Tools for Open Geospatial Science

Vaclav Petras, Anna Petrasova, Helena Mitrasova
North Carolina State University, Center for Geospatial Analytics, USA (vpetrasav@ncsu.edu, vpetrasav@ncsu.edu)

NC STATE UNIVERSITY Center for Geospatial Analytics

Course Syllabus

- Introduction to and motivation for open science
- Collaborative writing of scientific papers (Authoria, Markdown)
- Advanced tools for papers and reports (Overleaf, LaTeX)
- Revision control systems and web 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

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... (Hetrick et al., 2014; Hetrick, 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).

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:

> Automatization 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.lae.output=elevation=e

Languages used in the class: Python and Bash. Alternatives: R, Raku, Octave, Julia.
```

Versioning & Publishing Code

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

```
> git commit present_data.py --message "replaced part of the msc equation"
```

```
> git commit present_data.py --message "replaced part of the msc equation"
```

Used in the class: Git (GitHub). Alternatives: Subversion (SVN), Mercurial (hg), ... Housing option and on class: GitHub. Self-hosted open alternatives: GitLab, Gogs, GitTree, ... Alternative versions: 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.

```
> Who: Programming, visualization, and user interfaces (GUI, CLI, API), inputs, outputs, memory manage-
> ment and other common features, integration with existing analytical tools, long-term maintenance
> Guidelines: Python package. Python package. GRASS GIS model. QGIS plugin. ... Integration gradient:
> use of open-source tools and libraries, software standards, and community involvement.
```

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 an 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


