BHAMSLE: A Breakpoint Heuristic Algorithm for Maximum Simulated Likelihood Estimation of Advanced Discrete Choice Models

Tom Haering*[1] and Michel Bierlaire[1]

[1]École Polytechnique Fédérale de Lausanne (EPFL), School of Architecture, Civil and Environmental Engineering (ENAC), Transport and Mobility Laboratory, Switzerland, {tom.haering, michel.bierlaire}@epfl.ch

Short summary

This paper introduces BHAMSLE, a Breakpoint Heuristic Algorithm for Maximum Simulated Likelihood Estimation (MSLE), adapted from the Breakpoint Heuristic Algorithm (BHA) for choice-based pricing, bridging the gap between choice-based optimization and choice model estimation. Similarly to the BHA, BHAMSLE leverages indifference points—or breakpoints—in individual decision-making to systematically explore local optima. Benchmark comparisons with PandasBiogeme, the current state-of-the-art software for DCM estimation, demonstrate that BHAMSLE, both as a standalone estimation procedure and as an approach for obtaining high-quality starting points, substantially improves log-likelihood on different latent class logit as well as latent class mixed logit models across 100 random samples, with gains of up to 10% for observed choices and up to 16% for synthetic choices. Notably, small numbers of draws are often enough to observe significant gains, with larger samples further amplifying these improvements.

Keywords: Discrete Choice, Heuristic, Latent Class, Maximum Simulated Likelihood Estimation.

1 Introduction

Maximum likelihood estimation (MLE) is a widely used method for estimating parameters of a specified distribution based on observed data. It plays a significant role in fields such as physics (Hauschild & Jentschel, 2001), machine learning (Goodfellow et al., 2016), and discrete choice modeling (Bierlaire, 2023).

The estimation of a discrete choice model involves determining coefficient values that maximize the log-likelihood of the observed data. This process typically begins with initializing the coefficients, followed by iterative updates through an optimization algorithm until a predefined convergence criterion is met. Consequently, the initialization of coefficients—along with the chosen algorithm—directly influences the trajectory of the estimation process. For widely used models such as the multinomial logit, nested logit, and generic mixed logit, this initialization rarely poses a significant issue. These models generally exhibit a unique global optimum, making the estimation process relatively straightforward.

The latent class model specifically has seen an increase in popularity over the last decade. Where the mixed logit merely allows to control for unobserved heterogeneity, the latent class model generally also allows to get a better understanding of that heterogeneity. Despite this advantage, latent class models face a critical drawback—difficulty in estimation due to numerous local optima, with some model specifications yielding hundreds of potential solutions (Peer et al., 2016). As a result, the initialization of the coefficients and the specific estimation algorithm used heavily influence the identified solution. To ensure convergence to the global optimum, it is essential to perform multiple estimations with diverse initializations. This issue is well-documented in the literature; for instance, Jung & Wickrama (2008) emphasize the prevalence of local solutions in latent class modeling and advocate for repeated random initialization as a necessary practice.

For advanced discrete choice models such as mixed logit and latent class mixed logit models, additional challenges arise due to the complete lack of closed-form expressions in their choice probabilities. Consequently, determining optimal parameters relies on simulation techniques, like maximum simulated likelihood estimation (MSLE) (Train, 2003).

A general MSLE approach was introduced by Fernandez Antolin (2018), framing the problem as a mixed-integer linear program (MILP), demonstrating that MSLE can be seen as a choice-based

optimization problem. Traditionally, such problems integrate a DCM to account for stochastic behavior within an optimization context, often targeting endogenous parameters, such as the price of a product, to maximize revenue or other metrics. In the case of MSLE, one instead assumes fixed choice attributes, with the choice model parameters taking on the role of the decision variables, maximizing the simulated likelihood as the objective function. This perspective bridges a gap between choice-based optimization techniques and simulated likelihood estimation, suggesting potential for cross-applications between the two.
Building on this insight, this study introduces BHAMSLE (Breakpoint Heuristic Algorithm for MSLE), adapted from the Breakpoint Heuristic Algorithm (BHA) originally designed for choice-based pricing Haering et al. (2024). BHAMSLE systematically explores local optima by leveraging "breakpoints" that capture critical shifts in individuals' decision-making. The algorithm is evaluated on a set of latent class estimation problems, where we assess its performance both as a standalone estimation method and as an initialization tool for PandasBiogeme. The remainder of the paper is organized as follows: Section 2 provides the necessary notation and describes BHAMSLE, while Section 3 presents the case study and computational experiments. Finally, Section 4 offers concluding remarks and essential takeaways.

2 Methodology

In this section, we give the problem formulation for MSLE and introduce BHAMSLE. While we illustrate both the problem and algorithm specifically for the case of a latent class model, it is important to emphasize that the algorithm is general and can be applied to estimate any DCM.

