



UNIVERSIDADE FEDERAL
DO RIO DE JANEIRO



University
of Regina

Práticas em Ciência Aberta

Bruno Soares, Ph.D.
Guilherme Maricato, Ph.D.
Bruno Umbelino, M.Sc.





USO DE GITHUB

TOOLBOX

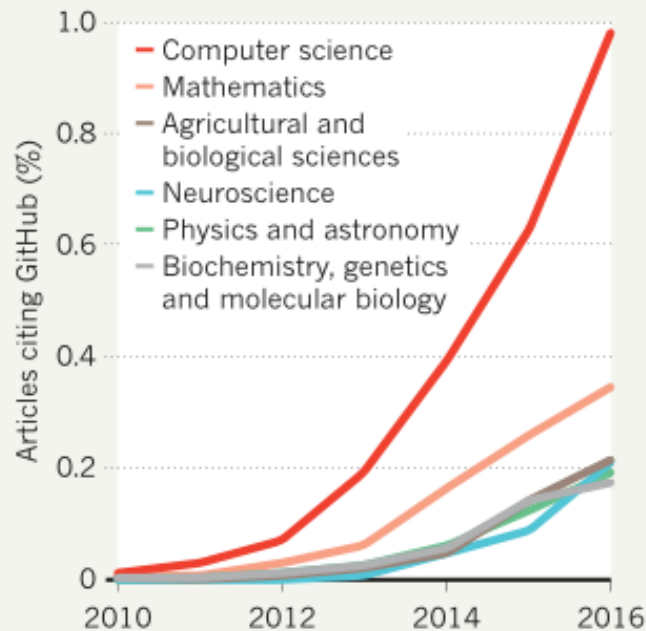
DEMOCRATIC DATABASES: SCIENCE ON GITHUB

Scientists are turning to a software-development site to share data and code.

NATURE | VOL 538 | 6 OCTOBER 2016

GROWING INFLUENCE OF GITHUB

An increasing proportion of research articles cite GitHub in their references.



PROBLEMA

Falta de organização



SOLUÇÃO MARCKDOWN



- Facilita a compreensão dos métodos;
- Métodos e execução *integrados*;
- *Estende* o que está apresentado nos artigos;
- Avançado: *Adeus Office!*

Figura 9.1: Ilustração por Allison Horst.



Sobre

1 Instalação

1.1 Instalação do R

1.2 Instalação do RStudio

1.3 Instalação de softwares adic...

2 RStudio

2.1 Telas

2.2 Atalhos

2.3 Projetos

2.4 Git e versionamento

2.5 Cheatsheets

2.6 Addins



Sobre

Este livro está em **construção** e em **revisão aberta**. Fique no nosso material.

Todo o conteúdo deste livro é gratuito e aberto.

Autoria:

- **Athos Damiani**
- **Beatriz Milz**
- **Caio Lente**
- **Daniel Falbel**
- **Fernando Correa**
- **Julio Trecenti**
- **Nicole Luduvicé**
- **Tereza Lacerda**
- **William Amorim**

Ciência de Dados - livro



Última atualização: 12/07/2022

PROBLEMA

Pacote e script são os mesmos, mas a função deixou de funcionar

What data does R keep if you run *distinct(data, Subject)*?

Depends. When did you last update {dplyr} ?

Subject	dv	condition	mediator
11543	70	treatment	5
11543	70	treatment	5
555	3	control	6
555	3	control	6
47888	110	placebo	3
47888	110	placebo	3

Before June 24 2016

Subject	dv	condition	mediator
11543	70	treatment	5
555	3	control	6
47888	110	placebo	3

After June 24 2016

Subject
11543
555
47888

SOLUÇÃO

Ferramentas que auxiliam no processo
GROUNDHOG e **renv**



Prós

Groundhog

- **Simplicidade:** Groundhog é conhecido por sua facilidade de uso, com comandos intuitivos e uma configuração simplificada.
- **Controle de versão:** Permite especificar versões exatas de pacotes, garantindo que o projeto funcione sempre com as mesmas versões.
- **Facilidade de compartilhamento:** Facilita o compartilhamento de projetos com outras pessoas, pois as dependências são facilmente rastreadas e instaladas.

renv

- **Gerenciamento de ambientes:** Permite criar ambientes isolados para cada projeto, evitando conflitos entre diferentes versões de pacotes.
- **Restauração de ambientes:** Facilita a restauração de ambientes, garantindo que o projeto possa ser reproduzido em qualquer máquina.
- **Integração com Git:** Integra-se perfeitamente com o Git, permitindo rastrear as dependências do projeto no histórico de commits.
- **Comunidade ativa:** Possui uma comunidade grande e ativa de usuários, oferecendo suporte e recursos abundantes.

