mlogit: Random Utility Models in R

Yves Croissant

mlogit: Random Utility Models in R

Yves Croissant

CEMOI – Université de la Réunion

Abstract

mlogit is a package for R which enables the estimation of random utility models with choice situation and/or alternative specific variables. The main extensions of the basic multinomial model (heteroscedastic, nested and random parameter models) are implemented.

Keywords: discrete choice models, maximum likelihood estimation, R, econometrics.

1. Introduction

Random utility models are the reference approach in economics when one wants to analyze the choice by a decision maker of one among a set of mutually exclusive alternatives. Since the seminal works of Daniel McFadden (McFadden 1974, 1978) who won the Nobel prize in economics "for his development of theory and methods for analyzing discrete choice", a large amount of theoretical and empirical literature has been developed in this field.[1]

These models rely on the hypothesis that the decision maker is able to rank the different alternatives by an order of preference represented by a utility function, the chosen alternative being the one which is associated with the highest level of utility. They are called random utility models because part of the utility is unobserved and is modeled as the realization of a random deviate.

Different hypotheses on the distribution of this random deviate lead to different flavors of random utility models. Early developments of these models were based on the hypothesis of identical and independent errors following a Gumbel distribution,[2] leading to the multinomial logit model (*MNL*). More general models have since been proposed, either based on less restrictive distribution hypotheses or by introducing individual heterogeneity.

---

[1]For a presentation of this literature, see Train (2009); the theoretical parts of this paper draw heavily on this book.

[2]This distribution has the distinctive advantage that it leads to a probability which can be written as an integral which has a closed form.

Maintaining the Gumbel distribution hypothesis but relaxing the i.i.d. (identically and independently distributed) hypothesis leads to more general logit models (the heteroscedastic and the nested logit models). Relaxing the Gumbel distribution hypothesis and using a normal distribution instead leads to the multinomial probit model which can deal with heteroscedasticity and correlation of the errors.

Individual heterogeneity can be introduced in the parameters associated with the covariates entering the observable part of the utility or in the variance of the errors. This leads respectively to the mixed effect models (*MXL*) and the scale heterogeneity model (*S-MNL*).

All these models can be estimated with the **mlogit** package (Croissant 2020b) which provides a unified interface and is available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/package=mlogit`. The first version of **mlogit** was posted in 2008 and it was the first R package allowing the estimation of random utility models. Since then, other package have emerged.[3] The following packages provide useful additions to **mlogit**:

- **mnlogit** (Hasan, Wang, and Mahani 2016) enables efficient estimation of *MNL* for large data sets,

- **gmnl** (Sarrias and Daziano 2017) estimates *MXL* and *S-MNL*, but also the so called generalized multinomial logit model *G-MNL* which nests them,

- latent-class multinomial logit models (*LC-MNL*), for which the heterogeneity is due to the fact that individuals belong to different classes and mixed-mixed models (*MM-MNL*) which are a mixture of *LC-MNL* and *MXL* can also be estimated using the **gmnl** package,

- Bayesian estimators for multinomial models are provided by packages **bayesm** (Rossi 2019), **MNP** (Imai and van Dyk 2017) and **RSGHB** (Dumont and Keller 2019).

The article is organized as follow. Section 2 explains how the usual formula-data and testing interface can be extended in order to describe in a natural way the model to be estimated. Section 3 describes the landmark multinomial logit model. Section 4 and 5 present two important extensions of this basic model: Section 4 presents models that relax the i.i.d. Gumbel hypothesis and Section 5 introduces slope heterogeneity by considering some parameters as random. Section 6 concludes.

2. Data management, model description and testing

The formula-data interface is a critical advantage of the R software. It provides a practical way to describe the model to be estimated and to store data. However, the usual interface is not flexible enough to deal correctly with random utility models. Therefore, **mlogit** provides tools to construct richer `data.frame`s and `formula`s.

2.1. Data management

Package **mlogit** is loaded using:

```
R> library("mlogit")
```

---

[3]See Sarrias and Daziano (2017, page 4) for a survey of relevant R packages.

It comes with several data sets that we will use to illustrate the features of the package. Data sets used for multinomial logit estimation concern some individuals, that make one or a sequential choice of one alternative among a set of mutually exclusive alternatives. The determinants of these choices are covariates that can depend on the alternative and the choice situation, only on the alternative or only on the choice situation.

To illustrate this typology of the covariates, consider the case of repeated choice of destinations for vacations by families:

- the length of the vacation and the season are choice situation specific variables,

- income and family size are individual specific variables,

- distance to destination and cost are alternative specific variables.

The unit of observation is therefore the choice situation, and it is also the individual if there is only one choice situation per individual observed, which is often the case.

Data sets used to estimate random utility models have therefore a specific structure that can be characterized by three indexes: the alternative, the choice situation and the individual. These three indexes will be denoted `alt`, `chid` and `id`. Note that the distinction between `chid` and `id` is only relevant if we have repeated observations for the same individual.

Data sets can have two different shapes: a "wide" shape (one row for each choice situation) or a "long" shape (one row for each alternative and, therefore, as many rows as there are alternatives for each choice situation).

Package **mlogit** deals with both formats by relying on the `dfidx` function from the **dfidx** package (Croissant 2020a), which takes as first argument a `data.frame` and returns an object of class 'dfidx', which in "long" format is a `data.frame` with a special column which contains the corresponding indexes.

Wide format

`Train`[4] is an example of a "wide" data set:

```
R> data("Train", package = "mlogit")
R> Train$choiceid <- 1:nrow(Train)
R> head(Train, 3)
```

```
  id choiceid choice price_A time_A change_A comfort_A price_B time_B
1  1        1      A    2400    150        0         1    4000    150
2  1        2      A    2400    150        0         1    3200    130
3  1        3      A    2400    115        0         1    4000    115
  change_B comfort_B
1        0         1
2        0         1
3        0         0
```

---

[4]Used by Ben-Akiva, Bolduc, and Bradley (1993) and Meijer and Rouwendal (2006).

This data set contains data about a stated preference survey in the Netherlands. Each individual has responded to several (up to 16) scenarios. For every scenario, two train tickets `A` and `B` are proposed to the user, with different combinations of four attributes: `price` (the price in cents of guilders), `time` (travel time in minutes), `change` (the number of changes) and `comfort` (the class of comfort, 0, 1 or 2, 0 being the most comfortable class).

This "wide" format is suitable to store choice situation (or individual specific) variables because, in this case, they are stored only once in the data. Otherwise, it is cumbersome for alternative specific variables because there are as many columns for such variables that there are alternatives.

For such a "wide" data set, the `shape` argument of `dfidx` is mandatory, as its default value is `"long"`. The alternative specific variables are indicated with the argument `varying` which is a numeric vector that indicates their position in the data frame. This argument is then passed to `stats::reshape` that coerces the original data frame in "long" format. Further arguments may be passed to `stats::reshape`. For example, as the names of the variables are of the form `price_A`, one must add `sep = "_"` (the default value being `"."`). The `choice` argument is also mandatory because the response has to be transformed in a logical value in the "long" format. Given that each individual faces different choice situations, the `id` variable containing the individual index nests the choice situation variable called `choiceid`. This nesting structure in the indexes can be specified as a list through the argument `idx`.

```
R> Tr <- dfidx(Train, shape = "wide", choice = "choice",
+    varying = 4:11, sep = "_", idx = list(c("choiceid", "id")),
+    idnames = c("chid, "alt"),
+    opposite = c("price", "comfort", "time", "change"))
```

Note the use of the `opposite` argument for the four covariates: We expect negative coefficients for all of them, taking the opposite of the covariates will lead to expected positive coefficients. We next convert `price` and `time` in more meaningful units, namely hours and euros (1 guilder was 2.20371 euros):

```
R> Tr$price <- Tr$price / 100 * 2.20371
R> Tr$time <- Tr$time / 60
R> head(Tr, 3)

~~~~~~~
 first 3 observations out of 5858
~~~~~~~
  choice      price time change comfort idx
1   TRUE -52.88904 -2.5      0      -1 1:A
2  FALSE -88.14840 -2.5      0      -1 1:B
3   TRUE -52.88904 -2.5      0      -1 2:A

~~~ indexes ~~~~
  chid id alt
1    1  1   A
2    1  1   B
3    2  1   A
indexes:  1, 1, 2
```

An `idx` attribute is added to the data, which contains the three relevant indexes: `choiceid` is the choice situation index, `alt` the alternative index and `id` is the individual index. This attribute is a `data.frame` which can be extracted using the `idx` function.

```
R> head(idx(Tr), 3)

~~~ indexes ~~~~
  choiceid id alt
1        1  1   A
2        1  1   B
3        2  1   A
indexes:  1, 1, 2
```

Long format

`ModeCanada`[5] is an example of a data set in "long" format. It presents the choice of individuals for a transport mode for the Ontario-Quebec corridor:

```
R> data("ModeCanada", package = "mlogit")
R> head(ModeCanada)

  case   alt choice dist  cost ivt ovt freq income urban noalt
1    1 train      0   83 28.25  50  66    4     45     0     2
2    1   car      1   83 15.77  61   0    0     45     0     2
3    2 train      0   83 28.25  50  66    4     25     0     2
4    2   car      1   83 15.77  61   0    0     25     0     2
5    3 train      0   83 28.25  50  66    4     70     0     2
6    3   car      1   83 15.77  61   0    0     70     0     2
```

There are four transport modes (`air`, `train`, `bus` and `car`) and most of the variables are alternative specific (`cost` for monetary cost, `ivt` for in vehicle time, `ovt` for out of vehicle time, `freq` for frequency). The only choice situation specific variables are `dist` (the distance of the trip), `income` (household income), `urban` (a dummy for trips which have a large city at the origin or the destination) and `noalt` the number of available alternatives. The advantage of this shape is that there are much fewer columns than in the "wide" format, the caveat being that values of `dist`, `income` and `urban` are repeated four times.

For data in "long" format the `shape` argument is no longer mandatory.

To replicate published results later in the text (in the first part of Section 3.5), we will use only a subset of the choice situations, namely those for which the four alternatives are available. This can be done using the `subset` function with the `subset` argument set to `noalt == 4` when estimating the model. This can also be done within `dfidx`, using the `subset` argument.

The information about the structure of the data can be explicitly indicated using choice situations and alternative indexes (in this data set `case` and `alt`, respectively) or, in part,

---

[5]Used in particular by Forinash and Koppleman (1993), Bhat (1995), Koppelman and Wen (1998) and Koppelman and Wen (2000).

guessed by the `dfidx` function. Here, after subsetting, we have 2779 choice situations with four alternatives, and the rows are ordered first by choice situation and then by alternative (`train`, `air`, `bus` and `car` in this order).

