



The Data Science Track

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Why do data science?

"It is not the critic who counts: not the man who points out how the strong man stumbles or where the doer of deeds could have done better. The credit belongs to the man who is actually in the arena, whose face is marred by dust and sweat and blood, who strives valiantly, who errs and comes up short again and again, because there is no effort without error or shortcoming, but who knows the great enthusiasms, the great devotions, who spends himself for a worthy cause; who, at the best, knows, in the end, the triumph of high achievement, and who, at the worst, if he fails, at least he fails while daring greatly, so that his place shall never be with those cold and timid souls who knew neither victory nor defeat."



Theodore Roosevelt, 26th President of the United States

[Statistics and the science game](#)

The key challenge in data science

"Ask yourselves, what problem have you solved, ever, that was worth solving, where you knew knew all of the given information in advance? Where you didn't have a surplus of information and have to filter it out, or you didn't have insufficient information and have to go find some?"



[Dan Myer, Mathematics Educator](#)

[The key word in data science is not data; it is science](#)

About us

Data intensive statistics in biology and medicine

- Brian Caffo
 - Website <http://www.bcaffo.com/>
 - Twitter [@bcaffo](https://twitter.com/bcaffo)
 - Github <https://github.com/bcaffo>
- Jeff Leek
 - Website <http://biostat.jhsph.edu/~jleek/>, <http://simplystatistics.org/>
 - Twitter [@jtleek](https://twitter.com/jtleek)
 - Github <https://github.com/jtleek>
- Roger Peng
 - Website <http://www.biostat.jhsph.edu/~rpeng/>, <http://simplystatistics.org/>
 - Twitter [@rdpeng](https://twitter.com/rdpeng)
 - Github <https://github.com/rdpeng>

Why data science?



<http://www.economist.com/node/15579717>

Why data science?

McKinsey Global Institute



June 2011

Big data: The next frontier
for innovation, competition,
and productivity

http://www.mckinsey.com/insights/business_technology/big_data_the_next_frontier_for_innovation

Why statistical data science?



For Today's Graduate, Just One Word: Statistics

By STEVE LOHR

Published: August 5, 2009

MOUNTAIN VIEW, Calif. — At Harvard, Carrie Grimes majored in anthropology and archaeology and ventured to places like Honduras, where she studied Mayan settlement patterns by mapping where artifacts were found. But she was drawn to what she calls “all the computer and math stuff” that was part of the job.

TWITTER

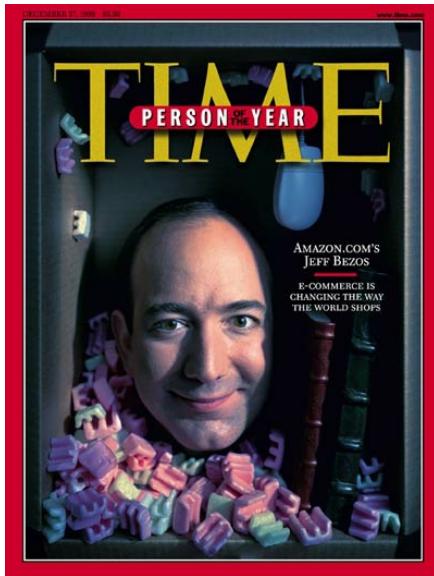
LINKEDIN

COMMENTS
(58)

SIGN IN TO E-
MAIL

http://www.nytimes.com/2009/08/06/technology/06stats.html?_r=0

Why are you lucky?



Why are you lucky?

Information Data Forum Leaderboard



**Improve Healthcare,
Win \$3,000,000.**

COMPETITION GOAL

Identify patients who will be admitted to a hospital within the next year, using historical claims data.

[Heritage Health Prize](#)

Why R?

The New York Times

Business Computing

Search All NYTimes.com

Capital One 360

WORLD U.S. N.Y. / REGION BUSINESS TECHNOLOGY SCIENCE HEALTH SPORTS OPINION ARTS STYLE TRAVEL JOBS REAL ESTATE AUTOS



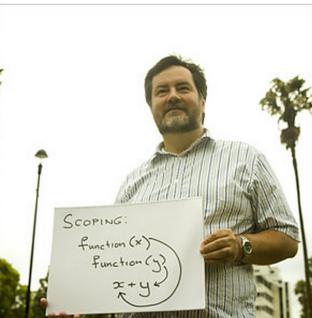
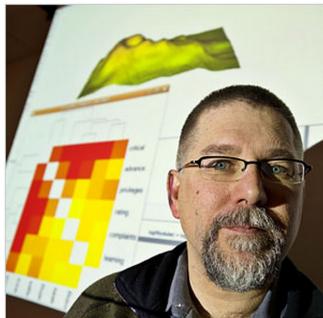
PROGRESS IS EVERYONE'S BUSINESS

See how Goldman Sachs has helped Hologic enable better outcomes for patients.

► WATCH THE VIDEO



Data Analysts Captivated by R's Power



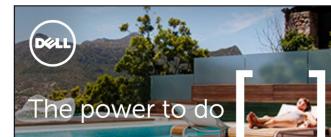
Log in to see what your friends are sharing Log In With Facebook
on nytimes.com. [Privacy Policy](#) | [What's This?](#)

What's Popular Now

Amiri Baraka,
Polarizing Poet
and Playwright,
Dies at 79



'Very Sad' Chris
Christie Extends
Apology in Bridge
Scandal



<http://www.nytimes.com/2009/01/07/technology/business-computing/07program.html?pagewanted=all>

Why R?

- It is free
- It has a comprehensive set of packages
 - Data access
 - Data cleaning
 - Analysis
 - Data reporting
- It has one of the best development environments - Rstudio <http://www.rstudio.com/>
- It has an amazing ecosystem of developers
- Packages are easy to install and "play nicely together"

Who is a data scientist?



[Daryl Morey](#)

Who is a data scientist?



Hilary Mason

Who is a data scientist?



