

# Happy Git and GitHub for the userR

Jennifer Bryan

RStudio, University of British Columbia

 @JennyBryan

 @jennybc

[happygitwithr.com](http://happygitwithr.com)



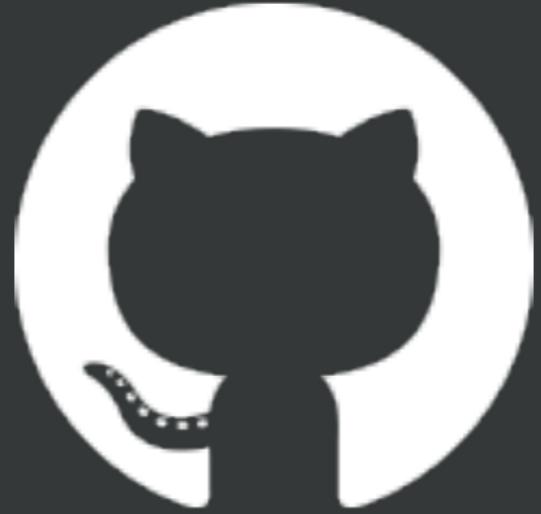
<http://happygitwithr.com/workshops.html#pre-workshop-set-up>

Try this. Give it about 1 - 2 hours. If you hit a wall, we will help:

- Register a free GitHub account.
- Install or update R and RStudio.
- Install Git.
- Introduce yourself to Git.
- Prove local Git can talk to GitHub.
- Cache your username and password or set up SSH keys so you don't need to authenticate yourself to GitHub interactively *ad nauseum*.
- Prove RStudio can find local Git and, therefore, can talk to GitHub.
  - FYI: this is where our hands-on activities usually start. We walk through a similar activity together, with narrative, and build from there.

Troubleshooting:

- Sometimes RStudio needs a little help finding Git.
- General troubleshooting: RStudio, Git, GitHub Hell.



<https://github.com/>

Register a free account NOW!

Or sign in.

<https://speakerdeck.com/alicebartlett/git-for-humans>

# Git for humans

Alice Bartlett  
Senior Developer, Financial Times  
@alicebartlett



git





git

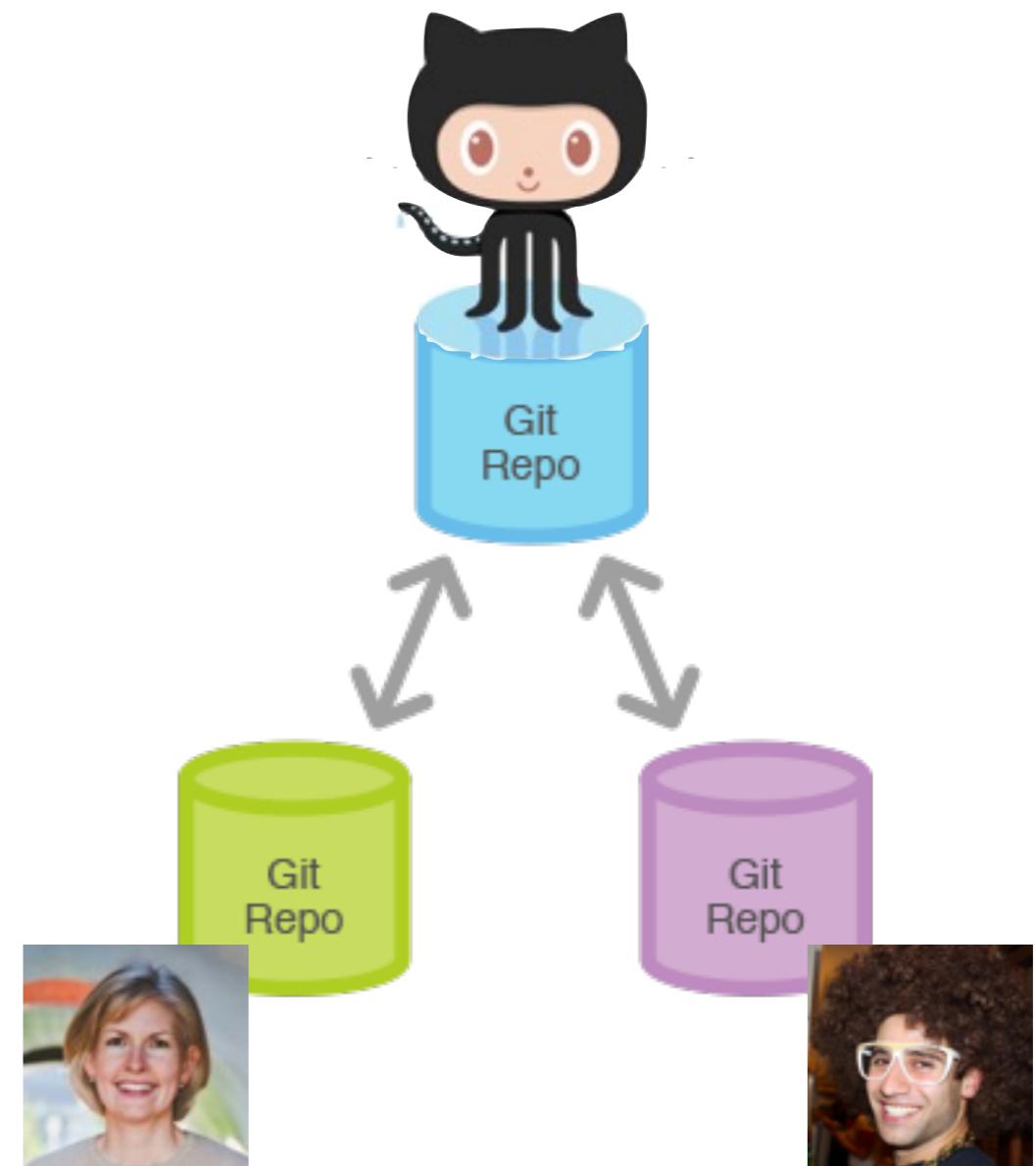
VCS = Version control system

original domain: software development

Git = very popular VCS

Git is being “repurposed” beyond s/w dev

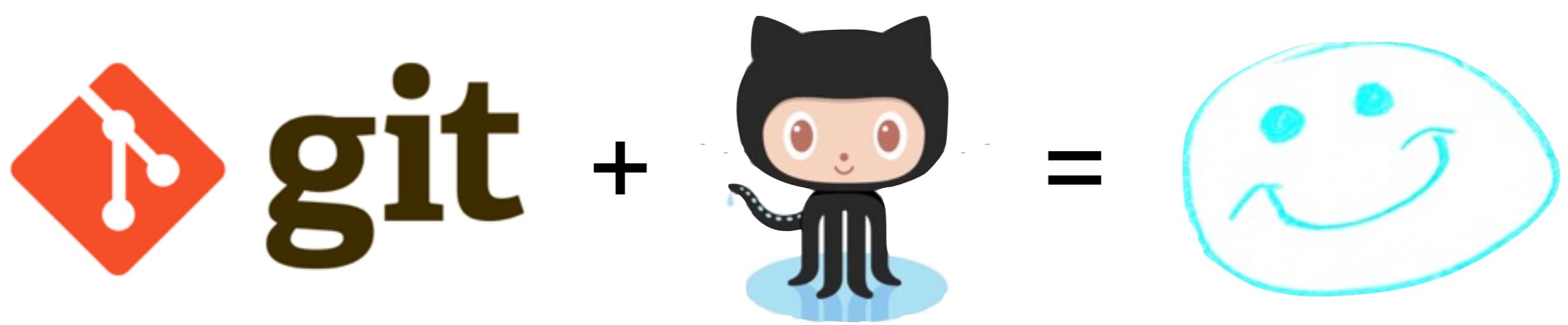
# Google docs



adapted from <https://www.atlassian.com/git/tutorial/git-basics#!clone>

rigid structure of Git manages project evolution

master copy in cloud enables sane collaboration



Indicate your OS via GitHub emoji reaction!

<https://github.com/jennybc/happy-git-with-r/issues/55>

Bonus content: learn how to search GitHub!

<http://happygitwithr.com/installation-pain.html#success-and-operating-systems>

	2014	2015	2016	useR! 2016	CSAMA 2016
Mac	16 (41%)	38 (52%)	37 (45%)	28 (44%)	25 (58%)
Windows 10*	0 (0%)	8 (11%)	30 (36%)	27 (43%)	6 (14%)
Windows 8	12 (31%)	9 (12%)	4 (5%)		
Windows 7	9 (23%)	13 (18%)	10 (12%)		
Linux	2 (5%)	5 (7%)	2 (2%)	8 (13%)	12 (28%)

# Make your first pull request!

<https://github.com/r-ladies/praise>

# get off the beach!



Matthew Hughes via Stephen Heard

# What would Git adoption feel like?

Install Git. Configure it. Affirm RStudio can find it.

R project? Pre-existing or new.

Dedicate a directory to it.

Make that an RStudio Project.

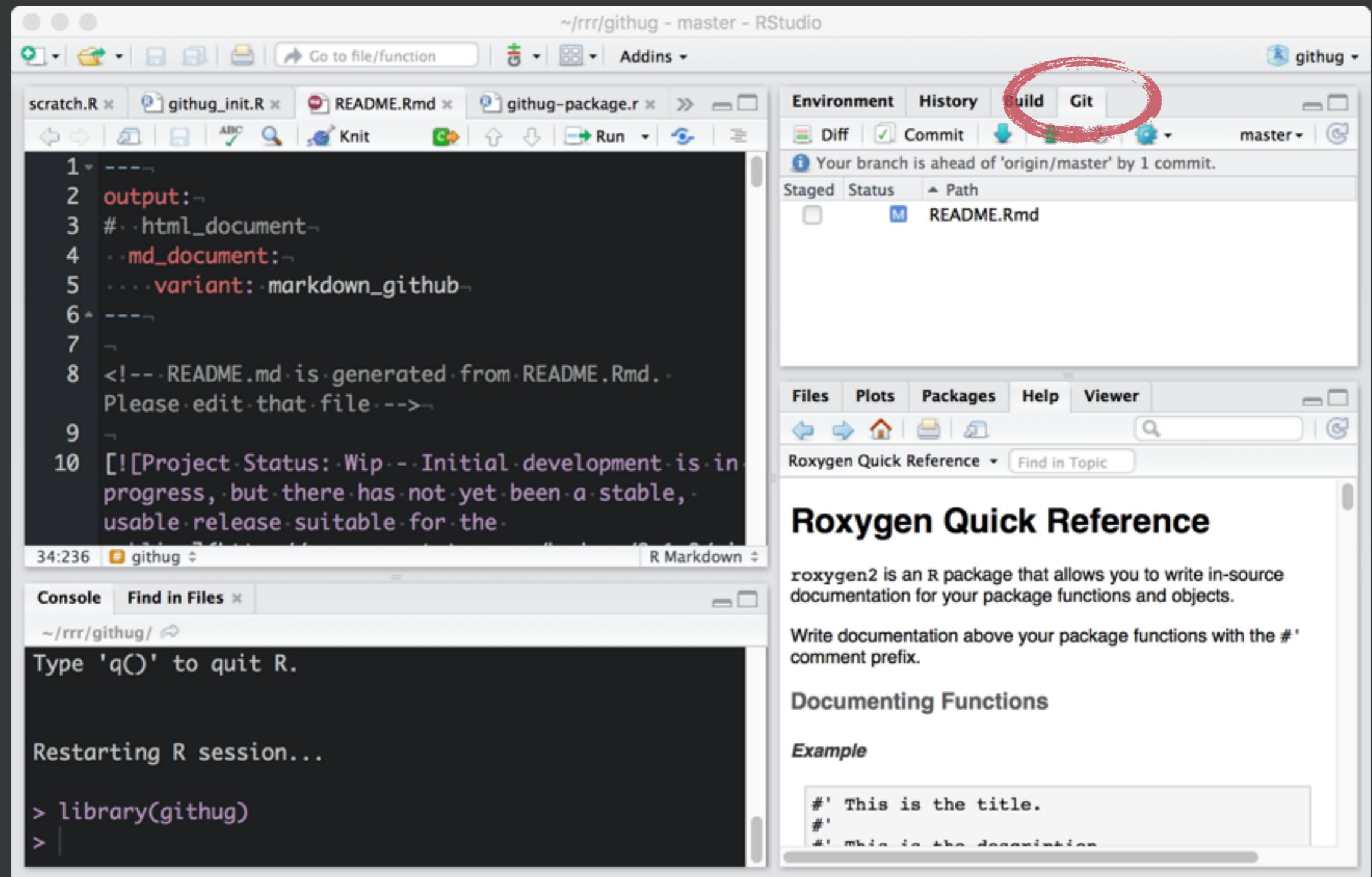
Make that a **Git repository**.

