Also, as much as possible the GUI should not get in their way if they just want to do some programming.

Deducer is designed to be used with the Java based R console JGR, though it supports a number of other R environments (e.g. Windows RGUI and RTerm).

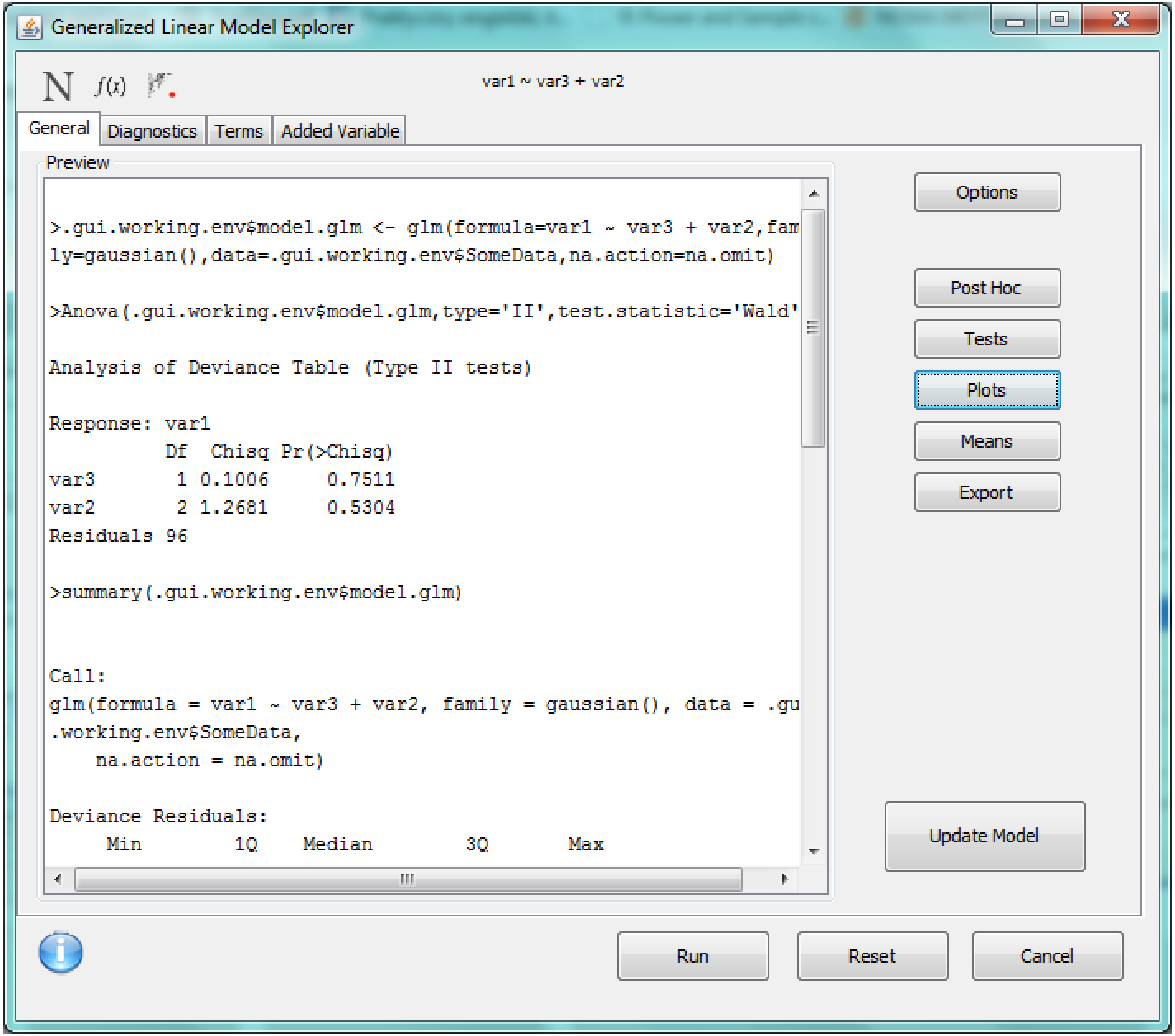
Download and Install

Deducer

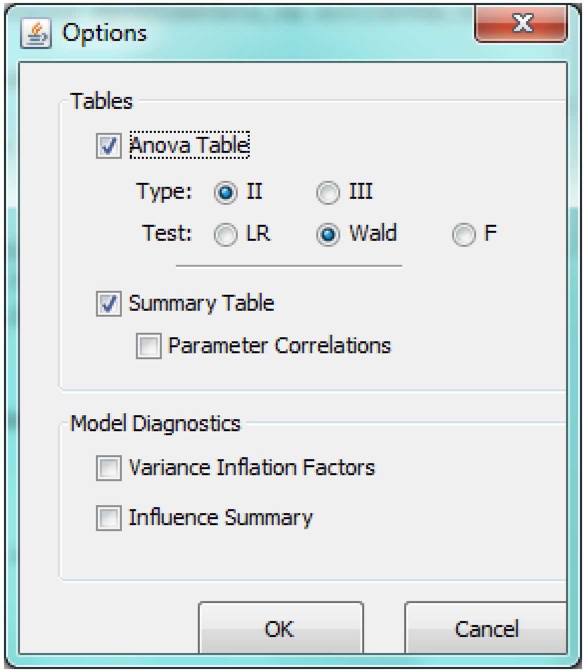


```
R version 3.0.3 (2014-
Copyright (C) 2014 The
Platform: i386-w64-mi

R is free software an
You are welcome to re
Type 'license()' or '

R is a collaborative
Type 'contributors()'
'citation()' on how t

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

>
Loading required package: JGR
Loading required package: rJava

Loading required package: JavaGD
Loading required package: iplots
starting httpd help server ... done
> JGR::package.manager()
```

Deducer



```
>.gui.working.env$model.glm <- glm(formula=var1 ~ var3 + var2,fam
ly=gaussian(),data=.gui.working.env$SomeData,na.action=na.omit)

>Anova(.gui.working.env$model.glm,type='II',test.statistic='Wald'

Analysis of Deviance Table (Type II tests)

Response: var1
          Df  Chisq Pr(>Chisq)
var3       1 0.1006     0.7511
var2       2 1.2681     0.5304
Residuals 96

>summary(.gui.working.env$model.glm)


Call:
glm(formula = var1 ~ var3 + var2, family = gaussian(), data = .gu
.working.env$SomeData,
    na.action = na.omit)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
```



Tables

☑ Anova Table

Type: ● II ○ III

Test: ○ LR ● Wald ○ F

☑ Summary Table

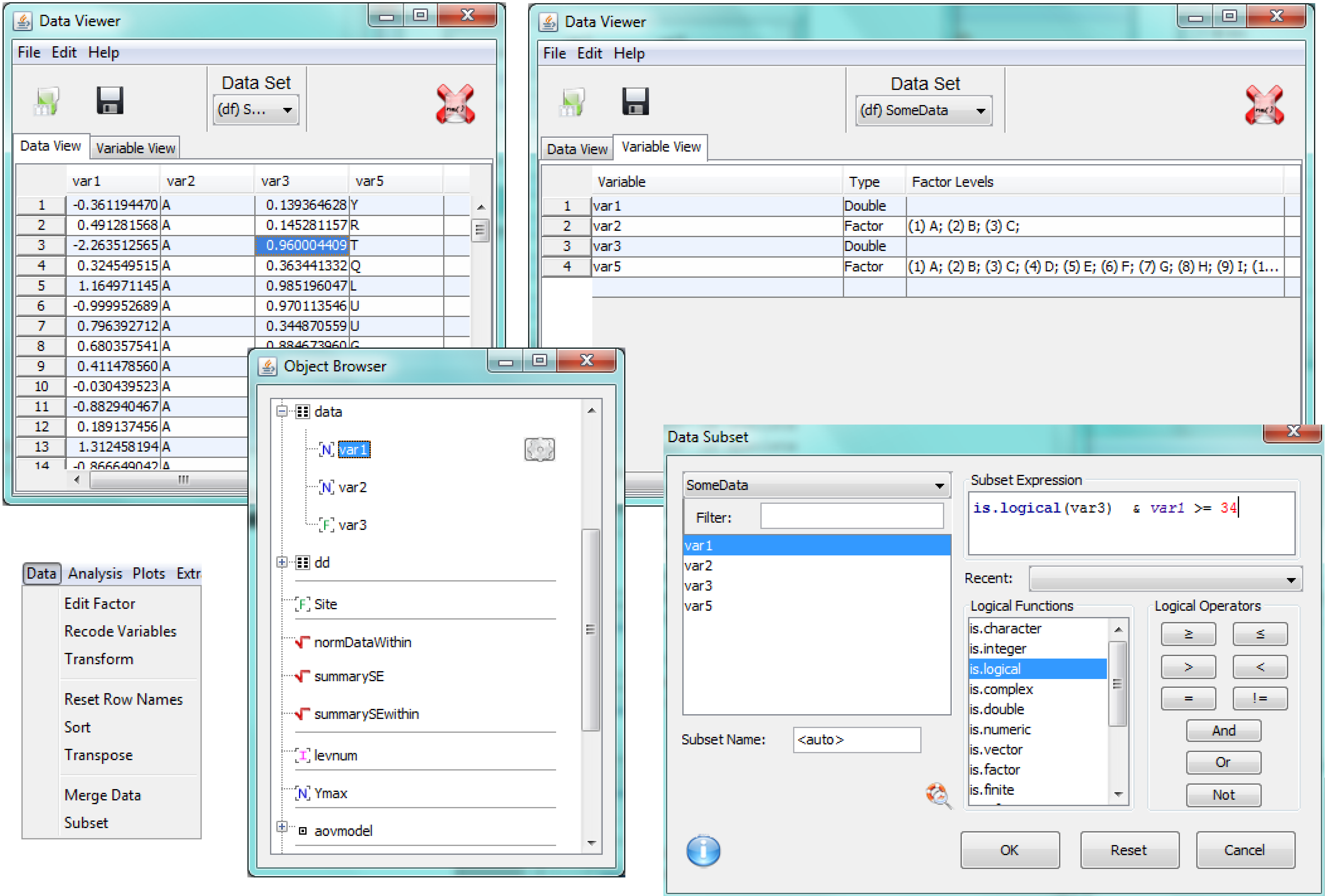
☐ Parameter Correlations

Model Diagnostics

☐ Variance Inflation Factors

☐ Influence Summary

Deducer



	var1	var2	var3	var5
1	-0.361194470	A	0.139364628	Y
2	0.491281568	A	0.145281157	R
3	-2.263512565	A	0.960004409	T
4	0.324549515	A	0.363441332	Q
5	1.164971145	A	0.985196047	L
6	-0.999952689	A	0.970113546	U
7	0.796392712	A	0.344870559	U
8	0.680357541	A	0.884673960	G
9	0.411478560	A		
10	-0.030439523	A		
11	-0.882940467	A		
12	0.189137456	A		
13	1.312458194	A		
14	-0.866649042	A		

Data Set: (df) SomeData

	Variable	Type	Factor Levels
1	var1	Double	
2	var2	Factor	(1) A; (2) B; (3) C;
3	var3	Double	
4	var5	Factor	(1) A; (2) B; (3) C; (4) D; (5) E; (6) F; (7) G; (8) H; (9) I; (1...

