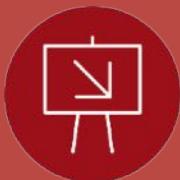




BANCO DE PORTUGAL
EUROSISTEMA

ACADEMIA
BANCO DE PORTUGAL



“Machine Learning” and “Big Data”: basic concepts and applications

Nelson Areal • Universidade do Minho
18 December 2018

Training session 1

Introductions

Nelson Areal

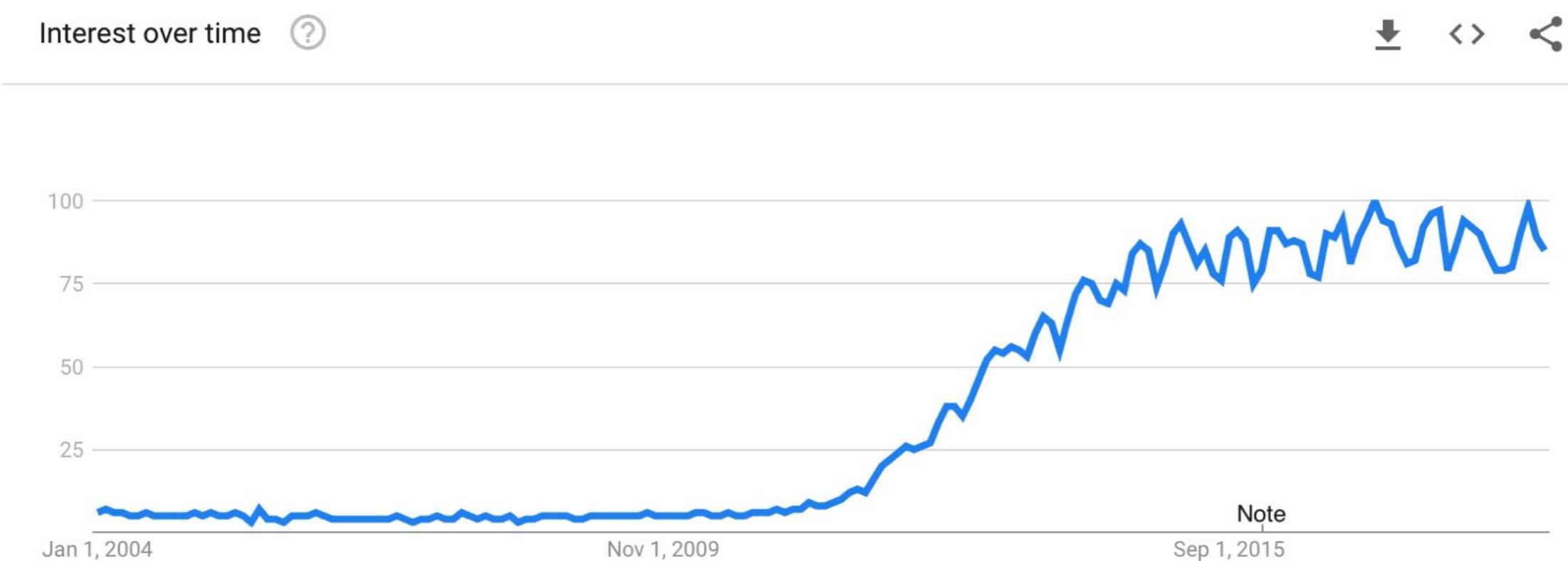
Management Department
School of Economics and Management
University of Minho

Twitter: @nareal

URL: <http://nelsonareal.net>

Big data

Big data



<https://trends.google.com/trends/explore?date=all&q=Big%20data>

Big data



a trip down to memory lane...

Big data



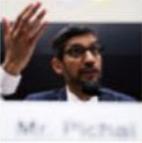
Francis X. Diebold

Diebold, Francis X. (2003) "**Big data dynamic factor models for macroeconomic measurement and forecasting**" In Advances in Economics and Econometrics: Theory and Applications, Eighth World Congress of the Econometric Society,"(edited by M. Dewatripont, LP Hansen and S. Turnovsky), pp. 115-122.

Big data

≡ SECTIONS HOME SEARCH

The New York Times

 Google's Pichai Faces Privacy and Bias Questions in Congress

 Huawei Executive Granted Bail by Canadian Court

 Amazon's Homegrown Chips Threaten Silicon Valley Giant Intel

Bits Bits
Business, Innovation, Technology, Society

The Origins of 'Big Data': An Etymological Detective Story

BY STEVE LOHR FEBRUARY 1, 2013 9:10 AM

[Email](#)
[Share](#)
[Tweet](#)
[Save](#)
[More](#)

Words and phrases are fundamental building blocks of language and culture, much as genes and cells are to the biology of life. And words are how we express ideas, so tracing their origin,


Lloyd Miller for The New York Times

<https://bits.blogs.nytimes.com/2013/02/01/the-origins-of-big-data-an-etymological-detective-story/>

Big data

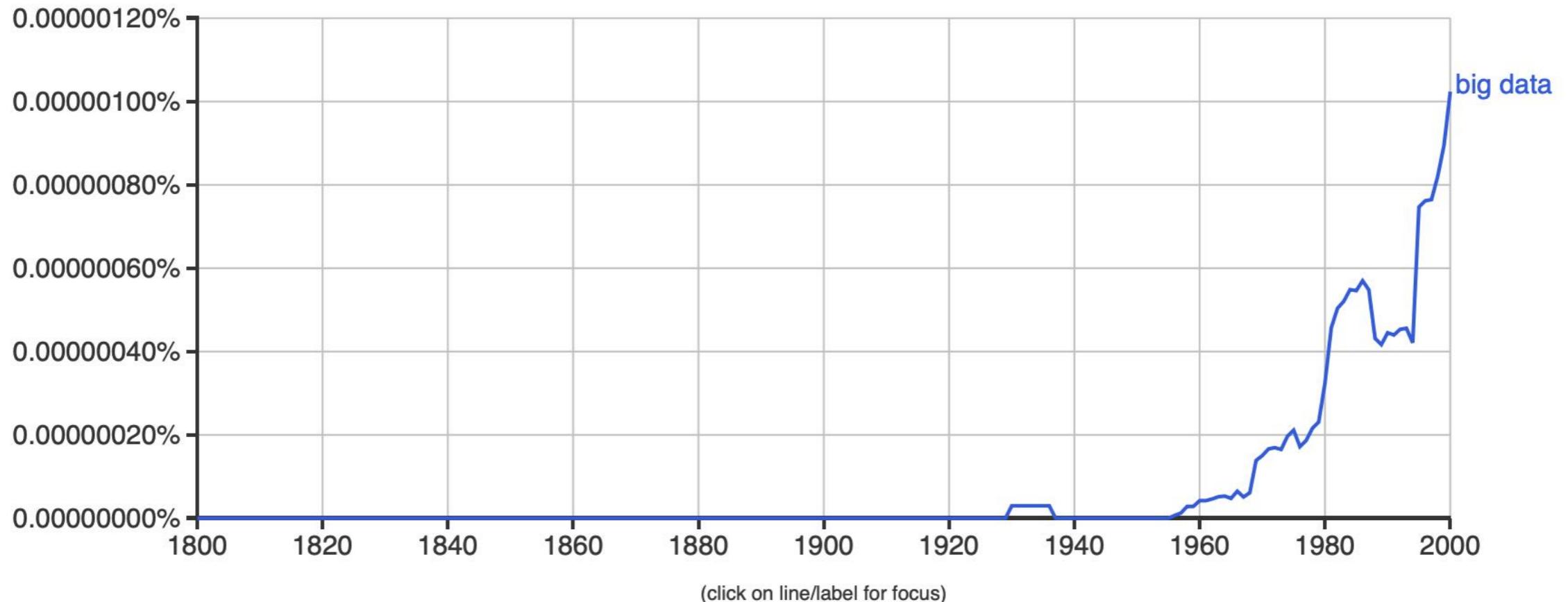
On the Origin(s) and Development of the Term “Big Data”*

Francis X. Diebold
University of Pennsylvania
fdiebold@sas.upenn.edu

First Draft, August 2012
This Draft, September 21, 2012

Abstract: I investigate the origins of the now-ubiquitous term “Big Data,” in industry and academics, in computer science and statistics/econometrics. Credit for coining the term must be shared. In particular, John Mashey and others at Silicon Graphics produced highly-relevant (unpublished, non-academic) work in the mid-1990s. The first significant academic references (independent of each other and of Silicon Graphics) appear to be Weiss and Indurkhya (1998) in computer science and Diebold (2000) in statistics/econometrics. Douglas Laney of Gartner also produced insightful work (again unpublished and non-academic) slightly later. Big Data the term is now firmly entrenched, Big Data the phenomenon continues unabated, and Big Data the discipline is emerging.

Big data



<https://books.google.com/ngrams/>

Big data

EARTH SCIENCE IN THE PUBLIC SERVICE

RESOURCE AND ENVIRONMENTAL DATA ANALYSIS

By DANIEL F. MERRIAM
Chairman, Department of Geology, Syracuse University

INTRODUCTION

All the predictions on the depletion of our national resources are pessimistic. We are exhausting our raw materials, and now it is only a question of when. It is of top importance, therefore, that optimum use be made of these resources as they are exploited, and it will be necessary to achieve a balance between use, conservation, and environmental considerations. As difficult as this may be, however, the real problem will be in finding new reserves to replace the depleted ones. Geologists will play a key role in the exploration for these new mineral resources.

of commodities."

The new philosophy will reflect team inquiry. The geologist now will integrate his thinking with computer scientists, statisticians, engineers, hydrologists, geochemists, and geophysicists in outlining areas of interest and in monitoring their development. Methods will reflect the latest available techniques and perhaps simulated real-world models based on millions or even billions of items of data. New ideas will be put forth by the younger better disciplined geologists who will have been trained to interpret and judge the interdisciplinary multi-based data that have been treated in a complex

<https://books.google.pt/books/content?id=64grjG0MLRUC&pg=PA41&img=1&zoom=3&hl=en&sig=ACfU3U0lfQ8jiFg0uvAuByxcbiyUNWuRpQ&ci=55%2C114%2C409%2C152&edge=0>

Big data

"In the future big data storage and retrieval will be put into use and small specialized systems will proliferate. Standards and quality control will gradually be installed at all levels and geologists will eventually record all their data on standard forms utilized worldwide. The US Geological Survey as already pointed out can continue to be an effective leader in this area of electronic data processing where so much expertise is available from years of experience" (Merriam, 1974)

Big data

My definition

Regular size

Biggish data

Big data

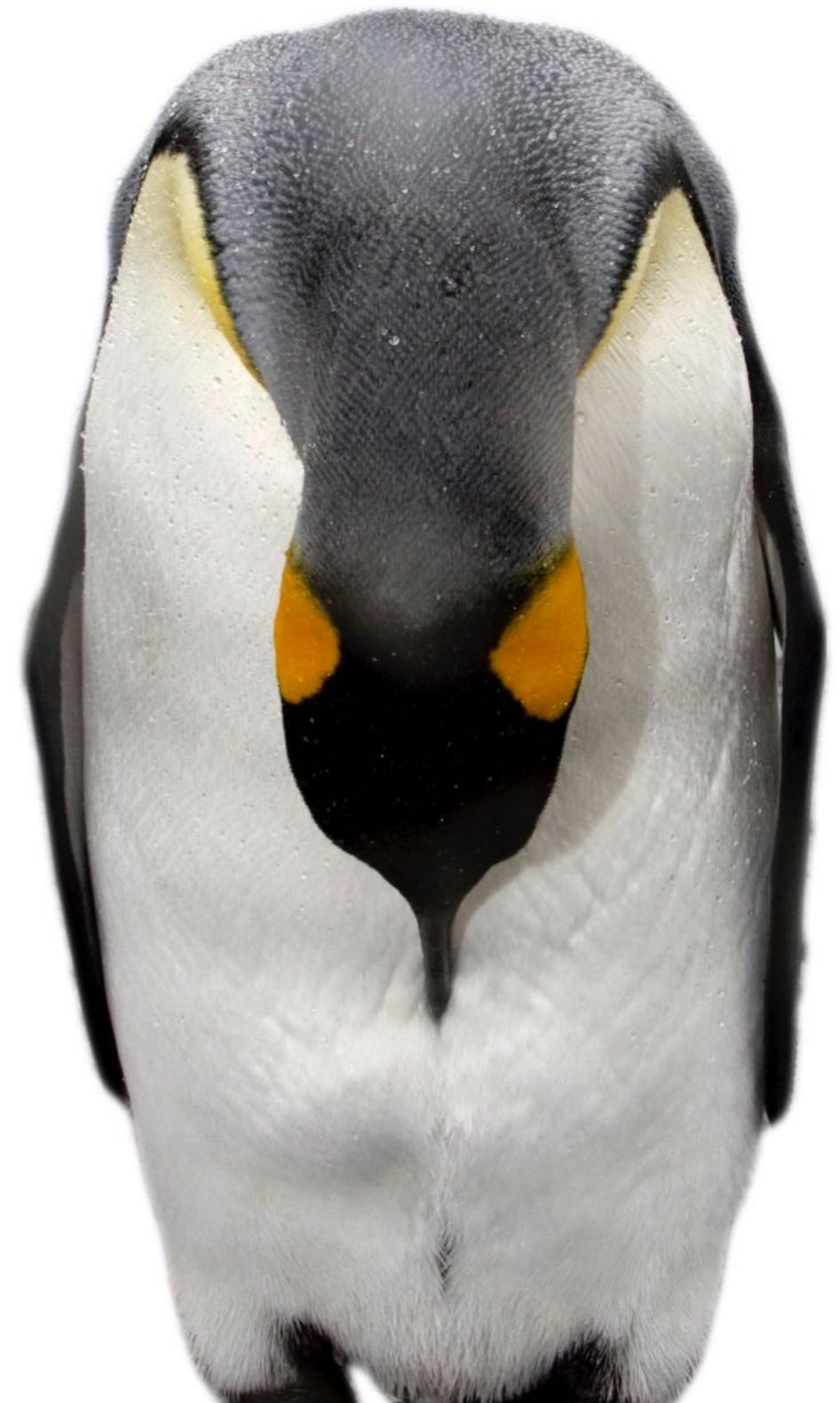
Anything that fits in memory



the deeper your pocket is the smaller the dataset will look

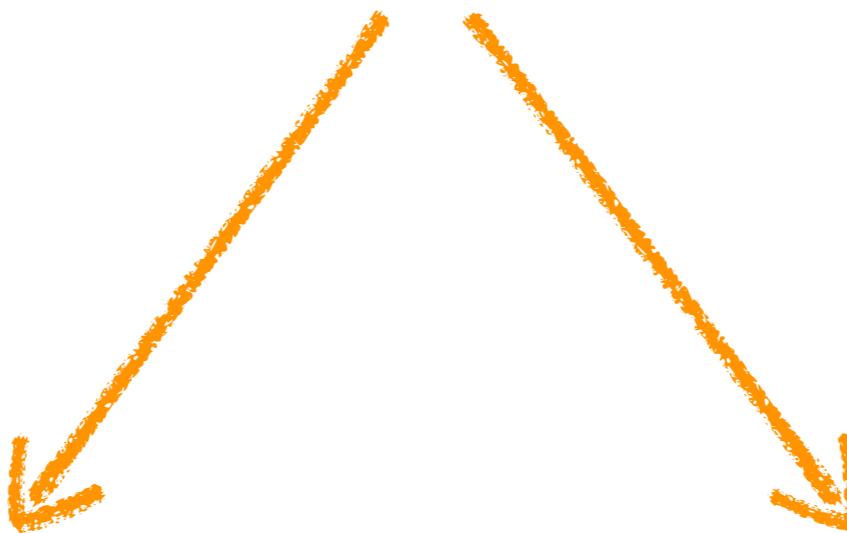
Big data

**Does size
matters?**



Big data

Does size matters?



**Harder to clean, validate
and manipulate**

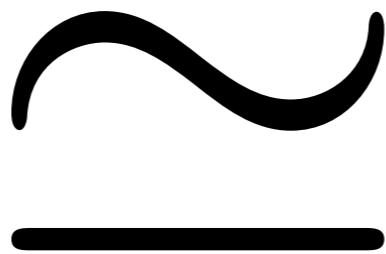
Results

Preliminares

Tools



Tools



"I suppose it is tempting, if the only tool you have is a hammer, to treat everything as if it were a nail." Maslow (1966)

Tools



Top Gear - Jeremy Clarkson's sophisticated way of fixing his BMW Estate

<https://www.youtube.com/watch?v=6xsVX6029rU>

Tools

Reliability



Computational Statistics & Data Analysis

Volume 40, Issue 4, 28 October 2002, Pages 713-721



On the accuracy of statistical procedures in Microsoft Excel 2000 and Excel XP

B D. McCullough ^a✉, Berry Wilson ^b✉

[Show more](#)

[https://doi.org/10.1016/S0167-9473\(02\)00095-6](https://doi.org/10.1016/S0167-9473(02)00095-6)

[Get rights and content](#)

Abstract

The problems that rendered Excel 97 unfit for use as a statistical package have not been fixed in either Excel 2000 or Excel 2002 (also called “Excel XP”). Microsoft attempted to fix errors in the standard normal random number generator and the inverse normal function, and in the former case actually made the problem worse.

McCullough, B. D., & Wilson, B. (2002). On the accuracy of statistical procedures in Microsoft Excel 2000 and Excel XP. *Computational Statistics and Data Analysis*, 40, 713-721.

Tools

Reliability

“In the small world where computer science overlaps with statistics, it was well known that Microsoft Excel was riddled with statistical errors. It was so well known that no one bothered to write about it.”

McCullough, B. (2006). The unreliability of Excel's statistical procedures. *Foresight: International Journal of Applied Forecasting*, 3, 44-45.

Tools

Reliability



Journal of Statistical Software

January 2013, Volume 52, Issue 7.

<http://www.jstatsoft.org/>

Spreadsheets in the Cloud – Not Ready Yet

Bruce D. McCullough
Drexel University

A. Talha Yalta
TOBB University of Economics
and Technology

Abstract

Cloud computing is a relatively new technology that facilitates collaborative creation and modification of documents over the internet in real time. Here we provide an introductory assessment of the available statistical functions in three leading cloud spreadsheets namely **Google Spreadsheet**, Microsoft **Excel Web App**, and **Zoho Sheet**. Our results show that the developers of cloud-based spreadsheets are not performing basic quality control, resulting in statistical computations that are misleading and erroneous. Moreover, the developers do not provide sufficient information regarding the software and the hardware, which can change at any time without notice. Indeed, rerunning the tests after several months we obtained different and sometimes worsened results.

Keywords: cloud computing, spreadsheet, accuracy, **Google Docs**, **Excel**, **Zoho**, Wilkinson tests.

McCullough, B. D & Yalta, A. T. (2010). Spreadsheets in the Cloud – Not Ready Yet. *Journal of Statistical Software*, 52(7), 1-14.

Tools

Reliability



Journal of Statistical Software

April 2010, Volume 34, Issue 4.

<http://www.jstatsoft.org/>

On the Numerical Accuracy of Spreadsheets

Marcelo G. Almiron
Universidade Federal
de Alagoas

Bruno Lopes
Universidade Federal
de Alagoas

Alyson L. C. Oliveira
Universidade Federal
de Alagoas

Antonio C. Medeiros
Universidade Federal
de Alagoas

Alejandro C. Frery
Universidade Federal
de Alagoas

Abstract

This paper discusses the numerical precision of five spreadsheets (**Calc**, **Excel**, **Gnumeric**, **NeoOffice** and **Oleo**) running on two hardware platforms (i386 and amd64) and on three operating systems (Windows Vista, Ubuntu Intrepid and Mac OS Leopard). The methodology consists of checking the number of correct significant digits returned by each spreadsheet when computing the sample mean, standard deviation, first-order autocorrelation, *F* statistic in ANOVA tests, linear and nonlinear regression and distribution functions. A discussion about the algorithms for pseudorandom number generation provided by these platforms is also conducted. We conclude that there is no safe choice among the spreadsheets here assessed: they all fail in nonlinear regression and they are not suited for Monte Carlo experiments.

