

Version Control with Git and GitHub

Harrie Jonkman

11/25/2021

Version Control with Git and GitHub using R

- ▶ Basics of using version control system to keep track of all your important R code
- ▶ But also facilitate collaboration with colleagues and the wider world.

Target

- ▶ Learning to use Git in combination with the web-based hosting service GitHub
- ▶ Looking at installing and configuring Git and GitHub
- ▶ Working with Git and Github stand alone
- ▶ Collaborate with someone else
- ▶ Preparation on next workshop on Distill
- ▶ All with R/RStudio

Set-up

- ▶ Github sites
- ▶ Presentation
- ▶ Working stand alone
- ▶ Collaborate with someone else
- ▶ Questions, discussion, preparation NSC-R website

Overall: We try to keep it simple

Caution

- ▶ Working with R and Git/GitHub, but the it's more elaborated with own menu system and syntax
- ▶ Git/Github can be complicated and can frustrate you
- ▶ Project can be restricted (try to get <50 MB and not bigger than 100 MB)

Why using version control?

"FINAL".doc



FINAL.doc!



FINAL_rev.2.doc



FINAL_rev.6.COMMENTS.doc



FINAL_rev.8.comments5.
CORRECTIONS.doc



FINAL_rev.18.comments7.



FINAL_rev.22.comments49



JAMES CHAMBERLAIN

Why using version control?

- ▶ Version control takes care of keeping a record of all versions of a particular file
- ▶ It allows you to revert back to previous version if you need to
- ▶ It keeps track of all your files in a single place
- ▶ It helps others (especially collaborators) to contribute to and reuse your work
- ▶ Files are always available from anywhere and on any computer if you are connected
- ▶ It helps you learn programming

What is Git?

- ▶ It is the version control system that lets you track changes to a set of files
- ▶ It can be any files that make up a project (.pdf, .rmd, .docc, .txt, .jpg etc)
- ▶ All the files that make up a project is called a **repository** (or just **repo**)

What is GitHub

- ▶ It is a web-based hosting service for Git repositories
- ▶ It allows you to create a remote copy of your local-version controlled project
- ▶ It can be a back up or archive of your projects
- ▶ It makes it accessible to you and others

Workproces

1. Typically (but not always) create a **remote** repository on GitHub
2. Then **clone** (say copying) this repository to our **local** computer
3. When it works, you work locally on your project as usual
4. You can take snapshots (call it **commits**) of these files after you've made important changes
5. Then we **push** these changes to the remote GitHub repository to make a backup or make it available for collaborators
6. If others work in the project/repository on their computer they make changes and you can **pull** it to your local repository

Workproces visual

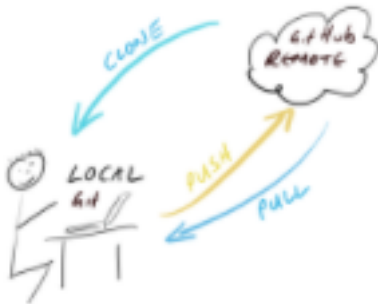


Figure 2: Workproces visual

Installing R/RStudio

- ▶ Be sure you have the latest version of R, here
- ▶ And RStudio here

Installing Git

- ▶ Look at Bryan, J. *Happy Git and GitHub for the useR*, chapter 6 here

Configure Git from RStudio

Options

Enable version control interface for RStudio projects

Git executable:
 [Browse...](#)

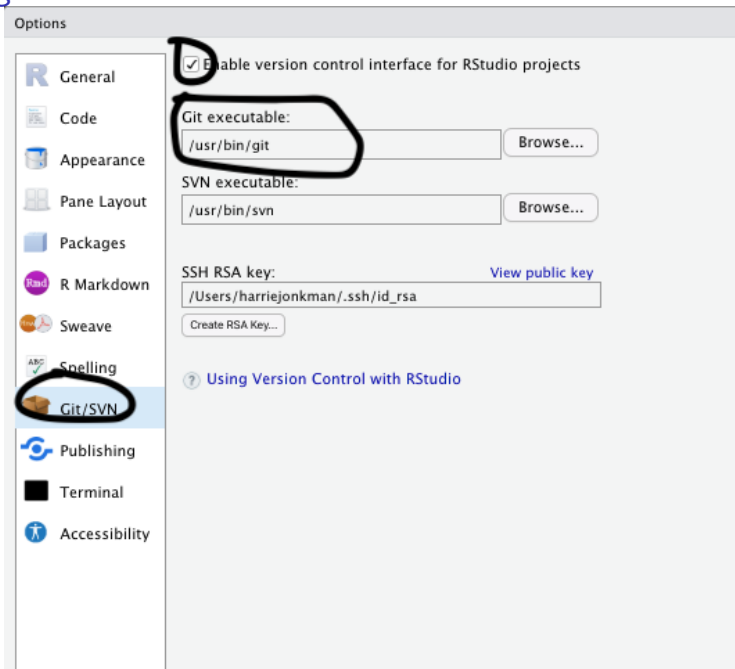
SVN executable:
 [Browse...](#)