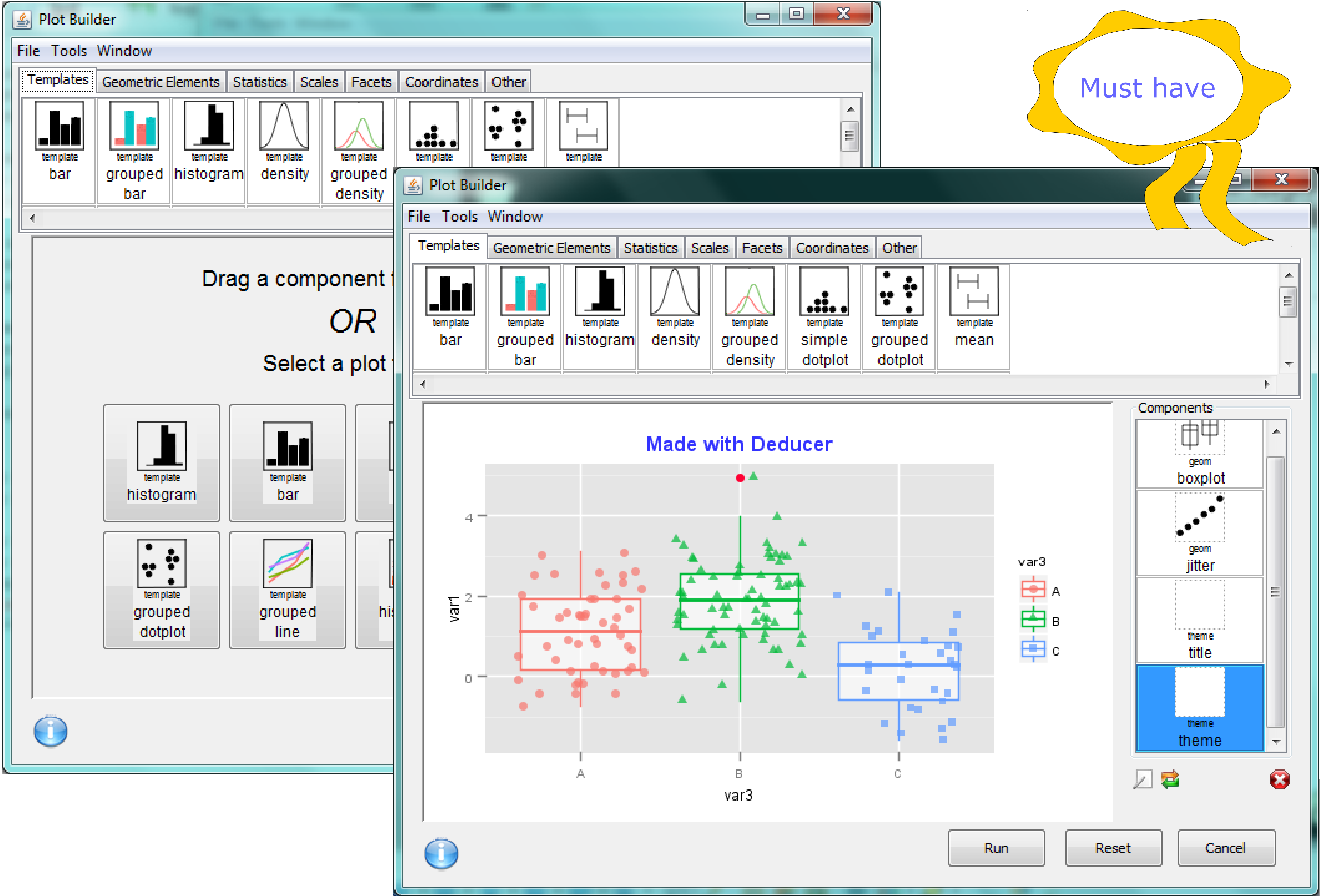
SomeData

Subset Expression:

```
is.logical(var3)  & var1 >= 34
```

Subset Name: <auto>

Deducer

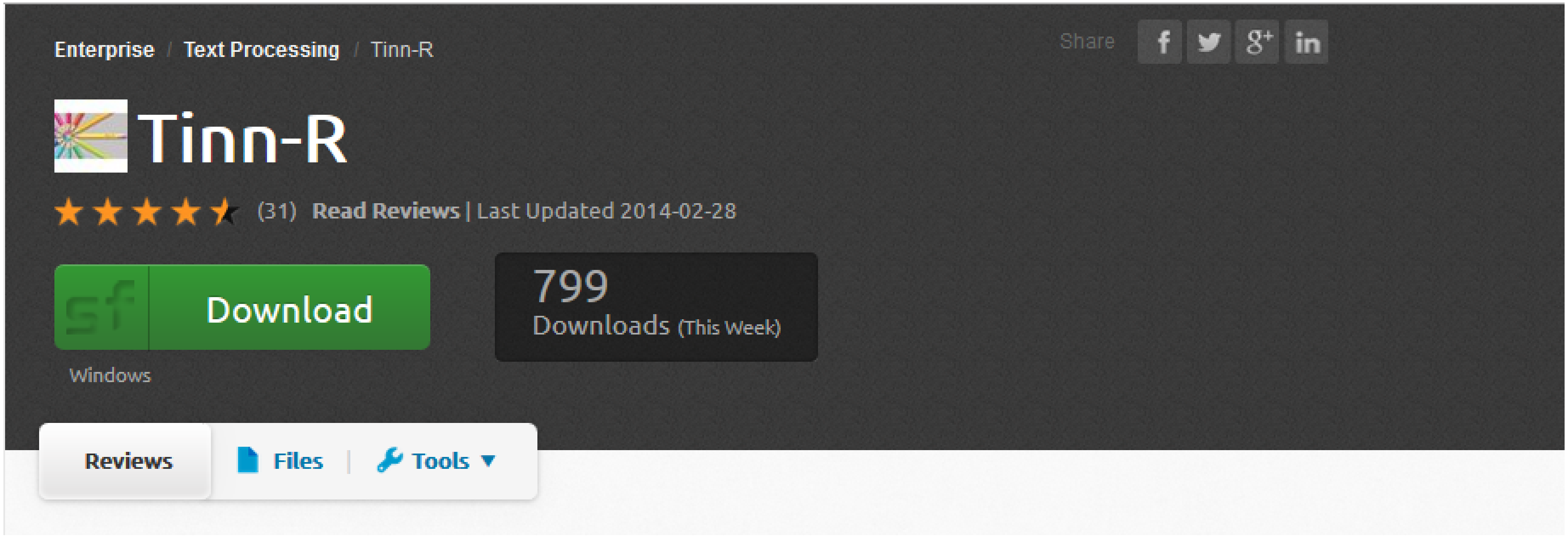


Drag a component
OR
Select a plot

R is somewhat ascetic, isn't it?

8. TinnR

http://sourceforge.net/projects/tinn-r



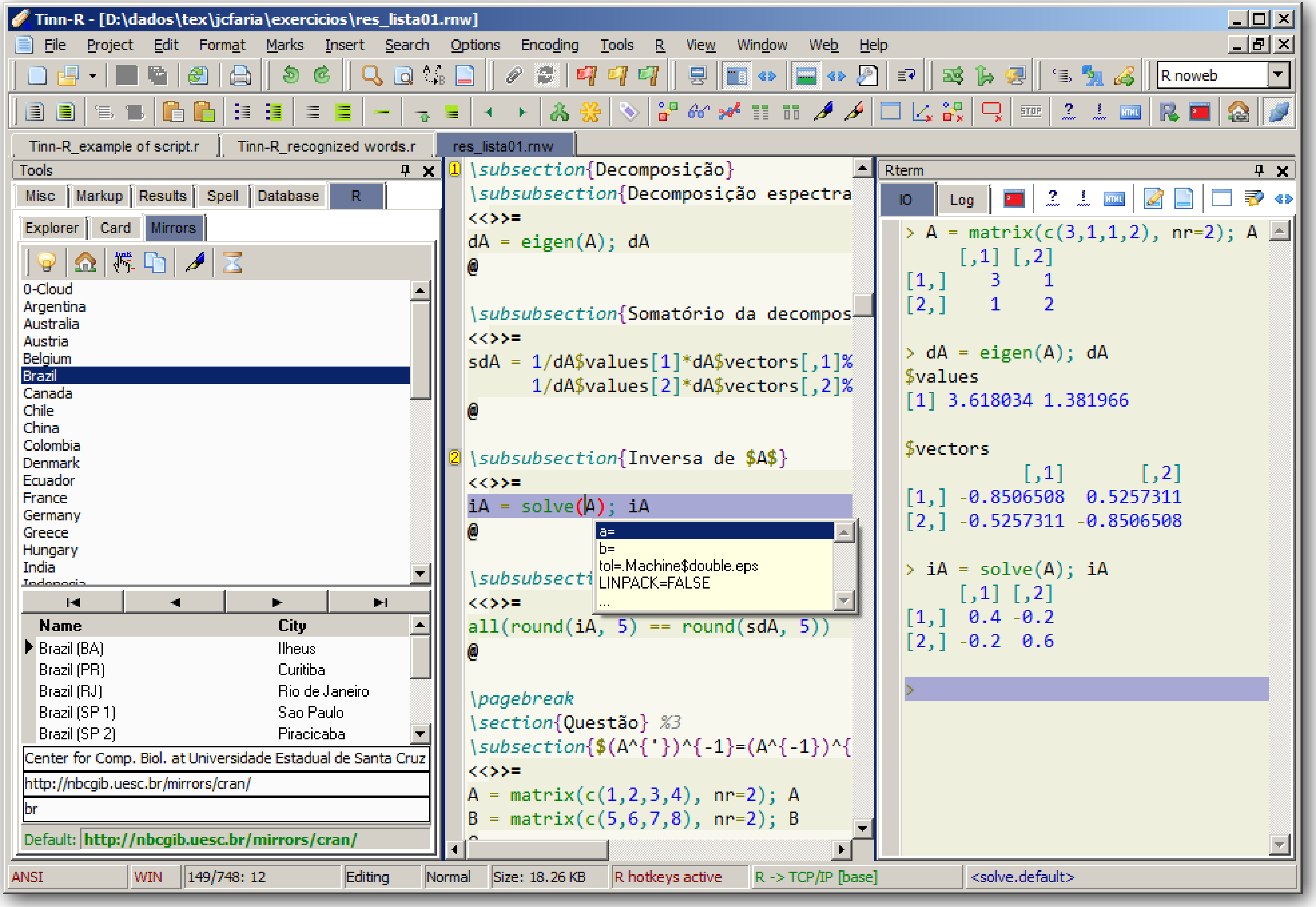
Enterprise / Text Processing / Tinn-R

Tinn-R

(31) Read Reviews | Last Updated 2014-02-28

799 Downloads (This Week)

TinnR



Tinn-R - [D:\dados\tex\jcfaria\exercicios\res_lista01.rnw]

0-Cloud
Argentina
Australia
Austria
Belgium
Brazil
Canada
Chile
China
Colombia
Denmark
Ecuador
France
Germany
Greece
Hungary
India

Name	City
Brazil (BA)	Ilheus
Brazil (PR)	Curitiba
Brazil (RJ)	Rio de Janeiro
Brazil (SP 1)	Sao Paulo
Brazil (SP 2)	Piracicaba

Center for Comp. Biol. at Universidade Estadual de Santa Cruz
http://nbcgib.uesc.br/mirrors/cran/
br

Default: **http://nbcgib.uesc.br/mirrors/cran/**

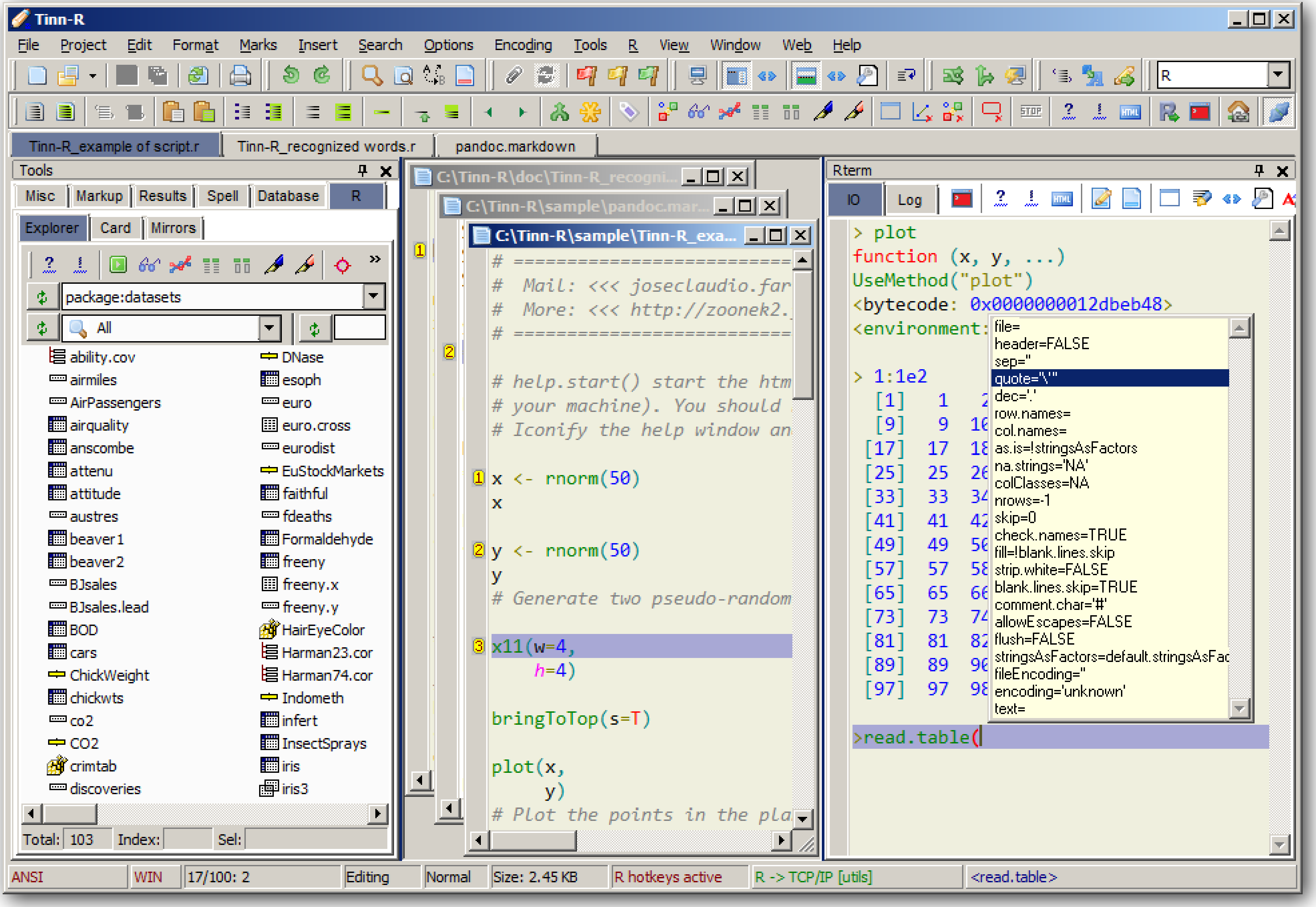
```
\subsection{Decomposição}
\subsubsection{Decomposição espectra
<<>>=
dA = eigen(A); dA
@

\subsubsection{Somatório da decompos
<<>>=
sdA = 1/dA$values[1]*dA$vectors[,1]%
      1/dA$values[2]*dA$vectors[,2]%
@

\subsubsection{Inversa de $A$}
<<>>=
iA = solve(A); iA
@

\subsubsecti
<<>>=
all(round(iA, 5) == round(sdA, 5))
@

\pagebreak
\section{Questão} %3
\subsection{$(A^{'})^{-1}=(A^{-1})^{
<<>>=
A = matrix(c(1,2,3,4), nr=2); A
B = matrix(c(5,6,7,8), nr=2); B
```

```
a=
b=
tol=.Machine$double.eps
LINPACK=FALSE
...
```

```
> A = matrix(c(3,1,1,2), nr=2); A
     [,1] [,2]
[1,]    3    1
[2,]    1    2

> dA = eigen(A); dA
$values
[1] 3.618034 1.381966

$vectors
           [,1]       [,2]
[1,] -0.8506508  0.5257311
[2,] -0.5257311 -0.8506508

> iA = solve(A); iA
     [,1] [,2]
[1,]  0.4 -0.2
[2,] -0.2  0.6

>
```

ANSI | WIN | 149/748: 12 | Editing | Normal | Size: 18.26 KB | R hotkeys active | R -> TCP/IP [base] | <solve.default>

TinnR



ability.cov
airmiles
AirPassengers
airquality
anscombe
attenu
attitude
austres
beaver1
beaver2
BJsales
BJsales.lead
BOD
cars
ChickWeight
chickwts
co2
CO2
crimtab
discoveries
DNase
esoph
euro
euro.cross
eurodist
EuStockMarkets
faithful
fdeaths
Formaldehyde
freeny
freeny.x
freeny.y
HairEyeColor
Harman23.cor
Harman74.cor
Indometh
infert
InsectSprays
iris
iris3

Total: 103 Index: Sel:

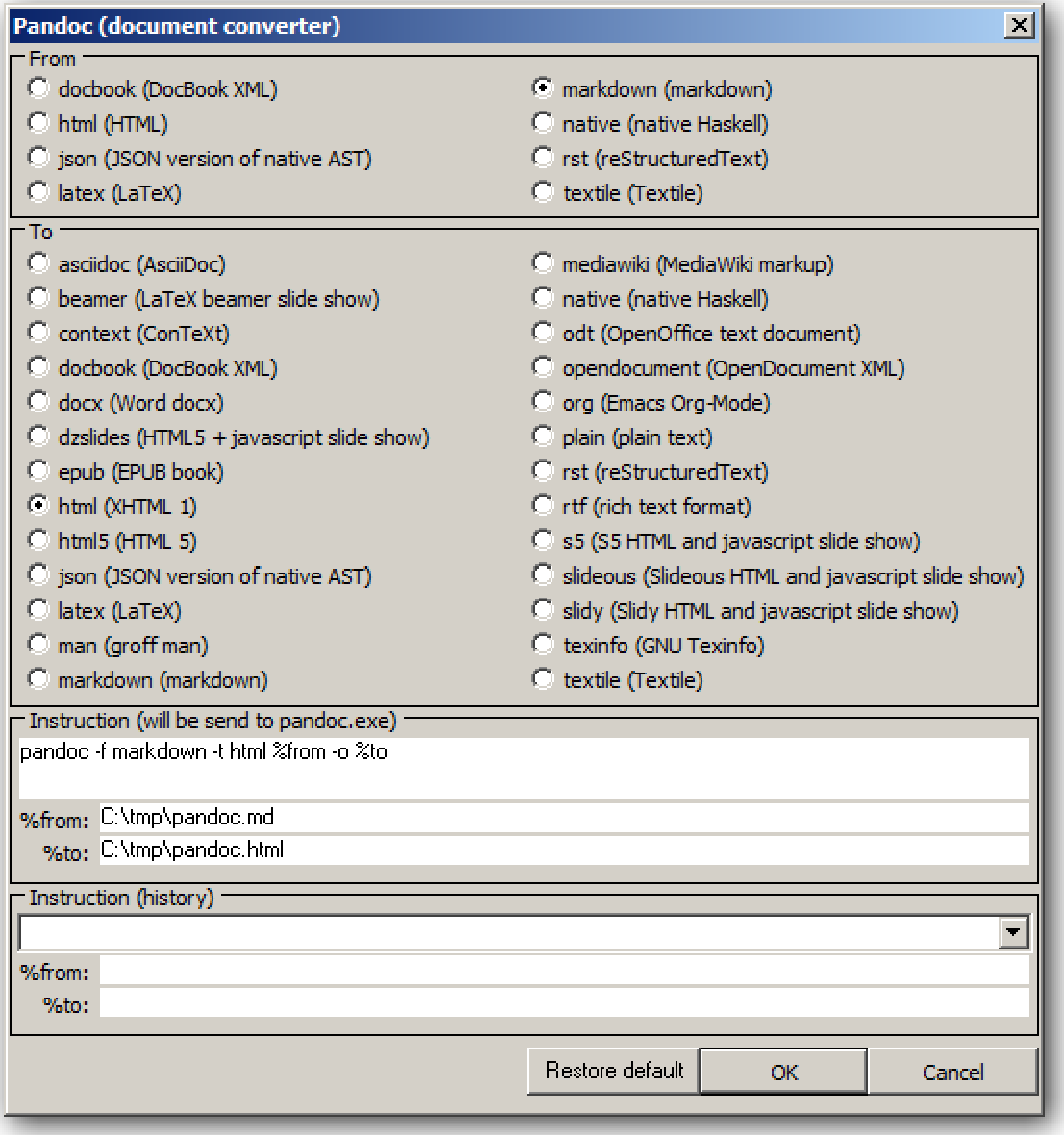
```
# ==========================
#  Mail: <<< joseclaudio.far
#  More: <<< http://zoonek2.
# ==========================

# help.start() start the htm
# your machine). You should
# Iconify the help window an

x <- rnorm(50)
x

y <- rnorm(50)
y
# Generate two pseudo-random

x11(w=4,
    h=4)

bringToTop(s=T)

plot(x,
     y)
# Plot the points in the pla
```

```
> plot
function (x, y, ...)
UseMethod("plot")
<bytecode: 0x0000000012dbeb48>
<environment:

> 1:1e2
  [1]   1   2
  [9]   9  16
 [17]  17  18
 [25]  25  26
 [33]  33  34
 [41]  41  42
 [49]  49  50
 [57]  57  58
 [65]  65  66
 [73]  73  74
 [81]  81  82
 [89]  89  90
 [97]  97  98

>read.table(
```

```
file=
header=FALSE
sep=''
quote="\'"
dec='.'
row.names=
col.names=
as.is=!stringsAsFactors
na.strings='NA'
colClasses=NA
nrows=-1
skip=0
check.names=TRUE
fill=!blank.lines.skip
strip.white=FALSE
blank.lines.skip=TRUE
comment.char='#'
allowEscapes=FALSE
flush=FALSE
stringsAsFactors=default.stringsAsFac
fileEncoding=''
encoding='unknown'
text=
```

ANSI | WIN | 17/100: 2 | Editing | Normal | Size: 2.45 KB | R hotkeys active | R -> TCP/IP [utils] | <read.table>

TinnR



Pandoc (document converter)

From

○ docbook (DocBook XML)	● markdown (markdown)
○ html (HTML)	○ native (native Haskell)
○ json (JSON version of native AST)	○ rst (reStructuredText)
○ latex (LaTeX)	○ textile (Textile)

To

○ asciidoc (AsciiDoc)	○ mediawiki (MediaWiki markup)
○ beamer (LaTeX beamer slide show)	○ native (native Haskell)
○ context (ConTeXt)	○ odt (OpenOffice text document)
○ docbook (DocBook XML)	○ opendocument (OpenDocument XML)
○ docx (Word docx)	○ org (Emacs Org-Mode)
○ dzslides (HTML5 + javascript slide show)	○ plain (plain text)
○ epub (EPUB book)	○ rst (reStructuredText)
● html (XHTML 1)	○ rtf (rich text format)
○ html5 (HTML 5)	○ s5 (S5 HTML and javascript slide show)
○ json (JSON version of native AST)	○ slideous (Slideous HTML and javascript slide show)
○ latex (LaTeX)	○ slidy (Slidy HTML and javascript slide show)
○ man (groff man)	○ texinfo (GNU Texinfo)
○ markdown (markdown)	○ textile (Textile)

Instruction (will be send to pandoc.exe)

```
pandoc -f markdown -t html %from -o %to
```

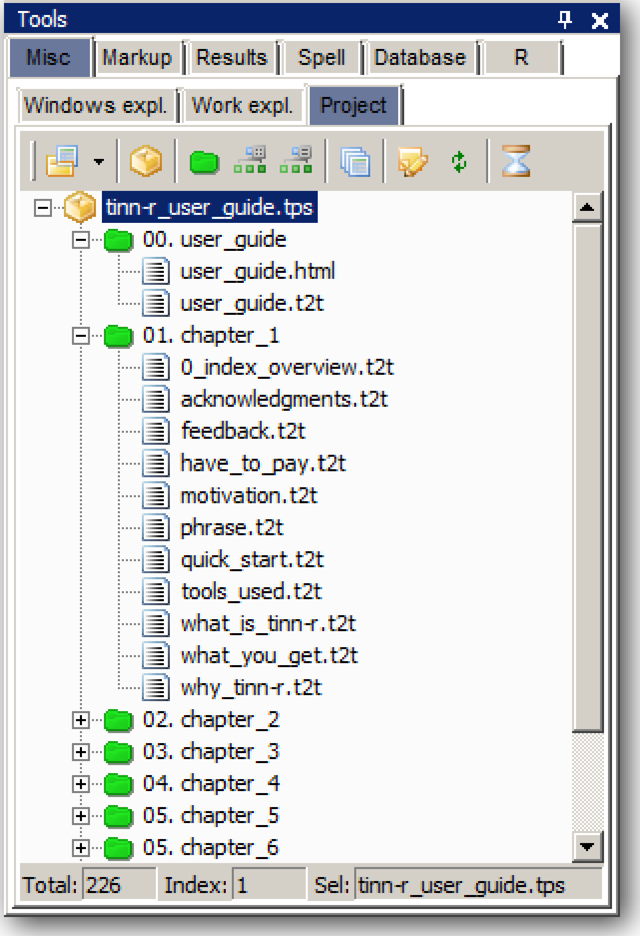
%from: C:\tmp\pandoc.md

%to: C:\tmp\pandoc.html

Instruction (history)

%from:

%to:



- tinn-r_user_guide.tps
  - 00. user_guide
    - user_guide.html
    - user_guide.t2t
  - 01. chapter_1
    - 0_index_overview.t2t
    - acknowledgments.t2t
    - feedback.t2t
    - have_to_pay.t2t
    - motivation.t2t
    - phrase.t2t
    - quick_start.t2t
    - tools_used.t2t
    - what_is_tinn-r.t2t
    - what_you_get.t2t
    - why_tinn-r.t2t
  - 02. chapter_2
  - 03. chapter_3
  - 04. chapter_4
  - 05. chapter_5
  - 05. chapter_6

Total: 226 Index: 1 Sel: tinn-r_user_guide.tps

13 reasons why **you will** love GNU R

I R is (extremely) cheap. In fact - it's free :)

II R has (extremely) wide range of capabilities

III R is (widely) supported by the world of science

IV R is supported by the community

V R is (increasingly) supported by the business

VI R is able to read data in many formats

VII Interoperability is easy to achieve

VIII R is truly cross-platform

IX R offers numerous ways of presenting data

X There are many options to optimize the code

XI R is able to handle large amount of data

XII R has a set of fancy tools and IDEs

XII FDA accepted using R for drug trials!

FDA: R is OK for drug trials!

In a poster presented at the UseR 2012 conference, FDA biostatistician Jae Brodsky reiterated the FDA policy regarding software used to prepare submissions for drug approvals with clinical trials: **Sponsors may use R in their submissions**. […] SAS **is not required** to be used for clinical trials.

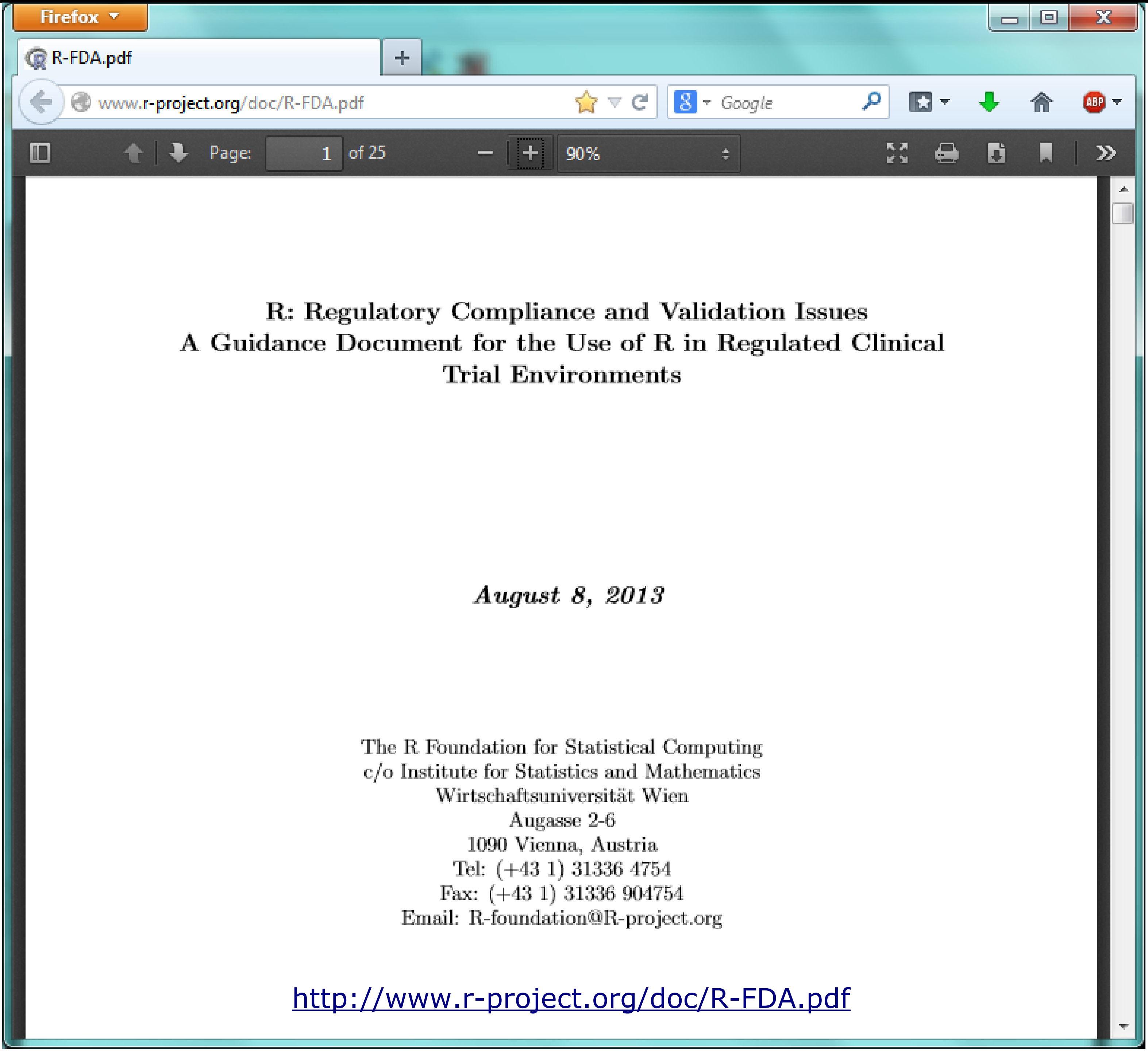
The FDA **does not endorse or require** any particular software to be used for clinical trial submissions, and there are **no regulations that restrict the use** of open source software (including R) at the FDA.

Nonetheless, any software (R included) used to prepare data analysis from clinical trials **must comply** with the various FDA regulations and guidances (e.g. 21 CFR Part 11). Even MS Excel can be made "21 CFR 11 Compliant".

The R Foundation helpfully **provides a guidance document** for the use of R in regulated clinical trial environments, which provides details of the specific FDA regulations and how R complies with them.

http://tinyurl.com/fda-r-ok

the (veRy important) guidance



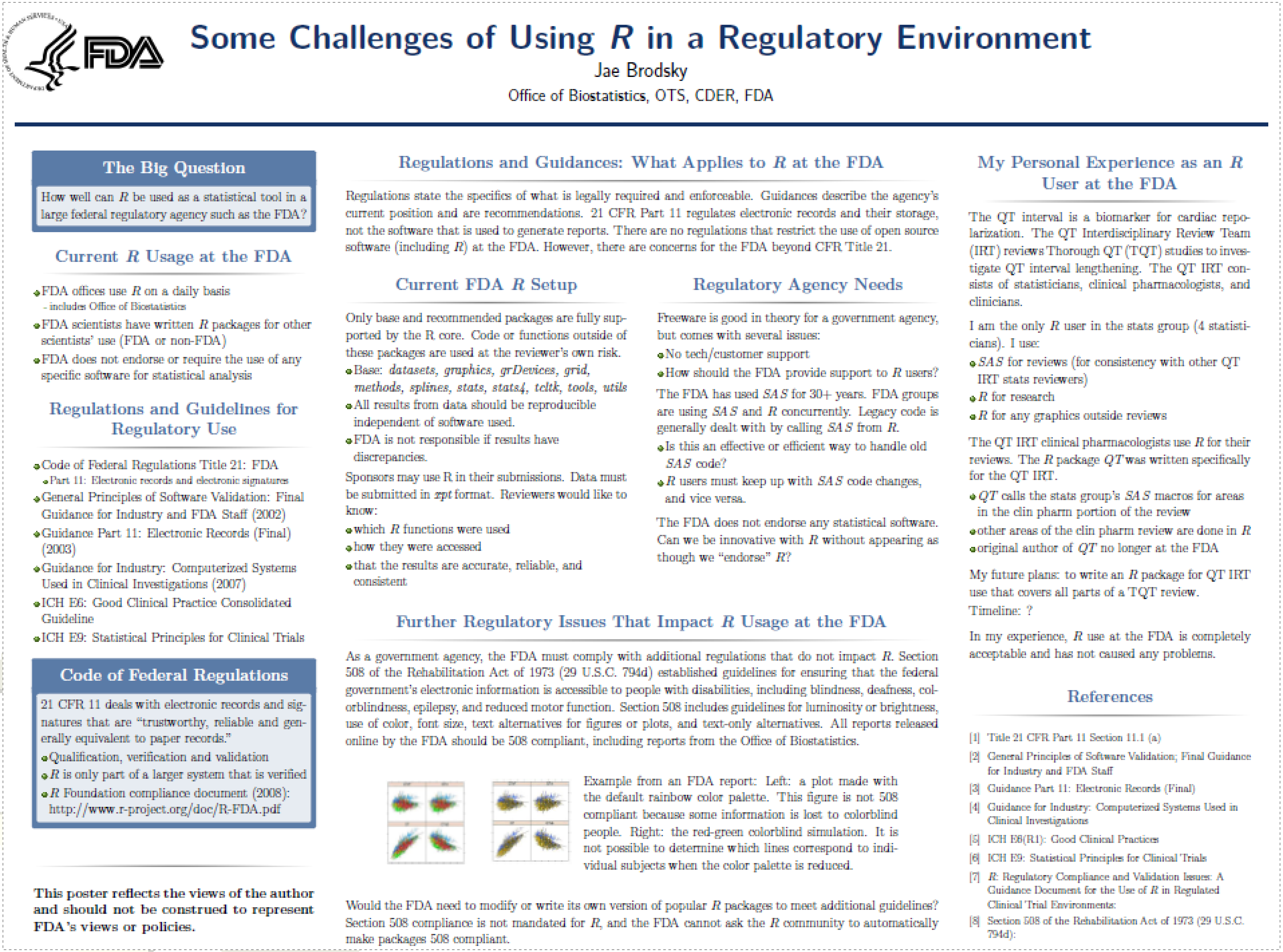
R: Regulatory Compliance and Validation Issues
A Guidance Document for the Use of R in Regulated Clinical Trial Environments

