







Using R to Create Quantitative Insights on the Benefit from Open Sourcing Company R Packages

James Black
 Data and Statistical Sciences
 Roche | Genentech

This talk depends on  (GitStats) a package funded by Roche and designed and implemented by Maciej Banas

Table of contents

-  A paradigm shift to open source
-  Accessing and using the data
-  Limitations
-  An eco-system of tools to get into this data

The WTF's in clinical reporting pre-2020



The majority of evidence that get's medicines approved **was in a language you must pay to use**



Huge internal codebases duplicated across pharma



Statisticians **designed in R but statistical programmers analysed in SAS**



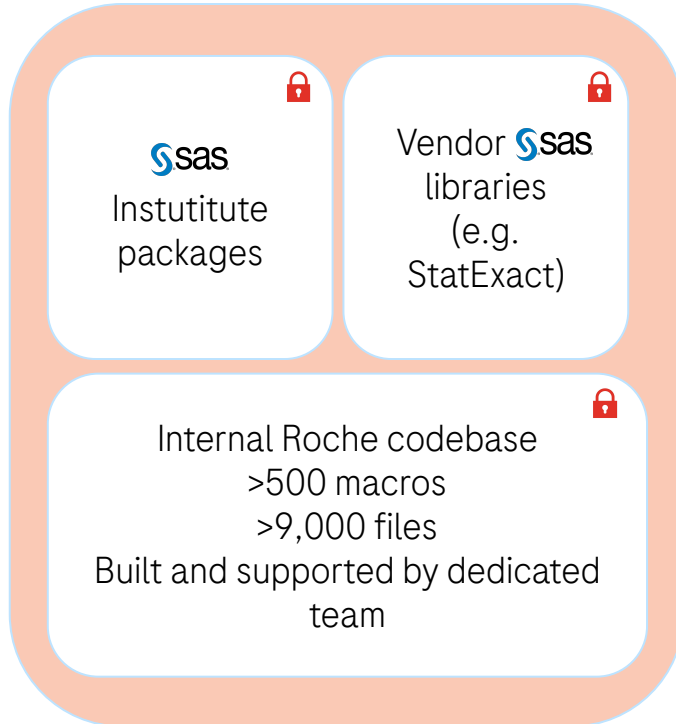
Talent wanted to use open source languages that dominate data science



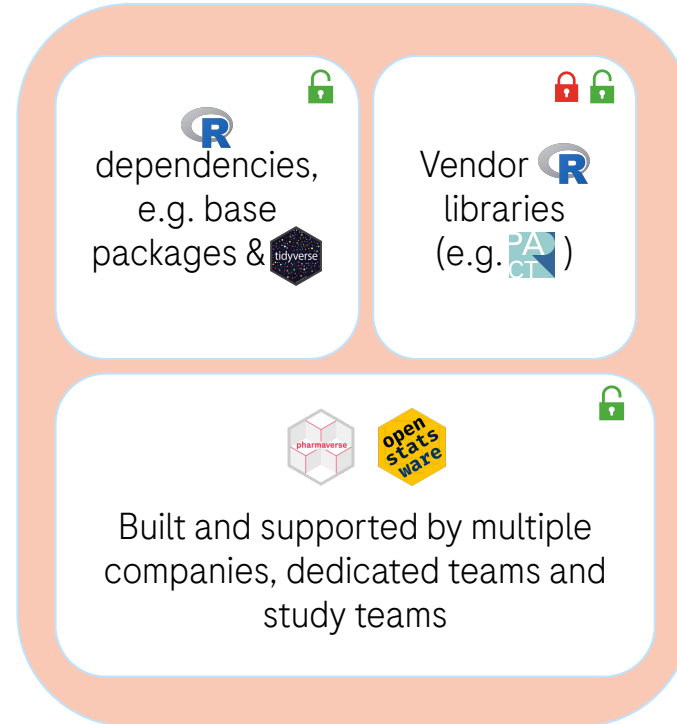
A paradigm shift to open source

What is a statistical programming codebase?