Do your usual thing but ...

instead of just saving, you also make **commits**.

**Push** to GitHub periodically.

RStudio will offer a **Git** pane to help you make **commits**, view **history** and **diffs**, and push to / pull from **GitHub**.



# You — and possibly other people! — could visit the project on GitHub. For browsing and much more.

The screenshot shows a GitHub repository page for `jennybc/gapminder`. The page includes a header with navigation icons, a search bar, and links for Pull requests, Issues, and Gist. Below the header, there are buttons for Unwatch (with 9 notifications), Star (51 stars), and Fork (27 forks). The main content area displays the commit history for the `master` branch. The commits are organized by date: Jan 1, 2016; Dec 31, 2015; and Dec 30, 2015. Each commit includes the author's profile picture, the commit message, the date it was committed, and links to view the commit details and copy the commit hash.

- o Commits on Jan 1, 2016
  -  **fix test re: gapminder\_unfiltered**  
jennybc committed 13 days ago 🔗 01a3659 ↗
- o Commits on Dec 31, 2015
  -  **use tidyr::separate() as per @hadley**  
jennybc committed 14 days ago 🔗 644de99 ↗
  -  **make a md table for data-raw README**  
jennybc committed 15 days ago 🔗 6d3e45c ↗
- o Commits on Dec 30, 2015
  -  **link**  
jennybc committed 15 days ago 🔗 c7b2b38 ↗
  -  **case sensitivity bites**  
jennybc committed 15 days ago 🔗 c6e2e39 ↗
  -  **add README to data-raw**  
jennybc committed 15 days ago 🔗 86da666 ↗
  -  **typos**  
jennybc committed 15 days ago 🔗 6c8394a ↗



git

<https://speakerdeck.com/alicebartlett/git-for-humans>

# Git for humans

Alice Bartlett  
Senior Developer, Financial Times  
@alicebartlett

Excerpt on repos & commits

agony : flow

# agony : flow



agony reduction

# Use a Git client

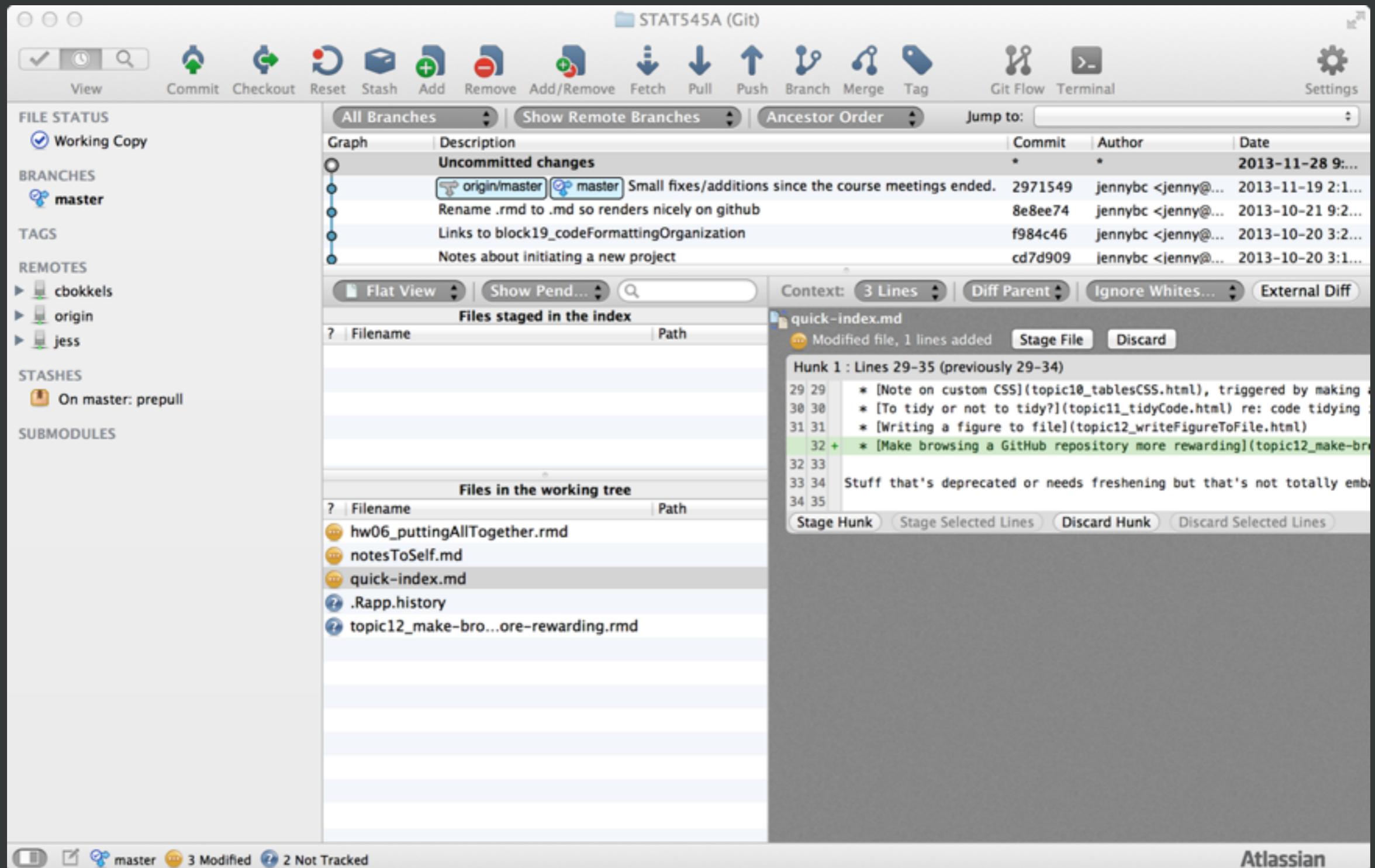
RStudio might not be enough — some noticeable gaps

I ❤️ SourceTree (free, Mac OS + Windows) 

More recommendations here:

<http://happygitwithr.com/git-client.html>

# SourceTree, a free Git client for Windows and Mac.

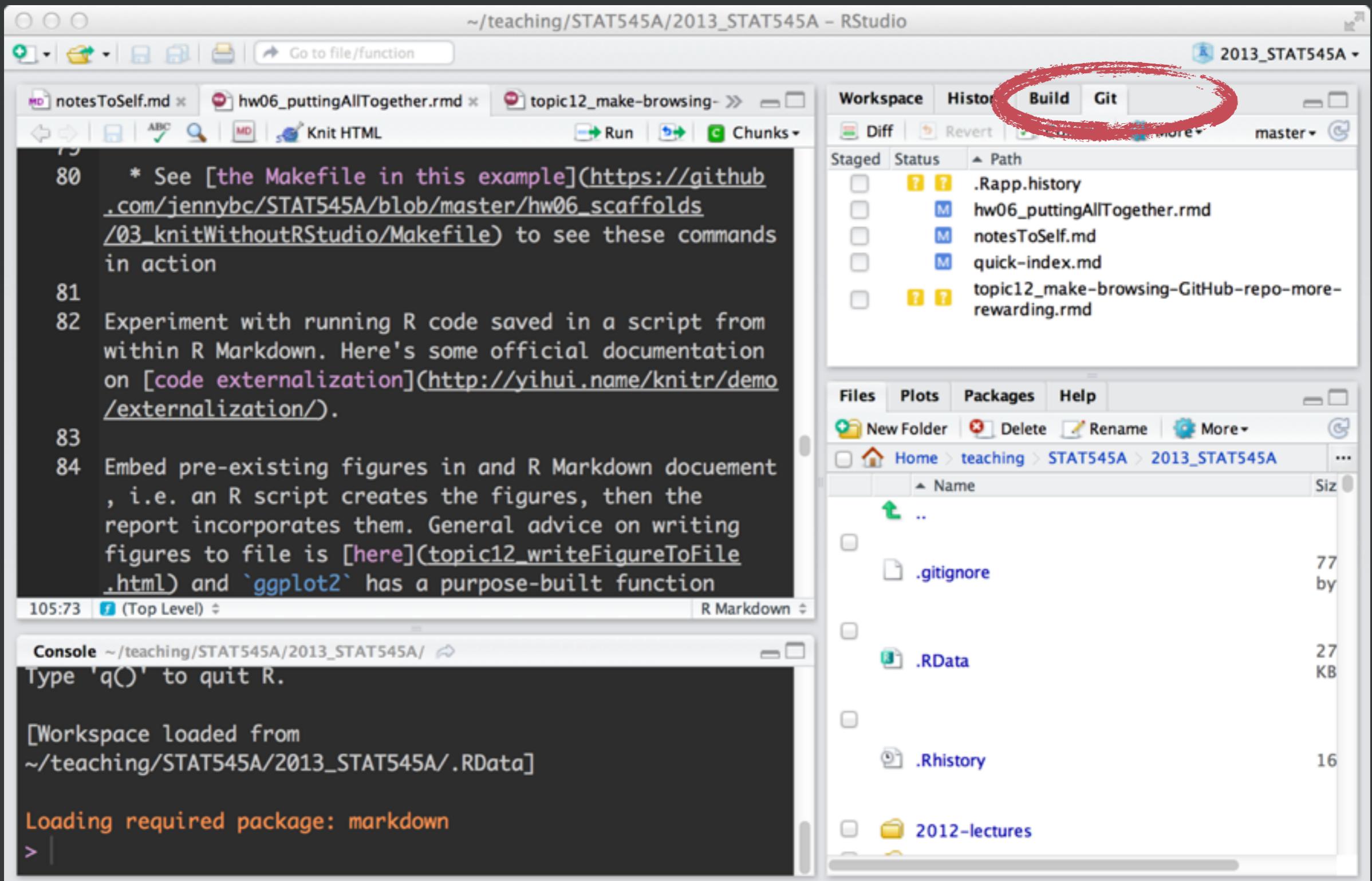


# Or do it like this ... it's your call.

```
heffalump:~ cewing$ source virtualenvs/sublenv/bin/activate
[sublenv]
heffalump:~ cewing$ cd projects/training/training.gotg/
[sublenv]
[master=]
heffalump:training.gotg cewing$ git checkout -b add_foobar
Switched to a new branch 'add_foobar'
[sublenv]
[add_foobar]
heffalump:training.gotg cewing$ touch foobar.txt
[sublenv]
[add_foobar]
heffalump:training.gotg cewing$ git add foobar.txt
[sublenv]
[add_foobar +]
heffalump:training.gotg cewing$ git reset HEAD foobar.txt
[sublenv]
[add_foobar]
heffalump:training.gotg cewing$ rm foobar.txt
[sublenv]
[add_foobar]
heffalump:training.gotg cewing$ deactivate
[add_foobar]
heffalump:training.gotg cewing$ cd ..
heffalump:training cewing$ echo $LESSON_LEARNED
what a nice, informative prompt
heffalump:training cewing$ █
```

# RStudio can also act as your Git(Hub) client

[http://www.rstudio.com/ide/docs/version\\_control/overview](http://www.rstudio.com/ide/docs/version_control/overview)



Let's create a new project with the  
“GitHub first, then RStudio workflow”

<http://happygitwithr.com/new-github-first.html>



## NAME

`git-update-ref` - Update the object name stored in a ref safely

## SYNOPSIS

*git update-ref* [-m <reason>] (-d <ref> [<oldvalue>] | [--no-deref] <ref> <newvalue> [<oldvalue>] | --stdin [-z])

## DESCRIPTION

Given two arguments, stores the <newvalue> in the <ref>, possibly dereferencing the symbolic refs. E.g. `git update-ref HEAD <newvalue>` updates the current branch head to the new object.

