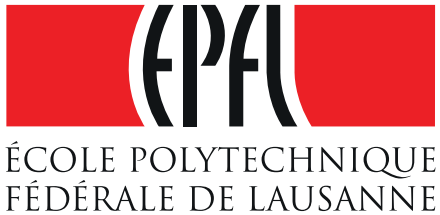

Automatic Utility Specification in Discrete Choice Models

Workshop on Discrete Choice Models

Nicola Ortelli

Lausanne, April 2019

Outline

Outline

Introduction

Motivation

- DCMs are widely used to understand and predict the choices of individuals.
- Such models are powerful, but specifying utility functions is time-consuming and prone to errors.
- In particular, any misspecification may lead to severe bias in the estimates.

Objective

- Automatize the development of utility functions in MNL models.
- Avoid erroneous use of domain knowledge.
- Explore the set of possible specifications in a thorough and "impartial" way.

Outline

The Multinomial Logit

Utility

$$U_{in} = V_{in} + \varepsilon_{in}$$

- All ε_{in} are independent and identically distributed: $\varepsilon_{in} \sim \mathcal{G}(0, 1), \forall i \in C_n$.

Systematic component

$$V_{in} = \sum_{k=1}^{K_i} \beta_{ik} x_{ink} = \boldsymbol{\beta_i^T x_{in}}$$

- V_{in} is assumed to be linear in the parameters.

The Multinomial Logit

Nonlinearities

$$V_{in} = \cdots + \beta_{ik} h_{ik}(x_{ink}) + \cdots$$

- Non-linear transformations of the variables may still be considered.
- Examples: polynomial, piecewise linear, logarithmic transformations, etc.

Taste variation

$$V_{in} = \cdots + \sum_{d=1}^{D_c} \beta_{ikd} \delta_d(c_{nk}) x_{ink} + \cdots$$

- Heterogeneity in a population may be captured by assigning different parameters to different socioeconomic groups.

The Multinomial Logit

To sum up...

$$V_{in} = \sum_{k=1}^{K_i} \sum_{d=1}^{D_c} \beta_{ikd} \delta_d(c_{nk}) h_{ik}(x_{ink})$$

- Developing a MNL consists in deciding on the form of V_{in}:
  - Which variables should be included?
  - Which quantities should they be interacting with?
  - Which non-linear transformations should be considered?

VNS for Automatic Utility Specification

Main idea

- Utility building as a combinatorial optimization problem.
- Each possible specification is a candidate in the search space.
- VNS to explore it in a "strategic" way.

Challenges

- Define a suitable set of neighborhood structures.
- Identify an appropriate objective function.
- Make the search as fast as possible.

Utility Building as a Combinatorial Optimization Problem

Neighborhood structures

- *Add* / *Remove*
- *Unlinearize* / *Linearize*
- *Interact* / *Uninteract*

Measures of performance

- $f_{\text{CV}} = -\sum_{h=1}^{H} \mathcal{L}_h(\boldsymbol{\beta})$
- $f_{\text{AIC}} = 2K - 2\mathcal{L}(\boldsymbol{\beta})$
- $f_{\text{BIC}} = \log(N)K - 2\mathcal{L}(\boldsymbol{\beta})$

Speeding up the VNS

- No "shaking" step.
- First improvement local search.
- Updated list of "less promising" moves.

On the Validity of the Obtained Models

Possible issues

- Nothing guarantees s_{opt} to be behaviorally realistic.
- No formal or informal tests are applied during the search.
- s_{opt} may therefore contain some inconsistencies.

Still...

- "Let the data speak for themselves".
- Any model should be coherent with the structure of the data it uses, rather than forcing the latter into unverifiable structural assumptions.
- In any case, inconsistencies are uncommon in VNS-generated models.

Generating Semi-Artificial Data

Procedure

- Specify a set of arbitrary utility functions $s^{\dagger} = \{U_{in} = V_{in} + \varepsilon_{in} \mid i \in C\}$.
- Estimate $s^{\dagger}$ using the original choices and compute all $P_n(i)$.
- Sample a new set of simulated choices from the $P_n(i)$.

Remarks

- Only the response variable is created from scratch.
- Variables from an existing dataset ensure realism.

Assessing the Quality of a Specification

Minimum set of operations

- Number of operations to reach $s^\dagger$ from s_{opt}.
- In other words, similarity as their distance in the neighborhood structures.

Kullback-Leibler distance

$$\mathcal{D}_{KL}(s^\dagger || s_{opt}) = \sum_{n=1}^{N} \sum_{i \in C_n} \log \left(\frac{P_n^{s^\dagger}(i)}{P_n^{s_{opt}}(i)} \right) P_n^{s^\dagger}(i),$$

- Reflects the "amount of information" that is lost by s_{opt}.
- The smaller $\mathcal{D}_{KL}(s^\dagger || s_{opt})$, the "more similar" the outcomes of the models.