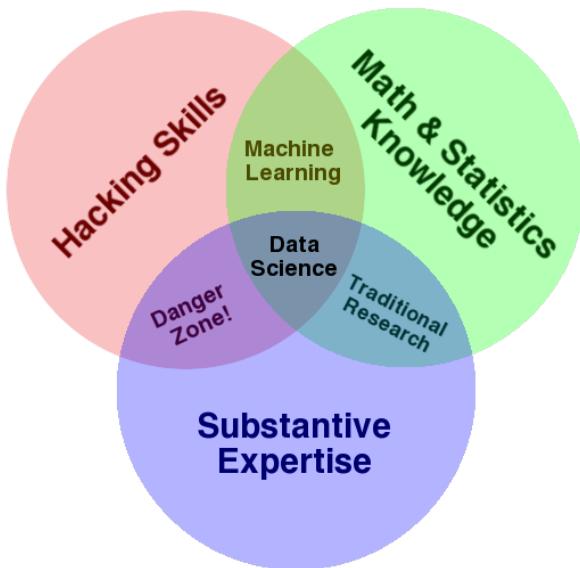
[Daphne Koller](#)

Who is a data scientist?



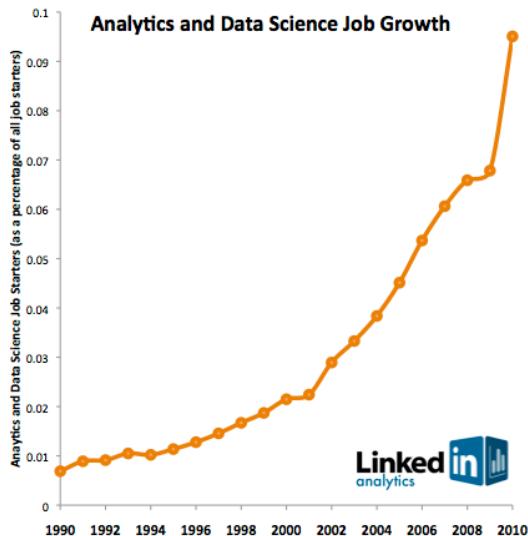
[Nate Silver](#)

Our goal



[Drew Conway](#)

Plus jobs



<http://radar.oreilly.com/2011/09/building-data-science-teams.html>

This course

- Introducing you to the track
- Getting tools set up
- Giving you basic background



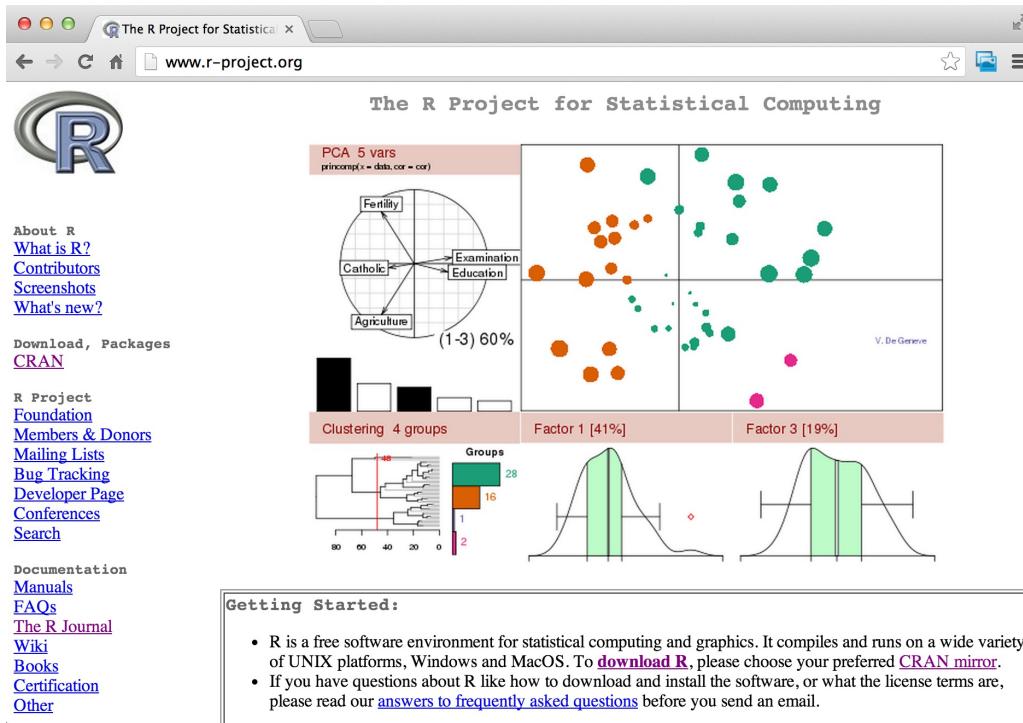
The Data Scientist's Toolbox

Johns Hopkins Bloomberg School of Public Health

What do data scientists do?

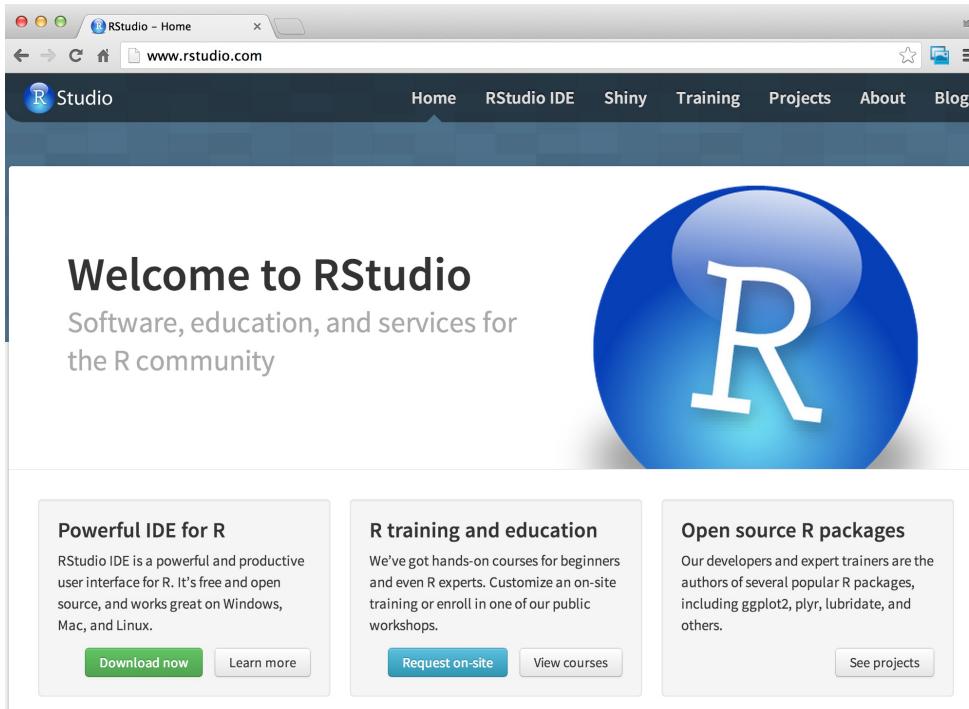
- Define the question
- Define the ideal data set
- Determine what data you can access
- Obtain the data
- Clean the data
- Exploratory data analysis
- Statistical prediction/modeling
- Interpret results
- Challenge results
- Synthesize/write up results
- Create reproducible code
- Distribute results to other people

