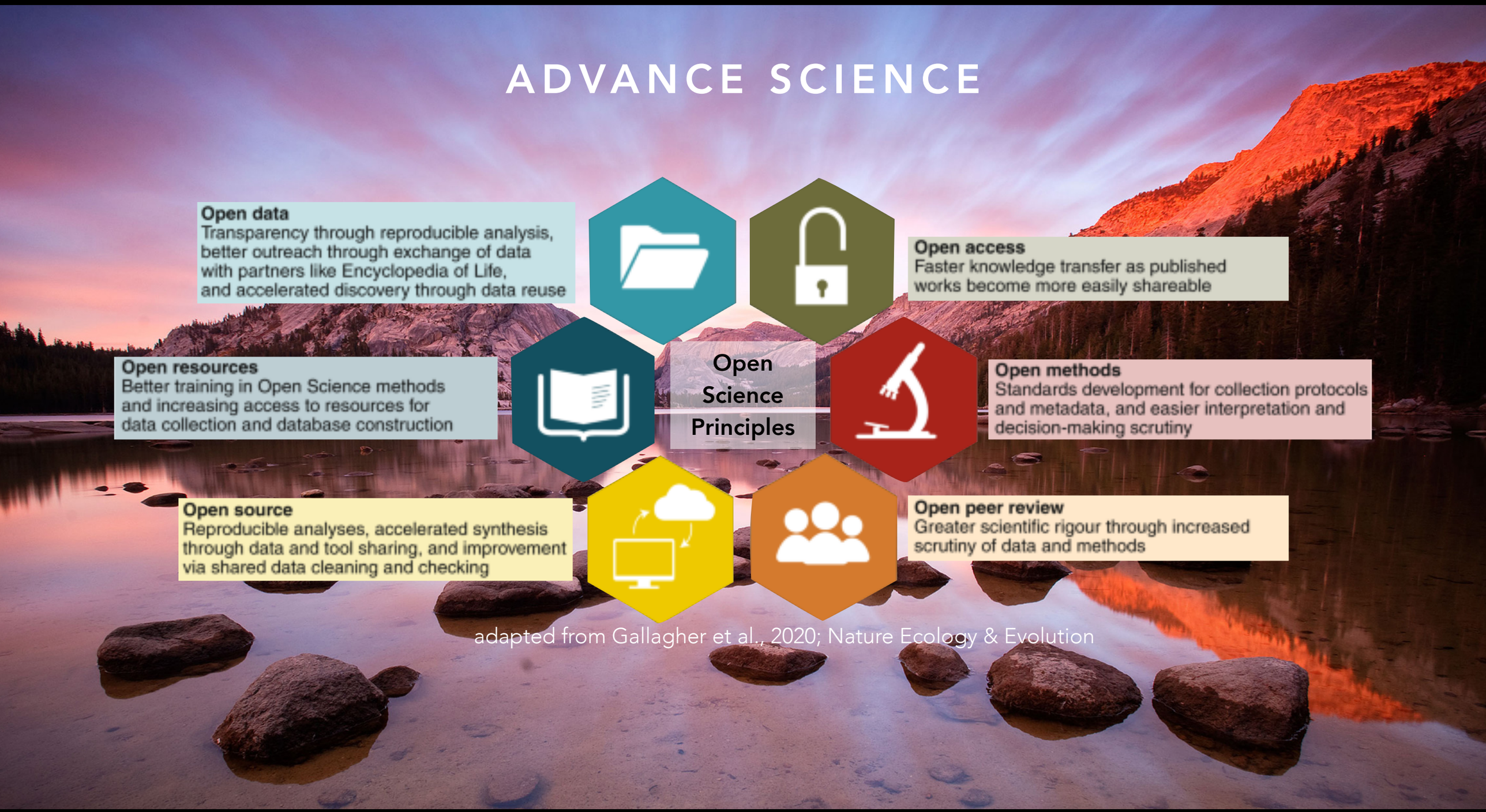
MAKING OPEN SCIENCE WORK FOR YOU: TOOLS FOR REPRODUCIBLE NEUROSCIENCE

STEPHANIE NOBLE, PHD
NEUROPRISM LAB
DEPARTMENT OF PSYCHOLOGY
DEPARTMENT OF BIOENGINEERING
CENTER FOR COGNITIVE & BRAIN SCIENCE
NORTHEASTERN UNIVERSITY

WHY DO OPEN SCIENCE?

WHY DO OPEN SCIENCE?

WHY DO OPEN SCIENCE?



adapted from Gallagher et al., 2020; Nature Ecology & Evolution

WHY DO OPEN SCIENCE?

ADVANCE SCIENCE

Open data
Transparency through reproducible analysis, better outreach through exchange of data with partners like Encyclopedia of Life, and accelerated discovery through data reuse

Open access
Faster knowledge transfer as published works become more easily shareable

Open resources
Better training in Open Science methods and increasing access to resources for data collection and database construction

Open Science Principles

Open methods
Standards development for collection protocols and metadata, and easier interpretation and decision-making scrutiny

Open source
Reproducible analyses, accelerated synthesis through data and tool sharing, and improvement via shared data cleaning and checking

Open peer review
Greater scientific rigour through increased scrutiny of data and methods

adapted from Gallagher et al., 2020; Nature Ecology & Evolution

ADVANCE YOUR CAREER

TOOLS FOR REPRODUCIBLE NEUROSCIENCE

- Quick & dirty onboarding for…

TOOLS FOR REPRODUCIBLE NEUROSCIENCE

- Quick & dirty onboarding for…
  - *Open code*
    - Focus: Version control
    - See also: Code testing, Reproducible workflows

TOOLS FOR REPRODUCIBLE NEUROSCIENCE

- Quick & dirty onboarding for…
  - *Open code*
    - Focus: Version control
    - See also: Code testing, Reproducible workflows
  - *Open data*
    - Focus: Public dataset sharing and reuse
    - See also: Data Management

TOOLS FOR REPRODUCIBLE NEUROSCIENCE

- Quick & dirty onboarding for…
  - *Open code*
    - Focus: Version control
    - See also: Code testing, Reproducible workflows
  - *Open data*
    - Focus: Public dataset sharing and reuse
    - See also: Data Management
- *Other topics in reproducible neuroscience*
  - Open publishing, Open community

TOOLS FOR REPRODUCIBLE NEUROSCIENCE

- Quick & dirty onboarding for…
  - *Open code*
    - Focus: Version control
    - See also: Code testing, Reproducible workflows
  - *Open data*
    - Focus: Public dataset sharing and reuse
    - See also: Data Management
  - *Other topics in reproducible neuroscience*
    - Open publishing, Open community

- Guiding stars
  - Why bother?
  - Quick Start (Fastest + easiest way to get started)
  - Common bottlenecks

OPEN CODE: VERSION CONTROL

track + manage changes to code

OPEN CODE: VERSION CONTROL

Why bother?

track + manage changes to code

OPEN CODE: VERSION CONTROL

Why bother?

- Save time, dissemination, recognition, & collaboration

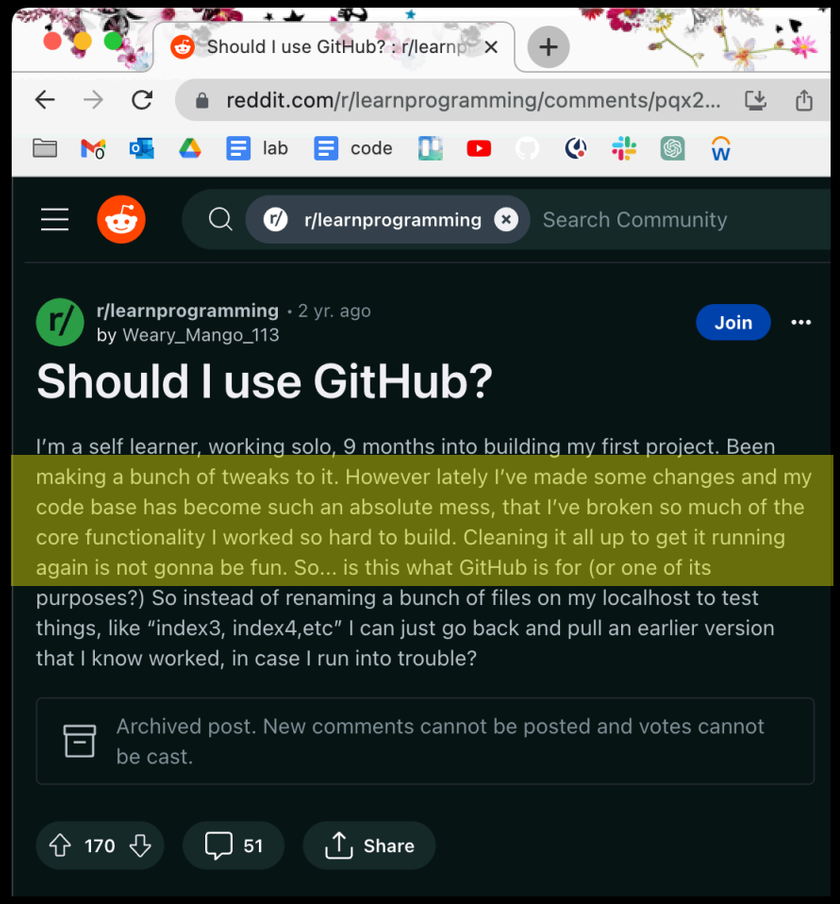
track + manage changes to code

OPEN CODE: VERSION CONTROL

Why bother?

- Save time, dissemination, recognition, & collaboration

- Most useful if your project involves…
  - Multiple “stages” of development
  - Multiple files
  - Code intended to be used by others (or future you, or you across computers, etc.)
- Contributing to an existing project
- Also to learn transferable skills: critical for many jobs



track + manage changes to code

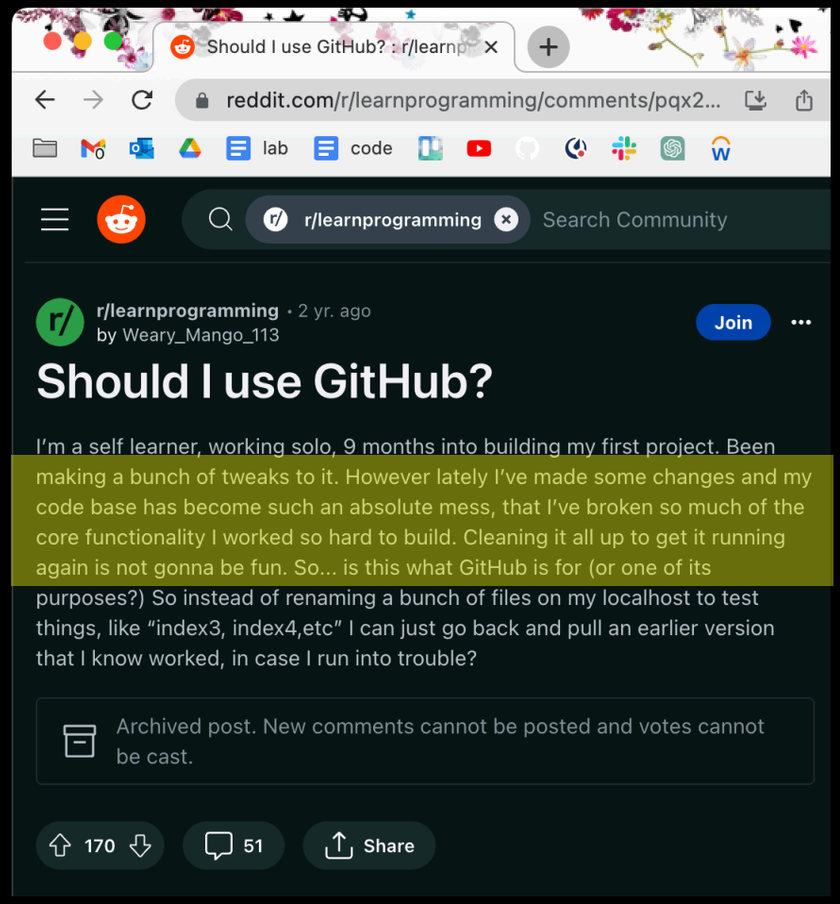
OPEN CODE: VERSION CONTROL

Why bother?

- Save time, dissemination, recognition, & collaboration

- Most useful if your project involves…
  - Multiple “stages” of development
  - Multiple files
  - Code intended to be used by others (or future you, or you across computers, etc.)
  - Contributing to an existing project
  - Also to learn transferable skills: critical for many jobs

- Less critical if your project involves…
  - A single file
  - A one-off quick & dirty test (but you’d be surprised…)



track + manage changes to code

OPEN CODE: VERSION CONTROL

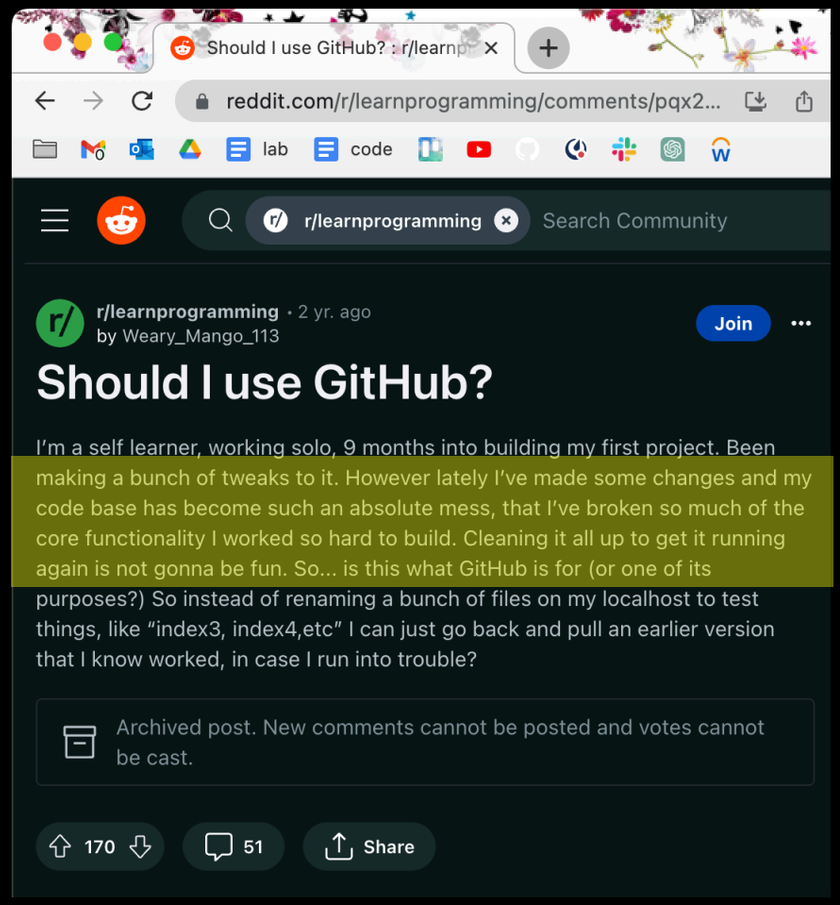
Why bother?

