# Estimating the MDCEV model with Biogeme

Michel Bierlaire Mengyi Wang

April 30, 2024

DRAFT Report TRANSP-OR 24xxxx
Transport and Mobility Laboratory
School of Architecture, Civil and Environmental Engineering
Ecole Polytechnique Fédérale de Lausanne
transp-or.epfl.ch

# Abstract

The abstract...

# 1 The MDCEV model

The multiple discrete-continuous extreme value model (MDCEV) is a choice model where the choice of multiple alternatives can occur simultaneously. It has been introduced by Bhat (2005), building on the Karush-Kuhn-Tucker multiple discrete-continuous economic model proposed by Wales and Woodland (1983). In this document, we introduce a generalization of the model, where the derivation is performed for a generic utility function. This is motivated by the need to obtain an implementation that is easily extendible to new models in the future.

Consider an individual, denoted as $n$, who is presented with a distinct set of items, represented as $\mathcal{C}_n$, containing $J_n$ items in total. Given a total budget of $E_n$, this individual decides on purchasing a quantity $y_{in}$ of each item. This decision must verify the following budget constraint:

$$\sum_{i \in \mathcal{C}_n} e_{in} = \sum_{i \in \mathcal{C}_n} p_{in} y_{in} \leq E_n,$$

where:

- $p_{in} > 0$ is the price per unit of item $i$ for the individual $n$.
- $e_{in} = p_{in} y_{in}$ represents the total expenditure by individual $n$ on item $i$.

We assume that at least one item is consumed. Note that it is mathematically equivalent to write the problem in terms of expenditures $e$ or in terms of quantities $y$. We adopt the former.

Each item $i$ is associated with a utility $U_{in}(x_{in}, e_{in}, \varepsilon_{in}; \theta)$, where $x_{in}$ are explanatory variables, $e_{in}$ is the expenditure of individual $n$ for alternative $i$, $\varepsilon_{in}$ is an error term independent from $x_{in}$ and $e_{in}$, and $\theta$ is a vector of parameters, to be estimated from data. We identify two properties that the utility functions must have. Both are relatively weak conditions. The first one is related to the optimality conditions, and are verified if the functions are concave, or quasi-concave. The second one is related to the specification of the error terms, needed for the derivation of the econometric model. It is assumed that there exists an order preserving function $\phi$ such that the random utility can be expressed in additive form, that is such that

$$\phi\left(\frac{\partial U_{in}}{\partial e_{in}}\right) = V_{in} + \varepsilon_{in}, \tag{1}$$

where $V_{in}$ is the deterministic part, and $\varepsilon_{in}$ is the error term. On top of this, the specification of the utility functions should be associated with specific behavioral assumptions. This aspect is not discussed in this document, in order to keep the model as general as possible.

The expenditure decisions $e_{1n}, \ldots, e_{J_n,n}$ are assumed to be the solution of the following optimization problem:

$$\max_{e_n} \sum_{i \in \mathcal{C}_n} U_{in}(x_{in}, e_{in}, \varepsilon_{in}; \theta) \tag{2}$$

subject to

$$\sum_{i \in \mathcal{C}_n} e_{in} \leq E_n, \tag{3}$$

$$e_{in} \geq 0. \tag{4}$$

The optimality conditions can be derived from the Lagrangian of the problem. We introduce a Lagrange multiplier $\lambda \geq 0$ associated with constraint (3) and a Lagrange multiplier $\mu_i \geq 0$ for each constraint (4). The Lagrangian is defined as

$$\mathcal{L}(e; \lambda, \mu) = -\sum_{i \in \mathcal{C}_n} U_{in} + \lambda \left( \sum_{i \in \mathcal{C}_n} e_i - E_n \right) - \sum_{i \in \mathcal{C}_n} \mu_i e_i. \tag{5}$$

The first order optimality conditions state that

$$\frac{\partial \mathcal{L}}{\partial e_i} = -\frac{\partial U_{in}}{\partial e_i} + \lambda - \mu_i = 0, \text{ and } \mu_i e_i = 0, \ \forall i \in \mathcal{C}_n. \tag{6}$$

Note that we assume that the second order optimality conditions are also verified. This is the case if the utility functions are concave, for instance.

At least one item is consumed. We assume without loss of generality that it is item 1. Therefore $e_{1n} > 0$ and $\mu_1 = 0$. Consequently, (6) can be written

$$\lambda = \frac{\partial U_{1n}}{\partial e_1}. \tag{7}$$

Consider a chosen alternative $i \neq 1$ such that $e_{in} > 0$. Using the same argument, we can write

$$\frac{\partial U_{in}}{\partial e_i} = \lambda = \frac{\partial U_{1n}}{\partial e_1}. \tag{8}$$

Therefore, if the dual variable $\lambda$ is known, the optimal expenditure for a chosen alternative $i$ can be obtained by solving Equation (8), analytically, or numerically. This quantity is denoted

$$e_{in}(x_{in}; \lambda, \theta), \tag{9}$$

and is such that

$$\frac{\partial U_{in}}{\partial e_i}\left(e_{in}(x_{in};\lambda,\theta)\right)=\lambda.$$

Consider an alternative $i$ such that $e_{in}=0$. In this case, $\mu_i=\lambda-\frac{\partial U_{in}}{\partial e_i}\geq 0$ and

$$\frac{\partial U_{in}}{\partial e_i}\leq\lambda=\frac{\partial U_{1n}}{\partial e_1}. \tag{10}$$

Note that we can transform the utility functions $U_{in}$ with any order preserving function $F$ without changing the solution of the problem. An order preserving function is a strictly increasing function $F$ of one variable such that $F'(u)>0$. In that case,

$$\phi\left(\frac{\partial U_{in}}{\partial e_i}\right)\leq\phi\left(\frac{\partial U_{1n}}{\partial e_1}\right)\Longleftrightarrow\frac{\partial U_{in}}{\partial e_i}\leq\frac{\partial U_{1n}}{\partial e_1}. \tag{11}$$

As the analyst is not able to observe the actual utility function, we assume that the utility function is a random variable. More specifically, we assume that there exists an order preserving transform $\phi$ of the utility functions such that

$$\phi\left(\frac{\partial U_{in}}{\partial e_i}\right)=V_{in}+\varepsilon_{in}, \tag{12}$$

where $V_{in}$ is the deterministic part, and $\varepsilon_{in}$ is a random disturbance. Therefore, the optimality conditions can be written as

$$V_{in}+\varepsilon_{in}=V_{1n}+\varepsilon_{1n},\qquad\text{if } e_{in}>0, \tag{13}$$

$$V_{in}+\varepsilon_{in}\leq V_{1n}+\varepsilon_{1n},\qquad\text{if } e_{in}=0. \tag{14}$$

We assume that the utility functions are defined in such a way that (13) defines a bijective relationship between $e_{in}$ and $\varepsilon_{in}$, for all $i\in\mathcal{C}_n$.

The above model is typically used in two modalities. For the estimation of the unknown parameters, the analyst needs the distribution of the expenditures given by the model in order to calculate the log-likelihood function for the observed expenditures. This distribution and the corresponding log-likelihood function are derived in Section 2. For the application of the model, the parameters are known, and the expenditures must be forecast. The procedure to calculate this forecast is described in Section 3.

# 2 Estimation of unknown parameters

We assume that we have observed a sample of individuals. For each individual $n$, we have access to the explanatory variables and the expenditures for each alternative: $x_{in}$, $e_{in}$, for $i\in\mathcal{C}_n$. The objective is to infer the value of the unknown parameters $\theta$ from this sample.

