INTRODUCTION

Know your Instructor

- Author "R for Business Analytics"
- Author " R for Cloud Computing"
- Founder "Decisionstats.com"
- University of Tennessee, Knoxville MS (courses in statistics and computer science)
- MBA (IIM Lucknow,India-2003)
- B.Engineering (DCE 2001)

http://linkedin.com/in/ajayohri

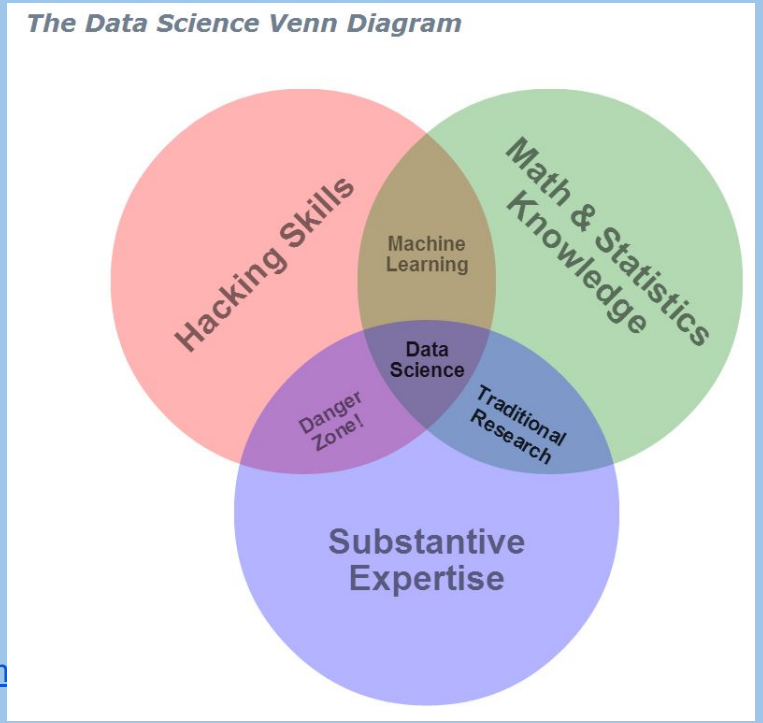
Classroom Rules

- From Instructor

- From Audience
  - mobile phones should be kindly switched off
    - Yes, this includes Whatsapp
  - Ask Questions at end of session
  - Take Notes
  - Please Take Notes

What is data science ?

Hacking (Programming) + Maths/Statistics + Domain Knowledge = Data Science



http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram

Oh really, is this a Data Scientist ?

a data scientist is simply a person who can

write code = in R,Python,Java, SQL, Hadoop (Pig,HQL,MR) etc

= for data storage, querying, summarization, visualization

= how efficiently, and in time (fast results?)

= where on databases, on cloud, servers

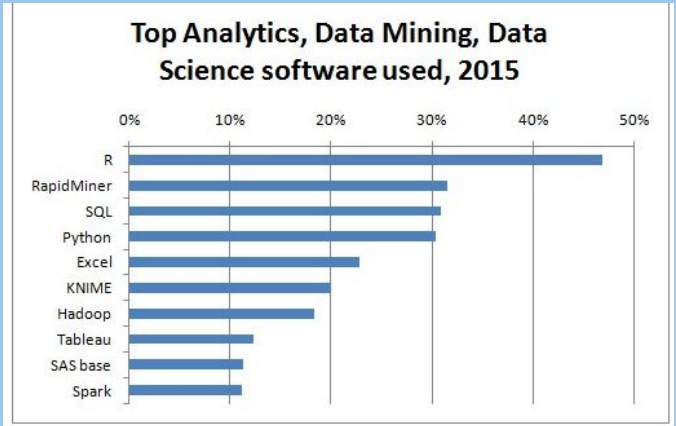
and understand **enough** statistics

to derive **insights** from data

so **business** can make **decisions**

Data Science with R

A popular language in Data Science



R moves up to 5th place in IEEE language rankings

IEEE Spectrum has just published its third annual ranking with its 2016 Top Programming Languages, and the R Language is once again near the top of the list, moving up one place to fifth position.

Language Rank	Types	Spectrum Ranking
1. C		100.0
2. Java		98.1
3. Python		98.0
4. C++		95.9
5. R		87.9
6. C#		86.7
7. PHP		82.8
8. JavaScript		82.2
9. Ruby		74.5
10. Go		71.9

As I said last year (when R moved up to take sixth place), this is an extraordinary result for a domain-specific language. The other four languages in the top 5 (C, Java, Python amd C++) are all general-purpose languages, suitable for just about any programming task. R by contrast is a language specifically for data science, and its high ranking here reflects both the critical importance of data science as a discipline today, and of R as the language of choice for data scientists

What Is R

https://www.r-project.org/about.html

R is an integrated suite of software facilities for data manipulation, calculation and graphical display. It includes

- an effective data handling and storage facility,
- a suite of operators for calculations on arrays, in particular matrices,
- a large, coherent, integrated collection of intermediate tools for data analysis,
- graphical facilities for data analysis and display either on-screen or on hardcopy, and
- a well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.

Install R

https://cran.r-project.org/bin/windows/base/

R-3.3.1 for Windows (32/64 bit)

Download R 3.3.1 for Windows (70 megabytes, 32/64 bit)

Installation and other instructions
New features in this version

If you want to double-check that the package you have downloaded exactly matches the package distributed by R, you can compare the md5sum of the .exe to the true fingerprint. You will need a version of md5sum for windows: both graphical and command line versions are available.

Frequently asked questions

- Does R run under my version of Windows?
- How do I update packages in my previous version of R?
- Should I run 32-bit or 64-bit R?

Please see the R FAQ for general information about R and the R Windows FAQ for Windows-specific information.

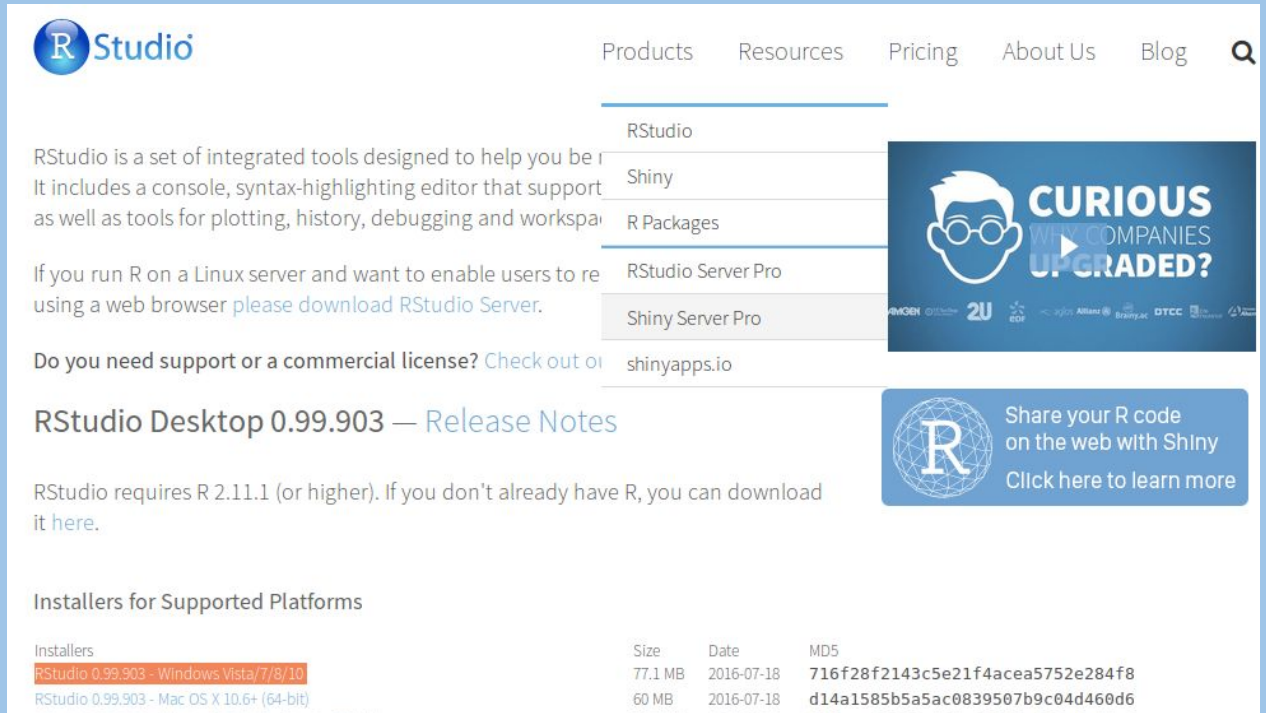
Other builds

- Patches to this release are incorporated in the r-patched snapshot build.
- A build of the development version (which will eventually become the next major release of R) is available in the r-devel snapshot build.
- Previous releases

Note to webmasters: A stable link which will redirect to the current Windows binary release is <CRAN MIRROR>/bin/windows/base/release.htm.

Install RStudio

https://www.rstudio.com/products/rstudio/download/



Statistical Software Landscape

SAS

Python (Pandas)

IBM SPSS

R

Julia

Clojure

Octave

Matlab

JMP

E views

Using R with other software

https://rforanalytics.wordpress.com/useful-links-for-r/using-r-from-other-software/

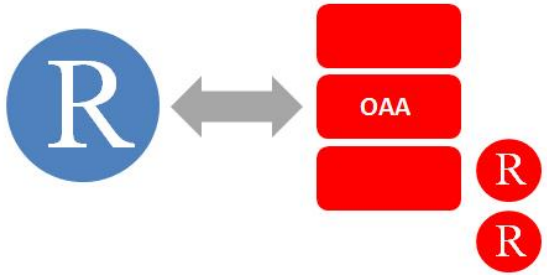


Tableau http://www.tableausoftware.com/new-features/r-integration

Qlik http://qliksolutions.ru/qlikview/add-ons/r-connector-eng/

Oracle R http://www.oracle.com/technetwork/database/database-technologies/r/r-enterprise/overview/index.html

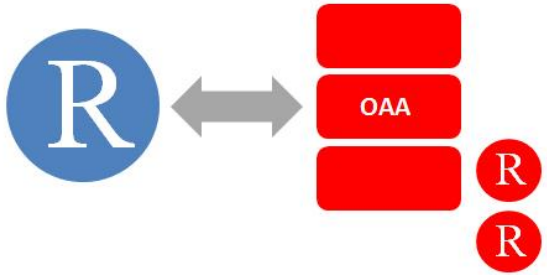
Rapid Miner https://rapid-i.com/content/view/202/206/lang,en/#r

JMP http://blogs.sas.com/jmp/index.php?/archives/298-JMP-Into-R!.html

Using R with other software

https://rforanalytics.wordpress.com/useful-links-for-r/using-r-from-other-software/

SAS/IML http://www.sas.com/technologies/analytics/statistics/iml/index.html



Teradata http://developer.teradata.com/applications/articles/in-database-analytics-with-ter

Pentaho http://bigdatatechworld.blogspot.in/2013/10/integration-of-rweka-with-pentaho-data.html

IBM SPSS

https://www14.software.ibm.com/webapp/iwm/web/signup.do?source=ibm-analytics&S_PKG=ov18855&S_TACT=M161003W&dynform=127&lang=en_US

TIBCO TERR

http://spotfire.tibco.com/discover-spotfire/what-does-spotfire-do/predictive-analytics/tibco-enterprise-runtime-for-r-terr

Some Advantages of R

open source

free

large number of algorithms and packages esp for statistics

flexible

very good for data visualization

superb community

rapidly growing

can be used with other software

Some Disadvantages of R

in memory (RAM) usage

steep learning curve

some IT departments frown on open source

verbose documentation

tech support

evolving ecosystem for corporates

Solutions for Disadvantages of R

in memory (RAM) usage ⇒ specialized packages, in database computing

steep learning curve ⇒ TRAINING !!!

some IT departments frown on open source ⇒ TRAINING and education!

verbose documentation ⇒ CRAN View , R Documentation

tech support ⇒ expanding pool of resources

evolving ecosystem for corporates ⇒ getting better with MS et al

R used by Government

- In the early days of the Deepwater Horizon disaster, NIST used uncertainty analysis in R to harmonize spill estimates from various sources, and to provide ranges of estimates to other agencies and the media.
- Before new drugs are allowed on the market, the FDA works with pharmaceutical companies to verify safety and efficacy through clinical trials. Despite a false perception that only commercial software may be used, many pharmaceutical companies are now using open-source R to analyze data from clinical trials.
- The National Weather Service uses R for research and development of models to predict river flooding.
- The newly-formed Consumer Financial Protection Bureau -- freed from the restrictions of a legacy IT infrastructure -- is championing the use of open-source technologies in government.
- Local governments are also building data-based applications. The SF Estuary Institute uses R and Google Maps to provide a tool to track pollution in the San Francisco Bay area.

http://gsnmagazine.com/node/26483?c=cyber_security

R used by Telecom

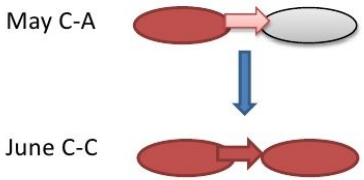
- Churn using

Social Network Analysis

http://www.slideshare.net/dataspora/social-network-a

Results: A Customer With a Canceller in Their Network Churns at Twice the Rate

Types of Connections (Edges)



reality	expected by chance	delta
X	Y	**2.0**

In essence, we are asking whether being connected to another canceller has any effect on one's rate of cancellation. It turns out that it does.

And if we only look at voluntary port-outs, we see that customers churn at 3x the rate.

R used by Insurance

a few more insurance related packages:

- ChainLadder – Reserving methods in R. The package provides Mack-, Munich-, Bootstrap, and Multivariate-chain-ladder methods, as well as the LDF Curve Fitting methods of Dave Clark and GLM-based reserving models.
- cplm – Monte Carlo EM algorithms and Bayesian methods for fitting Tweedie compound Poisson linear models
- lossDev – A Bayesian time series loss development model. Features include skewed-t distribution with time-varying scale parameter, Reversible Jump MCMC for determining the functional form of the consumption path, and a structural break in this path; by Christopher W. Laws and Frank A. Schmid
- actuar: Loss distributions modelling, risk theory (including ruin theory), simulation of compound hierarchical models and credibility theory check out the actuar package by C. Dutang, V. Goulet and M. Pigeon.
- favir: Formatted Actuarial Vignettes in R. FAViR lowers the learning curve of the R environment. It is a series of peer-reviewed Sweave papers that use a consistent style.
- mondate: R packackge to keep track of dates in terms of months
- lifecontingencies – Package to perform actuarial evaluation of life contingencies

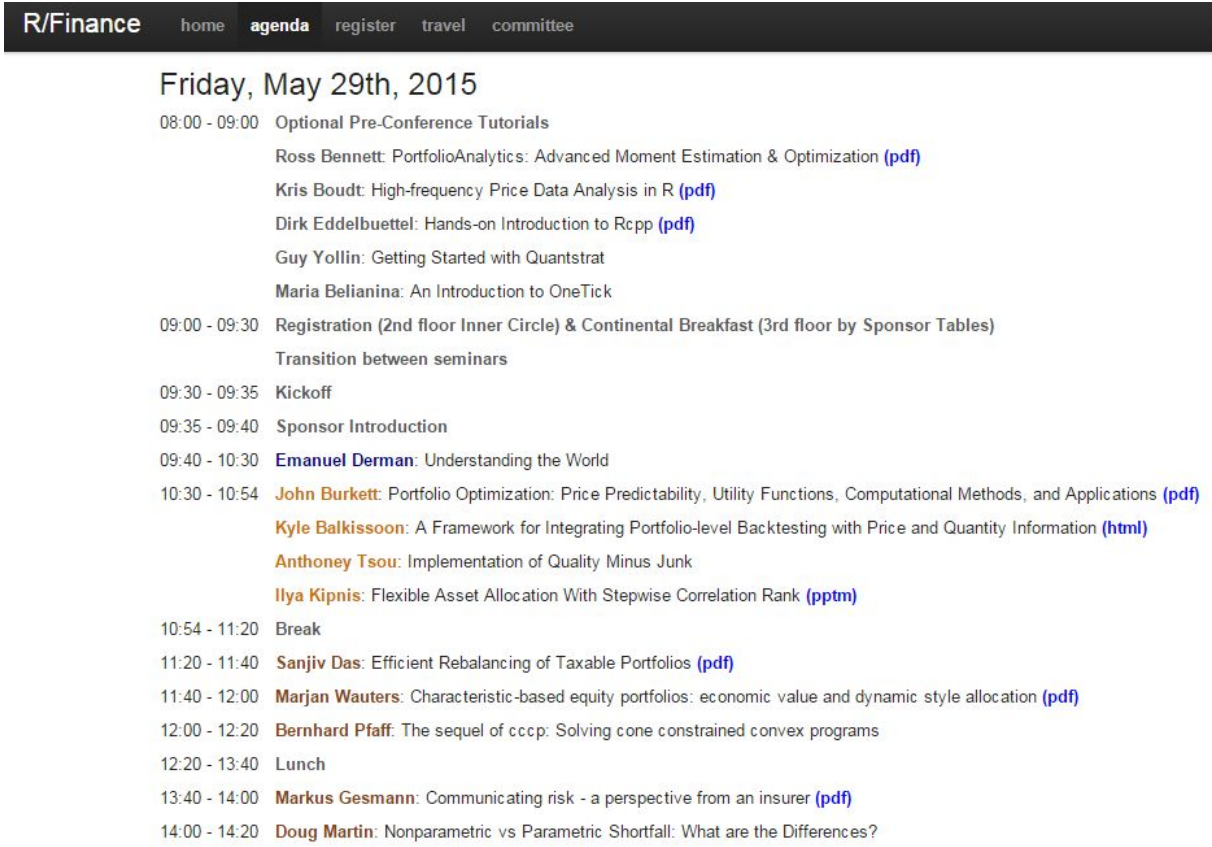
and

Introduction to R for Actuaries by Nigel de Silva

and http://www.rininsurance.com/

R in Finance

http://www.rinfinance.com/



R/Finance home agenda register travel committee

Friday, May 29th, 2015

08:00 - 09:00 **Optional Pre-Conference Tutorials**

Ross Bennett: PortfolioAnalytics: Advanced Moment Estimation & Optimization **(pdf)**

Kris Boudt: High-frequency Price Data Analysis in R **(pdf)**

Dirk Eddelbuettel: Hands-on Introduction to Rcpp **(pdf)**

Guy Yollin: Getting Started with Quantstrat

Maria Belianina: An Introduction to OneTick

09:00 - 09:30 **Registration (2nd floor Inner Circle) & Continental Breakfast (3rd floor by Sponsor Tables)**

Transition between seminars

09:30 - 09:35 **Kickoff**

09:35 - 09:40 **Sponsor Introduction**

09:40 - 10:30 **Emanuel Derman**: Understanding the World

10:30 - 10:54 **John Burkett**: Portfolio Optimization: Price Predictability, Utility Functions, Computational Methods, and Applications **(pdf)**

Kyle Balkissoon: A Framework for Integrating Portfolio-level Backtesting with Price and Quantity Information **(html)**

Anthoney Tsou: Implementation of Quality Minus Junk

Ilya Kipnis: Flexible Asset Allocation With Stepwise Correlation Rank **(pptm)**

10:54 - 11:20 Break

11:20 - 11:40 **Sanjiv Das**: Efficient Rebalancing of Taxable Portfolios **(pdf)**

11:40 - 12:00 **Marjan Wauters**: Characteristic-based equity portfolios: economic value and dynamic style allocation **(pdf)**

12:00 - 12:20 **Bernhard Pfaff**: The sequel of cccp: Solving cone constrained convex programs

12:20 - 13:40 **Lunch**

13:40 - 14:00 **Markus Gesmann**: Communicating risk - a perspective from an insurer **(pdf)**

14:00 - 14:20 **Doug Martin**: Nonparametric vs Parametric Shortfall: What are the Differences?

R in Finance

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

This CRAN Task View contains a list of packages useful for empirical work in Finance, grouped by topic.

- The Rmetrics suite of packages comprises fArma, fAsianOptions, fAssets, fBasics, fBonds, timeDate (formerly: fCalendar), fCopulae, fExoticOptions, fExtremes, fGarch, fImport,fNonlinear, fOptions, fPortfolio, fRegression, timeSeries (formerly: fSeries), fTrading, fUnitRoots and contains a very large number of relevant functions for different aspect of empirical and computational finance.
- The RQuantLib package provides several option-pricing functions as well as some fixed-income functionality from the QuantLib project to R.
- The quantmod package offers a number of functions for quantitative modelling in finance as well as data acqusition, plotting and other utilities.
- The portfolio package contains classes for equity portfolio management; the portfolioSim builds a related simulation framework. The backtest offers tools to explore portfolio-based hypotheses about financial instruments. The stockPortfolio package provides functions for single index, constant correlation and multigroup models. The pa package offers performance attribution functionality for equity portfolios.
- The PerformanceAnalytics package contains a large number of functions for portfolio performance calculations and risk management.

R in Pharma

http://blog.revolutionanalytics.com/2013/08/r-drug-development-and-the-fda.html

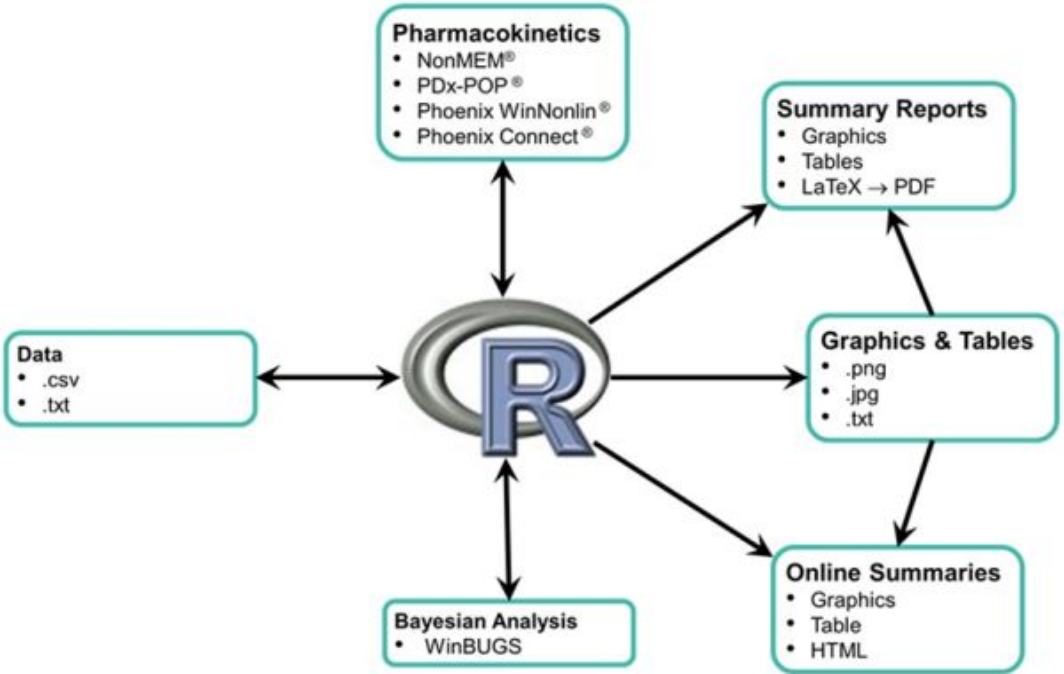
Opening the Doors to Open Source Programming in Drug Development.

R: Regulatory Compliance and Validation Issues A Guidance Document for the Use of R in Regulated Clinical Trial Environments in which he concluded that useR 2012 FDA statistician Jea Brodsky presented a poster described how FDA scientists “use R on a daily basis” and have themselves written R packages for use at various stages in the drug submission process.