MSLE problem formulation

Consider a set of $\mathcal{N} = \{1, \dots, N\}$ individuals, each choosing exactly one option amongst a set of choices $\mathcal{I} = \{1, \dots, I\}$. An individual may have access to only a subset of these choices, indicated by their choice set $C_n \subset \mathcal{I}$. The observed choice for individual $n \in \mathcal{N}$ is denoted by $y_n \in \mathcal{I}$. Each alternative is assigned a stochastic utility U_{in}, composed of a deterministic component V_{in} and a random error term ε_{in}. The deterministic component is represented by the linear combination of product attributes and socio-economic characteristics x_{ink} (where $k \in \mathcal{K} = \{1, \dots, K\}$ indexes the set of all such factors) with the endogenous parameters β_k that are to be estimated. The error term ε_{in} captures unobserved and irrational behavior. The utility U_{in} is now given by:

$$U_{in} = V_{in} + \varepsilon_{in} = \sum_{k \in \mathcal{K}} x_{ink}\beta_k + \varepsilon_{in} \quad n \in \mathcal{N},\ i \in C_n.$$

We furthermore assume that each individual $n \in \mathcal{N}$ selects the alternative $i \in C_n$ corresponding to the maximal utility U_{in}. In latent class models, the analyst tests the hypothesis that the population of individuals can be divided into a set of latent classes $\mathcal{C} = \{1, \dots, C\}$, each characterized by distinct preferences. The probability that an individual belongs to latent class $c \in \mathcal{C}$ is denoted by π_c and requires estimation. As $\sum_c \pi_c = 1$, only $C - 1$ of these probabilities are independently estimable, with the final probability determined by this condition. The probability that individual n selects alternative i given their membership in class c is given by:

$$P_{in|c} = \mathbb{P}(U^c_{in} \geq U^c_{jn},\ \forall j \in C_n),$$

where U^c_{in} represents the utility of alternative i for individual n, given class membership c. The unconditional probability P_{in} of individual n choosing option i is then described by $\sum_c \pi_c P_{in|c}$. For advanced DCMs, P_{in} may not have a closed-form expression, necessitating approximation through random draws to simulate error components and class memberships. For example, in a mixed logit model, a parameter β_m might be assumed to be distributed normally amongst the population with mean β^{mean}_m and standard deviation β^{std}_m. For each simulation scenario $r \in \mathcal{R} = \{1, \dots, R\}$ we can then describe the deterministic utility U_{inr} as:

$$U_{inr} = \Big[\sum_{k \in \mathcal{K} \setminus \{m\}} x_{ink}\beta_k \Big] + x_{inm}(\beta^{\text{mean}}_m + u_{nr}\beta^{\text{std}}_m) + \varepsilon_{inr} \quad n \in \mathcal{N},\ i \in C_n,\ r \in \mathcal{R},$$

where u_{nr} is a draw from $\mathcal{N}(0, 1)$ and ε_{inr} a draw from Gumbel(0, 1). The simulated choices for each scenario are captured by binary variables ω_{inr}, i.e. $\omega_{inr} = 1 \Leftrightarrow U_{inr} = \max_j U_{jnr}$. Now

$\frac{1}{R}\sum_r \omega_{inr}$ provides an unbiased estimator for P_{in}. The objective function to be maximized, the simulated log-likelihood $sLL(\pi, \beta)$, is given by:

$$sLL(\pi, \beta) = \sum_{n \in \mathcal{N}} \ln\Big(\frac{1}{R}\sum_{r \in \mathcal{R}} \omega_{y_n nr}\Big).$$

Breakpoint Heuristic Algorithm for MSLE (BHAMSLE)

BHAMSLE capitalizes on the idea of decision-making breakpoints, more specifically "entry" and "exit" breakpoints for each individual n and scenario r, signifying where the tuple (n, r) is captured or lost. These breakpoints represent a set of local optima that can be enumerated. The method can be categorized as a coordinate descent (ascent), iteratively optimizing one parameter at a time while fixing all others, terminating once no parameter can be improved further. Let $\gamma_g,\ g \in \mathcal{G} = \{1, \ldots, C-1\}$ represent the parameters that separate the unit interval into C partitions $P_1, \ldots, P_C$. Furthermore, draws from the uniform $[0, 1]$ distribution used to simulate class membership are denoted by $\sigma_{nr}, n \in \mathcal{N}, r \in \mathcal{R}$. The full algorithm is described in the following procedure:

1. Choose a starting point for the estimation, usually, $\beta_k^* = 0,\ k \in \mathcal{K},\ \gamma_g^* = \frac{g}{C},\ g \in \mathcal{G}$, and compute its objective value $o^* = SLL(\pi^*, \beta^*)$.
2. Set $j = 1$.
3. Fix all other parameters $\beta_k = \beta_k^*,\ k \neq j$ and $\gamma_g = \gamma_g^*,\ g \neq j - K$.
4. Compute the set of breakpoints, initialized as $\mathcal{B} = \{\}$:

   ```
   for $n \in \mathcal{N}, r \in \mathcal{R}$ :
       if $j \leq K$ :
           for $c \in \mathcal{C}$ :
               if $\sigma_{nr} \in P_c$ :
                   Compute the segment $[s_1, s_2]$ where $U^c_{y_n nr} \geq U^c_{inr}$ $\forall i \in C_n$. Add
                   $(s_1, n)$ as an entry breakpoint and $(s_2, n)$ as an exit breakpoint to $\mathcal{B}$.
               end
           end
       else
           Let $g \leftarrow j - K$.
           if $\sigma_{nr} \in (\gamma^*_{g-1}, \gamma^*_{g+1})$ :
               Let $W \leftarrow \{c \in \{g, g+1\} \mid U^c_{y_n nr} \geq U^c_{inr},\ \forall i \in \mathcal{I}\}$.
               if $W = \{g, g+1\}$ :
                   Add $(-\infty, n)$ as an entry breakpoint to $\mathcal{B}$.
               elseif $W = \{g\}$ :
                   Add $(\sigma_{nr}, n)$ as an entry breakpoint to $\mathcal{B}$.
               elseif $W = \{g+1\}$ :
                   Add $(\sigma_{nr}, n)$ as an exit breakpoint to $\mathcal{B}$.
               end
           end
       end
   end
   ```