"This paper discusses the numerical precision of five spreadsheets (Calc, Excel, Gnumeric, NeoOffice and Oleo) running on two hardware platforms (i386 and amd64) and on three operating systems (Windows Vista, Ubuntu Intrepid and Mac OS Leopard).

(...)

We conclude that there is no safe choice among the spreadsheets here assessed: they all fail in nonlinear regression and they are not suited for Monte Carlo experiments."

Almiron, M. G., Lopes, B., Oliveira, A. L. C., Medeiros, A. C., & Frery, A. C. (2010). On the numerical accuracy of spreadsheets. *Journal of Statistical Software*, 34(4), 1-29.

Tools

Reproducibility

The screenshot shows the Financial Times homepage with a prominent article by Chris Cook. The article discusses the critique of Carmen Reinhart and Ken Rogoff's research on public debt overhangs. The page includes a navigation bar with links like HOME, WORLD, US, COMPANIES, TECH, MARKETS, GRAPHICS, OPINION, WORK & CAREERS, LIFE & ARTS, and HOW TO SPEND IT. There are also links for Portfolio and Account Settings. The article is titled "Reinhart-Rogoff recrunch the numbers" and is dated April 17, 2013. It has 51 comments and a share button. The right sidebar features a "myFT" section with 15 notifications.

FINANCIAL TIMES

HOME WORLD US COMPANIES TECH MARKETS GRAPHICS OPINION WORK & CAREERS LIFE & ARTS HOW TO SPEND IT

Latest on World

Anbang sells \$2.4bn bank stake in latest disposal

Charts of the Year: have Italy's old demons returned?

Theresa May vows to fight no confidence vote - live

Nobel economics lessons on climate change Premium

Opinion FT Data + Add to myFT

Reinhart-Rogoff recrunch the numbers

CHRIS COOK + Add to myFT

Chris Cook APRIL 17, 2013

51

Carmen Reinhart and Ken Rogoff have had a bad day. The two [economic historians' research](#), which implied that public debt overhangs can hamper economic growth, was perhaps one of the most cited pieces of work in recent years. Their advice that high debt-GDP ratios – particularly above 90 per cent – are harmful to growth, has become a widely used point in discussion. And it's under attack by a trio at the University of Massachusetts, Amherst – Thomas Herndon, Michael Ash, and Robert Pollin.

As FT Alphaville [has noted](#), the issue is about one of Reinhart and Rogoff's most heavily cited papers on the importance of debt. This paper has been accused of being the victim of fat-fingered Excel coding, as well as selective use of data and odd weighting of how different episodes are weighted, which seemed – to the authors – to make little sense. [UPDATE, 14:46 GMT - Pollin and Ash have written a piece for the FT on what they think this all means for austerity [here](#).]

<https://www.ft.com/content/01fc06b8-fb6e-3e36-acb0-a1f8b47a7271>

Tools

Reproducibility



Does High Public Debt Consistently
Stifle Economic Growth?
A Critique of Reinhart and Rogoff

Thomas Herndon, Michael Ash and Robert Pollin

April 2013

WORKINGPAPER SERIES

Number 322



"We replicate Reinhart and Rogoff (2010a and 2010b) and find that **coding errors, selective exclusion of available data, and unconventional weighting of summary statistics lead to serious errors** that inaccurately represent the relationship between public debt and GDP growth among 20 advanced economies in the post-war period."

Tools

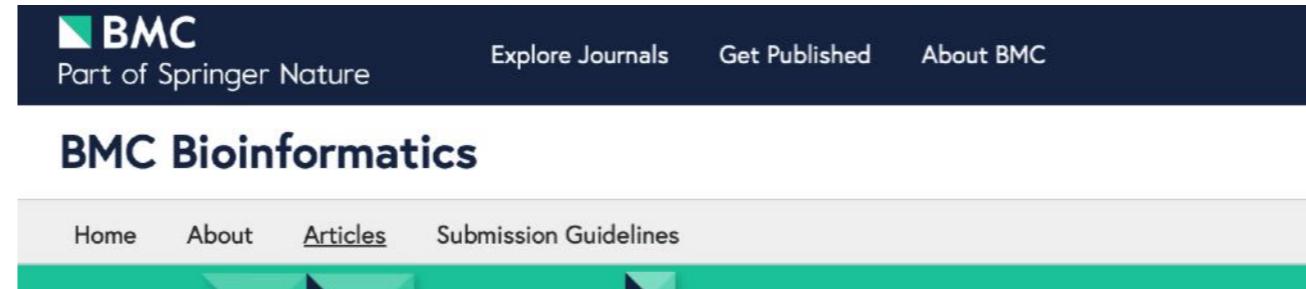
Reproducibility

A screenshot of a Microsoft Excel spreadsheet. The formula bar at the top shows the formula `=SUM(D3:D4)`. The spreadsheet has columns A through E and rows 1 through 12. Cell D7 contains the value 7. Cell D12 contains the value 2. The formula `=SUM(D3:D4)` is intended to sum the values in cells D3 and D4, but it only includes cell D3, resulting in an error. A tooltip box is displayed over cell D12, indicating the error: "Formula Omits Adjacent Cells". The tooltip also includes options: "Update Formula to Include Cells", "Help on this Error", "Ignore Error", "Edit in Formula Bar", and "Error Checking Options...".

	A	B	C	D	E
1					
2					
3					1
4					1
5					1
6					
7			Sum	2	
8					
9					
10					
11					
12					

Tools

Data formats



The screenshot shows the BMC Bioinformatics journal website. At the top, there's a dark blue header with the BMC logo (a green square with a white 'B') and the text "Part of Springer Nature". To the right are links for "Explore Journals", "Get Published", and "About BMC". Below the header, the title "BMC Bioinformatics" is displayed in a large, bold, dark blue font. Underneath the title is a navigation bar with links for "Home", "About", "Articles" (which is underlined in red), and "Submission Guidelines". A green horizontal bar runs across the page below the navigation bar.

Correspondence | [Open Access](#)

Mistaken Identifiers: Gene name errors can be introduced inadvertently when using Excel in bioinformatics

Barry R Zeeberg [†], Joseph Riss [†], David W Kane, Kimberly J Bussey, Edward Uchio, W Marston Linehan, J Carl Barrett and John N Weinstein [✉](#)

[†]Contributed equally

BMC Bioinformatics 2004 **5**:80

<https://doi.org/10.1186/1471-2105-5-80> | © Zeeberg et al; licensee BioMed Central Ltd. 2004

Received: 05 March 2004 | Accepted: 23 June 2004 | Published: 23 June 2004

Abstract

Background

When processing microarray data sets, we recently noticed that some gene names were being changed inadvertently to non-gene names.

Results

A little detective work traced the problem to default date format conversions and floating-point format conversions in the very useful Excel program package. The date conversions affect at least 30 gene names; the floating-point conversions affect at least 2,000 if Riken identifiers are included. These conversions are irreversible; the original gene names cannot be recovered.

Tools

Data formats

The screenshot shows a scientific article page from the BMC website. At the top, the BMC logo and "Part of Springer Nature" are visible, along with navigation links for "Explore Journals", "Get Published", and "About BMC". A search bar and login link are also present. The main header "Genome Biology" is highlighted with a yellow box. Below the header, there are links for "Home", "About", "Articles" (underlined), and "Submission Guidelines".

On the left, there are "Comment" and "Open Access" buttons. On the right, there are "Download PDF" and "Export citations" buttons. The article title is "Gene name errors are widespread in the scientific literature". It is authored by Mark Ziemann, Yotam Eren and Assam El-Osta. The publication details are "Genome Biology 2016 17:177" and the DOI is <https://doi.org/10.1186/s13059-016-1044-7>. The article was published on 23 August 2016.

The abstract section is titled "Abstract" and contains the following text: "The spreadsheet software Microsoft Excel, when used with default settings, is known to convert gene names to dates and floating-point numbers. A programmatic scan of leading genomics journals reveals that approximately one-fifth of papers with supplementary Excel gene lists contain erroneous gene name conversions."

On the right side, there are sections for "In These Collections", "Commentaries", "Metrics", "Article accesses: 91719", "Citations: 29", "more information", "Altmetric Attention Score: 1744", and a color bar. There is also a "Share This Article" button at the bottom.

<https://genomebiology.biomedcentral.com/articles/10.1186/s13059-016-1044-7>

Tools

Bad practices



\approx **\$6b USD**

“The London Whale”

2012

Tools

Bad practices

(...) Fourth, the model was approved despite observed operational problems. **The Model Review Group noted that the VaR computation was being done on spreadsheets using a manual process and it was therefore “error prone” and “not easily scalable.”** (...)

From: Report of JPMorgan Chase & Co. Management Task Force Regarding 2012 CIO Losses

Tools

Bad practices

(...) Sixth, CIO's implementation of the model was flawed. CIO relied on the model creator, who reported to the front office, to operate the model. **Data were uploaded manually without sufficient quality control. Spreadsheet-based calculations were conducted with insufficient controls and frequent formula and code changes were made.** Inadequate information technology resources were devoted to the process. Contrary to the action plan contained in the model approval, the process was never automated. (...)

From: Report of JPMorgan Chase & Co. Management Task Force Regarding 2012 CIO Losses

Tools

Bad practices

(...) Specifically, after subtracting the old rate from the new rate, the **spreadsheet divided by their sum instead of their average, as the modeler had intended**. This error likely had the effect of muting volatility by a factor of two and of lowering the VaR, although it is unclear by exactly what amount, particularly given that it is unclear whether this error was present in the VaR calculation for every instrument, and that it would have been offset to some extent by correlation changes. **It also remains unclear when this error was introduced in the calculation.**

From: Report of JPMorgan Chase & Co. Management Task Force Regarding 2012 CIO Losses

Tools

Bad practices

 Harvard Business Review   [Subscribe](#) | [Sign In](#) | [Register](#)

RISK MANAGEMENT

The JP Morgan “Whale” Report and the Ghosts of the Financial Crisis

by [Ben W. Heineman, Jr.](#)

JANUARY 24, 2013

 [SAVE](#)  [SHARE](#)  [COMMENT](#) (0)  [TEXT SIZE](#)  [PRINT](#)

The apparition of 2008 returns once more. Two recently released JP Morgan Chase (JPM) reports on the causes of the “London Whale” trading losses raise important questions about whether financial service firms can exorcise the spectral issues which were so central to the financial crisis. They read as if JPM and a key headquarters unit – the Chief Investment Office – had not learned a single lesson from the meltdown four years ago. And

WHAT TO READ NEXT



[The Hidden Risks in Emerging Markets](#)

RECOMMENDED



JPMorgan and the London Whale

FINANCE & ACCOUNTING CASE

\$8.95 [ADD TO CART](#)

<https://hbr.org/2013/01/the-jp-morgan-whale-report-and>

Tools

Notebooks

Literate programming

Interactivity

Different outputs from the same source document

Source control

Automatic report creation

Tools

Notebooks

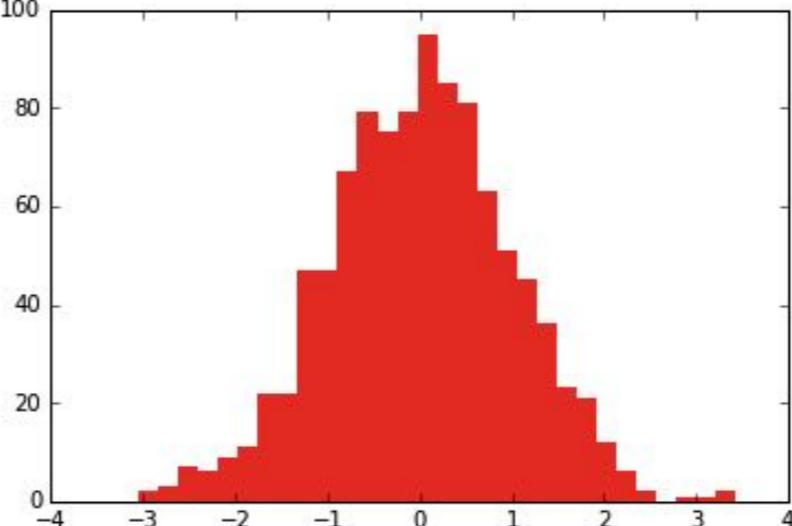


File Edit View Insert Cell Kernel Help

In [1]: `%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np`

In [11]: `# plot a normal distribution

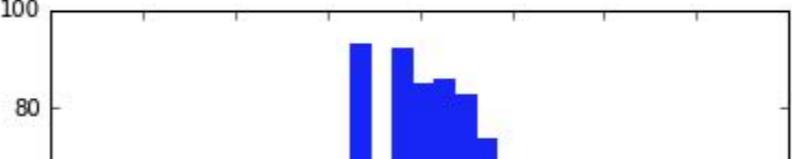
fig, ax = plt.subplots(1, 1)
handle = ax.hist(np.random.normal(0, 1, 1000), bins=30, lw=0, color='r')[0]`



A histogram showing a normal distribution centered at 0. The x-axis ranges from -4 to 4 with major ticks every 1 unit. The y-axis ranges from 0 to 100 with major ticks every 20 units. The distribution is symmetric and bell-shaped, colored red.

In [13]: `# plot a distribution with a different mean

fig, ax = plt.subplots(1, 1)
handle = ax.hist(np.random.normal(5, 1, 1000), bins=30, lw=0, color='b')[0]`



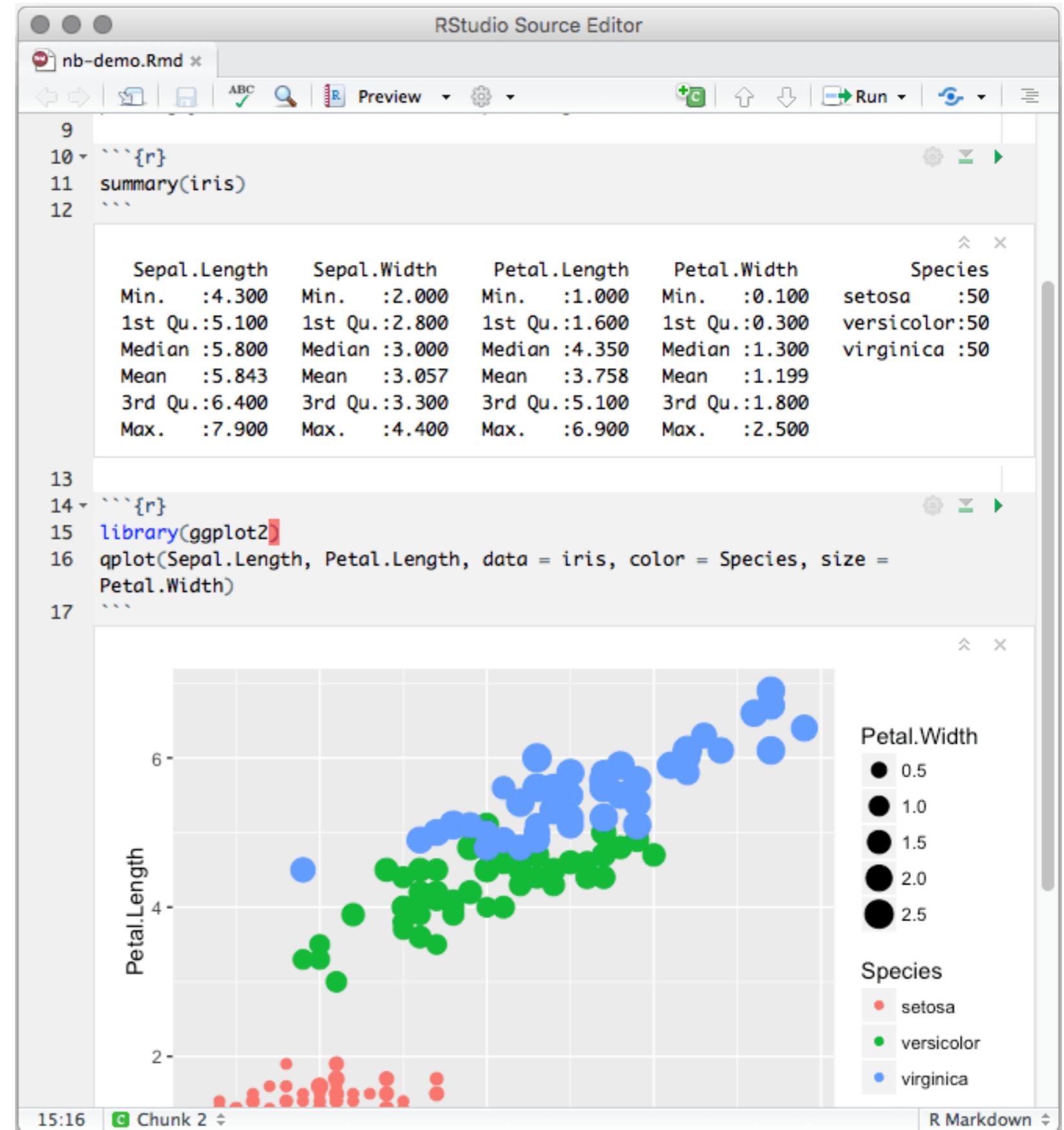
A histogram showing a distribution centered at 5. The x-axis ranges from -4 to 4 with major ticks every 1 unit. The y-axis ranges from 80 to 100 with major ticks every 20 units. The distribution is skewed to the right, colored blue.

<https://jupyter.org>

Tools

Notebooks

R Notebooks



Tools

Revision control systems



<https://git-scm.com>

Tools

Revision control systems



GitHub



Bitbucket



GitLab

Tools



Tools

Only change files programmatically

Use checksum hashes to guarantee
the original files remains unaltered

All the examples are in R

Data collection

Data collection

Data sources

Data collection

Data sources

 DATA.GOV

DATA TOPICS ▾ IMPACT APPLICATIONS DEVELOPERS CONTACT

The home of the U.S. Government's open data

Here you will find data, tools, and resources to conduct research, develop web and mobile applications, design data visualizations, and [more](#).

GET STARTED
SEARCH OVER [301,196 DATASETS](#)

Health Care Provider Charge Data 

BROWSE TOPICS



Agriculture



Climate



Consumer



Ecosystems



Education



Energy



Finance



Health



Local
Government



Manufacturing



Maritime



Ocean



Public Safety



Science &
Research

<https://www.data.gov>

Data collection

Data sources

data.gov.uk | Find open data

Publish your data Support

We've been improving data.gov.uk to help you find and use open government data.

[Discover what's changed](#) and [get in touch](#) to give us your feedback.

