

Brent Biseda

W200

Monday, 4 PM Pacific Time

Gerald Benoit

Project 1 Write Up

I was able to create an economic engine that evaluates natural gas and oil wells. The program requires a number of proprietary csv files that I request are kept confidential. The file may be run via commandline with the .py file and the csv files maintained in the /input/ folder. Alternatively, it may be run via the .ipynb. In both cases the files are output into the /output/ folder. For reference, I have included the outputs from a montecarlo simulation run with 100 iterations.

The DataLoader class accepts the following information then is passed to a field object which drills the wells.

- Type Curve information
- Operating cost file
- List of wells to be drilled (drill schedule)
- Commodity pricing information

After the data is loaded, the economics, visualizations, and reporting classes are able to produce outputs.

The project class has information hard coded to simulate the performance of gas wells based on historical information regarding the lognormal distribution parameters for Utica & Marcellus wells. This part of the code is something that I will be using at work & modifying to create specific montecarlo simulations.

The output csv will be able to be utilized in our marketing meetings.

Thoughts & Challenges:

The major challenge was creating the definition of many of the classes in such a way that there is at least some flexibility in the information that is able to be passed in from a csv. Midway through this project I refactored my code to implement a dictreader such that I could eliminate many specific variable names. This would allow the code to become more flexible in the future.

One challenge in this program is in defining areas to include try & except blocks. There are many possible ways that the code could break, particularly if the input files are not of the appropriate format. Therefore, many of the try and except blocks are around larger blocks of code to indicate where an error may have occurred. An example of this in particular is the try & except block around the montecarlo simulation method

The montecarlo method was my ultimate goal for this project. Because there are many different parameters to vary, I have left the code to be changed internally in the method as any simulation performed will be done on an adhoc basis and will involve many variables. For this reason, I did not dwell specifically on creating an interface to manage the variables in the montecarlo simulation.

With that said, I feel like this was a very challenging project and I spent a considerable amount of time on it. I have also been working on additional mapping visualizations which I have not included here because they require 3rd party libraries. Hopefully, I will have gotten far enough along to share this during the class presentation!

Regards,

Brent