The first way to read correctly this data frame is to ignore completely the two index variables. In this case, the only supplementary argument to provide is the `alt.levels` argument which is a character vector that contains the name of the alternatives in their order of appearance:

```
R> MC <- dfidx(ModeCanada, subset = noalt == 4,
+   alt.levels = c("train", "air", "bus", "car"))
```

Note that this can only be used if the data set is "balanced", which means that the same set of alternatives is available for all choice situations. It is also possible to provide the name of the variable that contains the alternatives through the argument `idx`:

```
R> MC <- dfidx(ModeCanada, subset = noalt == 4, idx = list(NA, "alt"))
```

The name of the variable that contains the information about the choice situations can also be indicated through the argument `idx`:

```
R> MC <- dfidx.data(ModeCanada, subset = noalt == 4, idx = "case",
+   alt.levels = c("train", "air", "bus", "car"))
```

Both alternative and choice situation variables can also be provided:

```
R> MC <- dfidx(ModeCanada, subset = noalt == 4,
+   idx = c("case", "alt"))
```

Moreover, as the two indexes are stored in the first two columns of the original data frame, the `idx` argument can be left unspecified

```
R> MC <- dfidx(ModeCanada, subset = noalt == 4)
```

and the indexes can be kept as stand alone series if the `drop.index` argument is set to `FALSE`:

```
R> MC <- dfidx(ModeCanada, subset = noalt == 4, idx = c("case", "alt"),
+   drop.index = FALSE)
R> head(MC)
```

```
~~~~~~~
 first 10 observations out of 11116
~~~~~~~
   case   alt choice dist    cost ivt ovt freq income urban noalt       idx
1  109 train      0  377   58.25 215  74    4     45     0     4 109:rain
2  109   air      1  377  142.80  56  85    9     45     0     4  109:air
3  109   bus      0  377   27.52 301  63    8     45     0     4  109:bus
4  109   car      0  377   71.63 262   0    0     45     0     4  109:car
5  110 train      0  377   58.25 215  74    4     70     0     4 110:rain
6  110   air      1  377  142.80  56  85    9     70     0     4  110:air
```

```
7   110    bus    0  377  27.52 301  63   8    70    0    4  110:bus
8   110    car    0  377  71.63 262   0   0    70    0    4  110:car
9   111  train    0  377  58.25 215  74   4    35    0    4  111:rain
10  111    air    1  377 142.80  56  85   9    35    0    4  111:air

~~~ indexes ~~~~
   case   alt
1   109 train
2   109   air
3   109   bus
4   109   car
5   110 train
6   110   air
7   110   bus
8   110   car
9   111 train
10  111   air
indexes:  1, 2
```

2.2. Model description

Standard `formulas` are not very practical to describe random utility models, as these models may use different sets of covariates. Actually, when working with random utility models, one has to consider at most four sets of covariates:

1. alternative and choice situation specific covariates x_{ij} with generic coefficients β and alternative specific covariates t_j with a generic coefficient ν,

2. choice situation specific covariates z_i with alternative specific coefficients γ_j,

3. alternative and choice situation specific covariates w_{ij} with alternative specific coefficients δ_j,

4. choice situation specific covariates v_i that influence the variance of the errors.

The first three sets of covariates enter the observable part of the utility which can be written, for alternative j:

$$V_{ij} = \alpha_j + \beta x_{ij} + \nu t_j + \gamma_j z_i + \delta_j w_{ij}.$$

As the absolute value of utility is irrelevant, only utility differences are useful to model the choice for one alternative. For two alternatives j and k, we obtain:

$$V_{ij} - V_{ik} = (\alpha_j - \alpha_k) + \beta(x_{ij} - x_{ik}) + (\gamma_j - \gamma_k)z_i + (\delta_j w_{ij} - \delta_k w_{ik}) + \nu(t_j - t_k).$$

It is clear from the previous expression that coefficients of choice situation specific variables (the intercept being one of those) should be alternative specific, otherwise they would disappear in the differentiation. Moreover, only differences of these coefficients are relevant and can

be identified. For example, with three alternatives 1, 2 and 3, the three coefficients $\gamma_1, \gamma_2, \gamma_3$ associated to a choice situation specific variable cannot be identified, but only two linear combinations thereof. Therefore, one has to make a choice of normalization and the simplest one is to simply set $\gamma_1 = 0$.

Coefficients for alternative and choice situation specific variables may (or may not) be alternative specific. For example, transport time is alternative specific, but 10 min in public transport may not have the same impact on utility than 10 min in a car. In this case, alternative specific coefficients are relevant. Monetary cost is also alternative specific, but in this case, one can consider that 1\$ is 1\$ whether it is spent for the use of a car or in public transports. In this case, a generic coefficient is appropriate.

The treatment of alternative specific variables does not differ much from the alternative and choice situation specific variables with a generic coefficient. However, if some of these variables are introduced, the ν parameter can only be estimated in a model without intercepts to avoid perfect multicollinearity.

Individual-related heteroscedasticity (see Swait and Louviere 1993) can be addressed by writing the utility of choosing j for individual i as $U_{ij} = V_{ij} + \sigma_i \epsilon_{ij}$, where ϵ has a variance that does not depend on i and j and $\sigma_i^2 = f(v_i)$ is a parametric function of some individual-specific covariates. Note that this specification induces choice situation heteroscedasticity, also denoted scale heterogeneity.[6] As the overall scale of utility is irrelevant, the utility can also be written as: $U_{ij}^* = U_{ij}/\sigma_i = V_{ij}/\sigma_i + \epsilon_{ij}$, i.e., with homoscedastic errors. If V_{ij} is a linear combination of covariates, the associated coefficients are then divided by σ_i.

A logit model with only choice situation specific variables is sometimes called a *multinomial logit model*, one with only alternative specific variables a *conditional logit model* and one with both kind of variables a *mixed logit model*. This is seriously misleading: *conditional logit model* is also a logit model for longitudinal data in the statistical literature and *mixed logit* is one of the names of a logit model with random parameters. Therefore, in what follows, we will use the name *multinomial logit model* for the model we have just described whatever the nature of the explanatory variables used.

Package **mlogit** makes use of objects of class 'Formula' provided by the **Formula** package (Zeileis and Croissant 2010). The **Formula** package provides richer `formula`s, which accept multiple responses (a feature not used here) and multiple sets of covariates. Moreover, specific `model.frame` and `model.matrix` methods which can be used with one or several sets of covariates are available.

To illustrate the use of objects of class 'Formula', we use again the `ModeCanada` data set and consider three sets of covariates that will be indicated in a three-part formula, which refers to the first three items of the four point list at the beginning of Section 2.2.

- `cost` (monetary cost) is an alternative specific covariate with a generic coefficient (part 1),

- `income` and `urban` are choice situation specific covariates (part 2),

- `ivt` (in vehicle travel time) is alternative specific and alternative specific coefficients are expected (part 3).

---

[6]This kind of heteroscedasticity should not be confused with alternative heteroscedasticity ($\sigma_j^2 \neq \sigma_k^2$) which is introduced in the heteroscedastic logit model described in Section 4.1.

```
R> library("Formula")
R> f <- Formula(choice ~ cost | income + urban | ivt)
```

Some parts of the formula may be omitted when there is no ambiguity. For example, the following sets of `formula`s are identical:

```
R> f2 <- Formula(choice ~ cost + ivt | income + urban)
R> f2 <- Formula(choice ~ cost + ivt | income + urban | 0)

R> f3 <- Formula(choice ~ 0 | income | 0)
R> f3 <- Formula(choice ~ 0 | income)

R> f4 <- Formula(choice ~ cost + ivt)
R> f4 <- Formula(choice ~ cost + ivt | 1)
R> f4 <- Formula(choice ~ cost + ivt | 1 | 0)
```

By default, an intercept is added to the model but it can be removed by using `+ 0` or `- 1` in the second part.

```
R> f5 <- Formula(choice ~ cost | income + 0 | ivt)
R> f5 <- Formula(choice ~ cost | income - 1 | ivt)
```

A `model.frame` method is provided for 'dfidx' objects. It differs from the `formula` method by the fact that the returned object is of class 'dfidx' and not an ordinary data frame, which means that the information about the structure of the data is not lost. Defining a specific `model.frame` method for 'dfidx' objects implies that the first argument of the function should be a 'dfidx' object, which results in an unusual order of the arguments in the function (the data first, and then the formula). Moreover, as the model matrix for random utility models has specific features, we add a supplementary argument called `pkg` to the `dfidx` function so that the returned object has a specific class (and inherits the 'dfidx' class):

```
R> MC <- dfidx(ModeCanada, subset = noalt == 4, pkg = "mlogit")
R> class(MC)

[1] "dfidx_mlogit" "dfidx" "data.frame"

R> f <- Formula(choice ~ cost | income  | ivt)
R> mf <- model.frame(MC, f)
R> class(mf)

[1] "dfidx_mlogit" "dfidx" "data.frame"
```

Using `mf` as the argument of `model.matrix` enables the construction of the relevant model matrix for random utility model as a specific `model.matrix` method for 'difdx_mlogit' objects is provided.

```
R> head(model.matrix(mf), 4)
```

```
  (Intercept):air (Intercept):bus (Intercept):car    cost income:air
1               0               0               0   58.25          0
2               1               0               0  142.80         45
3               0               1               0   27.52          0
4               0               0               1   71.63          0
  income:bus income:car ivt:train ivt:air ivt:bus ivt:car
1          0          0       215       0       0       0
2          0          0         0      56       0       0
3         45          0         0       0     301       0
4          0         45         0       0       0     262
```

The model matrix contains $J-1$ columns for every choice situation specific variables (`income` and the intercept), which means that the coefficient associated to the first alternative (`air`) is set to zero. It contains only one column for `cost` because we want a generic coefficient for this variable. It contains J columns for `ivt`, because it is an alternative specific variable for which we want alternative specific coefficients.

2.3. Testing

As for all models estimated by maximum likelihood, three testing procedures may be applied to test hypotheses about models fitted using **mlogit**. The set of hypotheses tested defines two models: the unconstrained model that does not take these hypotheses into account and the constrained model which imposes these hypotheses.

This in turn defines three principles of tests: the *Wald test*, based only on the unconstrained model, the *Lagrange multiplier test* (or *score test*), based only on the constrained model and the *likelihood ratio test*, based on the comparison of both models.

Two of these tests are implemented in the **lmtest** package (Zeileis and Hothorn 2002): `waldtest` and `lrtest`. The Wald test is also implemented as `linearHypothesis` in package **car** (Fox and Weisberg 2019), with a fairly different syntax. We provide special methods of `waldtest` and `lrtest` for 'mlogit' objects and we also provide a function for the Lagrange multiplier (or score) test called `scoretest`.

We will see later that the score test is especially useful for 'mlogit' objects when one is interested in extending the basic multinomial logit model because, in this case, the unconstrained model may be difficult to estimate. For the presentation of further tests, we provide a convenient `statpval` function which extracts the statistic and the p value from the objects returned by the testing function, which can be either of class 'anova' or 'htest'.

