Short video lecture

- Libraries, packages, modules
- Using miniconda package manager
- Linear Normal DGP in Python

Libraries

- General term for reusable code
- Collections of functions (and other types of code such as class definitions)
- Allow us to use functions others have written as reusable modular algorithms
- Open source, incredible variety
- Ecosystem + community

C libraries

- Standard library built in
- Download others as needed
- Use C libraries
  - `#include <stdio.h>`
  - `#include <math.h>`
  - might need compilation options that tell the compiler to use precompiled libraries, e.g. for math library
  - `gcc my_prog.c -o my_prog -lm`

Packages

- R, Python
- Structured collections of code with a language specific format
- Might also contain help files or documentation

R Library

- “The R library”: collection on your computer
- Install R packages into this library using the R package manager
  - R CLI & Positron:
  - `install.packages(“package_name”)`
- Use packages from this library
  - R: `library(package_name)`
  - makes all the functions available in the current workspace
  - R: `package_name::function_name`
  - use a particular function from a particular package that is not necessarily loaded

R

Python libraries

- Python package structure
  - directory with `.py` files
  - module = `.py` file with functions inside
- Several package managers
  - conda is default for data science
  - suggest: install miniconda
  - pip is common for basic python
  - Bash: `conda install package_name -c conda-forge`
- Use Python modules and functions
  - `import numpy as np`
  - `import matplotlib.pyplot as plt`
  - `from matplotlib.pyplot import plot`

Py

Linear Normal DGP in Python

- See code
- Numpy library
- Data structure: numpy ndarray
- Vectorized operations
- Random number generator
- Matplotlib library