

A Generalized size-structure assessment platform for Crustacea



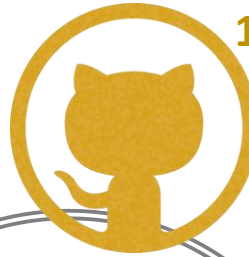
GMACS 2.01

The ‘Generalized Model for Assessing Crustacean Stocks’

Matthieu Veron and
a whole world

Crab Plan Team Meeting
January 2023
Anchorage, Alaska

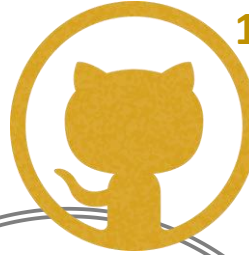
Outline of this “presentation”



1. A GitHub organization

- Various repositories
 - ✓ What can be found in this organization?
 - ✓ Which ones are useful for users and developers?
- How to copy/clone/fork a repository of interest ?
- How can I contribute to a repository?

Outline of this “presentation”



1. A GitHub organization

- ✓ Various repositories
- ✓ How to copy/clone/fork a repository of interest?
- ✓ How can I contribute to a repository?

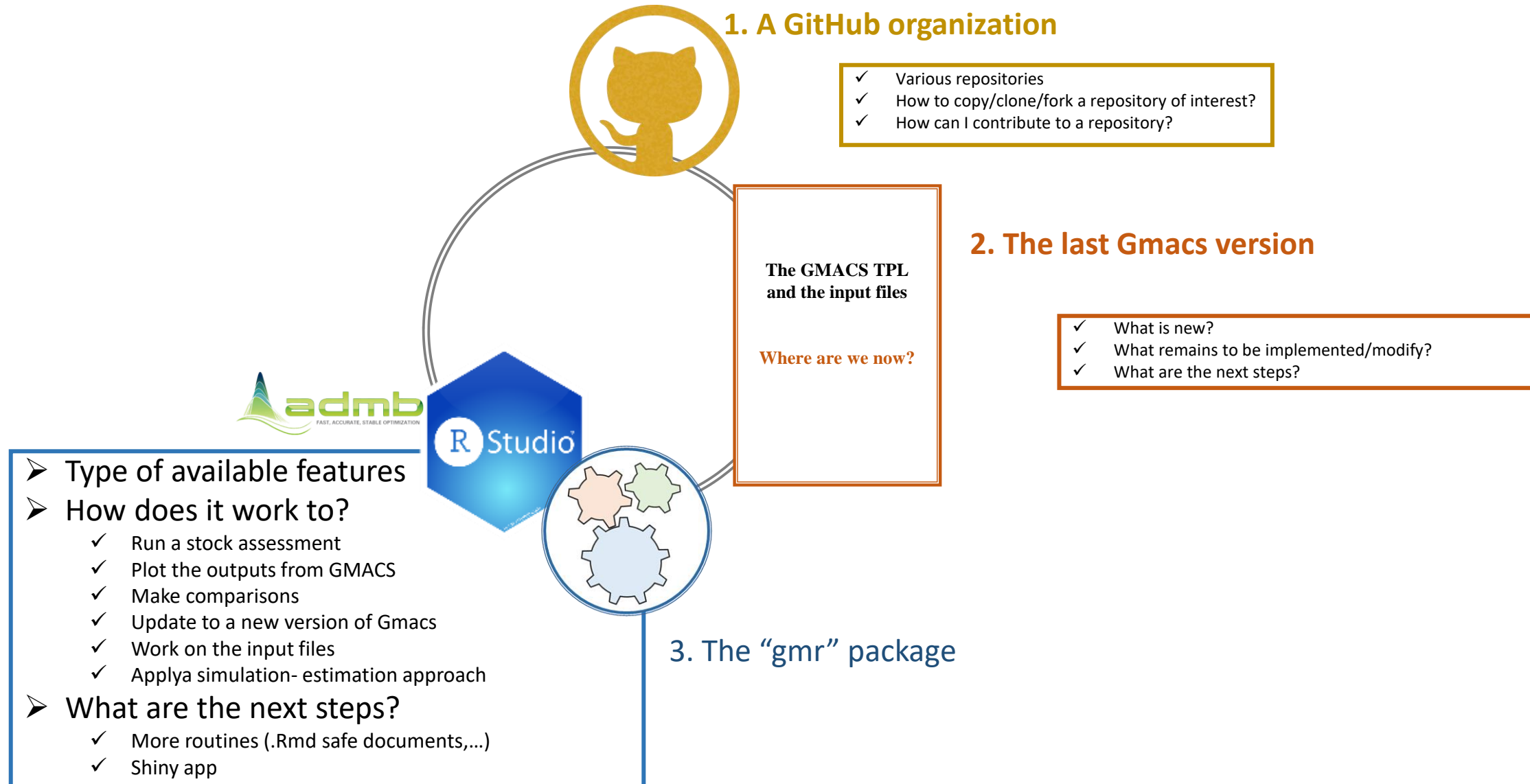
**The GMACS TPL
and the input files**

Where are we now?

2. The last Gmacs version

- What is new?
- What remains to be implemented/modify?
 - ✓ (Andre’s summer follies)
 - ✓ Some options seem useless
 - ✓ Inconsistencies ? (input files, ...)
- What are the next steps?

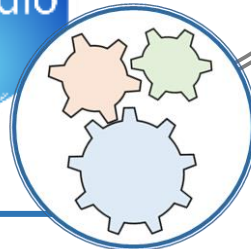
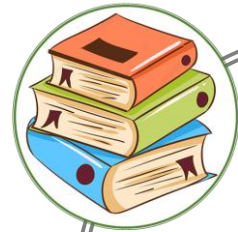
Outline of this “presentation”



Outline of this “presentation”

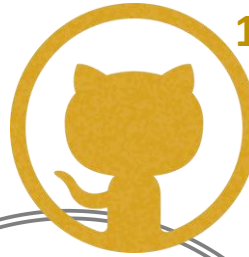
4. Various documentations and a user manual

- A paper in progress
- A user manual for preparing an assessment (with an example for a stock)
- A website for function help and vignettes



- ✓ Type of available features
- ✓ How does it work to:
- ✓ What are the next steps

1. A GitHub organization



- ✓ Various repositories
- ✓ How to copy/clone/fork a repository of interest?
- ✓ How can I contribute to a repository?

The GMACS TPL
and the input files

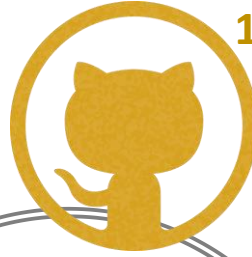
Where are we now?

2. The last Gmacs version

- ✓ What is new?
- ✓ What remains to be implemented/modify?
- ✓ What are the next steps?

3. The “gmr” package

Outline of this “presentation”



1. A GitHub organization

➤ Various repositories

- ✓ What can be found in this organization?
- ✓ Which ones are useful for users and developers?



1. A GitHub organization

➤ Various repositories

✓ Fully open source

<https://github.com/GMACS-project>

Welcome!

GMACS-project is a Github organization that hosts a number of open-source projects (under development) that aim to provide tools for Crustacean stock assessments. Specifically, this organization gives you access to repositories associated with {GMACS}, the Generalized size-structured Model for Assessing Crustacean Stocks.

This includes

- The [old repository](#) for Gmacs (holding old versions and files);
- The [new repository](#) for Gmacs where you can find:
 - i) the old development versions of GMACS in [GMACS_Orig](#) and [GMACS_Terminal_molt](#),
 - ii) the [current](#) developpement version and,
 - iii) the [latest](#) release for GMACS;
- A [User-manual](#) repository which is in development.
- An R package [gmr](#) allowing to work with GMACS.



