

Introducción a Git y GitHub desde RStudio



Irene Ramos &
Andrea Gómez Vargas

17 de abril 2023

CC-0 Licensed





THE CARPENTRIES

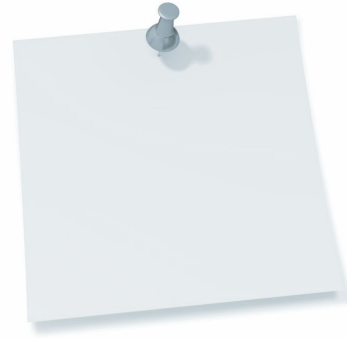
- The Carpentries enseña a investigadores de todo el mundo técnicas básicas de código y ciencia de datos.
- Todos los talleres de *Software Carpentry*, *Data Carpentry*, and *Library Carpentry* se basan en las lecciones de The Carpentries.
- Todas las personas que participen en las actividades de The Carpentries deben guiarse por el **Código de Conducta**.



Necesito ayuda.



Terminé el ejercicio.



Temario

- ¿Qué es el control de versiones?
- Configurando Git
- Repositorios remotos en GitHub
- Usando Git desde RStudio



Créditos

- Daisie Huang and Ivan Gonzalez (eds): "Software Carpentry: Version Control with Git." Version 2016.06, June 2016, <https://github.com/swcarpentry/git-novice>, 10.5281/zenodo.57467.
- Yanina Bellini Saibene, & Marysol Gatti. (2020, September). yabellini/Intro_to_Git_with_Rstudio: Primeros pasos de Git con R y RStudio (Version v1.0). Zenodo. <http://doi.org/10.5281/zenodo.4012062>



