Estimation of multinomial logit models in R : The mlogit Packages

Yves Croissant
Université de la Réunion

Abstract

mlogit is a package for R which enables the estimation of the multinomial logit models with individual 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.

An introductory example

The logit model is useful when one tries to explain discrete choices, *i.e.* choices of one among several mutually exclusive alternatives[1]. There are many useful applications of discrete choice modelling in different fields of applied econometrics, using individual data, which may be :

- *revealed preferences data* which means that the data are observed choices of individuals for, say, a transport mode (car, plane and train for example),
- *stated preferences data* ; in this case, individuals face a virtual situation of choice, for example the choice between three train tickets with different characteristics :
  - A : a train ticket which costs 10 euros, for a trip of 30 minutes and one change,
  - B : a train ticket which costs 20 euros, for a trip of 20 minutes and no change,
  - C : a train ticket which costs 22 euros, for a trip of 22 minutes and one change.

Suppose that, in a transport mode situation, we can define an index of satisfaction V_j for each alternative which depends linearly on cost (x) and time (z) :

$$\left\{ \begin{array}{rcl} V_1 &=& \alpha_1 + \beta x_1 + \gamma z_1 \\ V_2 &=& \alpha_2 + \beta x_2 + \gamma z_2 \\ V_3 &=& \alpha_3 + \beta x_3 + \gamma z_3 \end{array} \right.$$

[1]For an extensive presentation of the logit model, see Train (2003) and Louiviere, Hensher, and Swait (2000). The theoretical parts of this paper draw heavily on Kenneth Train's book.

In this case, the probability of choosing the alternative j is increasing with V_j. For sake of estimation, one has to transform the satisfaction index, which can take any real value so that it is restricted to the unit interval and can be interpreted as a probability. The multinomial logit model is obtained by applying such a transformation to the V_js. More specifically, we have :

$$\left\{\begin{array}{lcl} \mathrm{P}_1 & = & \frac{e^{V_1}}{e^{V_1}+e^{V_2}+e^{V_3}} \\ \mathrm{P}_2 & = & \frac{e^{V_2}}{e^{V_1}+e^{V_2}+e^{V_3}} \\ \mathrm{P}_3 & = & \frac{e^{V_3}}{e^{V_1}+e^{V_2}+e^{V_3}} \end{array}\right.$$

The two characteristics of probabilities are satisfied :

- $0 \leq \mathrm{P}_j \leq 1 \; \forall i = 1, 2, 3,$
- $\sum_{j=1}^{3} \mathrm{P}_j = 1$

Once fitted, a logit model is useful for predictions :

- enter new values for the explanatory variables,
- get
  - at an individual level the probabilities of choice,
  - at an aggregate level the market shares.

Consider, as an example, interurban trips between two towns (Lyon and Paris). Suppose that there are three modes (car, plane and train) and that the characteristics of the modes and the market shares are as follow :

	price	time	share
car	50	4	20%
plane	150	1	25%
train	80	2	55%

With a sample of travellers, one can estimate the coefficients of the logit model, *i.e.* the coefficients of time and price in the utility function.

The fitted model can then be used to predict the impact of some changes of the explanatory variables on the market shares, for example :

- the influence of train trips length on modal shares,
- the influence of the arrival of low cost companies.

To get the predictions, one just has to change the values of train time or plane price and compute the new probabilities, which can be interpreted at the aggregate level as predicted market shares.

1. Data management and model description

1.1. Data management

mlogit is loaded using :

```
library("mlogit")
#ra <- lapply(system("ls ~/Dropbox/Forge/mlogit/pkg/R/*R", intern = TRUE), source)
#library("Formula");library("statmod");library("zoo");library("MASS");library("lmtest")
```

It comes with several data sets that we'll use to illustrate the features of the library. Data sets used for multinomial logit estimation deals with some individuals, that make one or a sequential choice of one alternative among a set of several alternatives. The determinants of these choices are variables that can be alternative specific or purely individual specific. Such data have therefore a specific structure that can be characterised by three indexes :

- the alternative,
- the choice situation,
- the individual.

the last one being only relevant if we have repeated observations for the same individual.

Data sets can have two different shapes :

- a *wide* shape : in this case, there is one row for each choice situation,
- a *long* shape : in this case, there is one row for each alternative and, therefore, as many rows as there are alternatives for each choice situation.

This can be illustrated with three data sets.

- Fishing is a revealed preferences data sets that deals with the choice of a fishing mode,
- TravelMode (from the AER package) is also a revealed preferences data sets which presents the choice of individuals for a transport mode for inter-urban trips in Australia,

- Train is a stated preferences data sets for which individuals faces repeated virtual situations of choice for train tickets.

```
data("Fishing", package = "mlogit")
head(Fishing, 3)

##      mode price.beach price.pier price.boat price.charter
## 1 charter     157.930    157.930    157.930       182.930
## 2 charter      15.114     15.114     10.534        34.534
## 3    boat     161.874    161.874     24.334        59.334
##   catch.beach catch.pier catch.boat catch.charter   income
## 1      0.0678     0.0503     0.2601        0.5391 7083.332
## 2      0.1049     0.0451     0.1574        0.4671 1250.000
## 3      0.5333     0.4522     0.2413        1.0266 3750.000
```

There are four fishing modes (beach, pier, boat, charter), two alternative specific variables (price and catch) and one choice/individual specific variable (income)[2]. This "wide" format is suitable to store individual specific variables. Otherwise, it is cumbersome for alternative specific variables because there are as many columns for such variables that there are alternatives.

```
data("TravelMode", package="AER")
head(TravelMode)

##   individual  mode choice wait vcost travel gcost income size
## 1          1   air     no   69    59    100    70     35    1
## 2          1 train     no   34    31    372    71     35    1
## 3          1   bus     no   35    25    417    70     35    1
## 4          1   car    yes    0    10    180    30     35    1
## 5          2   air     no   64    58     68    68     30    2
## 6          2 train     no   44    31    354    84     30    2
```

There are four transport modes (air, train, bus and car)and most of the variable are alternative specific (wait, vcost, travel, gcost). The only individual specific variables are income and size. The advantage of this shape is that there are much fewer columns than in the wide format, the caveat being that values of income and size are repeated four times.

mlogit deals with both format. It provides a mlogit.data function that take as first argument a `data.frame` and returns a `data.frame` in "long" format with some information about the structure of the data.

For the Fishing data, we would use :

[2]Note that the distinction between choice situation and individual is not relevant here as these data are not panel data.

```
Fish <- mlogit.data(Fishing, shape="wide", varying=2:9, choice="mode")
```

The mandatory arguments are `choice`, which is the variable that indicates the choice made, the shape of the original `data.frame` and, if there are some alternative specific variables, `varying` which is a numeric vector that indicates which columns contains alternative specific variables. This argument is then passed to reshape that coerced the original `data.frame` in "long" format. Further arguments may be passed to reshape. For example, if the names of the variables are of the form var:alt, one can add sep = ':'.

```
head(Fish, 5)

##              mode   income     alt   price  catch chid
## 1.beach     FALSE 7083.332   beach 157.930 0.0678    1
## 1.boat      FALSE 7083.332    boat 157.930 0.2601    1
## 1.charter    TRUE 7083.332 charter 182.930 0.5391    1
## 1.pier      FALSE 7083.332    pier 157.930 0.0503    1
## 2.beach     FALSE 1250.000   beach  15.114 0.1049    2
```

The result is a `data.frame` in "long format" with one line for each alternative. The "choice" variable is now a logical variable and the individual specific variable (income) is repeated 4 times. An index attribute is added to the data, which contains the two relevant index : **chid** is the choice index and **alt** index. This attribute is a `data.frame` that can be extracted using the index function, which returns this `data.frame`.

```
head(index(Fish))

##             chid     alt
## 1.beach        1   beach
## 1.boat         1    boat
## 1.charter      1 charter
## 1.pier         1    pier
## 2.beach        2   beach
## 2.boat         2    boat
```

For data in "long" format like TravelMode, the `shape` (here equal to `long`) and the `choice` arguments are still mandatory.

The information about the structure of the data can be explicitly indicated or, in part, guessed by the mlogit.data function. Here, we have 210 choice situations which are indicated by a variable called individual. The information about choice situations can also be guessed from the fact that the data frame is balanced (every individual face 4 alternatives) and that the rows are ordered first by choice situations and then by alternative.

Concerning the alternative, there are indicated by the mode variable and they can also be guessed thanks to the ordering of the rows and the fact that the data frame is balanced.

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 :

```
TM <- mlogit.data(TravelMode,
choice = "choice", shape = "long", alt.levels = c("air", "train",
"bus", "car"))
```

It is also possible to provide an argument `alt.var` which indicates the name of the variable that contains the alternatives

```
TM <- mlogit.data(TravelMode ,choice = "choice", shape = "long",
                  alt.var = "mode")
```

The name of the variable that contains the information about the choice situations can be indicated using the `chid.var` argument :

```
TM <- mlogit.data(TravelMode, choice = "choice",
                  shape = "long", chid.var = "individual",
                  alt.levels = c("air", "train", "bus", "car"))
```

Both alternative and choice variable can be provided :

```
TM <- mlogit.data(TravelMode, choice = "choice", shape = "long",
                 chid.var = "individual", alt.var = "mode")
```

and dropped from the data frame using the `drop.index` argument :

```
TM <- mlogit.data(TravelMode, choice = "choice", shape = "long",
                 chid.var = "individual", alt.var = "mode", drop.index = TRUE)
head(TM)

##          choice wait vcost travel gcost income size
## 1.air     FALSE   69    59    100    70     35    1
## 1.train   FALSE   34    31    372    71     35    1
## 1.bus     FALSE   35    25    417    70     35    1
## 1.car      TRUE    0    10    180    30     35    1
## 2.air     FALSE   64    58     68    68     30    2
## 2.train   FALSE   44    31    354    84     30    2
```

The final example (Train) is in a “wide” format and contains panel data.

```
data("Train", package="mlogit")
head(Train, 3)

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

Each individual has responded to several (up to 16) scenario. To take this panel dimension into account, one has to add an argument `id` which contains the individual variable. The index attribute has now a supplementary column, the individual index.

```
Tr <- mlogit.data(Train, shape = 'wide', choice = "choice", id = "id",
                  varying = 4:11, sep = "_", alt.levels = c("A", "B"))
head(Tr, 3)

##     id choiceid choice alt price time change comfort chid
## 1.A  1        1   TRUE   A  2400  150      0       1    1
## 1.B  1        1  FALSE   B  4000  150      0       1    1
## 2.A  1        2   TRUE   A  2400  150      0       1    2

head(index(Tr), 3)

##     chid alt id
## 1.A    1   A  1
## 1.B    1   B  1
## 2.A    2   A  1
```

1.2. Model description

mlogit use the standard `formula`, `data` interface to describe the model to be estimated. However, standard `formula`s are not very practical for such models. More precisely, while working with multinomial logit models, one has to consider three kinds of variables :

- alternative specific variables x_{ij} with a generic coefficient β,
- individual specific variables z_i with an alternative specific coefficients γ_j,
- alternative specific variables w_{ij} with an alternative specific coefficient δ_j.

The satisfaction index for the alternative j is then :

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

Satisfaction being ordinal, only differences are relevant to modelize the choice for one alternative. This means that we'll be interested in the difference between the satisfaction index of two different alternatives j and k :

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

It is clear from the previous expression that coefficients for individual 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 may be identified. For example, with three alternatives 1, 2 and 3, the three coefficients $\gamma_1, \gamma_2, \gamma_3$ associated to an individual specific variable cannot be identified, but only two linear combinations of them. Therefore, one has to make a choice of normalization and the most simple one is just to set $\gamma_1 = 0$.

Coefficients for alternative specific variables may (or may not) be alternative specific. For example, transport time is alternative specific, but 10 mn in public transport may not have the same impact on utility than 10 mn in a car. In this case, alternative specific coefficients are relevant. Monetary time is also alternative specific, but in this case, one can consider than 1 euro is 1 euro whatever it is spent in car or in public transports[3]. In this case, a generic coefficient is relevant.

A model with only individual 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 follow, we'll use the name *multinomial logit model* for the model we've just described whatever the nature of the explanatory variables included in the model.

mlogit package provides objects of class `mFormula` which are extended model formulas and which are build upon `Formula` objects provided by the **Formula** package[4].

To illustrate the use of `mFormula` objects, let's use again the TravelMode data set. income and size (the size of the household) are individual specific variables. vcost (monetary cost) and travel (travel time) are alternative specific. We want to use a generic coefficient for the former and alternative specific coefficients for the latter. This is done using the mFormula function that build a three-parts formula :

```
f <- mFormula(choice ~ vcost | income + size | travel)
```

[3] At least if the monetary cost of using car is correctly calculated.

[4] See Zeileis and Croissant (2010) for a description of the **Formula** package.

By default, an intercept is added to the model, it can be removed by using `+0` or `-1` in the second part. Some parts may be omitted when there are no ambiguity. For example, the following couples of `formula`s are identical :

```
f2 <- mFormula(choice ~ vcost + travel | income + size)
f2 <- mFormula(choice ~ vcost + travel | income + size | 0)
```

```
f3 <- mFormula(choice ~ 0 | income | 0)
f3 <- mFormula(choice ~ 0 | income)
```

```
f4 <- mFormula(choice ~ vcost + travel)
f4 <- mFormula(choice ~ vcost + travel | 1)
f4 <- mFormula(choice ~ vcost + travel | 1 | 0)
```

Finally, we show below some `formula`s that describe models without intercepts (which is generally hardly relevant)

```
f5 <- mFormula(choice ~ vcost | 0 | travel)
f6 <- mFormula(choice ~ vcost | income + 0 | travel)
f6 <- mFormula(choice ~ vcost | income -1 | travel)
f7 <- mFormula(choice ~ 0 | income -1 | travel)
```

`model.matrix` and `model.frame` methods are provided for `mFormula` objects. The former is of particular interest, as illustrated in the following example :

```
f <- mFormula(choice ~ vcost | income  | travel)
head(model.matrix(f, TM))

##         train:(intercept) bus:(intercept) car:(intercept) vcost
## 1.air                   0               0               0    59
## 1.train                 1               0               0    31
## 1.bus                   0               1               0    25
## 1.car                   0               0               1    10
## 2.air                   0               0               0    58
## 2.train                 1               0               0    31
##         train:income bus:income car:income air:travel
## 1.air              0          0          0        100
## 1.train           35          0          0          0
## 1.bus              0         35          0          0
## 1.car              0          0         35          0
## 2.air              0          0          0         68
```