In order to calculate the (log)-likelihood of observed expenditures, we are interested in the distribution of the vector $e_n$ provided by the model. We have established that it is a function of the vector of disturbances $\varepsilon_n$: $e_n = H(\varepsilon_n)$. Consequently, if we assume a distribution for $\varepsilon_n$, characterized by a probability density function (pdf) $f_\varepsilon$ and a cumulative distribution function (CDF) $F_\varepsilon$, we can characterize the distribution of $e_n$.

We start by assuming that $e_{1n}$, the consumed quantity of item 1 is known and non zero. Consequently, the value of $\varepsilon_{1n}$ is known as well. In order to derive the pdf evaluated at $e$, we split the vector $e$ into its positive entries $e^+$ and its zero entries $e^0$, alternative 1 being excluded. In an analogous way, we denote $\mathcal{C}^+$ and $\mathcal{C}^0$ the corresponding sets of indices, of size $J^+$ and $J^0$, respectively.

For each $i \in \mathcal{C}^+$, we can use (13) to define $\varepsilon_n = H^{-1}(e_n)$, where $H^{-1}: \mathbb{R}^{J^+-1} \to \mathbb{R}^{J^+-1}$ is defined as

$$H_i^{-1}(e) = \varepsilon_{i+1,n} = V_{1n}(e_1) - V_{i+1,n}(e_{i+1}) + \varepsilon_{1n}. \tag{15}$$

Therefore, the density function can be decomposed as

$$\begin{aligned} f_e(e^+, e^0|e_1) &= f_e(e^+|e^0, e_1)\,\Pr(e^0|e_1) \\ &= f_\varepsilon(\varepsilon^+|\varepsilon^0, \varepsilon_1)\det\left(\frac{\partial H^{-1}}{\partial e}\right)\Pr(e^0|e_1), \end{aligned}$$

where

$$\begin{aligned} \Pr(e^0|e_1) &= \Pr(V_{in} + \varepsilon_{in} \le V_{1n} + \varepsilon_{1n},\ \forall i \in \mathcal{C}^0) \\ &= \Pr(\varepsilon_{in} \le V_{1n} - V_{in} + \varepsilon_{1n},\ \forall i \in \mathcal{C}^0) \\ &= F_\varepsilon(\varepsilon_{1n}, 1, \ldots, 1, (V_{1n} - V_{in} + \varepsilon_{1n})_{i\in\mathcal{C}^0}). \end{aligned}$$

From (15), we can calculate the entries of the Jacobian $\partial H^{-1}/\partial e$. Indeed,

$$\begin{aligned} \frac{\partial H_i^{-1}(e)}{\partial e_k} &= \frac{\partial V_{1n}}{\partial e_1}\frac{\partial e_1}{\partial e_k} && \text{if } k \neq i+1, \\ &= \frac{\partial V_{1n}}{\partial e_1}\frac{\partial e_1}{\partial e_k} - \frac{\partial V_{i+1,n}}{\partial e_{i+1}} && \text{if } k = i+1. \end{aligned}$$

From (3), we have

$$e_1 = E - \sum_{j\neq 1} e_j,$$

so that

$$\frac{\partial e_1}{\partial e_k} = -1.$$

Consequently,

$$
\begin{aligned}
\frac{\partial H_i^{-1}(e)}{\partial e_k} &= -\frac{\partial V_{1n}}{\partial e_1} && \text{if } k \neq i+1, \\
&= -\frac{\partial V_{1n}}{\partial e_1} - \frac{\partial V_{i+1,n}}{\partial e_{i+1}} && \text{if } k = i+1.
\end{aligned}
$$

If we denote

$$
c_i = -\frac{\partial V_{in}}{\partial e_i}, \tag{16}
$$

the Jacobian has the following structure:

$$
\partial H^{-1}/\partial e = \begin{pmatrix} c_1 + c_2 & c_1 & \cdots & c_1 \\ c_1 & c_1 + c_3 & \cdots & c_1 \\ & & \vdots & \\ c_1 & c_1 & \cdots & c_1 + c_{J_n} \end{pmatrix}.
$$

Therefore, the determinant is equal to

$$
\det(\partial H^{-1}/\partial e) = \left( \prod_{i=1}^{J^+} c_i \right) \left( \sum_{i=1}^{J^+} \frac{1}{c_i} \right).
$$

Note that this determinant depends only on the utility function, not on the distribution of the $\varepsilon_n$.

Therefore, the density function of the expenditures is given by

$$
f_e(e^+, e^0) = \left( \prod_{i=1}^{J^+} c_i \right) \left( \sum_{i=1}^{J^+} \frac{1}{c_i} \right) \int_{\varepsilon_1=-\infty}^{+\infty} f_\varepsilon(\varepsilon^+ | \varepsilon^0, \varepsilon_1) F_\varepsilon(\varepsilon_{1n}, \ldots) f_\varepsilon(\varepsilon_1) d\varepsilon_1, \tag{17}
$$

where $f_\varepsilon(\varepsilon_1)$ is the marginal distribution of $\varepsilon_1$, and

$$
F_\varepsilon(\varepsilon_{1n}, \ldots) = F_\varepsilon(\varepsilon_{1n}, 1, \ldots, 1, (V_{1n} - V_{in} + \varepsilon_{1n})_{i \in \mathcal{C}^0}).
$$

Equation (17) corresponds to Equation 11 in Bhat (2008).

If we assume that the $\varepsilon_{in}$ are independent, we obtain the MDCEV model introduced by Bhat (2005). In that case, the density (17) is

$$
f_e(e^+, e^0) = \mu^{J^+ - 1} \left( \prod_{i=1}^{J^+} c_i \right) \left( \sum_{i=1}^{J^+} \frac{1}{c_i} \right) \left( \frac{\prod_{i \in \mathcal{C}^+} e^{\mu V_{in}}}{(\sum_{i \in \mathcal{C}_n} e^{\mu V_{in}})^{J^+}} \right) (J^+ - 1)!, \tag{18}
$$

where the derivation is available in Bhat (2008). Therefore, the contribution of observation $n$ to the log likelihood is

$$
\begin{aligned}
\ln f_e(e^+, e^0) = & (J^+ - 1) \ln \mu \\
& + \sum_{i=1}^{J^+} \ln c_i \\
& + \ln \left( \sum_{i=1}^{J^+} \frac{1}{c_i} \right) \\
& + \mu \sum_{i \in \mathcal{C}^+} V_{in} \\
& - J^+ \ln \sum_{i \in \mathcal{C}_n} e^{\mu V_{in}} \\
& + \ln (J^+ - 1)!.
\end{aligned}
$$

Note that the last term is a constant, and is ignored by Biogeme.

# 3 Forecasting

We assume that we have a sample of individuals, either directly observed, or coming from a scenario. For each individual $n$, we have access to the explanatory variables for each alternative: $x_{in}$, for $i \in \mathcal{C}_n$. We also assume that an estimation of the unknown parameters $\theta$ is available. The objective is to forecast the distribution of expenditures $e_n$ for each individual $n$ in the sample.