Outline

The Swissmetro Dataset (Bierlaire *et al.*, 2001)

Characteristics

- Data collected to analyze the potential impact of the Swissmetro.
- Stated preference data.
- More than 10 000 "observations".
- Potential issues due to high correlation among variables.
- 20% is set aside as the test set.

Considered Variables and Statistics

Variable	min	max	mean	std.
GE *Traveler's gender. 0 if female, 1 if male.*	0	1	0.75	0.43
GA *Travel card ownership. 1 if the traveler owns one, 0 otherwise.*	0	1	0.14	0.35
TRN_{TT} *Train travel time* [*min*]. *Based on the car distance.*	31	1049	166.63	77.35
TRN_{CO} *Train cost* [*CHF*]. *If the traveler owns a GA, equal to its price.*	4	5040	514.34	1088.93
TRN_{HE} *Train headway* [*min*].	30	120	70.10	37.43
SM_{TT} *Swissmetro travel time* [*min*]. *A speed of 500 km/h is considered.*	8	796	87.47	53.55
SM_{CO} *Swissmetro cost* [*CHF*]. *Proportional to the rail fare.*	6	6720	670.34	1441.59
SM_{HE} *Train headway* [*min*].	10	30	20.02	8.16
CAR_{TT} *Car travel time* [*min*].	0	1560	123.80	88.71
CAR_{CO} *Car cost* [*CHF*]. *A fixed average cost per kilometer is considered.*	0	520	78.74	55.26

Experiments

Experiments on artificial data

- Four "playgrounds" of increasing complexity.
- Simulated choices from randomly built specifications.
- Different probabilities for variable inclusion, transformation and segmentation.

Specification type	$P_{s^\dagger}(include)$	$P_{s^\dagger}(tranform)$	$P_{s^\dagger}(segment)$	Use in playgrounds
Simple	$\mathcal{U}(0.1, 0.9)$	0	0	A B C D
With transformations	$\mathcal{U}(0.1, 0.9)$	$\mathcal{U}(0.3, 0.7)$	0	B D
With segmentations	$\mathcal{U}(0.1, 0.9)$	0	$\mathcal{U}(0.3, 0.7)$	C D
Full	$\mathcal{U}(0.1, 0.9)$	$\mathcal{U}(0.3, 0.7)$	$\mathcal{U}(0.3, 0.7)$	D

Playgrounds

Playground	Specifications	VNS structures
A	20 simple	*Add* / *Remove*
B	20 simple 20 with tra.	*Add* / *Remove* *Unlinearize* / *Linearize*
C	20 simple 20 with seg.	*Add* / *Remove* *Interact* / *Uninteract*
D	20 simple 20 with tra. 20 with seg. 20 full	*Add* / *Remove* *Unlinearize* / *Linearize* *Interact* / *Uninteract*

Outline

Playground A

$f(\cdot)$	Specifications (corr. retrieved)		Min. set of operations			$\mathcal{D}_{KL}(s^{\dagger}\|\|s_{opt})$		Iterations
			A/R	U/L	I/U	Train	Test	
CV	20 simple	(8/18)	0/0.65			0.13	0.03	19.3
AIC	20 simple	(10/18)	0/0.5			0.86	0.20	18.6
BIC	20 simple	(12/20)	0.4/0			1.05	0.26	17.1

Playground B

$f(\cdot)$	Specifications (corr. retrieved)		Min. set of operations A/R	U/L	I/U	$\mathcal{D}_{KL}(s^{\dagger}\|\|s_{opt})$ Train	Test	Iterations
CV	20 simple	(3/14)	0.15/1	−/0.65		1.82	0.43	30.2
	20 with tra.	(3/15)	0.1/0.9	0.5/0.4		2.93	0.72	39.2
AIC	20 simple	(6/16)	0.1/0.9	−/0.93		2.36	0.54	29.4
	20 with tra.	(4/16)	0.1/0.65	0.5/0.45		2.85	0.71	38.7
BIC	20 simple	(7/16)	0.5/0	−/0.25		1.36	0.33	23.2
	20 with tra.	(6/16)	0.6/0.05	0.55/0.05		2.88	0.73	31.2

Playground C

$f(\cdot)$	Specifications (corr. retrieved)		Min. set of operations A/R	U/L	I/U	$\mathcal{D}_{KL}(s^{\dagger}\|\|s_{opt})$ Train	Test	Iterations
CV	20 simple	(1/6)	0.15/1.05		−/2.3	4.34	1.29	49.4
	20 with seg.	(0/4)	0.15/1.5		0.4/2.15	4.62	1.37	87.9
AIC	20 simple	(4/11)	0.1/0.8		−/1.35	3.63	1.03	47.4
	20 with seg.	(2/5)	0.2/0.85		0.45/1.9	4.99	1.46	80.0
BIC	20 simple	(10/20)	0.5/0		−/0	1.09	0.27	28.9
	20 with seg.	(6/13)	0.5/0.1		1/0.3	3.73	1.15	69.2