5. Sort $\mathcal{B}$ in ascending order. Define $\Sigma_n = |\{\text{entry point } (x, y) \in \mathcal{B} : x = -\infty, y = n\}|,\ n \in \mathcal{N}$, $o = -N\ln(R) + \sum_n \ln(\Sigma_n)$ and $\mathcal{B} \leftarrow \{(x, y) \in \mathcal{B} : x \neq -\infty\}$. Then evaluate all $b \in \mathcal{B}$:

   ```
   for $b \in \mathcal{B}$ :
       if $b$ is an entry point :
           $o \mathrel{+}= \ln(\Sigma_n + 1) - \ln(\Sigma_n)$.
       else
           $o \mathrel{+}= \ln(\Sigma_n - 1) - \ln(\Sigma_n)$.
       end
       if $o > o^*$ :
           $o^* = o$, if $j \leq K$ set $\beta^*_j = b$, else set $\gamma^*_{j-K} = b$.
       end
   end
   ```

6. Set $j = j + 1$ (if now $j = K + C$, set $j = 1$) and repeat from step 3.
7. Terminate when no improvement is found over $K + C - 1$ iterations.

N	R	LL-Bio	sLL-B	LL-B	Gap (%)	(p_1, p_2)-Bio	(p_1, p_2)-B	T-Bio	T-B
500	1	-414.509	-804.719	-513.625	-23.91	(0.47, 0.53)	(0.50, 0.50)	9	0
500	5	-414.509	-804.719	-513.625	-23.91	(0.47, 0.53)	(0.50, 0.50)	7	0
500	10	-414.509	-692.120	-436.954	-5.41	(0.47, 0.53)	(0.60, 0.40)	8	0
500	20	-414.509	-421.346	-407.011	1.81	(0.47, 0.53)	(0.49, 0.51)	7	1
500	50	-414.509	-400.484	-397.845	4.02	(0.47, 0.53)	(0.55, 0.45)	3	2
500	100	-414.509	-398.038	-395.873	4.50	(0.47, 0.53)	(0.53, 0.47)	4	6
500	500	-414.509	-394.770	-395.622	4.56	(0.47, 0.53)	(0.55, 0.45)	3	31
500	1,000	-414.509	-395.112	-395.528	4.58	(0.47, 0.53)	(0.52, 0.48)	4	67
1,000	1	-828.307	-1609.440	-1027.450	-24.04	(0.49, 0.51)	(0.50, 0.50)	7	0
1,000	5	-828.307	-1609.440	-1027.450	-24.04	(0.49, 0.51)	(0.50, 0.50)	8	0
1,000	10	-828.307	-1402.890	-885.524	-6.91	(0.49, 0.51)	(0.55, 0.45)	9	1
1,000	20	-828.307	-853.607	-819.559	1.06	(0.49, 0.51)	(0.61, 0.39)	10	2
1,000	50	-828.307	-810.007	-796.224	3.87	(0.49, 0.51)	(0.57, 0.43)	4	6
1,000	100	-828.307	-798.965	-791.984	4.39	(0.49, 0.51)	(0.52, 0.48)	4	13
1,000	500	-828.307	-790.545	-790.466	4.57	(0.49, 0.51)	(0.55, 0.45)	4	66
1,000	1,000	-828.307	-788.956	-789.431	4.69	(0.49, 0.51)	(0.56, 0.44)	4	134

Table 1: BHAMSLE (B) vs. PandasBiogeme (Bio) on a latent class logit model with observed choices (N = population size, R = number of draws, sLL = simulated log-likelihood, LL = log-likelihood, T = estimation time in seconds)

Maintaining information on whether a breakpoint represents an entry or exit for a given individual n is crucial, as it enables the efficient processing of breakpoints in ascending order. This organization allows for the incremental computation of changes in the simulated log-likelihodd (sLL) in $\mathcal{O}(1)$ time per breakpoint. In contrast, evaluating the sLL objective function directly each time, as a generic global solver would, necessitates $\mathcal{O}(NR)$ operations. This distinction results in substantial computational savings, particularly for large-scale problems.

3 Results and discussion

To test our approach, we perform experiments on four different setups: latent class logit and latent class mixed logit models with observed vs. synthetically generated choices. All tests are performed in a single thread on a computational cluster node with two 2.4 GHz Intel Xeon Platinum 8360Y processors, utilizing 16 GB of RAM. We benchmark BHAMSLE against PandasBiogeme 3.2.14 (Bierlaire, 2023). For each test, we consider sample sizes $N = \{500, 1000\}$ and numbers of scenarios $R = \{50, 100, 500, 1000\}$, where for mixed logit models we increase the number of scenarios up to $R = 3000$. For every tuple (N, R) we take 100 samples from the full dataset (and respective distributions) and report the averaged obtained values. For the latent class mixed logit models, Biogeme's simulation module is used to compute the log-likelihood.

