Paul's paper workflow

Paul G. Constantine[a], Johnny Q. Coauthor[b]

[a]*Department of Applied Mathematics and Statistics, Colorado School of Mines, Golden, CO 80211*
[b]*Department of Stuff, University of Somewhere, Somewhere, TX 99999*

Abstract

This document describes how to set up the workflow for a paper collaborating with Paul.

Keywords: workflow, latex, bibtex

1. Introduction

This document describes the things you need to set up to follow Paul's workflow for writing papers. Here's the software you need to get started:

1. LaTeX distribution that enables you to run `latex`, `pdflatex`, `bibtex`, `dvips`, and `ps2pdf` from the command line,
2. text editor (I use Emacs),
3. Bibtex reference manager (I use JabRef (`jabref.org`) because it's based on Java so it's cross platform),
4. Dropbox for sharing a folder.

There are certainly plenty of ways to share LaTeX documents and papers. This is just what's worked for me. I have a directory structure for a paper with the following directories:

- `paper_v0`, where the paper tex files are along with a `figs` directory; if big changes happen, I might make a new `paper_vX` folder
- `references`, where I keep PDFs of papers I'm referencing

Email address: `paul.constantine@mines.edu` (Paul G. Constantine)

- `code`, where I keep scripts for numerical experiments (helps reproducibility and recreating figures)
- `arxiv`, I make a separate folder for the version I put on arXiv
- `reviewer_response`, where I keep a text file of the initial reviews and a LaTeX file with reviewer responses
- `latexdiff`, when I submit a revision I include the results of `latexdiff` (see example syntax in folder)
- `paper_r0`, a folder for the revision; this usually begins as a duplicate of `paper_v0`

2. LaTeX

There are several text editors that include tools for working with LaTeX. I don't really like any of them. Emacs has a natural "latex-mode" that does some syntax highlighting—and there's a lot you can do the integrate LaTeX with Emacs (see, for example, Nick Higham's blog: `https://nickhigham.wordpress.com/category/latex/`). However, I don't use any of it. I just use Emacs to create .tex documents.

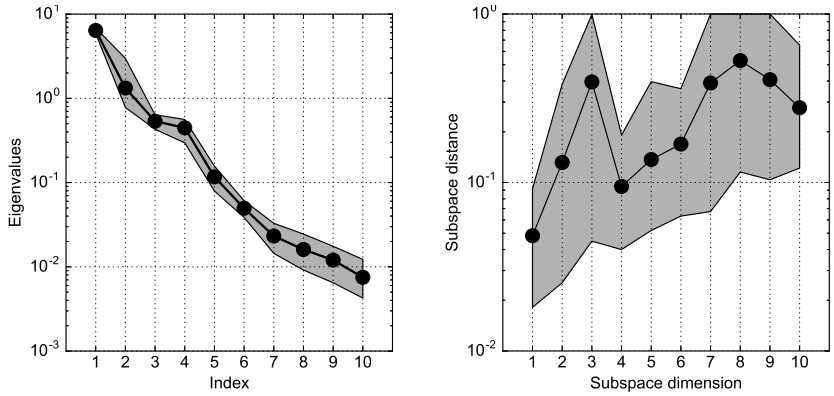
I compile a completed .tex file with a Makefile that executes LaTeX commands. From the command line, just type `make`. My Makefile's contain two ways to compile: (i) with `latex` and (ii) with `pdflatex`. The latter is preferred, but some journal templates need to use the former (for some reason).

Here are some sample environments I use regularly in papers. See Algorithm 1 and Figure 1 with Subfigures 1a and 1b.

Algorithm 1 My favorite algorithm

Given some stuff, do the following:

1. Step 1
2. Step 2



(a) Label 1 (b) Label 2

Figure 1: And this is where you put the caption

I usually make figures in the code directory and manually move them to the `paper_v0/figs` directory. That way, if I change the figure in the code because I'm playing around, then it doesn't immediately change the paper figure.

3. Code and examples

I put all the scripts that run the numerical experiments and the data they generate (assuming it's a resonable size) in the `code` directory. Doing this makes it easier to return to the code that generated a particular figure. I also use `git` and `bitbucket.org` to make the scripts accessible online. See my existing scripts at `bitbucket.org/paulcon`.

4. Bibtex and references

I use Jabref as a reference manager for citations like this one [1]. To get the bibtex records for papers, I find the paper on its journal's webpage and click something like 'Download citation' that most of the journals have. That's a good place to start, but the bibtex records are often imperfect.

Acknowledgments

Thanks to all my coauthors for helping me hone my workflow.

References

[1] P. G. Constantine, Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies, SIAM, Philadelphia, 2015.