Contras

Groundhog

- **Menos recursos:** Comparado ao renv, o Groundhog oferece menos recursos e opções de personalização.
- **Comunidade menor:** Possui uma comunidade menor de usuários, o que pode dificultar a obtenção de suporte em caso de problemas.
- **Foco em versões:** O foco principal é no controle de versões, não oferecendo tantos recursos para gerenciamento de ambientes.

renv

Maior complexidade:

Comparado ao Groundhog, o renv possui uma curva de aprendizado mais comandos e opções de configuração.

Requer mais cuidado: Exige mais cuidado na configuração e no gerenciamento dos ambientes para evitar problemas.

Dependência de projetos: Os ambientes são específicos para cada projeto, o que pode gerar duplicação de pacotes.



Qual é a melhor opção?

DEPENDE!!

A escolha entre Groundhog e renv dependerá das suas necessidades e preferências.

- Se você busca **simplicidade e facilidade de uso**, o Groundhog pode ser uma boa opção.
- Se você precisa de **recursos avançados para gerenciamento de ambientes e integração com Git**, o renv é a escolha mais recomendada.

RENV: comparing groundhog with renv

[🏠](#) > [RENV: comparing groundhog with renv](#)

Comparing 'groundhog' with 'renv'

last update: March 30th, 2022

tldr; academics should probably use groundhog, corporate data scientists should arguably use renv.

Probably the best well-known solution for version-control in R is the package [renv](#), which is a revamped and reformed version of the older packrat package. Here we focus on the relative advantages and disadvantages of renv vs groundhog.

renv's advantages

1. renv is integrated with R Studio
2. renv is developed by professional software engineers (rather than academic researchers)
3. renv works with more repositories than does groundhog (in addition to groundhog's CRAN, GitHub and GitLab, renv also works with BitBucket and BioConductor). However, adding BioConductor to the set of repositories groundhog works with is top of the to-do list.

groundhog advantages.

1. groundhog allows sharing individual .R scripts that are reproducible, with renv you have to work within a project structure.
2. groundhog does not require additional attachment files to make R code reproducible, with renv you have to also share a 'lockfile' (a textfile listing required package versions) and possibly several additional configuration files.
3. groundhog requires no learning curve, you simply rely on `groundhog.library()` instead of `library()`. Nothing else changes. renv, in contrast, requires several additional setup and ongoing steps to attain version-control. See the ['workflow'](#) that is recommended for its use. For instance, users need to actively save the packages they have loaded into their environment using the

Table Of Contents

- Nutshell
- About
- User guide
- Changelog
- GRAN: Groundhog's Archive of all CRAN binaries
- Comparing groundhog with 'renv'
- Comparing {groundhog} with Posit's P3M manager
- R Tools
- The Back End (how groundhog works)
- Running older versions of R
- Troubleshooting



GROUNDHOG

```
16 # instalando os pacotes
17 install.packages(c("ggplot2", "dplyr", "tidyr", "readxl", "parallel", "florabr"))
18
19 # carregando os pacotes
20 packages ← c("ggplot2", "dplyr", "tidyr", "readxl", "parallel", "florabr")
21 lapply(packages, library, character.only = TRUE)
22
23 # garantindo que as versoes dos pacotes utilizados serao as mesmas ao longo de toda a analise
24 install.packages("groundhog") #caso já tenha instalado, so carregar o pacote
25 library(groundhog)
26 groundhog.day="2024-09-20" #formato de data ano-mes-dia de criação do script/atualizacao
27 pkgs=c("ggplot2", "dplyr", "tidyr", "readxl", "parallel", "florabr")
28 groundhog.library(pkgs, groundhog.day)
29
```

Estrutura mínima de repositório “ideal”



- Segue uma estrutura hierárquica;
- Separação de arquivos;
- Nomes curtos (sem caracteres especiais);
- Não é um esquema absoluto;
- README é essencial

Estrutura mínima de repositório “ideal”




O tipo de pasta e conteúdo
vai depender do SEU
JULGAMENTO de
reprodutibilidade e a
especificidade da área de
pesquisa.

