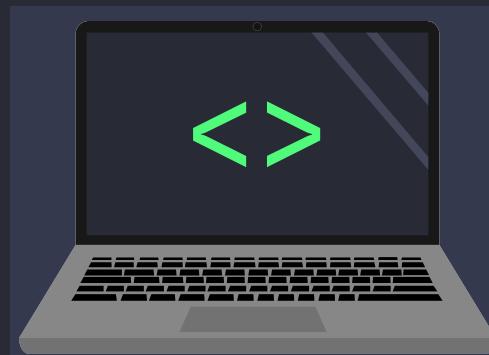


Open science: making data and code open, accessible, and reproducible.



```
```{r, presentation.info, echo = T}
author = "Ryan Eckert"
date = as.Date("04/10/2020", "%m/%d/%y")
cat(paste(author, date, sep = "\n"))
...``
```

Ryan Eckert  
2020-04-10



# Goals.

- We want:
  - to work on our code in a way that is **version controlled** and easily **communicable**

# Goals.

- We want:
  - to work on our code in a way that is **version controlled** and easily **communicable**
  - an effective way to share **data** and **code**

# Goals.

- We want:
  - to work on our code in a way that is version controlled and easily communicable
  - an effective way to share data and code
  - to make our code easily reproducible

# Goals.

- We want:
  - to work on our code in a way that is **version controlled** and easily **communicable**
  - an effective way to share **data** and **code**
  - to make our code easily **reproducible**
  - to **archive** our data and code so that it is always **accessible** and **citable**

# Goals.

- This allows **transparency** in our research and is **helpful** to others who are developing skillsets

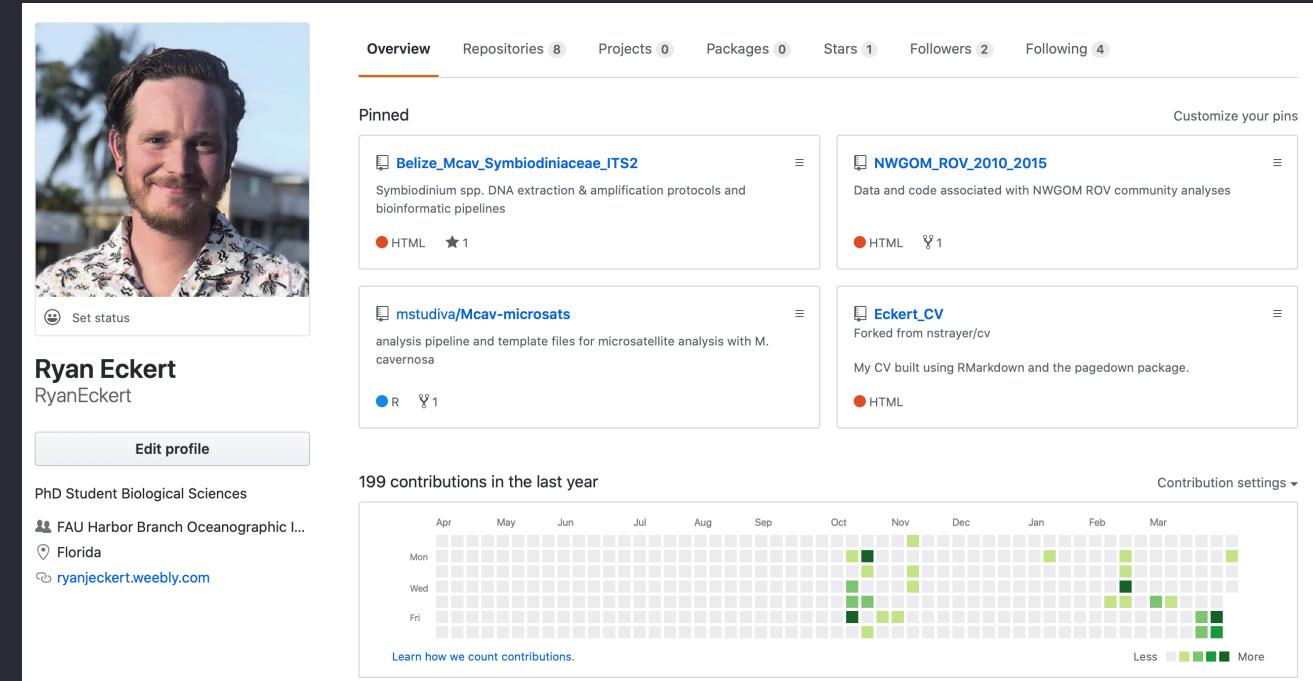
# Goals.

- This allows **transparency** in our research and is **helpful** to others who are developing skillsets
- We can accomplish all these goals effectively using **free** platforms: **R****Markdown**, **GitHub**, and **Zenodo**





- Development platform



A screenshot of a GitHub profile page for a user named Ryan Eckert. The profile picture shows a man with a beard and short hair, wearing a patterned shirt. Below the picture is a "Set status" button. The profile name is "Ryan Eckert" and the handle is "RyanEckert". There is a "Edit profile" button. Underneath the profile information, it says "PhD Student Biological Sciences" and "FAU Harbor Branch Oceanographic I...". It also lists "Florida" and a website link "ryanjeckert.weebly.com".

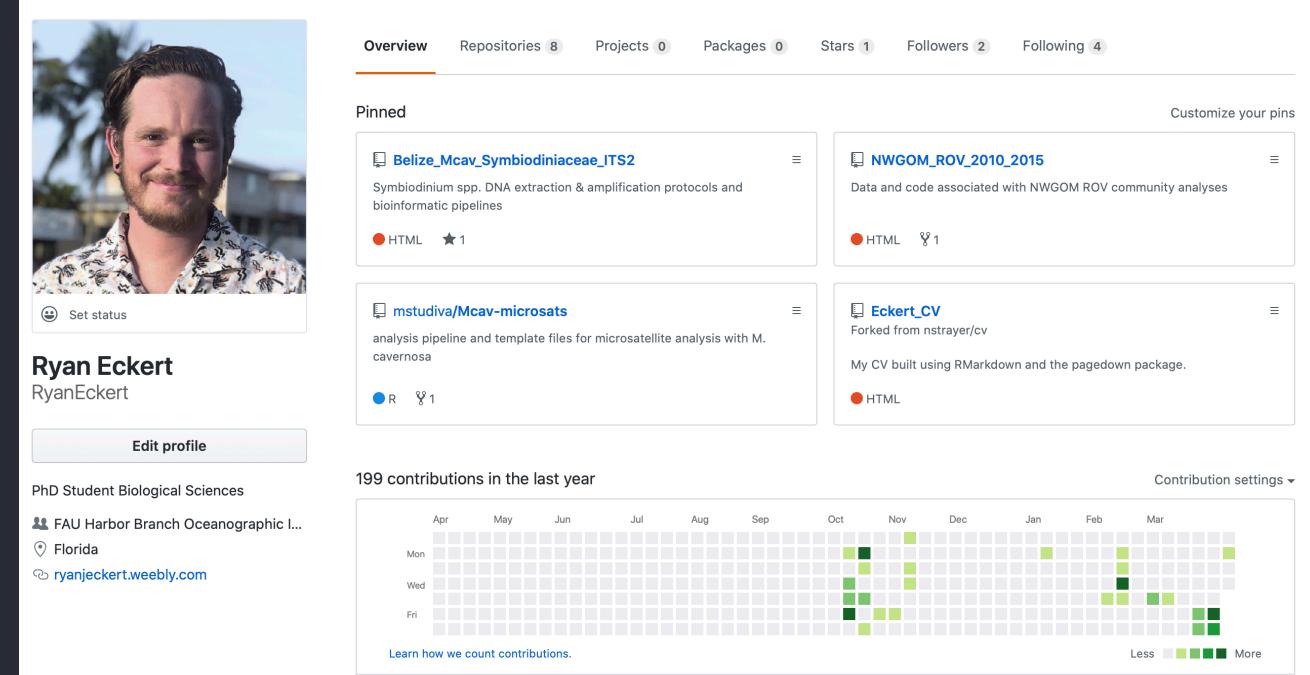
The main content area shows the user's activity. At the top, there are tabs for "Overview", "Repositories 8", "Projects 0", "Packages 0", "Stars 1", "Followers 2", and "Following 4". The "Overview" tab is selected. Below the tabs, there is a section titled "Pinned" which contains four pinned items:

- Belize\_Mcav\_SymbiodiniaceaeITS2**: Symbiodinium spp. DNA extraction & amplification protocols and bioinformatic pipelines. Forked from nstrayer/cv. Last updated 1 day ago.
- NWGOM\_ROV\_2010\_2015**: Data and code associated with NWGOM ROV community analyses. Forked from nstrayer/cv. Last updated 1 day ago.
- mstudiva/Mcav-microsats**: analysis pipeline and template files for microsatellite analysis with *M. cavernosa*. Forked from nstrayer/cv. Last updated 1 day ago.
- Eckert\_cv**: My CV built using RMarkdown and the pagedown package. Forked from nstrayer/cv. Last updated 1 day ago.

Below the pinned items, there is a section titled "199 contributions in the last year" with a grid showing contributions by month and day. A legend at the bottom right indicates that darker shades of green represent more contributions. The grid shows a high density of contributions in October and November, with a significant drop-off in December and January.

# GitHub.

- Development platform
- Site to host and review code



The screenshot shows a GitHub profile page for a user named Ryan Eckert. The profile picture is a photo of a man with a beard and short hair, wearing a patterned shirt. Below the picture is a "Set status" button. The name "Ryan Eckert" and the handle "RyanEckert" are displayed. A "Edit profile" button is visible. Underneath, it says "PhD Student Biological Sciences" and "FAU Harbor Branch Oceanographic I...". It also shows "Florida" with a location pin icon and a link to "ryanjeckert.weebly.com".

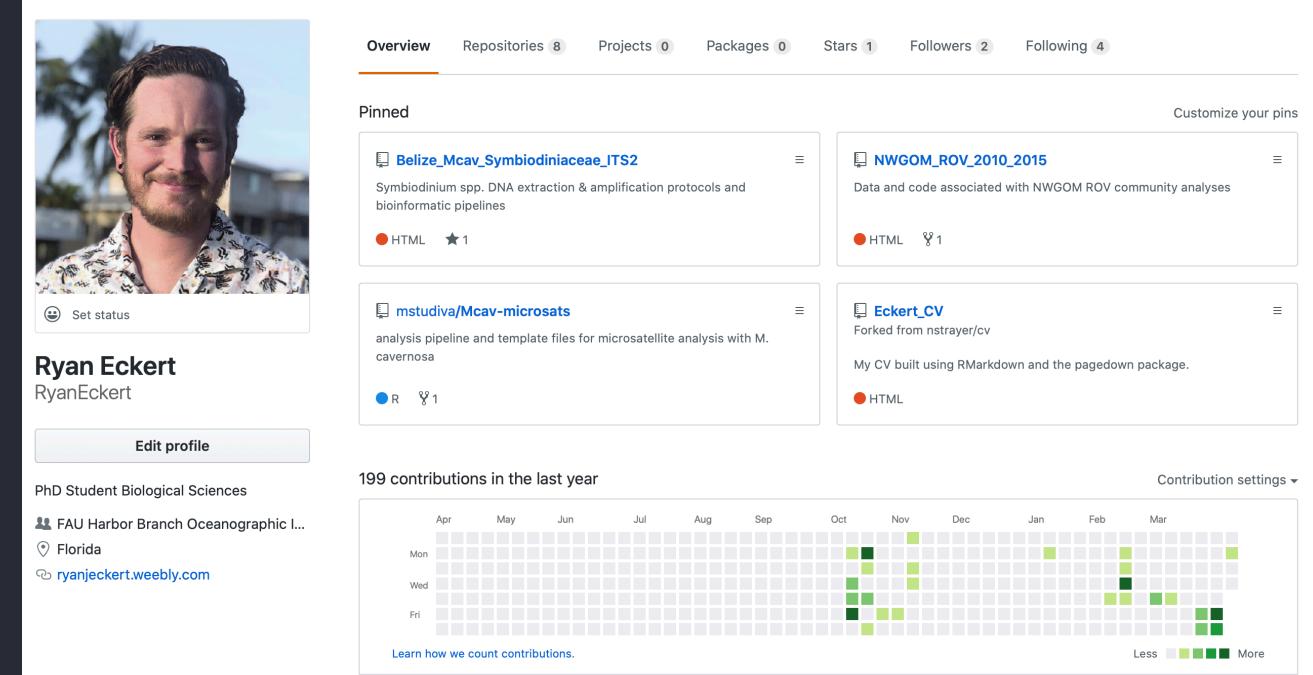
The main content area has tabs for "Overview", "Repositories 8", "Projects 0", "Packages 0", "Stars 1", "Followers 2", and "Following 4". The "Overview" tab is selected. Below the tabs, there's a section titled "Pinned" with four pinned items:

- Belize\_Mcav\_Symbiodiniaceae\_ITS2**: Symbiodinium spp. DNA extraction & amplification protocols and bioinformatic pipelines. (HTML, ★ 1)
- NWGOM\_ROV\_2010\_2015**: Data and code associated with NWGOM ROV community analyses. (HTML, 1)
- mstudiva/Mcav-microsats**: analysis pipeline and template files for microsatellite analysis with *M. cavernosa*. (R, 1)
- Eckert\_CV**: Forked from nstrayer/cv. My CV built using RMarkdown and the pagedown package. (HTML)

Below the pinned items is a section titled "199 contributions in the last year" with a grid chart showing activity by month and day. The legend indicates "Less" (light green) and "More" (dark green).

# GitHub.

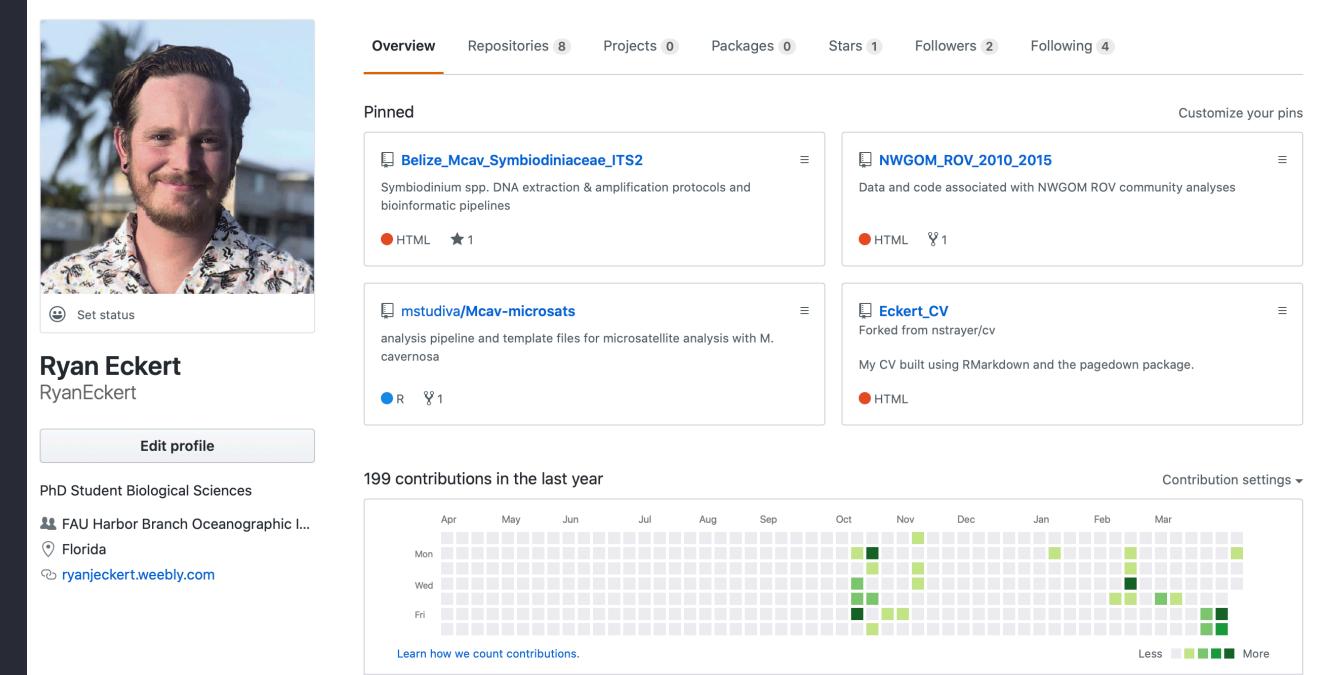
- Development platform
- Site to host and review code
- A way to collaborate on programs/projects



The screenshot shows a GitHub profile page for a user named Ryan Eckert. The profile picture is a photo of a man with a beard and short hair, wearing a patterned shirt. Below the picture is a "Set status" button. The name "Ryan Eckert" is displayed in bold, with "RyanEckert" in a smaller font underneath. A "Edit profile" button is visible. The bio section includes the text "PhD Student Biological Sciences", "FAU Harbor Branch Oceanographic I...", "Florida", and a link to "ryanjeckert.weebly.com". The "Overview" tab is selected, showing repository statistics: 8 repositories, 0 projects, 0 packages, 1 star, 2 followers, and 4 following. The "Pinned" section contains four pinned items: "Belize\_Mcav\_Symbiodiniaceae\_ITS2" (Symbiodinium spp. DNA extraction & amplification protocols and bioinformatic pipelines), "NWGOM\_ROV\_2010\_2015" (Data and code associated with NWGOM ROV community analyses), "mstudiva/Mcav-microsats" (analysis pipeline and template files for microsatellite analysis with *M. cavernosa*), and "Eckert\_CV" (My CV built using RMarkdown and the pagedown package). The "Contributions" section shows a grid of colored squares representing contributions by month and day over the last year, with a legend at the bottom indicating "Less" (light green) and "More" (dark green).