August 8, 2013

The R Foundation for Statistical Computing
c/o Institute for Statistics and Mathematics
Wirtschaftsuniversität Wien
Augasse 2-6
1090 Vienna, Austria
Tel: (+43 1) 31336 4754
Fax: (+43 1) 31336 904754
Email: R-foundation@R-project.org

http://www.r-project.org/doc/R-FDA.pdf

the (veRy important) poster



Some Challenges of Using *R* in a Regulatory Environment

Jae Brodsky

Office of Biostatistics, OTS, CDER, FDA

The Big Question

How well can *R* be used as a statistical tool in a large federal regulatory agency such as the FDA?

Current *R* Usage at the FDA

- FDA offices use *R* on a daily basis
  - includes Office of Biostatistics
- FDA scientists have written *R* packages for other scientists' use (FDA or non-FDA)
- FDA does not endorse or require the use of any specific software for statistical analysis

Regulations and Guidelines for Regulatory Use

- Code of Federal Regulations Title 21: FDA
  - Part 11: Electronic records and electronic signatures
- General Principles of Software Validation: Final Guidance for Industry and FDA Staff (2002)
- Guidance Part 11: Electronic Records (Final) (2003)
- Guidance for Industry: Computerized Systems Used in Clinical Investigations (2007)
- ICH E6: Good Clinical Practice Consolidated Guideline
- ICH E9: Statistical Principles for Clinical Trials

Code of Federal Regulations

21 CFR 11 deals with electronic records and signatures that are "trustworthy, reliable and generally equivalent to paper records."

- Qualification, verification and validation
- *R* is only part of a larger system that is verified
- *R* Foundation compliance document (2008): http://www.r-project.org/doc/R-FDA.pdf

This poster reflects the views of the author and should not be construed to represent FDA's views or policies.

Regulations and Guidances: What Applies to *R* at the FDA

Regulations state the specifics of what is legally required and enforceable. Guidances describe the agency's current position and are recommendations. 21 CFR Part 11 regulates electronic records and their storage, not the software that is used to generate reports. There are no regulations that restrict the use of open source software (including *R*) at the FDA. However, there are concerns for the FDA beyond CFR Title 21.

Current FDA *R* Setup

Only base and recommended packages are fully supported by the R core. Code or functions outside of these packages are used at the reviewer's own risk.

- Base: *datasets*, *graphics*, *grDevices*, *grid*, *methods*, *splines*, *stats*, *stats4*, *tcltk*, *tools*, *utils*
- All results from data should be reproducible independent of software used.
- FDA is not responsible if results have discrepancies.

Sponsors may use R in their submissions. Data must be submitted in *xpt* format. Reviewers would like to know:

- which *R* functions were used
- how they were accessed
- that the results are accurate, reliable, and consistent

Regulatory Agency Needs

Freeware is good in theory for a government agency, but comes with several issues:

- No tech/customer support
- How should the FDA provide support to *R* users?

The FDA has used *SAS* for 30+ years. FDA groups are using *SAS* and *R* concurrently. Legacy code is generally dealt with by calling *SAS* from *R*.

- Is this an effective or efficient way to handle old *SAS* code?
- *R* users must keep up with *SAS* code changes, and vice versa.

The FDA does not endorse any statistical software. Can we be innovative with *R* without appearing as though we "endorse" *R*?

Further Regulatory Issues That Impact *R* Usage at the FDA

As a government agency, the FDA must comply with additional regulations that do not impact *R*. Section 508 of the Rehabilitation Act of 1973 (29 U.S.C. 794d) established guidelines for ensuring that the federal government's electronic information is accessible to people with disabilities, including blindness, deafness, colorblindness, epilepsy, and reduced motor function. Section 508 includes guidelines for luminosity or brightness, use of color, font size, text alternatives for figures or plots, and text-only alternatives. All reports released online by the FDA should be 508 compliant, including reports from the Office of Biostatistics.

Example from an FDA report: Left: a plot made with the default rainbow color palette. This figure is not 508 compliant because some information is lost to colorblind people. Right: the red-green colorblind simulation. It is not possible to determine which lines correspond to individual subjects when the color palette is reduced.

Would the FDA need to modify or write its own version of popular *R* packages to meet additional guidelines? Section 508 compliance is not mandated for *R*, and the FDA cannot ask the *R* community to automatically make packages 508 compliant.

My Personal Experience as an *R* User at the FDA

The QT interval is a biomarker for cardiac repolarization. The QT Interdisciplinary Review Team (IRT) reviews Thorough QT (TQT) studies to investigate QT interval lengthening. The QT IRT consists of statisticians, clinical pharmacologists, and clinicians.

I am the only *R* user in the stats group (4 statisticians). I use:

- *SAS* for reviews (for consistency with other QT IRT stats reviewers)
- *R* for research
- *R* for any graphics outside reviews

The QT IRT clinical pharmacologists use *R* for their reviews. The *R* package *QT* was written specifically for the QT IRT.

- *QT* calls the stats group's *SAS* macros for areas in the clin pharm portion of the review
- other areas of the clin pharm review are done in *R*
- original author of *QT* no longer at the FDA

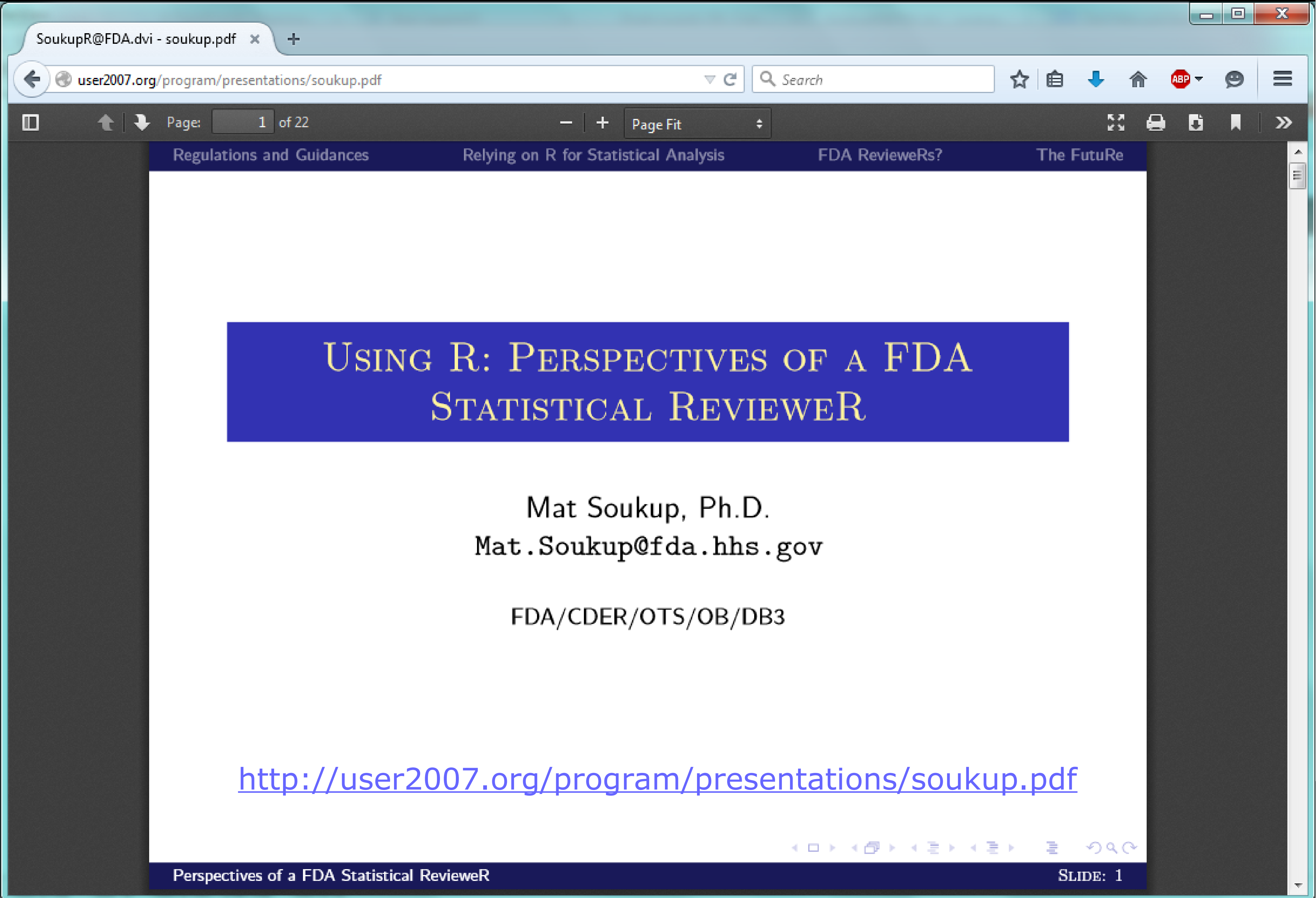
My future plans: to write an *R* package for QT IRT use that covers all parts of a TQT review. Timeline: ?

In my experience, *R* use at the FDA is completely acceptable and has not caused any problems.

References

[1] Title 21 CFR Part 11 Section 11.1 (a)
[2] General Principles of Software Validation; Final Guidance for Industry and FDA Staff
[3] Guidance Part 11: Electronic Records (Final)
[4] Guidance for Industry: Computerized Systems Used in Clinical Investigations
[5] ICH E6(R1): Good Clinical Practices
[6] ICH E9: Statistical Principles for Clinical Trials
[7] *R*: Regulatory Compliance and Validation Issues: A Guidance Document for the Use of *R* in Regulated Clinical Trial Environments:
[8] Section 508 of the Rehabilitation Act of 1973 (29 U.S.C. 794d):

some important information



Regulations and Guidances | Relying on R for Statistical Analysis | FDA RevieweRs? | The FutuRe

USING R: PERSPECTIVES OF A FDA STATISTICAL REVIEWER

Mat Soukup, Ph.D.
Mat.Soukup@fda.hhs.gov

FDA/CDER/OTS/OB/DB3

http://user2007.org/program/presentations/soukup.pdf

Perspectives of a FDA Statistical RevieweR

SLIDE: 1

FurtheR impoRtant issues

furtheR impoRtant issues

I Handling metadata

II Handling ODM and CDA metadata

III Issue with multilingual data

IV Differences between SAS and R

V Implementation of useful SAS functions

Handling metadata

Briefly, *metadata is data that describes other data*.

Metadata can be used for:

- Commenting and labeling (titling) objects (datasets, functions, graphs, tables)
- Adding instructions of how the content of a variable should be rendered, e.g.:
    - number of decimal digits, format of dates and times, length of texts
    - format of outputted text: font name, color, size, style
    - dictionary of which values should appear instead of raw data (translation)
- Searching purposes (keywords, tags)

Note: metadata doesn't change the underlying, raw value, it is just additional information.

Handling metadata

SAS uses metadata extensively to control how data is displayed. There are dozens of formats and “informats” specific to literals, numbers, currencies, dates and times.

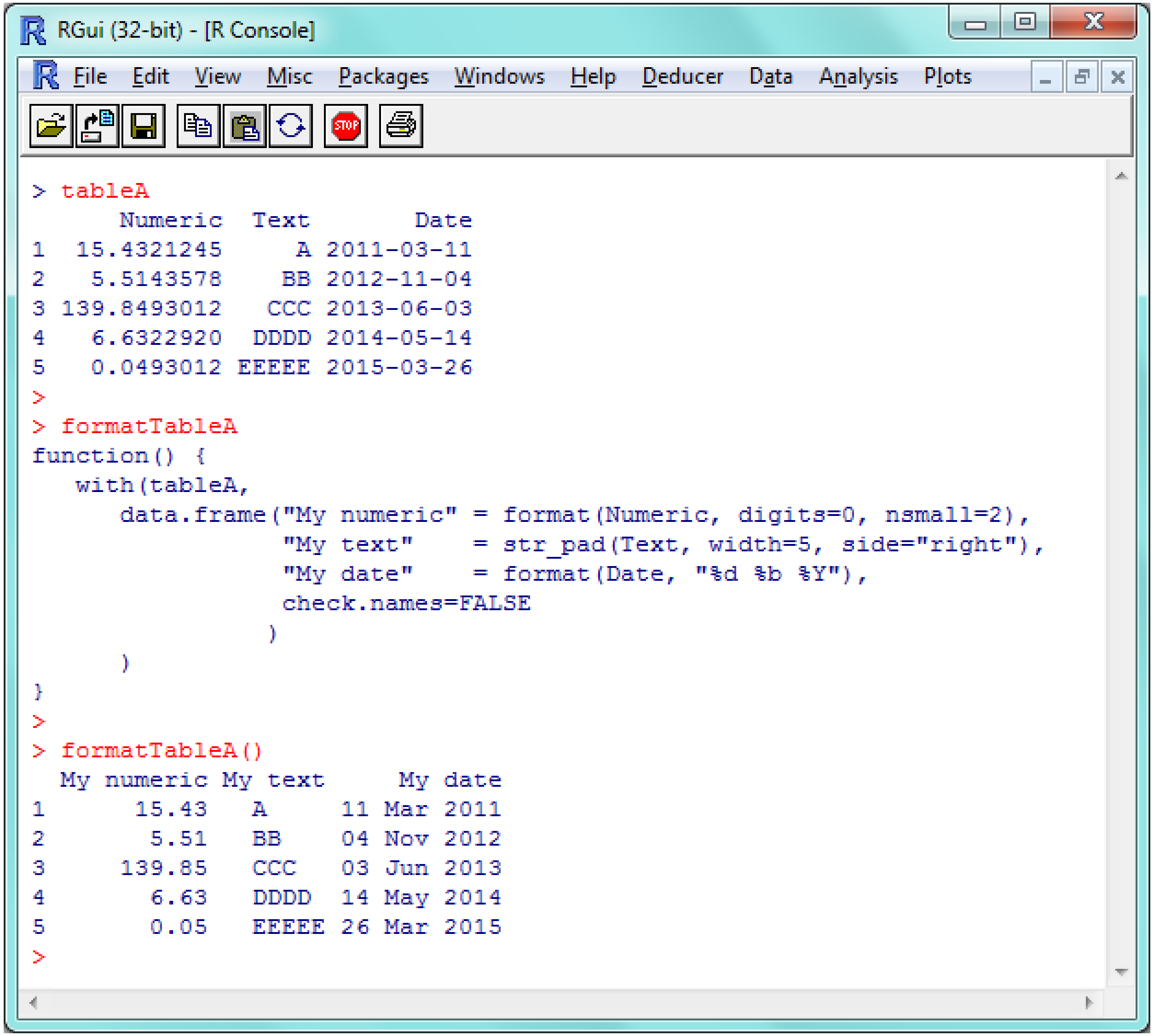
On the contrary, R doesn't use “formats”. There is no something like “*numeric(10,2)*” or “*char(20)*”. Without additional efforts data is displayed “as is”.