Repositório do capítulo da qualificação

biodiversity_ignorance_first_chapter (Private) Unwatch 3

main 1 Branch 0 Tags Add file Code

 BrunoUmbelino21

salvando o script r descrevendo os arquivos das pastas

0a32d44 · 5 months ago

10 Commits

PRISMA	finalizando alterações	5 months ago
RawData	finalizando alterações	5 months ago
figures	finalizando alterações	5 months ago
manuscript_file	finalizando alterações	5 months ago
table	finalizando alterações	5 months ago
.gitignore	Final version of files from thesis chapter one about systemati...	7 months ago
README.md	salvando o script r descrevendo os arquivos das pastas	5 months ago
biodiversity_ignorance_first_chapter.Rproj	Final version of files from thesis chapter one about systemati...	7 months ago

README + ...

General Overview

This repository contains all data used to build the paper "Perceptions of shortfalls in biodiversity research, insights and applications: a systematic review", part of my thesis project in developing at postgraduate in Ecology on Universidade Federal do Rio de Janeiro.

In this study we brought secondary data from literature published on Web of Science and Scopus database. For get those data we used Boolean terms "biodiversity AND ignorance".

Repository structure

Folders

- . RawData: original reference tables from SCOPUS in .xlsx format and from WOS in .txt format.
- . manuscript_file: manuscript about biodiversity ignorance. There are two types of files: one in .pdf format and another in .doc
- . PRISMA: there is a diagram from methodology applied
- . table: contain one .csv file with all references/studies remain.
- . figures: there is one file of integrative view of shortfalls with influences from human actions.

README + ...

This repository contains all data used to build the paper "Perceptions of shortfalls in biodiversity research, insights and applications: a systematic review", part of my thesis project in developing at postgraduate in Ecology on Universidade Federal do Rio de Janeiro.

In this study we brought secondary data from literature published on Web of Science and Scopus database. For get those data we used Boolean terms "biodiversity AND ignorance".

Repository structure

Folders

- . RawData: original reference tables from SCOPUS in .xlsx format and from WOS in .txt format.
- . manuscript_file: manuscript about biodiversity ignorance. There are two types of files: one in .pdf format and another in .doc
- . PRISMA: there is a diagram from methodology applied
- . table: contain one .csv file with all references/studies remain.
- . figures: there is one file of integrative view of shortfalls with influences from human actions.

Downloading this repository

The user can download this repo to a local folder in your computer or clone it:

```
download.file(url = "https://github.com/BrunoUmbelino21/biodiversity_ignorance_first_chapter/archive/main.zip", destfile = "biodiversity_ignorance_first_chapter.zip")
```

to unzip the .zip file in your computer type

```
unzip(zipfile = "biodiversity_ignorance_first_chapter.zip")
```

Authors

Bruno Umbelino, Stephanie Vaz, Mauricio de Almeida-Gomes, Marcus Vinicius Vieira

Outros exemplos

There and back to the present: a model-based framework to estimate phylogenetically constrained alpha diversity gradients

DOI [10.1101/2021.12.11.472171](https://doi.org/10.1101/2021.12.11.472171)

DOI [10.5281/zenodo.8136692](https://doi.org/10.5281/zenodo.8136692)

Authors

Leandro Duarte, Gabriel Nakamura, Vanderlei Debastiani, Renan Maestri, Maria João Veloso da Costa Ramos Pereira, Marcus Cianciaruso, José Alexandre F. Diniz-Filho

Repository overview

This repository contains data and R scripts that allows to reproduce the analyses presented in the manuscript entitled "There and back to the present: a model-based framework to estimate phylogenetically constrained alpha diversity gradients", in which we introduce an individual-based simulation approach coupled with Approximate Bayesian Computation (ABC) that allows to parameterize adaptation rates of species niche positions along the evolution of a monophyletic lineage, as well as the intensity of dispersal limitation between local assemblages potentially connected by dispersal (metacommunity). The analytic tool was implemented in an R package called [mcfly](#). A preprint version of this work can be found [here](#).

Obs -The preprint does not correspond to the most updated version of the study.