Open Source Software in the Biopharma Industry: Challenges and Opportunities,

R in Pharma

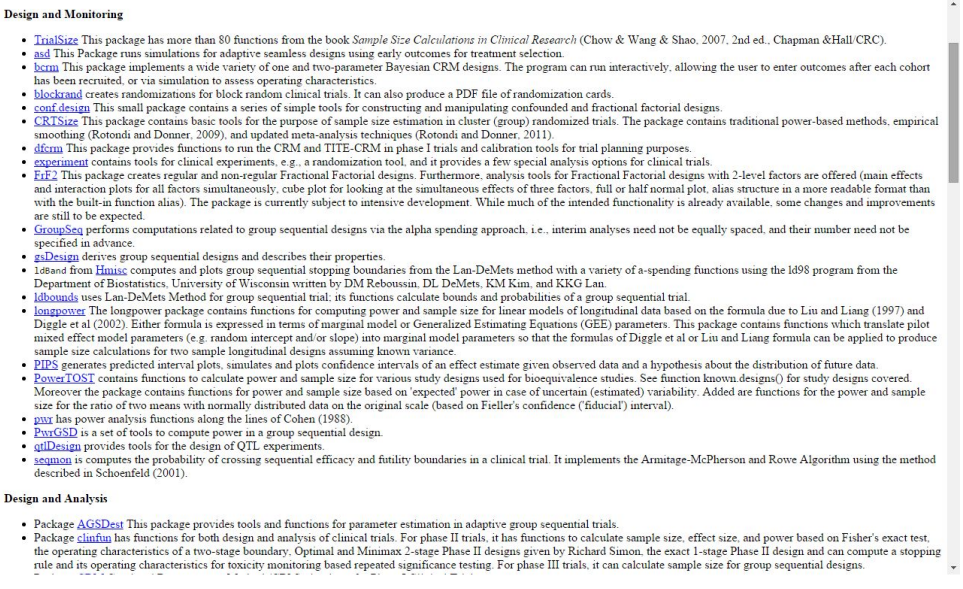
http://web.quanticate.com/bid/102741/Using-the-Statistical-Programming-Language-R-in-the-Pharma-Industry



R in Pharma

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

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.



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.
- CRTSize This package contains basic tools for the purpose of sample size estimation in cluster (group) randomized trials. The package contains traditional power-based methods, empirical smoothing (Rotondi and Donner, 2009), and updated meta-analysis techniques (Rotondi and Donner, 2011).
- dfcrm This package provides functions to run the CRM and TITE-CRM in phase I trials and calibration tools for trial planning purposes.
- experiment contains tools for clinical experiments, e.g., a randomization tool, and it provides a few special analysis options for clinical trials.
- FrF2 This package creates regular and non-regular Fractional Factorial designs. Furthermore, analysis tools for Fractional Factorial designs with 2-level factors are offered (main effects and interaction plots for all factors simultaneously, cube plot for looking at the simultaneous effects of three factors, full or half normal plot, alias structure in a more readable format than with the built-in function alias). The package is currently subject to intensive development. While much of the intended functionality is already available, some changes and improvements are still to be expected.
- GroupSeq performs computations related to group sequential designs via the alpha spending approach, i.e., interim analyses need not be equally spaced, and their number need not be specified in advance.
- gsDesign derives group sequential designs and describes their properties.
- ldBand from Hmisc computes and plots group sequential stopping boundaries from the Lan-DeMets method with a variety of a-spending functions using the ld98 program from the Department of Biostatistics, University of Wisconsin written by DM Reboussin, DL DeMets, KM Kim, and KKG Lan.
- ldbounds uses Lan-DeMets Method for group sequential trial; its functions calculate bounds and probabilities of a group sequential trial.
- longpower The longpower package contains functions for computing power and sample size for linear models of longitudinal data based on the formula due to Liu and Liang (1997) and Diggle et al (2002). Either formula is expressed in terms of marginal model or Generalized Estimating Equations (GEE) parameters. This package contains functions which translate pilot mixed effect model parameters (e.g. random intercept and/or slope) into marginal model parameters so that the formulas of Diggle et al or Liu and Liang formula can be applied to produce sample size calculations for two sample longitudinal designs assuming known variance.
- PIPS generates predicted interval plots, simulates and plots confidence intervals of an effect estimate given observed data and a hypothesis about the distribution of future data.
- PowerTOST contains functions to calculate power and sample size for various study designs used for bioequivalence studies. See function known.designs() for study designs covered. Moreover the package contains functions for power and sample size based on 'expected' power in case of uncertain (estimated) variability. Added are functions for the power and sample size for the ratio of two means with normally distributed data on the original scale (based on Fieller's confidence ('fiducial') interval).
- pwr has power analysis functions along the lines of Cohen (1988).
- PwrGSD is a set of tools to compute power in a group sequential design.
- qtlDesign provides tools for the design of QTL experiments.
- seqmon is computes the probability of crossing sequential efficacy and futility boundaries in a clinical trial. It implements the Armitage-McPherson and Rowe Algorithm using the method described in Schoenfeld (2001).

Design and Analysis

- Package AGSDest This package provides tools and functions for parameter estimation in adaptive group sequential trials.
- Package clinfun has functions for both design and analysis of clinical trials. For phase II trials, it has functions to calculate sample size, effect size, and power based on Fisher's exact test, the operating characteristics of a two-stage boundary, Optimal and Minimax 2-stage Phase II designs given by Richard Simon, the exact 1-stage Phase II design and can compute a stopping rule and its operating characteristics for toxicity monitoring based repeated significance testing. For phase III trials, it can calculate sample size for group sequential designs.

Companies using R

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

ANZ, the fourth largest bank in Australia, using R for credit risk analysis

Bank of America uses R for reporting.

The Consumer Financial Protection Bureau uses R for data analysis.

Facebook

Facebook and R:

- Analysis of Facebook Status Updates
- Facebook's Social Network Graph
- How Google and Facebook are using R
- Predicting Colleague Interactions with R

Refresher in Statistics

Mean

Arithmetic Mean- the sum of the values divided by the number of values.

The geometric mean is an average that is useful for sets of positive numbers that are interpreted according to their product and not their sum (as is the case with the arithmetic mean) e.g. rates of growth.

Median

the **median** is the number separating the higher half of a data sample, a population, or a probability distribution, from the lower hal

Mode-

The "mode" is the value that occurs most often.

Refresher in Statistics

Range

the **range** of a set of data is the difference between the largest and smallest values.

Variance

mean of squares of differences of values from mean

Standard Deviation

square root of its variance

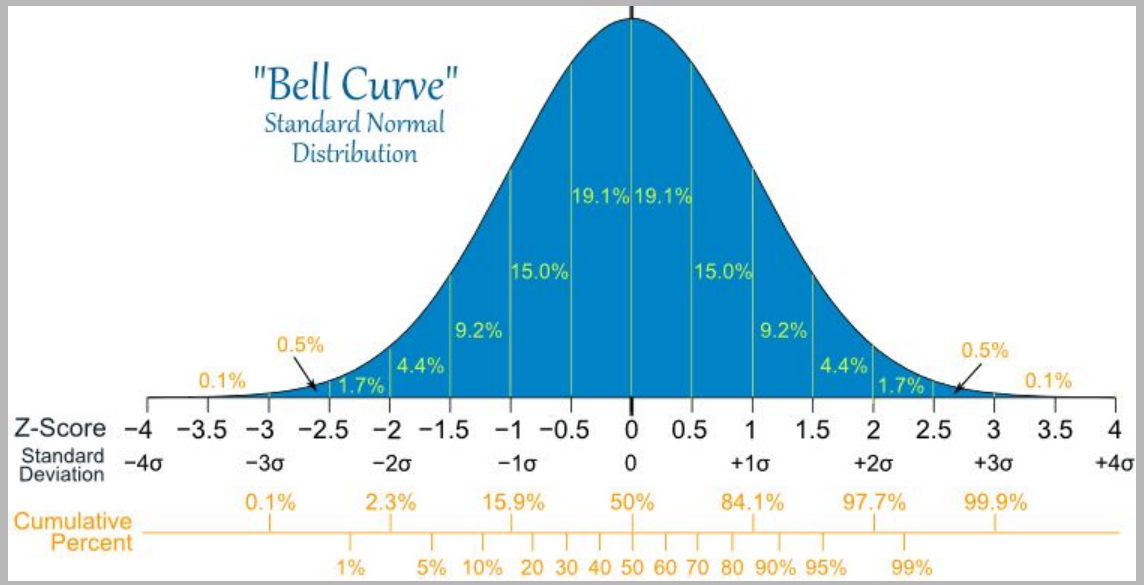
Frequency

a **frequency distribution** is a table that displays the **frequency** of various outcomes in a sample.

Distributions

Normal

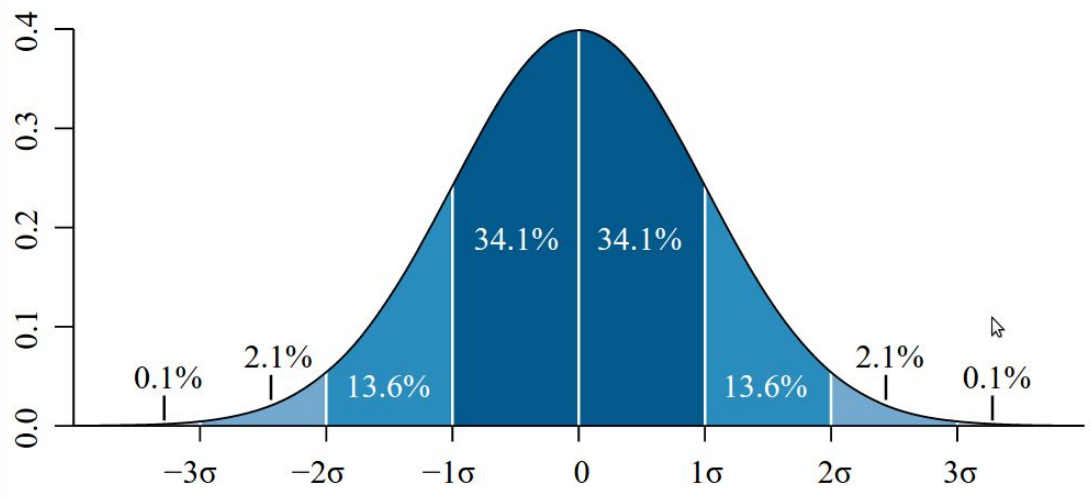
The simplest case of a normal distribution is known as the *standard normal distribution*. This is a special case where μ=0 and σ=1,



Refresher in Statistics

Probability Distribution

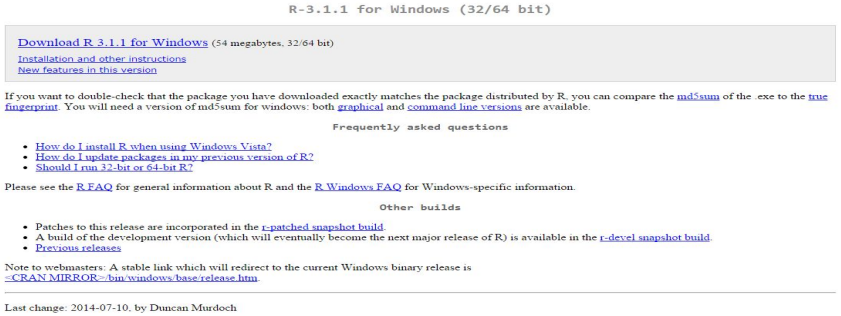
The probability density function (pdf) of the normal distribution, also called Gaussian or "bell curve", the most important continuous random distribution. As notated on the figure, the probabilities of intervals of values correspond to the area under the curve.



Pre Requisites

- Installation of R

http://cran.rstudio.com/bin/windows/base/

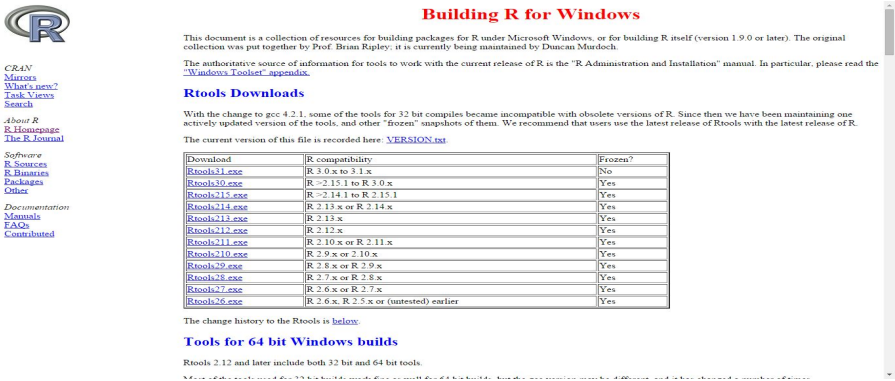


- R Studio

- R Packages

Pre Requisites

- Installation of R
  - Rtools
    - http://cran.rstudio.com/bin/windows/Rtools/



CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

Documentation
Manuals
FAQs
Contributed

Building R for Windows

This document is a collection of resources for building packages for R under Microsoft Windows, or for building R itself (version 1.9.0 or later). The original collection was put together by Prof. Brian Ripley; it is currently being maintained by Duncan Murdoch.

The authoritative source of information for tools to work with the current release of R is the "R Administration and Installation" manual. In particular, please read the "Windows Toolset" appendix.

Rtools Downloads

With the change to gcc 4.2.1, some of the tools for 32 bit compiles became incompatible with obsolete versions of R. Since then we have been maintaining one actively updated version of the tools, and other "frozen" snapshots of them. We recommend that users use the latest release of Rtools with the latest release of R.

The current version of this file is recorded here: VERSION.txt.

Download	R compatibility	Frozen?
Rtools31.exe	R 3.0.x to 3.1.x	No
Rtools30.exe	R >2.15.1 to R 3.0.x	Yes
Rtools215.exe	R >2.14.1 to R 2.15.1	Yes
Rtools214.exe	R 2.13.x or R 2.14.x	Yes
Rtools213.exe	R 2.13.x	Yes
Rtools212.exe	R 2.12.x	Yes
Rtools211.exe	R 2.10.x or R 2.11.x	Yes
Rtools210.exe	R 2.9.x or 2.10.x	Yes
Rtools29.exe	R 2.8.x or R 2.9.x	Yes
Rtools28.exe	R 2.7.x or R 2.8.x	Yes
Rtools27.exe	R 2.6.x or R 2.7.x	Yes
Rtools26.exe	R 2.6.x, R 2.5.x or (untested) earlier	Yes

The change history to the Rtools is below.

Tools for 64 bit Windows builds

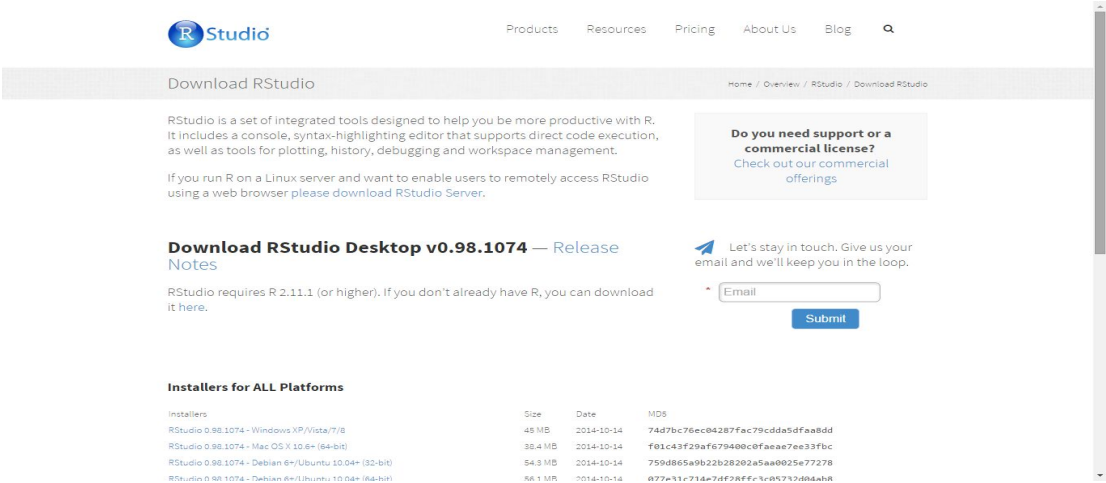
Rtools 2.12 and later include both 32 bit and 64 bit tools.

- R Studio
- R Packages

Pre Requisites

- Installation of R
  - RTools
- R Studio

http://www.rstudio.com/products/rstudio/download/

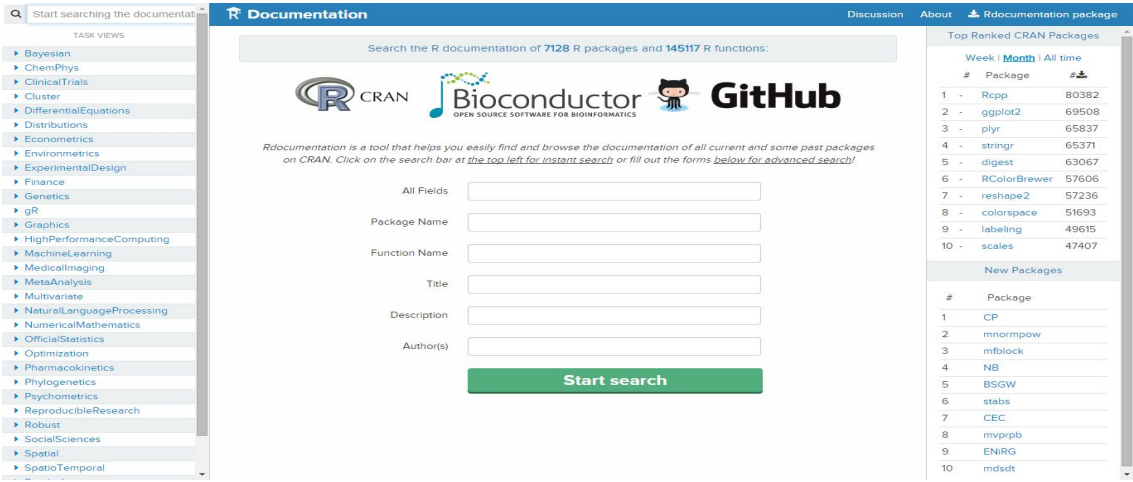


- R Packages

Pre Requisites

- Installation of R
  - RTools
- R Studio

http://www.rstudio.com/products/rstudio/download/



- R Packages about eight packages supplied with the R distribution and many more are available through the CRAN family of Internet sites covering a very wide range of modern statistics.

CRAN

107 sites in 49 regions

CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

Documentation
Manuals
FAQs
Contributed

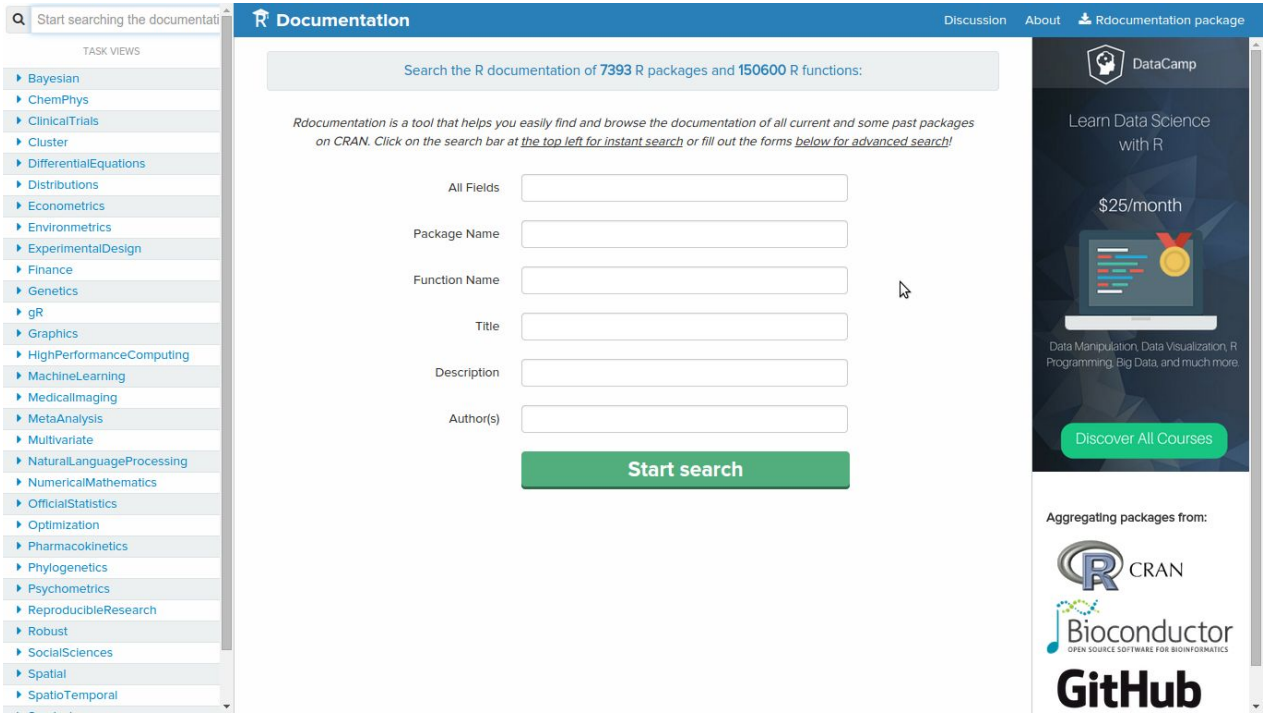
CRAN Mirrors

The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. Some statistics on the status of the mirrors can be found here: main page, windows release, windows old release.

