Tools for Open Geospatial Science

Vaclav Petras, Anna Petrasova, Helena Mitasova

North Carolina State University, Center for Geospatial Analytics, USA (wenzeslaus@gmail.com, vpetras@ncsu.edu)

NC STATE UNIVERSITY Center for Geospatial Analytics

Highlights

Course highlights

- Complete graduate-level course with all material available.
- Course taught at NCSU CGA, for 15 on-campus and off-campus students, fall semester 2017.

Motivation

- *It's impossible to conduct research without software...* (Hettrick et al., 2014; Hettrick, 2014)
- *Software [...] developed as part of novel methods is as important for the method's implementation [...] Such software [...] must be made available to readers upon publication.* (Nature Methods, 2007).
- Various authors over the last 20 years identified that text is less than sufficient way of communicating research and described reproducibility spectrum (Buckheit and Donoho, 1995; Peng, 2011; Rodriguez-Sanchez et al., 2016; Marwick, 2017).

Course Syllabus

- Introduction to and motivation for open science
- Collaborative writing of scientific papers (Authorea, Markdown)
- Advanced tools for papers and reports (Overleaf, LaTeX)
- Revision control systems and wiki technologies (Git, GitHub)
- How open source communities and development work
- QGIS, a free and open source geographic system
- Command line and remote access to computational resources
- Command line and Python tools for geospatial work (GDAL)
- GRASS GIS as software for geospatial research
- Publishing data on web (data repositories, OpenLayers)
- Combining text, code and results into one document (Jupyter)
- Publishing code as part of an open source project
- Reproducible computational environments (Docker)
- Writing and reproducing an open science paper

Scripting

Automation in lab work, repeatability by others, and review by peers is enabled by scripting. However, the natural way to capture workflows in graphical user interface is taking screenshots such as this series from GRASS GIS:

Scripting is a more efficient way of recording the same information which can be edited and automatically processed. The following Bash command is equivalent of the three screenshots above:

```
r.in.lidar input=points.las output=elevation -e
```

Languages used in the class: Python and Bash; Alternatives: R, Ruby, Octave, Julia, ...

Versioning & Publishing Code

File format and software-dependent concepts of file versioning are replaced by scalable and robust techniques.

Software used in class: Git (`git`). Alternatives: Subversion (`svn`), Mercurial (`hg`), ...
Hosting option used in class: GitHub. Self-hosted open alternatives: GitLab, Gogs, Gitolite, Trac, ...
Alternative services: GitLab, Bitbucket, ...

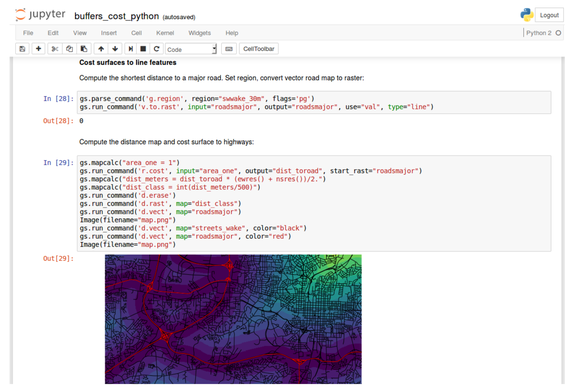
Publishing code can be as simple as uploading file online using a web browser, however more advanced ways such as integration into a larger open source project bring many benefits.

Why? Preprocessing, visualization, and user interface (GUI, CLI, API), inputs, outputs, memory management and other common features, integration with existing analytical tools, long-term maintenance. Options: R package, Python package, GRASS GIS module, QGIS plugin, ... Integration gradient: unofficial extension - integrated extension - code addition. Licensing and criteria to choosing the project (Schweik and English, 2012) need to be part of the course.

GRASS GIS GPL Free Software Free as in Freedom

Computational Notebooks

Computational notebooks are interactive documents with text, code, and figures. Notebooks work for most languages and can be exported to many formats for publication online or print. Content can range from class assignments to full papers or interactive figures in published papers.



