

15.S60: Computing for Optimization and Statistics

Assignment 1

Due: January 11, 2022 8pm EST

For this assignment, you will explore a few of the tools from class in more depth. You have the option to choose which tools to focus on based on what aligns with your interests and computational needs. **Please select TWO of the four exercises below.** Submit your two screenshots to Canvas as a single PDF.

1. More Practice with Terminal and Git

Learning objective: Identify files they do not need to be tracked and instruct Git to ignore them. Perform file movement and simple pattern matching operations using commands in Terminal.

- (a) Read through a 5 minute tutorial on ignoring files in Git here: <https://swcarpentry.github.io/git-novice/06-ignore/index.html>
- (b) Clone the Github repository containing the files for this exercise to your local machine:

```
git clone https://github.com/alexschmid3/ignorerepo.git
```
- (c) We first want to re-organize our files. It seems our outputs folder accidentally got moved into our data folder. Move the outputs folder and its contents into ignorerepo using a Terminal command. Commit the changes to the repository.
- (d) There are many files in this repository which will take up valuable disk space. We want to track changes on our main file, run_all.jl, everything in src, and the text file datastuff.txt, while ignoring the rest. Create a .gitignore file and add the appropriate files. Commit the .gitignore file.
- (e) We now must remove everything from our repo and add it back, ignoring those files specified in .gitignore file. Run:

```
git rm -rf --cached .
git add .
git commit -m "The .gitignore file works now"
```

- (f) Add some text to three files: datastuff.txt, results_exp1.csv, and readData.jl. Run git status to see the result and take a screenshot of the output. Ensure that the changes in datastuff.txt and readData.jl are tracked, while the CSV file is not.

2. Connecting Engaging and Github

Learning objective: Appropriately clone, push, and pull code between a local machine, Github, and Engaging, then run a simple batch job on Engaging.

- (a) Follow the link below for instructions on how to generate an SSH key for your Github account, add your key to the SSH agent, then add the key to your Github account: <https://docs.github.com/en/get-started/quickstart/set-up-git>
- (b) Fork <https://github.com/alexschmid3/clusterrepo.git> to create a copy of the entire repo on your own Github account. To do this, find the “Fork” button in the upper right hand corner.
- (c) Clone your forked copy of the repo to your local computer.
- (d) Clone your forked copy of the repo to the cluster by starting a session in Terminal (macOS) or PuTTY (Windows) as we did in class, then running:

```
git clone https://github.com/<username>/clusterrepo.git
```

You now have a copy of the repo on your machine, a copy on Github, and a copy on the cluster. You can make changes to the repo on your local machine, push the changes to Github, and then pull the changes from Github on the cluster. None of this will effect the original repository we forked (the one on alexschmid3's Github).

- (e) On your local machine, open the file `mybatchjob.sh` in a text editor. Edit the appropriate lines to ensure your email will receive the notifications when the batch job completes (there are three places to edit). Commit these changes and push them to Github.
- (f) Now, we need to update the version on the cluster. First, in the terminal of the cluster, navigate to the `clusterrepo` directory. Then, pull the changes from Github.

```
git pull
```

You can check whether this worked by printing out the contents of `mybatchjob.sh` and seeing if it reflects the changes you made.

- (g) Now let's kickoff the batch job using the following command:

```
sbatch mybatchjob.sh
```