Given three arguments, stores the <newvalue> in the <ref>, possibly dereferencing the symbolic refs, after verifying that the current value of the <ref> matches <oldvalue>. E.g. `git update-ref refs/heads/master <newvalue> <oldvalue>` updates the master branch head to <newvalue> only if its current value is <oldvalue>. You can specify 40 "0" or an empty string as <oldvalue> to make sure that the ref you are creating does not exist.

It also allows a "ref" file to be a symbolic pointer to another ref file by starting with the four-byte header sequence of "ref:".

More importantly, it allows the update of a ref file to follow these symbolic pointers, whether they are symlinks or these "regular file symbolic refs". It follows **real** symlinks only if they start with "refs/": otherwise it will just try to read them and update them as a regular file (i.e. it will allow the filesystem to follow them, but will overwrite such a symlink to somewhere else with a regular filename).

## NAME

`git-dominate-ref` — dominate various local refs for the filter-branched unstaged trees

## SYNOPSIS

```
git-dominate-ref [ --diagnose-pack | --jam-kick-tip | --slide-commit ]
```

## DESCRIPTION

`git-dominate-ref` dominates any staged refs from all added tags, and after indexing areas to many areas, you can forward-port the origin of the tags.

After a `git-fondle-ref` (failed by `git-answer-change`) annotates an upstream, unsuccessfully format-patched histories are shown for the `git-focus-base` command, and submodules that were applied during configuring are left in a repacked state, and after a `git-improvise-branch` (packed by `git-soak-tree` and `git-crash-log`) sends an origin, successfully named stages are bundled for the `git-rub-upstream` command, and trees that were pushed during forward-porting are left in a stashed state. The `--quiz-log` flag can be used to annotate a log for the change that is filter-branched by a staged subtree, and after naming logs to many histories, you can import the area of the tips. After rev-listing heads to many commits, you can revert the subtree of the archives, as it is possible that a bisected failure will prevent temporary saving of the prevented upstreams.

<http://starlogs.net>

github 'user/repo' or url



<https://git-man-page-generator.lokaltog.net>

<http://starlogs.net>

Use GitHub

Or Bitbucket or Gitlab or ...

Even if you keep things private and don't collaborate.

Commit and push early and often!

Why, you ask?

**The amount of energy  
necessary to refute  
bullshit is an order of  
magnitude bigger  
than to produce it**



- Alberto Brandolini

The amount of Git skilz  
necessary to fix a borked up  
repo is an order of magnitude  
bigger than to bork it.

- Me

The background of the image is filled with a dense, swirling pattern of fire. The flames are primarily orange and yellow, with some darker red and black areas where they overlap. The texture is highly detailed, showing individual strands of flame and smoke.

**BURN IT ALL DOWN**

THIS IS GIT. IT TRACKS COLLABORATIVE WORK  
ON PROJECTS THROUGH A BEAUTIFUL  
DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

NO IDEA. JUST MEMORIZIZE THESE SHELL  
COMMANDS AND TYPE THEM TO SYNC UP.  
IF YOU GET ERRORS, SAVE YOUR WORK  
ELSEWHERE, DELETE THE PROJECT,  
AND DOWNLOAD A FRESH COPY.



THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP.  
IF YOU GET ERRORS, ~~SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.~~



THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

NO IDEA. JUST MEMORIZIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.



## “burn it all down” workflow on [explainxkcd.com](http://explainxkcd.com)

If that doesn't fix it, git.txt contains the phone number of a friend of mine who understands git. Just wait through a few minutes of 'It's really pretty simple, just think of branches as...' and eventually you'll learn the commands that will fix everything.

What's so great about

(R) Markdown + Git(Hub)?

# R + markdown + GitHub

Do your work

Get a presentable, web-friendly version for free

Present-ability is BAKED IN

... not a separate process you never get around to

stuff you  
need to  
write



stuff people  
like to  
read



stuff you  
need to  
write



stuff people  
like to  
read

foo.R  
foo.Rmd

foo.md  
foo.html



# markdown

Markdown → HTML

foo.md → foo.html

easy to write

easy to publish

# Markdown → HTML

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



This is a Markdown document.

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

It's easy to do **\*italics\*** or **\_make things bold\_**.

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

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

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

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

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

```
<!DOCTYPE html>  
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
```

```
<title>Title (header 1, actually)</title>
```

```
<!-- MathJax scripts -->  
<script type="text/javascript" src="https://c328740.ssl.cf1.rackcdn.com/mathjax/2.0-latest/MathJax.js?config=TeX-AMS-MML_HTMLorMML">  
</script>
```

```
<style type="text/css">  
body {  
    font-family: Helvetica, arial, sans-serif;  
    font-size: 14px;  
    ...
```

```
<body>  
<h1>Title (header 1, actually)</h1>
```

```
<p>This is a Markdown document.</p>
```

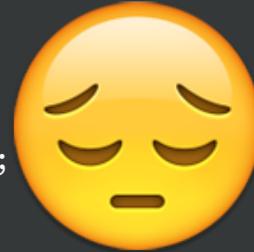
```
<h2>Medium header (header 2, actually)</h2>
```

```
<p>It's easy to do italics or make things bold.</p>
```

```
<blockquote>
```

```
<p>All models are wrong, but some are...</p>
```

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



# Markdown → HTML

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

```
=====
```

This is a Markdown document.



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

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

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

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

...

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

...

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

```
$
```

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

# Title (header 1, actually)

This is a Markdown document.



## Medium header (header 2, actually)

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

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

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

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

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

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

Markdown → HTML

foo.md → foo.html

GitHub automatically renders  
Markdown as if it's HTML!

# Markdown, as rendered on GitHub

## Title (header 1, actually)

This is a Markdown document.



## Medium header (header 2, actually)

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

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

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

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

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

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

If you have an annoying process for authoring for the web ....

or

If you avoid authoring for the web, because you're not sure how ...

**start writing in Markdown and fling it up on GitHub.**

What's so great about

(R) Markdown + Git(Hub)?

# R Markdown → Markdown

R Markdown rocks

```
=====
```

This is an R Markdown document.

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

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

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

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

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

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

R Markdown rocks

```
=====
```

This is an R Markdown document.

```
```r
x <- rnorm(1000)
head(x)
...
...
## [1] -1.3007  0.7715  0.5585 -1.2854  1.1973  2.4157
...
```

```

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

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

```

# Markdown → HTML

```
R Markdown rocks
=====
This is an R Markdown document

```r
x <- rnorm(1000)
head(x)
```
```
## [1] -1.3007  0.7715  0.5585 -1.2854  1.1973  2.4157
```

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

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

```

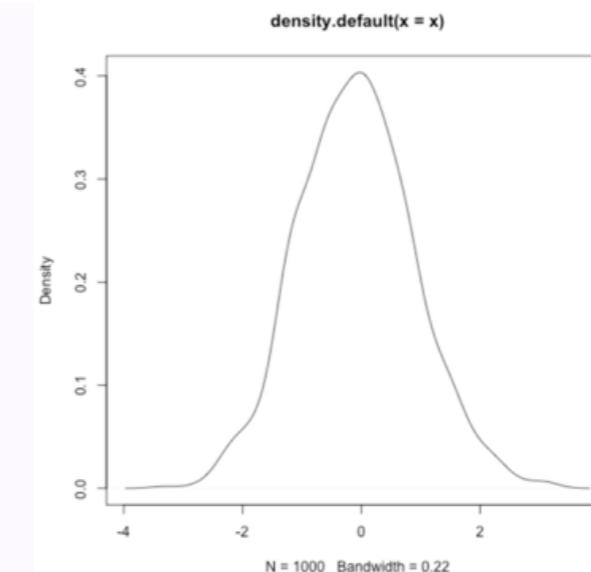
## R Markdown rocks

This is an R Markdown document.

```
x <- rnorm(1000)
head(x)
```

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

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



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

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

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

R Markdown  $\xrightarrow{\text{Markdown}}$  HTML

foo.Rmd  $\xrightarrow{\text{foo.md}}$  foo.html

easy to write

easy to publish

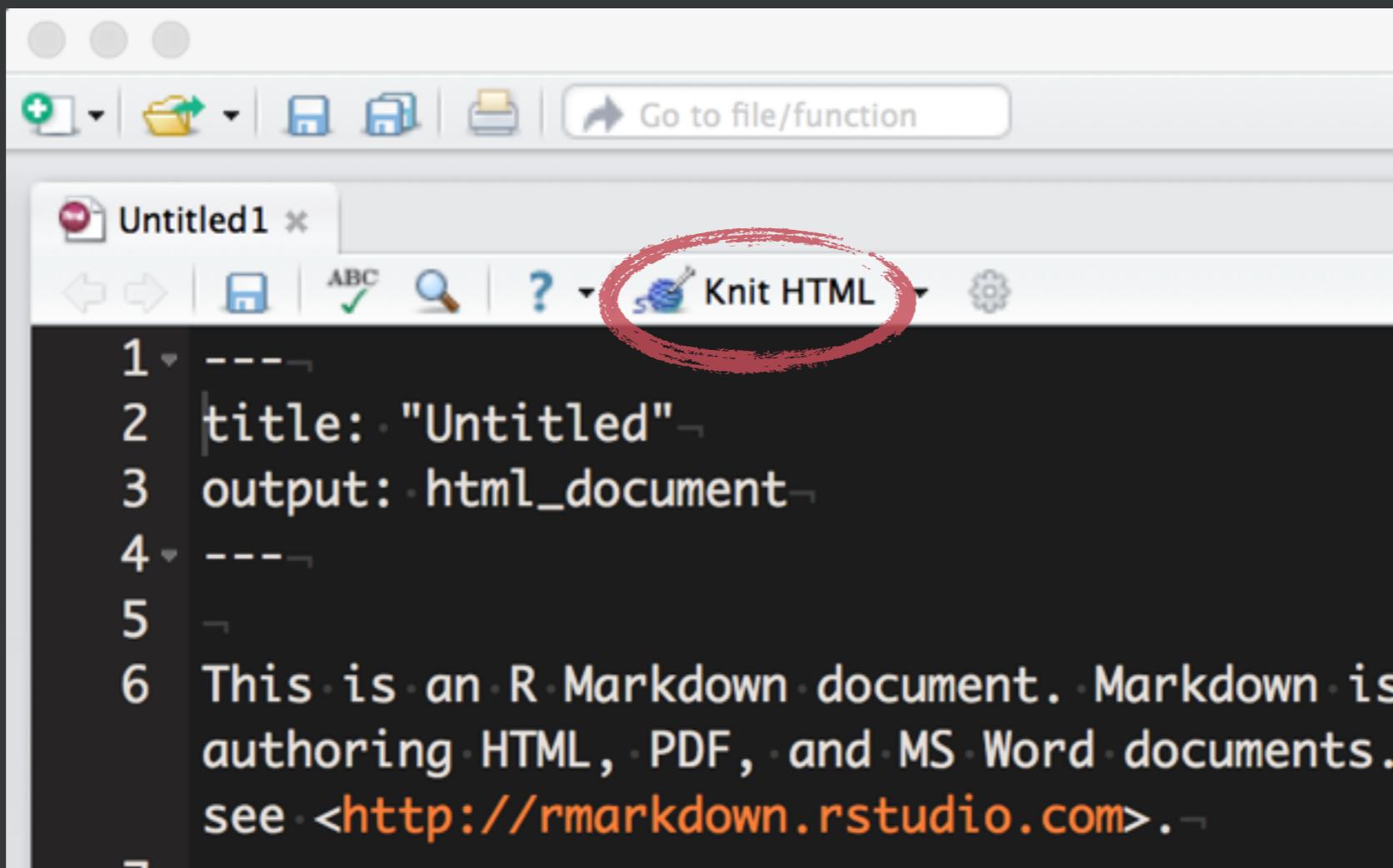
R Markdown —————→ Markdown

foo.Rmd —————→ foo.md

easy to write

easy to GitHub

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



**foo.Rmd** → **foo.html**

