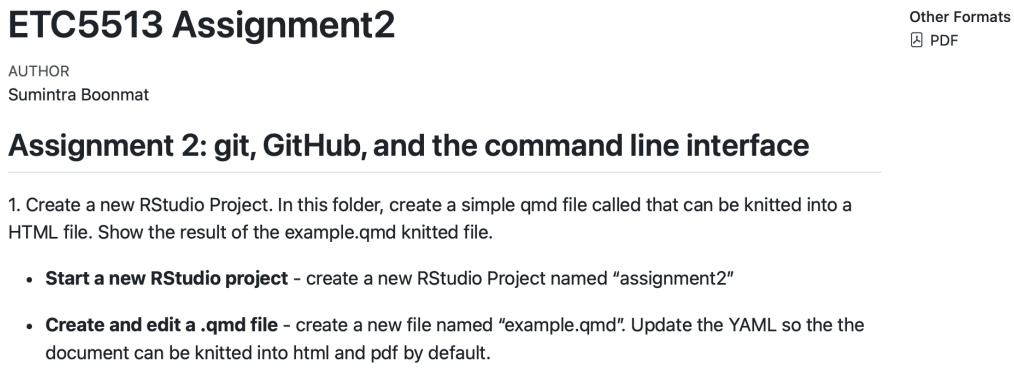
ETC5513 Assignment2

Sumintra Boonmat

Assignment 2: git, GitHub, and the command line interface

1. Create a new RStudio Project. In this folder, create a simple qmd file called that can be knitted into a HTML file. Show the result of the example.qmd knitted file.

- **Start a new RStudio project** - create a new RStudio Project named "assignment2"
- **Create and edit a .qmd file** - create a new file named "example.qmd". Update the YAML so the the document can be knitted into html and pdf by default.



2. From the command line interface, initialise this folder as a git repository and push it to the GitHub classroom repository.

- **Open the terminal** and change into the project directory.
  cd documents/assignment2
- **Initialize a Git repository** in the current folder. This sets up version control for your project
  git init
- **Create a *.gitignore* file** and ***add .Rproj.user/*** to exclude unnecessary RStudio project files from being tracked.

- **Stage all project files** for commit.
  git add .
- **Commit the changes** with a clear message describing what was done
  git commit -m “created a new project and qmd file”
- **On GitHub**, create a new repository manually. Make sure **not to include** a **README, .gitignore, or license**.
- **Copy the repository’s SSH URL** from GitHub.
- **Link the local repository to the remote GitHub repository**
  git remote add origin git@github.com:your-username/your-repository.git
  git remote add origin git@github.com:SumintraB/assignment2.git (this is the code URL to my repository)
- **Push your local repository to GitHub.**
  git push -u origin main

3. Create a new branch called testbranch. Modify the file example.qmd and add the changes to both the local and remote repositories.

- **Create and switch to a new branch called *testbranch***
  git branch testbranch - creates a new branch named testbranch
  git switch testbranch -switch to created branch
- **Modify the example.qmd**
- **Stage all the files**
  git add .
- **Make a commit**
  git push origin testbranch git commit -m “modified example.qmd for branching in step3”
- **Push the commit to the remote testbranch on Github**
  git push origin testbranch

4. Create a folder called data, and add the data from Assignment 1 to that folder. Amend the previous commit to include the data folder. Push this amended commit to the remote.