This involves generating R draws $\varepsilon_i^r$, $r = 1, \ldots, R$, from the error terms $\varepsilon_i$ and, for each draw $r$, solving the optimization problem defined in equations (2)–(4):

$$
\max_e \sum_{i \in \mathcal{C}} U_i(x_i, e_i, \varepsilon_i^r; \theta)
$$

subject to

$$
\begin{aligned}
\sum_{i \in \mathcal{C}} e_i & \leq E, \\
e_i & \geq 0.
\end{aligned}
$$

For the sake of notational clarity, we have dropped the index $n$ representing the individual. We denote by $e_r^*$ and $\lambda_r^*$ the optimal values of the expenditures and the dual variable, respectively. The distribution of $e$ is approximated by the empirical distribution of $e_r^*$, $r = 1, \ldots, R$. In the rest of this section, we also drop the index $r$

for notational simplicity. It is sufficient to keep in mind that $\varepsilon_i$ must be interpreted as a value, a draw from the distribution.

In Biogeme, solving the optimization problem is implemented in two ways. The *"brute force"* algorithm uses a generic solver from the package `scipy`. It does not exploit any information about the model specification. The second algorithm, the *analytical* algorithm, is inspired by Pinjari and Bhat (2021), and exploits some properties of the utility function in order to identify the optimal solution. We describe it here.

## 3.1 Properties

We first define $W_i$ as the derivative of the utility of alternative $i$ evaluated at zero expenditure:

$$W_i = \frac{\partial U_i}{\partial e_i}(e_i = 0). \tag{19}$$

It plays a central role in the algorithm.

Then, we assume that the model verifies the following properties, where $\mathcal{C}^* = \{i | e_i^* > 0\}$ is the set of alternatives that are chosen at the optimal solution, that is, all alternatives such that the optimal expenditure is non zero.

**Property 1.** *For each chosen alternative $i \in \mathcal{C}^*$, we have*

$$\lambda^* < W_i.$$

**Property 2.** *For each chosen alternative $i \in \mathcal{C}^*$, the optimal expenditure (9) is a decreasing function of $\lambda$:*

$$\frac{\partial e_i}{\partial \lambda} < 0.$$

**Property 3.** *For each chosen alternative $i \in \mathcal{C}^*$, there exists a lower bound $\lambda_i^\ell \geq 0$ such that*

$$\lambda_i^\ell \leq \lambda^*,$$

*and $e_{in}(x_{in}; \lambda, \theta)$ is well defined, and non negative for each $\lambda$ such that*

$$\lambda_i^\ell \leq \lambda \leq W_i.$$

Note that property 2 is verified if the utility function is strictly concave. Indeed, the sign of $\frac{\partial e_i}{\partial \lambda}$ is the same as the sign of $\frac{\partial \lambda}{\partial e_i}$. And, from the optimality condition (8),

$$\frac{\partial \lambda}{\partial e_i} = \frac{\partial^2 U_{in}}{\partial e_i^2}.$$

As discussed later, all models implemented in the current version of Biogeme have those properties. Therefore, the bisection algorithm can be applied.

We first derive some immediate corollaries of the above properties that are exploited in the design of the algorithm.

**Corollary 1.** *Consider a chosen alternative* $i \in \mathcal{C}^*$ *and a non chosen alternative* $j \notin \mathcal{C}^*$*. Then,*

$$W_i > \lambda^* \geq W_j. \tag{20}$$

*Proof.* Consider an unchosen alternative $j$. From the optimality condition (10), we have

$$\lambda^* \geq \frac{\partial U_j}{\partial e_j}(e_j^*) = \frac{\partial U_j}{\partial e_j}(e_j = 0) = W_j.$$

The result follows from Property 1. □

**Corollary 2.** *The total expenditure*

$$E(\lambda) = \sum_{i \in \mathcal{C}^*} e_i(\lambda)$$

*is a decreasing function of* $\lambda$*:*

$$\frac{\partial E(\lambda)}{\partial \lambda} < 0.$$

*At the optimal value* $\lambda^*$*, the constraints are verified and*

$$E(\lambda^*) = \sum_{i \in \mathcal{C}^*} e_i(\lambda^*) = E. \tag{21}$$

*In particular, it means that*

$$\lambda < \lambda^* \iff E(\lambda) > E,$$

*and*

$$\lambda > \lambda^* \iff E(\lambda) < E.$$

*Proof.* This is an immediate consequence of Property 2. □

The next result provides a condition to verify if an alternative is in the optimal choice set or not.

**Lemma 1.** *Let's assume that the numbering of the alternatives is organized in decreasing order of* $W_i$*, and that we have already established that alternatives* $1, \ldots, M$ *are chosen. We define*

$$\lambda_{\mathcal{C}}^{\ell} = \max_{i=1,\ldots,M} \lambda_i^{\ell},$$

*where* $\lambda_i^{\ell}$ *is the lower bound defined by Property 3. Then, alternative* $M+1$ *is chosen in the optimal solution if and only if* $\lambda_{\mathcal{C}}^{\ell} \leq W_{M+1}$ *and* $E(W_{M+1}) < E$*.*

*Proof.* First, note that the condition $\lambda_{\mathcal{C}}^{\ell} \leq W_{M+1}$ guarantees that $E(W_{M+1})$ is well defined and non negative. Using Equation (21), $E(W_{M+1}) < E$ is equivalent to $E(W_{M+1}) < E(\lambda^*)$. From Corollary 2, it is equivalent to establish that alternative $M+1$ is chosen in the optimal solution if and only if $W_{M+1} \geq \lambda_{\mathcal{C}}^{\ell}$ and $W_{M+1} > \lambda^*$. To show that, we consider both possibilities: whether $M+1$ is included in the choice set or not.

- If $M+1$ is chosen in the optimal solution, Property 1 guarantees that

  $$\lambda^* < W_{M+1},$$

  and Property 3 guarantees that $\lambda_{\mathcal{C}}^{\ell} \leq \lambda^*$ (as all alternatives in $\mathcal{C}$ are chosen), so that

  $$\lambda_{\mathcal{C}}^{\ell} < W_{M+1},$$

  thereby confirming the sufficient condition.

- For the necessary condition, we prove the contrapositive: if $M+1$ is not chosen in the optimal solution, than, $W_{M+1} < \lambda_{\mathcal{C}}^{\ell}$ or $W_{M+1} \leq \lambda^*$. If $M+1$ is not chosen, then $e_{M+1}^* = 0$ is optimal. From the optimality condition (10),

  $$\lambda^* \geq \frac{\partial U_{M+1}}{\partial e_{M+1}}(e_{M+1} = 0) = W_{M+1},$$

  which is true irrespectively if $W_{M+1} < \lambda_{M+1}^{\ell}$ is true or not.

□

## 3.2 The analytical algorithm

The algorithm consists of two phases:

- First, mathematical properties of the specific MDCEV model are exploited in order to identify the optimal set of chosen alternatives.
- Once the set of chosen alternatives has been identified, the optimal value of the dual variable is calculated using a bisection method.

### Identification of chosen alternatives

In order to identify the chosen alternatives, Corollary 1 suggests to sort the alternatives in decreasing order of $W_i$. For each alternative in this sequence, we check whether it is chosen using Lemma 1. If it is, we proceed to the next one; if not, we have identified all chosen alternatives, and we can terminate the process. It is described in Algorithm 1, where we use $\lambda_0^{\ell} = 0$.

**Algorithm 1:** Identification of the chosen alternatives

**Input:** $E$, $W_i$, $i = 1, \ldots, J$, such that $W_j > W_{j+1}$. $i = 1, \ldots, J-1$.
**Output:** Optimal choice set $\mathcal{C}^*$.