The main workhorse of data science



<http://www.r-project.org/>

Where we will work on coding

A screenshot of the RStudio website homepage. The page features a large blue header with the RStudio logo and navigation links for Home, RStudio IDE, Shiny, Training, Projects, About, and Blog. Below the header is a large white section with the text "Welcome to RStudio" and "Software, education, and services for the R community". To the right is a large blue circular logo with a white "R". Below this are three callout boxes: "Powerful IDE for R", "R training and education", and "Open source R packages".

Welcome to RStudio

Software, education, and services for the R community

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.

[Download now](#) [Learn more](#)

R training and education

We've got hands-on courses for beginners and even R experts. Customize an on-site training or enroll in one of our public workshops.

[Request on-site](#) [View courses](#)

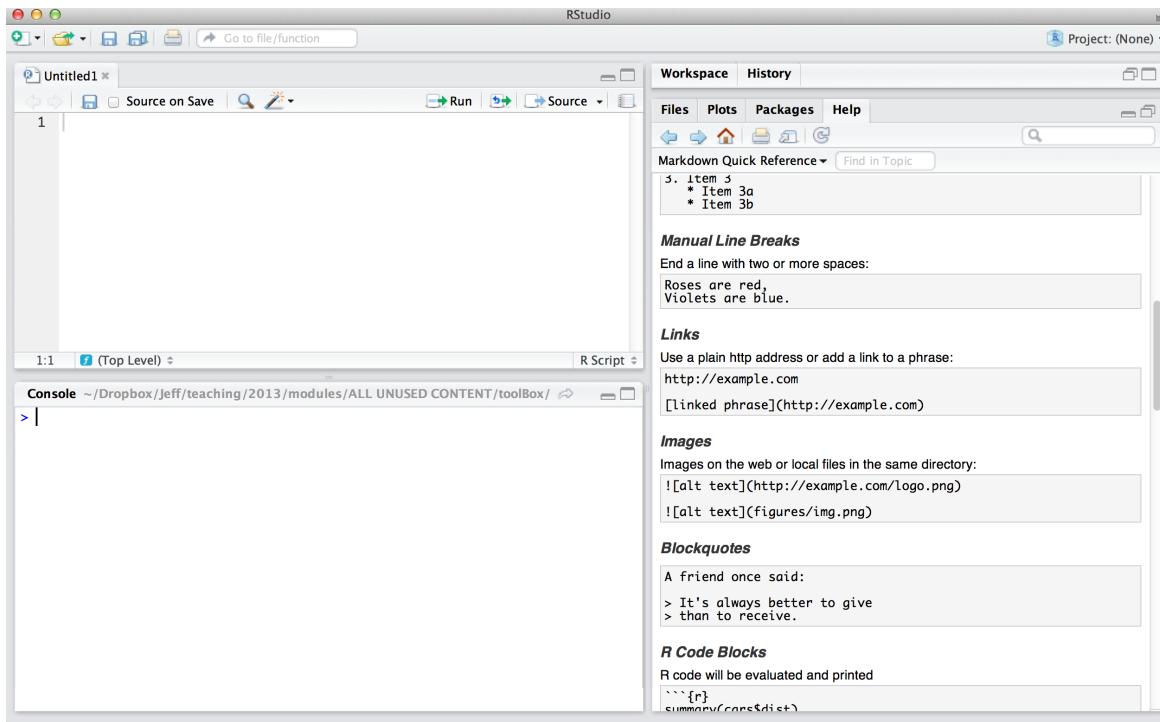
Open source R packages

Our developers and expert trainers are the authors of several popular R packages, including ggplot2, plyr, lubridate, and others.

[See projects](#)