There are only some options to control globally the number of decimal digits, formatting dates or setting timezone.

```
> options("digits")$digit
[1] 7
> (x <- 12.3456789)
[1] 12.34568
> options(digits = 4)
> x
[1] 12.35
> x == 12.3456789
[1] TRUE
```

Handling metadata

To control formats on the column or cell level one must write additional formatting code every time it is required (usually a function is created for this purpose).



```
> tableA
      Numeric  Text       Date
1  15.4321245     A 2011-03-11
2   5.5143578    BB 2012-11-04
3 139.8493012   CCC 2013-06-03
4   6.6322920  DDDD 2014-05-14
5   0.0493012 EEEEE 2015-03-26
>
> formatTableA
function() {
   with(tableA,
      data.frame("My numeric" = format(Numeric, digits=0, nsmall=2),
                 "My text"    = str_pad(Text, width=5, side="right"),
                 "My date"    = format(Date, "%d %b %Y"),
                 check.names=FALSE
                )
      )
}
>
> formatTableA()
  My numeric My text     My date
1      15.43   A     11 Mar 2011
2       5.51   BB    04 Nov 2012
3     139.85   CCC   03 Jun 2013
4       6.63   DDDD  14 May 2014
5       0.05   EEEEE 26 Mar 2015
>
```

Handling metadata

This separates metadata from data and is an anti-pattern leading to:

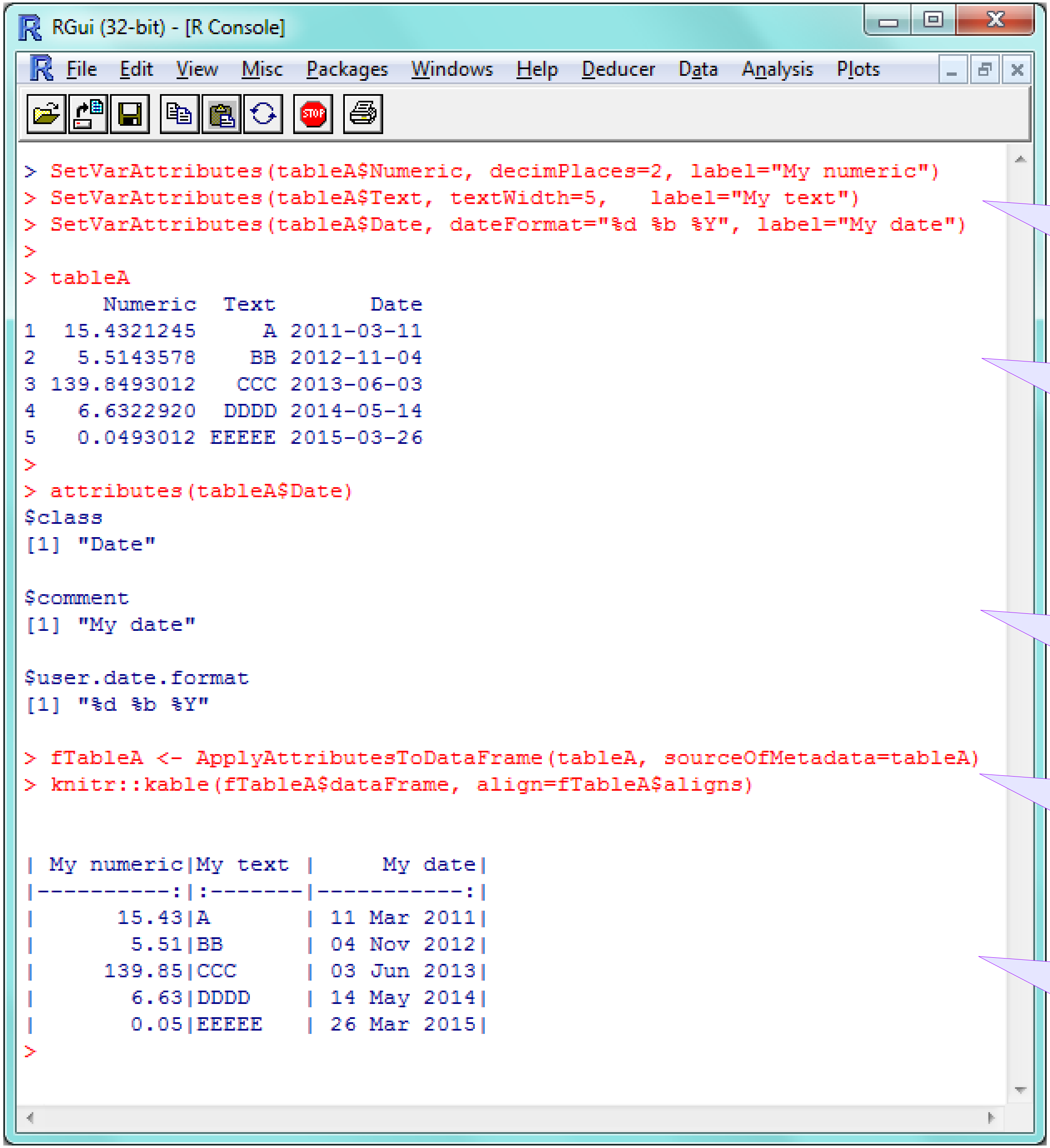
- Creation of dedicated formatting code specific to a data object. Amount of the code grows quickly as new variables are created *(formatTableA, formatListB, etc.)*
- Errors, when someone mistakes one metadata for other. This happens easily for large sets of variables.

The **golden rule** is that the ***data should be "self-describing"***.

This indicates the need to introduce a separate, independent, intermediate metadata layer between the data and the output rendering code, which consists of:

- Set of attributes describing data objects
- Functions assigning these attributes to data objects
- Functions retrieving values of the attributes and applying them to the raw data

Metadata in use



```
> SetVarAttributes(tableA$Numeric, decimPlaces=2, label="My numeric")
> SetVarAttributes(tableA$Text, textWidth=5,   label="My text")
> SetVarAttributes(tableA$Date, dateFormat="%d %b %Y", label="My date")
>
> tableA
      Numeric  Text        Date
1  15.4321245     A 2011-03-11
2   5.5143578    BB 2012-11-04
3 139.8493012   CCC 2013-06-03
4   6.6322920  DDDD 2014-05-14
5   0.0493012 EEEEE 2015-03-26
>
> attributes(tableA$Date)
$class
[1] "Date"

$comment
[1] "My date"

$user.date.format
[1] "%d %b %Y"

> fTableA <- ApplyAttributesToDataFrame(tableA, sourceOfMetadata=tableA)
> knitr::kable(fTableA$dataFrame, align=fTableA$aligns)


| My numeric|My text |     My date|
|----------:|:-------|-----------:|
|      15.43|A       | 11 Mar 2011|
|       5.51|BB      | 04 Nov 2012|
|     139.85|CCC     | 03 Jun 2013|
|       6.63|DDDD    | 14 May 2014|
|       0.05|EEEEE   | 26 Mar 2015|
>
```

Assigning attributes (metadata) to the original table

Original data stays unchanged

Metadata bound to a column of the original table

Applying metadata to the table

Displaying formatted result

Handling metadata

Fortunately, it is possible to create one's own metadata layer by using R attributes.

Author of this presentation wrote for his own use a set of functions for:

- Managing and using dictionaries (*code lists*)
- Binding metadata to objects
- Controlling the way data is displayed (applying metadata to datasets)
- Displaying the data with metadata overlapped
- Copying metadata between objects (when attributes were wiped out)
- Describing datasets in detail
- Reading labels of variables from sasXmlMap file
- Creating SAS Transport Files (XPT) using the attached metadata

They are supposed to be released as an R package in future.

Own metadata layer - example

Definition of an exemplary dataset:

```
> x <- c(5L,NA,6L,7L,1L)

> data <- data.frame(
     unlabeled = c(NA, "A", "B", "C", "D"),
     vlt = c("This is a long string", "Another long string", "Much longer string than previous",
                                                                              "A short string", NA),
     dv = c(NA, rnorm(4)),
     pct = x/sum(na.omit(x)),
     int = x,
     dt  = as.POSIXct(c("2011-02-01 08:00", "2010-04-21 12:30", NA, "1999-11-02 14:15", "2000-10-15 07:30")),
     dat = as.POSIXct(c("2011-02-01 08:00", "2010-04-21 12:30", NA, "1999-11-02 14:15", "2000-10-15 07:30")),
     tim = as.POSIXct(c("2011-02-01 08:00", "2010-04-21 12:30", NA, "1999-11-02 14:15", "2000-10-15 07:30")),
     trt = c(1,NA,2,4,3))
```

```
> # raw, unformatted data
> data
  unlabeled                       vlt         dv        pct int                  dt                 dat                 tim trt
1      <NA>     This is a long string         NA 0.26315789   5 2011-02-01 08:00:00 2011-02-01 08:00:00 2011-02-01 08:00:00   1
2         A       Another long string -1.4286644         NA  NA 2010-04-21 12:30:00 2010-04-21 12:30:00 2010-04-21 12:30:00  NA
3         B Much longer than previous  1.5035051 0.31578947   6                <NA>                <NA>                <NA>   2
4         C            A short string -1.3464709 0.36842105   7 1999-11-02 14:15:00 1999-11-02 14:15:00 1999-11-02 14:15:00   4
5         D                      <NA> -0.7571589 0.05263158   1 2000-10-15 07:30:00 2000-10-15 07:30:00 2000-10-15 07:30:00   3
```

Metadata - example

Definition of exemplary dictionaries:

```
# Look how a definition in SAS format can be used directly
> AddUserDictFromDef(definitions = 'value MyDict
                   1 = "Treatment A"
                   2 = "Treatment B"
                   3 = "Placebo";', SASFormat=TRUE)
```

```
> AddUserDict("sex",  codes  = c("F",      "M"),
                      labels = c("Female", "Male"))
```

```
> UCODELST
  codelst clcode     cllabel
1  MyDict      1 Treatment A
2  MyDict      2 Treatment B
3  MyDict      3     Placebo
4     sex      F      Female
5     sex      M        Male
```

Metadata - example

Add some metadata:

```
> SetVarAttributes(data$vlt,  label = "Very long text", textWidth = 15,  sasFormat = "$15.")
> SetVarAttributes(data$dv,   label = "Double value",   decimPlaces = 2, sasFormat = "8.2")
> SetVarAttributes(data$pct,  label = "Percentage",     decimPlaces = 1, asPercent = TRUE)
> SetSVarAttributes("data$int", label = "Integer")      # variable name passed as a literal
> SetVarAttributes(data$dat,  label = "Date",           dateFormat = "%d%b%Y", sasFormat = "yymmdd10.")
> SetVarAttributes(data$dt,   label = "Date & Time (ISO 8601)", dateFormat = "%Y-%m-%d %H:%M:%S",
                                                                                      sasFormat = "e8601dt19.")
> SetVarAttributes(data$tim,  label = "Time",           dateFormat = "%H:%M")
> SetVarAttributes(data$trt,  label = "Translated",     translDict = "MyDict")
> SetVarAttributes(data,      label = "My data set")
```

```
> attributes(data$dt)
$class
[1] "POSIXct" "POSIXt"

$tzone
[1] ""

$comment
[1] "Date & Time (ISO 8601)"

$user.date.format
[1] "%Y-%m-%d %H:%M:%S"

$user.sas.format
[1] "e8601dt19."
```

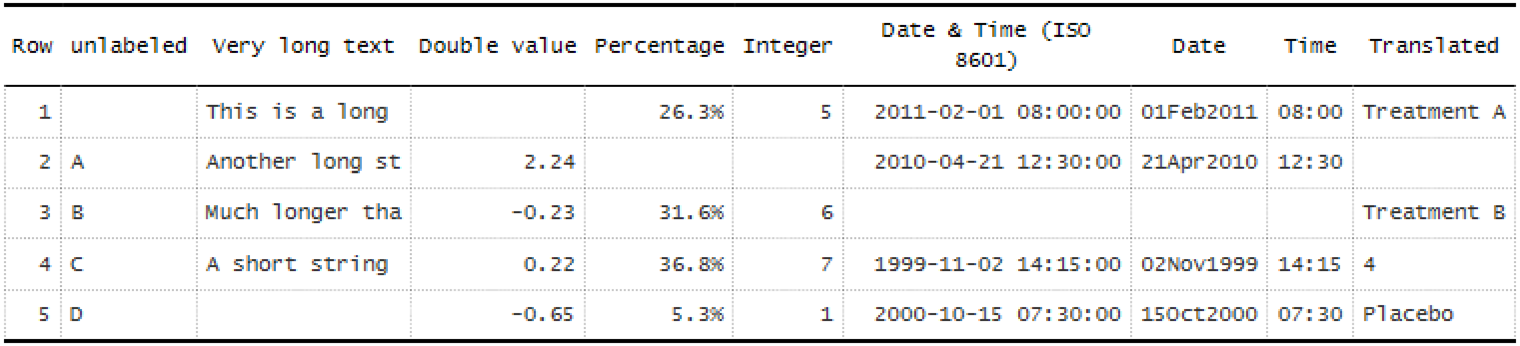
Attributes bound to
a column of a data.frame