1. [GMACS_Assessment_code](#) (public)
2. [gmr](#) (public)
3. [User-manual](#) (private for now)
4. “gmacs”: the old repository (public)



1. A GitHub organization

➤ Various repositories

✓ Fully open source

<https://github.com/GMACS-project>

Welcome!

GMACS-project is a Github organization that hosts a number of open-source projects (under development) that aim to provide tools for Crustacean stock assessments. Specifically, this organization gives you access to repositories associated with {GMACS}, the Generalized size-structured Model for Assessing Crustacean Stocks.

This includes

- The [old repository](#) for Gmacs (holding old versions and files);
- The [new repository](#) for Gmacs where you can find:
 - i) the old development versions of GMACS in [GMACS_Orig](#) and [GMACS_Terminal_molt](#),
 - ii) the [current](#) developpement version and,
 - iii) the [latest](#) release for GMACS;
- A [User-manual](#) repository which is in development.
- An R package [gmr](#) allowing to work with GMACS.

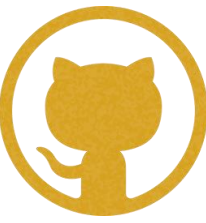


1. [GMACS_Assessment_code](#) (public)

2. [gmr](#) (public)

3. [User-manual](#) (private for now)

4. “gmacs”: the old repository



1. A GitHub organization

➤ Various repositories

1.1 The GMACS_Assessment_code repository

https://github.com/GMACS-project/GMACS_Assessment_code

File/Folder	Commit Message
Assessment_data	small updates
Assessments	Clean & update repo
GMACS_versions	Merge branch 'main' into Buck_PR
R_scripts	Clean & update repo
Rmarkdown_templates	Updates due to modification of the repo
SAFE_documents	Check WTS's version 2.01.LWTS
.gitattributes	Initial commit
.gitignore	2022-10-03. Got snow crab natural mortality test
README.md	Update README.md
_config.yml	Create _config.yml

1. "Assessment_data"

⇒ The files from the most recent assessment that has been accepted (gmacs.dat; .ctl; .dat; .prj) **** Stock assessment Authors ****



1. A GitHub organization

➤ Various repositories

1.1 The GMACS_Assessment_code repository

https://github.com/GMACS-project/GMACS_Assessment_code

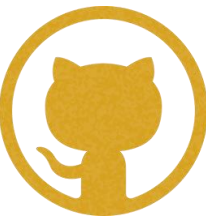
File/Folder	Commit Message
MatthVeron Update README.md	
Assessment_data	small updates
Assessments	Clean & update repo
GMACS_versions	Merge branch 'main' into Buck_PR
R_scripts	Clean & update repo
Rmarkdown_templates	Updates due to modification of the repo
SAFE_documents	Check WTS's version 2.01.LWTS
.gitattributes	Initial commit
.gitignore	2022-10-03. Got snow crab natural mortality test
README.md	Update README.md
_config.yml	Create _config.yml

1. "Assessment_data"

⇒ The files from the most recent assessment that has been accepted (gmacs.dat; .ctl; .dat; .prj) **** Stock assessment Authors ****

2. Assessments

⇒ All historical accepted assessment files (gmacs.dat; .ctl; .dat; .prj & gmacsAll.out) **** Stock assessment Authors ****



1. A GitHub organization

➤ Various repositories

1.1 The GMACS_Assessment_code repository

https://github.com/GMACS-project/GMACS_Assessment_code

File/Folder	Description
MatthVeron Update README.md	
Assessment_data	small updates
Assessments	Clean & update repo
GMACS_versions	Merge branch 'main' into Buck_PR
R_scripts	Clean & update repo
Rmarkdown_templates	Updates due to modification of the repo
SAFE_documents	Check WTS's version 2.01.LWTS
.gitattributes	Initial commit
.gitignore	2022-10-03. Got snow crab natural mortality test
README.md	Update README.md
_config.yml	Create _config.yml

1. "Assessment_data"

⇒ The files from the most recent assessment that has been accepted (gmacs.dat; .ctl; .dat; .prj) ** Stock assessment Authors **

2. Assessments

⇒ All historical accepted assessment files (gmacs.dat; .ctl; .dat; .prj & gmacsAll.out) ** Stock assessment Authors **

3. GMACS_versions : key folder

⇒ The most recent and release version of GMACS

⇒ Two historical development folders (Gmacs_orig & GMACS_terminal_molt)

⇒ Two ADpaths["OS"].txt



1. A GitHub organization

➤ Various repositories

1.1 The GMACS_Assessment_code repository

https://github.com/GMACS-project/GMACS_Assessment_code

File/Folder	Description
MatthVeron Update README.md	
Assessment_data	small updates
Assessments	Clean & update repo
GMACS_versions	Merge branch 'main' into Buck_PR
R_scripts	Clean & update repo
Rmarkdown_templates	Updates due to modification of the repo
SAFE_documents	Check WTS's version 2.01.LWTS
.gitattributes	Initial commit
.gitignore	2022-10-03. Got snow crab natural mortality test
README.md	Update README.md
_config.yml	Create _config.yml

1. "Assessment_data"

⇒ The files from the most recent assessment that has been accepted (gmacs.dat; .ctl; .dat; .prj) ** Stock assessment Authors **

2. Assessments

⇒ All historical accepted assessment files (gmacs.dat; .ctl; .dat; .prj & gmacsAll.out) ** Stock assessment Authors **

3. GMACS_versions : key folder

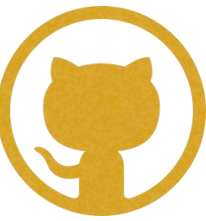
⇒ The most recent and release version of GMACS

⇒ Two historical development folders (Gmacs_orig & GMACS_terminal_molt)

⇒ Two ADpaths["OS"].txt

4. R_scripts (in dvpt)

⇒ R scripts that are intended to be "routines" to help working with GMACS














1. A GitHub organization

➤ Various repositories

1.1 The GMACS_Assessment_code repository

https://github.com/GMACS-project/GMACS_Assessment_code

 MatthVeron Update README.md	
 Assessment_data	small updates
 Assessments	Clean & update repo
 GMACS_versions	Merge branch 'main' into Buck_PR
 R_scripts	Clean & update repo
 Rmarkdown_templates	Updates due to modification of the repo
 SAFE_documents	Check WTS's version 2.01.LWTS
 .gitattributes	Initial commit
 .gitignore	2022-10-03. Got snow crab natural mortality test
 README.md	Update README.md
 _config.yml	Create _config.yml

1. "Assessment_data"

⇒ The files from the most recent assessment that has been accepted (gmacs.dat; .ctl; .dat; .prj) ** Stock assessment Authors **

2. Assessments

⇒ All historical accepted assessment files (gmacs.dat; .ctl; .dat; .prj & gmacsAll.out) ** Stock assessment Authors **

3. GMACS_versions : key folder

⇒ The most recent and release version of GMACS

⇒ Two historical development folders (Gmacs_orig & GMACS_terminal_molt)

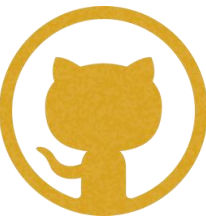
⇒ Two ADpaths["OS"].txt

4. R_scripts *(in dvpt)*

⇒ R scripts that are intended to be "routines" to help working with GMACS

5. Rmarkdown_templates *(in dvpt)*

⇒ Rmarkdown files (routines) to produce pdfs of outputs from GMACS, for comparison analyses, and when editing SAFE documents