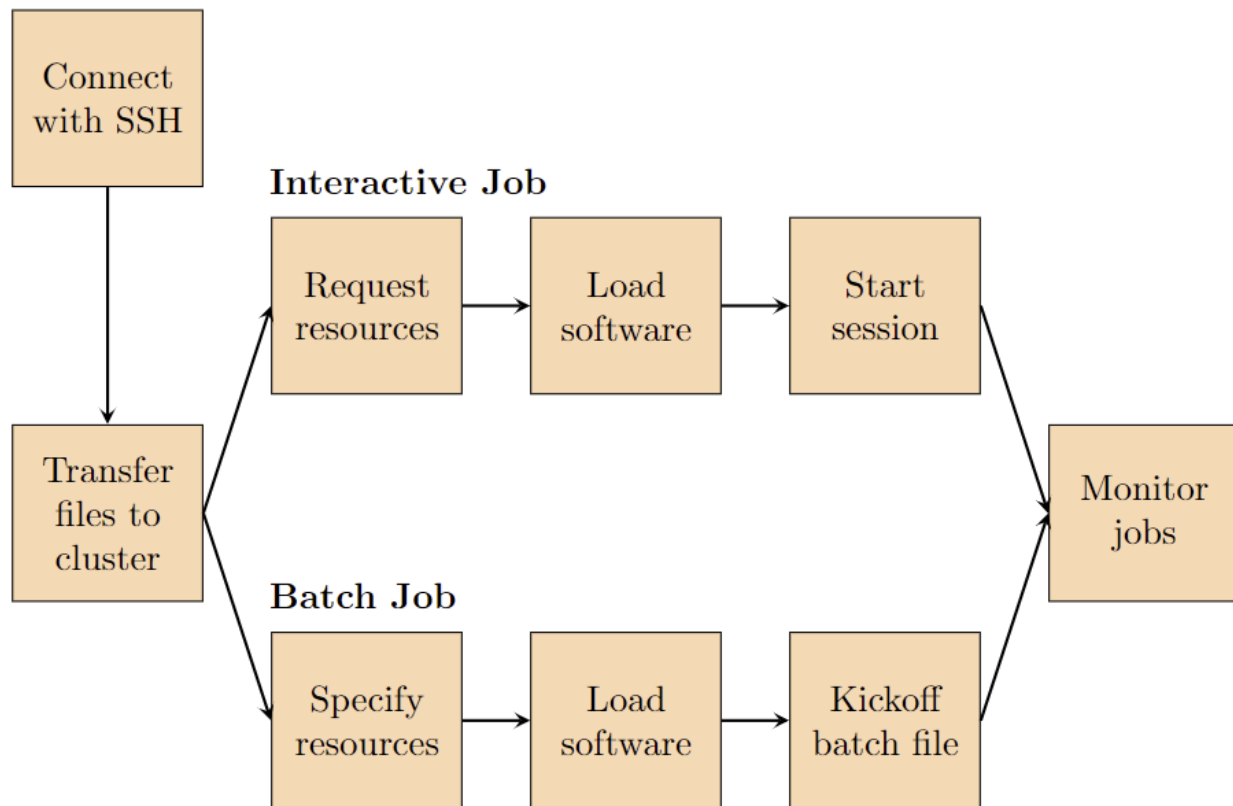
This will create a batch job that loads Julia, then runs a very short script. Depending on how busy the cluster is, your job may be queued. Once it begins, it should finish right away.

- (h) Check the output by printing out the output file, `myoutputfile.out`. Take a screenshot of the output from the terminal.

In the future, you can create your own Github repository for your projects and set up this same pipeline for a clean way to manage your computational experiments on the cluster!

3. LaTeX Figures with TikZ

Learning objective: Create a simple figure using the TikZ package and reference the figure in a LaTeX document.

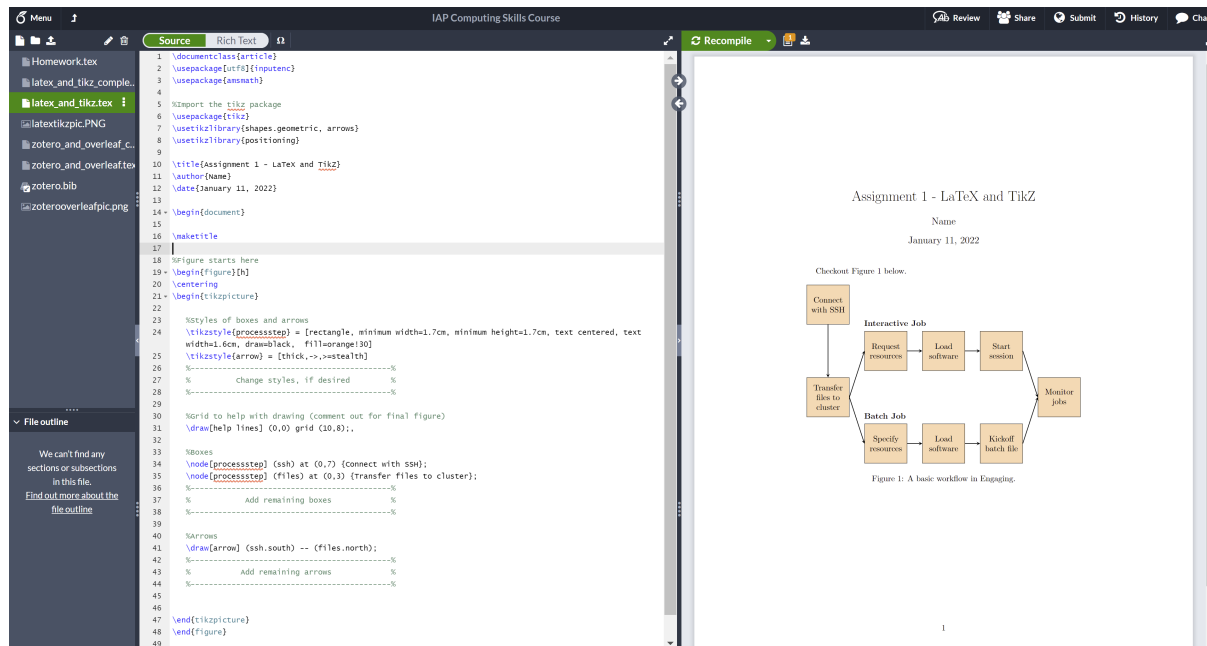


- (a) Recreate the above figure in Overleaf using TikZ, starting from the template in the course Github. If you aren't sure how to draw something, check here <https://www.edpif.org/documents/latex/intermediate/beamer-tikz/doc/minimaltikz.pdf> or search online.

