

# SISBD Module 1

## Accessing Biomedical Data

Jeff Leek

(@jtleek)

Raphael Gottardo

(@raphg)

# Preliminaries



**Raphael Gottardo**  
@raphg

 Follow

Register for Summer Institute in Statistics for Big Data  
[ow.ly/KNjBw](http://ow.ly/KNjBw) with [@rdpeng](#) [@hadleywickham](#) [@raphg](#) and  
the handsome [@jtleek](#)

1:52 PM - 25 Mar 2015



11



21

**Class name:** Accessing Biomedical Big Data

**Instructors:** [Raphael Gottardo](#), [Jeff Leek](#)

**TAs:** [Jean Morrison](#), [Brian Williamson](#)

**Course Website:** <http://sisbid.github.io/Module1>

**Goal:** Teach you how to get and clean data

**Pre-reqs:** Hopefully some R programming

**Where to get slides:** <https://github.com/SISBID/Module1>

Motivating  
Example

# An exciting result

## Genomic signatures to guide the use of chemotherapeutics

Anil Potti<sup>1,2</sup>, Holly K Dressman<sup>1,3</sup>, Andrea Bild<sup>1,3</sup>, Richard F Riedel<sup>1,2</sup>, Gina Chan<sup>4</sup>, Robyn Sayer<sup>4</sup>, Janel Cragun<sup>4</sup>, Hope Cottrill<sup>4</sup>, Michael J Kelley<sup>2</sup>, Rebecca Petersen<sup>5</sup>, David Harpole<sup>5</sup>, Jeffrey Marks<sup>5</sup>, Andrew Berchuck<sup>1,6</sup>, Geoffrey S Ginsburg<sup>1,2</sup>, Phillip Febbo<sup>1,2,3</sup>, Johnathan Lancaster<sup>4</sup> & Joseph R Nevins<sup>1,2,3</sup>

**Using *in vitro* drug sensitivity data coupled with Affymetrix microarray data, we developed gene expression signatures that predict sensitivity to individual chemotherapeutic drugs. Each signature was validated with response data from an independent set of cell line studies. We further show that many of these signatures can accurately predict clinical response in individuals treated with these drugs. Notably, signatures developed to predict response to individual agents, when combined, could also predict response to multidrug regimens. Finally, we integrated the chemotherapy response signatures with signatures of oncogenic pathway deregulation to identify new therapeutic strategies.** <http://www.nature.com/nm/journal/v12/n11/full/nm1491.html>

### ARTICLE LINKS

- ▶ Supplementary info

### ARTICLE TOOLS

- ✉ Send to a friend
- ✉ Export citation
- ✉ Export references
- ✉ Rights and permissions
- ✉ Order commercial reprints

### SEARCH PUBMED FOR

- ▶ Anil Potti
- ▶ Holly K Dressman
- ▶ Andrea Bild
- ▶ Richard F Riedel
- ▶ Robyn Sayer

# Stunning problems

## DERIVING CHEMOSENSITIVITY FROM CELL LINES: FORENSIC BIOINFORMATICS AND REPRODUCIBLE RESEARCH IN HIGH-THROUGHPUT BIOLOGY

BY KEITH A. BAGGERLY\* AND KEVIN R. COOMBES†

*U.T. M.D. Anderson Cancer Center*

High-throughput biological assays such as microarrays let us ask very detailed questions about how diseases operate, and promise to let us personalize therapy. Data processing, however, is often not described well enough to allow for exact reproduction of the results, leading to exercises in “forensic bioinformatics” where aspects of raw data and reported results are used to infer what methods must have been employed. Unfortunately, poor documentation can shift from an inconvenience to an active danger when it obscures not just methods but errors. In this report, we examine several related papers purporting to use microarray-based signatures of drug sensitivity derived from cell lines to predict patient response. Patients in clinical trials are currently being allocated to treatment arms on the basis of these results. However, we show in five case studies that the results incorporate several simple errors that may be putting patients at risk. One theme that emerges is that the most common errors are simple (e.g., row or column offsets); conversely, it is our experience that the most simple errors are common. We then discuss steps we are taking to avoid such errors in our own investigations.

<https://projecteuclid.org/euclid.aoas/1267453942>

# Timeline of events

From the article:

## Cancer trial errors revealed

**2006** Anil Potti, a cancer geneticist at Duke University in Durham, North Carolina, and others file patent applications on the idea of using gene-expression data to predict sensitivity to cancer drugs. Potti is first author on a paper in *Nature Medicine*<sup>1</sup>.

**2007** Potti is last author on a paper in the *Journal of Clinical Oncology* (*JCO*)<sup>2</sup>. Duke begins three clinical trials to test Potti's predictors in patients with breast or lung cancer.

**SEPTEMBER 2009** Keith Baggerly and Kevin Coombes, statisticians at the University of Texas M. D. Anderson Cancer Centre in Houston, publish a paper in *Annals of Applied Statistics*<sup>3</sup> stating that they could not replicate Potti's claims. Duke suspends the trials and asks a review panel to investigate.

**NOVEMBER 2009** Potti places data underlying the *JCO* paper online. Baggerly writes to Sally Kornbluth, Duke vice-dean for research, and Michael Cuffe, Duke vice-president for medical affairs, to point out differences from raw data.

**DECEMBER 2009** An unredacted copy of the report by Duke's review panel, later obtained by *Nature*, shows that the panel replicated Potti's claims using his data, but were unaware that those data contained discrepancies.

**JANUARY 2010** Duke restarts clinical trials.

**JULY 2010** *The Cancer Letter* reveals that Potti made false claims about his CV. Trials are suspended and an investigation begins. Harold Varmus, director of the National Cancer Institute in Bethesda, Maryland, asks the Institute of Medicine to review Duke's trials.

**NOVEMBER 2010** *JCO* paper is retracted. Duke closes the trials permanently. Potti resigns.

**DECEMBER 2010** Institute of Medicine study begins, but will now focus more generally on criteria for genomics predictor.

**JANUARY 2011** *Nature Medicine* paper is retracted.

<http://www.nature.com/news/2011/110111/full/469139a/box/1.html>

# Major fallout

Duke Lawsuit.pdf (page 1 of 90)

Previous Next Zoom Move Text Select Annotate Sidebar Search

NORTH CAROLINA DURHAM COUNTY IN THE GENERAL COURT OF JUSTICE  
DURHAM COUNTY FILED SEP 7 2011 SUPERIOR COURT DIVISION  
Richard Aiken, Jean K. Carroll, as Executor of the Estate of Harold G. Carroll, Jean K. Carroll, Individually, Peggy Cox, as Administratrix of the Estate of Paul F. Cox, Peggy Cox, Individually, Helene L. Fligel, Jason Gannon, as Personal Representative of the Estate of Jennifer L. Gannon, John Haddock, as Executor of the Estate of Karen Heath, Walter Jacobs, as Executor of the Estate of

413 2 M  
CLERK OF SUPERIOR COURT

COMPLAINT  
(JURY TRIAL DEMANDED)

[http://dig.abclocal.go.com/wtvd/docs/Duke\\_lawsuit\\_090811.pdf](http://dig.abclocal.go.com/wtvd/docs/Duke_lawsuit_090811.pdf)

<http://www.dukechronicle.com/articles/2015/05/03/duke-lawsuit-involving-cancer-patients-linked-anil-potti-settled>

# An interesting talk

## When is Reproducibility an Ethical Issue? Genomics, Personalized Medicine, and Human Error

Keith A. Baggerly

Bioinformatics and Computational Biology  
UT M. D. Anderson Cancer Center

[kabagg@mdanderson.org](mailto:kabagg@mdanderson.org)

BIRS Workshop, Aug 14, 2013



<http://www.birs.ca/events/2013/5-day-workshops/13w5083/videos/watch/201308141121-Baggerly.mp4>

# About me



How you feel about statistics

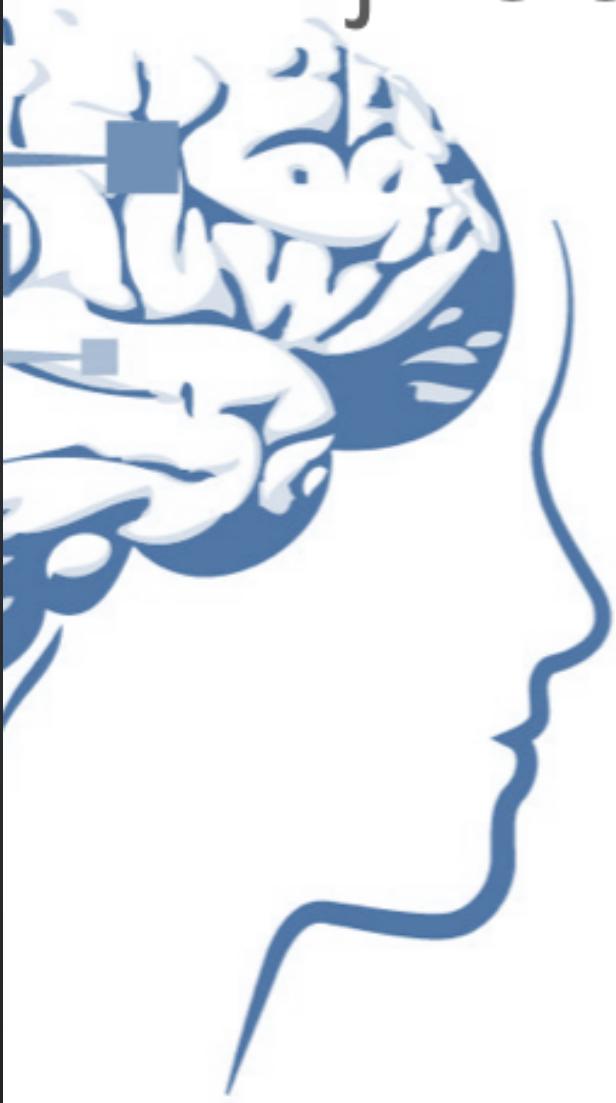


How I feel about statistics



@simplystats

[jhudatascience.org](http://jhudatascience.org)



# jtleek

Find me online @[jtleek](#),  
[@simplystats](#), Simply  
Statistics, and Github.

[Home](#)  
[Alumni](#)  
[Books](#)  
[Data](#)  
[Jobs](#)  
[Papers](#)  
[People](#)  
[Software](#)  
[Talks](#)  
[Teaching](#)

[www.jtleek.com](http://www.jtleek.com)

## Hi I'm Jeff

I work on figuring out how to go from raw data from next generation sequencing machines to results, turning public genomic data into clinically useful tools, and understanding how people use data analysis in real life.

I do [statistical research](#), write [data analysis software](#), [curate and create data sets](#), write a [blog about statistics](#), teach [people here at Hopkins](#), teach [a lot of people online](#), and work with [amazing students](#) who go [do awesome things](#). If you want to, come [do stuff with me](#)

If you want to keep up with everything we are working on, follow me on Twitter [@jtleek](#). The best way to contact me is my gmail account (I do not check my JHU email at all), or you can call me at my office **410-955-1166** (fair warning I have answered that phone ~3 times total since 2009), send me a fax **410-955-0958** (for real, fax is still a thing?!), or if you still use the pony express you could send me a letter at:

Johns Hopkins University  
Bloomberg School of Public Health  
Office E3624  
615 North Wolfe Street  
Baltimore, MD 21205-2179



# “Genomic data science”



# The Elements of Data Analytic Style



<https://leanpub.com/datastyle>

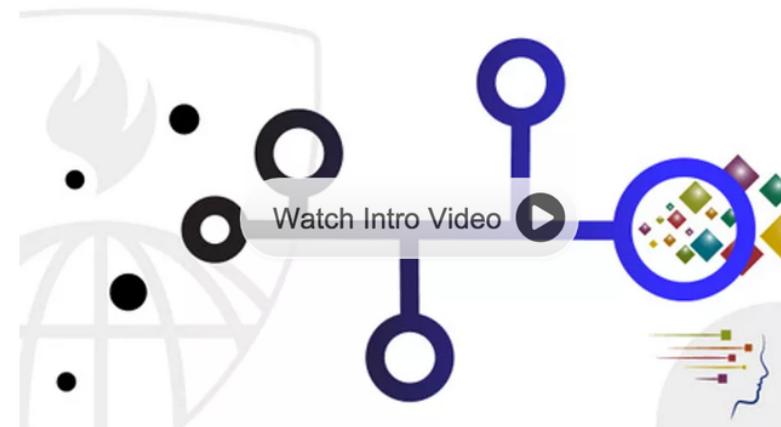
Jeff Leek



# Getting and Cleaning Data

Part of the [Data Science Specialization](#) »

Learn how to gather, clean, and manage data from a variety of sources.  
This is the third course in the Johns Hopkins Data Science Specialization.

[Edit Course Description](#)[Edit Session Descriptions](#) ▾[Edit Session Materials](#) ▾

## About the Course

Before you can work with data you have to get some. This course will cover the basic ways that data can be obtained. The course will cover obtaining data from the web, from APIs, from databases and from colleagues in various formats. It will also cover the basics of data cleaning and how to make data “tidy”. Tidy data dramatically speed downstream data analysis tasks. The course will also cover the components of a complete data set including raw data, processing instructions, codebooks, and processed data. The course will cover the basics needed for collecting, cleaning, and sharing data.

## Sessions

July 6, 2015 - August 1, 2015

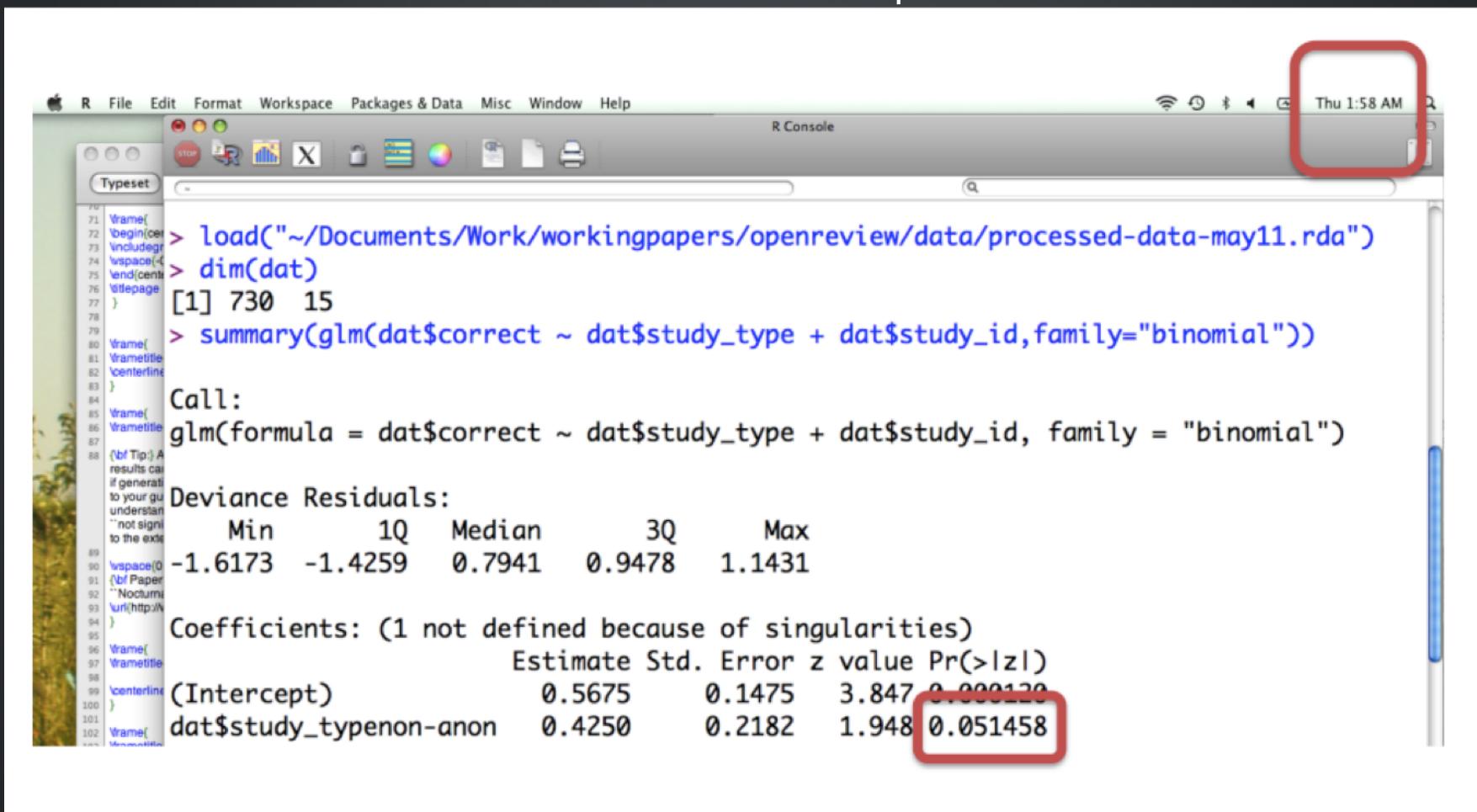
[Join Course](#)

## Eligible for

[Data Science Specialization](#)

This class

# What is the temptation?



```
R File Edit Format Workspace Packages & Data Misc Window Help
R Console
Thu 1:58 AM
```

```
> load("~/Documents/Work/workingpapers/openreview/data/processed-data-may11.rda")
> dim(dat)
[1] 730 15
> summary(glm(dat$correct ~ dat$study_type + dat$study_id,family="binomial"))

Call:
glm(formula = dat$correct ~ dat$study_type + dat$study_id, family = "binomial")

Deviance Residuals:
    Min      1Q  Median      3Q     Max 
-1.6173 -1.4259  0.7941  0.9478  1.1431 

Coefficients: (1 not defined because of singularities)
              Estimate Std. Error z value Pr(>|z|)    
(Intercept)   0.5675    0.1475  3.847 0.000120 ***
dat$study_typenon-anon 0.4250    0.2182  1.948 0.051458  

```

# Why this class?

## Abstract

Formula display:  **MathJax** [?](#)

## Background

Many groups, including our own, have proposed the use of DNA methylation profiles as biomarkers for various disease states. While much research has been done identifying DNA methylation signatures in cancer vs. normal etc., we still lack sufficient knowledge of the role that differential methylation plays during normal cellular differentiation and tissue specification. We also need thorough, genome level studies to determine the meaning of methylation of individual CpG dinucleotides in terms of gene expression.

## Results

In this study, we have used (insert statistical method here) to compile unique DNA methylation signatures from normal human heart, lung, and kidney using the Illumina Infinium 27 K methylation arrays and compared those to gene expression by RNA sequencing. We have identified unique signatures of global DNA methylation for human heart, kidney and liver, and showed that DNA methylation data can be used to correctly classify various tissues. It indicates that DNA methylation reflects tissue specificity and may play an important role in tissue differentiation. The integrative analysis of methylation and RNA-Seq data showed that gene methylation and its transcriptional levels were comprehensively correlated. The location of methylation markers in terms of distance to transcription start site and CpG island showed no effects on the regulation of gene expression by DNA methylation in normal tissues.

# Why this class?

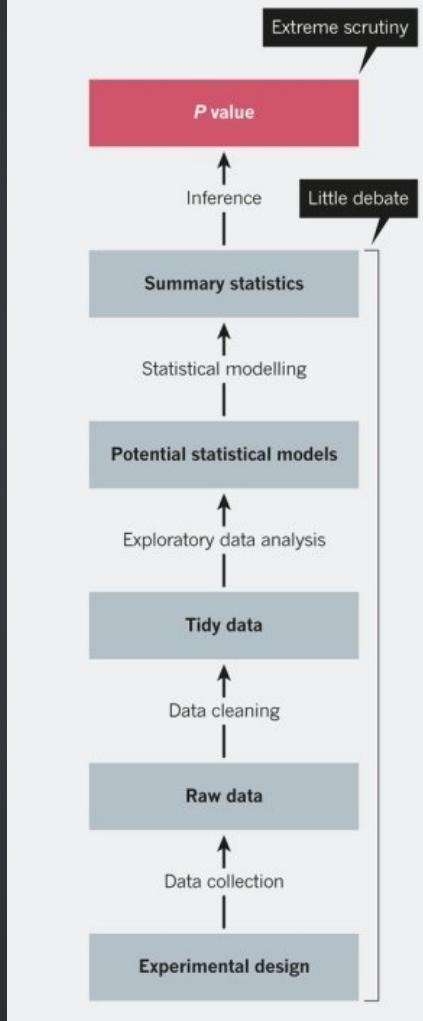


Emma, please insert NMR data here! where are they? and for this compound, just make up an elemental analysis...

<http://pubs.acs.org/doi/abs/10.1021/om4000067>

## DATA PIPELINE

The design and analysis of a successful study has many stages, all of which need policing.



# Why this class?

- Most of the attention is on the last step
- This course is about all the steps that come before
- They are *critical* for getting things right

Leek and Peng (2015) Nature

# Researcher degrees of freedom

*General Article*

## **False-Positive Psychology: Undisclosed Flexibility in Data Collection and Analysis Allows Presenting Anything as Significant**

**Joseph P. Simmons<sup>1</sup>, Leif D. Nelson<sup>2</sup>, and Uri Simonsohn<sup>1</sup>**

<sup>1</sup>The Wharton School, University of Pennsylvania, and <sup>2</sup>Haas School of Business, University of California, Berkeley



Psychological Science  
XX(X) 1–8  
© The Author(s) 2011  
Reprints and permission:  
[sagepub.com/journalsPermissions.nav](http://sagepub.com/journalsPermissions.nav)  
DOI: 10.1177/0956797611417632  
<http://pss.sagepub.com>  
SAGE

<http://pss.sagepub.com/content/22/11/1359.abstract>



Herein lies the dirty secret about most data scientists' work -- it's more data munging than deep learning. The best minds of my generation are deleting commas from log files, and that makes me sad. A Ph.D. is a terrible thing to waste.