If you want to use this repository you can clone or download to your computer machine using the following code:

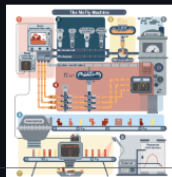
To clone using github from command line:

```
git clone https://github.com/GabrielNakamura/Data_MS-mcfly your_repo_title
```

To download using R:

```
download.file(url = "https://github.com/GabrielNakamura/Data_MS-mcfly/archive/main.zip", destfile = "Da
```

Repository structure



README

README and METADATA

Original study published as Preprint in EcoEvoRxiv "Is the audience gender-blind? Smaller audience in female talks highlights prestige differences in academia" (<https://doi.org/10.32942/X25607>). Submitted for recommendation to Peer Community in Ecology.

The present repository contains data, metadata, and modeling analysis source codes necessary to reproduce the EcoEncontros gender bias results. The EcoEncontros is a seminar series of weekly talks at the Ecology Graduate Program at the University of São Paulo (PPGE, IB-USP). The information from all talks between 2008 and 2019 from the EcoEncontros committee attendance list archives (N=344 talks) are presented. We retrieved data about the speaker (gender, academic level, and affiliation) and the seminar (date, title, abstract, and audience size). We inferred the speaker's gender by name and photo (always present on the seminars' posters). We have removed the speakers' names from the data files provided in order to maintain their anonymity.

The content is organized into four sections, each with one specific aim:

- `@_data_summary` : data description, wrangling, and summary statistics. It starts with raw presentation data from 2008 to 2019 (data folder; see metadata below).
- `1_speakers_genderPosition` : modeling source code of objective 1 analyses on gender bias in speakers and academic level. Here, we assess the proportions of female speakers by academic level, before and after affirmative actions. We do an extra analysis with only the speakers belonging to the PPGE community, to evaluate if the proportion of female speakers is proportional to the proportion of female academics in the PPGE community.
- `2_audience_genderPosition` : modeling source code of objective 2 analyses on gender bias in audience size of seminars and academic level. Here, the audience size (# number of attendants in the seminar) is modeled by the gender (male, female), the academic level (student, postdoc, professor) of the speaker, and affirmative actions (before and after).
- `3_text_genderAnalysis` : modeling source code of objective 3 containing text analysis from titles and abstracts of the talks.

When using this data, please cite the Zenodo permanent version of this repository:

Leite, Melina de Souza & Rodrigues Barreto, Júlia (2024). Data and Code from: Is the audience gender-blind? Smaller audience in female talks highlights prestige differences in academia (v0.9.9). Zenodo.
<https://doi.org/10.5281/zenodo.11288110>.

Metadata of the raw data file "presentations_PPGE_2008-2019.csv"

Variables:

- `id` = Unique label of presentation ordered in date.
- `date` = date of presentation in "DD-MM-YYYY" format.

Caminho para repositório online e desktop





ABC DOS TERMOS E FUNÇÕES

Clone

- Clonar (clone) nesse caso não é nada mais que criar uma cópia exata de um repositório remoto em sua máquina;
- Facilita o processo de não pensar;
- Não vai ser criminalizado por plágio;
- Segue o mesmo caminho de criar um repositório que VOCÊ CRIOU DO ZERO.



ABC DOS TERMOS E FUNÇÕES

Fork

- Você pode usar um fork para experimentar o código de um projeto existente, aprender novas técnicas e testar suas ideias sem o risco de danificar o projeto original;
- Você pode fazer alterações na sua cópia e em seguida enviar um **pull request** para o repositório original (mediante a aprovação de envio);
- Você pode usar um fork como base para criar seu próprio projeto, aproveitando o código existente e adaptando às suas necessidades.



ABC DOS TERMOS E FUNÇÕES

Qual a diferença entre Clone e Fork?

- **Clone:** Cria uma cópia do repositório na sua máquina local, permitindo que você faça alterações e envie para o repositório original (se tiver permissão).
- **Fork:** Cria uma cópia independente do repositório na sua conta do GitHub, permitindo que você faça alterações sem afetar o repositório original.



ABC DOS TERMOS E FUNÇÕES

Commit (“diário”)

- É uma forma de registrar as modificações que você fez em seus arquivos (adicionando, removendo ou alterando conteúdo) e fornecer uma mensagem descritiva sobre o que foi alterado;
- Isso permite acompanhar o progresso do projeto e entender como ele evoluiu ao longo do tempo;
- Isso permite acompanhar o progresso do projeto e entender como ele evoluiu ao longo do tempo;
- Se algo der errado, você pode voltar a um commit anterior para restaurar uma versão estável do seu projeto.