0-Cloud	
http://cran.rstudio.com/	Rstudio, automatic redirection to servers worldwide
Algeria	
http://cran.usthb.dz/	University of Science and Technology Houari Boumediene
Argentina	
http://mirror.fcaglp.unlp.edu.ar/CRAN/	Universidad Nacional de La Plata
Australia	
http://cran.csiro.au/	CSIRO
http://cran.ms.unimelb.edu.au/	University of Melbourne
Austria	
http://cran.at.r-project.org/	Wirtschaftsuniversitaet Wien
Belgium	
http://www.freestatistics.org/cran/	K.U.Leuven Association
Brazil	
http://nbcgib.uesc.br/mirrors/cran/	Center for Comp. Biol. at Universidade Estadual de Santa Cruz
http://cran-r.c3sl.ufpr.br/	Universidade Federal do Parana
http://cran.fiocruz.br/	Oswaldo Cruz Foundation, Rio de Janeiro
http://www.vps.fmvz.usp.br/CRAN/	University of Sao Paulo, Sao Paulo
http://brieger.esalq.usp.br/CRAN/	University of Sao Paulo, Piracicaba
Canada	
http://cran.stat.sfu.ca/	Simon Fraser University, Burnaby
http://mirror.its.dal.ca/cran/	Dalhousie University, Halifax
http://cran.utstat.utoronto.ca/	University of Toronto
http://cran.skazkaforyou.com/	iWeb, Montreal
http://cran.parentingamerica.com/	iWeb, Montreal
Chile	
http://dirichlet.mat.puc.cl/	Pontificia Universidad Catolica de Chile, Santiago
China	
http://ftp.ctex.org/mirrors/CRAN/	CTEX.ORG
http://mirror.bjtu.edu.cn/cran/	Beijing Jiaotong University, Beijing
http://mirrors.opencas.cn/cran/	Chinese Academy of Sciences, Beijing

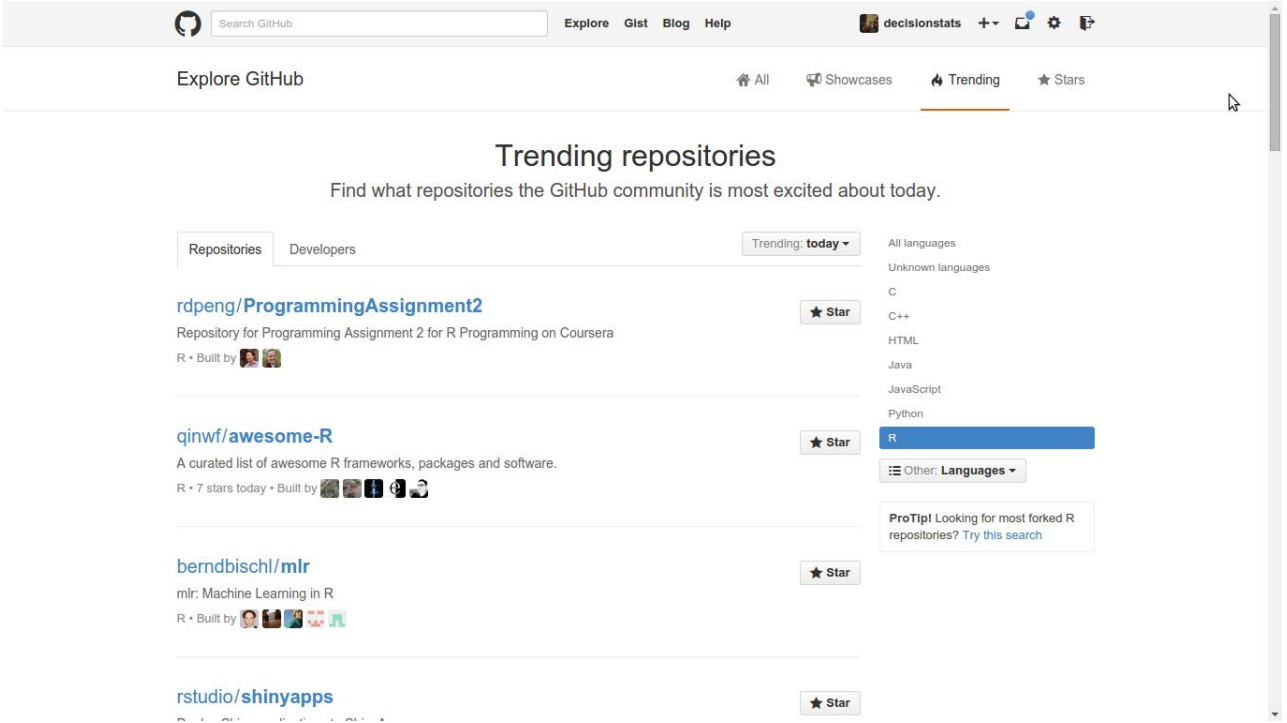
Non CRAN Repositories

http://www.rdocumentation.org/



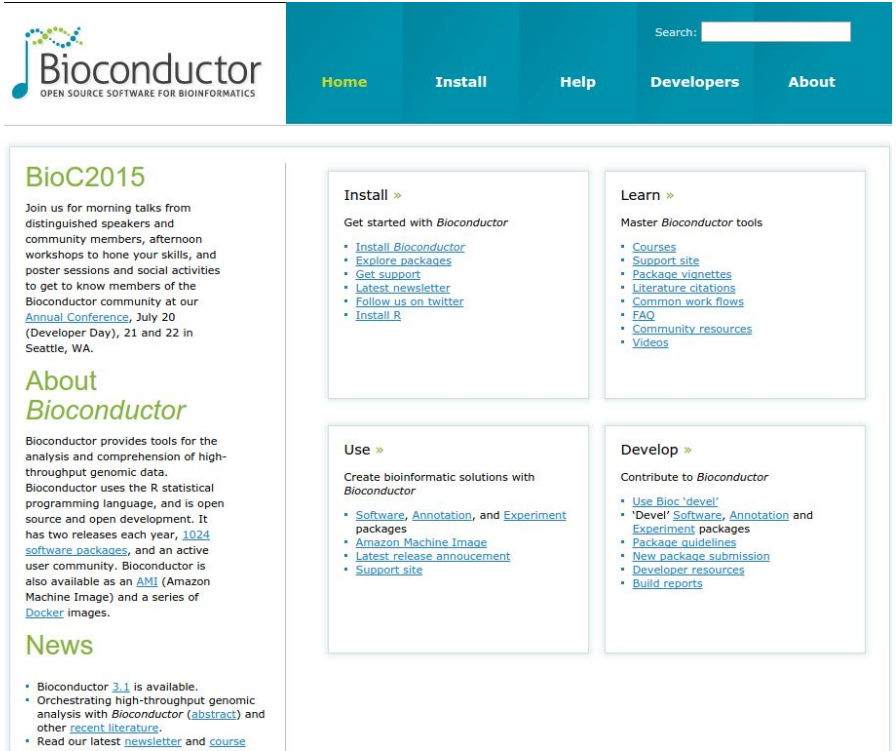
github

https://github.com/trending?l=R



bioconductor

http://www.bioconductor.org/



Install R

https://cran.r-project.org/bin/windows/base/

R-3.3.1 for Windows (32/64 bit)

Download R 3.3.1 for Windows (70 megabytes, 32/64 bit)

Installation and other instructions
New features in this version

If you want to double-check that the package you have downloaded exactly matches the package distributed by R, you can compare the md5sum of the .exe to the true fingerprint. You will need a version of md5sum for windows: both graphical and command line versions are available.

Frequently asked questions

- Does R run under my version of Windows?
- How do I update packages in my previous version of R?
- Should I run 32-bit or 64-bit R?

Please see the R FAQ for general information about R and the R Windows FAQ for Windows-specific information.

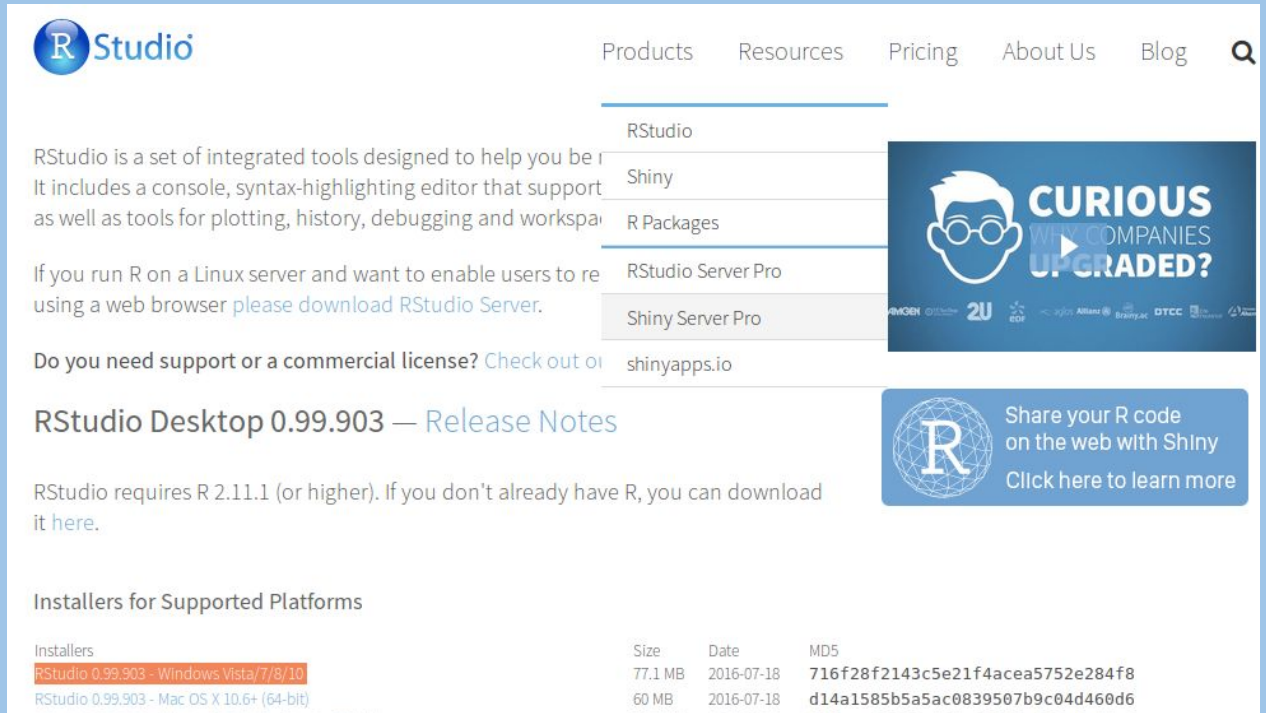
Other builds

- Patches to this release are incorporated in the r-patched snapshot build.
- A build of the development version (which will eventually become the next major release of R) is available in the r-devel snapshot build.
- Previous releases

Note to webmasters: A stable link which will redirect to the current Windows binary release is <CRAN MIRROR>/bin/windows/base/release.htm.

Install RStudio

https://www.rstudio.com/products/rstudio/download/



Pre Requisites

- R Packages

install.packages() INSTALLS
update.packages() UPDATES
library() LOADS

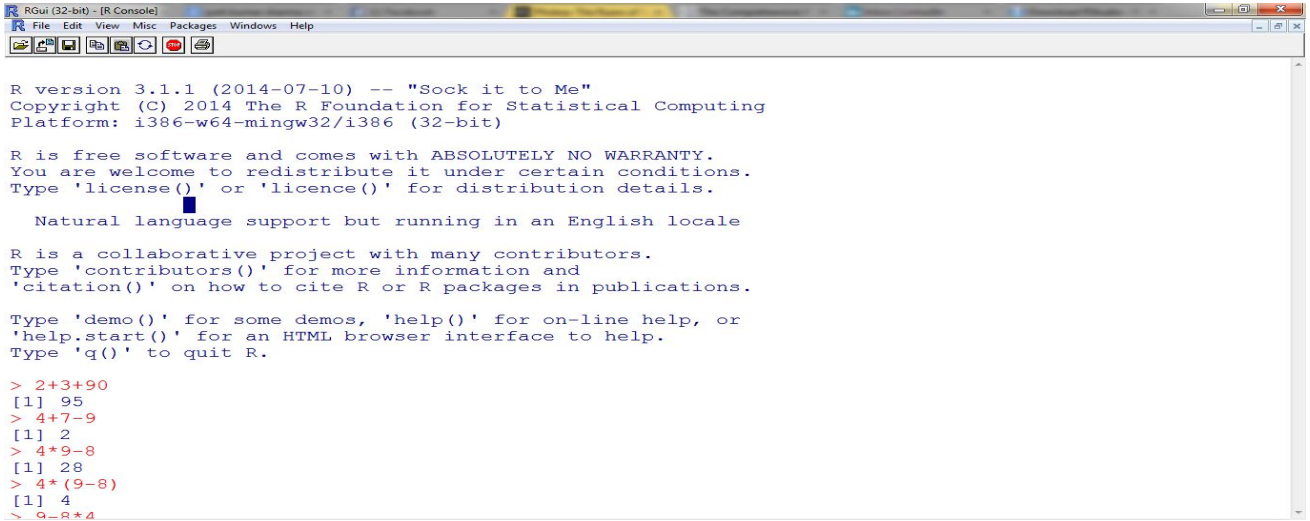
- Packages are **installed** once, updated periodically, but **loaded** every time