# GitHub.

- Development platform
- Site to host and review code
- A way to collaborate on programs/projects
- Fork repositories of interest



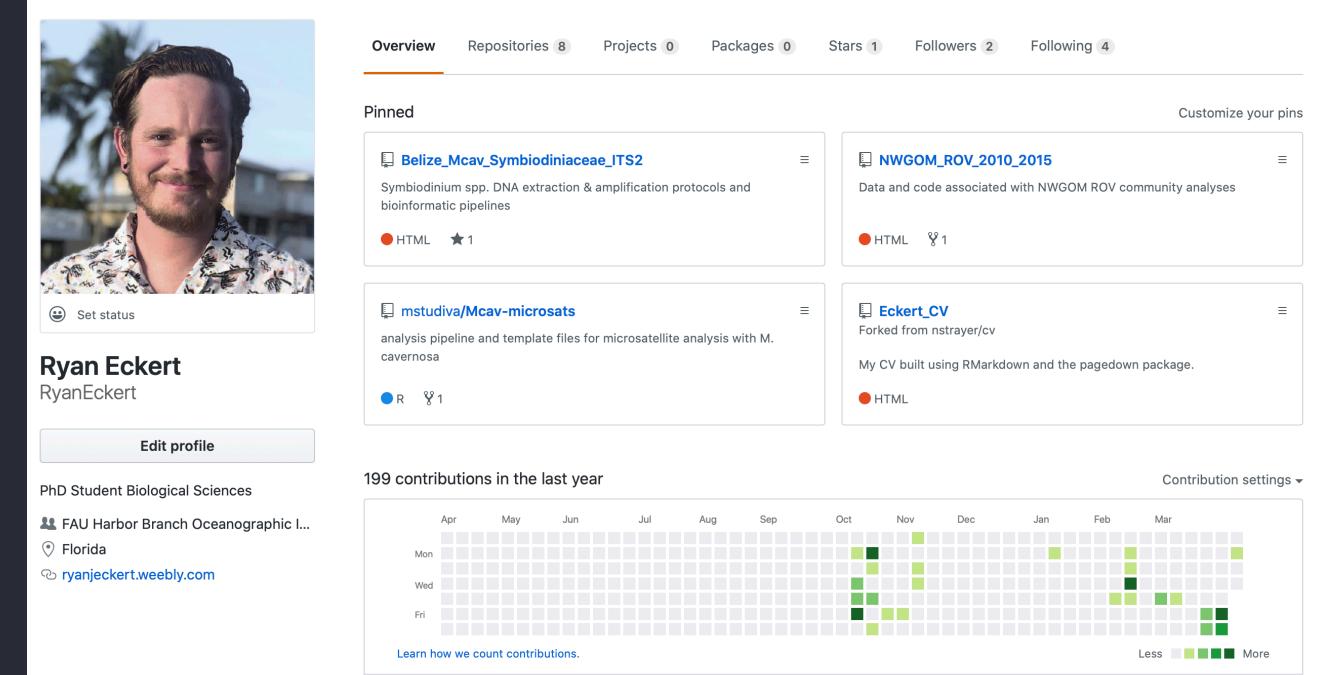
The screenshot shows a GitHub profile page for a user named Ryan Eckert. The profile picture is a portrait of a man with a beard. Below the picture is a "Set status" button. The name "Ryan Eckert" is displayed in bold, followed by the handle "RyanEckert". There is a "Edit profile" button. Underneath, it says "PhD Student Biological Sciences" and "FAU Harbor Branch Oceanographic I...". It also shows "Florida" with a location pin icon and a link to "ryanjeckert.weebly.com". On the right side of the profile, there is a navigation bar with tabs for "Overview", "Repositories 8", "Projects 0", "Packages 0", "Stars 1", "Followers 2", and "Following 4". Below the navigation bar is a section titled "Pinned" which lists four repositories:

- Belize\_Mcav\_Symbiodiniaceae\_ITS2**: Symbiodinium spp. DNA extraction & amplification protocols and bioinformatic pipelines. Last updated: 1 day ago. 1 star.
- NWGOM\_ROV\_2010\_2015**: Data and code associated with NWGOM ROV community analyses. Last updated: 1 day ago. 1 star.
- mstudiva/Mcav-microsats**: analysis pipeline and template files for microsatellite analysis with *M. cavernosa*. Last updated: 1 day ago. 1 star.
- Eckert\_CV**: Forked from nstrayer/cv. My CV built using RMarkdown and the pagedown package. Last updated: 1 day ago. 1 star.

Below the pinned repositories is a section titled "199 contributions in the last year" featuring a heatmap showing contribution activity by month and day. A legend at the bottom right indicates that darker shades of green represent more contributions. At the very bottom, there is a link to "Learn how we count contributions."

# GitHub.

- Development platform
- Site to host and review code
- A way to collaborate on programs/projects
- Fork repositories of interest
- Clone repositories from others



The screenshot shows a GitHub profile page for a user named Ryan Eckert. The profile picture is a portrait of a man with a beard. Below the picture is a "Set status" button. The name "Ryan Eckert" is displayed in bold, followed by the handle "RyanEckert". There is a "Edit profile" button. Underneath, it says "PhD Student Biological Sciences" and lists "FAU Harbor Branch Oceanographic I..." and "Florida". It also includes a link "ryanjeckert.weebly.com". On the right side of the profile, there is a summary bar with the following information: Overview (selected), Repositories 8, Projects 0, Packages 0, Stars 1, Followers 2, Following 4. Below this is a section titled "Pinned" which lists four repositories: "Belize\_Mcav\_Symbiodiniaceae\_ITS2", "NWGOM\_ROV\_2010\_2015", "mstudiva/Mcav-microsats", and "Eckert\_CV". At the bottom, there is a chart titled "199 contributions in the last year" showing activity across months and days of the week.

# GitHub.

- Development platform
- Site to host and review code
- A way to collaborate on programs/projects
- Fork repositories of interest
- Clone repositories from others
- Version control system (VCS)



Ryan Eckert  
RyanEckert

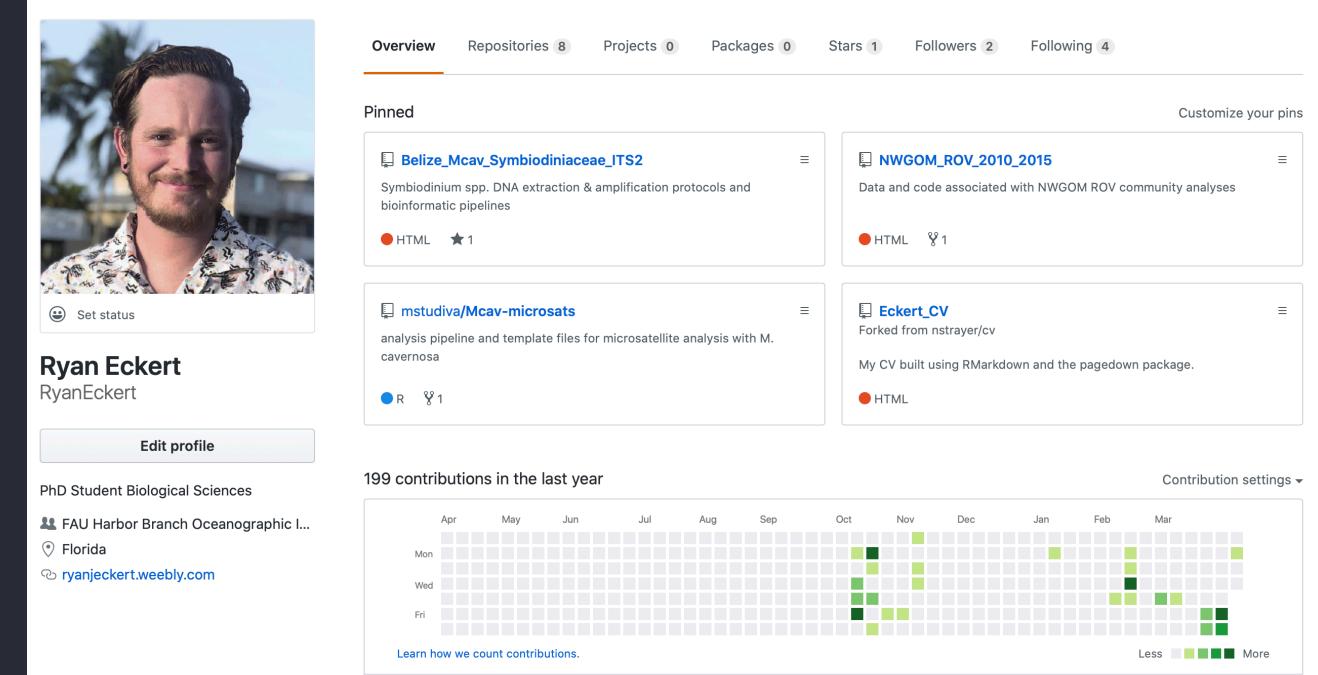
Edit profile

PhD Student Biological Sciences

FAU Harbor Branch Oceanographic I...

Florida

[ryanjeckert.weebly.com](http://ryanjeckert.weebly.com)



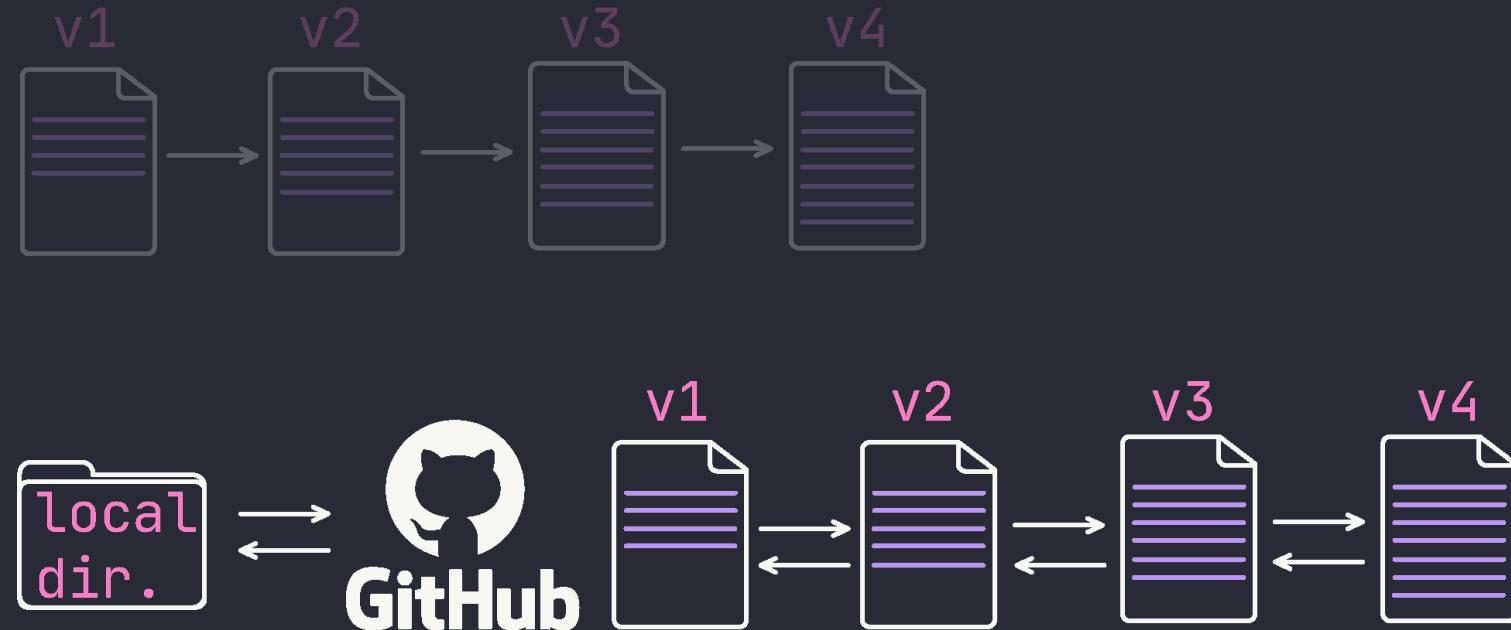
# GitHub.

- Working **locally** on code is often **destructive**—versions replace one another



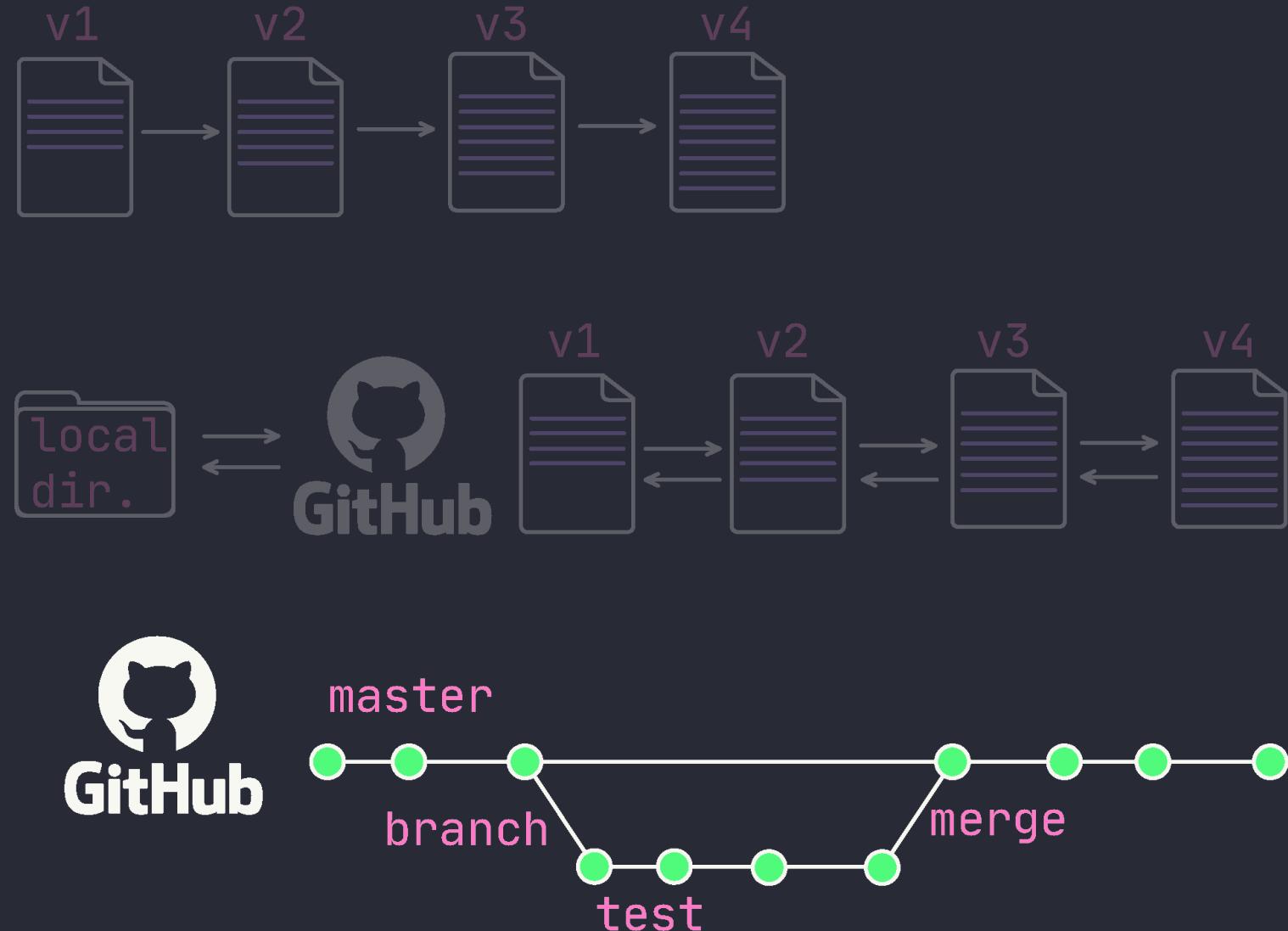
# GitHub.

- Working **locally** on code is often destructive—versions replace one another
- Using **GitHub** allows us to easily implement **version control**—changes are documented between versions \*This depends on how often you commit to GitHub\*



# GitHub.

- Working **locally** on code is often destructive—versions replace one another
- Using **GitHub** allows us to easily implement **version control**—changes are documented between versions \*This depends on how often you commit to GitHub\*
- Additionally, we can **branch** to work on issues away from the **master** experimentally. If we are happy with changes we can **merge** this branch back to the **master**

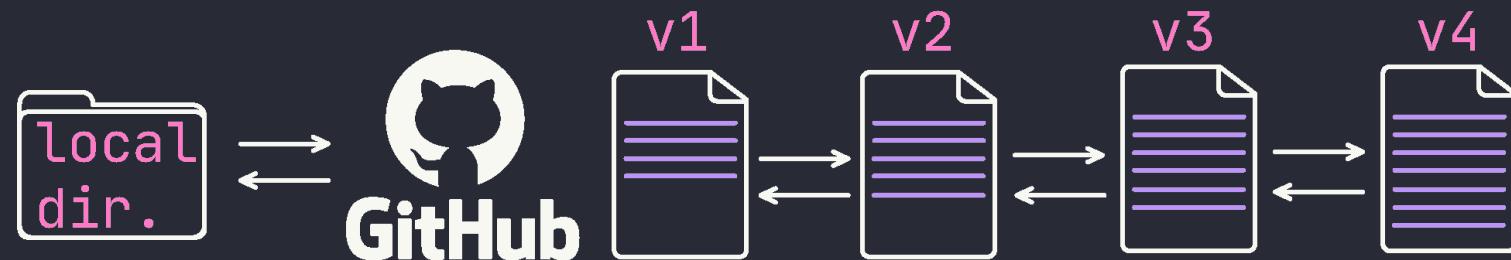


# GitHub.

- Working **locally** on code is often destructive—versions replace one another



- Using **GitHub** allows us to easily implement **version control**—changes are documented between versions \*This depends on how often you commit to GitHub\*



- Additionally, we can **branch** to work on issues away from the **master** experimentally. If we are happy with changes we can **merge** this branch back to the **master**



# Rmarkdown.

- Notebook interface

The screenshot shows the R Markdown notebook interface with two panes. The left pane is the R Console, displaying R code for creating a pie chart. The right pane is the R Markdown editor, showing the resulting pie chart and the corresponding R code.

**Console (Left):**

