Exploring the Parameter Space

Nick DeRobertis

March 3, 2021

1   Introduction to Parameter Exploration

- If you've merely done a calculation, you have a single answer and can't learn any more about the problem without redoing the calculations

- Instead because we are building models which can flexibly take any input and convert that into the appropriate output, it opens up a lot of possible extensions to the model

- This is our main focus for a large portion of the course: how can we extend any financial model to get a greater understanding of the underlying problem

- Sensitivity analysis is useful to understand how the full range of possible inputs affects the main results of your model

- Scenario analysis allows analyzing outcomes in example situations

- Monte Carlo Simulation allows assigning outputs to a probability distribution, which helps you understand the risk of your result

2   Introduction to Sensitivity Analysis

- The formal definition of sensitivity analysis may seem complicated, but all we are doing is running the model multiple times with different inputs and showing the outputs

- A key component of sensitivity analysis is the visualization, as now that there are many different outputs it is much easier to draw meaning from them when visualized

3   Sensitivity Analysis in Excel

- Data tables in Excel allow calculating a cell multiple times, changing some other cell. This is perfect for sensitivity analysis if we target an output cell and change an input cell

- One-way data tables change one input at a time, two-way data tables change two inputs at a time

- You are basically limited to changing two inputs at once, without doing some clever hacks

- Visualization rule of thumb: graph one-way data tables and use conditional formatting for two-way

- Conditional formatting changes the format of cells based on conditions, such as putting the largest numbers in green and the smallest in red

- Row input cell means that your inputs are going horizontally in a row. Column input cell means that your inputs going vertically in a column. For one-way data tables, you will use only one of the two. For two-way data tables, you will use both

4 Using Python Dictionaries

- There are three ways to loop through dictionaries: through the keys (the default), through the values (.values()), and through both at once (.items())

- If you want one way to do it, just always loop through the items as you have access to both the key and the value at once

- Combine dictionaries using update. Add items to dictionaries with brackets and assignment. Remove items from dictionaries with pop

5 Python List Comprehensions - Convenient List Building

- List comprehensions are an example of "syntactic sugar," or a feature of a programming language which is not necessary but makes things easier (makes the programming experience "sweeter")

- They allow us to write simple loops made to create lists with only a single line of code

6 Python Imports and Installing Packages

- Import can be used for third-party and built-in packages, but also for your own code offloaded into separate files

- I had you install Anaconda to get Python because it includes most of the Python packages we would want to use pre-installed. So we haven't had to install any package up to now

- With more than 250k packages out there and only about 200 installed in Anaconda, the time will come when you need to install something

- It will even happen in this course because we will use some packages I have created

7 Introduction to Sensitivity Analysis in Python

- With sensitivity analysis in Python, we can get DataFrames looking similar to the tables with conditional formatting we saw in Excel

- We can also output hexbin plots which show the same graphically

- Unlike Excel, we are not limited to analyzing two inputs at once, though it still makes sense to visualize the output in pairs of inputs

- The basic setup with pure Python is fairly simple: use loops over the possible input values and call your model function in the loop, saving the output associated with the inputs. Changing multiple inputs at once means multiple nested loops.

- Then you have to add the conditional formatting or hexbin plot

- The sensitivity package reduces this all to a couple lines of code, regardless of how many inputs you want to look at

8 Sensitivity Analysis in Python Example

- Here we will focus only on using the sensitivity package rather than carrying everything out manually

- We need to go back and add an optional argument to the model about whether it should print the results, otherwise we will have the results printed a huge number of times as we run the sensitivity analysis

- The sensitivity package is made to work with functions where each input is passed separately, whereas our model function takes a single dataclass instance. To make our model function work with the sensitivity package, we need to create a wrapper function which takes the separate arguments, creates the dataclass from them, passes that into the model function and returns the result. You can copy my snippet to do this with your model.

- List comprehensions are a nice easy way to specify values in a range, but you can also hard-code these lists

- Be careful not to look at too many input values as execution could be very slow. The progress bar will tell you how many cases of the model you are running and show how long it is taking.

- There are a number of options to customize the output from the library. You can change the names of the inputs and results, the color map, change direction of the colors, the grid size on the hexbin plots, and the number formatting in styled DataFrames.

9 Lab Exercise - Adding Sensitivity Analysis to Project 1 - Python

- The lab exercise here is the same as that we did for Excel but now just doing it in Python

- You are free to use the manual approach or the sensitivity package but I would recommend the sensitivity package

- Be sure to upgrade your sensitivity package before working on the exercise

- The lecture on adding sensitivity analysis to the Python Dynamic Salary Retirement model will be very helpful for this

- You may need to restructure your model before you can even begin adding sensitivity analysis. You must have a function which accepts the data and produces the final output from that data, and your functions must not access the overall model_data