Interfaces to R

- Console

Default

Customization



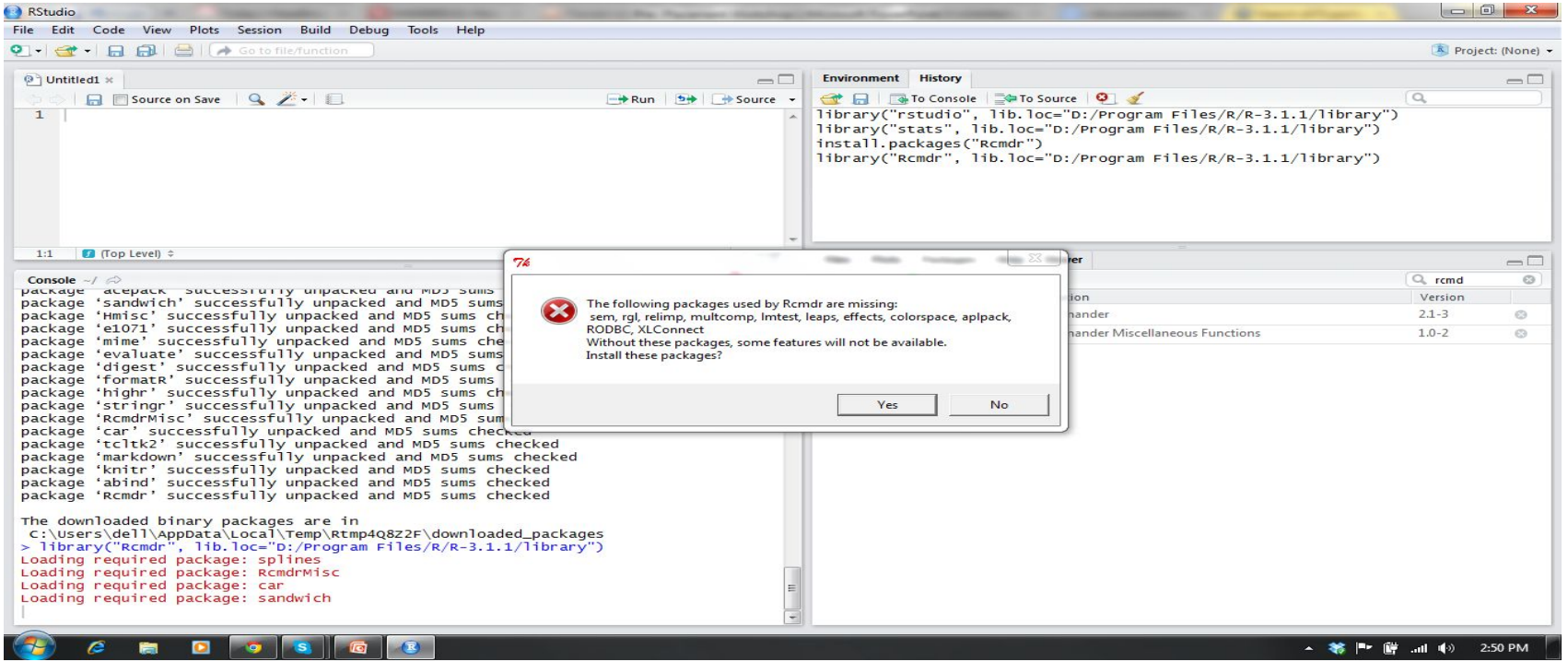
- IDE

- GUI

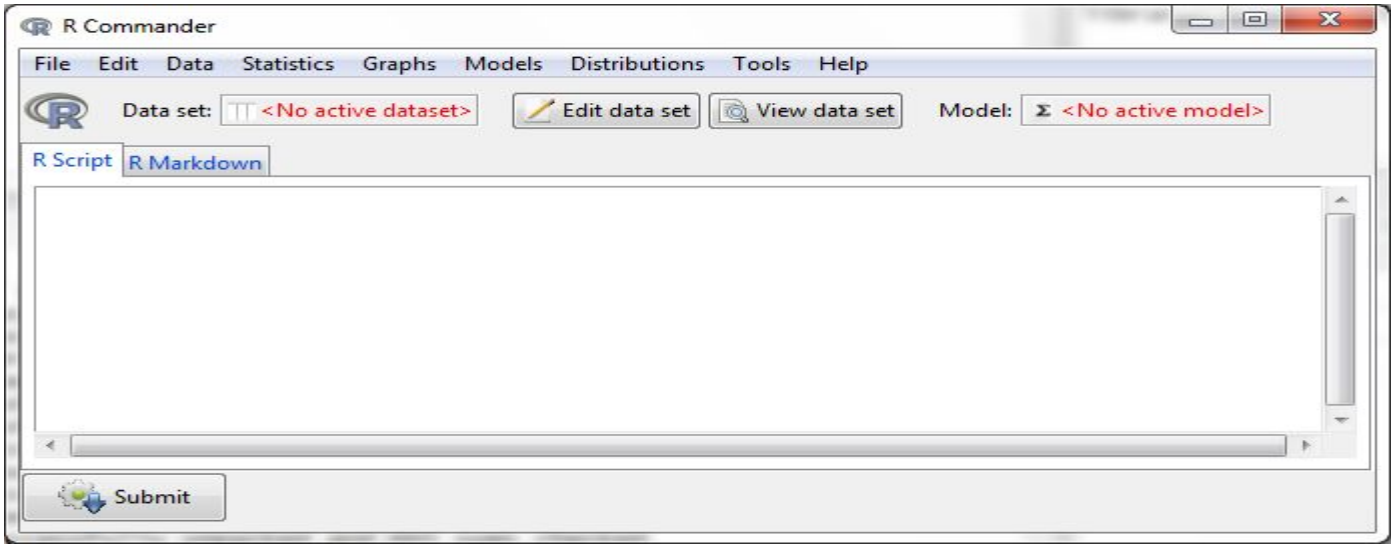
Graphical Interfaces to R

- R Commander
- Rattle
- Deducer

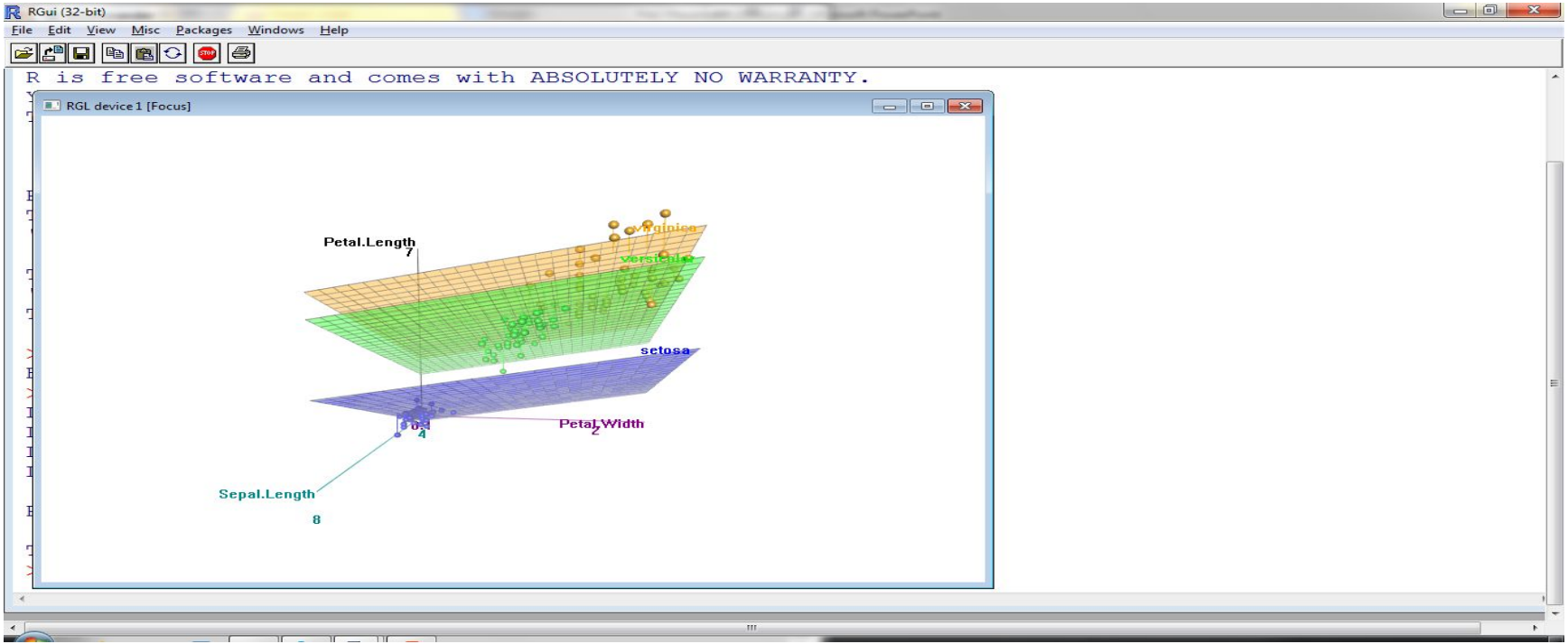
Installation of R Commander



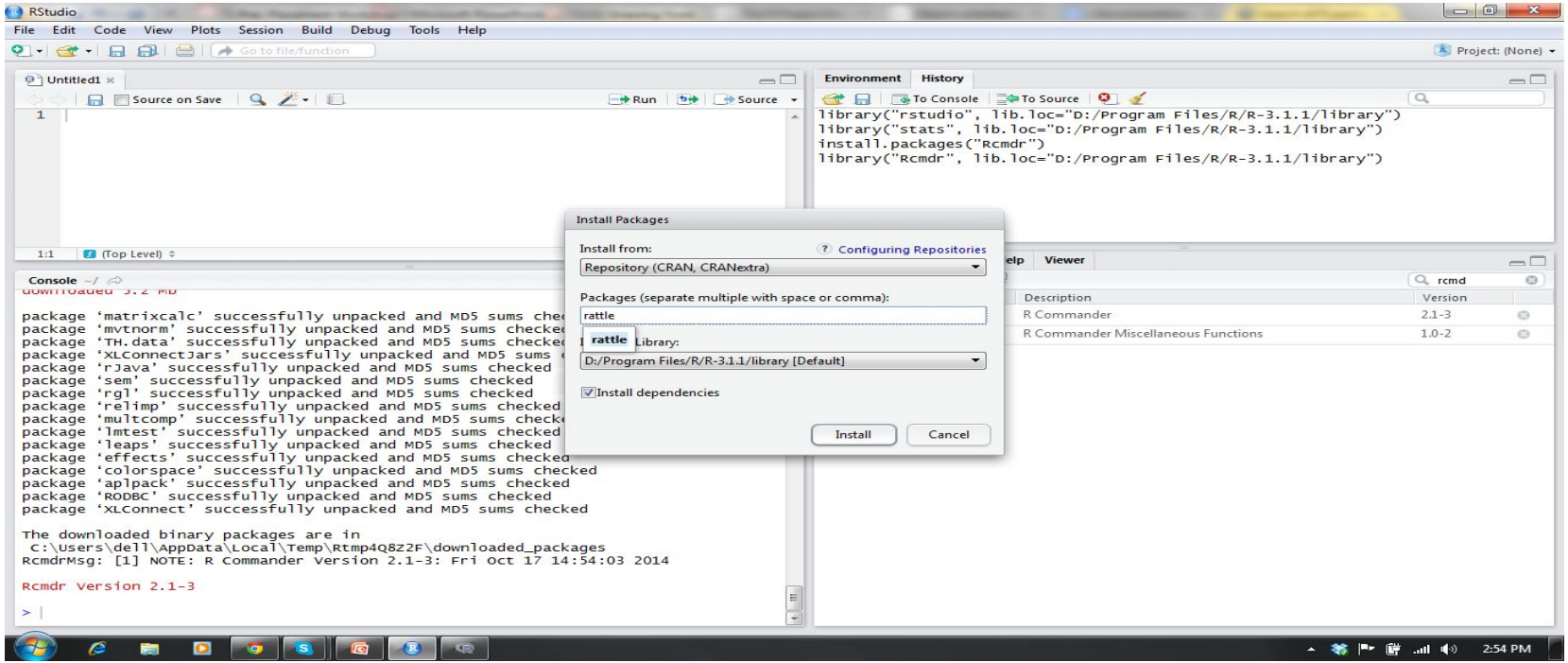
Overview of R Commander



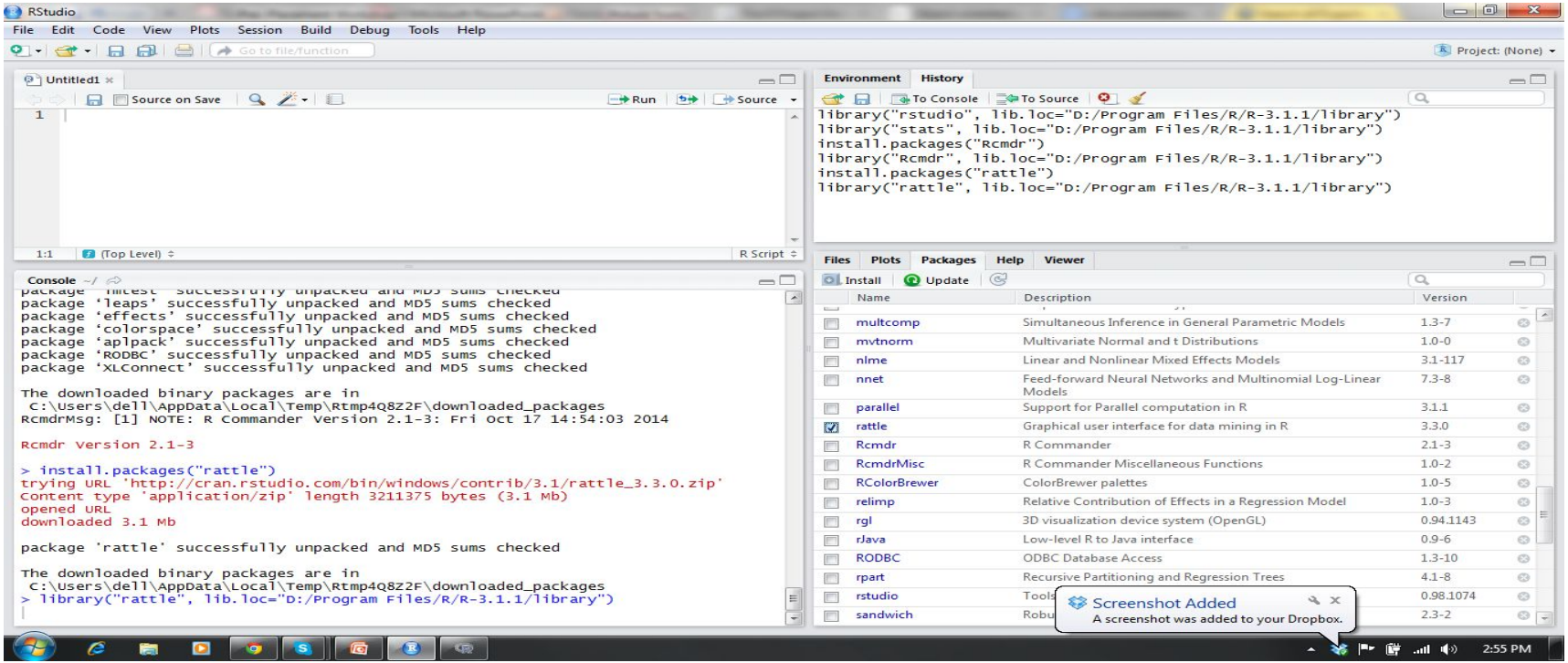
Demo
R Commander – 3D Graphs



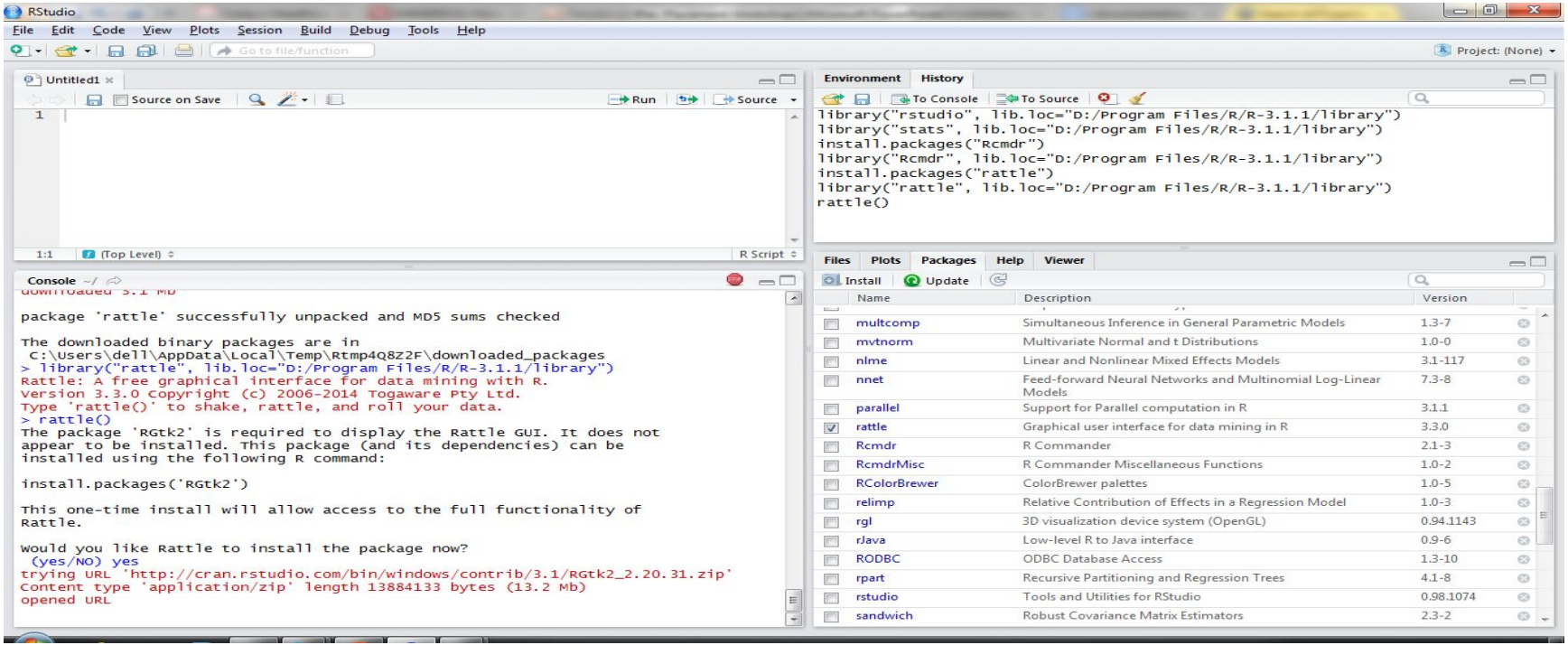
Installation of Rattle



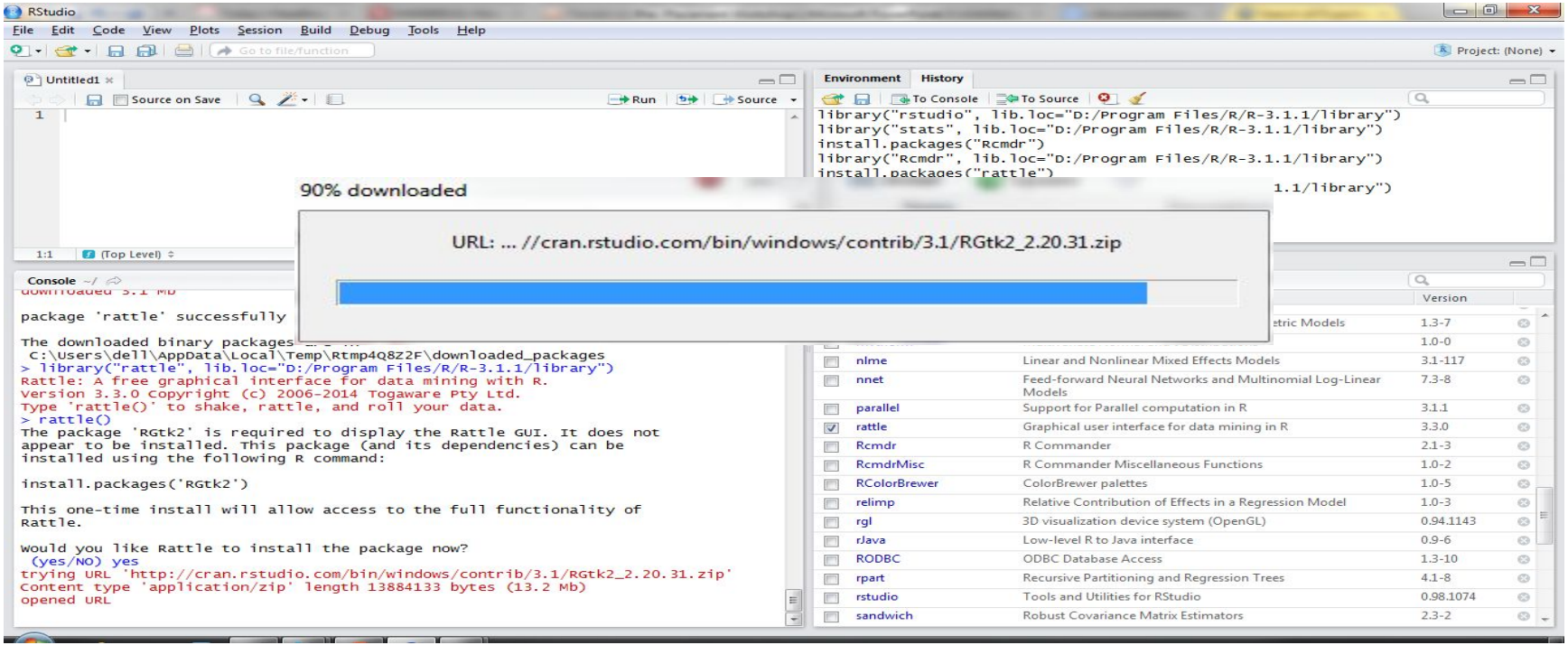
Installation of Rattle



Installation of Rattle

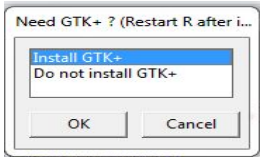


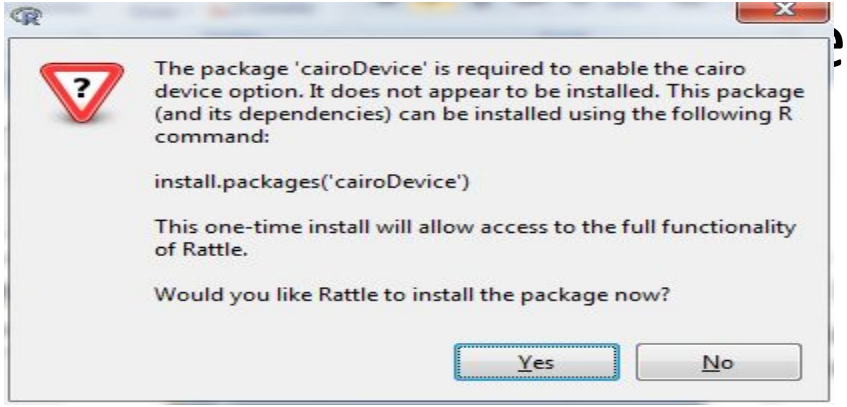
Installation of Rattle

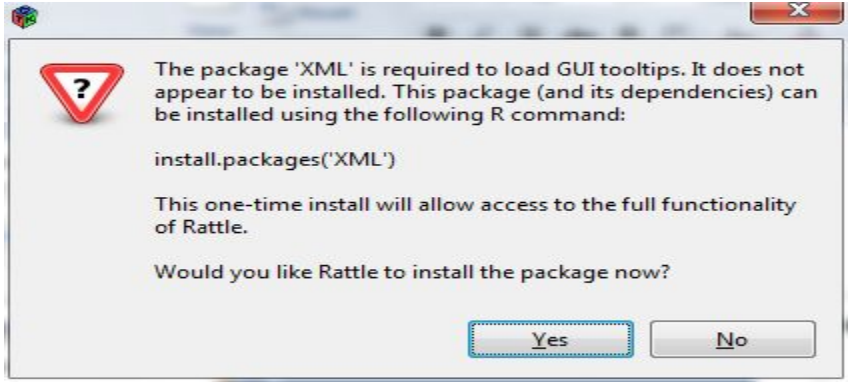


Installation of Rattle

- GTK+ Installation Necessary

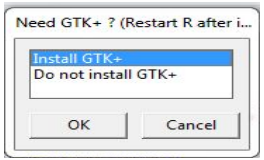




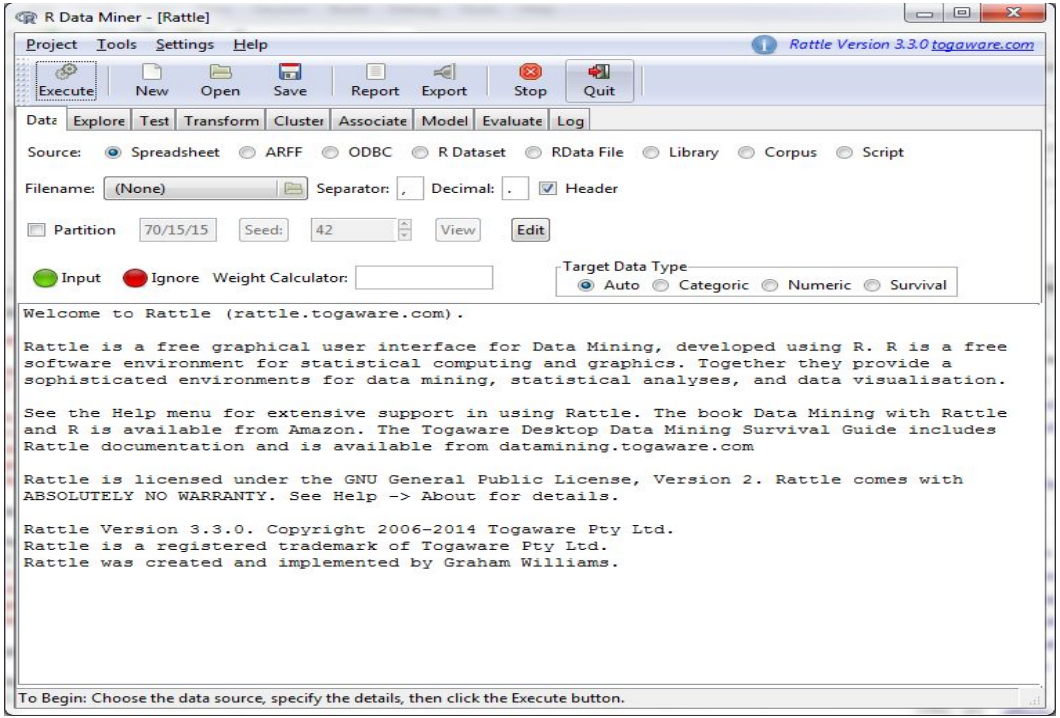


Installation of Rattle

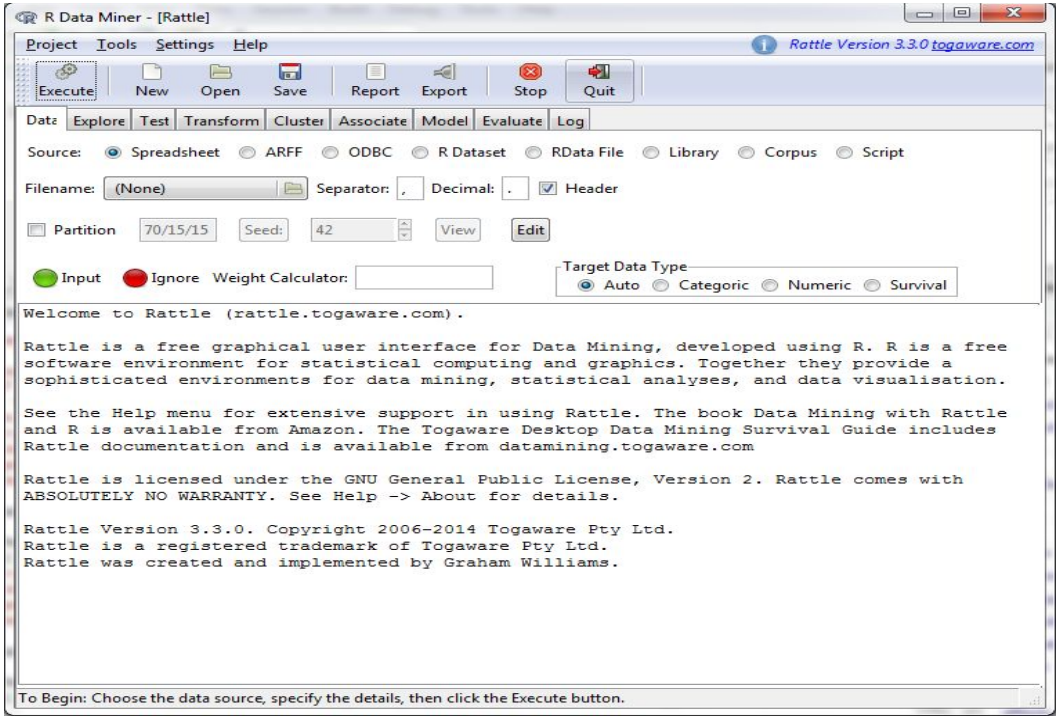
- GTK+ Installation Necessary



Overview of Rattle



Demo Rattle


R Data Miner - [Rattle]
Project Tools Settings Help
Rattle Version 3.3.0 togaware.com
Execute New Open Save Report Export Stop Quit
Data Explore Test Transform Cluster Associate Model Evaluate Log
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 book Data Mining with Rattle and R is available from Amazon. 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 3.3.0. Copyright 2006-2014 Togaware Pty Ltd.
Rattle is a registered trademark of Togaware Pty Ltd.
Rattle was created and implemented by Graham Williams.
To Begin: Choose the data source, specify the details, then click the Execute button.

RStudio

RStudio Desktop enables you with following advantages of native R console

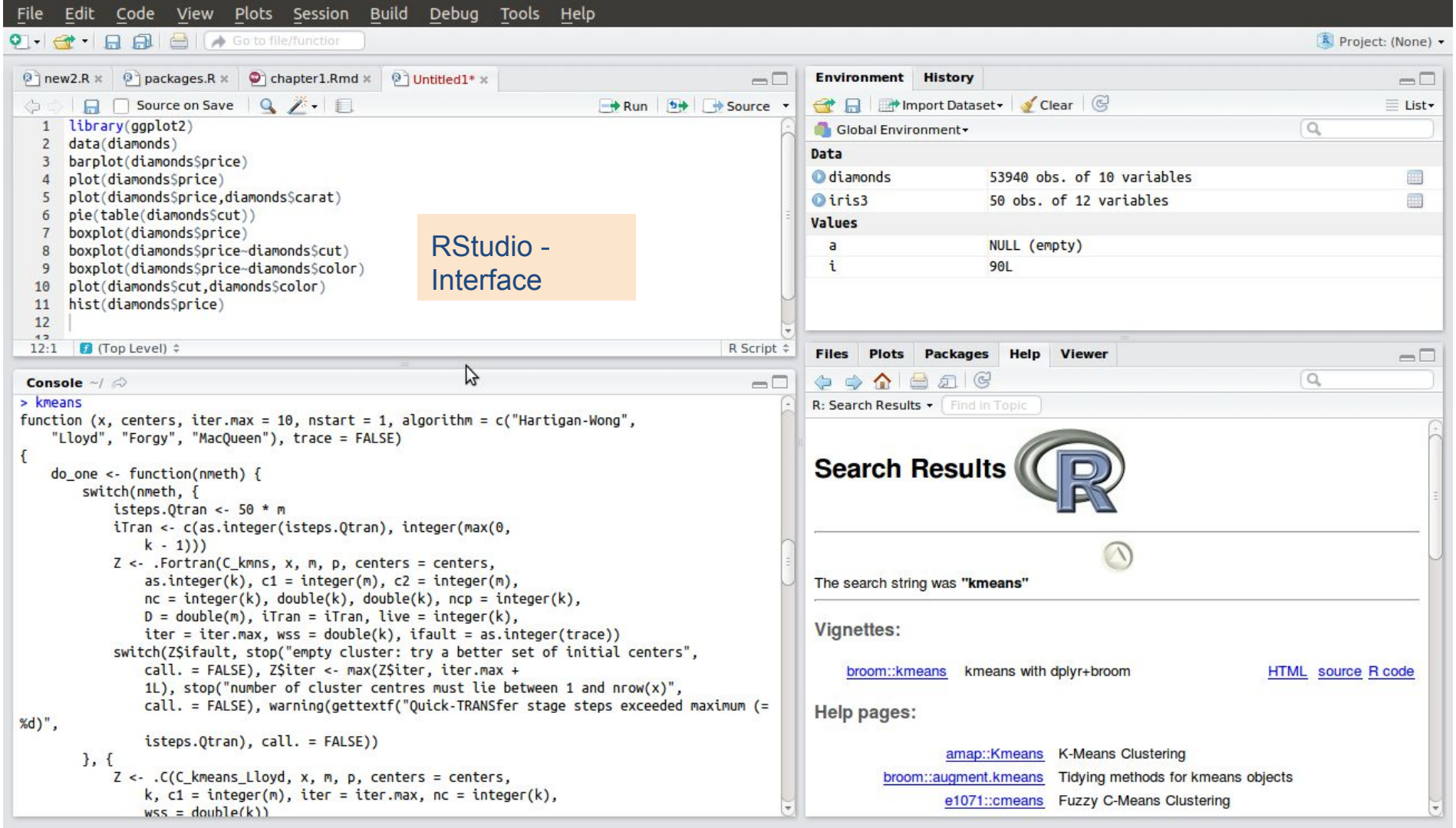
- Syntax highlighting, code completion, and smart indentation
- Execute R code directly from the source editor
- Quickly jump to function definitions
- Easily manage multiple working directories using projects
- Integrated R help and documentation
- Interactive debugger to diagnose and fix errors quickly
- Extensive package development tools

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

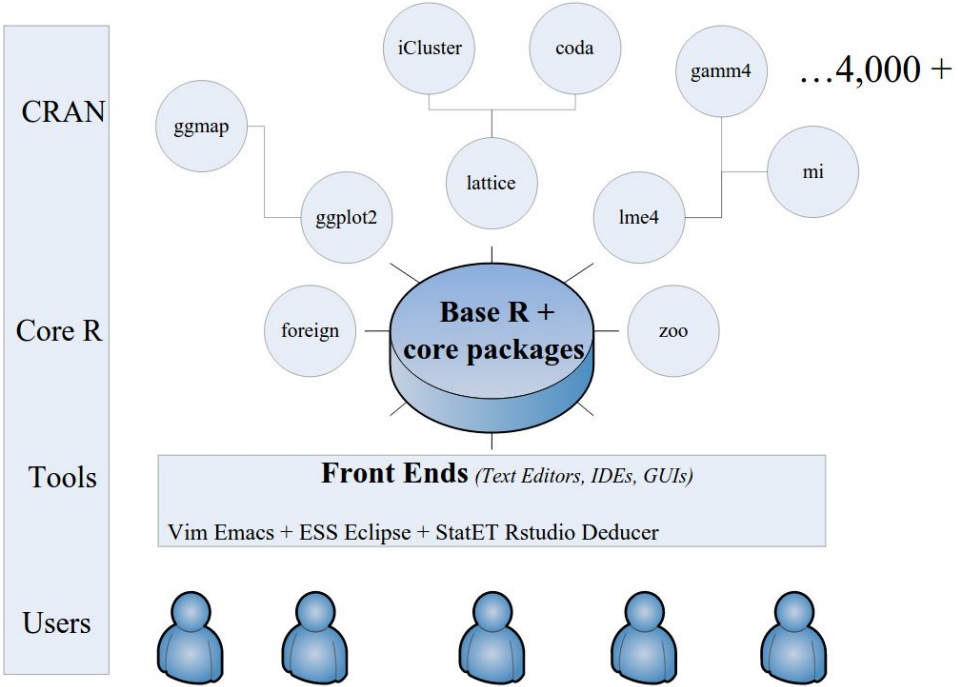
RStudio

RStudio Server enables you to provide a browser based interface (the RStudio IDE) to a version of R running on a remote Linux server. Deploying R and RStudio on a server has a number of benefits, including:

- The ability to access your R workspace from any computer in any location;
- Easy sharing of code, data, and other files with colleagues;
- Allowing multiple users to share access to the more powerful compute resources (memory, processors, etc.) available on a well equipped server; and
- Centralized installation and configuration of R, R packages, TeX, and other supporting libraries.