ABC DOS TERMOS E FUNÇÕES

Push

- Envia suas alterações locais para o repositório remoto;
- É como "enviar" suas atualizações para o servidor, tornando-as acessíveis para outros membros do projeto;
- Geralmente usado após fazer commits;



ABC DOS TERMOS E FUNÇÕES

Pull Request (PR) ou só Pull

- É usado para trazer as alterações (commits e branch) do repositório remoto para o seu repositório local;
- É como "baixar" as atualizações feitas por outros membros do projeto;
- Garante que sua cópia local esteja sempre atualizada com a versão mais recente do projeto;



ABC DOS TERMOS E FUNÇÕES

Branches (pastas)

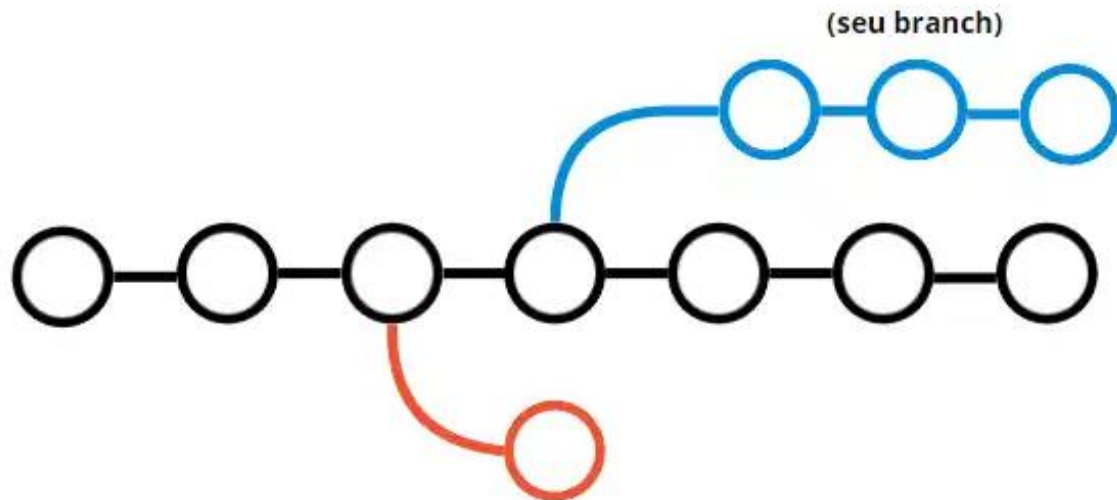
- É usado para trazer as alterações (commits) do repositório remoto para o seu repositório local;
- Várias pessoas podem trabalhar em diferentes funcionalidades ao mesmo tempo, cada uma em sua própria branch.
- Cada branch representa uma tarefa ou funcionalidade específica, facilitando o acompanhamento do progresso;
- Várias pessoas podem trabalhar em diferentes funcionalidades ao mesmo tempo, cada uma em sua própria branch.



