Computational Methods in Economics

ECON 21410

Instructor: Jeremy Bejarano

Introductions

- My name is Jeremy Bejarano
  - 5th year PhD student in economics
  - Macroeconomics, Finance, Asset Pricing
  - Enjoy mathematics, scientific computing, data, Python (PyData and Jupyter), HPC

About this class

- **Course Description:** This course introduces the basic programming and computational techniques necessary for solving and estimating economic models. The course covers topics in numerical methods, such as optimization, function approximation, and Monte Carlo techniques, as well as topics in data exploration, visualization, and estimation. Emphasis will be placed on developing effective programming and research practices. The course is structured through a series of applications in such topics as macroeconomic growth, real business cycles, and asset pricing. The course will be taught primarily in Python. Though helpful, no previous experience with computer programming is necessary for this class.

About this class

- Confusion with course description
  - Interest in Python?
  - Interest in R?
  - Stata?
  - Macroeconomics or Finance?
- Examples from other fields
  - Labor economics, Urban economics, Industrial organization

About this class

- ECON 21410: Computational Methods in Economics
- Prerequisites:
  - ECON 20100 (The Elements of Economic Analysis II)
  - ECON 21020 or ECON 21030 (Econometrics or Econometrics Honors)
- Recommended:
  - ECON 20200 (The Elements of Economic Analysis III – covers macro topics)

About this class

- Navigate to course GitHub page:
  - https://github.com/jmbejara/comp-econ-sp18
- Explain how
  1. course material, including schedule, is on GitHub,
  2. assignments will be turned in via GitHub,
  3. and grades will be posted on Canvas.
- Go over syllabus:
  - https://github.com/jmbejara/comp-econ-sp18/blob/master/Syllabus-Computational-Economics.pdf

Motivation

- Quotes from Gentskow and Shapiro
- Personal Demos
- Example class projects/assignments
- QuantEcon project. What is it?
- QuantEcon: About Python
  - https://lectures.quantecon.org/py/about_py.html

Motivation

- Gentzkow and Shapiro, “Code and Data for the Social Sciences: A Practitioner’s Guide”
- What does it mean to do empirical social science? Asking good questions. Digging up novel data. Designing statistical analysis. Writing up results. For many of us, most of the time, what it means is writing and debugging code. We write code to clean data, to transform data, to scrape data, and to merge data. We write code to execute statistical analyses, to simulate models, to format results, to produce plots. We stare at, puzzle over, fight with, and curse at code that isn’t working the way we expect it to. We dig through old code trying to figure out what we were thinking when we wrote it, or why we’re getting a different result from the one we got the week before.

Motivation

- Though we all write code for a living, few of the economists, political scientists, psychologists, sociologists, or other empirical researchers we know have any formal training in computer science. Most of them picked up the basics of programming without much effort, and have never given it much thought since. Saying they should spend more time thinking about the way they write code would be like telling a novelist that she should spend more time thinking about how best to use Microsoft Word. Sure, there are people who take whole courses in how to change fonts or do mail merge, but anyone moderately clever just opens the thing up and figures out how it works along the way.
- *This manual began with a growing sense that our own version of this self-taught seat-of-the-pants approach to computing was hitting its limits.*

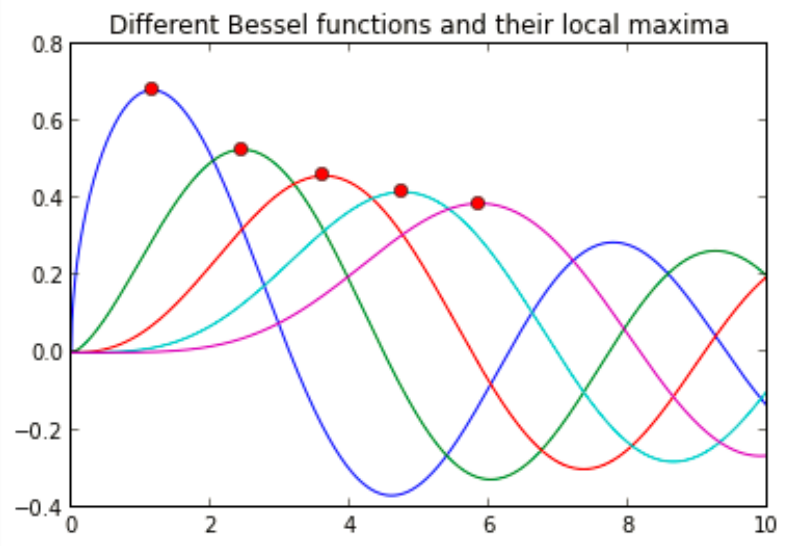
Motivation

In [1]:
```
from scipy import optimize, special
from numpy import *
from pylab import *

x = arange(0,10,0.01)

for k in arange(0.5,5.5):
     y = special.jv(k,x)
     plot(x,y)
     f = lambda x: -special.jv(k,x)
     x_max = optimize.fminbound(f,0,6)
     plot([x_max], [special.jv(k,x_max)],'ro')

title('Different Bessel functions and their local maxima')
show()
```



Motivation

- Pandas for Data Analysis
- Visualizations
  - https://seaborn.pydata.org/examples/index.html
  - http://nbviewer.jupyter.org/github/plotly/python-user-guide/blob/master/s3_bubble-charts/s3_bubble-charts.ipynb

Motivation

- Interactive visualization are easy to make and are great for personal portfolios.

Motivation

- Applications to topics in economics and finance
  - Schelling's Segregation Model
  - Portfolio Optimization
  - Estimation of Productivity
  - Business Cycle Accounting
  - Models of Entry and Exit
  - Asset Pricing

About You

- Who has had experience with computer programming before?
- Python? R? Stata? Anything else?
- Git or LaTeX?
- Who is interesting in pursuing a career related to economics? Data science? Software development?
- Who is interested in graduate school? In economics or other fields?

Preliminary Exam

- We will take a preliminary exam to get an idea of your experience with Python.
- We will be holding a Python crash course on Saturday.
  - If you get 100%, you can skip this crash course and the associated portion of assignment 0.
- Take Preliminary Exam.

An Introductory Example

- Introduce Google's Colaboratory
- If time permits:
  - "QuantEcon: "An Introductory Example"