```
## 2.train           30          0          0          0
##          train:travel bus:travel car:travel
## 1.air               0          0          0
## 1.train           372          0          0
## 1.bus               0        417          0
## 1.car               0          0        180
## 2.air               0          0          0
## 2.train           354          0          0
```

The model matrix contains $J-1$ columns for every individual specific variables (income and the intercept), which means that the coefficient associated to the first alternative (air) is fixed to 0.

It contains only one column for vcost because we want a generic coefficient for this variable.

It contains J columns for travel, because it is an alternative specific variable for which we want an alternative specific coefficient.

2. Random utility model and the multinomial logit model

2.1. Random utility model

The individual must choose one alternative among J different and exclusive alternatives. A level of utility may be defined for each alternative and the individual is supposed to choose the alternative with the highest level of utility. Utility is supposed to be the sum of two components[5]:

- a systematic component, denoted V_j, which is a function of different observed variables x_j. For sake of simplicity, it will be supposed that this component is a linear combination of the observed explanatory variables : $V_j = \beta_j^\top x_j$,
- an unobserved component ϵ_j which, from the researcher point of view, can be represented as a random variable. This error term includes the impact of all the unobserved variables which have an impact on the utility of choosing a specific alternative.

It is very important to understand that the utility and therefore the choice is purely deterministic from the decision maker's point of view. It is random form the searcher's point of view, because some of the determinants of the utility are unobserved, which implies that the choice can only be analyzed in terms of probabilities.

We have, for each alternative, the following utility levels :

[5]when possible, we'll omit the individual index to simplify the notations.

$$\left\{\begin{array}{lclcl} U_1 & = & \beta_1^\top x_1 + \epsilon_1 & = & V_1 + \epsilon_1 \\ U_2 & = & \beta_2^\top x_2 + \epsilon_2 & = & V_2 + \epsilon_2 \\ & \vdots & & \vdots & \\ U_J & = & \beta_J^\top x_J + \epsilon_J & = & V_J + \epsilon_J \end{array}\right.$$

alternative l will be chosen if and only if $\forall \;\; j \neq l \;\; U_l > U_j$ which leads to the following $J-1$ conditions :

$$\left\{\begin{array}{lcl} U_l - U_1 & = & (V_l - V_1) + (\epsilon_l - \epsilon_1) > 0 \\ U_l - U_2 & = & (V_l - V_2) + (\epsilon_l - \epsilon_2) > 0 \\ & \vdots & \\ U_l - U_J & = & (V_l - V_J) + (\epsilon_l - \epsilon_J) > 0 \end{array}\right.$$

As ϵ_j are not observed, choices can only be modeled in terms of probabilities from the researcher point of view. The $J-1$ conditions can be rewritten in terms of upper bonds for the $J-1$ remaining error terms :

$$\left\{\begin{array}{lcl} \epsilon_1 & < & (V_l - V_1) + \epsilon_l \\ \epsilon_2 & < & (V_l - V_2) + \epsilon_l \\ & \vdots & \\ \epsilon_J & < & (V_l - V_J) + \epsilon_l \end{array}\right.$$

The general expression of the probability of choosing alternative l is then :

$$(\mathrm{P}_l \mid \epsilon_l) = \mathrm{P}(U_l > U_1, \ldots, U_l > U_J)$$

$$(\mathrm{P}_l \mid \epsilon_l) = F_{-l}(\epsilon_1 < (V_l - V_1) + \epsilon_l, \ldots, \epsilon_J < (V_l - V_J) + \epsilon_l) \tag{1}$$

where F_{-l} is the multivariate distribution of $J-1$ error terms (all the ϵ's except ϵ_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 :

$$\mathrm{P}_l = \int (\mathrm{P}_l \mid \epsilon_l) f_l(\epsilon_l) d\epsilon_l$$

$$\mathrm{P}_l = \int F_{-l}((V_l - V_1) + \epsilon_l, \ldots, (V_l - V_J) + \epsilon_l) f_l(\epsilon_l) d\epsilon_l \tag{2}$$

where f_l is the marginal density function of ϵ_l.

2.2. 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 relies on three hypothesis :

H1 : independence of errors

If the hypothesis of independence of errors is made, the univariate distribution of the errors can be used :

$$\left\{\begin{array}{rcl} \mathrm{P}(U_l > U_1) & = & F_1(V_l - V_1 + \epsilon_l) \\ \mathrm{P}(U_l > U_2) & = & F_2(V_l - V_2 + \epsilon_l) \\ & \vdots & \\ \mathrm{P}(U_l > U_J) & = & F_J(V_l - V_J + \epsilon_l) \end{array}\right.$$

where F_j is the cumulative density of ϵ_j.

The conditional (1) and unconditional (2) probabilities are then :

$$(\mathrm{P}_l \mid \epsilon_l) = \prod_{j \neq l} F_j(V_l - V_j + \epsilon_l) \tag{3}$$

$$\mathrm{P}_l = \int \prod_{j \neq l} F_j(V_l - V_j + \epsilon_l)\ f_l(\epsilon_l)\ d\epsilon_l \tag{4}$$

which means that the evaluation of only a one-dimensional integral is required to compute the probabilities.

H2 : Gumbel distribution

Each ϵ follows a Gumbel distribution :

$$f(z) = \frac{1}{\theta} e^{-\frac{z-\mu}{\theta}} e^{-e^{-\frac{z-\mu}{\theta}}}$$

where μ is the location parameter and θ the scale parameter.

$$P(z < t) = F(t) = \int_{-\infty}^{t} \frac{1}{\theta} e^{-\frac{z-\mu}{\theta}} e^{-e^{-\frac{z-\mu}{\theta}}} dz = e^{-e^{-\frac{t-\mu}{\theta}}}$$

The first two moments of the Gumbel distribution are $\mathrm{E}(z) = \mu + \theta\gamma$, where γ is the Euler-Mascheroni constant (0.577) and $\mathrm{V}(z) = \frac{\pi^2}{6}\theta^2$.

The mean of ϵ_js is not identified if V_j contains an intercept. We can then, without loss of generality suppose that $\mu_j = 0 \quad \forall j$. Moreover, the overall scale of utility is not identified. Therefore, only $J-1$ scale parameters may be identified, and a natural choice of normalisation is to impose that one of the θ_j is equal to 1.

H3 identically distributed errors

As the location parameter is not identified for any error term, this hypothesis is essentially an homoscedasticity hypothesis, which means that the scale parameter of Gumbel distribution is the same for all the alternatives. As one of them has been previously fixed to 1, we can therefore suppose that, without loss of generality, $\theta_j = 1\ \forall j \in 1 \ldots J$ in case of homoscedasticity.

In this case, the conditional (3) and unconditional (4) probabilities further simplify to :

$$(\mathrm{P}_l \mid \epsilon_l) = \prod_{j \neq l} F(V_l - V_j + \epsilon_l) \tag{5}$$

$$\mathrm{P}_l = \int \prod_{j \neq l} F(V_l - V_j + \epsilon_l)\ f(\epsilon_l)\ d\epsilon_l \tag{6}$$

with F and f respectively the cumulative and the density of the standard Gumbel distribution (*i.e.* with position and scale parameters equal to 0 and 1).

2.3. Computation of the probabilities

With these hypothesis on the distribution of the error terms, we can now show that the probabilities have very simple, closed forms, which correspond to the logit transformation of the deterministic part of the utility.

Let's start with the probability that the alternative l is better than one other alternative j. With hypothesis 2 and 3, it can be written :

$$P(\epsilon_j < V_l - V_j + \epsilon_l) = e^{-e^{-(V_l - V_j + \epsilon_l)}} \tag{7}$$

With hypothesis 1, the probability of choosing l is then simply the product of probabilities (7) for all the alternatives except l :

$$(P_l \mid \epsilon_l) = \prod_{j \neq l} e^{-e^{-(V_l - V_j + \epsilon_l)}} \tag{8}$$

The unconditional probability is the expected value of the previous expression with respect to ϵ_l.

$$P_l = \int_{-\infty}^{+\infty} (P_l \mid \epsilon_l)\, e^{-\epsilon_l} e^{-e^{-\epsilon_l}} d\epsilon_l = \int_{-\infty}^{+\infty} \left(\prod_{j \neq l} e^{-e^{-(V_l - V_j + \epsilon_l)}} \right) e^{-\epsilon_l} e^{-e^{-\epsilon_l}} d\epsilon_l \tag{9}$$

We first begin by writing the preceding expression for *all* alternatives, including the l alternative.

$$P_l = \int_{-\infty}^{+\infty} \left(\prod_{j} e^{-e^{-(V_l - V_j + \epsilon_l)}} \right) e^{-\epsilon_l} d\epsilon_l$$

$$P_l = \int_{-\infty}^{+\infty} e^{-\sum_j e^{-(V_l - V_j + \epsilon_l)}} e^{-\epsilon_l} d\epsilon_l = \int_{-\infty}^{+\infty} e^{-e^{-\epsilon_l} \sum_j e^{-(V_l - V_j)}} e^{-\epsilon_l} d\epsilon_l$$

We then use the following change of variable

$$t = e^{-\epsilon_l} \Rightarrow dt = -e^{-\epsilon_l} d\epsilon_l$$

The unconditional probability is therefore the following integral :

$$P_l = \int_0^{+\infty} e^{-t \sum_j e^{-(V_l - V_j)}} dt$$

which has a closed form :

$$P_l = \left[- \frac{e^{-t \sum_j e^{-(V_l - V_j)}}}{\sum_j e^{-(V_l - V_j)}} \right]_0^{+\infty} = \frac{1}{\sum_j e^{-(V_l - V_j)}}$$

and can be rewritten as the usual logit probability :

$$P_l = \frac{e^{V_l}}{\sum_j e^{V_j}} \tag{10}$$

2.4. IIA hypothesis

If we consider the probabilities of choice for two alternatives l and m, we have :

$$P_l = \frac{e^{V_l}}{\sum_j e^{V_j}}$$

$$P_m = \frac{e^{V_m}}{\sum_j e^{V_j}}$$

The ration of these two probabilities is :

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

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 hypothesis (for independence of irrelevant alternatives).

If we use again the introductory example of urban trips between Lyon and Paris :

	price	time	share
car	50	4	20%
plane	150	1	20%
train	80	2	60%

Suppose that, because of low cost companies arrival, the price of plane is now 100$. The market share of plane will increase (for example up to 60%). With a logit model, share for train / share for car is 3 before the price change, and will remain the same after the price change. Therefore, the new predicted probabilities for car and train are 10 and 30%.

The *IIA* hypothesis relies on the hypothesis of independence of the error terms. 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 if some important variables are unobserved.

To see that, suppose that the utilities for two alternatives are :

$$U_{i1} = \alpha_1 + \beta_1 z_i + \gamma x_{i1} + \epsilon_{i1}$$
$$U_{i2} = \alpha_2 + \beta_2 z_i + \gamma x_{i2} + \epsilon_{i2}$$

with ϵ_{i1} and ϵ_{i2} uncorrelated. In this case, the logit model can be safely used, as the hypothesis of independence of the errors is satisfied.

If z_i is unobserved, the estimated model is :

$$U_{i1} = \alpha_1 + \gamma x_{i1} + \eta_{i1}$$
$$U_{i2} = \alpha_2 + \gamma x_{i2} + \eta_{i2}$$
$$\eta_{i1} = \epsilon_{i1} + \beta_1 z_i$$
$$\eta_{i2} = \epsilon_{i2} + \beta_2 z_i$$

The error terms are now correlated because of the common influence of omitted variables.

2.5. Estimation

The coefficients of the multinomial logit model are estimated by full information maximum likelihood.

The likelihood function

Let's start with a very simple example. Suppose there are four individuals. For given parameters and explanatory variables, we can calculate the probabilities. The likelihood for the sample is the probability associated to the sample :

	choice	P_{i1}	P_{i2}	P_{i3}	l_i
1	1	0.5	0.2	0.3	0.5
2	3	0.2	0.4	0.4	0.4
3	2	0.6	0.1	0.3	0.1
4	2	0.3	0.6	0.1	0.6

With random sample the joint probability for the sample is simply the product of the probabilities associated with every observation.

$$\mathrm{L} = 0.5 \times 0.4 \times 0.1 \times 0.6$$

A compact expression of the probabilities that enter the likelihood function is obtained by denoting y_{ij} a dummy variable which is equal to 1 if individual i made choice j and 0 otherwise.

The probability of the choice made for one individual is then :

$$\mathrm{P}_i = \prod_j \mathrm{P}_{ij}^{y_{ij}}$$

Or in log :

$$\ln \mathrm{P}_i = \sum_j y_{ij} \ln \mathrm{P}_{ij}$$

which leads to the log-likelihood function :

$$\ln L = \sum_i \ln \mathrm{P}_i = \sum_i \sum_j y_{ij} \ln \mathrm{P}_{ij}$$

Properties of the maximum likelihood estimator

Under regularity conditions, the maximum likelihood estimator is consistent and has an asymptotic normal distribution. The variance of the estimator is :

$$\mathrm{V}(\hat{\theta}) = \left(\mathrm{E}\left(-\frac{\partial^2 \ln L}{\partial\theta\partial\theta^\top}(\theta)\right)\right)^{-1}$$

This expression can not be computed because it depends on the thrue values of the parameters. Three estimators have been proposed :

- $\hat{\mathrm{V}}_1(\hat{\theta}) = \left(\mathrm{E}\left(-\frac{\partial^2 \ln L}{\partial\theta\partial\theta^\top}(\hat{\theta})\right)\right)^{-1}$: this expression can be computed if the expected value is computable,
- $\hat{\mathrm{V}}_2(\hat{\theta}) = \left(-\frac{\partial^2 \ln L}{\partial\theta\partial\theta^\top}(\hat{\theta})\right)^{-1}$
- $\hat{\mathrm{V}}_3(\hat{\theta}) = \sum_{i=1}^n \left(\frac{\partial \ln l_i}{\partial\theta}(\hat{\theta})\right)\left(\frac{\partial \ln l_i}{\partial\theta}(\hat{\theta})\right)^\top$: this expression is called the BHHH expression and doesn't require the computation of the hessian.

Numerical optimization

We seek to calculate the maximum of a function $f(x)$. This first order condition for a maximum is $f'(x_o) = 0$, but in general, there is no explicit solution for x_o, which then must be numerically approximated. In this case, the following algorithm can be used :

1. Start with a value x called x_t,
2. Approximate the function around x_t using a second order Taylor series : $l(x) = f(x_t) + (x - x_t)g(x_t) + 0.5(x - x_t)^2 h(x_t)$ where g and h are the first two derivatives of f,
3. find the maximum of $l(x)$. The first order condition is : $\frac{\partial l(x)}{\partial x} = g(x_t) + (x - x_t)h(x_t) = 0$. The solution is : $x_t - \frac{g(x_t)}{h(x_t)}$
4. call this value x_{t+1} and iterate until you get as close as required to the maximum.

This algorithm is illustrated on figure 1.

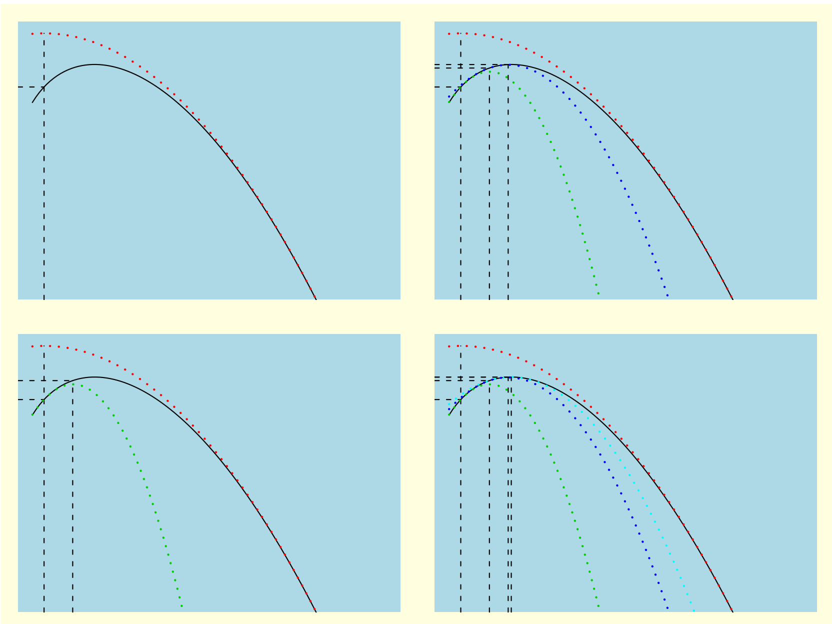

Figure 1: Numerical optimization

Consider now a function of several variables $f(x)$. The vector of first derivatives (called the gradient) is denoted g and the matrix of second derivatives (called the hessian) is denoted H. The second order approximation is :

$$l(x) = f(x_t) + (x - x_t)g(x_t) + 0.5(x - x_t)^\top H(x_t)(x - x_t)$$

The vector of first derivatives is :

$$\frac{\partial l(x)}{\partial x} = g(x_t) + H(x_t)(x - x_t)$$

$$x = x_t - H(x_t)^{-1}g(x_t)$$

Two kinds of routines are currently used for maximum likelihood estimation. The first one can be called "Newton-like" methods. In this case, at each iteration, an estimation of the hessian is calculated, whether using the second derivatives of the function (Newton-Ralphson method) or using the outer product of the gradient (BHHH). This approach is very powerful if the function is well-behaved, but it may perform poorly otherwise and fail after a few iterations.

The second one, called BFGS, updates at each iteration the estimation of the hessian. It is often more robust and may performs well in cases where the first one doesn't work.

Two optimization functions are included in core R: nlm which use the Newton-Ralphson method and optim which use BFGS (among other methods). Recently, the **maxLik** package (Toomet and Henningsen 2010) provides a unified approach. With a unique interface, all the previously described methods are available.

The behavior of maxLik can be controlled by the user using in the estimation function arguments like `print.level` (from 0-silent to 2-verbal), `iterlim` (the maximum number of iterations), `methods` (the method used, one of `nr`, `bhhh` or `bfgs`) that are passed to maxLik.

Gradient and Hessian for the logit model

For the multinomial logit model, the gradient and the hessian have very simple expressions.

$$\frac{\partial \ln P_{ij}}{\partial \beta} = x_{ij} - \sum_l P_{il} x_{il}$$

$$\frac{\partial \ln L}{\partial \beta} = \sum_i \sum_j (y_{ij} - P_{ij}) x_{ij}$$

$$\frac{\partial^2 \ln L}{\partial \beta \partial \beta'} = \sum_i \sum_j P_{ij} \left(x_{ij} - \sum_l P_{il} x_{il}\right) \left(x_{ij} - \sum_l P_{il} x_{il}\right)^\top$$

Moreover, the log-likelihood function is globally concave, which mean that there is a unique optimum which is the global maximum. In this case, the Newton-Ralphson method is very efficient and the convergence is achieved after just a few iterations.

2.6. Interpretation

In a linear model, the coefficients can be directly considered as marginal effects of the explanatory variables on the explained variable. This is not the case for the multinomial models. 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 explanatory variables, which can be be individual-specific (z_i) or alternative specific (x_{ij}) :

$$\frac{\partial P_{ij}}{\partial z_i} = P_{ij}\left(\beta_j - \sum_l P_{il}\beta_l\right)$$

$$\frac{\partial P_{ij}}{\partial x_{ij}} = \gamma P_{ij}(1 - P_{ij})$$

$$\frac{\partial P_{ij}}{\partial x_{il}} = -\gamma P_{ij} P_{il}$$

- For an alternative-specific variable, the sign of the coefficient is directly interpretable. 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 4 in order to have an upper bound of the marginal effect.
- For an individual specific variable, the sign of the coefficient is not necessarily the sign of the coefficient. Actually, the sign of the marginal effect is given by $(\beta_j - \sum_l P_{il}\beta_l)$, which is positive if the coefficient for the j alternative 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.

Marginal rates of substitution

Coefficients are marginal utilities, which are not interpretable because utility is ordinal. However, ratios of coefficients are marginal rates of substitution, which are interpretable. For example, if the observable part of utility is : $V = \beta_o + \beta_1 x_1 + \beta x_2 + \beta x_3$, join variations of x_1 and x_2 which ensure the same level of utility are such that : $dV = \beta_1 dx_1 + \beta_2 dx_2 = 0$ so that :

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

For example, if x_2 is transport cost (in euros), x_1 transport time (in hours), $\beta_1 = 1.5$ and $\beta_2 = 0.2$, $\frac{\beta_1}{\beta_2} = 30$ is the marginal rate of substitution of time in terms of euros and the value of 30 means that to reduce the travel time of one hour, the individual is willing to pay at most 30 euros more.

Consumer's surplus

Consumer's surplus has a very simple expression with multinomial logit models. It 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 alternative chosen. The expected utility, from the searcher's point of view is then :

$$\mathrm{E}(\max_j U_j)$$

where the expectation is taken on the values of all the error terms. If the marginal utility of income (α) is known and constant, the expected surplus is simply $\mathrm{E}(\max_j U_j)/\alpha$.

This expected surplus is a very simple expression in the context of the logit model, which is called the "log-sum". We'll demonstrate this fact in the context of two alternatives.

With two alternatives, the values of ϵ_1 and ϵ_2 can be depicted in a plane. This plane contains all the possible combinations of (ϵ_1, ϵ_2). Some of them leads to the choice of alternative 1 and the other to the choice of alternative 2. More precisely, alternative 1 is chosen if $\epsilon_2 \leq V_1 - V_2 + \epsilon_1$ and alternative 2 is chosen if $\epsilon_1 \leq V_2 - V_1 + \epsilon_2$. The first expression is the equation of a straight line in the plan which delimits the choice for the two alternatives.

We can then write the expected utility as the sum of two terms E_1 and E_2, with :

$$E_1 = \int_{\epsilon_1=-\infty}^{\infty} \int_{-\infty}^{V_1-V_2+\epsilon_1} (V_1 + \epsilon_1) f(\epsilon_1) f(\epsilon_2) d\epsilon_1 d\epsilon_2$$

and

$$E_2 = \int_{\epsilon_2=-\infty}^{\infty} \int_{-\infty}^{V_2-V_1+\epsilon_1} (V_2 + \epsilon_2) f(\epsilon_1) f(\epsilon_2) d\epsilon_1 d\epsilon_2$$

with $f(z) = \exp(-e^{-z})$ the density of the Gumbell distribution.

We'll derive the expression for E_1, by symmetry we'll guess the expression for E_2 and we'll then obtain the expected utility by summing E_1 and E_2.

$$E_1 = \int_{\epsilon_1=-\infty}^{\infty} (V_1 + \epsilon_1) \left(\int_{-\infty}^{V_1-V_2+\epsilon_1} f(\epsilon_2) d\epsilon_2 \right) f(\epsilon_1) d\epsilon_1$$

The expression in brackets is the cumulative density of ϵ_2. We then have :

$$E_1 = \int_{\epsilon_1=-\infty}^{\infty} (V_1 + \epsilon_1) e^{-e^{-(V_1-V_2)-\epsilon_1}} f(\epsilon_1) d\epsilon_1$$

$$E_1 = \int_{\epsilon_1=-\infty}^{\infty} (V_1 + \epsilon_1) e^{-\epsilon_1} e^{-ae^{-\epsilon_1}} d\epsilon_1$$

with $a = 1 + e^{-(V_1-V_2)} = \frac{e^{V_1}+e^{V_2}}{e^{V_1}} = \frac{1}{P_1}$
Let defines $z \mid e^{-z} = ae^{-\epsilon_1} \Leftrightarrow z = \epsilon_1 - \ln a$
We then have :

$$E_1 = \int_{\epsilon_1=-\infty}^{\infty} (V_1 + z + \ln a)/ae^{-z} e^{-e^{-z}} dz$$

$$E_1 = (V_1 + \ln a)/a + \mu/a$$

where μ is the expected value of a random variable which follows a standard Gumbell distribution, *i.e.* the Euler-Mascheroni constant.

$$E_1 = \frac{\ln(e^{V_1} + e^{V_2}) + \mu}{(e^{V_1} + e^{V_2})/e^{V_1}} = \frac{e^{V_1} \ln(e^{V_1} + e^{V_2}) + e^{V_1}\mu}{e^{V_1} + e^{V_2}}$$

By symmetry,

$$E_2 = \frac{e^{V_2} \ln(e^{V_1} + e^{V_2}) + e^{V_2}\mu}{e^{V_1} + e^{V_2}}$$

And then :

$$\mathrm{E}(U) = E_1 + E_2 = \ln(e^{V_1} + e^{V_2}) + \mu$$

More generally, in presence of J alternatives, we have :

$$\mathrm{E}(U) = \ln \sum_{j=1}^{J} e^{V_j} + \mu$$

and the expected surplus is, with α the constant marginal utility of income :

$$\mathrm{E}(U) = \frac{\ln \sum_{j=1}^{J} e^{V_j} + \mu}{\alpha}$$

2.7. Application

Train contains data about a stated preference survey in Netherlands. Users are asked to choose between two train trips characterized by four attributes :

- price : the price in cents of guilders,
- time : travel time in minutes,
- change : the number of changes,
- comfort : the class of comfort, 0, 1 or 2, 0 being the most comfortable class.

