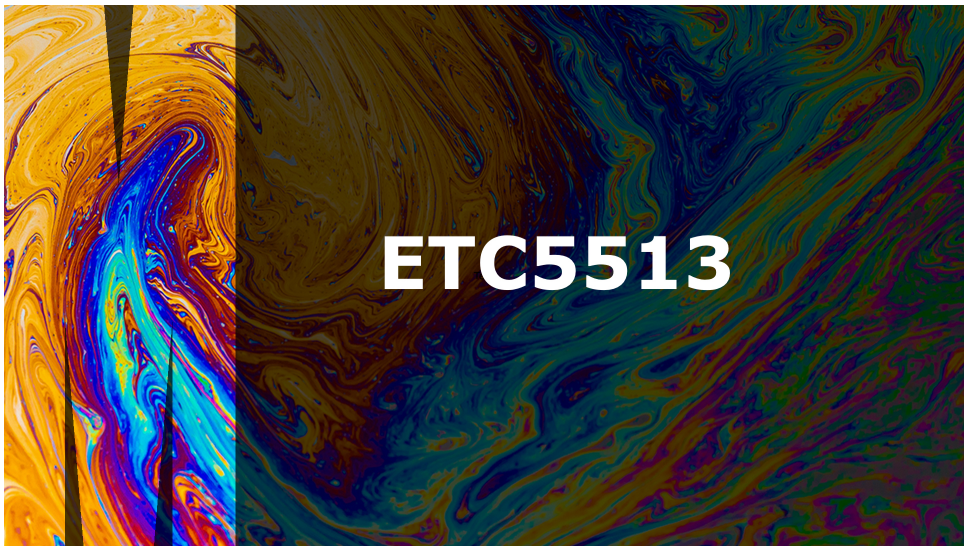

Reproducible and Collaborative Practices

Tutorial 7

Patricia Menéndez

Contents

Tutorial objectives:

- Practice version control workflow.
- Ammend commits.
- Visit past commits and create branches from those.
- Merging branches and deal with conflicts.
- Practice pull requests.
- Use GitKraken as a tool to visualize trees.
- Use git reset and git revert.

1 Forks, pull request, commits and inspecting differences between commits (work in pairs)

During this exercise make sure that you use your terminal/cli and also GitKraken to see the repo tree. Also keep an eye on your GitHub repo.

1. Create a new *public* GitHub repository called *Tutorial7-XX* and replace *XX* with your initials. Include a *README.md* file.
2. Create a new branch called *newbranchXX* and move the HEAD of your repo to the tip of *newbranchXX*
3. In your new branch add a new file called *exerciseXX.Rmd* and replace *XX* with your initials.
4. Stage, commit and push the changes into the remote repo
5. Create a new folder in *newbranchXX* called *Images* and add *Figure1.png* (you can find the figure in Moodle)
6. Stage, commit and push the changes into the remote repo.
7. Using *git log* and *git log –oneline* inspect the commits that you have made.
8. Exchange the details of your GitHub repo with your partner for the exercise.
9. Fork your partner's repository.
10. Clone locally your partner's repository **with all the branches**. **Hint:** See lecture slides page 36.
11. Continue working in your partner's branch and add the following to the *exerciseXX.Rmd* YAML:

```
title: "Reproducible and Collaborative Practices"
subtitle: "Tutorial 7"
author: "Your Name"
institute: "Department of Econometrics and Business Statistics"
output:
 pdf_document:
   toc: true
   toc_depth: 2
   number_sections: true
   highlight: tango
header-includes:
  - \usepackage{titling}
  - \pretitle{\begin{center}
    \includegraphics[width=5in,height=13in]{figs/front.jpg}
  - \posttitle{\end{center}}
  - \usepackage{fontawesome}
  - \usepackage[most]{tcolorbox}
  - \usepackage{xcolor}
  - \usepackage{sectsty}
  - \sectionfont{\color{olive}}
  - \usepackage{verbatim}
```

12. Use *git status* and *git log –oneline* to inspect your repo.
13. Stage *exerciseXX.Rmd*.
14. Unstage *exerciseXX.Rmd*.
15. Stage, commit and push the changes into the remote repo.
16. Amend this last commit. **Hint:**
    - git commit –amend
    - after that in your terminal you can use :q to get out of the text editor
    - git push –force

- You can also right click on your last commit in the GitKraken tree and select *edit commit message*.

17. Add one new section into the *exerciseXX.Rmd*.
18. Stage, commit and push the changes into the remote repo.
19. Inspect the differences between your last two commits. **Hint:** diff oldestcommit_SHA .. HEAD –color -words
20. Go back to a previous commit of your choice and create a new branch from there. **Hint:** git checkout SHA/SHA1.
21. Checkout into the new branch and add a new section into *exerciseXX.Rmd*. Then merge this new branch into your partner's branch.
22. Create a pull request to each other and accept the changes that your partner is suggesting for your repo dealing with any possible merging conflicts.
23. Once you have finished with this pull request merge the branch into master.

2 Closer look to commits, revert to previous commits and reset your repository.

1. Use GitKraken and inspect your previous exercise repo.
2. Inspect the tree and commits.
3. Create a new section in *exerciseXX.Rmd*.
4. Stage, commit and push the changes.
5. Add a new *latex list* inside the last section that you have created. **Hint:**
   `\begin{itemize} \item ... \end{itemize}`
6. Stage, commit and push the changes.
7. Go back and find the SHA number of the second last commit and use *git reset SHA*. Type *git status* and observe what has changed. How many commits do you have now in your repo now?
8. In the terminal use *git revert* to go back to one of your previous commits.