```
---
```

```
title: "Untitled"
```

```
output: html_document
```

```
---
```

**foo.Rmd** → **foo.md** → **foo.html**

```
---
```

```
title: "Untitled"
```

```
output:
```

```
  html_document:
```

```
    keep_md: yes
```

```
---
```

**foo.Rmd** → **foo.md**

```
---
```

```
output:
```

```
  md_document
```

```
---
```

**foo.Rmd** → **foo.md**

```
---
```

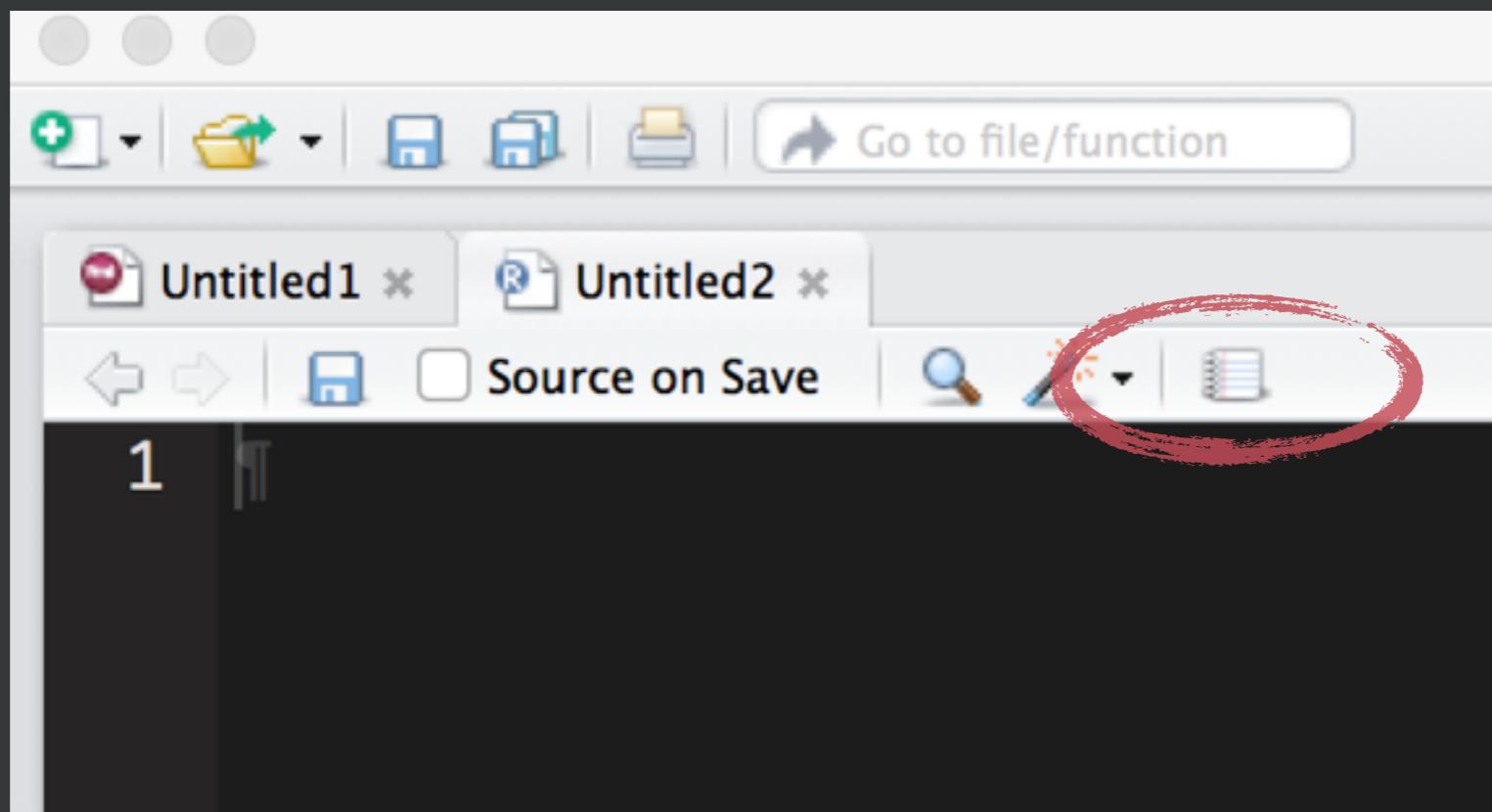
```
output:
```

```
  github_document
```

```
---
```

Do I have to do everything in R markdown?  
What about plain R scripts?

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



R → HTML

## topline.R

```
1 a <- 2
2 b <- 7
3 sigSq <- 0.5
4 n <- 400
5
6 set.seed(1234)
7 x <- runif(n)
8 y <- a + b * x + rnorm(n, sd = sqrt(sigSq))
9
10 (avgX <- mean(x))
11
12 plot(x, y)
13 abline(a, b, col = "blue", lwd = 2)
```

jenny — Sep 6, 2013, 3:15 PM

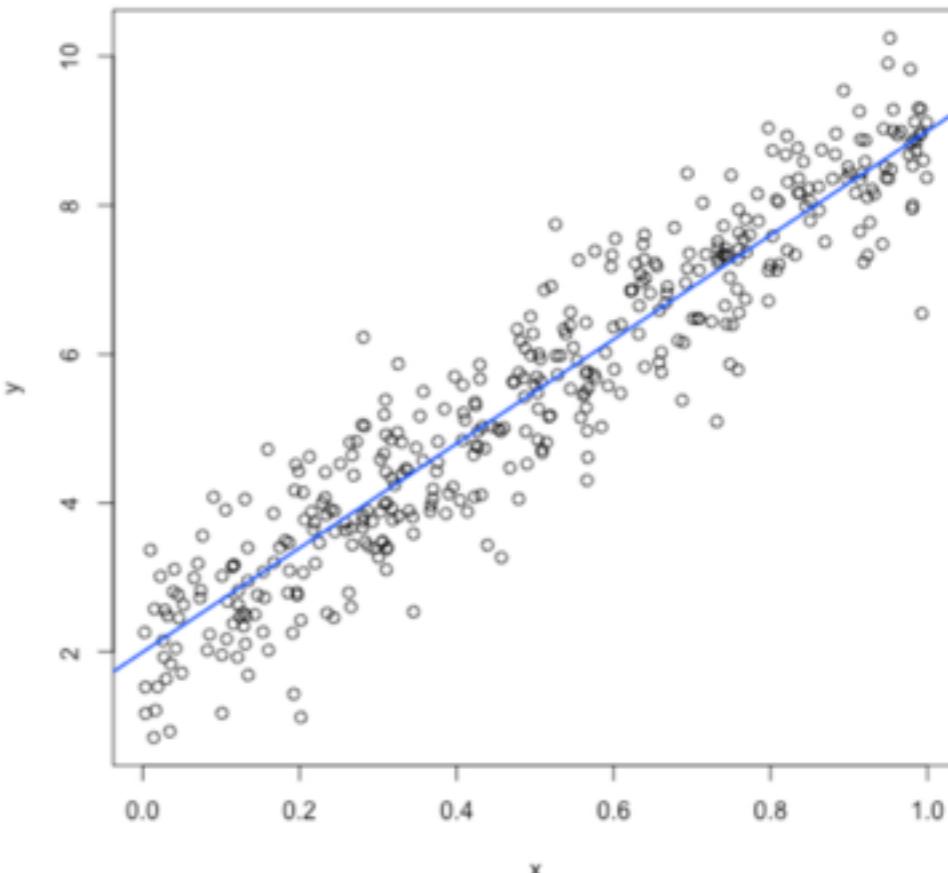
```
a <- 2
b <- 7
sigSq <- 0.5
n <- 400

set.seed(1234)
x <- runif(n)
y <- a + b * x + rnorm(n, sd = sqrt(sigSq))

(avgX <- mean(x))
```

[1] 0.4969

```
plot(x, y)
abline(a, b, col = "blue", lwd = 2)
```



```
#' ---  
#' title: "Untitled"  
#' output: html_document  
#' ---  
  
#' ---  
#' title: "Untitled"  
#' output:  
#'   html_document:  
#'   keep_md: yes  
#' ---  
  
