

Usando o Git/Github no RStudio

Beatriz Pinna

10/06/20

Etapas

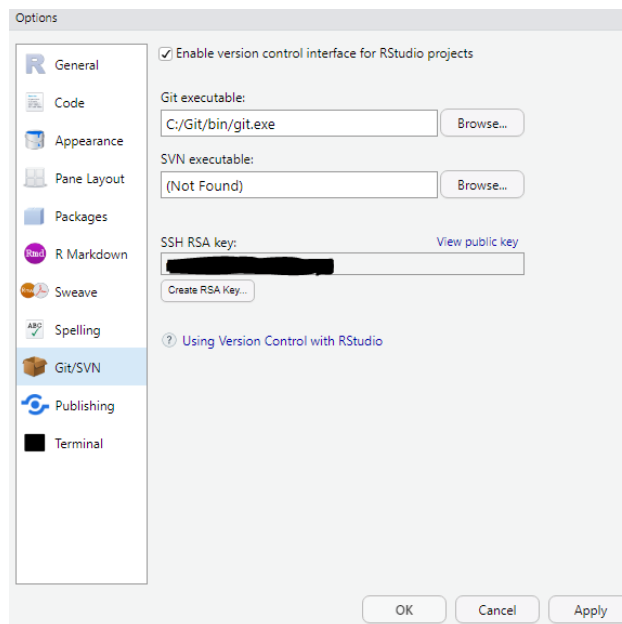
Este documento apresenta uma breve descrição para você que quer ou precisa controlar a versão dos seus projetos no R através do Github. Para isto, vamos seguir estas 3 etapas:

1. Instalar o Git
2. Criar uma conta no Github
3. Controle de Versão no RStudio

Instalação do Git

Git é um sistema de controle de versões distribuído, usado principalmente no desenvolvimento de software, mas pode ser usado para registrar o histórico de edições de qualquer tipo de arquivo. Faça o download do Git por esse link.

Veja se o Git aparece no RStudio em Tools > Global Options > Git/SVN.



Conta no Github

Para que você possa controlar o histórico do seu projeto você precisa criar uma conta gratuita no Github através desse link.

GitHub é uma plataforma de hospedagem de código-fonte com controle de versão usando o Git. Ele permite que programadores, utilitários ou qualquer usuário cadastrado na plataforma contribuam em projetos privados e/ou Open Source de qualquer lugar do mundo.

Controlde de versão no RStudio

Para configurar o Rstudio com o Git direto pelo RStudio é necessário instalar o pacote `usethis()`.

```
install.packages("usethis")  
library(usethis)
```

Após a instalação do pacote, vamos configurar seu e-mail associado ao Github.

```
use_git_config(user.name = "Beatriz Pinna",  
               user.email = "beatrizrpinna@gmail.com")
```

Agora vamos configurar o Github com o Rstudio. Para isso, vamos criar um token no Github utilizando a função a seguir.

```
browse_github_token()
```

Esta função irá abrir uma página do Github para criar o token. Em **Note** você poderá atribuir um novo nome e depois clicar em **Generate Token**. Copie o token gerado e abra o arquivo `.Renviron` com o comando abaixo.

```
edit_r_environ()
```

No arquivo `.Renviron` adicione **GITHUB_PAT=SEU_TOKEN**, salve o arquivo e reinicie o RStudio com o comando CTRL + SHIFT + F10. Agora já podemos trabalhar com projetos no Github clonando um repositório. Para isso, faça as seguintes etapas:


Crie um repositório pelo Github

No Github vá em **Repositories > New** para criar um novo repositório. Em **Repository name** coloque o nome que deseja e adicione uma descrição em **Description**, se quiser. Além disso, é possível escolher a permissão do seu repositório: público ou privado. Para finalizar, marque a opção para adicionar o arquivo README, que é uma introdução do seu projeto e será clonado no seu computador, e clique em **Create repository**.

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 *

 biapinna /

Great repository names are short and memorable. Need inspiration? How about [effective-memory](#)?

Description (optional)

☒  **Public**
Anyone can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

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

☒ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.

Add .gitignore: **None** | Add a license: **None** [?](#)

[Create repository](#)

Crie um projeto de controle de versão pelo RStudio




No Github abra o repositório que você criou e em **Clone or download** copie a URL, como aparece na imagem abaixo.

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

[Manage topics](#)


3 commits | 1 branch | 0 packages | 0 releases

Branch: master | [New pull request](#) | [Create new file](#) | [Upload files](#) | [Find file](#) | [Clone or download](#)

File	Commit
 .gitignore	Initial commit
 README.md	Update README.md
 Teste-Bia.Rproj	Initial commit

Clone with HTTPS [Use SSH](#)

Use Git or checkout with SVN using the web URL.