File Edit Code View Plots Session Build Debug Tools Help
Project: (None)
new2.R
packages.R
chapter1.Rmd
Untitled1*
Source on Save
Run
Source
1 library(ggplot2)
2 data(diamonds)
3 barplot(diamonds$price)
4 plot(diamonds$price)
5 plot(diamonds$price,diamonds$carat)
6 pie(table(diamonds$cut))
7 boxplot(diamonds$price)
8 boxplot(diamonds$price~diamonds$cut)
9 boxplot(diamonds$price~diamonds$color)
10 plot(diamonds$cut,diamonds$color)
11 hist(diamonds$price)
12:1
(Top Level)
R Script
RStudio - Interface
Environment
History
Import Dataset
Clear
List
Global Environment
Data
diamonds 53940 obs. of 10 variables
iris3 50 obs. of 12 variables
Values
a NULL (empty)
i 90L
Files
Plots
Packages
Help
Viewer
R: Search Results
Find in Topic
Search Results
The search string was "kmeans"
Vignettes:
broom::kmeans kmeans with dplyr+broom
HTML source R code
Help pages:
amap::Kmeans K-Means Clustering
broom::augment.kmeans Tidying methods for kmeans objects
e1071::cmeans Fuzzy C-Means Clustering
Console ~/
> kmeans
function (x, centers, iter.max = 10, nstart = 1, algorithm = c("Hartigan-Wong",
"Lloyd", "Forgy", "MacQueen"), trace = FALSE)
{
do_one <- function(nmeth) {
switch(nmeth, {
isteps.Qtran <- 50 * m
iTran <- c(as.integer(isteps.Qtran), integer(max(0,
k - 1)))
Z <- .Fortran(C_kmns, x, m, p, centers = centers,
as.integer(k), c1 = integer(m), c2 = integer(m),
nc = integer(k), double(k), double(k), ncp = integer(k),
D = double(m), iTran = iTran, live = integer(k),
iter = iter.max, wss = double(k), ifault = as.integer(trace))
switch(Z$ifault, stop("empty cluster: try a better set of initial centers",
call. = FALSE), Z$iter <- max(Z$iter, iter.max +
1L), stop("number of cluster centres must lie between 1 and nrow(x)",
call. = FALSE), warning(gettextf("Quick-TRANSfer stage steps exceeded maximum (=
%d)",
isteps.Qtran), call. = FALSE))
}, {
Z <- .C(C_kmeans_Lloyd, x, m, p, centers = centers,
k, c1 = integer(m), iter = iter.max, nc = integer(k),
wss = double(k))

R Landscape


CRAN
Core R
Tools
Users
iCluster
coda
gamm4
…4,000 +
ggmap
lattice
mi
ggplot2
lme4
foreign
Base R +
core packages
zoo
Front Ends (Text Editors, IDEs, GUIs)
Vim Emacs + ESS Eclipse + StatET Rstudio Deducer

R Documentation

http://cran.r-project.org/manuals.html

Manuals

edited by the R Development Core Team.

The following manuals for R were created on Debian Linux and may differ from the manuals for Mac or Windows on platf
version of the manuals for each platform are part of the respective R installations. The manuals change with R, hence we pr
version for the patched release version (R-patched) and finally a version for the forthcoming R version that is still in develo

Here they can be downloaded as PDF files, EPUB files, or directly browsed as HTML:

Manual	R-release
An Introduction to R is based on the former "Notes on R", gives an introduction to the language and how to use R for doing statistical analysis and graphics.	HTML | PDF | EPUB
R Data Import/Export describes the import and export facilities available either in R itself or via packages which are available from CRAN.	HTML | PDF | EPUB
R Installation and Administration	HTML | PDF | EPUB
Writing R Extensions covers how to create your own packages, write R help files, and the foreign language (C, C++, Fortran, ...) interfaces.	HTML | PDF | EPUB
A draft of **The R language definition** documents the language *per se*. That is, the objects that it works on, and the details of the expression evaluation process, which are useful to know when programming R functions.	HTML | PDF | EPUB
R Internals: a guide to the internal structures of R and coding standards for the core team working on R itself.	HTML | PDF | EPUB
The R Reference Index: contains all help files of the R standard and recommended packages in printable form. (9MB, approx. 3500 pages)	PDF

Translations of manuals into other languages than English are available from the contributed documentation section (only a

The LaTeX or Texinfo sources of the latest version of these documents are contained in every R source distribution (in the
be found in the respective archives of the R sources. The HTML versions of the manuals are also part of most R installation

Please check the manuals for R-devel before reporting any issues with the released versions.

R Documentation

Vignettes

ggplot2: An Implementation of the Grammar of Graphics

An implementation of the grammar of graphics in R. It combines the advantages of both base and lattice graph
by step from multiple data sources. It also implements a sophisticated multidimensional conditioning system a
information, documentation and examples.

Version:	1.0.1
Depends:	R (≥ 2.14), stats, methods
Imports:	plyr (≥ 1.7.1), digest, grid, gtable (≥ 0.1.1), reshape2, scales (≥ 0.2.3), proto, MASS
Suggests:	quantreg, Hmisc, mapproj, maps, hexbin, maptools, multcomp, nlme, testthat, knitr, mgcv
Enhances:	sp
Published:	2015-03-17
Author:	Hadley Wickham [aut, cre], Winston Chang [aut]
Maintainer:	Hadley Wickham <h.wickham at gmail.com>
BugReports:	https://github.com/hadley/ggplot2/issues
License:	GPL-2
URL:	http://ggplot2.org, https://github.com/hadley/ggplot2
NeedsCompilation:	no
Citation:	ggplot2 citation info
Materials:	README NEWS
In views:	Graphics, Phylogenetics
CRAN checks:	ggplot2 results

Downloads:

Reference manual:	ggplot2.pdf
Vignettes:	Contributing to ggplot2 development ggplot2 release process
Package source:	ggplot2_1.0.1.tar.gz
Windows binaries:	r-devel: ggplot2_1.0.1.zip, r-release: ggplot2_1.0.1.zip, r-oldrel: ggplot2_1.0.1
OS X Snow Leopard binaries:	r-release: not available, r-oldrel: ggplot2_1.0.1.tgz
OS X Mavericks binaries:	r-release: ggplot2_1.0.1.tgz
Old sources:	ggplot2 archive

Reverse dependencies:

Reverse depends:	alphahull, AmpliconDuo, aoristic, apsimr, bcrm, bde, benchmark, biomod2, bootnet, brms

CRAN Views

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

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

R Community

- email groups http://www.r-project.org/mail.html

R-announce

R-help

R-package-devel

R-devel

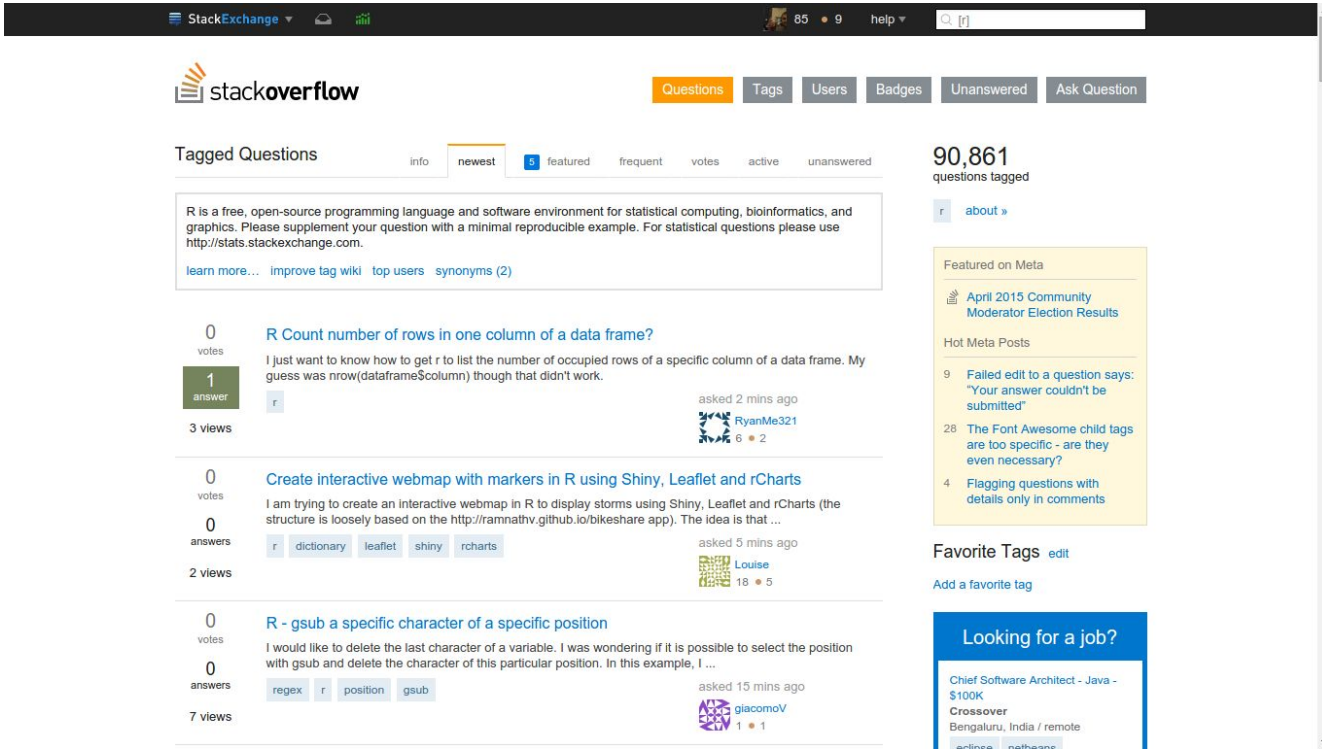
R-packages

Special Interest Groups

- Stack Overflow [r]
- Twitter #rstats
- Blogs at http://www.r-bloggers.com/ (573 blogs)

Stack Overflow

http://stackoverflow.com/questions/tagged/r



Twitter

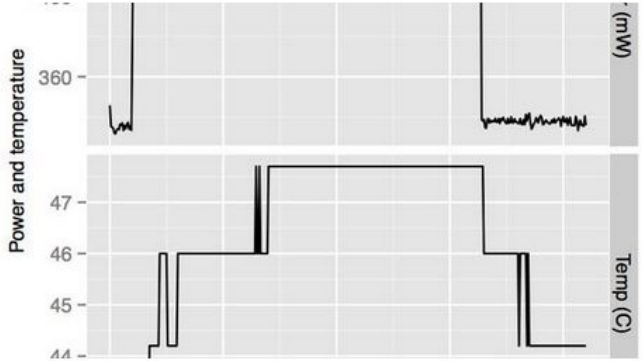
https://twitter.com/search?q=rstats&src=sprv

#rstats

Results for **#rstats**

Top / **All**

Mark Benson @markbenson · 5m
Power and heat are related. Here's an R plot I did that proves it on the Kindle Fire. **#rstats** vanilladraft.com/stmes/



View photo

Stéphane Fréchette @sfrechette · 8m
How to get your very own RStudio Server and Shiny Server with DigitalOcean r-bloggers.com/how-to-get-you… #datascience #feedly **#rstats** #shiny

Ankit kansal @sinisterankit · 9m
Interesting post on configuring parallel computing on #r #rstudio **#rstats** #dataprocessing #data

> **Learn R** @R_Programming
> How to do parallel computing with R? rstatistics.net/parallel-compu… #rstats #datascience

Help within R

?”keyword”

??”keyword”

Example-

```
> ?kmeans
```

```
> ??kmeans
```

Functions Used in this Lesson

function(x)

for

library

install.packages

update.packages

ls

rm

print

Citations and References

```
> citation()

To cite R in publications use:

  R Core Team (2015). R: A language and environment for statistical computing. R Foundation for
Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.
```

Introductory R

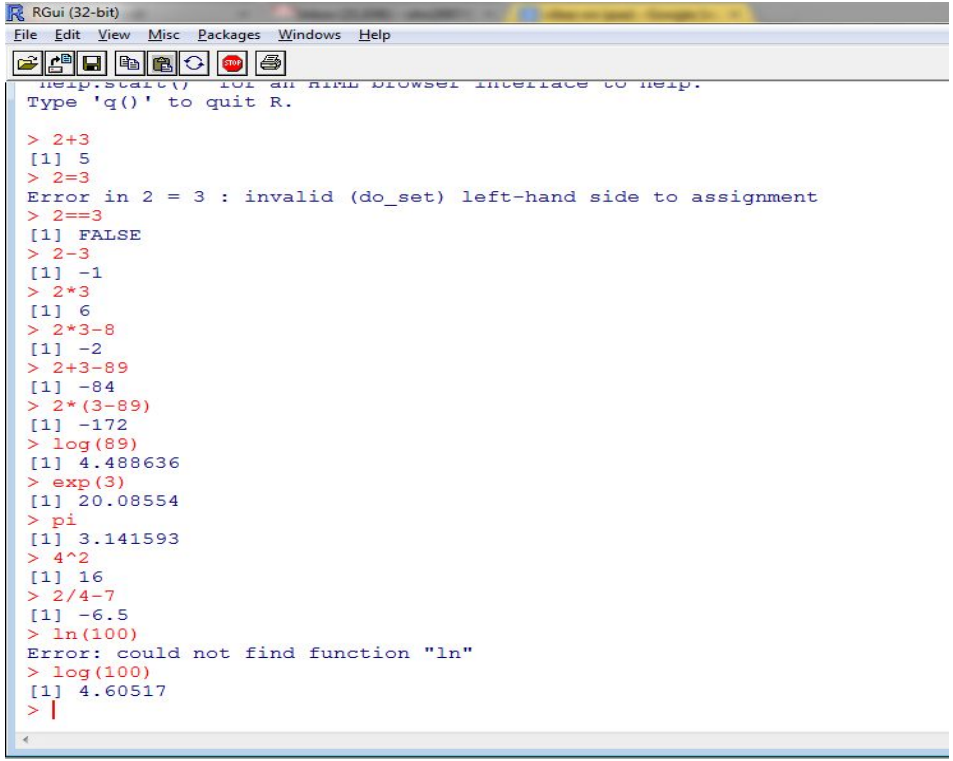
```
> Sys.Date()
[1] "2015-05-10"
> Sys.time()
[1] "2015-05-10 18:28:32 IST"
```

R as a Calculator
Basic Math on R Console

- +
- -
- Log
- Exp
- *
- /
- ()

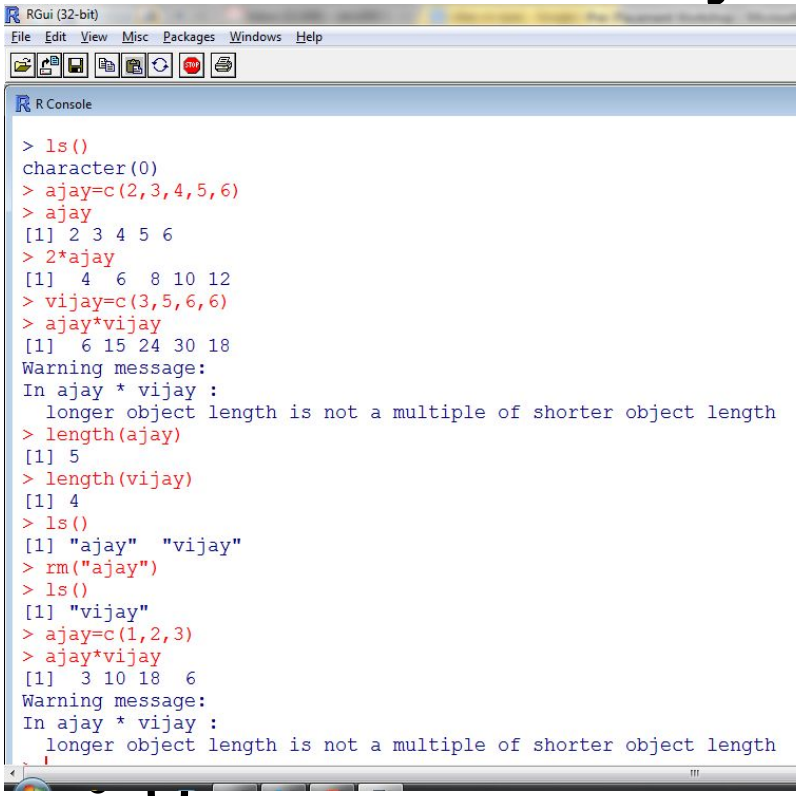
- mean
- sum
- sd
- log
- median
- exp

Demo-
Basic Math on R Console



Hint- Ctrl +L clears screen

Demo-
Basic Objects on R Console



Functions-
ls() – what objects are here
rm(“foo”) removes object named foo

Assignment
Using = or -> assigns object names to values

Hint- Up arrow ↑ gives you last typed command

Functions and Loops

- Loops

for (number in 1:5){ print (number) }

```
> for (number in 1:5){ print (number) }
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
> for (i in 1:5){ print (i) }
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
> for (i in 1:5){ rnorm(i,10,10) }
> for (i in 1:5){ print(rnorm(i,10,10)) }
[1] 1.090406
[1]  8.611727 16.670168
[1] 10.84623 13.13938 11.56230
[1]   6.068250 -18.723389  33.174107  -1.320091
[1] 13.939702 -9.037375 13.755986  9.459680  9.625309
> 
```

Functions and Loops

- Function

functionajay=function(a)(a^2+2*a+1)

```
> functionajay=function(a)(a^2+2*a+1)
> functionajay(1)
[1] 4
> functionajay(2)
[1] 9
> for (i in 1:5){ print(rnorm(i) }
Error: unexpected '}' in "for (i in 1:5){ print(rnorm(i) }"
>
> for (i in 1:5){ print(functionajay(i)) }
[1] 4
[1] 9
[1] 16
[1] 25
[1] 36
>
```

Hint: Always match brackets

Each (deserves a)

Each { deserves a }

Each [deserves a]

Other sources to learn R

swirlstats

http://swirlstats.com/

datacamp

https://www.datacamp.com/

codeschool

http://tryr.codeschool.com/

coursera

https://www.coursera.org/course/compdata

https://www.coursera.org/course/rprog

Good coding practices

- Use # for comment
- Use git for version control
- Use Rstudio for multiple lines of code

Functions in R

- custom functions
- source code for a function

```
Console ~/
> kmeans
function (x, centers, iter.max = 10, nstart = 1, algorithm = c("Hartigan-Wong",
    "Lloyd", "Forgy", "MacQueen"), trace = FALSE)
{
    do_one <- function(nmeth) {
        switch(nmeth, {
            isteps.Qtran <- 50 * m
            iTran <- c(as.integer(isteps.Qtran), integer(max(0,
                k - 1)))
            Z <- .Fortran(C_kmns, x, m, p, centers = centers,
                as.integer(k), c1 = integer(m), c2 = integer(m),
                nc = integer(k), double(k), double(k), ncp = integer(k),
                D = double(m), iTran = iTran, live = integer(k),
                iter = iter.max, wss = double(k), ifault = as.integer(trace))
            switch(Z$ifault, stop("empty cluster: try a better set of initial centers",
                call. = FALSE), Z$iter <- max(Z$iter, iter.max +
                1L), stop("number of cluster centres must lie between 1 and nrow(x)",
                call. = FALSE), warning(gettextf("Quick-TRANSfer stage steps exceeded maximum (=
%d)",
                isteps.Qtran), call. = FALSE))
        }, {
            Z <- .C(C_kmeans_Lloyd, x, m, p, centers = centers,
                k, c1 = integer(m), iter = iter.max, nc = integer(k),
                wss = double(k))
        }, {
            Z <- .C(C_kmeans_MacQueen, x, m, p, centers = as.double(centers),
                k, c1 = integer(m), iter = iter.max, nc = integer(k),
                wss = double(k))
        })
        if (m23 <- any(nmeth == c(2L, 3L))) {
            if (any(Z$nc == 0))
                warning("empty cluster: try a better set of initial centers",
                  call. = FALSE)
```

HOMEWORK TIME !

Learning Objectives

- how to input data in R using various ways
- how to check for correct data input
- how to use special packages for fast data input
- how to input data from statistical file formats
- how to input data from databases
- how to input data from web (web scraping)

What will you learn from this lesson

- data input from various kinds of format
- efficient data input via various packages
- sql to R
- web scraping
- piping in R
- using json in R

Environment

ls() -lists objects

rm()-removes an object

gc() -does garbage collection and frees up memory

File Edit Code View Plots Session Build Debug Tools Help

Project: (None)

Console ~/

```
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: i686-pc-linux-gnu (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.

  Natural language support but running in an English locale

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.

[Workspace loaded from ~/.RData]

> ls()
[1] "a"     "i"     "iris3"
> rm(a)
> ls()
[1] "i"     "iris3"
> gc()
         used (Mb) gc trigger (Mb) max used (Mb)
Ncells 334887  9.0     597831 16.0   407500 10.9
Vcells 624499  4.8    1215808  9.3  1215802  9.3
>
```

Environment | **History**

Import Dataset | Clear | List

Global Environment

Data	
iris3	50 obs. of 12 variables
Values	
i	90L

Files | **Plots** | **Packages** | **Help** | **Viewer**

R: Search Results | Find in Topic

Search Results

The search string was **"kmeans"**

Vignettes:

broom::kmeans kmeans with dplyr+broom — HTML source R code

Help pages:

amap::Kmeans K-Means Clustering
broom::augment.kmeans Tidying methods for kmeans objects
e1071::cmeans Fuzzy C-Means Clustering

File Edit Code View Plots Session Build Debug Tools Help

Go to file/function

Project: (None)

Console ~/

```
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: i686-pc-linux-gnu (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.

  Natural language support but running in an English locale

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.

[Workspace loaded from ~/.RData]

> ls()
[1] "a"     "i"     "iris3"
> rm(a)
> ls()
[1] "i"     "iris3"
> gc()
         used (Mb) gc trigger (Mb) max used (Mb)
Ncells 334887  9.0     597831 16.0   407500 10.9
Vcells 624499  4.8    1215808  9.3  1215802  9.3
>
```

Environment | **History**

Import Dataset | Clear | List

Global Environment

Data	
iris3	50 obs. of 12 variables
Values	
i	90L

Files | **Plots** | **Packages** | **Help** | **Viewer**

R: Search Results | Find in Topic

Search Results

The search string was **"kmeans"**

Vignettes:

broom::kmeans kmeans with dplyr+broom — HTML source R code

Help pages:

amap::Kmeans K-Means Clustering
broom::augment.kmeans Tidying methods for kmeans objects
e1071::cmeans Fuzzy C-Means Clustering

File System

getwd()- get working directory

setwd()- set or change working directory

dir() - lists files in working directory

File Edit Code View Plots Session Build Debug Tools Help

Go to file/function

Project: (None)

Console ~/Desktop/new/

```
> getwd()
[1] "/home/ajay/Desktop"
> setwd("/home/ajay/Desktop/new")
> dir()
[1] "obama"
>
```

Environment | **History**

Import Dataset | Clear | List

Global Environment

Data

iris3	50 obs. of 12 variables

Values

i	90L

Files | **Plots** | **Packages** | **Help** | **Viewer**

New Folder | Delete | Rename | More

Home

Name	Size	Modified
.RData	3.8 KB	May 2, 2015, 11:47 AM
.Rhistory	10.7 KB	May 10, 2015, 2:20 PM
17811636-Brain-function-as-gears-and-cogs-in-the-shape-of-a-human-head-as-a-medical-symbol-of-mental-health-c-Stock-Photo.jpg	139.8 KB	Apr 16, 2015, 9:41 AM
21.png	352.7 KB	May 4, 2015, 5:18 PM
2167434.jpg	32.8 KB	May 4, 2015, 5:36 PM
a.out	7.7 KB	May 2, 2015, 2:26 PM
anaconda		
animation		
animation2		
backports-3.18.1-1		
backports-3.18.1-1.tar.xz	8.6 MB	Dec 22, 2014, 3:14 AM
Call R and Python from base SAS.html	48.4 KB	May 5, 2015, 12:53 PM

File Edit Code View Plots Session Build Debug Tools Help

Go to file/function

Project: (None)