<http://adage.com/article/digitalnext/dear-madison-avenue-set-data-scientists-free/298676/>



TECHNOLOGY | For Big-Data Scientists, 'Janitor Work' Is Key Hurdle to Insights



TECHNOLOGY

# For Big-Data Scientists, 'Janitor Work' Is Key Hurdle to Insights

By STEVE LOHR AUG. 17, 2014



amazon webservices | intel CLOUD INSIGHTS

Why Novartis is Looking Beyond On-Premises... [READ >](#)

Case Study: Cloud Supercomputing from AWS Powers... [READ >](#)

**Get Started with AWS**

**CREATE A FREE ACCOUNT >**

<http://www.nytimes.com/2014/08/18/technology/for-big-data-scientists-hurdle-to-insights-is-janitor-work.html>

# What you wished data looked like

# What it actually looks like

```
@HWI-EAS121:4:100:1783:550#0/1
CGTTACGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACGGATCTCGTATGCCGTCTGCCTCGTGACAAGACAGGGG
+HWI-EAS121:4:100:1783:550#0/1
aaaaaa`b_aa`aa`YaX]aZ`aZM^Z]YRa]YSG[[ZREQLHESDHNDDHNMEEDDM PENITKFLFEEDDHEJQMEDDD
@HWI-EAS121:4:100:1783:1611#0/1
GGGTGGGCATTCCACTCGCAGTATGGGTTGCCGCACGACAGGCAGCGGT CAGCCTGCCTTGGCCTGGCCTTCGGAAA
+HWI-EAS121:4:100:1783:1611#0/1
a``^\\_ ``~~~a``a``a_``]a_``]`a____`_``^`]X]_`XTV_``]NX_XVX``]_TTTG[VTHPN]VFDZ
@HWI-EAS121:4:100:1783:322#0/1
CGTTTATGTTTGAATATGTCTTATCTAACGGTTATTTAGATGTTGGTCTTATTCTAACGGTCATATATTTCTA
+HWI-EAS121:4:100:1783:322#0/1
abaa``aaaaabbbaabbbbbbb`bbbb_bbbbbbbb`bbbaV``a``a``]``aT]a__V\\1]_``]a`]a_abbaV__
@HWI-EAS121:4:100:1783:1394#0/1
GGGTCTTATTGGTCTGGT GATCCCCCATATTCTCCGGTTGTTAACCGATCATCGGCATTACTCCGGCTGC
+HWI-EAS121:4:100:1783:1394#0/1
````[aa\b``[ ]abb][`a_abbb`a``bbbbbabaaaab_Vza_``bab_X`[a\HV_[_][^_X\T_VQQ
@HWI-EAS121:4:100:1783:207#0/1
CCCTGGAGATCGGAAGAGCGGTTCAGCAGGAATGCCGAGACGGATCTCGTATGCCGTCTGCTTGGAAAAAAACA
+HWI-EAS121:4:100:1783:207#0/1
abba`Xa``\\`aa]ba__bba[a_O_a`aa`aa`a]^V]X_a^YS\R_\H_[]\ZTDUZZUSOPX]]POP\GS\WSHHD
@HWI-EAS121:4:100:1783:455#0/1
GGGTAAATTCAAGGACAAATGTAATGGCTGCACAAAAAAATACATCTTCATGTTCCATTGCACCATTGACAAATACATATT
+HWI-EAS121:4:100:1783:455#0/1
abb_babbabaabbbbbbbbbbbaabbba`bb`ab_0_bab_Q_bbabaa_a
`
```

[http://brianknaus.com/software/srtoolbox/s\\_4\\_1\\_sequence80.txt](http://brianknaus.com/software/srtoolbox/s_4_1_sequence80.txt)

# What it actually looks like

The screenshot shows a web browser window with the URL [https://dev.twitter.com/docs/api/1/get\(blocks/blocking](https://dev.twitter.com/docs/api/1/get(blocks/blocking)) in the address bar. The page is titled "GET blocks/blocking | Twitter API". The main content area displays the API documentation for the GET blocks/blocking endpoint, including example requests and responses.

**Example Request**

GET [https://api.twitter.com/1/blocks/blocking.json?cursor=-1&include\\_entities=true](https://api.twitter.com/1/blocks/blocking.json?cursor=-1&include_entities=true)

```
1. {
2.   "previous_cursor": 0,
3.   "previous_cursor_str": "0",
4.   "next_cursor": 0,
5.   "users": [
6.     {
7.       "profile_sidebar_border_color": "C0DEED",
8.       "name": "Javier Heady \r",
9.       "profile_sidebar_fill_color": "DDEEF6",
10.      "profile_background_tile": false,
11.      "location": null,
12.      "created_at": "Thu Mar 01 00:16:47 +0000 2012",
13.      "profile_image_url":
14.        "http://a0.twimg.com/sticky/default_profile_images/default_profile_4_normal.png",
15.      "is_translator": false,
16.      "id_str": "509466276",
17.      "profile_link_color": "0084B4",
18.      "follow_request_sent": false,
19.      "contributors_enabled": false,
20.      "default_profile": true,
21.      "url": null,
22.      "favourites_count": 0,
23.      "utc_offset": null,
24.      "id": 509466276,
25.      "profile_image_url_https":
26.        "https://si0.twimg.com/sticky/default_profile_images/default_profile_4_normal.png",
27.      "listed_count": 0,
28.      "profile_use_background_image": true,
29.      "profile_text_color": "333333",
30.      "lang": "en",
31.      "protected": false,
```

**https://dev.twitter.com/docs/api/1/get(blocks/blocking**

# What it actually looks like

ALLERGIES		MEDICATION HISTORY
Last Updated: 01 Dec 2011 @ 0851		Last Updated: 11 Apr 2011 @ 1737
Allergy Name:	TRIMETHOPRIM	Medication: AMLODIPINE BESYLATE 10MG TAB
Location:	DAYT29	Instructions: TAKE ONE TABLET BY MOUTH TAKE ON GRAPEFRUIT JUICE--
Date Entered:	09 Mar 2011	Status: Active
Action:		Refills Remaining: 3
Allergy Type:	DRUG	Last Filled On: 20 Aug 2010
A Drug Class:	ANTI-INFECTIVES, OTHER	Initially Ordered On: 13 Aug 2010
Observed/Historical:	HISTORICAL	Quantity: 45
Comments:	The reaction to this allergy was MILD (NO SQUELAE)	Days Supply: 90
Allergy Name:	TRAMADOL	Pharmacy: DAYTON
Location:	DAYT29	Prescription Number: 2718953
Date Entered:	09 Mar 2011	Medication: IBUPROFEN 600MG TAB
Action:	URINARY RETENTION	Instructions: TAKE ONE TABLET BY MOUTH FOUR TI
Allergy Type:	DRUG	Status: Active
A Drug Class:	NON-OPIOID ANALGESICS	Refills Remaining: 3
Observed/Historical:	HISTORICAL	Last Filled On: 20 Aug 2010
Comments:	gradually worsening difficulty emptying bladder	Initially Ordered On: 01 Jul 2010

<http://blue-button.github.com/challenge/>

# #otherpeoplesdata

Home Notifications Messages  #otherpeoplesdata   

## #otherpeoplesdata

Top | Live | Accounts | Photos | Videos | More options ▾

Who to follow · Refresh · View all

-  Joseph N. Paulson @dorageh Followed by Hector Corrada ... 
-  RStudio @rstudio 
-  One R Tip a Day @RLangTip 

Find friends

Trends · Change

**#IndependenceDay**  
For At Least 4,000 Immigrants, This Independence Day Has Special...  
283K Tweets about this trend

**Happy 4th**  
Happy 4th: EPA touts 43-year-old DDT ban | WashingtonExaminer.com  
2.01M Tweets about this trend

**4th of July**  
Pizzas to be served to troops in Afghanistan

Patrick Durusau @patrickDurusau · Jul 1  
RP The challenge of combining 176 x #otherpeoplesdata... #integration  
[#opendata](#) [ow.ly/OSFGs](#)

Martin Bentley @astonsplat · Jun 30  
#OtherPeoplesData

Matthew @MCeeP  
Most helpful data column ever

Legacy4Life, A Theft @hieegg · Jun 27  
#hieegg hiegg.com #legacyforlife #PAYBACK ME YOUR #fraud #thieves of #otherpeoplesdata #HellboyIII da, #Legacy42 #legacy769

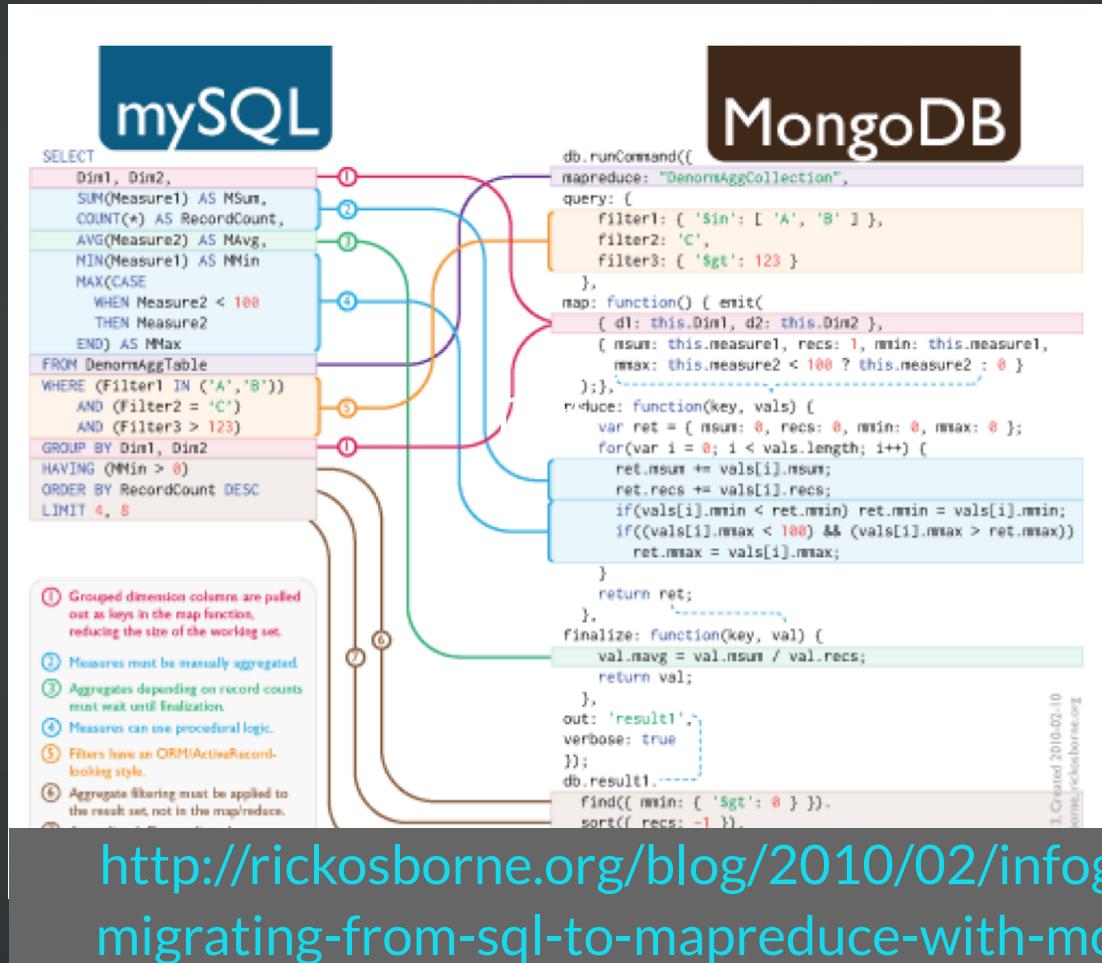
Katie Mack follows

Morgan Jackson @BioInFocus · Jun 25  
@thelabandfield @cbahla1 Also raises a curious/terrifying new usage for #OtherPeoplesData

Where you wish data was



# Where is data?



# Where is data?

The screenshot shows a web browser window with the following details:

- Title Bar:** GET blocks/blocking | Twitter
- URL:** https://dev.twitter.com/docs/api/1/get(blocks/blocking)
- Header:** Developers, Search, API Health, Blog, Discussions, Documentation, Sign in
- Content:**
  - Example Request:** GET https://api.twitter.com/1/blocks/blocking.json?cursor=-1&include\_entities=true
  - Text Example:** A large block of JSON code representing a user profile, starting with line 1 and ending at line 30.
  - Link:** https://dev.twitter.com/docs/api/1/get(blocks/blocking)

# Where is data?

The screenshot shows a web browser window for the Open Baltimore beta data portal at <https://data.baltimorecity.gov>. The page features a large banner image of a computer monitor displaying a cityscape. Overlaid on the banner are binary code patterns and the City of Baltimore seal. The navigation menu includes Home, Residents, Business, Visitors, Government, Office of the Mayor, Help, Sign Up, and Sign In. A central callout box encourages user feedback. Below the main content are three preview boxes showing sample data from different datasets.

**We Want Your Feedback!**

Have suggestions for a dataset? Please take a moment and visit the suggestion page at the bottom of this page. Or you can click the feedback tab to the left and join the discussion over at the forums. See you there!

Brought to you by  
CITY OF BALTIMORE  
STEPHANIE RAWLINGS-BLAKE  
MAYOR

id	propertyAddress	id	category
1	1000 University St	1	Residential
2	1400 University St	2	Residential
3	1401 University St	3	Residential
4	1402 University St	4	Residential
5	1403 University St	5	Residential
6	1404 University St	6	Residential
7	1405 University St	7	Residential
8	1406 University St	8	Residential
9	1407 University St	9	Residential
10	1408 University St	10	Residential
11	1409 University St	11	Residential
12	1410 University St	12	Residential
13	1411 University St	13	Residential
14	1412 University St	14	Residential
15	1413 University St	15	Residential
16	1414 University St	16	Residential
17	1415 University St	17	Residential
18	1416 University St	18	Residential
19	1417 University St	19	Residential
20	1418 University St	20	Residential
21	1419 University St	21	Residential
22	1420 University St	22	Residential
23	1421 University St	23	Residential
24	1422 University St	24	Residential
25	1423 University St	25	Residential
26	1424 University St	26	Residential
27	1425 University St	27	Residential
28	1426 University St	28	Residential
29	1427 University St	29	Residential
30	1428 University St	30	Residential
31	1429 University St	31	Residential
32	1430 University St	32	Residential
33	1431 University St	33	Residential
34	1432 University St	34	Residential
35	1433 University St	35	Residential
36	1434 University St	36	Residential
37	1435 University St	37	Residential
38	1436 University St	38	Residential
39	1437 University St	39	Residential
40	1438 University St	40	Residential
41	1439 University St	41	Residential
42	1440 University St	42	Residential
43	1441 University St	43	Residential
44	1442 University St	44	Residential
45	1443 University St	45	Residential
46	1444 University St	46	Residential
47	1445 University St	47	Residential
48	1446 University St	48	Residential
49	1447 University St	49	Residential
50	1448 University St	50	Residential
51	1449 University St	51	Residential
52	1450 University St	52	Residential
53	1451 University St	53	Residential
54	1452 University St	54	Residential
55	1453 University St	55	Residential
56	1454 University St	56	Residential
57	1455 University St	57	Residential
58	1456 University St	58	Residential
59	1457 University St	59	Residential
60	1458 University St	60	Residential
61	1459 University St	61	Residential
62	1460 University St	62	Residential
63	1461 University St	63	Residential
64	1462 University St	64	Residential
65	1463 University St	65	Residential
66	1464 University St	66	Residential
67	1465 University St	67	Residential
68	1466 University St	68	Residential
69	1467 University St	69	Residential
70	1468 University St	70	Residential
71	1469 University St	71	Residential
72	1470 University St	72	Residential
73	1471 University St	73	Residential
74	1472 University St	74	Residential
75	1473 University St	75	Residential
76	1474 University St	76	Residential
77	1475 University St	77	Residential
78	1476 University St	78	Residential
79	1477 University St	79	Residential
80	1478 University St	80	Residential
81	1479 University St	81	Residential
82	1480 University St	82	Residential
83	1481 University St	83	Residential
84	1482 University St	84	Residential
85	1483 University St	85	Residential
86	1484 University St	86	Residential
87	1485 University St	87	Residential
88	1486 University St	88	Residential
89	1487 University St	89	Residential
90	1488 University St	90	Residential
91	1489 University St	91	Residential
92	1490 University St	92	Residential
93	1491 University St	93	Residential
94	1492 University St	94	Residential
95	1493 University St	95	Residential
96	1494 University St	96	Residential
97	1495 University St	97	Residential
98	1496 University St	98	Residential
99	1497 University St	99	Residential
100	1498 University St	100	Residential
101	1499 University St	101	Residential
102	1400 University St	102	Residential
103	1401 University St	103	Residential
104	1402 University St	104	Residential
105	1403 University St	105	Residential
106	1404 University St	106	Residential
107	1405 University St	107	Residential
108	1406 University St	108	Residential
109	1407 University St	109	Residential
110	1408 University St	110	Residential
111	1409 University St	111	Residential
112	1410 University St	112	Residential
113	1411 University St	113	Residential
114	1412 University St	114	Residential
115	1413 University St	115	Residential
116	1414 University St	116	Residential
117	1415 University St	117	Residential
118	1416 University St	118	Residential
119	1417 University St	119	Residential
120	1418 University St	120	Residential
121	1419 University St	121	Residential
122	1420 University St	122	Residential
123	1421 University St	123	Residential
124	1422 University St	124	Residential
125	1423 University St	125	Residential
126	1424 University St	126	Residential
127	1425 University St	127	Residential
128	1426 University St	128	Residential
129	1427 University St	129	Residential
130	1428 University St	130	Residential
131	1429 University St	131	Residential
132	1430 University St	132	Residential
133	1431 University St	133	Residential
134	1432 University St	134	Residential
135	1433 University St	135	Residential
136	1434 University St	136	Residential
137	1435 University St	137	Residential
138	1436 University St	138	Residential
139	1437 University St	139	Residential
140	1438 University St	140	Residential
141	1439 University St	141	Residential
142	1440 University St	142	Residential
143	1441 University St	143	Residential
144	1442 University St	144	Residential
145	1443 University St	145	Residential
146	1444 University St	146	Residential
147	1445 University St	147	Residential
148	1446 University St	148	Residential
149	1447 University St	149	Residential
150	1448 University St	150	Residential
151	1449 University St	151	Residential
152	1450 University St	152	Residential
153	1451 University St	153	Residential
154	1452 University St	154	Residential
155	1453 University St	155	Residential
156	1454 University St	156	Residential
157	1455 University St	157	Residential
158	1456 University St	158	Residential
159	1457 University St	159	Residential
160	1458 University St	160	Residential
161	1459 University St	161	Residential
162	1460 University St	162	Residential
163	1461 University St	163	Residential
164	1462 University St	164	Residential
165	1463 University St	165	Residential
166	1464 University St	166	Residential
167	1465 University St	167	Residential
168	1466 University St	168	Residential
169	1467 University St	169	Residential
170	1468 University St	170	Residential
171	1469 University St	171	Residential
172	1470 University St	172	Residential
173	1471 University St	173	Residential
174	1472 University St	174	Residential
175	1473 University St	175	Residential
176	1474 University St	176	Residential
177	1475 University St	177	Residential
178	1476 University St	178	Residential
179	1477 University St	179	Residential
180	1478 University St	180	Residential
181	1479 University St	181	Residential
182	1480 University St	182	Residential
183	1481 University St	183	Residential
184	1482 University St	184	Residential
185	1483 University St	185	Residential
186	1484 University St	186	Residential
187	1485 University St	187	Residential
188	1486 University St	188	Residential
189	1487 University St	189	Residential
190	1488 University St	190	Residential
191	1489 University St	191	Residential
192	1490 University St	192	Residential
193	1491 University St	193	Residential
194	1492 University St	194	Residential
195	1493 University St	195	Residential
196	1494 University St	196	Residential
197	1495 University St	197	Residential
198	1496 University St	198	Residential
199	1497 University St	199	Residential
200	1498 University St	200	Residential
201	1499 University St	201	Residential
202	1400 University St	202	Residential
203	1401 University St	203	Residential
204	1402 University St	204	Residential
205	1403 University St	205	Residential
206	1404 University St	206	Residential
207	1405 University St	207	Residential
208	1406 University St	208	Residential
209	1407 University St	209	Residential
210	1408 University St	210	Residential
211	1409 University St	211	Residential
212	1410 University St	212	Residential
213	1411 University St	213	Residential
214	1412 University St	214	Residential
215	1413 University St	215	Residential
216	1414 University St	216	Residential
217	1415 University St	217	Residential
218	1416 University St	218	Residential
219	1417 University St	219	Residential
220	1418 University St	220	Residential
221	1419 University St	221	Residential
222	1420 University St	222	Residential
223	1421 University St	223	Residential
224	1422 University St	224	Residential
225	1423 University St	225	Residential
226	1424 University St	226	Residential
227	1425 University St	227	Residential
228	1426 University St	228	Residential
229	1427 University St	229	Residential
230	1428 University St	230	Residential
231	1429 University St	231	Residential
232	1430 University St	232	Residential
233	1431 University St	233	Residential
234	1432 University St	234	Residential
235	1433 University St	235	Residential
236	1434 University St	236	Residential
237	1435 University St	237	Residential
238	1436 University St	238	Residential
239	1437 University St	239	Residential
240	1438 University St	240	Residential
241	1439 University St	241	Residential
242	1440 University St	242	Residential
243	1441 University St	243	Residential
244	1442 University St	244	Residential
245	1443 University St	245	Residential
246	1444 University St	246	Residential
247	1445 University St	247	Residential
248	1446 University St	248	Residential
249	1447 University St	249	Residential
250	1448 University St	250	Residential
251	1449 University St	251	Residential
252	1450 University St	252	Residential
253	1451 University St	253	Residential
254	1452 University St	254	Residential
255	1453 University St	255	Residential
256	1454 University St	256	Residential
257	1455 University St	257	Residential
258	1456 University St	258	Residential
259	1457 University St	259	Residential
260	1458 University St	260	Residential
261	1459 University St	261	Residential
262	1460 University St	262	Residential
263	1461 University St	263	Residential
264	1462 University St	264	Residential
265	1463 University St	265	Residential
266	1464 University St	266	Residential
267	1465 University St	267	Residential
268	1466 University St	268	Residential
269	1467 University St	269	Residential
270	1468 University St	270	Residential
271	1469 University St	271	Residential
272	1470 University St	272	Residential
273	1471 University St	273	Residential
274	1472 University St	274	Residential
275	1473 University St	275	Residential
276	1474 University St	276	Residential
277	1475 University St	277	Residential
278	1476 University St	278	Residential
279	1477 University St	279	Residential
280	1478 University St	280	Residential
281	1479 University St	281	Residential
282	1480 University St	282	Residential
283	1481 University St	283	Residential
284	1482 University St	284	Residential
285	1483 University St	285	Residential
286	1484 University St	286	Residential
287	1485 University St	287	Residential
288	1486 University St	288	Residential
289	1487 University St	289	Residential
290	1488 University St	290	Residential
291	1489 University St	291	Residential
292	1490 University St	292	Residential
293	1491 University St	293	Residential
294	1492 University St	294	Residential
295	1493 University St	295	Residential
296	1494 University St	296	Residential
297	1495 University St	297	Residential
298	1496 University St	298	Residential
299	1497 University St	299	Residential
300	1498 University St	300	Residential
301	1499 University St	301	Residential
302	1400 University St	302	Residential
303	1401 University St	303	Residential
304	1402 University St	304	Residential
305	1403 University St	305	Residential
306	1404 University St	306	Residential
307	1405 University St	307	Residential
308	1406 University St	308	Residential
309	1407 University St	309	Residential
310	1408 University St	310	Residential
311	1409 University St	311	Residential
312	1410 University St	312	Residential
313	1411 University St	313	Residential
314	1412 University St	314	Residential
315	1413 University St	315	Residential
316	1414 University St	316	Residential
317	1415 University St	317	Residential
318	1416 University St	318	Residential
319	1417 University St	319	Residential
320	1418 University St	320	Residential
32			

**GET ALL THE  
DATA!!!**



# Data

# Source

# Brainstorming

<http://bit.ly/1FWssE2>

# Tools

# The workhorse



The screenshot shows the homepage of the R Project for Statistical Computing. The page features a large R logo on the left, followed by a navigation menu with links to Home, Download, CRAN, R Project, and R Foundation. The main content area has a large title "The R Project for Statistical Computing" and a "Getting Started" section. Below the title, there is a paragraph about R being a free software environment for statistical computing and graphics, mentioning its availability on various platforms and the option to download from CRAN mirrors. There is also a link to frequently asked questions. The "News" section lists several recent releases and events, including "The R Journal Volume 7/1" and several R versions released in 2015 and 2014.

**The R Project for Statistical Computing**

**Getting Started**

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

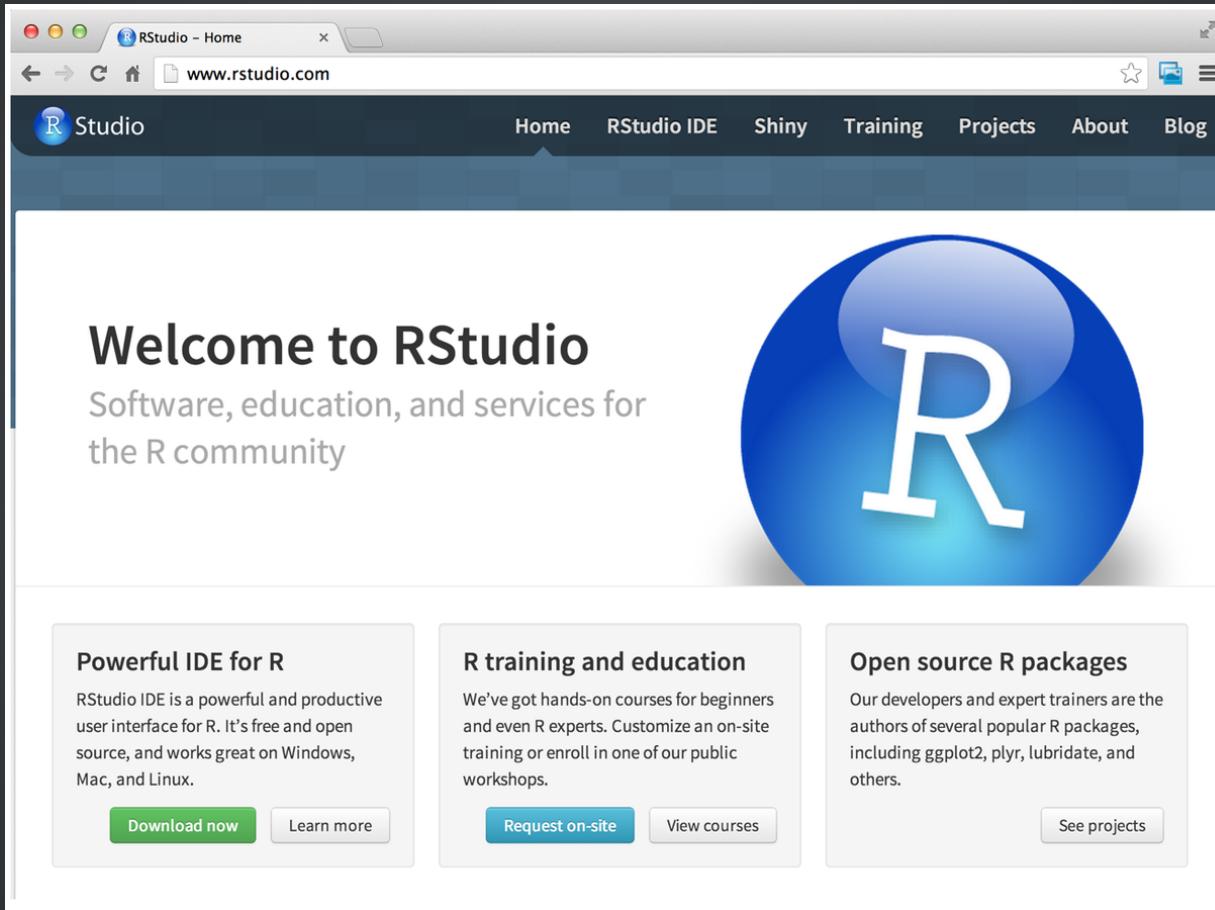
If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

**News**

- [The R Journal Volume 7/1](#) is available.
- [R version 3.2.1 \(World-Famous Astronaut\)](#) has been released on 2015-06-18.
- [R version 3.1.3 \(Smooth Sidewalk\)](#) has been released on 2015-03-09.
- [useR! 2015](#), will take place at the University of Aalborg, Denmark, June 30 - July 3, 2015.
- [useR! 2014](#), took place at the University of California, Los Angeles, USA June 30 - July 3, 2014.

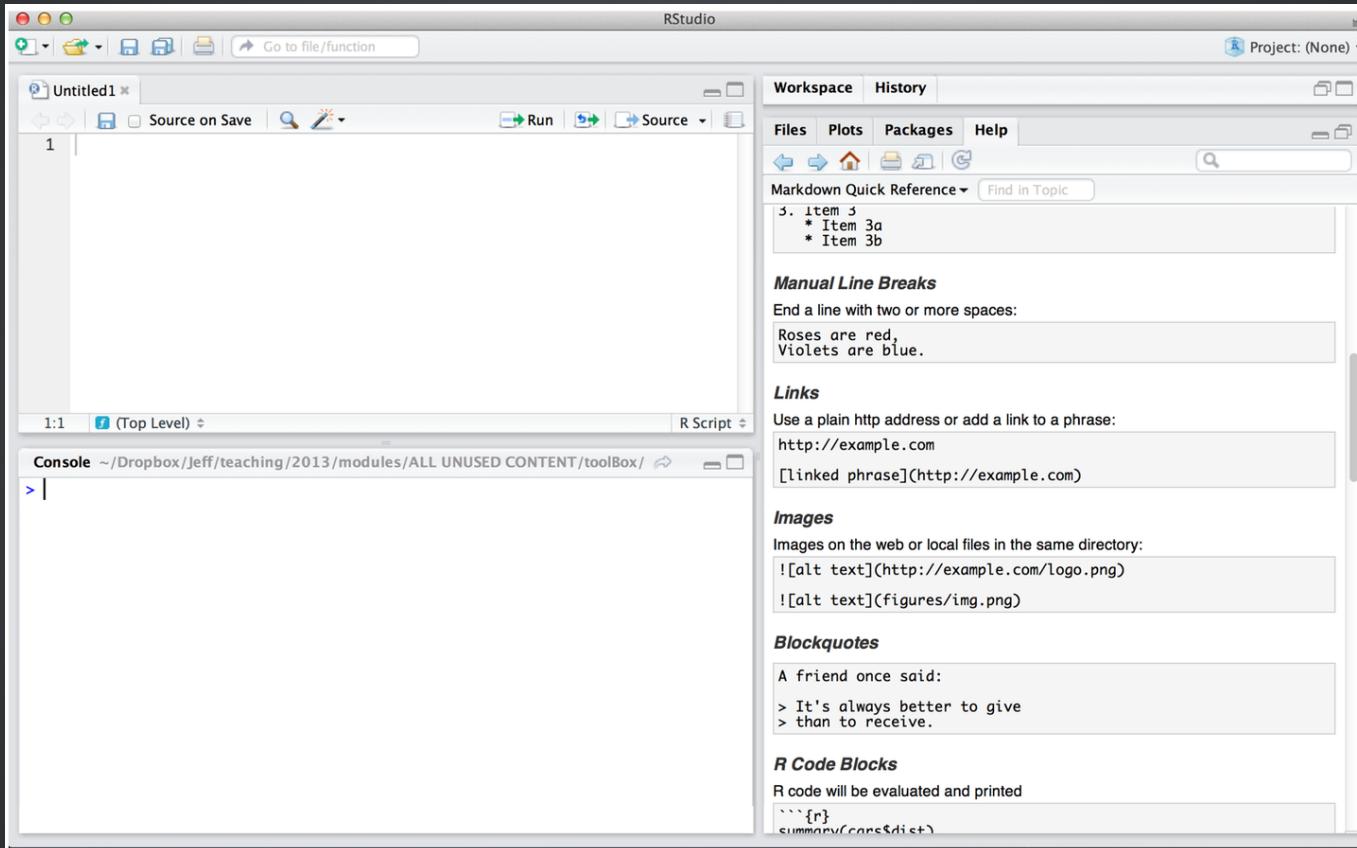
<http://www.r-project.org/>

# A good coding environment

A screenshot of a web browser displaying the RStudio homepage. The browser window has a title bar "RStudio - Home" and a URL bar "www.rstudio.com". The main content area shows the RStudio logo (a large blue circle with a white 'R') and the text "Welcome to RStudio: Software, education, and services for the R community". Below this are three main sections: "Powerful IDE for R", "R training and education", and "Open source R packages", each with descriptive text and call-to-action buttons.

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

# Rstudio Interface



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

# Some useful commands

**Cmd + Enter**

Evaluates line of code (Mac)

**Ctrl + Enter**

Evaluates line of code (Windows)

**Ctrl + 1**

Switch to script page

**Ctrl + 2**

Switch to console

# Rstudio

# Tour

<http://bit.ly/1Rdf+K1>

# R packages



[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](#)

[A3](#)  
[abbyyR](#)  
[abc](#)  
[ABCanalysis](#)  
[abc.data](#)  
[abcdeFBA](#)  
[ABCOptim](#)  
[abctools](#)  
[abd](#)  
[abf2](#)  
[abind](#)  
[abn](#)  
[abundant](#)  
[acc](#)  
[accelerometry](#)  
[AcceptanceSampling](#)  
[ACCLMA](#)  
[accrual](#)  
[accrued](#)  
[ACD](#)  
[acepack](#)

## Available CRAN Packages By Name

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

A3: Accurate, Adaptable, and Accessible Error Metrics for Predictive Models  
Access to Abbyy Optical Character Recognition (OCR) API  
Tools for Approximate Bayesian Computation (ABC)  
Computed ABC Analysis  
Data Only: Tools for Approximate Bayesian Computation (ABC)  
ABCDE\_FBA: A-Biologist-Can-Do-Everything of Flux Balance Analysis with this package  
Implementation of Artificial Bee Colony (ABC) Optimization  
Tools for ABC Analyses  
The Analysis of Biological Data  
Load Gap-Free Axon ABF2 Files  
Combine Multidimensional Arrays  
Data Modelling with Additive Bayesian Networks  
Abundant regression and high-dimensional principal fitted components  
A Package to Processes Accelerometer Data  
Functions for Processing Minute-to-Minute Accelerometer Data  
Creation and evaluation of Acceptance Sampling Plans  
ACC & LMA Graph Plotting  
Bayesian Accrual Prediction  
Data Quality Visualization Tools for Partially Accruing Data  
Categorical data analysis with complete or missing responses  
ace() and avas() for selecting regression transformations