- Save time, dissemination, recognition, & collaboration

- Most useful if your project involves…
  - Multiple “stages” of development
  - Multiple files
  - Code intended to be used by others (or future you, or you across computers, etc.)
  - Contributing to an existing project
  - Also to learn transferable skills: critical for many jobs

- Less critical if your project involves…
  - A single file
  - A one-off quick & dirty test (but you’d be surprised…)

- TLDR: Put in some time at the beginning to save lots of time / effort / heartache later



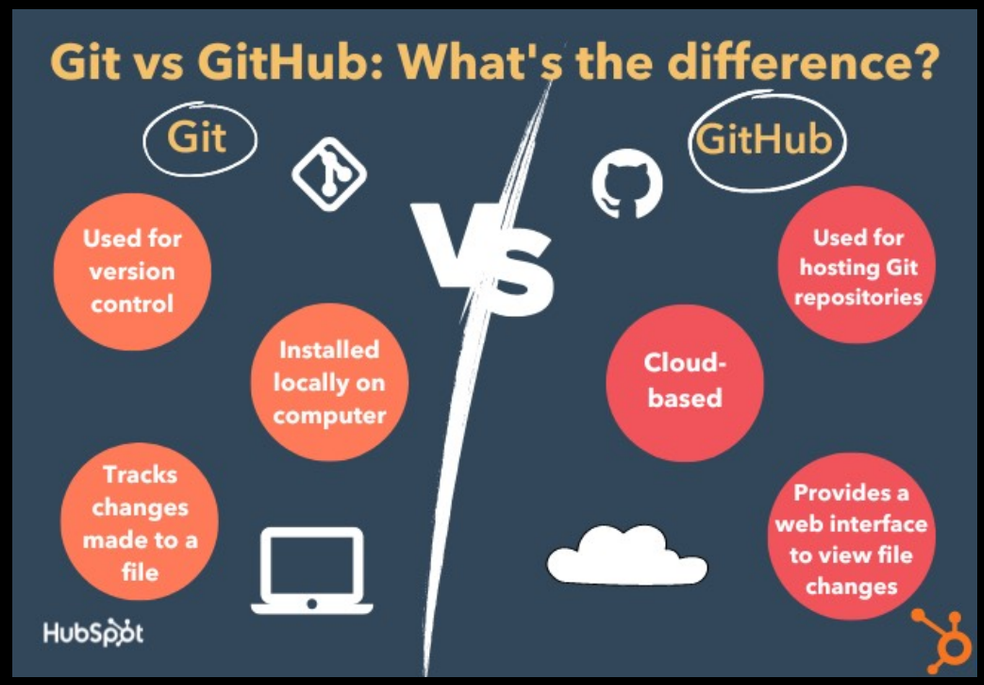
VERSION CONTROL: THE QUICK & DIRTY GUIDE

Quick Start

VERSION CONTROL: THE QUICK & DIRTY GUIDE

Quick Start

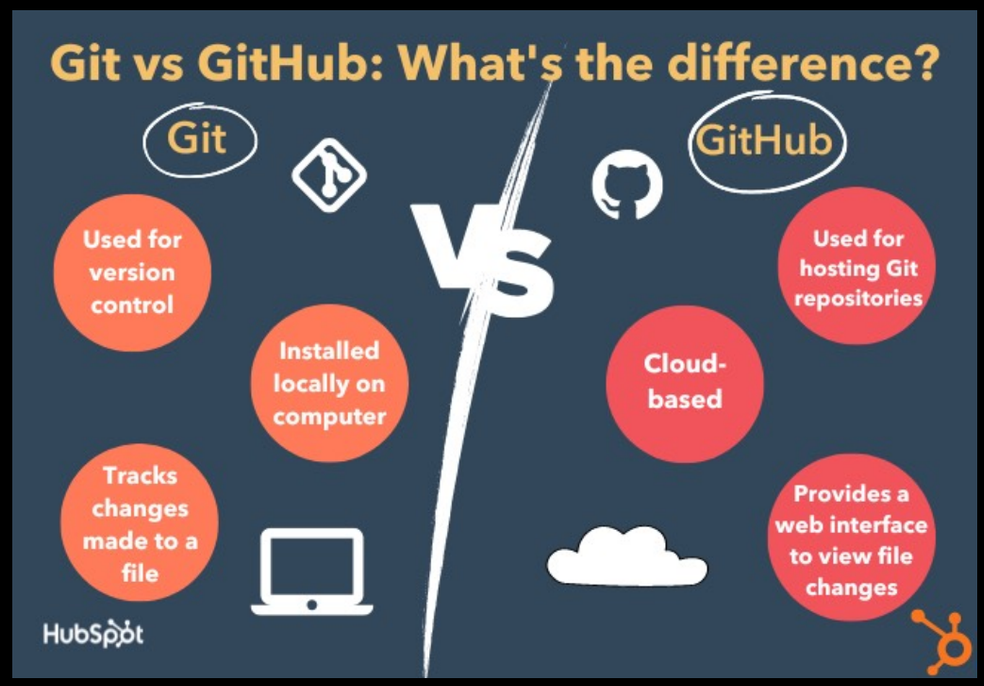
- First, some terminology:
  - **Git** ~ commands
    - lots of Git-specific terminology
  - **GitHub** ~ online storage



VERSION CONTROL: THE QUICK & DIRTY GUIDE

Quick Start

- First, some terminology:
  - **Git** ~ commands
    - lots of Git-specific terminology
  - **GitHub** ~ online storage



- Demo platform: **GitHub website**
  - Great for quick & dirty but has limited functionality
  - Other options (easy->complex): GitHub Desktop, IDE (editor) + extensions, command line, etc.

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Your first project

- Step 1. Make a GitHub account
- Step 2. Make a new repository
- Step 3. Create a branch
- Step 4. Make + commit changes

https://docs.github.com/en/get-started/quickstart/hello-world

VERSION CONTROL:
A QUICK & DIRTY GUIDE

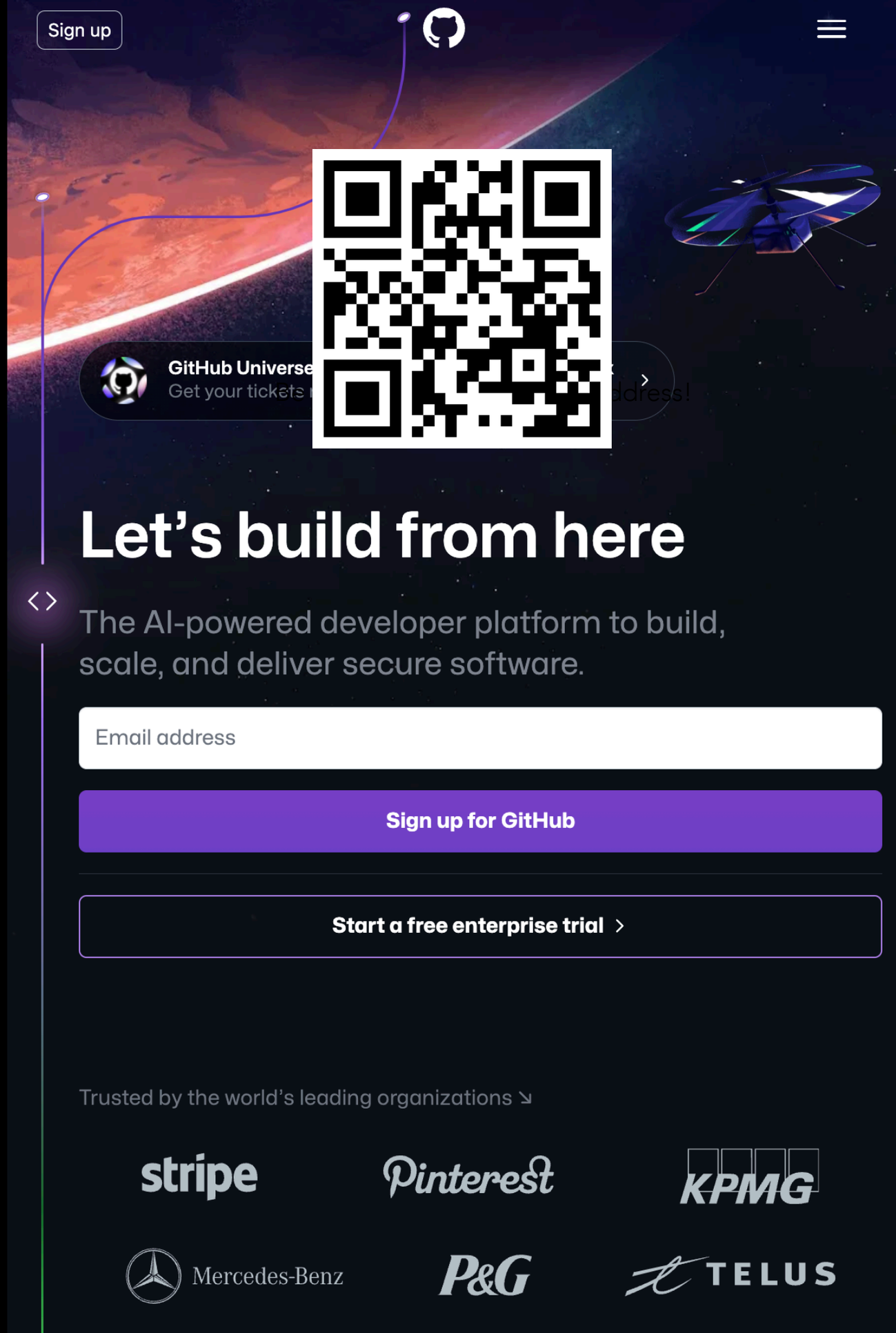
Quick Start: Your first project

- **Step 1. Make a GitHub account**
  (surprise, surprise)

- Step 2. Make a new repository

- Step 3. Create a branch

- Step 4. Make + commit changes



VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Your first project

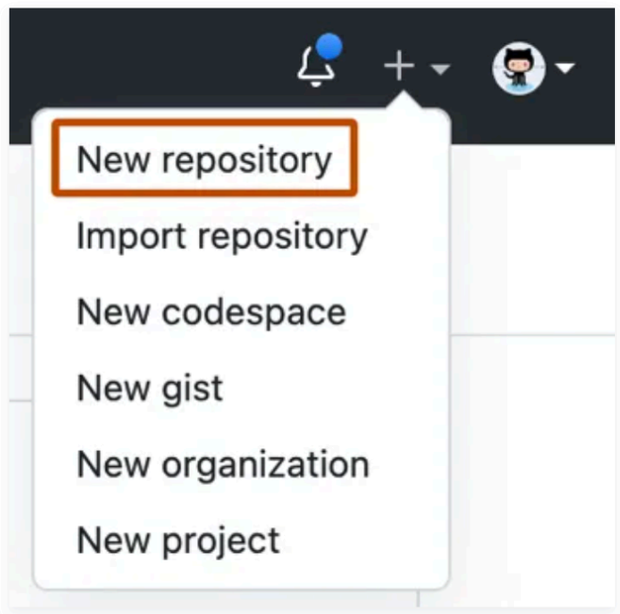
- Step 1. Make a GitHub account
  (surprise, surprise)

- **Step 2. Make a new repository** ← project

- Step 3. Create a branch

- Step 4. Make + commit changes

1. In the upper-right corner of any page, use the + drop-down menu, and select **New repository**.



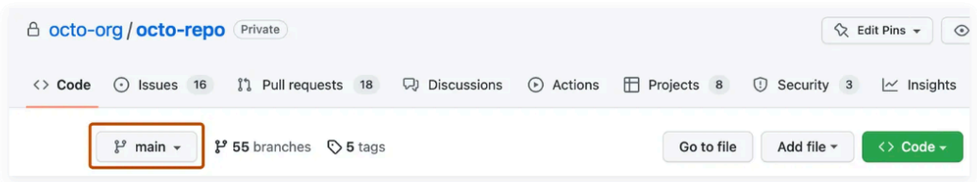
2. In the "Repository name" box, type `hello-world`.
3. In the "Description" box, type a short description.
4. Select whether your repository will be **Public** or **Private**.
5. Select **Add a README file**.
6. Click **Create repository**.

VERSION CONTROL:
A QUICK & DIRTY GUIDE

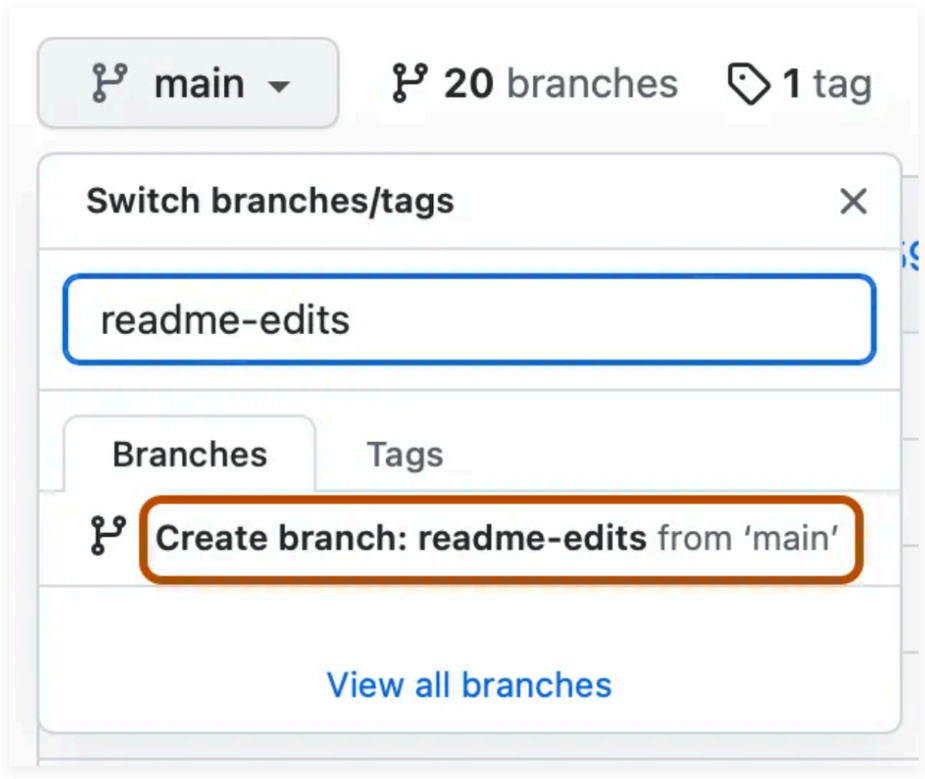
Quick Start: Your first project

- Step 1. Make a GitHub account
  (surprise, surprise)
- Step 2. Make a new repository ← project
- **Step 3. Create a branch** ↩
  version; useful to separate "development" from "main" (production-ready) branches
- Step 4. Make + commit changes

1. Click the **Code** tab of your `hello-world` repository.
2. Above the file list, click the dropdown menu that says **main**.



