Estimating hybrid choice models with the new version of Biogeme

Michel Bierlaire

transp-or.epfl.ch

Transport and Mobility Laboratory, EPFL
Motivation

- Standard random utility assumptions are usually violated
- Factors such as attitudes, perceptions, knowledge are not reflected
Example: perceptions

Exemple: subscription to *The Economist*

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Web only</td>
<td>@ $59</td>
</tr>
<tr>
<td>Print only</td>
<td>@ $125</td>
</tr>
<tr>
<td>Print and web</td>
<td>@ $125</td>
</tr>
</tbody>
</table>
Example: perceptions

Exemple: subscription to *The Economist*

<table>
<thead>
<tr>
<th>Experiment 1</th>
<th>Experiment 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web only @ $59</td>
<td>Web only @ $59</td>
</tr>
<tr>
<td>Print only @ $125</td>
<td>Print and web @ $125</td>
</tr>
<tr>
<td>Print and web @ $125</td>
<td>Print and web @ $125</td>
</tr>
</tbody>
</table>
Example: perceptions

Exemple: subscription to *The Economist*

<table>
<thead>
<tr>
<th></th>
<th>Experiment 1</th>
<th>Experiment 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>Web only @ $59</td>
<td>Web only @ $59</td>
</tr>
<tr>
<td>0</td>
<td>Print only @ $125</td>
<td></td>
</tr>
<tr>
<td>84</td>
<td>Print and web @ $125</td>
<td>Print and web @ $125</td>
</tr>
<tr>
<td></td>
<td>68</td>
<td>32</td>
</tr>
</tbody>
</table>

Source: Ariely (2008)

- Dominated alternative
- According to utility maximization, should not affect the choice
- But it affects the perception, which affects the choice.
### Example: perceptions

Population of 600 is threatened by a disease. Two alternative treatments to combat the disease have been proposed.

<table>
<thead>
<tr>
<th>Experiment 1</th>
<th>Experiment 2</th>
</tr>
</thead>
<tbody>
<tr>
<td># resp. = 152</td>
<td># resp. = 155</td>
</tr>
<tr>
<td>Treatment A:</td>
<td>Treatment C:</td>
</tr>
<tr>
<td>200 people saved</td>
<td>400 people die</td>
</tr>
<tr>
<td>Treatment B:</td>
<td>Treatment D:</td>
</tr>
<tr>
<td>600 people saved with prob. 1/3</td>
<td>0 people die with prob. 1/3</td>
</tr>
<tr>
<td>0 people saved with prob. 2/3</td>
<td>600 people die with prob. 2/3</td>
</tr>
</tbody>
</table>
# Example: perceptions

Population of 600 is threatened by a disease. Two alternative treatments to combat the disease have been proposed.

<table>
<thead>
<tr>
<th></th>
<th>Experiment 1</th>
<th>Experiment 2</th>
</tr>
</thead>
<tbody>
<tr>
<td># resp.</td>
<td>152</td>
<td>155</td>
</tr>
<tr>
<td><strong>Treatment A:</strong></td>
<td>72%</td>
<td>22%</td>
</tr>
<tr>
<td></td>
<td>200 people saved</td>
<td>400 people die</td>
</tr>
<tr>
<td><strong>Treatment B:</strong></td>
<td>28%</td>
<td>78%</td>
</tr>
<tr>
<td></td>
<td>600 people saved with</td>
<td>0 people die with prob. 1/3</td>
</tr>
<tr>
<td></td>
<td>prob. 1/3</td>
<td>0 people saved with</td>
</tr>
<tr>
<td></td>
<td></td>
<td>prob. 2/3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>600 people die with</td>
</tr>
<tr>
<td></td>
<td></td>
<td>prob. 2/3</td>
</tr>
</tbody>
</table>

Source: Tversky & Kahneman (1986)
Latent concepts

- **latent**: potentially existing but not presently evident or realized (from old French: hidden)
- Here: not directly observed
- Standard models are already based on a latent concept: utility

Drawing convention:

- Latent variable
- Observed variable
- structural relation: →
- measurement: →
- errors: →
Random utility

\[ V_{in} = \sum_k \beta_{ik} x_{ikn} \]

\[ P_n(i) = e^{V_{in}} / \sum_j e^{V_{jn}} \]
Attitudes

- Psychometric indicators
- Example: attitude towards the environment.
- For each question, response on a scale: strongly agree, agree, neutral, disagree, strongly disagree, no idea.
  - The price of oil should be increased to reduce congestion and pollution
  - More public transportation is necessary, even if it means additional taxes
  - Ecology is a threat to minorities and small companies.
  - People and employment are more important than the environment.
  - I feel concerned by the global warming.
  - Decisions must be taken to reduce the greenhouse gas emission.
Indicators cannot be used as explanatory variables. Mainly two reasons:

1. Measurement errors
   - Scale is arbitrary and discrete
   - People may overreact
   - Justification bias may produce exaggerated responses

2. No forecasting possibility
   - No way to predict the indicators in the future
Factor analysis

\[ I_i = \lambda_i + \sum_k L_{ik} X_k^* \]

Latent variables \( X_k^* \)

Indicators
Measurement equation

Explanatory variables

\[ \varepsilon_i \]

Latent variables \( X^* \)

\[ X^*_k = \sum_j \beta_j x_j \]

Indicators

\[ I_i = \lambda_i + \sum_k L_{ik} X^*_k \]
Measurement equation

Continuous model: regression

\[ I = f(X^*; \beta) + \varepsilon \]

Discrete model: thresholds

\[ I = \begin{cases} 
1 & \text{if} \quad -\infty < X^* \leq \tau_1 \\
2 & \text{if} \quad \tau_1 < X^* \leq \tau_2 \\
3 & \text{if} \quad \tau_2 < X^* \leq \tau_3 \\
4 & \text{if} \quad \tau_3 < X^* \leq \tau_4 \\
5 & \text{if} \quad \tau_4 < X^* \leq +\infty 
\end{cases} \]
Choice model

Explanatory variables

Utility

Latent variables

Choice

Indicators

$\varepsilon_{in}$

$\omega_{in}$
Estimation: likelihood

Structural equations:

1. Distribution of the latent variables:

\[ f_1(X^*_n|X_n; \lambda, \Sigma_\omega) \]

For instance

\[ X^*_n = h(X_n; \lambda) + \omega_n, \quad \omega_n \sim N(0, \Sigma_\omega). \]

2. Distribution of the utilities:

\[ f_2(U_n|X_n, X^*_n; \beta, \Sigma_\varepsilon) \]

For instance

\[ U_n = V(X_n, X^*_n; \beta) + \varepsilon_n, \quad \varepsilon_n \sim N(0, \Sigma_\omega). \]
Estimation: likelihood

Measurement equations:

1. Distribution of the indicators:

\[ f_3(I_n|X_n, X_n^*; \alpha, \Sigma_\nu) \]

For instance:

\[ I_n = m(X_n, X_n^*; \alpha) + \nu_n, \quad \nu_n \sim N(0, \Sigma_\nu). \]

2. Distribution of the observed choice:

\[ P(y_{in} = 1) = \Pr(U_{in} \geq U_{jn}, \forall j). \]
Indicators: continuous output

\[ f_3(I_n|X_n, X_n^*; \alpha, \Sigma_\nu) \]

For instance:

\[ I_n = m(X_n, X_n^*; \alpha) + \nu_n, \quad \nu_n \sim N(0, \sigma^2_{\nu_n}) \]

So,

\[ f_3(I_n|\cdot) = \frac{1}{\sigma_{\nu_n} \sqrt{2\pi}} \exp \left( -\frac{(I_n - m(\cdot))^2}{2\sigma^2_{\nu_n}} \right) \]

Define

\[ Z = \frac{I_n - m(\cdot)}{\sigma_{\nu_n}} \sim N(0, 1), \quad \phi(Z) = \frac{1}{\sqrt{2\pi}} e^{-Z^2/2} \]

and

\[ f_3(I_n|\cdot) = \frac{1}{\sigma_{\nu_n}} \phi(Z) \]
Indicators: discrete output

$$f_3(I_n | X_n, X^*_n; \alpha, \Sigma \nu)$$

For instance:

$$I_n = m(X_n, X^*_n; \alpha) + \nu_n, \quad \nu_n \sim \text{Logistic}(0, 1)$$

$$P(I_n = 1) = \Pr(m(\cdot) \leq \tau_1) = \frac{1}{1 + e^{-\tau_1 + m(\cdot)}}$$

$$P(I_n = 2) = \Pr(m(\cdot) \leq \tau_2) - \Pr(m(\cdot) \leq \tau_1) = \frac{1}{1 + e^{-\tau_2 + m(\cdot)}} - \frac{1}{1 + e^{-\tau_1 + m(\cdot)}}$$