```
install.packages("devtools")  
install.packages("dplyr")
```

# All Packages

## Bioconductor version 3.1 (Release)

Autocomplete biocViews search:

Software (1024)

- ▶ AssayDomain (345)
- ▶ BiologicalQuestion (313)
- ▶ Infrastructure (211)
- ▶ ResearchField (225)
- ▶ StatisticalMethod (293)
- ▶ Technology (645)
- ▶ WorkflowStep (525)
- ▶ AnnotationData (883)
- ▶ ExperimentData (241)

## Packages found under Software:

Show All  entries

Search table:

Package	Maintainer	Title
<a href="#">a4</a>	Tobias Verbeke, Willem Ligtenberg	Automated Affymetrix Array Analysis Umbrella Package
<a href="#">a4Base</a>	Tobias Verbeke, Willem Ligtenberg	Automated Affymetrix Array Analysis Base Package
<a href="#">a4Classif</a>	Tobias Verbeke, Willem Ligtenberg	Automated Affymetrix Array Analysis Classification Package
<a href="#">a4Core</a>	Tobias Verbeke, Willem Ligtenberg	Automated Affymetrix Array Analysis Core Package
<a href="#">a4Preproc</a>	Tobias Verbeke, Willem Ligtenberg	Automated Affymetrix Array Analysis Preprocessing Package
<a href="#">a4Reporting</a>	Tobias Verbeke, Willem Ligtenberg	Automated Affymetrix Array Analysis Reporting Package
<a href="#">ABarray</a>	Yongming Andrew Sun	Microarray QA and statistical data analysis for Applied Biosystems Genome Survey Microarray (AB1700) gene expression data.
		ABSeq: a new RNA-Seq analysis method based on differences and el
<a href="#">aCGH</a>	Peter Dimitrov	Classes and functions for Array Comparative Genomic Hybridization data.

<http://bioconductor.org/>

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



dgrtwo / broom

Watch

16



Convert statistical analysis objects from R into tidy format

146 commits

1 branch

8 releases

10 contributors



branch: master ▾

broom / +

Merge pull request #51 from zeehio/master ...



dgrtwo authored 3 hours ago

latest commit ec5c0bd980



R Merge pull request #51 from zeehio/master 3 hours ago



man-roxygen Overhaul of how augmenting works across many objects. In particular t... 7 months ago



man Add a `tidy` method for x,y,z lists 21 days ago



tests Changed `rowwise\_df\_tidiers` to allow the original data to be saved a... a month ago



vignettes Added `gam` to README. Removed rownames from glmnet output. Few typo ... 7 months ago



.Rbuildignore Update cran comments. 6 months ago



.gitignore 6 months ago



DESCRIPTION Merge pull request #51 from zeehio/master 3 hours ago

<https://github.com/dgrtwo/broom>