```
R> statpval <- function(x) {
+    if (inherits(x, "anova"))
+      result <- as.matrix(x)[2, c("Chisq", "Pr(>Chisq)")]
+    if (inherits(x, "htest")) result <- c(x$statistic, x$p.value)
+    names(result) <- c("stat", "p-value")
+    round(result, 3)
+  }
```

3. Random utility model and the multinomial logit model

3.1. Random utility model

The utility for alternative l is written as: $U_l = V_l + \epsilon_l$ where V_l is a function of some observable covariates and unknown parameters to be estimated, and ϵ_l is a random deviate which contains all the unobserved determinants of the utility. Alternative l is therefore chosen if $\epsilon_j < (V_l - V_j) + \epsilon_l \ \forall \ j \neq l$ and the probability of choosing this alternative is then:

$$\mathsf{P}(\epsilon_1 < V_l - V_1 + \epsilon_l, \epsilon_2 < V_l - V_2 + \epsilon_l, \ldots, \epsilon_J < V_l - V_J + \epsilon_l).$$

Denoting F_{-l} the cumulative density function of all the ϵ's except ϵ_l, this probability is:

$$(P_l \mid \epsilon_l) = F_{-l}(V_l - V_1 + \epsilon_l, \ldots, V_l - V_J + \epsilon_l).$$

Note that this probability is conditional on the value of ϵ_l. The unconditional probability (which depends only on β and on the value of the observed explanatory variables) is obtained by integrating out the conditional probability using the marginal density of ϵ_l, denoted f_l:

$$P_l = \int F_{-l}(V_l - V_1 + \epsilon_l, \ldots, V_l - V_J) + \epsilon_l) f_l(\epsilon_l) d\epsilon_l.$$

The conditional probability is an integral of dimension $J - 1$ and the computation of the unconditional probability adds one more dimension of integration.

3.2. The distribution of the error terms

The multinomial logit model (McFadden 1974) is a special case of the model developed in the previous section. It is based on three hypotheses.

The first hypothesis is the independence of the errors. In this case, the univariate distribution of the errors can be used, which leads to the following conditional and unconditional probabilities:

$$(P_l \mid \epsilon_l) = \prod_{j \neq l} F_j(V_l - V_j + \epsilon_l) \text{ and } P_l = \int \prod_{j \neq l} F_j(V_l - V_j + \epsilon_l) \ f_l(\epsilon_l) \ d\epsilon_l,$$

which means that the conditional probability is the product of $J - 1$ univariate cumulative density functions and only the evaluation of a one-dimensional integral is required to compute the unconditional probability.

The second hypothesis is that each ϵ follows a Gumbel distribution, whose density and probability functions are, respectively:

$$f(z) = \frac{1}{\theta} e^{-\frac{z-\mu}{\theta}} e^{-e^{-\frac{z-\mu}{\theta}}} \text{ and } F(z) = \int_{-\infty}^{z} f(t) dt = e^{-e^{-\frac{z-\mu}{\theta}}}, \tag{1}$$

where μ is the location parameter and θ the scale parameter. The first two moments of the Gumbel distribution are $\mathsf{E}(z) = \mu + \theta\gamma$, where γ is the Euler-Mascheroni constant (≈ 0.577) and $\mathsf{VAR}(z) = \frac{\pi^2}{6}\theta^2$. The mean of ϵ_j is not identified if V_j contains an intercept. We can then,

without loss of generality suppose that $\mu_j = 0$, $\forall j$. Moreover, the overall scale of utility is not identified. Therefore, only $J - 1$ scale parameters may be identified, and a natural choice of normalization is to impose that one of the θ_j is equal to one.

The last hypothesis is that the errors are identically distributed. As the location parameter is not identified for any error term, this hypothesis is essentially a homoscedasticity hypothesis, which means that the scale parameter of the Gumbel distribution is the same for all the alternatives. As one of them has been previously set to one, we can therefore suppose that, without loss of generality, $\theta_j = 1$, $\forall j \in 1 \ldots J$. The conditional and unconditional probabilities then further simplify to:

$$(P_l \mid \epsilon_l) = \prod_{j \neq l} e^{-e^{-(V_l - V_j + \epsilon_l)}} \text{ and } P_l = \int_{-\infty}^{+\infty} \prod_{j \neq l} e^{-e^{-(V_l - V_j + t)}} e^{-t} e^{-e^{-t}} dt.$$

The probabilities have then simple, closed forms, which correspond to the logit transformation of the deterministic part of the utility:

$$P_l = \frac{e^{V_l}}{\sum_{j=1}^{J} e^{V_j}}.$$

3.3. Independence of irrelevant alternatives property

If we consider the probabilities of choice for two alternatives l and m, we have $P_l = e^{V_l} / \sum_j e^{V_j}$ and $P_m = e^{V_m} / \sum_j e^{V_j}$. The ratio of these two probabilities is:

$$\frac{P_l}{P_m} = \frac{e^{V_l}}{e^{V_m}} = e^{V_l - V_m}.$$

This probability ratio for the two alternatives depends only on the characteristics of these two alternatives and not on those of other alternatives. This is called the IIA property (for independence of irrelevant alternatives). IIA relies on the hypothesis that the errors are identical and independent. It is not a problem by itself and may even be considered as a useful feature for a well specified model. However, this hypothesis may be in practice violated, especially if some important variables are omitted.

3.4. Interpretation

In a linear model, the coefficients are the marginal effects of the explanatory variables on the explained variable. This is not the case for the multinomial logit model. However, meaningful results can be obtained using relevant transformations of the coefficients.

Marginal effects

The marginal effects are the derivatives of the probabilities with respect to the covariates, which can be choice situation specific (z_i) or alternative specific (x_{ij}):

$$\begin{array}{rcl} \dfrac{\partial P_{il}}{\partial z_i} & = & P_{il} \left(\beta_l - \sum_j P_{ij} \beta_j \right) \\[2mm] \dfrac{\partial P_{il}}{\partial x_{il}} & = & \gamma P_{il}(1 - P_{il}) \\[2mm] \dfrac{\partial P_{il}}{\partial x_{ik}} & = & -\gamma P_{il} P_{ik}. \end{array}$$

- For a choice situation specific variable, the sign of the marginal effect is not necessarily the sign of the coefficient. Actually, the sign of the marginal effect is given by $\left(\beta_l - \sum_j P_{ij}\beta_j\right)$, which is positive if the coefficient for alternative l is greater than a weighted average of the coefficients for all the alternatives, the weights being the probabilities of choosing the alternatives. In this case, the sign of the marginal effect can be established with no ambiguity only for the alternatives with the lowest and the greatest coefficients.

- For an alternative specific variable, the sign of the coefficient can be directly interpreted. The marginal effect is obtained by multiplying the coefficient by the product of two probabilities which is at most 0.25. The rule of thumb is therefore to divide the coefficient by four in order to have an upper bound of the marginal effect.

Note that the last equation can be rewritten: $\frac{\partial P_{il}/P_{il}}{\partial x_{ik}} = -\gamma P_{ik}$. Therefore, when a characteristic of alternative k changes, the relative change of the probabilities for every alternatives except k are the same, which is a consequence of the IIA property.

Marginal rates of substitution

Coefficients are marginal utilities, which cannot be interpreted. However, ratios of coefficients are marginal rates of substitution. For example, if the observable part of utility is: $V = \beta_0 + \beta_1 x_1 + \beta x_2 + \beta x_3$, joint variations of x_1 and x_2 which ensure the same level of utility satisfy $dV = \beta_1 dx_1 + \beta_2 dx_2 = 0$ so that:

$$-\frac{dx_2}{dx_1} \mid_{dV=0} = \frac{\beta_1}{\beta_2}.$$

For example, if x_1 is transport time (in hours) and x_2 is transport cost (in \$), $\beta_1 = 1.5$ and $\beta_2 = 0.05$, $\frac{\beta_1}{\beta_2} = 30$ is the marginal rate of substitution of time in terms of \$ and the value of 30 means that to reduce the travel time of one hour, the individual is willing to pay at most 30\$ more. Stated more simply, time value is 30\$ per hour.

Consumer's surplus

Consumer's surplus has a very simple expression for multinomial logit models, which was first derived by Small and Rosen (1981). The level of utility attained by an individual is $U_j = V_j + \epsilon_j$, j being the chosen alternative. The expected utility, from the searcher's point of view is then $\mathsf{E}(\max_j U_j)$, where the expectation is taken over the values of all the error terms. Its expression is simply, up to an additive unknown constant, the logarithm of the denominator of the logit probabilities, often called the "log-sum":

$$\mathsf{E}(U) = \ln \sum_{j=1}^{J} e^{V_j} + C.$$

If the marginal utility of income α is known and constant, the expected surplus is simply $\mathsf{E}(U)/\alpha$.

3.5. Application

Random utility models are fitted using the `mlogit` function. Basically, only two arguments are mandatory, `formula` and `data`, if an object of class 'dfidx' (and not an ordinary `data.frame`) is provided.

`ModeCanada`

We first use the `ModeCanada` data set, which was already coerced to a 'dfidx' object (called MC) in the previous section. The same model can then be estimated using as `data` argument this 'dfidx' object:

```
R> ml.MC1 <- mlogit(choice ~ cost + freq + ovt | income | ivt, MC)
```

or a `data.frame`. In the latter case, further arguments that will be subsequently passed to `dfidx` should be indicated:

```
R> ml.MC1b <- mlogit(choice ~ cost + freq + ovt | income | ivt, ModeCanada,
+    subset = noalt == 4, idx = c("case", "alt"))
```

`mlogit` provides two further useful arguments:

- `reflevel` indicates which alternative is the "reference" alternative, i.e., the one for which the coefficients of choice situation specific covariates are set to zero,

- `alt.subset` indicates a subset of alternatives on which the estimation has to be performed; in this case, only the lines that correspond to the selected alternatives are used and all the choice situations where not selected alternatives have been chosen are removed.

We estimate the model on the subset of three alternatives (we exclude `bus` whose market share is negligible in our sample) and we set `car` as the reference alternative. Moreover, we use a total transport time variable computed as the sum of the in vehicle and the out of vehicle time variables.

```
R> MC$time <- with(MC, ivt + ovt)
R> ml.MC1 <- mlogit(choice ~ cost + freq | income | time, MC,
+    alt.subset = c("car", "train", "air"), reflevel = "car")
```

The main results of the model are computed and displayed using the `summary` method:

```
R> summary(ml.MC1)

Call:
mlogit(formula = choice ~ cost + freq | income | time, data = MC,
    alt.subset = c("car", "train", "air"), reflevel = "car",
    method = "nr")

Frequencies of alternatives:
```

```
    car    train      air
0.45757 0.16721 0.37523


nr method
6 iterations, 0h:0m:0s
g'(-H)^-1g = 6.94E-06
successive function values within tolerance limits

Coefficients :
                     Estimate  Std. Error  z-value  Pr(>|z|)
(Intercept):train -0.97034440  0.26513065  -3.6599 0.0002523 ***
(Intercept):air   -1.89856552  0.68414300  -2.7751 0.0055185 **
cost              -0.02849715  0.00655909  -4.3447 1.395e-05 ***
freq               0.07402902  0.00473270  15.6420 < 2.2e-16 ***
income:train      -0.00646892  0.00310366  -2.0843 0.0371342 *
income:air         0.02824632  0.00365435   7.7295 1.088e-14 ***
time:car          -0.01402405  0.00138047 -10.1589 < 2.2e-16 ***
time:train        -0.01096877  0.00081834 -13.4036 < 2.2e-16 ***
time:air          -0.01755120  0.00399181  -4.3968 1.099e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Log-Likelihood: -1951.3
McFadden R^2:  0.31221
Likelihood ratio test : chisq = 1771.6 (p.value = < 2.22e-16)
```

The frequencies of the different alternatives in the sample are first indicated. Next, some information about the optimization is displayed: the Newton-Raphson method (with analytic gradient and Hessian) is used, as it is the most efficient method for this simple model for which the log-likelihood function is globally concave. Note that very few iterations and computing time are required to estimate this model. Then the usual table of coefficients is displayed followed by some goodness of fit measures: the value of the log-likelihood function, which is compared to the value for the model containing only intercepts, which leads to the computation of the McFadden R^2 and to the likelihood ratio test.

The `fitted` method can be used either to obtain the probability of actual choices (`type = "outcome"`) or the probabilities for all the alternatives (`type = "probabilities"`).

```
R> head(fitted(ml.MC1, type = "outcome"))

      109       110       111       112       113       114
0.1909475 0.3399941 0.1470527 0.3399941 0.3399941 0.2440011


R> head(fitted(ml.MC1, type = "probabilities"), 4)

          car     train       air
109 0.4206404 0.3884120 0.1909475
```

```
110 0.3696476 0.2903582 0.3399941
111 0.4296769 0.4232704 0.1470527
112 0.3696476 0.2903582 0.3399941
```

Note that the log-likelihood is the sum of the logarithm of the fitted outcome probabilities and that, as the model contains intercepts, the average fitted probabilities for every alternative equal the market shares of the alternatives in the sample.

```
R> sum(log(fitted(ml.MC1, type = "outcome")))
```

```
[1] -1951.344
```

```
R> logLik(ml.MC1)
```

```
'log Lik.' -1951.344 (df=9)
```

```
R> apply(fitted(ml.MC1, type = "probabilities"), 2, mean)
```

```
      car     train       air
0.4575659 0.1672084 0.3752257
```

Predictions can be made using the `predict` method. If no data is provided, predictions are made for the sample mean values of the covariates.

```
R> predict(ml.MC1)
```

```
      car     train       air
0.5066362 0.2116876 0.2816761
```

Assume, for example, that we wish to predict the effect of a reduction of train transport time of 20%. We first create a new data frame simply by multiplying train transport time by 0.8 and then using the `predict` method on this new data frame.