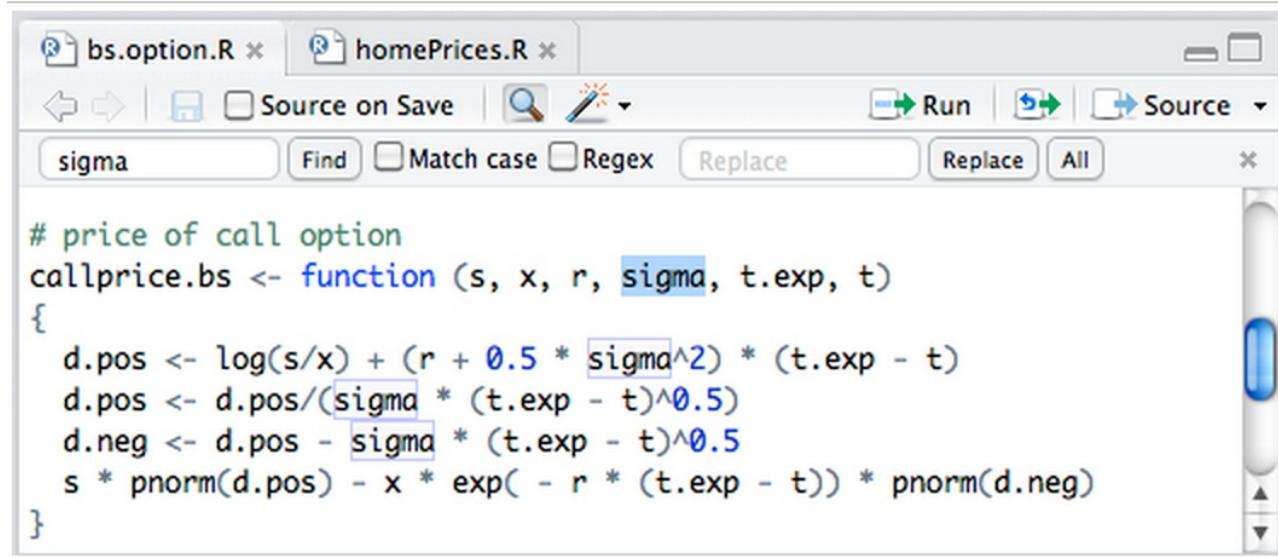
<http://www.rstudio.com/>

Rstudio's interface



<http://www.rstudio.com/>

Primary file types - R script



```
# price of call option
callprice.bs <- function (s, x, r, sigma, t.exp, t)
{
  d.pos <- log(s/x) + (r + 0.5 * sigma^2) * (t.exp - t)
  d.pos <- d.pos/(sigma * (t.exp - t)^0.5)
  d.neg <- d.pos - sigma * (t.exp - t)^0.5
  s * pnorm(d.pos) - x * exp(-r * (t.exp - t)) * pnorm(d.neg)
}
```

<http://www.rstudio.com/ide/docs/using/source>

Primary file types - R markdown document

The screenshot shows the RStudio interface with two panes. The left pane displays the R Markdown source code in an Rmd file named 'example.Rmd'. The right pane shows the generated HTML preview.

R Markdown Source (example.Rmd):

```
1 Header 1
2 -----
3 This is an R Markdown document. Markdown is a
4 simple formatting syntax for authoring web pages.
5
6 Use an asterisk mark, to provide emphasis such as
7 *italics* and **bold**.
8
9 Create lists with a dash:
10 - Item 1
11 - Item 2
12 - Item 3
13
14 ...
15 Code blocks display
16 with fixed-width font
17 ``
18
19 > Blockquotes are offset
20
```

HTML Preview:

Header 1

This is an R Markdown document. Markdown is a simple formatting syntax for authoring web pages.

Use an asterisk mark, to provide emphasis such as *italics* and **bold**.

Create lists with a dash:

- Item 1
- Item 2
- Item 3

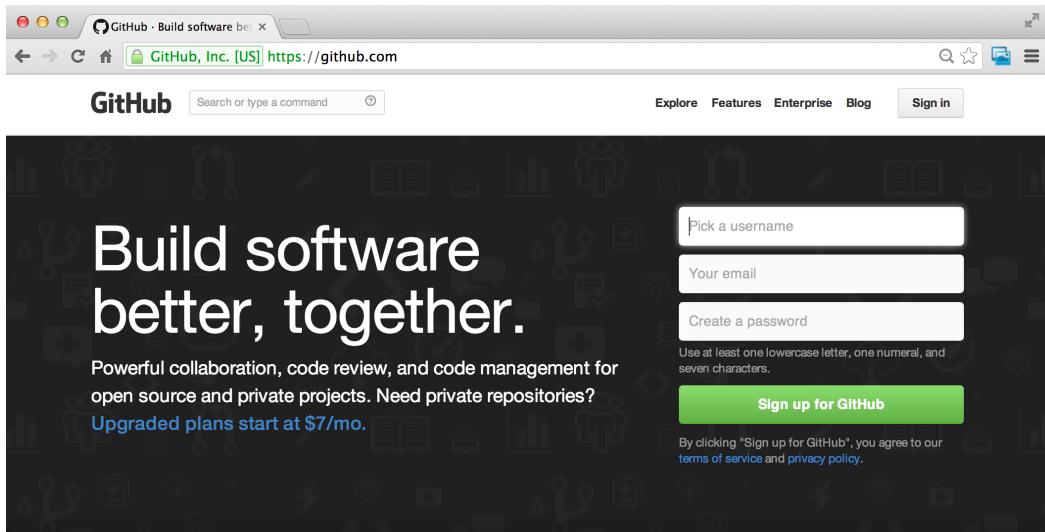
You can write `in-line` code with a back-tick.

Code blocks display
with fixed-width font

Blockquotes are offset

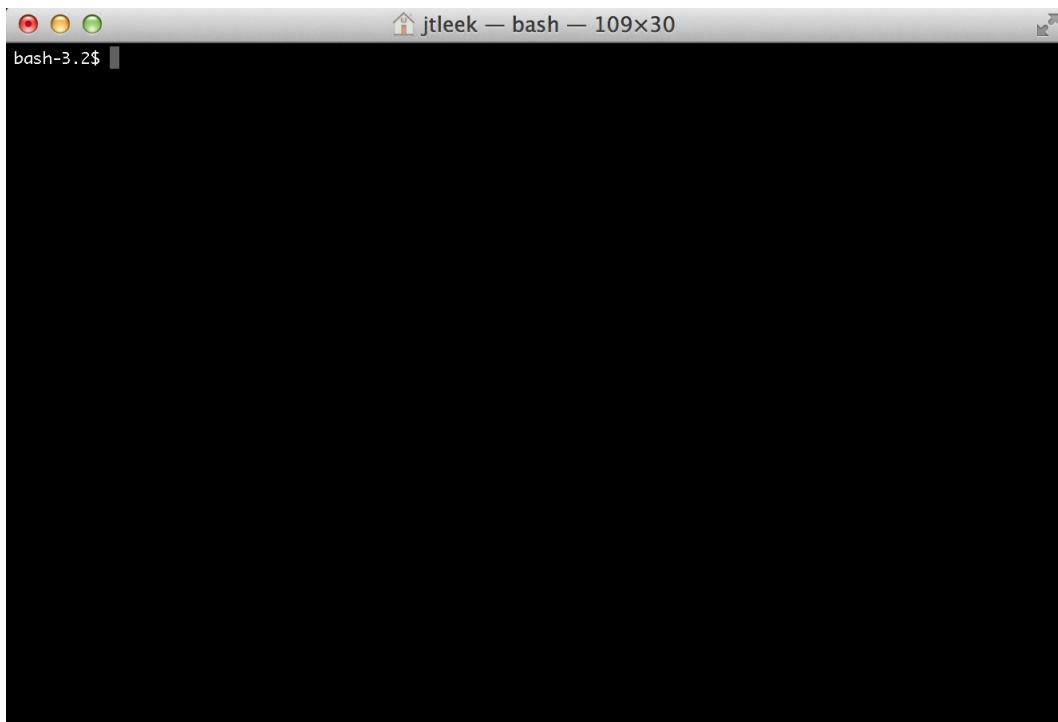
http://www.rstudio.com/ide/docs/authoring/using_markdown

Sharing your results - Github & Git



Why you'll love GitHub.
Powerful features to make software development more collaborative.