Flujo de trabajo

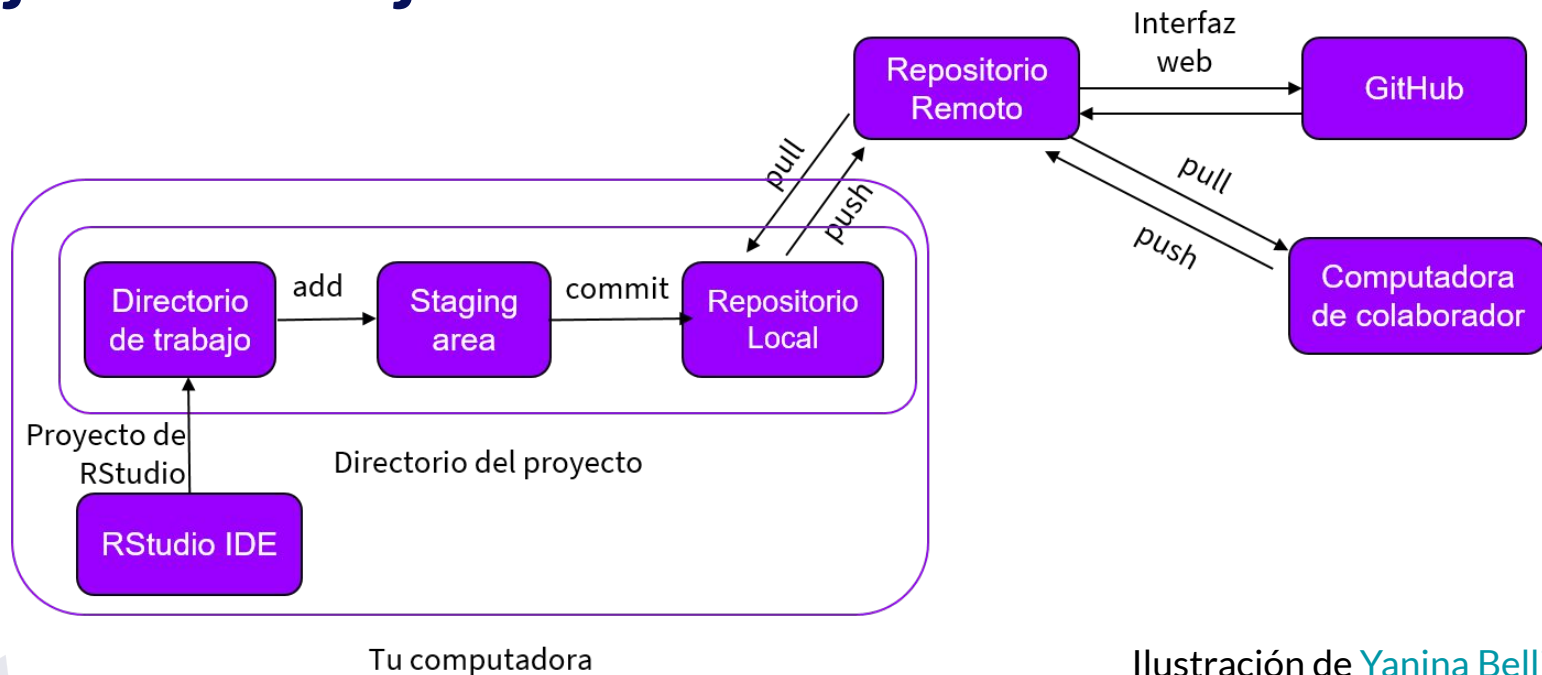


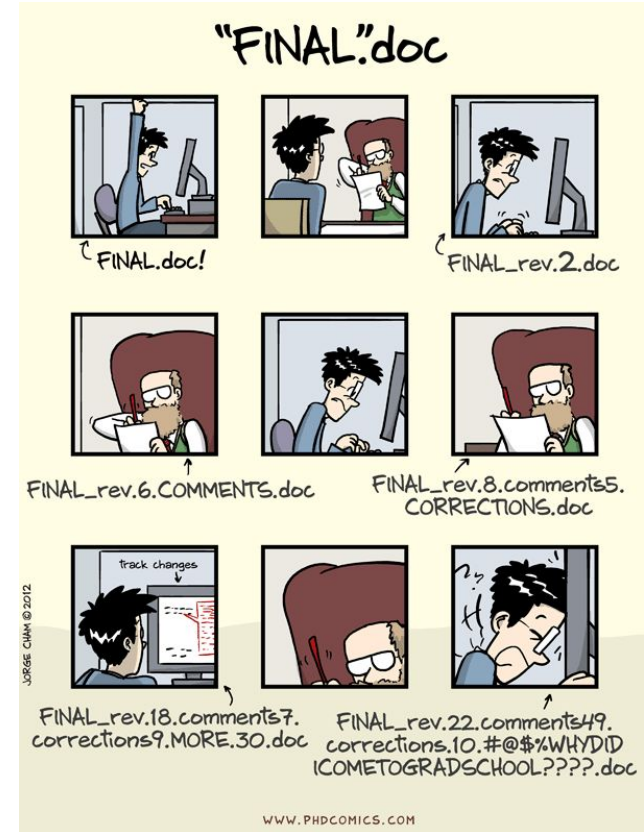
Ilustración de [Yanina Bellini & Marysol Gatti](#) CC-BY-SA

¿Qué es el control de versiones?



Control de versiones

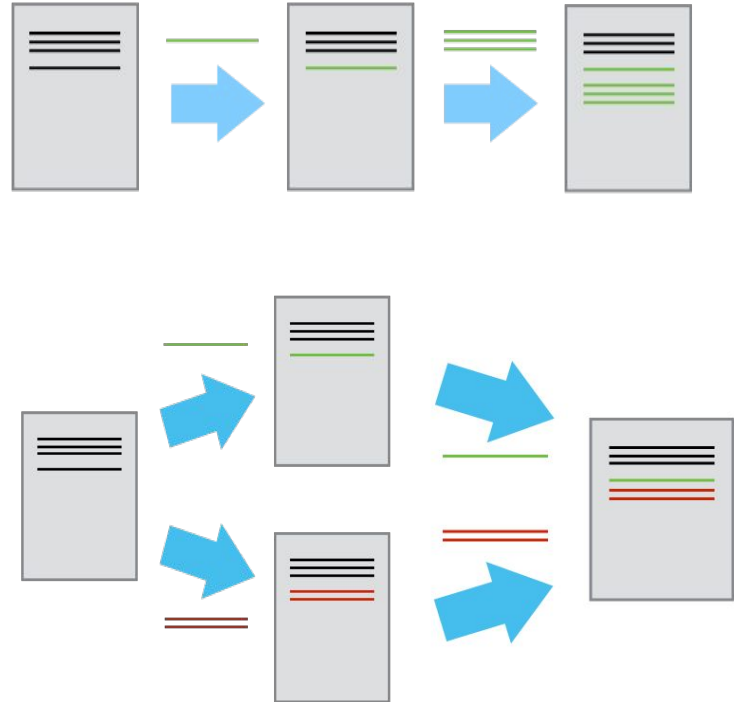
- Seguimiento de los cambios en cada paso.
- Controlar qué cambios incluir en cada versión.
- Mantener metadatos sobre los cambios



Fuente: phdcomics.com

Control de versiones

- Guardar cada paso de un documento
- Trabajo paralelo y fusión de cambios.



Control de versiones

- **Commit**

- El registro de los cambios en un fichero.

- **Repositorio**

- El historial de commits y sus metadatos.

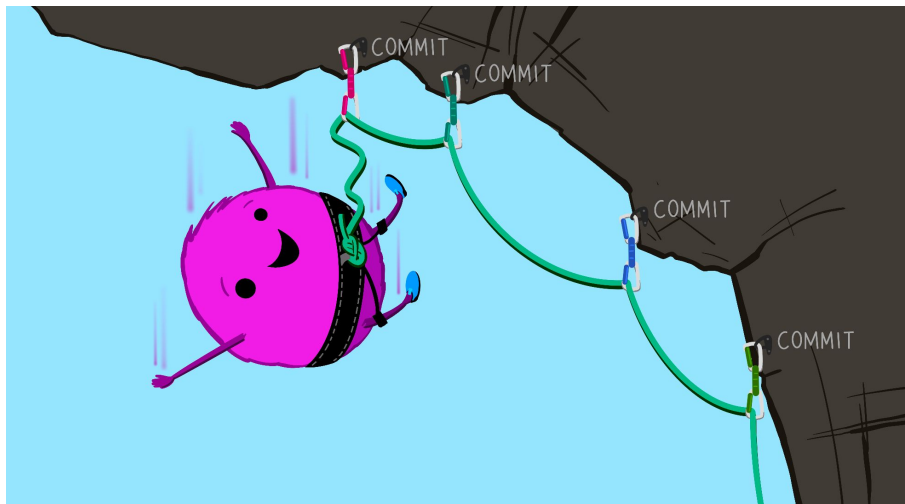


Ilustración de [Allison Horst](#) CC-BY



Comandos básicos

- **git add**
 - Agregar el archivo o directorio al staging area
- **git commit**
 - Guardar los cambios en el repositorio local.

