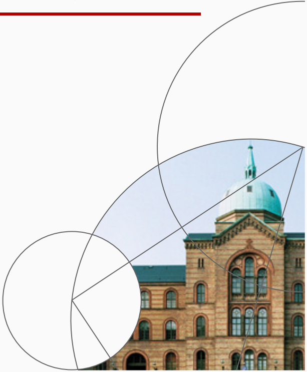
Center for
Economic
Behavior &
Inequality

13. Outroduction

Introduction to Programming and Numerical Analysis

Jeppe Druedahl

Spring 2019



Plan

1. Model projects

2. Exam project

3. Questions

Model projects

Broad range of topics

- **Take-away:** The tools you are learning in this course are very broadly applicable ⇒ use in other courses and for bachelor and master theses:
  1. Labor supply models with progressive taxation
  2. Estimation of IMDb ratings
  3. Cournot and Betrand competition models
  4. Solow models with human capital, land or climate externalities
  5. AS-AD models
  6. Ramsey models
  7. Overlapping generation (OLG) models
  8. CGE models
  9. RBC models
  10. Koopman models
  11. Portfolio theory
  12. Fertility

The holy trinity

- **The best projects** contain a combination of:
  1. Algebraic manipulation with *sympy*
  2. Numerical optimization with *scipy.optimize*
  3. Stochastic simulation with *numpy.random*
- **Some projects:** Too much focus on *sympy*.
- **Next year:** A combination of all three elements will be required.
- **This year:** They are all relevant for the exam.

Structure and commenting

- **Structure and commenting:** Still room for improvement...
    1. Lecture 5
    2. Examples repo
- **Some central stuff:**
    1. Use functions more
    2. Use modules more
    3. Order your comments
    4. Show and test intermediate results
    5. Try to simplify once you found *a* solution
- **»Programming is more than writing code«**
    1. In real life for **safety** (insurance against bugs)
    2. In science also for **replicability**
- **Two different takes:**
    1. The comments explain humans what the code does.
    2. The code makes the computer do what the comments say.

Exam project

Problems

- **Structure:** 3 problems with 3-6 sub-questions on solving and simulating models and analyzing their implications graphically and numerically.
- **Examples of a problems:**
  1. Solve consumer or firm problems (with non-standard constraints)
  2. Solve and simulate an AS-AD model
  3. Solve for the Walrus-equilibrium in an exchange economy
  4. Solve an extended Solow model
  5. Solve a two period dynamic optimization problem

  $\Rightarrow$ similar to the problems in the problem sets
- **Curriculum:** Lecture notebooks (÷ sections marked with *)
- **Packages:** No new packages are required, and using non-standard packages are actively discouraged.

Answering

1. **Focus on answering the questions** - nothing more, nothing less
2. Explain your **method in words** (or with an algorithm)
3. **Structure and comment your code!**
4. Explain your **results in words**
5. **Partial answers, attempts and considerations** are also **awarded** (something on everything is better than a lot on a few questions)

Disclaimer: Solving the full exam project in depth will be hard.

Hand-in

- **You should hand-in a single zip-file named with your groupname only.**
- The zip-file should contain:
  1. A general README.md for your portfolio
  2. A Feedback.txt file with a list of the groups each group member have given peer feedback to with links to the GitHub issues
  3. Your data analysis project (in the folder /dataproject)
  4. Your model analysis project (in the folder /modelproject)
  5. Your exam project (in the folder /examproject)

Questions

Questions

- **Any questions now?**
- **Online:**
  https://github.com/NumEconCopenhagen/lectures-2019/issues