<https://github.com/biapinna/Teste-Bia.git> 

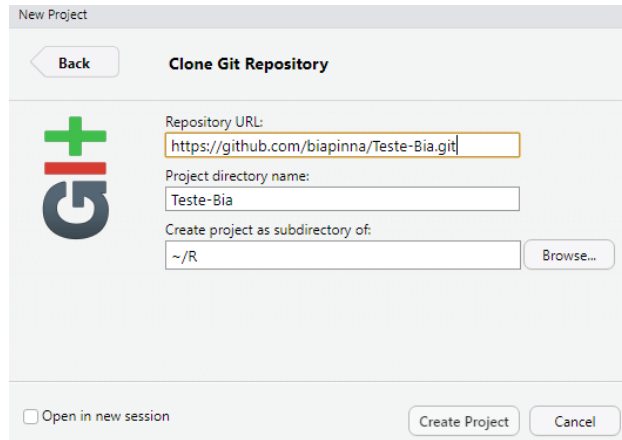
[Open in Desktop](#) | [Download ZIP](#)

README.md [Edit](#)

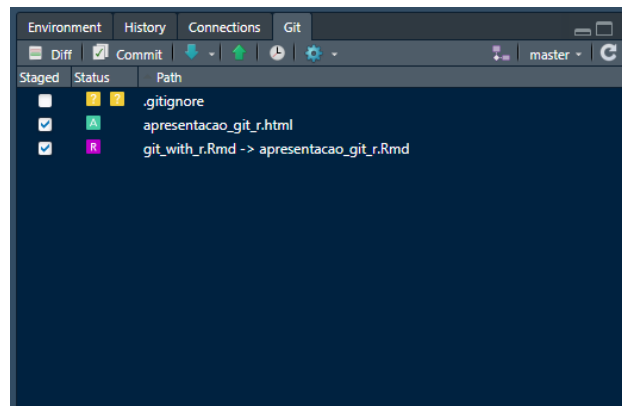
Teste-Bia

The goal of Teste-Bia is to blabla

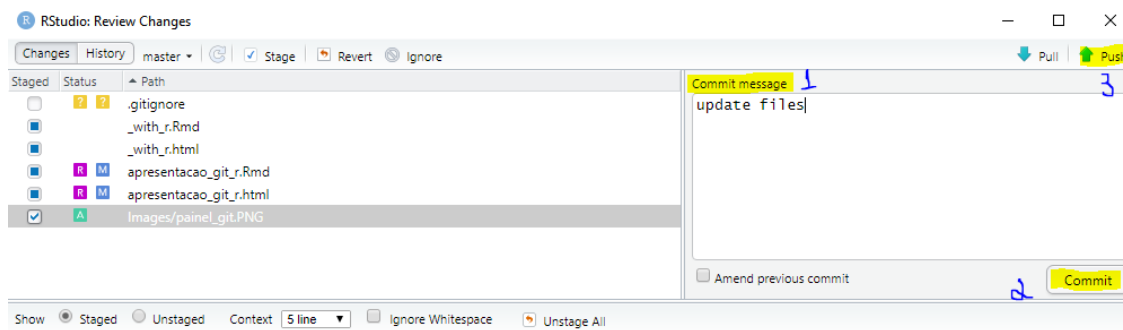
No Rstudio vá em File > New Project > Version Control > Git e crie um novo projeto de controle de versão. Em **Repository URL** cole o link do repositório do Github que você copiou. Após inserir o link, o campo **Project directory name** será preenchido automaticamente, e selecione o diretório em **Create project as subdirectory of** para salvar a cópia local do repositório.



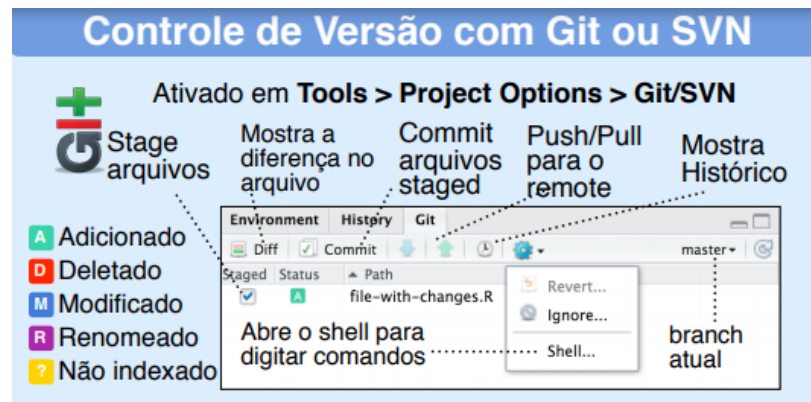
O Git através do RStudio fornece uma interface gráfica simples. Ao abrir o seu projeto versionados com Git no RStudio você verá o Painel do Git como na imagem abaixo.



É necessário selecionar os arquivos na coluna **Staged** para prosseguir com o **Commit**. De acordo com a imagem abaixo, primeiro escreva uma mensagem para identificar o **Commit**, clique no botão commit e aparecerá uma mensagem dos arquivos modificados, feche a mensagem e clique em **Push** para subir os arquivos no Github. O comando **Pull** baixa os códigos no Github para a sua máquina.



Interface gráfica do Rstudio com o Git:



Fonte: RStudio Cheatsheet

Referências

<https://pt.wikipedia.org/wiki/Git>

<https://pt.wikipedia.org/wiki/GitHub>

<https://beatrizmilz.github.io/RLadies-Git-RStudio-2019/#1>

https://github.com/saramortara/R-git-apresentacao/blob/master/trabalhos__reprodutivos.pdf

<https://www.curso-r.com/blog/2017-07-17-rstudio-e-github/>