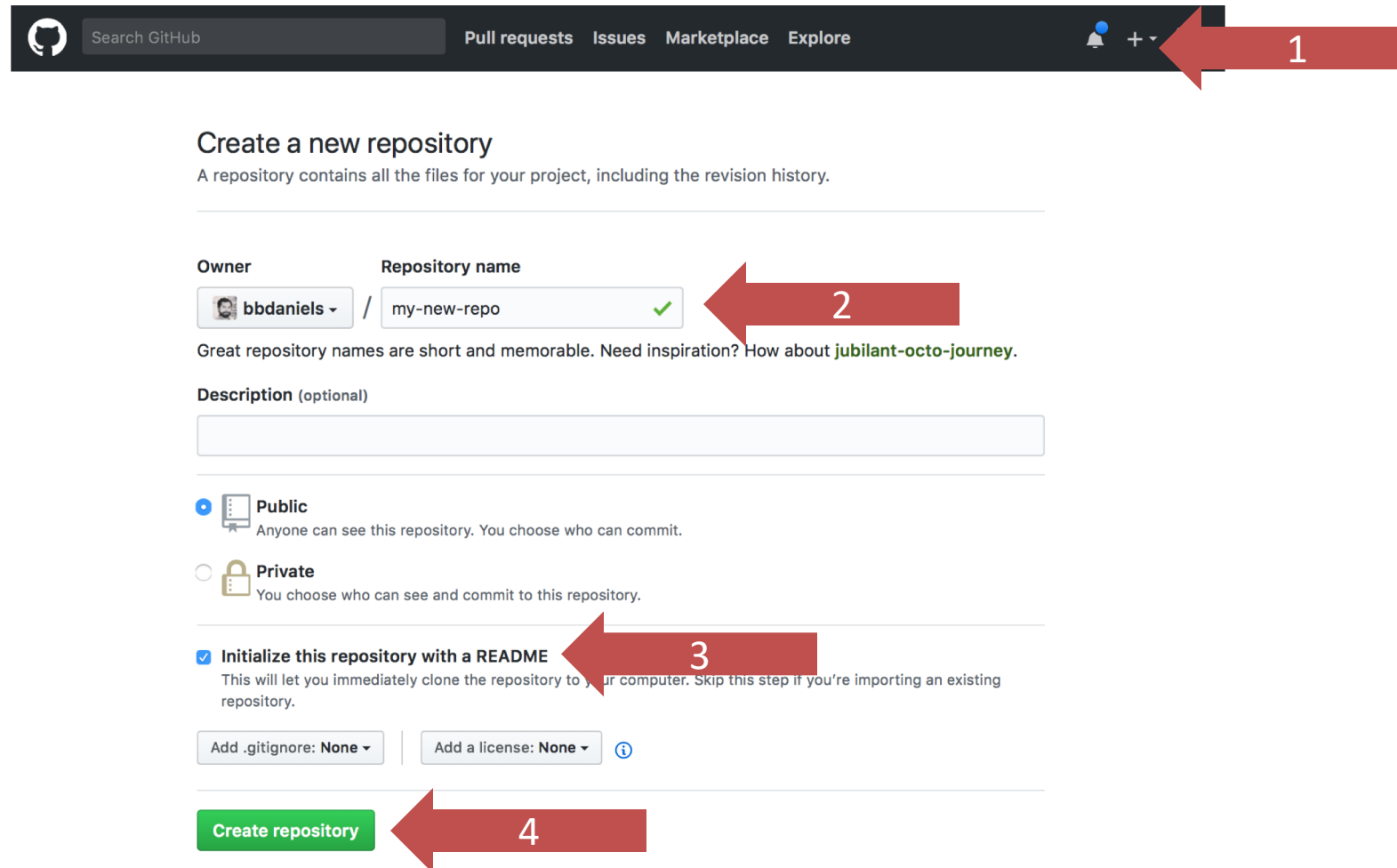


# Create a repository on GitHub



The screenshot shows the GitHub 'Create a new repository' page. Four red arrows with numbers 1 through 4 indicate the steps to create a repository:

- 1**: Points to the top navigation bar, specifically the '+' icon in the top right corner.
- 2**: Points to the 'Repository name' input field, which contains 'my-new-repo' and has a green checkmark.
- 3**: Points to the 'Initialize this repository with a README' checkbox, which is checked.
- 4**: Points to the 'Create repository' button at the bottom.