$M \leftarrow 0$.
$\lambda_\ell \leftarrow 0$.
**while** $\lambda^\ell \leq W_{M+1}$ *and* $E(W_{M+1}) < E$ **do**
    $M \leftarrow M + 1$,
    $\lambda_\ell \leftarrow \max(\lambda_\ell, \lambda^\ell_M)$.
$\mathcal{C}^* = \{1, \ldots, M\}$ if $M > 0$, $\emptyset$ otherwise.

**Bisection algorithm**

Let $\mathcal{C}^* = \{1, \ldots, M\}$ be the set of chosen alternatives, where the alternatives are numbered in decreasing order of $W_i$, which is assumed not empty, without loss of generality. The bisection algorithm consists in evaluating the optimal value of the dual variable $\lambda^*$ by updating a lower and an upper bound. The initialization is based on Property 3 and (20). Indeed, as alternative $M$ is chosen, $W_M$ is an upper bound on $\lambda^*$. If all alternatives are chosen, that is, if $M = J$, we use Property 3 and define

$$\lambda^\ell_{\mathcal{C}^*} = \max_{i \in \mathcal{C}^*} \lambda^\ell_M$$

as a lower bound. If $M < J$, (20) suggests also $W_{M+1}$ as a lower bound. Therefore, we use the best one:

$$\lambda_\ell = \begin{cases} \max(\lambda^\ell_{\mathcal{C}^*}, W_{M+1}) & \text{if } M < J, \\ \lambda^\ell_{\mathcal{C}^*} & \text{otherwise,} \end{cases}$$

and

$$\lambda_u = W_M.$$

The algorithm then exploits Corollary 2 to update either the lower or the upper bound. Indeed, if $E(\widehat{\lambda}) < E$, it means that $\widehat{\lambda} > \lambda^*$, and we have a better upper bound. Similarly, if $E(\widehat{\lambda}) > E$, it means that $\widehat{\lambda} < \lambda^*$, and we have a better lower bound. Once we have a sufficiently precise approximation of $\lambda^*$, we use (9) to obtain the optimal expenditures.

# 4 Model specifications

The theory above described so far is generic. In order to obtain an operational model, utility function and the order preserving function $\phi$ used in (1) must be

**Algorithm 2:** Find Optimal $\lambda^*$

**Input:** $E, \mathcal{C}^* = \{1, \ldots, M\}$, $\delta_\lambda$, $\delta_E$.
**Output:** $\lambda^*$.
**Output:** $\lambda^*$.

$\lambda_\ell \leftarrow \max(\lambda^\ell_M, W_{M+1})$, where $W_{M+1} = 0$ if $M = J$.
$\lambda_u \leftarrow W_M$.
**repeat**
    $\widehat{\lambda} = (\lambda_\ell + \lambda_u)/2$.
    **if** $E(\widehat{\lambda}) < E$ **then**
        $\lambda_u \leftarrow \widehat{\lambda}$,
    **else if** $E(\widehat{\lambda}) > E$ **then**
        $\lambda_\ell \leftarrow \widehat{\lambda}$.
**until** $\lambda_u - \lambda_\ell \leq \delta_\lambda$ ***or*** $\left|E(\widehat{\lambda}) - E\right| \leq \delta_E$.
$\lambda^* = (\lambda_\ell + \lambda_u)/2$.

specified. In terms of implementation, Biogeme uses the following functions for the estimation of the parameter:

- a function that generates the Biogeme expression for the transformed utility $V_{in}$, and,
- a function that generates the Biogeme expression for the entries of the Jacobian $c_{in}$ ( actually, one function calculates their logarithm $\ln c_{in}$, and another one their inverse, $1/c_{in}$).

For forecasting, it is needed to first check if each specification comply with Properties 1–3. If so, the analytical algorithm can be used. It requires the implementation of the following functions:

- a function calculating the value of the utility function $U_{in}$,
- a function calculating calculating the derivatives of the utility functions $\partial U_{in}/\partial e_i$,
- a function calculating the optimal expenditure (9) as a function of the dual variable.

We now develop these aspects for each model implemented in Biogeme.

## 4.1 Translated utility function

In the context of a time use model, Bhat (2005) uses the translated utility function introduced by Kim et al. (2002, Eq. (1)):

$$U_{in}(e_i) = \exp(\beta^T x_{in} + \varepsilon_{in})(e_i + \gamma_i)^{\alpha_i}, \tag{22}$$

where $0 < \alpha_i < 1$ and $\gamma_i > 0$ are parameters to be estimated. If there is an outside good $k$, the value of $\gamma_k$ is set to zero. Note that there is no price involved in this specification, as it models time and not goods. It is equivalent to set $p_i = 1$, for all $i$ in the model described above. Note also that, Kim et al. (2002) and Bhat (2005) impose the following restriction on $\alpha_i$: $0 < \alpha_i \leq 1$. However, in the context of this implementation, $\alpha_i = 1$ would create a singularity.

In Biogeme, this model is referred to as `translated`.

### First order conditions

We can calculate

$$\frac{\partial U_{in}}{\partial e_i} = \exp(\beta^T x_{in} + \varepsilon_{in})\alpha_i(e_i + \gamma_i)^{\alpha_i - 1}. \tag{23}$$

Evaluated at zero expenditure, we obtain

$$W_{in} = \exp(\beta^T x_{in} + \varepsilon_{in})\alpha_i\gamma_i^{\alpha_i - 1}. \tag{24}$$

For forecasting purposes, as described in Section 3, we need to obtain $e_i^*(\lambda)$, the solution of the equation

$$\lambda = \frac{\partial U_{in}}{\partial e_i}.$$

Here, we have

$$e_i^*(\lambda) = \left(\frac{\lambda}{\exp(\beta^T x_{in} + \varepsilon_{in})\alpha_i}\right)^{\frac{1}{\alpha_i - 1}} - \gamma_i \tag{25}$$

### Transformed utility

In this context, we use the logarithm as the order preserving function to obtain the following specification:

$$\phi\left(\frac{\partial U_{in}}{\partial e_i}\right) = \beta^T x_{in} + \varepsilon_{in} + \ln \alpha_i + (\alpha_i - 1)\ln(e_i + \gamma_i),$$

so that

$$V_{in} = \beta^T x_{in} + \ln \alpha_i + (\alpha_i - 1)\ln(e_i + \gamma_i),$$

**Entries of the Jacobian**

We have

$$c_{in} = -\frac{\partial V_{in}}{\partial e_i} = \frac{1-\alpha_i}{e_i+\gamma_i},$$

so that

$$\ln c_{in} = \ln(1-\alpha_i) - \ln(e_i+\gamma_i),$$

and

$$\frac{1}{c_{in}} = \frac{e_i+\gamma_i}{1-\alpha_i}.$$

**Properties**

1. In order to show that Property 1 is verified, we consider

$$\ln(W_{in}) = \beta^T x_{in} + \varepsilon_{in} + \ln(\alpha_i) + (\alpha_i - 1)\ln\gamma_i.$$

Consider a chosen alternative $i$. From the optimality KKT conditions, we have that

$$\begin{aligned}
\ln\lambda^* &= \beta^T x_{in} + \varepsilon_{in} + \ln\alpha_i + (\alpha_i-1)\ln(e_{in}+\gamma_i)\\
&= \beta^T x_{in} + \varepsilon_{in} + \ln\alpha_i + (\alpha_i-1)\ln(\frac{e_{in}}{\gamma_i}+1) + (\alpha_i-1)\ln\gamma_i\\
&= W_{in} + (\alpha_i-1)\ln(\frac{e_i}{\gamma_i}+1),
\end{aligned}$$

where $\lambda^*$ is the optimal dual variable associated with the budget constraint. Property 1 follows from the fact that

$$(\alpha_i-1)\ln(\frac{e_i}{\gamma_i}+1) < 0.$$

2. For Property 2, as discussed earlier, it is sufficient to show that the utility function is strictly concave.

$$\frac{\partial^2 U_{in}}{\partial e_i^2} = \exp(\beta^T x_{in} + \varepsilon_{in})\alpha_i(\alpha_i-1)(e_i+\gamma_i)^{\alpha_i-2}.$$

The strict concavity comes from the fact that $0 < \alpha_i < 1$.