Our historical approach



Our model today

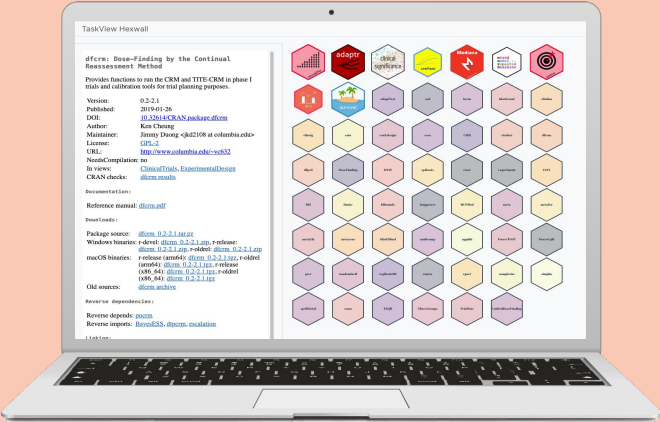


Core collaborations for clinical trial design and reporting

Biostatistics



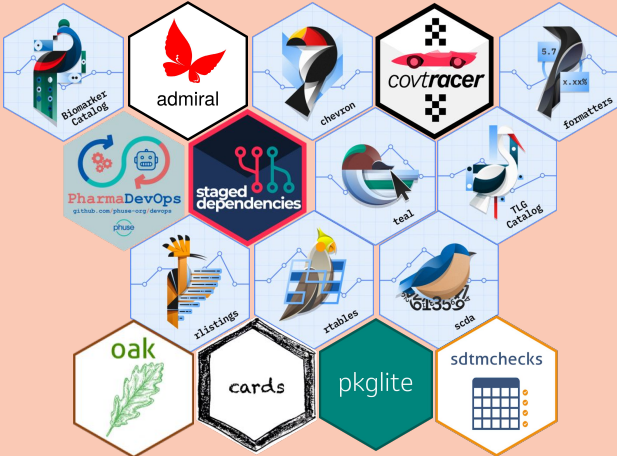
openstatsware.org



Statistical programming



pharmaverse.org





Why Open Source (OS)?

A photograph of a sunset over the ocean. The sun is low on the horizon, creating a bright, shimmering path of light across the water's surface. The sky is a mix of deep blue and orange, and the waves are gently breaking. The text "A rising tide lifts all boats" is overlaid in the center in a bold, white, sans-serif font.

A rising tide lifts all boats



*But what's the
measurable
value to **us** of OS?*

Can we classify what is an ‘internal’ vs ‘external’ contribution to a package?



Roche contribution



Work email from `.gitconfig`



Github handle used at work



Employment periods



External contribution



Any other email in `.gitconfig`



Any other Github handle



Internal account, but outside of employment window

Can we classify what is an 'internal' vs 'external' contribution to a package?



Roche contribution



Work email from `.gitconfig`



Github handle used at work



Employment periods



Quite accurate!