The form fields and options are as follows:

- Owner**: bbdaniels
- Repository name**: my-new-repo
- Description (optional)**: (empty text box)
- Visibility**: ☒ **Public** (Anyone can see this repository. You choose who can commit.)  
☐ **Private** (You choose who can see and commit to this repository.)
- Initialize this repository with a README**: ☒ (This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.)
- Add .gitignore**: None
- Add a license**: None

# Setup in GitHub Desktop

The screenshot shows the GitHub web interface for a repository named 'my-new-repo' by user 'bbdaniels'. The repository has 1 commit, 1 branch, 0 releases, and 1 contributor. The 'Clone or download' button is highlighted with a red arrow, and its dropdown menu is open, showing the 'Open in Desktop' option. A tooltip below the 'Open in Desktop' button reads: 'Clone bbdaniels/my-new-repo to your computer and use it in GitHub Desktop.'

Repository: bbdaniels / my-new-repo

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

No description, website, or topics provided. [Add topics](#) [Edit](#)

1 commit 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

Clone with HTTPS Use SSH

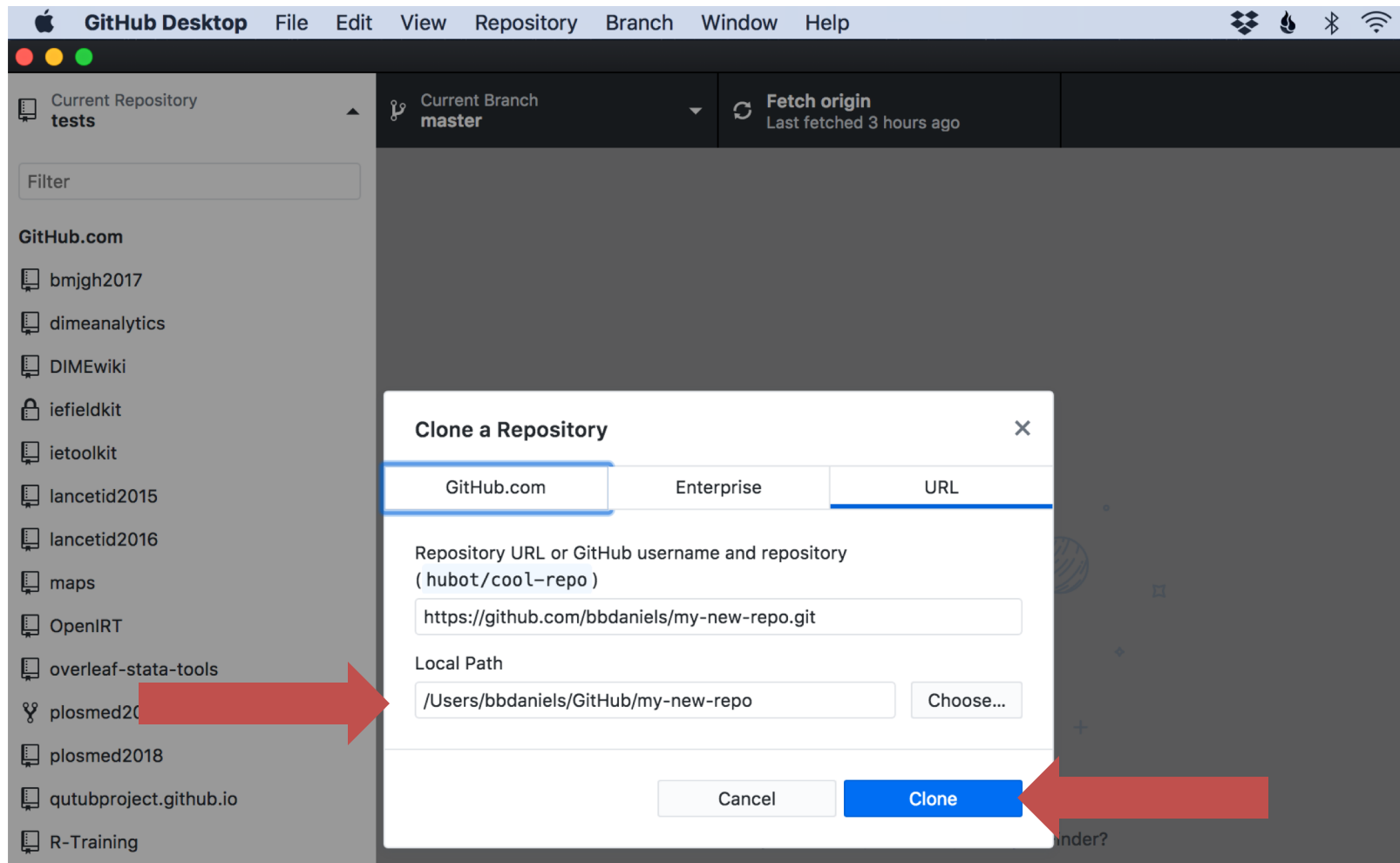
Use Git or checkout with SVN using the web URL.