```
data("Train", package="mlogit")
Tr <- mlogit.data(Train, shape = 'wide', choice = "choice",
                  varying = 4:11, sep = "_", alt.levels = c("A", "B"), id = "id")
```

We first convert price and time in more meaningful unities, hours and euros (1 guilder is 2.20371 euros) :

```
Tr$price <- Tr$price / 100 * 2.20371
Tr$time <- Tr$time / 60
```

We then estimate the model : both alternatives being virtual train trips, it is relevant to use only generic coefficients and to remove the intercept :

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

##
## Call:
## mlogit(formula = choice ~ price + time + change + comfort | -1,
##     data = Tr, method = "nr", print.level = 0)
##
## Frequencies of alternatives:
##       A       B
## 0.50324 0.49676
##
## nr method
## 5 iterations, 0h:0m:0s
## g'(-H)^-1g = 0.00014
## successive function values within tolerance limits
##
## Coefficients :
##           Estimate Std. Error  z-value  Pr(>|z|)
## price   -0.0673580  0.0033933 -19.8506 < 2.2e-16 ***
## time    -1.7205514  0.1603517 -10.7299 < 2.2e-16 ***
## change  -0.3263409  0.0594892  -5.4857 4.118e-08 ***
```

```
## comfort -0.9457256  0.0649455 -14.5618 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -1724.2
```

All the coefficients are highly significant and have the predicted negative sign (remind than an increase in the variable comfort implies using a less comfortable class). The coefficients are not directly interpretable, but dividing them by the price coefficient, we get monetary values :

```
coef(ml.Train)[-1]/coef(ml.Train)[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 access a more comfortable class.

The second example use the Fishing data. It illustrates the multi-part formula interface to describe the model, and the fact that it is not necessary to transform the data set using mlogit.data before the estimation, *i.e.* instead of using :

```
Fish <- mlogit.data(Fishing, shape="wide", varying=2:9, choice="mode")
ml.Fish <- mlogit(mode~price | income | catch, Fish)
```

it is possible to use mlogit with the original `data.frame` and the relevant arguments that will be internally passed to mlogit.data :

```
ml.Fish <- mlogit(mode~price | income | catch, Fishing, shape = "wide", varying = 2:9)
summary(ml.Fish)

##
## Call:
## mlogit(formula = mode ~ price | income | catch, data = Fishing,
##     shape = "wide", varying = 2:9, method = "nr", print.level = 0)
##
## Frequencies of alternatives:
##   beach    boat charter    pier
## 0.11337 0.35364 0.38240 0.15059
##
## nr method
## 7 iterations, 0h:0m:0s
```

```
## g'(-H)^-1g = 2.54E-05
## successive function values within tolerance limits
##
## Coefficients :
##                       Estimate  Std. Error  z-value  Pr(>|z|)
## boat:(intercept)     8.4184e-01  2.9996e-01   2.8065 0.0050080
## charter:(intercept)  2.1549e+00  2.9746e-01   7.2443 4.348e-13
## pier:(intercept)     1.0430e+00  2.9535e-01   3.5315 0.0004132
## price               -2.5281e-02  1.7551e-03 -14.4046 < 2.2e-16
## boat:income          5.5428e-05  5.2130e-05   1.0633 0.2876612
## charter:income      -7.2337e-05  5.2557e-05  -1.3764 0.1687088
## pier:income         -1.3550e-04  5.1172e-05  -2.6480 0.0080977
## beach:catch          3.1177e+00  7.1305e-01   4.3724 1.229e-05
## boat:catch           2.5425e+00  5.2274e-01   4.8638 1.152e-06
## charter:catch        7.5949e-01  1.5420e-01   4.9254 8.417e-07
## pier:catch           2.8512e+00  7.7464e-01   3.6807 0.0002326
##
## boat:(intercept)    **
## charter:(intercept) ***
## pier:(intercept)    ***
## price               ***
## boat:income
## charter:income
## pier:income         **
## beach:catch         ***
## boat:catch          ***
## charter:catch       ***
## pier:catch          ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -1199.1
## McFadden R^2:  0.19936
## Likelihood ratio test : chisq = 597.16 (p.value = < 2.22e-16)
```

Several methods can be used to extract some results from the estimated model. `fitted` returns the predicted probabilities for the outcome or for all the alternatives if outcome = FALSE.

```
head(fitted(ml.Fish))

##         1         2         3         4         5         6
## 0.3114002 0.4537956 0.4567631 0.3701758 0.4763721 0.4216448
```

```
head(fitted(ml.Fish, outcome=FALSE))

##        beach      boat   charter       pier
## 1 0.09299769 0.5011740 0.3114002 0.09442817
## 2 0.09151070 0.2749292 0.4537956 0.17976449
## 3 0.01410358 0.4567631 0.5125571 0.01657625
## 4 0.17065868 0.1947959 0.2643696 0.37017585
## 5 0.02858215 0.4763721 0.4543225 0.04072324
## 6 0.01029791 0.5572463 0.4216448 0.01081103
```

Finally, two further arguments can be usefully used while using mlogit

- `reflevel` indicates which alternative is the "reference" alternative, *i.e.* the one for which the coefficients are 0,
- `altsubset` indicates a subset 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 observations which correspond to choices for unselected alternatives are removed :

```
mlogit(mode~price | income | catch, Fish, reflevel='charter',
       alt.subset=c('beach', 'pier', 'charter'))

##
## Call:
## mlogit(formula = mode ~ price | income | catch, data = Fish,     alt.subset = c("beach",
##
## Coefficients:
## beach:(intercept)   pier:(intercept)              price
##       -1.9952e+00        -9.4859e-01        -2.8343e-02
##      beach:income        pier:income      charter:catch
##        2.7184e-05        -1.0359e-04         1.1719e+00
##       beach:catch         pier:catch
##        3.2090e+00         2.8101e+00
```

2.8. The rank-ordered logit model

Sometimes, in stated-preference surveys, the respondents are asked to give the full rank of their preference for all the alternative, and not only the prefered alternative. The relevant model for this kind of data is the rank-ordered logit model, which can be estimated as a standard multinomial logit model if the data is reshaped correctly[6]

[6]see for example Beggs, Cardell, and Hausman (1981), Chapman and Staelin (1982) and Hausman and Ruud (1987).

The ranking can be decomposed in a series of choices of the best alternative within a decreasing set of available alternatives. For example, with 4 alternatives, the probability that the ranking would be 3-1-4-2 can be writen as follow :

- alternative 3 is in the first position, the probability is then $\frac{e^{\beta^\top x_3}}{e^{\beta^\top x_1}+e^{\beta^\top x_2}+e^{\beta^\top x_3}+e^{\beta^\top x_4}}$,
- alternative 1 is in second position, the relevant probability is the logit probability that 1 is the chosen alternative in the set of alternatives (1-2-4) : $\frac{e^{\beta^\top x_1}}{e^{\beta^\top x_1}+e^{\beta^\top x_2}+e^{\beta^\top x_4}}$,
- alternative 4 is in third position, the relevant probability is the logit probability that 4 is the chosen alternative in the set of alternatives (2-4) : $\frac{e^{\beta^\top x_4}}{e^{\beta^\top x_2}+e^{\beta^\top x_4}}$,
- the probability of the full ranking is then simply the product of these 3 probabilities.

This model can therefore simply be fitted as a multinomial logit model ; the ranking for one individual amoung J alternatives is writen as $J-1$ choices among $J, J-1, \ldots, 2$ alternatives.

The estimation of the rank-ordered logit model is illustrated using the Game data set Fok, Paap, and van Dijk (2010). Respondents are asked to rank 6 gaming platforms. The covariates are a dummy own which indicates whether a specific platform is curently owned, the age of the respondent (age) and the number of hours spent on gaming per week (hours). The data set is available in wide (game) and long (game2) format. In wide format, the consists on J columns which indicate the ranking of each alternative.

```
data("Game", package = "mlogit")
data("Game2", package = "mlogit")
head(Game,2)

##   ch.Xbox ch.PlayStation ch.PSPortable ch.GameCube ch.GameBoy
## 1       2              1             3           5          6
## 2       4              2             3           5          6
##   ch.PC own.Xbox own.PlayStation own.PSPortable own.GameCube
## 1     4        0               1              0            0
## 2     1        0               1              0            0
##   own.GameBoy own.PC age hours
## 1           0      1  33  2.00
## 2           0      1  19  3.25

head(Game2, 7)

##   age hours    platform ch own chid
## 1  33  2.00     GameBoy  6   0    1
```

```
## 2  33  2.00    GameCube  5   0    1
## 3  33  2.00          PC  4   1    1
## 4  33  2.00 PlayStation  1   1    1
## 5  33  2.00  PSPortable  3   0    1
## 6  33  2.00        Xbox  2   0    1
## 7  19  3.25     GameBoy  6   0    2

nrow(Game)

## [1] 91

nrow(Game2)

## [1] 546
```

Note that Game contains 91 rows (there are 91 individuals) and that Game2 contains 546 rows (91 individuals *times* 6 alternatives)

To use mlogit.data, the `ranked` should `TRUE` :

```
G <- mlogit.data(Game2, shape="long", choice="ch", alt.var='platform', ranked=TRUE)
G <- mlogit.data(Game, shape="wide", choice="ch", varying=1:12, ranked=TRUE)
head(G)

##                 age hours         alt own chid    ch
## 1.GameBoy        33     2     GameBoy   0    1 FALSE
## 1.GameCube       33     2    GameCube   0    1 FALSE
## 1.PC             33     2          PC   1    1 FALSE
## 1.PSPortable     33     2  PSPortable   0    1 FALSE
## 1.PlayStation    33     2 PlayStation   1    1  TRUE
## 1.Xbox           33     2        Xbox   0    1 FALSE

nrow(G)

## [1] 1820
```

Note that the choice variable is now a logical variable and that the number of row is now 1820 (91 individuals $\times(6+5+4+3+2)$ alternatives).

Using PC as the reference level, we can then reproduce the results of the original reference :

```
summary(mlogit(ch~own|hours+age, G, reflevel="PC"))

##
## Call:
## mlogit(formula = ch ~ own | hours + age, data = G, reflevel = "PC",
##     method = "nr", print.level = 0)
##
## Frequencies of alternatives:
##          PC     GameBoy    GameCube  PSPortable PlayStation
##     0.17363     0.13846     0.13407     0.17363     0.18462
##        Xbox
##     0.19560
##
## nr method
## 5 iterations, 0h:0m:0s
## g'(-H)^-1g = 6.74E-06
## successive function values within tolerance limits
##
## Coefficients :
##                          Estimate Std. Error z-value  Pr(>|z|)
## GameBoy:(intercept)      1.570379   1.600251  0.9813 0.3264288
## GameCube:(intercept)     1.404095   1.603483  0.8757 0.3812185
## PSPortable:(intercept)   2.583563   1.620778  1.5940 0.1109302
## PlayStation:(intercept)  2.278506   1.606986  1.4179 0.1562270
## Xbox:(intercept)         2.733774   1.536098  1.7797 0.0751272
## own                      0.963367   0.190396  5.0598 4.197e-07
## GameBoy:hours           -0.235611   0.052130 -4.5197 6.193e-06
## GameCube:hours          -0.187070   0.051021 -3.6665 0.0002459
## PSPortable:hours        -0.233688   0.049412 -4.7294 2.252e-06
## PlayStation:hours       -0.129196   0.044682 -2.8915 0.0038345
## Xbox:hours              -0.173006   0.045698 -3.7858 0.0001532
## GameBoy:age             -0.073587   0.078630 -0.9359 0.3493442
## GameCube:age            -0.067574   0.077631 -0.8704 0.3840547
## PSPortable:age          -0.088669   0.079421 -1.1164 0.2642304
## PlayStation:age         -0.067006   0.079365 -0.8443 0.3985154
## Xbox:age                -0.066659   0.075205 -0.8864 0.3754227
##
## GameBoy:(intercept)
## GameCube:(intercept)
## PSPortable:(intercept)
## PlayStation:(intercept)
## Xbox:(intercept)        .
## own                     ***
```

