Monetary and Fiscal Policy
Econ 243

Prof. Harold Cole
517 Perelman Center
colehl@upenn.edu

Fall 2023
Date Tue-Thur 1:45-3:14
Room: MCNB 309

Modern economic policy analysis is almost entirely done using computers: (i) to solve elaborate quantitative models that capture the key elements thought to be important, (ii) to fit or calibrate these models to selected facts the data, and (iii) to simulate these models in order to derive predicted outcomes. Debates among policymakers largely occur by comparing the predictions of these models.

The Federal Reserve uses models to develop economic forecasts and projections of economic variables, such as inflation, unemployment rates, GDP growth, and interest rates, to predict future economic conditions. These forecasts aid in the formulation of monetary policy decisions. They also employ them to analyze the potential impacts of different monetary policy actions. By simulating the effects of changes in interest rates, money supply, and other policy variables, the Federal Reserve can assess how these actions might affect inflation, economic growth, employment, and other macroeconomic indicators. They also conduct stress tests of banks, and various risk and stability assessments within financial markets.

The Treasury uses models to analyze the potential effects of fiscal policy measures, such as government spending, labor or capital taxes, on the overall economy. The models simulate the interactions between various economic variables, such as investment, consumption, and employment. The models are used to assess the revenue implications of altering tax rates, tax brackets, deductions, and exemptions. The models are also used to analyze the distributional effects of fiscal policy measures and provide insights into the potential effects on income inequality. Finally they are used to assess the costs and benefits of fiscal policy initiatives.

This course aims to shed light on how all this is done. We develop versions of the models used in policy analysis (such as the New Keynesian and Ramsey capital/labor tax models), learn how to solve, calibrate and simulate them on a computer in order to derive predictions about possible outcomes and evaluate their welfare implications. To do this, the course will cover both both the standard analytic methods (like Lagrangians) and the computational methods commonly used in economics.

The course uses the open source computer language Python which is being used extensively in scientific computing and machine learning. Python's simple syntax, extensive libraries, and strong community support makes it an easy-to-use programming language. As a result, the step from a logic flow chart to actual code is surprisingly small. In the course we will teach the computational methods starting from scratch but I am assuming that have the standard background in calculus, probability theory, and statistics as per the requirements for the course. However, to aid you we will have an extensive math review.

Textbook: The main text is Cole, Harold L. *Monetary and Fiscal Policy Through a DSGE Lens*. Oxford University Press, 2020. The textbook should be in the bookstore. It can also be ordered on Amazon, and is available in a Kindle version.

Additional Material: There is some additional material that I will post on Canvas.

1. I have added some new material since the book was published, and I will make a couple of revised chapters available there. The new material includes a brief overview to machine learning which explains how it is used in scientific (numerical) computing.

2. I will also post a range of coding examples to help you get up to speed.

The material you obtain in class is not to be shared with those outside of the class.

Homework: There will be regular homework assignments, and these will play an important part in the course. We are trying to understand quantitative macroeconomic models. For that reason you will be asked to program up various versions of these models and compute the results. Normally, homework assignments will be due the following week and should be uploaded through Canvas.

Grading: There will be 2 midterms and a final exam. The final will be in-class and on the last day of classes. The weight will be 1/4 on the homework, 1/4 on each midterm and 1/4 on the final. I will handout supplemental problems for you to work on to aid you in preparing for the exams. The exams will draw on the lectures and the supplemental readings. *If someone misses one of the exams, I will base their grade on the max of their two exams. To be fair to everyone, I will also do the same for the class.*

Preliminary dates for the two Midterms are (i) Thursday Sept. 28, and (ii) Thursday Nov. 9.

Office Hours: My Zoom office hours are tentatively scheduled for Mondays @ 2-3 pm. But this will be subject to change during the semester. Please check the announcements on Canvas. I can also schedule regular office hours Tuesday-Thursday for those who need them.

T.A.: Artemii Korolkov is our TA and he is planning on holding office hours Fridays 9:30-11:00.

Tentative Topics List

1. Math review and Python programming introduction
2. DSGE model of money and production - deterministic version
    - Introducing the basic model - chapters 2-3
    - Extensions to the model - chapter 4
3. Stochastic Model
    - Money model - chapters 5-6
4. New Keynesian and Liquidity Models
    - New Keynesian model - chapters 10-11
    - Liquidity model - chapter 13
5. Adding Capital and Fiscal Policy
    - Basic model with capital chapter 14
    - Adding in Taxes chapters 15-17
    - Adding government expenditure chapter 18
    - Dynamic version with capital chapter 19
6. Introduction to Machine Learning in the RBC model
7. Special Section time permitting - study the impact of either
    (a) global warming or
    (b) declining populations.