https://github.com/bbdaniels/my-new-r

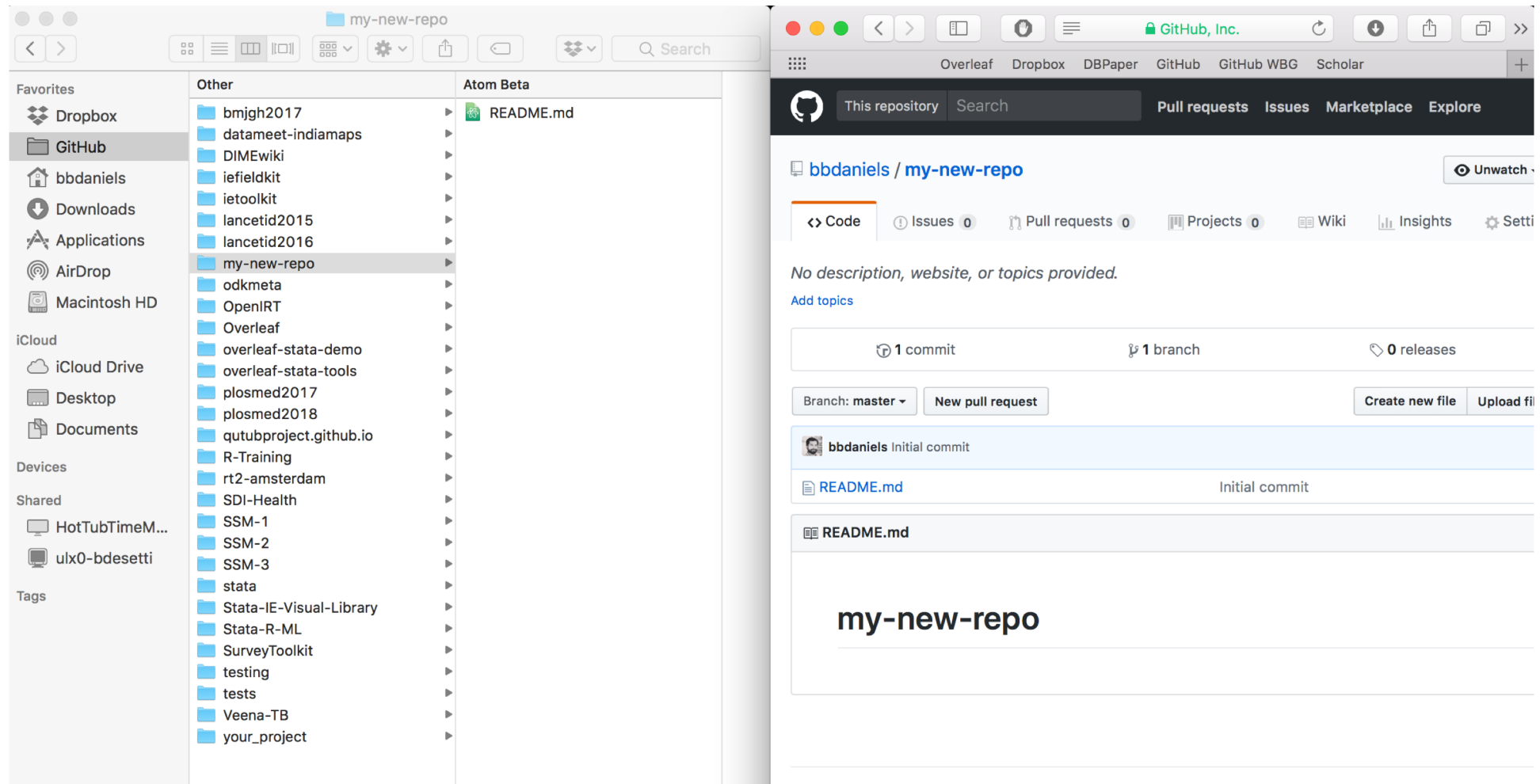
Open in Desktop Download ZIP

Clone bbdaniels/my-new-repo