```
## GameBoy:hours          ***
## GameCube:hours         ***
## PSPortable:hours       ***
## PlayStation:hours      **
## Xbox:hours             ***
## GameBoy:age
## GameCube:age
## PSPortable:age
## PlayStation:age
## Xbox:age
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -516.55
## McFadden R^2:  0.36299
## Likelihood ratio test : chisq = 588.7 (p.value = < 2.22e-16)
```

3. Relaxing the iid hypothesis

With hypothesis 1 and 3, the error terms are *iid* (identically and independently distributed), *i.e.* not correlated and homoscedastic. Extensions of the basic multinomial logit model have been proposed by relaxing one of these two hypothesis while maintaining the second hypothesis of Gumbell distribution.

3.1. The heteroskedastic logit model

The heteroskedastic logit model was proposed by Bhat (1995).

The probability that $U_l > U_j$ is :

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

which implies the following conditional and unconditional probabilities

$$(P_l \mid \epsilon_l) = \prod_{j \neq l} e^{-e^{-\frac{(V_l - V_j + \epsilon_l)}{\theta_j}}} \tag{11}$$

$$P_l = \int_{-\infty}^{+\infty} \prod_{j \neq l} \left(e^{-e^{-\frac{(V_l - V_j + \epsilon_l)}{\theta_j}}} \right) \frac{1}{\theta_l} e^{-\frac{\epsilon_l}{\theta_l}} e^{-e^{-\frac{\epsilon_l}{\theta_l}}} d\epsilon_l \tag{12}$$

We then apply the following change of variable :

$$u = e^{-\frac{\epsilon_l}{\theta_l}} \Rightarrow du = -\frac{1}{\theta_l} e^{-\frac{\epsilon_l}{\theta_l}} d\epsilon_l$$

The unconditional probability (12) can then be rewritten :

$$P_l = \int_0^{+\infty} \prod_{j \neq l} \left(e^{-e^{-\frac{V_l - V_j - \theta_l \ln u}{\theta_j}}} \right) e^{-u} du = \int_0^{+\infty} \left(e^{-\sum_{j \neq l} e^{-\frac{V_l - V_j - \theta_l \ln u}{\theta_j}}} \right) e^{-u} du$$

There is no closed form for this integral but it can be written the following way :

$$P_l = \int_0^{+\infty} G_l e^{-u} du$$

with

$$G_l = e^{-A_l} \quad A_l = \sum_{j \neq l} \alpha_j \quad \alpha_j = e^{-\frac{V_l - V_j - \theta_l \ln u}{\theta_j}}$$

This one-dimensional integral can be efficiently computed using a Gauss quadrature method, and more precisely the Gauss-Laguerre quadrature method :

$$\int_0^{+\infty} f(u) e^{-u} du = \sum_t f(u_t) w_t$$

where u_t and w_t are respectively the nodes and the weights.

$$P_l = \sum_t G_l(u_t) w_t$$

$$\frac{\partial G_l}{\partial \beta_k} = \sum_{j \neq l} \frac{\alpha_j}{\theta_j} (x_{lk} - x_{jk}) G_l$$

$$\frac{\partial G_l}{\partial \theta_l} = - \ln u \sum_{j \neq l} \frac{\alpha_j}{\theta_j} G_l$$

$$\frac{\partial G_l}{\partial \theta_j} = \ln \alpha_j \frac{\alpha_j}{\theta_j} G_l$$

To illustrate the estimation of the heteroscedastic logit model, we use the data used by (Bhat 1995). This data set is called ModeCanada.

```
data("ModeCanada", package = "mlogit")
```

As done in the article, we first restrict the sample to the user who don't choose the bus and choose a mode among the four modes available (train, air, bus and car).

```
busUsers <- with(ModeCanada, case[choice == 1 & alt == 'bus'])
Bhat <- subset(ModeCanada, !case %in% busUsers & alt != 'bus' & noalt == 4)
Bhat$alt <- Bhat$alt[drop = TRUE]
Bhat <- mlogit.data(Bhat, shape='long', chid.var = 'case',
                    alt.var = 'alt', choice='choice',
                    drop.index=TRUE)
```

This restricts the sample to 2769 users.

```
ml.MC <- mlogit(choice ~ freq + cost + ivt + ovt | urban + income, Bhat, reflevel = 'car')
hl.MC <- mlogit(choice ~ freq + cost + ivt + ovt | urban + income, Bhat, reflevel = 'car', h
summary(hl.MC)

##
## Call:
## mlogit(formula = choice ~ freq + cost + ivt + ovt | urban + income,
##     data = Bhat, reflevel = "car", heterosc = TRUE)
##
## Frequencies of alternatives:
##     car   train     air
## 0.45757 0.16721 0.37523
##
## bfgs method
## 10 iterations, 0h:0m:2s
## g'(-H)^-1g = 2.89E-07
## gradient close to zero
##
## Coefficients :
##                      Estimate Std. Error z-value  Pr(>|z|)
## train:(intercept)   0.6783934  0.3327626  2.0387   0.04148 *
## air:(intercept)     0.6567544  0.4681631  1.4028   0.16067
## freq                0.0639247  0.0049168 13.0014 < 2.2e-16 ***
## cost               -0.0269615  0.0042831 -6.2948 3.078e-10 ***
## ivt                -0.0096808  0.0010539 -9.1859 < 2.2e-16 ***
## ovt                -0.0321655  0.0035930 -8.9523 < 2.2e-16 ***
## train:urban         0.7971316  0.1207392  6.6021 4.054e-11 ***
## air:urban           0.4454726  0.0821609  5.4220 5.895e-08 ***
```

```
## train:income      -0.0125979  0.0039942 -3.1541   0.00161 **
## air:income         0.0188600  0.0032159  5.8646 4.503e-09 ***
## sp.train           1.2371829  0.1104610 11.2002 < 2.2e-16 ***
## sp.air             0.5403239  0.1118353  4.8314 1.356e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -1838.1
## McFadden R^2:  0.35211
## Likelihood ratio test : chisq = 1998 (p.value = < 2.22e-16)
```

The results obtained by Bhat (1995) can't be exactly reproduced because he uses some weights that are not available in the data set. However, we obtain very close values for the two estimated scale parameters for the train sp.train and for the air mode sp.air.

The second example uses the TravelMode data set and reproduces the first column of table 23.28 page 855 of Greene (2008).

```
data("TravelMode",package="AER")
TravelMode <- mlogit.data(TravelMode,choice="choice",shape="long",
                          alt.var="mode",chid.var="individual")
TravelMode$avinc <- with(TravelMode,(mode=='air')*income)
ml.TM <- mlogit(choice ~ wait + gcost + avinc, TravelMode,
                reflevel = "car")
hl.TM <- mlogit(choice ~ wait + gcost + avinc, TravelMode,
             reflevel = "car", heterosc = TRUE)
summary(hl.TM)

##
## Call:
## mlogit(formula = choice ~ wait + gcost + avinc, data = TravelMode,
##     reflevel = "car", heterosc = TRUE)
##
## Frequencies of alternatives:
##     car     air   train     bus
## 0.28095 0.27619 0.30000 0.14286
##
## bfgs method
## 43 iterations, 0h:0m:1s
## g'(-H)^-1g = 3.77E-07
## gradient close to zero
##
## Coefficients :
##                     Estimate Std. Error z-value Pr(>|z|)
```

```
## air:(intercept)    7.832450  10.950706  0.7152   0.4745
## train:(intercept)  7.171867   9.135295  0.7851   0.4324
## bus:(intercept)    6.865775   8.829608  0.7776   0.4368
## wait              -0.196843   0.288274 -0.6828   0.4947
## gcost             -0.051562   0.069444 -0.7425   0.4578
## avinc              0.040253   0.060680  0.6634   0.5071
## sp.air             4.024020   5.977821  0.6732   0.5008
## sp.train           3.854208   6.220456  0.6196   0.5355
## sp.bus             1.648749   2.826916  0.5832   0.5597
##
## Log-Likelihood: -195.66
## McFadden R^2:  0.31047
## Likelihood ratio test : chisq = 176.2 (p.value = < 2.22e-16)
```

Note that the ranking of the scale parameters differs from the previous example. In particular, the error of the air utility has the largest variance as it has the smallest one in the previous example.

The standard deviations print at the end of table 23.28 are obtained by multiplying the scale parameters by $\pi/\sqrt{6}$:

```
c(coef(hl.TM)[7:9], sp.car = 1)*pi/sqrt(6)

##   sp.air sp.train   sp.bus   sp.car
## 5.161007 4.943214 2.114603 1.282550
```

Note that the standard deviations of the estimated scale parameters are very high, which means that they are poorly identified.

3.2. 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.

We suppose that the alternatives can be put into M different nests. This implies the following multivariate distribution for the error terms.

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

The marginal distributions of the ϵs are still univariate extreme value, but there is now some correlation within nests. $1-\lambda_m$ is a measure of the correlation, *i.e.* $\lambda_m = 1$ implies no correlation. It can then be shown that the probability of choosing alternative j that belongs to the 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 compatible with the random utility maximisation hypothesis if all the nest elasticities are in the $0-1$ interval.

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

$$V_j = Z_j + W_l$$

We can then rewrite the probabilities as follow :

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

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

$$\left(\sum_{k\in B_l} e^{(Z_k+W_l)/\lambda_l}\right)^{\lambda_l} = \left(e^{W_l/\lambda_l}\sum_{k\in B_l} e^{Z_k/\lambda_l}\right)^{\lambda_l} = e^{W_l+\lambda_l I_l}$$

with $I_l = \ln\sum_{k\in B_l} e^{Z_k/\lambda_l}$ which is often called the inclusive value or the inclusive utility.

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

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 with the previous expression is that the deterministic parts of the utility for each alternative is not divided by the nest elasticity :

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

The differences between the two versions have been discussed in Koppelman and Wen (1998), Heiss (2002) and Hensher and Greene (2002).

The gradient is, for the first version of the model and denoting $N_m = \sum_{k\in B_m} e^{V_k/\lambda_m}$:

$$\left\{\begin{array}{rcl}
\frac{\partial \ln P_j}{\partial \beta} & = & \frac{x_j}{\lambda_l} + \frac{\lambda_l - 1}{\lambda_l}\frac{1}{N_l}\sum_{k\in B_l} e^{V_k/\lambda_l}x_k - \frac{1}{\sum_m N_m^{\lambda_m}}\sum_m N_m^{\lambda_m - 1}\sum_{k\in B_m} e^{V_k/\lambda_m}x_k \\
\frac{\partial \ln P_j}{\partial \lambda_l} & = & -\frac{V_j}{\lambda_l^2} + \ln N_l - \frac{\lambda_l - 1}{\lambda_l^2}\frac{1}{N_l}\sum_{k\in B_l} V_k e^{V_k/\lambda_l} \\
 & & -\frac{N_l^{\lambda_l}}{\sum_m N_m^{\lambda_m}}\left(\ln N_l - \frac{1}{\lambda_l N_l}\sum_{k\in B_l} V_k e^{V_k/\lambda_l}\right) \\
\frac{\partial \ln P_j}{\partial \lambda_m} & = & -\frac{N_m^{\lambda_m}}{\sum_m N_m^{\lambda_m}}\left(\ln N_m - \frac{1}{\lambda_m N_m}\sum_{k\in B_m} V_k e^{V_k/\lambda_m}\right)
\end{array}\right.$$

Denoting $P_{j|l} = \frac{e^{V_j/\lambda_l}}{N_l}$ the conditional probability of choosing alternative j if nest l is chosen, $P_l = \frac{N_l^{\lambda_l}}{\sum_m N_m^{\lambda_m}}$ the probability of choosing nest l, $\bar{x}_l = \sum_{k\in B_l} P_{k|l}x_k$ the weight average value of x in nest l, $\bar{x} = \sum_{m=1}^M P_m \bar{x}_m$ the weight average of x for all the nests and $\bar{V}_l = \sum_{k\in B_l} P_{k|l}V_k$

$$\left\{\begin{array}{rcl}
\frac{\partial \ln P_j}{\partial \beta} & = & \frac{1}{\lambda_l}\left[x_j - (1-\lambda_l)\bar{x}_l\right] - \bar{x} \\
\frac{\partial \ln P_j}{\partial \lambda_l} & = & -\frac{1}{\lambda_l^2}\left[V_j - \lambda_l^2 \ln N_l - (1-\lambda_l)\bar{V}_l\right] - \frac{P_l}{\lambda_l^2}\left[\lambda_l^2 \ln N_l - \lambda_l \bar{V}_l\right] \\
\frac{\partial \ln P_j}{\partial \lambda_m} & = & \frac{P_m}{\lambda_m}\left[\bar{V}_m - \lambda_m \ln N_m\right]
\end{array}\right.$$

$$\left\{\begin{array}{rcl}
\frac{\partial \ln P_j}{\partial \beta} & = & \frac{x_j - \{(1-\lambda_l)\bar{x}_l + \lambda_l \bar{x}\}}{\lambda_l} \\
\frac{\partial \ln P_j}{\partial \lambda_l} & = & -\frac{V_j - \left\{\lambda_l(1-P_l)\lambda_l \ln N_l + (1-\lambda_l(1-P_l))\bar{V}_l\right\}}{\lambda_l^2} \\
\frac{\partial \ln P_j}{\partial \lambda_m} & = & \frac{P_m}{\lambda_m}\left[\bar{V}_m - \lambda_m \ln N_m\right]
\end{array}\right.$$

For the unscaled version, the gradient is :

$$\left\{\begin{array}{rcl}
\frac{\partial \ln P_j}{\partial \beta} & = & x_j - (1-\lambda_l)\bar{x}_l - \sum_m \lambda_m P_m \bar{x}_m \\
\frac{\partial \ln P_j}{\partial \lambda_l} & = & (1-P_l)\ln N_l \\
\frac{\partial \ln P_j}{\partial \lambda_m} & = & -P_m \ln N_m
\end{array}\right.$$

Until now, we have supposed that every alternative belongs to one and only one nest. If some alternatives belong to several nests, we get an overlapping nests model. In this case, the notations should be slightly modified :

$$P_j = \frac{\sum_{l|j\in B_l} e^{V_j/\lambda_l} N_l^{\lambda_l-1}}{\sum_m N_m^{\lambda_m}}$$

$$P_j = \sum_{l|j\in B_l} \frac{e^{V_j/\lambda_l}}{N_l} \frac{N_l^{\lambda_l}}{\sum_m N_m^{\lambda_m}} = \sum_{l|j\in B_l} P_{j|l} P_l$$

$$\left\{\begin{array}{rcl}
\frac{\partial \ln P_j}{\partial \beta} & = & \sum_{l|j\in B_l} \frac{P_{j|l}P_l}{P_j} \frac{x_j - \{(1-\lambda_l)\bar{x}_l + \lambda_l \bar{x}\}}{\lambda_l} \\
\frac{\partial \ln P_j}{\partial \lambda_l} & = & -\frac{P_{j|l}P_l}{P_j} \frac{V_j - \left\{\lambda_l(1-P_j/P_{j|l})\lambda_l \ln N_l + (1-\lambda_l(1-P_j/P_{j|l}))\bar{V}_l\right\}}{\lambda_l^2} \\
\frac{\partial \ln P_j}{\partial \lambda_m} & = & \frac{P_m}{\lambda_m}\left[\bar{V}_m - \lambda_m \ln N_m\right]
\end{array}\right.$$

For the unscaled version of the model, the gradient is :

$$\left\{\begin{array}{rcl}
\frac{\partial \ln P_j}{\partial \beta} & = & \sum_{l|j\in B_l} \frac{P_{j|l}P_l}{P_j}(x_j - (1-\lambda_l)\bar{x}_l) - \sum_m \lambda_m P_m \bar{x}_m \\
\frac{\partial \ln P_j}{\partial \lambda_l} & = & P_l \left(\frac{P_{jl}}{P_j} - 1\right) \ln N_l \\
\frac{\partial \ln P_j}{\partial \lambda_m} & = & -P_m \ln N_m
\end{array}\right.$$

We illustrate the estimation of the unscaled nested logit model with an example used in (Greene 2008). The dataset, called TravelMode has already been used. Four transport modes are available and two nests are considered :

- the *ground* nest with bus, train and car modes,
- the *fly* nest with the air modes.

Note that the second nest is a "degenerate" nest, which means that it contains only one alternative. In this case, the nest elasticity is difficult to interpret, as it is related to the degree of correlation of the alternatives within the nests and that there is only one alternative in this nest. This parameter can only be identified in a very special case : the use of the unscaled version of the nested logit model with generic variable. This is exactly the situation considered by (Greene 2008) and presented in the table 21.11 p. 730.