1. A GitHub organization

➤ Various repositories

1.1 The GMACS_Assessment_code repository

https://github.com/GMACS-project/GMACS_Assessment_code

File/Folder	Last Commit
MatthVeron Update README.md	
Assessment_data	small updates
Assessments	Clean & update repo
GMACS_versions	Merge branch 'main' into Buck_PR
R_scripts	Clean & update repo
Rmarkdown_templates	Updates due to modification of the repo
SAFE_documents	Check WTS's version 2.01.LWTS
.gitattributes	Initial commit
.gitignore	2022-10-03. Got snow crab natural mortality test
README.md	Update README.md
_config.yml	Create _config.yml

1. "Assessment_data"

⇒ The files from the most recent assessment that has been accepted (gmacs.dat; .ctl; .dat; .prj) ** Stock assessment Authors **

2. Assessments

⇒ All historical accepted assessment files (gmacs.dat; .ctl; .dat; .prj & gmacsAll.out) ** Stock assessment Authors **

3. GMACS_versions : key folder

⇒ The most recent and release version of GMACS

⇒ Two historical development folders (Gmacs_orig & GMACS_terminal_molt)

⇒ Two ADpaths["OS"].txt

4. R_scripts *(in dvpt)*

⇒ R scripts that are intended to be routines to help working with GMACS

5. Rmarkdown_templates *(in dvpt)*

⇒ Rmarkdown files (routines) to produce pdf of outputs from GMACS, comparison analyses, and editing SAFE documents

6. SAFE_documents *(in dvpt)*

⇒ Rmarkdown files for each stock that can serve as a basis to produce SAFE documents with comparisons between various models



1. A GitHub organization

➤ Various repositories

✓ Fully open source

<https://github.com/GMACS-project>

Welcome!

GMACS-project is a Github organization that hosts a number of open-source projects (under development) that aim to provide tools for Crustacean stock assessments. Specifically, this organization gives you access to repositories associated with {GMACS}, the Generalized size-structured Model for Assessing Crustacean Stocks.

This includes

- The [old repository](#) for Gmacs (holding old versions and files);
- The [new repository](#) for Gmacs where you can find:
 - i) the old development versions of GMACS in [GMACS_Orig](#) and [GMACS_Terminal_molt](#),
 - ii) the [current](#) developpement version and,
 - iii) the [latest](#) release for GMACS;
- A [User-manual](#) repository which is in development.
- An R package [gmr](#) allowing to work with GMACS.

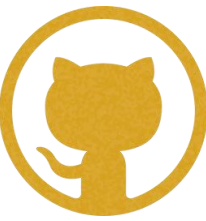


1. [GMACS_Assessment_code](#) (public)

2. [gmr](#) (public)

3. [User-manual](#) (private for now)

4. “gmacs”: the old repository



1. A GitHub organization

➤ Various repositories

1.2 The gmr repository

<https://github.com/GMACS-project/gmr>

	MatthVeron Update GetGmacsExe.R
📁	.github Update build-site.yaml
📁	R Update GetGmacsExe.R
📁	docs update website after release V1.2.0
📁	inst update website after release V1.2.0
📁	man Remove the export of Write_TPL
📁	pkgdown Updates for New release - V1.2.0
📁	vignettes small updates on Do_comp & website
📄	.Rbuildignore New Github Actions
📄	.gitignore Working on website
📄	DESCRIPTION Updates for New release - V1.2.0
📄	NAMESPACE Small updates
📄	NEWS.Rmd Updates for New release - V1.2.0
📄	NEWS.md Update NEWS.md
📄	README.Rmd Update Readme
📄	README.md Update Readme
📄	Work_on_gmr.R New Github Actions
📄	gmr.Rproj Update to pass the check-R-package

Basic structure of a package repository

1. **Three** main files: **Description**, **Namespace** and :
2. The **R folder**: contains all the functions of the package
3. The help documentation is stored in the “vignettes” folder
4. Package developed using the package [Roxygen](#)
 - ⇒ easy to do
 - ⇒ facilitates the maintenance of the package and the documentation
 - ⇒ facilitates the control of the implementation
 - ⇒ facilitates the publication on CRAN (ultimate goal for gmr)



1. A GitHub organization

➤ Various repositories

1.2 The gmr repository

<https://github.com/GMACS-project/gmr>



gmr - R package to work with a Generalized size-structured Model for Assessing Crustaceans
GMACS

- Links
- [Browse source code](#)
- [Report a bug](#)
- License
- GPL-2
- Citation
- [Citing gmr](#)
- Developers
- Matthieu Veron
Author, maintainer
- Cody Szuwalski
Author, contributor
- [More about authors...](#)

gmr is an R package providing tools to work with *GMACS*, a generalized size-structured assessment model for Crustaceans. Specifically, it includes an extensive library of functions developed to build and run GMACS for one or several crustacean stock(s) and to plot outputs of the model.

Installation

To install the `gmr` package, the `devtools` and `gdata` packages are required.

```

if(!require("devtools"))
  install.packages("devtools")
if(!require("devtools"))
  install.packages("gdata")

# Install development version from Github
.DirSrc <- "GMACS-project/gmr"
devtools::install_github(.DirSrc)

```

Now you can load the package and inspect the functions:

```

# Load the gmr package
library(gmr)
# Explore function
plot_growth

```

Using the functions requires output from the General Model for Assessing Crustacean Stocks (GMACS) which is described in the *GMACS* repo.

Basic structure of a package repository

1. All the functions are in the R folder
2. The help documentation are stored in the “vignettes” folder
3. Package developed using the package *Roxygen*
 - ⇒ Easy to do
 - ⇒ facilitates the maintenance of the package and the documentation
 - ⇒ Facilitates the control of the implementation
 - ⇒ Facilitates the publication on the CRAN (Final goal for gmr)

4. A [website](#) is build using the package [pkgdown](#)

To contribute (*see also later*)

⇒ just .. make sure to follow the “good coding practices” eg(= VS <- ; stats::rnorm vs rnorm(); ...)

⇒ where to find help?

Roxygen vignettes and *pkgdown* website

Practical R+Git Workflow for Scientists May-July
2022

Eli Holmes [workshops](#)



Hadley
Wickham's
[book](#)



1. A GitHub organization

➤ Various repositories

✓ Fully open source

<https://github.com/GMACS-project>

Welcome!

GMACS-project is a Github organization that hosts a number of open-source projects (under development) that aim to provide tools for Crustacean stock assessments. Specifically, this organization gives you access to repositories associated with {GMACS}, the Generalized size-structured Model for Assessing Crustacean Stocks.

This includes

- The [old repository](#) for Gmacs (holding old versions and files);
- The [new repository](#) for Gmacs where you can find:
 - i) the old development versions of GMACS in [GMACS_Orig](#) and [GMACS_Terminal_molt](#),
 - ii) the [current](#) developpement version and,
 - iii) the [latest](#) release for GMACS;
- A [User-manual](#) repository which is in development.
- An R package [gmr](#) allowing to work with GMACS.