Playground D

$f(\cdot)$	Specifications (corr. retrieved)		Min. set of operations			$\mathcal{D}_{KL}(s^{\dagger}\|\|s_{opt})$		Iterations
			A/R	U/L	I/U	Train	Test	
CV	20 simple	(1/5)	0.15/1	−/0.7	−/2.1	4.19	1.22	62.2
	20 with tra.	(1/4)	0.1/0.85	0.5/0.45	−/1.95	5.99	1.57	73.9
	20 with seg.	(0/4)	0.1/1.65	−/1.6	0.55/2.6	7.27	2.25	113.5
	20 full	(0/4)	0.4/1.0	0.6/0.8	1.6/3.4	13.37	5.11	145.7
AIC	20 simple	(3/10)	0.1/0.75	−/0.7	−/0.95	3.51	1.05	57.5
	20 with tra.	(2/6)	0.1/0.65	0.5/0.4	−/1.25	5.51	1.45	65.8
	20 with seg.	(1/3)	0.1/1.05	−/1.15	0.45/2.15	6.12	1.73	103.1
	20 full	(0/5)	0.3/1.35	0.35/0.65	1.15/2.35	9.77	3.32	126.1
BIC	20 simple	(7/16)	0.5/0	−/0.25	−/0	1.36	0.33	34.1
	20 with tra.	(5/16)	0.65/0.05	0.6/0.05	−/0.1	3.23	0.83	43.5
	20 with seg.	(3/12)	0.5/0.25	−/0.55	1/0.35	4.88	1.50	94.8
	20 full	(2/11)	1.05/0.35	0.7/0.45	2.6/1.35	14.44	4.76	85.2

Real Choices: CV

Parameter	Estimate	t-test
ASC-TRN $_{GA=0}$	10.9	16.5
ASC-TRN $_{GA=1}$	36.8	9.38
B-LOG(TRN_{TT}) $_{GE=0,\ GA=0}$	−2.14	−14.7
B-LOG(TRN_{TT}) $_{GE=0,\ GA=1}$	0.313	0.868 *
B-LOG(TRN_{TT}) $_{GE=1,\ GA=0}$	−3.3	−25.7
B-LOG(TRN_{TT}) $_{GE=1,\ GA=1}$	−0.397	−1.3 *
B-LOG(TRN_{CO}) $_{GA=0}$	−1.18	−15.2
B-LOG(TRN_{CO}) $_{GA=1}$	−3.64	−9.66
B-LOG(TRN_{HE})	−0.495	−7.52
ASC-SM $_{GE=0,\ GA=0}$	8.01	12.0
ASC-SM $_{GE=0,\ GA=1}$	38.9	9.49
ASC-SM $_{GE=1,\ GA=0}$	2.75	6.11
ASC-SM $_{GE=1,\ GA=1}$	36.3	9.12
B-LOG(SM_{TT}) $_{GA=0}$	−1.75	−23.7
B-LOG(SM_{TT}) $_{GA=1}$	−1.12	−4.54
B-LOG(SM_{CO}) $_{GA=0}$	−1.22	−21.4
B-LOG(SM_{CO}) $_{GA=1}$	−3.3	−8.87
B-SM_{HE} $_{GE=0}$	−0.0165	−2.76
B-SM_{HE} $_{GE=1}$	−0.00374	−1.11 *
ASC-CAR	0	—
B-LOG(CAR_{TT}) $_{GE=0,\ GA=0}$	−1.1	−7.99
B-LOG(CAR_{TT}) $_{GE=0,\ GA=1}$	1.54	2.6
B-LOG(CAR_{TT}) $_{GE=1,\ GA=0}$	−2.1	−20.6
B-LOG(CAR_{TT}) $_{GE=1,\ GA=1}$	0.399	0.764 *
B-CAR_{CO}	−0.00924	−10.0
LL at convergence	−6236.60	