External contribution



Any other email in `.gitconfig`



Any other Github handle



Internal account, but outside of employment window



People can be double counted, especially on packages started outside of Github



Getting to know the contributions from **commits**

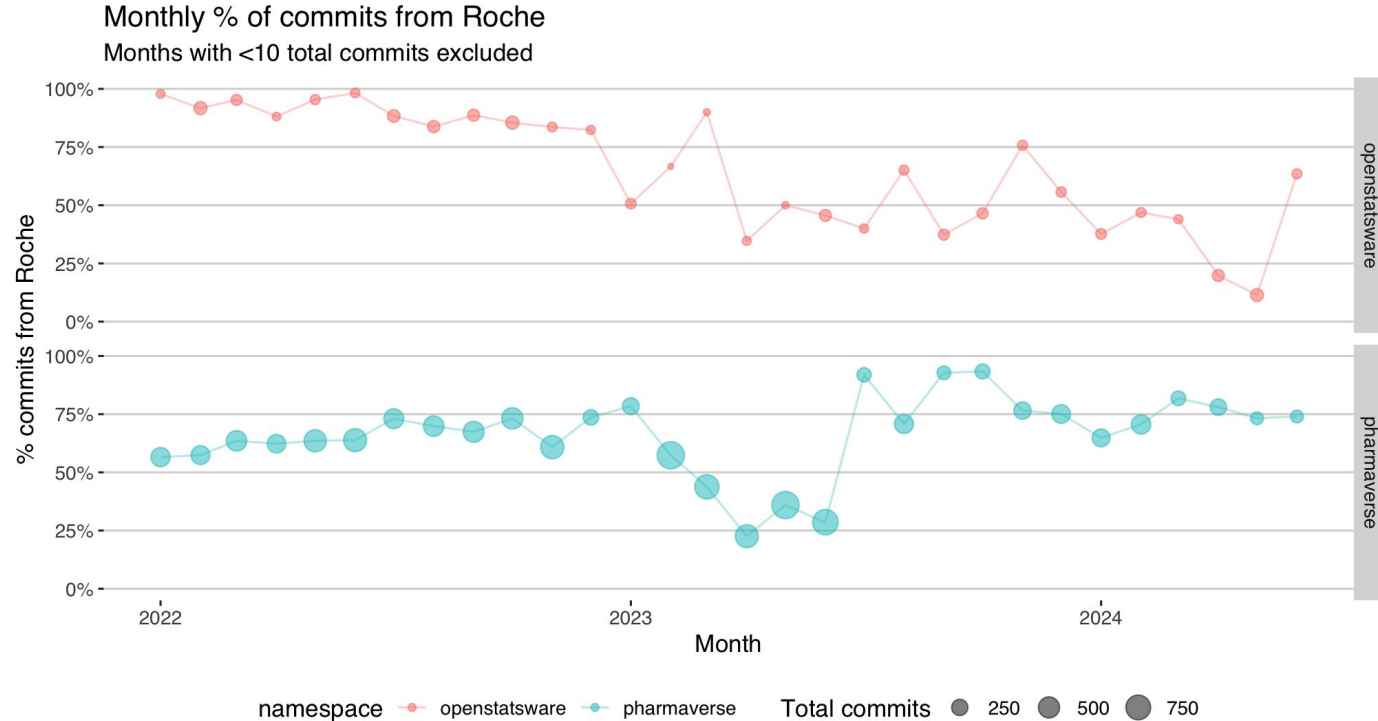


Getting commit data

```
create_gitstats() |>
  set_github_host(repos = c("pharmaverse/admiral", "insightengineering/teal")) |>
  get_commits(since = "2010-01-01")

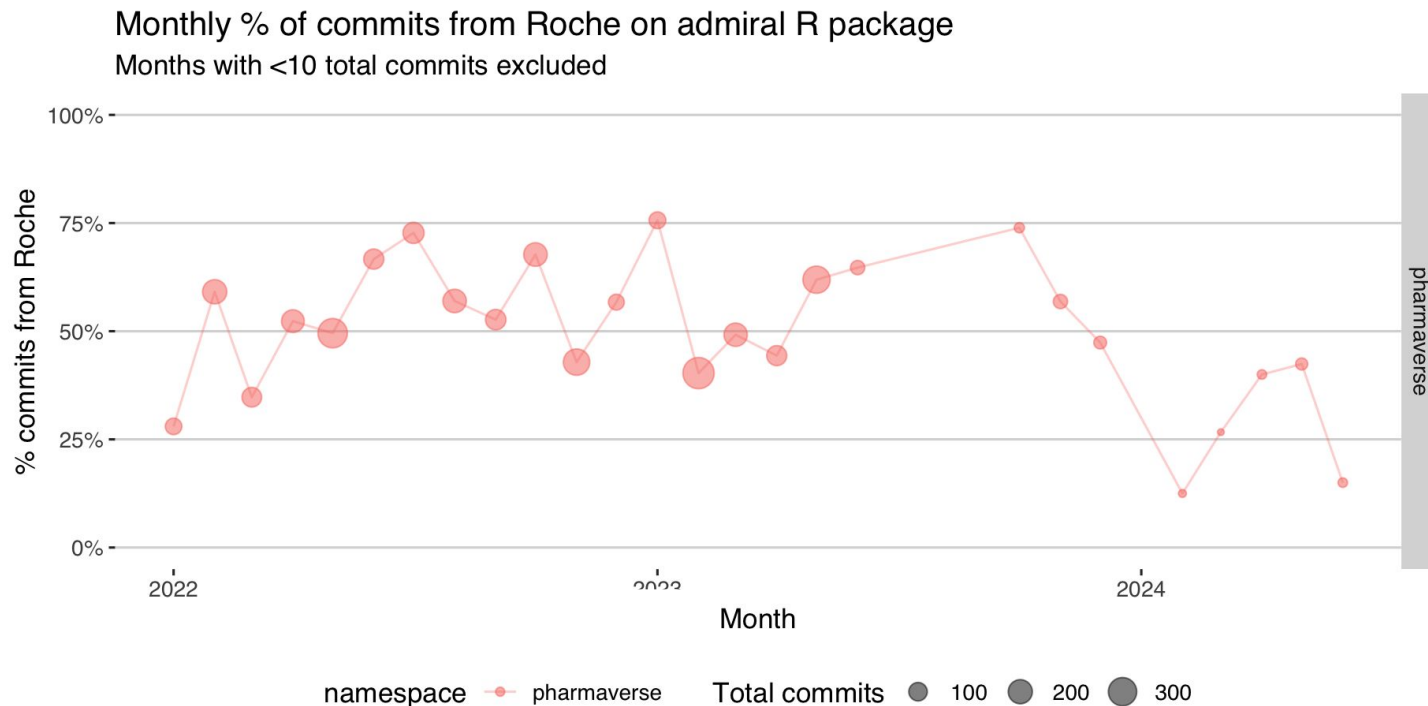
# A tibble: 7,264 × 10
  id      committed_date      author author_login author_name  additions
<chr>    <dtm>                <chr>  <chr>         <chr>         <int>
4 more variables: deletions <int> repository <chr> organization <chr> api_url <chr>
```

Our pan-study codebase is consistently being co-created with external contributors!





~½ the activity on admiral was external commits over the last 2 years





Contributions to the Roche relevant pharmaverse



















Only includes commit data for now, other forms of contribution will be added (e.g. issues)

Package	Age (y)	Roche + Externals		Roche metrics		Monthly trend ³
		Contributors	Commits	Contributors ¹	Commits ²	
pharmaverse						
diffdf	7	4	252	100%	100%	
rtables	6.6	28	1073	93%	100%	
datacutr	2	9	373	89%	99%	
tern	7.3	58	2910	90%	95%	
teal.modules.general	5.7	33	1361	94%	92%	
teal.code	2.3	20	182	90%	92%	
teal.slice	2.3	22	367	91%	91%	
teal	7.3	39	2818	85%	91%	
teal.modules.clinical	6.8	49	2030	92%	91%	
teal.data	2.3	24	296	92%	89%	
admiralophtha	2	22	581	59%	88%	
admiralonco	2.4	24	1033	46%	77%	
falcon	1.6	15	161	40%	76%	
admiral	3.4	79	4444	47%	62%	
admiraldev	2	25	701	60%	52%	
xportr	3.5	12	1129	33%	14%	
metatools	2.5	5	183	20%	5%	
admiralvaccine	1.8	18	1517	22%	3%	
metacore	3.4	10	388	20%	1%	