1. [GMACS_Assessment_code](#) (public)

2. [gmr](#) (public)

3. [User-manual](#) (private for now)

4. “gmacs”: the old repository



1. A GitHub organization

➤ Various repositories

1.3 The user-manual repository

MatthVeron Update README.md	
References	Set-up repo
_book	Create docs directory for Github pages
_bookdown_files/02-cross-refs_files/f...	Set-up repo
docs	Set theme jekyll-theme-cayman
.gitignore	Initial commit
.nojekyll	update user manual to publish github page
01-preface.Rmd	update book
02-File-orga.Rmd	update book
03-data-file.Rmd	Set-up repo
04-control-file.Rmd	Set-up repo
05-project-file.Rmd	Set-up repo
06-R-work-gmacs.Rmd	Set-up repo
07-running-gmacs.Rmd	Set-up repo
08-first-assessment.Rmd	Set-up repo
09-stock-assessment-document.Rmd	Set-up repo
10-FAQ.Rmd	Set-up repo
11-references.Rmd	Set-up repo
Book.Rproj	Set-up repo
Book.rds	Create docs directory for Github pages
README.md	Update README.md
_bookdown.yml	update user manual to publish github page
_common.R	Set-up repo
_output.yml	update book

Book build by the [bookdown](#) R package:

- How to fill in the GMACS input files
- How to run an assessment
- How to work with the results from an estimation run
- How to apply a simulation-estimation approach

Gmacs: mastering a modeling framework to assess Crustacean species

Table of contents

Welcome

- 1 Preface
 - 2 File organisation
 - 3 Data File
 - 4 Controle File
 - 5 Projection File
 - 6 Working with Gmacs files within R
 - 7 Running Gmacs and plotting results
 - 8 Building your first Assessment
 - 9 Producing Stock Assessment Document
 - 10 Frequently Asked Questions
- References

[View book source](#)

Welcome

This is the online version of “Gmacs: mastering a modeling framework to assess Crustacean species,” a user guide book **currently under development** and that aims to merge and synthesize all the current and past work done by the Gmacs development team since the inception of this project.

Statistical catch age models have several advantages over simple production type models in that age and size composition data can be used to better inform structural features such as recruitment variability, growth variability and total mortality rates. There are a number of generic age-structured models in use today (e.g., Stock Synthesis, CASAL, ...), but there are very few generic size-based, or staged-based models that are used in stock assessment.

The Generalized Model for Assessing Crustacean Stocks (GMACS) is an integrated analysis modelling framework designed to develop size-structured stock assessment models for crustacean species.

Crab stocks of Alaska are managed by the North Pacific Fishery Management Council (NPFMC). Some stocks are assessed with integrated size-structured assessment models of the form described in Punt et al. (2013). Currently, most of these stocks are assessed using a stock-specific assessment model (e.g. Zheng and Siddeek (2014)). In this context and with the aim of uniform the assessment process of these species, the Gmacs project aims to provide a software that will allow each stock to be assessed independently but using a single flexible modelling framework.

This book is designed to take you from knowing nothing about Gmacs and its utilities

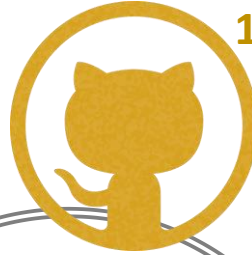
On this page

[Welcome](#)

[Structure of the book](#)

[License](#)

Outline of this “presentation”



1. A GitHub organization

- Various repositories
 - ✓ What can be found in this organization?
 - ✓ Which ones are useful for users and developers?
- How to copy/clone/fork a repository of interest ?

1. A GitHub organization

➤ How to copy/clone/fork a repository of interest?

➤ Copying a repository:

- ✓ **Why and when?** Because you want to use it as a template and make your own version of this repository – You won't affect the "origin" repository and won't get updates from that origin. This is a good way to start a new project based on a previous one.

➤ Cloning a repository:

- ✓ **Why and when?** If you are the owner of a repository (or have access to pull, write, commit and push), you will want to make a local copy of your repository. Cloning allows you to be directly connected to the original repository and make your changes without any restriction. Cloning is great to work on your own project or when you are member of an organization (but that is not the best way to do this in this case).

➤ **Forking a repository:**

- ✓ **Why and when?** To contribute to an existing project WITHOUT affecting the original repository. The fork you'll create will be independent from the origin repository. You will get all the updates from the origin (pulling) and be able to propose some modifications that will need to be accepted by the owner of the repository

1. A GitHub organization

➤ How to copy/clone/fork a repository of interest?

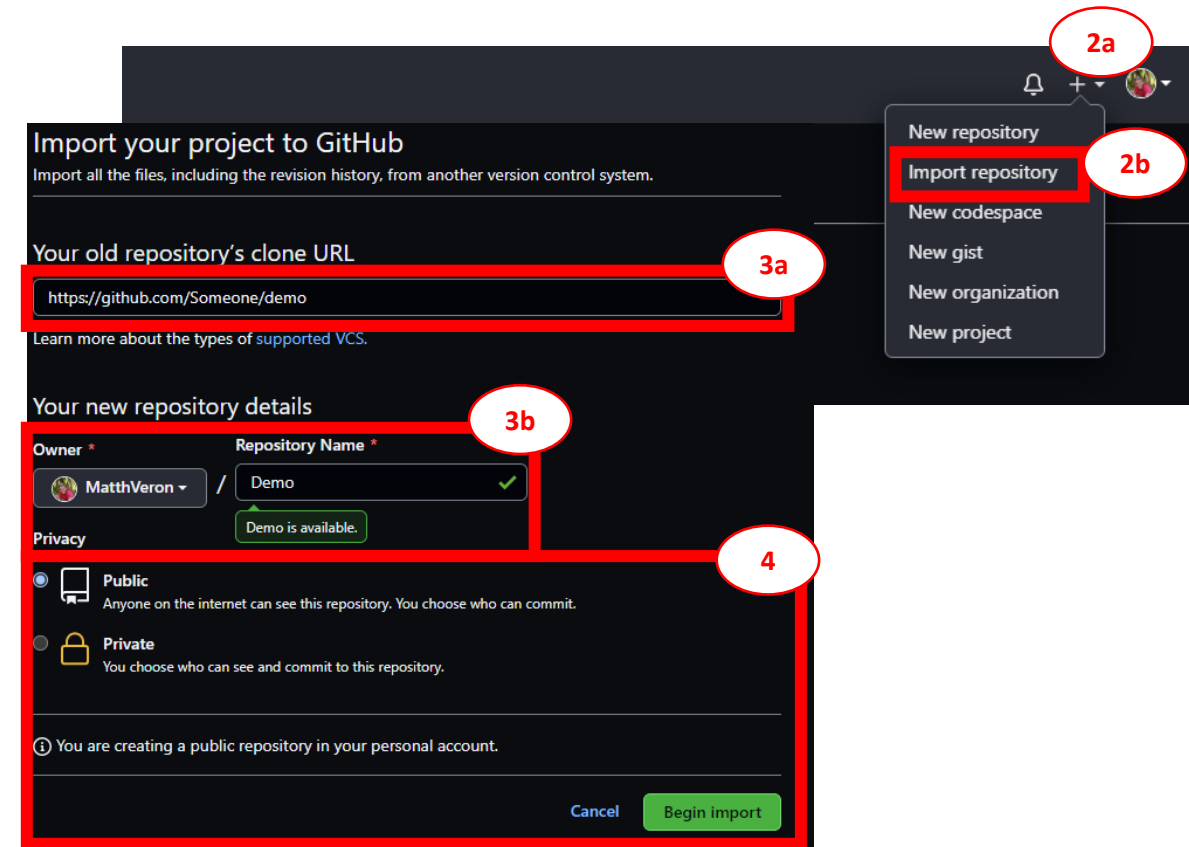
Why and when?