The first dataset is extracted from stated preference data on hypothetical mode choice collected in Switzerland (Bierlaire et al., 2001). Three alternatives are considered: Swissmetro (SM), rail, and car, with the latter being available only to car owners. In the first experiment, we hypothesize that there exists a portion of the population that has a different sensitivity to travel time than the rest. Thus a separate $\beta'_{\text{traveltime}}$ is estimated for this class. We refer to this class as class 2 and to the base model as class 1. The systematic utility equations for the two classes are:

$$\begin{aligned}
V^{(1)}_{\text{car}} &= \text{ASC}_{\text{car}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{car}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{car}}, \\
V^{(1)}_{\text{rail}} &= \text{ASC}_{\text{rail}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{rail}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{rail}} + \beta_{\text{headway}} \cdot \text{headway}_{\text{rail}}, \\
V^{(1)}_{\text{SM}} &= \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{SM}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{SM}} + \beta_{\text{headway}} \cdot \text{headway}_{\text{SM}}, \\
\\
V^{(2)}_{\text{car}} &= \text{ASC}_{\text{car}} + \beta'_{\text{traveltime}} \cdot \text{traveltime}_{\text{car}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{car}}, \\
V^{(2)}_{\text{rail}} &= \text{ASC}_{\text{rail}} + \beta'_{\text{traveltime}} \cdot \text{traveltime}_{\text{rail}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{rail}} + \beta_{\text{headway}} \cdot \text{headway}_{\text{rail}}, \\
V^{(2)}_{\text{SM}} &= \beta'_{\text{traveltime}} \cdot \text{traveltime}_{\text{SM}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{SM}} + \beta_{\text{headway}} \cdot \text{headway}_{\text{SM}}.
\end{aligned}$$

N	R	LL-Bio1	LL-Bio2	Gap (%)	(p_1, p_2)-Bio1	(p_1, p_2)-Bio2	T-Bio1	T-B	T-Bio2
500	1	-438.812	-438.760	0.01	(0.45, 0.55)	(0.44, 0.56)	17	0	20
500	5	-431.788	-428.005	0.88	(0.44, 0.56)	(0.41, 0.59)	14	0	13
500	10	-427.414	-428.099	-0.16	(0.45, 0.55)	(0.43, 0.57)	20	0	19
500	20	-426.447	-426.925	-0.11	(0.46, 0.54)	(0.49, 0.51)	23	1	20
500	50	-439.559	-435.483	0.93	(0.44, 0.56)	(0.42, 0.58)	24	3	17
500	100	-431.124	-433.809	-0.62	(0.40, 0.60)	(0.41, 0.59)	23	6	23
500	500	-490.745	-436.676	11.02	(0.33, 0.67)	(0.39, 0.61)	38	46	48
500	1,000	-488.010	-435.165	10.83	(0.34, 0.66)	(0.38, 0.62)	90	107	55
500	3,000	-474.640	-433.381	8.69	(0.34, 0.66)	(0.39, 0.61)	312	347	135
1,000	1	-877.418	-875.202	0.25	(0.39, 0.61)	(0.48, 0.52)	11	0	11
1,000	5	-868.597	-867.473	0.13	(0.41, 0.59)	(0.45, 0.55)	15	0	15
1,000	10	-855.605	-855.563	0.00	(0.46, 0.54)	(0.46, 0.54)	19	1	21
1,000	20	-856.742	-853.567	0.37	(0.45, 0.55)	(0.41, 0.59)	28	2	30
1,000	50	-869.742	-866.792	0.34	(0.44, 0.56)	(0.41, 0.59)	23	7	23
1,000	100	-888.778	-870.692	2.04	(0.44, 0.56)	(0.42, 0.58)	26	14	38
1,000	500	-867.117	-845.290	2.52	(0.44, 0.56)	(0.39, 0.61)	88	96	83
1,000	1,000	-869.915	-845.012	2.87	(0.43, 0.57)	(0.39, 0.61)	166	219	169
1,000	3,000	-868.542	-843.699	2.86	(0.43, 0.57)	(0.40, 0.60)	477	619	493

Table 2: Biogeme without (Bio1) and with (Bio2) BHAMSLE (B) starting point on a latent class mixed logit model with observed choices (N = population size, R = number of draws, sLL = simulated log-likelihood, LL = log-likelihood, T = estimation time in seconds)

Thus we have a total of seven parameters to be estimated. The results are presented in Table 1. We observe that, as the number of simulation draws R increases, the difference in the simulated log-likelihood (sLL) and the true log-likelihood (LL) for BHAMSLE decreases significantly, indicating that the approximation becomes more accurate with more draws. The comparison of log-likelihood values between the two methods demonstrates the effectiveness of the heuristic: starting from $R = 50$, BHAMSLE manages to consistently find higher quality solutions than Biogeme, with an average improvement of around 4.5% in log-likelihood. For the estimated probabilities of the two latent classes (p_1, p_2), Biogeme struggles to discern between the two classes, assigning practically uniform probabilities. For BHAMSLE, at low R, we observe the estimated probabilities remain close to uniform as well, but again around $R = 50$ draws we notice that BHAMSLE is able to capture a slightly higher probability for class 1, which likely is the main reason for the improved likelihood. Finally, in terms of runtime, BHAMSLE is significantly slower than Biogeme for larger numbers of draws. However, for small R, the estimation time is almost negligible, at only a couple of seconds.