```
devtools::install_github("broom")
```

# Average Trustworthiness

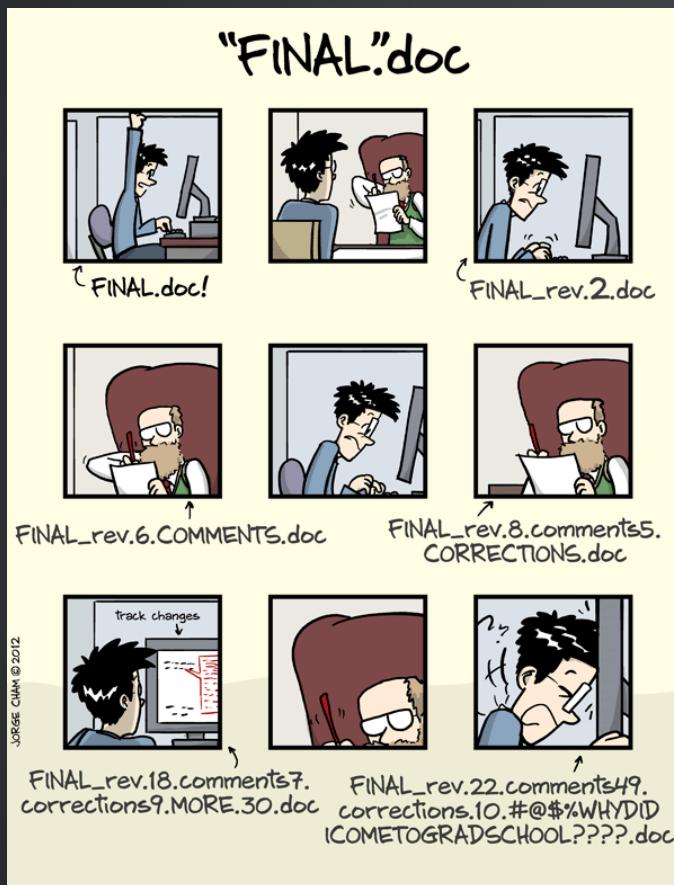


# Installing Packages

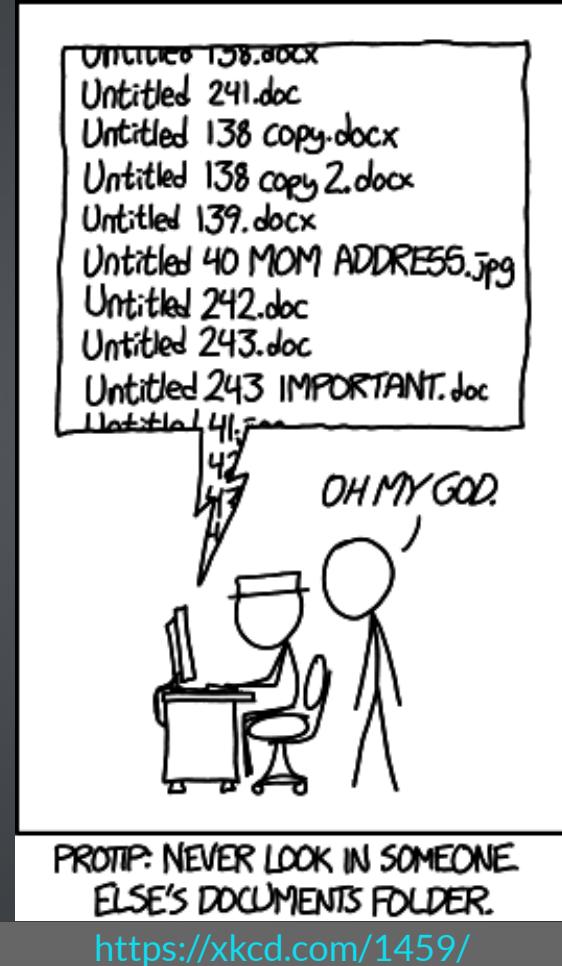
<http://bit.ly/1CdyISm>

# Version Control

# Just. No.



<http://www.phdcomics.com/comics/archive.php?comicid=1531>



# Unfortunate truth



**Michael Cook**

@mtrc

 Follow

"Version control is a truly vital concept that has unfortunately been implemented by madmen." Amen.  
[twitter.com/Pentadact/stat...](https://twitter.com/Pentadact/status/617075570761965568)

5:00 PM - 3 Jul 2015



9



11

<https://twitter.com/mtrc/status/617075570761965568>

# Git

 **git** --fast-version-control

[About](#)  
[Documentation](#)  
[Blog](#)  
**Downloads**  
  GUI Clients  
  Logos  
[Community](#)

The entire [Pro Git book](#) written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

[Search entire site...](#)

## Downloads

 Mac OS X    Windows  
 Linux    Solaris

Older releases are available and the [Git source repository](#) is on GitHub.

### GUI Clients

Git comes with built-in GUI tools (`git-gui`, `gitk`), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)



Latest source Release  
**2.4.5**  
Release Notes (2015-06-25)  
[Downloads for Mac](#)

### Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos →](#)

<http://git-scm.com/downloads>

# Github

A screenshot of a web browser displaying the GitHub homepage. The URL in the address bar is <https://github.com>. The page features a dark background with various white and light gray icons related to software development and collaboration. On the left, there is a large, bold text overlay that reads "Build software better, together.". Below this, a smaller text block says "Powerful collaboration, code review, and code management for open source and private projects. Need private repositories? Upgraded plans start at \$7/mo." To the right, there is a sign-up form with three input fields: "Pick a username", "Your email", and "Create a password". A note below the fields states "Use at least one lowercase letter, one numeral, and seven characters." A green "Sign up for GitHub" button is located below the password field. At the bottom of the page, a footer bar contains the GitHub logo and the text "https://github.com/".

GitHub · Build software better, together.

Explore Features Enterprise Blog

Sign up Sign in

# Build software better, together.

Powerful collaboration, code review, and code management for open source and private projects. Need private repositories? Upgraded plans start at \$7/mo.

Pick a username

Your email

Create a password

Use at least one lowercase letter, one numeral, and seven characters.

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We will send you account related emails occasionally.

https://github.com/

# Github repo

**GitHub** This repository Search Explore Features Enterprise Blog Sign up Sign in

SISBID / Module1 Watch 2 Star 2 Fork 5

Teaching material for Summer Institute in Statistics for Big Data Module 1.

27 commits 2 branches 0 releases 3 contributors

branch: gh-pages + Module1 / +

Add compiled html

File	Description	Time
lecture_notes	Added installation script and set up instructions	18 hours ago
.gitignore	First commit for lecture notes	27 days ago
.nojekyll	updated license, added webpage	5 days ago
LICENSE	updated license, added webpage	5 days ago
README.md	Fixed typo and links	18 hours ago
SISBD.Rproj	First commit for lecture notes	27 days ago
getting_started.md	Added installation script and set up instructions	18 hours ago
index.Rmd	Fixed typos and links	18 hours ago

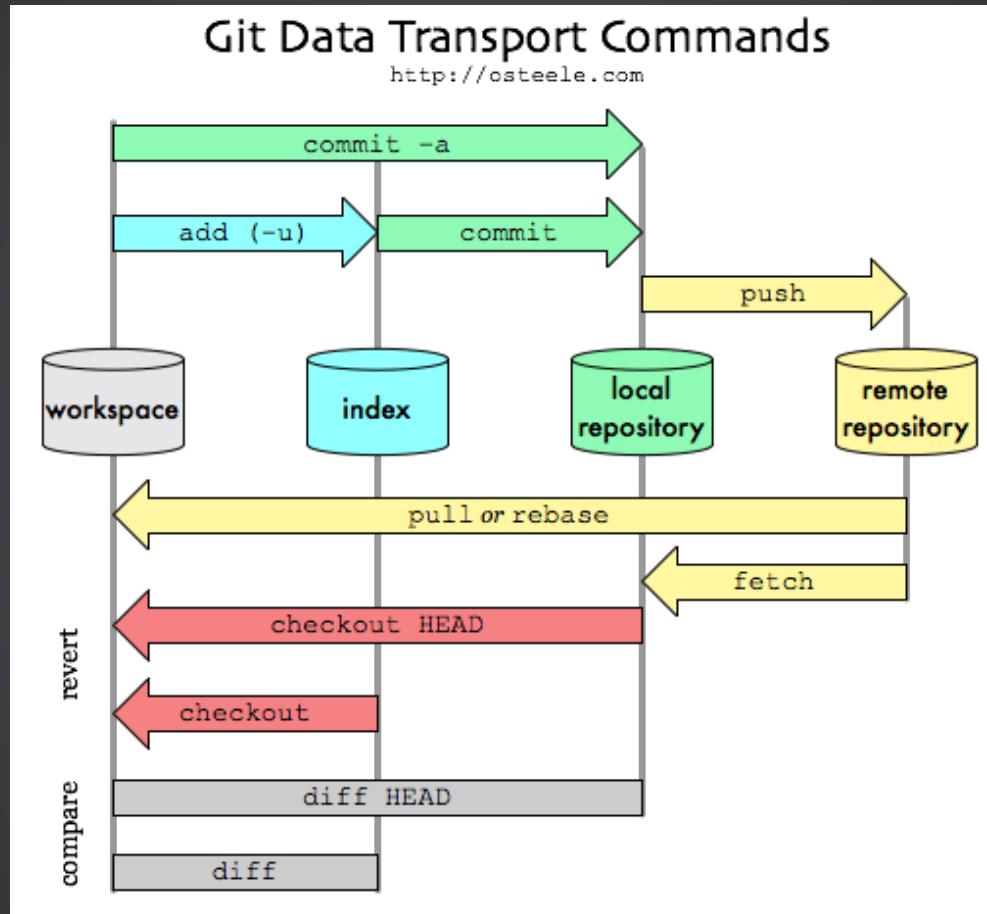
Code Issues 0 Pull requests 0 Pulse Graphs

HTTPS clone URL <https://github.com/> Clone in Desktop Download ZIP

You can clone with [HTTPS](#) or [Subversion](#).

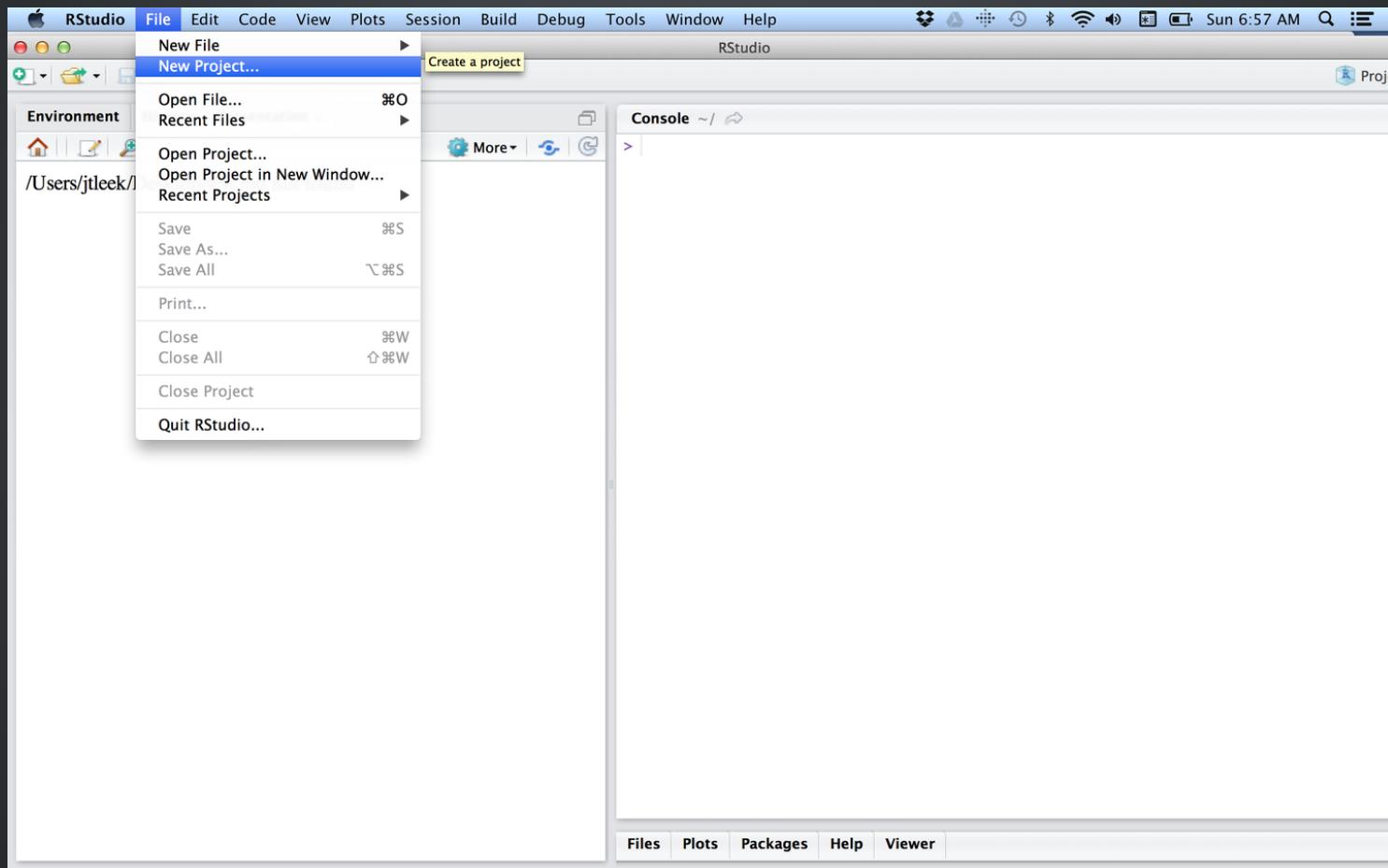
<https://github.com/jtleek/datasharing>

# Basic Scheme

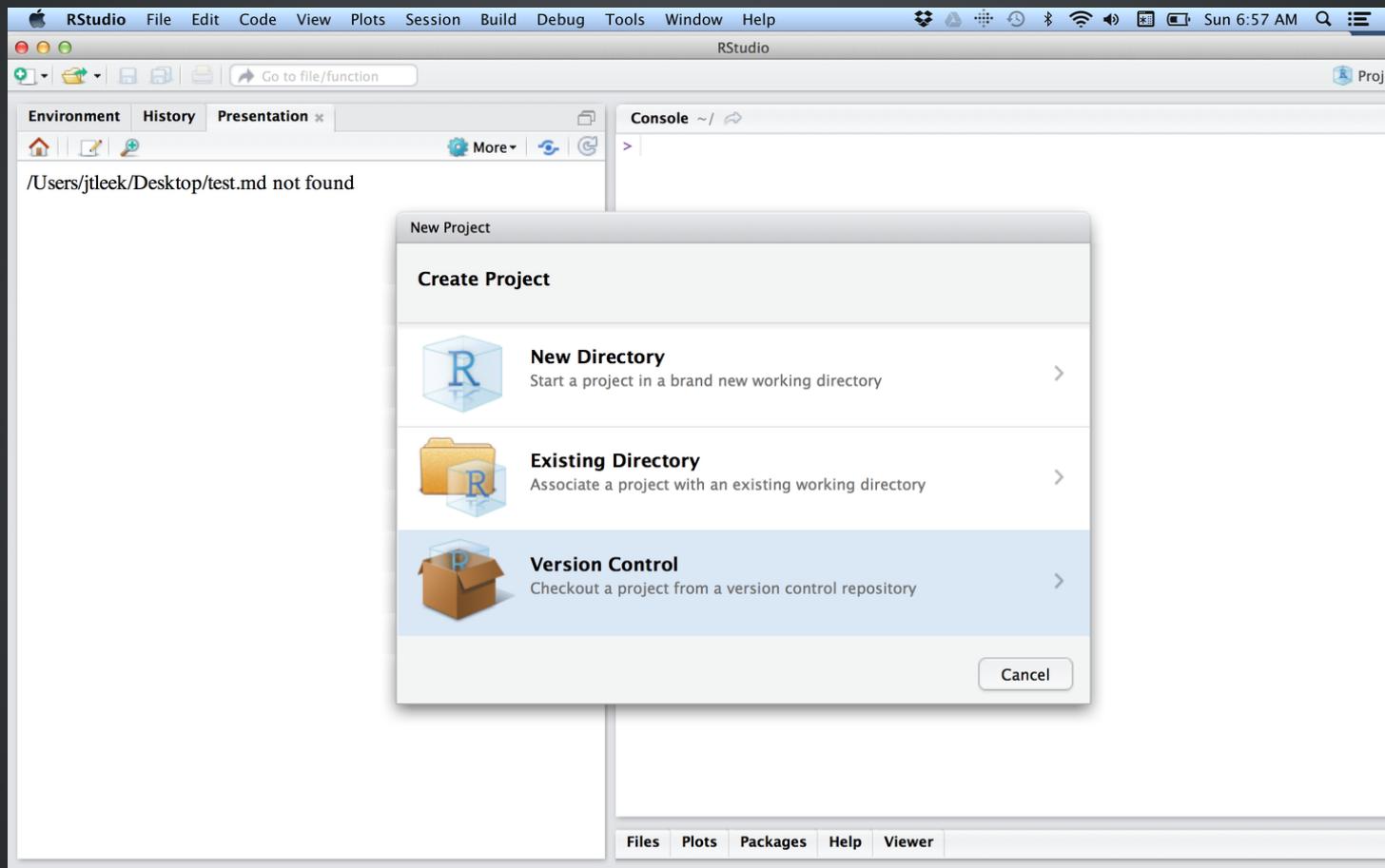


<http://gitready.com/beginner/2009/01/21/pushing-and-pulling.html>

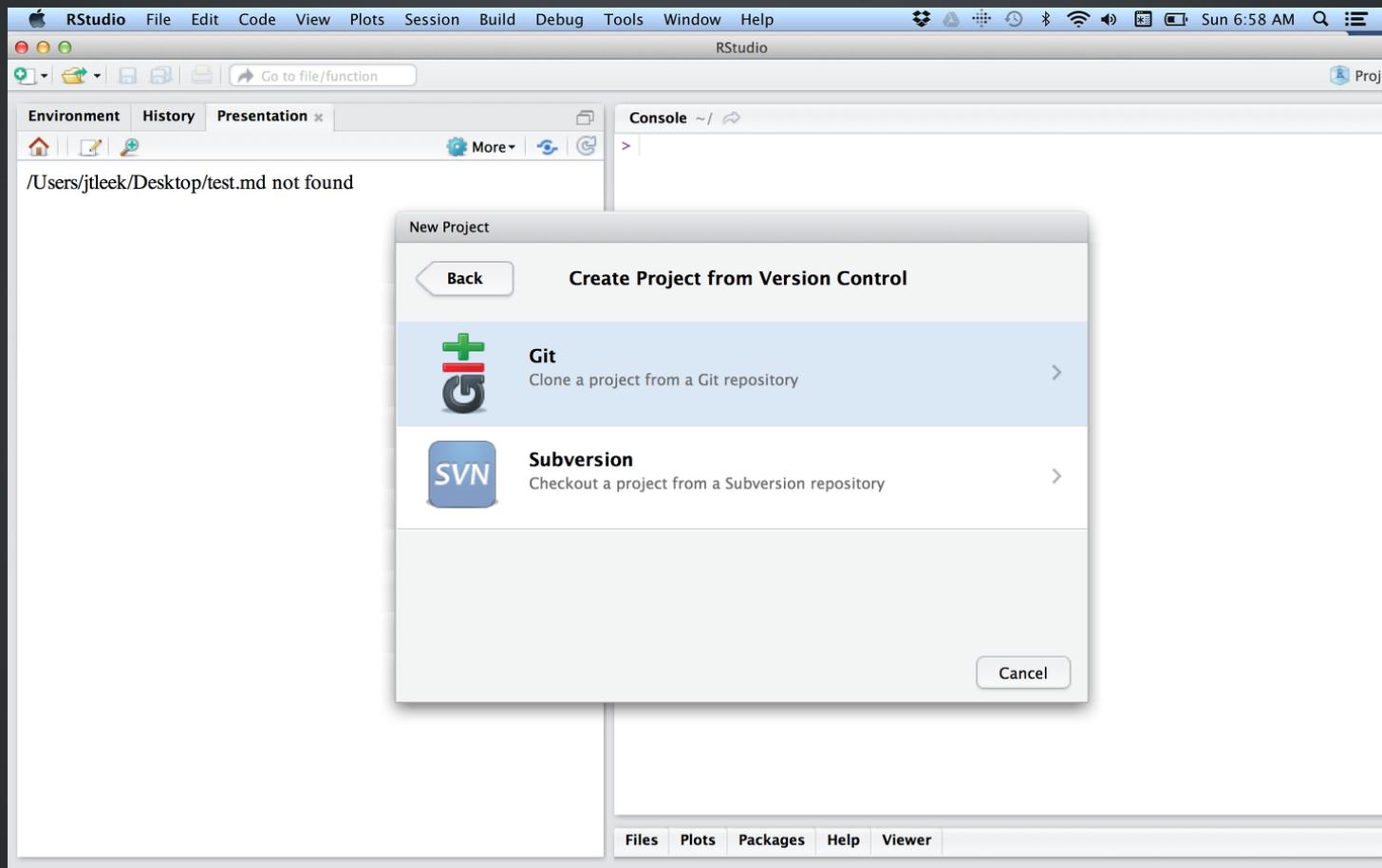
# Git/Github in Rstudio



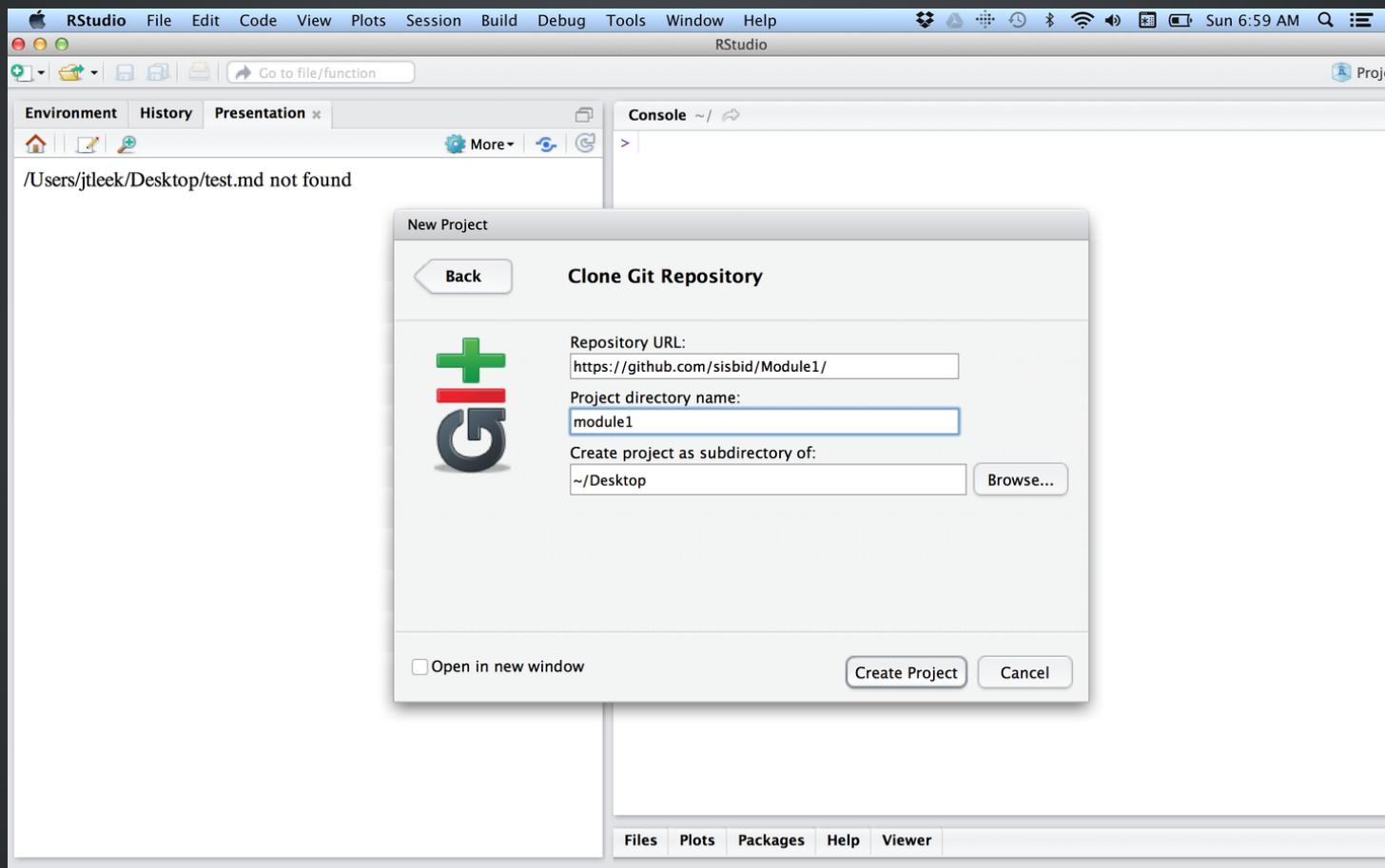
# Git/Github in Rstudio



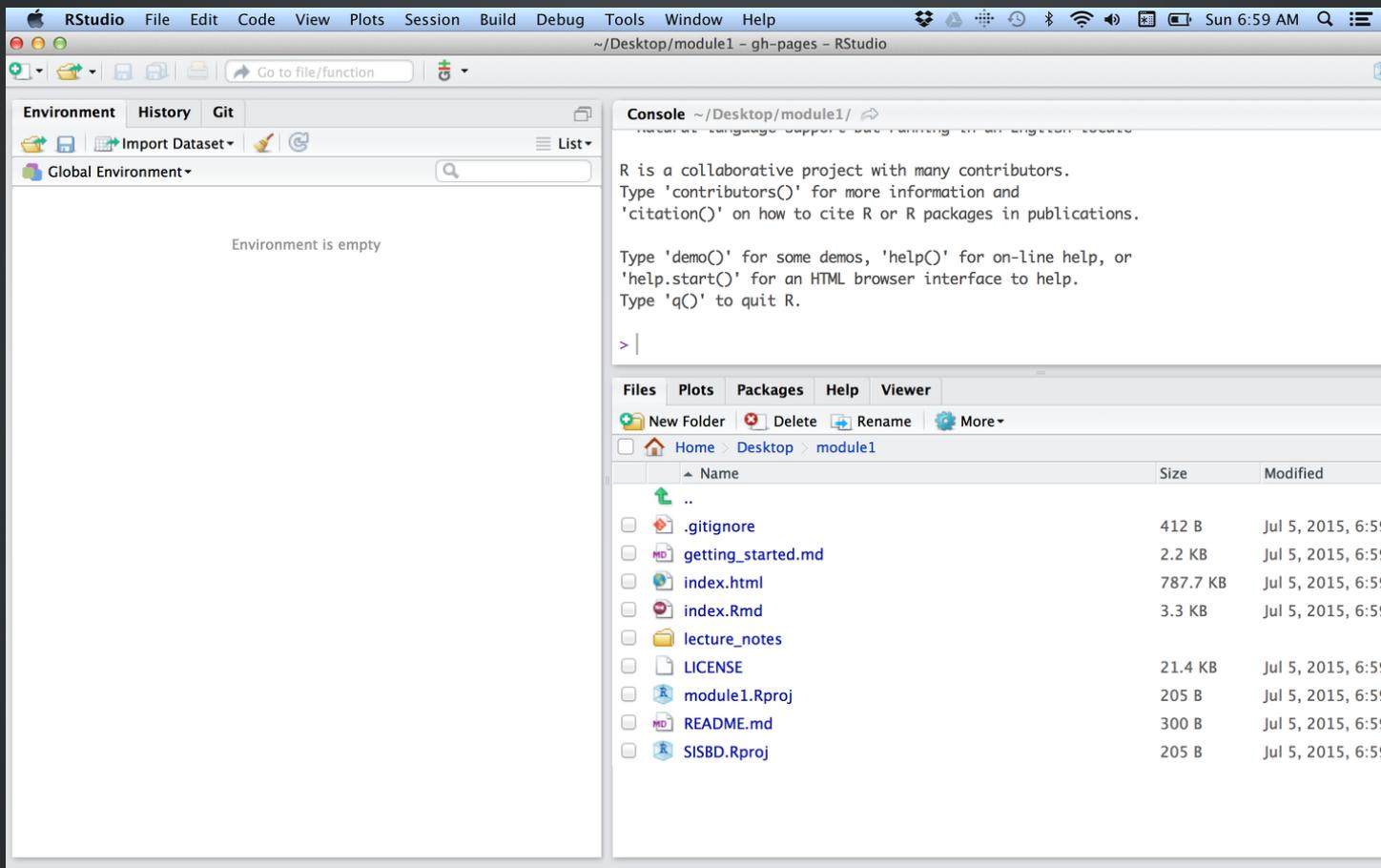
# Git/Github in Rstudio



# Git/Github in Rstudio



# Git/Github in Rstudio



# Installing Git/Github

<http://r-pkgs.had.co.nz/git.html#git-init>

# Git/Github Practice

<http://bit.ly/1H0qcwB>

Reproducible

Research

# Why reproducible research?

|| Your closest collaborator is you six months ago, but you don't reply to emails.

[http://kbroman.org/Tools4RR/assets/lectures/06\\_org\\_eda\\_withnotes.pdf](http://kbroman.org/Tools4RR/assets/lectures/06_org_eda_withnotes.pdf)

# Why reproducible research?

- | | Could you just re-run all that code with the [latest/different/best] parameters?
  - every collaborator

# Why reproducible research?

From the article:

## Cancer trial errors revealed

**2006** Anil Potti, a cancer geneticist at Duke University in Durham, North Carolina, and others file patent applications on the idea of using gene-expression data to predict sensitivity to cancer drugs. Potti is first author on a paper in *Nature Medicine*<sup>1</sup>.

**2007** Potti is last author on a paper in the *Journal of Clinical Oncology* (*JCO*)<sup>2</sup>. Duke begins three clinical trials to test Potti's predictors in patients with breast or lung cancer.

**SEPTEMBER 2009** Keith Baggerly and Kevin Coombes, statisticians at the University of Texas M. D. Anderson Cancer Centre in Houston, publish a paper in *Annals of Applied Statistics*<sup>3</sup> stating that they could not replicate Potti's claims. Duke suspends the trials and asks a review panel to investigate.

**NOVEMBER 2009** Potti places data underlying the *JCO* paper online. Baggerly writes to Sally Kornbluth, Duke vice-dean for research, and Michael Cuffe, Duke vice-president for medical affairs, to point out differences from raw data.

**DECEMBER 2009** An unredacted copy of the report by Duke's review panel, later obtained by *Nature*, shows that the panel replicated Potti's claims using his data, but were unaware that those data contained discrepancies.

**JANUARY 2010** Duke restarts clinical trials.

**JULY 2010** *The Cancer Letter* reveals that Potti made false claims about his CV. Trials are suspended and an investigation begins. Harold Varmus, director of the National Cancer Institute in Bethesda, Maryland, asks the Institute of Medicine to review Duke's trials.

**NOVEMBER 2010** *JCO* paper is retracted. Duke closes the trials permanently. Potti resigns.

**DECEMBER 2010** Institute of Medicine study begins, but will now focus more generally on criteria for genomics predictor.

**JANUARY 2011** *Nature Medicine* paper is retracted.

<http://www.nature.com/news/2011/110111/full/469139a/box/1.html>

# Why reproducible research?

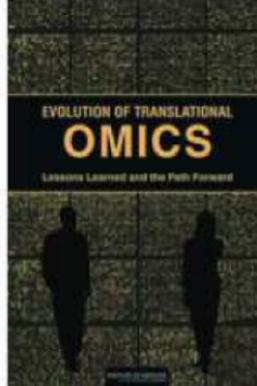
REPORT BRIEF MARCH 2012

INSTITUTE OF MEDICINE  
OF THE NATIONAL ACADEMIES  
Advising the nation • Improving health

For more information visit [www.iom.edu/translationalomics](http://www.iom.edu/translationalomics)

## Evolution of Translational Omics

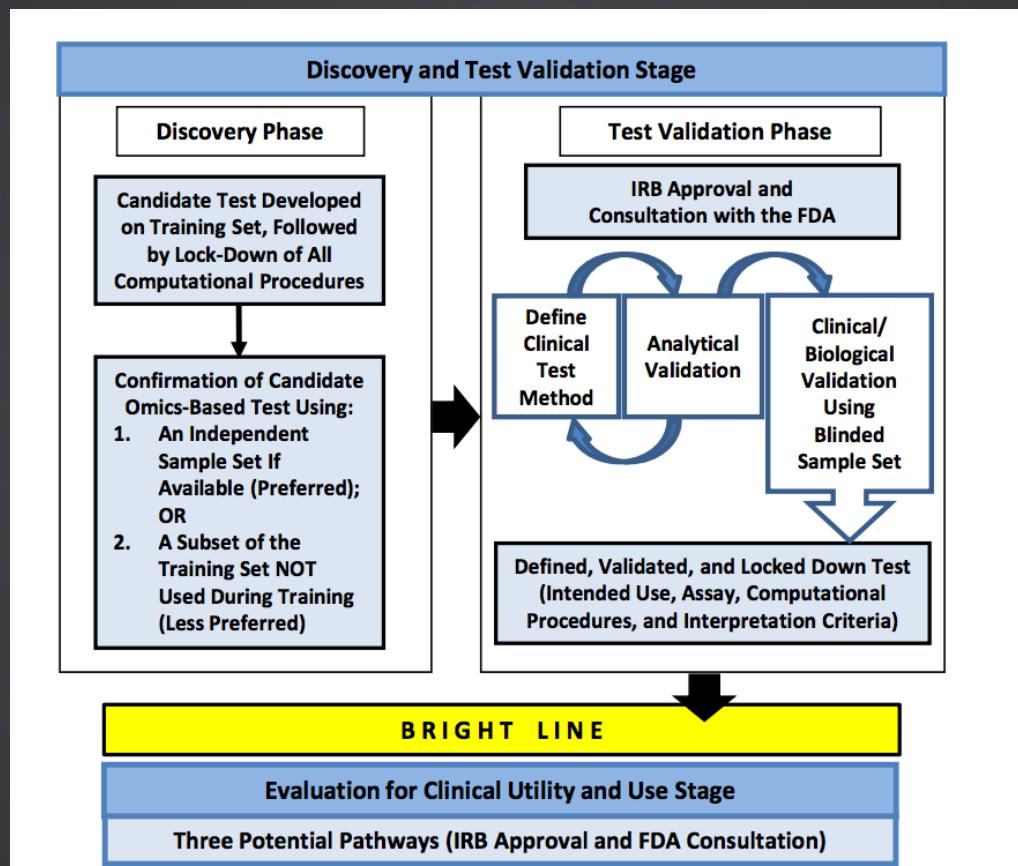
### Lessons Learned and the Path Forward



**Sequencing the human genome** opened a new era in biomedical science. Researchers have begun to untangle the complex roles of biology and genetics in specific diseases, and now better understand why particular therapies do or do not work in individual patients. New technologies have made it feasible to measure an enormous number of molecules within a tissue or cell; for example, genomics investigates thousands of DNA sequences, and proteomics examines large numbers of proteins. Collectively, these technologies

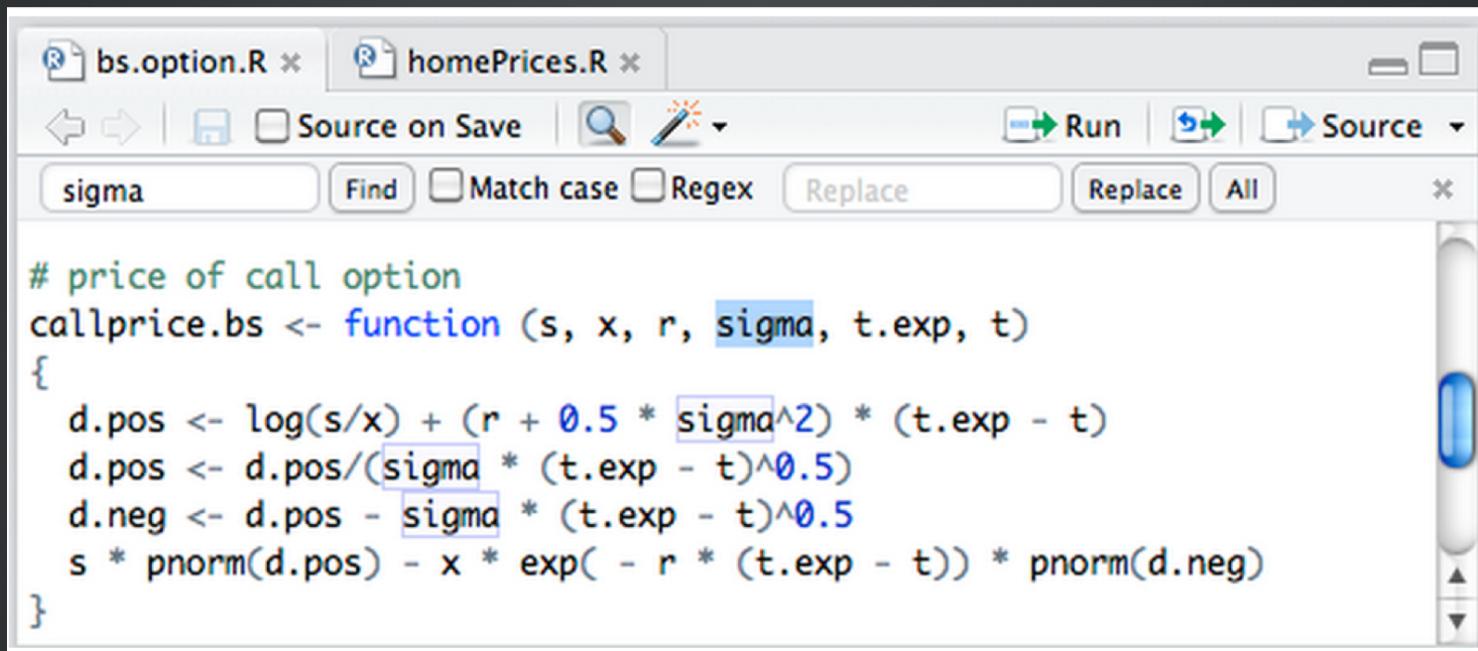
<http://www.iom.edu/Reports/2012/Evolution-of-Translational-Omics.aspx>

# The bright line



<http://www.iom.edu/Reports/2012/Evolution-of-Translational-Omics.aspx>

# R scripts



The screenshot shows the RStudio IDE interface with two files open in the top tab bar: "bs.option.R" and "homePrices.R". The main editor area contains the following R code:

```
# 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)
}
```

The variable "sigma" is highlighted in blue, indicating it is selected or being used in the current context.

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

**Markdown** → **HTML**

**foo.md** → **foo.html**

**easy to write  
(and read!)**

**easy to publish  
easy to read in  
browser**

<https://speakerdeck.com/jennybc/new-tools-and-workflows-for-data-analysis>

# Markdown

# HTML

```
Title (header 1, actually)
=====
```

This is a Markdown document.

```
## Medium header (header 2, actually)
```

It's easy to do *italics* or make things bold.

> All models are wrong, but some are useful. An approximate answer to the right problem is worth a good deal more than an exact answer to an approximate problem. Absolute certainty is a privilege of uneducated minds-and fanatics. It is, for scientific folk, an unattainable ideal. What you do every day matters more than what you do once in a while. We cannot expect anyone to know anything we didn't teach them ourselves. Enthusiasm is a form of social courage.

Code block below. Just affects formatting here but we'll get to R Markdown for the real fun soon!

```
```  
x <- 3 * 4  
```
```

I can haz equations. Inline equations, such as ... the average is computed as  $\frac{1}{n} \sum_{i=1}^n x_i$ . Or display equations like this:

```
$$  
\begin{equation*}  
|x| =  
\begin{cases} x & \text{if } x \geq 0, \\ -x & \text{if } x < 0. \end{cases}  
\end{equation*}
```



## Title (header 1, actually)



This is a Markdown document.

## Medium header (header 2, actually)

It's easy to do *italics* or **make things bold**.

All models are wrong, but some are useful. An approximate answer to the right problem is worth a good deal more than an exact answer to an approximate problem. Absolute certainty is a privilege of uneducated minds-and fanatics. It is, for scientific folk, an unattainable ideal. What you do every day matters more than what you do once in a while. We cannot expect anyone to know anything we didn't teach them ourselves. Enthusiasm is a form of social courage.

Code block below. Just affects formatting here but we'll get to R Markdown for the real fun soon!

```
x <- 3 * 4
```

I can haz equations. Inline equations, such as ... the average is computed as  $\frac{1}{n} \sum_{i=1}^n x_i$ . Or display equations like this:

$$|x| = \begin{cases} x & \text{if } x \geq 0, \\ -x & \text{if } x < 0. \end{cases}$$

# R markdown files

The screenshot shows the RStudio IDE interface with the following details:

- Title Bar:** Shows the file name "index.Rmd x".
- Toolbar:** Includes icons for Save, Undo, Redo, ABC, MD, and Knit HTML.
- Search Bar:** Contains fields for Find, Next, Prev, Replace, and All, along with checkboxes for In selection, Match case, Whole word, Regex, and Wrap.
- Code Editor:** Displays the R code for the "index.Rmd" file. The code includes sections for clustering, statistical modeling, and logistic regression, with various comments and annotations.
- Bottom Status Bar:** Shows the line number "186:1" and the status "Top Level".
- Right Panel:** Shows the "Run" and "Chunks" tabs.

```
index.Rmd x
Find Next Prev Replace All
In selection Match case Whole word Regex Wrap
252 ---
253 ## New clustering
254 ````{r, fig.height =6,fig.width=6}
255 hClusterUpdated = hclust(dist(t(log10(trainSpam[,1:55]+1))))
256 plot(hClusterUpdated)
257 ```
258
259 ---
260 ## Statistical prediction/modeling
261
262 * Should be informed by the results of your exploratory analysis
263 * Exact methods depend on the question of interest
264 * Transformations/processing should be accounted for when necessary
265 * Measures of uncertainty should be reported
266
267 ---
268 ## Statistical prediction/modeling
269 ````{r,cache=TRUE}
270 trainSpam$numType = as.numeric(trainSpam$type)-1
271 costFunction = function(x,y){sum(x!=y > 0.5)}
272 cvError = rep(NA,55)
273 library(boot)
274 for(i in 1:55){
275   lmFormula = as.formula(paste("numType~",names(trainSpam)[i],sep=""))
276   glmFit = glm(lmFormula,family="binomial",data=trainSpam)
277   cvError[i] = cv.glm(trainSpam,glmFit,costFunction,2)$delta[2]
278 }
279 which.min(cvError)
280 names(trainSpam)[which.min(cvError)]
281 ```
282
283
284 ---
```

[http://www.rstudio.com/ide/docs/authoring/using\\_markdown](http://www.rstudio.com/ide/docs/authoring/using_markdown)

# R Markdown

# Markdown

R Markdown rocks

This is an R Markdown document.

```
```{r}
x <- rnorm(1000)
head(x)
````
```

See how the R code gets executed and a representation thereof appears in the document? `knitr` gives you control over how to represent all conceivable types of output. In case you care, then average of the `r length(x)` random normal variates we just generated is `r round(mean(x), 3)`. Those numbers are NOT hard-wired but are computed on-the-fly. As is this figure. No more copy-paste ... copy-paste ... oops forgot to copy-paste.

```
```{r}
plot(density(x))
````
```

Note that all the previously demonstrated math typesetting still works. You don't have to choose between having math cred and being web-friendly!

Inline equations, such as ... the average is computed as  $\frac{1}{n} \sum_{i=1}^n x_i$ . Or display equations like this:

```
$$
\begin{equation*}
|x| =
\begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x \leq 0 \end{cases}
\end{equation*}
````
```

R Markdown rocks

This is an R Markdown document.

```
```{r}
x <- rnorm(1000)
head(x)
````
```

```
...
## [1] -1.3007  0.7715  0.5585 -1.2854  1.1973
2.4157
````
```

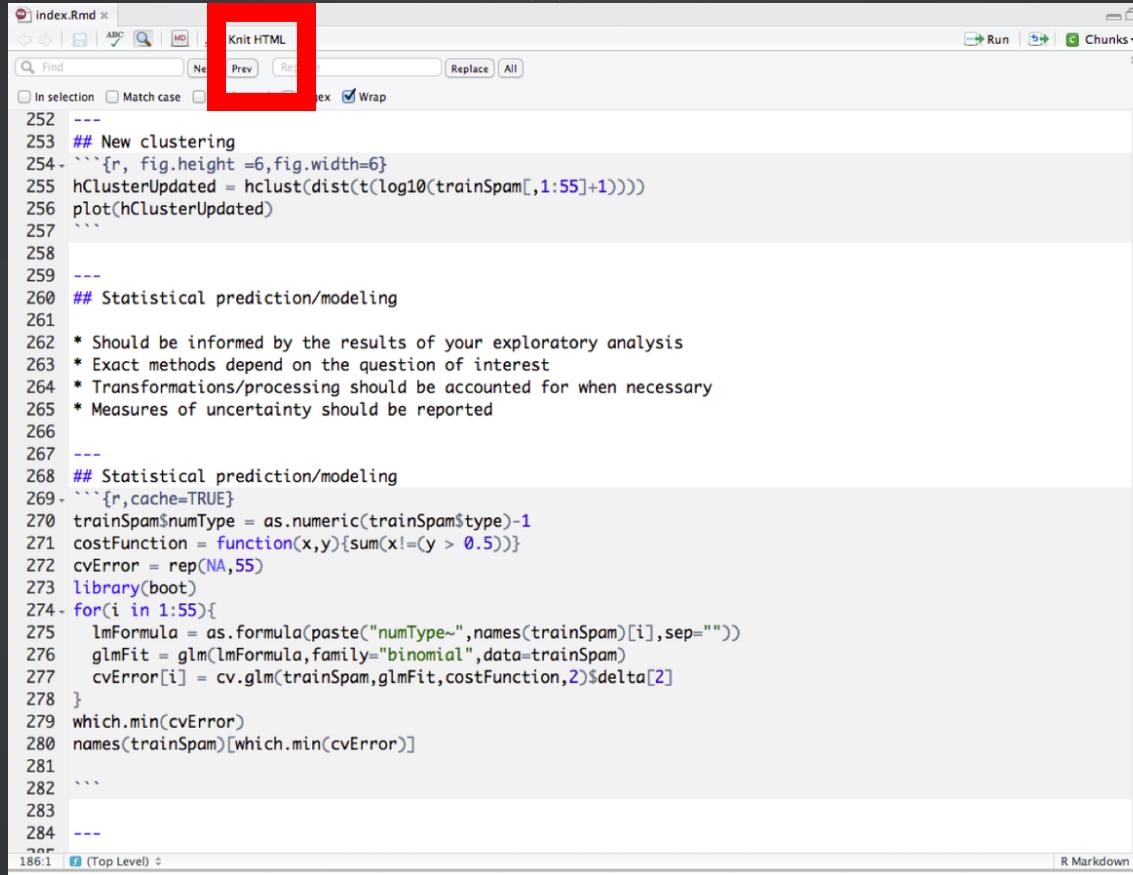
See how the R code gets executed and a representation thereof appears in the document? `knitr` gives you control over how to represent all conceivable types of output. In case you care, then average of the 1000 random normal variates we just generated is -0.081. Those numbers are NOT hard-wired but are computed on-the-fly. As is this figure. No more copy-paste ... copy-paste ... oops forgot to copy-paste.