3. Finally, for Property 3, (25) is well defined for any $\lambda \geq 0$. Moreover, $e_i^*(\lambda) \geq 0$ if $\lambda \leq W_i$. Therefore,

$$\lambda_i^\ell = 0.$$

## 4.2 Generalized translated utility function

Bhat (2008) generalizes the above formulation and introduces the following specification, where the utility functions $U_{in}$ are defined as

$$U_{1n} = \exp(\beta^T x_{1n} + \varepsilon_{1n}) \frac{1}{\alpha_1} \left( \frac{e_1}{p_1} \right)^{\alpha_1}, \tag{26}$$

for the "outside good" that is always consumed and, for $i > 1$,

$$U_{in} = \exp(\beta^T x_{in} + \varepsilon_{in}) \frac{\gamma_i}{\alpha_i} \left[ \left( \frac{e_i}{p_i \gamma_i} + 1 \right)^{\alpha_i} - 1 \right], \tag{27}$$

where $\beta$, $0 < \alpha_i < 1$ and $\gamma_i > 0$ are parameters to be estimated. Note that this model introduces prices, so that $e_i/p_i$ is the quantity consumed.

In Biogeme, this model is referred to as `generalized`.

### First order conditions

We can calculate

$$\frac{\partial U_{1n}}{\partial e_1} = \exp(\beta^T x_{1n} + \varepsilon_{1n}) \frac{1}{p_1} \left( \frac{e_1}{p_1} \right)^{\alpha_1 - 1}, \tag{28}$$

and

$$\frac{\partial U_{in}}{\partial e_i} = \exp(\beta^T x_{in} + \varepsilon_{in}) \frac{1}{p_i} \left( \frac{e_i}{p_i \gamma_i} + 1 \right)^{\alpha_i - 1}. \tag{29}$$

Evaluated at zero expenditure, we obtain

$$W_{in} = \frac{1}{p_i} \exp(\beta^T x_{in} + \varepsilon_{in}).$$

For forecasting purposes, as described in Section 3, we need to obtain $e_i^*(\lambda)$, the solution of the equation

$$\lambda = \frac{\partial U_{in}}{\partial e_i}.$$

Here, we have

$$e_1^*(\lambda) = p_1 \left( \frac{p_1 \lambda}{\exp(\beta^T x_{1n} + \varepsilon_{1n})} \right)^{\frac{1}{\alpha_1 - 1}}, \tag{30}$$

and

$$e_i^*(\lambda) = p_i \gamma_i \left[ \left( \frac{p_i \lambda}{\exp(\beta^T x_{in} + \varepsilon_{in})} \right)^{\frac{1}{\alpha_i - 1}} - 1 \right]. \tag{31}$$

Note that these expressions correspond to Equations (11) in Pinjari and Bhat (2021).

### Transformed utility

We use again the logarithm as the order preserving function to obtain the following specification:

$$\phi\left(\frac{\partial U_{1n}}{\partial e_1}\right) = \beta^T x_{1n} + \varepsilon_{1n} + (\alpha_1 - 1)\ln e_1 - \alpha_1 \ln p_1,$$

and

$$\phi\left(\frac{\partial U_{in}}{\partial e_i}\right) = \beta^T x_{in} + \varepsilon_{in} - \ln p_i + (\alpha_i - 1)\ln\left(\frac{e_i}{p_i\gamma_i} + 1\right),$$

so that

$$V_{1n} = \beta^T x_{1n} + (\alpha_1 - 1)\ln e_1 - \alpha_1 \ln p_1,$$

and

$$V_{in} = \beta^T x_{in} - \ln p_i + (\alpha_i - 1)\ln\left(\frac{e_i}{p_i\gamma_i} + 1\right).$$

### Entries of the Jacobian

We have

$$c_{1n} = -\frac{\partial V_{1n}}{\partial e_1} = \frac{1-\alpha_1}{e_1},$$

and

$$c_{in} = -\frac{\partial V_{in}}{\partial e_i} = \frac{1-\alpha_i}{e_i + p_i\gamma_i}.$$

Therefore,

$$\begin{aligned}
\ln(c_{1n}) &= \ln(1-\alpha_1) - \ln(e_1),\\
\ln(c_{in}) &= \ln(1-\alpha_i) - \ln(e_i + p_i\gamma_i),\\
1/c_{1n} &= \frac{e_1}{1-\alpha_1},\\
1/c_{in} &= \frac{e_i + p_i\gamma_i}{1-\alpha_i}.
\end{aligned}$$

### Properties

1. In order to show that Property 1 is verified, we consider

   $$\ln W_{in} = \beta^T x_{in} + \varepsilon_{in} - \ln(p_i).$$

   Consider a chosen alternative $i$. From the optimality KKT conditions, we have that

   $$\ln \lambda^* = W_{in} + (\alpha_i - 1)\ln\left(\frac{e_i}{p_i\gamma_i} + 1\right),$$

and

$$\ln W_{in} = \ln \lambda^* + (1-\alpha_i) \ln \left( \frac{e_i}{p_i \gamma_i} + 1 \right),$$

where $\lambda$ is the dual variable associated with the budget constraint. Property 1 follows from the fact that

$$(1-\alpha_i) \ln \left( \frac{e_i}{p_i \gamma_i} + 1 \right) \geq 0.$$

2. For Property 2, it is sufficient to show that the utility is strictly concave. For the outside good,

$$\frac{\partial^2 U_{1n}}{\partial e_1^2} = (\alpha_1 - 1) \frac{1}{p_1} \exp(\beta^T x_{1n} + \varepsilon_{1n}) \frac{1}{p_1} \left( \frac{e_1}{p_1} \right)^{\alpha_1 - 2},$$

$$\frac{\partial^2 U_{in}}{\partial e_i^2} = (\alpha_i - 1) \frac{1}{p_i \gamma_i} \exp(\beta^T x_{in} + \varepsilon_{in}) \frac{1}{p_i} \left( \frac{e_i}{p_i \gamma_i} + 1 \right)^{\alpha_i - 2}$$

The strict concavity comes from the fact that $0 < \alpha_i < 1$.

3. Finally, for Property 3, (30) and (31) are well defined for any $\lambda \geq 0$. Moreover, if $i > 1$, $e_i^*(\lambda) \geq 0$ if $\lambda \leq W_i$. Therefore,

$$\lambda_i^\ell = 0.$$

## 4.3 The $\gamma$-profile

If $\alpha_i \to 0$, (27) collapses to the linear expenditure system (LES) form, defined as follows:

$$U_{1n}(e_1) = \exp(\beta^T x_{1n} + \varepsilon_{1n}) \ln \left( \frac{e_1}{p_1} \right), \tag{32}$$

and

$$U_{in}(e_i) = \exp(\beta^T x_{in} + \varepsilon_{in}) \gamma_i \ln \left( \frac{e_i}{p_i \gamma_i} + 1 \right), \tag{33}$$

where $\beta$, $\gamma_i > 0$ is a parameter to be estimated.