You want to use it as a template and make your own version of it without modifying the original one

1. Get the **URL** of the repository you want to copy (in a browser)
2. On your GitHub page: In the upper right, click the **+** (2a) and click “Import repository” (2b)
3. Paste the URL (3a) and give a name to your new repository (3b)
4. Select if you want to make it “Public” or “Private” and click “begin Import” (4)

You now have a new repository “Demo” on GitHub. You need now to create a local copy of that repo to work on it

⇒ Cloning



1. A GitHub organization

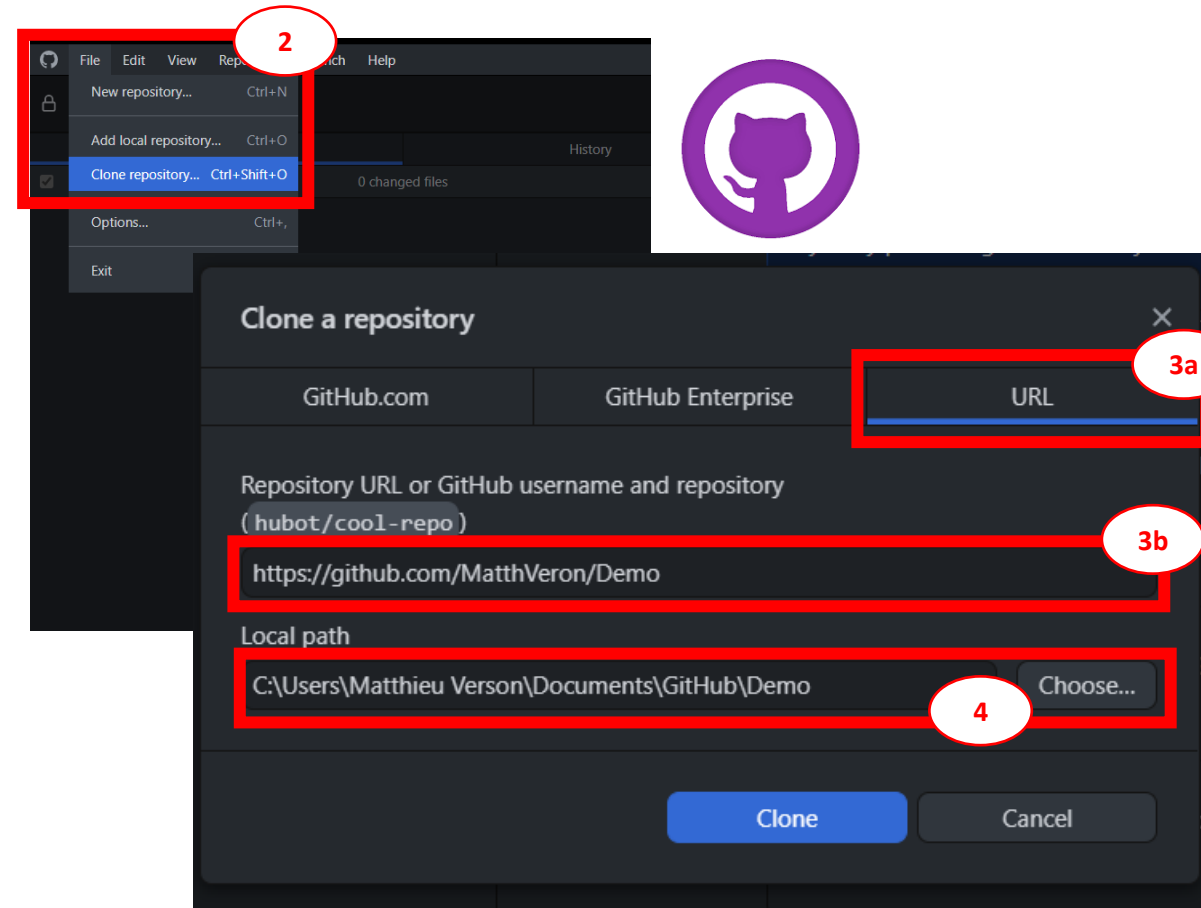
➤ How to copy/clone/fork a repository of interest?

Why and when?

You want to make a local copy of your repository while being connected to the original repository on Github so you can push your code modifications without any restriction.

1. Copy the **URL** of YOUR repository
(<https://www.github.com/yourname/Demo>)
2. On **GitHub Desktop**, click **File>Clone Repository** (2)
3. Navigate to the URL box (3a) and paste it in the “URL bar” (3b)
4. Select/check the path where you are saving this repo on your local machine

You now have a now a local copy of “**Demo**” and can start to make your changes, commit them and push them to the origin (repo on GitHub)



1. A GitHub organization

➤ How to copy/clone/**fork** a repository of interest?

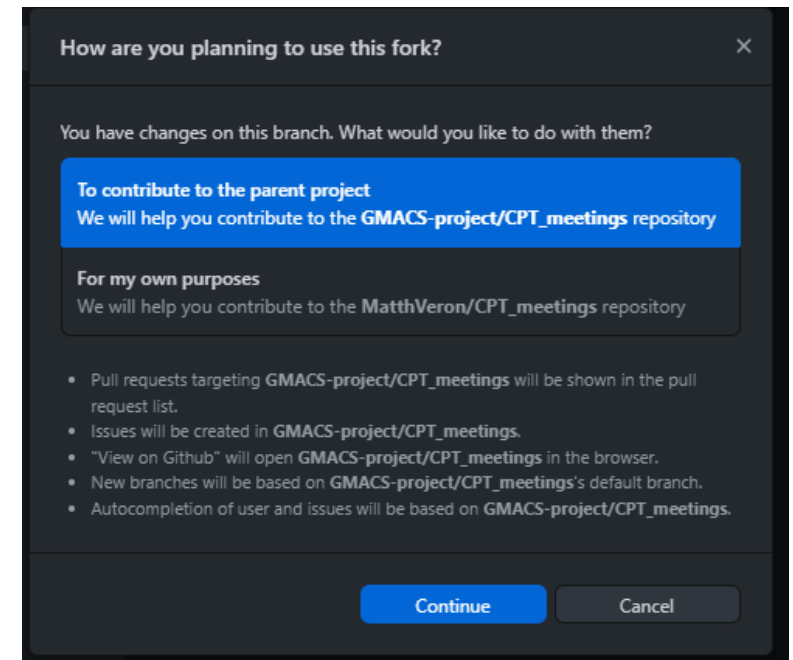
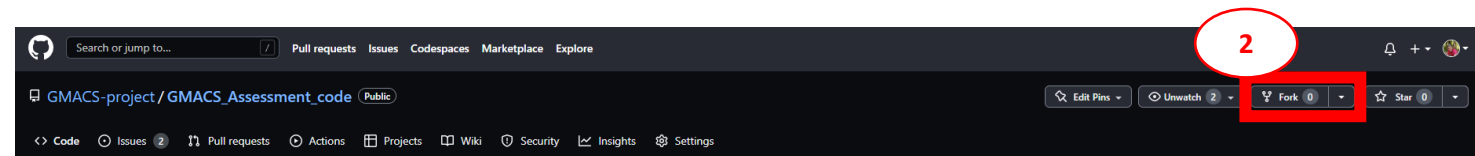
Why and when?

To contribute to an existing project WITHOUT affecting the original repository. You will work independently of the original repository. You can easily pull the changes made on this repository and suggest your contributions.

This is what we want for all the repository on the GMACS organization

1. Go to the GitHub repository you want to fork (here, [GMACS Assessment Code](#) repo of the organization)

2. In the upper right, click on “Fork”
When forking a repo you may see appear this window
=> You want to contribute to the parent project



1. A GitHub organization

➤ How to copy/clone/fork a repository of interest?