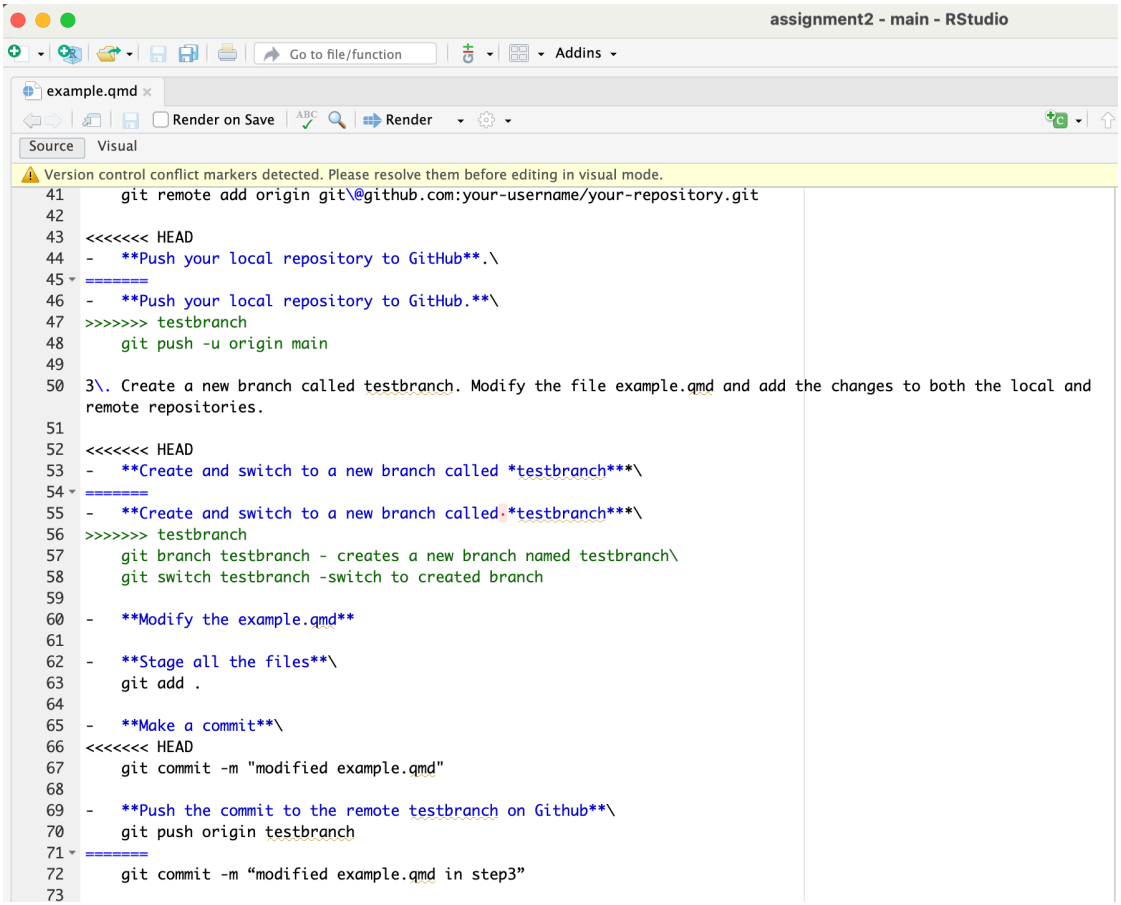
- **Create a new folder named *data* and place the data from assignment1**
- **Stage all the new files**
  git add .
- **Make a commit with message**
  git commit -m “added new data folder”
- **Push the changes to the remote testbranch**
  git push origin testbranch

5. Switch back to the main branch and modify example.qmd in a different way such that it will conflict with testbranch. Commit and push these changes

- **Switch back to main**
  git switch main
- **Modify change on example.qmd file**- update the same part as on testbranch so it creates conflict with the testbranch.
- **Stage and commit the change**
  git add .
  git commit -m “modified example.qmd on main”
- **Push the changes to main**
  git push origin main