```
R> NMC <- MC
R> NMC[idx(NMC)$alt == "train", "time"] <- 0.8 *
+    NMC[idx(NMC)$alt == "train", "time"]
R> Oprob <- fitted(ml.MC1, type = "probabilities")
R> Nprob <- predict(ml.MC1, newdata = NMC)
R> rbind(old = apply(Oprob, 2, mean), new = apply(Nprob, 2, mean))
```

```
          car     train       air
old 0.4575659 0.1672084 0.3752257
new 0.4044736 0.2635801 0.3319462
```

If, for the first individuals in the sample, we compute the ratio of the probabilities of the `air` and the `car` mode, we obtain:

```
R> head(Nprob[, "air"] / Nprob[, "car"])
```

```
      109       110       111       112       113       114
0.4539448 0.9197791 0.3422401 0.9197791 0.9197791 0.6021092
```

```
R> head(Oprob[, "air"] / Oprob[, "car"])
```

```
      109       110       111       112       113       114
0.4539448 0.9197791 0.3422401 0.9197791 0.9197791 0.6021092
```

which is an illustration of the IIA property. If train time changes, it changes the probabilities of choosing `air` and `car`, but not their ratio.

We next compute the surplus for individuals of the sample induced by train time reduction. This requires the computation of the log-sum term (also called inclusive value or inclusive utility) for every choice situation, which is:

$$\text{iv}_i = \ln \sum_{j=1}^{J} e^{\beta^\top x_{ij}}.$$

For this purpose, we use the `logsum` function, which works on a vector of `coefficients` and a `model.matrix`. The basic use of `logsum` consists on providing as unique argument (called `coef`) an 'mlogit' object. In this case, the `model.matrix` and the `coef` are extracted from the same model.

```
R> ivbefore <- logsum(ml.MC1)
```

To compute the log-sum after train time reduction, we must provide a `model.matrix` which is not the one corresponding to the fitted model. This can be done using the `X` argument which is a matrix or an object from which a `model.matrix` can be extracted. This can also be done by filling the `data` argument (a `data.frame` or an object from which a `data.frame` can be extracted using a `model.frame` method), and eventually the `formula` argument (a `formula` or an object for which the `formula` method can be applied). If no formula is provided but if `data` is an 'dfidx' object, the formula is extracted from it.

```
R> ivafter <- logsum(ml.MC1, data = NMC)
```

Surplus variation is then computed as the difference of the log-sums divided by the opposite of the cost coefficient which can be interpreted as the marginal utility of income:

```
R> surplus <- - (ivafter - ivbefore) / coef(ml.MC1)["cost"]
R> summary(surplus)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.5852  2.8439  3.8998  4.6971  5.8437 31.3912
```

Consumer's surplus variation range from 0.6 to 31$, with a median value of about 4$.

Marginal effects are computed using the `effects` method. By default, they are computed at the sample mean, but a `data` argument can be provided. The variation of the probability and of the covariate can be either absolute or relative. This is indicated with the `type` argument which is a combination of two `a` (as absolute) and `r` (as relative) characters. For example, `type = "ar"` means that what is measured is an absolute variation of the probability for a relative variation of the covariate.

```
R> effects(ml.MC1, covariate = "income", type = "ar")
```

```
        car       train         air
-0.1822177 -0.1509079   0.3331256
```

The results indicate that, for a 100% increase of income, the probability of choosing `air` increases by 33 percentage points, while the probabilities of choosing `car` and `train` decrease by 18 and 15 percentage points.

For an alternative specific covariate, a matrix of marginal effects is displayed.

```
R> effects(ml.MC1, covariate = "cost", type = "rr")
```

```
            car       train         air
car   -0.9131273   0.9376923   0.9376923
train  0.3358005  -1.2505014   0.3358005
air    1.2316679   1.2316679  -3.1409703
```

The cell in the lth row and the cth column indicates the change of the probability of choosing alternative c when the cost of alternative l changes. As `type = "rr"`, elasticities are computed. For example, a 10% change of train cost increases the probabilities of choosing car and air travel by 3.36%. Note that the relative changes of the probabilities of choosing one of these two modes are equal, which is a consequence of the IIA property.

Finally, in order to compute travel time valuation, we divide the coefficients of travel times (in minutes) by the coefficient of monetary cost (in $).

```
R> coef(ml.MC1)[grep("time", names(coef(ml.MC1)))] /
+    coef(ml.MC1)["cost"] * 60
```

```
  time:car time:train    time:air
  29.52728   23.09447    36.95360
```

The value of travel time ranges from 23 for train to 37$ per hour for plane.

NOx

The second example is a data set used by Fowlie (2010), called `NOx`. She analyzed the effect of an emissions trading program (the NOx budget program which seeks to reduce the emission of nitrogen oxides) on the behavior of producers. More precisely, coal electricity plant managers

may adopt one out of fifteen different technologies in order to comply with the emissions defined by the program. Some of them require high investment (the capital cost is `kcost`) and are very efficient to reduce emissions, some other require much less investment but are less efficient and the operating cost (denoted `vcost`) is then higher, especially because pollution permits must be purchased to offset emissions exceeding their allocation.

The focus of the paper is on the effects of the regulatory environment on the manager's behavior. Some firms are deregulated, whereas other are either regulated or public. Rate of return is applied for regulated firms, which means that they perceive a "fair" rate of return on their investment. Public firms also enjoy significant cost of capital advantages. Therefore, the main hypothesis of the paper is that public and regulated firms will adopt much more capitalistic intensive technologies than deregulated and public ones, which means that the coefficient of capital cost should take a higher negative value for deregulated firms. Capital cost is interacted with the age of the plant (measured as a deviation from the sample mean age), as firms should weigh capital costs more heavily for older plants, as they have less time to recover these costs.

Multinomial logit models are estimated for the three sub-samples defined by the regulatory environment. The 15 technologies are not available for every plant, the sample is therefore restricted to available technologies, using the `available` covariate. Three technology dummies are introduced: `post` for post-combustion pollution control technology, `cm` for combustion modification technology and `lnb` for low NOx burners technology.

A last model is estimated for the whole sample, but the parameters are allowed to be proportional to each other. The scedasticity function is described in the fourth part of the formula, it contains here only one covariate, `env`. Note also that for the last model, the author introduced a specific capital cost coefficient for deregulated firms.[7]

```
R> data("NOx", package = "mlogit")
R> NOx$kdereg <- with(NOx, kcost * (env == "deregulated"))
R> NOxml <- dfidx(NOx, idx = list(c("chid", "id"), "alt"))
R> ml.pub <- mlogit(choice ~ post + cm + lnb + vcost + kcost + kcost:age |
+    - 1, subset = available & env == "public", data = NOxml)
R> ml.reg <- update(ml.pub, subset = available & env == "regulated")
R> ml.dereg <- update(ml.pub, subset = available & env == "deregulated")
R> ml.pool <- mlogit(choice ~ post + cm + lnb + vcost + kcost + kcost:age +
+    kdereg | - 1 | 0 | env, subset = available == 1, data = NOxml,
+    method = "bhhh")
R> library("texreg")
R> texreg(list(Public = ml.pub, Deregulated = ml.dereg, Regulated = ml.reg,
+    Pooled = ml.pool), caption = "Environmental compliance choices.",
+    omit.coef = "(post)|(cm)|(lnb)", float.pos = "t!", label = "tab:nox",
+    dcolumn = TRUE)
```

Results are presented in Table 1, using the **texreg** package (Leifeld 2013). Coefficients are very different on the sub-samples defined by the regulatory environment. Note in particular that the capital cost coefficient is positive and insignificant for public and regulated firms, as it is significantly negative for deregulated firms. Errors seems to have significant larger

---

[7]Note the use of the `method` argument, set to `"bhhh"`. See Section 5.3 for details.

	Public	Deregulated	Regulated	Pooled
vcost	-1.56^{***}	-0.19^{***}	-0.28^{***}	-0.31^{***}
	(0.36)	(0.06)	(0.06)	(0.04)
kcost	0.04	-0.06^{**}	0.01	0.01
	(0.11)	(0.02)	(0.03)	(0.02)
kcost:age	-0.08	-0.04^{**}	-0.02^{*}	-0.02^{***}
	(0.04)	(0.01)	(0.01)	(0.01)
kdereg				-0.07^{***}
				(0.01)
sig.envderegulated				0.32^{**}
				(0.12)
sig.envpublic				-0.33^{***}
				(0.08)
AIC	168.92	690.15	731.48	1634.22
Log Likelihood	-78.46	-339.07	-359.74	-808.11
Num. obs.	113	227	292	632

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Table 1: Environmental compliance choices.

variance for deregulated firms and lower ones for public firms compared to regulated firms. The hypothesis that the coefficients (except the `kcost` one) are identical up to a multiplicative scalar can be performed using a likelihood ratio test:

```
R> stat <- 2 * (logLik(ml.dereg) + logLik(ml.reg) + logLik(ml.pub) -
+     logLik(ml.pool))
R> stat

'log Lik.' 61.67179 (df=6)

R> pchisq(stat, df = 9, lower.tail = FALSE)