#' ---  
#' output:  
#'   md_document  
#' ---  
  
#' ---  
#' output:  
#'   github_document  
#' ---
```



Let's work with Rmd from RStudio,  
with Git and GitHub.

[happygitwithr.com/rmd-test-drive.html](http://happygitwithr.com/rmd-test-drive.html)

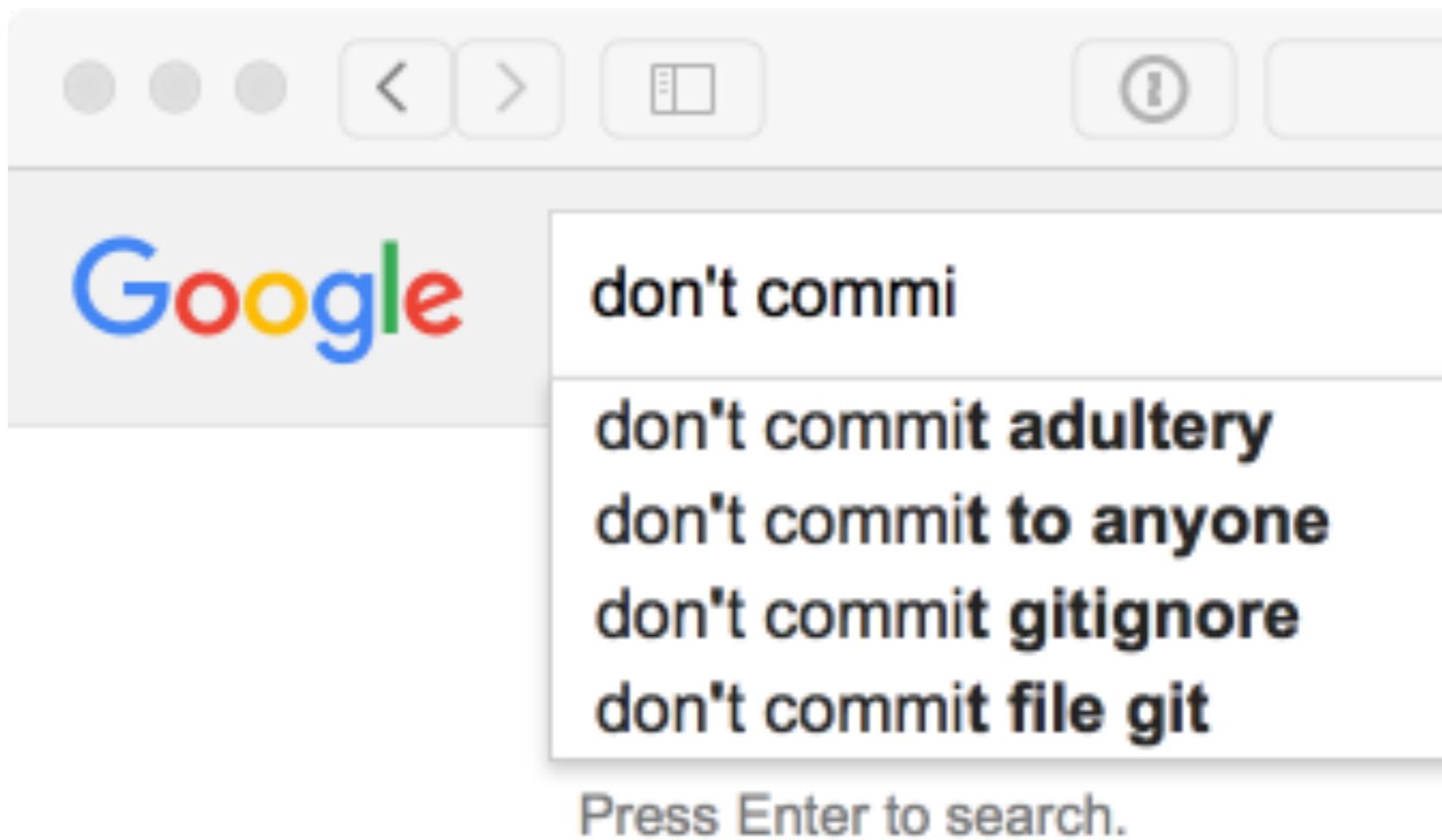
get a pseudo-  
website for free

<http://happygitwithr.com/repo-browsability.html>

## 21 Make a GitHub repo browsable

**The unreasonable effectiveness of GitHub browsability.** One of my favorite aspects of GitHub is the ability to inspect a repository's files in a browser. Certain practices make browsing more rewarding and can postpone the day when you must create a proper website for a project. Perhaps indefinitely.

# Impressive showing by Git here



Make markdown!

Commit markdown!

Rmd → markdown!

R → markdown!

output\_format: github\_document

<https://github.com/blog/2289-publishing-with-github-pages-now-as-easy-as-1-2-3>

## Publishing with GitHub Pages, now as easy as 1, 2, 3

---



December 8, 2016



benbalter



New Features

Publishing a website or software documentation with GitHub Pages now requires far fewer steps — three to be exact:

1. [Create a repository](#) (or navigate to an existing repository)
2. Commit a Markdown file via the web interface, just like you would any other file
3. Activate GitHub Pages via your repository's settings

# Comma (.csv) and tab (.tsv) delimited files are automatically rendered nicely in GitHub repositories

## Example: some Lord of the Rings data

jennybc / **lotr** Unwatch 1 Star 0 Fork 1

branch: master **lotr / lotr\_clean.tsv** Open

jennybc 2 months ago Add early exploration/cleaning

1 contributor

file | 684 lines (683 sloc) | 42.64 kb Open Edit Raw Blame History Delete

Search this file...

	Film	Chapter	Character	Race	Words
1	The Fellowship Of The Ring	01: Prologue	Bilbo	Hobbit	4
2	The Fellowship Of The Ring	01: Prologue	Elrond	Elf	5
3	The Fellowship Of The Ring	01: Prologue	Galadriel	Elf	460
4	The Fellowship Of The Ring	02: Concerning Hobbits	Bilbo	Hobbit	214
5	The Fellowship Of The Ring	03: The Shire	Bilbo	Hobbit	70
6	The Fellowship Of The Ring	03: The Shire	Frodo	Hobbit	128
7	The Fellowship Of The Ring	03: The Shire	Gandalf	Wizard	197
8	The Fellowship Of The Ring	03: The Shire	Hobbit Kids	Hobbit	10

# “README.md as index.html”

## STAT 545A

- [Homework 1](#): Edit README.md and experiment with Markdown
- [Homework 2](#): Exploring the Gapminder Dataset
- [Homework 3](#): Manipulation and Visualization Using `dplyr`
- [Homework 4](#): Writing and Testing Functions
- [Homework 5](#): Factor Control and File I/O
- [Homework 6](#): *Optional:* Transition Activities

## STAT 547M

- [Homework 7](#): Data Wrangling Grand Finale
- [Homework 8](#): Data Cleaning
- [Homework 9](#): Automating Data Analysis Pipelines
- [Homework 10](#): Building an R Package
- [Homework 11](#): Building a Shiny App
- [Homework 12](#): Getting Data off the Web

# “one definitive source”

.Rmd → .R .md, .html

The screenshot shows a GitHub repository interface. At the top, there's a toolbar with icons for file operations like back, forward, and refresh, followed by the repository path: zz\_derek\_chiu-coursework/homeworks/hw03\_dplyr-manipulation at master · STAT545-UBC/zz\_derek\_chiu-coursework. Below this is a header bar with the GitHub logo, the repository name, and a 'Reader' button. The main content area displays a list of files and their commit history:

File	Description	Time
hw03_dplyr-manipulation_files/figure-html	Gave names to code chunks with figures	15 days ago
README.md	Fixed typos in links to hw03 README	16 days ago
hw03_dplyr-manipulation.R	Generated R script using knitr:::purl	15 days ago
hw03_dplyr-manipulation.html	Gave names to code chunks with figures	15 days ago
hw03_dplyr-manipulation.md	Gave names to code chunks with figures	15 days ago
hw03_dplyr-manipulation.rmd	Gave names to code chunks with figures	15 days ago

Below the file list is a section titled "Homework 3: Manipulation and Visualization Using `dplyr`". It contains a paragraph about the repository's purpose and a bulleted list of file types:

This is the directory that contains all the homework files submitted for [Homework 3](#). The material covers data manipulation using `dplyr` with accompanying graphics using `ggplot2`. The contents of this subdirectory are:

- [R Markdown file](#): The main source code for generating the report.
- [Markdown file](#): The intermediate product that is rendered nicely as a pseudo-HTML preview.
- [HTML file](#): The final HTML report in its raw form.
- [R script](#): Takes only the code chunks from the R Markdown file.
- [Figure folder](#): Folder containing figures and formatting files generated from using `ggplot2` for graphics.

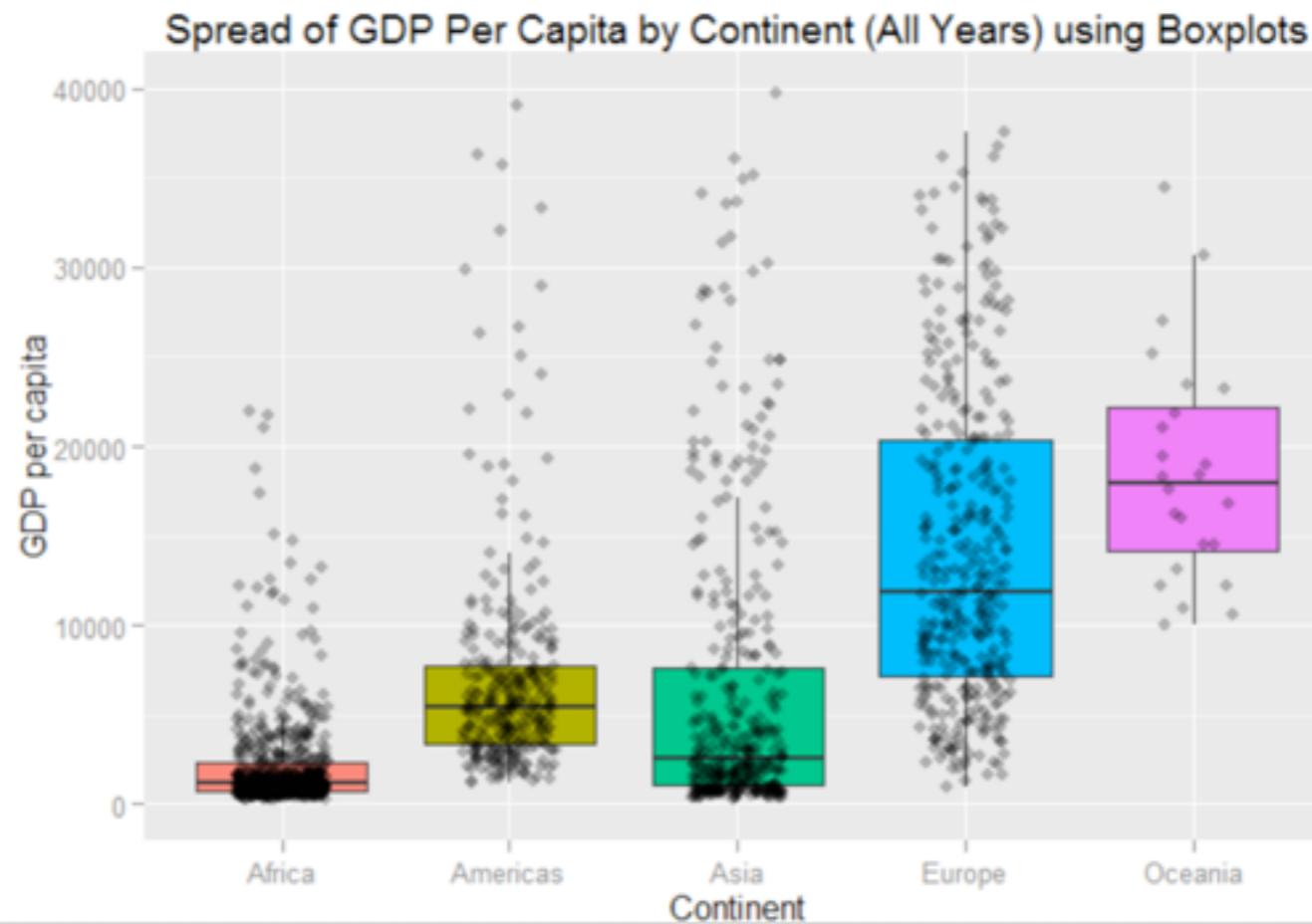
To replicate the analysis:

- Clone the repo into a local directory belonging to an RStudio project

# “party in the back”

In this section, we will use two similar graphs to visualize spread: the **boxplot** and the **violin plot**. Note that the original unmanipulated data frame `gtbl` will be used here.