Metadata - example

Display the data with metadata applied:

```
# Raw data
> data
  unlabeled                       vlt         dv        pct int                  dt                 dat                 tim trt
1      <NA>     This is a long string         NA 0.26315789   5 2011-02-01 08:00:00 2011-02-01 08:00:00 2011-02-01 08:00:00   1
2         A       Another long string -1.4286644         NA  NA 2010-04-21 12:30:00 2010-04-21 12:30:00 2010-04-21 12:30:00  NA
3         B Much longer than previous  1.5035051 0.31578947   6                <NA>                <NA>                <NA>   2
4         C            A short string -1.3464709 0.36842105   7 1999-11-02 14:15:00 1999-11-02 14:15:00 1999-11-02 14:15:00   4
5         D                      <NA> -0.7571589 0.05263158   1 2000-10-15 07:30:00 2000-10-15 07:30:00 2000-10-15 07:30:00   3
```

```
# Data with metadata overlapped  (output is in RMarkdown format)
> PrintDataFrame(data, sourceOfMetadata = data)

| Row|unlabeled |Very long text   | Double value| Percentage| Integer| Date & Time (ISO 8601)|      Date|  Time|Translated   |
|---:|:---------|:----------------|------------:|----------:|-------:|----------------------:|---------:|-----:|:------------|
|   1|          |This is a long   |             |      26.3%|       5|    2011-02-01 08:00:00| 01Feb2011| 08:00|Treatment A  |
|   2|A         |Another long st  |        -1.43|           |        |    2010-04-21 12:30:00| 21Apr2010| 12:30|             |
|   3|B         |Much longer tha  |         1.50|      31.6%|       6|                       |          |      |Treatment B  |
|   4|C         |A short string   |        -1.35|      36.8%|       7|    1999-11-02 14:15:00| 02Nov1999| 14:15|4            |
|   5|D         |                 |        -0.76|       5.3%|       1|    2000-10-15 07:30:00| 15Oct2000| 07:30|Placebo      |
```



Row	unlabeled	Very long text	Double value	Percentage	Integer	Date & Time (ISO 8601)	Date	Time	Translated
1		This is a long		26.3%	5	2011-02-01 08:00:00	01Feb2011	08:00	Treatment A
2	A	Another long st	2.24			2010-04-21 12:30:00	21Apr2010	12:30	
3	B	Much longer tha	-0.23	31.6%	6				Treatment B
4	C	A short string	0.22	36.8%	7	1999-11-02 14:15:00	02Nov1999	14:15	4
5	D		-0.65	5.3%	1	2000-10-15 07:30:00	15Oct2000	07:30	Placebo

Metadata - example

Describe dataset with metadata attached:

```
> DescribeTable(data)  # simple, default form
                                   My data set
____________________________________________________________________________
Rows: 5

dat*       DAT  maxL: 19 W:    D:  T:  N:  4 NA:  1  U:  5 | 01Feb2011, 21Apr2010, NA, 02Nov1999, 15Oct2000
dt*        DAT  maxL: 19 W:    D:  T:  N:  4 NA:  1  U:  5 | 2011-02-01 08:00:00, 2010-04-21 12:30:00, NA, 1999-11-02 14:15:00, 2000-10-15 07:30:00
dv*        dbl  maxL: 18 W:    D:2 T:  N:  4 NA:  1  U:  5 | Min -1.43, Q1 -1.39, Me -1.05, Q3 0.37, Max 1.5 | NA, -1.42866436243726, 1.50350508808 ...
int*       int  maxL:  2 W:    D:0 T:  N:  4 NA:  1  U:  5 | Min 1, Q1 3, Me 5.5, Q3 6.5, Max 7 | 5, NA, 6, 7, 1
pct*       dbl  maxL: 18 W:    D:1 T:  N:  4 NA:  1  U:  5 | Min 0.05, Q1 0.16, Me 0.29, Q3 0.34, Max 0.37 | 0.263157894736842, NA, 0.3157894736842 ...
tim*       DAT  maxL: 19 W:    D:  T:  N:  4 NA:  1  U:  5 | 08:00, 12:30, NA, 14:15, 07:30
trt*       dbl  maxL:  2 W:    D:2 T:Y N:  4 NA:  1  U:  5 | Min 1, Q1 1.5, Me 2.5, Q3 3.5, Max 4 | (MyDict) Treatment A, NA, Treatment B, 4, Placebo
unlabeled  txt  maxL:  2 W:    D:  T:  N:  4 NA:  1  U:  5 | NA, A, B, C, D
vlt*       txt  maxL: 25 W: 15 D:  T:  N:  4 NA:  1  U:  5 | This is a long string, Another long string, Much longer than previous, A short string, NA
```

```
> DescribeTable(data, displayLabels=TRUE, displaySASFormats = TRUE)    # more detailed
                                      My data set
____________________________________________________________________________
Rows: 5

dat       [Date]                  DAT  maxL: 19 W:    D:  SASf: yymmdd10. T:  N:  4 NA:  1  U:  5 | 01Feb2011, 21Apr2010, NA, 02Nov1999, 15Oct2000
dt        [Date & Time (ISO 8601)]DAT  maxL: 19 W:    D:  SASf:e8601dt19. T:  N:  4 NA:  1  U:  5 | 2011-02-01 08:00:00, 2010-04-21 12:30:00, NA, 1 ...
dv        [Double value]          dbl  maxL: 18 W:    D:2 SASf:       8.2 T:  N:  4 NA:  1  U:  5 | Min -1.43, Q1 -1.39, Me -1.05, Q3 0.37, Max 1.5 ...
int       [Integer]               int  maxL:  2 W:    D:0 SASf:           T:  N:  4 NA:  1  U:  5 | Min 1, Q1 3, Me 5.5, Q3 6.5, Max 7 | 5, NA, 6,  ...
pct       [Percentage]            dbl  maxL: 18 W:    D:1 SASf:           T:  N:  4 NA:  1  U:  5 | Min 0.05, Q1 0.16, Me 0.29, Q3 0.34, Max 0.37 | ...
tim       [Time]                  DAT  maxL: 19 W:    D:  SASf:           T:  N:  4 NA:  1  U:  5 | 08:00, 12:30, NA, 14:15, 07:30
trt       [Translated]            dbl  maxL:  2 W:    D:2 SASf:           T:Y N:  4 NA:  1  U:  5 | Min 1, Q1 1.5, Me 2.5, Q3 3.5, Max 4 | (MyDict) ...
unlabeled                         txt  maxL:  2 W:    D:  SASf:           T:  N:  4 NA:  1  U:  5 | NA, A, B, C, D
vlt       [Very long text]        txt  maxL: 25 W: 15 D:  SASf:      $15. T:  N:  4 NA:  1  U:  5 | This is a long string, Another long string, Muc ...
```

Metadata - example

Transfer attributes between two datasets.

```
> newData <- sqldf("SELECT * FROM data")    # sqldf wipes out attributes

> DescribeTable(newData, displayLabels=TRUE, displaySASFormats = TRUE)
Rows: 5

dat        DAT  maxL: 19 W:    D:  SASf: T:  N:  4 NA:  1  U:  5 | 2011-02-01 08:00:00, 2010-04-21 12:30:00, NA, 1999-11-02 14:15:00, 2000-10-15 07 ...
dt         DAT  maxL: 19 W:    D:  SASf: T:  N:  4 NA:  1  U:  5 | 2011-02-01 08:00:00, 2010-04-21 12:30:00, NA, 1999-11-02 14:15:00, 2000-10-15 07 ...
dv         dbl  maxL: 18 W:    D:2 SASf: T:  N:  4 NA:  1  U:  5 | Min -0.65, Q1 -0.44, Me 0, Q3 1.23, Max 2.24 | NA, 2.24243108561811, -0.22569800 ...
int        int  maxL:  2 W:    D:0 SASf: T:  N:  4 NA:  1  U:  5 | Min 1, Q1 3, Me 5.5, Q3 6.5, Max 7 | 5, NA, 6, 7, 1
pct        dbl  maxL: 18 W:    D:2 SASf: T:  N:  4 NA:  1  U:  5 | Min 0.05, Q1 0.16, Me 0.29, Q3 0.34, Max 0.37 | 0.263157894736842, NA, 0.3157894 ...
tim        DAT  maxL: 19 W:    D:  SASf: T:  N:  4 NA:  1  U:  5 | 2011-02-01 08:00:00, 2010-04-21 12:30:00, NA, 1999-11-02 14:15:00, 2000-10-15 07 ...
trt        dbl  maxL:  2 W:    D:2 SASf: T:  N:  4 NA:  1  U:  5 | Min 1, Q1 1.5, Me 2.5, Q3 3.5, Max 4 | 1, NA, 2, 4, 3
unlabeled  txt  maxL:  2 W:    D:  SASf: T:  N:  4 NA:  1  U:  5 | NA, A, B, C, D
vlt        txt  maxL: 25 W:    D:  SASf: T:  N:  4 NA:  1  U:  5 | This is a long string, Another long string, Much longer than previous, A short s ...
```

```
> CopyUserAttributes(srcDS = data, destDS = newData, env=environment())
Copied table attribute: [newData]@comment = My data set
Copied column attribute: [vlt]@comment = Very long text
Copied column attribute: [vlt]@user.str.len = 15
Copied column attribute: [vlt]@user.sas.format = $15.
...
Copied column attribute: [trt]@comment = Translated
Copied column attribute: [trt]@user.transl.dict = MyDict
```

```
> DescribeTable(newData, displayLabels=TRUE, displaySASFormats = TRUE)
                                                                      My data set
______________________________________________________________________________________________________________________________________________________
Rows: 5

dat        [Date]                  DAT  maxL: 19 W:    D:  SASf: yymmdd10. T:  N:  4 NA:  1  U:  5 | 01Feb2011, 21Apr2010, NA, 02Nov1999, 15Oct2000
dt         [Date & Time (ISO 8601)]DAT  maxL: 19 W:    D:  SASf:e8601dt19. T:  N:  4 NA:  1  U:  5 | 2011-02-01 08:00:00, 2010-04-21 12:30:00, NA, 1 ...
dv         [Double value]          dbl  maxL: 18 W:    D:2 SASf:       8.2 T:  N:  4 NA:  1  U:  5 | Min -0.65, Q1 -0.44, Me 0, Q3 1.23, Max 2.24 |  ...
int        [Integer]               int  maxL:  2 W:    D:0 SASf:           T:  N:  4 NA:  1  U:  5 | Min 1, Q1 3, Me 5.5, Q3 6.5, Max 7 | 5, NA, 6,  ...
pct        [Percentage]            dbl  maxL: 18 W:    D:1 SASf:           T:  N:  4 NA:  1  U:  5 | Min 0.05, Q1 0.16, Me 0.29, Q3 0.34, Max 0.37 | ...
tim        [Time]                  DAT  maxL: 19 W:    D:  SASf:           T:  N:  4 NA:  1  U:  5 | 08:00, 12:30, NA, 14:15, 07:30
trt        [Translated]            dbl  maxL:  2 W:    D:2 SASf:           T:Y N:  4 NA:  1  U:  5 | Min 1, Q1 1.5, Me 2.5, Q3 3.5, Max 4 | (MyDict) ...
unlabeled                          txt  maxL:  2 W:    D:  SASf:           T:  N:  4 NA:  1  U:  5 | NA, A, B, C, D
vlt        [Very long text]        txt  maxL: 25 W: 15 D:  SASf:      $15. T:  N:  4 NA:  1  U:  5 | This is a long string, Another long string, Muc ...
```

Metadata - example

Transfer attributes when datasets are copied via <- or =

```
keep.attr <- function(x)
{
     a <- attributes(x)
     a[c('names','row.names','class','dim','dimnames')] <- NULL
     a
}

keep <- function(.Data, ..., .Attr=NULL)
{
     cl <- union('keep', class(.Data))
     do.call('structure', c(list(.Data, class=cl, ...), .Attr))
}

'[.keep' <- function(.Data, ...)
     keep(NextMethod(), .Attr=keep.attr(.Data))

'[<-.keep' <- function(.Data, ...)
     keep(NextMethod(), .Attr=keep.attr(.Data))
```

```
> newData <- data  # dataset "data" has attributes which will be copied

> DescribeTable(newData, displayLabels=TRUE, displaySASFormats = TRUE)
                                                                      My data set
_______________________________________________________________________________________________________________________________________________________
Rows: 5

dat       [Date]                  DAT  maxL: 19 W:    D:  SASf: yymmdd10. T:  N:  4 NA:  1  U:  5 | 01Feb2011, 21Apr2010, NA, 02Nov1999, 15Oct2000
dt        [Date & Time (ISO 8601)]DAT  maxL: 19 W:    D:  SASf:e8601dt19. T:  N:  4 NA:  1  U:  5 | 2011-02-01 08:00:00, 2010-04-21 12:30:00, NA, 1 ...
dv        [Double value]          dbl  maxL: 18 W:    D:2 SASf:       8.2 T:  N:  4 NA:  1  U:  5 | Min -0.65, Q1 -0.44, Me 0, Q3 1.23, Max 2.24 |  ...
int       [Integer]               int  maxL:  2 W:    D:0 SASf:           T:  N:  4 NA:  1  U:  5 | Min 1, Q1 3, Me 5.5, Q3 6.5, Max 7 | 5, NA, 6,  ...
pct       [Percentage]            dbl  maxL: 18 W:    D:1 SASf:           T:  N:  4 NA:  1  U:  5 | Min 0.05, Q1 0.16, Me 0.29, Q3 0.34, Max 0.37 | ...
tim       [Time]                  DAT  maxL: 19 W:    D:  SASf:           T:  N:  4 NA:  1  U:  5 | 08:00, 12:30, NA, 14:15, 07:30
trt       [Translated]            dbl  maxL:  2 W:    D:2 SASf:           T:Y N:  4 NA:  1  U:  5 | Min 1, Q1 1.5, Me 2.5, Q3 3.5, Max 4 | (MyDict) ...
unlabeled                         txt  maxL:  2 W:    D:  SASf:           T:  N:  4 NA:  1  U:  5 | NA, A, B, C, D
vlt       [Very long text]        txt  maxL: 25 W: 15 D:  SASf:      $15. T:  N:  4 NA:  1  U:  5 | This is a long string, Another long string, Muc ...
```