Feature

Linha Principal

Bug



branch alternativa

commit
2.1

commit
2.2

**branch
master**

commit1

commit2

commit3

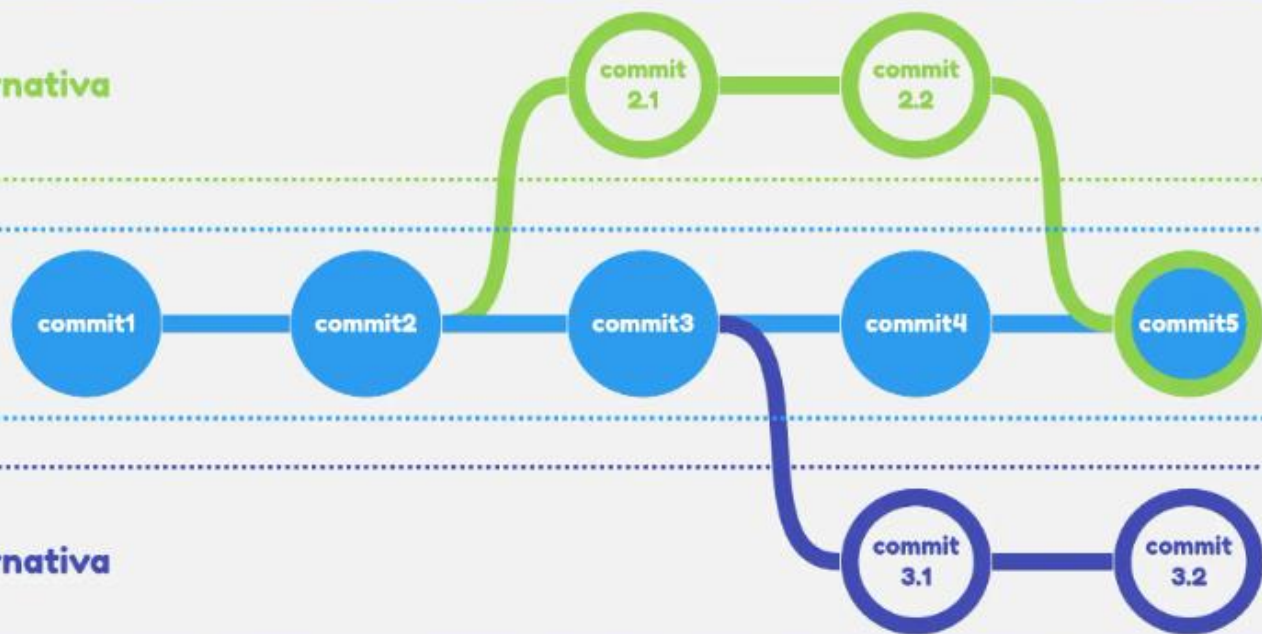
commit4

commit5

branch alternativa

commit
3.1

commit
3.2



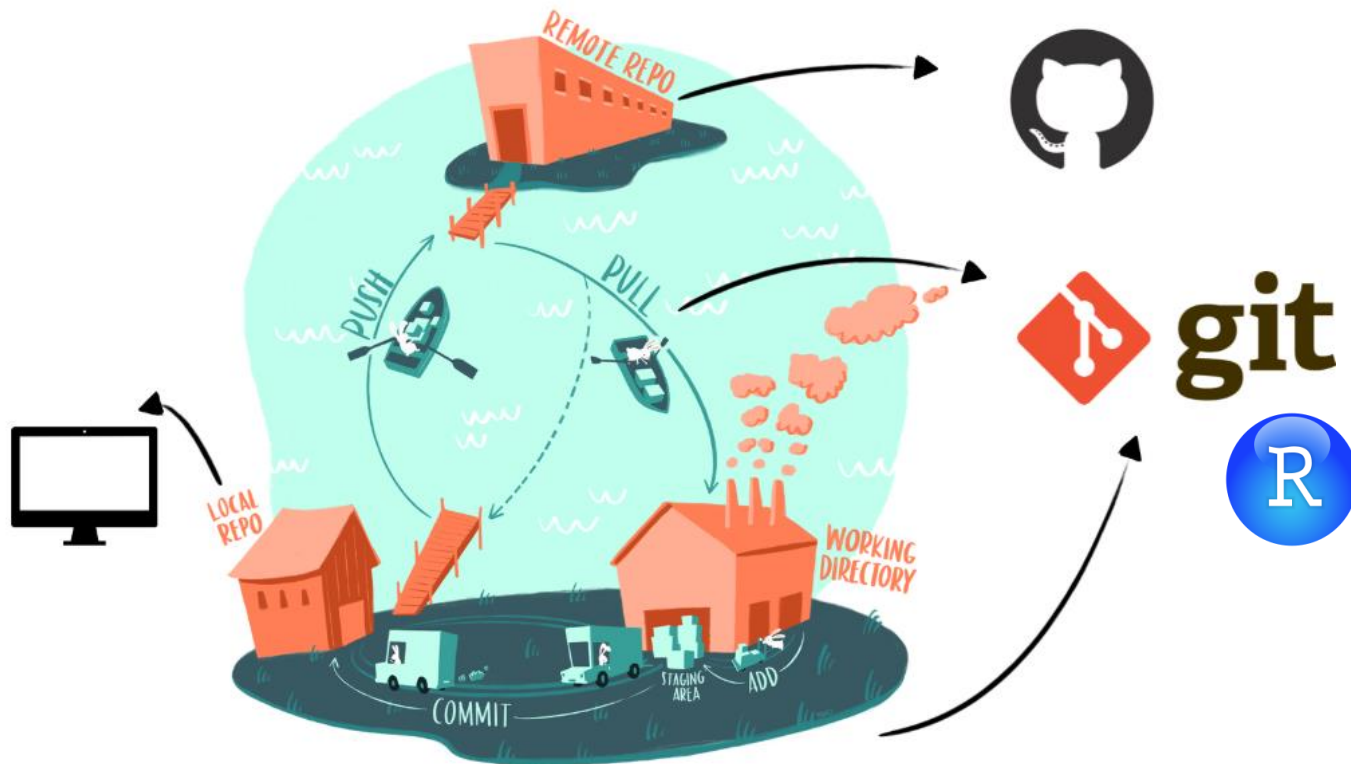


ABC DOS TERMOS E FUNÇÕES

Gitignore

- É um arquivo de texto especial que reside na raiz do seu repositório Git e serve como um "guardião", instruindo o Git sobre quais arquivos e pastas ele deve ignorar.
- Arquivos de configuração ou dados pessoais podem conter informações sensíveis que você não deseja compartilhar publicamente;
- Arquivos temporários gerados pelo seu editor de código ou sistema operacional são irrelevantes para o histórico do seu projeto e podem gerar confusão.

Caminho para repositório online e desktop





Coffee break



Passo 1: baixar o Git