Where to run Github commands - the shell





Getting help

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Asking questions

- In a standard class
 - There are 30-100 people
 - You raise your hand and ask a question
 - The instructor responds
- In a MOOC
 - There are almost 100,000 people
 - You post a question to the message board
 - Others vote on your questions
 - Your instructor responds (as often as possible)
 - Your peers respond (as often as possible)

Often the fastest answer is the one you find yourself

- It's important to try to answer your own questions first
- If the answer to your question is in the help file or the top hit on Google, the answer to your question will be, "Read the documentation" or "Google it" (<http://lmgtfy.com/>)
- If you figure out the answer and see the same questions on the forum, post the solution you found

Some important R functions

Access help file

```
?rnorm
```

Search help files

```
help.search("rnorm")
```

Get arguments

```
args("rnorm")
```

```
function (n, mean = 0, sd = 1)  
NULL
```

Some important R functions

See code

```
rnorm
```

```
function (n, mean = 0, sd = 1)
.Internal(C_rnorm, n, mean, sd)
<bytecode: 0x7f9173a9f630>
<environment: namespace:stats>
```

R reference card

<http://cran.r-project.org/doc/contrib/Short-refcard.pdf>

How to ask an R question

- What steps will reproduce the problem?
- What is the expected output?
- What do you see instead?
- What version of the product (e.g. R, packages, etc.) are you using?
- What operating system?

How to ask a data analysis question

- What is the question you are trying to answer?
- What steps/tools did you use to answer it?
- What did you expect to see?
- What do you see instead?
- What other solutions have you thought about?

Be specific in the title of your questions

- Bad:
 - HELP! Can't fit linear model!
 - HELP! Don't understand PCA!
- Better
 - R 2.15.0 lm() function produces seg fault with large data frame, Mac OS X 10.6.3
 - Applied principal component analysis to a matrix - what are U, D, and V^T ?
- Even better
 - R 2.15.0 lm() function on Mac OS X 10.6.3 -- seg fault on large data frame
 - Using principal components to discover common variation in rows of a matrix, should I use U, D or V^T ?

Etiquette for forums/help sites: DOs

- Describe the goal
- Be explicit
- Provide the minimum information
- Be courteous (never hurts)
- Follow up and post solutions
- Use the forums rather than email

Etiquette for forums/help sites: DON'Ts

- Immediately assume you found a bug
- Grovel as a substitute for doing your work
- Post homework questions on mailing lists (people don't like doing your homework)
- Email multiple mailing lists at once/the wrong mailing list
- Ask others to fix your code without explaining the problem
- Ask about general data analysis questions on R forums.

Credits

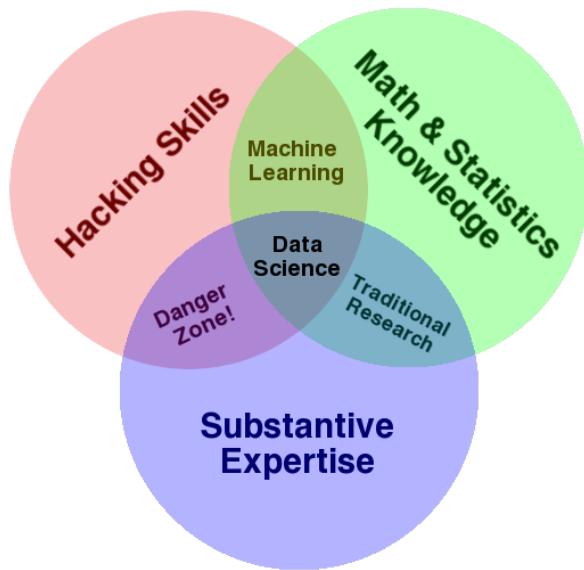
- Roger's [Getting Help Video](#)
- Inspired by Eric Raymond's "How to ask questions the smart way"



Finding answers

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

One of the key data science traits



[Drew Conway](#)

Key characteristics of hackers

- Willing to find answers on their own
- Knowledgeable about where to find answers on their own
- Unintimidated by new data types or packages
- Unafriad to say they don't know the answer
- *Polite but relentless*

[Google knows it too](#)

Where to look for different types of questions

- R programming (see also: <http://bit.ly/Ufaadn>)
 - Search the archive of the class forums
 - Read the manual/help files
 - Search on the web
 - Ask a skilled friend
 - Post to the class forums
 - Post to the [R mailing list](#) or [Stackoverflow](#)
- Data Analysis/Statistics
 - Search the archive of the class forums
 - Search on the web
 - Ask a skilled friend
 - Post to the class forums

A note on Googling data science questions

- The best place to start for general questions is the forums
- [Stackoverflow](#) (use the tag "[r]"), [R mailing list](#) for software questions, [CrossValidated](#) for more general questions
- Otherwise Google "[data type] data analysis" or "[data type] R package"
- Try to identify what data analysis is called for your data type
 - [Biostatistics](#) for medical data
 - [Data Science](#) for data from web analytics
 - [Machine learning](#) for data in computer science/computer vision
 - [Natural language processing](#) for data from texts
 - [Signal processing](#) for data from electrical signals
 - [Business analytics](#) for data on customers
 - [Econometrics](#) for economic data
 - [Statistical process control](#) for data about industrial processes

Credits

- Roger's [Getting Help Video](#)
- Inspired by Eric Raymond's "How to ask questions the smart way"



R Programming Overview

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

R programming content

- Data types
- Subsetting
- Reading and writing data
- Control structures
- Functions
- Scoping
- Vectorized operations
- Dates and times
- Debugging
- Simulation
- Optimization

Reading Lines of a Text File

`readLines` can be useful for reading in lines of webpages

```
## This might take time
con <- url("http://www.jhsph.edu", "r")
x <- readLines(con)
> head(x)
[1] "<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">"
[2] ""
[3] "<html>"
[4] "<head>"
[5] "\t<meta http-equiv=\"Content-Type\" content=\"text/html; charset=utf-8
```

Something's Wrong!