Console ~/Desktop/new/

```
> getwd()
[1] "/home/ajay/Desktop"
> setwd("/home/ajay/Desktop/new")
> dir()
[1] "obama"
>
```

Environment | **History**

Import Dataset | Clear | List

Global Environment

Data

iris3	50 obs. of 12 variables

Values

i	90L

Files | **Plots** | **Packages** | **Help** | **Viewer**

New Folder | Delete | Rename | More

Home

	Name	Size	Modified
	.RData	3.8 KB	May 2, 2015, 11:47 AM
	.Rhistory	10.7 KB	May 10, 2015, 2:20 PM
	17811636-Brain-function-as-gears-and-cogs-in-the-shape-of-a-human-head-as-a-medical-symbol-of-mental-health-c-Stock-Photo.jpg	139.8 KB	Apr 16, 2015, 9:41 AM
	21.png	352.7 KB	May 4, 2015, 5:18 PM
	2167434.jpg	32.8 KB	May 4, 2015, 5:36 PM
	a.out	7.7 KB	May 2, 2015, 2:26 PM
	anaconda		
	animation		
	animation2		
	backports-3.18.1-1		
	backports-3.18.1-1.tar.xz	8.6 MB	Dec 22, 2014, 3:14 AM
	Call R and Python from base SAS.html	48.4 KB	May 5, 2015, 12:53 PM

File Edit Code View Plots Session Build Debug Tools Help

Go to file/function

Project: (None)

Console ~/Desktop/new/

```
> getwd()
[1] "/home/ajay/Desktop"
> setwd("/home/ajay/Desktop/new")
> dir()
[1] "obama"
>
```

Environment | **History**

Import Dataset | Clear | List

Global Environment

Data

iris3	50 obs. of 12 variables

Values

i	90L

Files | **Plots** | **Packages** | **Help** | **Viewer**

New Folder | Delete | Rename | More

Home

Name	Size	Modified
.RData	3.8 KB	May 2, 2015, 11:47 AM
.Rhistory	10.7 KB	May 10, 2015, 2:20 PM
17811636-Brain-function-as-gears-and-cogs-in-the-shape-of-a-human-head-as-a-medical-symbol-of-mental-health-c-Stock-Photo.jpg	139.8 KB	Apr 16, 2015, 9:41 AM
21.png	352.7 KB	May 4, 2015, 5:18 PM
2167434.jpg	32.8 KB	May 4, 2015, 5:36 PM
a.out	7.7 KB	May 2, 2015, 2:26 PM
anaconda		
animation		
animation2		
backports-3.18.1-1		
backports-3.18.1-1.tar.xz	8.6 MB	Dec 22, 2014, 3:14 AM
Call R and Python from base SAS.html	48.4 KB	May 5, 2015, 12:53 PM

Assigning

objectname=read.csv(filepath,parameters)

OR

objectname<-read.csv(filepath,parameters)

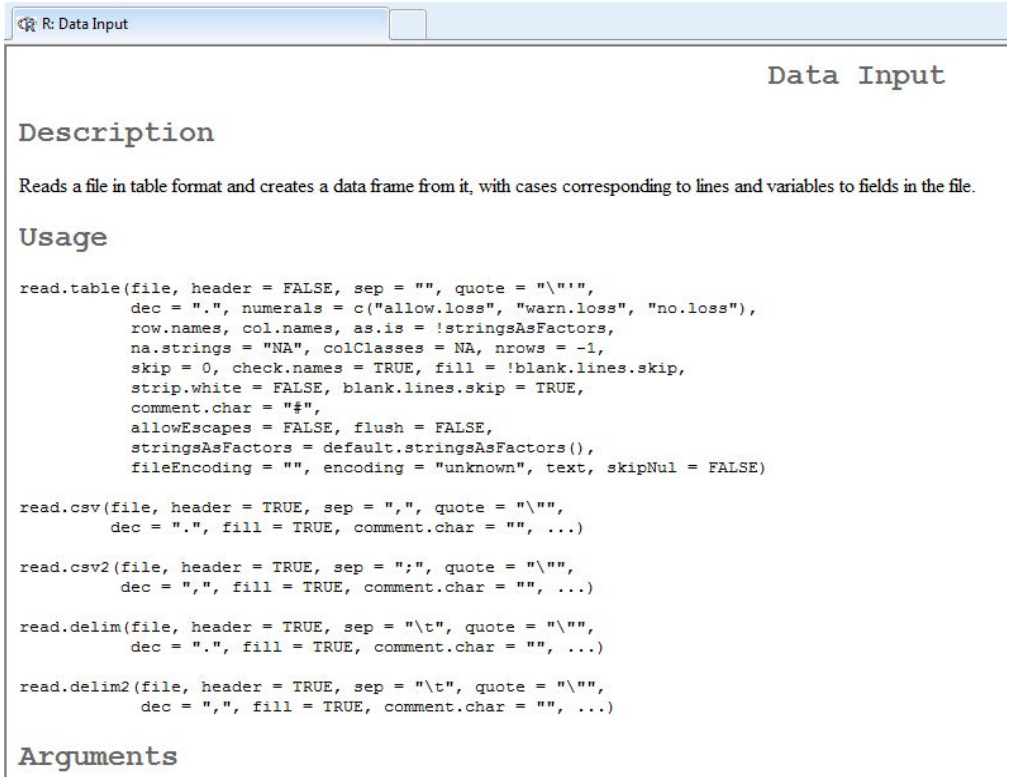
Data Input

read.table() or read.csv()

read.spss()

read.sas7bdat()

read.table()



R: Data Input

Data Input

Description

Reads a file in table format and creates a data frame from it, with cases corresponding to lines and variables to fields in the file.

Usage

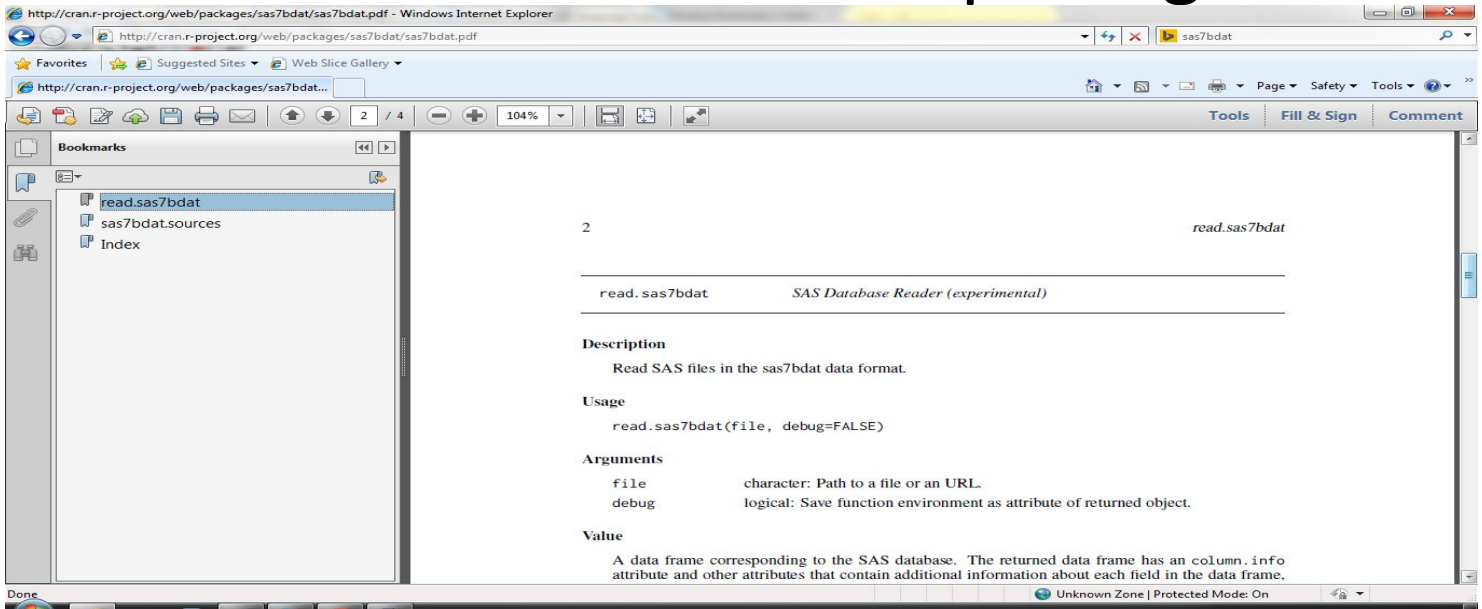
```
read.table(file, header = FALSE, sep = "", quote = "\"'",
           dec = ".", numerals = c("allow.loss", "warn.loss", "no.loss"),
           row.names, col.names, as.is = !stringsAsFactors,
           na.strings = "NA", colClasses = NA, nrows = -1,
           skip = 0, check.names = TRUE, fill = !blank.lines.skip,
           strip.white = FALSE, blank.lines.skip = TRUE,
           comment.char = "#",
           allowEscapes = FALSE, flush = FALSE,
           stringsAsFactors = default.stringsAsFactors(),
           fileEncoding = "", encoding = "unknown", text, skipNul = FALSE)

read.csv(file, header = TRUE, sep = ",", quote = "\"",
         dec = ".", fill = TRUE, comment.char = "", ...)

read.csv2(file, header = TRUE, sep = ";", quote = "\"",
          dec = ",", fill = TRUE, comment.char = "", ...)

read.delim(file, header = TRUE, sep = "\t", quote = "\"",
           dec = ".", fill = TRUE, comment.char = "", ...)

read.delim2(file, header = TRUE, sep = "\t", quote = "\"",
            dec = ",", fill = TRUE, comment.char = "", ...)
```

Arguments

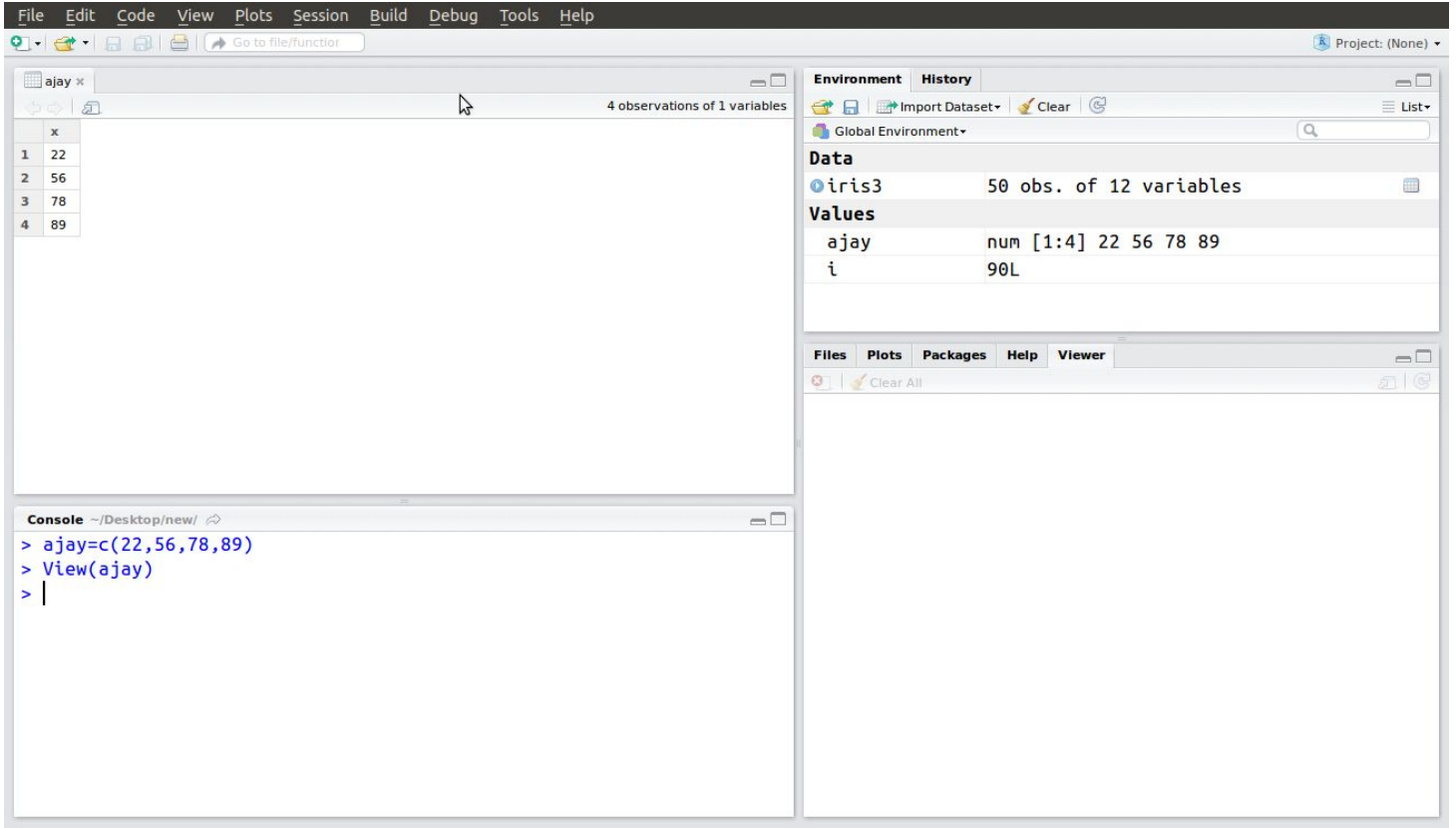
https://stat.ethz.ch/R-manual/R-devel/library/utils/html/read.table.html

Statistical formats

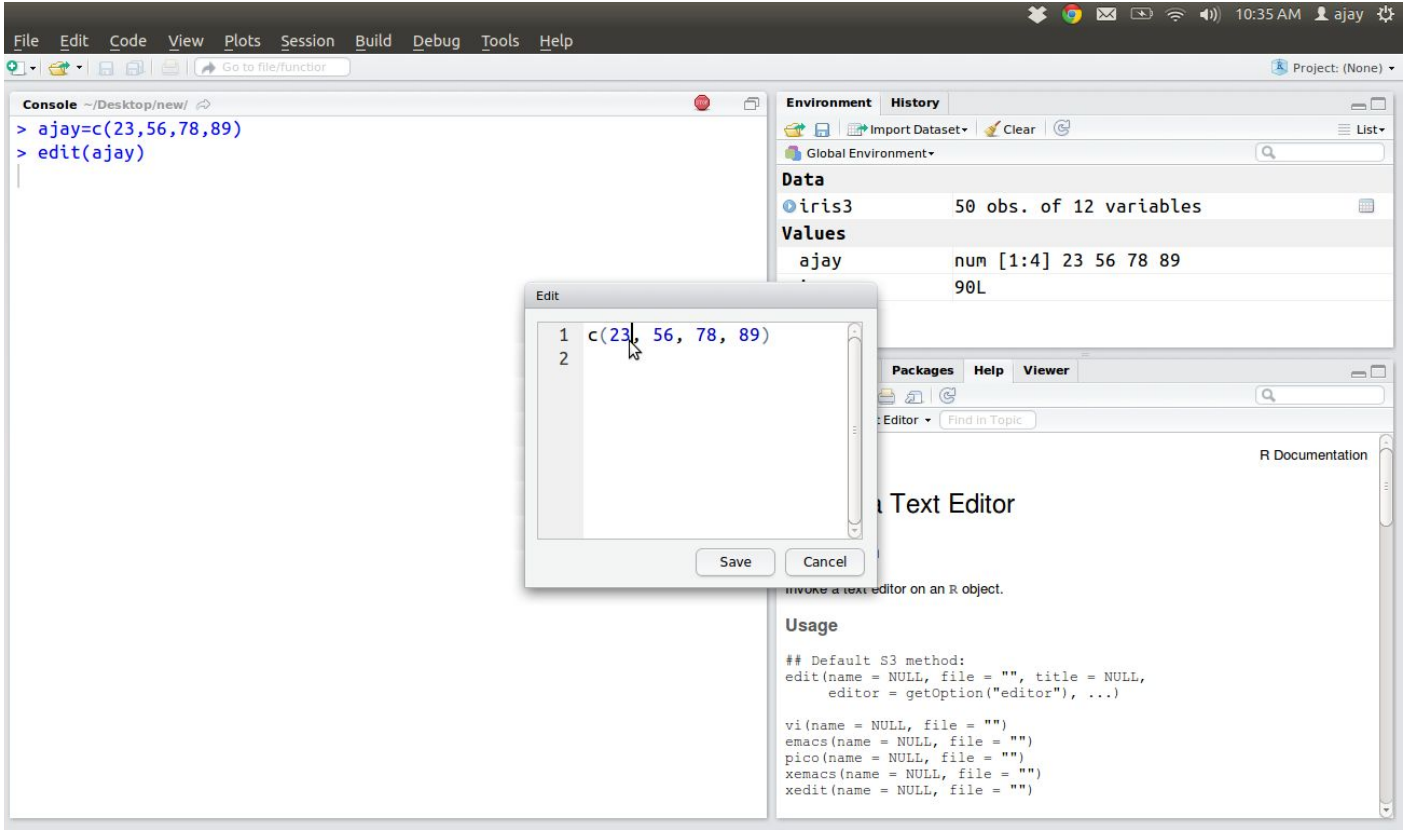
- read.spss from foreign package
- read.sas7bdat from sas7bdat package



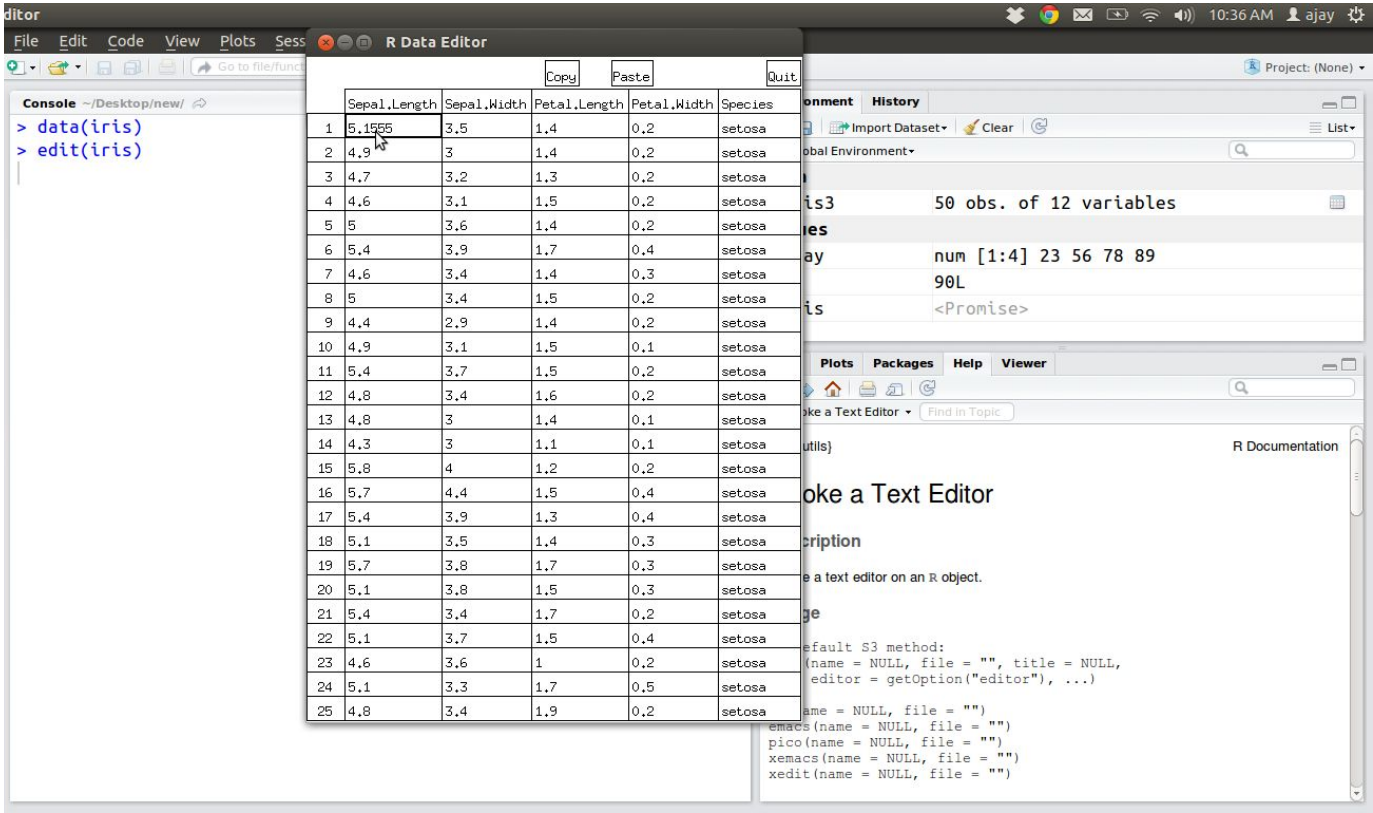
Manual Entry



Manual Editing



Manual Editing



readr from Hadley

The goal of readr is to provide a fast and friendly way to read tabular data into R. The most important functions are:

- Read delimited files: `read_delim()`, `read_csv()`, `read_tsv()`, `read_csv2()`.
- Read fixed width files: `read_fwf()`, `read_table()`.
- Read lines: `read_lines()`.
- Read whole file: `read_file()`.
- Re-parse existing data frame: `type_convert()`.

https://github.com/hadley/readr

readr from Hadley

Source Data - https://bit.ly/dsdata

```
> library(readr)
> system.time(read_csv("BigDiamonds.csv"))
|======================================================================================| 100%   49 MB
   user  system elapsed
  2.396   0.068   2.448
Warning message:
597311 problems parsing 'BigDiamonds.csv'. See problems(...) for more details.
```

readxl from Hadley

Readxl supports both the legacy `.xls` format and the modern xml-based `.xlsx` format. `.xls`support is made possible the with libxls C library, which abstracts away many of the complexities of the underlying binary format. To parse `.xlsx`, we use the RapidXML C++ library.

```
read_excel("my-old-spreadsheet.xls")
read_excel("my-new-spreadsheet.xlsx")

read_excel("my-spreadsheet.xls", sheet = "data")
read_excel("my-spreadsheet.xls", sheet = 2)

read_excel("my-spreadsheet.xls", na = "NA")
```

https://github.com/hadley/readxl

data.table

fread is the fastest way to read data

```
> b=fread("BigDiamonds.csv")
Read 598024 rows and 13 (of 13) columns from 0.049 GB file in 00:00:04
```

data.table

fread is the fastest way to read data

```
> b=fread("BigDiamonds.csv")
Read 598024 rows and 13 (of 13) columns from 0.049 GB file in 00:00:04
```

data.table

fread is the fastest way to read data

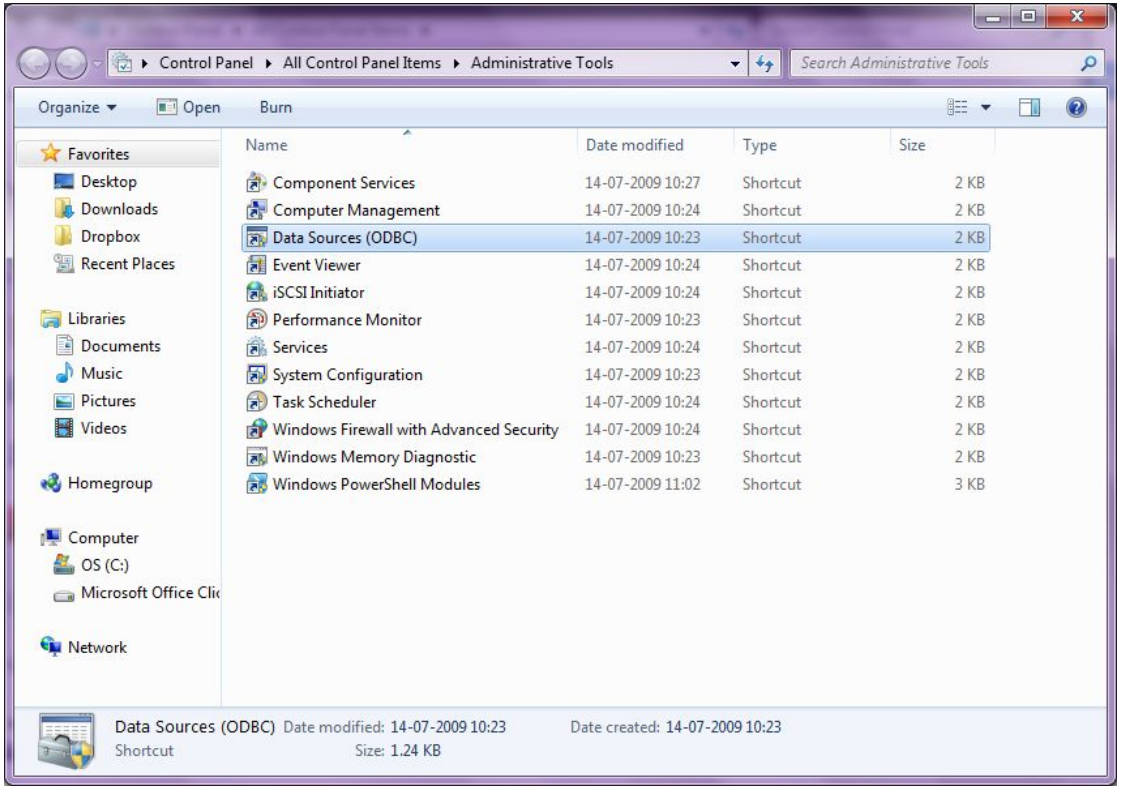
```
> system.time(read_csv("BigDiamonds.csv"))
   user  system elapsed
  2.552   0.028   2.581
Warning message:
597311 problems parsing 'BigDiamonds.csv'. See problems(...) for more details.
> system.time(fread("BigDiamonds.csv"))
   user  system elapsed
  1.532   0.012   1.540
> system.time(read.csv("BigDiamonds.csv"))
   user  system elapsed
 10.892   0.032  10.922
```

Some learnings

1. Multiple packages can do the same thing faster or slower in R
2. Knowing the right package is the essential difference as a data scientist
3. Putting code within system.time() helps measure speed

also see http://adv-r.had.co.nz/Profiling.html for advanced ways to spped up code

Creating DSN (Optional)



Creating DSN (in Windows)

A Data Source Name (DSN) is the logical name that is used by Open Database Connectivity (ODBC) to refer to the drive and other information that is required to access data. The name is use by Internet Information Services for a connection to an ODBC data source, such as a Microsoft SQL Server database.