[Don't show this message again](#)

Find open data

Find data published by central government, local authorities and public bodies to help you build products and services

Q

Business and economy

Small businesses, industry, imports, exports and trade

Environment

Weather, flooding, rivers, air quality, geology and agriculture

Mapping

Addresses, boundaries, land ownership, aerial photographs, seabed and land terrain

Crime and justice

Courts, police, prison, offenders, borders and immigration

Government

Staff numbers and pay, local councillors and department business plans

Society

Employment, benefits, household finances, poverty and population

Defence

Armed forces, health and safety,

Government spending

Includes all payments by

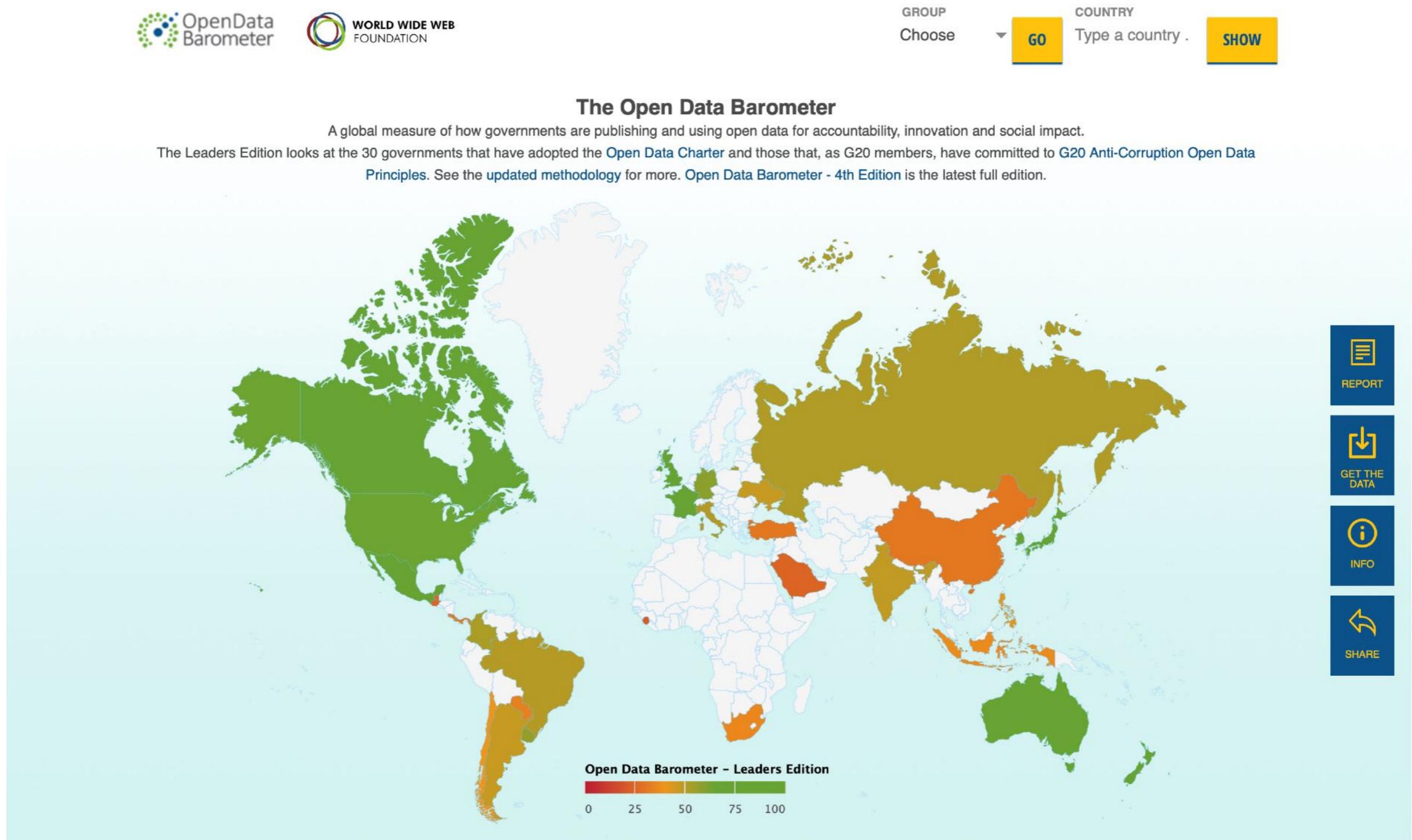
Towns and cities

Includes housing, urban planning,

<https://data.gov.uk>

Data collection

Data sources



<https://opendatabarometer.org/>

Data collection

Data sources

The screenshot shows the homepage of the Global Open Data Index. At the top left is the logo 'GLOBAL OPEN DATA INDEX' with a sunburst icon. At the top right is the 'OPEN KNOWLEDGE INTERNATIONAL' logo with a sunburst icon. The top navigation bar includes links for Places, Datasets, Download, Insights, Methodology, About, and Help. The main title 'TRACKING THE STATE OF OPEN GOVERNMENT DATA' is prominently displayed in large, bold, dark letters. Below it, a subtitle reads: 'The Global Open Data Index provides the most comprehensive snapshot available of the state of open government data publication'. At the bottom, there are three call-to-action boxes: 'Compare countries' (with a globe icon), 'Discuss findings' (with a speech bubble icon), and 'Get the insights' (with a lightbulb icon). The URL <https://index.okfn.org> is at the very bottom.

Places Datasets Download Insights Methodology About Help ▾

TRACKING THE STATE OF OPEN GOVERNMENT DATA

The Global Open Data Index provides the most comprehensive snapshot available of the state of open government data publication

Compare countries
Ranked table and map views of participating countries.

Discuss findings
Discuss your findings in our forum.

Get the insights
Insights from local and thematic perspectives.

<https://index.okfn.org>

Data collection

Data sources

Registry of Open Data on AWS



About

This registry exists to help people discover and share datasets that are available via AWS resources. [Learn more about sharing data on AWS](#).

[See all usage examples for datasets listed in this registry.](#)

Search datasets (currently 88 matching datasets)

[Search datasets](#)

Add to this registry

If you want to add a dataset or example of how to use a dataset to this registry, please follow the instructions on the [Registry of Open Data on AWS GitHub repository](#).

Unless specifically stated in the applicable dataset documentation, datasets available through the Registry of Open Data on AWS are not provided and maintained by AWS. Datasets are provided and maintained by a variety of third parties under a variety of licenses. Please check dataset licenses and related documentation to determine if a dataset may be used for your application.

Sentinel-2

[earth observation](#) [satellite imagery](#) [gis](#) [natural resource](#) [sustainability](#) [disaster response](#)

The [Sentinel-2 mission](#) is a land monitoring constellation of two satellites that provide high resolution optical imagery and provide continuity for the current SPOT and Landsat missions. The mission provides a global coverage of the Earth's land surface every 5 days, making the data of great use in on-going studies. L1C data are available from June 2015 globally. L2A data are available from April 2017 over wider Europe region, planned to be expanded globally in July 2018.

[Details →](#)

Usage examples

- [EOS Land Viewer by Earth Observing System](#)
- [Sentinel Playground by Sinergise](#)
- [EO Browser by Sinergise](#)
- [FME Landsat-8/Sentinel-2 File Selector by Safe Software](#)
- [Using Vector tiles and AWS Lambda, we can build a really simple API to get Landsat and Sentinel images by Remote Pixel](#)

[See 16 usage examples →](#)

Landsat 8

[earth observation](#) [satellite imagery](#) [gis](#) [natural resource](#) [sustainability](#) [disaster response](#)

An ongoing collection of satellite imagery of all land on Earth produced by the Landsat 8 satellite.

[Details →](#)

Usage examples

- [Exploring the Chile wildfires with Landsat and Sentinel-2 imagery by Timothy Whitehead](#)
- [Development Seed Geolambda by Matthew Hanson](#)
- [Integrate imagery from the full Landsat archive into your own apps, maps, and analysis with Landsat image services by Esri](#)
- [FME Landsat-8/Sentinel-2 File Selector by Safe Software](#)
- [Sentinel Playground for Landsat by Sinergise](#)

[See 13 usage examples →](#)

<https://registry.opendata.aws>

Data collection

Data sources

Google Cloud Platform Select a project ▾

Search for solutions

Marketplace

Datasets

Filter by

106 results

TYPE

Datasets 

CATEGORY

Advertising (7)

Analytics (6)

Big data (4)

Climate (19)

Databases (1)

Developer tools (1)

Economics (22)

Encyclopedic (29)

Finance (3)

Genomics (3)

Health (8)

Machine learning (1)

Maps (1)

Public safety (13)

Science & research (45)

Social (3)

Transportation (1)

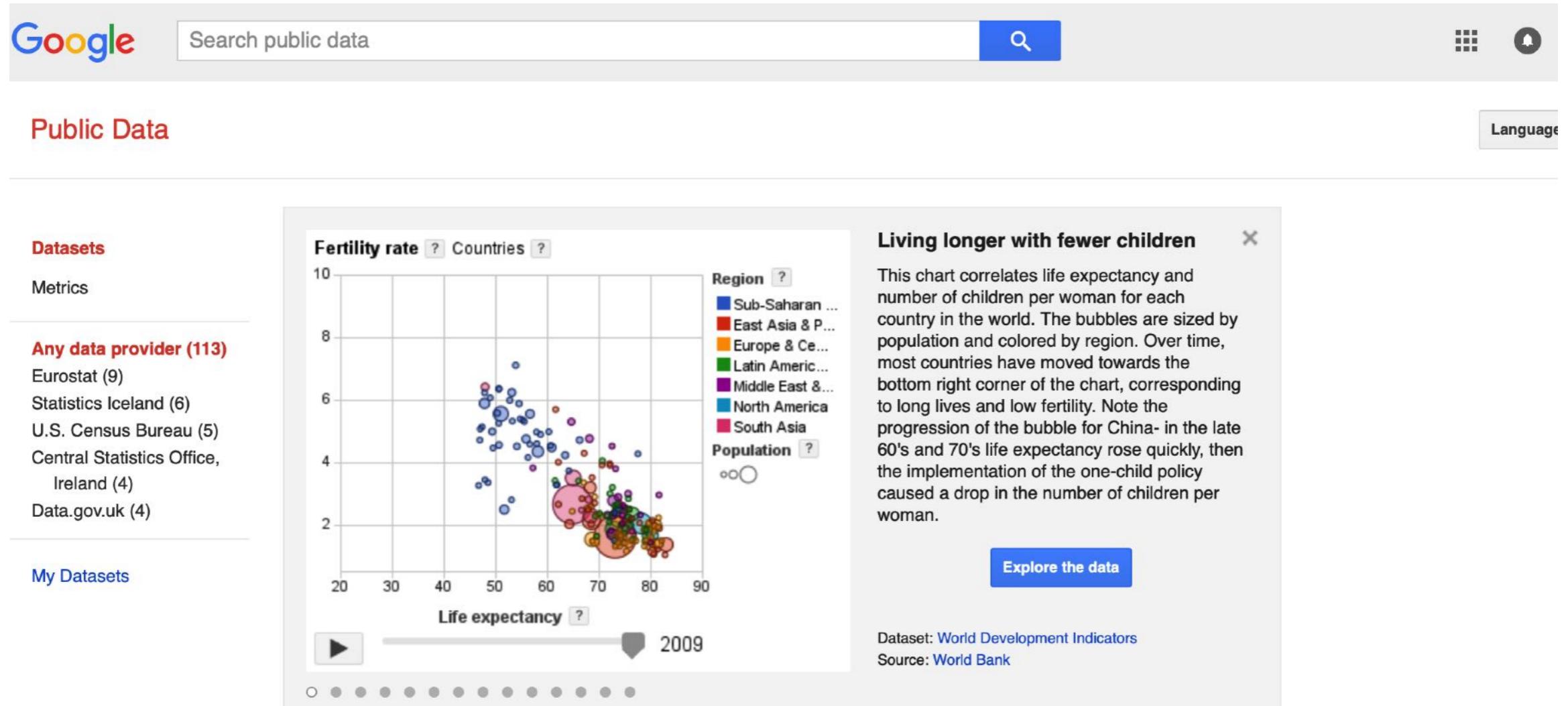
Other (11)

 1000 Cannabis Genome Project BigQuery Public Data Genomic samples of various cannabis strains	 Argentina Real Estate Listings Properati Monthly property listing data for Argentina since 2016	 Austin Crime Data City of Austin City of Austin crime data for 2014 and 2015	 Bitcoin Blockchain BigQuery Public Data Bitcoin blockchain transactions and blocks
 Brazil Real Estate Listings Properati Monthly property listing data for Brazil since 2016	 Bureau of Labor Statistics U.S. Bureau of Labor Statistics U.S. economic statistics for inflation, prices and unemployment	 ChEMBL Data Google Patents Public Datasets	 Chicago Crime Data City of Chicago Chicago Police Department crime data from 2001 to present
 Chicago Taxi Trips City of Chicago Chicago taxi trips from 2013 to present	 Chile Real Estate Listings Properati Monthly property listing data for Chile since 2016	 Columbia Real Estate Listings Properati Monthly property listing data for Columbia since 2016	 Cooperative Patent Classification Data Google Patents Public Datasets

<https://console.cloud.google.com/marketplace/browse?filter=solution-type:dataset>

Data collection

Data sources



Looking for other datasets? Find more with [Google Dataset Search](#).

[World Development Indicators](#)

[World Bank](#)

This dataset contains the World Development Indicators (WDI).

<https://www.google.com/publicdata/directory>

Data collection

Data sources

Awesome Public Datasets



NOTICE: This repo is automatically generated by [apd-core](#). Please **DO NOT** modify this file directly. We have provided [a new way](#) to contribute to Awesome Public Datasets. The original PR entrance directly on repo is closed forever.

- I am well.
- Please fix me.

This list of a topic-centric public data sources in high quality. They are collected and tidied from blogs, answers, and user responses. Most of the data sets listed below are free, however, some are not. Other amazingly awesome lists can be found in [sindresorhus's awesome](#) list.

Table of Contents

- [Agriculture](#)
- [Biology](#)
- [Climate+Weather](#)
- [ComplexNetworks](#)
- [ComputerNetworks](#)
- [DataChallenges](#)
- [EarthScience](#)
- [Economics](#)
- [Education](#)
- [Energy](#)

<https://github.com/awesomedata/awesome-public-datasets>

Data collection

Data sources



REUTILIZAÇÕES EM DESTAQUE

A screenshot of the "PartilhaJustica.gov.pt" website. The header includes the logo and the tagline "Mais participação, mais transparência, mais Justiça.". The menu bar has options for "PARTILHA", "TRANSPARÉNCIA", "PARTICIPAÇÃO", and "DESTAQUES". The "DESTAQUES" section highlights three items: "Mais de 33 mil certificados de registo criminal foram emitidos online em 2017", "Registo de marcas aumenta 16 por cento", and "Ações Executivas civis registam".

ÚLTIMAS REUTILIZAÇÕES

A list of recent data reuse projects from the portal:

- Lisboa Aberta**
Agência para a... 11 de maio de 2018 | Aplicação
- Partilha.justica.gov.pt**
Agência para a... 3 de maio de 2018 | Visualização
- Mapa do Cidadão**

<https://dados.gov.pt/pt/>

Data collection

Consuming text files

Consuming text files

Text is one of the most reliable
and common format to share files

Consuming text files

They can be:

- **csv** - comma separated file
- **tsk** - tab separated file
- **fwf** - fixed width file
- or use another delimiter

Consuming text files

Potential problems

Different file encoding your program expects

Quoted/unquoted strings

Comments in the records

Different number of columns per row

Different line break characters

Invisible characters

Consuming text files

Potential problems

```
read_delim(file, delim, quote = "\\"", escape_backslash = FALSE,  
escape_double = TRUE, col_names = TRUE, col_types = NULL,  
locale = default_locale(), na = c("", "NA"), quoted_na = TRUE,  
comment = "", trim_ws = FALSE, skip = 0, n_max = Inf,  
guess_max = min(1000, n_max), progress = show_progress(),  
skip_empty_rows = TRUE)
```

Consuming text files

Potential problems

Different line break characters

dos2unix

```
tr -d "\015" < file_name > new_file_name
```

Consuming text files

Tools

Command line tool to work with csv files

csvkit

<https://csvkit.readthedocs.io/en/0.9.0/index.html#>

Consuming text files

Tools

tail

head

wc

tr

sed

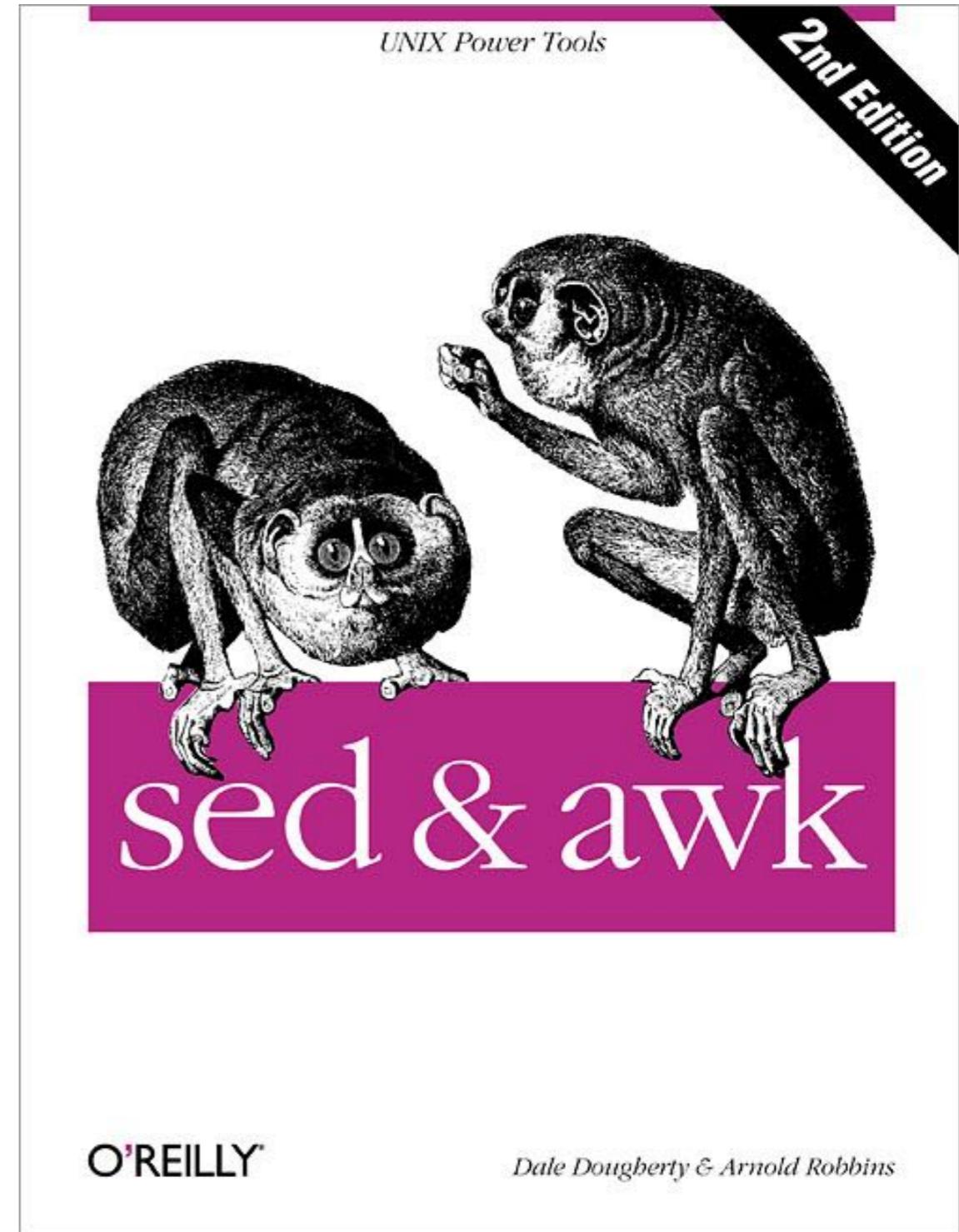
cut

grep

awk

Consuming text files

Tools



<https://www.oreilly.com/library/view/sed-awk/1565922255/>

O'REILLY

Dale Dougherty & Arnold Robbins

Consuming text files

Tools

O'REILLY®



Data
Science
at the
Command Line

FACING THE FUTURE WITH TIME-TESTED TOOLS

<https://www.datascienceatthecommandline.com>

Jeroen Janssens

Consuming text files

Tools

The
Pragmatic
Programmers

Small, Sharp, Software Tools

Harness the Combinatoric
Power of Command-Line Tools
and Utilities



<https://pragprog.com/book/bhcldev/small-sharp-software-tools>

Brian P. Hogan
edited by Tammy Coron

Consuming text files

Example

The screenshot shows a web browser window displaying the CMS.gov website. The URL in the address bar is cms.gov. The page title is "Medicare Provider Utilization and Payment Data: Part D Prescriber". The main content area describes the Part D Prescriber Public Use File (PUF) and provides links to various datasets and tools. On the left, there is a sidebar with a list of related datasets. At the bottom, there is a "Downloads" section with links to PDF files. The CMS logo is at the top left, and the navigation menu includes links for Medicare, Medicaid/CHIP, Medicare-Medicaid Coordination, Private Insurance, Innovation Center, Regulations & Guidance, Research, Statistics, Data & Systems, and Outreach & Education.