Preliminary List of Supplemental Readings:

1. G. McCandless and W. Weber, "Some Monetary Facts," Federal Reserve Bank of Minneapolis Quarterly Review.
2. Atkeson and Kehoe, "Depression and Deflation: Is there an empirical link?," Federal Reserve Bank of Minneapolis Staff Report.
3. Cole and Kocherlakota, "Why Zero Interest Rates are Good and How to Get Them," Federal Reserve Bank of Minneapolis, Quarterly Review, Spring 1998.
4. R. King, "The Phillips Curve and U.S. Macroeconomic Policy: Snapshots, 1958-1996," Economic Quarterly of the Federal Reserve Bank of Richmond, Fall 2008.

5. Atkeson, Chari and Kehoe, "Taxing Capital Income: A Bad Idea," Federal Reserve Bank of Minneapolis Quarterly Review Summer 1999.

6. E. Prescott, "Why Do Americans Work So Much More Than Europeans?," Federal Reserve Bank of Minneapolis Staff Report 321.

7. Hassler, John, Per Krusell, and Jonas Nycander. "Climate policy." Economic Policy 31.87 (2016): 503-558.

Background Reading: You may find it helpful to consult a few sources for material on optimization, probability theory and computing. The obvious source are your prior textbooks. Some additional suggestions are:

1. A.K. Dixit, Optimization in Economic Theory, chapters 1-5.

2. `https://jeremykun.com/2013/01/04/probability-theory-a-primer/`

Computing: Learning to program is a critical aspect of the course. We will use Python in the class because: (i) it is open source and hence free, (ii) it has a very simple structure aimed at making programming easier (iii) it has become one of the main languages used in data science, (iv) there is a rich array of explanatory and introductory material to aid you in learning how to program. For example, if you type a question in google "Python how do I ... " you will generally find a number of clear answers.

Here is what you need to do to get started:

1. Download and install Anaconda with Python 3.x. Get the individual edition which is free:

2. `https://www.anaconda.com/products/individual`

   To see how to install Anaconda and Python view the first 5:30 minutes of this video and then install Anaconda and Python. Make sure to get the highest 3.x version (Corey Shafer Python Tutorial: Anaconda - Installation and Using Conda)

   `https://www.youtube.com/watch?v=YJC6ldI3hWk`

3. It might be wise to make sure that everything is up-to-date after the install. "conda update –all" . See `https://www.anaconda.com/blog/keeping-anaconda-date`

4. We are going to be using *Jupyter notebooks* to program your code. To understand them,

   (a) watch the following video (Corey Shafer Jupyter Notebook Tutorial: Introduction, Setup, and Walkthrough)

   `https://www.youtube.com/watch?v=HW29067qVWk`

   (b) Download Trapezoid file and run it to make sure that everything works. Also, it is a great example of sample code which you can build off of.

   (c) Check out Python_first_steps.ipynb which you should be able to download and run.

Extra Material on Python: There are a wide range of introductory videos for Python programming. Most of what you need will be covered in class but here are some supplemental materials.

1. If you know very little about computer programming, watch the first 8 minutes of this video: Learn Java Programming with Beginners Tutorial

   `https://youtu.be/uWYPVz_i7W4`

2. For a gentle introduction to Python, watch the first 3 hours and 30 minutes of the following video (Giraffe Academy Learn Python - Full Course for Beginners) This is what I used to learn it.

   `https://www.youtube.com/watch?v=rfscVS0vtbw`

3. If we end up using Pandas here is something on this:

   "Python Pandas Tutorial (Part 1): Getting Started with Data Analysis - Installation and Loading Data" by Corey Schafer.

   Ignore installation material since it is already installed via Anaconda.

4. Here is a great cheat sheet which compares Matlab, Python and Julia

   `https://cheatsheets.quantecon.org`

Using AI: ChatGPT and other AI tools offer the possibility of writing code for you. *Anyone who comes up with a good example of using ChatGPT can earn an extra point on their homework. You can only do this twice.*

- In my experience AI can be helpful but may be "buggy"; however potentially great.
- Here is one source on this: `https://blog.enterprisedna.co/how-to-use-chat-gpt/`
- And here is another: `https://www.zdnet.com/article/how-to-use-chatgpt-to-write-code/`