Real Choices: AIC

Parameter	Estimate	t-test
ASC-TRN $_{GE=0,\ GA=0}$	14.9	16.1
ASC-TRN $_{GE=0,\ GA=1}$	33.2	9.49
ASC-TRN $_{GE=1,\ GA=0}$	18.2	25.2
ASC-TRN $_{GE=1,\ GA=1}$	34.8	10.0
B-LOG(TRN_{TT}) $_{GE=0,\ GA=0}$	−2.03	−11.8
B-LOG(TRN_{TT}) $_{GE=0,\ GA=1}$	0.195	0.534 ∗
B-LOG(TRN_{TT}) $_{GE=1,\ GA=0}$	−3.11	−22.3
B-LOG(TRN_{TT}) $_{GE=1,\ GA=1}$	−0.297	−0.962 ∗
B-LOG(TRN_{CO}) $_{GA=0}$	−1.19	−15.0
B-LOG(TRN_{CO}) $_{GA=1}$	−3.64	−9.65
B-LOG(TRN_{HE})	−0.5	−7.59
ASC-SM $_{GE=0,\ GA=0}$	11.9	21.5
ASC-SM $_{GE=0,\ GA=1}$	44.5	9.73
ASC-SM $_{GE=1,\ GA=0}$	10.5	26.5
ASC-SM $_{GE=1,\ GA=1}$	30.8	8.7
B-LOG(SM_{TT}) $_{GA=0}$	−1.65	−23.1
B-LOG(SM_{TT}) $_{GA=1}$	−1.08	−4.37
B-LOG(SM_{CO}) $_{GE=0,\ GA=0}$	−1.19	−12.1
B-LOG(SM_{CO}) $_{GE=0,\ GA=1}$	−4.51	−8.61
B-LOG(SM_{CO}) $_{GE=1,\ GA=0}$	−1.24	−20.1
B-LOG(SM_{CO}) $_{GE=1,\ GA=1}$	−2.84	−7.22
B-SM_{HE} $_{GE=0}$	−0.0163	−2.74
B-SM_{HE} $_{GE=1}$	−0.00432	−1.28 ∗
ASC-CAR	0	—
B-CAR_{TT} $_{GE=0,\ GA=0}$	−0.00579	−5.15
B-CAR_{TT} $_{GE=0,\ GA=1}$	0.0312	2.61
B-CAR_{TT} $_{GE=1,\ GA=0}$	−0.0148	−19.8
B-CAR_{TT} $_{GE=1,\ GA=1}$	0.00411	0.703 ∗
B-CAR_{CO}	−0.00948	−10.2
LL at convergence	−6223.81	

Real Choices: BIC

Parameter	Estimate	t-test
ASC-TRN: GA=0	16.9	26.4
ASC-TRN: GA=1	33.9	10.9
B-LOG(TRN_{TT}) $_{GE=0,\ GA=0}$	−2.39	−20.3
B-LOG(TRN_{TT}) $_{GE=0,\ GA=1}$	−0.682	−3.54
B-LOG(TRN_{TT}) $_{GE=1,\ GA=0}$	−2.83	−24.6
B-LOG(TRN_{TT}) $_{GE=1,\ GA=1}$	−0.575	−3.21
B-LOG(TRN_{CO}) $_{GA=0}$	−1.22	−15.9
B-LOG(TRN_{CO}) $_{GA=1}$	−3.58	−9.59
B-LOG(TRN_{HE})	−0.49	−7.46
ASC-SM $_{GE=0,\ GA=0}$	11.6	28.9
ASC-SM $_{GE=0,\ GA=1}$	33.2	10.7
ASC-SM $_{GE=1,\ GA=0}$	10.2	27.7
ASC-SM $_{GE=1,\ GA=1}$	34.7	10.9
B-LOG(SM_{TT})	−1.61	−23.5
B-LOG(SM_{CO}) $_{GA=0}$	−1.22	−21.4
B-LOG(SM_{CO}) $_{GA=1}$	−3.26	−8.79
ASC-CAR	0	—
$B\text{-}CAR_{TT\ GE=0}$	−0.00527	−4.79
$B\text{-}CAR_{TT\ GE=1}$	−0.0145	−19.8
$B\text{-}CAR_{CO}$	−0.00936	−10.1
LL at convergence	−6250.88	

Comparison with the Benchmark (Bierlaire *et al.*, 2001)

	VNS-CV	VNS-AIC	VNS-BIC	Benchmark
LL on train set	−6236.60	−6223.81	−6250.88	−6759.69
LL on test set	−1500.74	−1533.21	−1493.72	−1695.30
Estimated parameters	24	28	19	10
Considered variables	10	10	9	12
Iterations	172	149	199	–

Outline

Conclusion

Summary

- Attempt at building an automatic utility specification procedure.
- Translation into a combinatorial optimization problem.
- Validity and effectiveness were tested on semi-artificial and real data.

Main findings

- Precision in retrieving the correct specifications varies greatly.
- VNS-BIC yields impressive results even in the most intricate settings.
- Novel specifications on real data.

Conclusion

Directions for further research

- Assess the adaptability and scalability of the method:
  - other datasets,
  - more variables under consideration,
  - more complex specifications,
  - different choice models.
- Explore new approaches:
  - automatic relevance determination,
  - causal inference,
  - model averaging.