3. Type a branch name, `readme-edits`, into the text box.
4. Click **Create branch: readme-edits from main**.

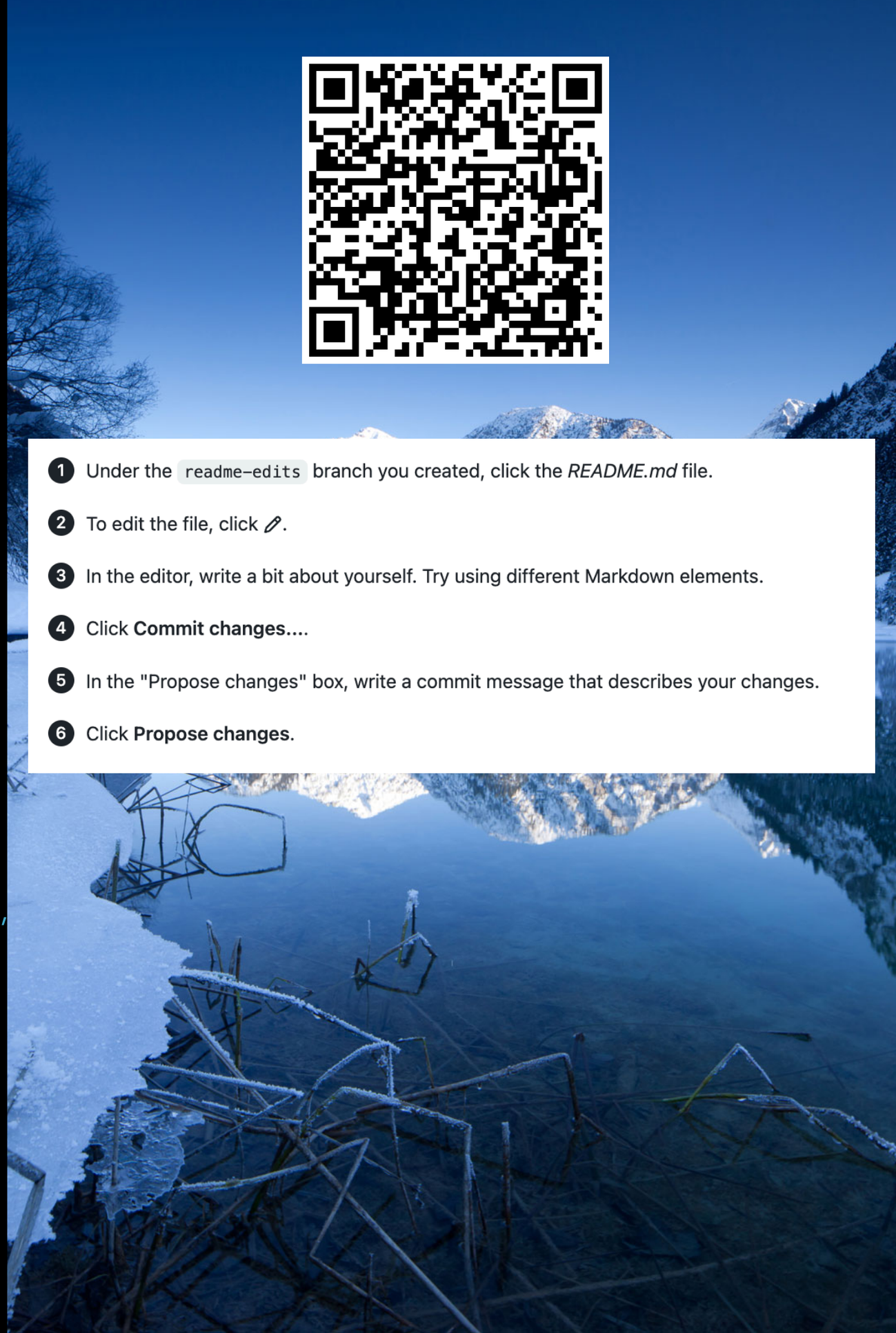


Now you have two branches, `main` and `readme-edits`. Right now, they look exactly the same. Next you'll add changes to the new branch.

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Your first project

- Step 1. Make a GitHub account
  (surprise, surprise)
- Step 2. Make a new repository
  ← project
- Step 3. Create a branch
  ↖ version; useful to separate "development" from "main" (production-ready) branches
- **Step 4. Make + commit changes**
  ↙ save + explain changes, like a checkpoint

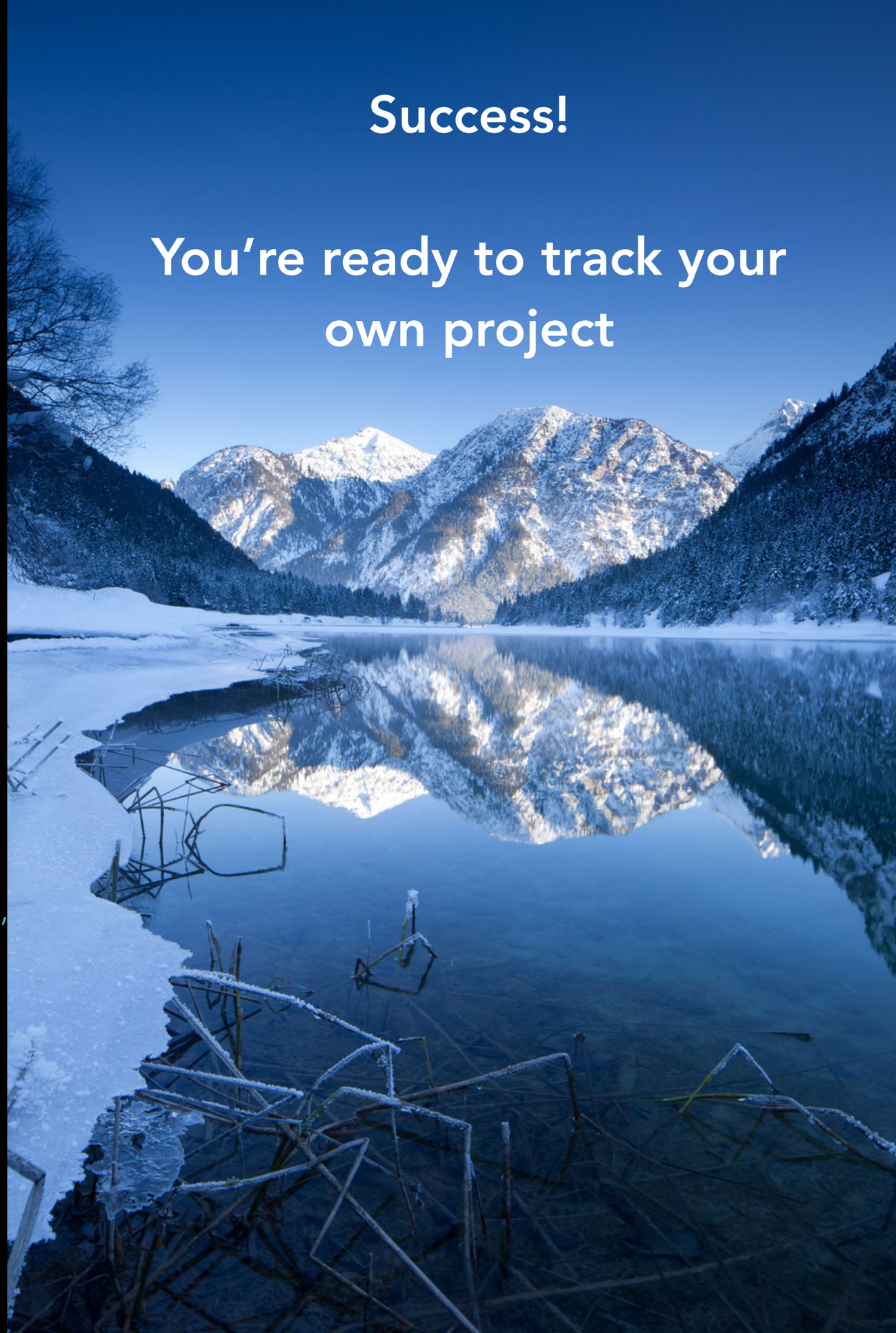


VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Your first project

- Step 1. Make a GitHub account
  (surprise, surprise)
- Step 2. Make a new repository
  project
- Step 3. Create a branch
  version; useful to separate "development" from "main" (production-ready) branches
- Step 4. Make + commit changes
  save + explain changes, like a checkpoint

For your very first project, you can definitely stop here. Get comfortable getting your code out there!



VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Your first project

- Step 1. Make a GitHub account
  (surprise, surprise)
- Step 2. Make a new repository
  project
- Step 3. Create a branch
  version; useful to separate "development" from "main" (production-ready) branches
- Step 4. Make + commit changes
  save + explain changes, like a checkpoint

- **Step 5. Make a pull request**
  request to merge dev't branch into main
- Step 6. Merge pull request

1. Click the **Pull requests** tab of your `hello-world` repository.
2. Click **New pull request**
3. In the **Example Comparisons** box, select the branch you made, `readme-edits`, to compare with `main` (the original).
4. Look over your changes in the diffs on the Compare page, make sure they're what you want to submit.

Showing 1 changed file with 3 additions and 3 deletions. Split Unified

6 README.md

```
@@ -1,3 +1,3 @@
- # test-area-2
- edit1
- edit2
+ # About me
+
+ My name is Mona Lisa.
```

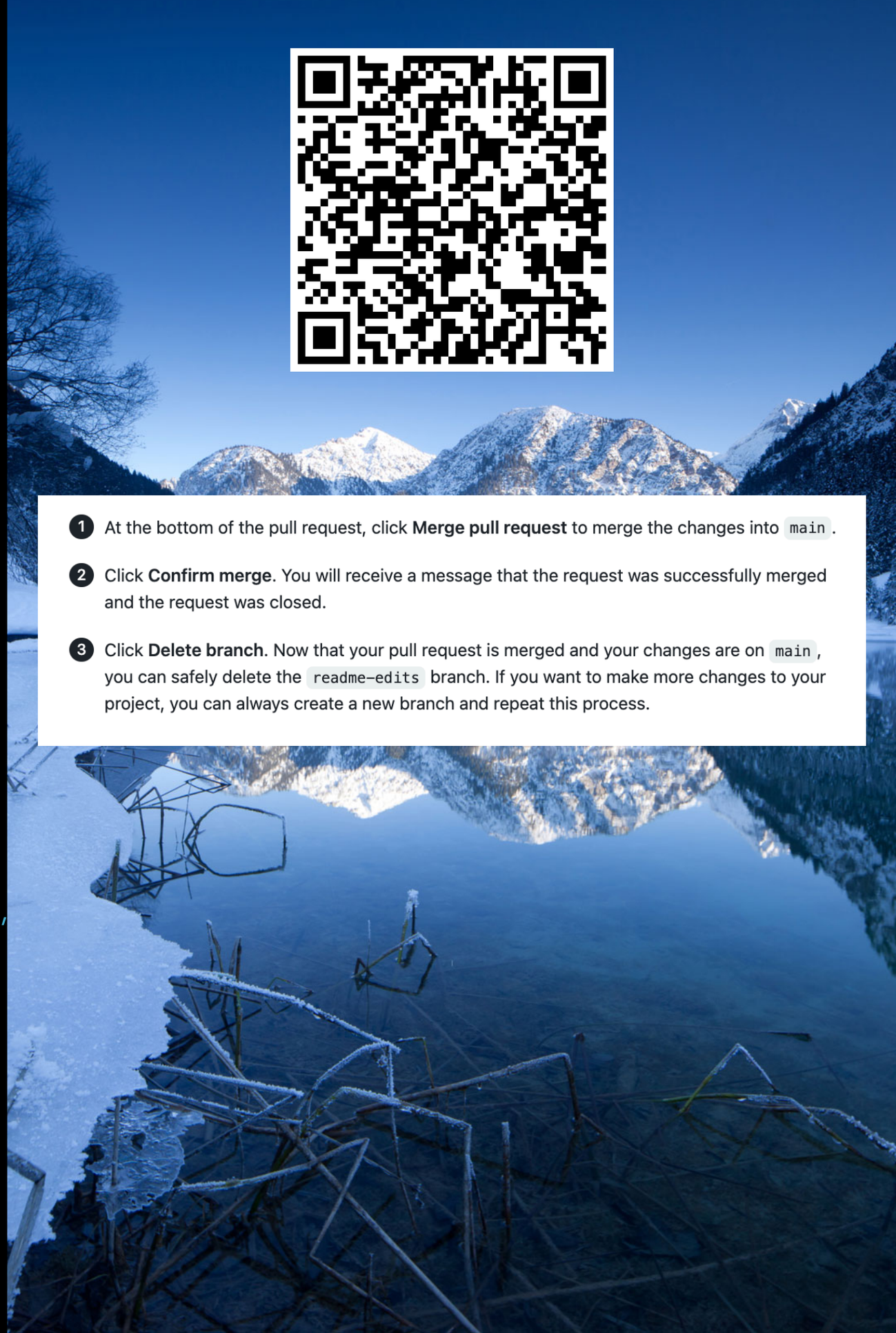
5. Click **Create pull request**.
6. Give your pull request a title and write a brief description of your changes. You can include emojis and drag and drop images and gifs.
7. Optionally, to the right of your title and description, click the ⚙ next to **Reviewers**, **Assignees**, **Labels**, **Projects**, or **Milestone** to add any of these options to your pull request. You do not need to add any yet, but these options offer different ways to collaborate using pull requests. For more information, see "About pull requests."
8. Click **Create pull request**.