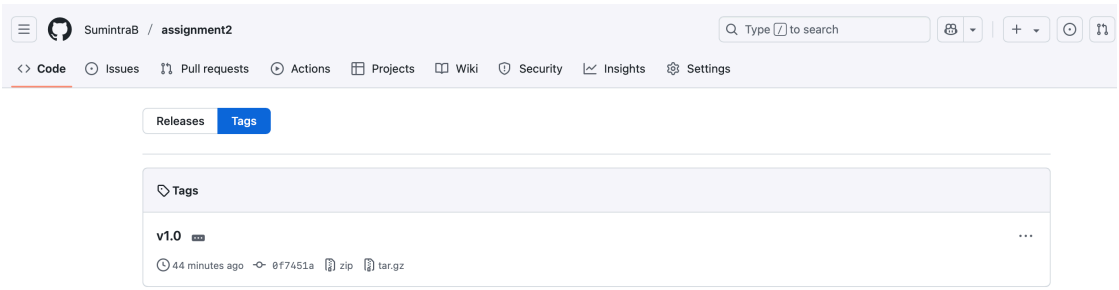
6. Merge the changes in testbranch onto main. Show the conflict and fixing the merge conflict. Push the changes to GitHub, showing the status and changes in GitHub.

- **Ensure sure you are on the main branch**
  git branch
  (if not)
  git switch main
- **Merge the testbranch into main**
  git merge testbranch
- **If the conflict appears, fix the conflict by choosing the preferred version and remove**
  «««<
  =======
  »»»>
- **Stage and commit the resolved conflict**
  git add .
  git commit -m “conflict resolved”
- **Push the updated to main branch**
  git push origin main



7. Tag this commit v1.0 on main using an annotated tag.

   - **Create an annotated tag**
     git tag -a v1.0
   - **Add the tag message**
     “Version 1.0 - merged testbranch and resolved conflict”
   - **Push the tag to remote**
     git push origin v1.0

