Git and GitHub

Juan Carlos Villaseñor-Derbez (JC)

Hands-on: Building Your First Repository

The overall flow:

0. Create a GitHub account (if you don't have one), install git, authenticate (instructions)
1. Create a new repository on GitHub
2. Clone it to your computer
3. Add your project files
4. Make your first commit
5. Push to GitHub

Step 0: Make sure you have git, and an account and all that

```
which git
```

Step 1: Create a GitHub Repository

1. Go to github.com and sign in
2. Click the "+" icon in the top right
3. Select "New repository"
4. Choose a descriptive name: `my_first_repo`
5. Add a description: "Learning how to use GitHub"
6. Make it **Public**
7. Check "Add a README file"
8. Select gitignore tailored to R
9. Click "Create repository"

Step 2: Clone to Your Computer

In RStudio: 1. Go to File → New Project 2. Choose "Version Control" 3. Select "Git" 4. Paste your repository URL 5. Choose where to save it locally - Not on iDrive, not on DropBox, not on GoogleDrive… 6. Click "Create Project" 7. Verify that you now have a `README.md` and `.gitignore` files

Step 3: Set Up Your Project Structure

To create these folders in your repository:

```
environmental-data-project/
  data/
      raw/
      processed/
|      output/
  scripts/
      01_processing/
      02_analysis/
      03_content/
  results/
      figures/
      tables/
  docs/
  README.md
```

Use:

```
EVR628tools::create_dirs()
```

Step 4: Create / Modify the .gitignore File

Create a `.gitignore` file to exclude unnecessary files:

```
# R specific
.Rhistory
.RData
.Ruserdata
*.Rproj.user/

# System files
.DS_Store
Thumbs.db
```

Note: You can always add specific files with `git add -f filename`, edit the `.gitignore` file or use the git pane

Step 5: Add Your First "Analysis"

Create a simple R script in `scripts/analysis/`:

```
# Example: environmental_data_analysis.R
library(EVR628tools)
library(tidyverse)

# Create a simple plot
```

```r
p <- ggplot(data_milton, aes(x = iso_time, y = wind_speed)) +
  geom_point()

# Save plot
ggsave(plot = p, filename = "results/figures/first_plot.png") # <-- This shouldn't work, why
```

Step 6: Update Your README

Create a comprehensive README.md

```
# My first repo

## Description
Analysis of environmental data for EVR 628 course.

## Project Structure
- `data/`: Raw and processed data files
- `scripts/`: R scripts for analysis
- `results/`: Output figures and tables
- `docs/`: Documentation

## Author
[Your Name and email?]
```

Step 7: Stage-Commit-Push

- Stage, commit your R script
- Push
- Stage, commit your README
- Stage, commit your plot
- Push

Now lets colaborate

Adding someone to your repo

- Find a partner
- Decide who will be partner A and who will be partner B

Parter A

- Go to settings -> collaborators -> Add collaborator
- Add partner B's username and send an invitation
- Wait until B tells you they are done (~5 minutes)

...

- Pull

Parter B

- Give partner A your username
- Check your email
- Go to the repository
- Repeat the cloning process (in a different location)
- Make one change to partner A's graph
- Stage the file, then commit, and push
- Tell partner A you are done

Branching and pull requesting

Parter A

Parter B