SSH RSA key: [View public key](#)

[Create RSA Key...](#)

[? Using Version Control with RStudio](#)

General
Code
Appearance
Pane Layout
Packages
R Markdown
Sweave
Spelling
Git/SVN
Publishing
Terminal
Accessibility



Register a GitHub account

- ▶ See Bryan, J., *Happy Git and GitHub for the useR*, chapter 4, here

Setting up a project, different options:

- ▶ Option 1: Set up a remote GitHub repository first and then connect to an RStudio project to this repository (easier, showed here)
- ▶ Option 2: Connect an existing project to a GitHub repository (complexer, not showed)

GitHub first

- ▶ Create a **repository (repo)** on GitHub
- ▶ Go to the GitHub page and sign in if necessary
- ▶ Click on the 'Repositories' tab at the top
- ▶ Click on the green 'New' button on the right

This is what you see

The screenshot shows a web browser window with the URL `github.com/Jonkman1?tab=repositories`. The page displays the profile of Harrie Jonkman, who is a senior researcher. Below the profile, there is a list of repositories:

- NSCR_Web** (Public): Website for NSC-R webshops. Updated 5 days ago. Includes tags for HTML, 1 fork, and Creative Commons Zero v1.0 Universal license.
- GithubfromR** (Public): Updated 14 days ago.
- HarriesHoekje** (Public): Hier mijn blog over ontwikkelingen in de moderne data-analyse. Updated 15 days ago. Includes the HTML tag.

A green button labeled "New" with a repository icon is circled in the top right corner of the repository list.

https://github.com/account wing · ☆ 0

Test_alone

- ▶ Give your new repo a name(let us call it Test_alone)
- ▶ Select Public
- ▶ Tick on the 'Initialize this repository with a README'
- ▶ Click on ' Create repository'

This is what you see

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Owner *



Repository name

test_alone



Great repository names are short and memorable. Need inspiration? How about [miniature-octo-winner?](#)

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.



Add a README file

This is where you can write a long description for your project. [Learn more](#).

Add .gitignore

Choose which files not to track from a list of templates. [Learn more](#).

Choose a license

A license tells others what they can and can't do with your code. [Learn more](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

Clone or download to R/RStudio

- ▶ The new GitHub repository is created
- ▶ Now we **clone** (or download) it
- ▶ We copy the `https://..` URL that pops up

Switch to RStudio

- ▶ Click on File -> New Project
- ▶ Select Version Control

This is what you see

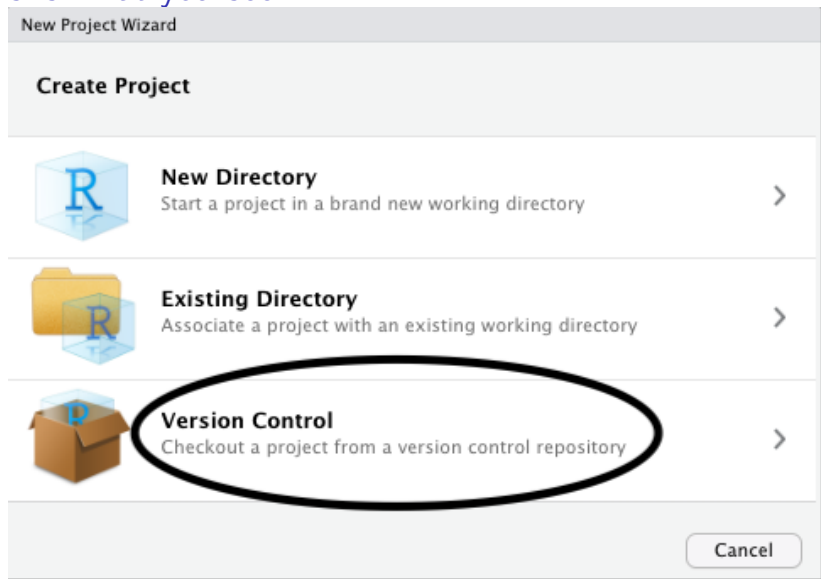


Figure 6: Create Project/Version Control

Paste it

- ▶ Now paste the URL you copied from GitHub in the Repository URL box
- ▶ It will fill out the Project Directory Name: with correct repository name
- ▶ Select where you want to place it on your machine (Browse)