'log Lik.' 6.377284e-10 (df=6)
```

The hypothesis is strongly rejected.

4. Logit models relaxing the i.i.d. hypothesis

In the previous section, we assumed that the error terms were i.i.d., i.e., uncorrelated and homoscedastic. Extensions of the basic multinomial logit model have been proposed by relaxing one of these two hypotheses while maintaining the hypothesis of a Gumbel distribution.

4.1. The heteroscedastic logit model

The heteroscedastic logit model was proposed by Bhat (1995). The probability that $U_l > U_j$ is:

$$\mathsf{P}(\epsilon_j < V_l - V_j + \epsilon_l) = e^{-e^{-\frac{(V_l - V_j + \epsilon_l)}{\theta_j}}},$$

which implies the following conditional and unconditional probabilities

$$(P_l \mid \epsilon_l) = \prod_{j \neq l} e^{-e^{-\frac{(V_l - V_j + \epsilon_l)}{\theta_j}}},$$

$$
\begin{aligned}
P_l &= \int_{-\infty}^{+\infty} \prod_{j \neq l} \left(e^{-e^{-\frac{(V_l - V_j + t)}{\theta_j}}} \right) \frac{1}{\theta_l} e^{-\frac{t}{\theta_l}} e^{-e^{-\frac{t}{\theta_l}}} dt \\
&= \int_0^{+\infty} \left(e^{-\sum_{j \neq l} e^{-\frac{V_l - V_j - \theta_l \ln t}{\theta_j}}} \right) e^{-t} dt.
\end{aligned}
$$

There is no closed form for this integral, but it can be efficiently computed using a Gauss quadrature method, and more precisely the Gauss-Laguerre quadrature method.

4.2. The nested logit model

The nested logit model was first proposed by McFadden (1978). It is a generalization of the multinomial logit model that is based on the idea that some alternatives may be joined in several groups (called nests). The error terms may then present some correlation in the same nest, whereas error terms of different nests are still uncorrelated.

Denoting $m = 1 \ldots M$ the nests and B_m the set of alternatives belonging to nest m, the cumulative distribution of the errors is:

$$\exp \left(-\sum_{m=1}^M \left(\sum_{j \in B_m} e^{-\epsilon_j / \lambda_m} \right)^{\lambda_m} \right).$$

The marginal distributions of the ϵ's are still univariate extreme value, but there is now some correlation within nests where $1 - \lambda_m$ is a measure of the correlation, i.e., $\lambda_m = 1$ implies no correlation. In the special case where $\lambda_m = 1 \; \forall m$, the errors are i.i.d. Gumbel errors and the nested logit model reduces to the multinomial logit model. It can then be shown that the probability of choosing alternative j that belongs to nest l is:

$$P_j = \frac{e^{V_j / \lambda_l} \left(\sum_{k \in B_l} e^{V_k / \lambda_l} \right)^{\lambda_l - 1}}{\sum_{m=1}^M \left(\sum_{k \in B_m} e^{V_k / \lambda_m} \right)^{\lambda_m}},$$

and that this model is a random utility model if all the λ parameters are in the 0–1 interval.[8]

Let us now write the deterministic part of the utility of alternative j as the sum of two terms: the first one (Z_j) being specific to the alternative and the second one (W_l) to the nest it belongs to:

$$V_j = Z_j + W_l.$$

---

[8]A slightly different version of the nested logit model (Daly 1987) is often used, but is not compatible with the random utility maximization hypothesis. Its difference from the previous expression is that the deterministic parts of the utility for each alternative are not divided by the nest elasticity. The differences between the two versions have been discussed in Koppelman and Wen (1998), Heiss (2002) and Hensher and Greene (2002).

We can then rewrite the probabilities as follows:

$$
\begin{aligned}
P_j &= \frac{e^{(Z_j+W_l)/\lambda_l}}{\sum_{k\in B_l} e^{(Z_k+W_l)/\lambda_l}} \times \frac{\left(\sum_{k\in B_l} e^{(Z_k+W_l)/\lambda_l}\right)^{\lambda_l}}{\sum_{m=1}^{M}\left(\sum_{k\in B_m} e^{(Z_k+W_m)/\lambda_m}\right)^{\lambda_m}} \\
&= \frac{e^{Z_j/\lambda_l}}{\sum_{k\in B_l} e^{Z_k/\lambda_l}} \times \frac{\left(e^{W_l/\lambda_l}\sum_{k\in B_l} e^{Z_k/\lambda_l}\right)^{\lambda_l}}{\sum_{m=1}^{M}\left(e^{W_m/\lambda_m}\sum_{k\in B_m} e^{Z_k/\lambda_m}\right)^{\lambda_m}}.
\end{aligned}
$$

Then denote $I_l = \ln\sum_{k\in B_l} e^{Z_k/\lambda_l}$ which is often called the log-sum, the inclusive value or the inclusive utility.[9] We then can write the probability of choosing alternative j as:

$$
P_j = \frac{e^{Z_j/\lambda_l}}{\sum_{k\in B_l} e^{Z_k/\lambda_l}} \times \frac{e^{W_l+\lambda_l I_l}}{\sum_{m=1}^{M} e^{W_m+\lambda_m I_m}}.
$$

The first term $P_{j|l}$ is the conditional probability of choosing alternative j if nest l is chosen. It is often referred as the *lower model*. The second term P_l is the marginal probability of choosing nest l and is referred as the *upper model*. $W_l + \lambda_l I_l$ can be interpreted as the expected utility of choosing the best alternative in l, W_l being the expected utility of choosing an alternative in this nest (whatever this alternative is) and $\lambda_l I_l$ being the expected extra utility gained by being able to choose the best alternative in the nest. The inclusive values link the two models. It is then straightforward to show that IIA applies within nests, but not for two alternatives in different nests.

A consistent but inefficient way of estimating the nested logit model is to estimate separately its two components. The coefficients of the lower model are first estimated, which enables the computation of the inclusive values I_l. The coefficients of the upper model are then estimated, using I_l as covariates. Maximizing directly the likelihood function of the nested model leads to a more efficient estimator.

4.3. Applications

ModeCanada

Bhat (1995) estimated the heteroscedastic logit model on the `ModeCanada` data set. Using package **mlogit**, the heteroscedastic logit model is obtained by setting the `heterosc` argument to `TRUE`:

```
R> ml.MC <- mlogit(choice ~ freq + cost + ivt + ovt | urban + income, MC,
+    reflevel = "car", alt.subset = c("car", "train", "air"))
R> hl.MC <- mlogit(choice ~ freq + cost + ivt + ovt | urban + income, MC,
+    reflevel = "car", alt.subset = c("car", "train", "air"),
+    heterosc = TRUE)
R> coef(summary(hl.MC))[11:12, ]

         Estimate Std. Error    z-value      Pr(>|z|)
sp.train 1.2371829  0.1104610 11.200182 0.000000e+00
sp.air   0.5403239  0.1118353  4.831425 1.355592e-06
```

---

[9]We have already encountered this expression in Section 3.4.

The variance of the error terms of `train` and `air` are respectively higher and lower than the variance of the error term of `car` (set to one). Note that the z values and p values of the output are not particularly meaningful, as the hypothesis that the coefficient is zero (and not one) is tested. The homoscedasticity hypothesis can be tested using any of the three tests. A particular convenient syntax is provided in this case. For the likelihood ratio and the Wald test, one can use only the fitted heteroscedastic model as argument. In this case, it is guessed that the hypothesis that the user wants to test is the homoscedasticity hypothesis.

```
R> lr.heter <- lrtest(hl.MC, ml.MC)
R> wd.heter <- waldtest(hl.MC, heterosc = FALSE)
```

or, more simply:

```
R> lrtest(hl.MC)
R> waldtest(hl.MC)
```

The Wald test can also be computed using the `linearHypothesis` function from the **car** package:

```
R> library("car")
R> lh.heter <- linearHypothesis(hl.MC, c("sp.air = 1", "sp.train = 1"))
```

For the score test, we provide the constrained model as argument, which is the standard multinomial logit model and the supplementary argument which defines the unconstrained model, which is in this case `heterosc = TRUE`.

```
R> sc.heter <- scoretest(ml.MC, heterosc = TRUE)
R> sapply(list(wald = wd.heter, lh = lh.heter, score = sc.heter,
+     lr = lr.heter), statpval)
```

```
          wald     lh score     lr
stat    25.196 25.196 9.488  6.888
p-value  0.000  0.000 0.009  0.032
```

The homoscedasticity hypothesis is strongly rejected using the Wald test, but only at the 1% and 5% level for, respectively, the score and the likelihood ratio tests.

JapaneseFDI

Head and Mayer (2004) analyzed the choice of one of the 57 European regions belonging to 9 countries by Japanese firms to implement a new production unit.

```
R> data("JapaneseFDI", package = "mlogit")
R> jfdi <- dfidx(JapaneseFDI, idx = list("firm", c("region", "country")),
+     idnames = c("chid", "alt"), drop.index = FALSE)
```

Note that the argument `idx` is a list of two elements, where the second element contains the names of the column containing the regions and the countries, indicating the nestedness of

regions within countries and where region is a grouping variable, feature used later to easily define the nests. There are two sets of covariates: the wage rate `wage`, the unemployment rate `unemp`, a dummy indicating whether the region is eligible to European funds `elig` and the area `area` are observed at the regional level and are therefore relevant for the estimation of the lower model, whereas the social charges rate `scrate` and the corporate tax rate `ctaxrate` are observed at the country level and are therefore suitable for the upper model.

We first estimate a multinomial logit model:

```
R> ml.fdi <- mlogit(choice ~ log(wage) + unemp + elig + log(area) + scrate +
+    ctaxrate | 0, data = jfdi)
```

Note that, as the covariates are only alternative specific, the intercepts are not identified and therefore have been removed. We next estimate the lower model, which analyzes the choice of a region within a given country. Therefore, for each choice situation, we estimate the choice of a region on the subset of regions of the country which has been chosen. Moreover, observations concerning Portugal and Ireland are removed as these two countries are mono-regions.

```
R> lm.fdi <- mlogit(choice ~ log(wage) + unemp + elig + log(area) | 0,
+    data = jfdi, subset = country == choice.c & !country %in% c("PT", "IE"))
```

We next use the fitted lower model in order to compute the inclusive value at the country level:

$$\text{iv}_{ig} = \ln \sum_{j \in B_g} e^{\beta^\top x_{ij}},$$

where B_g is the set of regions for country g. When a grouping variable is provided in the `dfidx` function, inclusive values are by default computed for every group g (global inclusive values are obtained by setting the `type` argument to `"global"`). By default, `output` is set to `"chid"` and the result is a vector (if `type = "global"`) or a matrix (if `type = "region"`) with row number equal to the number of choice situations. If `output` is set to `"obs"`, a vector of length equal to the number of lines of the data in "long" format is returned. The following code indicates different ways to use the `logsum` function:

```
R> lmformula <- formula(lm.fdi)
R> head(logsum(ml.fdi, data = jfdi, formula = lmformula, type = "group"), 2)

        BE       DE       ES       FR       IE       IT       NL       PT
3 3.595818 5.415838 3.593702 5.153709 1.933707 5.051387 4.077845 2.702028
4 4.113243 5.765190 4.445012 5.383095 1.960462 5.687569 4.490379 3.200124
        UK
3 4.900622
4 5.378561


R> head(logsum(ml.fdi, data = jfdi, formula = lmformula, type = "global"))

       3        4        5        7        8        9
6.736116 7.182139 7.121855 7.084245 7.133368 7.133368
```

```
R> head(logsum(ml.fdi, data = jfdi, formula = lmformula, output = "obs"))
```

```
[1] 3.595818 3.595818 3.595818 3.595818 5.415838 5.415838
```

```
R> head(logsum(ml.fdi, data = jfdi, formula = lmformula, type = "global",
+    output = "obs"))
```

```
[1] 6.736116 6.736116 6.736116 6.736116 6.736116 6.736116
```

To add the inclusive values in the original `data.frame`, we use `output = "obs"` and the `type` argument can be omitted as its default value is `"group"`:

```
R> JapaneseFDI$iv <- logsum(lm.fdi, data = jfdi, formula = lmformula,
+    output = "obs")
```

We next select the relevant variables for the estimation of the upper model, select unique lines in order to keep only one observation for every choice situation / country combination and finally we coerce the response (`choice.c`) to a logical for the chosen country.

```
R> JapaneseFDI.c <- subset(JapaneseFDI,
+    select = c("firm", "country", "choice.c", "scrate", "ctaxrate", "iv"))
R> JapaneseFDI.c <- unique(JapaneseFDI.c)
R> JapaneseFDI.c$choice.c <- with(JapaneseFDI.c, choice.c == country)
```

Finally, we estimate the upper model, using the previously computed inclusive value as a covariate.

```
R> jfdi.c <- dfidx(JapaneseFDI.c, choice = "choice.c",
+    idnames = c("chid", "alt"))
R> um.fdi <- mlogit(choice.c ~ scrate + ctaxrate + iv | 0, data = jfdi.c)
```

If one wants to obtain different `iv` coefficients for different countries, the `iv` covariate should be introduced in the third part of the formula and the coefficients for the two mono-region countries (Ireland and Portugal) should be set to one using the `constPar` argument.

```
R> um2.fdi <- mlogit(choice.c ~ scrate + ctaxrate | 0 | iv, data = jfdi.c,
+    constPar = c("iv:PT" = 1, "iv:IE" = 1))
```

We next estimate the full-information maximum likelihood nested model. It is obtained by adding a `nests` argument to the `mlogit` function. This should be a named list of alternatives (here regions), the names being the nests (here the countries). More simply, if a group variable has been indicated while using `dfidx`, `nests` can be a Boolean.

Two flavors of nested models can be estimated, using the `un.nest.el` argument which is a Boolean. If `TRUE`, one imposes that the coefficient associated with the inclusive utility is the same for every nest, which means that the degree of correlation inside each nest is the same. If `FALSE`, a different coefficient is estimated for every nest.

	Mult. logit	Lower model	Upper model	Upper model	Nested logit (1)	Nested logit (2)
log(wage)	0.47	1.21*			0.46	0.77**
	(0.25)	(0.48)			(0.25)	(0.25)
unemp	−8.90***	−9.77***			−7.62***	−6.95**
	(1.69)	(2.39)			(1.60)	(2.28)
elig	−0.25	−0.89**			−0.34	−0.18
	(0.21)	(0.33)			(0.20)	(0.18)
log(area)	0.31***	0.29***			0.29***	0.15**
	(0.05)	(0.06)			(0.05)	(0.05)
scrate	−2.26***		−2.49***	0.15	−2.44***	−0.88
	(0.38)		(0.38)	(1.72)	(0.38)	(0.86)
ctaxrate	−4.82***		−3.69***	−1.78	−4.13***	−2.35
	(0.59)		(0.60)	(1.32)	(0.66)	(1.25)
iv			0.66***		0.85***	
			(0.06)		(0.08)	
iv:BE				0.72***		0.48*
				(0.08)		(0.19)
iv:DE				0.72***		0.52**
				(0.08)		(0.17)
iv:ES				0.86***		0.77***
				(0.05)		(0.20)
iv:FR				0.75***		0.67***
				(0.04)		(0.09)
iv:IT				0.62***		0.25**
				(0.05)		(0.08)
iv:NL				0.69***		0.18
				(0.06)		(0.10)
iv:UK				0.87***		0.86***
				(0.06)		(0.10)
AIC	3469.13	1738.04	1746.58	1713.86	3467.96	3437.01
Log Lik.	−1728.57	−865.02	−870.29	−845.93	−1726.98	−1703.50
Num. obs.	452	421	452	452	452	452

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

Table 2: Choice by Japanese firms of a European region.