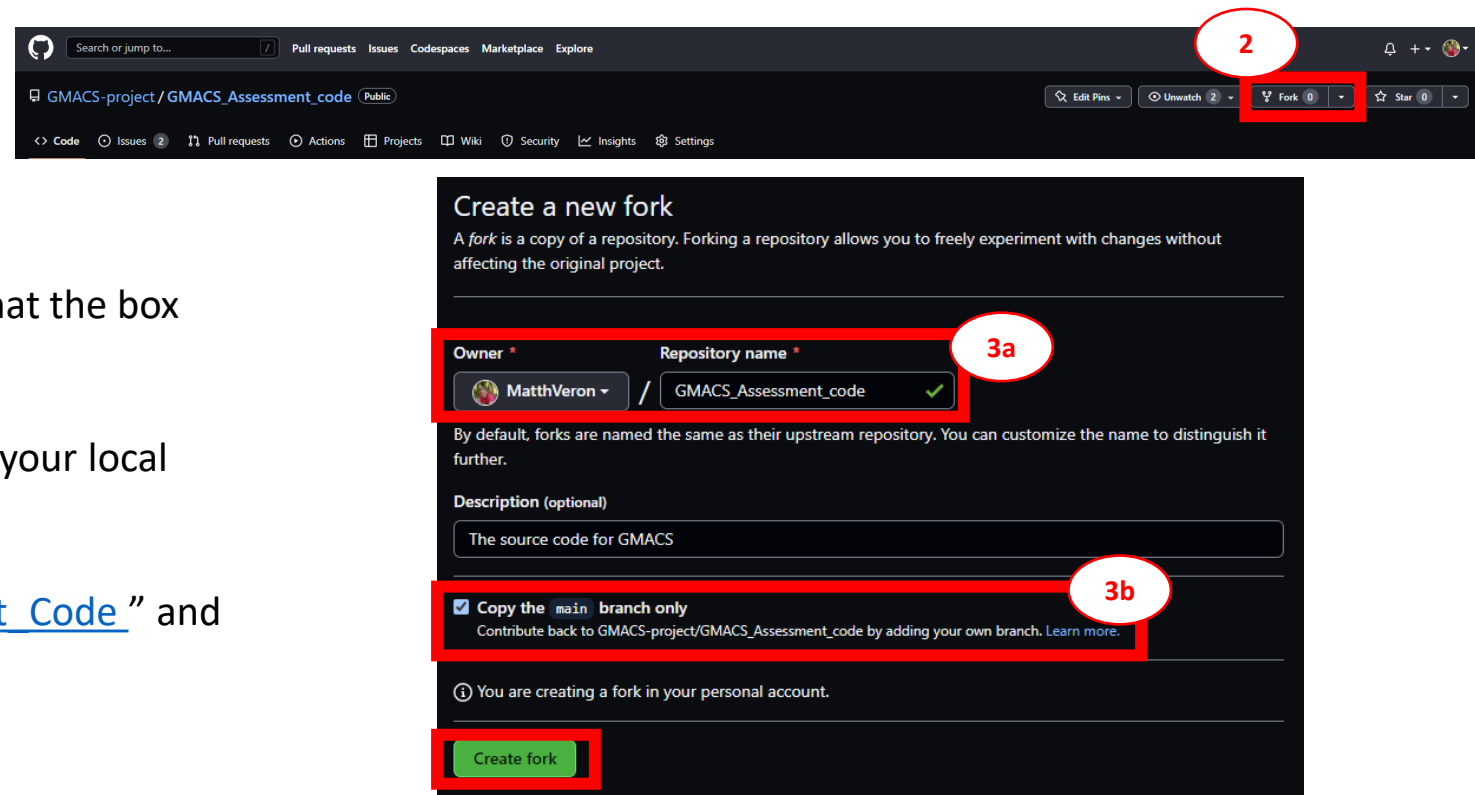
Why and when?

To contribute to an existing project WITHOUT affecting the original repository. You will work independently of the original repository. You can easily pull the changes made on this repository and suggest your contributions.

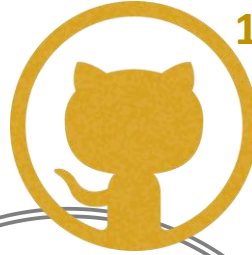
This is what we want for all the repository on the GMACS organization

1. Go to the GitHub repository you want to fork (here, [GMACS Assessment Code](#) repo of the organization)
2. In the upper right, click on “Fork”
3. Name (if you want/have to) the repo (3a) and check that the box “Copy the main branch only” is ticked (3b)
4. Fork the repo and use the previous step to clone it on your local machine.

You now have a now a local copy of “[GMACS Assessment Code](#)” and can start to work on it.



Outline of this “presentation”



1. A GitHub organization

- Various repositories
 - ✓ What can be found in this organization?
 - ✓ Which ones are useful for users and developers?
- How to copy/clone/fork a repository of interest ?
- How can I contribute to a repository?

1. A GitHub organization

➤ How can I contribute to a repository?

✓ Forking the repository & cloning it to have a local copy

✓ Do your first contributions/changes on your fork

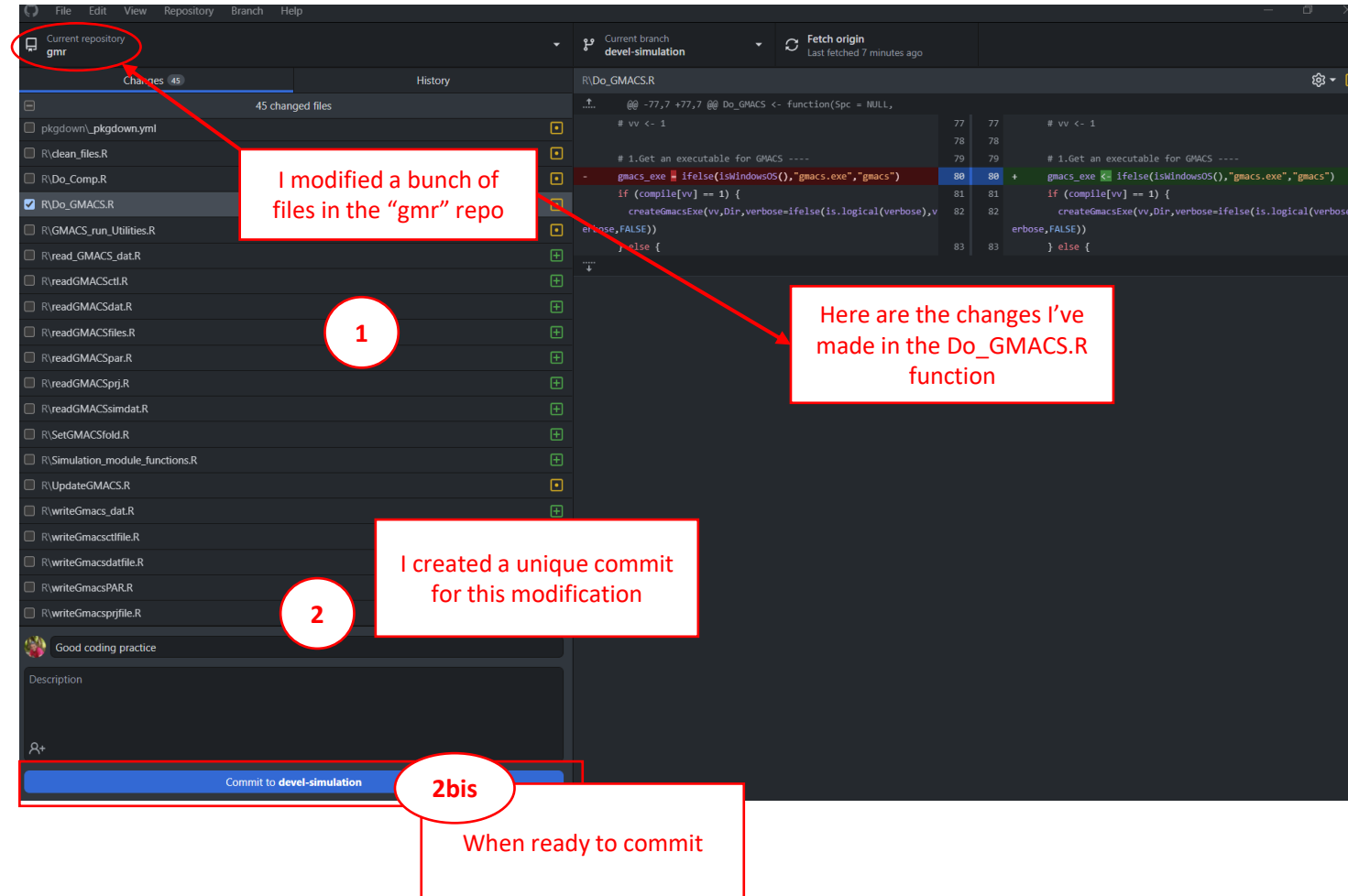
- Suggest enhancements (e.g., modifying pre-existent code)
- Create a new function (in a code, a new R file)
- Propose changes to some part of the documentation (e.g., README, a function, ...)
- Propose changes in the structure of the files
- ... Many possibilities