The screenshot shows the 'git for windows' website. At the top left, the logo 'git for windows' is displayed, with 'VERSION 2.48.1' below it. To the right of the logo are links for 'FAQ', 'REPOSITORY', and 'MAILING LIST'. A large orange diamond containing a white branching diagram is on the left. To its right, the text reads 'We bring the awesome **Git** SCM to Windows'. Below this text are two blue buttons: 'Download' and 'Contribute'. Under the buttons is the URL 'https://gitforwindows.org/'. At the bottom, a blue banner contains the text 'Tools & Features'.

git for windows

FAQ REPOSITORY MAILING LIST

VERSION 2.48.1

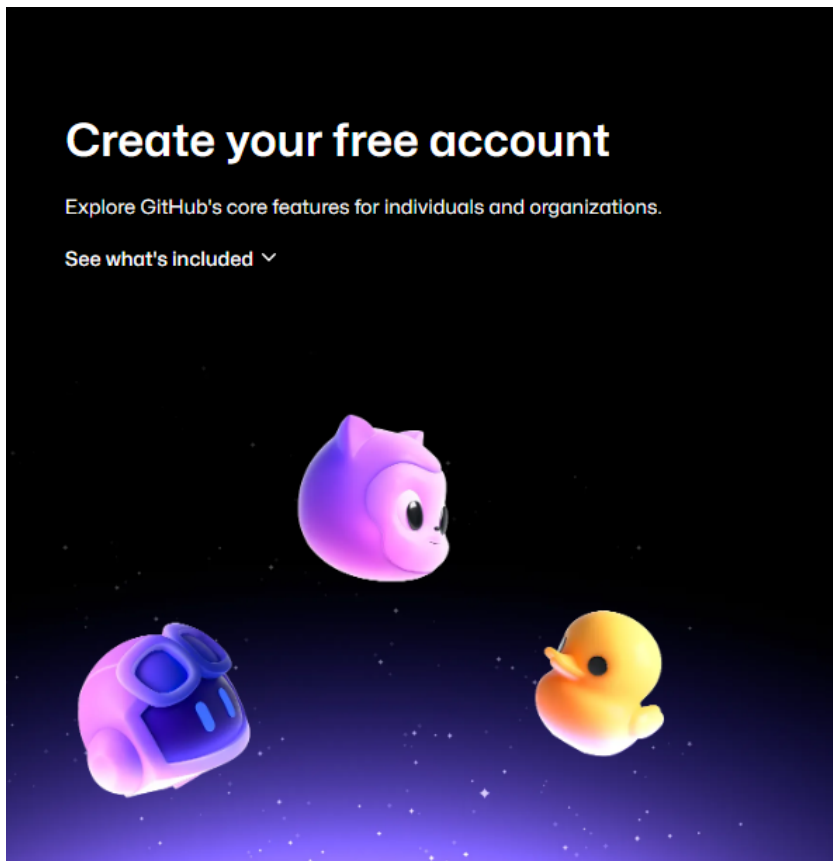
We bring the
awesome **Git** SCM to
Windows

Download Contribute

<https://gitforwindows.org/>

Tools & Features

Passo 2: criar uma conta no GitHub



Already have an account? [Sign in →](#)

Sign up to GitHub

Email*

Password*

Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Username*

Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen.

[Continue >](#)

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.

Curso de Ciência Aberta – USP