In the second experiment, we make use of a similar model specification, but this time for class 2 we hypothesize that there exists a portion of the population that shows different intrinsic preferences for the alternatives than the rest of the people, thus separate alternative specific constants ASC'_{car} and $\text{ASC}'_{\text{rail}}$ are estimated. For class 1, we consider the $\beta_{\text{traveltime}}$ parameter to be normally distributed amongst the population, resulting in a latent class mixed logit model. To this extent, we denote $\beta^{\text{mixed}}_{\text{traveltime}} = \beta_{\text{traveltime}} + \beta^{\text{std}}_{\text{traveltime}} \cdot U_n$, where $U_n \sim \mathcal{N}(0, 1)$, and replace $\beta_{\text{traveltime}}$ by this new parameter for class 1, keeping everything else the same. We give the new systematic equations below:

$$\begin{aligned}
V^{(1)}_{\text{car}} &= \text{ASC}_{\text{car}} + \beta^{\text{mixed}}_{\text{traveltime}} \cdot \text{traveltime}_{\text{car}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{car}}, \\
V^{(1)}_{\text{rail}} &= \text{ASC}_{\text{rail}} + \beta^{\text{mixed}}_{\text{traveltime}} \cdot \text{traveltime}_{\text{rail}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{rail}} + \beta_{\text{headway}} \cdot \text{headway}_{\text{rail}}, \\
V^{(1)}_{\text{SM}} &= \beta^{\text{mixed}}_{\text{traveltime}} \cdot \text{traveltime}_{\text{SM}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{SM}} + \beta_{\text{headway}} \cdot \text{headway}_{\text{SM}}, \\
V^{(2)}_{\text{car}} &= \text{ASC}'_{\text{car}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{car}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{car}}, \\
V^{(2)}_{\text{rail}} &= \text{ASC}'_{\text{rail}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{rail}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{rail}} + \beta_{\text{headway}} \cdot \text{headway}_{\text{rail}}, \\
V^{(2)}_{\text{SM}} &= \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{SM}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{SM}} + \beta_{\text{headway}} \cdot \text{headway}_{\text{SM}}.
\end{aligned}$$

N	R	LL-Bio1	LL-Bio2	Gap (%)	T-Bio1	T-B	T-Bio2
500	1	-219.870	-219.870	0.00	2	0	3
500	5	-219.870	-219.870	0.00	2	0	2
500	10	-219.870	-215.130	2.16	4	0	1
500	20	-219.870	-218.138	0.79	2	2	2
500	50	-219.870	-192.996	12.22	2	9	2
500	100	-219.870	-192.892	12.27	3	24	2
500	500	-219.870	-192.808	12.31	1	211	1
500	1,000	-219.870	-192.802	12.31	1	445	1
1,000	1	-455.687	-455.687	0.00	2	0	2
1,000	5	-455.687	-455.687	0.00	2	0	4
1,000	10	-455.687	-451.570	0.90	2	0	2
1,000	20	-455.687	-383.782	15.78	2	4	3
1,000	50	-455.687	-382.670	16.02	2	24	2
1,000	100	-455.687	-382.289	16.11	2	51	2
1,000	500	-455.687	-382.002	16.17	1	506	1
1,000	1,000	-455.687	-381.962	16.18	2	1,121	1

Table 3: Biogeme without (Bio1) and with (Bio2) BHAMSLE (B) starting point on a latent class logit model with synthetic choices (N = population size, R = number of draws, sLL = simulated log-likelihood, LL = log-likelihood, T = estimation time in seconds)

The total number of parameters to estimate thus increases to nine. In order to examine the effectiveness of BHAMSLE in providing good starting points for estimation, we focus on the estimation results for Biogeme with the standard starting point (Bio1) and the results when using the solution from BHAMSLE as a starting point (Bio2). These results are shown in Table 2. We observe that for smaller values of R, the differences between LL-Bio1 and LL-Bio2 are small. However, as the number of simulation draws increases, the impact of the heuristic becomes more apparent. For samples of size $N = 500$, significant improvements in log-likelihood are achieved using the BHAMSLE starting point, yielding up to 10% better solutions. For $N = 1000$, we still achieve an improvement of around 3%. The number of simulation draws necessary for good estimation results is higher here, with BHAMSLE providing better starting points starting from $R = 500$. It is important to note that it is also possible for BHAMSLE to provide a starting point that is worse than the standard starting point, as seen for example with $N = 500, R = 100$. This likely stems from the fact that small numbers of draws are not enough to efficiently capture the mixed parameter. The differences in the estimated latent class probabilities are again not large, but substantial enough to make a difference. It appears that the true distribution of the two classes lies close to 40% for class 1 vs. 60% for class 2. Using the heuristic as a starting point consistently guides Biogeme towards this improved local optimum. The runtimes for both methods are substantially increased, but the gap between Biogeme and BHAMSLE is smaller than for the latent class model logit model, with Biogeme completing the estimation about 1.5 times faster than the heuristic.