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Your first project

- Step 1. Make a GitHub account
  (surprise, surprise)
- Step 2. Make a new repository
  ← project
- Step 3. Create a branch
  ↖ version; useful to separate "development" from "main" (production-ready) branches
- Step 4. Make + commit changes
  ↑ save + explain changes, like a checkpoint

- Step 5. Make a pull request
  ↑ request to merge dev't branch into main
- **Step 6. Merge pull request**
  ↑ accept request, resolving conflicts if necessary



VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Your first project

- Step 1. Make a GitHub account
  (surprise, surprise)
- Step 2. Make a new repository
  ← project
- Step 3. Create a branch
  ↖ version; useful to separate "development" from "main" (production-ready) branches
- Step 4. Make + commit changes
  ↑ save + explain changes, like a checkpoint

- Step 5. Make a pull request
  ↑ request to merge dev't branch into main
- **Step 6. Merge pull request**
  ↑ accept request, resolving conflicts if necessary

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Your first project

- Step 1. Make a GitHub account
- Step 2. Make a new repository
- Step 3. Create a branch
- Step 4. Make + commit changes

- Step 5. Make a pull request
- Step 6. Merge pull request

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Contributing to a project
when you're not on the core team

- Step 1. Make a GitHub account
- Step 2. Make a new repository
- Step 3. Create a branch
- Step 4. Make + commit changes

- Step 5. Make a pull request
- Step 6. Merge pull request

https://docs.github.com/en/get-started/quickstart/contributing-to-projects

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Contributing to a project
when you're not on the core team

- Step 1. Make a GitHub account

copy to your account

Fork an existing

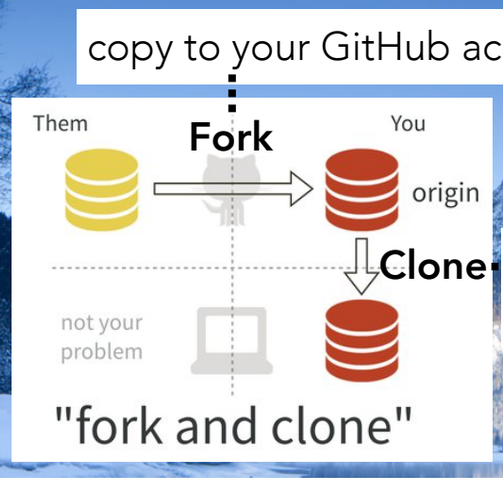
- **Step 2. ~~Make a new~~ repository**

- Step 3. Create a branch

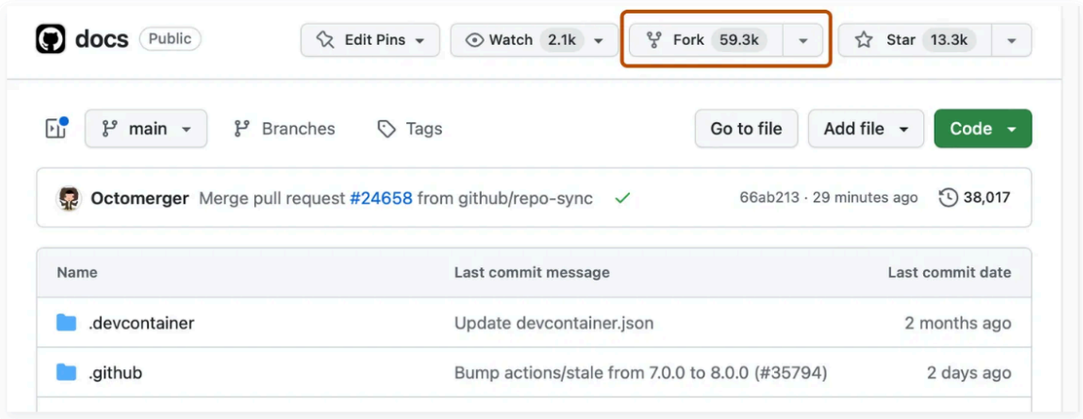
- Step 4. Make + commit changes

- Step 5. Make a pull request

- Step 6. Merge pull request



1. Navigate to the `Spoon-Knife` project at https://github.com/octocat/Spoon-Knife.
2. In the top-right corner of the page, click **Fork**.



3. Under "Owner," select the dropdown menu and click an owner for the forked repository.
4. By default, forks are named the same as their upstream repositories. Optionally, to further distinguish your fork, in the "Repository name" field, type a name.
5. Optionally, in the "Description" field, type a description of your fork.
6. Optionally, select **Copy the DEFAULT branch only**.

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Contributing to a project
when you're not on the core team

- Step 1. Make a GitHub account

copy to your account

Fork an existing

- Step 2. ~~Make a new~~ repository

- **Step 3. Create a branch**

- Step 4. Make + commit changes

- Step 5. Make a pull request

- Step 6. Merge pull request

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Contributing to a project
when you're not on the core team

- Step 1. Make a GitHub account

copy to your account

Fork an existing

- Step 2. ~~Make a new~~ repository

- Step 3. Create a branch

- **Step 4. Make + commit changes**

- Step 5. Make a pull request

- Step 6. Merge pull request

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Contributing to a project
when you're not on the core team

- Step 1. Make a GitHub account

copy to your account

Fork an existing

- Step 2. ~~Make a new~~ repository

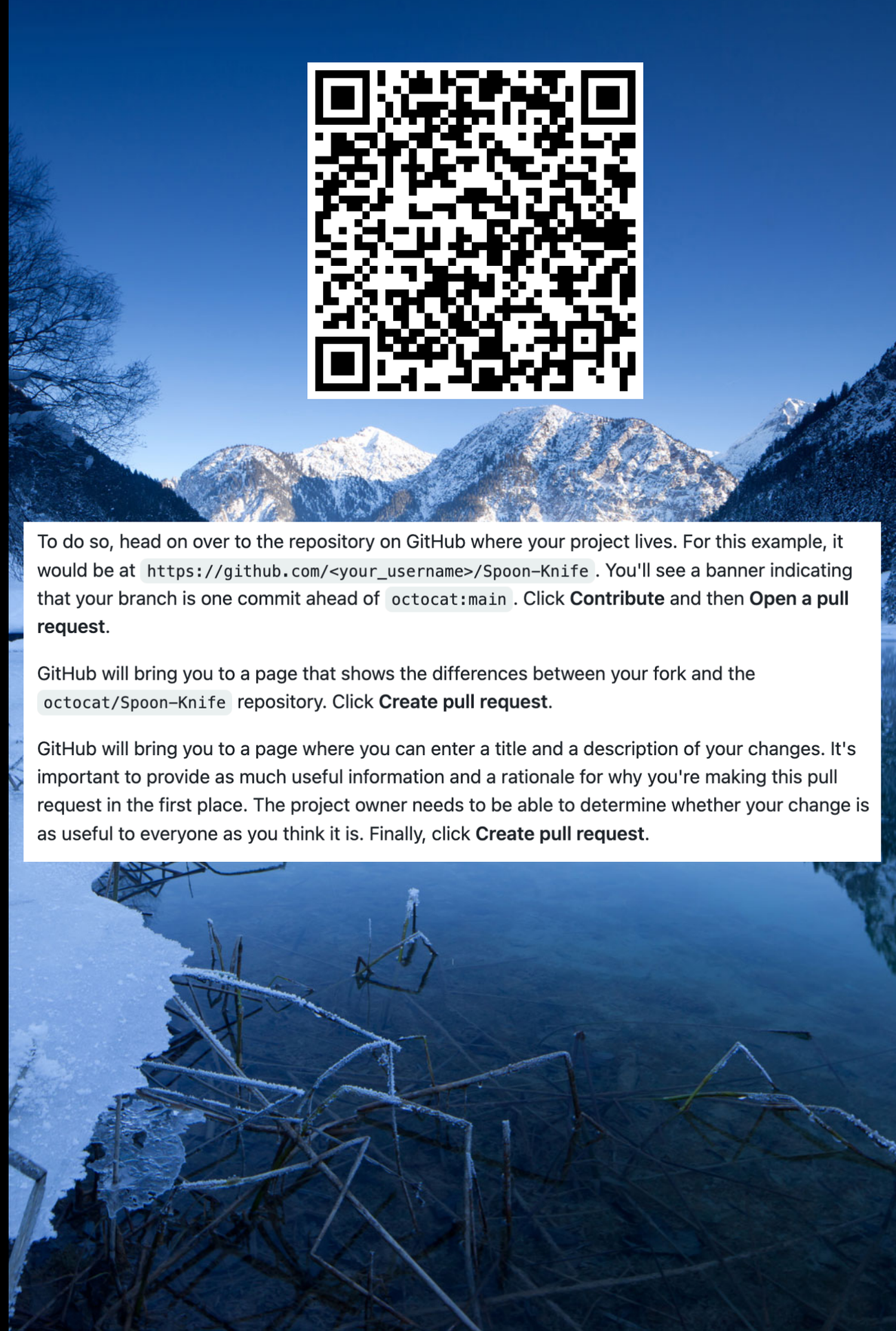
- Step 3. Create a branch

- Step 4. Make + commit changes

- **Step 5. Make a pull request**

  This time, between your fork and the original repo

- Step 6. Merge pull request



VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Contributing to a project
when you're not on the core team

- Step 1. Make a GitHub account

copy to your account

Fork an existing

- Step 2. ~~Make a new~~ repository

- Step 3. Create a branch

- Step 4. Make + commit changes

- Step 5. Make a pull request

This time, between your fork and the original repo

- **Step 6. ~~Merge pull request~~**

Manage feedback

The ball is now in their court, but you can help!

VERSION CONTROL:
A QUICK & DIRTY GUIDE

Quick Start: Contributing to a project
when you're not on the core team

- Step 1. Make a GitHub account

copy to your account

Fork an existing

- Step 2. ~~Make a new~~ repository

- Step 3. Create a branch

- Step 4. Make + commit changes

- Step 5. Make a pull request

This time, between your fork and the original repo

- **Step 6. ~~Merge pull request~~**

Manage feedback

The ball is now in their court, but you can help!

VERSION CONTROL: THE QUICK & DIRTY GUIDE

Common bottlenecks

VERSION CONTROL: THE QUICK & DIRTY GUIDE

Common bottlenecks

- Using commands you don't understand can be dangerous
  - Example dangerous commands: *git reset*, *git checkout -f* (force), *git push -f*
    - Overwrite files you're working on locally, mess up the repo, destroy records, etc.
  - **Solution:** Learn about a command before you use it and think twice if it's described as dangerous. If you are desperate and no one else is working with you, manually make a local copy of your work first. Don't do this at all on team projects.

VERSION CONTROL: THE QUICK & DIRTY GUIDE

Common bottlenecks

- Using commands you don't understand can be dangerous
  - Example dangerous commands: *git reset*, *git checkout -f* (force), *git push -f*
    - Overwrite files you're working on locally, mess up the repo, destroy records, etc.
  - **Solution:** Learn about a command before you use it and think twice if it's described as dangerous. If you are desperate and no one else is working with you, manually make a local copy of your work first. Don't do this at all on team projects.

- Merging can be messy as your project becomes more complicated
  - **Solution:** integrating branches is an art. For an actively developing project, try to regularly keep development branches up-to-date with the latest version of the master

VERSION CONTROL: THE QUICK & DIRTY GUIDE

Common bottlenecks

- Using commands you don't understand can be dangerous
  - Example dangerous commands: *git reset*, *git checkout -f* (force), *git push -f*
    - Overwrite files you're working on locally, mess up the repo, destroy records, etc.
  - **Solution:** Learn about a command before you use it and think twice if it's described as dangerous. If you are desperate and no one else is working with you, manually make a local copy of your work first. Don't do this at all on team projects.

- Merging can be messy as your project becomes more complicated
  - **Solution:** integrating branches is an art. For an actively developing project, try to regularly keep development branches up-to-date with the latest version of the master

- **General advice:** Don't worry too much. Commit often, learn as you go along, and be encouraged that using git means you have lots of new (safe) options to undo mistakes (https://ohshitgit.com/)

OPEN CODE: OTHER RESOURCES

More Git Resources

OPEN CODE: OTHER RESOURCES

More Git Resources

- 5 minute Git (GitHub Desktop): https://alicelepissier.com/git-tutorial/slides.html#1
- Interactive intro to command line Git: https://learngitbranching.js.org/
- Deeper dive into command line Git (OHBM OSSIG): https://ohbm.github.io/hackathon2021/traintrack/

OPEN CODE: OTHER RESOURCES

More Git Resources

- 5 minute Git (GitHub Desktop): https://alicelepissier.com/git-tutorial/slides.html#1
- Interactive intro to command line Git: https://learngitbranching.js.org/
- Deeper dive into command line Git (OHBM OSSIG): https://ohbm.github.io/hackathon2021/traintrack/

Related topics

- **Code testing (OHBM OSSIG):** https://www.youtube.com/watch?v=gfPP2pQ8Rms&ab_channel=OHBMOpenScienceSIG
  - Benefit: quickly know whether code is accurate or a you broke something (an expensive lesson)
  - Quick recommendation: Can be useful for most people. Take a look.
- **Reproducible workflows (OHBM OSSIG):** https://www.youtube.com/watch?v=tk2eZSrM8oA&ab_channel=OHBMOpenScienceSIG ; https://brainlife.io/about/
  - Benefit: Maximize others' ability to reproduce (version control, testing, coding best practices, environments/containers, etc.)
  - Quick recommendation: More useful for intermediate / expert applications

OPEN DATA: PUBLIC DATASET SHARING & REUSE

OPEN DATA: PUBLIC DATASET SHARING & REUSE

SHARING DATA PUBLICLY

Why bother?

SHARING DATA PUBLICLY

Why bother?

- Dissemination, recognition (citations), & collaboration
  - You won't get scooped, but many people release with their first publication
- NIH wants you to (cf. Data Management & Sharing Plan)
- Trend towards sharing full statistical maps
- Reusable for others and you, too
- Bonus: BIDS apps work with BIDS data (fMRIPrep, MRIQC)

SHARING DATA PUBLICLY

Why bother?