$$\vdots$$

$$P(I_n = 5) = 1 - \Pr(m(\cdot) \leq \tau_4) = 1 - \frac{1}{1 + e^{-\tau_4 + m(\cdot)}}$$
Indicators: discrete output

\[ \Pr(\tau_{q-1} \leq m(\cdot) \leq \tau_q) \]
Estimation: likelihood

Assuming $\omega_n$, $\varepsilon_n$ and $\nu_n$ are independent, we have

$$
\mathcal{L}_n(y_n, I_n|X_n; \alpha, \beta, \lambda, \Sigma_\varepsilon, \Sigma_\nu, \Sigma_\omega) =

\int_{X^*} P(y_n|X_n, X^*; \beta, \Sigma_\varepsilon)f_3(I_n|X_n, X^*; \alpha, \Sigma_\nu)f_1(X^*|X_n; \lambda, \Sigma_\omega)\,dX^*.
$$

Maximum likelihood estimation:

$$
\max_{\alpha, \beta, \lambda, \Sigma_\varepsilon, \Sigma_\nu, \Sigma_\omega} \sum_n \log (\mathcal{L}_n(y_n, I_n|X_n; \alpha, \beta, \lambda, \Sigma_\varepsilon, \Sigma_\nu, \Sigma_\omega))
$$

Source: Walker (2001)
Software issues

- Specification of the models
- Compute integrals
- Loop on data
- Compute derivatives
Biogeme

Current version:
- Designed for discrete choice (MEV models).
- Syntax: specific to Biogeme.
- Models are pre-implemented, with their derivatives.
- Users focus only on the utility functions.
- Additional features for random parameters, and panel data.

New version:
- Designed for any model
- Syntax: python 3.1 (case sensitive) python.org
- Compiler: python → C++ (likelihood and derivatives)
- Users can create the model from scratch, or use pre-implemented libraries
- Additional features for integration, and loops.
Example: a logit model

Import libraries

```python
from biogeme import *
from headers import *
from logit import *
from loglikelihood import *
from statistics import *
```
Example: a logit model

Define the data file:

dataFile = "sample.dat"

Define the parameters to be estimated:

ASC1 = Beta('ASC1', 0, -10000, 10000, 1)
ASC2 = Beta('ASC2', 0, -10000, 10000, 0)
ASC3 = Beta('ASC3', 0, -10000, 10000, 0)
ASC4 = Beta('ASC4', 0, -10000, 10000, 0)
ASC5 = Beta('ASC5', 0, -10000, 10000, 0)
ASC6 = Beta('ASC6', 0, -10000, 10000, 0)
BETA1 = Beta('BETA1', 0, -10000, 10000, 0)
BETA2 = Beta('BETA2', 0, -10000, 10000, 0)
Example: a logit model

Define the utility functions (structural equations):

\[
\begin{align*}
V_1 &= ASC_1 + BETA_1 \times x_{11} + x_{12} \times BETA_2 \\
V_2 &= ASC_2 + BETA_1 \times x_{21} + BETA_2 \times x_{22} \\
V_3 &= ASC_3 + BETA_1 \times x_{31} + BETA_2 \times x_{32} \\
V_4 &= ASC_4 + BETA_1 \times x_{41} + x_{42} \times BETA_2 \\
V_5 &= ASC_5 + BETA_1 \times x_{51} + BETA_2 \times x_{52} \\
V_6 &= ASC_6 + BETA_1 \times x_{61} + BETA_2 \times x_{62}
\end{align*}
\]

Assign alternatives IDs to utility functions:

\[
V = \{1: V_1, \\
2: V_2, \\
3: V_3, \\
4: V_4, \\
5: V_5, \\
6: V_6\}
\]

Python data structure: dictionary

http://docs.python.org/py3k/tutorial/datastructures.html#dictionaries
Example: a logit model

Assign availabilities to alternatives:

```python
av = {1: av1,
     2: av2,
     3: av3,
     4: av4,
     5: av5,
     6: av6}
```

Measurement equation: logit model

- use the library
- or write it yourself
Example: a logit model

Using the library:

```plaintext
prob = logit_av(V,av,Choice)
rowIterator('obsIter', Datafile(dataFile))
BIOGEME_OBJECT.ESTIMATE = Sum(log(prob),'obsIter')
```