Used in the class: Jupyter Notebook; Alternatives: R Markdown (Notebook), Emacs Org-mode, ...

Runtime Environments

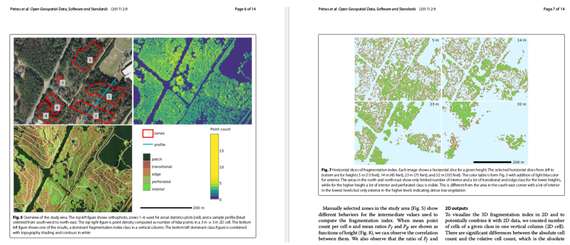
To run any code, various dependencies need to be available, which is often challenging. Solutions include Docker, Vagrant, and virtual machines which create full and self-contained environments. The following is a example of environment description for Docker:

```
# Dockerfile
FROM ubuntu:16.04
RUN apt-get update
RUN apt-get install -y python sqlite3
WORKDIR /data
```

Examples of Research

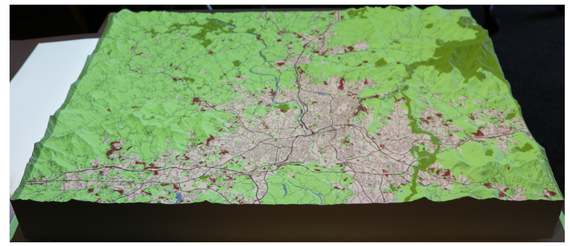
Lidar Analysis

Lidar analysis paper by Petras et al. (2017) is an example of research which provided scripts needed to produce figures presented in the paper as well as reusable code which was published as a module in GRASS GIS Addons repository.



Urbanization

Landscape change and urbanization paper by Petrasova et al. (2016) is an example of research which turned unpublished code into reusable tool for modeling available as a set of modules in GRASS GIS Addons repository.



References & Resources

Buckheit, J. B. and Donoho, D. L. (1995). Wavelab and reproducible research. In *Wavelets and statistics*, pages 55–81. Springer.

Hettrick, S. (2014). It\'s impossible to conduct research without software, say 7 out of 10 uk researchers. *Software and research*, 5:1536.

Hettrick, S., Antonioletti, M., Carr, L., Chue Hong, N., Crouch, S., De Roure, D., Emsley, I., Goble, C., Hay, A., Inupakutika, D., and et al. (2014). Uk research software survey 2014.

Marwick, B. (2017). Computational reproducibility in archaeological research: Basic principles and a case study of their implementation. *Journal of Archaeological Method and Theory*, 24(2):424–450.

Nature Methods (2007). Social software. *Nature Methods*, 4:189.

Peng, R. D. (2011). Reproducible research in computational science. *Science*, 334(6060):1226–1227.

Petras, V., Newcomb, D. J., and Mitasova, H. (2017). Generalized 3D fragmentation index derived from lidar point clouds. *Open Geospatial Data, Software and Standards*, 2(1):9.

Petras, V., Petrasova, A., Harmon, B., Meentemeyer, R. K., and Mitasova, H. (2015). Integrating free and open source solutions into geospatial science education. *ISPRS International Journal of Geo-Information*, 4(2):942–956.

Petrasova, A., Petras, V., Van Berkel, D., Harmon, B., Mitasova, H., and Meentemeyer, R. (2016). Open source approach to urban growth simulation. *Int Arch Photogramm Remote Sens Spat Inf Sci*, 41:B7.

Rodriguez-Sanchez, F., Pérez-Luque, A. J., Bartomeus, I., and Varela, S. (2016). Ciencia reproducible: qué, por qué, cómo. *Revista Ecosistemas*, 25(2):83–92.

Schweik, C. M. and English, R. C. (2012). *Internet success: a study of open-source software commons*. MIT Press.

goo.gl/g8pFiE

Course material is available at the above address and the formats and software used for publishing is described in Petras et al. (2015).