```
```{r}
plot(density(x))
````
```

```
![plot of chunk unnamed-chunk-2](figure/unnamed-
chunk-2.png)
```

```
library(rmarkdown  
render("foo.Rmd")
```

# R markdown -> HTML



The screenshot shows the RStudio IDE interface with an R Markdown file named "index.Rmd" open. The code editor displays R code for data clustering and statistical modeling. A red box highlights the "Knit HTML" button in the toolbar, which is used to convert the R Markdown document into an HTML file. The code includes comments explaining the steps of the analysis, such as clustering, prediction modeling, and cross-validation.

```
252 ---  
253 ## New clustering  
254 ````{r, fig.height = 6,fig.width=6}  
255 hClusterUpdated = hclust(dist(t(log10(trainSpam[,1:55]+1))))  
256 plot(hClusterUpdated)  
257 ````  
258  
259 ---  
260 ## Statistical prediction/modeling  
261  
262 * Should be informed by the results of your exploratory analysis  
263 * Exact methods depend on the question of interest  
264 * Transformations/processing should be accounted for when necessary  
265 * Measures of uncertainty should be reported  
266  
267 ---  
268 ## Statistical prediction/modeling  
269 ````{r,cache=TRUE}  
270 trainSpam$numType = as.numeric(trainSpam$type)-1  
271 costFunction = function(x,y){sum(x!=y > 0.5)}  
272 cvError = rep(NA,55)  
273 library(boot)  
274 for(i in 1:55){  
275   lmFormula = as.formula(paste("numType~",names(trainSpam)[i],sep=""))  
276   glmFit = glm(lmFormula,family="binomial",data=trainSpam)  
277   cvError[i] = cv.glm(trainSpam,glmFit,costFunction,2)$delta[2]  
278 }  
279 which.min(cvError)  
280 names(trainSpam)[which.min(cvError)]  
281  
282 ````  
283  
284 ---  
285
```

[http://www.rstudio.com/ide/docs/authoring/using\\_markdown](http://www.rstudio.com/ide/docs/authoring/using_markdown)

# R markdown

## Lab

<http://bit.ly/1LRk3ds>

# Downloading

## Data

# Organizing yourself

Project/  
raw/  
processed/  
code/  
README .md

## Relative path (do this)

```
setwd( "../data" )  
setwd( "./files" )
```

```
setwd( "..\tmp" )
```

## Absolute path (not this)

```
setwd( "/Users/jtleek/data" )  
setwd( "~/Desktop/files/data" )
```

```
setwd( "C:\\\\Users\\\\Andrew\\\\Downloads" )
```

# Finding and creating files

```
if( !file.exists( "data" ) ) {  
    dir.create( "data" )  
}
```

# Listing files

```
list.files("data")
[1] "chicago.rds" "geo"
```

# Downloading files

```
fileUrl <- "https://data.baltimorecity.gov/api/views/dz54-2aru/rows.csv?accessType=DOWNLOA
```

```
download.file(fileUrl,  
  destfile=".data/cameras.csv",  
  method="curl")  
list.files("./data")  
dateDownloaded <- date()  
dateDownloaded
```

# Maybe easier (on Macs)

```
fileUrl <- "https://data.baltimorecity.gov/api/views/dz54-2aru/rows.csv?accessType=DOWNLOA
```

```
install.packages("downloader")
library(downloader)
download(fileUrl,
         destfile=".data/cameras.csv")
list.files("./data")
dateDownloaded <- date()
dateDownloaded
```

Reading

CSV

# Reading local files

```
cameras = read.table("./data/cameras.csv",
                      header=T,
                      sep=",")  
  
cameras = read.csv("./data/cameras.csv")
```

# For big(ger) local files

```
install.packages("readr")
library(readr)
cameras = read_csv("./data/cameras.csv")
```

# Quick comparison

```
dat = matrix(rnorm(100*1000),  
             nrow=1000)  
  
write.csv(dat,file="dat.csv")  
  
system.time(read.csv("dat.csv"))  
    user  system elapsed  
0.269    0.003   0.272  
  
system.time(read_csv("dat.csv"))  
    user  system elapsed  
0.025    0.001   0.026
```

Reading

Excel

# Most common file format

The screenshot shows a Microsoft Excel web page. At the top, there's a navigation bar with links for HOME, MY OFFICE, PRODUCTS, SUPPORT, IMAGES, TEMPLATES, and STORE. Below the navigation is a search bar with the placeholder 'Search all of Office.com'. To the right of the search bar are two buttons: 'Buy with Office' and 'Try 1 month FREE'. The main content area features a large green banner on the left with the Microsoft Excel logo and the word 'Analyze.' Below it is a link 'What's new in Excel? >'. To the right of the banner is a photograph of a computer monitor displaying an Excel spreadsheet titled 'Employee Travel Expense Trends - Social'. The spreadsheet contains a bar chart with data for months from Jan to Dec, showing expenses for various categories like Airfare, Conventions, Hotels, and Meals. Below the monitor, there are three buttons: 'Discover' (in red), 'Visualize' (in white), and 'Share' (in white). A footer at the bottom of the page reads 'Discover and reveal the insights hidden in your data'.

<http://office.microsoft.com/en-us/excel/>

# Get an Excel File

```
fileUrl <- "https://data.baltimorecity.gov/api/views/dz54-2aru/rows.xlsx?accessType=DOWNLOA
```

```
download(fileUrl,  
        destfile=".data/cameras.xlsx")  
list.files("./data")  
dateDownloaded <- date()  
dateDownloaded
```

# Read it

```
install.packages("readxl")  
cameras = read_excel("./data/cameras.xlsx"  
                     sheet=1)
```

# Sheets and Cells

# If you want to read cells

```
install.packages("xlsx")
library(xlsx)
colIndex = 2:3
rowIndex = 1:4

cameras = read.xlsx("./data/cameras.xlsx",
                     sheetIndex=1,
                     header=TRUE,
                     colIndex= colIndex,
                     rowIndex = rowIndex)
```

# Result

```
> cameras
    direction      street
  1          N/B    Caton Ave
  2          S/B    Caton Ave
  3          E/B  Wilkens Ave
```

Reading  
Google Sheets

# Google Sheets

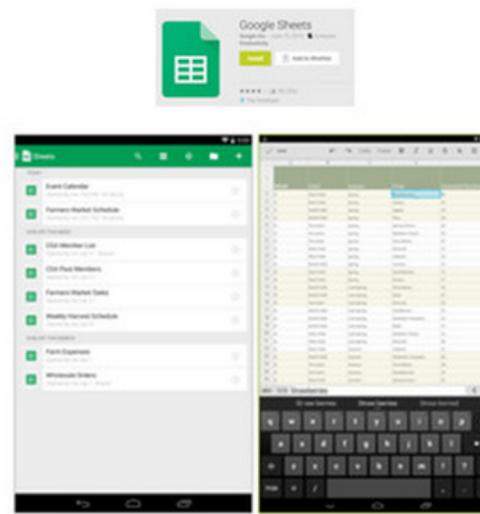
A screenshot of the Google Sheets interface. The title bar reads "Untitled spreadsheet". The menu bar includes File, Edit, View, Insert, Format, Data, Tools, Add-ons, and Help. The top toolbar contains various icons for file operations, text styling (bold, italic, underline), and data manipulation. The spreadsheet area shows a grid from row 1 to 21 and column A to L. Row 1 is highlighted in light gray, and cell A1 is selected, indicated by a blue border. The rest of the cells are white.

<http://google.com/sheets>

## iOS



## Android



enter data from a phone  
enter data w/o WiFi

Slide from: <https://speakerdeck.com/jennybc/googlesheets-talk-at-user2015>

# googlesheets package

```
library(devtools)
install_github("jennybc/cellranger")
install_github("jennybc/googlesheets")
library(googlesheets)
?gs_read
?"cell-specification"
```

cameras

jtteek@gmail.com

File Edit View Insert Format Data Tools Add-ons Help Last edit was seconds ago

Comments Share

Sheets home address

|    | A                   | B         | C             | D               | E                          | F                               | G | H | I | J | K | L | M |
|----|---------------------|-----------|---------------|-----------------|----------------------------|---------------------------------|---|---|---|---|---|---|---|
| 1  | address             | direction | street        | crossStreet     | intersection               | Location 1                      |   |   |   |   |   |   |   |
| 2  | S CATON AVE & B     | N/B       | Caton Ave     | Benson Ave      | Caton Ave & Benson Ave     | (39.2693779962, -76.6688185297) |   |   |   |   |   |   |   |
| 3  | S CATON AVE & B     | S/B       | Caton Ave     | Benson Ave      | Caton Ave & Benson Ave     | (39.2693157898, -76.6688698176) |   |   |   |   |   |   |   |
| 4  | WILKENS AVE & P     | E/B       | Wilkens Ave   | Pine Heights    | Wilkens Ave & Pine Height  | (39.2720252302, -76.676960806)  |   |   |   |   |   |   |   |
| 5  | THE ALAMEDA & E     | S/B       | The Alameda   | 33rd St         | The Alameda & 33rd St      | (39.3285013141, -76.5953545714) |   |   |   |   |   |   |   |
| 6  | E 33RD ST & THE     | E/B       | E 33rd        | The Alameda     | E 33rd & The Alameda       | (39.3283410623, -76.5953594625) |   |   |   |   |   |   |   |
| 7  | ERDMAN AVE & N      | E/B       | Erdman        | Macon St        | Erdman & Macon St          | (39.3068045671, -76.5593167803) |   |   |   |   |   |   |   |
| 8  | ERDMAN AVE & N      | W/B       | Erdman        | Macon St        | Erdman & Macon St          | (39.306966535, -76.5593122365)  |   |   |   |   |   |   |   |
| 9  | N CHARLES ST & E    | S/B       | Charles       | Lake Ave        | Charles & Lake Ave         | (39.3690535299, -76.625826716)  |   |   |   |   |   |   |   |
| 10 | E MADISON ST & N    | W/B       | Madison       | Caroline St     | Madison & Caroline St      | (39.2993257666, -76.5976760827) |   |   |   |   |   |   |   |
| 11 | ORLEANS ST & N      | E/B       | Orleans       | Linwood Ave     | Orleans & Linwood Ave      | (39.2958661981, -76.5764270078) |   |   |   |   |   |   |   |
| 12 | EASTERN AVE & K     | E/B       | Eastern       | Kane St         | Eastern & Kane St          | (39.2877626582, -76.5371017795) |   |   |   |   |   |   |   |
| 13 | EDMONDSON AVE       | E/B       | Edmonson      | Cooks Lane      | Edmonson & Cooks Lane      | (39.2923680595, -76.7017056326) |   |   |   |   |   |   |   |
| 14 | W FRANKLIN ST &     | W/B       | Franklin      | Pulaski St      | Franklin & Pulaski St      | (39.2937082594, -76.6503837515) |   |   |   |   |   |   |   |
| 15 | ORLEANS ST & N      | E/B       | Orleans       | Gay St          | Orleans & Gay St           | (39.2947203114, -76.606128007)  |   |   |   |   |   |   |   |
| 16 | S MARTIN LUTHER     | N/B       | MLK Jr. Blvd. | Washington Blvd | MLK Jr. Blvd. & Washington | (39.2834598231, -76.6261138807) |   |   |   |   |   |   |   |
| 17 | HILLEN RD & ARGONNE | S/B       | Hillen Rd     | Argonne Drive   | Hillen Rd & Argonne Driv   | (39.3399907644, -76.588021025)  |   |   |   |   |   |   |   |
| 18 | W NORTH AVE & N     | W/B       | North Ave     | Howard St       | North Ave & Howard St      | (39.3110873669, -76.6193071428) |   |   |   |   |   |   |   |
| 19 | E PATAPSCO AVE      | W/B       | Patapsco      | 4th St          | Patapsco & 4th St          | (39.2372692804, -76.6054039252) |   |   |   |   |   |   |   |
| 20 | REISTERSTOWN F      | S/B       | Reisterstown  | Fallstaff Road  | Reisterstown & Fallstaff   | (39.3621351031, -76.7102427408) |   |   |   |   |   |   |   |
| 21 | PARK HEIGHTS AV     | N/B       | Park Heights  | Hayward Ave     | Park Heights & Hayward     | (39.3499204055, -76.6788706721) |   |   |   |   |   |   |   |
| 22 | PARK HEIGHTS AVE    | S/B       | Park Heights  | Hayward Ave     | Park Heights & Hayward     | (39.3499204055, -76.6788706721) |   |   |   |   |   |   |   |

\https://docs.google.com/spreadsheets/d/1xYzntyuetrn7mOIE7iwUtz1nS3f2uREK6y0vv\_AXKNO/pubhtml

cameras ★

jleek@gmail.com ▾

File Edit View Insert Format Data Tools Add-ons Help Last edit was 3 days ago

Comments Share

Share...

New

Open...

Rename...

Make a copy...

Move to folder...

Move to trash

Import...

See revision history ⌘+Option+Shift+G

Spreadsheet settings...

Publish to the web...

Email as attachment...

Print ⌘P

DATA AVE W/B

REISTERSTOWN F S/B

PARK HEIGHTS AV N/B

PARK HEIGHTS AV S/B

S MARTIN LUTHER KING S/B

W NORTHERN PKWY E/B

|    | C                        | D              | E                               | F                               | G                               | H | I | J | K | L | M |
|----|--------------------------|----------------|---------------------------------|---------------------------------|---------------------------------|---|---|---|---|---|---|
|    | street                   | crossStreet    | intersection                    | Location 1                      |                                 |   |   |   |   |   |   |
| 1  | ave                      | Benson Ave     | Caton Ave & Benson Ave          | (39.2693779962, -76.6688185297) |                                 |   |   |   |   |   |   |
| 2  | ave                      | Benson Ave     | Caton Ave & Benson Ave          | (39.2693157898, -76.6689698176) |                                 |   |   |   |   |   |   |
| 3  | Ave                      | Pine Heights   | Wilkens Ave & Pine Height       | (39.2720252302, -76.676960806)  |                                 |   |   |   |   |   |   |
| 4  | meda                     | 33rd St        | The Alameda & 33rd St           | (39.3285013141, -76.5953545714) |                                 |   |   |   |   |   |   |
| 5  |                          | The Alameda    | E 33rd & The Alameda            | (39.3283410623, -76.5953594625) |                                 |   |   |   |   |   |   |
| 6  |                          | Macon St       | Erdman & Macon St               | (39.3068045671, -76.5593167803) |                                 |   |   |   |   |   |   |
| 7  |                          | Macon St       | Erdman & Macon St               | (39.306966535, -76.5593122365)  |                                 |   |   |   |   |   |   |
| 8  |                          | Lake Ave       | Charles & Lake Ave              | (39.3690535299, -76.625826716)  |                                 |   |   |   |   |   |   |
| 9  |                          | Caroline St    | Madison & Caroline St           | (39.2993257666, -76.5976760827) |                                 |   |   |   |   |   |   |
| 10 |                          | Linwood Ave    | Orleans & Linwood Ave           | (39.2958661981, -76.5764270078) |                                 |   |   |   |   |   |   |
| 11 |                          | Kane St        | Eastern & Kane St               | (39.2877626582, -76.5371017795) |                                 |   |   |   |   |   |   |
| 12 |                          | Cooks Lane     | Edmonson & Cooks Lane           | (39.2923680595, -76.7017056326) |                                 |   |   |   |   |   |   |
| 13 |                          | Pulaski St     | Franklin & Pulaski St           | (39.2937082594, -76.6503837515) |                                 |   |   |   |   |   |   |
| 14 |                          | Gay St         | Orleans & Gay St                | (39.2947203114, -76.606128007)  |                                 |   |   |   |   |   |   |
| 15 |                          | Blvd.          | MLK Jr. Blvd. & Washington Blvd | (39.2834598231, -76.6261138807) |                                 |   |   |   |   |   |   |
| 16 |                          | Argonne Drive  | Hillen Rd & Argonne Driv        | (39.3399907644, -76.588021025)  |                                 |   |   |   |   |   |   |
| 17 |                          | Howard St      | North Ave & Howard St           | (39.3110873669, -76.6193071428) |                                 |   |   |   |   |   |   |
| 18 |                          | Patapsco       | 4th St                          | (39.2372692804, -76.6054039252) |                                 |   |   |   |   |   |   |
| 19 |                          |                | & 4th St                        |                                 |                                 |   |   |   |   |   |   |
| 20 | REISTERSTOWN F S/B       | Fallstaff Road | Reisterstown & Fallstaff Rd     | (39.3621351031, -76.7102427408) |                                 |   |   |   |   |   |   |
| 21 | PARK HEIGHTS AV N/B      | Park Heights   | Hayward Ave                     | Park Heights & Hayward          | (39.3499204055, -76.6788706721) |   |   |   |   |   |   |
| 22 | PARK HEIGHTS AV S/B      | Park Heights   | Hayward Ave                     | Park Heights & Hayward          | (39.3499204055, -76.6788706721) |   |   |   |   |   |   |
| 23 | S MARTIN LUTHER KING S/B | MLK Jr. Blvd   | Pratt St                        | MLK Jr. Blvd & Pratt St         | (39.2860268994, -76.6278460704) |   |   |   |   |   |   |
| 24 | W NORTHERN PKWY E/B      | Northern Pkwy  | Greenspring Ave                 | Northern Pkwy & Greens          | (39.3550243172, -76.6604587972) |   |   |   |   |   |   |

[https://docs.google.com/spreadsheets/d/1xYzntyuetrm7mOIE7iwUtz1nS3f2uREK6y0vv\\_AXKNO/pubhtml](https://docs.google.com/spreadsheets/d/1xYzntyuetrm7mOIE7iwUtz1nS3f2uREK6y0vv_AXKNO/pubhtml)

# Reading a Google Sheet

```
sheets_url = "https://docs.google.com/spreadsheets/d/1xYzntyuetrm7mO1E7iwUtz1nS3f2uREK6y0vv_AXKN0/p"
```

```
gsurl11 = gs_url(sheets_url)
dat = gs_read(gsurl11)
```

```
dat
```

Source: local data frame [80 x 6]

|   |   | address                | direction |     |  |
|---|---|------------------------|-----------|-----|--|
| 1 | S | CATON AVE & BENSON AVE |           | N/B |  |
| 2 | S | CATON AVE & BENSON AVE |           | S/B |  |

Google  
Sheets Lab

<http://bit.ly/1Cgzjxb>

# Reading

# Data: JSON

# JSON

```
{  
    "firstName": "John",  
    "lastName": "Smith",  
    "isAlive": true,  
    "age": 25,  
    "address": {  
        "streetAddress": "21 2nd Street",  
        "city": "New York",  
        "state": "NY",  
        "postalCode": "10021-3100"  
    },  
    "phoneNumbers": [  
        {  
            "type": "home",  
            "number": "212 555-1234"  
        },  
        {  
            "type": "office",  
            "number": "646 555-4567"  
        }  
    ],  
    "children": [],  
    "spouse": null  
}
```

<https://en.wikipedia.org/wiki/JSON>

# Why JSON matters

The screenshot shows a web browser window with the URL <https://developer.github.com/v3/search/>. The page content is a JSON object representing search results for the term "Tetris".