- Dissemination, recognition (citations), & collaboration
  - You won't get scooped, but many people release with their first publication
- NIH wants you to (cf. Data Management & Sharing Plan)
- Trend towards sharing full statistical maps
- Reusable for others and you, too
- Bonus: BIDS apps work with BIDS data (fMRIPrep, MRIQC)

- Common bottlenecks:
  - Some medical institutions won't let you
  - As usual, involves some effort (but worth the benefits)

SHARING DATA PUBLICLY

nature human behaviour

Perspective | Published: 07 December 2020

A hitchhiker's guide to working with large, open-source neuroimaging datasets

Corey Horien ✉, Stephanie Noble, Abigail S. Greene, Kangjoo Lee, Daniel S. Barron, Siyuan Gao, David O'Connor, Mehraveh Salehi, Javid Dadashkarimi, Xilin Shen, Evelyn M. R. Lake, R. Todd Constable & Dustin Scheinost ✉

Nature Human Behaviour **5**, 185–193 (2021) | Cite this article

4328 Accesses | **23** Citations | **83** Altmetric | Metrics

Abstract

Large datasets that enable researchers to perform investigations with unprecedented rigor are growing increasingly common in neuroimaging. Due to the simultaneous increasing popularity of open science, these state-of-the-art datasets are more accessible than ever to researchers around the world. While analysis of these samples has pushed the field forward, they pose a new set of challenges that might cause difficulties for novice users. Here we offer practical tips for working with large datasets from the end-user's perspective. We cover all aspects of the data lifecycle: from what to consider when downloading and storing the data to

SHARING DATA PUBLICLY

nature human behaviour

Perspective | Published: 07 December 2020

A hitchhiker's guide to working with large, open-source neuroimaging datasets

Corey Horien, Stephanie Noble, Abigail S. Greene, Kangjoo Lee, Daniel S. Barron, Siyuan Gao, David O'Connor, Mehraveh Salehi, Javid Dadashkarimi, Xilin Shen, Evelyn M. R. Lake, R. Todd Constable & Dustin Scheinost

SHARING DATA PUBLICLY

nature human behaviour

Perspective | Published: 07 December 2020

A hitchhiker's guide to working with large, open-source neuroimaging datasets

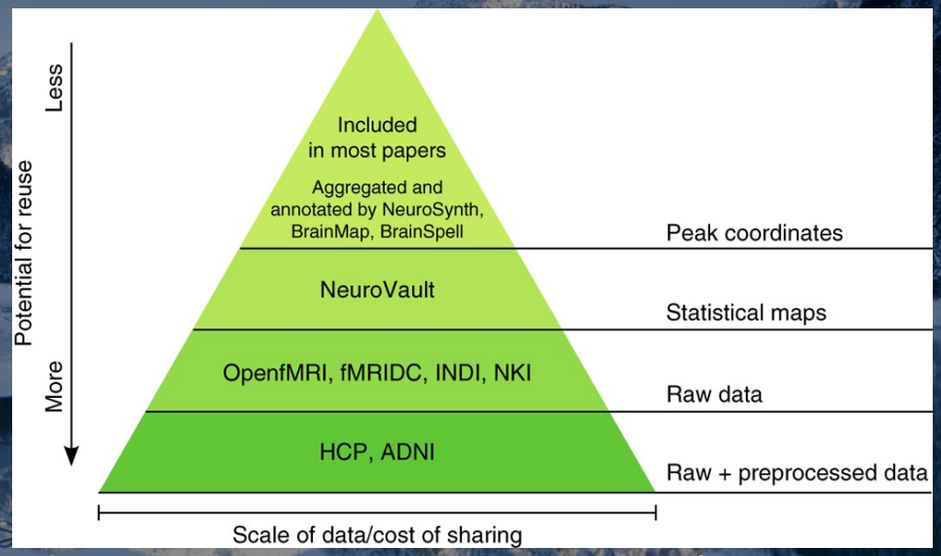
Corey Horien, Stephanie Noble, Abigail S. Greene, Kangjoo Lee, Daniel S. Barron, Siyuan Gao, David O'Connor, Mehraveh Salehi, Javid Dadashkarimi, Xilin Shen, Evelyn M. R. Lake, R. Todd Constable & Dustin Scheinost

SHARING DATA PUBLICLY:
A QUICK & DIRTY GUIDE

Quick Start

- **Step 1. Determine what to share**
  Raw or preprocessed?
- Step 2. Confirm you're allowed to share
- Step 3. Find a database and sign up
- Step 5. Anonymize
- Step 6. Confirm other requirements
- Step 7. Upload!

Imagine a dataset you could contribute



https://www.nature.com/articles/nn.3818

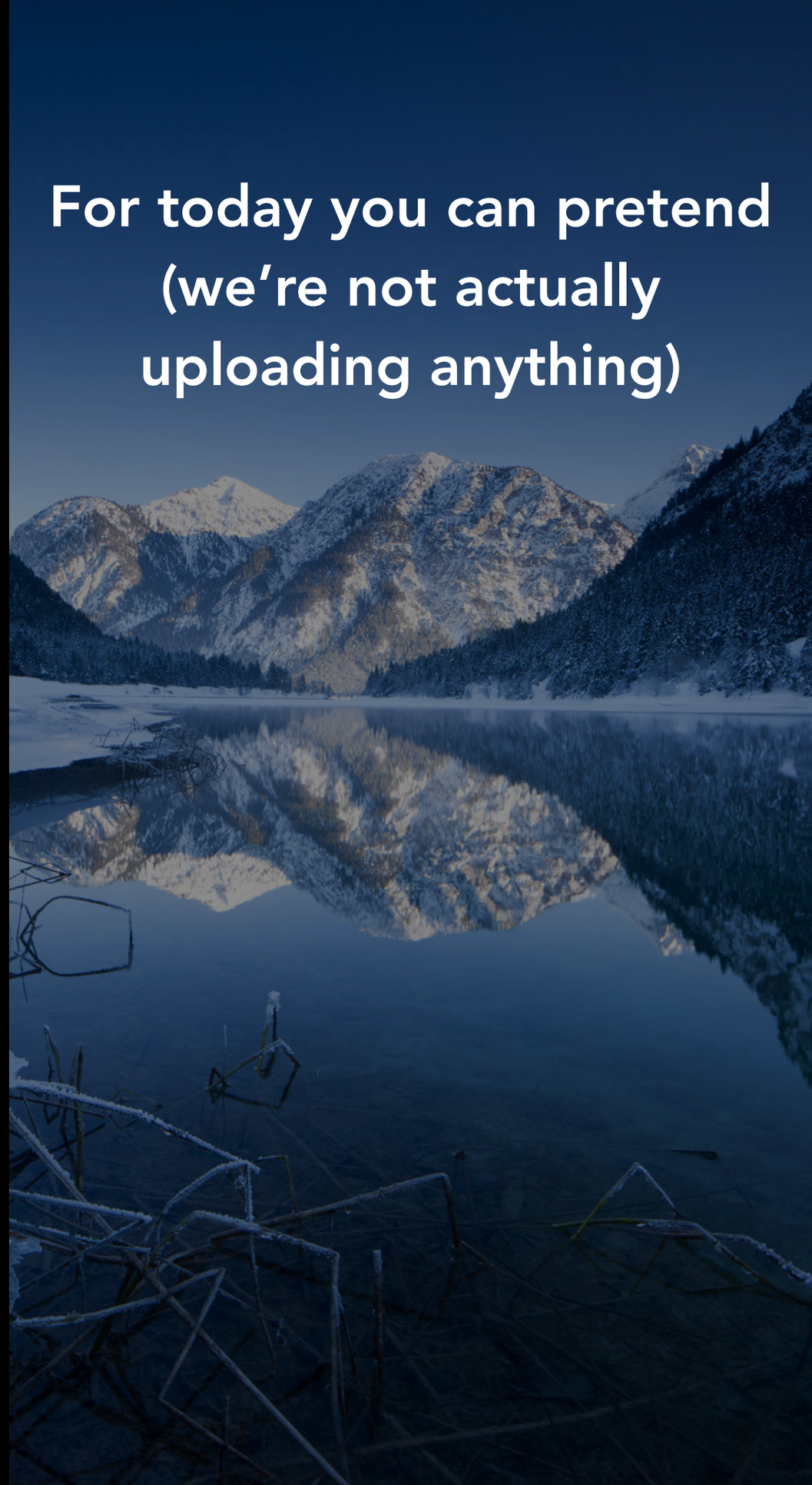
How much could you reasonably share?

SHARING DATA PUBLICLY:
A QUICK & DIRTY GUIDE

Quick Start

- Step 1. Determine what to share
  Raw or preprocessed?
- **Step 2. Confirm you're allowed to share**
  Consent, IRB, etc
- Step 3. Find a database and sign up
- Step 5. Anonymize
- Step 6. Confirm other requirements
- Step 7. Upload!

For today you can pretend (we're not actually uploading anything)



SHARING DATA PUBLICLY:
A QUICK & DIRTY GUIDE

Quick Start

- Step 1. Determine what to share
  Raw or preprocessed?
- Step 2. Confirm you're allowed to share
  Consent, IRB, etc
- **Step 3. Find a database and sign up**
  Review contribution guide to check: 1) you can share the level of data you want, & 2) you have all required data
- Step 5. Anonymize
- Step 6. Confirm other requirements
- Step 7. Upload!

Guides for Two Potential Databases:

INDI guide

http://fcon_1000.projects.nitrc.org/indi/indi_data_contribution_guide.pdf

Requires (1) T1, (1) resting state scan, age, sex, & handedness

OpenNeuro guide

https://docs.openneuro.org/user_guide.html

SHARING DATA PUBLICLY:
A QUICK & DIRTY GUIDE

Quick Start

- Step 1. Determine what to share
  Raw or preprocessed?

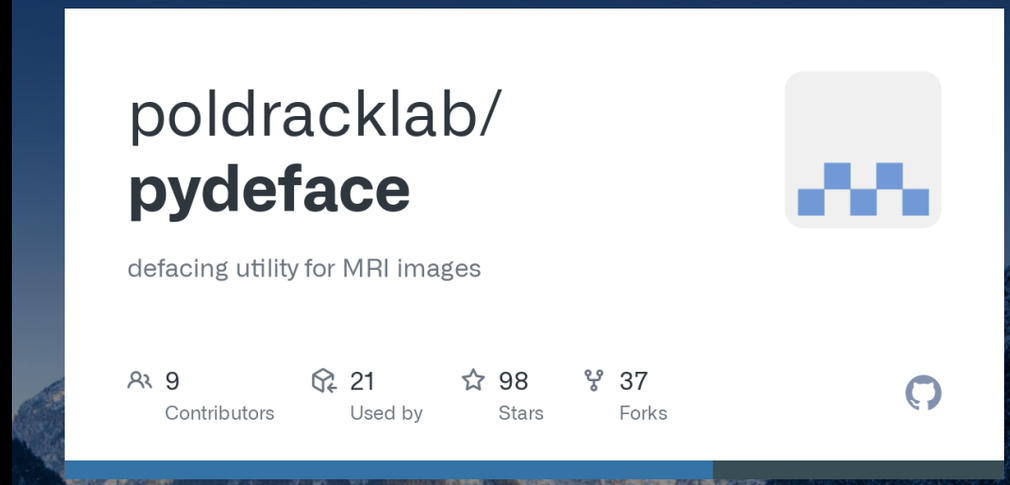
- Step 2. Confirm you're allowed to share
  Consent, IRB, etc

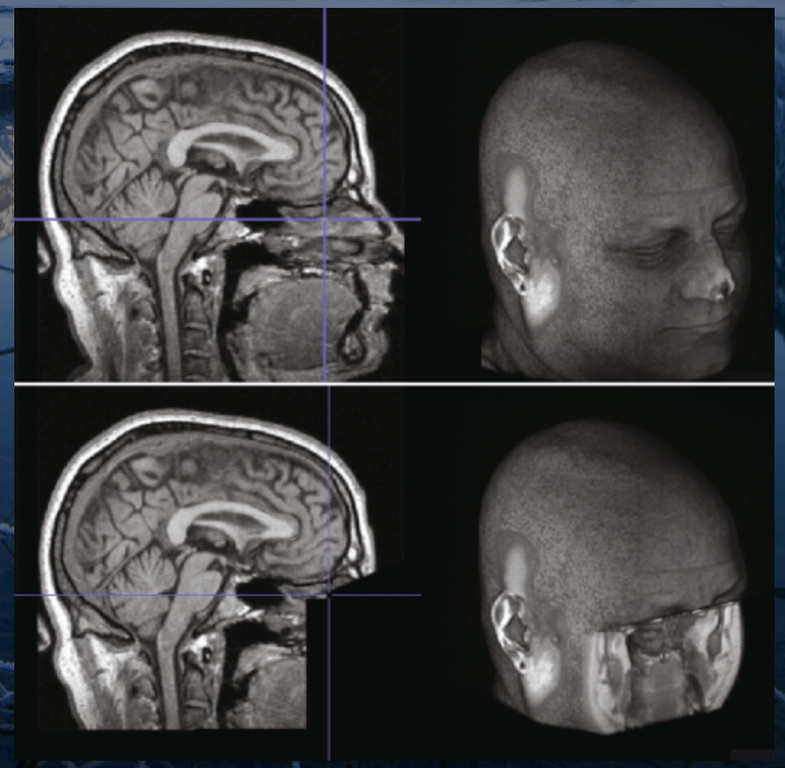
- Step 3. Find a database and sign up
  Review contribution guide to check: 1) you can share the level of data you want, & 2) you have all required data

- **Step 5. Anonymize**
  Deface structural scan, rename, remove name and DOB, etc. (see contribution guide)

- Step 6. Confirm other requirements

- Step 7. Upload!





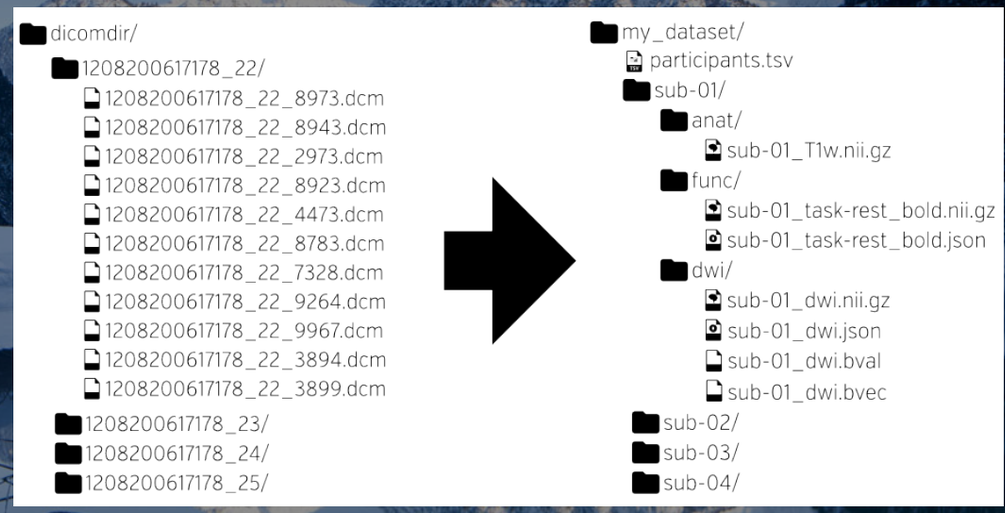
https://www.physicamedica.com/article/S1120-1797(21)00095-8/fulltext

SHARING DATA PUBLICLY:
A QUICK & DIRTY GUIDE

Quick Start

- Step 1. Determine what to share
  Raw or preprocessed?
- Step 2. Confirm you're allowed to share
  Consent, IRB, etc
- Step 3. Find a database and sign up
  Review contribution guide to check: 1) you can share the level of data you want, & 2) you have all required data
- Step 5. Anonymize
  Deface structural scan, rename, remove name and DOB, etc. (see contribution guide)
- **Step 6. Confirm other requirements**
  E.g., BIDS format (OpenNeuro), reorient (optional but desirable for INDI), metadata, description, etc.
- Step 7. Upload!