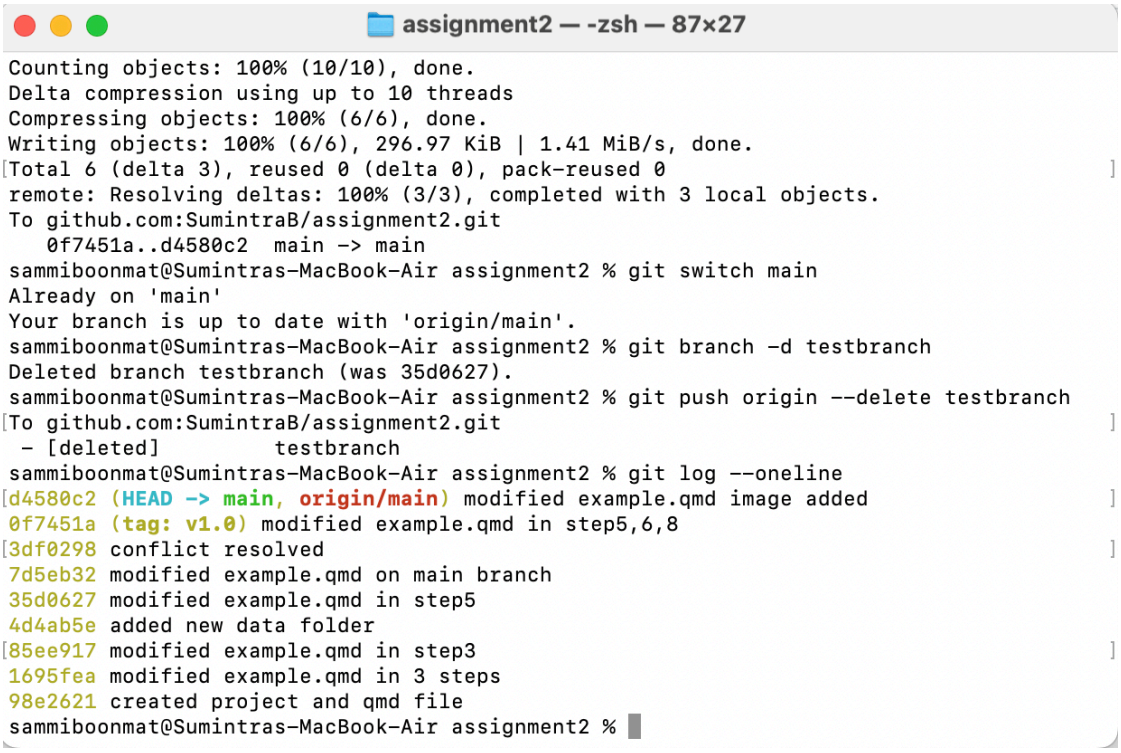


8. Delete branch testbranch locally and on the remote.

   - **Ensure you are on main branch** - branch cannot be deleted if you are in the branch that wish to be delete
     git switch main
   - **Delete the branch locally**
     git branch -d testbranch
   - **Delete the branch from the remote**
     git push origin —delete testbranch