```
{  
  "text_matches": [  
    {  
      "object_url": "https://api.github.com/repositories/3081286",  
      "object_type": "Repository",  
      "property": "name",  
      "fragment": "Tetris",  
      "matches": [  
        {  
          "text": "Tetris",  
          "indices": [  
            0,  
            6  
          ]  
        }  
      ],  
      {  
        "object_url": "https://api.github.com/repositories/3081286",  
        "object_type": "Repository",  
        "property": "description",  
        "fragment": "A C implementation of Tetris using Pennsim through LC4",  
        "matches": [  
          {  
            "text": "Tetris",  
            "indices": [  
              22,  
              28  
            ]  
          }  
        ]  
      }  
    }  
  ]  
}
```

A large blue rectangular box highlights the URL at the bottom of the page: <https://developer.github.com/v3/search/>.

<https://developer.github.com/v3/search/#text-match-metadata>

# Reading JSON

```
github_url = "https://api.github.com/users/jtleek/repos"
```

```
install.packages("jsonlite")
library(jsonlite)
jsonData <- fromJSON(github_url)
dim(jsonData)
[1] 30 67
```

```
jsonData$name
[1] "ballgown"           "capitalIn21stCen
[4] "dataanalysis"        "datascientist"
[7] "datawomenontwitter" "derfinder"
[10] "DSM"                 "EDA-Project"
```

# Data frame structure from JSON

```
table(sapply(jsonData, class))
      character   data.frame     integer     logical
                50             1             9             7

dim(jsonData$owner)
[1] 30 17

names(jsonData$owner)
[1] "login"          "id"
[5] "url"            "html_url"
[9] "gists_url"      "starred_url"
```

# Reading

## Data: HTML

# This is data

ReCount: analysis-ready

bowtie-bio.sourceforge.net/recount/

**Notes**  
Brief description of experiment.

Please note that to use the ExpressionSets below, you will need to install [Bioconductor](#) and run the command `library(Biobase)`

» **The Datasets**

| Study      | PMID                     | Species | Number of biological replicates       | Number of uniquely aligned reads | ExpressionSet                   | Count table                     | Phenotype table                 | Notes                                           |
|------------|--------------------------|---------|---------------------------------------|----------------------------------|---------------------------------|---------------------------------|---------------------------------|-------------------------------------------------|
| bodymap    | <a href="#">22496456</a> | human   | 19                                    | 2,197,622,796                    | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | Illumina Human BodyMap 2.0 -- tissue comparison |
| cheung     | <a href="#">20856902</a> | human   | 41                                    | 834,584,950                      | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | HapMap - CEU                                    |
| core       | <a href="#">19056941</a> | human   | 2                                     | 8,670,342                        | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | lung fibroblasts                                |
| gilad      | <a href="#">20009012</a> | human   | 6                                     | 41,356,738                       | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | liver; males and females                        |
| maqc       | <a href="#">20167110</a> | human   | 14<br>(technical)**<br>2 (biological) | 71,970,164                       | <a href="#">original pooled</a> | <a href="#">original pooled</a> | <a href="#">original pooled</a> | experiment: MAQC-2                              |
| montgomery | <a href="#">20220756</a> | human   | 60                                    | *886,468,054                     | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | HapMap - CEU                                    |
| pickrell   | <a href="#">20220758</a> | human   | 69                                    | *886,468,054                     | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | HapMap - YRI                                    |
| gutten     | <a href="#">18500741</a> | human   | 4                                     | 6,573,642                        | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | cell type                                       |
| katz.mouse | <a href="#">21057496</a> | mouse   | 4                                     | 14,368,471                       | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | control vs.<br>CUG-BP1 knockdown                |

<http://bowtie-bio.sourceforge.net/recount/>

# View the source

The screenshot shows a web browser window with the URL [bowtie-bio.sourceforge.net/recount/](http://bowtie-bio.sourceforge.net/recount/). A context menu is open over the first row of a table, with the 'View Page Source' option highlighted by a red box.

**Notes**  
Brief description of experiment.  
Please note that to use the ExpressionSets below, you will

**Back**  
Forward  
Reload  
Save As...  
Print...

**The Datasets**

| Study      | PMID                     | Species | Number of biological replicates    | Number of samples | View Page Source                | onSet                           | Count                           | Phenotype table      | Notes                                           |
|------------|--------------------------|---------|------------------------------------|-------------------|---------------------------------|---------------------------------|---------------------------------|----------------------|-------------------------------------------------|
| bodymap    | <a href="#">22496456</a> | human   | 19                                 | 834,584,950       | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a> | Illumina Human BodyMap 2.0 -- tissue comparison |
| cheung     | <a href="#">20856902</a> | human   | 41                                 | 8,670,342         | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a> | HapMap - CEU                                    |
| core       | <a href="#">19056941</a> | human   | 2                                  | 41,356,738        | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a> | lung fibroblasts                                |
| gilad      | <a href="#">20009012</a> | human   | 6                                  | 71,970,164        | <a href="#">original pooled</a> | <a href="#">original pooled</a> | <a href="#">original pooled</a> | <a href="#">link</a> | liver; males and females                        |
| maqc       | <a href="#">20167110</a> | human   | 14 (technical)**<br>2 (biological) | *886,468,054      | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a> | experiment: MAQC-2                              |
| montgomery | <a href="#">20220756</a> | human   | 60                                 | 6,573,643         | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a> | HapMap - CEU                                    |
| pickrell   | <a href="#">20220758</a> | human   | 69                                 | 223,929,919       | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a> | HapMap - YRI                                    |
| sultan     | <a href="#">18599741</a> | human   | 4                                  | 14,368,471        | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a> | cell type comparison                            |
| wang       | <a href="#">18978772</a> | human   | 22                                 | 14,368,471        | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a> | tissue comparison                               |
| katz.mouse | <a href="#">21057496</a> | mouse   | 4                                  | 14,368,471        | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a>            | <a href="#">link</a> | control vs. CUG-BP1 knockdown                   |

# What the computer sees

Brief description of experiment.  
<br /><br />

Please note that to use the ExpressionSets below, you will need to install [Bioconductor](http://www.bioconductor.org/) and run the command  
<tt>library(Biobase)</tt>

```
<h3>The Datasets</h3>
<div id="recounttab">
<table class="sortable"><tbody>
<tr>
<td>Study</td>
<td>PMID</td>
<td>Species</td>
<td>Number of biological replicates</td>
<td>Number of uniquely aligned reads</td>
<td>ExpressionSet</td>
<td>Count table</td>
<td>Phenotype table</td>
<td>Notes</td>
</tr>

<tr>
<td>bodymap</td>
<td><a href="http://www.ncbi.nlm.nih.gov/pubmed/22496456">22496456</a></td>
<td>human</td>
<td>19</td>
<td>2,197,622,796</td>
<td><a href=".(ExpressionSets/bodymap_eset.RData">link </a></td>
<td><a href=".(/countTables/bodymap_count_table.txt">link</a></td>
<td><a href=".(/phenotypeTables/bodymap_phenodata.txt">link</a></td>
<td> Illumina Human BodyMap 2.0 -- tissue comparison</td>
</tr>

<tr>
<td>cheung</td>
<td><a href="http://www.ncbi.nlm.nih.gov/pubmed?term=20856902">20856902</a></td>
<td>human</td>
<td>41</td>
<td>834,584,950</td>
<td><a href=".(ExpressionSets/cheung_eset.RData">link </a></td>
<td><a href=".(/countTables/cheung_count_table.txt">link</a></td>
<td><a href=".(/phenotypeTables/cheung_phenodata.txt">link</a></td>
<td> HapMap - CEU</td>
</tr>

<tr><td>core</td>
<td><a href="http://www.ncbi.nlm.nih.gov/pubmed?term=19056941">19056941</a></td>
<td>human</td>
```

# Inspect element

Screenshot of a web browser showing a table of datasets from ReCount. A context menu is open over the first row, with the "Inspect Element" option highlighted by a red box.

The table has columns: Study, PMID, Species, Number of biological replicates, OnSet, Count table, Phenotype table, and Notes.

Context menu options include: Back, Forward, Reload, Save As..., Print..., Translate to English, View Page Source, View Page Info, and Inspect Element.

Study	PMID	Species	Number of biological replicates	OnSet	Count table	Phenotype table	Notes
bodymap	<a href="#">22496456</a>	human	19	<a href="#">link</a>	<a href="#">link</a>	<a href="#">link</a>	Illumina Human BodyMap 2.0 -- tissue comparison
cheung	<a href="#">20856902</a>	human	41	834,584,950	<a href="#">link</a>	<a href="#">link</a>	<a href="#">link</a>
core	<a href="#">19056941</a>	human	2	8,670,342	<a href="#">link</a>	<a href="#">link</a>	lung fibroblasts
gilad	<a href="#">20009012</a>	human	6	41,356,738	<a href="#">link</a>	<a href="#">link</a>	liver; males and females
maqc	<a href="#">20167110</a>	human	14 (technical)** 2 (biological)	71,970,164	original pooled	original pooled	original pooled experiment: MAQC-2
montgomery	<a href="#">20220756</a>	human	60	*886,468,054	<a href="#">link</a>	<a href="#">link</a>	<a href="#">link</a>
pickrell	<a href="#">20220758</a>	human	69	*886,468,054	<a href="#">link</a>	<a href="#">link</a>	<a href="#">link</a>
sultan	<a href="#">18599741</a>	human	4	6,573,643	<a href="#">link</a>	<a href="#">link</a>	<a href="#">link</a>
wang	<a href="#">18978772</a>	human	22	223,929,919	<a href="#">link</a>	<a href="#">link</a>	<a href="#">link</a>
katz.mouse	<a href="#">21057496</a>	mouse	4	14,368,471	<a href="#">link</a>	<a href="#">link</a>	<a href="#">link</a>

# Copy XPath

The screenshot shows a web browser window with the URL [bowtie-bio.sourceforge.net/recount/](http://bowtie-bio.sourceforge.net/recount/). A table titled "The Datasets" is displayed, listing various studies with their details like PMID, species, and number of replicates. A context menu is open over the table, with the "Copy XPath" option highlighted by a red box. The browser's developer tools are visible at the bottom, showing the element inspector and styles panel.

Study	PMID	Species	Number of biological replicates	Number of uniquely aligned reads	ExpressionSet	Count table	Phenotype table	Notes
bodymap	22496456	human	19	2,197,622,796	link	link	link	Illumina Human BodyMap 2.0 -- tissue comparison
bottomly	21455293	mouse	21	343,445,340	link	link	link	2 inbred mouse strains
cheung	20856902	human	41	834,584,950	link	link	link	HapMap - CEU
core	19056941	human	2	8,670,342	link	link	link	lung fibroblasts
gilad	20009012	human	6	41,356,738	link	link	link	liver; males and females

Please note that to use the ExpressionSets below, you will need to install [Bioconductor](#) and run the command `library(BioBase)`

Elements Network Sources Timeline Profiles Resources Audits Console

Add Attribute Force Element State Edit as HTML

Copy XPath

Styles Computed Event Listeners »

```
element.style { }
media="screen" #recounttab table {
  margin: 1em;
  margin-top: 15px;
  border-collapse: collapse;
}
media="screen" *
  padding: 0;
  margin: 0;
}
table {
  user agent stylesheet
  display: table;
  border-collapse: separate;
```

# rvest package

```
recount_url = "http://bowtie-bio.sourceforge.net/recount"
```

```
install.packages("rvest")
library(rvest)
htmlfile = html(recount_url)

nds = html_nodes(htmlfile,
                 xpath='//*[@id="recounttab"]/table')
dat = html_table(nds)
dat = as.data.frame(dat)
```

# Data from the internetz!

```
head(dat)
```

	X1	X2	X3	X4
1	Study	PMID	Species	Number of biological replicates
2	bodymap	22496456	human	19
3	cheung	20856902	human	41
4	core	19056941	human	2
5	gilad	20009012	human	6

# Reading

# Data: APIs

# APIs: Application programming interface

The screenshot shows the Facebook Developers website at <https://developers.facebook.com>. The main navigation bar includes links for Developers, My Apps, Products, Docs, Tools & Support, and News. A search bar and a Log In button are also present.

The main content area features a large banner for "Facebook Analytics for Apps". The banner text reads: "Facebook Analytics for Apps" and "Understand how your customers use your app across all of their devices." It includes a "Learn More" button. To the right of the banner is a screenshot of the Facebook Analytics dashboard, which displays various metrics and graphs for an app's performance.

Below the banner are four sections: "App Monetization", "App Invites", "Social Plugins", and "Messenger". Each section has an icon, a title, a brief description, and a "Learn More" link.

A prominent blue bar at the bottom of the page contains the URL <https://developers.facebook.com/>.

App Monetization		App Invites		Social Plugins		Messenger	
Monetize with ads and publisher tools	Learn More	Let people recommend your app to friends	Learn More	Easily make your app or website social	Learn More	Grow your app by powering conversations	Learn More

# In Biology Too!

NCBI Resources How To

Bookshelf Books Search Help

## Entrez Programming Utilities Help

Bethesda (MD): National Center for Biotechnology Information (US); 2010-.  
Copyright and Permissions

Search this book

Views  
PubReader  
Print View  
Cite this Page

Other titles in this collection  
NCBI Help Manual

Related information  
NLM Catalog

Recent Activity  
Turn Off Clear

Entrez Programming Utilities Help  
Web scraping technologies in an API world.  
PubMed  
Zaykin DV[auth] (34)  
PubMed

Introduction to the E-utilities

- YouTube E-utilities Introduction
- Please see the [Release Notes](#) for details and changes.

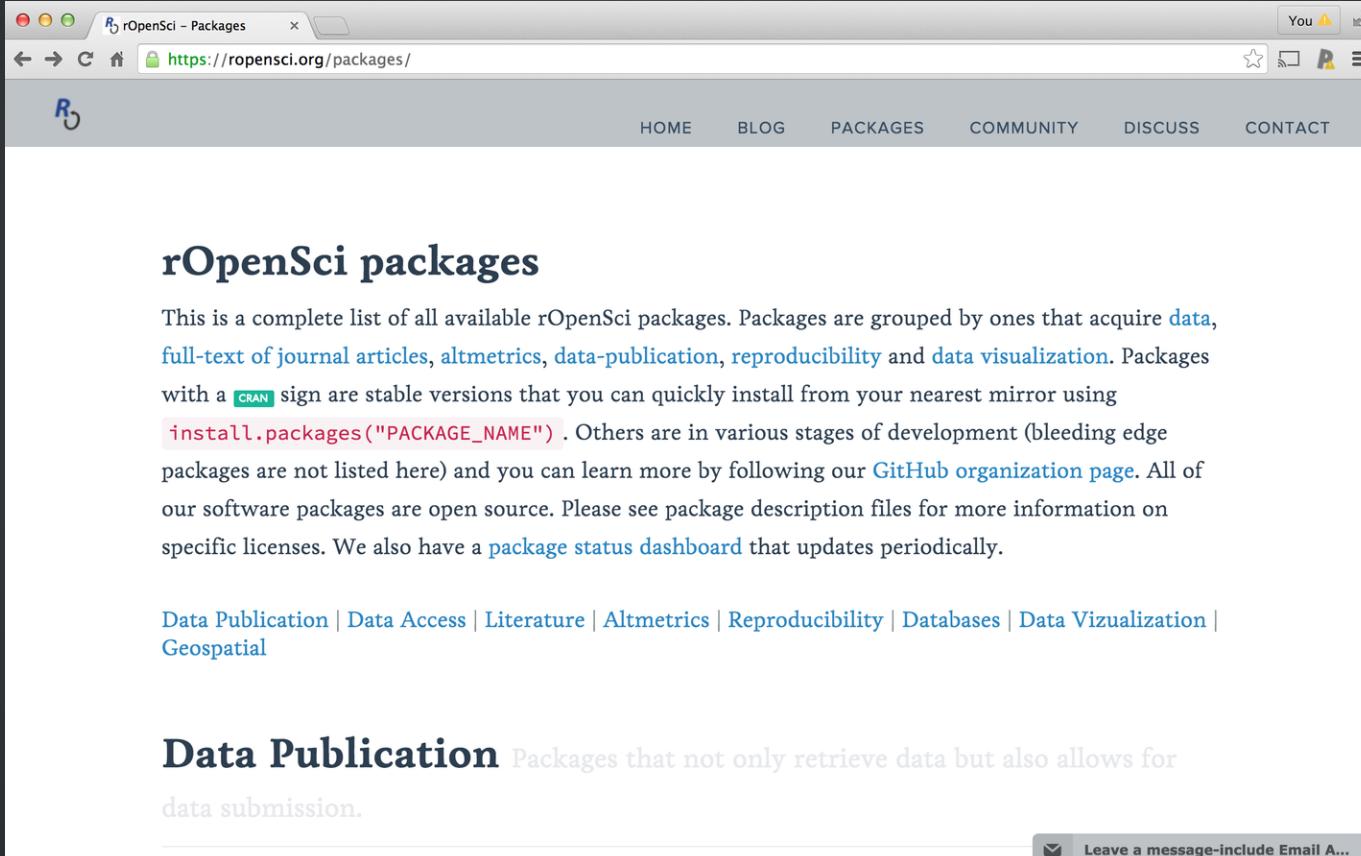
The Entrez Programming Utilities (E-utilities) are a set of eight server-side programs that provide a stable interface into the Entrez query and database system at the National Center for Biotechnology Information (NCBI). The E-utilities use a fixed URL syntax that translates a standard set of input parameters into the values necessary for various NCBI software components to search for and retrieve the requested data. The E-utilities are therefore the structured interface to the Entrez system, which currently includes 38 databases covering a variety of biomedical data, including nucleotide and protein sequences, gene records, three-dimensional molecular structures, and the biomedical literature.

Contents

Expand All Collapse All

<http://www.ncbi.nlm.nih.gov/books/NBK25501/>

# Step 0: Did someone already do this?



The screenshot shows a web browser window with the title bar "rOpenSci - Packages". The address bar displays the URL "https://ropensci.org/packages/". The page content is titled "rOpenSci packages". It describes a complete list of available rOpenSci packages, grouped by categories like data, full-text of journal articles, altmetrics, data-publication, reproducibility, and data visualization. It mentions CRAN stable versions and bleeding edge packages. A code snippet for installing packages is shown. Below the main text, there's a navigation bar with links to Data Publication, Data Access, Literature, Altmetrics, Reproducibility, Databases, Data Vizualization, and Geospatial. At the bottom, there's a "Data Publication" section and a "Leave a message" input field.

This is a complete list of all available rOpenSci packages. Packages are grouped by ones that acquire [data](#), [full-text of journal articles](#), [altmetrics](#), [data-publication](#), [reproducibility](#) and [data visualization](#). Packages with a [CRAN](#) sign are stable versions that you can quickly install from your nearest mirror using `install.packages("PACKAGE_NAME")`. Others are in various stages of development (bleeding edge packages are not listed here) and you can learn more by following our [GitHub organization page](#). All of our software packages are open source. Please see package description files for more information on specific licenses. We also have a [package status dashboard](#) that updates periodically.

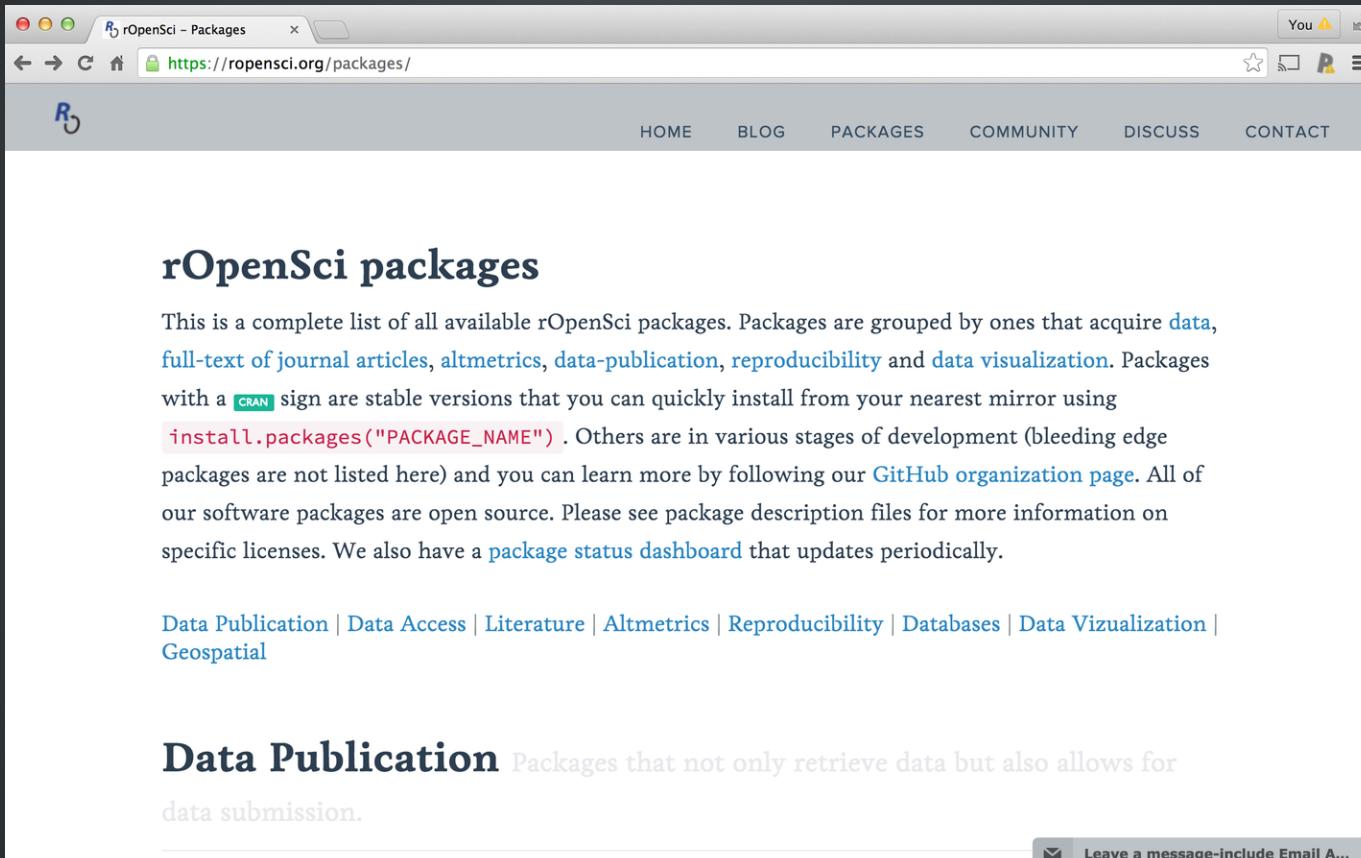
[Data Publication](#) | [Data Access](#) | [Literature](#) | [Altmetrics](#) | [Reproducibility](#) | [Databases](#) | [Data Vizualization](#) | [Geospatial](#)

**Data Publication** Packages that not only retrieve data but also allows for data submission.

Leave a message-include Email A...

<https://ropensci.org/>

# Figshare



The screenshot shows a web browser window with the URL <https://ropensci.org/packages/>. The page title is "rOpenSci - Packages". The main content area features a large heading "rOpenSci packages". Below it is a detailed paragraph about the package list, mentioning data acquisition, altmetrics, reproducibility, and visualization. It also discusses CRAN stable versions and GitHub development. At the bottom of the page, there's a footer with links to Data Publication, Data Access, Literature, Altmetrics, Reproducibility, Databases, Data Visualization, and Geospatial. A "Data Publication" section is highlighted with a dark background and white text.

This is a complete list of all available rOpenSci packages. Packages are grouped by ones that acquire [data](#), [full-text of journal articles](#), [altmetrics](#), [data-publication](#), [reproducibility](#) and [data visualization](#). Packages with a [CRAN](#) sign are stable versions that you can quickly install from your nearest mirror using `install.packages("PACKAGE_NAME")`. Others are in various stages of development (bleeding edge packages are not listed here) and you can learn more by following our [GitHub organization page](#). All of our software packages are open source. Please see package description files for more information on specific licenses. We also have a [package status dashboard](#) that updates periodically.

Data Publication | Data Access | Literature | Altmetrics | Reproducibility | Databases | Data Vizualization | Geospatial

**Data Publication** Packages that not only retrieve data but also allows for data submission.

[!\[\]\(5f19047134c6df3b36406db388ba5f61\_img.jpg\) Leave a message-include Email A...](#)

<https://ropensci.org/>

# Figshare API wrapper