to your computer and use it in GitHub Desktop.

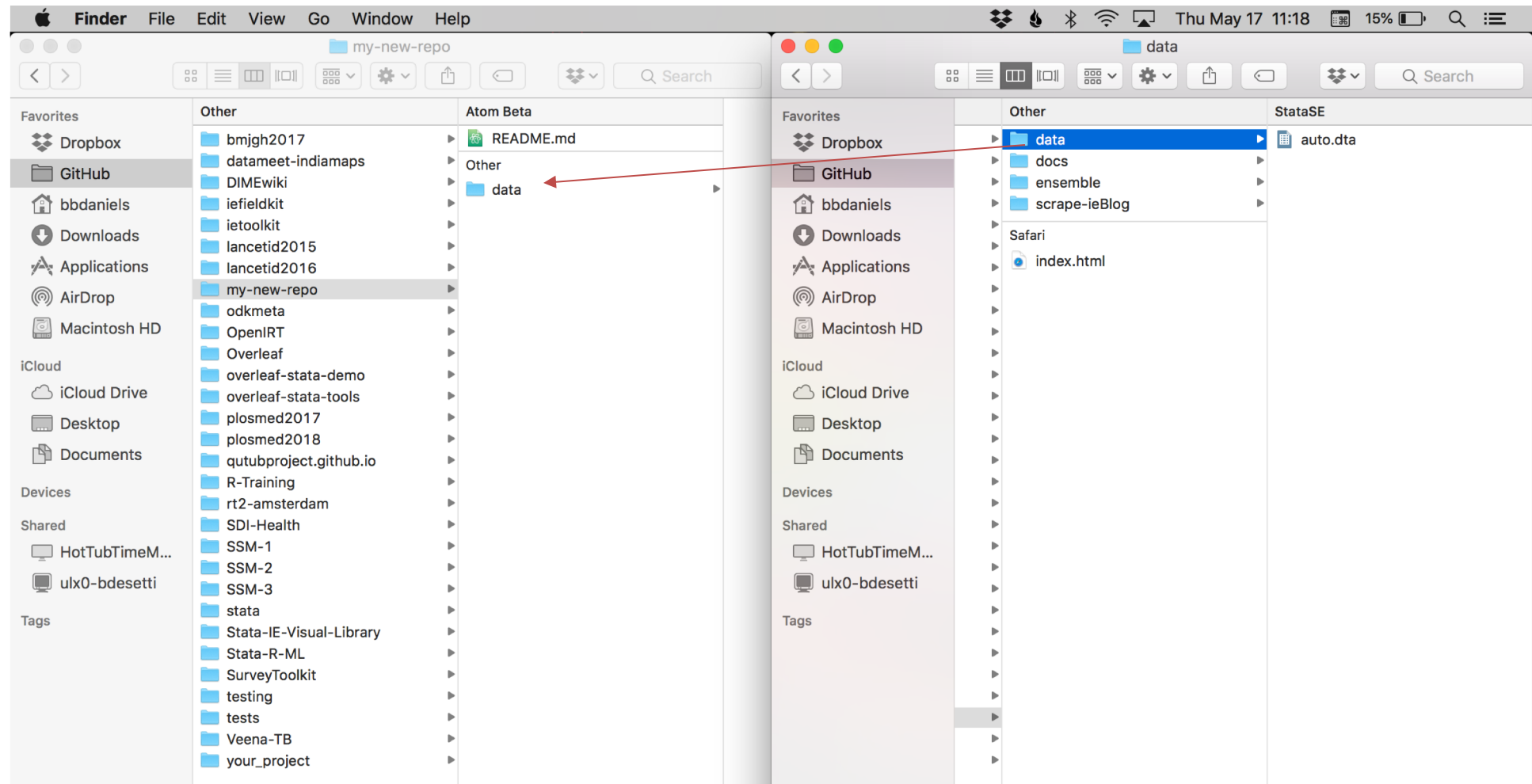
# Set local destination for “clone” repository



