Section 001
CDS 411 *Modeling and Simulation II*

Course toolbox

Class 1

Getting started with Anaconda

Anaconda: things to discuss

- What is Anaconda?: https://www.anaconda.com/what-is-anaconda/
- Where to find Anaconda: https://www.anaconda.com/download
- The (new) initial setup, read the release notes for details: https://conda.io/docs/release-notes.html#id24
- What are virtual environments, documentation here: https://conda.io/docs/user-guide/tasks/manage-environments.html

The environment.yaml file

```yaml
---
name: cds411
channels:
  - defaults
dependencies:
  - jupyter=1.0.0
  - jupyterlab=0.34.3
  - matplotlib=2.2.3
  - nodejs
  - numpy=1.15.0
  - pandas=0.23.4
  - pandoc=2.2.3.2-0
  - python=3.6.5
  - scikit-learn=0.19.1
  - scipy=1.1.0
  - seaborn=0.9.0
  - statsmodels=0.9.0
  - pip:
    - yapf==0.23.0
...
```

The environment.yaml file

```yaml
---
name: cds411
channels:
  - defaults
dependencies:
  - jupyter=1.0.0
  - jupyterlab=0.34.3
  - matplotlib=2.2.3
  - nodejs
  - numpy=1.15.0
  - pandas=0.23.4
  - pandoc=2.2.3.2-0
  - python=3.6.5
  - scikit-learn=0.19.1
  - scipy=1.1.0
  - seaborn=0.9.0
  - statsmodels=0.9.0
  - pip:
    - yapf==0.23.0
...
```

- Save the `environment.yaml` file to your computer, then run

```
conda env create --file environment.yaml
```

- To activate environment after the installation completes, run

```
conda activate cds411
```

Conda commands to remember

```
# Using the conda package manager
conda search <package-name>    # Search for <package-name>
conda install <package-name>   # Install <package-name>
conda list                     # List installed packages

# Create new virtualenv named <my-env> with Python 3.6
conda create -n <my-env> python=3.6

# Managing your conda environments
conda env list                 # List your virtualenvs
conda activate <my-env>        # Use <my-env> virtualenv
conda deactivate               # Deactivate virtualenv
conda remove -n <my-env> --all # Remove virtualenv

# Install environment using environment.yaml file
conda env create --file environment.yaml

# Update environment using environment.yaml file
conda env update --file environment.yaml
```

JupyterLab

JupyterLab: things to discuss

- What is JupyterLab?
- Install plugins
  - Install the `git` plugin:

    ```
    jupyter labextension install @jupyterlab/git
    ```

  - Install the GitHub plugin:

    ```
    jupyter labextension install @jupyterlab/github
    ```

- **Launching JupyterLab**

  Open up a command prompt, activate the `cds411` conda environment, and run

  ```
  jupyter lab
  ```

  By default, your web browser may launch and show you an interface. If it doesn't, open your browser and type `localhost:8888` in the address bar and press enter.

- JupyterLab tour: where can you find things and suggested settings

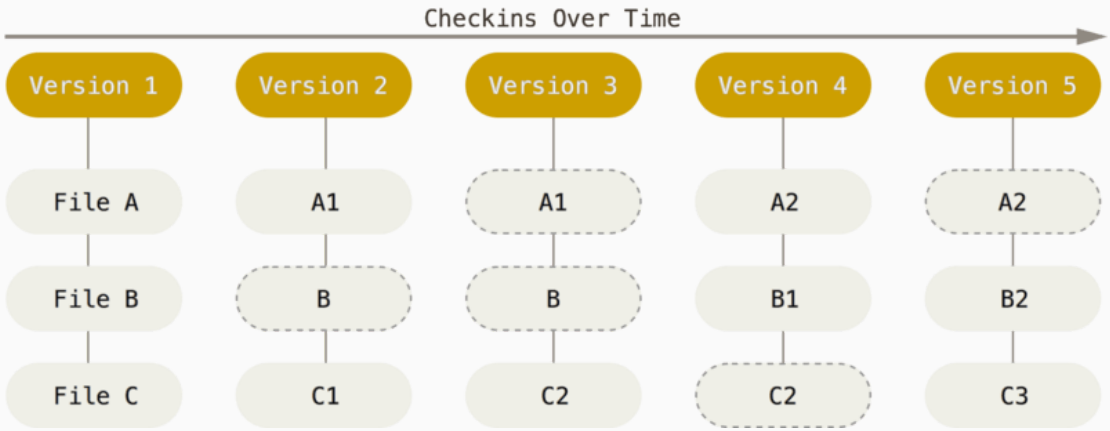
`git` and GitHub

Version control

> *What is "version control", and why should you care? Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.*

Source: https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control

How git does version control



> *Git thinks of its data more like a series of snapshots of a miniature filesystem. With Git, every time you commit, or save the state of your project, Git basically takes a picture of what all your files look like at that moment and stores a reference to that snapshot. To be efficient, if files have not changed, Git doesn't store the file again, just a link to the previous identical file it has already stored. Git thinks about its data more like a stream of snapshots.*

Source: https://git-scm.com/book/en/v2/Getting-Started-Git-Basics

GitHub

- What is GitHub?
- Navigating the site: http://book.cds101.com/navigating-the-github-site.html
- The repository page: http://book.cds101.com/repositories.html

Credits

License

Creative Commons Attribution-ShareAlike 4.0 International