How do you know that something is wrong with your function?

- What was your input? How did you call the function?
- What were you expecting? Output, messages, other results?
- What did you get?
- How does what you get differ from what you were expecting?
- Were your expectations correct in the first place?
- Can you reproduce the problem (exactly)?

lapply

`lapply` takes three arguments: a list `x`, a function (or the name of a function) `FUN`, and other arguments via its `...` argument. If `x` is not a list, it will be coerced to a list using `as.list`.

```
> lapply
function (x, FUN, ...)
{
  FUN <- match.fun(FUN)
  if (!is.vector(x) || is.object(x))
    x <- as.list(x)
  .Internal(lapply(x, FUN))
}
```

The actual looping is done internally in C code.



Getting and Cleaning Data Overview

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Getting and Cleaning Data Content

- Raw vs. tidy data
- Downloading files
- Reading data
 - Excel, XML, JSON, MySQL, HDF5, Web, ...
- Merging data
- Reshaping data
- Summarizing data
- Finding and replacing
- Data resources

Connecting and listing databases

```
ucscDb <- dbConnect(MySQL(), user = "genome", host = "genome-mysql.cse.ucsc.edu")
result <- dbGetQuery(ucscDb, "show databases;")
dbDisconnect(ucscDb)
result
```

Merging data - merge()

```
mergedData2 <- merge(reviews, solutions, by.x = "solution_id", by.y = "id",
  all = TRUE)
head(mergedData2[, 1:6], 3)
reviews[1, 1:6]
```

Raw versus processed data

Raw data

- The original source of the data
- Often hard to use for data analyses
- Data analysis *includes* processing
- Raw data may only need to be processed once

http://en.wikipedia.org/wiki/Raw_data

Processed data

- Data that is ready for analysis
- Processing can include merging, subsetting, transforming, etc.
- There may be standards for processing
- All steps should be recorded

http://en.wikipedia.org/wiki/Computer_data_processing



Exploratory Analysis Overview

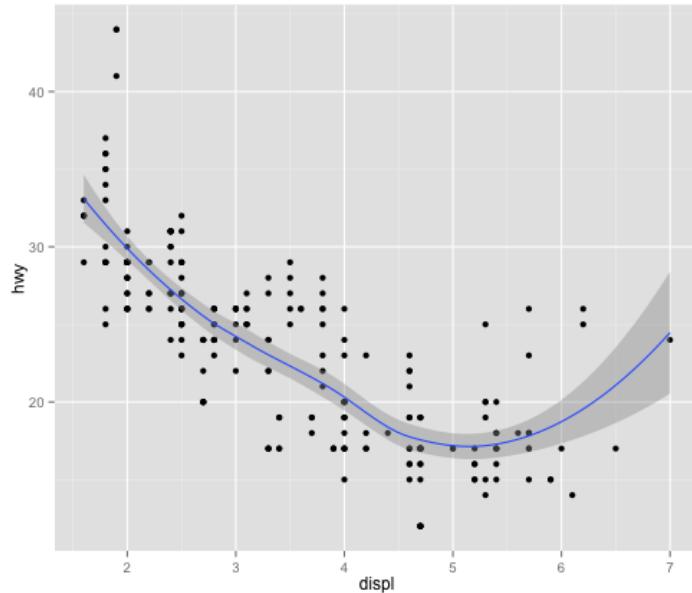
Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Exploratory Analysis Content

- Principles of Analytic Graphics
- Exploratory graphs
- Plotting Systems in R
 - base
 - lattice
 - ggplot2
- Hierarchical clustering
- K-Means clustering
- Dimension reduction

Adding a geom

```
qplot(displ, hwy, data = mpg, geom = c("point", "smooth"))
```

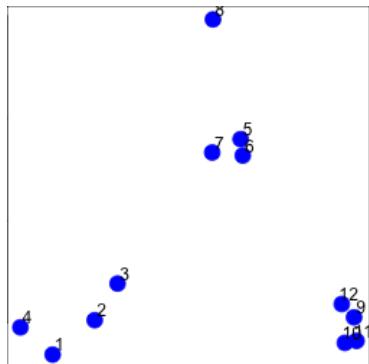


Principles of Analytic Graphics

- Principle 1: Show comparisons
- Principle 2: Show causality, mechanism, explanation
- Principle 3: Show multivariate data
- Principle 4: Integrate multiple modes of evidence
- Principle 5: Describe and document the evidence
- Principle 6: Content is king

K-means clustering - example

```
set.seed(1234)
par(mar = c(0, 0, 0, 0))
x <- rnorm(12, mean = rep(1:3, each = 4), sd = 0.2)
y <- rnorm(12, mean = rep(c(1, 2, 1), each = 4), sd = 0.2)
plot(x, y, col = "blue", pch = 19, cex = 2)
text(x + 0.05, y + 0.05, labels = as.character(1:12))
```





Reproducible Research Overview

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Reproducible Research Content

- Structure of a Data Analysis
- Organizing a Data Analysis
- Markdown
- LaTeX
- R Markdown
- Evidence-based data analysis
- RPubs

Steps in a data analysis

- Define the question
- Define the ideal data set
- Determine what data you can access
- Obtain the data
- Clean the data
- Exploratory data analysis
- Statistical prediction/modeling
- Interpret results
- Challenge results
- Synthesize/write up results
- Create reproducible code

Data analysis files

- Data
 - Raw data
 - Processed data
- Figures
 - Exploratory figures
 - Final figures
- R code
 - Raw scripts
 - Final scripts
 - R Markdown files (optional)
- Text
 - Readme files
 - Text of analysis



Statistical Inference Overview

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Statistical Inference Content

- Basic probability
- Likelihood
- Common distributions
- Asymptotics
- Confidence intervals
- Hypothesis tests
- Power
- Bootstrapping
- Non-parametric tests
- Basic bayesian statistics

Example

Suppose that the proportion of help calls that get addressed in a random day by a help line is given by

$$f(x) = \begin{cases} 2x & \text{for } 0 < x < 1 \\ 0 & \text{otherwise} \end{cases}$$

Is this a mathematically valid density?