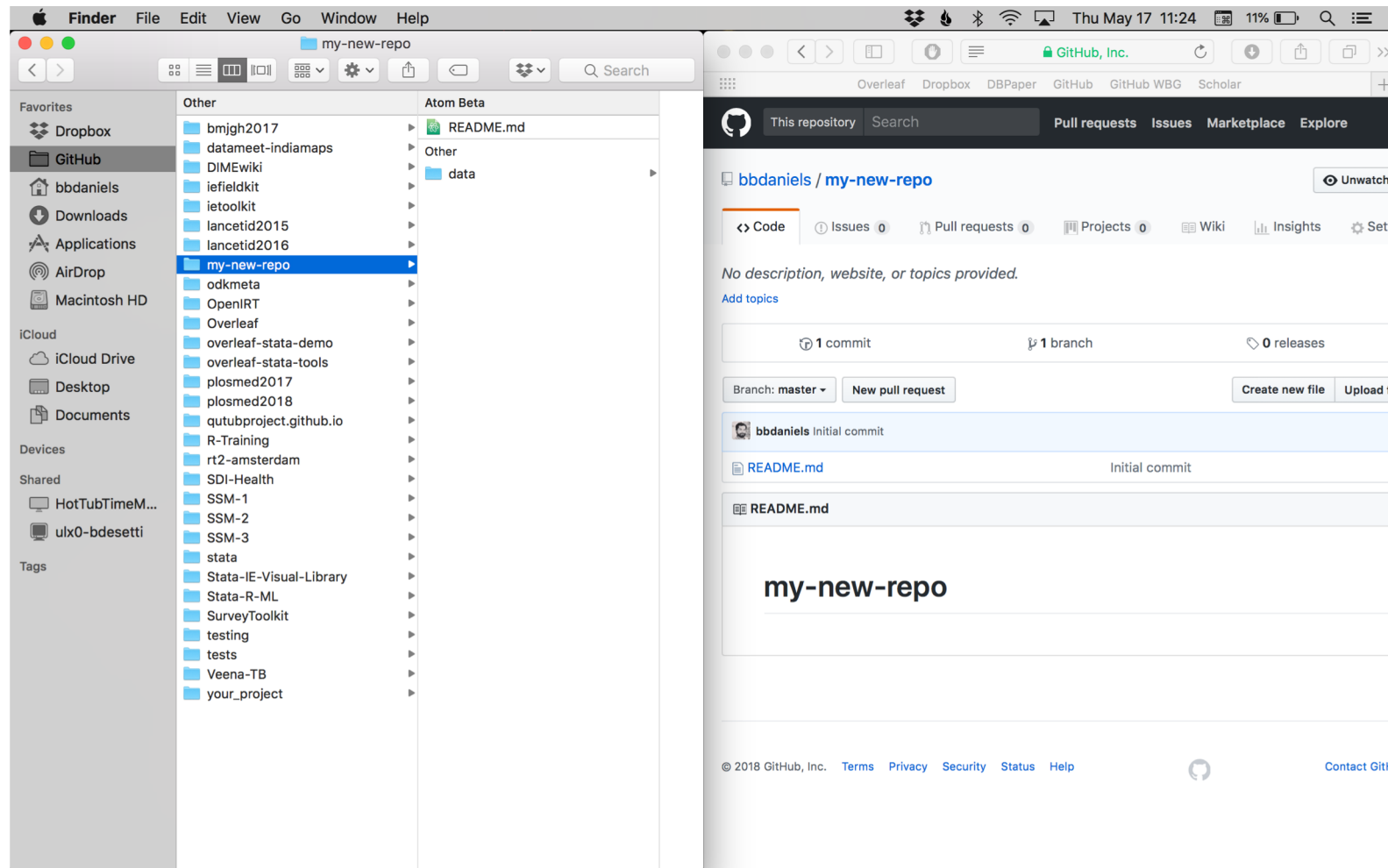
# The “local” and “remote” instances are identical



