Pre-publication Replication Example

Lars Vilhuber and David Wasser

August 28, 2019

Walking Through An Example

- We will work through an example of a pre-publication replication together
- We will use a replication package that has already been published on openICPSR: #110803
- You will need to set up an account on openICPSR
- Do each step on your computer
- Keep the Wiki open in another tab
- **We will not use Jira for this example, but Jira directions are included for your reference**

Outline

Outline

Access article and download materials

- Find the issue on Jira and advance it from **Open** to **In Progress**
- The pdf of the manuscript will be attached to the Jira issue
- The replication materials are on openICPSR: log in and search for the project number on the Find Data page
  - On the real Jira ticket, the project URL will already be in the Code Provenance field
- You should see a project named "Data and Code for Uncertainty and Business Cycles Replication File"
- On the righthand side, click **Download This Project**
- Fill out Entry Questionnaire

Outline

Create repo

- Repositories for AEA Verification are on https://bitbucket.org/aeaverification/
- Follow the instructions in the Wiki for creating a new repo
- **Today only**: Add your netid to the end of the repo name
- Repository name should be the same as the Jira ticket number, e.g. AEAREP-14
- **Today only**: Make the repo name "TEST-netid"

Populate repo

- Follow the instructions in the Wiki for cloning a repo
- Save the downloaded materials from the paper in your repo
- Now we will use git to `add`, `commit`, and `push` these files to Bitbucket
- In the terminal, navigate to the repo. Then:
  1. Use `git add` to add the appropriate files (careful!)

Populate repo

- Follow the instructions in the Wiki for cloning a repo
- Save the downloaded materials from the paper in your repo
- Now we will use git to `add`, `commit`, and `push` these files to Bitbucket
- In the terminal, navigate to the repo. Then:
  1. Use `git add` to add the appropriate files (careful!)
  2. Commit: `git commit -m "`*`REPO NAME`*` #comment Write your commit message here"`
     - ★ This is "smart commit" that syncs Jira and Bitbucket

Populate repo

- Follow the instructions in the Wiki for cloning a repo
- Save the downloaded materials from the paper in your repo
- Now we will use git to `add`, `commit`, and `push` these files to Bitbucket
- In the terminal, navigate to the repo. Then:
  1. Use `git add` to add the appropriate files (careful!)
  2. Commit: `git commit -m "`*`REPO NAME`*` #comment Write your commit message here"`
     - ★ This is "smart commit" that syncs Jira and Bitbucket
  3. Push: `git push origin master`

Populate repo

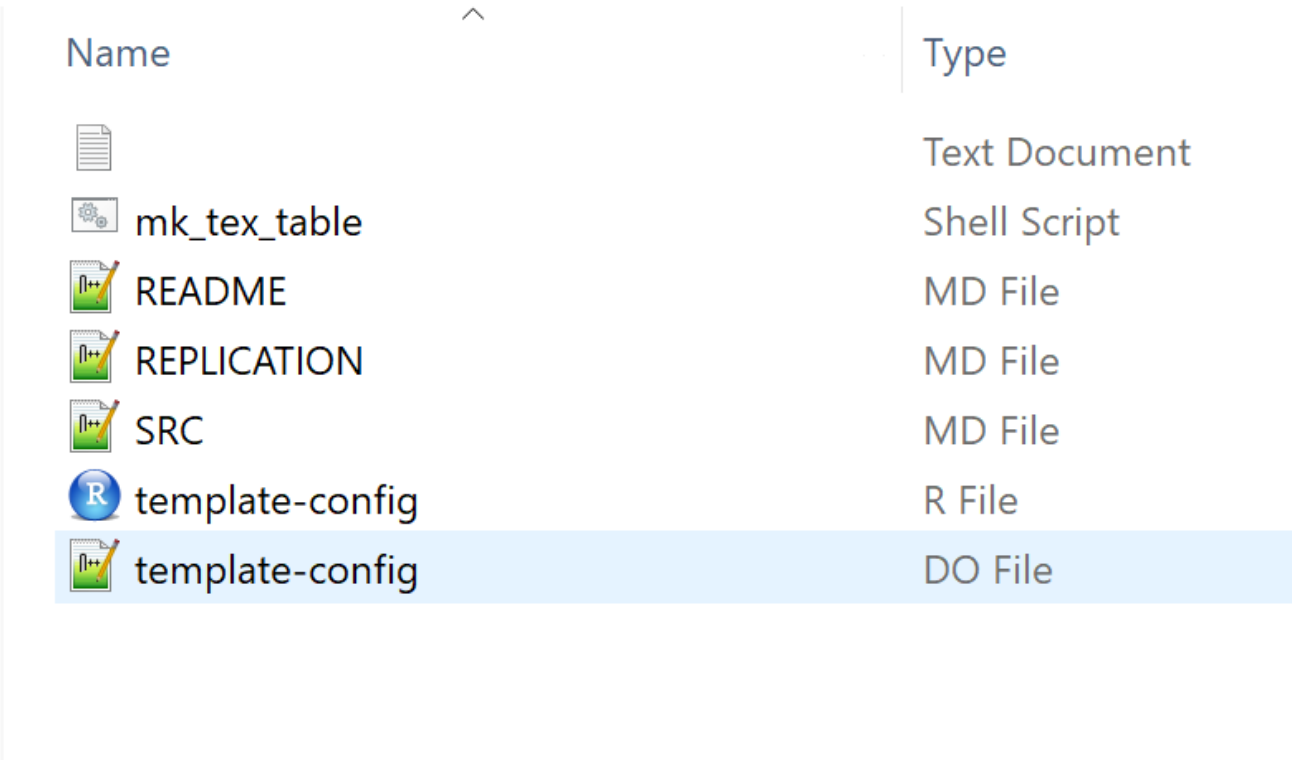
- Follow the instructions in the Wiki for cloning a repo
- Save the downloaded materials from the paper in your repo
- Now we will use git to `add`, `commit`, and `push` these files to Bitbucket
- In the terminal, navigate to the repo. Then:
  1. Use `git add` to add the appropriate files (careful!)
  2. Commit: `git commit -m "`*`REPO NAME`*` #comment Write your commit message here"`
     - ⋆ This is "smart commit" that syncs Jira and Bitbucket
  3. Push: `git push origin master`
- Check Bitbucket–our files should be there now

Outline

Download template

- Follow the instructions in the Wiki to download the template

Template looks like this



Outline

Download template

- Save only the necessary files in the repo!
- For this example, you should only save: REPLICATION.md, .gitignore, and SRC
- This replication is in Matlab, but most of the replications you do will be in Stata
- You can follow the post-publication example for Stata-specific help from this point forward
- The most important difference between the Stata process and the Matlab process is the confog.do file in Stata (see the post-publication example for more on this)

Outline

Jira Process

- Fill out the Entry Questionnaire, advance Jira to Code
- Fill out code provenance (if it has not been filled out already), advance Jira to Data
- Fill out location of data–typically the openICPSR project number, advance Jira to Verification

Outline

Verification

- Let's try to replicate Figure 1 today
- Log into Ciser
- Create a directory in Documents for this example called "jira_example"
- Use `git clone` to get your Bitbucket repo and its contents onto Ciser
- Open gen_figure1.m in Matlab and run it
- What do you find?

Outline

Committing and pushing to repo

- After running the code, we will write our report and then `git commit` and `git push` again.
- Always remember: commit frequently, push (at least) daily