Computing in Economics

Keith Finlay

Department of Economics
Tulane University

10/26/2014

Outline

1. Economics
2. Computational applications
3. Search
4. Optimization
5. Simulation
6. Conclusion

Economics

- Economics is the study of human choices given scarce resources
- There is plenty of overlap with the other social sciences
- What distinguishes economics is an axiomatic mathematical framework for examining human behavior
- (On average, I would argue that) our empirical work is more rigorous than that of the other social scientists
- Our use of computation reflects the theory and the empirical rigor

Outline

1 Economics

2 Computational applications

3 Search

4 Optimization

5 Simulation

6 Conclusion

An outline of computational applications in Economics

- Common to all fields
  - Collecting data about the world
  - Searching databases of prior literature
  - Document preparation
- Tools used in theoretical economics
  - Finding analytic solutions to symbolic math problems
  - Numerical analysis of symbolic math problems without analytic solutions
- Tools used in empirical economics
  - The whole set of tools used in statistics
  - The most common tasks are linear algebra and optimization
- Both use numerical simulation, Monte Carlo simulation, linear/dynamic programming, Taylor series approximation, derivative computation, integral approximation, solving differential equations
- Some specialized tools for specific subfields: graph theory to model networks

Outline

1. Economics
2. Computational applications
3. Search
4. Optimization
5. Simulation
6. Conclusion

Search

- Learning how to search for information is about
  - Knowing where to look for specialized information
  - Using the best tools in the most effective ways
  - Judging the quality of data source
- Right now is a great time to be searching!
  - When I was in college, literature reviews were done in the library using a database called EconLit
  - Now, Google Scholar is essentially the only tool necessary: `http://scholar.google.com`
- It's also a great time to collect data for research projects

Search

Where would you find these data?

- Unemployment rates for every state, for every month, since 1980 until recently?
- Price data for illegal drugs
- State laws related to implementation of the Affordable Care Act

Search

Learn how to use Google more effectively

- If you want to learn how to find the hard-to-find, you need to learn how your search tools are built
- Google still has the best index of web content, but they've changed how that data is presented to you
  - Internet search databases are populated by crawling through all linked content on the web
  - The algorithms are called spiders and what they do is called crawling
  - In general, Google highlights information that has a high PageRank, which is a metric of how much other content points at a particular page
  - Over the years, Google has started customizing content on its search page based on what it knows about you

Search

Learn how to use Google more effectively

- Here are some techniques for controlling the information Google presents
- `filetype:pdf` searches only for content in pdf format
- `"search terms"` forces search terms to appear in document text and in that order
- `site:tulane.edu` searches only on a given domain or base URL
- I use these the most, but also `intitle:` to search in title, `inurl:` to search in URL, `-search` to exclude search terms, and `cache:` to search in page caches
- You can find more examples here: `https://support.google.com/websearch/answer/136861?p=adv_operators`

Search

Search the web's past

- Use the Internet Archive's Wayback Machine to search through cached versions of websites: `https://archive.org/web`
- I've used this in my work to document how policies have changed over time

Search

Web scraping

- You can make your own datasets using information on the web
- Suppose you wanted to collect firm data from Facebook company pages
- On the next slide is example code form Python

Search

Web scraping

```
import urllib2
import json
list_companies = ["walmart", "cisco", "pepsi", "facebook"]
graph_url = "http://graph.facebook.com/"
for company in list_companies:
    #make graph api url with company username
    current_page = graph_url + company

    #open public page in facebook graph api
    web_response = urllib2.urlopen(current_page)
    readable_page = web_response.read()
    json_fbpage = json.loads(readable_page)

    #print page data to console
    print company + " page"
    print json_fbpage["id"]
    print json_fbpage["likes"]
    print json_fbpage["talking_about_count"]
    print json_fbpage["username"]
```

Outline

1 Economics

2 Computational applications

3 Search

4 Optimization

5 Simulation

6 Conclusion

Mathematical optimization

- Mathematical optimization is the selection of a best element (with regard to some criteria) from some set of available alternatives
  - In the most common applications, we maximize or minimize some real-valued function
- There are two primary applications of optimization in economics
  - In economic theory, we model the behavior of individuals as maximizing utility and the behavior of firms as maximizing profit
  - In statistics and econometrics, we minimize some function of what our statistical models get wrong or maximize the likelihood that we observe the data we have given estimated model parameters

Optimization in economic theory

Labor supply example

Suppose we want to model the labor supply decision of an individual:

- Variables: consumption C, leisure L, wage w, time available for work T
- Utility given by $U(C, L) = C^{\frac{1}{2}} L^{\frac{1}{2}}$
- Constraint connects labor earnings to consumption: $C \leq w(T - L)$
- Constrained optimization problem: maximize $U(C, L)$ such that $C \leq w(T - L)$
- In general, we assume that individuals make choices at the frontiers of their constraints:
  $\max \ C^{\frac{1}{2}} L^{\frac{1}{2}}$ s.t. $C = w(T - L) \Rightarrow \max \ (w(T - L))^{\frac{1}{2}} L^{\frac{1}{2}}$
- There is a general solution to this problem, but we can visualize the optimization problem easily if we assume $T = 16$ hours and $w = \$10$: `http://www.wolframalpha.com/input/?i=maximize+%2810%2816-L%29%29%5E%28.5%29L%5E%28.5%29`

Optimization in econometrics

Ordinary least squares

- Linear regression is one of the most commonly used statistical techniques
- The OLS population model is

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k + \varepsilon = x\beta + \varepsilon$$

- Ordinary Least Squares (OLS) refers to choice of β by minimizing the sum of the squared residuals:

$$\begin{aligned} \arg\min_{\widehat{\beta}} \varepsilon'\varepsilon &= (y - x\beta)'(y - x\beta) \\ &= y'y - \beta'x'y - y'x\beta + \beta'x'x\beta \\ &= y'y - 2y'x\beta + \beta'x'x\beta \end{aligned}$$

Optimization in econometrics

Ordinary least squares

- Taking the derivative of the sum of squared residuals with respect to the parameter vector β, gives us a vector of first order conditions we call the least squares normal equations:

$$\Rightarrow \frac{\partial}{\partial \beta} = -2x'y + 2x'x\beta = 0$$

- The solution to our optimization problem is the OLS equation:

$$\widehat{\beta} = (x'x)^{-1}x'y$$

Optimization in econometrics

Maximum likelihood estimation

- Outcome, y, depends on x, θ (e.g., $\theta = \{\beta, \sigma\}$)
- Estimation chooses $\widehat{\theta}_{ML}$ to maximize probability of the realized data $\{y_i, x_i\}_{i=1}^N$
- Probability of observing the realized data conditional on θ depends on the *assumptions* concerning the underlying data-generating process
- The likelihood function, $\mathcal{L}(\theta)$, gives the total probability of observing the realized data as a function of θ
- The pdf of a random variable, y, is $f(y|\theta)$, where θ captures parameters of the distribution
- Given a sample of size N, the joint distribution is $f(y_1, ..., y_N|\theta)$

Optimization in econometrics

Maximum likelihood estimation

- Assuming independence between observations, the joint distribution is the product of the marginal distributions

$$f(y_1, ..., y_N|\theta) = \prod_{i=1}^{N} f(y_i|\theta)$$

- This joint density is the *likelihood function*

$$\mathcal{L}(\theta|y) = \prod_{i=1}^{N} f(y_i|\theta)$$

- It is usually easier to work with the *log-likelihood function*

$$\begin{aligned} \mathcal{L}(\theta|y) &= \prod_{i=1}^{N} f(y_i|\theta) \\ \Rightarrow \ln\left[\mathcal{L}(\theta|y)\right] &= \sum_{i=1}^{N} \ln[f(y_i|\theta)] \end{aligned}$$

  which is just a monotonic transformation

Optimization in econometrics

Maximum likelihood estimation

- ML estimates of θ, $\widehat{\theta}_{ML}$, are obtained by maximizing $\ln\left[\mathcal{L}(\theta|y)\right]$

$$\widehat{\theta}_{ML} = \arg\max_{\theta} \ln\left[\mathcal{L}(\theta|y)\right]$$

  which entails solving the *likelihood equation*

$$\frac{\partial \ln\left[\mathcal{L}(\theta|y)\right]}{\partial\theta} = 0$$

- This is an example of a problem that does not have a closed-form or analytic solution
- We have to use some optimization algorithm to find the maximum likelihood

Optimization in econometrics

Maximum likelihood estimation

- The most famous method for finding where a function equals zero is Newton's method (or the Newton-Raphson method)
- We're looking for the point x_* where $f(x_*) = 0$
- Given some starting point, a better guess would be $x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$
- The process is repeated in iteration $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$ until a sufficiently accurate value is reached
- Here's a nice animation of the process: `https://upload.wikimedia.org/wikipedia/commons/e/e0/NewtonIteration_Ani.gif`

Outline

1 Economics

2 Computational applications

3 Search

4 Optimization

5 Simulation

6 Conclusion

Uses of simulation in economics

- Simulation is the imitation of the operation of a process or system over time
- Some uses of simulation in economics:
  - Numerical analysis with models that don't have closed-form solutions
  - Monte Carlo simulation to evaluate the performance of statistical techniques
  - Bootstrap simulation (resampling with replacement) to improve the performance of statistical methods

Numerical analysis

- Consider the labor supply example from before
- In some theoretical economic models, there is no single equation that demonstrates the effects of changes of all the model's variables on the economic behavior or outcome of interest
- We can simulate different possibilities by choosing different possible values for the variables in the model
- Then, we solve as many of the equations as possible to evaluate a more parsimonious version of the model
- This is precisely what we did with the labor supply example