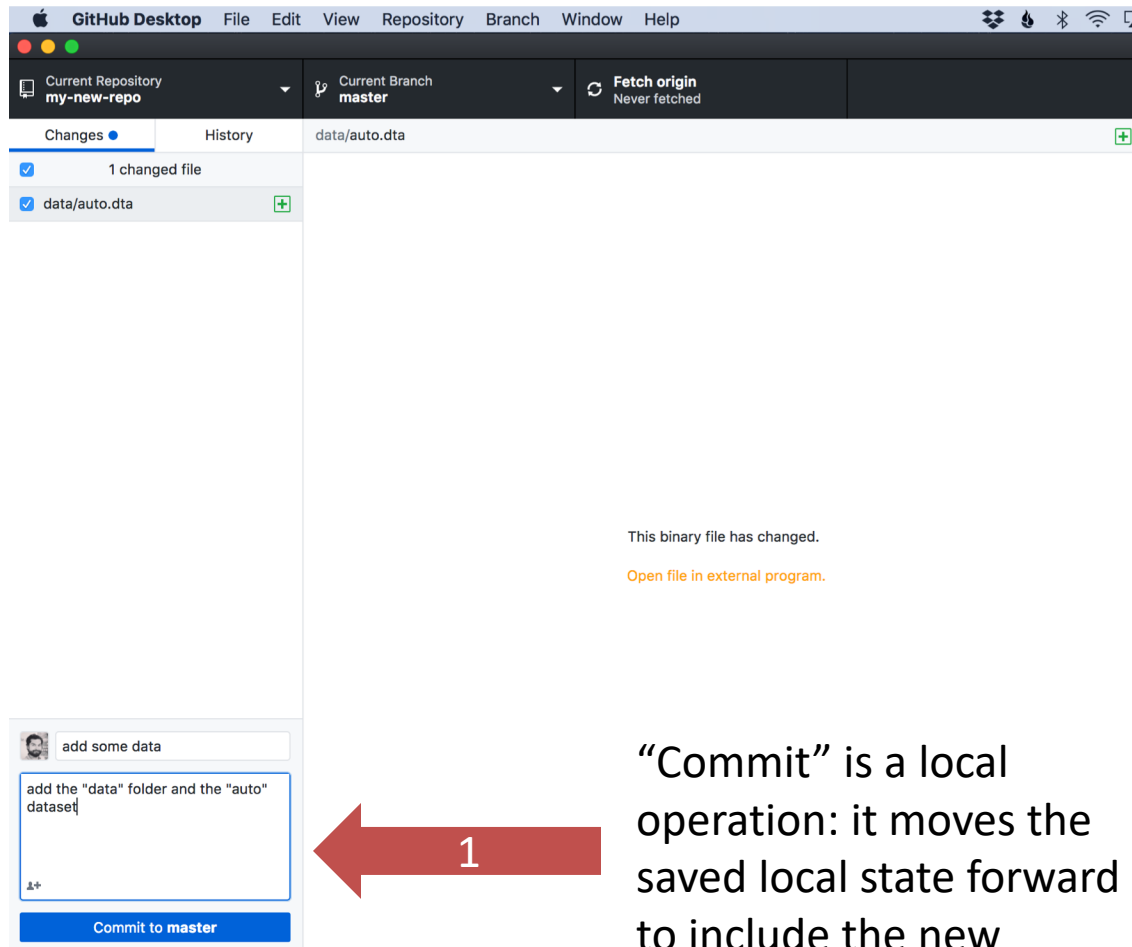
# Add some data locally...