The normal distribution

- A random variable is said to follow a **normal** or **Gaussian** distribution with mean μ and variance σ^2 if the associated density is

$$(2\pi\sigma^2)^{-1/2} e^{-(x-\mu)^2/2\sigma^2}$$

If X a RV with this density then $E[X] = \mu$ and $\text{Var}(X) = \sigma^2$

- We write $X \sim N(\mu, \sigma^2)$
- When $\mu = 0$ and $\sigma = 1$ the resulting distribution is called **the standard normal distribution**
- The standard normal density function is labeled ϕ
- Standard normal RVs are often labeled Z

Example bootstrap code

```
B <- 1000
n <- length(gmVol)
resamples <- matrix(sample(gmVol,
                           n * B,
                           replace = TRUE),
                      B, n)
medians <- apply(resamples, 1, median)
sd(medians)
[1] 3.148706
quantile(medians, c(.025, .975))
 2.5%    97.5%
582.6384 595.3553
```



Regression Models Overview

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Regression Models Content

- Linear regression
- Multiple Regression
- Confounding
- Residuals and diagnostics
- Prediction using linear models
- Model misspecification
- Scatterplot smoothing/splines
- Machine learning via regression
- Resampling inference in regression, bootstrapping, permutation tests
- Weighted regression
- Mixed models (random intercepts)

A historically famous idea, Regression to the Mean

- Why is it that the children of tall parents tend to be tall, but not as tall as their parents?
- Why do children of short parents tend to be short, but not as short as their parents?
- Why do parents of very short children, tend to be short, but not as short as their child? And the same with parents of very tall children?
- Why do the best performing athletes this year tend to do a little worse the following?

Basic regression model with additive Gaussian errors

- Least squares is an estimation tool, how do we do inference?
- Consider developing a probabilistic model for linear regression

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$$

- Here the ϵ_i are assumed iid $N(0, \sigma^2)$.
- Note, $E[Y_i | X_i = x_i] = \mu_i = \beta_0 + \beta_1 x_i$
- Note, $\text{Var}(Y_i | X_i = x_i) = \sigma^2$.
- Likelihood equivalent model specification is that the Y_i are independent $N(\mu_i, \sigma^2)$.

Multivariable regression analyses

- An insurance company is interested in how last year's claims can predict a person's time in the hospital this year.
 - They want to use an enormous amount of data contained in claims to predict a single number. Simple linear regression is not equipped to handle more than one predictor.
- How can one generalize SLR to incorporate lots of regressors for the purpose of prediction?
- What are the consequences of adding lots of regressors?
 - Surely there must be consequences to throwing variables in that aren't related to Y?
 - Surely there must be consequences to omitting variables that are?



Practical Machine Learning Overview

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Practical Machine Learning Content

- Prediction study design
- Types of Errors
- Cross validation
- The caret package
- Plotting for prediction
- Preprocessing
- Predicting with regression
- Predicting with trees
- Boosting
- Bagging
- Model blending
- Forecasting

Basic terms

In general, **Positive** = identified and **negative** = rejected. Therefore:

- **True positive** = correctly identified
- **False positive** = incorrectly identified
- **True negative** = correctly rejected
- **False negative** = incorrectly rejected

Medical testing example:

- **True positive** = Sick people correctly diagnosed as sick
- **False positive** = Healthy people incorrectly identified as sick
- **True negative** = Healthy people correctly identified as healthy
- **False negative** = Sick people incorrectly identified as healthy.

http://en.wikipedia.org/wiki/Sensitivity_and_specificity

Correlated predictors

```
library(caret)
library(kernlab)
data(spam)

inTrain <- createDataPartition(y = spam$type, p = 0.75, list = FALSE)
training <- spam[inTrain, ]
testing <- spam[-inTrain, ]

M <- abs(cor(training[, -58]))
diag(M) <- 0
which(M > 0.8, arr.ind = TRUE)
```

```
##          row col
## num415    34  32
## direct    40  32
## num857    32  34
## num857    32  40
```

Basic idea behind boosting

1. Start with a set of classifiers h_1, \dots, h_k
 - Examples: All possible trees, all possible regression models, all possible cutoffs.
2. Create a classifier that combines classification functions: $f(x) = \text{sgn}\left(\sum_{t=1}^T \alpha_t h_t(x)\right)$.
 - Goal is to minimize error (on training set)
 - Iterative, select one h at each step
 - Calculate weights based on errors
 - Upweight missed classifications and select next h

[Adaboost on Wikipedia](#)

