

STID 1 - Programmation Statistique

TD1

GitHub et introduction à R

Anthony SARDELLITTI

2023-01-01

Contents

1	Utiliser GitHub	1
1.1	Qu'est ce que Git	1
1.2	Création d'un repository	1
1.3	Utiliser GitHub	2
1.3.1	Option 1 : Avec GitHub Desktop (NOOB)	2
1.3.2	Option 2 : Avec GitBash (GOAT)	2
1.3.3	Cheat Sheet - GitBash	2
1.4	Effectuer une pull request	3
2	Les bases de R	3
2.1	Installer R et RStudio	3
2.2	Les objets et les vecteurs	3

1 Utiliser GitHub

1.1 Qu'est ce que Git

Un peu de lecture [ici](#).

1.2 Création d'un repository

1. Créer un compte personnel sur GitHub Vous utiliserez ce compte pour tous les cours de programmation statistique.
2. Créer un repository nommé *programmation_r* avec un fichier **README.md**

- [lien utile : bonnes pratiques](#). Vous pouvez créer un repository public ou privé.
- [lien utile : créer un repository](#)

3. Créer une branche appelée *td1*

- [lien utile : créer une branche](#)

1.3 Utiliser GitHub

1.3.1 Option 1 : Avec GitHub Desktop (NOOB)

1. [Installer GitHub Desktop](#)
2. Un peu de lecture pour la suite [ici](#)
3. Cloner votre repository *programmation_r* sur votre machine
4. Connecter à la branche *td1* sur GitHub Desktop
5. Ajouter une modification à votre fichier **README.md** directement dans le fichier dans votre dossier
6. Faire un commit de cette modification sur votre branche *td1* sur GitHub Desktop
7. Pousser le commit vers votre repository distant sur GitHub Desktop
8. Vérifier si votre repository distant a pris en compte cette modification sur votre branche *td1*

1.3.2 Option 2 : Avec GitBash (GOAT)

1.3.2.1 Configurer GitBash

1. Installer Git
 2. Installer GitBash
 3. Configurer la connexion SSH à votre repository distant
- [lien utile : connexion SSH](#)
4. Cloner votre repository *programmation_r* sur votre machine

1.3.2.2 Utiliser GitHub avec GitBash

0. Un peu de lecture [ici](#).
1. Connecter à la branche *td1*
2. Ajouter une modification à votre fichier **README.md**
3. Faire un commit de cette modification sur votre branche *td1*
4. Pousser le commit vers votre repository distant
5. Vérifier si votre repository distant a pris en compte cette modification sur votre branche *td1*

1.3.3 Cheat Sheet - GitBash

1.3.3.1 Configurer la clé SSH

1. Ouvrir l'application GitBash
2. Changer le chemin vers le dossier dans lequel vous souhaitez configurer votre clé ssh : `cd "C:/Users/Anthony/Documents/SSH"`
3. Générer la clé ssh : `ssh-keygen`
4. Saisir `id_rsa` comme file name pour sauvegarder la clé ssh
5. Entrer une passphrase qui permettra de se connecter à votre compte GitHub à chaque connexion
6. Ajouter la clé en tant que nouvelle clé dans les paramètres de votre compte GitHub web (voir lien utile cité précédemment)

7. Se connecter en local à votre compte GitHub : `eval "$(ssh-agent -s)"` puis `ssh-add ./id_rsa`
8. Changer le chemin vers le dossier dans lequel vous souhaitez cloner votre repository : `cd "C:/Users/Anthony/Documents/R"`
9. Cloner votre repository distant par exemple : `git@github.com:asardell/std1_2023.git`

1.3.3.2 Se connecter à Git au quotidien

1. Ouvrir l'application GitBash
2. Changer le chemin vers le dossier dans lequel vous avez stocké la clé ssh : `cd "C:/Users/Anthony/Documents/SSH"`
3. Se connecter en local à votre compte GitHub : `eval "$(ssh-agent -s)"` puis `ssh-add ./id_rsa`
4. Changer le chemin vers le dossier où est cloné votre repository : `cd "C:\Users\Anthony\Documents\R\std1_2023"`

1.3.3.3 Utiliser les commandes de bases Voici quelques commandes de bases :

- créer une branche : `git branch branch_name`
- changer de branche : `git switch branch_name`
- mettre à jour la branche en local avec la branche distante : `git pull origin branch_name`
- créer un commit : `git add .` puis `git commit -m "votre commentaire pour décrire la modification"`
- pousser un commit : `git push origin branch_name`
- mettre à jour votre repository local avec le repository distant : `git fetch`

1.4 Effectuer une pull request

Depuis votre compte GitHub web.

1. Effectuer une pull request de votre branche *td1* sur votre branch par défaut (*main* ou *master*).
- [lien utile : tutoriel pull request](#)
2. Effectuer un merge de vos deux branches à partir de votre pull request précédente.
- [lien utile : tutoriel merge request](#)

2 Les bases de R

2.1 Installer R et RStudio

1. [Installer R sur le site officiel \(CRAN\)](#)
2. [Installer RStudio](#)

2.2 Les objets et les vecteurs

1. Dans le cours suivant ([lien ici](#)), suivre la partie sur les objets en effectuant quelques tests et manipulation dans un script. Le but n'est pas de tout retenir mais de comprendre la mécanique du langage.
2. Testez vous sur un quiz. Si vous souhaitez être classé pour obtenir un bonus, saisissez votre numéro étudiant. Le code du quiz vous sera transmis lors du TD.