```
data("TravelMode",package="AER")
TravelMode <- mlogit.data(TravelMode,choice="choice",shape="long",
                          alt.var="mode",chid.var="individual")
TravelMode$avinc <- with(TravelMode,(mode=='air')*income)
nl.TM <- mlogit(choice ~ wait + gcost + avinc, TravelMode, reflevel = "car",
                nests = list(fly = "air", ground = c("train", "bus", "car")),
                unscaled=TRUE)
summary(nl.TM)
```

```
##
## Call:
## mlogit(formula = choice ~ wait + gcost + avinc, data = TravelMode,
##     reflevel = "car", nests = list(fly = "air", ground = c("train",
##         "bus", "car")), unscaled = TRUE)
##
## Frequencies of alternatives:
##     car     air   train     bus
## 0.28095 0.27619 0.30000 0.14286
##
## bfgs method
## 17 iterations, 0h:0m:0s
## g'(-H)^-1g = 1.02E-07
## gradient close to zero
##
## Coefficients :
##                    Estimate Std. Error z-value  Pr(>|z|)
## air:(intercept)    6.042373   1.331325  4.5386 5.662e-06 ***
## train:(intercept)  5.064620   0.676010  7.4919 6.795e-14 ***
## bus:(intercept)    4.096325   0.628870  6.5138 7.328e-11 ***
## wait              -0.112618   0.011826 -9.5232 < 2.2e-16 ***
## gcost             -0.031588   0.007434 -4.2491 2.147e-05 ***
## avinc              0.026162   0.019842  1.3185   0.18732
## iv:fly             0.586009   0.113056  5.1833 2.180e-07 ***
## iv:ground          0.388962   0.157904  2.4633   0.01377 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -193.66
## McFadden R^2:  0.31753
## Likelihood ratio test : chisq = 180.21 (p.value = < 2.22e-16)
```

The second example deals with a choice of a heating mode. The data set is called HC. There are seven alternatives, four of them provide also cooling : gaz central with cooling `gcc`, electric central with cooling `ecc`, electric room with cooling `erc` and heat pump with cooling `hpc` ; the other three provide only heating, these are electric central `ec`, electric room `er` and gaz central `gc`.

```
data("HC", package = "mlogit")
HC <- mlogit.data(HC, varying = c(2:8, 10:16), choice = "depvar", shape = "wide")
head(HC)

##       depvar  icca occa income alt   ich   och chid
```

```
## 1.ec   FALSE 27.28 2.95     20  ec 24.50 4.09    1
## 1.ecc  FALSE 27.28 2.95     20 ecc  7.86 4.09    1
## 1.er   FALSE 27.28 2.95     20  er  7.37 3.85    1
## 1.erc   TRUE 27.28 2.95     20 erc  8.79 3.85    1
## 1.gc   FALSE 27.28 2.95     20  gc 24.08 2.26    1
## 1.gcc  FALSE 27.28 2.95     20 gcc  9.70 2.26    1
```

`icca` and `occa` are the investment and the operating cost of the cooling part of the system. This is only relevant for the cooling modes and therefore we have to set the value to 0 for non-cooling modes.

```
cooling.modes <- HC$alt %in% c("gcc", "ecc", "erc","hpc")
HC$icca[!cooling.modes] <- HC$occa[!cooling.modes] <- 0
```

We now estimate a nested logit model with two nests : the cooling/non-cooling systems :

```
ml.HC <- mlogit(depvar~occa+icca+och+ich, HC)
nl.HC <- mlogit(depvar~occa+icca+och+ich, HC,
                nests = list(cooling = c('ecc', 'erc', 'gcc', 'hpc'),
                  noncool = c('ec', 'gc', 'er')))
summary(nl.HC)

##
## Call:
## mlogit(formula = depvar ~ occa + icca + och + ich, data = HC,
##     nests = list(cooling = c("ecc", "erc", "gcc", "hpc"), noncool = c("ec",
##         "gc", "er")))
##
## Frequencies of alternatives:
##    ec   ecc    er   erc    gc   gcc   hpc
## 0.004 0.016 0.032 0.004 0.096 0.744 0.104
##
## bfgs method
## 18 iterations, 0h:0m:0s
## g'(-H)^-1g = 2.24E-07
## gradient close to zero
##
## Coefficients :
##                    Estimate Std. Error z-value Pr(>|z|)
## ecc:(intercept)  2.171367   3.401923  0.6383  0.52329
## er:(intercept)  -2.455199   1.071462 -2.2914  0.02194 *
## erc:(intercept)  1.756250   3.547708  0.4950  0.62057
```

```
## gc:(intercept)  -0.208090   0.469091 -0.4436  0.65733
## gcc:(intercept)  2.234177   3.383645  0.6603  0.50907
## hpc:(intercept)  1.272654   3.618232  0.3517  0.72504
## occa            -0.966387   0.708161 -1.3646  0.17237
## icca            -0.051249   0.081461 -0.6291  0.52927
## och             -0.868681   0.445484 -1.9500  0.05118 .
## ich             -0.205005   0.090851 -2.2565  0.02404 *
## iv:cooling       0.333827   0.172073  1.9400  0.05238 .
## iv:noncool       0.328934   0.212062  1.5511  0.12087
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -188.03
## McFadden R^2:  0.16508
## Likelihood ratio test : chisq = 74.354 (p.value = 5.2125e-14)
```

The two nest elasticities are about 0.3, which implies a correlation of 0.7, which is quite high. The two nest elasticities are very close to each other, and it is possible to enforce the equality by updating the model with the argument `un.nest.el` set to `TRUE`.

```
nl.HC.u <- update(nl.HC, un.nest.el = TRUE)
```

3.3. The general extreme value model

Derivation of the general extreme value model

McFadden (1978) developed a general model that suppose that the join distribution of the error terms follow a a multivariate extreme value distribution. Let G be a function with J arguments $y_j \geq 0$. G has the following characteristics :

i) it is non negative $G(y_1, \ldots, y_J) \geq 0 \; \forall j$,

ii) it is homogeneous of degree 1 in all its arguments $G(\lambda y_1, \ldots \lambda y_J) = \lambda G(y_1, \ldots, y_J)$,

iii) for all its argument, $\lim_{y_j \rightarrow +\infty} = G(y_1, \ldots y_J) = +\infty$,

iv) for distinct arguments, $\frac{\partial^k G}{\partial y_i, \ldots, y_j}$ is non-negative if k is odd and non-positive if k is even.

Assume now that the joint cumulative distribution of the error terms can be written :

$$F(\epsilon_1, \epsilon_2, \ldots, \epsilon_J) = \exp\left(-G\left(e^{-\epsilon_1}, e^{-\epsilon_2}, \ldots, e^{-\epsilon_J}\right)\right)$$

We first show that this is a multivariate extreme value distribution. This implies :

1. if F is a joint cumulative distribution of probability, for any $\lim_{\epsilon_j \Rightarrow -\infty} F(\epsilon_1 \dots \epsilon_J) = 0$,
2. if F is a joint cumulative distribution of probability, $\lim_{\epsilon_1, \dots \epsilon_J \rightarrow +\infty} F(\epsilon_1 \dots \epsilon_J) = 1$,
3. all the cross-derivates of any order of F should be non-negative,
4. if F is a multivariate extreme value distribution, the marginal distribution of any ϵ_j, which is $\lim_{\epsilon_k \rightarrow +\infty \forall k \neq j} F(\epsilon_1 \dots \epsilon_J)$ should be an extreme value distribution.

For point 1, if $\epsilon_j \rightarrow -\infty$, $y_j \rightarrow +\infty$, $G \rightarrow +\infty$ and then $F \rightarrow 0$.
For point 2, if $(\epsilon_1, \dots, \epsilon_J) \rightarrow +\infty$, $G \rightarrow 0$ and then $F \rightarrow 1$.
For point 3, let denote[7] :

$$Q_k = Q_{k-1} G_k - \frac{\partial Q_{k-1}}{\partial y_k} \text{ and } Q_1 = G_1$$

Q_k is a sum of signed terms that are products of cross derivatives of G of various order. If each term of Q_{k-1} are non-negative, so is $Q_{k-1}G_k$ (from iv, the first derivatives are non-negative. Moreover "each term in $\frac{\partial Q_{k-1}}{\partial y_k}$ is non positive, since one of the derivatives within each term has increased in order, changing from even to odd or vice-versa, with a hypothesized change in sign (hypothesis iv). Hence each term in Q_k is non negative and, by induction, Q_k is non-negative for $k = 1, 2, \dots J$.
Suppose that the $k-1$-order cross-derivative of F can be written :

$$\frac{\partial^{k-1} F}{\partial \epsilon_1 \dots \partial \epsilon_{k-1}} = e^{-\epsilon_1} \dots e^{-\epsilon_k} Q_{k-1} F$$

Then , the k-order derivative is :

$$\frac{\partial^{k} F}{\partial \epsilon_1 \dots \partial \epsilon_{k}} = e^{-\epsilon_1} \dots e^{-\epsilon_k} Q_{k} F$$

$Q_1 = G_1$ is non-negative, so are $Q_2, Q_3, \dots Q_k$ and therefore all the cross-derivatives of any order are non-negatives.
To demonstrate the fourth point, we compute the marginal cumulative distribution of ϵ_l which is :

$$F(\epsilon_l) = \lim_{\epsilon_j \rightarrow +\infty \forall j \neq l} F(\epsilon_1, \dots, \epsilon_l, \dots \epsilon_J) = \exp\left(-G\left(0, \dots, e^{-\epsilon_l}, \dots, 0\right)\right)$$

with G being homogeneous of degree one, we have :

$$G\left(0, \dots, e^{-\epsilon_l}, \dots, 0\right) = a_l e^{-\epsilon_l}$$

[7] cited from McFadden (1978).

with $a_l = G(0, \ldots, 1, \ldots, 0)$. The marginal distribution of ϵ_l is then :

$$F(\epsilon_l) = \exp\left(-a_l e^{-\epsilon_l}\right)$$

which is an uni-variate extreme value distribution.

We note compute the probabilities of choosing an alternative :

We denote G_l the derivative of G respective to the l^{th} argument. The derivative of F respective to the ϵ_l is then :

$$F_l(\epsilon_1, \epsilon_2, \ldots, \epsilon_J) = e^{-\epsilon_l} G_l\left(e^{-\epsilon_1}, e^{-\epsilon_2}, \ldots, e^{-\epsilon_J}\right) \exp\left(-G\left(e^{-\epsilon_1}, e^{-\epsilon_2}, \ldots, e^{-\epsilon_J}\right)\right)$$

which is the density of ϵ_l for given values of the other $J-1$ error terms.

The probability of choosing alternative l is the probability that $U_l > U_j \; \forall j \neq l$ which is equivalent to $\epsilon_j < V_l - V_j + \epsilon_l$.

This probability is then :

$$\begin{array}{rcl}
P_l & = & \int_{-\infty}^{+\infty} F_l(V_l - V_1 + \epsilon_l, V_l - V_2 + \epsilon_l, \ldots, V_l - V_J + \epsilon_l) d\epsilon_l \\
& = & \int_{-\infty}^{+\infty} e^{-\epsilon_l} G_l\left(e^{-V_l+V_1-\epsilon_l}, e^{-V_l+V_2-\epsilon_l}, \ldots, e^{-V_l+V_J-\epsilon_l}\right) \\
& \times & \exp\left(-G\left(e^{-V_l+V_1-\epsilon_l}, e^{-V_l+V_2-\epsilon_l}, \ldots, e^{-V_l+V_J-\epsilon_l}\right)\right) d\epsilon_l
\end{array}$$

G being homogeneous of degree one, one can write :

$$G\left(e^{-V_l+V_1-\epsilon_l}, e^{-V_l+V_2-\epsilon_l}, \ldots, e^{-V_l+V_J-\epsilon_l}\right) = e^{-V_l} e^{-\epsilon_l} \times G\left(e^{V_1}, e^{V_2}, \ldots, e^{V_J}\right)$$

Homogeneity of degree one implies homogeneity of degree 0 of the first derivative :

$$G_l\left(e^{-V_l+V_1-\epsilon_l}, e^{-V_l+V_2-\epsilon_l}, \ldots, e^{-V_l-V_J-\epsilon_l}\right) = G_l\left(e^{V_1}, e^{V_2}, \ldots, e^{V_J}\right)$$

The probability of choosing alternative i is then :

$$P_l = \int_{-\infty}^{+\infty} e^{-\epsilon_l} G_l\left(e^{V_1}, e^{V_2}, \ldots, e^{V_J}\right) \exp\left(-e^{-\epsilon_l} e^{-V_l} G\left(e^{V_1}, e^{V_2}, \ldots, e^{V_J}\right)\right) d\epsilon_l$$

$$P_l = G_l \int_{-\infty}^{+\infty} e^{-\epsilon_l} \exp\left(-e^{-\epsilon_l} e^{-V_l} G\right) d\epsilon_l$$

$$P_l = G_l \frac{1}{e^{-V_l} G} \left[\exp\left(-e^{-\epsilon_l} e^{-V_l} G\right)\right]_{-\infty}^{+\infty} = \frac{G_l}{e^{-V_l} G}$$

Finally, the probability of choosing alternative i can be written :

$$P_l = \frac{e^{V_l} G_l\left(e^{V_1}, e^{V_2}, \ldots, e^{V_J}\right)}{G\left(e^{V_1}, e^{V_2}, \ldots, e^{V_J}\right)}$$

Among this vast family of models, several authors have proposed some nested logit models with overlapping nests Koppelman and Wen (2000) and Wen and Koppelman (2001).

Paired combinatorial logit model

Koppelman and Wen (2000) proposed the *paired combinatorial logit model*, which is a nested logit model with nests composed by every combination of two alternatives. This model is obtained by using the following G function :

$$G(y_1, y_2, \dots, y_n) = \sum_{k=1}^{J-1} \sum_{l=k+1}^{J} \left(y_k^{1/\lambda_{kl}} + y_l^{1/\lambda_{kl}} \right)^{\lambda_{kl}}$$

The *pcl* model is consistent with random utility maximisation if $0 < \lambda_{kl} \leq 1$ and the multinomial logit results if $\lambda_{kl} = 1 \, \forall (k, l)$. The resulting probabilities are :

$$P_l = \frac{\sum_{k \neq l} e^{V_l/\lambda_{lk}} \left(e^{V_k/\lambda_{lk}} + e^{V_l/\lambda_{lk}} \right)^{\lambda_{lk}-1}}{\sum_{k=1}^{J-1} \sum_{l=k+1}^{J} \left(e^{V_k/\lambda_{lk}} + e^{V_l/\lambda_{lk}} \right)^{\lambda_{lk}}}$$

which can be expressed as a sum of $J-1$ product of a conditional probability of choosing the alternative and the marginal probability of choosing the nest :

$$P_l = \sum_{k \neq l} P_{l|lk} P_{lk}$$

with :

$$P_{l|lk} = \frac{e^{V_l/\lambda_{lk}}}{e^{V_k/\lambda_{lk}} + e^{V_l/\lambda_{lk}}}$$

$$P_{lk} = \frac{\left(e^{V_k/\lambda_{lk}} + e^{V_l/\lambda_{lk}} \right)^{\lambda_{lk}}}{\sum_{k=1}^{J-1} \sum_{l=k+1}^{J} \left(e^{V_k/\lambda_{lk}} + e^{V_l/\lambda_{lk}} \right)^{\lambda_{lk}}}$$

We reproduce the example used by Koppelman and Wen (2000) on the same subset of the ModeCanada than the one used by Bhat (1995). Three modes are considered and there are therefore three nests. The elasticity of the train-air nest is set to one. To estimate this model, one has to set the `nests` to `pcl`. All the nests of two alternatives are then automatically created. The restriction on the nest elasticity for the train-air nest is performed by using the `constPar` argument.