```
Console R Markdown
~/Documents/GitHub/NWCOM_ROV_2010-2015/code/ ↵
+ library(raster)
+ theme_classic()
>
> coralPerc = coralPercA + theme(
+ axis.title.x = element_blank(),
+ axis.text.x = element_blank(),
+ axis.title.y = element_text(color = "black", size = 8),
+ axis.text.y = element_text(color = "black", size = 8),
+ legend.position = "none",
+ legend.justification = c(0,1),
+ legend.title = element_text(color = "black", size = 8),
+ legend.text = element_text(color = "black", size = 6),
+ legend.key = element_blank(),
+ legend.key.size = unit(0.5,"line"),
+ legend.background = element_blank(),
+ panel.background = element_blank(),
+ panel.border = element_blank()
+)
>
> coralPerc
>
```

**R Markdown Editor (Right):**

```
nwgomAnalyses.Rmd* ↵
Additionally we WILL make a color palette for our coral family data where each of the four
orders we are interested in have a ramp of color in a different hue.

```{r, color palettes etc.}
options("scipen" = 4)

64 colPal = divPal= c("#00B8AA", "#374649", "#FD625E", "#F2C811", "#788580")
65
66 getPalette = colorRampPalette(brewer.pal(8, "Dark2"))
67 getPalette2 = colorRampPalette(redmonder.pal(8, "qPBI"))

68 AL = colorRampPalette(c("FFFFFF", getPalette2(4)[1]))(8)[3:8]
69 AN = colorRampPalette(c("FFFFFF", getPalette2(4)[2]))(4)[2:4]
70 H = getPalette2(4)[3]
71 S = colorRampPalette(c("FFFFFF", getPalette2(4)[4]))(11)[2:11]

72 coralPalette = c(AL,AN,H,S)
73
74 pie(rep(1,length(coralPalette)), col = coralPalette)
75
76
77 ```

42:59 42:59 42:59
```

Pie Chart:

A pie chart divided into 20 equal-sized segments, each labeled with a number from 1 to 20. The segments are colored in a gradient: light orange for 1-4, teal for 5-8, light blue for 9-12, purple for 13-16, and pink for 17-20.