1-comp-literacy\homework\latex-and-tikz.tex

Don't worry about matching exactly, just be sure to convey the same information.

- (b) Add a caption of your choice.
- (c) Add a label to the figure.
- (d) Add a line of text to the LaTeX document that references the label of the figure.
- (e) Change the color of the boxes to the color of your choice.
- (f) Take a screenshot of your final document in Overleaf, with the LaTeX shown on the left and the resulting one page document including the figure and text on the right. For example,



4. Create a Zotero Bibliography in Overleaf

Learning objective: Create a small Zotero library, link it to Overleaf, and build a bibliography to cite the papers from the library.

- (a) Add five papers of your choice to your Zotero library. These can be from your downloads or from an online source like Google Scholar. Feel free to cite five news articles, books, or other materials if that is more relevant for you.
- (b) Ensure that all the citation information for the five papers was added properly to your Zotero library (i.e. author names, title, dates, etc. captured correctly). If not, edit the citation information directly in Zotero desktop. Doing this small bit of editing upfront will ensure that everything in your Zotero library is ready to cite quickly when you go to write your next literature review.
- (c) Set up a connection between your Overleaf and your Zotero library.
- (d) Create a BibTeX file for your Zotero library.
- (e) Create a short LaTeX document in Overleaf starting from the template in the Github.

1-comp-literacy\homework\zotero-and-overleaf.tex

Include some text that cites all five of your references and a References section with all of the proper citations. Nothing fancy! No need to even read the papers or summarize them, just practice adding papers to Zotero and referencing them in LaTeX.

- (f) Take a screenshot of your final document in Overleaf, with the LaTeX shown on the left and the resulting one page document on the right. For example,

Menu

1

IAP Computing Skills Course

Review

Share

Submit

History

Chat

Homework.tex

zotero_and_overleaf.tex

zotero.bib

File outline

We can't find any sections or subsections in this file.

Find out more about the file outline

Source

Rich Text

Recompile

```

1 \documentclass{article}
2 \usepackage{ctex} \usepackage{amsnath}
3 \usepackage{amsmath}
4
5 %Bibliography package
6 \usepackage{biblatex}
7
8 \addbibresource{zotero.bib}
9
10 \title{Assignment 1 - Zotero and Overleaf}
11 \author{Name}
12 \date{January 11, 2022}
13
14 \begin{document}
15
16 \maketitle
17
18 %------%
19 % Put text and references here %
20 %------%
21
22
23
24 %Prints the bibliography at the end of the document
25 \printbibliography
26
27 \end{document}
28

```

Assignment 1 - Zotero and Overleaf

Name

JANUARY 11, 2022

Here is some text citing my first paper, [2]. I found another paper on Google Scholar, [4], and my advisor recommended a third [1]. I saw a cool new preprint on arXiv, [5], which pairs nicely with this review paper I saw on Twitter, [3].

References

- [1] R. Bakhari, E. Hadjiconstantinou, and A. Mingoski. "An Exact Algorithm for the Capacitated Vehicle Routing Problem Based on a Two-Commodity Network Flow Formulation". en. In: *Operations Research* 52.5 (Oct. 2004), pp. 723-728. issn: 0030-364X, 1526-5663. doi: 10.1287/opre.1040.0141. url: <http://pubsonline.informs.org/doi/abs/10.1287/opre.1040.0141> (visited on 12/31/2021).
- [2] Natasha Beldar et al. "The Continuous-Time Service Network Design Problem". en. In: *Operations Research* 65.5 (Oct. 2017), pp. 1303-1321. issn: 0030-364X, 1526-5663. doi: 10.1287/opre.2017.1624. url: <http://pubsonline.informs.org/doi/abs/10.1287/opre.2017.1624> (visited on 12/31/2021).
- [3] Quentin Cappart et al. "Combinatorial optimization and reasoning with graph neural networks". en. In: *arXiv:2102.09544 [cs, math, stat]* (Apr. 2021). url: <http://arxiv.org/abs/2102.09544> (visited on 12/31/2021).
- [4] Sanjiv Doshi et al. "A Time Bucket Formulation for the Traveling Salesman Problem with Time Windows". en. In: *EVFORUM Journal on Computing* 24.1 (Feb. 2012), pp. 132-147. issn: 1091-9856, 1526-5528. doi: 10.1287/ijoc.1100.0432. url: <http://pubsonline.informs.org/doi/abs/10.1287/ijoc.1100.0432> (visited on 12/31/2021).
- [5] Jie Zhou et al. "Graph neural networks: A review of methods and applications". en. In: *AI Open* 1 (2020), pp. 57-81. issn: 2666-0510. doi: 10.1016/j.aiopen.2021.01.001. url: <https://linkinghub.elsevier.com/retrieve/pii/S2666051021000012> (visited on 12/31/2021).