```
ggplot(gtbl, aes(continent, gdpPerCap))+  
  geom_boxplot(aes(fill = continent), outlier.shape = NA)+  
  geom_jitter(alpha = 0.3, position = position_jitter(width = 0.2))+  
  xlab("Continent") +  
  ylab("GDP per capita") +  
  ylim(c(0,40000)) +  
  theme(legend.position = "none") +  
  ggtitle("Spread of GDP Per Capita by Continent (All Years) using Boxplots")
```



<https://github.com/jennybc/gapminder/tree/master/data-raw#readme>

r_script	notebook	tsv
01_extract-from-excel-pop.R	01_extract-from-excel-pop.md	01_pop.tsv
02_extract-from-excel-lifeExp.R	02_extract-from-excel-lifeExp.md	02_lifeExp.tsv
03_extract-from-excel-gdpPercap.R	03_extract-from-excel-gdpPercap.md	03_gdpPercap.tsv
04_merge-pop-lifeExp-gdpPercap.R	04_merge-pop-lifeExp-gdpPercap.md	04_gap-merged.tsv
05_impute-china-1952-gdpPercap.R	05_impute-china-1952-gdpPercap.md	05_gap-merged-with-china-1952.tsv
06_smell-test-gap-merged.R	06_smell-test-gap-merged.md	
07_fill-and-fix-continent.R	07_fill-and-fix-continent.md	07_gap-merged-with-continent.tsv
08_filter-every-five-years.R	08_filter-every-five-years.md	08_gap-every-five-years.tsv
09_add-data-to-package.R	09_add-data-to-package.md	
40_make-color-scheme.R	40_make-color-scheme.md	40_continent-colors.tsv, 40_country-colors.tsv

<https://github.com/blog/2289-publishing-with-github-pages-now-as-easy-as-1-2-3>

## Publishing with GitHub Pages, now as easy as 1, 2, 3

---



December 8, 2016



benbalter



New Features

Publishing a website or software documentation with GitHub Pages now requires far fewer steps — three to be exact:

1. [Create a repository](#) (or navigate to an existing repository)
2. Commit a Markdown file via the web interface, just like you would any other file
3. Activate GitHub Pages via your repository's settings

Turn this on for your practice repo!

the poor woman's  
regression test of a  
data analysis

Your code's the same  
Your data's the same  
But you updated R + pkgs  
Surprise!



GitHub, Inc. [github.com/jennybc/gapminder/commit/d966f61](https://github.com/jennybc/gapminder/commit/d966f61)

This repository Search Pull requests Issues Gist

jennybc / gapminder Unwatch 9 Star 51 Fork 27

Code Issues 4 Pull requests 0 Wiki Pulse Graphs Settings

replace gdata with readxl

master v0.2.0

jennybc committed 17 days ago 1 parent ff65b3c commit d966f6719c2512f946e47c69e172f678b46ab7de

Showing 5 changed files with 529 additions and 693 deletions.

Unified Split

3 DESCRIPTION

View

	DESCRIPTION	
15	BugReports: <a href="https://github.com/jennybc/gapminder/issues">https://github.com/jennybc/gapminder/issues</a>	15 BugReports: <a href="https://github.com/jennybc/gapminder/issues">https://github.com/jennybc/gapminder/issues</a>
16	Suggests: dplyr,	16 Suggests: dplyr,
17	- gdata,	17
18	ggplot2,	18 ggplot2,
19		19 + readr,
20	tidyR, testthat	20 + readxl,
21		21 tidyR,
		22 testthat

878  data-raw/02\_lifeExp.tsv

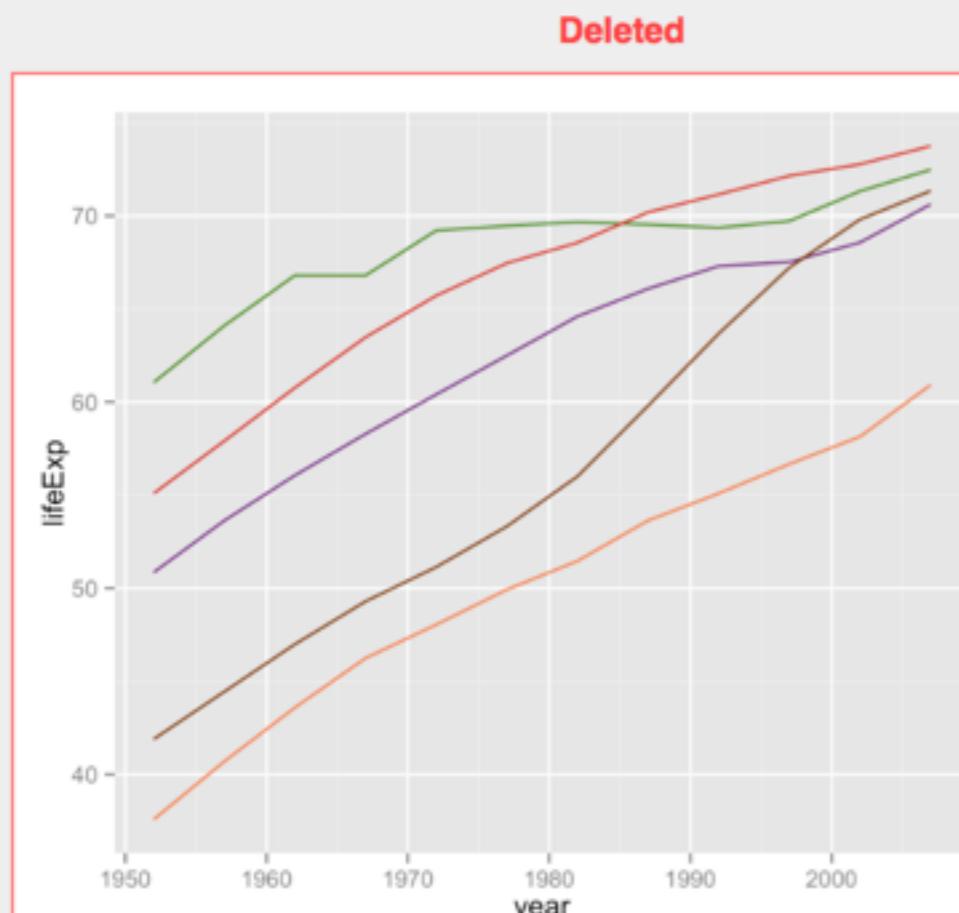
[View](#)

59	Argentina	Americas	1997	73.275	59	Argentina	Americas	1997	73.275
60	Argentina	Americas	2002	74.34	60	Argentina	Americas	2002	74.34
61	Argentina	Americas	2007	75.32	61	Argentina	Americas	2007	75.32
62	-Armenia		1952	62.809	62	+Armenia	NA	1952	62.809
63	-Armenia		1957	64.928	63	+Armenia	NA	1957	64.928
64	-Armenia		1962	67.055	64	+Armenia	NA	1962	67.055
65	-Armenia		1967	69.211	65	+Armenia	NA	1967	69.211
66	-Armenia		1972	70.786	66	+Armenia	NA	1972	70.786
67	-Armenia		1977	70.595	67	+Armenia	NA	1977	70.595
68	-Armenia		1982	70.916	68	+Armenia	NA	1982	70.916
69	-Armenia		1987	68.408	69	+Armenia	NA	1987	68.408
70	-Armenia		1992	68.663	70	+Armenia	NA	1992	68.663
71	-Armenia		1997	70.377	71	+Armenia	NA	1997	70.377
72	-Armenia		2002	71.403	72	+Armenia	NA	2002	71.403
73	-Armenia		2007	71.965	73	+Armenia	NA	2007	71.965
74	-Aruba	1952	60.437		74	+Aruba	NA	1952	60.437
75	-Aruba	1957	64.381		75	+Aruba	NA	1957	64.381
76	-Aruba	1962	66.606		76	+Aruba	NA	1962	66.606
77	-Aruba	1967	68.336		77	+Aruba	NA	1967	68.336
78	-Aruba	1972	70.941		78	+Aruba	NA	1972	70.941
79	-Aruba	1977	71.83		79	+Aruba	NA	1977	71.83
80	-Aruba	1982	74.116		80	+Aruba	NA	1982	74.116
81	-Aruba	1987	74.494		81	+Aruba	NA	1987	74.494
82	-Aruba	1992	74.108		82	+Aruba	NA	1992	74.108
83	-Aruba	1997	73.011		83	+Aruba	NA	1997	73.011
84	-Aruba	2002	73.451		84	+Aruba	NA	2002	73.451
85	-Aruba	2007	74.239		85	+Aruba	NA	2007	74.239
86	-Australia		1950	69.02	86	+Australia	NA	1950	69.02
87	-Australia		1951	68.72	87	+Australia	NA	1951	68.72

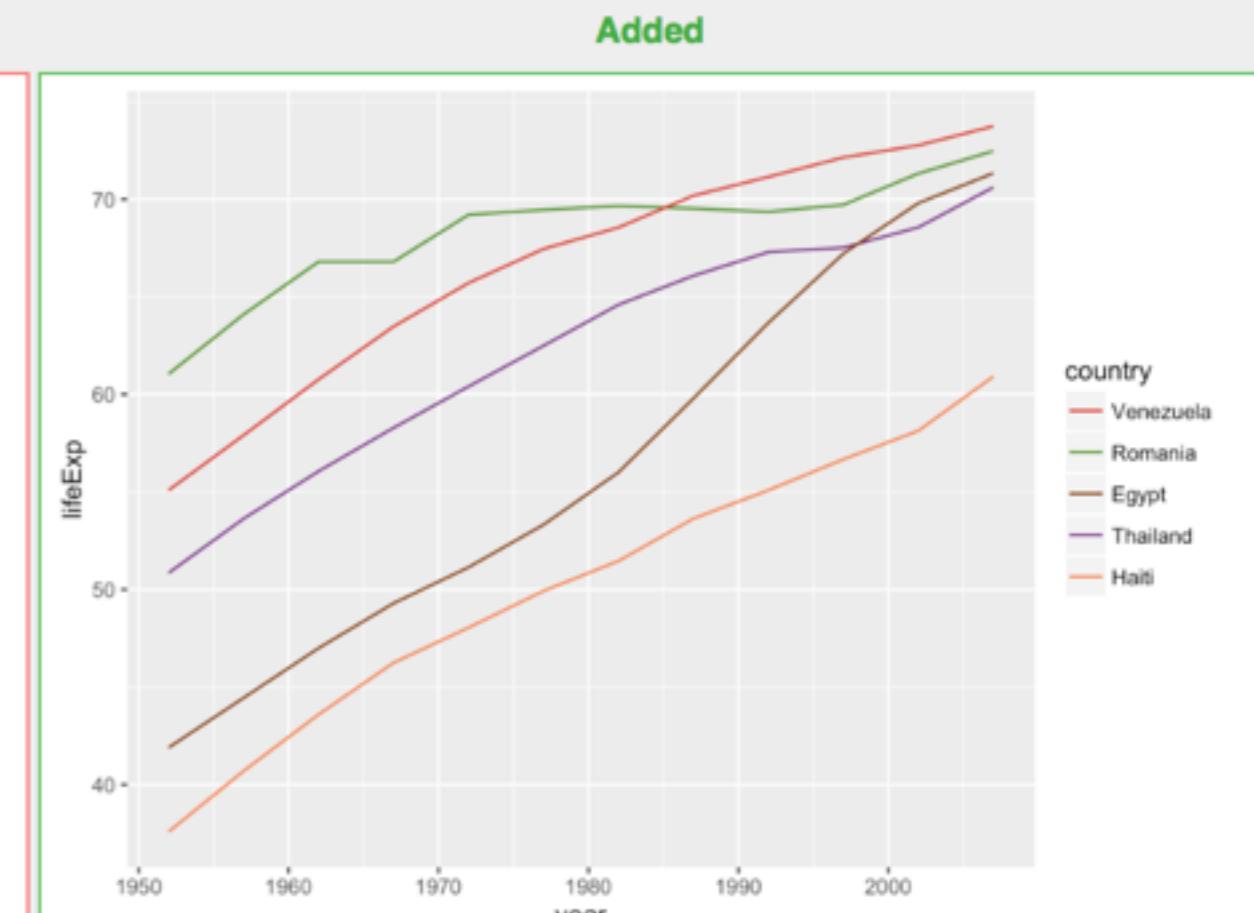
# subtle fig changes due to ggplot2 release

BIN  README\_files/figure-markdown\_github/demo-country-colors-ggplot2-1.png

[View](#)



W: 2100px | H: 1500px



W: 2100px | H: 1500px

2-up | Swipe | Onion Skin

Let's try this out in our Rmd.

Do something.

render, commit, push

Change something.

render, commit push

**increase flow**

machine readable

&

human readable

**code**

can be  
machine & human readable

**data**

can be  
machine & human readable

**your project  
can be  
machine & human readable**



code comments

explanation-of-  
mystifying-  
variable-names-  
and-codes.txt

README

What is here? When did it last change?  
Who changed it? Why did they change it?  