Category	Value
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1
15	1
16	1
17	1
18	1
19	1
20	1

Rmarkdown.

- Notebook interface
- Create dynamic documents
- Create websites, CVs, books, articles, presentations, etc.

The screenshot shows the R Markdown interface with two main panes. The left pane is the R Console, displaying R code related to creating a color palette and a pie chart. The right pane is the R Markdown editor, showing the generated Rmd file and the resulting pie chart.

Console Output:

```
Console R Markdown
~/Documents/GitHub/NWCOM_ROV_2010-2015/code/ ...
+ library(rmarkdown)
+ theme_classic()
>
> coralPerc = coralPercA + theme(
+   axis.title.x = element_blank(),
+   axis.text.x = element_blank(),
+   axis.title.y = element_text(color = "black", size = 8),
+   axis.text.y = element_text(color = "black", size = 8),
+   legend.position = "none",
+   legend.justification = c(0,1),
+   legend.title = element_text(color = "black", size = 8),
+   legend.text = element_text(color = "black", size = 6),
+   legend.key = element_blank(),
+   legend.key.size = unit(0.5,"line"),
+   legend.background = element_blank(),
+   panel.background = element_blank(),
+   panel.border = element_blank()
+ )
>
> coralPerc
```

R Markdown Editor:

```
nwgomAnalyses.Rmd* ...
Additionally we WILL make a color palette for our coral family data where each of the four orders we are interested in have a ramp of color in a different hue.<br>
```{r, color palettes etc.}
options("scipen" = 4)

colPal = divPal= c("#00B8AA", "#374649", "#FD625E", "#F2C811", "#788580")
getPalette = colorRampPalette(brewer.pal(8, "Dark2"))
getPalette2 = colorRampPalette(redmonder.pal(8, "qPBI"))

AL = colorRampPalette(c("FFFFFF", getPalette2(4)[1]))(8)[3:8]
AN = colorRampPalette(c("FFFFFF", getPalette2(4)[2]))(4)[2:4]
H = getPalette2(4)[3]
S = colorRampPalette(c("FFFFFF", getPalette2(4)[4]))(11)[2:11]

coralPalette = c(AL,AN,H,S)

pie(rep(1,length(coralPalette)), col = coralPalette)
```

```

A pie chart is displayed in the editor, divided into 20 equal-sized segments, each labeled with a number from 1 to 20 around its circumference.

Rmarkdown.

- Notebook interface
- Create dynamic documents
- Create websites, CVs, books, articles, presentations, etc.

The screenshot shows the R Markdown interface with two panes. The left pane is the R Console, displaying R code related to theme customization and color palettes. The right pane is the R Markdown editor, showing the corresponding Rmd file and a generated pie chart. The pie chart has 20 segments, each labeled with a number from 1 to 20, representing proportions of a dataset.

```
Console R Markdown ~Documents/GitHub/NWCOM_ROV_2010-2015/code/ ... Addins ...  
+ library(rmarkdown)  
+ theme_classic()  
>  
> coralPerc = coralPercA + theme(  
+   axis.title.x = element_blank(),  
+   axis.text.x = element_blank(),  
+   axis.title.y = element_text(color = "black", size = 8),  
+   axis.text.y = element_text(color = "black", size = 8),  
+   legend.position = "none",  
+   legend.justification = c(0,1),  
+   legend.title = element_text(color = "black", size = 8),  
+   legend.text = element_text(color = "black", size = 6),  
+   legend.key = element_blank(),  
+   legend.key.size = unit(0.5,"line"),  
+   legend.background = element_blank(),  
+   panel.background = element_blank(),  
+   panel.border = element_blank()  
)  
>  
> coralPerc  
>  
nwgomAnalyses.Rmd *  
Additionally we WILL make a color palette for our coral family data where each of the four  
orders we are interested in have a ramp of color in a different hue.<br>  
61 ````{r, color palettes etc.}  
62 options("scipen" = 4)  
63  
64 colPal = divPal= c("#00B8AA", "#374649", "#FD625E", "#F2C811", "#788580")  
65  
66 getPalette = colorRampPalette(brewer.pal(8, "Dark2"))  
67 getPalette2 = colorRampPalette(redmonder.pal(8, "qPBI"))  
68  
69 AL = colorRampPalette(c("FFFFFF", getPalette2(4)[1]))(8)[3:8]  
70 AN = colorRampPalette(c("FFFFFF", getPalette2(4)[2]))(4)[2:4]  
71 H = getPalette2(4)[3]  
72 S = colorRampPalette(c("FFFFFF", getPalette2(4)[4]))(11)[2:11]  
73  
74 coralPalette = c(AL,AN,H,S)  
75  
76 pie(rep(1,length(coralPalette)), col = coralPalette)  
77 ...  
42:59 42:59 Chunk 1 ...  
Environment History Connections Git  
Diff Commit Pull Push History More ...  
Staged Status Path New Branch master C
```

| Segment | Proportion |
|---------|------------|
| 1 | ~0.05 |
| 2 | ~0.05 |
| 3 | ~0.05 |
| 4 | ~0.05 |
| 5 | ~0.05 |
| 6 | ~0.05 |
| 7 | ~0.05 |
| 8 | ~0.05 |
| 9 | ~0.05 |
| 10 | ~0.05 |
| 11 | ~0.05 |
| 12 | ~0.05 |
| 13 | ~0.05 |
| 14 | ~0.05 |
| 15 | ~0.05 |
| 16 | ~0.05 |
| 17 | ~0.05 |
| 18 | ~0.05 |
| 19 | ~0.05 |
| 20 | ~0.05 |

Rmarkdown.

- Three main components:

Rmarkdown.

- Three main components:
- YAML header
 - Set options

```
1 ---  
2 title: "Symbiodiniaceae *ITS2* analyses"  
3 author: "Ryan Eckert -- ryan.j.eckert@gmail.com"  
4 output:  
5   html_document:  
6     theme: flatly  
7     toc: yes  
8     toc_depth: 3  
9     toc_float: yes  
10    pdf_document:  
11      toc: yes  
12      toc_depth: '3'  
13 ---
```

Rmarkdown.

- Three main components:
- YAML header
 - Set options
- Body text
 - Html, markdown, plain text

```
388 ## Running PERMANOVA in R
389 Now let's see how different communities are from each other with
PERMANOVA.
390 We will utilize the `adonis()` function in *vegan*.
391 We will use Bray-Curtis similarity for our distance matrix and
run a total of 9,999 permutations, and test the effects of Site,
Depth, and the interaction between Site and Depth. Dispersion is
heteroschedastic, but PERMANOVA is robust to deviations in
392 homogeneity of variance ([Anderson and Walsh,
2013](https://esajournals.onlinelibrary.wiley.com/doi/10.1890/12-2010.1)target="_blank"))
```

Rmarkdown.

- Three main components:
- YAML header
 - Set options
- Body text
 - Html, markdown, plain text
- Code Chunks
 - Specify language and options for knitting

```
394 ````{r, permanova}
395 set.seed(694)
396 its2Adonis = adonis(its2ProfsNorm[, c(5:ncol(its2ProfsNorm))] ~
397 Depth_zone * Sample_site,
398 data = its2ProfsNorm, permutations = 9999, method = "bray")
399 its2Adonis
400 ````
```

Rmarkdown.

- Knit to a document
 - .html; .pdf; .docx & more

Running PERMANOVA in R

Now let's see how different communities are from each other with PERMANOVA. We will utilize the `adonis()` function in *vegan*. We will use Bray-Curtis similarity for our distance matrix and run a total of 9,999 permutations, and test the effects of Site, Depth, and the interaction between Site and Depth. Dispersion is heteroschedastic, but PERMANOVA is robust to deviations in homogeneity of variance ([Anderson and Walsh, 2013](#))

```
set.seed(694)
its2Adonis = adonis(its2ProfsNorm[, c(5:ncol(its2ProfsNorm))] ~ Depth_zone * Sample_site,
data = its2ProfsNorm, permutations = 9999, method = "bray")

its2Adonis

##
## Call:
## adonis(formula = its2ProfsNorm[, c(5:ncol(its2ProfsNorm))] ~ Depth_zone * Sample_site
## , data = its2ProfsNorm, permutations = 9999, method = "bray")
##
## Permutation: free
## Number of permutations: 9999
##
## Terms added sequentially (first to last)
##
##                               Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
## Depth_zone                  3    3.331  1.11039  6.8801 0.07925 0.0001 ***
## Sample_site                  3    0.705  0.23497  1.4559 0.01677 0.1711
## Depth_zone:Sample_site     9    1.687  0.18740  1.1611 0.04012 0.2675
## Residuals                   225   36.313  0.16139          0.86386
## Total                        240   42.036                      1.00000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

PERMANOVA reveals that **Depth** has a significant effect on Symbiodiniaceae associations in our *M. cavernosa* samples.

Rmarkdown *quick* cheat sheet.

- # Header 1
- ## Header 2
- ##### Header 6

- ***
- *Italics*
- **Bold**
- Super^{^script^}
- Math: \$A = \pi * r^2\$

Header 1

Header 2

Header 6

Italics

Bold

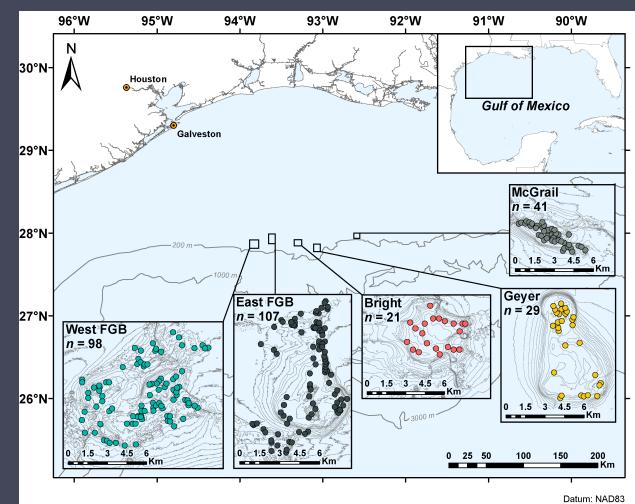
Super^{script}

Math: $A = \pi * r^2$

Rmarkdown *quick* cheat sheet.

- * Unordered
 - * list
 - + with bullets
- 1. Ordered
 - 2. list
 - + with numbers
- ! [Figure 1](path/to/image.png)

- Unordered
 - list
 - with bullets
- 1. Ordered
 - 2. list
 - with numbers



Rmarkdown *quick* cheat sheet.

