Bonus Lecture! Creating Your Own Python Module/Package

05/13/22

Agenda

- Part 1: Reviewing Functions
- Part 2: Moving Your Functions to a Module (.py file)
- Part 3: Making Your Own Local Package
- Part 4: Publishing a Package on PyPi (pip)

Repository for Today

- Repo Link: https://github.com/coding-dojo-data-science/making-your-own-python-modules-and-package
    - There are 2 notebooks covering different parts.

- In 1_Creating Your Own Python Modules & Package.ipynb:
    - Part 1: Reviewing Functions
    - Part 2: Moving Your Functions to a Module (.py file)
    - Part 3: Making Your Own Local Package

- In 2_Publishing a Package on PyPi (pip).ipynb
    - Part 4: Publishing a Package on PyPi (pip)

Parts 1-3: Functions, Modules, & Local Packages

See: 1_Creating Your Own Python Modules & Package.ipynb

Part 1: Reviewing Functions

- Concepts to review:
    - Defining Function Arguments/Parameters:
        - Positional Arguments
        - Keyword Arguments
    - Return Statements
    - Function Scope
    - Docstrings

Part 2: Moving Functions to an External File (Module)

- Using VS Code to work with your repo contents.
  - Creating a .py file
  - Moving functions over
  - Considerations/Gotchas

- Using autoreload inside your jupyter notebook

- Using docstring-generation extension

Part 3: Creating a Local Python Package

- Create a new folder (name is the package name)
- Add a __init__.py file.
- Move your module (.py) from earlier into the new folder.
- Import the modules inside of your __init__.py.

- Now that we have been able to import our package from inside the same repo, we want to import it anywhere on our local machine.

- If we add the the folder name that contains to package to our terminal's settings file, we can use it anywhere on our local machine.

Part 4: Publishing a PyPi Package

See: 2_Publishing a Package on PyPi (pip).ipynb

Making a Python Package with CookieCutter

- See: 2_Publishing a Package on PyPi (pip).ipynb
- Use the instructions in the notebook above IN CONJUNCTION with the official cookiecutter tutorial

IMPORTANT GOTCHAS

- You cannot publish a package using the same version number twice.
    - Make sure to use bump2version before building the new package version.
    - If you accidentally created a new package without changing the version:
        - Delete the contents of the "dist/" folder in your project repo.
        - Re-run the build commands and try uploading again

- Be carefully about requirements you add to your setup.py file.
    - Add version #'s to your requirements whenever possible.

LECTURE RECORDING

(Added 01.01.24)

https://youtu.be/8WJyabUoJdw