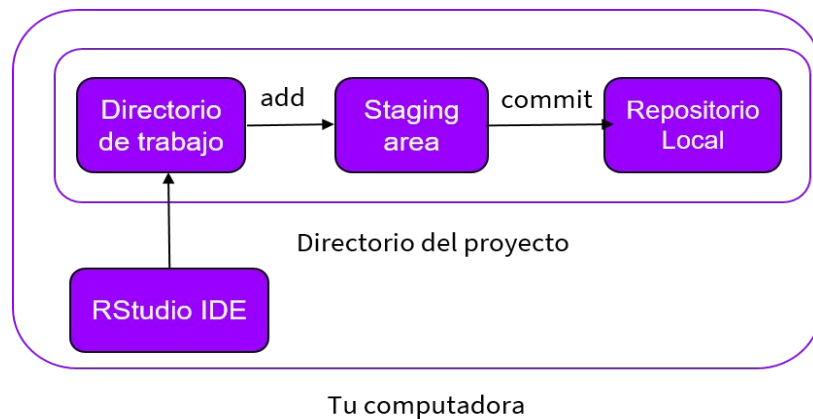


Ilustración de [Yanina Bellini & Marysol Gatti](#) CC-BY-SA



Configurando Git



Configurando Git

- Configuraciones al usar Git por **primera vez**
 - Nombre de usuario
 - Correo electrónico
 - Finales de línea
 - Otras preferencias



Ejercicio: Configuraciones iniciales

- Nombre de usuario

```
$ git config --global user.name "Vlad Dracula"
```

- Correo electrónico

```
$ git config --global user.email "vlad@tran.sylvan.ia"
```

- Opción:

```
$ git config --global user.email "vlad@users.noreply.github.com"
```



Ejercicio: Configuraciones iniciales

- Finales de línea
 - Windows:

```
$ git config --global core.autocrlf true
```

- OS y Linux:

```
$ git config --global core.autocrlf input
```



Resultado: Revisar configuración \$ git config --list

```
MINGW64:/c/Users/agomezvargas
agomezvargas@NE27586 MINGW64 ~
$ git config --global user.name "Soy Andrea"
agomezvargas@NE27586 MINGW64 ~
$ git config --global user.mail "SoyAndrea@users.noreply.github.com"
agomezvargas@NE27586 MINGW64 ~
$ git config --global core.autocrlf true
agomezvargas@NE27586 MINGW64 ~
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Users/agomezvargas/AppData/Local/Programs/Git/mingw64/etc/ssl/
certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.name=Soy Andrea
user.mail=SoyAndrea@users.noreply.github.com
core.autocrlf=true
agomezvargas@NE27586 MINGW64 ~
$
```



Repositorios remotos en GitHub



Repositorios remotos en GitHub

- ¿Cómo podemos compartir los cambios con otras personas en la web?
- GitHub
 - Servicio de alojamiento
 - Un repositorio Git local puede conectarse a un repositorio remoto
- **El primer paso en un nuevo proyecto**



Comandos básicos

- **git pull**
 - Actualiza el repositorio local con el remoto.
- **git push**
 - Envía los cambios del repositorio local al remoto.

● **pull** antes que **push**

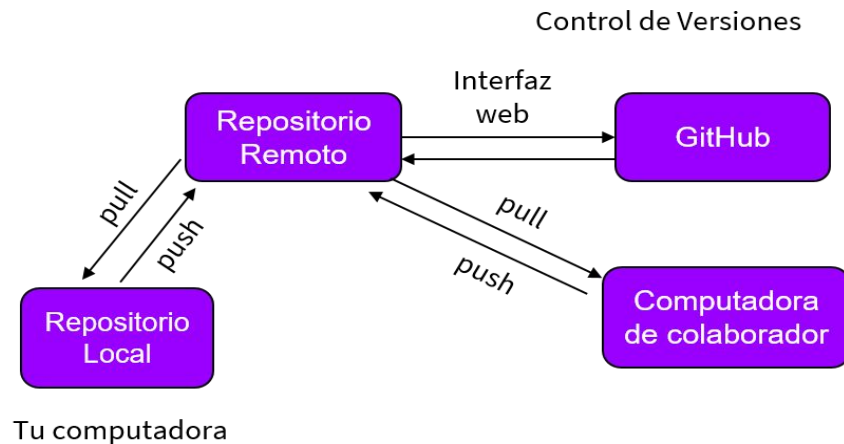
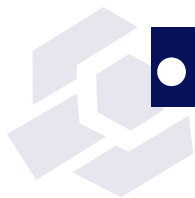
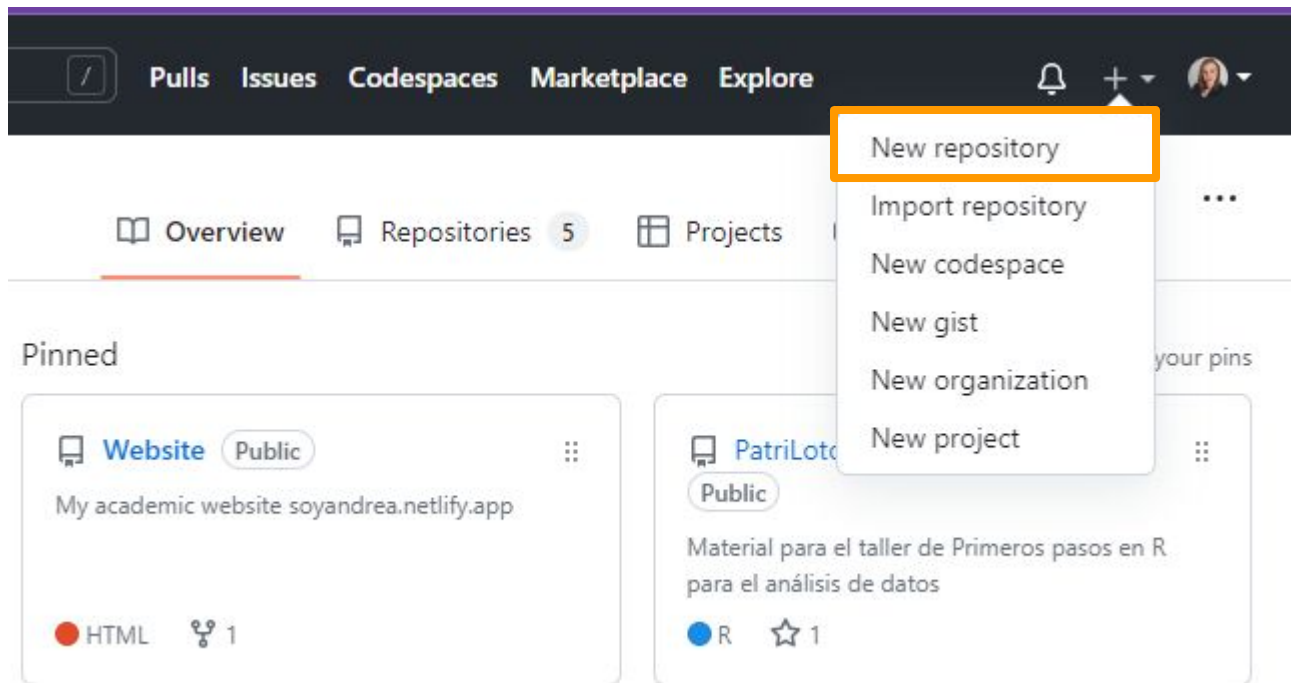