- ````{language, name, parameters}
Code chunk
```
- print code `inline`
- `r paste("run code inline")`
- [Hyperlink](https://ryanjeckert.weebly.com){target="\_blank"}

```
```{language, name, parameters}  
Code chunk  
```
```

Print code **inline**

run code inline

Hyperlink

# Zenodo.

- Open-access repository

# Zenodo.

- Open-access repository
- Can permanently **archive** data, metadata, and code

# Zenodo.

- Open-access repository
- Can permanently archive data, metadata, and code
- Link to GitHub repo and create a DOI

# Zenodo.

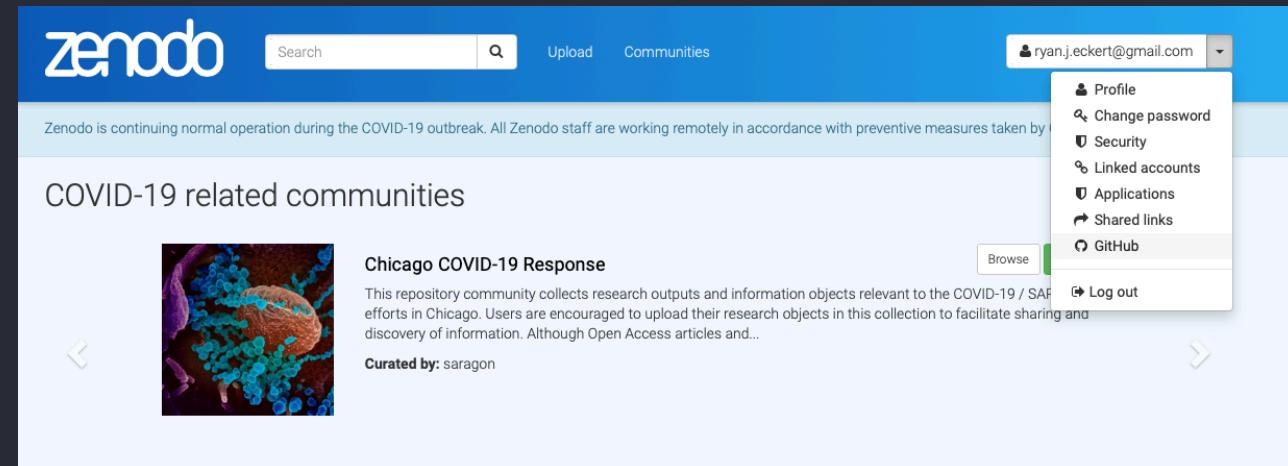
- Log in to Zenodo ([zenodo.org](https://zenodo.org))  
with GitHub account

# Zenodo.

- Log in to Zenodo ([zenodo.org](https://zenodo.org)) with GitHub account
- Authorize application: Zenodo

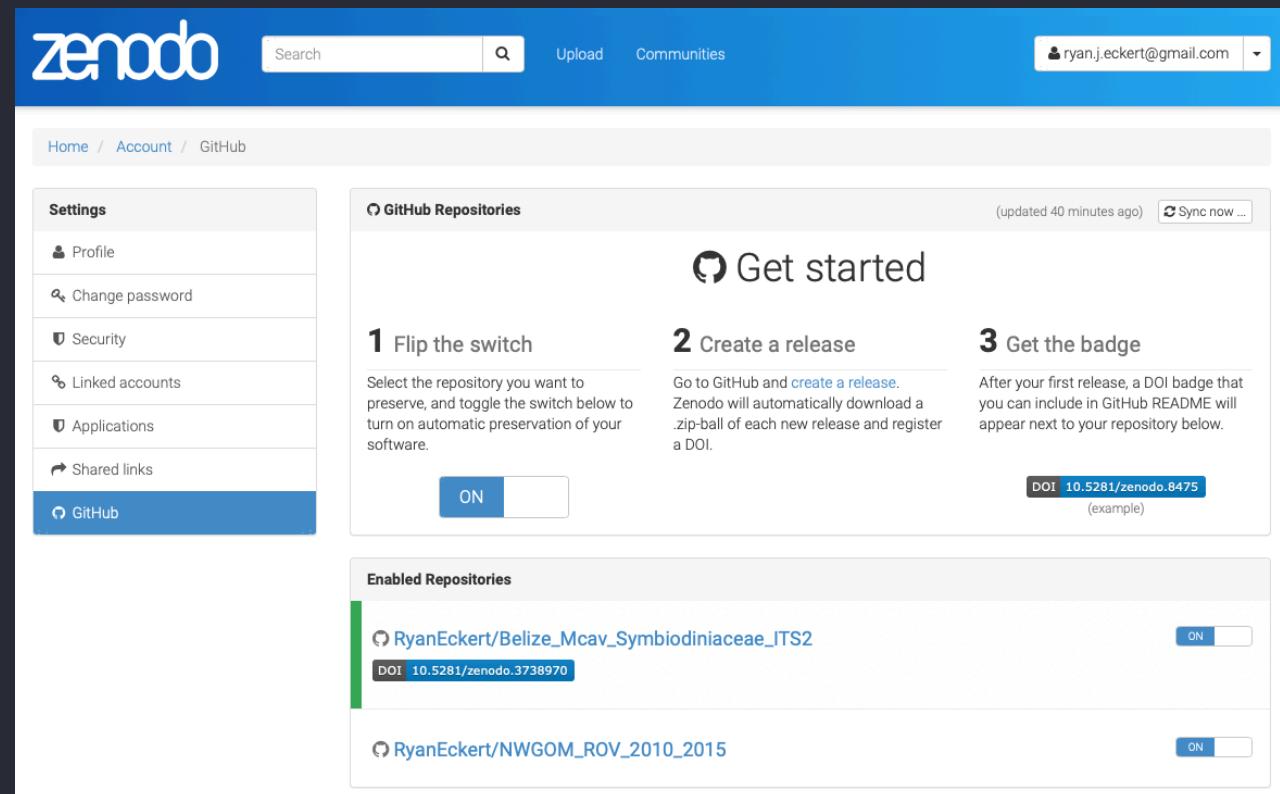
# Zenodo.

- Log in to Zenodo (zenodo.org) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo



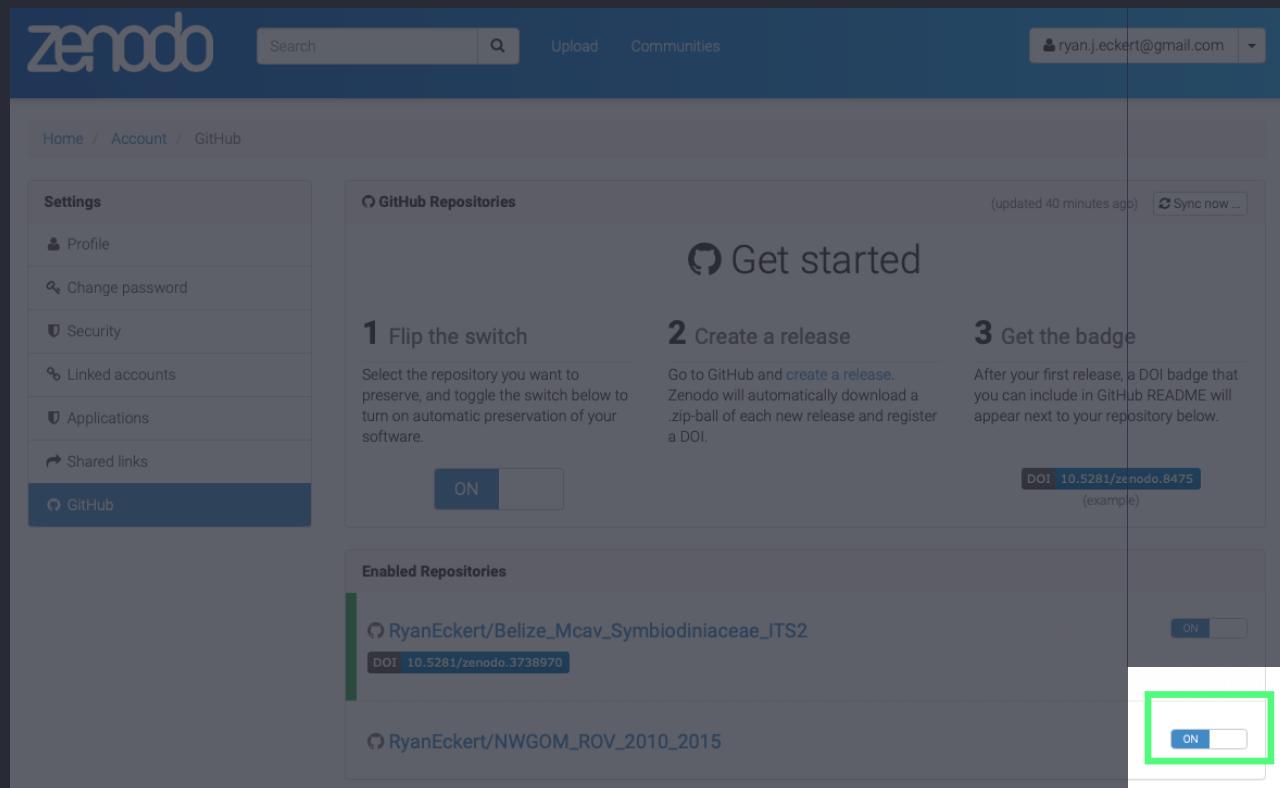
# Zenodo.

- Log in to Zenodo ([zenodo.org](https://zenodo.org)) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo



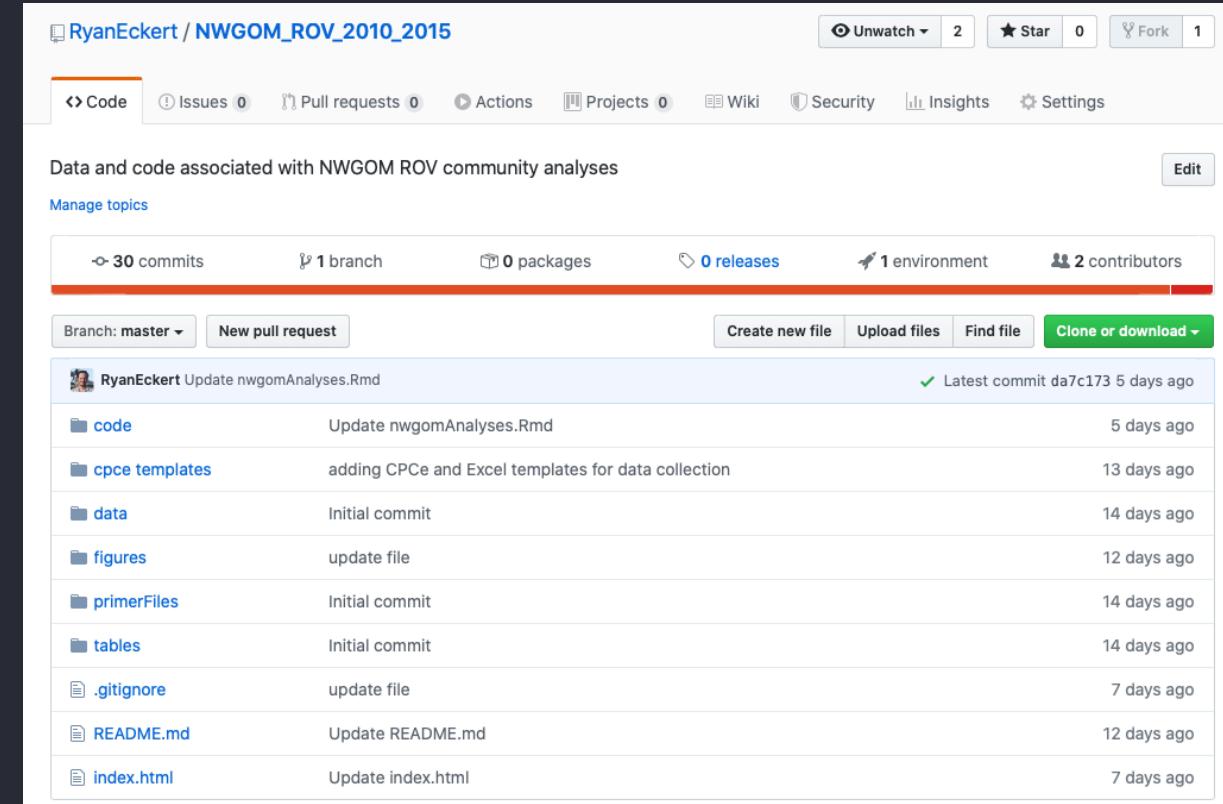
# Zenodo.

- Log in to Zenodo (zenodo.org) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo



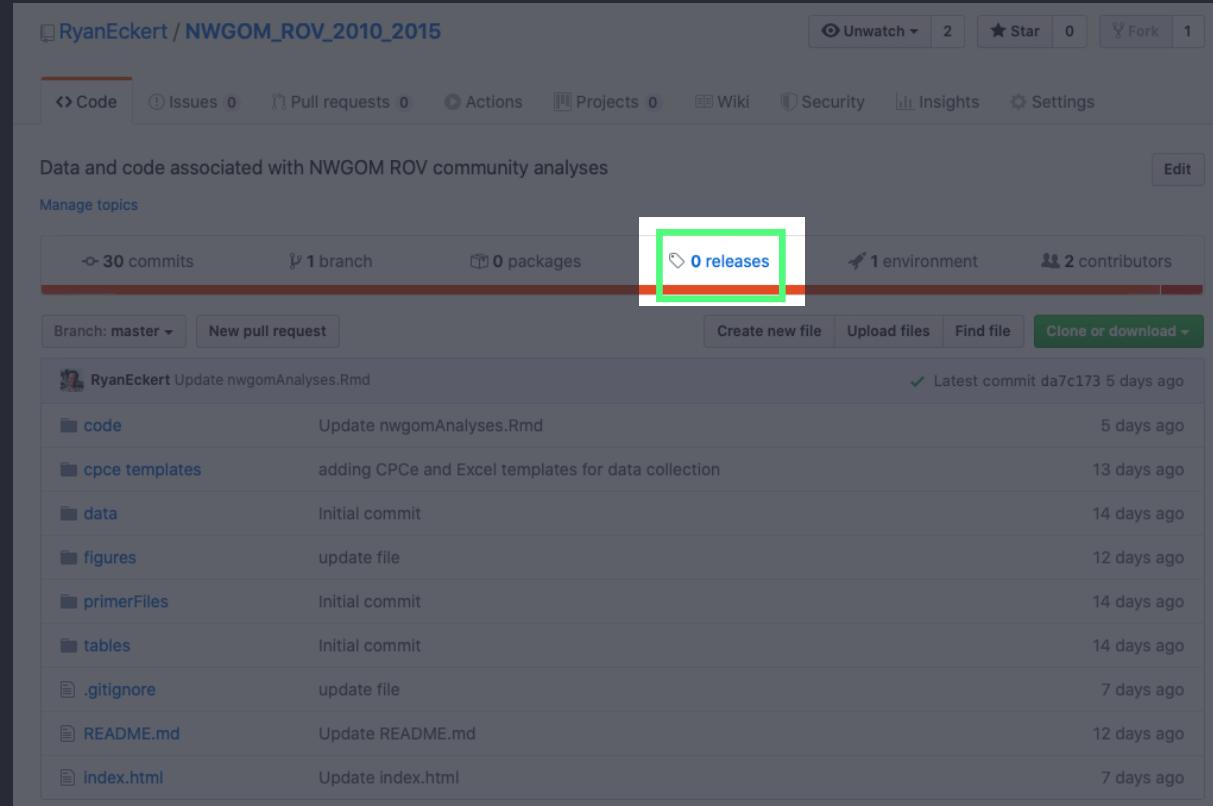
# Zenodo.

- Log in to Zenodo (zenodo.org) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo
- Create a release on GitHub



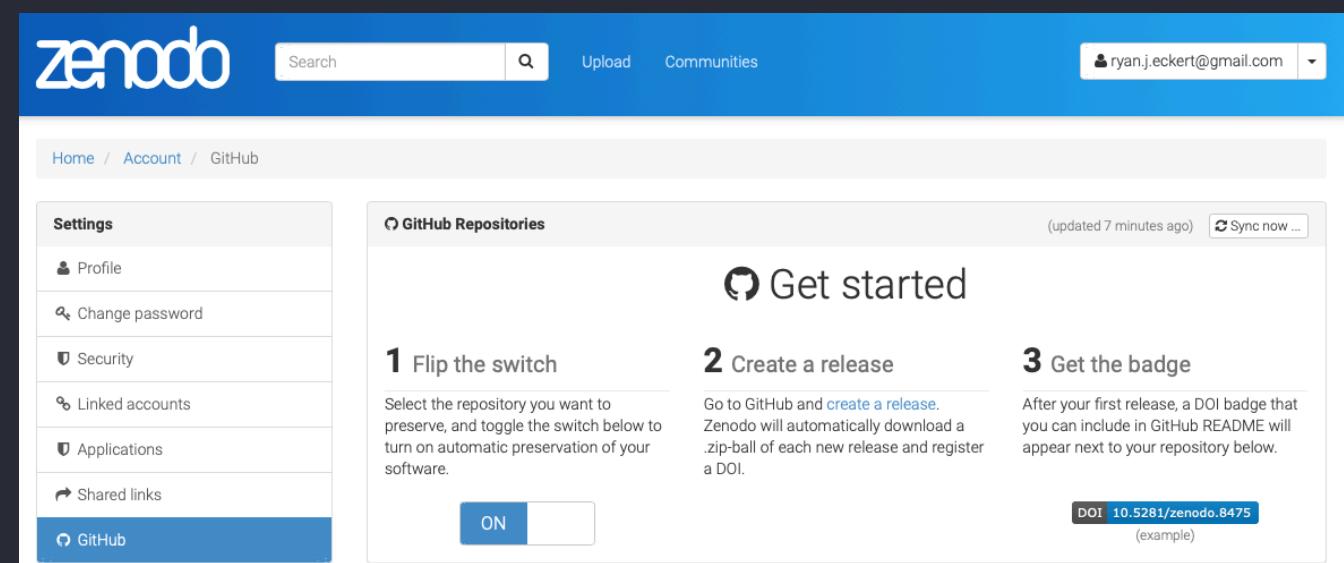
# Zenodo.

- Log in to Zenodo (zenodo.org) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo
- Create a release on GitHub



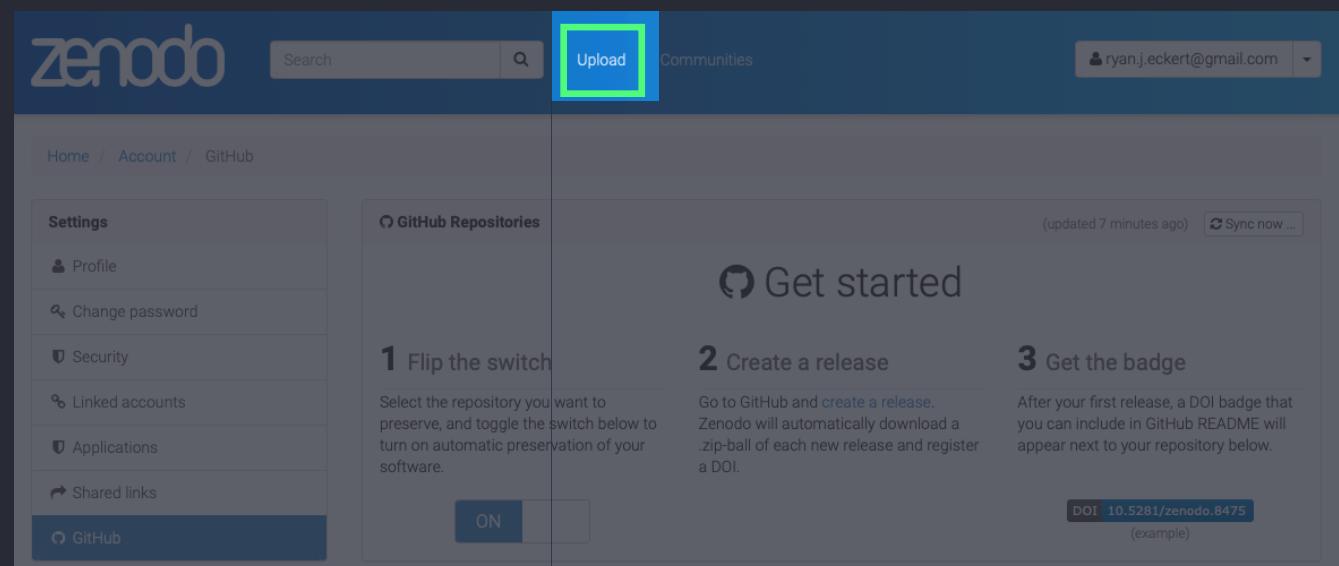
# Zenodo.

- Log in to Zenodo ([zenodo.org](https://zenodo.org)) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo
- Create a release on GitHub
- Complete info for release on Zenodo under upload



# Zenodo.

- Log in to Zenodo ([zenodo.org](https://zenodo.org)) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo
- Create a release on GitHub
- Complete info for release on Zenodo under upload



# Zenodo.

- Log in to Zenodo ([zenodo.org](https://zenodo.org)) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo
- Create a release on GitHub
- Complete info for release on Zenodo under upload

The screenshot shows the Zenodo web interface for a specific release. At the top, there's a navigation bar with 'zenodo' logo, search bar, upload button, and communities section. A user profile for 'ryan.j.eckert@gmail.com' is visible. Below the header, the date 'April 10, 2020' is shown, along with 'Software' and 'Open Access' buttons. The main content area displays a release titled 'RyanEckert/NWGOM\_ROV\_2010\_2015: NWGOM ROV transect analysis'. It includes a brief description: 'Pre-release of data analysis from mesophotic ROV transects in the NWGOM from 2010–2015.' Below the title, there's a preview section showing the contents of a ZIP file: 'NWGOM\_ROV\_2010\_2015-v1.0.0-alpha.zip'. This section lists files like '.gitignore', 'README.md', 'code', '.Rproj.user', '171C38C8', 'build\_options', 'pos', 'files-pane.pper', 'source-pane.pper', 'windowlayoutstate.pper', 'workbench-pane.pper', 'rmd-outputs', 'saved\_source\_markers', 'sources', 'prop', '8AFE4F94', and 'INDEX'. To the right of the preview, file sizes are listed: 68 Bytes, 3.8 kB, 199 Bytes, 171 Bytes, 24 Bytes, 295 Bytes, 65 Bytes, 316 Bytes, 27 Bytes, 61 Bytes, and 169 Bytes. Below the preview, there's a table for 'Files (10.8 MB)'. The table has columns for 'Name' and 'Size'. It lists 'RyanEckert/NWGOM\_ROV\_2010\_2015-v1.0.0-alpha.zip' (10.8 MB) and 'md5:efd86c2cb42cce6afcd87d46dbde1c2' (10.8 MB). Buttons for 'Preview' and 'Download' are next to the first file. On the right side of the page, there are sections for 'Available in' (GitHub, OpenAIRE), 'Publication date:' (April 10, 2020), 'DOI:' (DOI 10.5281/zenodo.3747111), 'Related identifiers:' (Supplement to [https://github.com/RyanEckert/NWGOM\\_ROV\\_2010\\_2015/tree/v1.0.0-alpha](https://github.com/RyanEckert/NWGOM_ROV_2010_2015/tree/v1.0.0-alpha)), and 'License (for files):' (Other (Open)).