```
pcl <- mlogit(choice~freq+cost+ivt+ovt, Bhat, reflevel='car',
              nests='pcl', constPar=c('iv.train.air'))
summary(pcl)
```

```
##
## Call:
## mlogit(formula = choice ~ freq + cost + ivt + ovt, data = Bhat,
##     reflevel = "car", nests = "pcl", constPar = c("iv.train.air"))
##
## Frequencies of alternatives:
##     car   train     air
## 0.45757 0.16721 0.37523
##
## bfgs method
## 16 iterations, 0h:0m:0s
## g'(-H)^-1g = 2.08E-07
## gradient close to zero
##
## Coefficients :
##                      Estimate  Std. Error  z-value  Pr(>|z|)
## train:(intercept)  1.30439316  0.16544227   7.8843 3.109e-15 ***
## air:(intercept)    1.99012922  0.35570613   5.5949 2.208e-08 ***
## freq               0.06537827  0.00435688  15.0057 < 2.2e-16 ***
## cost              -0.02448565  0.00316570  -7.7347 1.044e-14 ***
## ivt               -0.00761538  0.00067374 -11.3032 < 2.2e-16 ***
## ovt               -0.03223993  0.00237097 -13.5978 < 2.2e-16 ***
## iv.car.train       0.42129039  0.08613435   4.8911 1.003e-06 ***
## iv.car.air         0.27123320  0.09061319   2.9933   0.00276 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -1903
## McFadden R^2:  0.32927
## Likelihood ratio test : chisq = 1868.3 (p.value = < 2.22e-16)
```

The generalized nested logit model

Wen and Koppelman (2001) proposed the *generalized nested logit model*. This model is obtained by using the following G function :

$$G(y_1, y_2, \ldots, y_n) = \sum_m \left(\sum_{j \in B_m} (\alpha_{jm} y_j)^{1/\lambda_m} \right)^{\lambda_m}$$

with α_{jm} the allocation parameter which indicates which part of alternative j is assigned to nest m, with the condition $\sum_m \alpha_{jm} = 1 \; \forall j$ and λ_m the logsum parameter for nets m, with $0 < \lambda_m \leq 1$.

The resulting probabilities are :

$$P_j = \frac{\sum_m \left[\left(\alpha_{jm} e^{V_j}\right)^{1/\lambda_m} \left(\sum_{k \in N_m} \left(\alpha_{km} e^{V_k}\right)^{1/\lambda_m} \right)^{\lambda_m - 1} \right]}{\sum_m \left(\sum_{k \in B_m} (\alpha_{km} e^{V_k})^{1/\lambda_m} \right)^{\lambda_m}}$$

which can be expressed as a sum of products of a conditional probability of choosing the alternative and the marginal probability of choosing the nest :

$$P_j = \sum_m P_{j|m} P_m$$

with :

$$P_{j|m} = \frac{\left(\alpha_{jm} e^{V_j}\right)^{1/\lambda_m}}{\sum_{k \in B_m} (\alpha_{km} e^{V_k})^{1/\lambda_m}}$$

$$P_m = \frac{\left(\sum_{k \in N_m} \left(\alpha_{km} e^{V_k}\right)^{1/\lambda_m} \right)^{\lambda_m}}{\sum_m \left(\sum_{k \in B_m} (\alpha_{km} e^{V_k})^{1/\lambda_m} \right)^{\lambda_m}}$$

4. The random parameters (or mixed) logit 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.

4.1. The probabilities

For the standard logit model, the probabilities are :

$$P_{il} = \frac{e^{\beta' x_{il}}}{\sum_j e^{\beta' x_{ij}}}$$

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

$$P_{il} = \frac{e^{\beta_i' x_{il}}}{\sum_j e^{\beta_i' x_{ij}}}$$

Two strategies of estimation can then be considered :

- estimate the coefficients for each individual in the sample,

- consider the coefficients as random variables.

The first approach is of limited interest, because it would require numerous observations for each individual and because we are not interested on the value of the coefficients for a given individual. The second approach leads to the mixed logit model.

The probability that individual i will choose alternative l is :

$$P_{il} \mid \beta_i = \frac{e^{\beta_i' x_{il}}}{\sum_j e^{\beta_i' x_{ij}}}$$

This is the probability for individual i conditional on the vector of individual-specific coefficients β_i. To get the unconditional probability, we have to compute the average of these conditional probabilities for all the values of β_i.

Suppose that $V_{il} = \alpha + \beta_i x_{il}$, *i.e.* there is only one individual-specific coefficient and that the density of β_i is $f(\beta, \theta)$, θ being the vector of the parameters of the distribution of β. The unconditional probability is then :

$$P_{il} = \mathrm{E}(P_{il} \mid \beta_i) = \int_\beta (P_{il} \mid \beta) f(\beta, \theta) d\beta$$

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 :

$$P_{il} = \mathrm{E}(P_{il} \mid \beta_i) = \int_{\beta_1} \int_{\beta_2} \dots \int_{\beta_K} (P_{il} \mid \beta) f(\beta, \theta) d\beta_1 d\beta_2 \dots d\beta_K$$

This is a K-dimensional integral which cannot easily estimated by quadrature methods. In these kind of situations, the only practical method is 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.

4.2. Panel data

It is often the case, especially with stated preference survey, that we have repeated observations for the same individuals. This panel dimension can be taken into account in the mixed logit model. More specifically, we'll compute one probability for each individual and this is this probability that is included in the log-likelihood function. For a given vector of coefficients β_i, the probability that alternative l is chosen for the kth observation of the individual i is :

$$P_{ikl} = \frac{e^{\beta_i x_{ikl}}}{\sum_j e^{\beta_i x_{ikj}}}$$

The probability for the chosen probability for the kth observation for the individual i is :

$$P_{ik} = \prod_l P_{ikl}{}^{y_{ikl}}$$

Finally, the joint probability for the K observations of individual i is :

$$P_i = \prod_k \prod_l P_{ikl}{}^{y_{ikl}}$$

4.3. Simulations

The probabilities for the random parameter logit are integrals with no closed form. Moreover, the degree of integration is the number of random parameters. In practice, these models are estimated using simulation techniques, *i.e.* the expected value is replaced by an arithmetic mean. More precisely, the computation is done using the following steps :

- make an initial hypothesis about the distribution of the random parameters
- draw R numbers on this distribution,
- for each draw β^r, compute the probability : $P_{il}^r = \frac{e^{\beta^r x_{il}}}{\sum_j e^{\beta^r x_{ij}}}$
- compute the average of these probabilities : $\bar{\mathrm{P}}_{il} = \sum_{r=1}^n \mathrm{P}_{il}/R$
- compute the log-likelihood for these probabilities,
- iterate until the maximum.

Drawing from densities

To estimate a model using simulations, one needs to draw pseudo random numbers from a specified distribution. For this purpose, what is actually needed is a function that draws pseudo random numbers from a uniform distribution between 0 and 1. These numbers are then transformed using the quantile function of the required distribution.

For example, suppose one needs to drawn numbers from the Gumbell distribution. The cumulative distribution of a Gumbell variable is $F(x) = e^{-e^{-x}}$. The quantile function is obtained by inverting this function :

$$\Rightarrow F^{-1}(x) = -\ln(-\ln x)$$

and R draws from a Gumbell distribution are obtained by computing $F^{-1}(x)$ for R draws from the uniform distribution between 0 and 1. This is illustrated on figure 2.

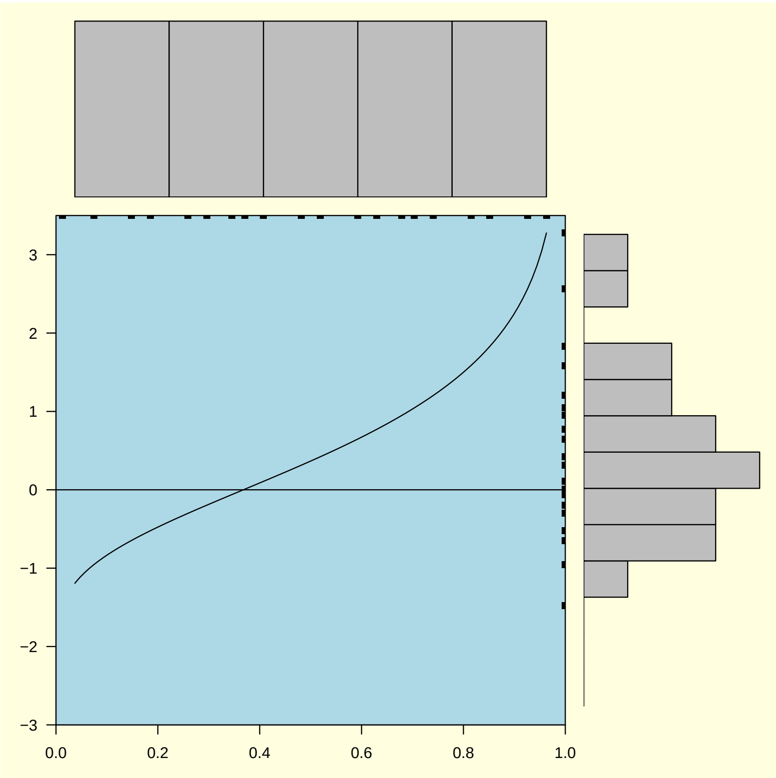

Figure 2: Uniform to Gumbell deviates

The problem is that there may not be a good coverage of the relevant interval instead numerous draws are made. More deterministic methods like Halton draws may be used instead.

Halton sequence

To generate a Halton sequence, we use a prime (e.g. 3). The sequence is then :

- 0 — 1/3 — 2/3,
- 0+1/9 — 1/3+1/9 — 2/3+1/9 — 0+2/9 — 1/3+2/9 — 2/3+2/9,
- 0+1/27 — 1/3+1/27 — 2/3+1/9+1/27 — 1/3+2/9+1/27 — 2/3+2/9+1/27 — 1/3+1/9+2/27 — 2/3+1/9+2/27 — 1/3+2/9+2/27 — 2/3+2/9+2/27

This Halton sequence is illustrated in figure 3.

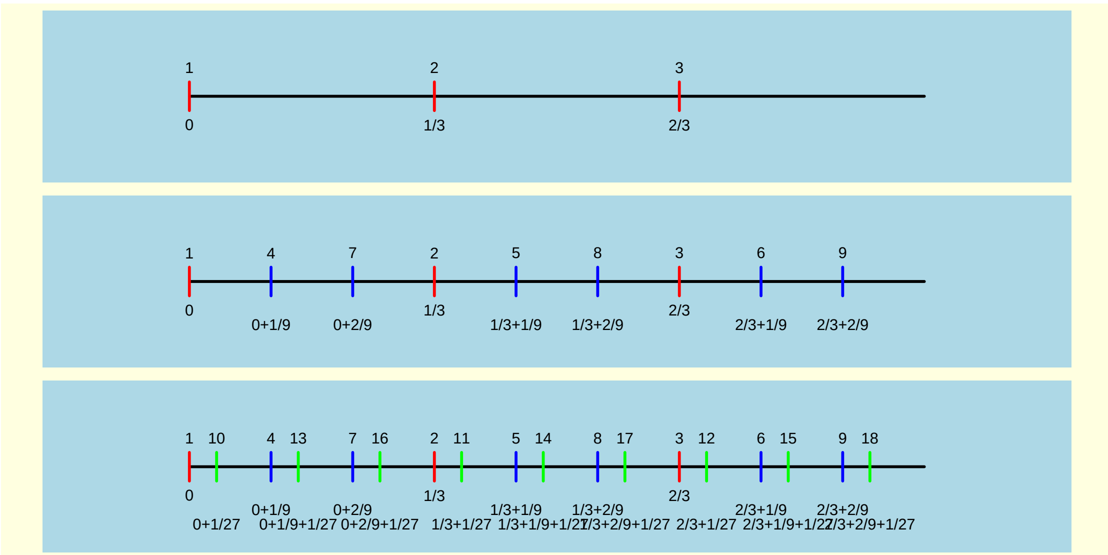

Figure 3: Halton sequences

The use of Halton sequences for two random coefficients is illustrated in figure 4.

On figure 4, one can see that, when using pseudo-random numbers, we have a bad coverage of the unit square, which means that there are some holes (some portions of the unit square where there are no observation and some redundancies (some portions

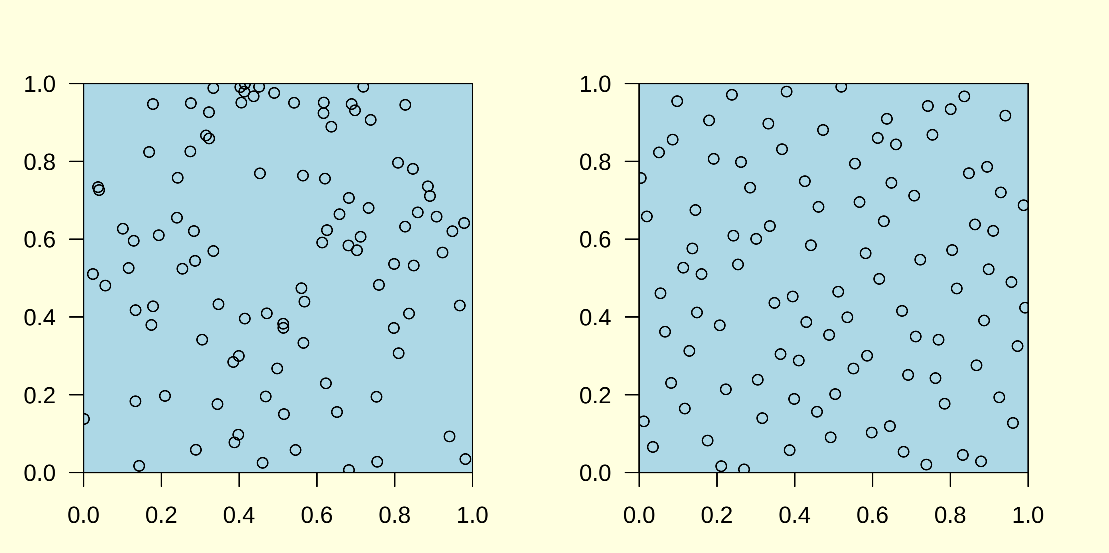

Figure 4: Halton sequences vs random numbers in two dimensions

of the unit square where there are almost identical observations). The coverage of the unit square is much better with Halton draws.

Correlation

It is often relevant to introduce correlations between random parameters. This is done using Choleski decomposition. Let Ω be the covariance matrix of two random parameters. As a covariance matrix is necessarily positive definite, it can be written $\Omega = C^\top C$, with C an upper triangular matrix :

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

so that :

$$\Omega = C^\top C = \begin{pmatrix} c_{11}^2 & c_{11}c_{12} \\ c_{11}c_{12} & c_{12}^2 + c_{22}^2 \end{pmatrix}$$

If $c_{12} = 0$, Ω reduces to a diagonal matrix and the remaining two parameters (c_{11}, c_{22}) are the standard deviations of the two random coefficients. To obtain a couple of correlated coefficients, one has to post-multiply a matrix of uncorrelated coefficients by the Choleski matrix.

If $\mathrm{V}(\eta_1, \eta_2) = I$, then the variance of $(\nu_1, \nu_2) = (\eta_1\eta_2)C$ is Ω

As an example, suppose that the covariance matrix is :

$$\Omega = \begin{pmatrix} 0.5 & 0.8 \\ 0.8 & 2.0 \end{pmatrix}$$

The Choleski matrix is :

$$C = \begin{pmatrix} 0.71 & 1.13 \\ 0 & 0.85 \end{pmatrix}$$

Starting with two uncorrelated parameters (η_1, η_2), we obtain the following two correlated coefficients (ν_1, ν_2) with covariance matrix Ω :

$$\left\{ \begin{array}{lcl} \nu_1 & = & 0.71\eta_1 \\ \nu_2 & = & 1.13\eta_1 + 0.85\eta_2 \end{array} \right.$$

This situation is illustrated by the figure 5.

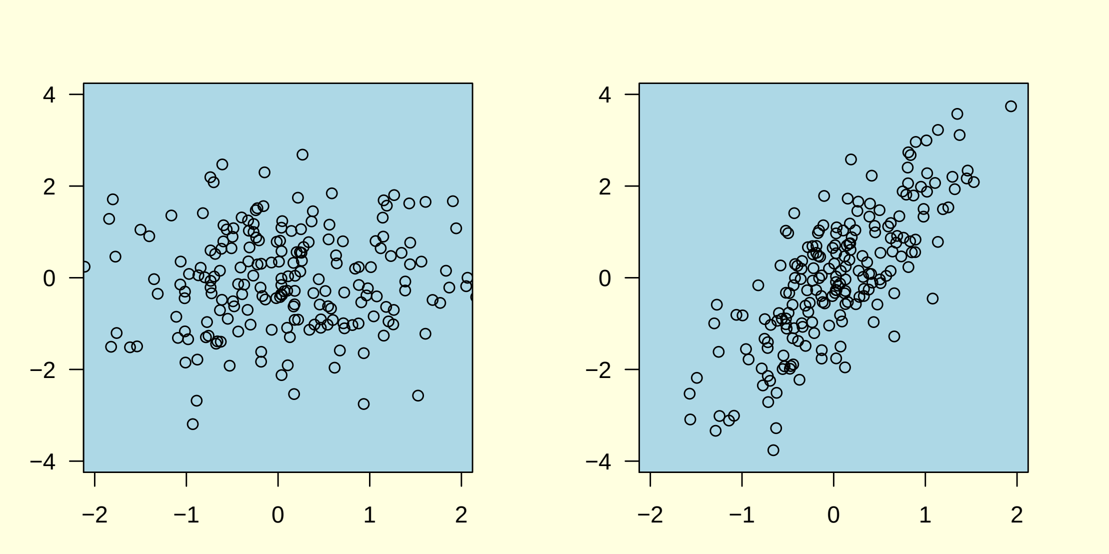

Figure 5: Correlation

4.4. Application

We use the Train data set to illustrate the estimation of a mixed logit model. The random parameter logit model is estimated by providing a `rpar` argument to mlogit. This argument is a named vector, the names being the random coefficients and the values the name of the law (for example `'n'` for a normal distribution). `R` is the number of draws, `halton` indicates whether halton draws should be used (`NA` indicates that default halton draws are used), `panel` and `correlation` are logical values that indicate that the panel version of the mixed logit model is estimated and that the correlation between random coefficients is taken into account.

We estimate a model with three random parameters, time, change and comfort. Two mixed logit models are estimated : Train.mxlc is a correlated model and Train.mxlu is an uncorrelated model. A basic multinomial model ml is also estimated.