```
R> nl.fdi <- mlogit(choice ~ log(wage) + unemp + elig + log(area) + scrate +
+    ctaxrate | 0, data = jfdi, nests = TRUE, un.nest.el = TRUE)
R> nl2.fdi <- update(nl.fdi, un.nest.el = FALSE, constPar = c("iv:PT" = 1,
+    "iv:IE" = 1))
```

The results of the fitted models are presented in Table 2.

```
R> library("texreg")
R> texreg(list("Mult. logit" = ml.fdi, "Lower model" = lm.fdi,
+    "Upper model" = um.fdi, "Upper model" = um2.fdi,
+    "Nested logit (1)" = nl.fdi, "Nested logit (2)" = nl2.fdi),
+    fontsize = "small", float.pos = "t!",  label = "tab:nlogit",
+    caption = "Choice by Japanese firms of a European region.",
+    dcolumn = TRUE)
```

For the nested logit models, two tests are of particular interest:

- the test of no nests, which means that all the nest elasticities are equal to one,

- the test of unique nest elasticities, which means that all the nest elasticities are equal to each other.

For the test of no nests, the nested model is provided as the unique argument for the `lrtest` and the `waldtest` function. For the `scoretest`, the constrained model (i.e., the multinomial logit model) is provided as the first argument and the second argument is `nests`, which describes the nesting structure that one wants to test.

```
R> lr.nest <- lrtest(nl2.fdi)
R> wd.nest <- waldtest(nl2.fdi)
R> sc.nest <- scoretest(ml.fdi, nests = TRUE, constPar = c("iv:PT" = 1,
+    "iv:IE" = 1))
```

The Wald test can also be performed using the `linearHypothesis` function:

```
R> lh.nest <- linearHypothesis(nl2.fdi, c("iv:BE = 1", "iv:DE = 1",
+    "iv:ES = 1", "iv:FR = 1", "iv:IT = 1", "iv:NL = 1", "iv:UK = 1"))
R> sapply(list(wald = wd.nest, lh = lh.nest, score = sc.nest, lr = lr.nest),
+    statpval)
```

```
            wald      lh score      lr
stat     208.407 208.407 60.28 50.122
p-value    0.000   0.000  0.00  0.000
```

The three tests reject the null hypothesis of no correlation. We next test the hypothesis that all the nest elasticities are equal.

```
R> lr.unest <- lrtest(nl2.fdi, nl.fdi)
R> wd.unest <- waldtest(nl2.fdi, un.nest.el = TRUE)
R> sc.unest <- scoretest(ml.fdi, nests = TRUE, un.nest.el = FALSE,
+    constPar = c("iv:IE" = 1, "iv:PT" = 1))
R> lh.unest <- linearHypothesis(nl2.fdi, c("iv:BE = iv:DE", "iv:BE = iv:ES",
+    "iv:BE = iv:FR", "iv:BE = iv:IT", "iv:BE = iv:NL", "iv:BE = iv:UK"))
R> sapply(list(wald = wd.unest, lh = lh.unest, score = sc.unest,
+    lr = lr.unest), statpval)
```

```
            wald     lh score      lr
stat      73.535 73.535 60.28 46.954
p-value    0.000  0.000  0.00  0.000
```

Once again, the three tests strongly reject the hypothesis.

5. The random parameters (or mixed) logit model

5.1. Derivation of the model

A mixed logit model or random parameters logit model is a logit model for which the parameters are assumed to vary from one individual to another. It is therefore a model that takes the heterogeneity of the population into account.

The probabilities

For the standard logit model, the probability that individual i chooses alternative j is:

$$P_{il}(x_i, \beta) = \frac{e^{\beta^\top x_{il}}}{\sum_j e^{\beta^\top x_{ij}}}.$$

Suppose now that the coefficients are individual-specific. The probabilities are then:

$$\frac{e^{\beta_i^\top x_{il}}}{\sum_j e^{\beta_i^\top x_{ij}}}.$$

A first approach consists of estimating the parameters for every individual. However, these parameters are identified and can be consistently estimated only if a large number of choice situations per individual is available, which is scarcely the case. A more appealing approach consists on considering the β_i's as random draws from a distribution whose parameters are estimated, which leads to the mixed logit model. The *conditional* probability that individual i will choose alternative l for a given value of β_i is:

$$L_{il}(x_i, \beta_i) = \frac{e^{\beta_i^\top x_{il}}}{\sum_j e^{\beta_i^\top x_{ij}}}.$$

To get the *unconditional* probability, we have to integrate out this conditional probability, using the density function of β. Suppose that $V_{il} = \alpha + \beta_i x_{il}$, i.e., there is only one individual-specific coefficient and that the density of β is $f(\beta, \theta)$, θ being the vector of the parameters of the distribution of β. The unconditional probability is then:

$$P_{il}(x_i, \theta) = \int_\beta \frac{e^{\beta x_{il}}}{\sum_j e^{\beta x_{ij}}} f(\beta, \theta) d\beta,$$

which is a one-dimensional integral that can be efficiently estimated by quadrature methods. If $V_{il} = \beta_i^\top x_{il}$ where β_i is a vector of length K and $f(\beta, \theta)$ is the joint density of the K individual-specific coefficients, the unconditional probability is:

$$P_{il}(x_i, \theta) = \int_{\beta_1} \int_{\beta_2} \dots \int_{\beta_K} \frac{e^{\beta^\top x_{il}}}{\sum_j e^{\beta^\top x_{ij}}} f(\beta, \theta) d\beta_1 d\beta_2 \dots d\beta_K.$$

This is a K-dimensional integral which cannot easily be estimated by quadrature methods. The only practical method is then to use simulations. More precisely, R draws of the parameters are taken from the distribution of β, the probability is computed for every draw and

the unconditional probability, which is the expected value of the conditional probabilities, is estimated by the average of the R probabilities.

Individual parameters

The expected value of a random coefficient ($\mathsf{E}(\beta)$) is simply estimated by the mean of the R draws on its distribution: $\bar{\beta} = \sum_{r=1}^{R} \beta_r$. Individual parameters are obtained by first computing the probabilities of the observed choice of i for every value of β_r:

$$P_{ir} = \frac{\sum_j y_{ij} e^{\beta_r^\top x_{ij}}}{\sum_j e^{\beta_r^\top x_{ij}}},$$

where y_{ij} is a dummy equal to one if i has chosen alternative j. The expected value of the parameter for an individual is then estimated by using these probabilities to weigh the β values:

$$\hat{\beta}_i = \frac{\sum_{r=1}^{R} P_{ir} \beta_r}{\sum_{r=1}^{R} P_{ir}}.$$

Panel data

If there are repeated observations for the same individuals, the longitudinal dimension of the data can be taken into account in the mixed logit model, assuming that the random parameters of individual i are the same for all their choice situations. Denoting y_{itl} a dummy equal to one if i chooses alternative l for the tth choice situation, the conditional probability of the observed choice is:

$$L_{it}(x_{it}, y_{it}, \beta_i) = \prod_j \frac{\sum_j y_{itj} e^{\beta_i^\top x_{itl}}}{\sum_j e^{\beta_i^\top x_{itj}}}.$$

The joint conditional probability for the T observations of individual i is then:

$$L_i(x_i, y_i, \beta_i) = \prod_t \prod_j \frac{\sum_j y_{itj} e^{\beta_i^\top x_{itj}}}{\sum_j e^{\beta_i^\top x_{itj}}},$$

which leads to the following unconditional probabilities:

$$P_i(x_i, y_i, \theta) = \int_{\beta_1} \int_{\beta_2} \cdots \int_{\beta_K} \prod_t \prod_j \frac{\sum_j y_{itj} e^{\beta^\top x_{itj}}}{\sum_j e^{\beta^\top x_{itj}}} f(\beta, \theta) d\beta_1 d\beta_2 \ldots d\beta_K.$$

5.2. Application

The random parameter logit model is estimated by providing a `rpar` argument to the function `mlogit`. This argument is a named vector, the names being the random coefficients and the values the name of the law of distribution. Currently, the normal (`"n"`), log-normal (`"ln"`), zero-censored normal (`"cn"`), uniform (`"u"`) and triangular (`"t"`) distributions are available. For these distributions, two parameters are estimated which are, for normal related distributions, the mean and the standard-deviation of the underlying normal distribution and

for the uniform and triangular distribution, the mean and the half range of the distribution. For these last two distributions, zero-bounded variants are also provided (`"zbt"` and `"zbu"`). These two zero-bounded distributions are defined by only one parameter (the mean) and their definition domain varies from zero to twice the mean.

Several considerations may lead to the choice of a specific distribution:

- if correlated coefficients are required, the natural choice is a (transformed-) normal distribution, `"n"`, `"ln"` and `"cn"`,

- it is often the case that one wants to impose that the distribution of a random parameter takes only positive or negative values. For example, the price coefficient should be negative for every individual. In this case, `"zbt"` and `"zbu"` can be used. The use of `"ln"` and `"cn"` can also be relevant but, in this case, if only negative values are expected, one should consider the distribution of the opposite of the random price coefficient. This can easily be done using the `opposite` argument of the function `dfidx`[10],

- the use of unbounded distributions often leads to implausible values of some statistics of the random parameters, especially the mean. This is particularly the case for the log-normal distribution, which has an heavy left tail. In this case, the use of bounded distributions like the uniform and the triangular distributions can be used.

Furthermore, the argument `R` is the number of draws, `halton` indicates whether Halton draws (see Train 2009, Chapter 9) should be used (`NA` and `NULL` indicate respectively that default Halton draws are used and that pseudo-random numbers are used), `panel` is a Boolean which indicates if the panel data version of the log-likelihood should be used.

Correlations between random parameters can be introduced only for normal-related distributed random parameters, using the `correlation` argument, as there is no obvious way to introduce correlation for random parameters following an uniform or a triangular distribution. If `TRUE`, all the normal-related random parameters are correlated. The `correlation` argument can also be a character vector indicating the random parameters that are assumed to be correlated.

Train

We use the `Train` data set, previously coerced to an object of class 'dfidx' called `Tr`. We first estimate the multinomial model: both alternatives being virtual train trips, it is relevant to use only generic coefficients and to remove the intercept:

```
R> Train.ml <- mlogit(choice ~ price + time + change + comfort | - 1, Tr)
R> coef(summary(Train.ml))
```

	Estimate	Std. Error	z-value	Pr(>|z|)
price	0.06735804	0.003393252	19.850585	0.000000e+00
time	1.72055142	0.160351702	10.729861	0.000000e+00
change	0.32634094	0.059489152	5.485722	4.117843e-08
comfort	0.94572555	0.064945464	14.561842	0.000000e+00

---

[10]See Section 2.1.

All the coefficients are highly significant and have the predicted positive sign (remember that an increase in the variable `comfort` implies using a less comfortable class). The coefficients cannot be directly interpreted, but dividing them by the price coefficient, we get monetary values:

```
R> coef(Train.ml)[-1] / coef(Train.ml)[1]


    time    change   comfort
25.54337   4.84487  14.04028
```

We obtain the value of 26 euros for an hour of traveling, 5 euros for a change and 14 euros to travel in a more comfortable class. We then estimate a model with three random parameters: `time`, `change` and `comfort`. We first estimate the uncorrelated mixed logit model:

```
R> Train.mxlu <- mlogit(choice ~ price + time + change + comfort | - 1, Tr,
+    panel = TRUE, rpar = c(time = "n", change = "n", comfort = "n"),
+    R = 100, correlation = FALSE, halton = NA, method = "bhhh")
R> names(coef(Train.mxlu))


[1] "price"      "time"       "change"      "comfort"    "sd.time"
[6] "sd.change"  "sd.comfort"
```

Compared to the multinomial logit model, there are now three more coefficients which are the standard deviations of the distribution of the three random parameters. The correlated model is obtained by setting the `correlation` argument to `TRUE`.

