AST3100 "Code development and packaging mini-course" Assignment 3

Due on Mar. 17 2020 at the start of class

The purpose of this third assignment is to build a test suite for the Python package that you are creating as part of this course, run it with `pytest`, and set it up to run automatically on `Travis CI`. Write your tests in a `tests/` sub-directory of your top-level directory as discussed in the notes.

Task 1: Write tests for your code that check that in cases where you know the answer, your code returns the correct answer Do this for some of the functions and methods in your package. If you think that none of the functions in your package have a known solution, then you need to think hard (because there probably is one) or re-design your package such that it consists of functions for which some known solutions exist.

Task 2: Write tests for your code that check that the outputs that your functions and methods provide are consistent with known properties of the outputs Even if you don't know the answer, what property do you expect the output to have? This can be as simple as checking the expected range of the output, or that the output satisfies some constraints.

Task 3: Install `coverage.py` and run your tests with `coverage` and `pytest` to determine the test coverage of your test suite following the steps given in the notes.

Task 4: Setup your project on `Travis CI` and run the build-and-test integration automatically Write a `.travis.yml` file that installs all dependencies (with `pip` if you can), installs your code, and runs the test suite, displaying the test coverage.

Task 5: Use Miniconda on `Travis CI` to install dependencies and run again.

Task 6 (extra credit): Setup your project on `AppVeyor` and run the build-and-test integration automatically by writing a `.appveyor.yml` file and adding it to your repository.