```
data("Train", package = "mlogit")
Tr <- mlogit.data(Train, shape = "wide", varying = 4:11,
                  choice = "choice", sep = "_",
                  opposite = c("price", "time", "change", "comfort"),
                  alt.levels=c("A", "B"), id.var ="id")
```

```
Train.ml <- mlogit(choice ~ price + time + change + comfort, Tr)
Train.mxlc <- mlogit(choice ~ price + time + change + comfort, Tr,
               panel = TRUE, rpar = c(time = "cn", change = "n", comfort = "ln"),
               correlation = TRUE, R = 100, halton = NA)
Train.mxlu <- update(Train.mxlc, correlation = FALSE)
```

The summary method supplies the usual table of coefficients, and also some statistics about the random parameters. Random parameters may be extracted using the function rpar which take as first argument a `mlogit` object, as second argument `par` the parameter(s) to be extracted and as third argument `norm` the coefficient (if any) that should be used for normalization. This is usually the coefficient of the price (taken as a non random parameter), so that the effects can be interpreted as monetary values. This function returns a `rpar` object, and several methods/functions are provided to describe it :

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

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.00000  0.00000 13.67638 30.05839 51.82477      Inf

med(time.value)

## [1] 13.67638

mean(time.value)

## [1] 30.05839

stdev(time.value)

## [1] 37.58519
```

In case of correlated random parameters further functions are provided to analyse the correlation of the coefficients :

```
cor.mlogit(Train.mxlc)

##                  time     change    comfort
## time      1.00000000 -0.1236958 0.06010036
## change   -0.12369582  1.0000000 0.80243793
## comfort   0.06010036  0.8024379 1.00000000

cov.mlogit(Train.mxlc)

##                  time      change    comfort
## time      0.01260989 -0.02249451 0.01703551
## change   -0.02249451  2.62259734 3.28019485
## comfort   0.01703551  3.28019485 6.37154524

stdev(Train.mxlc)

##        time      change     comfort
##   0.0746228   1.6194435 513.2326703
```

5. Multinomial Probit

5.1. The model

The multinomial probit is obtained with the same modeling that we used while presenting the random utility model. The utility of an alternative is still the sum of two components : $U_j = V_j + \epsilon_j$.

but the joint distribution of the error terms is now a multivariate normal with mean 0 and with a matrix of covariance denoted Ω[8].

Alternative l is chosen if :

$$\left\{\begin{array}{ccc} U_1 - U_l & = & (V_1 - V_l) + (\epsilon_1 - \epsilon_l) < 0 \\ U_2 - U_l & = & (V_2 - V_l) + (\epsilon_2 - \epsilon_l) < 0 \\ & \vdots & \\ U_J - U_l & = & (V_J - V_l) + (\epsilon_J - \epsilon_l) < 0 \end{array}\right.$$

wich implies, denoting $V_j^l = V_j - V_l$:

[8] see Hausman and Wise (1978) and Daganzo (1979).

$$\left\{\begin{array}{ccccc} \epsilon^l_1 & = & (\epsilon_1 - \epsilon_l) & < & -V^l_1 \\ \epsilon^l_2 & = & (\epsilon_2 - \epsilon_l) & < & -V^l_2 \\ & \vdots & & \vdots & \\ \epsilon^l_J & = & (\epsilon_J - \epsilon_l) & < & -V^l_J \end{array}\right.$$

The initial vector of errors ϵ are transformed using the following transformation :

$$\epsilon^l = M^l\epsilon$$

where the transformation matrix M^l is a $(J-1)\times J$ matrix obtained by inserting in an identity matrix a l^{th} column of -1. For example, if $J = 4$ and $l = 3$:

$$M^3 = \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

The covariance matrix of the error differences is obtained using the following matrix :

$$\mathrm{V}\left(\epsilon^l\right) = \mathrm{V}\left(M^l\epsilon\right) = M^l\mathrm{V}\left(\epsilon\right){M^l}^\top = M^l\Omega {M^l}^\top$$

The probability of choosing l is then :

$$P_l = \mathrm{P}(\epsilon^l_1 < -V^l_1 \;\&\; \epsilon^l_2 < -V^l_2 \;\&\; \ldots\; \epsilon^l_J < -V^l_J) \tag{13}$$

with the hypothesis of distribution, this writes :

$$P_l = \int_{-\infty}^{-V^l_1}\int_{-\infty}^{-V^l_2}\ldots\int_{-\infty}^{-V^l_J}\phi(\epsilon^l)d\epsilon^l_1 d\epsilon^l_2\ldots d^l_J \tag{14}$$

with :

$$\phi\left(\epsilon^l\right) = \frac{1}{(2\pi)^{(J-1)/2} \mid \Omega^l \mid^{1/2}} e^{-\frac{1}{2}\epsilon^l{\Omega^l}^{-1}\epsilon^l} \tag{15}$$

Two problems arise with this model :

- the first one is that the identified parameters are the elements of Ω^l and not of Ω. We must then carefully investigate the meanings of these elements.
- the second one is that the probability is a $J-1$ integral, which should be numerically computed. The relevant strategy in this context is to use simulations.

5.2. Identification

The meaning-full parameters are those of the covariance matrix of the error Ω. For example, with $J = 3$:

$$\Omega = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{pmatrix}$$

$$\Omega^1 = M^1 \Omega M^{1\top} = \begin{pmatrix} \sigma_{11} + \sigma_{22} - 2\sigma_{12} & \sigma_{11} + \sigma_{23} - \sigma_{12} - \sigma_{13} \\ \sigma_{11} + \sigma_{23} - \sigma_{12} - \sigma_{13} & \sigma_{11} + \sigma_{33} - 2\sigma_{13} \end{pmatrix}$$

The overall scale of utility being unidentified, one has to impose the value of one of the variance, for example the first one is fixed to 1. We then have :

$$\Omega^1 = \begin{pmatrix} 1 & \frac{\sigma_{11}+\sigma_{23}-\sigma_{12}-\sigma_{13}}{\sigma_{11}+\sigma_{22}-2\sigma_{12}} \\ \frac{\sigma_{11}+\sigma_{23}-\sigma_{12}-\sigma_{13}}{\sigma_{11}+\sigma_{22}-2\sigma_{12}} & \frac{\sigma_{11}+\sigma_{33}-2\sigma_{13}}{\sigma_{11}+\sigma_{22}-2\sigma_{12}} \end{pmatrix}$$

Therefore, out the 6 structural parameters of the covariance matrix, only 3 can be identified. Moreover, it's almost impossible to interpret these parameters.

More generally, with J alternatives, the number of the parameters of the covariance matrix is $(J + 1) \times J/2$ and the number of identified parameters is $J \times (J - 1)/2 - 1$.

5.3. Simulations

Let L^l be the Choleski decomposition of the covariance matrix of the error differences :

$$\Omega^l = L^l L^{l\top}$$

This matrix is a lower triangular matrix of dimension $(J - 1)$:

$$L^l = \begin{pmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ l_{31} & l_{32} & l_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{(J-1)1} & l_{(J-1)2} & l_{(J-1)3} & \dots & l_{(J-1)(J-1)} \end{pmatrix}$$

Let η be a vector of standard normal deviates :

$$\eta \sim N(0, I)$$

Therefore, we have :

$$\mathrm{V}\left(L^l \eta\right) = L^l V(\eta) L^{l\top} = L^l I L^{l\top} = \Omega^l$$

Therefore, if we draw a vector of standard normal deviates η and apply to it this transformation, we get a realization of ϵ^l.

This joint probability can be written as a product of conditional and marginal probabilities :

$$\begin{array}{rcl} P_l & = & \mathrm{P}(\epsilon^l_1 < -V^l_1 \ \& \ \epsilon^l_2 < -V^l_2 \ \& \ \ldots \ \& \ \epsilon^l_J < -V^l_J)) \\ & = & \mathrm{P}(\epsilon^l_1 < -V^l_1)) \\ & \times & \mathrm{P}(\epsilon^l_2 < -V^l_2 \mid \epsilon^l_1 < -V^l_1) \\ & \times & \mathrm{P}(\epsilon^l_3 < -V^l_3 \mid \epsilon^l_1 < -V^l_1 \ \& \ \epsilon^l_2 < -V^l_2) \\ & \vdots & \\ & \times & \mathrm{P}(\epsilon^l_J < -V^l_J \mid \epsilon^l_1 < -V^l_1 \ \& \ \ldots \ \& \ \epsilon^l_{J-1} < -V^l_{J-1})) \end{array}$$

The vector of error differences deviates is :

$$\begin{pmatrix} \epsilon^l_1 \\ \epsilon^l_2 \\ \epsilon^l_3 \\ \vdots \\ \epsilon^l_J \end{pmatrix} = \begin{pmatrix} l_{11} & 0 & 0 & \ldots & 0 \\ l_{21} & l_{22} & 0 & \ldots & 0 \\ l_{31} & l_{32} & l_{33} & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{(J-1)1} & l_{(J-1)2} & l_{(J-1)3} & \ldots & l_{(J-1)(J-1)} \end{pmatrix} \times \begin{pmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \vdots \\ \eta_J \end{pmatrix}$$

$$\begin{pmatrix} \epsilon^l_1 \\ \epsilon^l_2 \\ \epsilon^l_3 \\ \vdots \\ \epsilon^l_J \end{pmatrix} = \begin{pmatrix} l_{11}\eta_1 \\ l_{21}\eta_1 + l_{22}\eta_2 \\ l_{31}\eta_1 + l_{32}\eta_2 + l_{33}\eta_3 \\ \vdots \\ l_{(J-1)1}\eta_1 + l_{(J-1)2}\eta_2 + \ldots + l_{(J-1)(J-1)}\eta_{J-1} \end{pmatrix}$$

Let's now investigate the marginal and conditional probabilities :

- the first one is simply the marginal probability for a standard normal deviates, therefore we have : $\mathrm{P}(\epsilon^l_1 < -V^l_1) = \Phi\left(-\frac{V^l_1}{l_{11}}\right)$
- the second one is, for a given value of η_1 equal to $\Phi\left(-\frac{V^l_2 + l_{21}\eta_1}{l_{22}}\right)$. We then have to compute the mean of this expression for any value of η_1 lower than $-\frac{V^l_1}{l_{11}}$. We then have, denoting $\bar{\phi}_1$ the truncated normal density :

$$\mathrm{P}(\epsilon^l_2 < -V^l_2) = \int_{-\infty}^{-\frac{V^l_1}{l_{11}}} \Phi\left(-\frac{V^l_2 + l_{21}\eta_1}{l_{22}}\right) \bar{\phi}_1(\eta_1) d\eta_1$$

- the third one is, for given values of η_1 and η_2 equal to : $\Phi\left(-\frac{V^l_3 + l_{31}\eta_1 + l_{32}\eta_2}{l_{33}}\right)$. We then have :

$$\mathrm{P}(\epsilon_3^l < -V_3^l) = \int_{-\infty}^{-\frac{V_1^l}{l_{11}}} \int_{-\infty}^{-\frac{V_2^l + l_{21}\eta_1}{l_{22}}} \Phi\left(-\frac{V_3^l + l_{31}\eta_1 + l_{32}\eta_2}{l_{33}}\right) \bar{\phi}_1(\eta_1)\bar{\phi}_2(\eta_2) d\eta_1 d\eta_2$$

- and so on.

This probabilities can easily be simulated by drawing numbers from a truncated normal distribution.

This so called GHK algorithm[9] (for Geweke, Hajivassiliou and Keane who developed this algorithm) can be described as follow :

1. compute $\Phi\left(-\frac{V_1^l}{l_{11}}\right)$
2. draw a number called η_1^r from a standard normal distribution upper-truncated at $-\frac{V_1^l}{l_{11}}$ and compute $\Phi\left(-\frac{V_2^l + l_{21}\eta_1^r}{l_{22}}\right)$
3. draw a number called η_2^r from a standard normal distribution upper-truncated at $-\frac{V_2^l + l_{21}\eta_1^r}{l_{22}}$ and compute $\Phi\left(-\frac{V_3^l + l_{31}\eta_1^r + l_{32}\eta_2^r}{l_{33}}\right)$
4. ... draw a number called η_{J-1}^r from a standard normal distribution upper-truncated at $-\frac{V_{J-1}^l + l_{(J-1)1}\eta_1^r + \ldots V_{J-1}^l + l_{(J-1)(J-2)}\eta_{J-2}^r}{l_{(J-1)(J-1)}}$
5. multiply all these probabilities and get a realization of the probability called P_l^r.
6. repeat all these steps many times and average all these probabilities ; this average is an estimation of the probability : $\bar{P}_l = \sum_{r=1}^R P_l^r / R$.

Several points should be noted concerning this algorithm :

- the utility differences should be computed respective to the chosen alternative for each individual,
- the Choleski decomposition used should relies on the same covariance matrix of the errors. One method to attained this goal is to start from a given difference, *e.g.* the difference respective with the first alternative. The vector of error difference is then ϵ^1 and its covariance matrix is $\Omega^1 = L^1 L^{1\top}$. To apply a difference respective with an other alternative l, we construct a matrix called S^l which is obtained by

[9] see for example Geweke, Keane, and Runkle (1994).

using a $J-2$ identity matrix, adding a first row of 0 and inserting a column of -1 at the $l-1^{\text{th}}$ position. For example, with 4 alternatives and $l=3$, we have :

$$S^3 = \begin{pmatrix} 0 & -1 & 0 \\ 1 & -1 & 0 \\ 0 & -1 & 1 \end{pmatrix}$$

The elements of the choleski decomposition of the covariance matrix is then obtained as follow :

$$\Omega^l = S^l \Omega^1 S^{l\top} = L^l L^{l\top}$$

- to compute draws from a normal distribution truncated at a, the following trick is used : take a draw μ from a uniform distribution (between 0 and 1) ; then $\eta = \Phi^{-1}\left(\mu\Phi(a)\right)$ is a draw from a normal distribution truncated at a

5.4. Applications

We use again the Fishing data frame, with only a subset of three alternatives used. The multinomial probit model is estimated using mlogit with the `probit` argument equal to `TRUE`

```
data("Fishing", package = "mlogit")
Fish <- mlogit.data(Fishing, shape="wide", varying=2:9, choice="mode")
Fish.mprobit <- mlogit(mode~price | income | catch, Fish, probit = TRUE, alt.subset=c('beach
```

```
summary(Fish.mprobit)

##
## Call:
## mlogit(formula = mode ~ price | income | catch, data = Fish,
##     alt.subset = c("beach", "boat", "pier"), probit = TRUE)
##
## Frequencies of alternatives:
##   beach    boat    pier
## 0.18356 0.57260 0.24384
##
## bfgs method
## 14 iterations, 0h:0m:19s
## g'(-H)^-1g = 9.77E-07
## gradient close to zero
##
## Coefficients :
```

```
##                      Estimate  Std. Error z-value  Pr(>|z|)
## boat:(intercept)  7.2514e-01  3.5809e-01  2.0250 0.0428661 *
## pier:(intercept)  6.2393e-01  2.7396e-01  2.2774 0.0227617 *
## price            -1.2154e-02  1.7697e-03 -6.8681 6.505e-12 ***
## boat:income       2.4005e-06  3.6698e-05  0.0654 0.9478448
## pier:income      -6.5419e-05  4.0832e-05 -1.6022 0.1091198
## beach:catch       1.5479e+00  4.3002e-01  3.5995 0.0003188 ***
## boat:catch        4.0010e-01  4.1600e-01  0.9618 0.3361595
## pier:catch        1.2747e+00  5.5863e-01  2.2819 0.0224968 *
## boat.pier         5.4570e-01  4.6263e-01  1.1795 0.2381809
## pier.pier         6.9544e-01  2.9294e-01  2.3740 0.0175973 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -478.43
## McFadden R^2:  0.32751
## Likelihood ratio test : chisq = 465.99 (p.value = < 2.22e-16)
```

6. Tests

6.1. The three tests

As for all models estimated by maximum likelihood, three testing procedures may be applied to test hypothesis about models fitted using mlogit. The hypothesis tested define two models :

- the unconstrained model that doesn't take these hypothesis into account,
- the constrained model that impose these hypothesis.

This in turns define three principles of tests :

- the *Wald test* is based only on the unconstrained model,
- the *Lagrange multiplier test* (or *score test*) is based only on the constrained model,
- the *Likelihood ratio test* is based on the comparison of both models.

The three principles of test are better understood using figure 6.

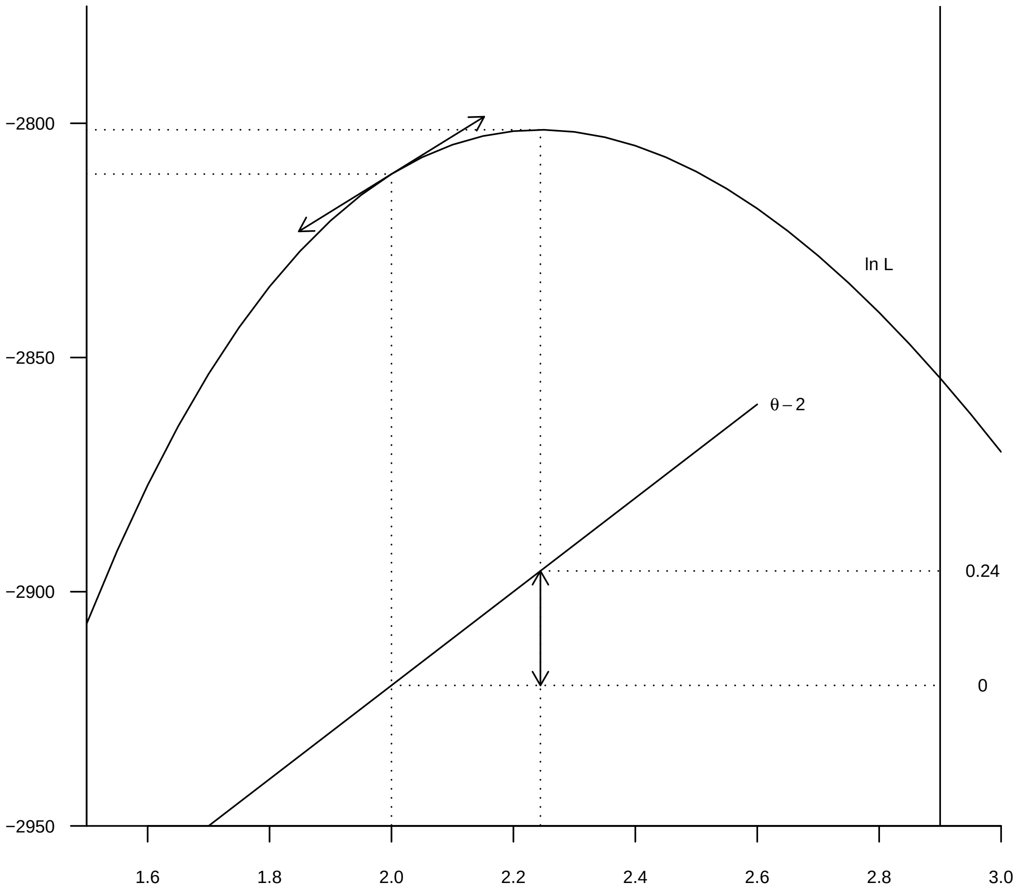

Figure 6: The three tests

In this one dimensional setting, the hypothesis is of the form $\theta = \theta_o$, which can be written $f(\theta) = \theta - \theta_o$, with $f(\theta) = 0$ if the hypothesis is unforced. This is the equation of a straight line on figure 6 . The constrained model is just $\hat{\theta}_c = \theta_o$, *i.e.* the constrained model is not estimated. The unconstrained model corresponds to the maximum of the curve that represents the log-likelihood function.