Ilustración de [Yanina Bellini & Marysol Gatti](#) CC-BY-SA



Ejercicio: Crear un repositorio en GitHub

1. clic en signo +,
elegir Nuevo
repo



2. Nos encontramos con la configuración estándar.

Debemos **indicar un nombre** y seleccionar las opciones que deseamos modificar o directamente clic en crear repositorio (**create repository**) con configuraciones por default.



Create a new repository

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

Owner *

SoyAndrea

Repository name *

/ workshopCSV ✓

Great repository names are [workshopCSV is available.](#) Need inspiration? How about [silver-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.](#)

.gitignore template: None

Choose a license

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

License: None

You are creating a public repository in your personal account.

Create repository

Debemos **indicar un nombre**
y seleccionar las opciones que
deseamos modificar




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 *


 SoyAndrea ▾

 /


workshopCSV ✓

Great repository names are workshopCSV is available. Need inspiration? How about [silver-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.

Clic en crear repositorio
(**create repository**) con
configuraciones por
default o modificadas.



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.](#)

.gitignore template: None ▼

Choose a license

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

License: None ▼

 You are creating a public repository in your personal account.

Create repository

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH `https://github.com/SoyAndrea/workshopCSV.git`

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# workshopCSV" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/SoyAndrea/workshopCSV.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/SoyAndrea/workshopCSV.git
git branch -M main
git push -u origin main
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

Tan pronto es creado el repositorio, GitHub te muestra la URL creada para el repositorio seguida de otras opciones de comandos

SoyAndrea / workshopCSV Private

Unwatch 1 Fork 0 Star 0

Code Issues Pull requests Actions Projects Security Insights

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH `https://github.com/SoyAndrea/workshopCSV.git`

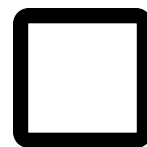
Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.



¡Repositorio creado en GitHub!