- **Iterators:**
  - `Sum(formula,iterator)`,
  - `Prod(formula,iterator)`

- **Function to be maximized:** `BIOGEME_OBJECT.ESTIMATE`
Example: a logit model

Not using the library:

\[ P(i) = \frac{e^{V_i}}{\sum_j a_j e^{V_j}} = \frac{1}{\sum_j a_j e^{V_j-V_i}}, \quad \log P(i) = -\log(\sum_j a_j e^{V_j-V_i}) \]

chosen = Elem(V,Choice)
den = 0
for i,v in V.items() :
    den += av[i] * exp(v-chosen)
logprob = -log(den)
rowIterator('obsIter', Datafile(dataFile))
BIOGEME_OBJECT.ESTIMATE = Sum(logprob,'obsIter')

- `Elem(dict,index)` returns the element of a dictionary when the index varies across the sample.

- Syntax for loops from python.
Example: a latent variable

Attitude towards car: structural equation

\[
\omega = \text{RandomVariable('omega')}
\]
\[
\text{attCar} = b_{\text{cteAttCar}} + b_{\text{female}} \times \text{female} + b_{\text{sigma}} \times \omega
\]

Measurement equations

\[
z_{04} = (I_{04} - b_{\alpha_{04}} - b_{\lambda_{04}} \times \text{attCar}) / b_{\sigma_{04}}
\]
\[
f_{04} = \text{normalpdf}(z_{04})/ b_{\sigma_{04}}
\]

\[
z_{05} = (I_{05} - b_{\alpha_{05}} - b_{\lambda_{05}} \times \text{attCar}) / b_{\sigma_{05}}
\]
\[
f_{05} = \text{normalpdf}(z_{05})/ b_{\sigma_{05}}
\]

\[
z_{10} = (I_{10} - b_{\alpha_{10}} - b_{\lambda_{10}} \times \text{attCar}) / b_{\sigma_{10}}
\]
\[
f_{10} = \text{normalpdf}(z_{10})/ b_{\sigma_{10}}
\]

\[
z_{12} = (I_{12} - b_{\alpha_{12}} - b_{\lambda_{12}} \times \text{attCar}) / b_{\sigma_{12}}
\]
\[
f_{12} = \text{normalpdf}(z_{12})/ b_{\sigma_{12}}
\]
Example: a latent variable

Likelihood

condLike = P * f04 * f05 * f10 * f12
like = Integrate(condLike * normalpdf(omega),'omega')
rowIterator('obsIter', Datafile(dataFile))
BIOGEME_OBJECT.ESTIMATE = Sum(log(like),'obsIter')

- `Integrate` uses Gauss-Hermite integration
- If integrals of more than one dimension, simulation can be used.
- We illustrate simulation on the same example.
Example: a latent variable, with simulation

\[
\omega = \text{bioNormal}('\omega')
\]
\[
\text{condLike} = P \ast f04 \ast f05 \ast f10 \ast f12
\]
\[
\text{drawIterator}('\text{drawIter}')
\]
\[
\text{like} = \text{Sum}('\text{condLike}', '\text{drawIter}')
\]
\[
\text{rowIterator}('\text{obsIter}', \text{Datafile(dataFile)})
\]
\[
\text{BIOGEME\_OBJECT.ESTIMATE} = \text{Sum}(\text{log}('\text{like}'), '\text{obsIter}')
\]

- **Sum** is used here to approximate the integral
- Same syntax, with another iterator
- Other iterators exist, in particular for panel data.
Sample enumeration

- Copy the value of the estimated parameters from the file `mymodel_param.py`
- Define the quantities that must be computed
  
  \[
  P_1 = \text{Integrate}(\text{cond}P_1 \times \text{density}, '\omega')
  \]
  \[
  \text{costElasticity} = \text{Derive}(P_1, '\text{cost}_1') \times \text{cost}_1 / P_1
  \]
- Gather them in a dictionary, with appropriate labels:
  
  \[
  S = \{'\text{Prob. alt. 1}': P_1, \\
  '\text{Cost elasticity alt. 1}': \text{costElasticity}\}
  \]
- Instruct BIOGEME to simulate:
  
  \[
  \text{BIOGEME_OBJECT.SIMULATE} = \text{Enumerate}(S, '\text{obsIter}')
  \]
Technicalities

- Alpha version is running, not distributed yet.
- Requires a GNU environment (Linux or Mac OS X)
- Exploits multiple processors
- Output in HTML, with sortable tables