- The Wald test is based on $f(\hat{\theta}_{nc}) = R\hat{\theta}_{nc} - q$ which is depicted by the arrow in figure 6. More generally, it is a vector of length J, whose expected value should be 0 if the hypothesis is true : $\mathrm{E}(R\hat{\theta}_{nc} - q) = R\theta - q$. Its variance is : $\mathrm{V}(R\hat{\theta}_{nc} - q) = R\mathrm{V}(\hat{\theta}_{nc})R^\top$. $(R\hat{\theta}_{nc} - q) \sim N\left(R\theta - q, R\mathrm{V}(\hat{\theta}_{nc})R^\top\right)$. If the hypothesis are true, the quadratic form is a chi-squared with J degrees of freedom :

$$t_{\text{wald}} = (R\hat{\theta}_{nc} - q)^\top \left(R\mathrm{V}(\hat{\theta}_{nc})R^\top\right)^{-1} (R\hat{\theta}_{nc} - q)$$

- The Lagrange multiplier is based on the gradient (the slope of the likelihood curve) evaluated at the constrained model : $\frac{\partial \ln L}{\partial \theta}(\hat{\theta}_c)$. Here again, this should be a random vector with expected value equal to 0 if H_o is true. The variance of the gradient is : $\mathrm{V}\left(\frac{\partial \ln L}{\partial \theta}(\hat{\theta}_c)\right) = \mathrm{E}\left(\frac{\partial^2 \ln L}{\partial \theta \partial \theta^\top}(\theta)\right)$. $\frac{\partial \ln L}{\partial \theta}(\hat{\theta}_c) \sim N(0, \mathrm{E}\left(\frac{\partial^2 \ln L}{\partial \theta \partial \theta^\top}\right))$. If the hypothesis are true, the quadratic form is a chi-squared with J degrees of freedom :

$$t_{\text{score}} = \left(\frac{\partial \ln L}{\partial \theta}(\hat{\theta}_c)\right)^\top \mathrm{V}\left(\frac{\partial \ln L}{\partial \theta}(\hat{\theta}_c)\right)^{-1} \left(\frac{\partial \ln L}{\partial \theta}(\hat{\theta}_c)\right)$$

- Finally, the likelihood ratio test compares both models. More specifically, the statistic is twice the value of the log-likelihood for the two models and, if the hypothesis are true, is a chi-squared with J degrees of freedom :

$$t_{\text{lr}} = 2\left(\ln L_{nc} - \ln L_c\right)$$

Two of these tests are implemented in the **lmtest** package (Zeileis and Hothorn 2002) : waldtest and lrtest. The wald test is also implemented in linearHypothesis from package **car** 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'll see later that the score test is especially useful for `mlogit` objects when one is interested in extending the basic multinomial logit model. In this case, the unconstrained model is much more difficult to estimate than the constrained model which is the basic multinomial logit model. The score test, which is based on the constrained model is therefore very simple to compute.

For now, we'll just demonstrate the use of the testing in the usual setting where the two models are provided. This can done by passing two fitted models to the testing function, or just one model and a formula which describes the second model.

We've previously estimated the following model :

```
ml.Fish <- mlogit(mode~price | income | catch, Fishing, shape = "wide", varying = 2:9)
```

The hypothesis that the income doesn't influence the choice for a fishing mode is a joint hypothesis that three coefficients are zero. The constrained model can be obtained by updating the previous model :

```
ml.Fish.c <- update(ml.Fish, . ~ . | . - income | .)
```

The wald and likelihood ratio tests are then obtained by providing the two models as arguments :

```
waldtest(ml.Fish, ml.Fish.c)

## Wald test
##
## Model 1: mode ~ price | income | catch
## Model 2: mode ~ price | 1 | catch
##   Res.Df Df  Chisq Pr(>Chisq)
## 1   1171
## 2   1174 -3 28.613  2.701e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

lrtest(ml.Fish, ml.Fish.c)

## Likelihood ratio test
##
## Model 1: mode ~ price | income | catch
## Model 2: mode ~ price | 1 | catch
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  11 -1199.1
## 2   8 -1214.2 -3 30.138  1.291e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

scoretest(ml.Fish.c, ml.Fish)

##
##  score test
##
## data:  mode ~ price | income | catch
## chisq = 29.71, df = 3, p-value = 1.588e-06
## alternative hypothesis: unconstrained model
```

or just one of them and a formula that describes the second one :

```
lrtest(ml.Fish, . ~ . | . - income | .)
lrtest(ml.Fish, mode ~ price | 1 | catch)
lrtest(ml.Fish.c, . ~ . | . + income | .)
lrtest(ml.Fish.c, mode ~ price | income | catch)
waldtest(ml.Fish, . ~ . | . - income | .)
waldtest(ml.Fish, mode ~ price | 1 | catch)
waldtest(ml.Fish.c, . ~ . | . + income | .)
waldtest(ml.Fish.c, mode ~ price | income | catch)
scoretest(ml.Fish.c, . ~ . | . + income | .)
scoretest(ml.Fish.c, mode ~ price | income | catch)
```

6.2. Test of heteroscedasticity

The homoscedascticity 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 pass only the fitted model as argument. In this case, it is guessed that the hypothesis that the user wants to test is the homoscedasticity hypothesis. We'll test the homoscedasticity hypothesis for the two heteroscedastic models (hl.MC and hl.TM estimated previously, with the RdModeCanada and the TravelMode and data sets.

```
lrtest(hl.MC, ml.MC)

## Likelihood ratio test
##
## Model 1: choice ~ freq + cost + ivt + ovt | urban + income
## Model 2: choice ~ freq + cost + ivt + ovt | urban + income
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  12 -1838.1
## 2  10 -1841.6 -2 6.8882    0.03193 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

waldtest(hl.MC, heterosc = FALSE)

##
##  Wald test
##
## data:  homoscedasticity
## chisq = 25.196, df = 2, p-value = 3.38e-06
```

or, more simply :

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

The wald test can also be computed using the linearHypothesis function from the **car** package (Fox and Weisberg 2010) :

```
library("car")
linearHypothesis(hl.MC, c('sp.air=1', 'sp.train=1'))

## Linear hypothesis test
##
## Hypothesis:
## sp.air = 1
## sp.train = 1
##
## Model 1: restricted model
## Model 2: choice ~ freq + cost + ivt + ovt | urban + income
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1   2759
## 2   2757  2 25.195   3.38e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 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.

```
scoretest(ml.MC, heterosc = TRUE)

##
##  score test
##
## data:  heterosc = TRUE
## chisq = 9.4883, df = 2, p-value = 0.008703
## alternative hypothesis: heteroscedastic model
```

The homoscedasticity hypothesis is strongly rejected using any of the three tests.

For the hl.TM model, the standard deviations of the estimated scale parameters are very high, which means that they are poorly identified. This is confirmed by the fact that the homoscedasticity hypothesis is not rejected :

```
c(wald = waldtest(hl.TM)$statistic,
  lr = lrtest(hl.TM)$Chisq[2],
  score = scoretest(ml.TM, heterosc = TRUE)$statistic)

##  wald.chisq          lr score.chisq
##    3.635586    6.935712   21.565985
```

6.3. Test about the nesting structure

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 1,
- the test of unique nest elasticities, which means that all the nest elasticities are equal to each other.

To illustrate the use of these tests, we'll use the nl.HC model estimated using the HC data set.

For the test of no nests, the nested model is provided as the unique argument for the lrtests and the waldtest function. For the scoretest, the constrainted 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.

```
lrtest(nl.HC)

## Likelihood ratio test
##
## Model 1: depvar ~ occa + icca + och + ich
## Model 2: depvar ~ occa + icca + och + ich
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  12 -188.03
## 2  10 -192.88 -2 9.6853   0.007886 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

waldtest(nl.HC)

##
##  Wald test
##
## data:  no nests
## chisq = 15.307, df = 2, p-value = 0.0004744
```

```
scoretest(ml.HC, nests = list(cooling = c('ecc', 'erc', 'gcc', 'hpc'),
                   noncool = c('ec', 'gc', 'er')))

##
##  score test
##
## data:  cooling, noncool
## chisq = 15.176, df = 2, p-value = 0.0005065
## alternative hypothesis: nested model
```

The wald test can also be performed using the linearHypothesis function :

```
linearHypothesis(nl.HC, c("iv:cooling = 1", "iv:noncool = 1"))

## Linear hypothesis test
##
## Hypothesis:
## iv:cooling = 1
## iv:noncool = 1
##
## Model 1: restricted model
## Model 2: depvar ~ occa + icca + och + ich
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1    240
## 2    238  2 15.307  0.0004744 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The three tests reject the null hypothesis of no correlation at the 1% level. The two nests elasticities being very closed, we'd like to test the equality between both elasticities. This can be performed using the three tests. For the score test, we provide the constrained model that is called nl.HC.u

```
lrtest(nl.HC,nl.HC.u)

## Likelihood ratio test
##
## Model 1: depvar ~ occa + icca + och + ich
## Model 2: depvar ~ occa + icca + och + ich
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  12 -188.03
## 2  11 -188.03 -1 0.0012     0.9723
```

```
waldtest(nl.HC, un.nest.el = TRUE)

##
##  Wald test
##
## data:  unique nest elasticity
## chisq = 0.0010706, df = 1, p-value = 0.9739

scoretest(nl.HC.u, un.nest.el = FALSE)

##
##  score test
##
## data:  un.nest.el = FALSE
## chisq = 0.0013939, df = 1, p-value = 0.9702
## alternative hypothesis: unique nest elasticity

linearHypothesis(nl.HC, "iv:cooling = iv:noncool")

## Linear hypothesis test
##
## Hypothesis:
## iv:cooling - iv:noncool = 0
##
## Model 1: restricted model
## Model 2: depvar ~ occa + icca + och + ich
##
##   Res.Df Df  Chisq Pr(>Chisq)
## 1    239
## 2    238  1 0.0011     0.9739
```

6.4. Test of random parameters

The three tests can be applied to test the specification of the model, namely the presence of random coefficients and their correlation. Actually, three nested models can be considered :

- a model with no random effects,
- a model with random but uncorrelated effects,
- a model with random and correlated effects.

These three models have been previously estimated for the example based on the Train data set under the names of Train.ml, Train.mxlu and Train.mxlc.

We first present the three tests of no random-uncorrelated effects.

```
lrtest(Train.mxlu, Train.ml)

## Likelihood ratio test
##
## Model 1: choice ~ price + time + change + comfort
## Model 2: choice ~ price + time + change + comfort
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   8 -1589.5
## 2   5 -1723.8 -3 268.57  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

waldtest(Train.mxlu)

##
##  Wald test
##
## data:  no random effects
## chisq = 884810, df = 3, p-value < 2.2e-16

scoretest(Train.ml, rpar = c(time = "n", change = "n", comfort = "n"), R = 100,
          correlation = FALSE, halton = NA, panel = TRUE)

##
##  score test
##
## data:  rpar(time='n',change='n',comfort='n')
## chisq = 298.31, df = 3, p-value < 2.2e-16
## alternative hypothesis: no uncorrelated random effects
```

Next, we present the three tests of no random-correlated effects.

```
lrtest(Train.mxlc, Train.ml)

## Likelihood ratio test
##
## Model 1: choice ~ price + time + change + comfort
## Model 2: choice ~ price + time + change + comfort
```

```
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  11 -1556.2
## 2   5 -1723.8 -6 335.22  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

waldtest(Train.mxlc)

##
##  Wald test
##
## data:  no random effects
## chisq = 160550, df = 6, p-value < 2.2e-16

scoretest(Train.ml, rpar = c(time = "n", change = "n", comfort = "n"), R = 100,
          correlation = TRUE, halton = NA, panel = TRUE)

##
##  score test
##
## data:  rpar(time='n',change='n',comfort='n')
## chisq = 294.73, df = 6, p-value < 2.2e-16
## alternative hypothesis: no correlated random effects
```

Finally, we present the three tests of no correlation, the existence of random parameters being maintained.

```
lrtest(Train.mxlc, Train.mxlu)

## Likelihood ratio test
##
## Model 1: choice ~ price + time + change + comfort
## Model 2: choice ~ price + time + change + comfort
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1  11 -1556.2
## 2   8 -1589.5 -3 66.653  2.221e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

waldtest(Train.mxlc, correlation = FALSE)

##
```

```
##  Wald test
##
## data:  uncorrelated random effects
## chisq = 129210, df = 3, p-value < 2.2e-16

scoretest(Train.mxlu, correlation = TRUE)

##
##  score test
##
## data:  correlation = TRUE
## chisq = 1536.9, df = 3, p-value < 2.2e-16
## alternative hypothesis: uncorrelated random effects
```

References

Beggs S, Cardell S, Hausman J (1981). "Assessing the potential demand for electric cars." *Journal of econometrics*, **16**, 1–19.

Bhat C (1995). "A heterocedastic extreme value model of intercity travel mode choice." *Transportation Research B*, **29**(6), 471–483.

Chapman R, Staelin R (1982). "Exploiting rank ordered choice set data within the stochastic utility model." *Journal of marketing research*, **14**, 288–301.

Daganzo C (1979). *Multinomial probit: the theory and its application to demand forecasting*. Academic Press, New York.

Daly A (1987). "Estimating 'tree' logit models." *Transportation Research B*, pp. 251–267.

Fok D, Paap R, van Dijk B (2010). "A rank-ordered logit model with unobserved heterogeneity in ranling capabilities." *Journal of applied econometrics*. Forthcoming.

Fox J, Weisberg S (2010). *An R Companion to Applied Regression*. Second edition. Sage, Thousand Oaks CA. URL http://socserv.socsci.mcmaster.ca/jfox/Books/Companion.

Geweke J, Keane M, Runkle D (1994). "Alternative computational approaches to inference in the multinomial probit model." *Review of economics and statistics*, **76**, 609–632.

Greene W (2008). *Econo metric Analysis*. 6 edition. Prentice Hall.

Hausman J, Ruud P (1987). "Specifying and testing econometric models for rank-ordered data." *Journal of econometrics*, **34**, 83–103.

Hausman J, Wise D (1978). "A conditional probit model for qualitative choice: discrete decisions recognizing interdemendence and heterogeneous preferences." *Econometrica*, **48**, 403–429.

Heiss F (2002). "Structural choice analysis with nested logit models." *The Stata Journal*, **2**(3), 227–252.

Hensher DA, Greene WH (2002). "Specification and estimation of the nested logit model: alternative normalisations." *Transportation Research Part B*, **36**, 1–17.

Koppelman FS, Wen CH (1998). "Alternative nested logit models: structure, properties and estimation." *Transportation Research B*, **32**(5), 289–298.

Koppelman FS, Wen CH (2000). "The paired combinatorial logit model: properties, estimation and application." *Transportation Research B*, **34**, 75–89.

Louiviere J, Hensher D, Swait J (2000). *Stated choice methods: analysis and appliacation*. Cambridge University Press, Cambridge, UK.

McFadden D (1974). "The measurment of urban travel demand." *Journal of public economics*, **3**, 303–328.

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.

Small KA, Rosen HS (1981). "Applied welfare economics with discrete choice models." *Econometrica*, **49**, 105–130.

Toomet O, Henningsen A (2010). *maxLik: Maximum Likelihood Estimation*. R package version 0.8-0, URL http://CRAN.R-project.org/package=maxLik.

Train KE (2003). *Discrete choice methods with simulation*. Cambridge University Press, Cambridge, UK.

Wen CH, Koppelman FS (2001). "The generalized nested logit model." *Transportation Research part B*, **35**, 627–641.

Zeileis A, Croissant Y (2010). "Extended Model Formulas in R: Multiple Parts and Multiple Responses." *Journal of Statistical Software*, **34**(1), 1–13. URL http://www.jstatsoft.org/v34/i01/.

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

Index

Contents

List of Figures

Affiliation:

Yves Croissant
Faculté de Droit et d'Economie
Université de la Réunion
15, avenue René Cassin
BP 7151
F-97715 Saint-Denis Messag Cedex 9
Telephone: +33/262/938446
E-mail: yves.croissant@univ-reunion.fr