1

✓ Commit your changes and push them to your fork on GitHub:

- On GitHub Desktop, in your repository folder, you'll see the changes you've made for each file. You can either make a single commit for all these files or select (one or more of them) to create a common commit.

2



1. A GitHub organization

➤ How can I contribute to a repository?

✓ Forking the repository & cloning it to have a local copy

✓ Do your first contributions/changes on your fork

- Suggest enhancements (e.g., modify pre-existent code)
- Create a new function (in a code, a new R file)
- 1 ○ Propose changes to some part of the documentation (e.g., README, a function, ...)
- Propose changes in the structure of the files
- ... Many possibilities

✓ Commit your changes and **push** them to your fork on GitHub:

- 3 ○ On GitHub Desktop, in your repository folder, you'll see the changes you've made for each file. You can either make a single commit for all these files or select (one or more of them) to create a common commit.

Current repository: gmr

Current branch: devel-simulation

Push origin: Last fetched 11 minutes ago

44 changed files

1 I modified a bunch of files in the "gmr" repo

2 I create a unique commit for this modification

2bis I committed

3 I can push this change to the "gmr" repo

Here are the changes I've made in the Do_GMACS.R function

```
1 + % Generated by roxygen2: do not edit by hand
2 + % Please edit documentation in R/clean_files.R
3 + \name{clean_bat_Sim}
4 + \alias{clean_bat_Sim}
5 + \title{Clean the output files of the Gmacs simulation module}
6 + \usage{
7 +   clean_bat_Sim(path = ".", verbose = TRUE)
8 + }
9 + \arguments{
10 +   \item{path}{(character string)- names or file path representing
11 +
12 +   \item{verbose}{(logical) flag to print file names being deleted (TRUE/FALSE)}
13 + }
14 + }
15 + nothing
16 + }
17 + \description{
18 +   Function to clean gmacs output files in a folder when a
19 +   simulation-estimation approach has been considered.
20 + }
21 + \details{
22 +
23 + }
```

1. A GitHub organization

➤ How can I contribute to a repository?

Since GMACS 2.0 is a product from both UW and NOAA and, because it is probably intended to be hosted in the [NMFS Fish Tools Organization](#) (already the case for the old-old version), we strive to follow the [contribution guide](#) provided by this organization.

To facilitate contribution, please consider these guidelines (and don't hesitate to make suggestions 😊).

1. A GitHub organization

➤ How can I contribute to a repository?

- ✓ Forking the repository & cloning it to have a local copy
 - ✓ Do your first contributions/changes on your fork
 - ✓ Commit your changes and push them to your fork on GitHub:
-
- ✓ When you're ready to submit your contributions to the owner/collaborators of the upstream repository, you'll need to create a **Pull Request (PR)**:
 - This lets you tell others what changes you've made
 - It allows a discussion session about those changes between you and your peers
 - It provides another opportunity to collaborate and review potential changes before merging your contribution into the main branch of the upstream repository

1. A GitHub organization

➤ How can I contribute to a repository?

✓ Example:

- I created a repository “CPT_meetings” that can be intended to be used during each CPT meeting to work collaboratively on aspects that we would like to keep private. You should all have access to this repository.

- I forked this repository from my Github account (1) and created a new “Plot_PR.R” file (2) to show how to make a PR. Now I can make a PR by clicking on “Contribute” (3-4). Open the PR, specify what you’ve done and create it.

Well Done !!!!

The image illustrates the process of creating a pull request on GitHub. It consists of three main parts:

- Repository Page:** A screenshot of the GitHub repository page for 'MathVeron/CPT_meetings'. A red box labeled '1' highlights the repository name. Another red box labeled '3' highlights the 'Contribute' button. A third red box labeled '3' highlights the 'Plot_PR.R' file in the file list.
- Contribute Modal:** A modal window that appears after clicking 'Contribute'. It contains the text: 'This branch is 1 commit ahead of GMACS-project:main. Open a pull request to contribute your changes upstream.' and a large green 'Open pull request' button. A red box labeled '4' highlights this modal.
- Open a pull request form:** A screenshot of the 'Open a pull request' form. A red box labeled '5' highlights the top right corner of this form. The form shows the base repository (GMACS-project/CPT_meetings) and the head repository (MathVeron/CPT_meetings). It indicates 'Able to merge' and has a 'Create pull request' button at the bottom right.

1. A GitHub organization

➤ How can I contribute to a repository?

✓ Make your first Pull Request

The folder “Jan_2023” has the “same structure” as the [GMACS versions](#) of the [GMACS Assessment Code](#) repo.

The idea is:

- To fork the “CPT_meetings” repository and get a local copy on your machine
- Use the “gmacsbase.tpl” file located in the “Dvpt_version” folder to create a new “gmacsbase_*yourname*.tpl” that you’ll save in the “demo_PR_tpl” folder
- Modify this “gmacsbase_*yourname*.tpl” by adding a comment/changing a function/...
- Push your changes to your fork
- Submit a PR to merge your changes with the main branch of the “CPT_meetings” repository

1. A GitHub organization

- How can I contribute to a repository?

Working on case study:
first Pull request

....

2. The last version of GMACS

The GMACS TPL
and the input files

Where are we now?

➤ What is new?

- ✓ Incorporation of Buck modifications (new check files, a bit of formatting)
- ✓ Integration of the simulation-estimation module

- ✓ For developers and others
 - Developer names section

```
// ***** //  
// Developers  
// ***** //  
// AEP: Andre E. Punt  
// CS: Cody Szuwalski  
// JI: Jim Ianelli  
// WTS: William T. Stockhausen  
// AW: Athol Whitten  
// DW: Darcey Webber  
// MV: Matthieu Veron  
// ***** //
```

Who modify the code and what has been done?

```
// 2022-10-31 ** WTS ** (Version to 2.01.WTS) - 1. Added cmdline input flag "testingflag" to turn on sandbox - 2. Added alternative data file reader (TCSAM02 format)  
// 2022-11-16 ** WTS ** (Version to 2.01.WTS) - 1. Added lots of diagnostic output when reading input files - 2. Added ECHOSTR, WriteCtlStr, WriteProjStr macros for 1 - 3. Reformatted calc_relative_abundance in preparation for adding ability to handle immature data  
// 2022-12-22 ** MV ** (Version to 2.01.L02) - 1. Fix a small bug in calc_natural_mortality() - 2. Rename WTS's version - 3. Modify the DatFileReader.cpp to make it compatible with different OS - Fix a small bug in the control file section (MrelFem)  
// 2022-12-31 ** MV ** (Version to 2.01.L03) - 1. Add the simulation approach developed by AE- AE modified the code so initial values for selex is not re-set in the PARAMETER_SECTION
```

2. The last version of GMACS

➤ What is new?