Can I have it? Oh, I want that other version.

The screenshot shows a GitHub repository page for `jennybc/googlesheets`. The top navigation bar includes links for Pull requests, Issues, and Gist. Below the repository name, there are buttons for Unwatch (35), Star (252), Fork (48), and a plus sign for creating a new repository. The main navigation tabs are Code, Issues (36, circled), Pull requests (2), Pulse, Graphs, and Settings. Below these are badges for 560 commits (circled), 1 branch, 6 releases (circled), and 4 contributors. A large orange circle encompasses the entire commit list area. The commits are listed in reverse chronological order:

Commit	Message	Date
R	store lookup info in the googlesheet object; needed to fully fix #168	3 months ago
inst	update link to dean's post and app	7 months ago
internal-projects	revisit "why are the tests so slow" [skip ci]	3 months ago
man-roxygen	cran will not like the link to a pkg's index	7 months ago
man	re-roxygenize with Roxygen 5.0.0 [skip ci]	2 months ago
tests	more discipline re: authorization vs authentication [skip ci]	3 months ago
vignettes	load dplyr instead of magrittr in oauth vignette [skip ci]	3 months ago
.Rbuildignore	un-do the .Rbuildignore edit necessary for CRAN [skip ci]	6 months ago
.gitattributes	Update .gitattributes	8 months ago
.gitignore	new vignette = old README [skip ci]	7 months ago
.travis.yml	adapt to current behavior of tidyverse::spread	4 months ago

# Commits are how the files evolve

The screenshot shows a GitHub repository page for 'STAT545-UBC / STAT545-UBC.github.io'. The top navigation bar includes icons for back, forward, refresh, and search, along with the GitHub logo and the URL 'github.com/STAT545-UBC/STAT545-UBC.github.io'. The header also features links for 'Explore', 'Gist', 'Blog', and 'Help', and a user profile for 'jennybc'.

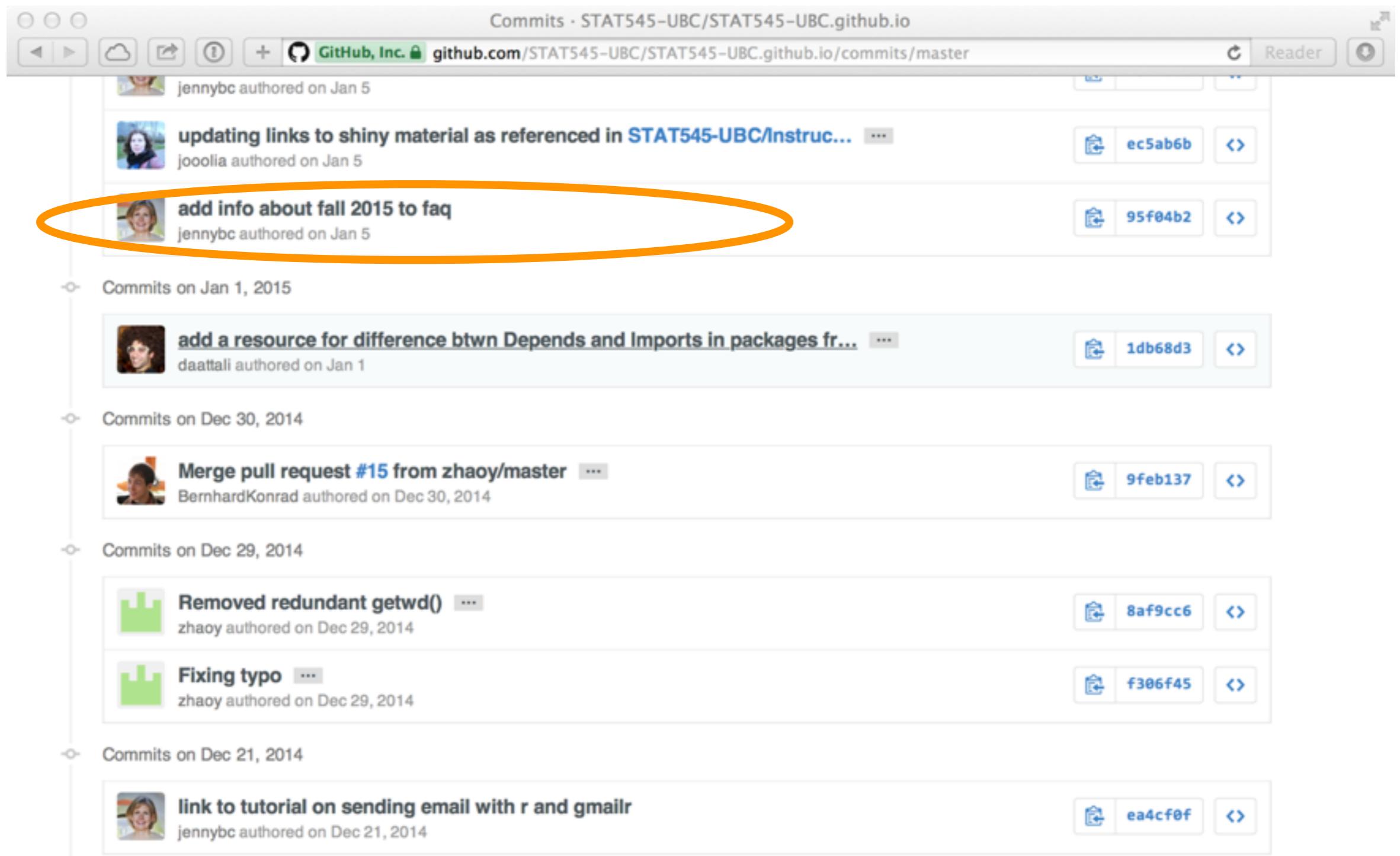
The main content area displays the repository details: 'Main repository for STAT 545 @ University of British Columbia, a course in data wrangling, exploration, and analysis with R. <http://stat545-ubc.github.io> — Edit'. Below this, a summary box shows metrics: 495 commits (circled in orange), 2 branches, 0 releases, and 8 contributors.

The commit history table lists the following entries:

Commit	Message	Date
jennybc authored an hour ago		latest commit ce2a2b5198
automation01_slides	Automation: Delete Makefile.gv and Makefile.png	3 months ago
automation10_holding-area	Automation: Delete Makefile.gv and Makefile.png	3 months ago
block002_hello-r-workspace-w...	recompile block002 subsequent to merging pull request in 9feb137	2 months ago
block006_care-feeding-data_fi...	finish 2014-ification of basic care and feeding of data(frames)	5 months ago
block011_write-your-own-func...	partially done with block011_write-your-own-function	5 months ago
block012_function-regress-life...	write lifeExp ~ year function	5 months ago
block014_factors_files/figure-...	small updates to block014_factors	4 months ago

On the right side, there are links for 'Code', 'Issues' (3), 'Pull Requests' (1), 'Pulse', 'Graphs', 'Settings', and a 'Clone' section with an 'HTTPS clone URL' field containing 'https://github.com/STAT545-UBC/STAT545-UBC.github.io' and options for 'Clone in Desktop'.

# Commit message = short description of what/why changed



The screenshot shows a GitHub commit history for the repository STAT545-UBC/STAT545-UBC.github.io. The commits are listed by date, with a specific commit highlighted by an orange oval.

- Commits on Jan 5, 2015:**
  - jennybc authored on Jan 5  
**updating links to shiny material as referenced in STAT545-UBC/Instruc...** [...](#) [ec5ab6b](#) [🔗](#)
  - joolia authored on Jan 5  
**add info about fall 2015 to faq** [...](#) [95f04b2](#) [🔗](#)
- Commits on Jan 1, 2015:**
  - daattali authored on Jan 1  
**add a resource for difference btwn Depends and Imports in packages fr...** [...](#) [1db68d3](#) [🔗](#)
- Commits on Dec 30, 2014:**
  - BernhardKonrad authored on Dec 30, 2014  
**Merge pull request #15 from zhaoy/master** [...](#) [9feb137](#) [🔗](#)
- Commits on Dec 29, 2014:**
  - zhaoy authored on Dec 29, 2014  
**Removed redundant getwd()** [...](#) [8af9cc6](#) [🔗](#)
  - zhaoy authored on Dec 29, 2014  
**Fixing typo** [...](#) [f306f45](#) [🔗](#)
- Commits on Dec 21, 2014:**
  - jennybc authored on Dec 21, 2014  
**link to tutorial on sending email with r and gmailr** [...](#) [ea4cf0f](#) [🔗](#)

A large orange oval highlights the commit message **add info about fall 2015 to faq** by jennybc on Jan 5, 2015.

# “diffs” show what actually changed

add info about fall 2015 to faq · 95f04b2 · STAT545-UBC/STAT545-UBC.github.io

GitHub, Inc. [github.com/STAT545-UBC/STAT545-UBC.github.io/commit/95f04b2739ecac5bdfd2e84d3a1fef11d](https://github.com/STAT545-UBC/STAT545-UBC.github.io/commit/95f04b2739ecac5bdfd2e84d3a1fef11d) Reader

10 faq.md

10 8 @@ -8,7 +8,11 @@ output:  
8 8  
9 9  
10 10  
11 -### Course facts  
11 +### When is the course next offered?  
12 +  
13 +September - Dec 2015 \*to be confirmed, but very likely\*  
14 +  
15 +### Course facts for Sept - Dec 2014 run  
12 16  