Commit data +
Scheduled Quarto =
**Monthly report of internal
vs external contribution
trends**



Getting to know the contributions from **LoC**

Package	% Roche ¹
openstatsware	
 jpost	100%
 rbmi	98%
 crmPack	92%
 mmrm	65%
 simIDM	64%
brms.mmrm	0%
pharmaverse	
 datacutr	100%
differ	100%
 admiralophtha	94%
 tern	93%
teal.slice	88%
teal.data	84%
teal.modules.clinical	81%
 teal	80%
teal.modules.general	67%
 admiralonce	66%
 admiral	63%
 rtables	49%
teal.code	46%
 falcon	38%
 admiraldev	32%
 xportr	26%
 admiralvaccine	18%
 metatools	4%
 metacore	1%
pkglite	0%

¹ A line of code is attributed to Roche if the person ever worked for Roche. `xportr` is an example where lines of code were contributed before the person joined Roche's team.

Our list of github handles of Roche employees may not be complete. Our data on when employed (if they joined or left Roche) is manually added, so may not be exact.



Snapshot of whether internal or external last touched each line of code

Exclude 'generated' code:

- `man/`
- `misc/`
- `inst/`
- `data/`
- `pkgdown/`

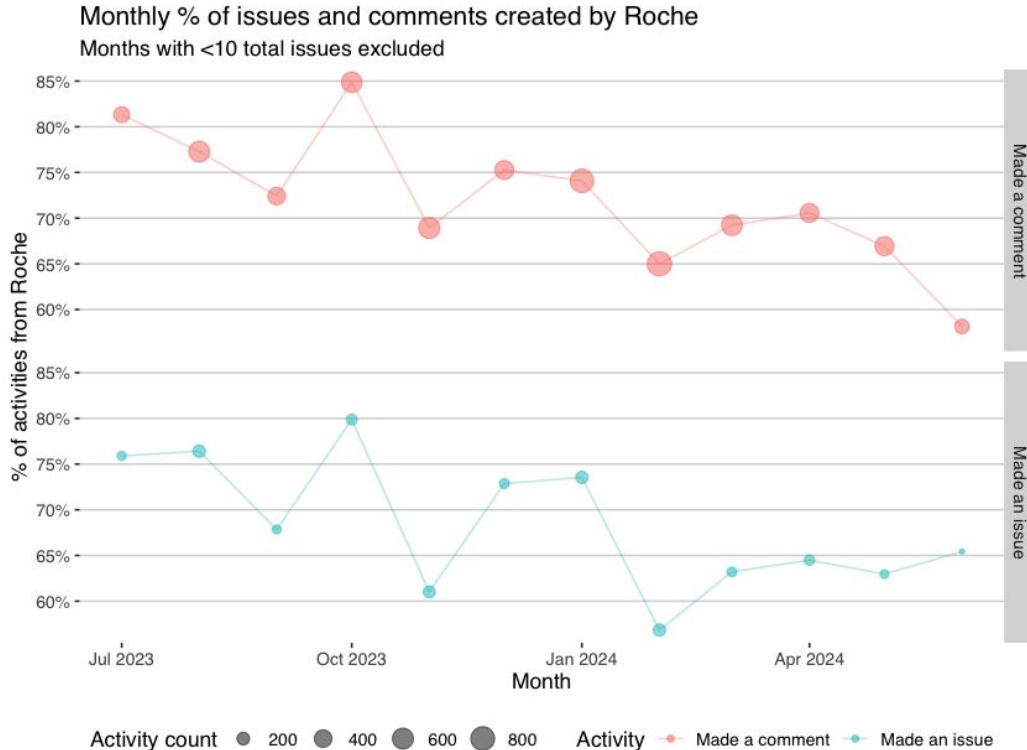
Process (automated via for loop):

1. Clone each repo into temporary directory
2. Run `git ls-files -- '!:man/*' '!:misc/*' '!:inst/*' '!:data/*' '!:pkgdown/*' | while read f; do git blame -w --line-porcelain -- "$f" | grep -I '^author-mail'; done | sort -f | uniq -ic | sort -n`
3. Read results into R



Getting to know the contributions from **issues**

Are others in engaging in discussions around our tools?