Medicare Provider Utilization and Payment Data

- [Medicare Provider Utilization and Payment Data: Physician and Other Supplier](#)
- [Medicare Provider Utilization and Payment Data: Inpatient](#)
- [Medicare Provider Utilization and Payment Data: Outpatient](#)
- Medicare Provider Utilization and Payment Data: Part D Prescriber**
- [Medicare Provider Utilization and Payment Data: Referring Durable Medical Equipment, Prosthetics, Orthotics and Supplies](#)
- [Medicare Provider Utilization and Payment Data: Home Health Agencies](#)
- [Medicare Provider Utilization and Payment Data: Skilled Nursing Facilities](#)
- [Medicare Provider Utilization and Payment Data: Hospice Providers](#)
- [Public Comment on the Release of Medicare Physician Data](#)

Medicare Provider Utilization and Payment Data: Part D Prescriber

The Part D Prescriber Public Use File (PUF) provides information on prescription drugs prescribed by individual physicians and other health care providers and paid for under the Medicare Part D Prescription Drug Program. The Part D Prescriber PUF is based on information from CMS's Chronic Conditions Data Warehouse, which contains Prescription Drug Event records submitted by Medicare Advantage Prescription Drug (MAPD) plans and by stand-alone Prescription Drug Plans (PDP). The dataset identifies providers by their National Provider Identifier (NPI) and the specific prescriptions that were dispensed at their direction, listed by brand name (if applicable) and generic name. For each prescriber and drug, the dataset includes the total number of prescriptions that were dispensed, which include original prescriptions and any refills, and the total drug cost. The total drug cost includes the ingredient cost of the medication, dispensing fees, sales tax, and any applicable administration fees and is based on the amount paid by the Part D plan, Medicare beneficiary, government subsidies, and any other third-party payers.

Although the Part D Prescriber PUF has a wealth of information on payment and utilization for Medicare Part D prescriptions, the dataset has a number of limitations. Of particular importance is the fact that the data may not be representative of a physician's entire practice or all of Medicare as it only includes information on beneficiaries enrolled in the Medicare Part D prescription drug program (i.e., approximately two-thirds of all Medicare beneficiaries). In addition, the data are not intended to indicate the quality of care provided. For additional limitations, please review the methodology document available below.

[Medicare Part D Prescriber Data CY 2016](#)
[Medicare Part D Prescriber Data CY 2015](#)
[Medicare Part D Prescriber Data CY 2014](#)
[Medicare Part D Prescriber Data CY 2013](#)

[Medicare Part D Opioid Prescribing Mapping Tool](#)

Inquiries regarding this data can be sent to MedicareProviderData@cms.hhs.gov.

To receive email notifications, please sign up for the Medicare Provider Data GovDelivery subscription [here](#).

Downloads

[Part D Prescriber PUF Methodology \[PDF, 428KB\]](#)
[Part D Prescriber PUF Frequently Asked Questions \[PDF, 103KB\]](#)

Page last Modified: 05/15/2018 1:53 PM
[Help with File Formats and Plug-Ins](#)

<https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/Medicare-Provider-Charge-Data/Part-D-Prescriber.html>

Consuming text files

Example

Part D Prescriber Detailed Data for 2013

Original compressed file: **532 MB**

Uncompressed file: **2.8 GB**

Consuming text files

Example

The results depend on:

- the hardware
- operating system
- system load
- and also the data itself



Consuming text files

Example

```
wc -l PartD_Prescriber_PUF_NPI_Drug_13.txt
```

23645874 PartD_Prescriber_PUF_NPI_Drug_13.txt

23 645 873
records

2.8 GB

Consuming text files

Example

```
head PartD_Prescriber_PUF_NPI_Drug_13.txt
```

```
+ ~/Documents/work/uminho/aulas/2018-2019/workshop-bp/data/PartD_Prescriber_PUF_NPI_DRUG_13 head
d PartD_Prescriber_PUF_NPI_Drug_13.txt
npi      nppes_provider_last_org_name    nppes_provider_first_name      nppes_provider_city      nppe
s_provider_state      specialty_description      description_flag      drug_name      generic_name
bene_count      total_claim_count      total_30_day_fill_count      total_day_supply      total_drug_c
ost      bene_count_ge65      bene_count_ge65_suppress_flag      total_claim_count_ge65      ge65_suppress_flag t
otal_30_day_fill_count_ge65      total_day_supply_ge65      total_drug_cost_ge65
1003000126      ENKESHAFI      ARDALAN CUMBERLAND      MD      Internal Medicine      S      ISOS
ORBIDE MONONITRATE ER      ISOSORBIDE MONONITRATE      11      11      307      171.59      *      *
1003000126      ENKESHAFI      ARDALAN CUMBERLAND      MD      Internal Medicine      S      LEVO
FLOXACIN      LEVOFLOXACIN      26      26      26      165      227.1      15      15      1
5      106      159.72
1003000126      ENKESHAFI      ARDALAN CUMBERLAND      MD      Internal Medicine      S      LISI
NOPRIL      LISINOPRIL      17      19      19      570      100.37      #      #
1003000126      ENKESHAFI      ARDALAN CUMBERLAND      MD      Internal Medicine      S      METO
PROLOL TARTRATE METOPROLOL TARTRATE      28      30      31      916      154.65      #      #
1003000126      ENKESHAFI      ARDALAN CUMBERLAND      MD      Internal Medicine      S      PRED
NISONE      PREDNISONE      14      14      14      133      44.72      *      *
1003000126      ENKESHAFI      ARDALAN CUMBERLAND      MD      Internal Medicine      S      SIMV
ASTATIN      SIMVASTATIN      16      17      17      487      102.06      #      #
1003000126      ENKESHAFI      ARDALAN CUMBERLAND      MD      Internal Medicine      S      WARF
ARIN SODIUM      WARFARIN SODIUM      11      11      13.6      324      165.45      *      *
1003000142      KHALIL      RASHID      TOLEDO      OH      Anesthesiology      S      ACETAMINOPHEN-CODEINE      ACET
AMINOPHEN WITH CODEINE      11      22      22      563      242.98      #      #
1003000142      KHALIL      RASHID      TOLEDO      OH      Anesthesiology      S      BACLOFEN      BACLOFEN      1
5      15      450      139.81      *      *
+ ~/Documents/work/uminho/aulas/2018-2019/workshop-bp/data/PartD_Prescriber_PUF_NPI_DRUG_13 |
```

Consuming text files

Example

tail PartD_Prescriber_PUF_NPI_Drug_13.txt

PartD_Prescriber_PUF_NPI_Drug_13.txt										
#										
1992999825	DESCHENES	GEOFFREY	SEATTLE	WA	Otolaryngology	S	AMOX	TR-POTA		
SSIUM CLAVULANATE		AMOXICILLIN/POTASSIUM CLAV		14	16	16	195	378.51	#	
#										
1992999825	DESCHENES	GEOFFREY	SEATTLE	WA	Otolaryngology	S	FLUTICASONE			
PROPIONATE	FLUTICASONE	PROPIONATE	38	74	84	2491	2388.37	#	62	7
2	2131	2023.8								
1992999825	DESCHENES	GEOFFREY	SEATTLE	WA	Otolaryngology	S	IPRATROPIUM			
BROMIDE	IPRATROPIUM	BROMIDE		27	30.5	755	619.17	*	27	3
0.5	755	619.17								
1992999825	DESCHENES	GEOFFREY	SEATTLE	WA	Otolaryngology	S	OMEPRAZOLE	0		
MEPRAZOLE			34	49	1442	433.48	*	19	33	9903
32.82										
1992999825	DESCHENES	GEOFFREY	SEATTLE	WA	Otolaryngology	S	OXYCODONE-AC			
ETAMINOPHEN	OXYCODONE	HCL/ACETAMINOPHEN		13	13	13	69	84.1	*	*
1992999833	SHAW	L. NOAH NEW YORK	NY		Psychoanalyst	T	SELEGILINE	HCL	SELE	
GILINE HCL			11	11	330	563.03	*	11	11	3305
63.03										
1992999833	SHAW	L. NOAH NEW YORK	NY		Psychoanalyst	T	STALEVO	100	CARB	
IDOPA/LEVODOPA/ENTACAPONE			11	11	330	6929.04	*	11	1	
1	330	6929.04								
1992999874	JOFFE	GABRIELLA	MECHANICSVILLE	VA	Internal Medicine		S	LEVO		
FLOXACIN	LEVOFLOXACIN	17	17	17	102	181.09	#	#		
1992999874	JOFFE	GABRIELLA	MECHANICSVILLE	VA	Internal Medicine		S	PANT		
OPRAZOLE SODIUM	PANTOPRAZOLE SODIUM			12	12	360	120.78	*	12	1
2	360	120.78								
1992999874	JOFFE	GABRIELLA	MECHANICSVILLE	VA	Internal Medicine		S	PRED		
NISONE	PREDNISONE	16	16	18	247	93.47	#	#		

Consuming text files

Example

Reading text files

Consuming text files

Example

Using the `readr` package:

```
prescriber_data_readr <- read_tsv(file = "../data/  
PartD_Prescriber_PUF_NPI_DRUG_13/  
PartD_Prescriber_PUF_NPI_DRUG_13.txt")
```

	user	system	elapsed
	60.197	6.876	82.337

Consuming text files

Example

Using the `data.table` package:

```
prescriber_data_fread <- fread(file = "../data/  
PartD_Prescriber_PUF_NPI_DRUG_13/  
PartD_Prescriber_PUF_NPI_DRUG_13.txt")
```

	user	system	elapsed
	41.217	9.128	25.446

Consuming text files

Example

Saving data in a binary format

Writing binary files

Example

Using the default R compressed file format:

```
saveRDS(prescriber_data_readr,  
        file=".~/data/prescriber_c.rds")
```

	user	system	elapsed	
	215.866	5.148	223.887	527MB

Writing binary files

Example

Using the default R uncompressed file format:

```
saveRDS(prescriber_data_readr,  
        file=".~/data/prescriber_u.rds",  
        compress = FALSE)
```

user	system	elapsed	
65.163	12.063	80.531	5.3GB

Writing binary files

fst package



“The [fst package](#) for R provides a fast, easy and flexible way to serialize data frames.

With access speeds of multiple GB/s, fst is specifically designed to unlock the potential of high speed solid state disks that can be found in most modern computers.

Data frames stored in the fst format have full random access, both in column and rows.”

<http://www.fstpackage.org>



Writing binary files

fst package

Using fst file format with the default compression:

```
write.fst(prescriber_data_readr,  
          path=".//data/prescriber_c.fst")
```

user	system	elapsed	
14.177	4.898	13.372	2.3 GB

Writing binary files

fst package

Using the uncompressed fst file format:

```
write.fst(prescriber_data_readr,  
          path = "../data/prescriber_u.fst",  
          compress = 0)
```

user	system	elapsed	
26.165	16.720	27.069	4.5 GB

Writing binary files

Feather package

“Feather provides binary columnar serialization for data frames.

*It is designed to make reading and writing data frames efficient,
and to make sharing data across data analysis languages easy.”*

No data compression!

<https://github.com/wesm/feather>



Writing binary files

Feather package

```
write_feather(prescriber_data_readr,  
              path=".~/data/prescriber.feather")
```

user	system	elapsed	
7.571	3.756	12.975	4.4 GB

Reading and writing binary files

Example

Reading data in a binary format

Reading binary files

Example

Using the compressed R file format:

```
aux <- readRDS(file =  
  ".../data/prescriber_c.rds")
```

user	system	elapsed	
79.030	1.936	82.347	527MB

Reading binary files

Example

Using the uncompressed R file format:

```
aux <- readRDS(file =  
  "../data/prescriber_u.rds")
```

user	system	elapsed	
79.859	5.630	90.014	5.3 GB

Reading binary files

Example

Using the compressed fst file format:

```
aux <- read_fst(path =  
                  "../data/prescriber_c.fst")
```

	user	system	elapsed	
	16.754	3.888	16.950	2.3 GB

Reading binary files

Example

Using the uncompressed fst file format:

```
aux <- read_fst(path =  
                  "../data/prescriber_u.fst")
```

user	system	elapsed	
11.033	4.555	17.682	4.5 GB

Reading binary files

Example

Using the feather file format:

```
aux <- read_feather(path =  
                      "../data/prescriber.feather")
```

	user	system	elapsed	
	14.996	9.072	47.238	4.4 GB

Reading and writing binary files

Example

Summary:

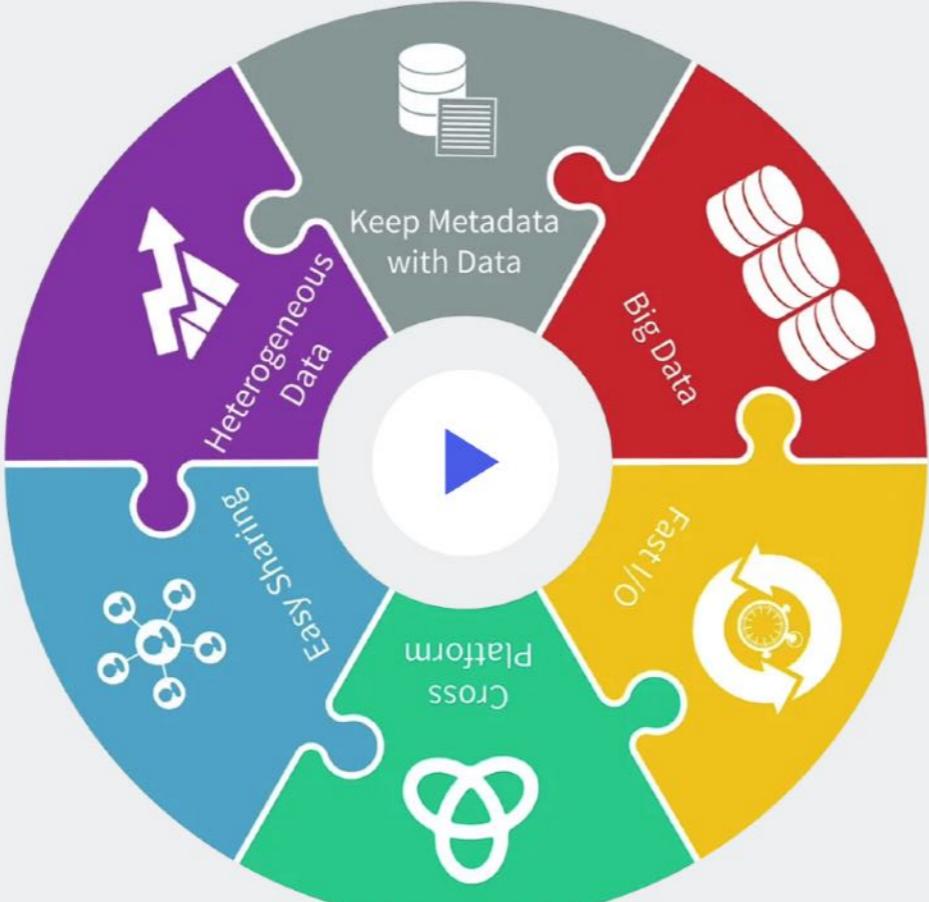
	Write	Read	File size
R format compressed	224s	82s	527 MB
R format uncompressed	81s	90s	5.3 GB
fst compressed	13s	17s	2.3 GB
fst uncompressed	27s	18s	4.5 GB
feather	13s	47s	4.4 GB

Consuming text files

Additional tools

About Us Solutions Community Downloads Documentation Support Portal [Twitter](#) [LinkedIn](#)

What is HDF5®?



- HETEROGENEOUS DATA**
HDF® supports n-dimensional datasets and each element in the dataset may itself be a complex object.
- EASY SHARING**
- CROSS PLATFORM**
- FAST I/O**
- BIG DATA**
- KEEP METADATA WITH DATA**

<https://www.hdfgroup.org/solutions/hdf5/>

Data collection

Web data sources

Data collection

Web data sources

Web scraping

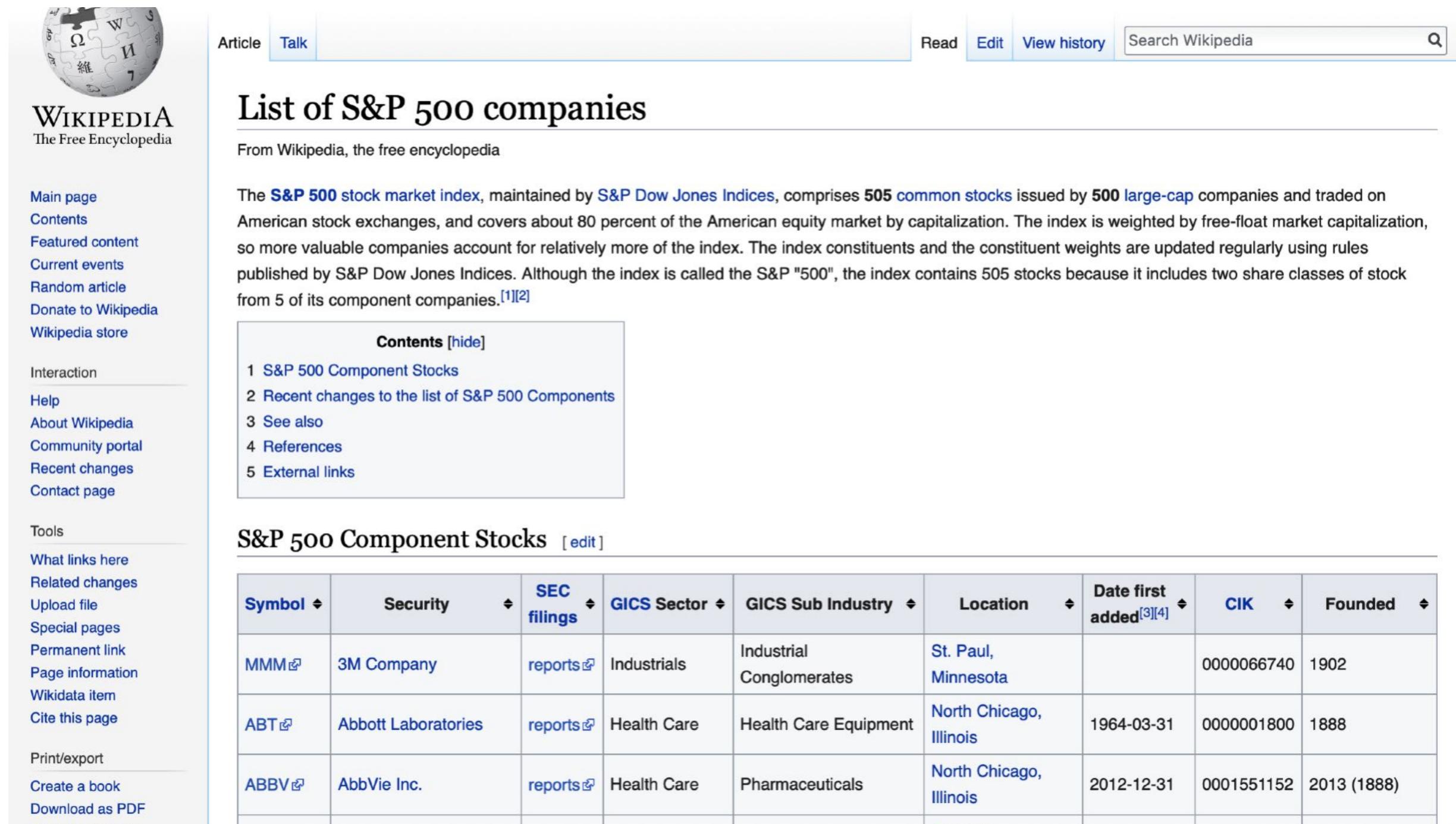
Web scraping

Friendly URLs

with data in tables

Web scraping

Friendly URLs



The screenshot shows a Wikipedia page titled "List of S&P 500 companies". The page has a standard header with tabs for "Article" (selected), "Talk", "Read", "Edit", "View history", and a search bar. The main content area starts with a summary of the S&P 500 index, mentioning it contains 505 stocks from 500 companies. Below this is a "Contents" sidebar with links to sections like "S&P 500 Component Stocks", "Recent changes", and "External links". The main body of the page is a table listing three companies: 3M Company, Abbott Laboratories, and AbbVie Inc., along with their SEC filings, GICS sectors, sub-industries, locations, dates first added, CIK numbers, and founding years.

WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export
Create a book
Download as PDF

Article Talk Read Edit View history Search Wikipedia

List of S&P 500 companies

From Wikipedia, the free encyclopedia

The **S&P 500 stock market index**, maintained by **S&P Dow Jones Indices**, comprises **505 common stocks** issued by **500 large-cap** companies and traded on American stock exchanges, and covers about 80 percent of the American equity market by capitalization. The index is weighted by free-float market capitalization, so more valuable companies account for relatively more of the index. The index constituents and the constituent weights are updated regularly using rules published by S&P Dow Jones Indices. Although the index is called the S&P "500", the index contains 505 stocks because it includes two share classes of stock from 5 of its component companies.^{[1][2]}

Contents [hide]

1 S&P 500 Component Stocks
2 Recent changes to the list of S&P 500 Components
3 See also
4 References
5 External links

S&P 500 Component Stocks [edit]

Symbol	Security	SEC filings	GICS Sector	GICS Sub Industry	Location	Date first added ^{[3][4]}	CIK	Founded
MMM	3M Company	reports	Industrials	Industrial Conglomerates	St. Paul, Minnesota		0000066740	1902
ABT	Abbott Laboratories	reports	Health Care	Health Care Equipment	North Chicago, Illinois	1964-03-31	0000001800	1888
ABBV	AbbVie Inc.	reports	Health Care	Pharmaceuticals	North Chicago, Illinois	2012-12-31	0001551152	2013 (1888)

https://en.wikipedia.org/wiki/List_of_S%26P_500_companies

Web scraping

Friendly URLs

```
library(rvest)
```

```
page <- read_html("https://en.wikipedia.org/wiki/  
List_of_S%26P_500_companies")
```

```
constituents <- page %>% html_nodes("table") %>%  
. [[1]] %>%  
html_table()
```

Web scraping

Friendly URLs

```
library(rvest)
```

```
page <- read_html("https://en.wikipedia.org/wiki/  
List_of_S%26P_500_companies")
```

```
constituents <- page %>% html_nodes("table") %>%  
. [[1]] %>%  
html_table()
```

Web scraping

Friendly URLs

The screenshot shows the homepage of the base: website. The header features the IMPIC logo (Instituto dos Mercados Públicos do Imobiliário e da Construção) and the word "base:" in large letters, with "CONTRATOS PÚBLICOS ONLINE" below it. To the right are links for "Área Reservada" and flags for Portugal and the European Union. A navigation menu at the top includes "NOTÍCIAS", "DOCUMENTAÇÃO", "ORIENTAÇÕES TÉCNICAS", "FAQ'S", "RELATÓRIOS", "LEGISLAÇÃO", "SANÇÕES ACESSÓRIAS", and "BENS MÓVEIS". Below the menu is a search bar with fields for "Pesquisa:", "Contratos", "Anúncios", "Entidades", "Incrementos 10% preço contratual", and "Despachos e Deliberações". There are buttons for "Pesquisar pelo objeto do contrato", "PESQUISAR", and "Pesquisa Avançada". The background of the page is a large image of a coastal town with buildings and a rocky shore. At the bottom, there are two sections: "Notícias" on the left featuring the IMPIC logo and a link to "IMPEDIMENTO DE FUNCIONAMENTO DA PLATAFORMA DE COMPRAS PÚBLICAS GATEWIT"; and "Relatórios, Circulares e Notas informativas...." on the right featuring a blue button with the text "Contratação Pública em Portugal".

<http://www.base.gov.pt/Base/pt/Homepage>

Web scraping

Friendly URLs

Resultados

Foram encontrados **958252** resultados para a sua pesquisa.

Pesquisou por:



Exportar Resultados (Excel)

Objeto do Contrato	Preço contratual	Publicação	Adjudicante	Adjudicatário	
Construção de passeios e muros e tratamento de águas pluviais...	23.970,77 €	17-12-2018	Freguesia de Meirinhas	Construções da Cancelinha, Lda	+
Prestação de Serviços Médicos no Serviço de Neuropediatria - C.P....	36.639,00 €	17-12-2018	Hospital Garcia de Orta, E. P. E.	JOÃO NUNO FERREIRA DE CARVALHO	+
Aquisição de Equipamento para infraestrutura Wireless em Escolas e Jardins...	74.901,15 €	17-12-2018	Município de Lisboa	InstalPlus - Sistemas de Comunicação Informática, Lda	+
Aquisição aparelhos Ar Condicionado	6.200,00 €	17-12-2018	Guarda Nacional Republicana	Rolfrio - Sociedade Comercial de Equipamentos Hoteleiros, Lda	+
Requalificação do Largo Antunes Lima	388.279,16 €	17-12-2018	Município de Vila Verde	Bruno Barbosa Araújo Unipessoal Lda.	+
Aquisição de serviços de assessoria especializada no domínio das TIC...	25.179,00 €	17-12-2018	Município de Vila Verde	Ambiglobal - Prestação de Serviços de Segurança, Higiene e Saúde no Trabalho Lda.	+
AQUISIÇÃO DE EQUIPAMENTO PARA O TEATRO MUNICIPAL DE MATOSINHOS CONSTANTINO...	7.480,50 €	17-12-2018	Município de Matosinhos	JOSE PEDRO FERREIRA DOS SANTOS	+
O objeto deste contrato consiste na elaboração de estudos para...	43.000,00 €	17-12-2018	Município de Vila Verde	Whiteenergy.com - Soluções Informáticas Lda.	+
Aquisição de serviços de trabalhos de limpeza de faixas de...	14.630,00 €	17-12-2018	Município de Torres Vedras	João Carlos Beirante Santos Martinho	+

Web scraping

Friendly URLs

http://www.base.gov.pt/Base/pt/ResultadosPesquisa?type=contratos&query=texto%26tipo%3D0%26tipocontrato%3D0%26cpv%3D%26numeroanuncio%3D%26aqinfo%3D%26adjudicante%3D%26adjudicataria%3D%26desdeprecocontrato_false%3D%26desdeprecocontrato%3D%26ateprecocontrato_false%3D%26ateprecocontrato%3D%26desdedatacontrato%3D%26atedatacontrato%3D%26desdedatapublicacao%3D%26atedatapublicacao%3D%26desdeprazoexecucao%3D%26ateprazoexecucao%3D%26desdedatafecho%3D%26atedatafecho%3D%26desdeprecoefectivo_false%3D%26desdeprecoefectivo%3D%26ateprecoefectivo_false%3D%26ateprecoefectivo%3D%26pais%3D0%26distrito%3D0%26concelho%3D0

Web scraping

Selenium

 **SeleniumHQ**
Browser Automation

[edit this page](#) search selenium: Projects Download Documentation Support About

What is Selenium?

Selenium automates browsers. That's it! What you do with that power is entirely up to you. Primarily, it is for automating web applications for testing purposes, but is certainly not limited to just that. Boring web-based administration tasks can (and should!) be automated as well.

Selenium has the support of some of the largest browser vendors who have taken (or are taking) steps to make Selenium a native part of their browser. It is also the core technology in countless other browser automation tools, APIs and frameworks.



Which part of Selenium is appropriate for me?

Selenium WebDriver 	Selenium IDE 
--	--

If you want to

- create robust, browser-based regression automation suites and tests
- scale and distribute scripts across many environments

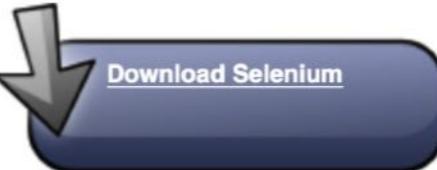
If you want to

- create quick bug reproduction scripts
- create scripts to aid in automation-aided

Selenium is a suite of tools to automate web browsers across many platforms.

Selenium...

- runs in [many browsers](#) and [operating systems](#)
- can be controlled by many [programming languages](#) and [testing frameworks](#).

 [Download Selenium](#)

<https://www.seleniumhq.org>

Web scraping

Selenium

The screenshot shows the official website of the Portuguese Social Security (Segurança Social). The top navigation bar includes links for 'A Segurança Social' (highlighted in green), 'Documentos e Formulários', 'Simulações', 'Após Sociais e Programas', 'Sou Empregador', and 'Sou Cidadão'. The main content area displays the 'Lista de devedores na Segurança Social' (List of debtors in the Social Security) page. A sidebar on the left provides links to various sections like 'Objetivos e princípios', 'História', and 'Estatísticas'. The central text explains the legal basis for the publication of debtor lists and the process for debt collection. At the bottom, there are links for further information and a footer with accessibility and cookie policy links.

SEGURANÇA SOCIAL

Sou Cidadão Sou Empregador Apoios Sociais e Programas Simulações Documentos e Formulários A Segurança Social

Pesquisa i

Objetivos e princípios
História
Organismos
As Organizações Internacionais e a Segurança Social
Prémios
Estatísticas
Orçamento e conta da Segurança Social
Lista de devedores na Segurança Social
Centros de documentação
Serviços de atendimento
Linha Segurança Social/Atendimento Automático
Atendimento por marcação
Bolsa de Médicos

Estou em: A Segurança Social > Lista de devedores na Segurança Social

Lista de devedores na Segurança Social

Em cumprimento do disposto no artigo 94º, nº 1, da Lei n.º 42/2016 de 28 de dezembro (Orçamento do Estado para 2017), no artigo 214º do Código dos Regimes Contributivos do Sistema Previdencial de Segurança Social e no artigo 64º, nº 5, alínea a), da Lei Geral Tributária, procede-se, no ano de 2017, pelo presente meio, à publicitação das listas dos devedores à segurança social com processos de execução fiscal ativos.

Esta lista integra devedores que, por ter terminado o prazo de pagamento voluntário sem terem cumprido as suas obrigações e, no prazo e termos legais, não terem requerido e enquadrado o pagamento da dívida em prestações, prestado garantia ou requerido a sua dispensa, não têm a sua situação contributiva regularizada.

A organização das referidas listas respeitou integralmente o teor da autorização nº 676/2006, de 19 de junho, da Comissão Nacional de Proteção de Dados.

Lista de Devedores

Selecionar

Para esclarecimentos adicionais contacte a Secção de Processo Executivo do distrito do seu domicílio ou sede ou o Serviço de Atendimento Telefónico através do n.º 300 036 036 (9h00-18h00-dias úteis)

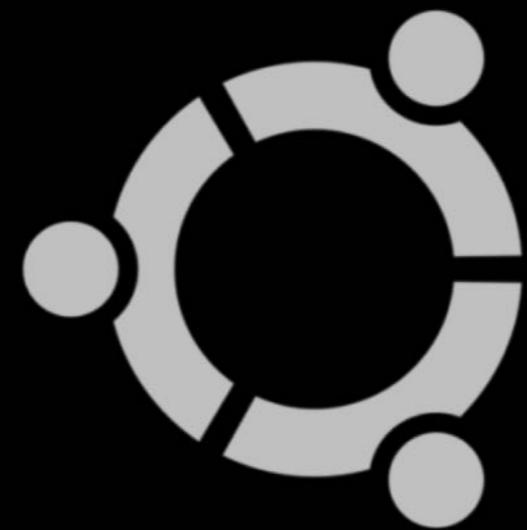
Para mais informações sobre as listas de devedores, critérios de inclusão e forma de exclusão, clique [aqui](#).

[D] W3C WCAG 1.0

Notícias Perguntas frequentes Links úteis Linhas de apoio Mapa do site Política de cookies

<http://www.seg-social.pt/lista-de-devedores-na-seguranca-social>

Web scraping



Selenium

Data collection

Web data sources

Using APIs

Using APIs

Application Programming Interface (API)

an API is the part of a server that handles requests and responses

Using APIs

Application Programming Interface (API)

easy to use

allow automatic update of data sets

Using APIs

An example

Load the libraries:

```
library(jsonlite)
library(httr)
library(purrr)
library(dplyr)
library(tidyr)
```

Using APIs

An example

Load the first page:

```
url <- paste0("api.worldbank.org/v2/indicators?  
format=json&page=", 1)  
res <- GET(url)  
parsed <- jsonlite::fromJSON(content(res, "text"))
```

Using APIs

An example

```
[  
  {  
    "page": 1,  
    "pages": 313,  
    "per_page": "50",  
    "total": 15650  
  },  
  [  
    {  
      "id": "2.0.cov.Math.pl_3.prv",  
      "name": "Coverage: Mathematics Proficiency Level 3, Private schools",  
      "unit": "",  
      "source": {  
        "id": "37",  
        "value": "LAC Equity Lab"  
      },  
      "sourceNote": "The coverage rate is the childhood access rate of a given opportunity used in calculating the Human Opportunities Index (HOI). The coverage rate does not take into account inequality of access between different circumstance groups.",  
      "sourceOrganization": "LAC Equity Lab tabulations using PISA Data.",  
      "topics": [  
        {  
          "id": "11",  
          "value": "Poverty "  
        }  
      ]  
    },  
    {  
      "id": "2.0.cov.Math.pl_3.prv",  
      "name": "Coverage: Mathematics Proficiency Level 3, Private schools",  
      "unit": "",  
      "source": {  
        "id": "37",  
        "value": "LAC Equity Lab"  
      },  
      "sourceNote": "The coverage rate is the childhood access rate of a given opportunity used in calculating the Human Opportunities Index (HOI). The coverage rate does not take into account inequality of access between different circumstance groups.",  
      "sourceOrganization": "LAC Equity Lab tabulations using PISA Data.",  
      "topics": [  
        {  
          "id": "11",  
          "value": "Poverty "  
        }  
      ]  
    }  
  ]
```

Using APIs

An example

A function to load the page, get the data in text format and convert the json into a data frame:

```
get_data <- function(page_n){  
  url <- paste0("api.worldbank.org/v2/indicators?  
format=json&page=",  
                page_n)  
  res <- GET(url)  
  parsed <- jsonlite::fromJSON(content(res, "text"),  
                                flatten = TRUE)  
  Sys.sleep(1)  
  return(as_data_frame(parsed[[2]]))  
}
```

Using APIs

An example

Download the data for all pages:

```
get_data <- safely(get_data)
```

```
pages <- data_frame(page_n = 1:parsed[[1]]$pages)
indicators <- pages %>%
  mutate(data = map(page_n, get_data))
```

Using APIs

An example

Check for errors:

```
indicators %>%  
  mutate(error = map(data, "error")) %>%  
  mutate(error = map_int(error, ~ length(.x))) %>%  
  dplyr::filter(error > 0)
```

Using APIs

An example

Get a data frame with all the indicators

```
indicators_clean <- indicators %>%  
  mutate(result = map(data, "result")) %>%  
  select(result) %>%  
  unnest()
```

Using APIs

An example

RStudio

Project: (None)

api_wb.Rmd* indicators_clean

Filter

	id	name	unit	sourceNote	sourceOrganization	topics
1	2.0.cov.Math.pl_3.prv	Coverage: Mathematics Proficiency Level 3, Private sc...		The coverage rate is the childhood access rate of a gi...	LAC Equity Lab tabulations using PISA Data.	2 variables
2	2.0.cov.Math.pl_3.all	Coverage: Mathematics Proficiency Level 3		The coverage rate is the childhood access rate of a gi...	LAC Equity Lab tabulations using PISA Data.	2 variables
3	2.0.cov.Math.pl_2.pub	Coverage: Mathematics Proficiency Level 2, Public sch...		The coverage rate is the childhood access rate of a gi...	LAC Equity Lab tabulations using PISA Data.	2 variables
4	2.0.cov.Math.pl_2.prv	Coverage: Mathematics Proficiency Level 2, Private sc...		The coverage rate is the childhood access rate of a gi...	LAC Equity Lab tabulations using PISA Data.	2 variables
5	2.0.cov.Math.pl_2.all	Coverage: Mathematics Proficiency Level 2		The coverage rate is the childhood access rate of a gi...	LAC Equity Lab tabulations using PISA Data.	2 variables
6	2.0.cov.Int	Coverage: Internet		The coverage rate is the childhood access rate of a gi...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
7	2.0.cov.FPS	Coverage: Finished Primary School		The coverage rate is the childhood access rate of a gi...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
8	2.0.cov.Ele	Coverage: Electricity		The coverage rate is the childhood access rate of a gi...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
9	2.0.cov.Cel	Coverage: Mobile Phone		The coverage rate is the childhood access rate of a gi...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
10	1.3_ACCESS.ELECTRICITY.URBAN	Access to electricity (% of urban population)		Access to electricity is the percentage of total populat...	World Bank Global Electrification Database 2014	list()
11	1.2_ACCESS.ELECTRICITY.RURAL	Access to electricity (% of rural population)		Access to electricity is the percentage of rural populat...	World Bank Global Electrification Database 2013	list()
12	1.2.PSev.Poor4uds	Poverty Severity (\$4 a day)-Urban		The poverty severity index combines information on ...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
13	1.2.PSev.2.5usd	Poverty Severity (\$2.50 a day)-Urban		The poverty severity index combines information on ...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
14	1.2.PSev.1.90usd	Poverty Severity (\$1.90 a day)-Urban		The poverty severity index combines information on ...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
15	1.2.PGap.Poor4uds	Poverty Gap (\$4 a day)-Urban		The poverty gap captures the mean aggregate income...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
16	1.2.PGap.2.5usd	Poverty Gap (\$2.50 a day)-Urban		The poverty gap captures the mean aggregate income...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
17	1.2.PGap.1.90usd	Poverty Gap (\$1.90 a day)-Urban		The poverty gap captures the mean aggregate income...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
18	1.2.HCount.Vul4to10	Vulnerable (\$4–10 a day) Headcount-Urban		The poverty headcount index measures the proportio...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
19	1.2.HCount.Poor4uds	Poverty Headcount (\$4 a day)-Urban		The poverty headcount index measures the proportio...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
20	1.2.HCount.Ofcl	Official Moderate Poverty Rate-Urban		The poverty headcount index measures the proportio...	LAC Equity Lab tabulations of data from National Stati...	2 variables
21	1.2.HCount.Mid10to50	Middle Class (\$10–50 a day) Headcount-Urban		The poverty headcount index measures the proportio...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
22	1.2.HCount.2.5usd	Poverty Headcount (\$2.50 a day)-Urban		The poverty headcount index measures the proportio...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
23	1.2.HCount.1.90usd	Poverty Headcount (\$1.90 a day)-Urban		The poverty headcount index measures the proportio...	LAC Equity Lab tabulations of SEDLAC (CEDLAS and th...	2 variables
24	1.1_YOUTH.LITERACY.RATE	Literacy rate, youth total (% of people ages 15–24)		The number of persons aged 15 to 24 years who can ...	Source of information: UNESCO Institute for Statistics ...	list()
25	1.1_TOTAL.FINAL.ENERGY.CONSUM	Total final energy consumption (TFEC)		Total final energy consumption (TFEC): This indicator ...	World Bank and International Energy Agency (IEA Stati...	list()

Showing 1 to 27 of 15,650 entries, 8 total columns

Environment History Connections

Using APIs

An example

The screenshot shows the World Bank Data Help Desk website. At the top, there is a navigation bar with links for Home, About, Data (which is underlined), Research, Learning, News, Projects & Operations, Publications, Countries, and Topics. The Data link is highlighted with a red underline. To the right of the navigation bar is a search bar with a magnifying glass icon. Below the navigation bar, there is a large red header with the word "Data" in white. Underneath the header, the page title is "Developer Information". There is a backlink to "Knowledge Base". On the left side, there is a sidebar with a list of API documentation topics: Developer Information: Overview, About This API Documentation, New Features and Enhancements in the V2 API, Basic API Call Structures, Country API Queries, Aggregate API Queries, Indicator API Queries, Topic API Queries, Advanced Data API Queries, Metadata API Queries, SDMX API Queries, API: Error Codes, Data Catalog API, Climate Data API, Development Best Practices, and wbopendata: Stata module to access World Bank databases. On the right side, there are three boxes: one for new users to sign in, one for developer info with terms of use, and one for contact support and give feedback.