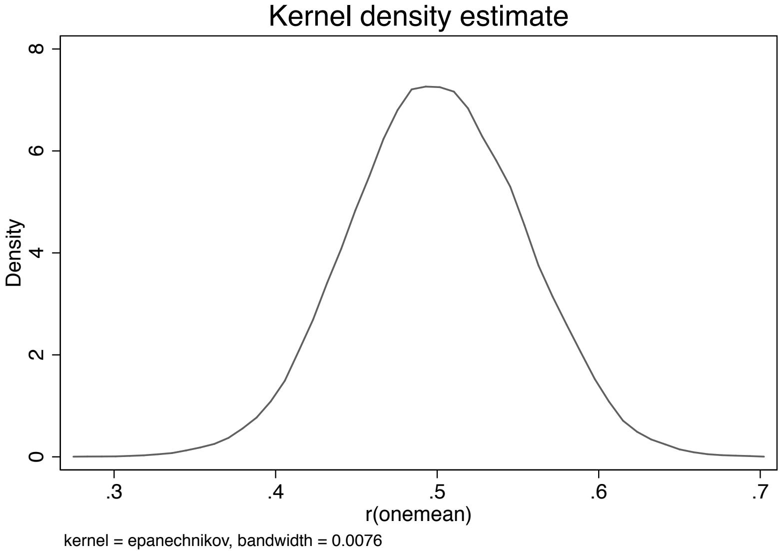
Monte Carlo simulation

- Monte Carlo experiments are a class of computational algorithms that rely on repeated random sampling to obtain numerical results
- We generally use MC simulations many times over in order to obtain the distribution of an unknown probabilistic entity
- Here's a Stata example for simulating the sampling distribution of means from 30-unit samples:

```
program onesample, rclass
    clear
    set obs 30
    generate x = runiform()
    summarize x
    return scalar onemean = r(mean)
end
simulate xbar = r(onemean), reps(10000) : onesample
```

Monte Carlo simulation

Sampling distribution of means



Bootstrap sampling

- In statistics, bootstrapping can refer to any test or metric that relies on random sampling with replacement
- Bootstrapping allows us to assign measures of accuracy to sample estimates
- It can be used to estimate sampling distributions of almost any statistic using very simple methods
- In many statistical applications, we rely on the Central Limit Theorem for asymptotic properties of estimators
- But CLT requires large samples or relatively symmetric population distributions
- We can use resampling with small samples or skewed population distributions

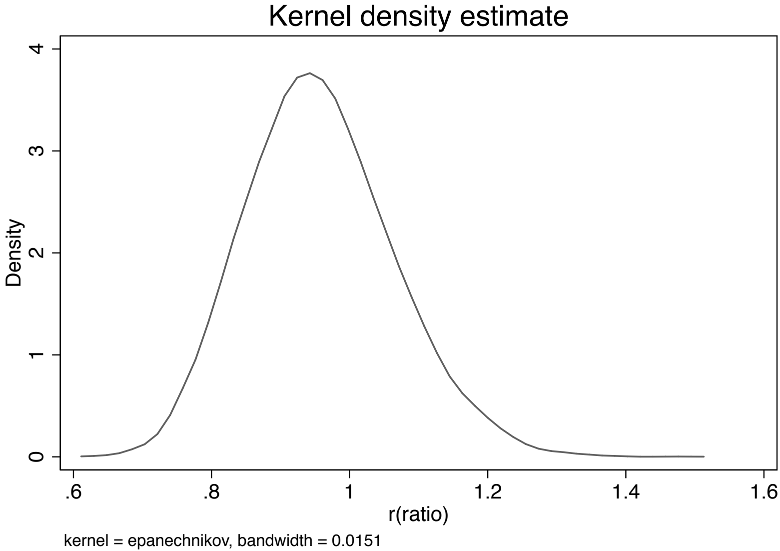
Bootstrap sampling

- Here's a Stata example for bootstrapping the ratio of the average price of a domestic car vs. the average price of a foreign car:

```
program define muratio, rclass
    syntax varlist(min=2 max=2)
    tempname ymu
    summarize `1', meanonly
    scalar `ymu' = r(mean)
    summarize `2', meanonly
    return scalar ratio = `ymu'/r(mean)
end
sysuse auto
gen p_dom=price if foreign==0
gen p_for=price if foreign==1
bootstrap r(ratio), reps(10000) : muratio p_dom p_for
```

Bootstrap sampling

Sampling distribution of ratio



Outline

1 Economics

2 Computational applications

3 Search

4 Optimization

5 Simulation

6 Conclusion

Practical advice

- Get a BS in Economics and Math
- Learn Python (a very useful, popular, and well-supported scripting language)
- Learn R (the premier open-source statistical computing language)
- Learn GIS (graphical information systems)