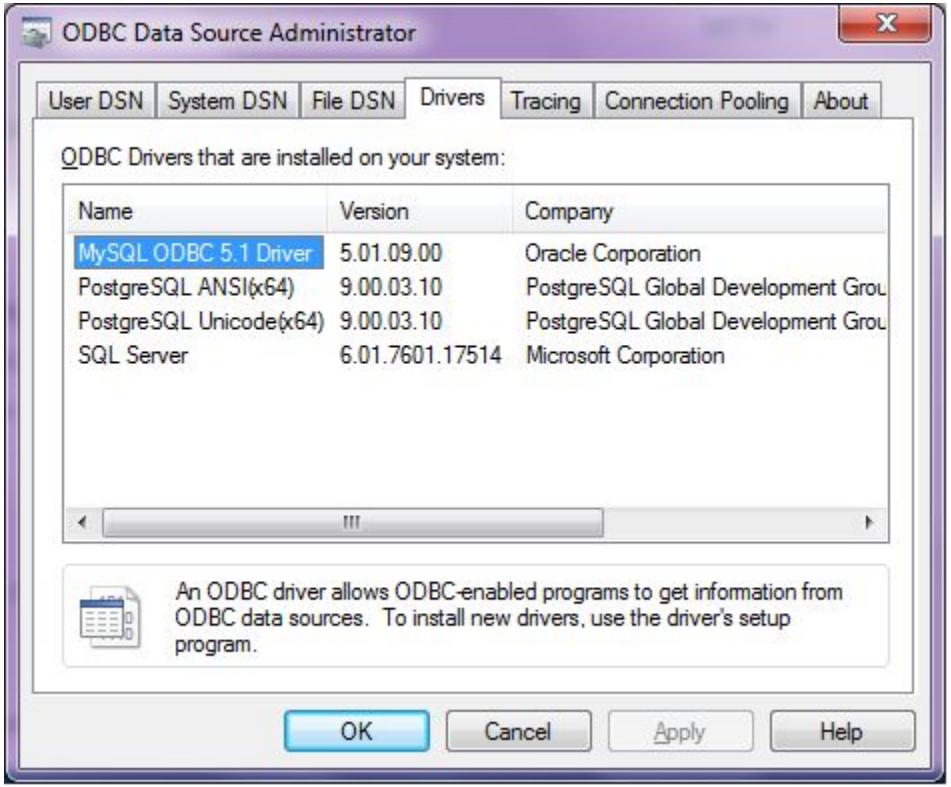
https://support.microsoft.com/en-us/kb/kbview/300596

Creating DSN (in Windows)

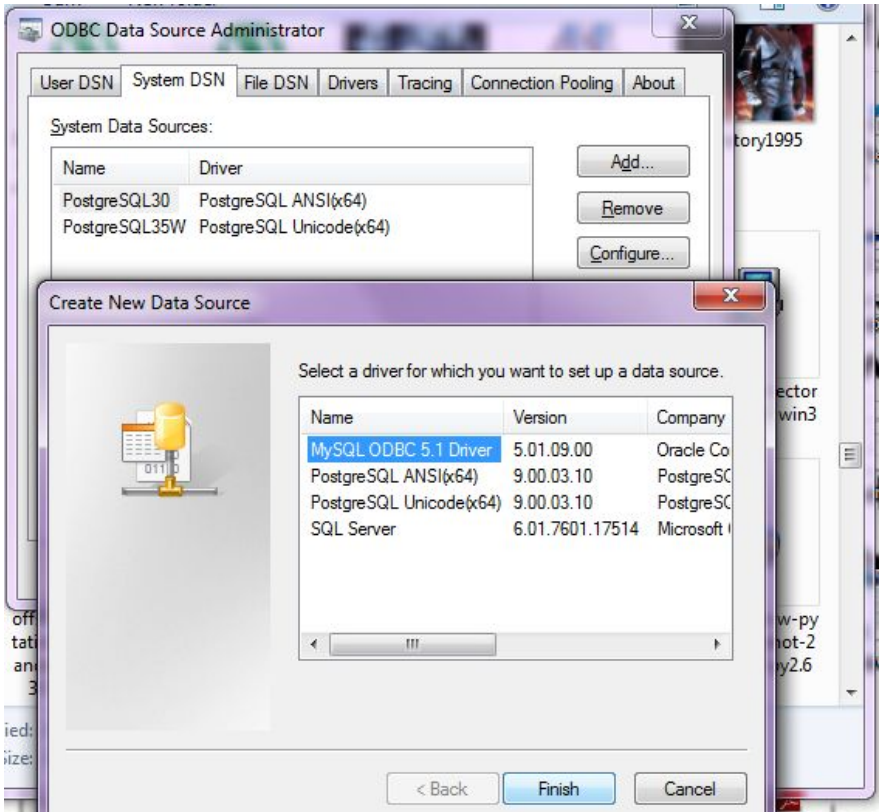
1. Click **Start**, point to **Control Panel**, double-click **Administrative Tools**, and then double-click **Data Sources(ODBC)**.
2. Click the **System DSN** tab, and then click **Add**.
3. Click the database driver that corresponds with the database type to which you are connecting, and then click **Finish**.
4. Type the data source name. Make sure that you choose a name that you can remember. You will need to use this name later.
5. Click **Select**.
6. Click the correct database, and then click **OK**.
7. Click **OK**, and then click **OK**.

https://support.microsoft.com/en-us/kb/kbview/300596

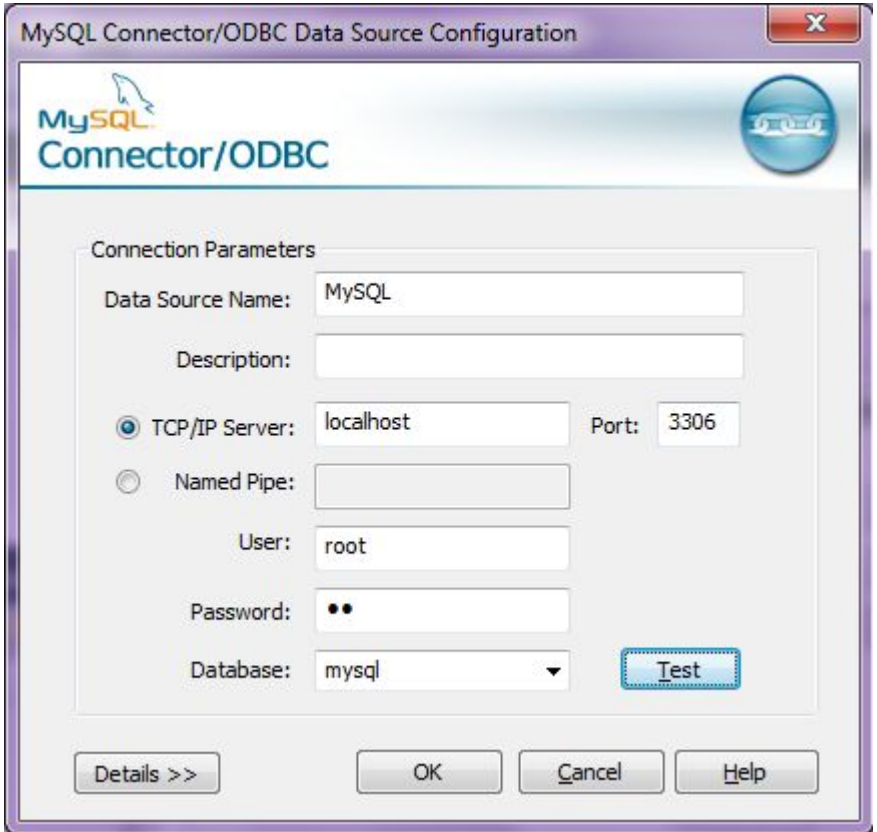
Creating DSN



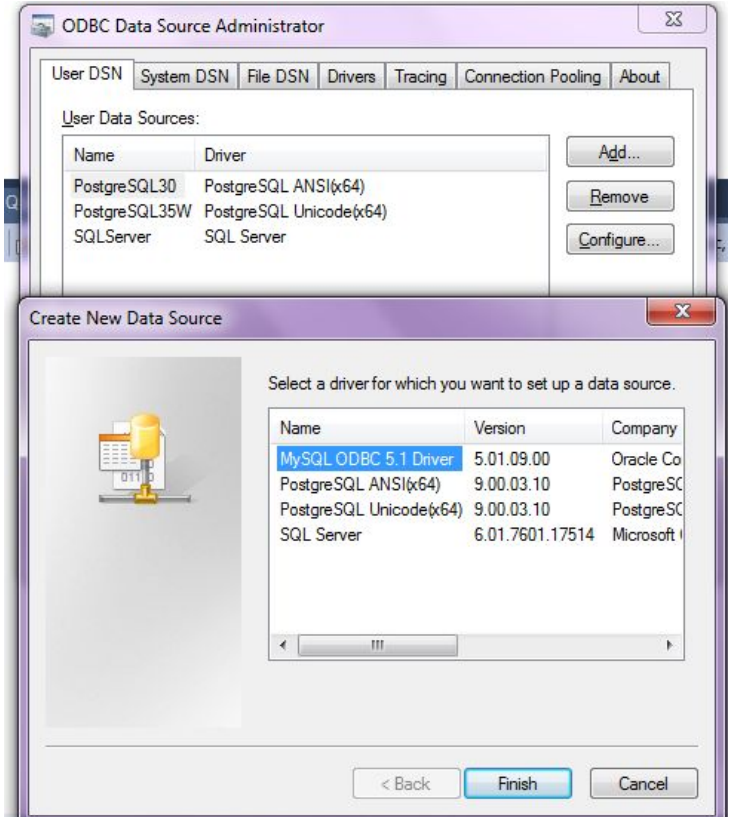
Creating DSN



Creating DSN



Creating DSN



RODBC

```
> library(RODBC)
> odbcDataSources()
> ajay=odbcConnect(“MySQL”,uid=”root”,pwd=”XX”)
> ajay
> sqlTables(ajay)
>tested=sqlFetch(ajay,”host”)
```

From Databases

The **RODBC** package provides access to databases through an **ODBC** interface.

The primary functions are

- **odbcConnect(*dsn*, uid="", pwd="")** Open a connection to an ODBC database
- **sqlFetch(*channel*, *sqltable*)** Read a table from an ODBC database into a data frame

Hint- a good site to revise R
http://www.statmethods.net

sqlite

http://cran.r-project.org/web/packages/RSQLite/RSQLite.pdf embeds the SQLite database engine in R and provides an interface compliant with the DBI package.

SQLite is a software library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine.

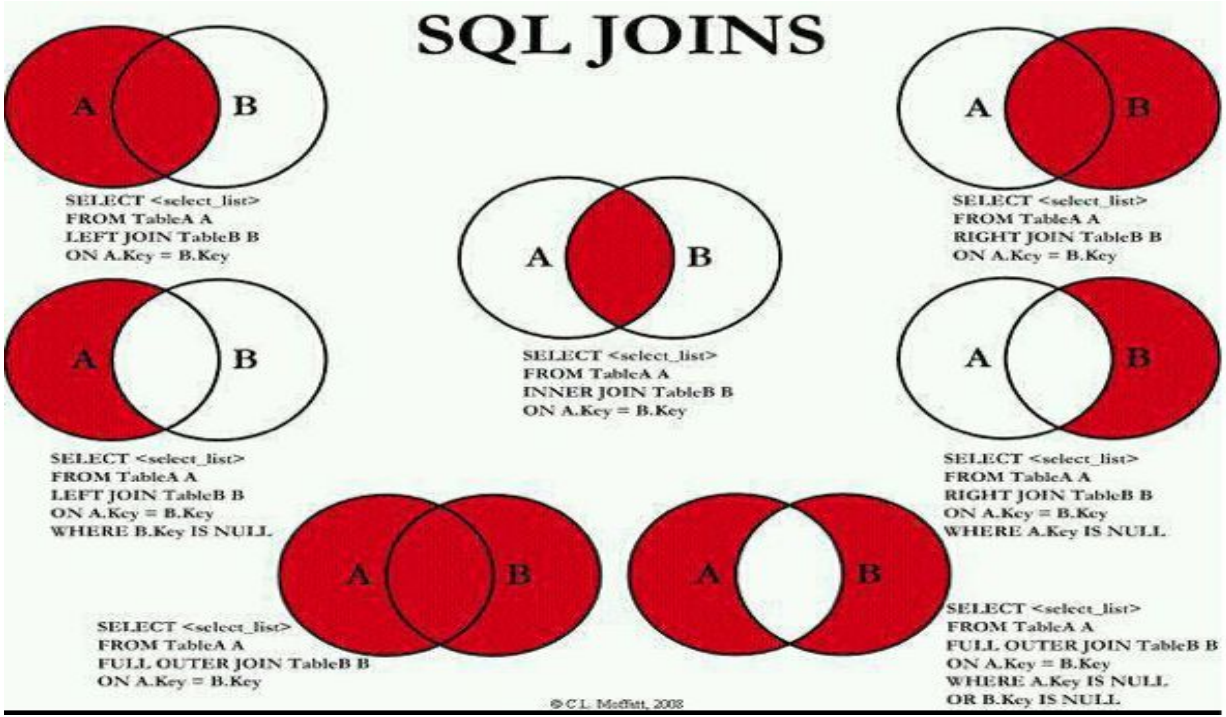
SQLite is the most widely deployed database engine in the world

```
library(RSQLite)
con <- dbConnect("SQLite", dbname = "sample_db")
 # read csv file into sql database
dbWriteTable(con, name="sample_data", value="sample_data.csv", row.names=FALSE, header=TRUE, sep = ",")
```

http://cran.r-project.org/web/packages/sqldf/index.html Manipulate R data frames using SQL

`read.csv.sql` in the `sqldf` package imports data into a temporary SQLite database and then reads it into R.

A Detour to SQL Joins (Optional)



RMySQL

```
install.packages("RMySQL")
library(RMySQL)

mydb = dbConnect(MySQL(), user='user', password='password', dbname='database_name', host='host')


dbListTables(mydb)

dbListFields(mydb, 'some_table')


dbSendQuery(mydb, 'drop table if exists some_table, some_other_table')

dbWriteTable(mydb, name='table_name', value=data.frame.name)
```

Other databases

Teradata https://github.com/Teradata/teradataR

PostgreSQL http://cran.r-project.org/web/packages/RPostgreSQL/

MongoDB http://cran.r-project.org/web/packages/mongolite/index.html

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

MonetDB http://cran.r-project.org/web/packages/MonetDB.R/index.html

Other data sources

Cassandra with R http://cran.r-project.org/web/packages/RCassandra/RCassandra.pdf

Neo4j with R

http://things-about-r.tumblr.com/post/47392314578/venue-recommendation-a-simple-use-case-connecting-r

R with Hadoop Stack https://github.com/RevolutionAnalytics/RHadoop/wiki

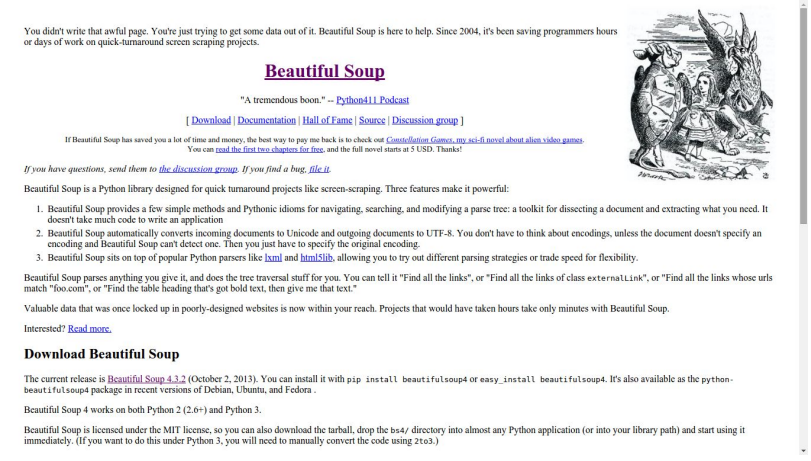
- NEW! ravro - read and write files in avro format
- plyrmr - higher level plyr-like data processing for structured data, powered by `rmr`
- rmr - functions providing Hadoop MapReduce functionality in R
- rhdfs - functions providing file management of the HDFS from within R
- rhbase - functions providing database management for the HBase distributed database from within R

https://amplab-extras.github.io/SparkR-pkg/ SparkR is an R package to use Spark from R.

Web Scraping

Web scraping (**web** harvesting or **web** data extraction) is a computer software technique of extracting information from websites.

example - python (scrapy and beautiful soup)



You didn't write that awful page. You're just trying to get some data out of it. Beautiful Soup is here to help. Since 2004, it's been saving programmers hours or days of work on quick-turnaround screen scraping projects.

Beautiful Soup

"A tremendous boon." -- Python411 Podcast

[Download | Documentation | Hall of Fame | Source | Discussion group]

If Beautiful Soup has saved you a lot of time and money, the best way to pay me back is to check out *Constellation Games*, my sci-fi novel about alien video games. You can read the first two chapters for free, and the full novel starts at 5 USD. Thanks!

If you have questions, send them to the discussion group. If you find a bug, file it.

Beautiful Soup is a Python library designed for quick turnaround projects like screen-scraping. Three features make it powerful:

1. Beautiful Soup provides a few simple methods and Pythonic idioms for navigating, searching, and modifying a parse tree: a toolkit for dissecting a document and extracting what you need. It doesn't take much code to write an application
2. Beautiful Soup automatically converts incoming documents to Unicode and outgoing documents to UTF-8. You don't have to think about encodings, unless the document doesn't specify an encoding and Beautiful Soup can't detect one. Then you just have to specify the original encoding.
3. Beautiful Soup sits on top of popular Python parsers like lxml and html5lib, allowing you to try out different parsing strategies or trade speed for flexibility.

Beautiful Soup parses anything you give it, and does the tree traversal stuff for you. You can tell it "Find all the links", or "Find all the links of class `externalLink`", or "Find all the links whose urls match "foo.com", or "Find the table heading that's got bold text, then give me that text."

Valuable data that was once locked up in poorly-designed websites is now within your reach. Projects that would have taken hours take only minutes with Beautiful Soup.

Interested? Read more.

Download Beautiful Soup

The current release is Beautiful Soup 4.3.2 (October 2, 2013). You can install it with `pip install beautifulsoup4` or `easy_install beautifulsoup4`. It's also available as the `python-beautifulsoup4` package in recent versions of Debian, Ubuntu, and Fedora .

Beautiful Soup 4 works on both Python 2 (2.6+) and Python 3.

Beautiful Soup is licensed under the MIT license, so you can also download the tarball, drop the `bs4/` directory into almost any Python application (or into your library path) and start using it immediately. (If you want to do this under Python 3, you will need to manually convert the code using `2to3`.)

Web Scraping

- readlines

Hint : R is case sensitive
readlines is not the same as readLines

```
> url="http://nytimes.com"
> ajay=readlines(url)
Error: could not find function "readlines"
> ajay=readLines(url)
> head(ajay)
[1] "<!DOCTYPE html>"                                                                                                  $
[2] "<!--[if (gt IE 9)|!(IE)]> <!--> <html lang=\"en\" class=\"no-js edition-domestic app-homepage\"  itemscope xmlns:og$
[3] "<!--[if IE 9]> <html lang=\"en\" class=\"no-js ie9 lt-ie10 edition-domestic app-homepage\" xmlns:og=\"http://opengr$
[4] "<!--[if IE 8]> <html lang=\"en\" class=\"no-js ie8 lt-ie10 lt-ie9 edition-domestic app-homepage\" xmlns:og=\"http:/$
[5] "<!--[if (lt IE 8)]> <html lang=\"en\" class=\"no-js lt-ie10 lt-ie9 lt-ie8 edition-domestic app-homepage\" xmlns:og=$
[6] "<head>"                                                                                                           $
> tail(ajay)
[1] "<div id=\"ab3\" class=\"ad ab3-ad hidden\"></div>"                                                                $
[2] "<div id=\"prop1\" class=\"ad prop1-ad hidden\"></div>"                                                            $
[3] "<div id=\"prop2\" class=\"ad prop2-ad hidden\"></div>"                                                            $
[4] "<div id=\"Anchor\" class=\"ad anchor-ad hidden\"></div>"                                                          $
[5] "<script type=\"text/javascript\">window.NREUM||(NREUM={});NREUM.info={\"beacon\":\"beacon-6.newrelic.com\",\"licens$
[6] "</html>"                                                                                                          $
> |
```

Hint : Use head() and tail() to inspect objects

Other packages are XML and Curl
Case Study- http://decisionstats.com/2013/04/14/using-r-for-cricket-analysis-rstats/

curl

cURL is a computer software project providing a library and command-line tool for transferring data using various protocols. The **cURL** project produces two products, libcurl and **cURL**.