THE WORLD BANK | Working for a World Free of Poverty

English Español Français عربي Русский 中文 ►

Search

Home About Data Research Learning News Projects & Operations Publications Countries Topics

Data

Developer Information

← Knowledge Base

[Developer Information: Overview](#)
[About This API Documentation](#)
[New Features and Enhancements in the V2 API](#)
[Basic API Call Structures](#)
[Country API Queries](#)
[Aggregate API Queries](#)
[Indicator API Queries](#)
[Topic API Queries](#)
[Advanced Data API Queries](#)
[Metadata API Queries](#)
[SDMX API Queries](#)
[API: Error Codes](#)
[Data Catalog API](#)
[Climate Data API](#)
[Development Best Practices](#)
[wbopendata: Stata module to access World Bank databases](#)

New and returning users may [sign in](#)

Thank you for visiting the World Bank's Data Help Desk. Please review the [terms of use](#) for this website. Your continued use of this website constitutes your acceptance of these terms and conditions.

[Developer Info](#)

Search

Contact support

Give feedback

General Suggestions 3
New Data 0
Website Improvements 3

<https://datahelpdesk.worldbank.org/knowledgebase/topics/125589-developer-information>

Data collection

**Data cleaning
and merging datasets**

Data cleaning

Regular expressions

Regular expressions are sequences that define patterns of characters that can be used for search and replace

Data cleaning

Regular expressions

remove extra spaces

remove stop words

remove prepositions in names

remove accents

remove punctuation

Usage examples

Data cleaning

Regular expressions

A function to trim whitespaces:

```
trim <- function (x){  
  gsub("^\\s+|\\s+$", "", x) %>%  
  gsub("[ ]+", " ", .)  
}
```

```
trim("    Alice    Maravilha    ")
```

```
[1] "Alice Maravilha"
```

Data cleaning

Regular expressions

A function to extract the first and last name:

```
first_last <- function (x){  
  gsub("^[A-z]+.*[A-z]+$",  
        "\\\\1 \\\\2", x)  
}
```

```
first_last("Joaquim Adalberto do Porto e de Lisboa")
```

```
[1] "Joaquim Lisboa"
```

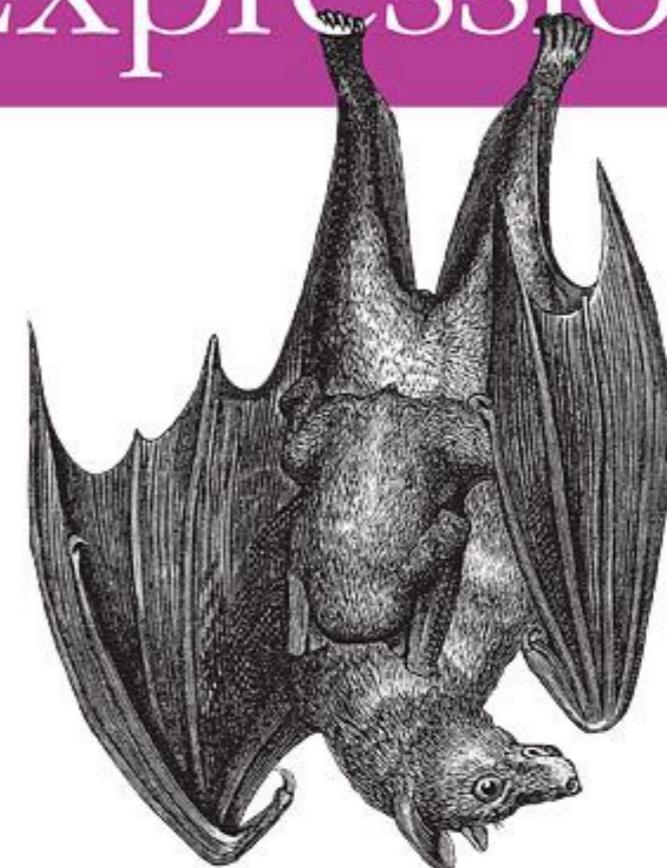
Data cleaning

Regular expressions

Unraveling Regular Expressions, Step-by-Step

Introducing

Regular Expressions



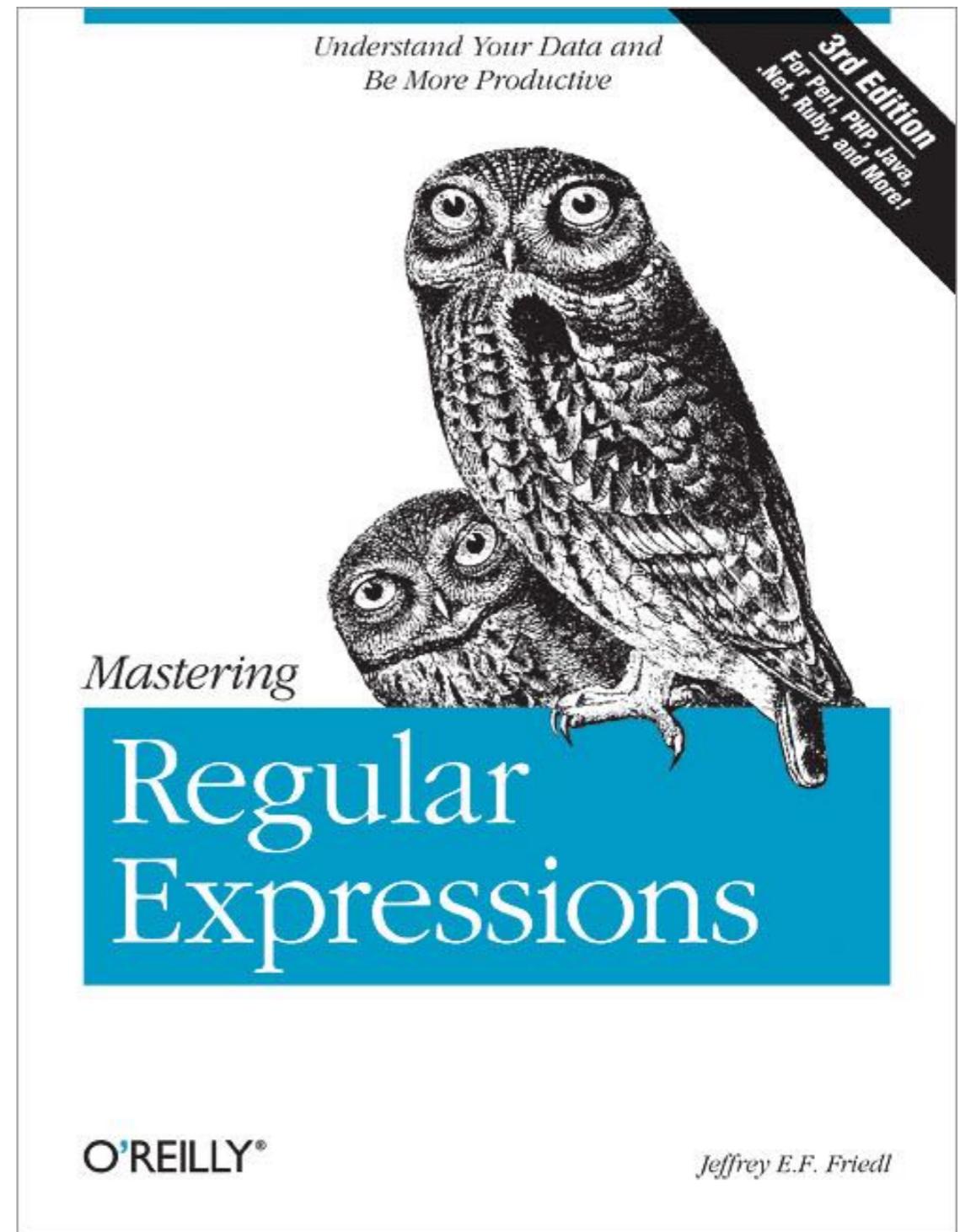
O'REILLY®

Michael Fitzgerald

<http://shop.oreilly.com/product/0636920012337.do>

Data cleaning

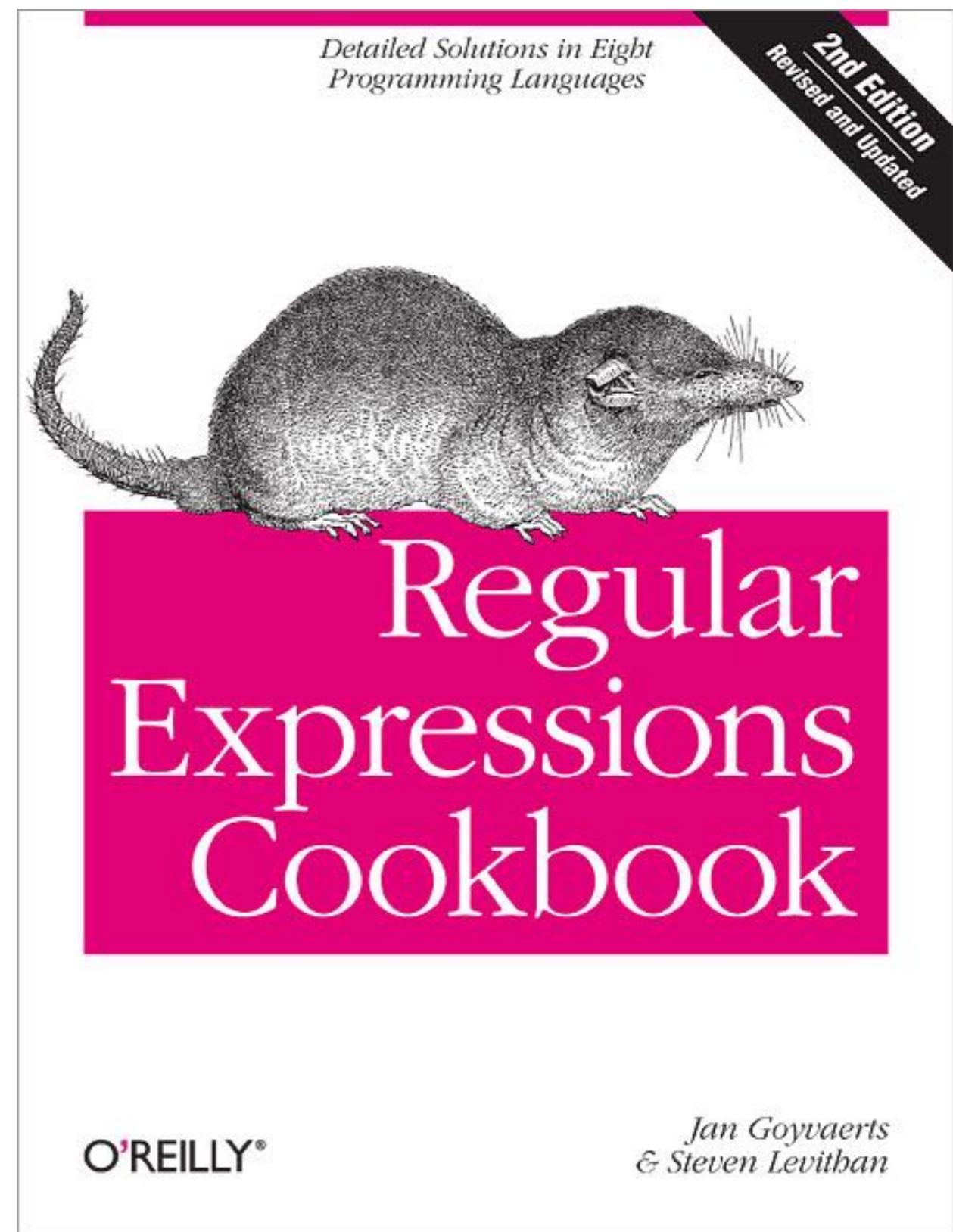
Regular expressions



<http://shop.oreilly.com/product/9780596528126.do>

Data cleaning

Regular expressions



<http://shop.oreilly.com/product/0636920023630.do>

Record disambiguation

Openrefine



“A free, open source, powerful tool for working with messy data”

<http://openrefine.org>

Record disambiguation

Openrefine

demonstration

Record disambiguation

Other tools

RecordLinkage: Record Linkage in R

<https://cran.r-project.org/web/packages/RecordLinkage/index.html>

fastLink: Fast Probabilistic Record Linkage with Missing Data

<https://cran.r-project.org/web/packages/fastLink/index.html>

Working with large data sets

Working with large data sets

Biggish data

Biggish data

Example

The screenshot shows a web browser window for cms.gov. The main content area displays the "Medicare Provider Utilization and Payment Data: Part D Prescriber" page. The page title is "Medicare Provider Utilization and Payment Data: Part D Prescriber". Below the title, there is a detailed description of the Part D Prescriber Public Use File (PUF). The PUF provides information on prescription drugs prescribed by individual physicians and other health care providers and paid for under the Medicare Part D Prescription Drug Program. The Part D Prescriber PUF is based on information from CMS's Chronic Conditions Data Warehouse, which contains Prescription Drug Event records submitted by Medicare Advantage Prescription Drug (MAPD) plans and by stand-alone Prescription Drug Plans (PDP). The dataset identifies providers by their National Provider Identifier (NPI) and the specific prescriptions that were dispensed at their direction, listed by brand name (if applicable) and generic name. For each prescriber and drug, the dataset includes the total number of prescriptions that were dispensed, which include original prescriptions and any refills, and the total drug cost. The total drug cost includes the ingredient cost of the medication, dispensing fees, sales tax, and any applicable administration fees and is based on the amount paid by the Part D plan, Medicare beneficiary, government subsidies, and any other third-party payers.

Although the Part D Prescriber PUF has a wealth of information on payment and utilization for Medicare Part D prescriptions, the dataset has a number of limitations. Of particular importance is the fact that the data may not be representative of a physician's entire practice or all of Medicare as it only includes information on beneficiaries enrolled in the Medicare Part D prescription drug program (i.e., approximately two-thirds of all Medicare beneficiaries). In addition, the data are not intended to indicate the quality of care provided. For additional limitations, please review the methodology document available below.

[Medicare Part D Prescriber Data CY 2016](#)
[Medicare Part D Prescriber Data CY 2015](#)
[Medicare Part D Prescriber Data CY 2014](#)
[Medicare Part D Prescriber Data CY 2013](#)

[Medicare Part D Opioid Prescribing Mapping Tool](#)

Inquiries regarding this data can be sent to MedicareProviderData@cms.hhs.gov.

To receive email notifications, please sign up for the Medicare Provider Data GovDelivery subscription [here](#).

Downloads

[Part D Prescriber PUF Methodology \[PDF, 428KB\]](#)
[Part D Prescriber PUF Frequently Asked Questions \[PDF, 103KB\]](#)

Page last Modified: 05/15/2018 1:53 PM
[Help with File Formats and Plug-Ins](#)

2013 to 2016

<https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/Medicare-Provider-Charge-Data/Part-D-Prescriber.html>

Biggish data

Tools

All examples use **dplyr** and **dbplyr**



allows using the database as an in-memory data frame

Biggish data

No changes of default configuration!

better results are possible on the same hardware

Working with large data sets

Biggish data

Relational databases

sqlite



“SQLite is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine.”

<https://www.sqlite.org>

sqlite

Advantages

self-contained file

no need to install additional software

supports databases up to 140 TB

files can be read in different languages

sqlite

Disadvantages

no user management

not adequate for concurrent writes to the database

not fast enough for interactive analysis

not optimized for data processing

sqlite

Example

```
# install.packages(c("dbplyr", "RSQLite"))
library(dplyr)
library(dbplyr)
library(readr)
library(DBI)
library(purrr)
```

sqlite

Example

Create the connection to the database:

```
dbcon <- dbConnect(RSQLite::SQLite(),  
                    ".../data/prescriptions.sqlite")
```

sqlite

Example

```
files <- data_frame(year = 13:16)

read_and_load_csv <- function(year, dbcon){
  prescriber_data <- read_tsv(
    file = paste0(
      "../data/PartD_Prescriber_PUF_NPI_DRUG_", year, "/"
      PartD_Prescriber_PUF_NPI_DRUG_", year, ".txt"))

  prescriber_data <- prescriber_data %>%
    mutate(year = year)

  dbWriteTable(dbcon,
    "prescriptions",
    prescriber_data,
    append = TRUE)
}

walk(files$year, read_and_load_csv,
      dbcon)
```

653.18s about 10 min

sqlite

Example

there are faster ways to load the data...

sqlite

Example

```
dbListTables(dbcon)
```

```
## [1] "prescriptions"
```

```
dbListFields(dbcon, "prescriptions")
```

```
[1] "npi"
[2] "nppes_provider_last_org_name"
[3] "nppes_provider_first_name"
[4] "nppes_provider_city"
[5] "nppes_provider_state"
[6] "specialty_description"
[7] "description_flag"
[8] "drug_name"
[9] "generic_name"
[10] "bene_count"
[11] "total_claim_count"
[12] "total_30_day_fill_count"
[13] "total_day_supply"
[14] "total_drug_cost"
[15] "bene_count_ge65"
[16] "bene_count_ge65_suppress_flag"
[17] "total_claim_count_ge65"
[18] "ge65_suppress_flag"
[19] "total_30_day_fill_count_ge65"
[20] "total_day_supply_ge65"
[21] "total_drug_cost_ge65"
[22] "year"
```

sqlite

Example

Create a reference to the table:

```
prescriptions <- tbl(dbcon,  
                      "prescriptions")
```

it can now be used as a data frame!

sqlite

Example

Count the number of rows in the table:

```
prescriptions %>% tally()
```

```
97 255 685
```

takes 43s

sqlite

Example

Count the number of distinct prescribers per state and year:

```
qry <- prescriptions %>%
  group_by(nppes_provider_state, year) %>%
  summarise(n = n_distinct(npi)) %>%
  arrange(desc(n))

## # A tibble: 244 x 3
## # Groups: nppes_provider_state [61]
##   nppes_provider_state year     n
##   <chr>        <int> <int>
## 1 CA            16    91469
## 2 CA            15    89246
## 3 CA            14    86586
## 4 CA            13    83802
## 5 NY            16    67635
## 6 NY            15    65499
...
takes 3 min
```

sqlite

Example

Creating indices can help:

```
dbExecute(dbcon,  
'CREATE INDEX nppes_provider_state_year ON  
prescriptions (nppes_provider_state, year);')
```

...but it will increase the database size!

takes 149s

sqlite

Example

```
qry <- prescriptions %>%  
  group_by(nppes_provider_state, year) %>%  
  summarise(n = n_distinct(npi)) %>%  
  arrange(desc(n)) %>%  
  collect()
```

takes about 7 min!

sqlite

not fast enough for run-time analysis!



sqlite

can still be useful

work with a subset of the data in memory
and run the code in batch mode for the complete dataset



PostgreSQL



“PostgreSQL is a powerful, open source object-relational database system with over 30 years of active development that has earned it a strong reputation for reliability, feature robustness, and performance.”