9. Show the commit log in condensed form in the terminal.

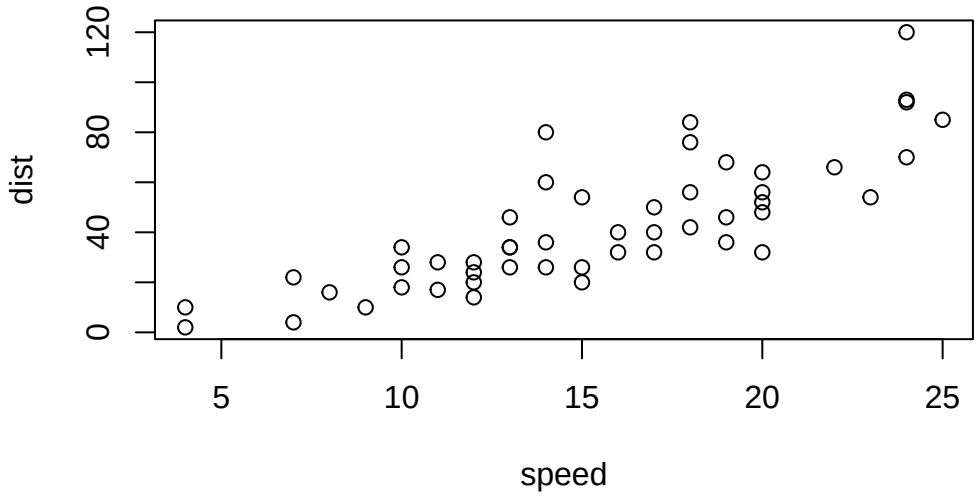
   git log —oneline



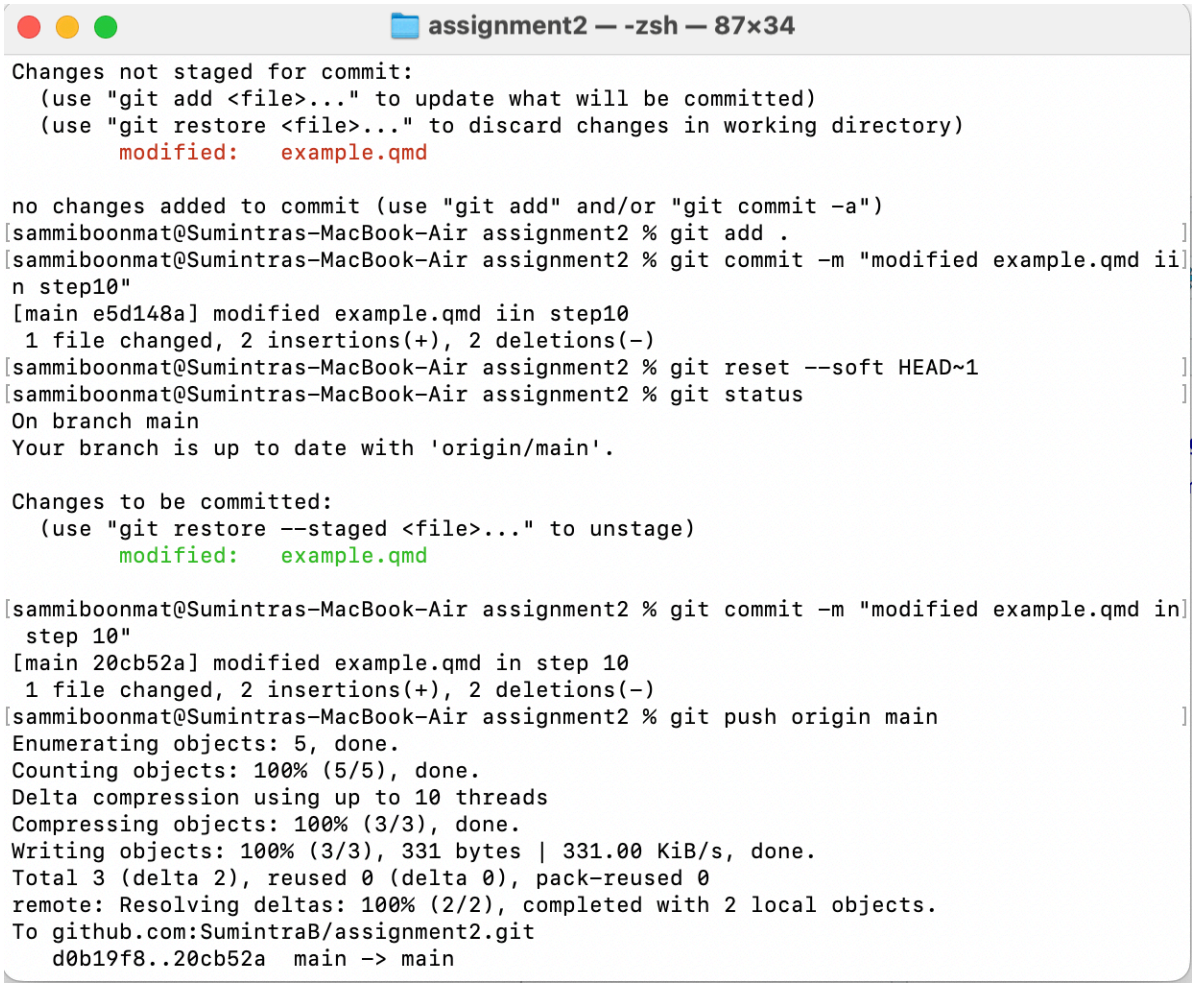
10. On main, create a new section in example.qmd that includes an easy to make plot. Commit the changes, and demonstrate using the command line interface how to undo the commit *without* losing your local changes.

Basic Plot

```
plot(cars)
```



- **Create a new simple plot**
- **Save the file and stage all changes**
  git add .
- **Commit the change**
  git commit -m “modified example.qmd in step 10”
- **Undo the commit, while keeping the changes on the file**
  git reset —soft HEAD~1


assignment2 — -zsh — 87×34
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified: example.qmd
no changes added to commit (use "git add" and/or "git commit -a")
sammiboonmat@Sumintras-MacBook-Air assignment2 % git add .
sammiboonmat@Sumintras-MacBook-Air assignment2 % git commit -m "modified example.qmd ii
n step10"
[main e5d148a] modified example.qmd iin step10
1 file changed, 2 insertions(+), 2 deletions(-)
sammiboonmat@Sumintras-MacBook-Air assignment2 % git reset --soft HEAD~1
sammiboonmat@Sumintras-MacBook-Air assignment2 % git status
On branch main
Your branch is up to date with 'origin/main'.
Changes to be committed:
(use "git restore --staged <file>..." to unstage)
modified: example.qmd
sammiboonmat@Sumintras-MacBook-Air assignment2 % git commit -m "modified example.qmd in
step 10"
[main 20cb52a] modified example.qmd in step 10
1 file changed, 2 insertions(+), 2 deletions(-)
sammiboonmat@Sumintras-MacBook-Air assignment2 % git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 10 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 331 bytes | 331.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:SumintraB/assignment2.git
d0b19f8..20cb52a main -> main