For next two tests we consider a different data set. It is extracted from revealed preference data on mode choice collected in London (Hillel et al., 2018). There are four alternatives available to all individuals: walking, cycling, public transport (pt), and driving. This time, instead of using observed choices, we use synthetic choices: In a pre-processing step, using a separately estimated logit model, every individual in the sample is assigned to either class 1, which represents the base model, or class 2, in which the time-sensitivity parameter $\beta_{\text{traveltime}}$ is divided by a factor of 5 to generate the choice. We therefore estimate a separate travel time sensitivity parameter $\beta'_{\text{traveltime}}$ for that class. The probability to be assigned to class 1 is 70%, and the probability for class 2 is 30%. We investigate which estimation method performs better in discovering these now known latent population segments. The systematic equations for the utilities are:

N	R	Ratio-Bio1	Ratio-B	Ratio-Bio2	(p_1, p_2)-Bio1	(p_1, p_2)-B	(p_1, p_2)-Bio2
500	1	0.67	0.62	0.67	(0.46, 0.54)	(0.50, 0.50)	(0.46, 0.54)
500	5	0.67	0.55	0.67	(0.46, 0.54)	(0.50, 0.50)	(0.46, 0.54)
500	10	0.67	0.64	0.51	(0.46, 0.54)	(0.47, 0.53)	(0.09, 0.91)
500	20	0.67	2.88	2.90	(0.46, 0.54)	(0.55, 0.45)	(0.64, 0.36)
500	50	0.67	3.83	3.47	(0.46, 0.54)	(0.53, 0.47)	(0.65, 0.35)
500	100	0.67	4.57	4.92	(0.46, 0.54)	(0.67, 0.33)	(0.68, 0.32)
500	500	0.67	4.69	4.92	(0.46, 0.54)	(0.67, 0.33)	(0.68, 0.32)
500	1,000	0.67	4.67	4.91	(0.46, 0.54)	(0.67, 0.33)	(0.68, 0.32)
1,000	1	0.84	1.09	0.84	(0.45, 0.55)	(0.50, 0.50)	(0.45, 0.55)
1,000	5	0.84	1.30	0.84	(0.45, 0.55)	(0.50, 0.50)	(0.45, 0.55)
1,000	10	0.84	12.17	0.85	(0.45, 0.55)	(0.53, 0.47)	(0.50, 0.50)
1,000	20	0.84	2.52	2.05	(0.45, 0.55)	(0.53, 0.47)	(0.62, 0.38)
1,000	50	0.84	4.28	4.98	(0.45, 0.55)	(0.58, 0.42)	(0.70, 0.30)
1,000	100	0.84	4.38	4.98	(0.45, 0.55)	(0.63, 0.37)	(0.70, 0.30)
1,000	500	0.84	4.69	4.97	(0.45, 0.55)	(0.70, 0.30)	(0.70, 0.30)
1,000	1,000	0.84	4.63	4.97	(0.45, 0.55)	(0.70, 0.30)	(0.70, 0.30)

Table 4: Ratios and probabilities for Biogeme without (Bio1) and with (Bio2) BHAMSLE (B) starting point on a latent class logit model with synthetic choices (N = population size, R = number of draws, Ratio = $\beta_{\text{traveltime}}\ /\beta'_{\text{traveltime}}$)

$$\begin{aligned}
V^{(1)}_{\text{walking}} &= \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{walking}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{walking}},\\
V^{(1)}_{\text{cycling}} &= \text{ASC}_{\text{cycling}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{cycling}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{cycling}},\\
V^{(1)}_{\text{pt}} &= \text{ASC}_{\text{pt}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{pt}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{pt}},\\
V^{(1)}_{\text{driving}} &= \text{ASC}_{\text{driving}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{driving}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{driving}},\\[1ex]
V^{(2)}_{\text{walking}} &= \beta'_{\text{traveltime}} \cdot \text{traveltime}_{\text{walking}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{walking}},\\
V^{(2)}_{\text{cycling}} &= \text{ASC}_{\text{cycling}} + \beta'_{\text{traveltime}} \cdot \text{traveltime}_{\text{cycling}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{cycling}},\\
V^{(2)}_{\text{pt}} &= \text{ASC}_{\text{pt}} + \beta'_{\text{traveltime}} \cdot \text{traveltime}_{\text{pt}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{pt}},\\
V^{(2)}_{\text{driving}} &= \text{ASC}_{\text{driving}} + \beta'_{\text{traveltime}} \cdot \text{traveltime}_{\text{driving}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{driving}}.
\end{aligned}$$

Thus we have a total of seven parameters to be estimated. We again focus on the impact when using the solution from BHAMSLE as a starting point for Biogeme. The results are presented in Tables 3 and 4. Table 3 shows that the BHAMSLE starting point significantly improves the log-likelihood values starting from $R = 50$, with the gap between LL-Bio1 and LL-Bio2 reaching up to 16%. In terms of estimation times, running the heuristic as a pre-processing step starts being computationally expensive only around $R = 500$ draws. Table 4 highlights that, with the help of BHAMSLE, Biogeme is significantly better at discovering the correct ratio between the two estimated travel time sensitivity parameters. Between 50-100 draws are enough for BHAMSLE to find solutions with a ratio close to the true value 5. For the estimated class membership probabilities (p_1, p_2) we observe similar outcomes: Starting from $R = 50$ draws, BHAMSLE is able to guide Biogeme very closely to the correct distribution of $(0.70, 0.30)$. In contrast, Biogeme without the heuristic starting point struggles to identify the correct proportions, remaining closer to uniform splits.