This is what you see

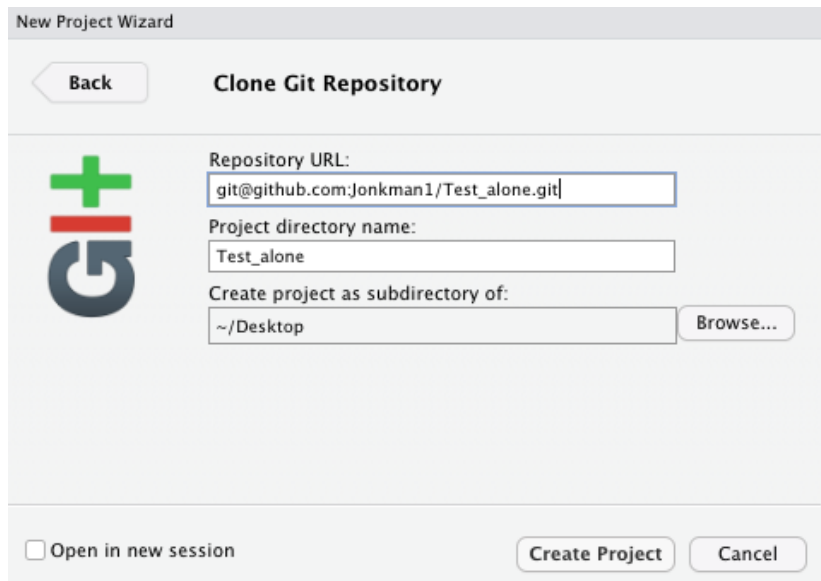
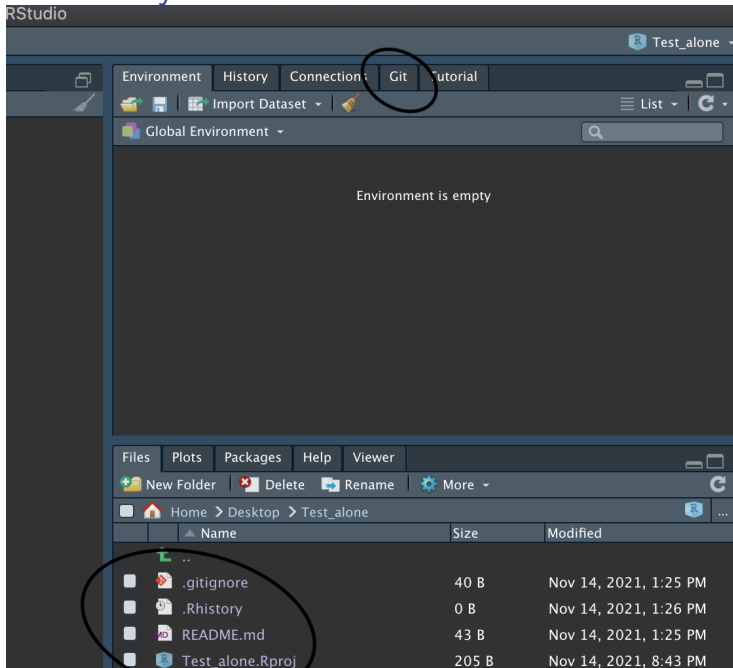


Figure 7: Git repo inside directory

Repo on your machine

- ▶ RStudio will create a new directory with the same name as your repo on your local computer
- ▶ Will **clone** your remote repository to this directory
- ▶ It will place three files: `Test_alone.Rproj`, `README.md` and `.gitignore`
- ▶ You will see a Git tab on your machine

This is what you see



Using Git by showing you two practices

- ▶ Work on your own (test_alone)
- ▶ Working with some one else (test_together)

References

- ▶ Douglas, A., Roos, D., Mancini, F., Couto, A. and Lusseau, D., *An introduction to R*, chapter 9, [here](#). Thank you.
- ▶ Bryan, J., *Happy Git and GitHub for the useR*, [here](#)
- ▶ Lendway, L., *github_for_collaboration*, [here](#)
- ▶ On youtube, [here](#)
- ▶ On youtube also, [here](#)