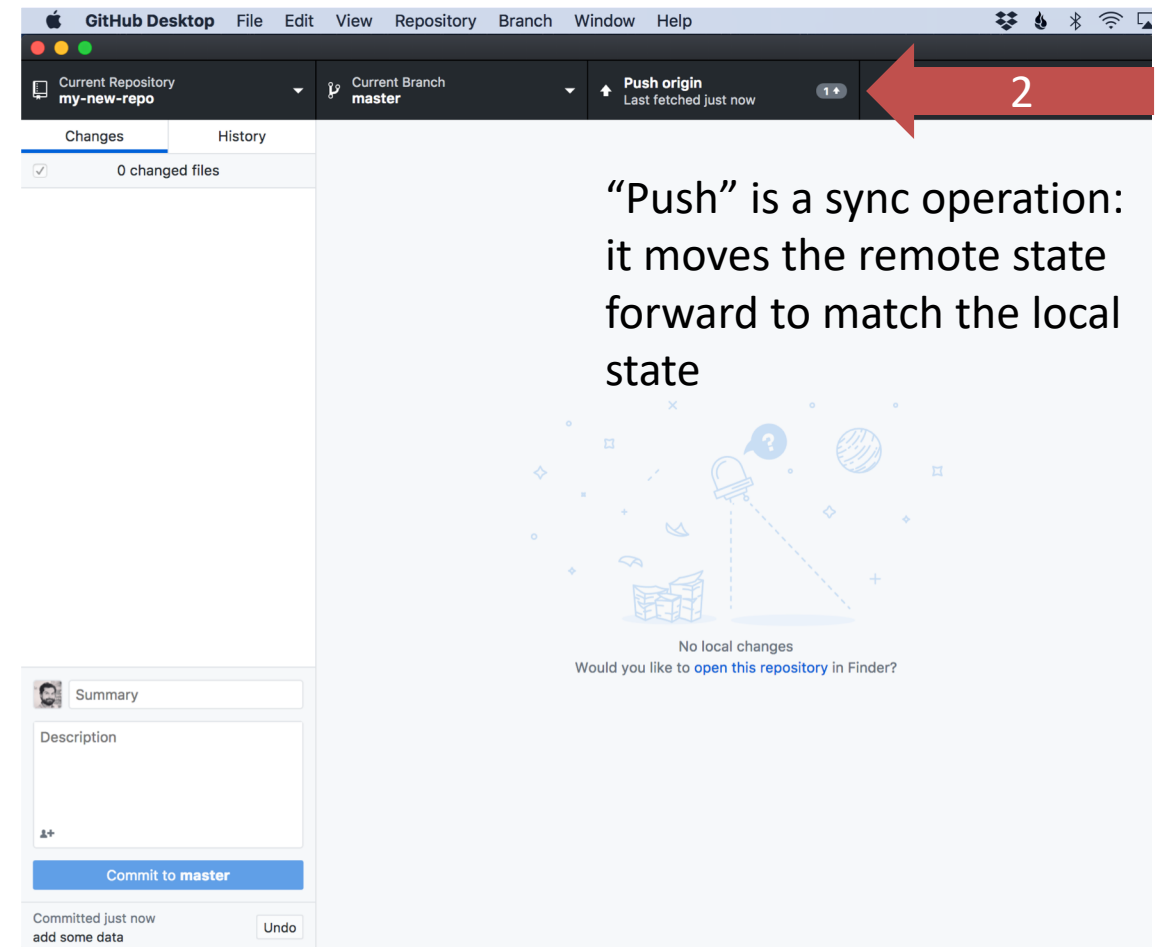
# The local and remote are *not* in sync