BIDS: Brain Imaging Data Structure



https://hackmd.io/@effigies/bids-derivatives-readme

SHARING DATA PUBLICLY:
A QUICK & DIRTY GUIDE

Quick Start

- Step 1. Determine what to share
  Raw or preprocessed?
- Step 2. Confirm you're allowed to share
  Consent, IRB, etc
- Step 3. Find a database and sign up
  Review contribution guide to check: 1) you can share the level of data you want, & 2) you have all required data
- Step 5. Anonymize
  Deface structural scan, rename, remove name and DOB, etc. (see contribution guide)
- Step 6. Confirm other requirements
  E.g., BIDS format (OpenNeuro), reorient (optional but desirable for INDI), metadata, description, etc.
- **Step 7. Upload!**

SHARING DATA PUBLICLY

Common bottlenecks

SHARING DATA PUBLICLY

Common bottlenecks

- Code for preparing data can sometimes be finicky, and reorganization can get annoying

SHARING DATA PUBLICLY

Common bottlenecks

- Code for preparing data can sometimes be finicky, and reorganization can get annoying
- Not a deal-breaker, just adjust time expectations accordingly

SHARING DATA PUBLICLY

Common bottlenecks

- Code for preparing data can sometimes be finicky, and reorganization can get annoying
  - Not a deal-breaker, just adjust time expectations accordingly
- May be prohibited by some medical institutions, especially VA

SHARING DATA PUBLICLY

More Data Sharing & Reuse Resources

SHARING DATA PUBLICLY

More Data Sharing & Reuse Resources

- Where to share data and sharing results - see Horien et al., 2021: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7992920/
- More considerations regarding data sharing (Open Research at the Wellcome Centre for Integrative Neuroimaging): https://open.win.ox.ac.uk/pages/open-science/community/Open-WIN-Community/docs/data/
- BIDS & BIDS Apps
  - Great guide to both: Markiewicz et al. https://hackmd.io/@effigies/bids-derivatives-readme
  - https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005209
  - https://psy6983.brainhackmtl.org/modules/bids/

SHARING DATA PUBLICLY

More Data Sharing & Reuse Resources

- Where to share data and sharing results - see Horien et al., 2021: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7992920/
- More considerations regarding data sharing (Open Research at the Wellcome Centre for Integrative Neuroimaging): https://open.win.ox.ac.uk/pages/open-science/community/Open-WIN-Community/docs/data/
- BIDS & BIDS Apps
    - Great guide to both: Markiewicz et al. https://hackmd.io/@effigies/bids-derivatives-readme
    - https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005209
    - https://psy6983.brainhackmtl.org/modules/bids/

Related topics

- Data Management (DataLad): https://www.datalad.org/
    - Benefit: record how you've changed your data
    - Expert level: moderate/high. Recommended for large projects and tracking processing.

OPEN DATA: PUBLIC DATASET SHARING & REUSE

OPEN DATA: PUBLIC DATASET SHARING & REUSE

USING PUBLICLY AVAILABLE DATA

Why bother?

USING PUBLICLY AVAILABLE DATA

Why bother?

- Save time, $$, effort
  - Feasibility in grants
- Larger samples than most labs can collect
  - Facilitates rigorous & reproducible science
  - Many of the most important recent papers in NI have relied on open datasets
- Learn transferable skills

USING PUBLICLY AVAILABLE DATA

Why bother?

- Save time, $$, effort
  - Feasibility in grants
- Larger samples than most labs can collect
  - Facilitates rigorous & reproducible science
  - Many of the most important recent papers in NI have relied on open datasets
- Learn transferable skills

- Common bottlenecks:
  - For large (n>500) datasets, need moderately large computational resources (most of you have access to HPC, though)
  - May not include specialized populations (though more and more populations are being released)

USING PUBLICLY AVAILABLE DATA

nature human behaviour

Perspective | Published: 07 December 2020

A hitchhiker's guide to working with large, open-source neuroimaging datasets

Corey Horien, Stephanie Noble, Abigail S. Greene, Kangjoo Lee, Daniel S. Barron, Siyuan Gao, David O'Connor, Mehraveh Salehi, Javid Dadashkarimi, Xilin Shen, Evelyn M. R. Lake, R. Todd Constable & Dustin Scheinost

USING PUBLICLY AVAILABLE DATA

nature human behaviour

Perspective | Published: 07 December 2020

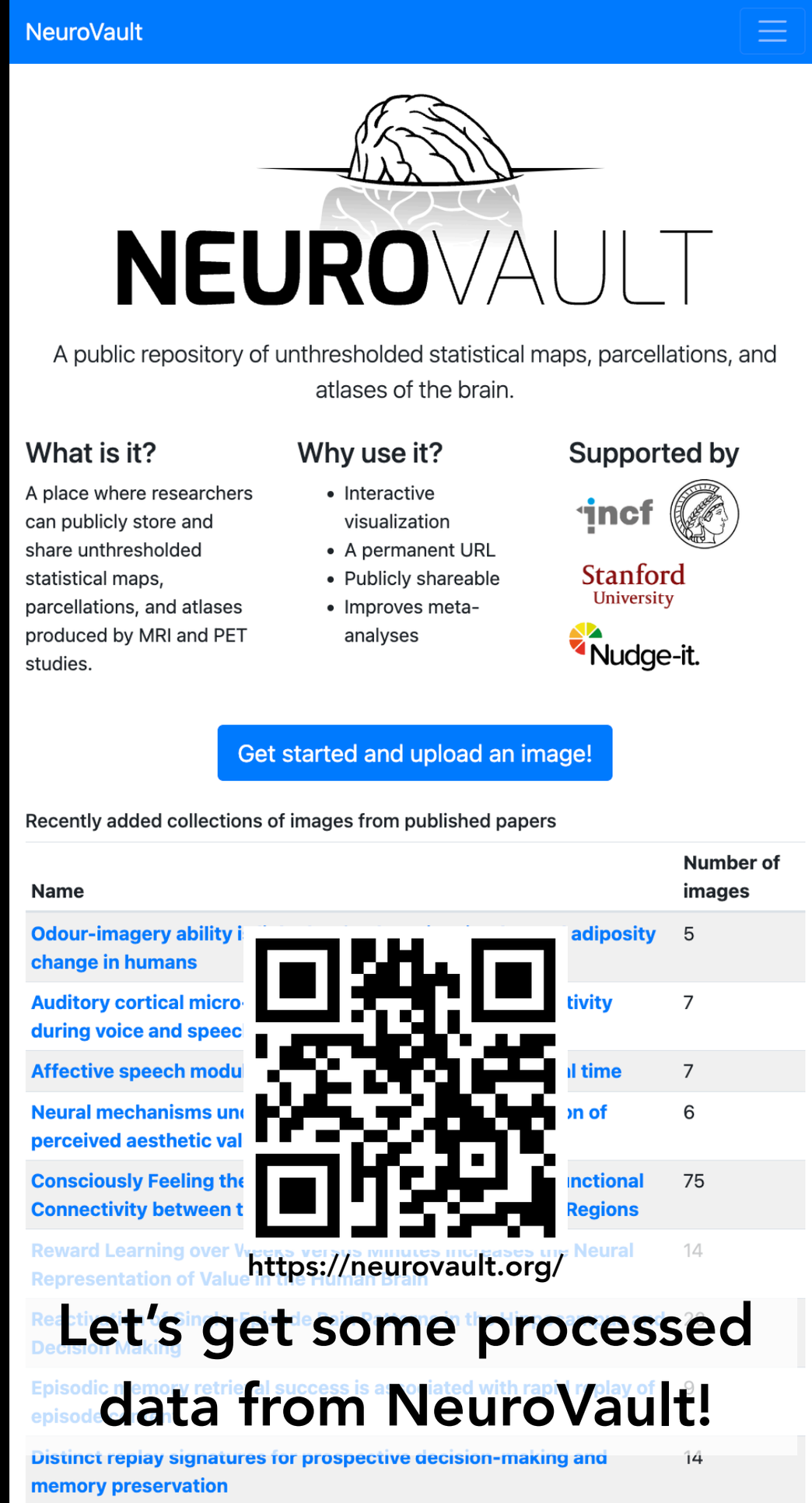
A hitchhiker's guide to working with large, open-source neuroimaging datasets

Corey Horien, Stephanie Noble, Abigail S. Greene, Kangjoo Lee, Daniel S. Barron, Siyuan Gao, David O'Connor, Mehraveh Salehi, Javid Dadashkarimi, Xilin Shen, Evelyn M. R. Lake, R. Todd Constable & Dustin Scheinost

PUBLIC DATA REUSE:
A QUICK & DIRTY GUIDE

Quick Start

- **Step 1. Determine what data you want**
  What do you want to do? Which databases have that data?
- Step 2. Confirm ethical approval
- Step 3. Confirm storage & computing
- Step 4. If required: create account, sign data use agreement, etc.
- Step 5. Download some data!



https://neurovault.org/

Let's get some processed data from NeuroVault!

PUBLIC DATA REUSE:
A QUICK & DIRTY GUIDE

Quick Start

- Step 1. Determine what data you want
  What do you want to do? Which databases have that data?
- **Step 2. Confirm ethical approval**
  Generally straightforward to seek IRB Exemption 4 unless data identifiable in combination
- Step 3. Confirm storage & computing
- Step 4. If required: create account, sign data use agreement, etc.
- Step 5. Download some data!

PUBLIC DATA REUSE:
A QUICK & DIRTY GUIDE

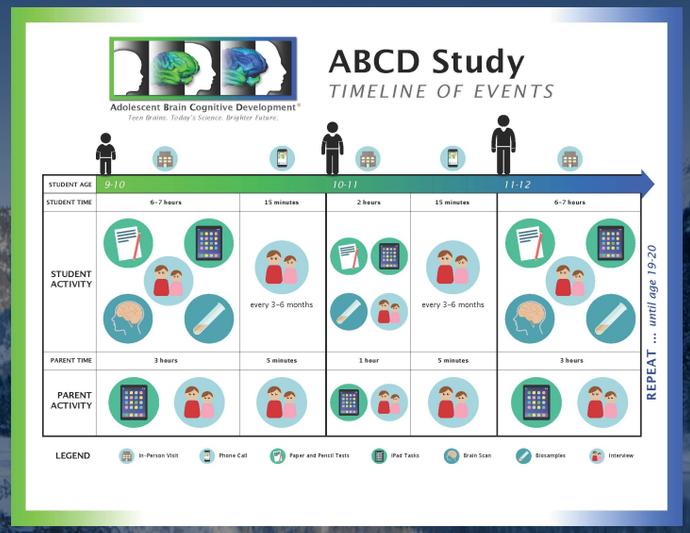
Quick Start

- Step 1. Determine what data you want
  - What do you want to do? Which databases have that data?
- Step 2. Confirm ethical approval
  - Generally straightforward to seek IRB Exemption 4 unless data identifiable in combination
- **Step 3. Confirm storage & computing**
  - Storage: Plan for additions, backups, + any intermediates
  - Computing: Time your pipeline + anticipate how to scale
- Step 4. If required: create account, sign data use agreement, etc.
- Step 5. Download some data!

Example: ABCD release 1 (~10,000 individuals)

Storage

- Raw NifTI data: ~13.5 TB
- Preprocessed connectivity matrices: ~25.6 MB
- More data released over the years



Computing

- Connectome-behavior regression <1 day on laptop
- Processing raw data >1 mo on HPC or AWS
  - Scaling up: time a couple subjects + account for parallelization
- Special considerations, e.g., HCP already on AWS so credits, fewer transfer limits, etc.

In-lab cluster < university cluster < AWS

$$

user expertise

In-lab maintenance

PUBLIC DATA REUSE:
A QUICK & DIRTY GUIDE

Quick Start

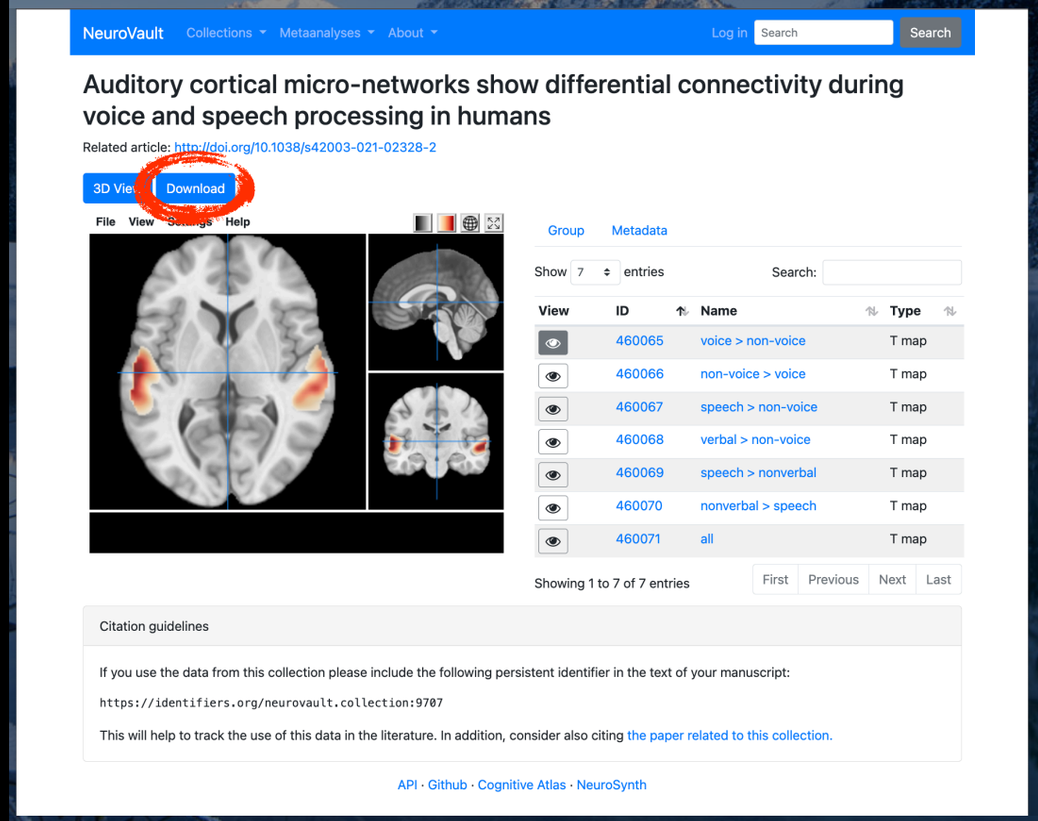
- Step 1. Determine what data you want
  What do you want to do? Which databases have that data?
- Step 2. Confirm ethical approval
  Generally straightforward to seek IRB Exemption 4 unless data identifiable in combination
- Step 3. Confirm storage & computing
  - Storage: Plan for additions, backups, + any intermediates
  - Computing: Time your pipeline + anticipate how to scale
- **Step 4. If required: create account, sign data use agreement, etc.**
- Step 5. Download some data!