```
install.packages("figshare")
library(figshare)
leeksearch = fs_search("Leek")

length(leeksearch)
[1] 56

leeksearch[[1]]
$article_id
[1] 1405638

$title
[1] "GEUVADIS expressed regions coverage matrix"

$DOI
[1] "http://dx.doi.org/10.6084/m9.figshare.1405638"

	description
```

# Do it yourself

The screenshot shows a web browser window displaying the GitHub API documentation for the v3 search endpoint. The URL in the address bar is <https://developer.github.com/v3/search/>. The page has a dark background with white text. At the top, there's a navigation bar with links for 'Reference', 'Webhooks', 'Guides', and 'Libraries'. Below this, the main content area has a title 'Search' and a list of search types: 'Search repositories', 'Search code', 'Search issues', 'Search users', and 'Text match metadata'. There are also sections for 'About the Search API' and 'Ranking search results'. A sidebar on the right contains a tree-like navigation menu with sections like 'Overview', 'Activity', 'Gists', 'Git Data', 'Issues', 'Miscellaneous', 'Organizations', 'Pull Requests', 'Repositories', 'Search' (which is expanded to show 'Repositories', 'Code', 'Issues', 'Users', and 'Legacy Search'), 'Users', and 'Enterprise 2.2'. At the bottom of the page, there's a footer with the URL <https://developer.github.com/v3/> and a note about authentication.

API

Reference Webhooks Guides Libraries

## Search

- i. [Search repositories](#)
- ii. [Search code](#)
- iii. [Search issues](#)
- iv. [Search users](#)
- v. [Text match metadata](#)

### About the Search API

The Search API is optimized to help you find the specific item you're looking for (e.g., a specific user, a specific file in a repository, etc.). Think of it the way you think of performing a search on Google. It's designed to help you find the one result you're looking for (or maybe the few results you're looking for). Just like searching on Google, you sometimes want to see a few pages of search results so that you can find the item that best meets your needs. To satisfy that need, the GitHub Search API provides **up to 1,000 results for each search**.

### Ranking search results

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

### Rate limit

quests using [Basic Authentication](#), [OAuth](#), or [client ID](#)

▶ Overview  
▶ Activity  
▶ Gists  
▶ Git Data  
▶ Issues  
▶ Miscellaneous  
▶ Organizations  
▶ Pull Requests  
▶ Repositories  
▼ Search  
  Repositories  
  Code  
  Issues  
  Users  
  Legacy Search  
▶ Users  
▶ Enterprise 2.2

# Read the docs

API

Reference   Webhooks   Guides   Libraries

## Search

- i. [Search repositories](#)
- ii. [Search code](#)
- iii. [Search issues](#)
- iv. [Search users](#)
- v. [Text match metadata](#)

### About the Search API

The Search API is optimized to help you find the specific item you're looking for (e.g., a specific user, a specific file in a repository, etc.). Think of it the way you think of performing a search on Google. It's designed to help you find the one result you're looking for (or maybe the few results you're looking for). Just like searching on Google, you sometimes want to see a few pages of search results so that you can find the item that best meets your needs. To satisfy that need, the GitHub Search API provides **up to 1,000 results for each search**.

### Ranking search results

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

▶ Overview

▶ Activity

▶ Gists

▶ Git Data

▶ Issues

▶ Miscellaneous

▶ Organizations

▶ Pull Requests

▶ Repositories

▼ Search

[Repositories](#)

[Code](#)

[Issues](#)

[Users](#)

[Legacy Search](#)

▶ Users

# Read the docs

The screenshot shows a dark-themed web page for the GitHub API documentation. At the top, there's a navigation bar with links for 'API', 'Reference', 'Webhooks', 'Guides', and 'Libraries'. Below the navigation, the word 'Search' is prominently displayed. To the right of 'Search', there's a button labeled '▶ Overview'. The main content area has a white background and contains several sections:

- Rate limit**: A section explaining the custom rate limit for authenticated requests (30 per minute) and unauthenticated requests (10 per minute). It links to 'rate limit documentation' for more details.
- Ranking search results**: A section explaining how results are sorted by best match based on the `score` field.
- Repositories**: A list of search categories including 'Repositories', 'Code', 'Issues', 'Users', and 'Legacy Search'.
- Search**: A list of search categories including 'Repositories', 'Code', 'Issues', 'Users', and 'Legacy Search'.
- Users**: A list of search categories including 'Repositories', 'Code', 'Issues', 'Users', and 'Legacy Search'.

The 'Search' category under 'Repositories' is currently selected, indicated by a blue border around its link.

# Read the docs

API

Reference   Webhooks   Guides   Libraries

## Example

Suppose you want to find the definition of the `addClass` function inside `jQuery`. Your query would look something like this:

```
https://api.github.com/search/code?q=addClass+in:file+language:js+repo:jquery/jquery
```

Here, we're searching for the keyword `addClass` within a file's contents. We're making sure that we're only looking in files where the language is JavaScript. And we're scoping the search to the `repo:jquery/jquery` repository.

### Ranking search results

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

- Code
- Issues
- Users
- Legacy Search
- ▶ Users

# A dissected example

API Reference Webhooks Guides Libraries

## Search

- i. [Search repositories](#)
- ii. [Search code](#)
- iii. [Search issues](#)
- iv. [Search users](#)
- v. [Text match metadata](#)

**<https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran&type>**

[About the Search API](#)

The Search API is optimized to help you find the specific item you're looking for (e.g., a specific user, a specific file in a repository, etc.). Think of it the way you think of performing a search on Google. It's designed to help you find the one result you're looking for (or maybe the few results you're looking for). Just like searching on Google, you sometimes want to see a few pages of search results so that you can find the item that best meets your needs. To satisfy that need, the GitHub Search API provides **up to 1,000 results for each search**.

**Ranking search results**

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

- ▶ Overview
- ▶ Activity
- ▶ Gists
- ▶ Git Data
- ▶ Issues
- ▶ Organizations
- ▶ Pull Requests
- ▶ Repositories
- ▼ Search
  - Repositories
  - Code
  - Issues
  - Users
  - Legacy Search
- ▶ Users

# The Base URL

The screenshot shows the GitHub API documentation for the Search API. The main content area displays the URL for a search query: `https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran`. Below the URL, there's a section titled "About the Search API" which explains the purpose and functionality of the search feature. Another section, "Ranking search results", discusses how results are sorted. On the right side, a sidebar navigation menu is visible, showing various categories like Overview, Activity, Gists, Git Data, Issues, Miscellaneous, Organizations, Pull Requests, Repositories, and Users.

API

Reference Webhooks Guides Libraries

## Search

- i. [Search repositories](#)
- ii. [Search code](#)
- iii. [Search issues](#)
- iv. [Text match metadata](#)

`https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran`

About the Search API

The Search API is optimized to help you find the specific item you're looking for (e.g., a specific user, a specific file in a repository, etc.). Think of it the way you think of performing a search on Google. It's designed to help you find the one result you're looking for (or maybe the few results you're looking for). Just like searching on Google, you sometimes want to see a few pages of search results so that you can find the item that best meets your needs. To satisfy that need, the GitHub Search API provides **up to 1,000 results for each search**.

**Ranking search results**

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

▶ Overview

▶ Activity

▶ Gists

▶ Git Data

▶ Issues

▶ Miscellaneous

▶ Organizations

▶ Pull Requests

▶ Repositories

▼ Search

    Repositories

    Code

    Issues

    Users

    Legacy Search

▶ Users

# Search repositories (repos)

The screenshot shows the GitHub API documentation for the Search API. At the top, there are navigation links for 'API', 'Reference', 'Webhooks', 'Guides', and 'Libraries'. On the left, a sidebar lists search categories: 'Search repositories', 'Search code', 'Search issues', 'Search pull requests', and 'Text match metadata'. Below this, a large example URL is displayed: `https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran`. To the right of the URL is a detailed sidebar menu with sections like 'Overview', 'Activity', 'Gists', 'Git Data', 'Issues', 'Miscellaneous', 'Organizations', 'Pull Requests', 'Repositories', 'Search' (which is expanded to show 'Repositories', 'Code', 'Issues', 'Users', 'Legacy Search'), and 'Users'.

API Reference Guides Libraries

Search

- i. [Search repositories](#)
- ii. [Search code](#)
- iii. [Search issues](#)
- iv. [Search pull requests](#)
- v. [Text match metadata](#)

`https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran`

About the Search API

The Search API is optimized to help you find the specific item you're looking for (e.g., a specific user, a specific file in a repository, etc.). Think of it the way you think of performing a search on Google. It's designed to help you find the one result you're looking for (or maybe the few results you're looking for). Just like searching on Google, you sometimes want to see a few pages of search results so that you can find the item that best meets your needs. To satisfy that need, the GitHub Search API provides **up to 1,000 results for each search**.

**Ranking search results**

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

# Create a query

API

Reference   Webhooks   Guides   Libraries

## Search

i. [Search repositories](#)

ii. [Search code](#)

iii. [Search issues](#)

iv. [Search pull requests](#)

v. [Text match metadata](#)

**<https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran>**

### About the Search API

The Search API is optimized to help you find the specific item you're looking for (e.g., a specific user, a specific file in a repository, etc.). Think of it the way you think of performing a search on Google. It's designed to help you find the one result you're looking for (or maybe the few results you're looking for). Just like searching on Google, you sometimes want to see a few pages of search results so that you can find the item that best meets your needs. To satisfy that need, the GitHub Search API provides **up to 1,000 results for each search**.

### Ranking search results

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

▶ Overview

▶ Activity

▶ Gists

▶ Git Data

▶ Issues

▶ Miscellaneous

▶ Organizations

▶ Pull Requests

▶ Repositories

▼ Search

  Repositories

  Code

  Issues

  Users

  Legacy Search

▶ Users

# Date the repo was created

The screenshot shows a dark-themed GitHub API documentation page. At the top, there's a navigation bar with 'API' on the left and 'Reference', 'Webhooks', 'Guides', and 'Libraries' on the right. Below this is a sidebar with sections like 'Overview', 'Activity', 'Gists', 'Git Data', 'Issues', 'Miscellaneous', 'Organizations', 'Pull Requests', 'Repositories', and 'Search'. The 'Search' section is expanded, showing 'Repositories', 'Code', 'Issues', 'Users', 'Legacy Search', and 'Users' again. The main content area has a heading 'Search' and a list of items: 'i. Search repositories', 'ii. Search code', 'iii. Search issues', 'iv. Search users', and 'v. Text match metadata'. Below this is a large URL box containing the GitHub API endpoint for searching repositories: `https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran`. Underneath the URL, there's a section titled 'About the Search API' with a detailed description of how it works.

API

Reference Webhooks Guides Libraries

Search

- i. [Search repositories](#)
- ii. [Search code](#)
- iii. [Search issues](#)
- iv. [Search users](#)
- v. [Text match metadata](#)

`https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran`

About the Search API

The Search API is optimized to help you find the specific item you're looking for (e.g., a specific user, a specific file in a repository, etc.). Think of it the way you think of performing a search on Google. It's designed to help you find the one result you're looking for (or maybe the few results you're looking for). Just like searching on Google, you sometimes want to see a few pages of search results so that you can find the item that best meets your needs. To satisfy that need, the GitHub Search API provides **up to 1,000 results for each search**.

**Ranking search results**

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

# Language of repo

API

Reference   Webhooks   Guides   Libraries

## Search

- i. [Search repositories](#)
- ii. [Search code](#)
- iii. [Search issues](#)
- iv. [Search pull requests](#)
- v. [Text match metadata](#)

<https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran>

### About the Search API

The Search API is optimized to help you find the specific item you're looking for (e.g., a specific user, a specific file in a repository, etc.). Think of it the way you think of performing a search on Google. It's designed to help you find the one result you're looking for (or maybe the few results you're looking for). Just like searching on Google, you sometimes want to see a few pages of search results so that you can find the item that best meets your needs. To satisfy that need, the GitHub Search API provides **up to 1,000 results for each search**.

### Ranking search results

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

▶ Overview

▶ Activity

▶ Gists

▶ Git Data

▶ Issues

▶ Miscellaneous

▶ Organizations

▶ Pull Requests

▶ Repositories

▼ Search

  ▶ Repositories

  ▶ Code

  ▶ Issues

  ▶ Users

  ▶ Legacy Search

▶ Users

# Ignore repos from "cran"

The screenshot shows a dark-themed GitHub API documentation page. At the top, there's a navigation bar with 'API' on the left and 'Reference', 'Webhooks', 'Guides', and 'Libraries' on the right. Below the navigation is a sidebar with sections like 'Overview', 'Activity', 'Gists', 'Git Data', 'Issues', 'Miscellaneous', 'Organizations', 'Pull Requests', 'Repositories', 'Search' (which is expanded to show 'Repositories', 'Code', 'Issues', 'Users', 'Legacy Search'), and 'Users'. The main content area has a heading 'Search' and a list of search types: 'Search repositories', 'Search code', 'Search issues', 'Search pull requests', and 'Text match metadata'. A large, semi-transparent text box highlights a GitHub API endpoint: `https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran`. Below this, there's a section titled 'About the Search API' with a detailed description of how it works, mentioning up to 1,000 results per search. There's also a 'Ranking search results' section explaining the scoring mechanism.

API

Reference Webhooks Guides Libraries

Search

- i. [Search repositories](#)
- ii. [Search code](#)
- iii. [Search issues](#)
- iv. [Search pull requests](#)
- v. [Text match metadata](#)

`https://api.github.com/search/repositories?q=created:2014-08-13+language:r+-user:cran`

About the Search API

The Search API is optimized to help you find the specific item you're looking for (e.g., a specific user, a specific file in a repository, etc.). Think of it the way you think of performing a search on Google. It's designed to help you find the one result you're looking for (or maybe the few results you're looking for). Just like searching on Google, you sometimes want to see a few pages of search results so that you can find the item that best meets your needs. To satisfy that need, the GitHub Search API provides **up to 1,000 results for each search**.

Ranking search results

Unless another sort option is provided as a query parameter, results are sorted by best match, as indicated by the `score` field for each item returned. This is a computed value representing the relevance of an item relative to the other items in the result set. Multiple factors are combined to boost the most relevant item to the top of the result list.

▶ Overview

▶ Activity

▶ Gists

▶ Git Data

▶ Issues

▶ Miscellaneous

▶ Organizations

▶ Pull Requests

▶ Repositories

▼ Search

- Repositories
- Code
- Issues
- Users
- Legacy Search

▶ Users

# httr package

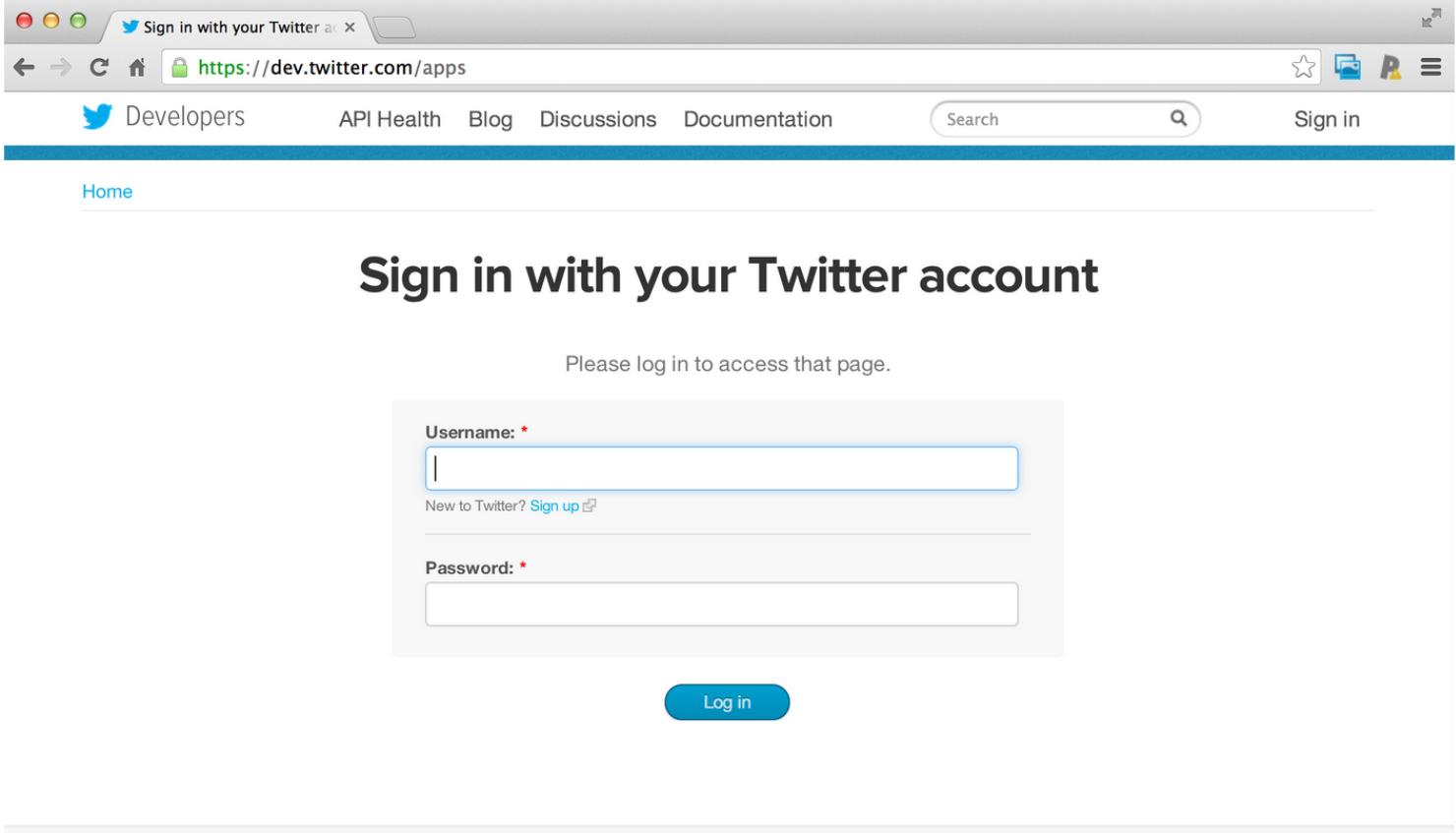
```
query_url = "https://api.github.com/search/repositories?q=created:2014-08-13+language:r+user:cran"
```

```
install.packages("httr")
library(httr)
req = GET(query_url)
names(content(req))
[1] "total_count"           "incomplete_results"

content(req)$items[[1]]
$id
[1] 22907468

$name
[1] "computel"
```

# Not all APIs are "open"



The screenshot shows a web browser window with the following details:

- Title Bar:** "Sign in with your Twitter account" (partially visible).
- Address Bar:** <https://dev.twitter.com/apps>
- Header:** Developers, API Health, Blog, Discussions, Documentation, Search, Sign in.
- Section:** Home
- Main Content:** "Sign in with your Twitter account". Below it, a message says "Please log in to access that page." A login form follows:
  - Username:** Input field with placeholder text "New to Twitter? [Sign up](#)".
  - Password:** Input field.
  - Buttons:** "Log in" button.

<https://dev.twitter.com/apps>

# Not all APIs are "open"

```
myapp = oauth_app("twitter",
                  key="yourConsumerKeyHere", secret="yourConsumerSecretHere")
sig = sign_oauth1.0(myapp,
                     token = "yourTokenHere",
                     token_secret = "yourTokenSecretHere")
homeTL = GET("https://api.twitter.com/1.1/statuses/home_timeline.json", sig)
```

# But you can often get cool data

```
json1 = content(homeTL)
json2 = jsonlite::fromJSON(toJSON(json1))
json2[1,1:4]

      created_at           id       id_str
1 Mon Jan 13 05:18:04 +0000 2014 4.225984e+17 422598398940684288

1 Now that P. Norvig's regex golf IPython notebook hit Slashdot, let's see if
```

Web + API

Lab

<http://bit.ly/1S2oAsj>