Usando Git desde RStudio





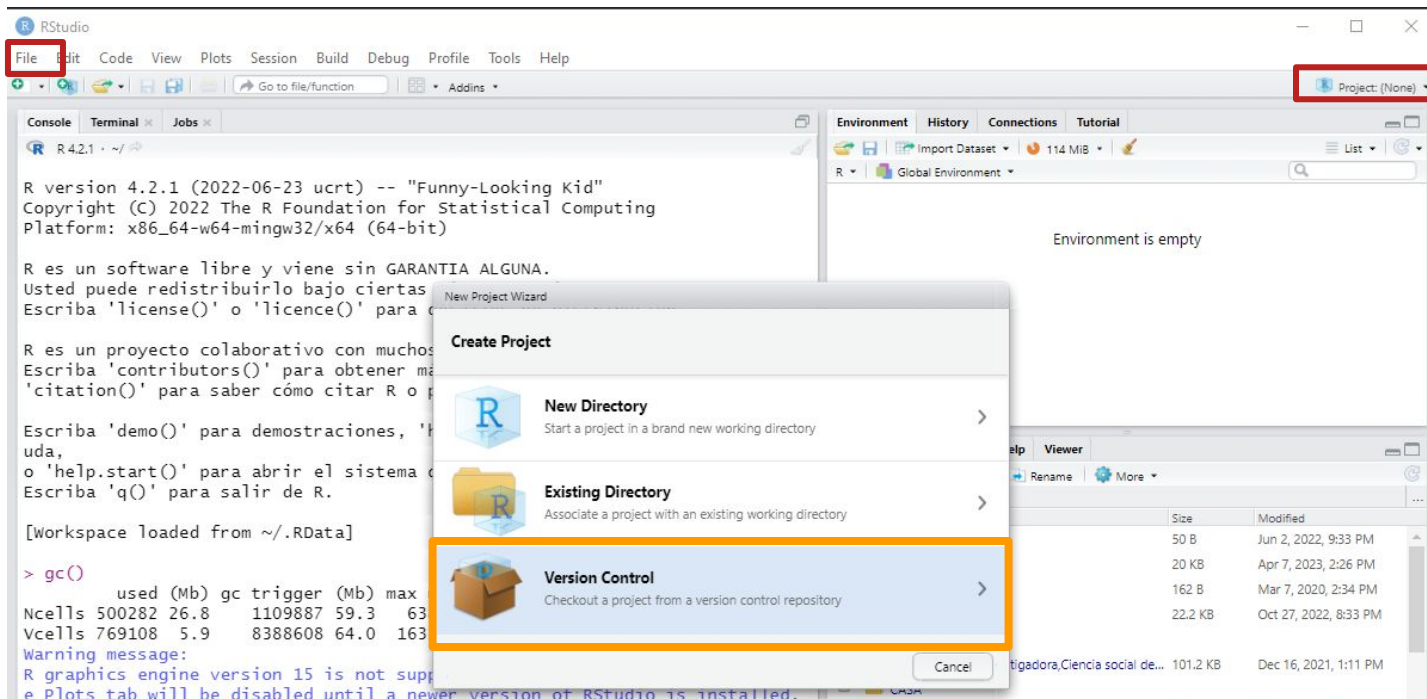
RStudio

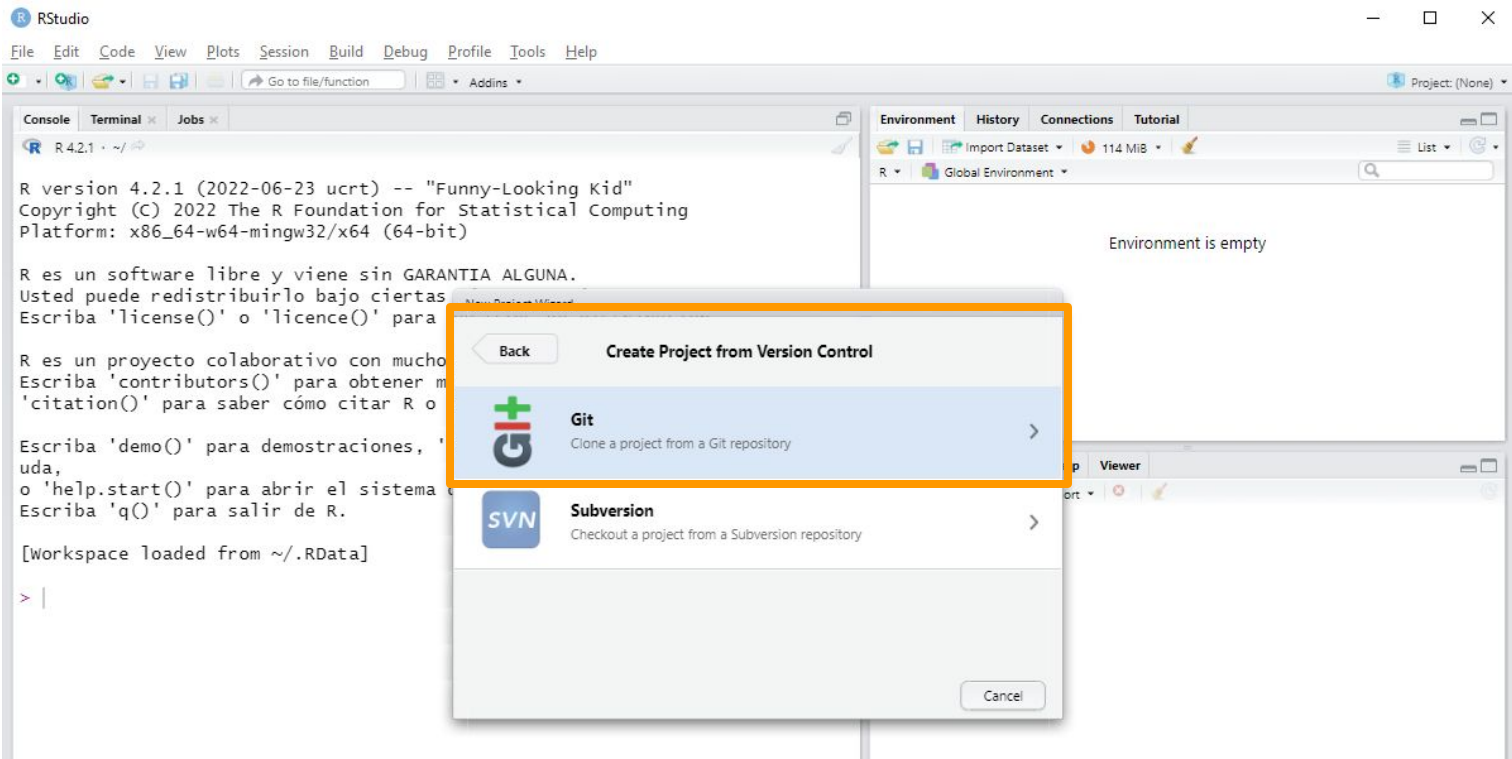
- Tiene una buena **interfaz de trabajo** para realizar las operaciones más comunes de control de versiones con git & github.
- Permite crear un proyecto asociado a un directorio determinado.