- ✓ Incorporation of Buck modifications (new check files, a bit of formatting)
- ✓ Integration of the simulation-estimation module
- ✓ For developers and others
 - Developer names section
 - Submodels labelling section

The GMACS TPL
and the input files

Where are we now?

```
// Submodels labeling
// ***** //
// Label 130: Projection inputs
// Label 200: INITIALIZATION_SECTION
// Label 400: calc_objective_function
// Label 401: catch_likelihood
// Label 402: index_likelihood
// Label 403: length_likelihood
// Label 404: recruitment_likelihood
// Label 405: growth_likelihood
// Label 500: calc_spr_reference_points2
// Label 501: calc_brute_equilibrium
// Label 502: project_biomass_OFL
// Label 503: calc_predicted_project
// Label 504: project_biomass
// Label 505: compute_OFL_and_ABC
// Label 506: project_one_year
// Label 600: CreateOutput
// Label 700: write_eval
// ***** //
```

2. The last version of GMACS

➤ What is new?

- ✓ Incorporation of Buck modifications (new check files, a bit of formatting)
- ✓ Integration of the simulation-estimation module

- ✓ For developers and others
 - Developer names section
 - Submodels labelling section
 - Name of the version and date of the last compilation (update)
 - ❑ Change when releasing a new version of GMACS (part of the process)

The GMACS TPL
and the input files

Where are we now?

```
!! TheHeader = adstring("## GMACS Version 2.01.L03; Previous compilation on: 2023-01-01 20:28:19; Last compilation on: 2023-01-01 20:29:06");
```

2. The last version of GMACS

➤ What remains to be implemented/modify?

- ✓ Incorporation of Buck modifications (new check files, a bit of formatting)
- ✓ Integration of the simulation-estimation module

- ✓ Environmental linkages
- ✓ Random walk in selectivity deviations
- ✓ Random walk deviations in selectivity
- ✓ Penalty on the random deviations
 - Option when M changes as a random walk?
- ✓ Correcting the indefinite `init_bounded_number_vector`

- ✓ Change data / control input files

**The GMACS TPL
and the input files**

Where are we now?

3. The “gmr” package



➤ Type of available features

- ✓ Functions to work with GMACS files (`list()`)
 - Functions to read the input/output files
 - Functions to write the GMACS input files
 - Functions to clean the compilation-process files and the output files from GMACS

- ✓ Functions to run an assessment
 - Functions to create gmacs.exe
 - Functions to run GMACS
 - Functions to make comparisons between models

- ✓ Functions to update a new version of GMACS

- ✓ Functions to plot the output from GMACS

- ✓ Functions to conduct a simulation-estimation analysis

=> See details on the “gmr” website or directly into R (e.g., `?readGMACSctl` to get the documentation of the function)

3. The "gmr" package



➤ Installing gmr from GitHub

```
# Install and load the packages----
# 1.Install devtools and gdata on your machine ----
if (!require("devtools")) {           # install devtools
  install.packages("devtools")
}
if (!require("gdata")) {             # needed to manipulate data
  install.packages("gdata")
}

# 2. Install / update gmr package ----
# The package is can be download from the GitHub organization (Src)
Src <- "GMACS-project/gmr"
# Get the latest version of gmr? (0: no; 1: install for the first time;
Update <- 0
# 2: update the package)
mylib <- "~/R/win-library/4.1"      # the library directory to remove the
# gmr package from
# remotes::install_github() will work to install gmr on your machine
if(Update == 1) devtools::install_github(Src)
# Updating to the latest version of gmr
if(Update == 2){
  remove.packages("gmr", lib=mylib)
  devtools::install_github(Src)
}
```

3. The “gmr” package



➤ How does it work?

✓ Let's go through the process of:

The folder “Jan_2023/GMACS_versions” holds several scripts allowing to conduct the following tasks:

- Update “gmr” and create the GMACS executable
 - [S1 Gmacs exe.R](#)
- Run an assessment;
 - [S2 Run Gmacs.R](#)
- Compare a new version of GMACS with the one currently used for the stock assessment and submitting an update of GMACS on GitHub:
 - [S3 Comp Assess Vers.R](#)
- Plot the output of an assessment (only pdf format right now);
 - [S4 plot GMACS.R](#)
- Conduct a simulation-estimation analysis
 - [S5 conduct simulations.R](#)

3. The "gmr" package



➤ What are the next steps?

- ✓ Clean the pre-existing (plotting) functions
- ✓ Create functions to check for changes when modifying directly input files in R
- ✓ Create vignettes
- ✓ Continue to implement new routines
 - Running GMACS
 - Updating GMACS
 - Make comparison between GMACS version / models
 - Write SAFE document (.Rmd)
- ✓ Develop a Shiny App for GMACS ??

4. Various documentations and a user manual



➤ A paper in progress

- ✓ Describes all the equations and options in GMACS
- ✓ Uses BBRKC as a case study for a simulation estimation approach
- ✓ Describes the outputs from an assessment

- ✓ Allows to introduce the 'gmr' package and the user manual

4. Various documentations and a user manual

- A paper in progress
- A user manual

- ✓ Describes :
 - Available options
 - How to set up an assessment (input files)
 - How to generate plots and make comparison
 - How to realize a simulation-estimation study
- ✓ Use BBRKC as an exemple
- ✓ FAQ

⇒ Currently in development



Gmacs: mastering a modeling framework to assess Crustacean species

Table of contents

Welcome

1 Preface

2 File organisation

3 Data File

4 Controle File

5 Projection File

6 Working with Gmacs files within R

7 Running Gmacs and plotting results

8 Building your first Assessment

9 Producing Stock Assessment Document

10 Frequently Asked Questions

References

[View book source](#) 



4. Various documentations and a user manual

- A paper in progress
- A user manual
- A website for function help and vignettes

✓ [GMR](#)



gmr - R package to work with a Generalized size-structured Model for Assessing Crustaceans

GMACS

`gmr` is an R package providing tools to work with *GMACS*, a generalized size-structured assessment model for Crustaceans. Specifically, it includes an extensive library of functions developed to build and run GMACS for one or several crustacean stock(s) and to plot outputs of the model.

Installation

To install the `gmr` package, the `devtools` and `gdata` packages are required.

```
if(!require("devtools"))
  install.packages("devtools")
if(!require("devtools"))
  install.packages("gdata")

# Install development version from Github
.DirSrc <- "GMACS-project/gmr"
devtools::install_github(.DirSrc)
```

Now you can load the package and inspect the functions:

```
# Load the gmr package
library(gmr)
# Explore function
plot_growth
```

Using the functions requires output from the General Model for Assessing Crustacean Stocks (GMACS) which is described in the [GMACS](#) repo.

Links

[Browse source code](#)

[Report a bug](#)

License

GPL-2

Citation

[Citing gmr](#)

Developers

Matthieu Veron
Author, maintainer

Cody Szuwalski
Author, contributor

[More about authors...](#)

⇒ Still in development

Contents

Preparing runs and reading input files

Conducting a stock assessment using GMACS

Updating a version of GMACS for a new release

Realizing a simulation-estimation analysis

Manipulating outputs from the GMACS function

Making results tables

Creating plots

Terminals

Plotting options and ggplot utilities