

Retour d'expérience sur l'écriture d'un bookdown à 2 via github

Sébastien Déjean (IMT, UPS) et Thibault Laurent (TSE, CNRS)

09/12/2021

Problématique

Notre objectif

Co-écrire un manuel sur une utilisation avancée de R

Nos besoins

- ▶ trouver l'outil adéquat pour écrire un manuel associant du code R et ses sorties avec du texte
- ▶ être capable de travailler le document chacun de notre côté et voir les contributions de chacun

Nos solutions

Bookdown et Github, utilisable depuis RStudio

Etape 1 : préparation du Bookdown en local

Pré-requis : une distribution *LaTeX* et l'installation du package R **bookdown**

- ▶ En local : depuis RStudio, création de l'ensemble des fichiers qui permettent de générer le bookdown. Les fichiers sont contenus dans un répertoire **R_book**
- ▶ pour plus d'informations sur bookdown : <https://thinkr.fr/rediger-avec-bookdown-pourquoi-comment/>

Etape 2 : Envoi du répertoire **R_book** sur Github

Pré-requis : avoir un compte Github, avoir la commande git installée, ouvrir un terminal et se placer dans le répertoire **R_book**

- ▶ aller sur <https://github.com/>, se connecter avec son compte et créer un nouveau répertoire **R_book**. Pour donner les droits de modification à une tierce personne, il faut aller dans "Manage access", puis "add people"
- ▶ Initialiser un dossier dépôt git dans le répertoire **R_book**

```
git init
```

- ▶ Ajouter les fichiers dans le dépôt git

```
git add .
```

- ▶ Identification de l'utilisateur (facultatif) :

```
git config --global user.email thibault.laurent@tse-fr.eu  
git config --global user.name "tibo31"
```

Etape 2 : Envoi du répertoire **R_book** sur Github

- Identification via un jeton créé préalablement sur Github (voir notice):

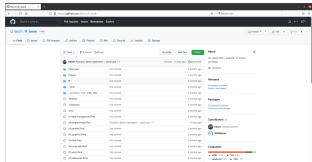
```
git remote set-url origin https://replacer_par_token@github.com:replacer/replacer
```

- Valider les modifications apportées au HEAD. Notez que tout commit ne se fera pas dans le dépôt distant.

```
git commit -m "Description du commit"
```

- ▶ Transférer les fichiers depuis la machine locale vers le serveur github

```
git push -u origin main
```

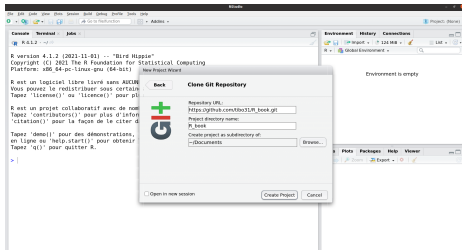


Nos fichiers sont sur Github !

Etape 3 : récupérer **R_book** depuis Github (avec RStudio)

Pré-requis : avoir un compte Github, avoir git installée (git-scm.com), avoir le package R **usethis** installé

► Depuis RStudio, New project, puis Version control:



Remarque : pour cloner le répertoire, pas besoin de compte Github. Pour pouvoir faire des changements en local et les envoyer sur Github, il faut s'identifier et avoir les droits de modification

Etape 4 : coupler le répertoire **R_book** sur la machine locale avec Github (avec RStudio)

- ▶ il nous faut créer un jeton, la commande suivante nous renvoie sur Github (il faut se connecter avec le compte qui a les droits de modification). A la fin de cette étape, on obtient un code qu'il faut copier dans le presse papier

```
usethis::create_github_token()
```

- ▶ pour valider le code sur notre machine locale, on utilise la commande suivante:

```
gitcreds::gitcreds_set()
```

- ▶ on peut vérifier que les informations de notre compte Github ont bien été prises en compte:

```
usethis::gh_token_help()
```

A présent, toutes les modifications faites en local pourront être transférées sur le serveur Github depuis l'onglet Git

Etape 5 : mettre à jour les modifications faites en local sur Github (avec RStudio)

- ▶ Dans l'onglet Git, on sélectionne les fichiers qu'on a modifié localement et qu'on souhaite transférer sur Github. On appuie sur le bouton `Commit` et il est recommandé d'écrire un message clair pour ses collaborateurs. A la fin de cette étape, on n'a toujours pas envoyé les changements sur Github.
- ▶ Pour finaliser le transfert sur Github, il faut cliquer sur `Push`.
- ▶ Les nouveaux fichiers sont sur Github

Etape 6 : reprendre son projet et mettre à jour les modifications de Github en local (avec RStudio)

- ▶ Quand on reprend son projet, il est recommandé de faire un pull chaque fois : cela synchronise les fichiers de Github potentiellement modifiés, avec les mêmes fichiers en local.
- ▶ pour accéder à l'historique des modifications : aller dans Commit, puis History

Non traité ici mais bien expliquer ici : <https://thinkr.fr/travailler-avec-git-via-rstudio-et-versionner-son-code/>

- ▶ Gestion des conflits
- ▶ Créer une nouvelle branche
- ▶ merger une branche
- ▶ voyage, voyage