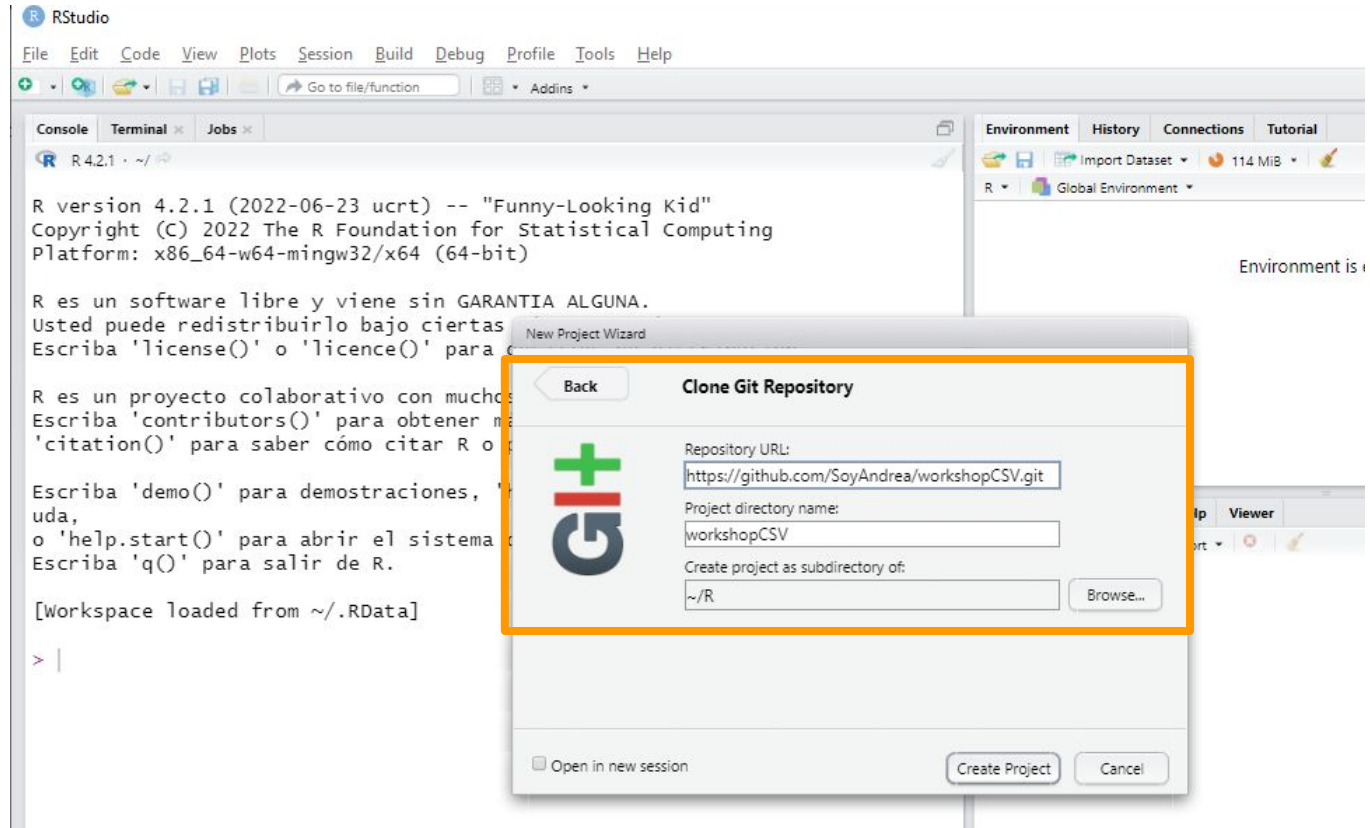
Clonación de un repositorio desde Rstudio

File >
New
Project >
Versión
Control





¡Y ya está! Ahora
tenemos un
proyecto en R que
contiene tu
propio repositorio
de GitHub.



Git desde RStudio

- `git pull`
- `git add`
- `git commit`
- `git push`

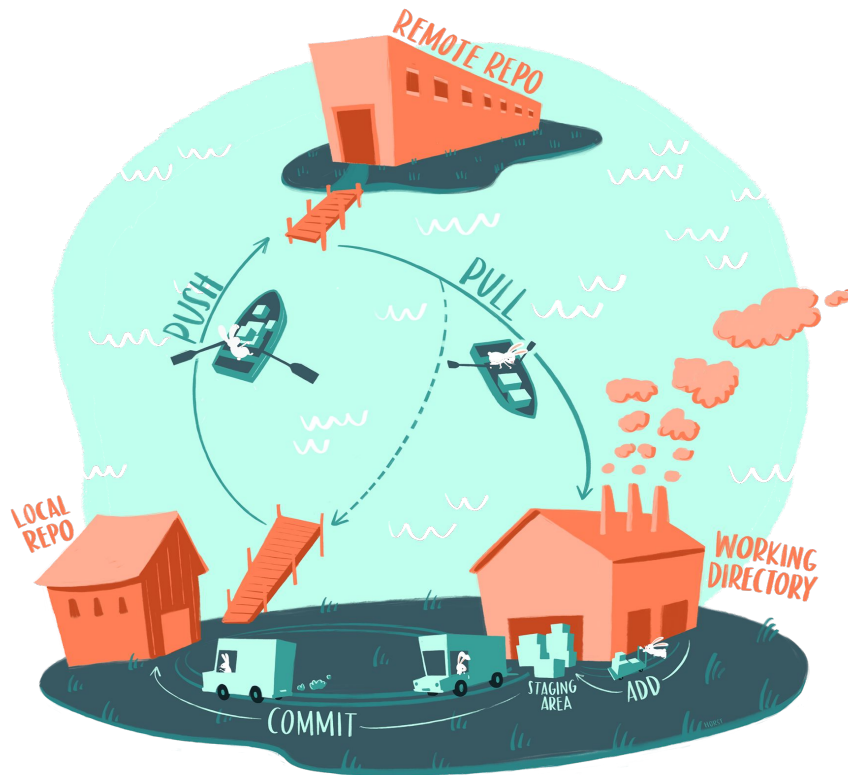
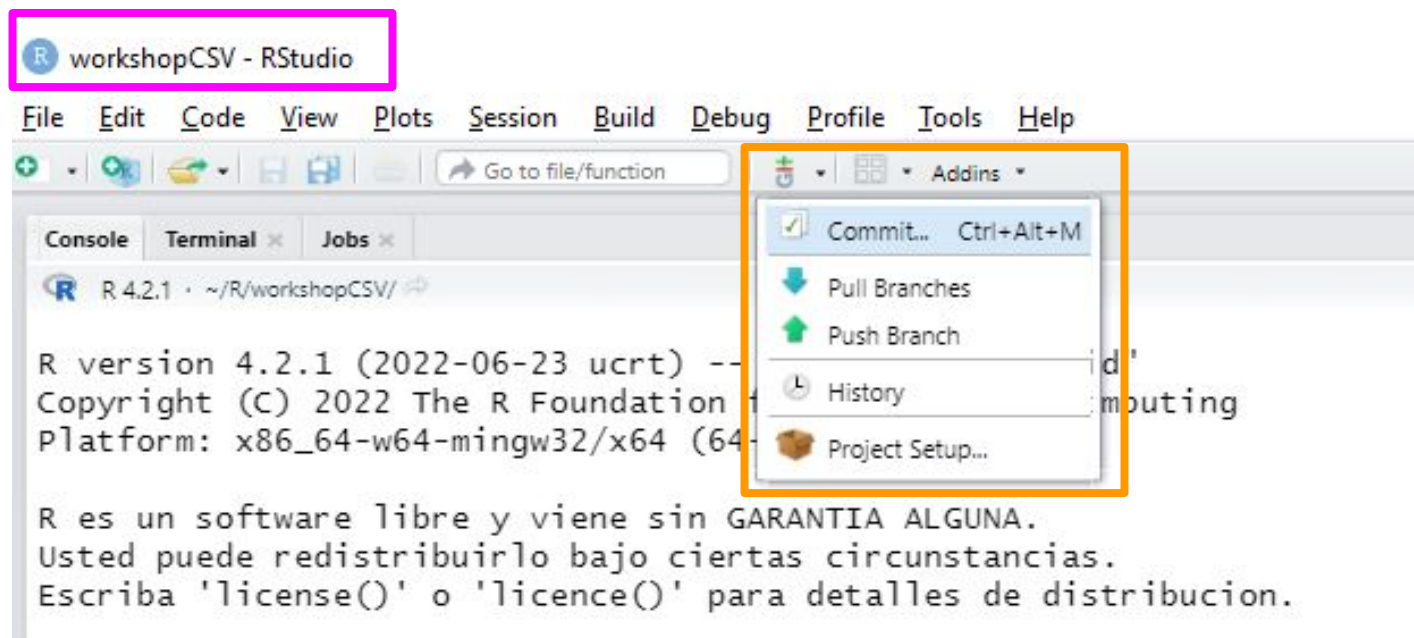