```
R> Train.mxlc <- update(Train.mxlu, correlation = TRUE)
R> names(coef(Train.mxlc))


 [1] "price"              "time"               "change"
 [4] "comfort"            "chol.time:time"     "chol.time:change"
 [7] "chol.change:change" "chol.time:comfort"  "chol.change:comfort"
[10] "chol.comfort:comfort"
```

There are now six parameters which are the elements of the Choleski decomposition of the covariance matrix of the three random parameters. These six parameters are therefore the elements of the following matrix

$$C = \begin{pmatrix} c_{11} & 0 & 0 \\ c_{12} & c_{22} & 0 \\ c_{13} & c_{23} & c_{33} \end{pmatrix}$$

such that:

$$CC^\top = \begin{pmatrix} c_{11}^2 & c_{11}c_{12} & c_{11}c_{13} \\ c_{11}c_{12} & c_{12}^2 + c_{22}^2 & c_{12}c_{23} + c_{22}c_{23} \\ c_{11}c_{13} & c_{12}c_3 + c_{22}c_{23} & c_{13}^2 + c_{23}^2 c_{33}^2 \end{pmatrix} = \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_2^2 & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_3^2 \end{pmatrix},$$

where σ_i^2 and σ_{ij} are respectively the variance of the random parameter i and the covariance between two random parameters i and j. Therefore, the first estimated parameter can be simply interpreted as the standard deviation of the first random parameter, but the five other can not be interpreted easily.

Random parameters may be extracted using the function `rpar` which takes as first argument an 'mlogit' object and as second argument `par` the parameter(s) to be extracted. This function returns a 'rpar' object and a `summary` method is provided to describe it:

```
R> marg.ut.time <- rpar(Train.mxlc, "time")
R> summary(marg.ut.time)
```

```
   Min.  1st Qu.   Median      Mean  3rd Qu.      Max.
  -Inf 1.283749 4.893752 4.893752 8.503756      Inf
```

The estimated random parameter is in the "preference space", which means that it is the marginal utility of time.

Parameters in the "willingness to pay" (WTP) space are easier to interpret. They can be estimated directly (a feature not supported by **mlogit**) or can be obtained from the marginal utility by dividing it by the coefficient of a covariate expressed in monetary value (a price for example), taken as a non random parameter. The ratio can then be interpreted as a monetary value (or willingness to pay). To obtain the distribution of the random parameters in the WTP space, one can use the `norm` argument of `rpar`:

```
R> wtp.time <- rpar(Train.mxlc, "time", norm = "price")
R> summary(wtp.time)
```

```
    Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
   -Inf   8.753119 33.367588 33.367588 57.982056      Inf
```

The median value (and the mean value as the distribution is symmetric) of transport time is about 33 euros. Several methods/functions are provided to extract the individual statistics (`mean`, `med` and `stdev` respectively for the mean, the median and the standard deviation):

```
R> mean(rpar(Train.mxlc, "time", norm = "price"))
```

```
[1] 33.36759
```

```
R> med(rpar(Train.mxlc, "time", norm = "price"))
```

```
[1] 33.36759
```

```
R> stdev(rpar(Train.mxlc, "time", norm = "price"))
```

```
[1] 36.49347
```

In case of correlated random parameters, as the estimated parameters cannot be directly interpreted, a vcov method for 'mlogit' objects is provided. It has a what argument whose default value is "coefficient". In this case the usual covariance matrix of the coefficients is return. If what = "rpar", the covariance matrix of the correlated random parameters is returned if type = "cov" (the default) and the correlation matrix (with standard deviations on the diagonal) is returned if type = "cor". The object is of class 'vcov.mlogit' and a summary method for this object is provided which computes, using the delta method, the standard errors of the parameters of the covariance or the correlation matrix.

```
R> vcov(Train.mxlc, what = "rpar")


             time       change  comfort
time    28.6460389 -0.2787999 5.557933
change  -0.2787999  3.1047367 1.232467
comfort  5.5579334  1.2324667 7.895535


R> vcov(Train.mxlc, what = "rpar", type = "cor")


             time        change   comfort
time     5.35219945 -0.02956296 0.3695645
change  -0.02956296  1.76202630 0.2489270
comfort  0.36956453  0.24892701 2.8098994


R> summary(vcov(Train.mxlc, what = "rpar", type = "cor"))


                   Estimate Std. Error z-value  Pr(>|z|)
sd.time            5.352199   0.381135 14.0428 < 2.2e-16 ***
sd.change          1.762026   0.144592 12.1862 < 2.2e-16 ***
sd.comfort         2.809899   0.178295 15.7599 < 2.2e-16 ***
cor.time:change   -0.029563   0.232414 -0.1272  0.898782
cor.time:comfort   0.369565   0.114068  3.2399  0.001196 **
cor.change:comfort 0.248927   0.110321  2.2564  0.024047 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


R> summary(vcov(Train.mxlc, what = "rpar", type = "cov"))


                   Estimate Std. Error z-value  Pr(>|z|)
var.time           28.64604    4.07982  7.0214 2.197e-12 ***
var.change          3.10474    0.50955  6.0931 1.107e-09 ***
var.comfort         7.89553    1.00198  7.8799 3.276e-15 ***
cov.time:change    -0.27880    0.51550 -0.5408    0.5886
cov.time:comfort    5.55793    0.89161  6.2336 4.559e-10 ***
cov.change:comfort  1.23247    0.30131  4.0903 4.308e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The correlation can be restricted to a subset of random parameters by filling the `correlation` argument with a character vector indicating the corresponding covariates:

```
R> Train.mxlc2 <- update(Train.mxlc, correlation = c("time", "comfort"))
R> vcov(Train.mxlc2, what = "rpar", type = "cor")
```

```
            time    comfort
time    5.5726158 0.3909467
comfort 0.3909467 3.0631462
```

The presence of random coefficients and their correlation can be investigated using any of the three tests. Actually, three nested models can be considered, a model with no random effects, a model with random but uncorrelated effects and a model with random and correlated effects. We first present the three tests three tests of no correlated random effects:

```
R> lr.mxc <- lrtest(Train.mxlc, Train.ml)
R> wd.mxc <- waldtest(Train.mxlc)
R> lh.mxc <- linearHypothesis(Train.mxlc, c("chol.time:time = 0",
+    "chol.time:change =  0", "chol.time:comfort = 0",
+    "chol.change:change = 0", "chol.change:comfort = 0",
+    "chol.comfort:comfort = 0"))
R> sc.mxc <- scoretest(Train.ml, rpar = c(time = "n", change = "n",
+    comfort = "n"), R = 100, correlation = TRUE, halton = NA,
+    panel = TRUE)
R> sapply(list(wald = wd.mxc, lh = lh.mxc, score = sc.mxc, lr = lr.mxc),
+    statpval)
```

```
            wald      lh   score      lr
stat     288.287 288.287 208.765 388.057
p-value    0.000   0.000   0.000   0.000
```

The hypothesis of no correlated random parameters is strongly rejected. We then present the three tests of no correlation, the existence of random parameters being maintained.

```
R> lr.corr <- lrtest(Train.mxlc, Train.mxlu)
R> lh.corr <- linearHypothesis(Train.mxlc, c("chol.time:change = 0",
+    "chol.time:comfort = 0", "chol.change:comfort = 0"))
R> wd.corr <- waldtest(Train.mxlc, correlation = FALSE)
R> sc.corr <- scoretest(Train.mxlu, correlation = TRUE)
R> sapply(list(wald = wd.corr, lh = lh.corr, score = sc.corr, lr = lr.corr),
+    statpval)
```

```
            wald      lh   score      lr
stat     103.195 103.195  10.483  42.621
p-value    0.000   0.000   0.015   0.000
```

The hypothesis of no correlation is strongly rejected with the Wald and the likelihood ratio test, only at the 5% level for the score test.

RiskyTransport

The second example is a study by Leon and Miguel (2017) who consider a mode-choice model for transit from Freetown's airport (Sierra-Leone) to downtown. Four alternatives are available: ferry, helicopter, water-taxi and hovercraft. A striking characteristic of their study is that all these alternatives experienced fatal accidents in recent years, so that the fatality risk is non-negligible and differs much from an alternative to another. For example, the probabilities of dying using the water taxi and the helicopter are respectively of 2.55 and 18.41 out of 100000 passenger-trips. This feature enables the authors to estimate the value of a statistical life. For an individual i, the utility of choosing alternative j is:

$$U_{ij} = \beta_{il}(1 - p_j) + \beta_{ic}(c_j + w_i t_j) + \epsilon_{ij},$$

where p_j is the probability of dying while using alternative j, c_j and t_j are the monetary cost and the transport time of alternative j and w_i is the wage rate of individual i (which is supposed to be his valuation of transportation time); $C_{ij} = c_j + w_i t_j$ is therefore the individual specific generalized cost for alternative j, β_{il} and β_{ic} are the (individual specific) marginal utilities of surviving and of expense. The value of the statistical life (VSL) is then defined by:

$$\text{VSL}_i = -\frac{\beta_{il}}{\beta_{ic}} = \frac{\Delta C_{ij}}{\Delta(1 - p_j)}.$$

The two covariates of interest are `cost` (the generalized cost in \$PPP) and `risk` (mortality per 100000 passenger-trips). The `risk` variable being purely alternative specific, intercepts for the alternatives cannot therefore be estimated. To avoid endogeneity problems, the authors introduce as covariates marks the individuals gave to five attributes of the alternatives: comfort of the seat, noise level, crowdedness, convenience of the docking or landing location and the "quality" of the clientele. We first estimate a multinomial logit model.

```
R> data("RiskyTransport", package = "mlogit")
R> RT <- dfidx(RiskyTransport, choice = "choice",
+    idx = list(c("chid", "id"), "mode"), idnames = c("chid", "alt"))
R> ml.rt <- mlogit(choice ~ cost + risk  + seats + noise + crowdness +
+    convloc + clientele | 0, data = RT, weights = weight)
```

Note the use of the `weights` argument in order to set weights to the observations, as done in the original study.

```
R> coef(ml.rt)[c("risk", "cost")]

      risk         cost
-0.093907630 -0.009540895
```

The ratio of the coefficients of risk and of cost is 9.84 (hundred of thousands of \$), which means that the estimated value of the statistical life is a bit less than one million \$. We next consider a mixed logit model. The coefficients of `cost` and `risk` are assumed to be random, following a zero-bounded triangular distribution.