# Zenodo.

- Log in to Zenodo ([zenodo.org](https://zenodo.org)) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo
- Create a release on GitHub
- Complete info for release on Zenodo under upload

The screenshot shows the Zenodo web interface for a dataset titled "RyanEckert/NWGOM\_ROV\_2010\_2015: NWGOM ROV transect analysis". The dataset was created on April 10, 2020, and is marked as "Open Access". A green box highlights the "Edit" button in the top right corner of the main content area. Below it, a yellow box highlights the "New version" button. The central part of the screen displays a file tree for "NWGOM\_ROV\_2010\_2015-v1.0.0-alpha.zip", showing various files and their sizes. To the right, there are sections for "Available in" (GitHub, OpenAIRE), "Publication date" (April 10, 2020), "DOI" (10.5281/zenodo.3747111), "Related identifiers" (Supplement to https://github.com/RyanEckert/NWGOM\_ROV\_2010\_2015/tree/v1.0.0-alpha), and "License (for files)" (Other (Open)).

# Zenodo.

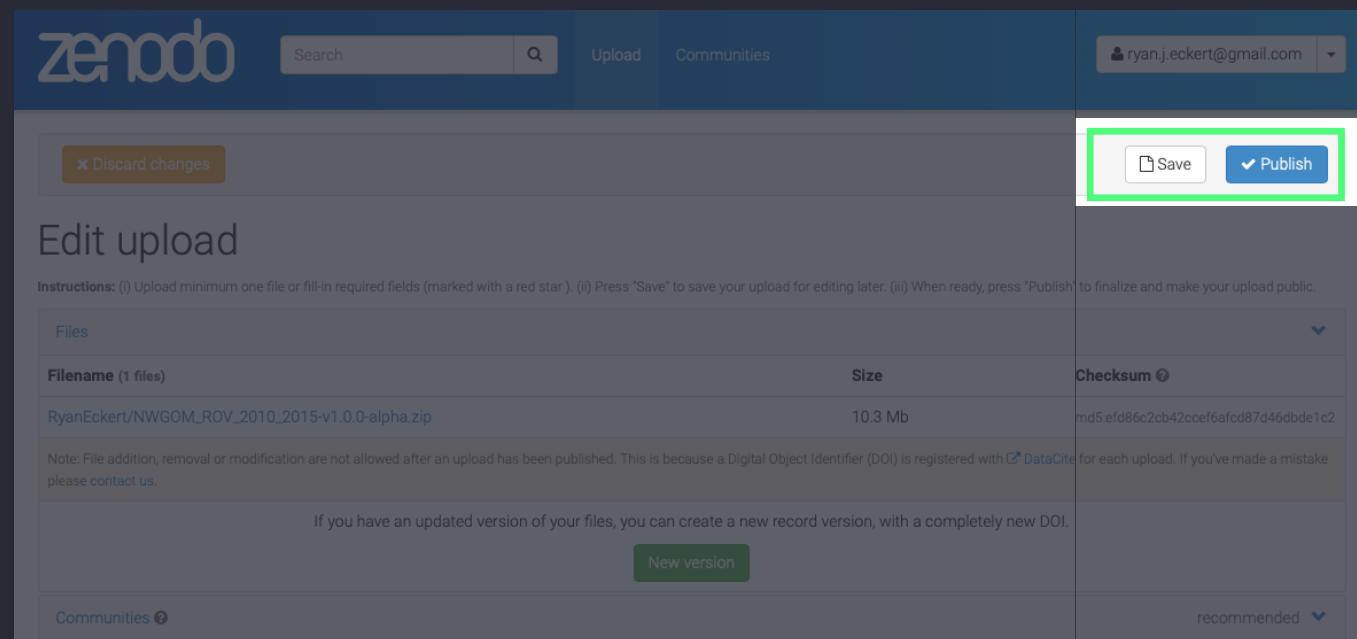
- Log in to Zenodo ([zenodo.org](https://zenodo.org)) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo
- Create a release on GitHub
- Complete info for release on Zenodo under upload
- Save edits and publish

The screenshot shows the Zenodo web interface for managing uploads. At the top, there's a blue header bar with the Zenodo logo, a search bar, an 'Upload' button, and a 'Communities' link. On the right, a user profile is shown with the email 'ryan.j.eckert@gmail.com'. Below the header, a large orange button labeled 'Discard changes' is visible. The main area is titled 'Edit upload' and contains instructions: '(i) Upload minimum one file or fill-in required fields (marked with a red star). (ii) Press "Save" to save your upload for editing later. (iii) When ready, press "Publish" to finalize and make your upload public.' A table lists a single file: 'RyanEckert/NWGOM\_ROV\_2010\_2015-v1.0.0-alpha.zip' (Size: 10.3 Mb, Checksum: md5:efd86c2cb42ccf6afcd87d46d1de1c2). A note below states: 'Note: File addition, removal or modification are not allowed after an upload has been published. This is because a Digital Object Identifier (DOI) is registered with DataCite for each upload. If you've made a mistake please [contact us](#).' At the bottom, a green button says 'New version' and a link 'Communities' with a 'recommended' dropdown.

| Files                                           | Size    | Checksum                            |
|-------------------------------------------------|---------|-------------------------------------|
| RyanEckert/NWGOM_ROV_2010_2015-v1.0.0-alpha.zip | 10.3 Mb | md5:efd86c2cb42ccf6afcd87d46d1de1c2 |

# Zenodo.

- Log in to Zenodo ([zenodo.org](https://zenodo.org)) with GitHub account
- Authorize application: Zenodo
- Go to GitHub tab in Zenodo
- Toggle switch to “ON” for selected repo
- Create a release on GitHub
- Complete info for release on Zenodo under upload
- Save edits and publish



# Zenodo.

- Now you should see a DOI badge on Zenodo

DOI 10.5281/zenodo.3747110

# Zenodo.

- Now you should see a DOI badge on Zenodo
- Since we know how to write markdown we can include the DOI badge on our GitHub and in our Rmarkdown documents.

The screenshot shows a Zenodo document page. At the top, there's a sidebar with a dark blue header containing the text "About this document". Below the header, the sidebar lists several items: "Basic setup of R environment", "Import data", "Percent cover and coral density by bank", "Alpha diversity metrics", "Hierarchical cluster analysis", and "Non-metric multidimensional scaling". To the right of the sidebar, the main content area has a title "NW GOM ROV analyses" in large blue text. Below the title, it says "Ryan Eckert – ryan.j.eckert@gmail.com" and "version: April 10, 2020". A blue button labeled "DOI 10.5281/zenodo.3747110" is present. Below the DOI button is a link "GitHub repository".