For the last experiment, we perform a similar altercation to class 1 as in experiment 1, this time replacing β_{cost} by a normally distributed $\beta^{\text{mixed}}_{\text{cost}} = \beta_{\text{cost}} + \beta^{\text{std}}_{\text{cost}} \cdot U_n$, with $U_n \sim \mathcal{N}(0, 1)$, together with adding a third latent class, which is hypothesized to be “lazy”, which in this context means that they do not consider walking or cycling in their choice set. Class 2 remains the same as in the previous experiment. We assign individuals to class 1 with a probability of 50%, class 2 with 30%, and class 3 with 20%. We give the new systematic equations for the utilities of classes 1 and 3 below:

N	R	LL-Bio1	LL-Bio2	Gap (%)	T-Bio1	T-B	T-Bio2
500	1	-249.850	-242.219	3.05	101	0	12
500	5	-244.705	-244.801	-0.04	21	0	19
500	10	-242.226	243.108	-0.36	28	34	25
500	20	-241.122	-238.055	1.27	41	82	38
500	50	-241.074	-241.647	-0.24	69	209	69
500	100	-239.411	-238.177	0.52	169	505	159
500	500	-239.935	-230.889	3.77	674	3012	693
500	1,000	-242.083	-231.634	4.32	1,524	5,153	1,642
500	3,000	-241.139	-230.855	4.26	4,827	12,197	5,141
1,000	1	-511.452	-501.856	1.88	53	0	31
1,000	5	-506.297	-480.593	5.08	26	0	28
1,000	10	-485.990	-488.870	-0.59	33	1	39
1,000	20	-484.787	-485.038	-0.05	69	1	66
1,000	50	-481.808	-482.956	-0.24	167	756	168
1,000	100	-482.799	-479.026	0.78	305	541	362
1,000	500	-490.314	-468.533	4.44	1,240	4,391	1,395
1,000	1,000	-495.295	-473.412	4.42	3,188	7,522	2,812
1,000	3,000	-493.228	-470.436	4.62	6,862	13,756	8,756

Table 5: Biogeme without (Bio1) and with (Bio2) BHAMSLE (B) starting point on a latent class mixed logit model with synthetic choices (N = population size, R = number of draws, sLL = simulated log-likelihood, LL = log-likelihood, T = estimation time in seconds)

$$
\begin{aligned}
V^{(1)}_{\text{walking}} &= \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{walking}} + \beta^{\text{mixed}}_{\text{cost}} \cdot \text{cost}_{\text{walking}},\\
V^{(1)}_{\text{cycling}} &= \text{ASC}_{\text{cycling}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{cycling}} + \beta^{\text{mixed}}_{\text{cost}} \cdot \text{cost}_{\text{cycling}},\\
V^{(1)}_{\text{pt}} &= \text{ASC}_{\text{pt}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{pt}} + \beta^{\text{mixed}}_{\text{cost}} \cdot \text{cost}_{\text{pt}},\\
V^{(1)}_{\text{driving}} &= \text{ASC}_{\text{driving}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{driving}} + \beta^{\text{mixed}}_{\text{cost}} \cdot \text{cost}_{\text{driving}},\\[1ex]
V^{(3)}_{\text{pt}} &= \text{ASC}_{\text{pt}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{pt}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{pt}},\\
V^{(3)}_{\text{driving}} &= \text{ASC}_{\text{driving}} + \beta_{\text{traveltime}} \cdot \text{traveltime}_{\text{driving}} + \beta_{\text{cost}} \cdot \text{cost}_{\text{driving}}.
\end{aligned}
$$

Thus we have a total of nine parameters to be estimated. The results are presented in Tables 5 and 6. Similarly to the second experiment, Table 5 reveals that in order to find a good starting point for latent class mixed logit, a higher number of draws is necessary. In this case, starting from $R = 500$ draws we observe significant improvements (up to 4.5%) in log-likelihood for Biogeme with the BHAMSLE starting point. For this amount of draws, the estimation time is proportionately high, with BHAMSLE on average taking more than twice the amount of time compared to Biogeme when estimating the model with $N = R = 1,000$. Table 6 presents the estimated class membership probabilities for each method. The expected probabilities are (0.50, 0.30, 0.20) based on the synthetic choice generation. Both methods struggle to recreate these proportions exactly, but for high enough numbers of draws ($R \geq 500$) BHAMSLE consistently provides improvements on Biogeme in terms of detecting the true segmentation. For example, at $N = 1,000$ and $R = 3,000$, Biogeme without a good starting point estimates the probabilities as (0.26, 0.63, 0.11), compared to (0.50, 0.31, 0.19) when guided by BHAMSLE, which aligns closely with the expected values.