Data can be pulled via  (GitStats)

Issue creation can be a proxy for others using, or exploring our tools.

Difficult to figure out the scope of an issue, as tags not used consistently.



So what can we easily pull?

Easily attainable insights



We know how people are **contributing code**, or **engaging in discussions** on each package.



The GitStats package makes it trivial to pull data out of github.com and gitlab.com



We can state objectively how much internal and external involvement we have on each of our critical open source R packages.



Limitations on this data

Limitations

- **Accuracy**
 - You must be able to define your internal contributions!
 - It's easy to double count external contributors
- **Construct validity**
 - **Commits** range from fixing typos to releasing new features
 - **Blame** assigns you the whole line, even if you edited a character
 - A user error **issue** is treated the same as a great feature proposal
- **Content validity**
 - Contributions go beyond Github interactions, so this is one piece of the puzzle!



An eco-system of tools to get into this data

Tools available



GitStats R package [This talk]
r-world-devs.github.io/GitStats



GrimoireLab for collecting and curating data
chaoss.github.io/grimoirelab
App built on framework: cauldron.io

Exploring external contributions to the R codebase used by Roche to design and analyse late-stage clinical trials

AUTHOR
James Black, PhD

AFFILIATION
Data & Statistical Sciences, Roche

PUBLISHED
March 12, 2024

ABSTRACT
R is increasingly used in the pharmaceutical industry as the backbone for the pan-study codebase for the design and analysis of clinical trials. In parallel with this shift to R, many companies are open sourcing, and collaborating, on the post-competitive code used across studies. The Pharmaverse and openstatsware are two example initiatives for statistical programming, and biostatistics, respectively.

While numerous benefits come from companies open sourcing their R codebase, from better talent acquisition, to transparency with regulators, activity on git repos provides an insight into the return on investment (ROI) from external contributions to the codebase a company depends on. In this document we explore the ROI as assessed via external contributions to the late-stage codebase at Roche, shedding light on the tangible benefits derived from collaborative development in the pharmaceutical domain.

KEYWORDS
Open Source, Pharmaverse, openstatsware

Table of contents

- 1 Background
- 1.1 Aim
- 2 Quantitative analysis
- 3 Quantitative analysis
- 4 Discussion
- 5 References

Notebooks
Article Notebook

Document is currently a draft

1 Background

In July 2021, Roche stopped development with propriety statistical software, to focus on a new backbone of R packages for the analysis of clinical trials. A 10+-year old codebase written in a propriety language (named *STREAM*) went into maintenance only updates, and development resources were shifted in their entirety to the new R backbone codebase, that comprised *rOAK*, *admiral* (Straub et al. 2023) and *NEST* (NEST 2023), which form the core of the *pharmaverse* (pharmaverse 2023). The design of clinical trials and exploratory data analysis at Roche has a longer history of R use, with packages like *rpact* (Anders Bilgrau and Krøgholt 2023) and *crmPack* (Sabanés Bové et al. 2019) used for many years. This has continued to increase in recent years through initiatives like *openstatsware* (openstatsware 2023), that aim to collaboratively fill software gaps in clinical trial design as open source software.

1.1 Aim

In this document we explore the ROI from the perspective of an organisation, from both qualitative and quantitative assessments, shedding light on the tangible benefits derived from collaborative development in the pharmaceutical domain.

2 Quantitative analysis

Regardless of whether a company open sources its own code, with our industries away from propriety languages we are likely to be both depending on and extending open source software. A core question is then whether there is an added benefit open sourcing our own code, and actively contributing back to projects we use.

Doing now what patients need next