<https://www.postgresql.org>

PostgreSQL

Advantages

feature rich

active development

can be replicated across a cluster of computers
and use load balancing

good geographical information system (GIS)
support

many queries run in parallel

PostgreSQL

Disadvantages

Not optimized for data processing

Requires installing a server

although it can run R functions natively using the PL/R
(R Procedural Language for PostgreSQL)

<https://github.com/postgres-plr/plr>

PostgreSQL

Example

```
dbcon <- dbConnect(RPostgreSQL::PostgreSQL(),  
                    host = "localhost",  
                    dbname = "prescriptions",  
                    user = "cms",  
                    password = "secret")
```

PostgreSQL

Example

Loading the data into the database
took about **29 min**

the loading procedure is exactly the same
as for the sqlite database

PostgreSQL

Example

Count the number of rows in the table:

```
prescriptions %>% tally()
```

97 255 685

takes 2.6 min

PostgreSQL

Example

Count the number of distinct prescribers per state and year:

```
qry <- prescriptions %>%
  group_by(nppes_provider_state, year) %>%
  summarise(n = n_distinct(npi)) %>%
  arrange(desc(n))

## # A tibble: 244 x 3
## # Groups: nppes_provider_state [61]
##   nppes_provider_state year     n
##   <chr>        <int> <int>
## 1 CA            16    91469
## 2 CA            15    89246
## 3 CA            14    86586
## 4 CA            13    83802
## 5 NY            16    67635
## 6 NY            15    65499
...
takes 7.8 min
```

PostgreSQL

Example

Creating indices can help:

```
dbExecute(dbcon,  
'CREATE INDEX nppes_provider_state_year ON  
prescriptions (nppes_provider_state, year);')
```

...but it will increase the database size!

takes 331.5s

PostgreSQL

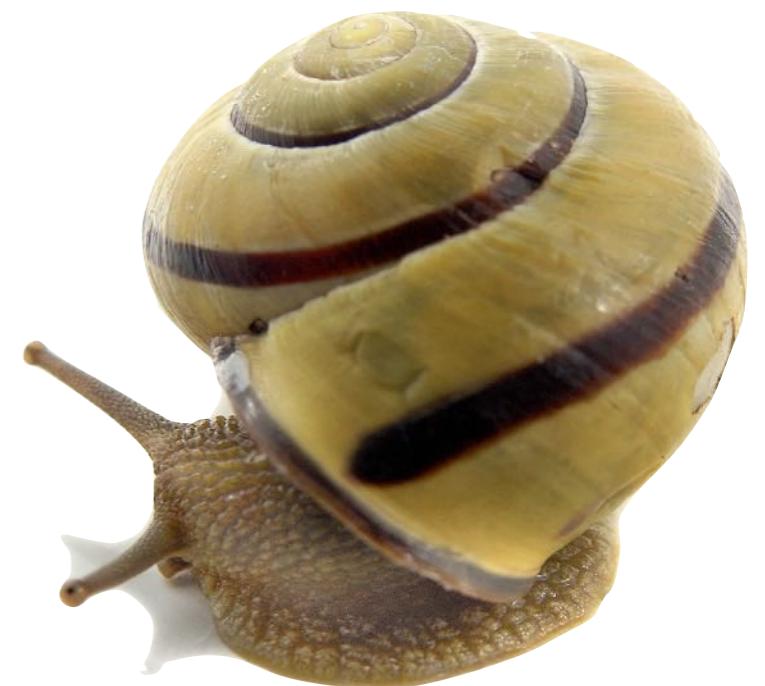
Example

```
qry <- prescriptions %>%  
  group_by(nppes_provider_state, year) %>%  
  summarise(n = n_distinct(npi)) %>%  
  arrange(desc(n)) %>%  
  collect()
```

takes about 7.8 min

PostgreSQL

not fast enough for run-time analysis!



PostgreSQL

Parallelize queries across nodes in a cluster

Citus plugin



<https://www.citusdata.com>

Working with large data sets

Biggish data

Column store databases

Column store databases

These databases store data tables **by column** rather than by row and, this usually allows **faster queries**

Column store databases



“MonetDB is designed for multi-core parallel execution on desktops to reduce response time for complex query processing.”

It is SQL compliant

<https://www.monetdb.org>

MonetDB

Example

The MonetDBLite is compatible with dbplyr

all queries remain the same

no need to install a database server

the database creates the indices automatically

MonetDB

Example

```
dbcon <- dbConnect(MonetDBLite::MonetDBLite(),  
                    ". ./data/monet")
```

MonetDB

Example

Loading the data into the database
took about **12 min**

the loading procedure is exactly the same
as for the sqlite and PostgreSQL databases

MonetDB

Example

Count the number of rows in the table:

```
prescriptions %>% tally()
```

```
97 255 685
```

takes 0.15s

MonetDB

Example

Count the number of distinct prescribers per state and year:

```
qry <- prescriptions %>%
  group_by(nppes_provider_state, year) %>%
  summarise(n = n_distinct(npi)) %>%
  arrange(desc(n))

## # A tibble: 244 x 3
## # Groups: nppes_provider_state [61]
##   nppes_provider_state year     n
##   <chr>        <int> <int>
## 1 CA            16    91469
## 2 CA            15    89246
## 3 CA            14    86586
## 4 CA            13    83802
## 5 NY            16    67635
## 6 NY            15    65499
...
takes 5.37s
```

MonetDB

Example

Find the number of individual doctors per speciality in 2016, ordering the results in descending order:

```
qry <- prescriptions %>%  
  dplyr::filter(year == 16,  
    !is.na(nppes_provider_first_name)) %>%  
  group_by(specialty_description) %>%  
  summarise(n = n_distinct(npi)) %>%  
  arrange(desc(n)) %>%  
  collect()
```

takes 15.8s

MonetDB

Example

	specialty_description	n
1	Internal Medicine	108859
2	Nurse Practitioner	108035
3	Family Practice	99198
4	Dentist	98750
5	Physician Assistant	69747
6	Emergency Medicine	36287
7	Psychiatry	23075
8	Obstetrics & Gynecology	21914
9	Student in an Organized Health Care Education/Training	21906
10	Optometry	21258
11	Cardiovascular Disease (Cardiology)	19743
12	Orthopedic Surgery	19325
13	Ophthalmology	18395
14	General Surgery	16920

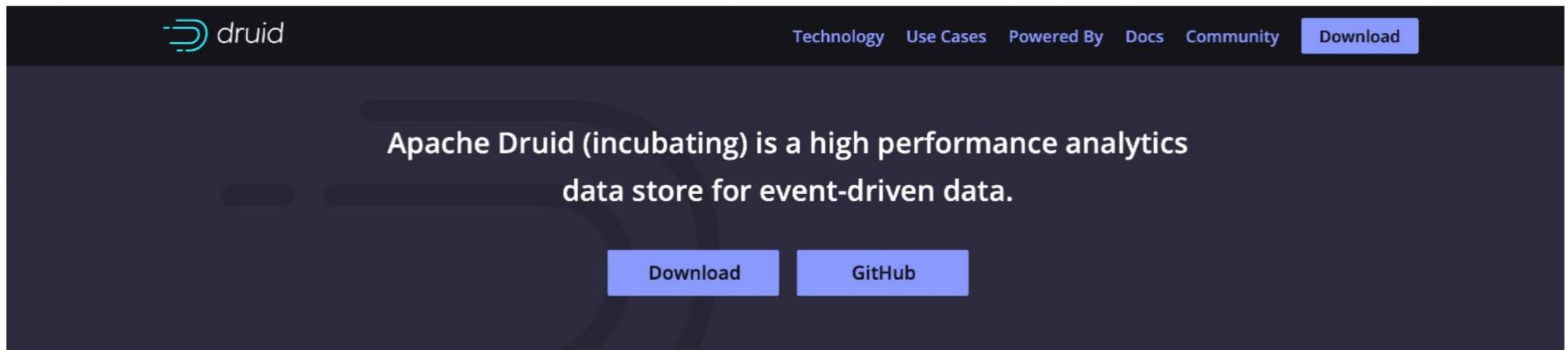
MonetDB



**fast enough to deal with 97m records
out-of-memory on a laptop**

Column store databases

Alternatives



The screenshot shows the Apache Druid homepage. At the top, there's a dark header with the "druid" logo, navigation links for Technology, Use Cases, Powered By, Docs, Community, and a prominent blue "Download" button. Below the header, a large central text area contains the tagline: "Apache Druid (incubating) is a high performance analytics data store for event-driven data." Underneath this text are two blue "Download" and "GitHub" buttons. The background of the main content area is a dark gradient.

Overview



Analyze event streams

Druid provides fast analytical queries, at high concurrency, on both real-time and historical data. Druid is often used to power interactive UIs.



Utilize reimaged architecture

Druid is a new class of data store that combines ideas from [OLAP/analytic databases](#), [timeseries databases](#), and [search systems](#) to enable new use cases with streaming and batch data.



Build next generation data stacks

Druid integrates natively with message buses (Kafka, AWS Kinesis, etc) and data lakes (HDFS, AWS S3, etc). Druid works especially well as a query layer for streaming data architectures.



Unlock new workflows

Druid is built for rapid, ad-hoc analytics on both real-time and historical data. Explain trends, explore data, and quickly iterate on queries to answer questions.



Deploy anywhere

Druid can be deployed in any *NIX environment on commodity hardware, both in the cloud and on premise. Druid is cloud-native: scaling up and down is as simple as adding and removing processes.

Latest releases

[Druid 0.12.3 Released](#)

Sep 18 2018

[Druid 0.12.2 Released](#)

Aug 9 2018

[Druid 0.12.1 Released](#)

Jun 8 2018

[Druid 0.12.0 Released](#)

Mar 8 2018

Upcoming Events

[DEC 6 Big Data and Cloud Meetup](#)

Demystifying AWS Analytics and Building Streaming Analytics using Kafka & Druid

[Join a Druid Meetup!](#)

Featured Content

[How Druid enables analytics at Airbnb](#)

Working with large data sets

Biggish data

Document databases

Document databases



“MongoDB is a document database (...) stores data in flexible, JSON-like documents, meaning fields can vary from document to document and data structure can be changed over time

MongoDB is a distributed database at its core, so high availability, horizontal scaling, and geographic distribution are built in and easy to use”

<https://www.mongodb.com/what-is-mongodb>

Document databases

JSON - Javascript Object Notation

```
{  
    "firstName": "John",  
    "lastName": "Smith",  
    "isAlive": true,  
    "age": 27,  
    "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"  
        },  
        {  
            "type": "mobile",  
            "number": "123 456-7890"  
        }  
    ],  
    "children": [],  
    "spouse": null  
}
```

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

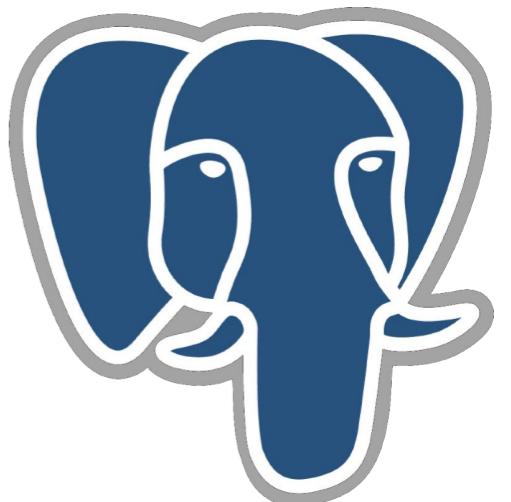
Document databases

Use cases

**Storing data from APIs with no
clear schema definition**

Document databases

It is also possible to store JSON documents
in a PostgreSQL database



Working with large data sets

Bigger data

Spark



“Apache Spark™ is a unified analytics engine for large-scale data processing.”

<https://spark.apache.org>

Spark



Download Libraries ▾ Documentation ▾ Examples Community ▾ Developers ▾

Spark SQL is Apache Spark's module for working with structured data.

Integrated

Seamlessly mix SQL queries with Spark programs.

Spark SQL lets you query structured data inside Spark programs, using either SQL or a familiar [DataFrame API](#). Usable in Java, Scala, Python and R.

```
results = spark.sql(  
    "SELECT * FROM people")  
names = results.map(lambda p: p.name)
```

Apply functions to results of SQL queries.

Uniform Data Access

Connect to any data source the same way.

DataFrames and SQL provide a common way to access a variety of data sources, including Hive, Avro, Parquet, ORC, JSON, and JDBC. You can even join data across these sources.

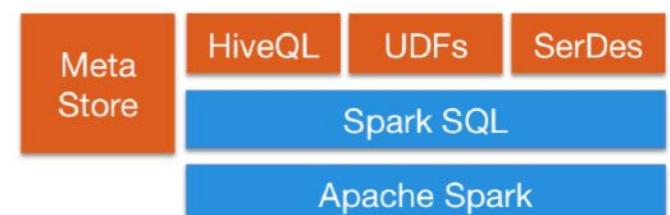
```
spark.read.json("s3n://...")  
    .registerTempTable("json")  
results = spark.sql(  
    """SELECT *  
    FROM people  
    JOIN json ...""")
```

Query and join different data sources.

Hive Integration

Run SQL or HiveQL queries on existing warehouses.

Spark SQL supports the HiveQL syntax as well as Hive SerDes and UDFs, allowing you to access existing Hive warehouses.



Spark SQL can use existing Hive metastores, SerDes, and UDFs.

Spark

Can run on a **local machine**,
on a local cluster or
on a **cluster in the cloud**

Spark

R has an interface to **Spark** that allows using **dplyr**

not much changes!

Spark

Example

Load the required packages:

```
library(sparklyr)  
library(dplyr)  
library(readr)  
library(purrr)
```

you'll need to install JDK
and spark locally

Spark

Example

Tune the Spark configuration and open the connection to Spark:

```
Sys.setenv(JAVA_HOME = "/Library/Java/  
JavaVirtualMachines/jdk1.8.0_191.jdk/Contents/Home/")  
config <- spark_config()  
config$`sparklyr.shell.driver-java-options` <-  
paste0("-Djava.io.tmpdir=", "../data/spark/")  
config$`sparklyr.shell.driver-memory` <- "4G"  
config$`sparklyr.shell.executor-memory` <- "4G"  
config$`spark.yarn.executor.memoryOverhead` <- "512"  
  
dbcon <- spark_connect(master = "local",  
                         config = config)
```

Spark

Example

Function to read the
csv files and
copy them to Spark

```
files <- data_frame(year = 13:16)

read_and_load_csv <- function(year, dbcon){

  aux <- spark_read_csv(
    dbcon,
    name = "aux",
    path = paste0(
      "./data/PartD_Prescriber_PUF_NPI_DRUG_",
      year, "/PartD_Prescriber_PUF_NPI_DRUG_",
      year, ".txt"),
    delimiter = "\t",
    overwrite = TRUE,
    memory = FALSE)

  if ("prescriber" %in% src_tbls(dbcon)){
    prescriber_data <- tbl(dbcon, "prescriber")
    aux %>%
      mutate(year = year) %>%
      sdf_bind_rows(prescriber_data) %>%
      sdf_register("prescriber")
  } else {
    aux %>%
      mutate(year = year) %>%
      sdf_register("prescriber")
  }
}
```

Spark

Example

Read each csv file and copy them to Spark:

```
walk(files$year, read_and_load_csv,  
      dbcon)
```

	user	system	elapsed
	4.926	0.390	160.748

Spark

Example

Count the number of distinct prescribers per state and year:

```
qry <- prescriptions %>%  
  group_by(nppes_provider_state, year) %>%  
  summarise(n = n_distinct(npi)) %>%  
  arrange(desc(n))
```

	user	system	elapsed
	1.464	0.265	115.086

Spark

Example

Find the number of individual doctors per speciality in 2016, ordering the results in descending order:

```
qry <- prescriptions %>%
  dplyr::filter(year == 16,
    !is.na(nppes_provider_first_name)) %>%
  group_by(specialty_description) %>%
  summarise(n = n_distinct(npi)) %>%
  arrange(desc(n)) %>%
  collect()
```

	user	system	elapsed
	0.732	0.118	56.614

Spark

Example

**Deploy a virtual machine
in the cloud to run Spark**



Spark

Example

Azure Distributed Data Engineering Toolkit

“(...) is a python CLI application for provisioning on-demand Spark on Docker clusters in Azure. It’s a cheap and easy way to get up and running with a Spark cluster, and a great tool for Spark users who want to experiment and start testing at scale.”

<https://github.com/Azure/aztk>



Spark

Example

The screenshot shows an RStudio interface with the following components:

- Top Bar:** Contains navigation icons (back, forward, search, etc.), a URL bar set to "localhost:8787", and a toolbar with various RStudio and system icons.
- Environment Tab:** Shows a table of prescription counts by provider specialty. The specialties listed are Family Practice, Dentist, Physician Assistant, Emergency Medicine, Psychiatry, Obstetrics & Gynecology, Student in an Organized Health Care Education/Train..., and Optometry. The counts range from 99199 down to 21258.
- Code Editor:** Displays R code for querying a database and performing a group-by operation on prescriptions. The code includes `tbl_cache`, `group_by`, `summarise`, `arrange`, and `collect` functions. It also shows a timing output for the query execution.
- Files Tab:** Browsing the "notebooks" directory, showing files like `derby.log`, `metastore_db`, `spark_cluster.nb.html` (selected), and `spark_cluster.Rmd`.
- Console:** Shows the command `spark_disconnect(dbcon)`.

After provisioning the virtual machine, everything is the same

Spark

Example

1 F8 VM

The F-series virtual machines support 2-GiB RAM and 16 GB of local solid state drive (SSD) per CPU core, and are optimized for compute intensive workloads. The F-series is based on the 2.4 GHz Intel Xeon® E5-2673 v3 (Haswell) processor, which can achieve clock speeds as high as 3.2 GHz with the Intel Turbo Boost Technology 2.0. These virtual machines are suitable for scenarios like batch processing, web servers, analytics, and gaming.

For persistent storage, use the variant Fs virtual machines and purchase Premium Storage separately. The pricing and billing meters for Fs sizes are the same as F-series.