# GitHub Desktop notes the changes and allows you to push them to the remote (“origin”)



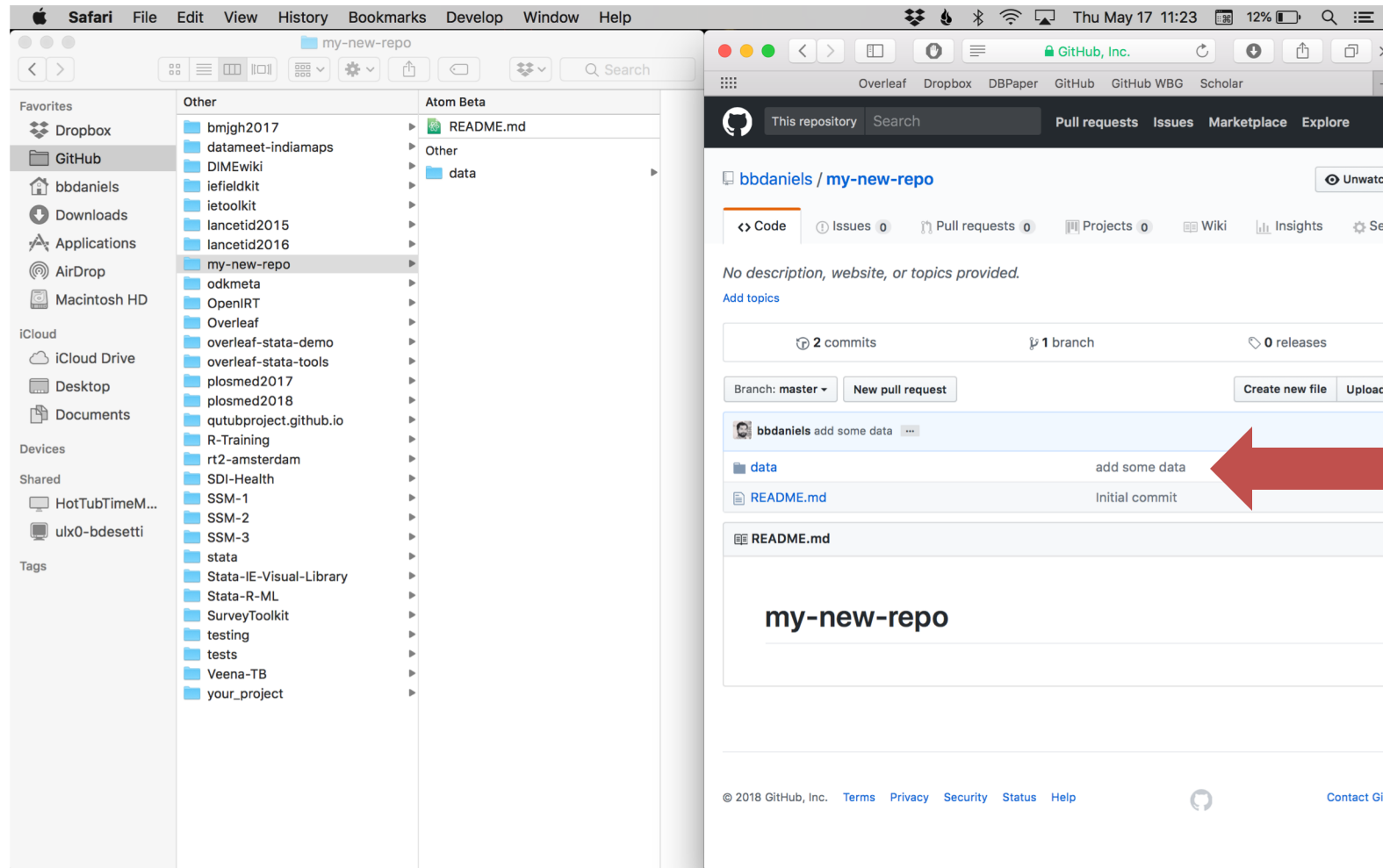
“Commit” is a local operation: it moves the saved local state forward to include the new changes



“Push” is a sync operation: it moves the remote state forward to match the local state



# The local and remote are again in sync



Each item  
records  
when it  
was last  
modified

# Git can do much more!

---

- For now, you have created a version history of your files with a publicly accessible remote backup
- Git is designed such that this remote backup serves as a coordination point for multiple people making changes simultaneously, through “branching”
- We will cover this later!
- [Pro Git](#) is a good introduction to all of Git’s many features
- And [GitKraken](#) is a better desktop client for interacting with the version history and the advanced features



**i2i**  
DIME  
TRANSFORM DEVELOPMENT