In Biogeme, this model is referred to as `gamma_profile`.

### First order conditions

We have

$$\frac{\partial U_{1n}}{\partial e_1} = \exp(\beta^T x_{1n} + \varepsilon_{1n}) \frac{1}{e_1},$$

and

$$\frac{\partial U_{in}}{\partial e_i} = \exp(\beta^T x_{in} + \varepsilon_{in}) \frac{\gamma_i}{e_i + p_i \gamma_i}.$$

Evaluated at zero expenditure, we obtain

$$W_{in} = \frac{1}{p_i} \exp(\beta^T x_{in} + \varepsilon_{in}).$$

For forecasting purposes, as described in Section 3, we need to obtain $e_i(\lambda)$, the solution of the equation

$$\lambda = \frac{\partial U_{in}}{\partial e_i}.$$

Here, we have

$$e_1(\lambda) = \exp(\beta^T x_{1n} + \varepsilon_{1n}) \frac{1}{\lambda}, \tag{34}$$

and

$$e_i(\lambda) = \exp(\beta^T x_{in} + \varepsilon_{in}) \frac{\gamma_i}{\lambda} - p_i \gamma_i. \tag{35}$$

### Transformed utility

Taking again the logarithm as the order preserving function, we have

$$\phi\left(\frac{\partial U_{1n}}{\partial e_1}\right) = \beta^T x_{1n} + \varepsilon_{1n} - \ln e_1,$$

and

$$\phi\left(\frac{\partial U_{in}}{\partial e_i}\right) = \beta^T x_{in} + \varepsilon_{in} + \ln \gamma_i - \ln(e_i + p_i \gamma_i),$$

so that

$$V_{1n} = \beta^T x_{1n} - \ln e_i,$$

and

$$V_{in} = \beta^T x_{in} + \ln \gamma_i - \ln(e_i + p_i \gamma_i).$$

### Entries of the Jacobian

Finally, we have

$$c_{1n} = \frac{1}{e_1},$$

and

$$c_{in} = \frac{1}{e_i + p_i\gamma_i},$$

so that

$$\begin{aligned} \ln(c_{1n}) &= -\ln(e_1), \\ \ln(c_{in}) &= -\ln(e_i + p_i\gamma_i), \\ 1/c_{1n} &= e_1, \\ 1/c_{in} &= e_i + p_i\gamma_i. \end{aligned}$$

### Properties

1. In order to show that Property 1 is verified, we consider

   $$\ln W_{in} = \beta^T x_{in} + \varepsilon_{in} - \ln(p_i).$$

   Consider a chosen alternative $i$. From the optimality KKT conditions, we have that

   $$\ln\lambda = \beta^T x_{in} + \varepsilon_{in} + \ln\gamma_i - \ln(e_i + p_i\gamma_i) = \ln W_{in} + \ln(p_i) + \ln\gamma_i - \ln(e_i + p_i\gamma_i)$$

   and

   $$\ln W_{in} = \ln\lambda - \ln(p_i) - \ln\gamma_i + \ln(e_i + p_i\gamma_i) = \ln\lambda + \ln\left(\frac{e_i}{p_i\gamma_i} + 1\right),$$

   where $\lambda$ is the dual variable associated with the budget constraint. Property 1 follows from the fact that

   $$\ln\left(\frac{e_i}{p_i\gamma_i} + 1\right) > 0.$$

2. For Property 2, it is sufficient to show that the utility is strictly concave. We have

   $$\frac{\partial^2 U_{1n}}{\partial e_1^2} = -\exp(\beta^T x_{1n} + \varepsilon_{1n}) e_1^{-2},$$

   and

   $$\frac{\partial^2 U_{in}}{\partial e_i^2} = -\exp(\beta^T x_{in} + \varepsilon_{in})\gamma_i(e_i + p_i\gamma_i)^{-2},$$

   and the strict concavity is verified as $\gamma_i > 0$.

3. Finally, for Property 3, (34) and (35) are well defined for any $\lambda > 0$. Moreover, if $i > 1$, $e_i^*(\lambda) \geq 0$ if $\lambda \leq W_i$. Therefore,

$$\lambda_i^\ell = 0.$$

## 4.4 The non-monotonic model

Wang et al. (2023) introduce the following specification in order to accommodate non-monotonic preferences, motivated in the context of time consumption. With our notations, the specification is

$$U_{1n}(e_1) = \frac{1}{\alpha_1} e^{\beta^T x_{in}} e_1^{\alpha_1} + (\theta^T z_{in} + \varepsilon_{in}) e_1.$$

for the outside good, and

$$U_{in}(e_i) = \frac{\gamma_i}{\alpha_i} e^{\beta^T x_{in}} \left[ \left( \frac{e_i}{\gamma_i} + 1 \right)^{\alpha_i} - 1 \right] + (\theta^T z_{in} + \varepsilon_{in}) e_i.$$

where $\beta$, $\theta$, $0 < \alpha_i < 1$, and $\gamma_i > 0$ are parameters to be estimated.

In Biogeme, this model is referred to as `non_monotonic`. Note that the prices must all be the same, so that the model is homoscedastic.

### First order conditions

We have

$$\frac{\partial U_{1n}}{\partial e_1} = e^{\beta^T x_{1n}} e_1^{\alpha_1 - 1} + \theta^T z_{1n} + \varepsilon_{1n},$$

and

$$\frac{\partial U_{in}}{\partial e_i} = e^{\beta^T x_{in}} \left( \frac{e_i}{\gamma_i} + 1 \right)^{\alpha_i - 1} + \theta^T z_{in} + \varepsilon_{in}.$$

Evaluated at zero expenditure, we obtain

$$W_{in} = e^{\beta^T x_{in}} + \theta^T z_{in} + \varepsilon_{in}.$$

For forecasting purposes, as described in Section 3, we need to obtain $e_i(\lambda)$, the solution of the equation

$$\lambda = \frac{\partial U_{in}}{\partial e_i}.$$

Here, we have

$$e_1(\lambda) = \left( e^{-\beta^T x_{1n}} (\lambda - \theta^T z_{1n} - \varepsilon_{1n}) \right)^{\frac{1}{\alpha_1 - 1}}, \tag{36}$$

and

$$e_i(\lambda) = \gamma_i \left[ \left( e^{-\beta^T x_{in}} (\lambda - \theta^T z_{in} - \varepsilon_{in}) \right)^{\frac{1}{\alpha_i - 1}} - 1 \right]. \tag{37}$$

Note that this formula is valid is

$$\lambda \geq \theta^T z_{in} + \varepsilon_{in}.$$

Also, $e_i(\lambda) \geq 0$ if

$$\lambda \leq W_{in}.$$

### Transformed utility

In this case, no order preserving function is necessary, as the error term appear directly in the formulation. Therefore,

$$V_{1n} = e^{\beta^T x_{1n}} e_1^{\alpha_1 - 1} + \theta^T z_{1n}.$$

and

$$V_{in} = e^{\beta^T x_{in}} \left( \frac{e_i}{\gamma_i} + 1 \right)^{\alpha_i - 1} + \theta^T z_{in}.$$