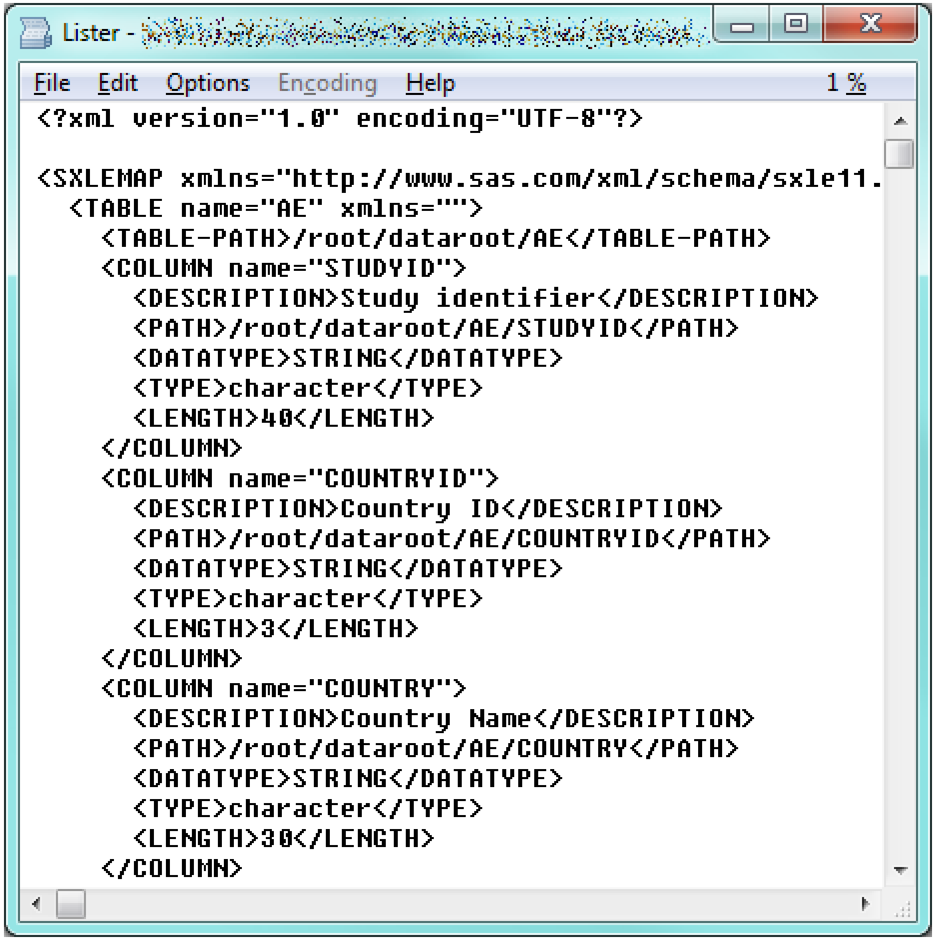
Metadata – reading *.sasXmlMap

With R one can easily parse additional XML files containing definition of metadata.

```r
# Quick and dirty reading labels from the *.sasXmlMap file. This can be easily extended.
ApplyLabelsFromSasXmlMap <- function(collectionOfTables, pathToSasXmlMap) {

     nameOfTablesCollection <- deparse(substitute(collectionOfTables))

     if(file.exists(pathToSasXmlMap)) {

          DOM <- xmlParse(pathToSasXmlMap)

          for(tab in names(collectionOfTables)) {

               for(col in toupper(colnames(collectionOfTables[[tab]]))) {

                    labelList <- xpathApply(DOM, paste0("//TABLE[@name = '", tab, "']/COLUMN[@name = '", col,
                                                                                 "']/DESCRIPTION"), xmlValue)

                    if(length(labelList) > 0) {

                         cat(paste("Table:", tab, " Column:", col, "=", labelList[[1]], "\n"))

                         SetSVarAttributes(varName = paste0(nameOfTablesCollection, "$", tab, "$",
                                                                       tolower(col)), label = labelList[[1]])
                    }
               }
          }
     }
}
```

Metadata – reading *.sasXmlMap

With R one can easily parse additional XML files containing definition of metadata.



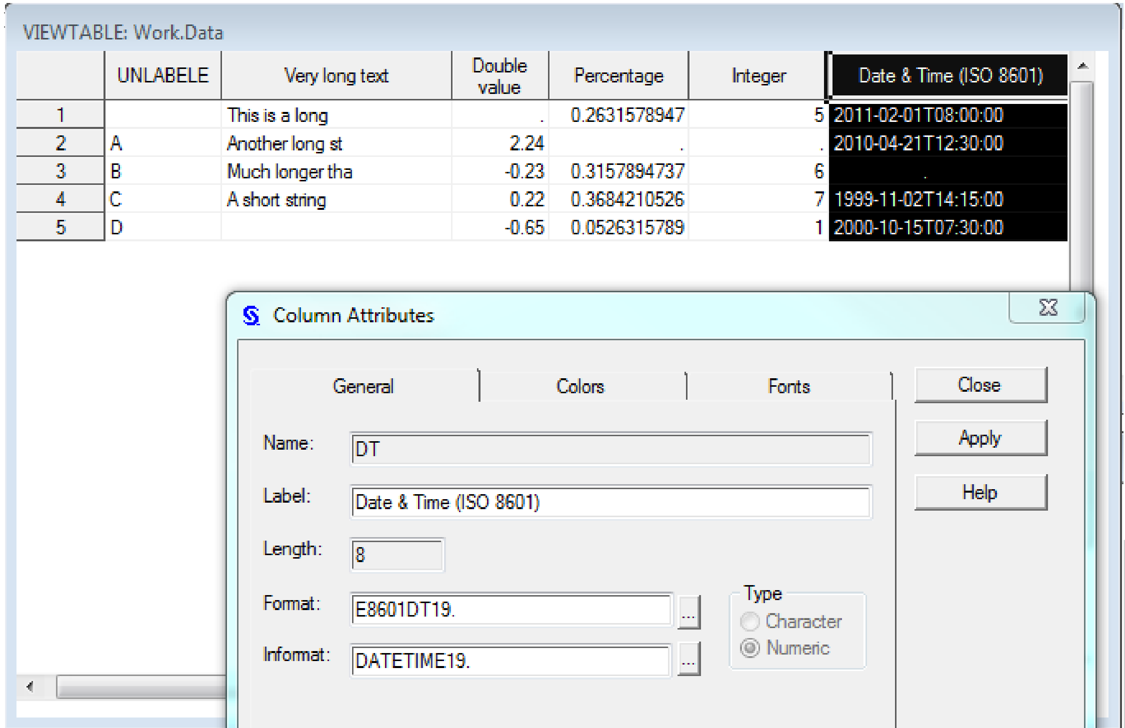
```
<?xml version="1.0" encoding="UTF-8"?>

<SXLEMAP xmlns="http://www.sas.com/xml/schema/sxle11.
  <TABLE name="AE" xmlns="">
    <TABLE-PATH>/root/dataroot/AE</TABLE-PATH>
    <COLUMN name="STUDYID">
      <DESCRIPTION>Study identifier</DESCRIPTION>
      <PATH>/root/dataroot/AE/STUDYID</PATH>
      <DATATYPE>STRING</DATATYPE>
      <TYPE>character</TYPE>
      <LENGTH>40</LENGTH>
    </COLUMN>
    <COLUMN name="COUNTRYID">
      <DESCRIPTION>Country ID</DESCRIPTION>
      <PATH>/root/dataroot/AE/COUNTRYID</PATH>
      <DATATYPE>STRING</DATATYPE>
      <TYPE>character</TYPE>
      <LENGTH>3</LENGTH>
    </COLUMN>
    <COLUMN name="COUNTRY">
      <DESCRIPTION>Country Name</DESCRIPTION>
      <PATH>/root/dataroot/AE/COUNTRY</PATH>
      <DATATYPE>STRING</DATATYPE>
      <TYPE>character</TYPE>
      <LENGTH>30</LENGTH>
    </COLUMN>
```

```
> # labels may be overwritten later in code
> ApplyLabelsFromSasXmlMap(cdash, "test.sasXmlMap")

Table: AE  Column: STUDYID = Study identifier
Table: AE  Column: COUNTRYID = Country ID
Table: AE  Column: COUNTRY = Country Name
Table: AE  Column: SITEID = Study site identifier
Table: AE  Column: SITE = Site Name
Table: AE  Column: SUBJID = Study subject identifier
Table: AE  Column: RANDNO = Study subject
randomization number
Table: AE  Column: AESPID = Adverse event's number
Table: AE  Column: USERNAME = Login Name
Table: AE  Column: USER_FULL_NAME = User Name
Table: AE  Column: LAST_MODIFIED = Last Modifed
Table: AE  Column: RECORD_STATUS = Record Status
Table: AE  Column: AEYN = Adverse events
Table: AE  Column: AETERM = Diagnosis or signs

...
```

Metadata – creating SAS Transport File

Having metadata attached to a data.frame and using SASxport package SAS Transport files can be created easily.

```
> MakeSASTransportFile(data, "c:/tmp/test.xpt")
Processing: test.xpt

*** Setting metadata... ***
 Processing column: unlabeled of class: character
 Processing column: vlt of class: character | Specified width: 15 | Assigned SAS format: $15.
 Processing column: dv of class: numeric | Specified decimal places: 2 | Assigned SAS format: 8.2
 Processing column: pct of class: numeric | Specified decimal places: 1 | Assigned DEFAULT NUMERIC SAS format: Best12.
 Processing column: int of class: integer  | Assigned DEFAULT NUMERIC SAS format: Best12.
 Processing column: dt of class: POSIXct | Specified format: %Y-%m-%d %H:%M:%S | Assigned SAS format: e8601dt19.
 Processing column: dat of class: POSIXct | Specified format: %d%b%Y | Assigned SAS format: yymmdd10.
 Processing column: tim of class: POSIXct | Specified format: %H:%M
 Processing column: trt of class: numeric  | Assigned DE

*** Exporting the dataset into SAS transport file..
*** Done ***
```



VIEWTABLE: Work.Data

	UNLABELE	Very long text	Double value	Percentage	Integer	Date & Time (ISO 8601)
1		This is a long	.	0.2631578947	5	2011-02-01T08:00:00
2	A	Another long st	2.24	.	.	2010-04-21T12:30:00
3	B	Much longer tha	-0.23	0.3157894737	6	.
4	C	A short string	0.22	0.3684210526	7	1999-11-02T14:15:00
5	D		-0.65	0.0526315789	1	2000-10-15T07:30:00

Column Attributes

Name: DT

Label: Date & Time (ISO 8601)

Length: 8

Format: E8601DT19.

Informat: DATETIME19.

Type
○ Character
● Numeric

furtheR impoRtant issues

I Handling metadata

II Handling ODM and CDA metadata

III Issue with multilingual data

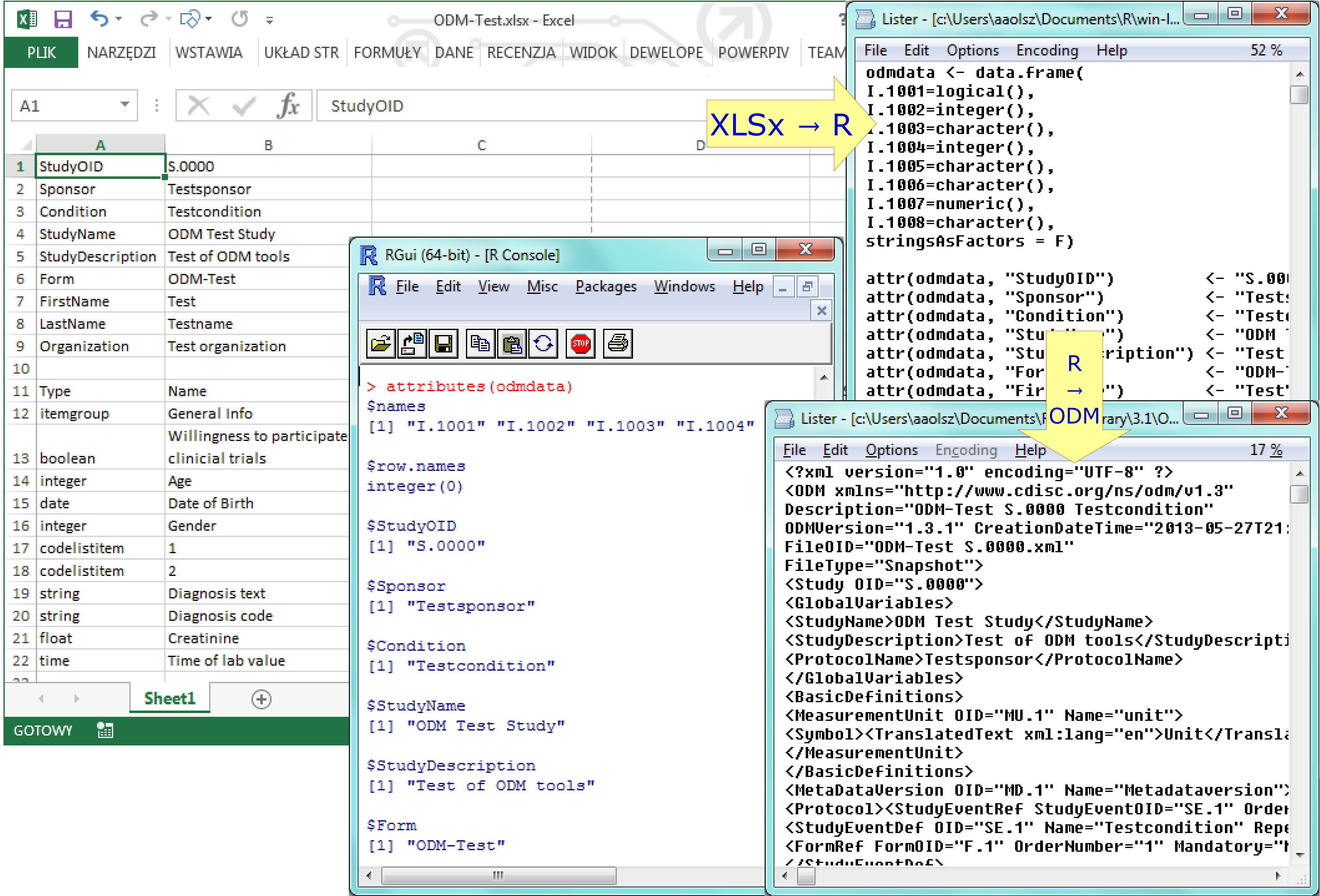
IV Differences between SAS and R

V Implementation of useful SAS functions

Handling ODM and CDA metadata

By using the ODMconverter package one can work with ODM and CDA formats in R. Both formats can be reciprocally translated into each other. There are also functions for creating R data.frames decorated with ODM metadata as well as creating ODM definitions based on metadata bound to an existing R data.frame.