The screenshot shows a GitHub repository page for "NWGOM\_ROV\_2010\_2015". The header includes the repository name, a "Code" button, and various navigation links like "Issues 0", "Pull requests 0", "Actions", "Projects 0", "Wiki", "Security", "Insights", and "Settings". It also shows the number of stars (2), forks (1), and the last commit (1430244 now). The main content area shows a commit from "RyanEckert" updating "README.md". Below the commit, it says "1 contributor". The file stats are "77 lines (59 sloc) | 3.71 KB". At the bottom of the page, there's a section titled "NWGOM ROV transect analyses" with a "DOI 10.5281/zenodo.3747110" button, the author "Ryan Eckert -- ryan.j.eckert@gmail.com", and the version "version: March 28, 2020".

# Zenodo.

- Now you should see a DOI badge on Zenodo
- Since we know how to write markdown we can include the DOI badge on our GitHub and in our Rmarkdown documents.



About this document

Basic setup of R environment  
Import data  
Percent cover and coral density by bank  
Alpha diversity metrics  
Hierarchical cluster analysis  
Non-metric multidimensional scaling

## NW GOM ROV analyses

Ryan Eckert – [ryan.j.eckert@gmail.com](mailto:ryan.j.eckert@gmail.com)  
version: April 10, 2020

[DOI 10.5281/zenodo.3747110](https://doi.org/10.5281/zenodo.3747110)

[GitHub repository](#)

### About this document

RyanEckert / NWGOM\_ROV\_2010\_2015

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security Insights Settings

Branch: master NWGOM\_ROV\_2010\_2015 / README.md Find file Copy path

RyanEckert Update README.md 1430244 now

1 contributor

77 lines (59 sloc) | 3.71 KB Raw Blame History

## NWGOM ROV transect analyses

DOI [10.5281/zenodo.3747110](https://doi.org/10.5281/zenodo.3747110)

Ryan Eckert -- [ryan.j.eckert@gmail.com](mailto:ryan.j.eckert@gmail.com)

version: March 28, 2020

# Programs & packages.

- Rstudio (<https://rstudio.com/products/rstudio/download/>)



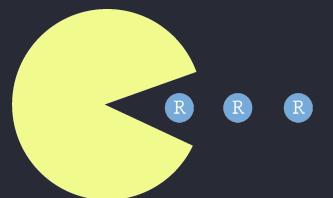
# Programs & packages.

- Rstudio (<https://rstudio.com/products/rstudio/download/>)
- rmarkdown ( `install.packages("rmarkdown")` )



# Programs & packages.

- Rstudio (<https://rstudio.com/products/rstudio/download/>)
- rmarkdown ( `install.packages("rmarkdown")` )
- pacman ( `install.packages("pacman")` )



# Programs & packages.

- Rstudio (<https://rstudio.com/products/rstudio/download/>)
- rmarkdown ( `install.packages("rmarkdown")` )
- pacman ( `install.packages("pacman")` )
- tidyverse ( `install.packages("tidyverse")` )



# Programs & packages.

- patchwork ( `(install.packages("patchwork"))` )



# Programs & packages.

- `patchwork` ( `(install.packages("patchwork"))` )
- `GitHub Desktop` (<https://desktop.github.com>)



# Programs & packages.

- **patchwork** ( `(install.packages("patchwork"))` )
- **GitHub Desktop** (<https://desktop.github.com>)
- **Atom** (<https://atom.io/download/>)



# Programs & packages.

- **patchwork** ( `(install.packages("patchwork"))` )
- **GitHub Desktop** (<https://desktop.github.com>)
- **Atom** (<https://atom.io/download/>)
- **Terminal** (Mac)



# Best practices/recommendations.

- Initialize a GitHub repo for data analysis

# Best practices/recommendations.

- Initialize a GitHub repo for data analysis
- Create an R project file for new repo (.Rproj)

# Best practices/recommendations.

- Initialize a GitHub repo for data analysis
- Create an R project file for new repo (.Rproj)
- START writing code in Rmarkdown

# Best practices/recommendations.

- Initialize a GitHub repo for data analysis
- Create an R project file for new repo (.Rproj)
- START writing code in Rmarkdown
- Spend time upfront creating publishable figures

# Best practices/recommendations.

- Commit to GitHub often while working to maintain good VC

# Best practices/recommendations.

- Commit to GitHub often while working to maintain good VC