### Entries of the Jacobian

We have

$$c_{1n} = -\frac{\partial V_{1n}}{\partial e_1} = e^{\beta^T x_{1n}} (1 - \alpha_1) e_1^{\alpha_1 - 2},$$

and

$$c_{in} = -\frac{\partial V_{in}}{\partial e_i} = e^{\beta^T x_{in}} \frac{1 - \alpha_i}{\gamma_i} \left( \frac{e_i}{\gamma_i} + 1 \right)^{\alpha_i - 2}.$$

For the outside good, we have

$$\ln c_{1n} = \beta^T x_{1n} + \ln(1 - \alpha_1) + (\alpha_i - 2) \ln e_1,$$

and

$$\frac{1}{c_{1n}} = e^{-\beta^T x_{1n}} \frac{1}{1 - \alpha_1} e_1^{2 - \alpha_1}.$$

For the other goods, we have

$$\ln c_{in} = \beta^T x_{in} + \ln(1 - \alpha_i) - \ln \gamma_i + (\alpha_i - 2) \ln \left( \frac{e_i}{\gamma_i} + 1 \right).$$

and

$$\frac{1}{c_{in}} = e^{-\beta^T x_{in}} \frac{\gamma_i}{1 - \alpha_i} \left( \frac{e_i}{\gamma_i} + 1 \right)^{2 - \alpha_i}.$$

**Properties**

1. We consider
$$W_{in} = \exp(\beta^T x_{in}) + \theta^T z_{in} + \varepsilon_{in}.$$
Consider a chosen alternative $i$. From the optimality KKT conditions, we have that
$$\lambda^* = e^{\beta^T x_{in}} \left(\frac{e_i}{\gamma_i} + 1\right)^{\alpha_i - 1} + \theta^T z_{in} + \varepsilon_{in}, \tag{38}$$
and
$$\lambda^* - \theta^T z_{in} - \varepsilon_{in} = e^{\beta^T x_{in}} \left(\frac{e_i}{\gamma_i} + 1\right)^{\alpha_i - 1}.$$
As
$$\left(\frac{e_i}{\gamma_i} + 1\right)^{\alpha_i - 1} < 1,$$
we have
$$\lambda^* - \theta^T z_{in} - \varepsilon_{in} < e^{\beta^T x_{in}},$$
and
$$\lambda^* < W_{in},$$
so that Property 1 is verified.

2. For Property 2, it is sufficient to show that the utility is strictly concave. We have
$$\frac{\partial^2 U_{1n}}{\partial e_1^2} = (\alpha_1 - 1) e^{\beta^T x_{1n}} e_1^{\alpha_1 - 2},$$
and
$$\frac{\partial^2 U_{in}}{\partial e_i^2} = (\alpha_i - 1) \frac{1}{\gamma_i} e^{\beta^T x_{in}} \left(\frac{e_i}{\gamma_i} + 1\right)^{\alpha_i - 2},$$
and the strict concavity is verified as $0 < \alpha_i < 1$.

3. For Property 3, as commented above, (36) and (37) are well defined if
$$\lambda \geq \theta^T z_{in} + \varepsilon_{in}.$$
and the expenditure is non negative if $\lambda \leq W_i$, which is guaranteed by Property 1. Therefore,
$$\lambda_i^\ell = \theta^T z_{in} + \varepsilon_{in}.$$

# 5 Model specification with Biogeme

## 5.1 Data

**Internal note:** MENGYI: please provide a description of the data here.

In terms of implementation, the data preparation is the same as for any regular Biogeme model. In particular, it involves the following steps.

1. The data is stored in a Pandas data frame, typically read from a file:

   ```
   df = pd.read_csv('data.csv')
   ```

2. It is converted into a Biogeme database object:

   ```
   database = db.Database('mdcev_example', df)
   ```

3. The name of each column is associated with a Python variable for subsequent use:

   ```
   PersonID = Variable('PersonID')
   weight = Variable('weight')
   ...
   ```

## 5.2 Baseline utilities

The baseline utilities are defined in the exact same way as for regular Biogeme models, in the form of a dictoinary associating the identifier of each alternative with the specification of the utility function.

```
shopping = (
    cte_shopping
    + metropolitan_shopping * metro
    + male_shopping * male
    + age_15_40_shopping * age15_40
    + spouse_shopping * spousepr
    + employed_shopping * employed
)

socializing = (
    cte_socializing
    + number_members_socializing * hhsize
    + male_socializing * male
    + age_41_60_socializing * age41_60
```

```
    + bachelor_socializing * bachigher
    + sunday_socializing * Sunday
)

recreation = (
    cte_recreation
    + number_members_recreation * hhsize
    + male_recreation * male
    + age_15_40_recreation * age15_40
    + spouse_recreation * spousepr
)

personal = (
    age_41_60_personal * age41_60
    + bachelor_personal * bachigher
    + white_personal * white
    + sunday_personal * Sunday
)

baseline_utilities = {
    1: shopping,
    2: socializing,
    3: recreation,
    4: personal,
}
```

A key difference with traditional Biogeme models is the dependent variable, which consists in an observed quantity for each alternative. This is also captured by a dictionary. In this example, the quantities correspond to an amount of time, expressed in minutes, and translated into hours.

```
consumed_quantities = {
    1: t1 / 60.0,
    2: t2 / 60.0,
    3: t3 / 60.0,
    4: t4 / 60.0,
}
```

## 5.3 MDCEV model

Each model introduced in Section 4 is associated with an object in Biogeme.

**Translated utility function** We first import the corresponding Python class:

```
from biogeme.mdcev import Translated
```

The parameters are then defined. Note that most of them bust be positive. There are two ways to impose positivity. The first (adopted in this example) is to impose a positive lower bound, such as $10^{-4}$. For example, the $\gamma$ parameters (see Section 4 for their definition) can be defined as follows:

```
# Gamma parameters. Must be positive.
lowest_positive_value = 0.0001
gamma_shopping = Beta('gamma_shopping', 1,
   lowest_positive_value, None, 0)
gamma_socializing = Beta('gamma_socializing', 1,
   lowest_positive_value, None, 0)
gamma_recreation = Beta('gamma_recreation', 1,
   lowest_positive_value, None, 0)
gamma_personal = Beta('gamma_personal', 1,
   lowest_positive_value, None, 0)
```

Another option would be to estimate the logarithm of the parameters, as follows:

```
# Gamma parameters. Must be positive.
gamma_shopping = exp(Beta('log_gamma_shopping', 0, None,
   None, 0))
gamma_socializing = exp(Beta('log_gamma_socializing', 0,
   None, None, 0))
gamma_recreation = exp(Beta('log_gamma_recreation', 0,
   None, None, 0))
gamma_personal = exp(Beta('log_gamma_personal', 0, None,
   None, 0))
```

The $\alpha$ parameters are defined similarly, imposing that they lie between 0 and 1.

```
alpha_shopping = Beta('alpha_shopping', 0.5,
   lowest_positive_value, 1, 0)
alpha_socializing = Beta('alpha_socializing', 0.5,
   lowest_positive_value, 1, 0)
alpha_recreation = Beta('alpha_recreation', 0.5,
   lowest_positive_value, 1, 0)
alpha_personal = Beta('alpha_personal', 0.5,
   lowest_positive_value, 1, 0)
```

Finally, the scale parameter of the error term:

```
scale_parameter = Beta('scale', 1, lowest_positive_value,
   None, 0)
```