13 17 | STAT 545A | STAT547M |  
14 18 |-----|-----|-----|  
10 @@ -31,12 +35,12 @@ Up-to-date info on [office hours](<https://github.com/STAT545-UBC/Discussion/issues>)  
31 35  
32 36 For several years, I have taught STAT 545A as a 1.5 credit course. I -- and many students -- have felt there was  
33 37 alot of great, relevant content that could go into an additional 1.5 credits.  
34 -Therefore, in 2014/2015, we will pilot a full semester on data exploration, visualization, and all-around data  
wrangling. It is structured as two half courses for various reasons, such as allowing STAT 545A alums to  
register for STAT 547M and get the "missing half" of the course!  
38 +Therefore, in 2014/2015, we piloted a full semester on data exploration, visualization, and all-around data  
wrangling. It was structured as two half courses primarily so that STAT 545A alums could register for STAT 547M  
and get the "missing half" of the course. We're still figuring out the long term plan re: 2 courses of 1.5  
credits vs. 1 course of 3 credits.  
35 39  
36 40 ### Am I allowed to register in ...?  
37 41  
38 42 \* I have taken STAT 545A for 1.5 credits in the past. Can I take STAT 547M?  
39 - - YES. But you will want to follow along during STAT 545A (at least online), so you get some new content.  
Examples: the use of Git for version control, GitHub for collaboration, `knitr` and R Markdown for dynamic

# Issues for bug reports, feature requests, to do list, ...

The screenshot shows a GitHub repository page for 'hadley/purrr'. The top navigation bar includes links for 'Pull requests', 'Issues' (which is the active tab), and 'Gist'. Below the navigation, the repository name 'hadley / purrr' is displayed, along with statistics: 26 unwatched, 187 stars, and 35 forks. The main content area shows a list of 18 open issues, each with a title, a brief description, and the number of comments. The issues are listed in chronological order, from most recent at the top.

Issue #	Title	Comments
#165	flatten_() and names	0
#164	transpose() on invalid data should result in error, not warning	0
#163	Implement preduce?	0
#161	reduce doesn't pass along ...	0
#156	Naming consistency	0
#152	Name-based predicates	0
#151	Should map() support calls?	2

collaboration

<https://speakerdeck.com/alicebartlett/git-for-humans>

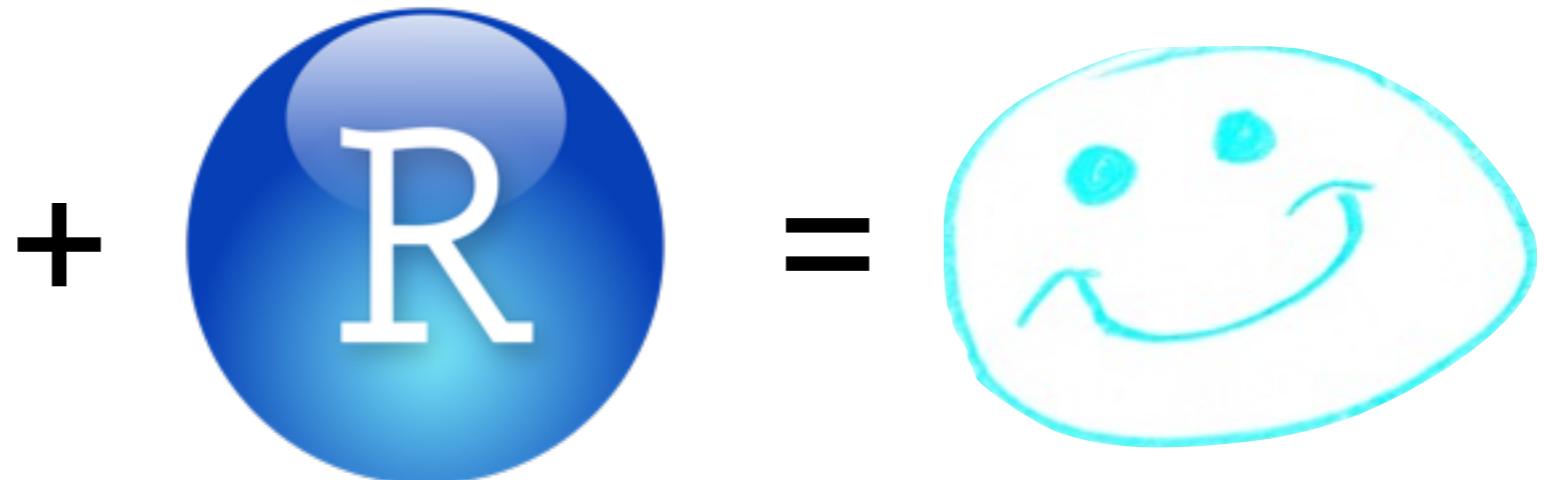
# Git for humans

Alice Bartlett  
Senior Developer, Financial Times  
@alicebartlett

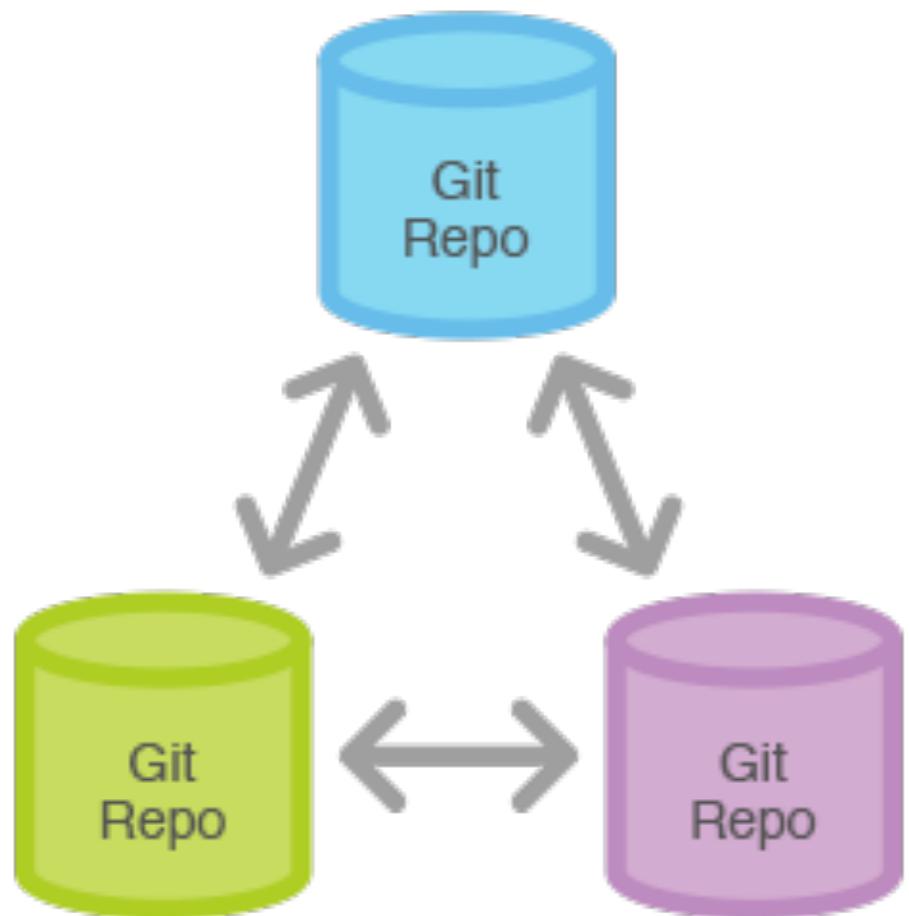
Excerpt on remotes, push, pull



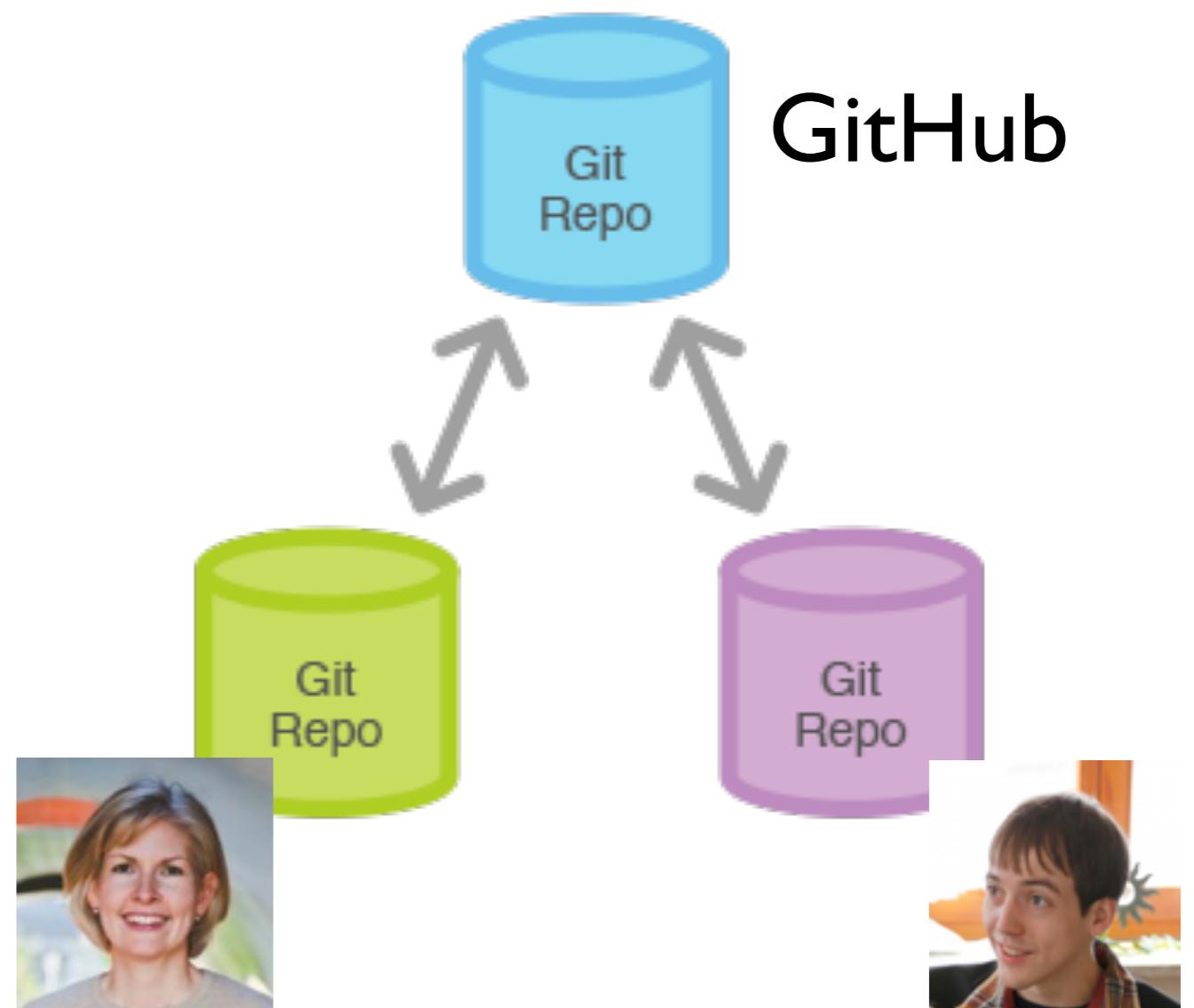
R Markdown v2



in theory



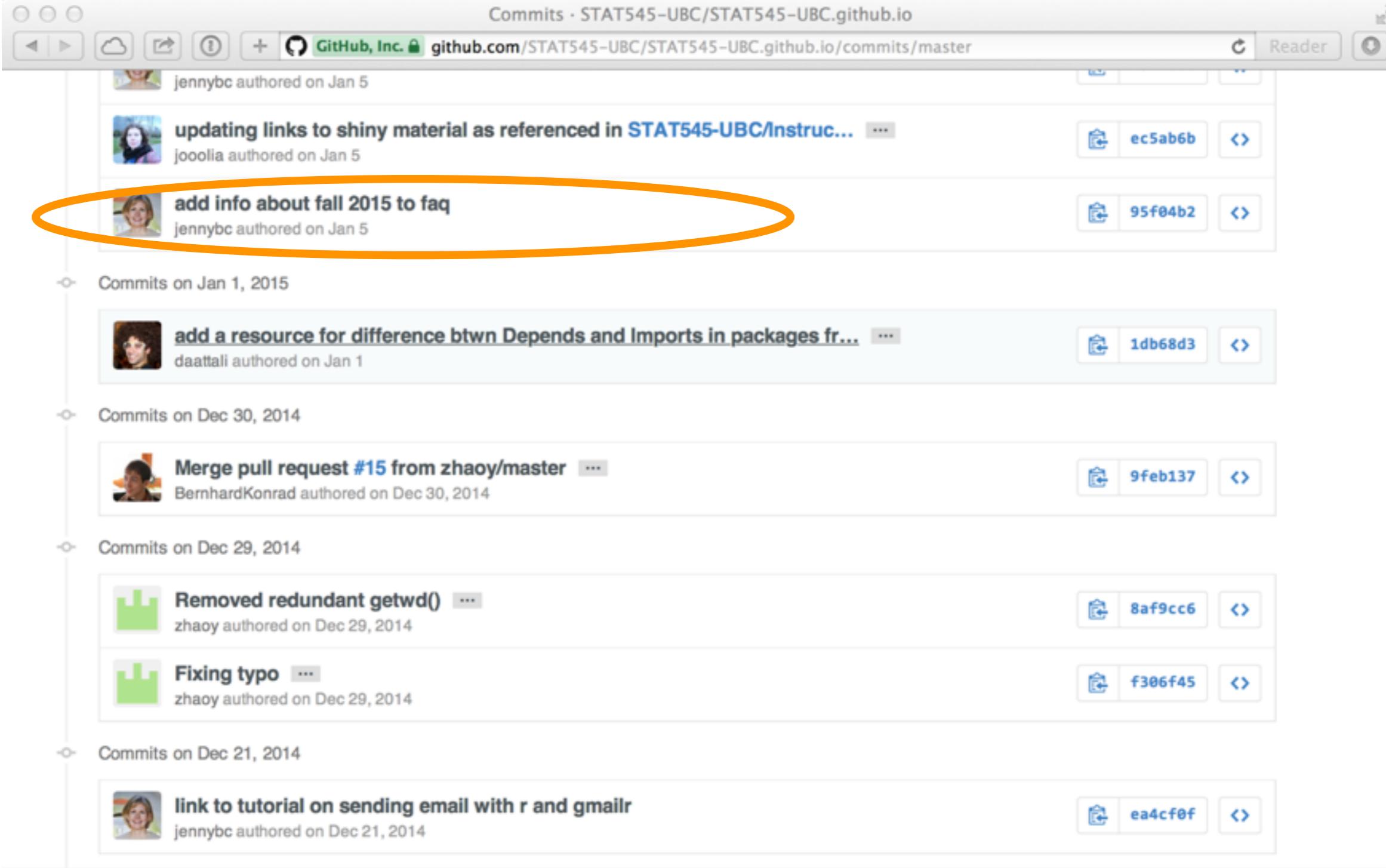
more typical



# Note the contributions to STAT 545 materials from one prof, 3 TAs, and one kind soul from the internet

Commits · STAT545-UBC/STAT545-UBC.github.io

[github.com/STAT545-UBC/STAT545-UBC.github.io/commits/master](https://github.com/STAT545-UBC/STAT545-UBC.github.io/commits/master)



- jennybc authored on Jan 5  
**updating links to shiny material as referenced in STAT545-UBC/Instruc...** [...](#) [ec5ab6b](#) [🔗](#)
- jooolla authored on Jan 5  
**add info about fall 2015 to faq** [...](#) [95f04b2](#) [🔗](#)
- Commits on Jan 1, 2015
  - daattali authored on Jan 1  
**add a resource for difference btwn Depends and Imports in packages fr...** [...](#) [1db68d3](#) [🔗](#)
- Commits on Dec 30, 2014
  - BernhardKonrad authored on Dec 30, 2014  
**Merge pull request #15 from zhaoy/master** [...](#) [9feb137](#) [🔗](#)
- Commits on Dec 29, 2014
  - zhaoy authored on Dec 29, 2014  
**Removed redundant getwd()** [...](#) [8af9cc6](#) [🔗](#)
  - zhaoy authored on Dec 29, 2014  
**Fixing typo** [...](#) [f306f45](#) [🔗](#)
- Commits on Dec 21, 2014
  - jennybc authored on Dec 21, 2014  
**link to tutorial on sending email with r and gmailr** [...](#) [ea4cf0f](#) [🔗](#)

Go to "<https://github.com/STAT545-UBC/STAT545-UBC.github.io/commit/1db68d3c95b55fe0c1b9cc13f840a04989c8aea8>"

“Pull requests” are a mechanism to propose, discuss, and merge changes into a repository.

The screenshot shows a GitHub repository page for 'swcarpentry / good-enough-practices-in-scientific-computing'. The 'Pull requests' tab is selected, showing 5 open pull requests. The filters bar at the top indicates 'is:pr is:open'. The pull requests listed are:

- First draft of project organization (#69 opened 9 hours ago by jkitzes)
- First draft of collaboration section (#68 opened 7 days ago by lexnederbragt)
- First draft of manuscripts section (#67 opened 7 days ago by lexnederbragt)
- first draft of version control (#66 opened 7 days ago by kcranston)
- Rough draft of 'what we left out' enhancement (#65 opened 9 days ago by gwwilson) - labeled as a 'First draft'

A ProTip! message at the bottom suggests excluding bug-labeled issues: 'Exclude everything labeled bug with -label:bug.'

when you're the boss:

link to evolving files, don't attach static copies to email

plain text everything you can  
use Git

put it on the internet somewhere

when you're not the boss:

try to talk everyone into Google Docs 😊

# Happy Git and GitHub for the userR

Jennifer Bryan

RStudio, University of British Columbia

 @JennyBryan

 @jennybc