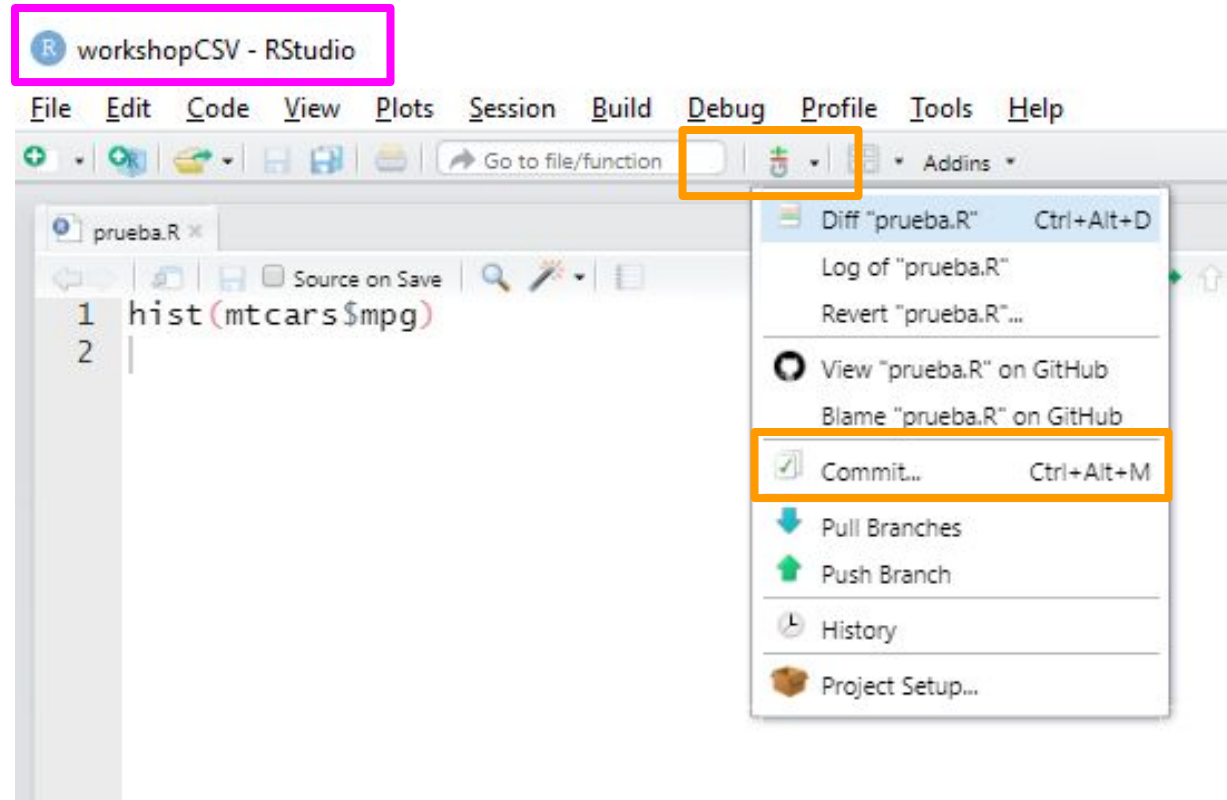
Ilustración de [Allison Horst](#) CC-BY

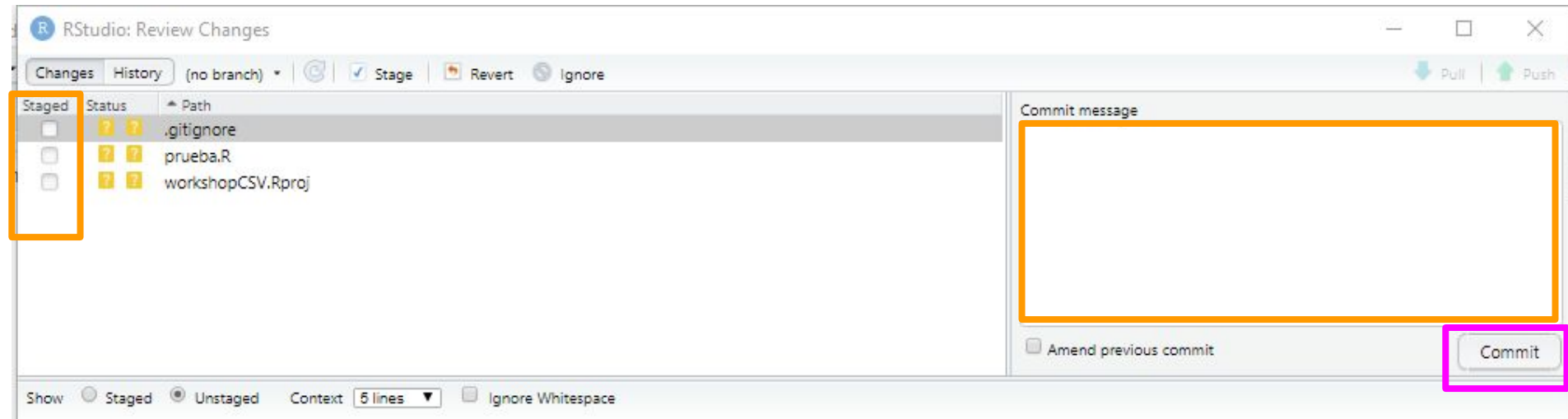
Ejercicio: Cambios desde RStudio



Una vez que hemos guardado nuestros archivos editados, ahora podemos usar RStudio para hacer permanentes los cambios.

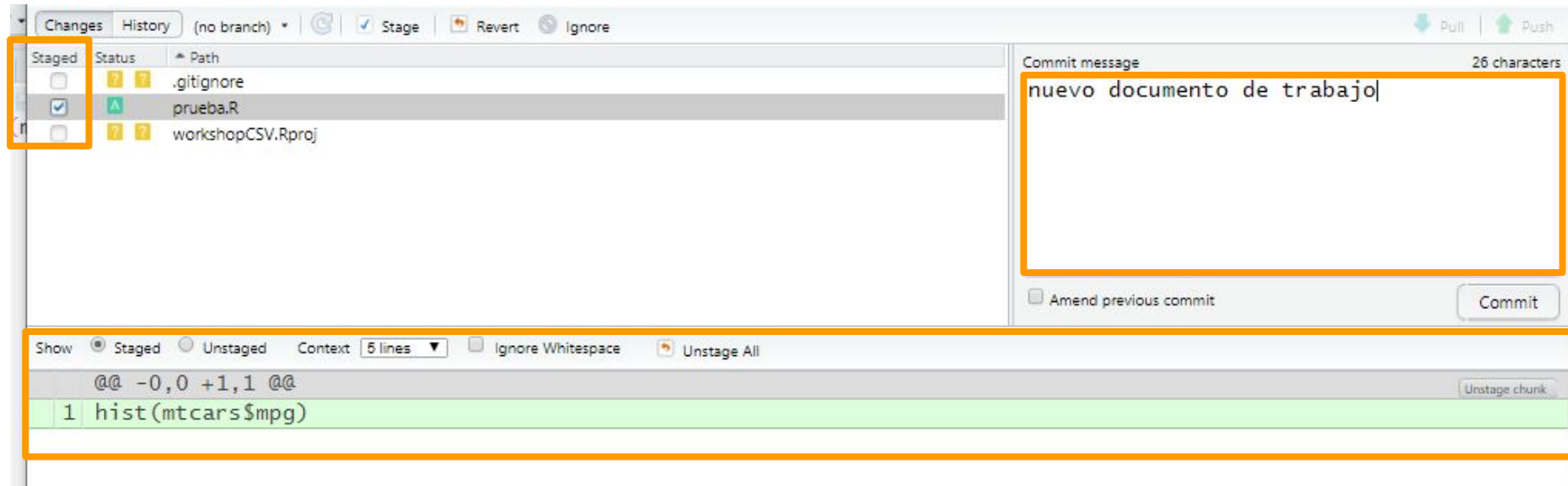
Usa el botón **Git** y haz click a “**Commit**”





Selecciona qué archivos quieres hacer commit (marca las casillas en la columna “**Staged**”) y luego escribe un mensaje para el commit (en el panel superior derecho - **commit message**).





Los iconos en la columna “Status” indican el estado actual de cada archivo. También puedes ver los cambios de cada archivo haciendo click en su nombre. Una vez que todo esté de la forma que quieres, haz clic en “Commit”:



RStudio: Review Changes

Changes History master

| Staged | Status | Path |
|--------------------------|--------|-------------------|
| <input type="checkbox"/> | ? | .gitignore |
| <input type="checkbox"/> | ? | workshopCSV.Rproj |

Show ☒ Staged ☐ Unstaged

Git Commit

Close

```
>>> C:/Program Files/Git/bin/git.exe commit -F C:/Users/Andrea/AppDa  
