

STAT-S 610 Lab 5: Code Review

November 6, 2020

Code review is an important part of contributing to and maintaining a repository shared among collaborators. When changes are made to a repository, those changes must be reviewed by collaborators to make sure they are up to standards before they are merged to the main branch. During this review process, some things that both the contributor and the reviewers might have in mind are:

- Is the code correct? Do the algorithms make mathematical and logical sense?
- Is the code readable? Does it adhere to the team's code style guide?
- Can the code be improved in any way? Are there any inefficiencies or bottlenecks?
- What is the scope of this contribution? Do we want to add additional features or make the code more generalizable?
- Are there any tests we may want to add?

The basic code review process can be summed up into five parts:

1. Create a new branch and a pull request with your contributions.
2. Have reviewers look over the code to see if they approve of your contribution.
3. Make necessary changes to the branch.
4. Once code reviewers are happy with your contribution, they will approve your pull request.
5. Merge your pull request into the main branch.

Creating a pull request

1. Open your terminal and go to the directory for your repository. This should already be on GitHub. Make sure you are on the main/master branch and that it is synced with the remote repository by using `fetch/pull`.
2. Create a new branch using the `checkout` command and switch to it using the `branch` command. Make your changes and add them. Finally, `commit` your changes (don't forget to include a commit message).
3. We are finally ready to create the pull request. Enter the command `git push -u origin new-branch` (where `new-branch` is replaced with the name of your branch). A URL should be displayed on your terminal. Go to the URL, fill in or correct the empty fields, assign reviewers, and click on "Create pull request".

Reviewing code

There is no hard and fast rule here. For the purposes of this lab, we will follow Hadley Wickham's R style guide (in addition to making sure the code is mathematically/logically correct and so on).

In the URL for the pull request, there are fields where reviewers can add comments both on the pull request in general and on specific lines of code. This is a place where the contributor and the reviewers can have conversations about the contribution and what additional changes should be made.

As a reviewer, you should not feel shy about asking questions or pointing out bad code. The golden standard is for a reviewer to read the code and be able to understand everything that is happening. If this is not the case, make comments pointing out which parts of the code are unclear. Do not be afraid to point out code that is inefficient, inelegant, or limited in scope, even if it is “correct”. This is an opportunity to be as nitpicky and pedantic as you want.

Committing new changes

This part is the same as on Lab 4 and Homework 6. Based on your reviewers’ comments and suggestions, make additional changes to the code, then use the `add`, `commit`, and `push` commands. There may be several rounds of steps 2 and 3.

Approving and merging

Once the reviewers are happy with the changes, they can approve the pull request, at which point it can be merged to the main branch.

Once the pull request is merged, sync up your local main branch with the remote:

```
git checkout master  
git pull
```

Then use `git branch -d new-branch` to delete the local branch corresponding to the pull request.