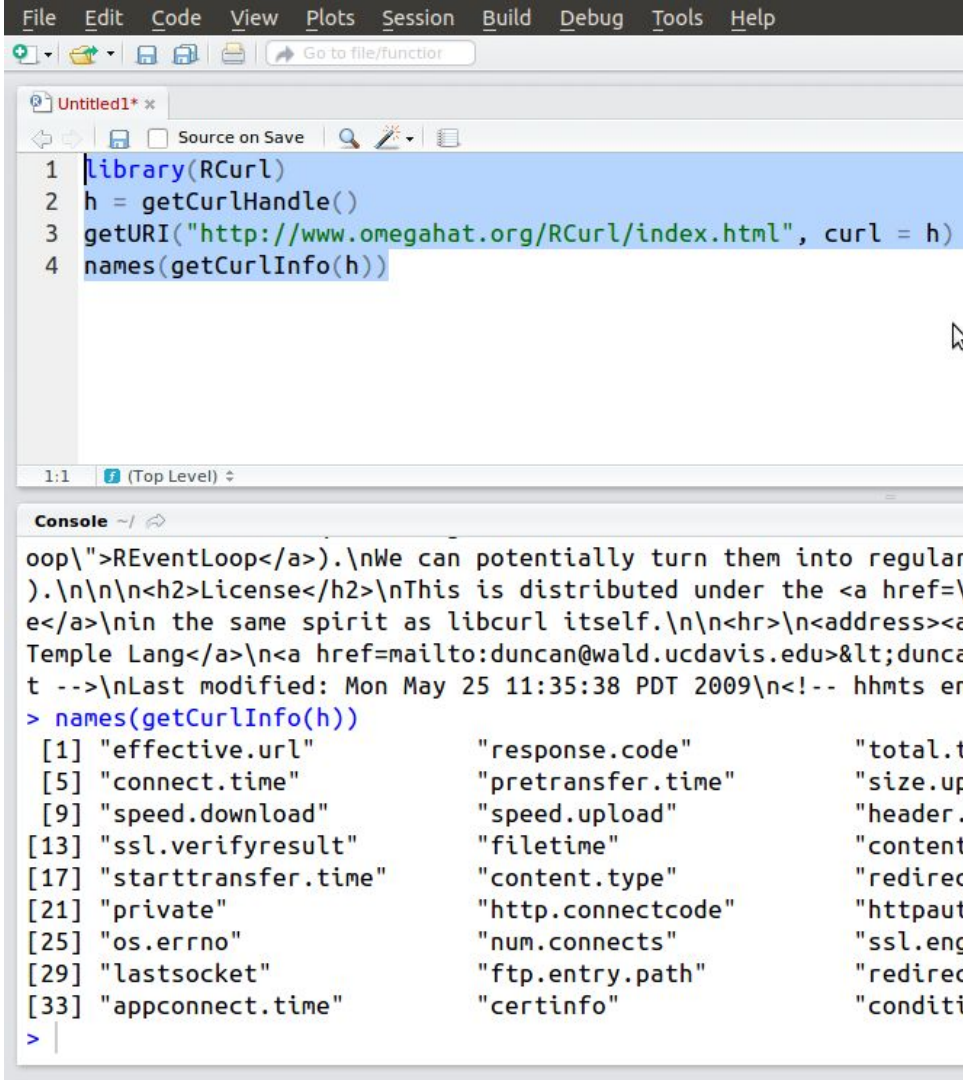
The RCurl package is an R-interface to the libcurl library that provides HTTP facilities. This allows us to download files from Web servers, post forms, use HTTPS (the secure HTTP), use persistent connections, upload files, use binary content, handle redirects, password authentication, etc.

The primary top-level entry points are

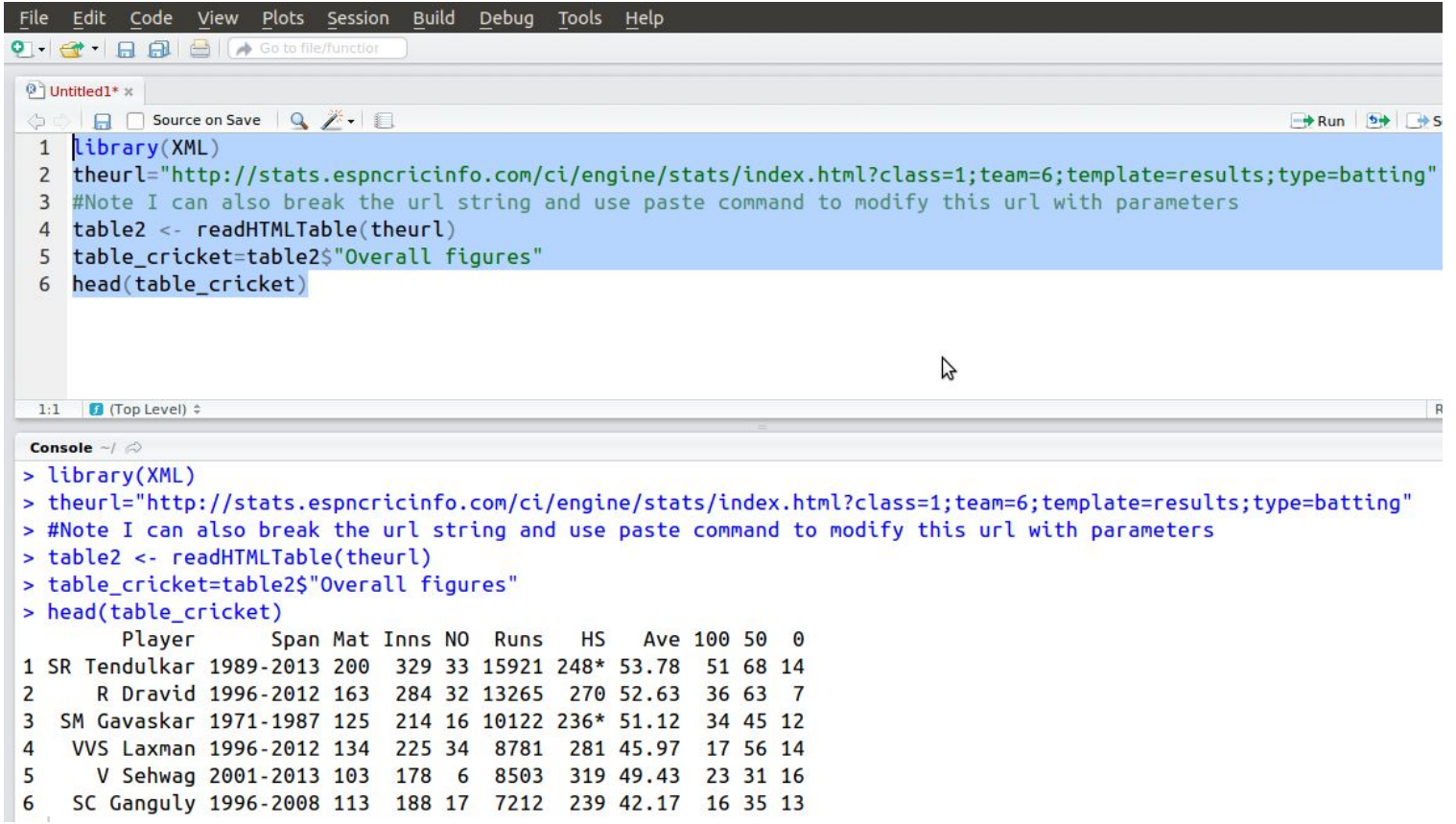
- getURL()
- getURLContent()
- getForm()
- postForm()

http://www.omegahat.org/RCurl/RCurlJSS.pdf

Rcurl


File Edit Code View Plots Session Build Debug Tools Help
Untitled1*
Source on Save
1 library(RCurl)
2 h = getCurlHandle()
3 getURI("http://www.omegahat.org/RCurl/index.html", curl = h)
4 names(getCurlInfo(h))
1:1 (Top Level)
Console ~/
oop\">REventLoop).\nWe can potentially turn them into regular
).\n\n\n<h2>License</h2>\nThis is distributed under the <a href=\
e\nin the same spirit as libcurl itself.\n\n<hr>\n<address><a
Temple Lang\n<a href=mailto:duncan@wald.ucdavis.edu><dunca
t -->\nLast modified: Mon May 25 11:35:38 PDT 2009\n<!-- hhmts en
> names(getCurlInfo(h))
[1] "effective.url" "response.code" "total.t
[5] "connect.time" "pretransfer.time" "size.up
[9] "speed.download" "speed.upload" "header.
[13] "ssl.verifyresult" "filetime" "content
[17] "starttransfer.time" "content.type" "redirec
[21] "private" "http.connectcode" "httpaut
[25] "os.errno" "num.connects" "ssl.eng
[29] "lastsocket" "ftp.entry.path" "redirec
[33] "appconnect.time" "certinfo" "conditi
>

XML


File Edit Code View Plots Session Build Debug Tools Help
Untitled1*
Source on Save
Run
1 library(XML)
2 theurl="http://stats.espncricinfo.com/ci/engine/stats/index.html?class=1;team=6;template=results;type=batting"
3 #Note I can also break the url string and use paste command to modify this url with parameters
4 table2 <- readHTMLTable(theurl)
5 table_cricket=table2$"Overall figures"
6 head(table_cricket)
1:1
(Top Level)
Console ~/
> library(XML)
> theurl="http://stats.espncricinfo.com/ci/engine/stats/index.html?class=1;team=6;template=results;type=batting"
> #Note I can also break the url string and use paste command to modify this url with parameters
> table2 <- readHTMLTable(theurl)
> table_cricket=table2$"Overall figures"
> head(table_cricket)
Player Span Mat Inns NO Runs HS Ave 100 50 0
1 SR Tendulkar 1989-2013 200 329 33 15921 248* 53.78 51 68 14
2 R Dravid 1996-2012 163 284 32 13265 270 52.63 36 63 7
3 SM Gavaskar 1971-1987 125 214 16 10122 236* 51.12 34 45 12
4 VVS Laxman 1996-2012 134 225 34 8781 281 45.97 17 56 14
5 V Sehwag 2001-2013 103 178 6 8503 319 49.43 23 31 16
6 SC Ganguly 1996-2008 113 188 17 7212 239 42.17 16 35 13

json format

jsonlite for json data

http://arxiv.org/abs/1403.2805

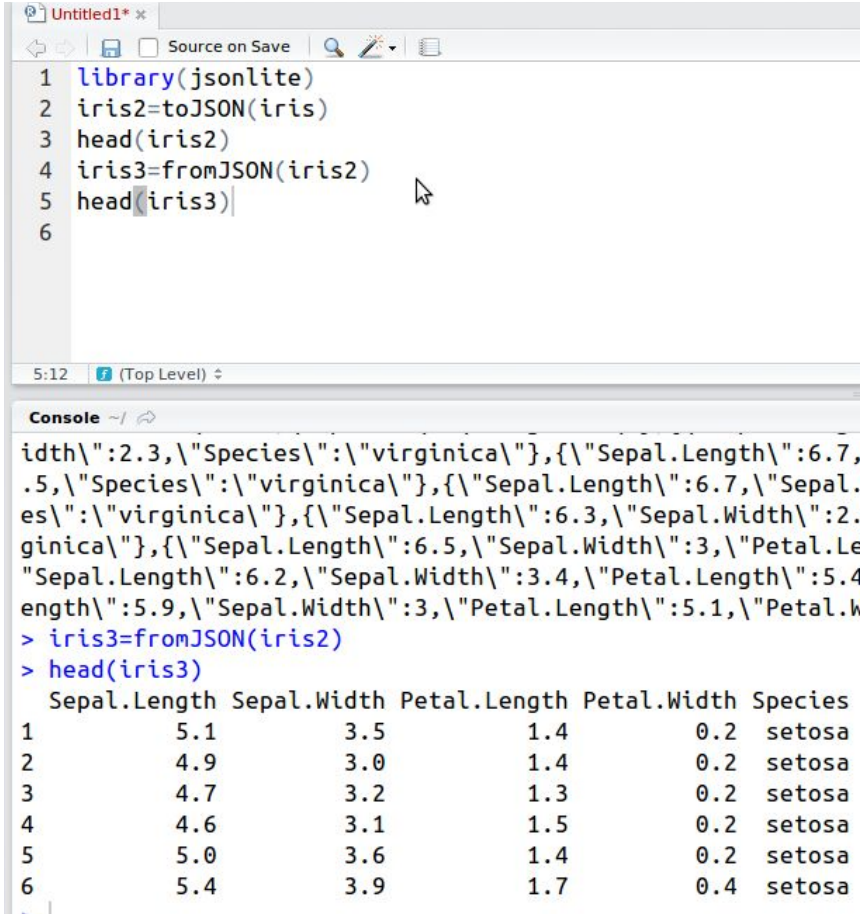
```
> library(jsonlite)

Attaching package: ‘jsonlite’

The following object is masked from ‘package:utils’:

    View

> library(httr)
> library(curl)
> zips <- stream_in(curl("https://media.mongodb.org/zips.json"))
opening curl input connection.
Found 29353 lines...
binding pages together (no custom handler).
closing curl input connection.
>
> head(zips)
    _id        city                  loc   pop state
1 01001      AGAWAM -72.62274, 42.07021 15338    MA
2 01002     CUSHMAN -72.51565, 42.37702 36963    MA
3 01005       BARRE -72.10835, 42.40970  4546    MA
4 01007 BELCHERTOWN -72.41095, 42.27510 10579    MA
5 01008   BLANDFORD -72.93611, 42.18295  1240    MA
6 01010   BRIMFIELD -72.18846, 42.11654  3706    MA
```


```
{ "_id" : "01001", "city" : "AGAWAM", "loc" : [ -72.622739, 42.070206 ], "pop" : 15338, "state" : "MA" }
{ "_id" : "01002", "city" : "CUSHMAN", "loc" : [ -72.51564999999999, 42.377017 ], "pop" : 36963, "state" : "MA" }
{ "_id" : "01005", "city" : "BARRE", "loc" : [ -72.10835400000001, 42.409698 ], "pop" : 4546, "state" : "MA" }
{ "_id" : "01007", "city" : "BELCHERTOWN", "loc" : [ -72.41095300000001, 42.275103 ], "pop" : 10579, "state" : "MA" }
{ "_id" : "01008", "city" : "BLANDFORD", "loc" : [ -72.936114, 42.182949 ], "pop" : 1240, "state" : "MA" }
{ "_id" : "01010", "city" : "BRIMFIELD", "loc" : [ -72.188455, 42.116543 ], "pop" : 3706, "state" : "MA" }
{ "_id" : "01011", "city" : "CHESTER", "loc" : [ -72.988761, 42.279421 ], "pop" : 1688, "state" : "MA" }
{ "_id" : "01012", "city" : "CHESTERFIELD", "loc" : [ -72.833309, 42.38167 ], "pop" : 177, "state" : "MA" }
{ "_id" : "01013", "city" : "CHICOPEE", "loc" : [ -72.607962, 42.162046 ], "pop" : 23396, "state" : "MA" }
{ "_id" : "01020", "city" : "CHICOPEE", "loc" : [ -72.576142, 42.176443 ], "pop" : 31495, "state" : "MA" }
{ "_id" : "01022", "city" : "WESTOVER AFB", "loc" : [ -72.558657, 42.196672 ], "pop" : 1764, "state" : "MA" }
{ "_id" : "01026", "city" : "CUMMINGTON", "loc" : [ -72.905767, 42.435296 ], "pop" : 1484, "state" : "MA" }
{ "_id" : "01027", "city" : "MOUNT TOM", "loc" : [ -72.67992099999999, 42.264319 ], "pop" : 16864, "state" : "MA" }
{ "_id" : "01028", "city" : "EAST LONGMEADOW", "loc" : [ -72.505565, 42.067203 ], "pop" : 13367, "state" : "MA" }
{ "_id" : "01030", "city" : "FEEDING HILLS", "loc" : [ -72.675077, 42.07182 ], "pop" : 11985, "state" : "MA" }
{ "_id" : "01031", "city" : "GILBERTVILLE", "loc" : [ -72.19858499999999, 42.332194 ], "pop" : 2385, "state" : "MA" }
{ "_id" : "01032", "city" : "GOSHEN", "loc" : [ -72.844092, 42.466234 ], "pop" : 122, "state" : "MA" }
```

json format

jsonlite for json data

http://arxiv.org/abs/1403.2805



Using APIs for data

https://ropensci.org/

CRAN Task View: Web Technologies and Services

Maintainer: Scott Chamberlain, Thomas Leeper, Patrick Mair, Karthik Ram, Christopher Gandrud
Contact: scott at ropensci.org
Version: 2015-03-20

This task view contains information about using R to obtain and parse data from the web. The base version of R does not ship with many tools for interacting with the web. Thankfully, there are an increasingly large number of tools for interacting with the web. A list of available packages and functions is presented below, grouped by the type of activity. If you have any comments or suggestions for additions or improvements for this taskview, go to GitHub and submit an issue , or make some changes and submit a pull request . If you can't contribute on GitHub, send Scott an email . If you have an issue with one of the packages discussed below, please contact the maintainer of that package. If you know of a web service, API, data source, or other online resource that is not yet supported by an R package, consider adding it to the package development to do list on GitHub .

Tools for Working with the Web from R

Parsing Data from the Web

- *downloading files* : `download.file()` is in base R and commonly used way to download a file. However, downloading files over HTTPS is not supported in R's internal method for `download.file()`. The `download()` function in the package downloader wraps `download.file()`, and takes all the same arguments, but works for https across platforms.
- *tabular data as txt, csv, etc.* : You can use `read.table()`, `read.csv()`, and friends to read a table directly from a URL, or after acquiring the csv file from the web via e.g., `getURL()` from RCurl. `read.csv()` works with http but not https, i.e.: `read.csv("http://...")`, but not `read.csv("https://...")`. You can download a file first before reading the file in R, and you can use downloader to download over https. `read.table()` and friends also have a `text` parameter so you can read a table if a table is encoded as a string with line breaks, etc.
- *JSON I/O* : JSON is *javascript object notation* . There are three packages for reading and writing JSON: rjson, RJSONIO, and jsonlite. jsonlite includes a different parser from RJSONIO called yajl . We recommend using jsonlite. Check out the paper describing jsonlite by Jeroen Ooms http://arxiv.org/abs/1403.2805 .
- *XML/HTML I/O* : The package XML contains functions for parsing XML and HTML, and supports xpath for searching XML (think regex for strings). A helpful function to read data from one or more HTML tables is `readHTMLTable()`. XML also includes XPATH parsing ability, see `xpathApply()` and `xpathSApply()`. The XML2R package is a collection of convenient functions for coercing XML into data frames (development version on GitHub). An alternative to XML is selectr , which parses CSS3 Selectors and translates them to XPath 1.0 expressions. XML package is often used for parsing xml and html, but selectr translates CSS selectors to XPath, so can use the CSS selectors instead of XPath. The selectorgadget browser extension can be used to identify page elements. RHTMLForms reads HTML documents and obtains a description of each of the forms it contains, along with the different elements and hidden fields. scrapeR provides additional tools for scraping data from HTML and XML documents.
- rvest : rvest scrapes html from web pages, and is designed to work with magrittr to make it easy to express common web scraping tasks.
- The tldextract package extract top level domains and subdomains from a host name. It's a port of a Python library of the same name .
- webutils: Utility functions for developing web applications. Parsers for `application/x-www-form-urlencoded` as well as `multipart/form-data`. Source on Github
- urltools: URL encoding, decoding, parsing, and parameter extraction. Source on Github
- The repmis package contains a `source_data()` command to load and cache plain-text data from a URL (either http or https). It also includes `source_Dropbox()` for downloading/caching plain-text data from non-public Dropbox folders and `source_XlsxData()` for downloading/caching Excel xlsx sheets.
- rsdmx provides tools to read data and metadata documents exchanged through the Statistical Data and Metadata Exchange (SDMX) framework. The package currently focuses on the SDMX XML standard format (SDMX-ML). project website (Github) .

Curl, HTTP, FTP, HTML, XML, SOAP

- RCurl: A low level curl wrapper that allows one to compose general HTTP requests and provides convenient functions to fetch URIs, get/post forms, etc. and process the results returned by the Web server. This provides a great deal of control over the HTTP/FTP connection and the form of the request while providing a higher-level interface than is available just using R socket connections. It also provide tools for Web authentication.

ff package

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

The ff package provides data structures that are stored on disk but behave (almost) as if they were in RAM by transparently mapping only a section (pagesize) in main memory - the effective virtual memory consumption per ff object.

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

Basic (statistical) functionality for package ff

Example- http://www.bnosac.be/index.php/blog/22-if-you-are-into-large-data-and-work-a-lot-package-ff

```
> require(ffbase)
> hhp <-
read.table.ffdf(file="/home/jan/Work/RForgeBNOSAC/github/RBelgium_HeritageHealthPrize/Data/Claims.csv", FUN =
"read.csv", na.strings = "")
```

Also see http://cran.r-project.org/web/packages/bigmemory/index.html

Create, store, access, and manipulate massive matrices. Matrices are allocated to shared memory and may use memory-mapped files. Packages biganalytics, bigtabulate, synchronicity, and bigalgebra provide advanced functionality

RevoScaleR package

RevoScaleR has its own file format, XDF, which is able to rapidly access data by row or by column and to read some data sequentially. XDF file data is stored in the same binary format used in memory, which eliminates the need for conversion when it is brought into memory.

http://www.revolutionanalytics.com/revolution-r-enterprise-scaler

rhdf5

This R/Bioconductor package provides an interface between HDF5 and R. HDF5's main features are the ability to store and access very large and/or complex datasets and a wide variety of metadata on mass storage (disk) through a completely portable file format.

http://www.bioconductor.org/packages/release/bioc/html/rhdf5.html

HDF5 is a data model, library, and file format for storing and managing data. It supports an unlimited variety of datatypes, and is designed for flexible and efficient I/O and for high volume and complex data. HDF5 is portable and is extensible, allowing applications to evolve in their use of HDF5.

https://www.hdfgroup.org/HDF5/

HDF5 simplifies the file structure to include only two major types of object:

- Datasets, which are multidimensional arrays of a homogeneous type
- Groups, which are container structures which can hold datasets and other groups

rhdf5

platforms all | downloads top 5% | posts 10 / 2 / 0.9 / 1 | in Bioc > 10 years
build ok | commits 0.83 | test coverage unknown

HDF5 interface to R

Bioconductor version: Release (3.1)

This R/Bioconductor package provides an interface between HDF5 and R. HDF5's main features are the ability to store and access very large and/or complex datasets and a wide variety of metadata on mass storage (disk) through a completely portable file format. The rhdf5 package is thus suited for the exchange of large and/or complex datasets between R and other software package, and for letting R applications work on datasets that are larger than the available RAM.

Author: Bernd Fischer, Gregoire Pau

Maintainer: Bernd Fischer <b.fischer at dkfz.de>

Citation (from within R, enter `citation("rhdf5")`):

Fischer B and Pau G. *rhdf5: HDF5 interface to R*. R package version 2.12.0.

Installation

To install this package, start R and enter:

```
source("http://bioconductor.org/biocLite.R")
biocLite("rhdf5")
```

Documentation

To view documentation for the version of this package installed in your system, start R and enter:

```
browseVignettes("rhdf5")
```

PDF	R Script	rhdf5
PDF		Reference Manual
Text		NEWS

Documentation »

Bioconductor

- Package vignettes and manuals.
- Workflows for learning and use.
- Course and conference material.
- Videos.
- Community resources and tutorials.

R / CRAN packages and documentation

Support »

Please read the posting guide. Post questions about Bioconductor to one of the following locations:

- Support site - for questions about Bioconductor packages
- Bioc-devel mailing list - for package developers

Functions Used in this lesson

- toJSON and fromJSON

Packages

- data.table
- jsonlite
- rvest

Revision

getwd

setwd

dir

ls

rm

Install.packages

library

fread vs read,csv

Df[i,j]

Df$column

str

summary

table

citation

help

Revision

mean

std

median

length

Vector

data.frame

Indexing

class

nrow

ncol

head

tail

Citations and References

M Dowle, T Short, S Lianoglou, A Srinivasan with contributions from R Saporta and E Antonyan (2014) data.table: Extension of data.frame. R package version 1.9.4. http://CRAN.R-project.org/package=data.table

Jeroen Ooms (2014). The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects. arXiv:1403.2805 [stat.CO] URL http://arxiv.org/abs/1403.2805

Hadley Wickham (2015). rvest: Easily Harvest (Scrape) Web Pages. R package version 0.2.0. http://CRAN.R-project.org/package=rvest