[master (root-commit) 2acc1c2] nuevo documento de trabajo  
1 file changed, 1 insertion(+)  
create mode 100644 prueba.R
```

Pull Push

Commit



workshopCSV - master - RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function

prueba.R

```
1 hist(mtcars$mpg)
2
```

Diff "prueba.R" Ctrl+Alt+D
Log of "prueba.R"
Revert "prueba.R" ...
View "prueba.R" on GitHub
Blame "prueba.R" on GitHub
Commit... Ctrl+Alt+M
Pull Branches
Push Branch
History

Environment History Connections Git Tutorial

Import Dataset 167 MiB

Global Environment

Environment is empty

RStudio: Review Changes

Changes History master (all commits)

| Subject | Author | Date | SHA |
|--|---|-----------------|----------|
| origin/master nuevo documento de trabajo | SoyAndrea <45582182+SoyAndrea@users.noreply.github> | 4/10/23 3:05 AM | 2acc1c2e |

Commits 1-1 of 1

SHA 2acc1c2e934e726abd1e68e63ad19f861bb98ae0
Author SoyAndrea <45582182+SoyAndrea@users.noreply.github.com>
Date 4/10/23 3:05 AM
Subject nuevo documento de trabajo

Para saber más

- Happy Git and GitHub for the useR:
<https://happygitwithr.com/index.html>



Gracias

[2023-04-17-csvconf-r-git](#)
[carpentries.org](#)