- Submit raw sequences to NCBI SRA (or equivalent)

# Best practices/recommendations.

- Commit to GitHub often while working to maintain good VC
- Submit raw sequences to NCBI SRA (or equivalent)
- When publishing, create a Zenodo archive for GitHub repo and create DOI

# Tips & tricks.

- Look at GitHub repos/code from papers with analyses/plots you want to use

# Tips & tricks.

- Look at GitHub repos/code from papers with analyses/plots you want to use
- Stackoverflow.com

# Tips & tricks.

- Look at GitHub repos/code from papers with analyses/plots you want to use
- Stackoverflow.com
- Use `tidyverse` for data manipulation/plotting and `patchwork` for multi-panel figures

# Tips & tricks.

- Look at GitHub repos/code from papers with analyses/plots you want to use
- Stackoverflow.com
- Use `tidyverse` for data manipulation/plotting and `patchwork` for multi-panel figures
- Color palettes: `paletteer` (`install.packages("paletteer")` ); create your own with: `colorRampPalette()`

# Tips & tricks.

- Use **dark themes** in Terminal, Atom, Rstudio, etc. to reduce eye strain/fatigue (I like **Dracula** or **SMYCK**)



# Tips & tricks.

- Use dark themes in Terminal, Atom, Rstudio, etc. to reduce eye strain/fatigue (I like Dracula or SMYCK)
- Change font to help with readability  
(I prefer JetBrains Mono)



# Tips & tricks.

- Use dark themes in Terminal, Atom, Rstudio, etc. to reduce eye strain/fatigue (I like Dracula or SMYCK)
- Change font to help with readability (I prefer JetBrains Mono)
- Keep everything **CONSISTENT** and **UNIQUE**
  - No variables with the same name over and over (i.e. df, plot)



# Tips & tricks.

- I use **caterpillarCase** for variables and objects
  - `thisMakesItEasierToRead` (for me)
  - `I_prefer_it_over_all_these_damn_underscores-or-hyphens`

# Tips & tricks.

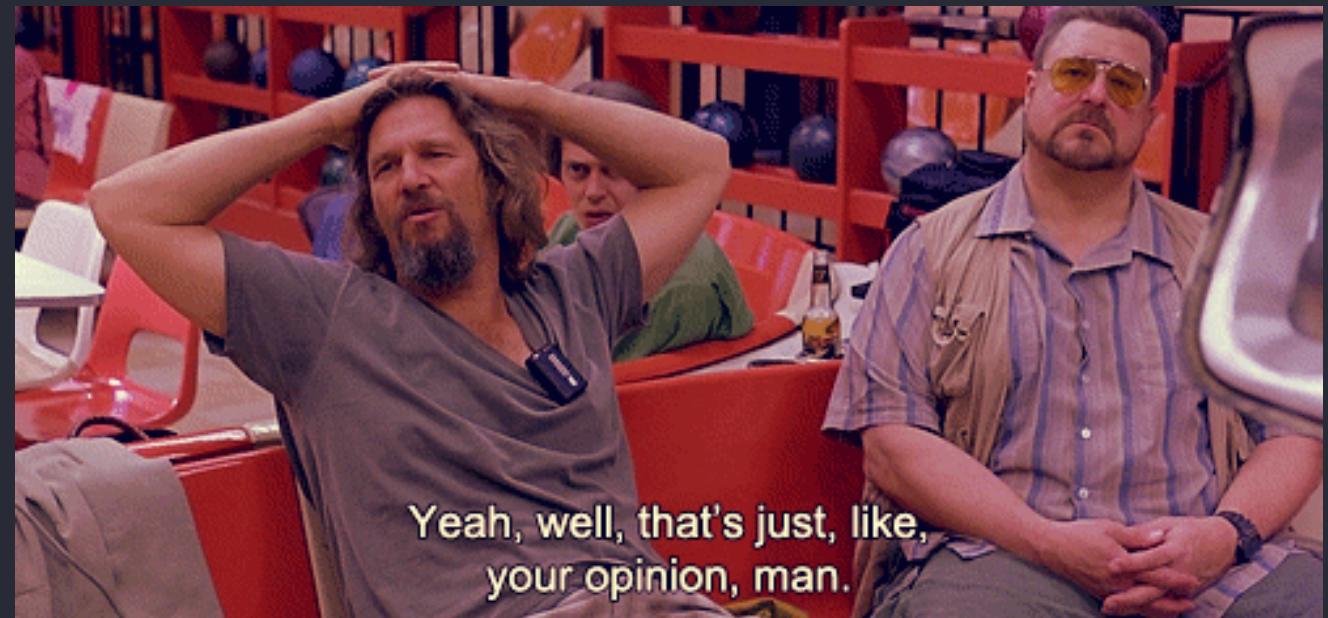
- I use **caterpillarCase** for variables and objects
  - `thisMakesItEasierToRead` (for me)
  - `I_prefer_it_over_all_these_damn_underscores-or-hyphens`
- Keep variables and objects **short** when possible
  - `northwestGulfOfMexicoRovOnlyCoralSpecies` = `read.csv("nwgomCoral.csv")`
  - `nwgomCoral` = `read.csv("nwgomCoral.csv")`

# Tips & tricks.

- I use `caterpillarCase` for variables and objects
  - `thisMakesItEasierToRead` (for me)
  - `I_prefer_it_over_all_these_damn_underscores-or-hyphens`
- Keep variables and objects `short` when possible
  - `northwestGulfOfMexicoRovOnlyCoralSpecies` = `read.csv("nwgomCoral.csv")`
  - `nwgomCoral` = `read.csv("nwgomCoral.csv")`
- I keep spaces in my code when allowed/optional (e.g. `R`)
  - `This = much(easier = "on the eyes", in my opinion)`
  - `This=becomes(very="tedious",to,read,over=time)`

# There is not one right way.

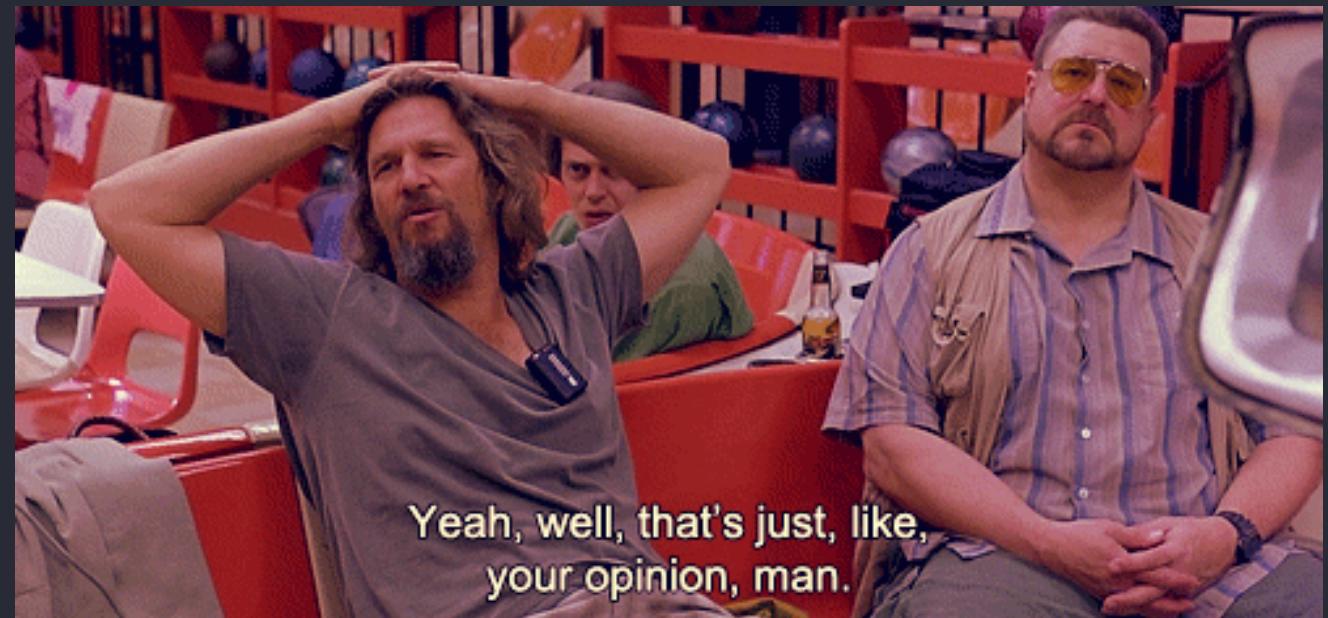
This is typically how *I* do things.



# There is not one right way.

This is typically how *I* do things.

I have found that this workflow works  
*well* for *me*.

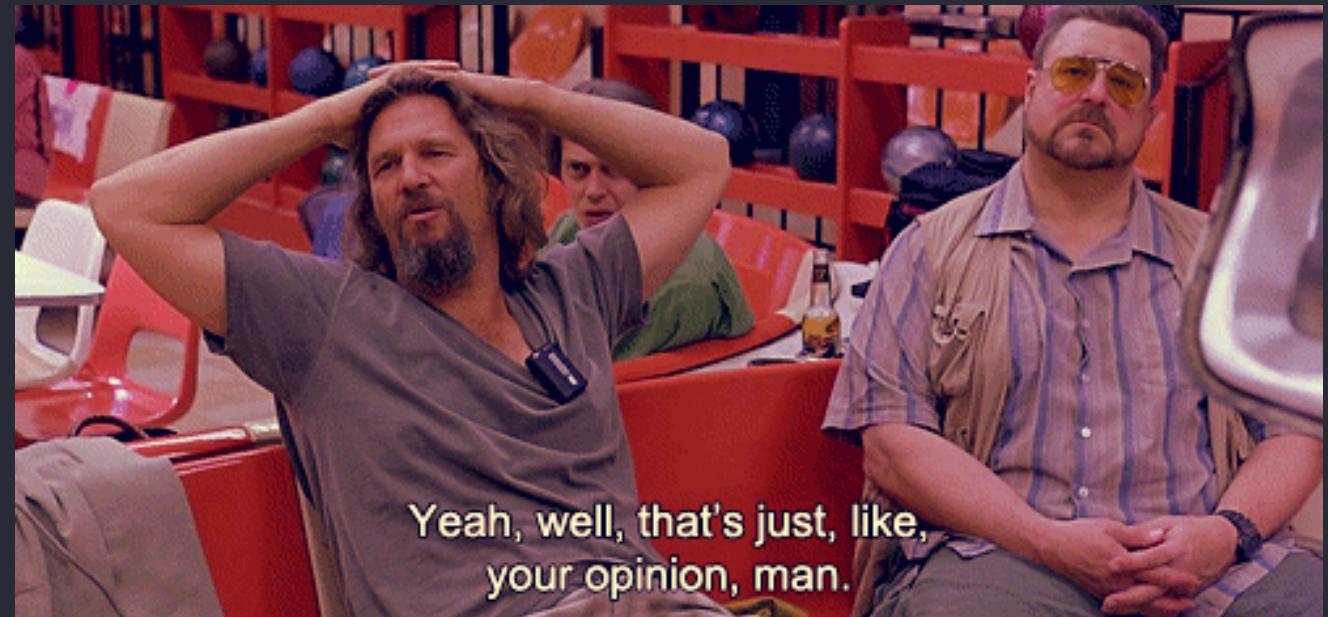


# There is not one right way.

This is typically how *I* do things.

I have found that this workflow works  
*well* for *me*.

Ultimately, these are guidelines to  
help introduce ways to work  
effectively and commit to *open*  
science.



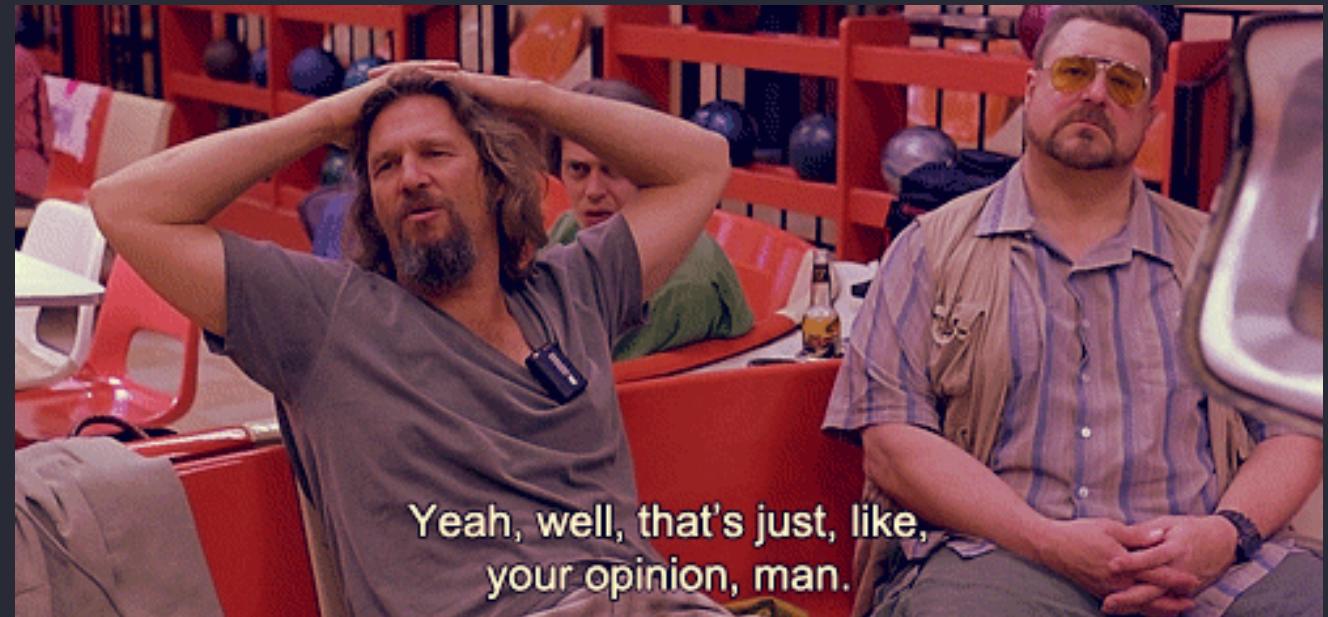
# There is not one right way.

This is typically how *I* do things.

I have found that this workflow works  
*well* for *me*.

Ultimately, these are guidelines to help introduce ways to work effectively and commit to *open science*.

Take these ideas and implement a system that works *well* for *you*.



# There is not one right way.

This is typically how *I* do things.

I have found that this workflow works  
*well* for *me*.

Ultimately, these are guidelines to  
help introduce ways to work  
effectively and commit to *open*  
science.

Take these ideas and implement a  
system that works *well* for *you*.

*Freak what you feel.*