<http://webee.technion.ac.il/people/rmeir/BoostingTutorial.pdf>



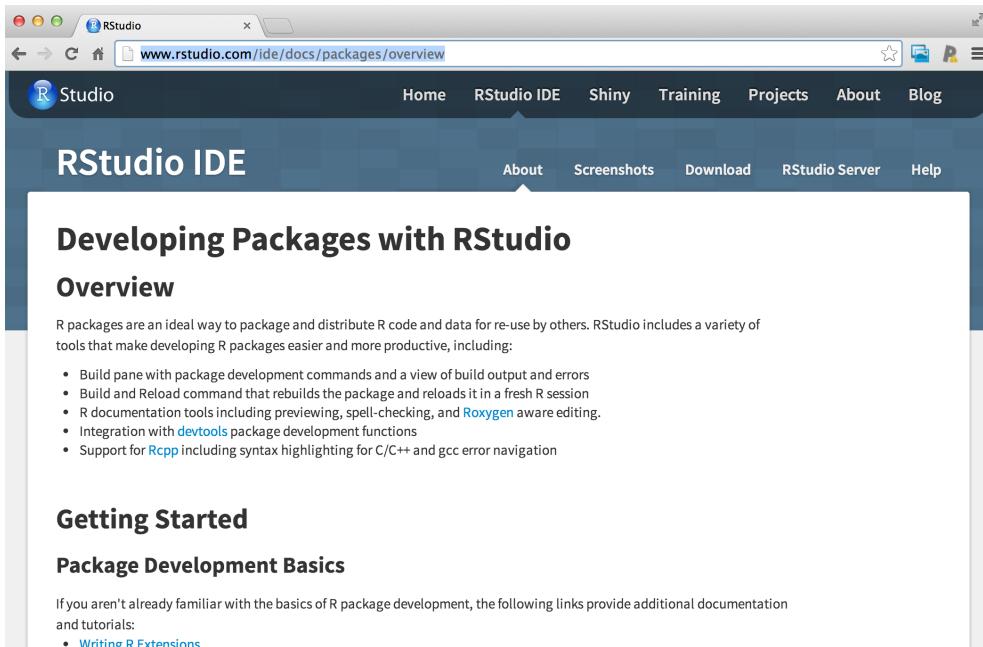
Building Data Products Overview

Jeffrey Leek
Johns Hopkins Bloomberg School of Public Health

Building Data Products Content

- R packages
 - devtools
 - roxygen
 - testthat
- rCharts
- Slidify
- Shiny

R packages - for the engineers



The screenshot shows a web browser window with the URL www.rstudio.com/ide/docs/packages/overview in the address bar. The page is titled "RStudio IDE" and features a navigation bar with links for Home, RStudio IDE, Shiny, Training, Projects, About, and Blog. Below the navigation bar, there is a secondary navigation menu with links for About, Screenshots, Download, RStudio Server, and Help. The main content area is titled "Developing Packages with RStudio Overview". It contains text about the benefits of R packages and a bulleted list of features provided by RStudio's IDE. At the bottom, there is a "Getting Started" section and a "Package Development Basics" section, along with a link to additional resources.

Developing Packages with RStudio

Overview

R packages are an ideal way to package and distribute R code and data for re-use by others. RStudio includes a variety of tools that make developing R packages easier and more productive, including:

- Build pane with package development commands and a view of build output and errors
- Build and Reload command that rebuilds the package and reloads it in a fresh R session
- R documentation tools including previewing, spell-checking, and [Roxygen](#) aware editing.
- Integration with [devtools](#) package development functions
- Support for [Rcpp](#) including syntax highlighting for C/C++ and gcc error navigation

Getting Started

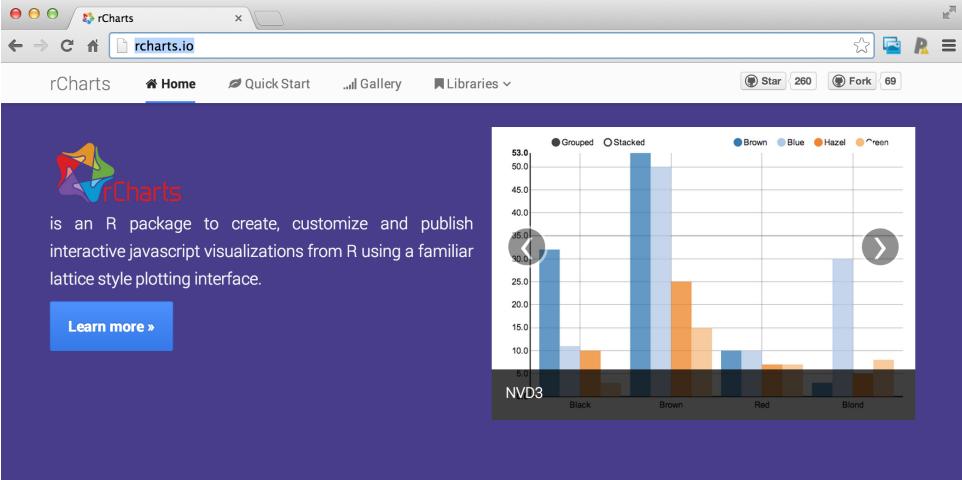
Package Development Basics

If you aren't already familiar with the basics of R package development, the following links provide additional documentation and tutorials:

- [Writing R Extensions](#)

<http://cran.r-project.org/web/packages/> <http://www.rstudio.com/ide/docs/packages/overview>

rCharts - for marketing



The screenshot shows the rCharts.io website. At the top, there's a navigation bar with links for Home, Quick Start, Gallery, Libraries, and a search bar. Below the navigation is a main content area with a purple header containing the rCharts logo and a brief description: "is an R package to create, customize and publish interactive javascript visualizations from R using a familiar lattice style plotting interface." A blue "Learn more »" button is present. To the right of the text is a stacked bar chart titled "NVD3". The chart has four categories on the x-axis: Black, Brown, Red, and Blond. Each category has four bars representing different colors: Brown (dark blue), Blue (light blue), Hazel (orange), and Green (yellow). The y-axis ranges from 0.0 to 53.0. A legend at the top right identifies the chart types: "Grouped" (solid circle) and "Stacked" (open circle). A legend below the chart identifies the colors: Brown, Blue, Hazel, and Green. Below the main content area are three callout boxes: "Familiar Plotting Interface", "Multiple Charting Libraries", and "Easy to Share".

Familiar Plotting Interface

rCharts uses a plotting interface that R users are already familiar with. You can use a

Multiple Charting Libraries

rCharts supports multiple javascript charting libraries, each with its own strengths. Each of

Easy to Share

rCharts allows you to share your visualization in multiple ways. You can save it as a

<http://rcharts.io/> <http://ramnathv.github.io/rChartsNYT/>

Shiny - for your users

The screenshot shows a web browser window for 'RStudio - Shiny' displaying the URL 'www.rstudio.com/shiny/'. The page has a dark blue header with the 'Shiny' logo and navigation links for Home, RStudio IDE, Shiny, Training, Projects, About, and Blog. Below the header, there's a secondary navigation bar with links for Shiny, About Shiny, Showcase, Tutorial, Shiny Server, and Shiny Hosting. The main content area features a large heading 'Easy web applications in R'. A sub-section titled 'Shiny makes it super simple for R users like you to turn analyses into interactive web applications that anyone can use.' is followed by a paragraph explaining how users can choose input parameters and incorporate outputs. Below this, a section titled 'Shiny in action' shows a basic application with a dropdown menu for 'Number of bins in histogram (approximate)' set to 20, and a checkbox for 'Show individual observations' which is unchecked. To the right, the 'ui.R' and 'server.R' files are shown. The 'ui.R' file contains the UI code for the histogram application, and the 'server.R' file contains the server logic.

Number of bins in histogram (approximate):

20

Show individual observations

```
ui.R
server.R

shinyUI(bootstrapPage(
  selectInput(inputId = "n_breaks",
    label = "Number of bins in histogram (approximate)",
    choices = c(10, 20, 35, 50),
    ...
  )
))
```

<http://www.rstudio.com/shiny/> <http://www.rstudio.com/shiny/showcase/>