4 Conclusions

This work aims to enhance the estimation of advanced discrete choice models (DCMs) by introducing BHAMSLE, a Breakpoint Heuristic Algorithm for Maximum Simulated Likelihood Estimation. By adapting the principles of the Breakpoint Heuristic Algorithm, originally developed for choice-based pricing, BHAMSLE leverages decision-making breakpoints to systematically explore local

N	R	(p_1, p_2, p_3)-Bio1	(p_1, p_2, p_3)-B	(p_1, p_2, p_3)-Bio2
500	1	(0.46, 0.42, 0.12)	(0.33, 0.33, 0.33)	(0.51, 0.32, 0.17)
500	5	(0.50, 0.37, 0.13)	(0.33, 0.33, 0.33)	(0.49, 0.37, 0.14)
500	10	(0.51, 0.36, 0.13)	(0.33, 0.33, 0.33)	(0.50, 0.36, 0.14)
500	20	(0.52, 0.34, 0.15)	(0.33, 0.33, 0.33)	(0.52, 0.30, 0.18)
500	50	(0.43, 0.53, 0.04)	(0.37, 0.32, 0.32)	(0.49, 0.34, 0.17)
500	100	(0.32, 0.54, 0.14)	(0.34, 0.34, 0.32)	(0.49, 0.31, 0.20)
500	500	(0.31, 0.56, 0.13)	(0.38, 0.30, 0.32)	(0.50, 0.32, 0.18)
500	1,000	(0.30, 0.58, 0.12)	(0.42, 0.31, 0.27)	(0.52, 0.32, 0.16)
500	3,000	(0.31, 0.57, 0.12)	(0.45, 0.31, 0.24)	(0.51, 0.33, 0.16)
1,000	1	(0.44, 0.46, 0.13)	(0.33, 0.33, 0.33)	(0.52, 0.28, 0.20)
1,000	5	(0.46, 0.41, 0.13)	(0.33, 0.33, 0.33)	(0.51, 0.30, 0.18)
1,000	10	(0.46, 0.42, 0.12)	(0.33, 0.33, 0.33)	(0.53, 0.27, 0.19)
1,000	20	(0.50, 0.30, 0.19)	(0.33, 0.33, 0.33)	(0.51, 0.30, 0.19)
1,000	50	(0.34, 0.51, 0.15)	(0.31, 0.40, 0.29)	(0.49, 0.31, 0.20)
1,000	100	(0.33, 0.52, 0.14)	(0.36, 0.33, 0.31)	(0.49, 0.32, 0.19)
1,000	500	(0.16, 0.79, 0.05)	(0.46, 0.31, 0.23)	(0.49, 0.34, 0.20)
1,000	1,000	(0.25, 0.65, 0.10)	(0.43, 0.34, 0.22)	(0.48, 0.32, 0.20)
1,000	3,000	(0.26, 0.63, 0.11)	(0.46, 0.32, 0.22)	(0.50, 0.31, 0.19)

Table 6: Probabilities for Biogeme without (Bio1) and with (Bio2) BHAMSLE (B) starting point on a latent class logit model with synthetic choices (N = population size, R = number of draws)

optima, bridging the gap between choice-based optimization and choice model estimation, and providing a novel method for robust initialization for complex estimation problems. We demonstrate through numerical experiments that BHAMSLE is effective in improving log-likelihood values for both latent class logit and latent class mixed logit models, achieving gains of up to 10% for observed choices and up to 16% for synthetic choices. Additionally, the results highlighted the heuristic's ability to recover latent population segments, even in complex scenarios involving mixed parameters and restricted choice sets. While the current state-of-the-art software, PandasBiogeme, remains computationally faster for standard initializations, BHAMSLE offers a significant advantage by reducing the need for costly random re-initialization, particularly in settings where achieving an accurate fit is crucial. The proposed method is general and can be applied to any DCM. Future research should thus focus on extending the application of BHAMSLE to more complex DCMs, exploring its performance under different model specifications and real-world datasets, and integrating it with parallelization techniques to further enhance computational efficiency.

References

Bierlaire, M. (2023). *A short introduction to biogeme* (Technical Report No. TRANSP-OR 230620). Lausanne, Switzerland: Transport and Mobility Laboratory, Ecole Polytechnique Fédérale de Lausanne.

Bierlaire, M., Axhausen, K., & Abay, G. (2001). The acceptance of modal innovation: The case of swissmetro. In *Swiss transport research conference.*

Fernandez Antolin, A. (2018). *Dealing with correlations in discrete choice models* (Unpublished doctoral dissertation). Ecole Polytechnique Fédérale de Lausanne, Switzerland.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Machine learning basics. *Deep learning*, *1*(7), 98–164.

Haering, T., Torres, R., Legault, Fabian, & Bierlaire, M. (2024). *Heuristics and exact algorithms for choice-based capacitated and uncapacitated continuous pricing* (Technical Report No. TRANSP-OR 240918). Lausanne, Switzerland: Transport and Mobility Laboratory, Ecole Polytechnique Fédérale de Lausanne.

Hauschild, T., & Jentschel, M. (2001). Comparison of maximum likelihood estimation and chi-square statistics applied to counting experiments. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, *457*(1-2), 384–401.

Hillel, T., Elshafie, M. Z., & Jin, Y. (2018). Recreating passenger mode choice-sets for transport simulation: A case study of london, uk. *Proceedings of the Institution of Civil Engineers-Smart Infrastructure and Construction*, *171*(1), 29–42.

Jung, T., & Wickrama, K. A. (2008). An introduction to latent class growth analysis and growth mixture modeling. *Social and personality psychology compass*, *2*(1), 302–317.

Peer, S., Knockaert, J., & Verhoef, E. T. (2016). Train commuters' scheduling preferences: Evidence from a large-scale peak avoidance experiment. *Transportation Research Part B: Methodological*, *83*, 314–333.

Train, K. (2003). *Discrete choice methods with simulation.* Cambridge University Press. (http://emlab.berkeley.edu/books/choice.html)