Once they have been defined, they need to be associated with the alternative identifiers.

```
gamma_parameters = {
    1: gamma_shopping,
    2: gamma_socializing,
    3: gamma_recreation,
    4: gamma_personal,
}

alpha_parameters = {
    1: alpha_shopping,
    2: alpha_socializing,
    3: alpha_recreation,
    4: alpha_personal,
}
```

Once all the ingredients have been prepared, the object can be created:

```
the_translated = Translated(
    model_name='translated',
    baseline_utilities=baseline_utilities,
    gamma_parameters=gamma_parameters,
    alpha_parameters=alpha_parameters,
    scale_parameter=scale_parameter,
    weights=weight,
)
```

**Generalized translated utility function** The specification of this model is almost identical to the previous one.

```
from biogeme.mdcev import Generalized

lowest_positive_value = 0.0001
gamma_shopping = Beta('gamma_shopping', 1,
   lowest_positive_value, None, 0)
gamma_socializing = Beta('gamma_socializing', 1,
   lowest_positive_value, None, 0)
gamma_recreation = Beta('gamma_recreation', 1,
   lowest_positive_value, None, 0)
gamma_personal = Beta('gamma_personal', 1,
   lowest_positive_value, None, 0)

alpha_shopping = Beta('alpha_shopping', 0.5,
   lowest_positive_value, 1, 0)
alpha_socializing = Beta('alpha_socializing', 0.5,
   lowest_positive_value, 1, 0)
```

```
alpha_recreation = Beta('alpha_recreation', 0.5,
    lowest_positive_value, 1, 0)
alpha_personal = Beta('alpha_personal', 0.5,
    lowest_positive_value, 1, 0)

scale_parameter = Beta('scale', 1, lowest_positive_value,
    None, 0)

gamma_parameters = {
    1: gamma_shopping,
    2: gamma_socializing,
    3: gamma_recreation,
    4: gamma_personal,
}

alpha_parameters = {
    1: alpha_shopping,
    2: alpha_socializing,
    3: alpha_recreation,
    4: alpha_personal,
}

the_generalized = Generalized(
    model_name='generalized',
    baseline_utilities=baseline_utilities,
    gamma_parameters=gamma_parameters,
    alpha_parameters=alpha_parameters,
    scale_parameter=scale_parameter,
    weights=weight,
)
```

**The $\gamma$-profile** This specification does not involve $\alpha$ parameters.

```
from biogeme.mdcev import GammaProfile
lowest_positive_value = 0.0001
gamma_shopping = Beta('gamma_shopping', 1,
    lowest_positive_value, None, 0)
gamma_socializing = Beta('gamma_socializing', 1,
    lowest_positive_value, None, 0)
gamma_recreation = Beta('gamma_recreation', 1,
    lowest_positive_value, None, 0)
gamma_personal = Beta('gamma_personal', 1,
    lowest_positive_value, None, 0)

scale_parameter = Beta('scale', 1, lowest_positive_value,
    None, 0)

gamma_parameters = {
```

```
    1: gamma_shopping,
    2: gamma_socializing,
    3: gamma_recreation,
    4: gamma_personal,
}

the_gamma_profile = GammaProfile(
    model_name='gamma_profile',
    baseline_utilities=baseline_utilities,
    gamma_parameters=gamma_parameters,
    scale_parameter=scale_parameter,
    weights=weight,
)
```

**The non-monotonic model** On top of the baseline utilities, the non-monotonic model involves also another component of utility, called the $\mu$-utilities. In this illustration, we use the same specification for both.

```
from biogeme.mdcev import NonMonotonic
lowest_positive_value = 0.0001
gamma_shopping = Beta('gamma_shopping', 1,
   lowest_positive_value, None, 0)
gamma_socializing = Beta('gamma_socializing', 1,
   lowest_positive_value, None, 0)
gamma_recreation = Beta('gamma_recreation', 1,
   lowest_positive_value, None, 0)
gamma_personal = Beta('gamma_personal', 1,
   lowest_positive_value, None, 0)

alpha_shopping = Beta('alpha_shopping', 0.5,
   lowest_positive_value, 1, 0)
alpha_socializing = Beta('alpha_socializing', 0.5,
   lowest_positive_value, 1, 0)
alpha_recreation = Beta('alpha_recreation', 0.5,
   lowest_positive_value, 1, 0)
alpha_personal = Beta('alpha_personal', 0.5,
   lowest_positive_value, 1, 0)

scale_parameter = Beta('scale', 1, lowest_positive_value,
   None, 0)

gamma_parameters = {
    1: gamma_shopping,
    2: gamma_socializing,
    3: gamma_recreation,
    4: gamma_personal,
}
```

```
alpha_parameters = {
    1: alpha_shopping,
    2: alpha_socializing,
    3: alpha_recreation,
    4: alpha_personal,
}


the_non_monotonic = NonMonotonic(
    model_name='non_monotonic',
    baseline_utilities=baseline_utilities,
    mu_utilities=baseline_utilities,
    gamma_parameters=gamma_parameters,
    alpha_parameters=alpha_parameters,
    scale_parameter=scale_parameter,
    weights=weight,
)
```

## 5.4 Estimation of the parameters

Once the object characterizing the model has been created, the estimation of the parameters is simply done as follows for the $\gamma$ profile:

```
results = the_gamma_profile.estimate_parameters(
    database=database,
    number_of_chosen_alternatives=number_chosen,
    consumed_quantities=consumed_quantities,
)
```

## 5.5 Forecasting

Finally, the forecasting is performed as follows. Here, we perform the forecast for two rows of the database.

```
two_rows_of_database: Database = database.extract_rows([0, 1])
budget_in_hours = 500
number_of_draws = 20000
optimal_consumptions: list[pd.DataFrame] =
    the_gamma_profile.forecast(
    database=two_rows_of_database,
    total_budget=budget_in_hours,
    number_of_draws=number_of_draws,
    brute_force=False,
)
```

The result is a list of data frames, one per entry in the database (two, in this example). Each data frame contains the forecasting results for each realization of the error term. Note that the parameter `brute_force` selecting the forecasting algorithm is optional and set to `False` by default. It means that the analytical algorithm is used.

# References


Bhat, C. R. (2005). A multiple discrete-continuous extreme value model: formulation and application to discretionary time-use decisions, *Transportation Research Part B: Methodological* **39**(8): 679 – 707.

Bhat, C. R. (2008). The multiple discrete-continuous extreme value (MDCEV) model: Role of utility function parameters, identification considerations, and model extensions, *Transportation Research Part B: Methodological* **42**(3): 274–303. A Tribute to the Career of Frank Koppelman.
**URL:** *https://www.sciencedirect.com/science/article/pii/S0191261507000677*

Kim, J., Allenby, G. M. and Rossi, P. E. (2002). Modeling consumer demand for variety, *Marketing Science* **21**(3): 229–250.

Pinjari, A. R. and Bhat, C. (2021). Computationally efficient forecasting procedures for kuhn-tucker consumer demand model systems: Application to residential energy consumption analysis, *Journal of Choice Modelling* **39**: 100283.
**URL:** *https://www.sciencedirect.com/science/article/pii/S1755534521000166*

Wales, T. J. and Woodland, A. D. (1983). Estimation of consumer demand systems with binding non-negativity constraints, *Journal of Econometrics* **21**(3): 263–285.

Wang, M., Ye, X. and Bierlaire, M. (2023). Accommodating non-monotonic preference in multiple discrete-continuous extreme value (MDCEV) choice model: Formulation and application to time use decisions. submitted to TRB.