PUBLIC DATA REUSE:
A QUICK & DIRTY GUIDE

Quick Start

- Step 1. Determine what data you want
  - What do you want to do? Which databases have that data?
- Step 2. Confirm ethical approval
  - Generally straightforward to seek IRB Exemption 4 unless data identifiable in combination
- Step 3. Confirm storage & computing
  - Storage: Plan for additions, backups, + any intermediates
  - Computing: Time your pipeline + anticipate how to scale
- Step 4. If required: create account, sign data use agreement, etc.
- **Step 5. Download some data!**

https://neurovault.org/collections/9810/



PUBLIC DATA REUSE

Common bottlenecks

PUBLIC DATA REUSE

Common bottlenecks

- Expertise + setup for getting very large datasets (>5,000; e.g., UKB) can be nontrivial

PUBLIC DATA REUSE

Common bottlenecks

- Expertise + setup for getting very large datasets (>5,000; e.g., UKB) can be nontrivial
- Processing large datasets (>1,000; e.g., HCP) is time-intensive (6-9 months for ABCD release 1)
  - if you're completely new, may be worth starting with preprocessed data to get a feel for it

PUBLIC DATA REUSE

Common bottlenecks

- Expertise + setup for getting very large datasets (>5,000; e.g., UKB) can be nontrivial
- Processing large datasets (>1,000; e.g., HCP) is time-intensive (6-9 months for ABCD release 1)
  - if you're completely new, may be worth starting with preprocessed data to get a feel for it
- AWS can be (accidentally) expensive—best to get an expert involved
  - Before you run anything: always test on a small scale first
  - During: check outputs periodically
  - After: double check you've closed your instance

PUBLIC DATA REUSE

PUBLIC DATA REUSE

Other Resources

- Large, publicly available datasets (Horien et al., 2021): https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7992920/
- In addition to official guides, many datasets have third-party guides, e.g.,:
    - HCP Tutorial (Andy's Brain Blog): https://andysbrainbook.readthedocs.io/en/latest/HCP/HCP_1_DownloadData.html
    - A Hitchhiker's Guide to OpenNeuro: Secondary Analysis on the Web's Largest Repository of Open Neuroimaging Data (Buckser, 2022): http://d-scholarship.pitt.edu/42175/
- High performance computing (Barch et al., 2019) https://training.incf.org/sites/default/files/CloudBasedComputer_MATRIX.pdf

PUBLIC DATA REUSE

Other Resources

- Large, publicly available datasets (Horien et al., 2021): https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7992920/
- In addition to official guides, many datasets have third-party guides, e.g.,:
    - HCP Tutorial (Andy's Brain Blog): https://andysbrainbook.readthedocs.io/en/latest/HCP/HCP_1_DownloadData.html
    - A Hitchhiker's Guide to OpenNeuro: Secondary Analysis on the Web's Largest Repository of Open Neuroimaging Data (Buckser, 2022): http://d-scholarship.pitt.edu/42175/
- High performance computing (Barch et al., 2019) https://training.incf.org/sites/default/files/CloudBasedComputer_MATRIX.pdf

Related topics

- Data Management (DataLad): https://www.datalad.org/
    - Scaling up: Managing 80TB and 15 million files from the HCP release: https://handbook.datalad.org/en/latest/usecases/HCP_dataset.html

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE:
OPEN PUBLISHING

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: OPEN PUBLISHING

- What is it
  - Open access, non-profit journals (Imaging Neuroscience! Aperture!), preprints, registered reports, non-traditional research objects, etc.

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: OPEN PUBLISHING

- What is it
  - Open access, non-profit journals (Imaging Neuroscience! Aperture!), preprints, registered reports, non-traditional research objects, etc.

- Why bother?
  - Again, dissemination, recognition (citations), & collaboration
  - Often faster, can be lower cost

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: OPEN PUBLISHING

- What is it
  - Open access, non-profit journals (Imaging Neuroscience! Aperture!), preprints, registered reports, non-traditional research objects, etc.

- Why bother?
  - Again, dissemination, recognition (citations), & collaboration
  - Often faster, can be lower cost

- Fastest + easiest way to get started
  - Post on arXiv / OSF, find community-led low fee open-access journals (Aperture, Imaging Neuroscience - fee waivers!)

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: OPEN PUBLISHING

- What is it
  - Open access, non-profit journals (Imaging Neuroscience! Aperture!), preprints, registered reports, non-traditional research objects, etc.

- Why bother?
  - Again, dissemination, recognition (citations), & collaboration
  - Often faster, can be lower cost

- Fastest + easiest way to get started
  - Post on arXiv / OSF, find community-led low fee open-access journals (Aperture, Imaging Neuroscience - fee waivers!)

- Common bottlenecks
  - Growing recognition, competing against “glam” journals
    - …but there is lots of community momentum + worth the benefits
  - Cost (depending)

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: OPEN COMMUNITY

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: OPEN COMMUNITY

- What it is
  - Actively inclusive & collaborative communities, e.g., Brainhack, OHBM Open Science Room, Neuromatch

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: OPEN COMMUNITY

- What it is
  - Actively inclusive & collaborative communities, e.g., Brainhack, OHBM Open Science Room, Neuromatch
- Why bother?
  - Network, support, training, future collaborations, etc.

Neuron

CellPress

NeuroView

Brainhack: Developing a culture of open, inclusive, community-driven neuroscience

Rémi Gau,[1,76,*] Stephanie Noble,[2,76] Katja Heuer,[3,4,76] Katherine L. Bottenhorn,[5,76] Isil P. Bilgin,[6,7,76] Yu-Fang Yang,[8,76] Julia M. Huntenburg,[9,76] Johanna M.M. Bayer,[10,11,76] Richard A.I. Bethlehem,[12,13,76] Shawn A. Rhoads,[14] Christoph Vogelbacher,[15] Valentina Borghesani,[16] Elizabeth Levitis,[17,18] Hao-Ting Wang,[19,20,21] Sofie Van Den Bossche,[22] Xenia Kobeleva,[23,24] Jon Haitz Legarreta,[25] Samuel Guay,[26] Selim Melvin Atay,[27] Gael P. Varoquaux,[28,29] Dorien C. Huijser,[30,31] Malin S. Sandström,[32] Peer Herholz,[33] Samuel A. Nastase,[34] AmanPreet Badhwar,[16,35,36] Guillaume Dumas,[37,38] Simon Schwab,[39] Stefano Moia,[40,41] Michael Dayan,[42] Yasmine Bassil,[43] Paula P. Brooks,[34] Matteo Mancini,[20,44,45] James M. Shine,[46] David O'Connor,[47] Xihe Xie,[48] Davide Poggiali,[49] Patrick Friedrich,[50]

CONSORTIA

The Brainhack Community includes Nasim Anousheh, Aurina Arnatkeviciute, Guillaume Auzias, Dipankar Bachar, Elise Bannier, Ruggero Basanisi, Arshitha Basavaraj, Marco Bedini, Pierre Bellec, R. Austin Benn, Kathryn Berluti, Steffen Bollmann, Saskia Bollmann, Claire Bradley, Jesse Brown, Augusto Buchweitz, Patrick Callahan, Micaela Y. Chan, Bramsh Q. Chandio, Theresa Cheng, Sidhant Chopra, Ai Wern Chung, Thomas G. Close, Etienne Combrisson, Giorgia Cona, R. Todd Constable, Claire Cury, Kamalaker Dadi, Pablo F. Damasceno, Samir Das, Fabrizio De Vico Fallani, Krista DeStasio, Erin W. Dickie, Lena Dorfschmidt, Eugene P. Duff, Elizabeth DuPre, Sarah L. Dziura, Nathalia B. Esper, Oscar Esteban, Shreyas Fadnavis, Guillaume Flandin, Jessica E. Flannery, John Flournoy, Stephanie J. Forkel, Alexandre R. Franco, Saampras Ganesan, Siyuan Gao, José C. García Alanis, Eleftherios Garyfallidis, Tristan Glatard, Enrico Glerean, Javier Gonzalez-Castillo, Cassandra D. Gould van Praag, Abigail S. Greene, Geetika Gupta, C. Alice Hahn, Yaroslav O. Halchenko, Daniel Handwerker, Thomas S. Hartmann, Valérie Hayot-Sasson, Stephan Heunis, Felix Hoffstaedter, Daniela M. Hohmann, Corey Horien, Horea-Ioan Ioanas, Alexandru D. Iordan, Chao Jiang, Michael Joseph, Jason Kai, Agah Karakuzu, David N. Kennedy, Anisha Keshavan, Ali R. Khan, Gregory Kiar, P. Christiaan Klink, Vincent Koppelmans, Serge Koudoro, Angela R. Laird, Georg Langs, Marissa L. Laws, Roxane Licandro, Sook-Lei Liew, Tomislav Lipic, Krisanne Litinas, Daniel J. Lurie, Désirée Lussier, Christopher R. Madan, Lea-Theresa Mais, Sina Mansour L., J.P. Manzano-Patron, Dimitra Maoutsa, Matheus Marcon, Daniel S. Margulies, Giorgio Marinato, Daniele Marinazzo, Christopher J. Markiewicz, Camille Maumet, Felipe Meneguzzi, David Meunier, Michael P. Milham, Kathryn L. Mills, Davide Momi, Clara A. Moreau, Aysha Motala, Iska Moxon-Emre, Thomas E. Nichols, Dylan M. Nielson, Gustav Nilsonne, Lisa Novello, Caroline O'Brien, Emily Olafson, Lindsay D. Oliver, John A. Onofrey, Edwina R. Orchard, Kendra Oudyk, Patrick J. Park, Mahboobeh Parsapoor, Lorenzo Pasquini, Scott Peltier, Cyril R. Pernet, Rudolph Pienaar, Pedro Pinheiro-Chagas, Jean-Baptiste Poline, Anqi Qiu, Tiago Quendera, Laura C. Rice, Joscelin Rocha-Hidalgo, Saige Rutherford, Mathias Schaninger, Dustin Scheinost, Deena Shariq, Thomas B. Shaw, Viviana Siless, Molly Simmonite, Nikoloz Sirmpilatze, Hayli Spence, Julia Sprenger, Andrija Stajduhar, Martin Szinte, Sylvain Takerkart, Angela Tam, Link Tejavibulya, Michel Thiebaut de Schotten, Ina Thome, Laura Tomaz da Silva, Nicolas Traut, Lucina Q. Uddin, Antonino Vallesi, John W. VanMeter, Nandita Vijayakumar, Matteo Visconti di Oleggio Castello, Jakub Vohryzek, Jaksa Vukojević, Kirstie Jane Whitaker, Lucy Whitmore, Steve Wideman, Suzanne T. Witt, Hua Xie, Ting Xu, Chao-Gan Yan, Fang-Cheng Yeh, B.T. Thomas Yeo, and Xi-Nian Zuo.

Brainhack is an innovative meeting format that promotes scientific collaboration and education in an open, inclusive environment. This NeuroView describes the myriad benefits for participants and the research community and how Brainhacks complement conventional formats to augment scientific progress.

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: OPEN COMMUNITY

- What it is
  - Actively inclusive & collaborative communities, e.g., Brainhack, OHBM Open Science Room, Neuromatch
- Why bother?
  - Network, support, training, future collaborations, etc.
- Fastest + easiest way to get started
  - Join an activity or volunteer! So many opportunities that all usually hybrid: Brainhack Global, Brainhack OHBM, OHBM Open Science Room, Neurohackademy

Neuron

CellPress

NeuroView

Brainhack: Developing a culture of open, inclusive, community-driven neuroscience

Rémi Gau,[1,76,*] Stephanie Noble,[2,76] Katja Heuer,[3,4,76] Katherine L. Bottenhorn,[5,76] Isil P. Bilgin,[6,7,76] Yu-Fang Yang,[8,76] Julia M. Huntenburg,[9,76] Johanna M.M. Bayer,[10,11,76] Richard A.I. Bethlehem,[12,13,76] Shawn A. Rhoads,[14] Christoph Vogelbacher,[15] Valentina Borghesani,[16] Elizabeth Levitis,[17,18] Hao-Ting Wang,[19,20,21] Sofie Van Den Bossche,[22] Xenia Kobeleva,[23,24] Jon Haitz Legarreta,[25] Samuel Guay,[26] Selim Melvin Atay,[27] Gael P. Varoquaux,[28,29] Dorien C. Huijser,[30,31] Malin S. Sandström,[32] Peer Herholz,[33] Samuel A. Nastase,[34] AmanPreet Badhwar,[16,35,36] Guillaume Dumas,[37,38] Simon Schwab,[39] Stefano Moia,[40,41] Michael Dayan,[42] Yasmine Bassil,[43] Paula P. Brooks,[34] Matteo Mancini,[20,44,45] James M. Shine,[46] David O'Connor,[47] Xihe Xie,[48] Davide Poggiali,[49] Patrick Friedrich,[50]

CONSORTIA

The Brainhack Community includes Nasim Anousheh, Aurina Arnatkeviciute, Guillaume Auzias, Dipankar Bachar, Elise Bannier, Ruggero Basanisi, Arshitha Basavaraj, Marco Bedini, Pierre Bellec, R. Austin Benn, Kathryn Berluti, Steffen Bollmann, Saskia Bollmann, Claire Bradley, Jesse Brown, Augusto Buchweitz, Patrick Callahan, Micaela Y. Chan, Bramsh Q. Chandio, Theresa Cheng, Sidhant Chopra, Ai Wern Chung, Thomas G. Close, Etienne Combrisson, Giorgia Cona, R. Todd Constable, Claire Cury, Kamalaker Dadi, Pablo F. Damasceno, Samir Das, Fabrizio De Vico Fallani, Krista DeStasio, Erin W. Dickie, Lena Dorfschmidt, Eugene P. Duff, Elizabeth DuPre, Sarah L. Dziura, Nathalia B. Esper, Oscar Esteban, Shreyas Fadnavis, Guillaume Flandin, Jessica E. Flannery, John Flournoy, Stephanie J. Forkel, Alexandre R. Franco, Saampras Ganesan, Siyuan Gao, José C. García Alanis, Eleftherios Garyfallidis, Tristan Glatard, Enrico Glerean, Javier Gonzalez-Castillo, Cassandra D. Gould van Praag, Abigail S. Greene, Geetika Gupta, C. Alice Hahn, Yaroslav O. Halchenko, Daniel Handwerker, Thomas S. Hartmann, Valérie Hayot-Sasson, Stephan Heunis, Felix Hoffstaedter, Daniela M. Hohmann, Corey Horien, Horea-Ioan Ioanas, Alexandru D. Iordan, Chao Jiang, Michael Joseph, Jason Kai, Agah Karakuzu, David N. Kennedy, Anisha Keshavan, Ali R. Khan, Gregory Kiar, P. Christiaan Klink, Vincent Koppelmans, Serge Koudoro, Angela R. Laird, Georg Langs, Marissa L. Laws, Roxane Licandro, Sook-Lei Liew, Tomislav Lipic, Krisanne Litinas, Daniel J. Lurie, Désirée Lussier, Christopher R. Madan, Lea-Theresa Mais, Sina Mansour L., J.P. Manzano-Patron, Dimitra Maoutsa, Matheus Marcon, Daniel S. Margulies, Giorgio Marinato, Daniele Marinazzo, Christopher J. Markiewicz, Camille Maumet, Felipe Meneguzzi, David Meunier, Michael P. Milham, Kathryn L. Mills, Davide Momi, Clara A. Moreau, Aysha Motala, Iska Moxon-Emre, Thomas E. Nichols, Dylan M. Nielson, Gustav Nilsonne, Lisa Novello, Caroline O'Brien, Emily Olafson, Lindsay D. Oliver, John A. Onofrey, Edwina R. Orchard, Kendra Oudyk, Patrick J. Park, Mahboobeh Parsapoor, Lorenzo Pasquini, Scott Peltier, Cyril R. Pernet, Rudolph Pienaar, Pedro Pinheiro-Chagas, Jean-Baptiste Poline, Anqi Qiu, Tiago Quendera, Laura C. Rice, Joscelin Rocha-Hidalgo, Saige Rutherford, Mathias Schaninger, Dustin Scheinost, Deena Shariq, Thomas B. Shaw, Viviana Siless, Molly Simmonite, Nikoloz Sirmpilatze, Hayli Spence, Julia Sprenger, Andrija Stajduhar, Martin Szinte, Sylvain Takerkart, Angela Tam, Link Tejavibulya, Michel Thiebaut de Schotten, Ina Thome, Laura Tomaz da Silva, Nicolas Traut, Lucina Q. Uddin, Antonino Vallesi, John W. VanMeter, Nandita Vijayakumar, Matteo Visconti di Oleggio Castello, Jakub Vohryzek, Jaksa Vukojević, Kirstie Jane Whitaker, Lucy Whitmore, Steve Wideman, Suzanne T. Witt, Hua Xie, Ting Xu, Chao-Gan Yan, Fang-Cheng Yeh, B.T. Thomas Yeo, and Xi-Nian Zuo.