Handling ODM and CDA metadata



	A	B
1	StudyOID	S.0000
2	Sponsor	Testsponsor
3	Condition	Testcondition
4	StudyName	ODM Test Study
5	StudyDescription	Test of ODM tools
6	Form	ODM-Test
7	FirstName	Test
8	LastName	Testname
9	Organization	Test organization
10		
11	Type	Name
12	itemgroup	General Info
13	boolean	Willingness to participate clinicial trials
14	integer	Age
15	date	Date of Birth
16	integer	Gender
17	codelistitem	1
18	codelistitem	2
19	string	Diagnosis text
20	string	Diagnosis code
21	float	Creatinine
22	time	Time of lab value

```
odmdata <- data.frame(
I.1001=logical(),
I.1002=integer(),
I.1003=character(),
I.1004=integer(),
I.1005=character(),
I.1006=character(),
I.1007=numeric(),
I.1008=character(),
stringsAsFactors = F)

attr(odmdata, "StudyOID")           <- "S.00
attr(odmdata, "Sponsor")            <- "Test
attr(odmdata, "Condition")          <- "Test
attr(odmdata, "Stu       ")         <- "ODM 
attr(odmdata, "Stu      ription")   <- "Test
attr(odmdata, "For       ")         <- "ODM-
attr(odmdata, "Fir       ")         <- "Test
```

```
> attributes(odmdata)
$names
[1] "I.1001" "I.1002" "I.1003" "I.1004"

$row.names
integer(0)

$StudyOID
[1] "S.0000"

$Sponsor
[1] "Testsponsor"

$Condition
[1] "Testcondition"

$StudyName
[1] "ODM Test Study"

$StudyDescription
[1] "Test of ODM tools"

$Form
[1] "ODM-Test"
```

```
<?xml version="1.0" encoding="UTF-8" ?>
<ODM xmlns="http://www.cdisc.org/ns/odm/v1.3"
Description="ODM-Test S.0000 Testcondition"
ODMVersion="1.3.1" CreationDateTime="2013-05-27T21:
FileOID="ODM-Test S.0000.xml"
FileType="Snapshot">
<Study OID="S.0000">
<GlobalVariables>
<StudyName>ODM Test Study</StudyName>
<StudyDescription>Test of ODM tools</StudyDescripti
<ProtocolName>Testsponsor</ProtocolName>
</GlobalVariables>
<BasicDefinitions>
<MeasurementUnit OID="MU.1" Name="unit">
<Symbol><TranslatedText xml:lang="en">Unit</Transla
</MeasurementUnit>
</BasicDefinitions>
<MetaDataVersion OID="MD.1" Name="Metadataversion">
<Protocol><StudyEventRef StudyEventOID="SE.1" Order
<StudyEventDef OID="SE.1" Name="Testcondition" Repe
<FormRef FormOID="F.1" OrderNumber="1" Mandatory="N
```

furtheR impoRtant issues

I Handling metadata

II Handling ODM and CDA metadata

III Issue with multilingual data

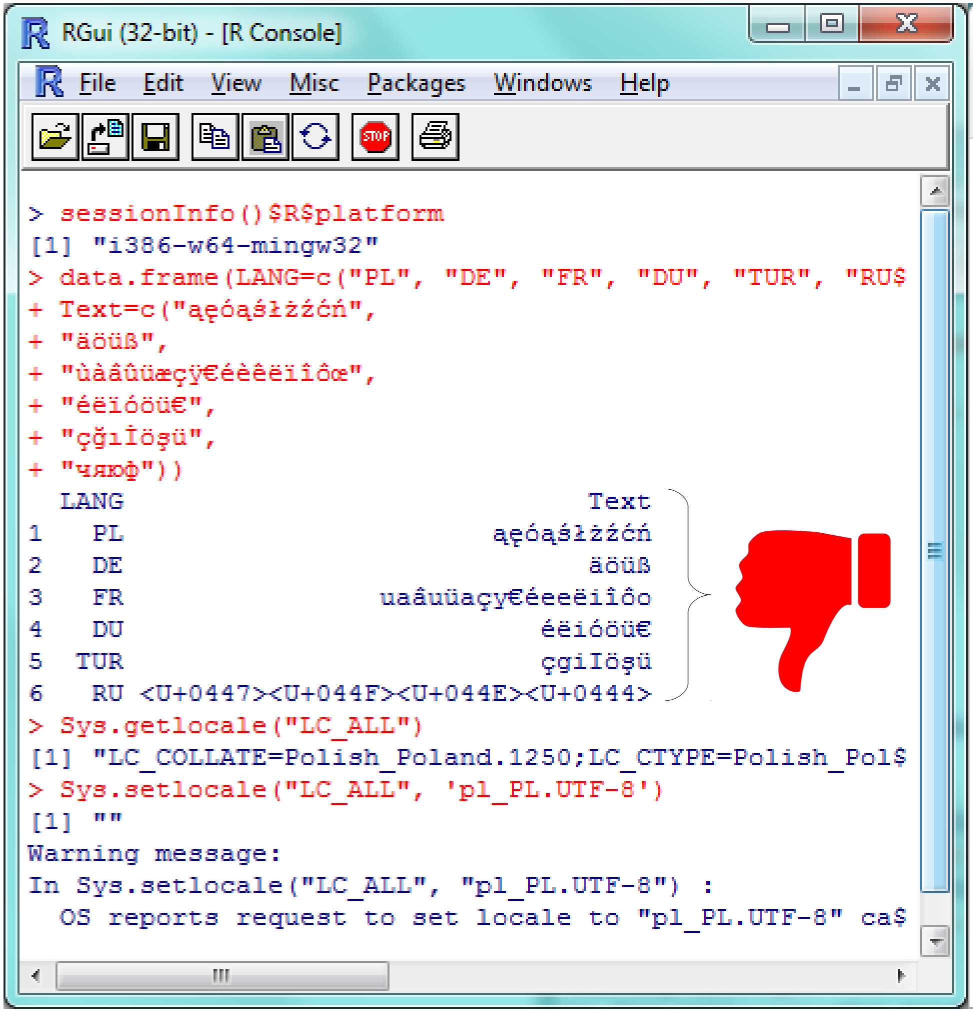
IV Differences between SAS and R

V Implementation of useful SAS functions

Issue with multilingual data

If multilingual data has to be processed, the best choice is to do it under Linux or MacOS. Both are unicode-enabled operating systems in contrast to Windows.

Windows



```
> sessionInfo()$R$platform
[1] "i386-w64-mingw32"
> data.frame(LANG=c("PL", "DE", "FR", "DU", "TUR", "RU$
+ Text=c("ąęóąśłżźćń",
+ "äöüß",
+ "ùàâûüæçÿ€éèêëïîôœ",
+ "éëïóöü€",
+ "çğıİöşü",
+ "чяюф"))
  LANG                                Text
1   PL                          ąęóąśłżźćń
2   DE                                äöüß
3   FR                   uaâuüaçy€éeeëiîôo
4   DU                             éëióöü€
5  TUR                             çgiIöşü
6   RU <U+0447><U+044F><U+044E><U+0444>
> Sys.getlocale("LC_ALL")
[1] "LC_COLLATE=Polish_Poland.1250;LC_CTYPE=Polish_Pol$
> Sys.setlocale("LC_ALL", 'pl_PL.UTF-8')
[1] ""
Warning message:
In Sys.setlocale("LC_ALL", "pl_PL.UTF-8") :
  OS reports request to set locale to "pl_PL.UTF-8" ca$
```

Linux

```
> sessionInfo()$R$platform
[1] "x86_64-pc-linux-gnu"
> data.frame(LANG=c("PL", "DE", "FR", "DU", "TUR", "RU"),
+ Text=c("ąęóąśłżźćń",
+ "äöüß",
+ "ùàâûüæçÿ€éèêëïîôœ",
+ "éëïóöü€",
+ "çğıİöşü",
+ "зчяющылф"))
  LANG              Text
1   PL        ąęóąśłżźćń
2   DE              äöüß
3   FR ùàâûüæçÿ€éèêëïîôœ
4   DU           éëïóöü€
5  TUR           çğıİöşü
6   RU          зчяющылф
> Sys.getlocale("LC_ALL")
[1] "LC_CTYPE=pl_PL.utf8;LC_NUMERIC=C;LC_TIME=pl_PL.utf8;LC_CO
LLATE=pl_PL.utf8;LC_MONETARY=pl_PL.utf8;LC_MESSAGES=pl_PL.utf8
;LC_PAPER=pl_PL.utf8;LC_NAME=C;LC_ADDRESS=C;LC_TELEPHONE=C;LC_
MEASUREMENT=pl_PL.utf8;LC_IDENTIFICATION=C"
> 
```

furtheR impoRtant issues

I	Handling metadata
II	Handling ODM and CDA metadata
III	Issue with multilingual data
IV	**Differences between SAS and R**
V	Implementation of useful SAS functions

Differences between SAS and R

It is worth noting that:

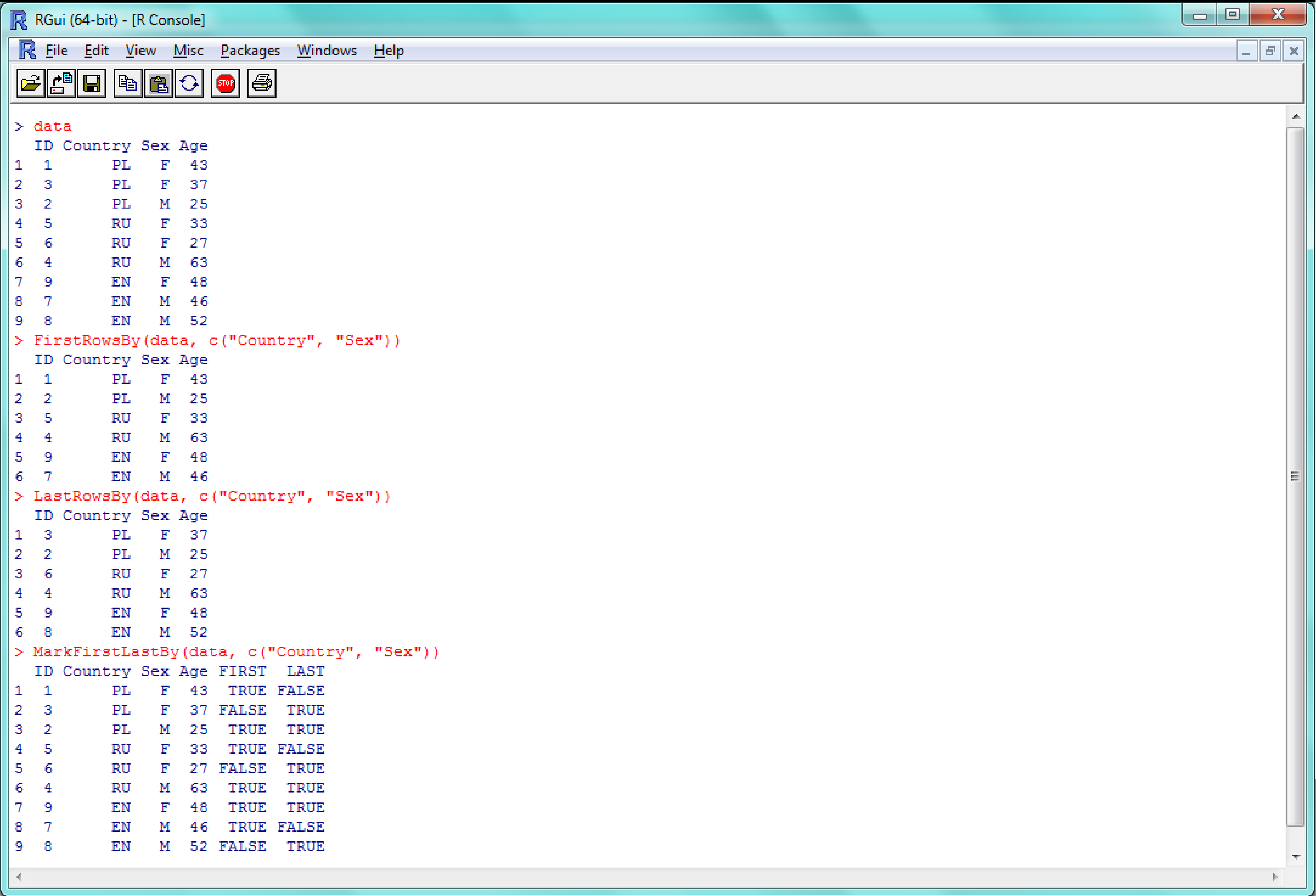
- R and SAS use different types of sum of square, type I and III respectively
  - R can be set to calculate SS of the same type (in fact – of any type)
- R uses different default contrast coding than SAS
  - R can be set to use the same contrasts (contr.SAS)
- Dates of origin of POSIX date/time differ across both packages:
  - in R: 1970-01-01 00:00:00 UTC
  - in SAS: 1960-01-01 00:00:00 GMT
- R uses different algorithm for calculating quantiles. Luckily, SAS-compliant algorithm is implemented (type 3)

furtheR impoRtant issues

R implementation of useful SAS functions #1

```
FirstRowsBy <- function(dataFrame, byColumns) {
     DT <- data.table(dataFrame, key=byColumns)
     as.data.frame(DT[unique(DT[, key(DT), with = FALSE]), mult = 'first'])
}

LastRowsBy <- function(dataFrame, byColumns) {
     DT <- data.table(dataFrame, key=byColumns)
     as.data.frame(DT[unique(DT[, key(DT), with = FALSE]), mult = 'last'])
}

MarkFirstLastBy <- function(dataFrame, byColumns) {
     idName <- paste0("ID", paste0(sample(c(LETTERS, letters, 0:9), size=5), collapse=""))
     dataFrame[, idName] <- seq_len(nrow(dataFrame))

     firstIDs <- FirstRowsBy(dataFrame = dataFrame, byColumns = byColumns)[, idName]
     lastIDs  <- LastRowsBy(dataFrame = dataFrame, byColumns = byColumns)[, idName]

     dataFrame$FIRST <-  dataFrame[, idName] %in% firstIDs
     dataFrame$LAST  <-  dataFrame[, idName] %in% lastIDs

     dataFrame[, idName] <- NULL
     return(dataFrame)
}
```

R implementation of useful SAS functions #1



```
> data
  ID Country Sex Age
1  1      PL   F  43
2  3      PL   F  37
3  2      PL   M  25
4  5      RU   F  33
5  6      RU   F  27
6  4      RU   M  63
7  9      EN   F  48
8  7      EN   M  46
9  8      EN   M  52
> FirstRowsBy(data, c("Country", "Sex"))
  ID Country Sex Age
1  1      PL   F  43
2  2      PL   M  25
3  5      RU   F  33
4  4      RU   M  63
5  9      EN   F  48
6  7      EN   M  46
> LastRowsBy(data, c("Country", "Sex"))
  ID Country Sex Age
1  3      PL   F  37
2  2      PL   M  25
3  6      RU   F  27
4  4      RU   M  63
5  9      EN   F  48
6  8      EN   M  52
> MarkFirstLastBy(data, c("Country", "Sex"))
  ID Country Sex Age FIRST  LAST
1  1      PL   F  43  TRUE FALSE
2  3      PL   F  37 FALSE  TRUE
3  2      PL   M  25  TRUE  TRUE
4  5      RU   F  33  TRUE FALSE
5  6      RU   F  27 FALSE  TRUE
6  4      RU   M  63  TRUE  TRUE
7  9      EN   F  48  TRUE  TRUE
8  7      EN   M  46  TRUE FALSE
9  8      EN   M  52 FALSE  TRUE
```

Conclusion

Undoubtedly, R **does meet** all the requirements

involved in biostatistician's work.

It is **perfectly fitted** for this purpose.

R can entirely

replace SAS

as well as **cooperate** with.

Thank you for your attention!

Please find the latest version of this document at:

http://www.scribd.com/AdrianOlszewski

or, in HTML version at:

http://www.r-clinical-research.com

This presentation was published first time on Scribd on April 1st, 2014

Last update: 13th June, 2015

HTML version was created with PDF2HTML provided by Lu Wang