Dynamic optimization project

due 12/13/11, by 12:30pm

Assignment: Invent, describe, and numerically solve a dynamic optimization problem. I expect some degree of originality and/or creativity in your choice of problem: no trivial extensions of the example I solved numerically in class or examples that your classmates have already completed.[1] That said, you may use an example from another class or from a book, as long as the numerical value function iteration you perform is 100% your work (that is, you would be able to explain each step without difficulty if asked, and no copy and pasting from an external source was used at any time).

You don't need any sort of complete model — you can take as exogenous anything you want, including normally endogenous variables like prices — just one agent's dynamic optimization problem. Make assumptions for functional forms and parameters, and explain your choices. I encourage you to come tell me about your idea before you begin working on it.

Your completed assignment should include:

1. A short, typed write-up including information on what the problem is, what the state and control variables are, what functional forms you assumed and why, what state space you are considering, and any other information you view as relevant. You can likely do this part in one page.

2. A graph of your value function over the state space you chose, derived using numerical methods.

3. A graph depicting the value function after, for example, 1 iteration, 5 iterations, 10 iterations, etc, that shows it converging to the true value function (you will choose the actual iterations to graph based on what is the most visually compelling).

4. A graph of your optimal policy rule(s) over the state space you chose.

5. At least one simple comparative statics exercise. For example, you can vary the discount factor and discuss how this affects the optimal policy rule(s).

6. Either compilable code which is heavily and clearly commented explaining what each step does, or an Excel file and a write-up describing each step you took in iterating the value function in Excel. The former is recommended, both because you will need to be able to write a program in the future, and because being able to show me your code will make it easier to meet the parameters of the assignment.

7. A footnote with a (one-sentence) acknowledgement of anyone and anything you consulted on this project (example: "In completing this project, I consulted Jeremy Sandford, classmate X, classmate Y, my dog Sparkles, and a bottle of Lagavulin, and I am solving example 4.3 from page 87 of the Adda and Cooper book, using code inspired by something I found on website Z.").

I recommend you do the numerical estimation in Matlab or another suitable programming language (Mathematica, Maple, R, Perl, Fortran, whatever). I recommend Matlab over other languages mainly because Tom Ahn[2] will require its use next fall in Eco 706; other languages will work just as well. If you have never used a programming language before, I can help you get started to some extent, but you will find

[1] I would consider a trivial extension to be doing a problem from class with different parameter values, or with a different reward function, unless there's an economic reason to suspect that changing these things will change the nature of the problem.

[2] Also known as "The Hammer".

numerous "getting started" tutorials online and you will find the return to simply messing around with the program is very high. To further incentivize you to learn how to do this using a programming language, I will give ten bonus points on top of your base score if you make a legitimate attempt to do this assignment using a programming language, instead of Excel.

If you do use Excel for the numerical portion, use Excel 2007 or later, which has 16,000 columns (earlier versions have 256). Use as fine of a grid over your state space as is reasonable, much finer than I used in class).

Grading: This assignment will comprise a "take-home" portion of your final exam, worth 37.5% of your final exam, and 15% of your course grade (so the in-class portion of the final exam will be worth 25% of your course grade). Your grade will be based on the following:

- **80%:** The clarity and quality of your write-up, including any code, and the apparent correctness of your results.
- **20%:** The degree of originality and difficulty attempted.[3]
- **10% bonus:** Completing the assignment using a programming language.
- **10% bonus:** An additional 10 percentage point bonus will be assigned to the most creative project.

[3]Don't go overboard in attempting something that is too difficult. Examples of good projects to attempt would be the monopoly problem from midterm 2, the stochastic cake-eating problem in Adda and Cooper section 3.2, one of the dynamic programming problems Paul Shea assigned in 702 last spring, the vintner problem from HW 5, or a project of your own invention of similar difficulty.