Brainhack is an innovative meeting format that promotes scientific collaboration and education in an open, inclusive environment. This NeuroView describes the myriad benefits for participants and the research community and how Brainhacks complement conventional formats to augment scientific progress.

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: OPEN COMMUNITY

- What it is
  - Actively inclusive & collaborative communities, e.g., Brainhack, OHBM Open Science Room, Neuromatch

- Why bother?
  - Network, support, training, future collaborations, etc.

- Fastest + easiest way to get started
  - Join an activity or volunteer! So many opportunities that all usually hybrid: Brainhack Global, Brainhack OHBM, OHBM Open Science Room, Neurohackademy

- Common bottlenecks
  - You (or your mentor) may not have the funds or need to justify
    - Many virtual options
    - Justification: provides training and networking (Gau et al., 2021)

Neuron

CellPress

NeuroView

Brainhack: Developing a culture of open, inclusive, community-driven neuroscience

Rémi Gau,[1,76,*] Stephanie Noble,[2,76] Katja Heuer,[3,4,76] Katherine L. Bottenhorn,[5,76] Isil P. Bilgin,[6,7,76] Yu-Fang Yang,[8,76] Julia M. Huntenburg,[9,76] Johanna M.M. Bayer,[10,11,76] Richard A.I. Bethlehem,[12,13,76] Shawn A. Rhoads,[14] Christoph Vogelbacher,[15] Valentina Borghesani,[16] Elizabeth Levitis,[17,18] Hao-Ting Wang,[19,20,21] Sofie Van Den Bossche,[22] Xenia Kobeleva,[23,24] Jon Haitz Legarreta,[25] Samuel Guay,[26] Selim Melvin Atay,[27] Gael P. Varoquaux,[28,29] Dorien C. Huijser,[30,31] Malin S. Sandström,[32] Peer Herholz,[33] Samuel A. Nastase,[34] AmanPreet Badhwar,[16,35,36] Guillaume Dumas,[37,38] Simon Schwab,[39] Stefano Moia,[40,41] Michael Dayan,[42] Yasmine Bassil,[43] Paula P. Brooks,[34] Matteo Mancini,[20,44,45] James M. Shine,[46] David O'Connor,[47] Xihe Xie,[48] Davide Poggiali,[49] Patrick Friedrich,[50]

CONSORTIA

The Brainhack Community includes Nasim Anousheh, Aurina Arnatkeviciute, Guillaume Auzias, Dipankar Bachar, Elise Bannier, Ruggero Basanisi, Arshitha Basavaraj, Marco Bedini, Pierre Bellec, R. Austin Benn, Kathryn Berluti, Steffen Bollmann, Saskia Bollmann, Claire Bradley, Jesse Brown, Augusto Buchweitz, Patrick Callahan, Micaela Y. Chan, Bramsh Q. Chandio, Theresa Cheng, Sidhant Chopra, Ai Wern Chung, Thomas G. Close, Etienne Combrisson, Giorgia Cona, R. Todd Constable, Claire Cury, Kamalaker Dadi, Pablo F. Damasceno, Samir Das, Fabrizio De Vico Fallani, Krista DeStasio, Erin W. Dickie, Lena Dorfschmidt, Eugene P. Duff, Elizabeth DuPre, Sarah L. Dziura, Nathalia B. Esper, Oscar Esteban, Shreyas Fadnavis, Guillaume Flandin, Jessica E. Flannery, John Flournoy, Stephanie J. Forkel, Alexandre R. Franco, Saampras Ganesan, Siyuan Gao, José C. García Alanis, Eleftherios Garyfallidis, Tristan Glatard, Enrico Glerean, Javier Gonzalez-Castillo, Cassandra D. Gould van Praag, Abigail S. Greene, Geetika Gupta, C. Alice Hahn, Yaroslav O. Halchenko, Daniel Handwerker, Thomas S. Hartmann, Valérie Hayot-Sasson, Stephan Heunis, Felix Hoffstaedter, Daniela M. Hohmann, Corey Horien, Horea-Ioan Ioanas, Alexandru D. Iordan, Chao Jiang, Michael Joseph, Jason Kai, Agah Karakuzu, David N. Kennedy, Anisha Keshavan, Ali R. Khan, Gregory Kiar, P. Christiaan Klink, Vincent Koppelmans, Serge Koudoro, Angela R. Laird, Georg Langs, Marissa L. Laws, Roxane Licandro, Sook-Lei Liew, Tomislav Lipic, Krisanne Litinas, Daniel J. Lurie, Désirée Lussier, Christopher R. Madan, Lea-Theresa Mais, Sina Mansour L., J.P. Manzano-Patron, Dimitra Maoutsa, Matheus Marcon, Daniel S. Margulies, Giorgio Marinato, Daniele Marinazzo, Christopher J. Markiewicz, Camille Maumet, Felipe Meneguzzi, David Meunier, Michael P. Milham, Kathryn L. Mills, Davide Momi, Clara A. Moreau, Aysha Motala, Iska Moxon-Emre, Thomas E. Nichols, Dylan M. Nielson, Gustav Nilsonne, Lisa Novello, Caroline O'Brien, Emily Olafson, Lindsay D. Oliver, John A. Onofrey, Edwina R. Orchard, Kendra Oudyk, Patrick J. Park, Mahboobeh Parsapoor, Lorenzo Pasquini, Scott Peltier, Cyril R. Pernet, Rudolph Pienaar, Pedro Pinheiro-Chagas, Jean-Baptiste Poline, Anqi Qiu, Tiago Quendera, Laura C. Rice, Joscelin Rocha-Hidalgo, Saige Rutherford, Mathias Schaninger, Dustin Scheinost, Deena Shariq, Thomas B. Shaw, Viviana Siless, Molly Simmonite, Nikoloz Sirmpilatze, Hayli Spence, Julia Sprenger, Andrija Stajduhar, Martin Szinte, Sylvain Takerkart, Angela Tam, Link Tejavibulya, Michel Thiebaut de Schotten, Ina Thome, Laura Tomaz da Silva, Nicolas Traut, Lucina Q. Uddin, Antonino Vallesi, John W. VanMeter, Nandita Vijayakumar, Matteo Visconti di Oleggio Castello, Jakub Vohryzek, Jaksa Vukojević, Kirstie Jane Whitaker, Lucy Whitmore, Steve Wideman, Suzanne T. Witt, Hua Xie, Ting Xu, Chao-Gan Yan, Fang-Cheng Yeh, B.T. Thomas Yeo, and Xi-Nian Zuo.

Brainhack is an innovative meeting format that promotes scientific collaboration and education in an open, inclusive environment. This NeuroView describes the myriad benefits for participants and the research community and how Brainhacks complement conventional formats to augment scientific progress.

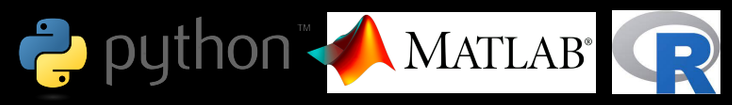
OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE:
MODERN SCIENTIFIC PROGRAMMING

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: MODERN SCIENTIFIC PROGRAMMING

- Highlights
  - Python (historically software dev), Matlab (historically engineering), R (historically stats)
    - "Historically", but times are changing
    - Neuroimaging-specific libraries: Nipype, Nilearn
  - Programming assistance: GitHub copilot, ChatGPT (but be careful)
  - Containers (e.g., Docker)

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: MODERN SCIENTIFIC PROGRAMMING

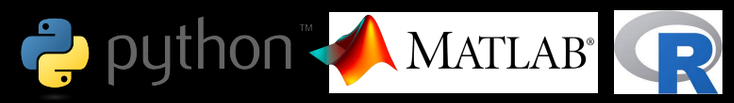
- Highlights
  - Python (historically software dev), Matlab (historically engineering), R (historically stats)
    - "Historically", but times are changing
    - Neuroimaging-specific libraries: Nipype, Nilearn
  - Programming assistance: GitHub copilot, ChatGPT (but be careful)
  - Containers (e.g., Docker)



- Why bother?
  - Building blocks of modern neuroscience = more rigorous projects, transferrable skills

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: MODERN SCIENTIFIC PROGRAMMING

- Highlights
  - Python (historically software dev), Matlab (historically engineering), R (historically stats)
    - "Historically", but times are changing
    - Neuroimaging-specific libraries: Nipype, Nilearn
  - Programming assistance: GitHub copilot, ChatGPT (but be careful)
  - Containers (e.g., Docker)

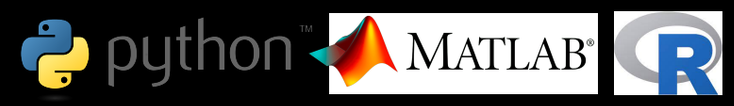


- Why bother?
  - Building blocks of modern neuroscience = more rigorous projects, transferrable skills

- Fastest + easiest way to get started
  - Take on a project or project-oriented course that uses that language (no, seriously)
  - Datacamp, codecademy, etc is more for secondary support
  - Programming assistance = double-edged sword

OTHER TOPICS IN REPRODUCIBLE NEUROSCIENCE: MODERN SCIENTIFIC PROGRAMMING

- Highlights
  - Python (historically software dev), Matlab (historically engineering), R (historically stats)
    - "Historically", but times are changing
    - Neuroimaging-specific libraries: Nipype, Nilearn
  - Programming assistance: GitHub copilot, ChatGPT (but be careful)
  - Containers (e.g., Docker)



- Why bother?
  - Building blocks of modern neuroscience = more rigorous projects, transferrable skills

- Fastest + easiest way to get started
  - Take on a project or project-oriented course that uses that language (no, seriously)
  - Datacamp, codecademy, etc is more for secondary support
  - Programming assistance = double-edged sword

- Common bottlenecks
  - Where to start?
    - Python is easiest + most flexible
      - but good to learn what your community uses most
    - Start with a real but manageable project (e.g., containers are more of an intermediate topic)

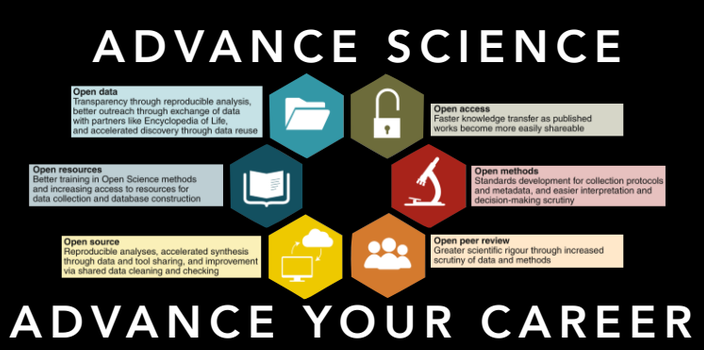
CONCLUSION

ADVANCE SCIENCE

ADVANCE YOUR CAREER

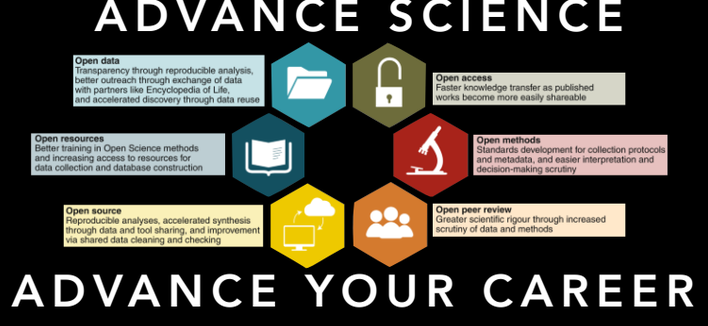
CONCLUSION

- Open neuroscience practices not only benefit the community, but also:
  - Save you $$, time, & effort
  - Facilitate dissemination & recognition of your work
  - Teach you transferrable skills



CONCLUSION

- Open neuroscience practices not only benefit the community, but also:
  - Save you $$, time, & effort
  - Facilitate dissemination & recognition of your work
  - Teach you transferrable skills



- Baby steps
  - **Open code:** Put one piece of code on GitHub
  - **Open data:** Get some preprocessed HCP data
  - **Open publishing:** Upload a manuscript to arXiv / OSF
  - **Open community:** Join a virtual Brainhack, OSR, or Neurohackademy event
  - **Modern scientific programming:** Follow a NiLearn tutorial

ACKNOWLEDGMENTS

Now hiring—join the Neuro-PRISM team!

Dustin Scheinost & MINDS Lab (Yale)

Joshua Curtiss (MGH)

Amanda Mejia (Indiana)

Thomas Nichols (Oxford)

Todd Constable & Connectivity Lab (Yale)

Tyrone Cannon (Yale)

John Murray (Yale) & Max Shinn (Oxford)

Marvin Chun & Kwangsun Yoo (Yale)

Andrew Zalesky & Sina Mansour (Melbourne)

Joshua Vogelstein & Eric Bridgeford (Johns Hopkins)

Marisa Spann (Columbia)

Argyris Stringaris (UCL) & Dylan Nielson (NIMH)

Jia-Hong Gao (Peking) & Guoyuan Yang (SJTU)

Daniel Barron (Yale)

Chris Benjamin (Yale)

Karim Ibrahim (Yale)

NIH DSPAN F99/K00
F99NS108557
K00MH122372

NIH BRAIN K99/R00
K99MH130894

THANK YOU! QUESTIONS?

sNeuroble

@sNeuroble

sNeuroble

BioImage Suite Web

fast & portable image analysis