ADD TO ESTIMATE	INSTANCE	CORE	RAM	TEMPORARY STORAGE	PAY AS YOU GO	ONE YEAR RESERVED (% SAVINGS)	THREE YEAR RESERVED (% SAVINGS)
+ F1	F1	1	2.00 GiB	16 GiB	€0.049/hour	€0.035/hour (~28%)	€0.024/hour (~51%)
+ F2	F2	2	4.00 GiB	32 GiB	€0.097/hour	€0.07/hour (~28%)	€0.048/hour (~51%)
+ F4	F4	4	8.00 GiB	64 GiB	€0.192/hour	€0.141/hour (~27%)	€0.094/hour (~51%)
+ F8	F8	8	16.00 GiB	128 GiB	€0.383/hour	€0.281/hour (~27%)	€0.188/hour (~51%)
+ F16	F16	16	32.00 GiB	256 GiB	€0.767/hour	€0.561/hour (~27%)	€0.376/hour (~51%)

<https://azure.microsoft.com/en-us/pricing/details/virtual-machines/linux/>

Spark

Example

Count the number of distinct prescribers per state and year:

```
qry <- prescriptions %>%  
  group_by(nppes_provider_state, year) %>%  
  summarise(n = n_distinct(npi)) %>%  
  arrange(desc(n))
```

	user	system	elapsed
	0.112	0.060	39.163

Spark

Example

Find the number of individual doctors per speciality in 2016, ordering the results in descending order:

```
qry <- prescriptions %>%
  dplyr::filter(year == 16,
    !is.na(nppes_provider_first_name)) %>%
  group_by(specialty_description) %>%
  summarise(n = n_distinct(npi)) %>%
  arrange(desc(n)) %>%
  collect()
```

	user	system	elapsed
	0.065	0.042	18.491

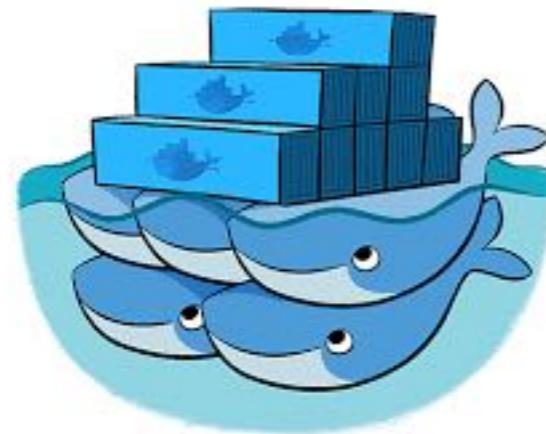
Spark

Example

it is equally easy to provision a cluster with Hadoop
these is where the speed gains occur

Spark

Example



docker swarm



kubernetes

Distributed computing

Distributed computing

Parallel computation

Parallel computation



Parallel computation

Example

**Estimate a Normal-GARCH(1,1) model
for each of the S&P-500 constituents and
plot the conditional volatility,
over the period from
January, 1 2000 to December, 13 2018**

Parallel computation

Example

Embarrassingly parallel



Parallel computation

Example

Get the tickers:

```
tickers <- readRDS(file =
  "../data/sp_500_constituents.rds")  
  
tickers <- tickers %>%
  select(Symbol, Security) %>%
  setNames(c("ticker", "co_name")) %>%
  mutate(ticker = ticker %>%
    gsub("\\.", "-", .))
```

Parallel computation

Example

Function to download the price data from Yahoo Finance:

```
get_data <- function(ticker){  
  
  aux <- getSymbols(Symbols = ticker,  
                     from = "2000-01-01",  
                     to = "2018-12-13",  
                     auto.assign = FALSE)  
  aux <- Ad(aux)/lag(Ad(aux)) - 1  
  aux <- na.omit(aux)  
  return(aux)  
  
}
```

Parallel computation

Example

Download the price data from Yahoo Finance for each stock:

```
returns <- tickers %>%
  mutate(ret = map(ticker, get_data))
```

Parallel computation

Example

Function to estimate the Normal-GARCH(1,1) model:

```
estimate_garch_model <- function(ret){  
  
  spec <- ugarchspec(variance.model =  
    list(model = "sGARCH",  
         garchOrder = c(1,1)),  
    mean.model =  
    list(armaOrder = c(0,0),  
         include.mean = TRUE),  
    distribution.model = "std")  
  fit <- ugarchfit(data = ret,  
                    spec = spec)  
}
```

Parallel computation

Example

Estimate the Normal-GARCH(1,1) model for each stock:

```
results <- foreach(i = 1:NROW(returns),  
                     .packages = "rugarch")  
    %dopar%  
    estimate_garch_model(returns$ret[[i]])
```

	user	system	elapsed
	332.186	11.141	355.618

Parallel computation

Example

Estimate the Normal-GARCH(1,1) model for each stock, **but now in parallel** (using the package **doParallel**):

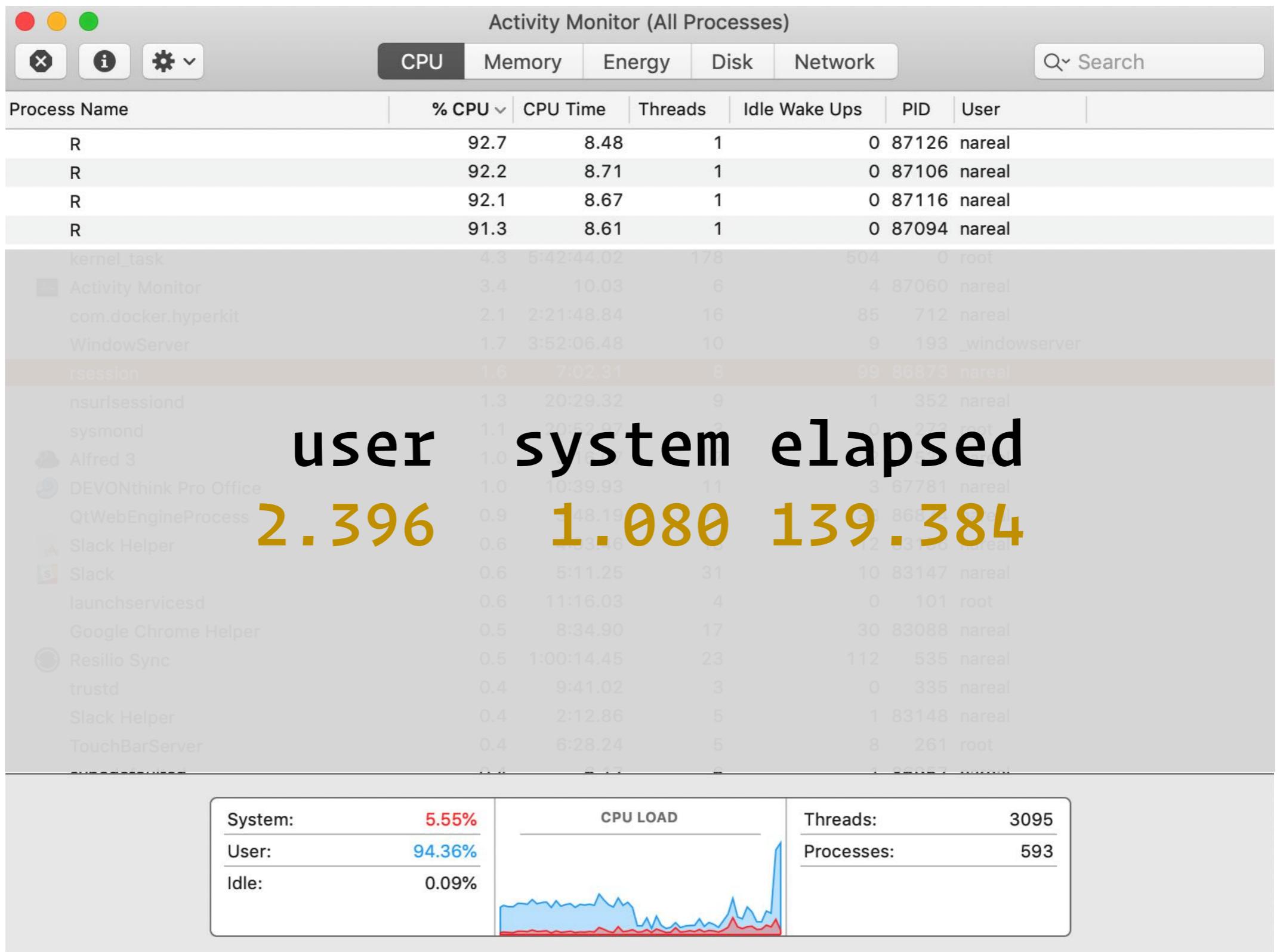
```
cl <- makeCluster(4)
registerDoParallel(cl)

results <- foreach(i = 1:NROW(returns),
                     .packages = "rugarch") %dopar%
estimate_garch_model(returns$ret[[i]])

stopCluster(cl)
```

Parallel computation

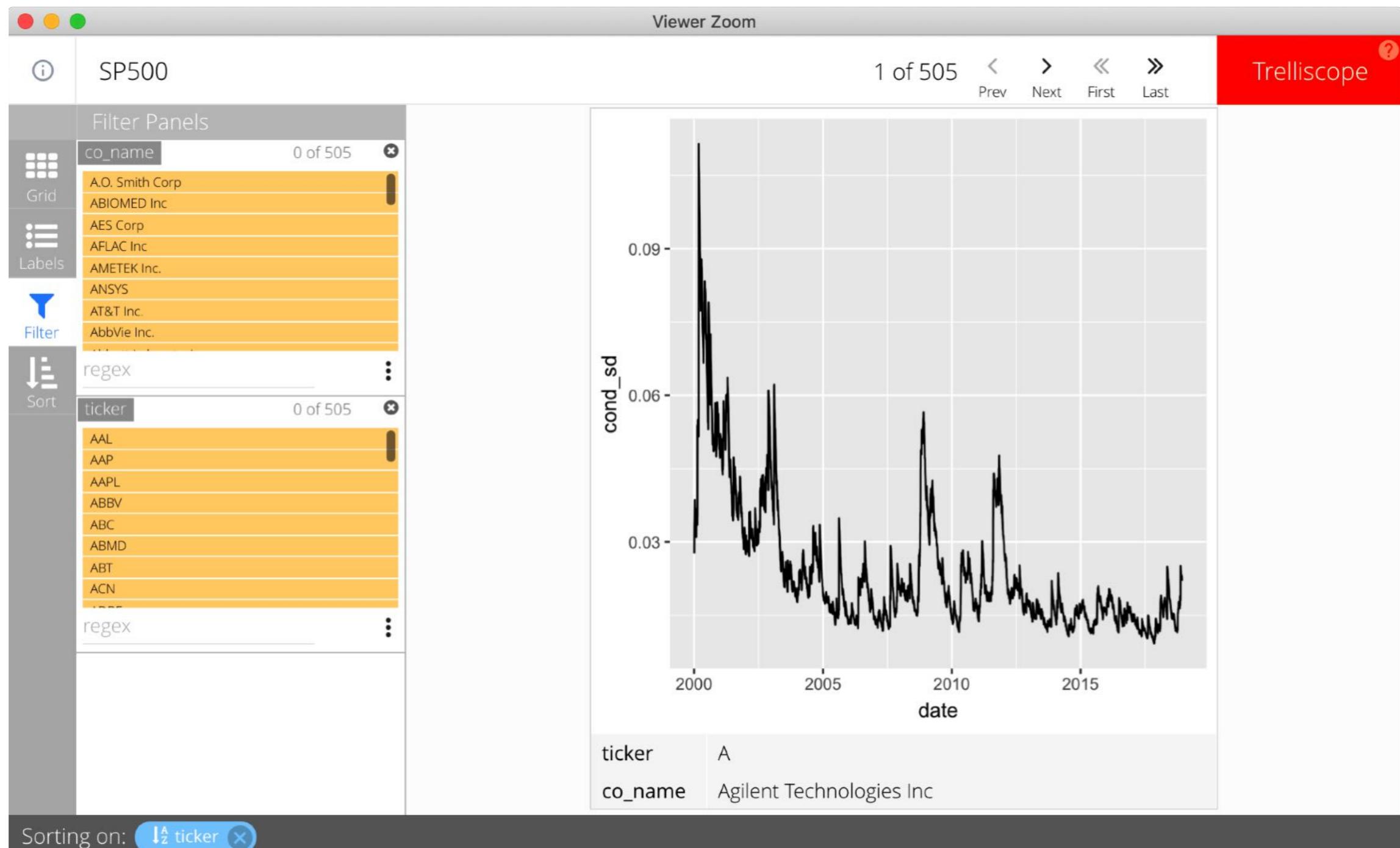
Example



Parallel computation

Example

Use trelliscopejs to display the results:



Distributed computing

Distributed computing

Distributed computation

To spread the computation over a cluster of computers is now trivial

just change the **doParallel** backend
for instance using snow (for a local cluster),
or **doAzureParallel** for a cloud cluster

Distributed computation

Example

The screenshot shows a dark-themed Microsoft Azure documentation page. At the top right, there is a link to 'Contact Sales: 1-800-867-1389'. Below the header, there is a navigation bar with links for 'ion', 'Pricing', 'Training', 'Marketplace', 'Partners', 'Support', 'Blog', and 'More'. At the bottom of the page, there are three buttons: 'Feedback' (with a speech bubble icon), 'Edit' (with a pencil icon), and 'Share' (with a share icon).

Tutorial: Run a parallel R simulation with Azure Batch

01/23/2018 • 6 minutes to read • Contributors 

Run your parallel R workloads at scale using [doAzureParallel](#), a lightweight R package that allows you to use Azure Batch directly from your R session. The doAzureParallel package is built on top of the popular [foreach](#) R package. doAzureParallel takes each iteration of the foreach loop and submits it as an Azure Batch task.

This tutorial shows you how to deploy a Batch pool and run a parallel R job in Azure Batch directly within RStudio. You learn how to:

- ✓ Install doAzureParallel and configure it to access your Batch and storage accounts
- ✓ Create a Batch pool as a parallel backend for your R session
- ✓ Run a sample parallel simulation on the pool

Prerequisites

- An installed [R](#) distribution, such as [Microsoft R Open](#). Use R version 3.3.1 or later.
- [RStudio](#), either the commercial edition or the open-source [RStudio Desktop](#).

<https://docs.microsoft.com/en-us/azure/batch/tutorial-r-doazureparallel>

Distributed computing

GPU

GPU



specialized processors, **not** suitable for general purpose computing

allows massive parallel execution of code

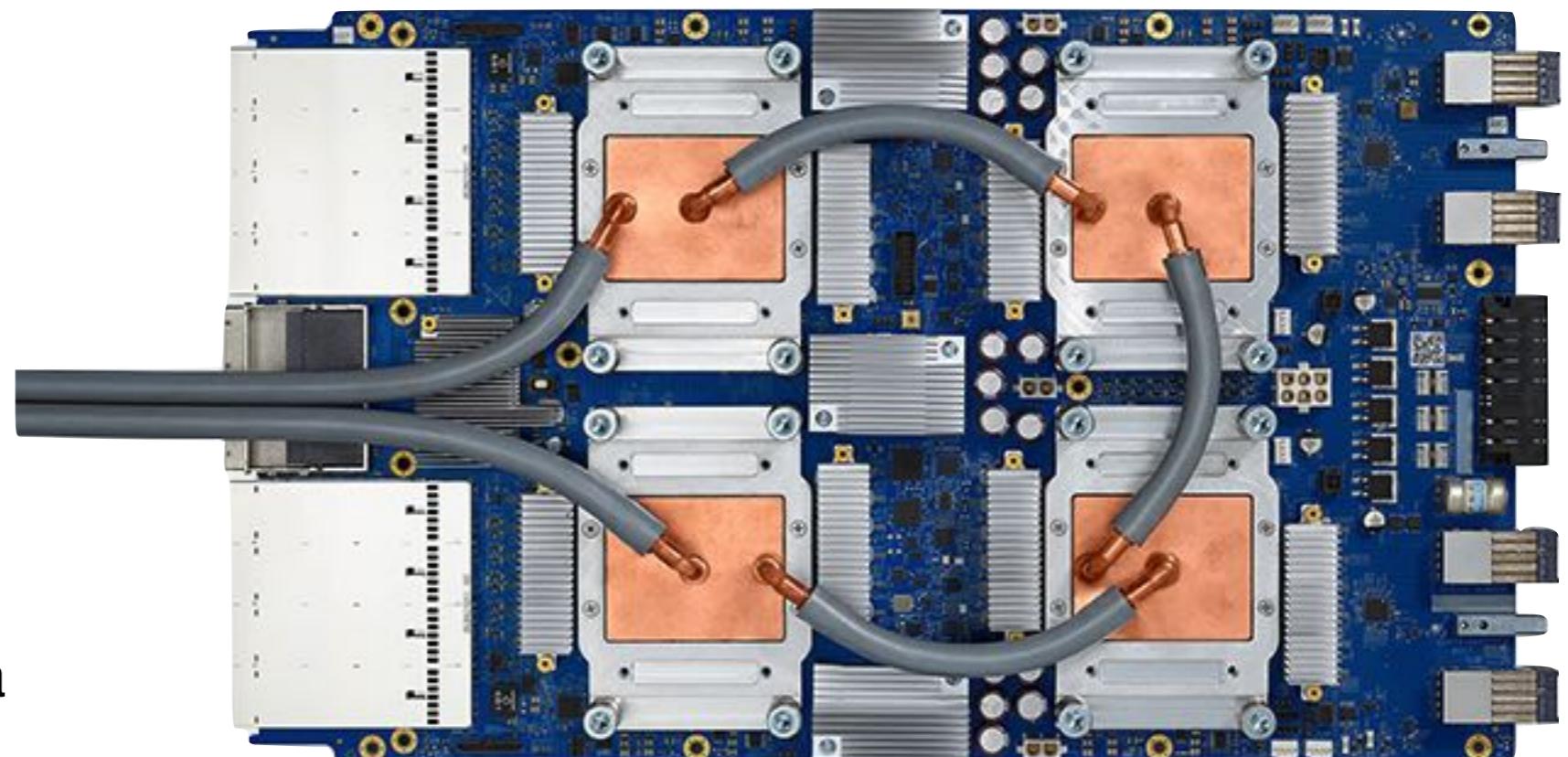
needs specific software to be used

there are basic linear algebra libraries that use it

often used in deep learning with great benefit

GPU

Google Tensor Processing Unit



Cloud TPU v3 Beta
420 teraflops
128 GB HBM

optimized for neural networks

GPU



MLlib library has support for GPUs

GPU



Google Cloud

cloudera



GPU

 **NVIDIA DEVELOPER** NEWS BLOG FORUMS

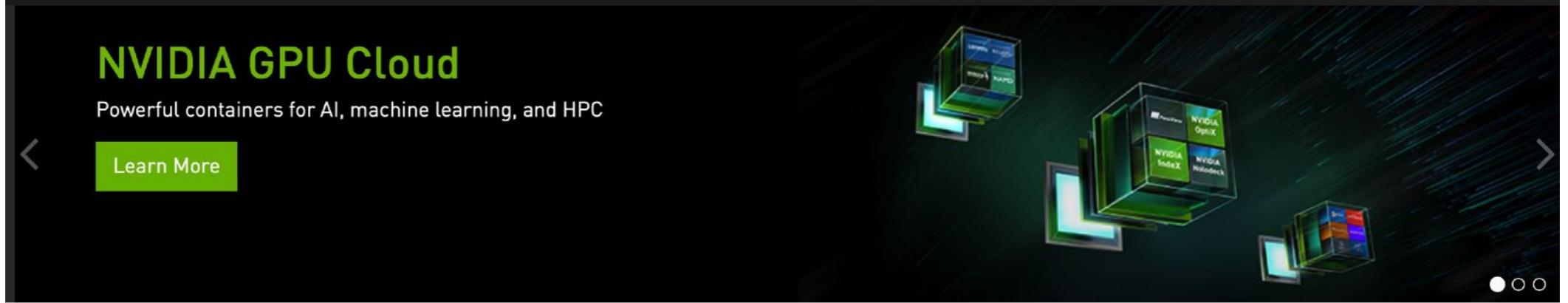
Join Login

RTX GAMEWORKS DESIGNWORKS VRWORKS COMPUTEWORKS JETPACK DRIVE CLARA

NVIDIA GPU Cloud

Powerful containers for AI, machine learning, and HPC

[Learn More](#)



Home

 Deep Learning	 Machine Learning	 Inference
 High Performance Computing	 Autonomous Machines	 Autonomous Vehicles
 Ray Tracing	 Game Development	 Design and Visualization

 **GPU-Accelerated Containers**

Featuring software for AI, machine learning, and HPC, the NVIDIA GPU Cloud (NGC) container registry provides GPU-accelerated containers that are tested and optimized to take full advantage of NVIDIA GPUs.

[Get Started](#)

Join the NVIDIA Developer Program

Access everything you need to develop with NVIDIA products

[Learn More](#)

<https://developer.nvidia.com/nvidia-developer-zone>

GPU

The screenshot shows the homepage of the Paperspace website. At the top, there is a navigation bar with a logo containing a 'P' inside a circle, followed by links for 'PRODUCTS', 'SOLUTIONS', 'LEARN', 'PRICING', and 'SIGN IN'. The background of the page is a vibrant red color with a subtle network graph pattern. The main title 'THE CLOUD BUILT FOR MACHINE LEARNING' is displayed in large, bold, dark blue capital letters. Below the title, there is a descriptive text block in dark blue that reads: 'Super powerful GPU-backed VMs in the cloud. The easiest way to get started with Machine Learning, Artificial Intelligence, and Data Science'.

PRODUCTS SOLUTIONS LEARN PRICING SIGN IN

THE CLOUD BUILT FOR MACHINE LEARNING

Super powerful GPU-backed VMs in the cloud.
The easiest way to get started with Machine Learning, Artificial Intelligence, and Data Science

<https://www.paperspace.com>