	Multinomial logit	Mixed logit
cost	−0.010***	−0.019***
	(0.001)	(0.001)
risk	−0.094***	−0.103***
	(0.011)	(0.016)
seats	0.152	0.108
	(0.244)	(0.233)
noise	−0.029	0.142
	(0.265)	(0.229)
crowdness	−0.919***	−0.716**
	(0.244)	(0.223)
convloc	−0.377	−0.150
	(0.202)	(0.197)
clientele	−0.257	−0.331
	(0.265)	(0.254)
AIC	3250.747	3177.250
Log Likelihood	−1618.374	−1581.625
Num. obs.	1793	1793

***$p < 0.001$; **$p < 0.01$; *$p < 0.05$

Table 3: Transportation choices.

```
R> mx.rt <- mlogit(choice ~ cost + risk  + seats + noise + crowdness +
+     convloc + clientele | 0, data = RT, weights = weight,
+     rpar = c(cost = "zbt", risk = "zbt"), R = 100, halton = NA,
+     panel = TRUE)
```

The results are presented in Table 3.

```
R> texreg(list("Multinomial logit" = ml.rt, "Mixed logit" = mx.rt),
+     digits = 3, float.pos = "t!", label = "tab:risktr",
+     dcolumn = TRUE, caption = "Transportation choices.")
```

Note that the log-likelihood is much larger for the mixed effect logit. Individual-level parameters can be extracted using the `fitted` method, with the type argument set to `parameters`.

```
R> indpar <- fitted(mx.rt, type = "parameters")
R> head(indpar)


        id         cost        risk
1 8020605 -0.02096705 -0.10105817
2 8260102 -0.01666475 -0.11211057
3 8260104 -0.01728864 -0.08302831
4 8260106 -0.01494848 -0.07319789
5 8260108 -0.01687051 -0.06501337
6 8260201 -0.02133116 -0.10285434
```
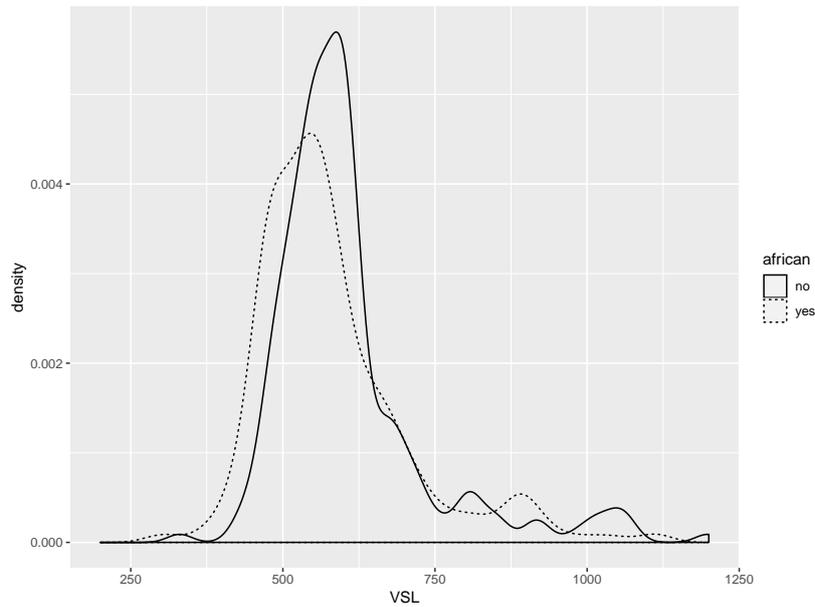
Figure 1: The value of a statistical life.

We can then compute the VSL for every individual and analyze their distribution, using quantiles and plotting in Figure 1 the empirical density of VSL for African and non-African travelers (as done in Leon and Miguel 2017, Table 4, p. 219 and Figure 5, p. 223).[11]

```
R> indpar$VSL <- with(indpar, risk / cost * 100)
R> quantile(indpar$VSL, c(0.025, 0.975))

     2.5%      97.5%
 432.4199 1054.3428

R> mean(indpar$VSL)

[1] 608.94
```

Note that computing the VSL as the ratio of two random parameters which can take zero value can lead to extremely high values if the individual parameter for cost is close to zero.[12]

```
R> max(indpar$cost)

[1] -0.002924437

R> max(indpar$VSL)
```

---

[11]Note that individual-specific parameters should be interpreted with caution, as they are consistent estimates of individual parameters only if the number of choice situations for every individual is large (see Train 2009, p. 266).

[12]See Daly, Hess, and Train (2012) for a discussion of the specifications of mixed logit models which assure finite moments of the distribution of willingness to pay.

```
[1] 3131.825
```

This is not the case here as (absolute) minimum value of `cost` is -0.003 which leads to a maximum value of VSL of 3131$.

```
R> library("ggplot2")
R> RT$id <- RT$id
R> indpar <- merge(unique(subset(as.data.frame(RT),
+    select = c("id", "african"))), indpar)
R> ggplot(indpar) + geom_density(aes(x = VSL, linetype = african)) +
+    scale_x_continuous(limits = c(200, 1200))
```

5.3. Optimization problems

Except for the simple multinomial logit model, the user can encounter computational problems while trying to estimate a model. This is especially true for some mixed logit models, in particular for those which introduce log-normally distributed random parameters and correlation. In this situation, the following approaches can be used:

- use the BHHH method instead of the default, which is BFGS (by setting the `method` argument to `"bhhh"`),

- provide user defined starting values by filling the `start` argument,

- set some parameters to constants, using the `constPar` argument and then update this fitted preliminary model by using the parameters obtained as starting values and removing the `constPar` argument,

- if the convergence is still not achieved, try another model. For example, for mixed logit models, switch from a log-normal to a triangular distribution or from correlated to uncorrelated random parameters.

6. Conclusions

Package **mlogit** estimates a large set of random utility models with a unified and user-friendly interface, together with functions and methods which compute and return useful results, like predicted probabilities, inclusive values, marginal effects, consumer surplus and individual parameters.

Some of the models that can be fitted using **mlogit** have not been presented in this article, namely:

- the rank-ordered logit model which is relevant in situations where individuals do not choose one among a set of mutually exclusive alternatives, but rank them,

- the overlapping nested logit model which is a nested logit model where some alternatives may belong to more than one nest,

- the paired combinatorial logit model which is a nested logit model where every possible combination of two alternatives is a nest,

- the multinomial probit model, obtained when the errors are assumed to follow a multi-variate normal distribution.

All these models are illustrated in several vignettes of the **mlogit** package.

References

Ben-Akiva M, Bolduc D, Bradley M (1993). "Estimation of Travel Choice Models with Randomly Distributed Values of Time." In *Innovations in Travel Behavior Analysis, Demand Forecasting, and Modeling Networks*, number 1413 in Transportation Research Record, pp. 88–97. Transportation Research Board.

Bhat CR (1995). "A Heteroscedastic Extreme Value Model of Intercity Travel Mode Choice." *Transportation Research Part B: Methodological*, **29**(6), 471–483. doi:10.1016/0191-2615(95)00015-6.

Croissant Y (2020a). **dfidx**: *Indexed Data Frames*. R package version 0.0-3, URL https://CRAN.R-project.org/package=dfidx.

Croissant Y (2020b). **mlogit**: *Multinomial Logit Models*. R package version 1.1-1, URL https://CRAN.R-project.org/package=mlogit.

Daly A (1987). "Estimating "Tree" Logit Models." *Transportation Research Part B: Methodological*, **21**(4), 251–267. doi:10.1016/0191-2615(87)90026-9.

Daly A, Hess S, Train K (2012). "Assuring Finite Moments for Willingness to Pay in Random Coefficient Models." *Transportation*, **39**(1), 19–31. doi:10.1007/s11116-011-9331-3.

Dumont J, Keller J (2019). **RSGHB**: *Functions for Hierarchical Bayesian Estimation: A Flexible Approach*. R package version 1.2.2, URL https://CRAN.R-project.org/package=RSGHB.

Forinash CV, Koppleman FS (1993). "Application and Interpretation of Nested Logit Models and Intercity Mode Choice." In *Innovations in Travel Behavior Analysis, Demand Forecasting, and Modeling Networks*, number 1413 in Transportation Research Record, pp. 98–106. Transportation Research Board.

Fowlie M (2010). "Emissions Trading, Electricity Restructuring, and Investment in Pollution Abatement." *American Economic Review*, **100**(3), 837–69. doi:10.1257/aer.100.3.837.

Fox J, Weisberg S (2019). *An R Companion to Applied Regression*. 3rd edition. Sage Publications, Thousand Oaks. URL http://socserv.socsci.mcmaster.ca/jfox/Books/Companion.

Hasan A, Wang Z, Mahani AS (2016). "Fast Estimation of Multinomial Logit Models: R Package **mnlogit**." *Journal of Statistical Software*, **75**(3), 1–24. doi:10.18637/jss.v075.i03.

Head K, Mayer T (2004). "Market Potential and the Location of Japanese Investment in the European Union." *The Review of Economics and Statistics*, **86**(4), 959–972. doi:10.1162/0034653043125257.

Heiss F (2002). "Structural Choice Analysis with Nested Logit Models." *The Stata Journal*, **2**(3), 227–252. doi:10.1177/1536867x0200200301.

Hensher DA, Greene WH (2002). "Specification and Estimation of the Nested Logit Model: Alternative Normalisations." *Transportation Research Part B: Methodological*, **36**(1), 1–17. doi:10.1016/s0191-2615(00)00035-7.

Imai K, van Dyk D (2017). **MNP**: *R Package for Fitting the Multinomial Probit Model*. R package version 3.1-0, URL https://CRAN.R-project.org/package=MNP.

Koppelman FS, Wen CH (1998). "Alternative Nested Logit Models: Structure, Properties and Estimation." *Transportation Research Part B: Methodological*, **32**(5), 289–298. doi:10.1016/s0191-2615(98)00003-4.

Koppelman FS, Wen CH (2000). "The Paired Combinatorial Logit Model: Properties, Estimation and Application." *Transportation Research Part B: Methodological*, **34**(2), 75–89. doi:10.1016/s0191-2615(99)00012-0.

Leifeld P (2013). "**texreg**: Conversion of Statistical Model Output in R to LaTeX and HTML Tables." *Journal of Statistical Software*, **55**(8), 1–24. doi:10.18637/jss.v055.i08.

Leon G, Miguel E (2017). "Risky Transportation Choices and the Value of a Statistical Life." *American Economic Journal: Applied Economics*, **9**(1), 202–228. doi:10.1257/app.20160140.

McFadden D (1974). "The Measurement of Urban Travel Demand." *Journal of Public Economics*, **3**(4), 303–328. doi:10.1016/0047-2727(74)90003-6.

McFadden D (1978). "Spatial Interaction Theory and Planning Models." In A Karlqvist (ed.), *Modeling the Choice of Residential Location*, pp. 75–96. North-Holland, Amsterdam.

Meijer E, Rouwendal J (2006). "Measuring Welfare Effects in Models with Random Coefficients." *Journal of Applied Econometrics*, **21**(2), 227–244. doi:10.1002/jae.841.

Rossi P (2019). **bayesm**: *Bayesian Inference for Marketing/Micro-Econometrics*. R package version 3.1-4, URL https://CRAN.R-project.org/package=bayesm.

Sarrias M, Daziano R (2017). "Multinomial Logit Models with Continuous and Discrete Individual Heterogeneity in R: The **gmnl** Package." *Journal of Statistical Software*, **79**(2), 1–46. doi:10.18637/jss.v079.i02.

Small KA, Rosen HS (1981). "Applied Welfare Economics with Discrete Choice Models." *Econometrica*, **49**(1), 105–130. doi:10.2307/1911129.

Swait J, Louviere J (1993). "The Role of the Scale Parameter in the Estimation and Use of Multinomial Logit Models." *Journal of Marketing Research*, **30**(3), 305–314. doi:10.2307/3172883.

Train KE (2009). *Discrete Choice Methods with Simulation.* 2nd edition. Cambridge University Press, Cambridge.

Zeileis A, Croissant Y (2010). "Extended Model Formulas in R: Multiple Parts and Multiple Responses." *Journal of Statistical Software*, **34**(1), 1–13. `doi:10.18637/jss.v034.i01`.

Zeileis A, Hothorn T (2002). "Diagnostic Checking in Regression Relationships." *R News*, **2**(3), 7–10. URL `https://CRAN.R-project.org/doc/Rnews/`.

Affiliation:

Yves Croissant
CEMOI
Université de la Réunion
15, Avenue René Cassin
BP 7151
F-97715 Saint-Denis Messag Cedex 9, Réunion
Telephone: +33/262/938446
E-mail: `yves.croissant@univ-reunion.fr`