Intelligent Transportation System Laboratory Department of Mathematics

Massachusetts Institute of Technology

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Master Thesis

Sampling of Alternatives for Logit Mixture Models

by

Inès Azaiez

Under the supervision of Professors
Michel Bierlaire and Moshe Ben Akiva
and the assistance of Cristian Angelo Guevara-Cue

Spring 2010

Acknowledgement

I would like to give special thanks to professor Michel Bierlaire for believing in me and whose guidance and support have been a constant source of encouragement. His ideas and advice have been essential in driving the progress of this report.

I would like also to express my gratitude to professor Moshe Ben-Akiva. His encouragement, support, understanding and above all, his constructive and greatly appreciated help and feedback, were invaluable to this project.

I would like also to thank Angelo Guevara for being helpful and offering kind assistance.

Doing my master thesis at MIT has been a great and memorable experience for me and I would like to thank my colleagues in the ITS lab: Swapnil, Marty, Steffen, Heiko, Angelo and all the friends I met in Boston for making my life here enjoyable.
I would like also to thank my friends Utsav and Joao for their kind help in proofreading and correcting my report.
I especially would like to thank my friend Peter for being continuously supportive and thoughtful.

I am also grateful to the EPFL-WISH foundation for providing support for my Master project at MIT.

Finally, I owe my deepest gratitude to my parents, my aunt Faten and uncle Paolo for their unconditional love, support and encouragements.

Contents

Chapter 1

Introduction

Discrete choice theory is concerned with understanding how individuals make choices given a specific set of alternatives. Each alternative in the choice set can be characterized by a set of attributes that are likely to affect the choice of the individual. The model may include some characteristics of the individuals which are likely to explain their choice like the age, the gender, the income.
Starting with the simple binary logit model, research got, during the 1960s and 1970s, to the multinomial logit (MNL) model which has been by far the most used model specification. Indeed, the MNL model provides a convenient closed form for the underlying choice probabilities and has a globally concave likelihood function which avoids computational burden. However, one important restriction in the MNL model is the independence from the irrelevant alternatives property (IIA). In fact, the MNL model has been derived based on the assumption that the error terms are *i.i.d.* across alternatives and individuals. This leads to the IIA property which states that the ratio of the probabilities of choosing any two alternatives is independent of the attributes of any other alternative in the choice set. Although mathematically convenient, this assumption appears restrictive and inaccurate for modeling behavior. For the sake of realism, more complex models have been developed in order to relax the *i.i.d.* assumption like the nested logit models and even more general models like the Multivariate Extreme Value models (MEV) where a more complex structure of correlation among the alternatives can be captured.

The Logit Mixture model, also called mixed logit (K. E. Train, reference[2]) and Logit Kernel Model (J. Walker, reference [3]), is the latest among a set of models developed out of discrete choice theory. The main improvement is that the Logit Mixture model includes a number of additional parameters that capture observed and unobserved heterogeneity[1] among the decision makers in sensitivity to exogenous variables. Although the theory of Logit Mixture models existed from the 1970s, the parameters' estimation was seen as a computational burden which hindered their use. In fact, the choice probabilities in such models are multidimensional integrals over a mixing distribution. The integrals have an

[1]Heterogeneity means individual variations in tastes and preferences

open form and hence must be computed numerically. The breakthrough came with the development of simulation methods which enabled these non closed form models to be estimated with relative ease through simulation using random draws from a chosen mixing distribution.

The Logit Mixture model is hence one of the most attractive and accurate model in discrete choice theory since it avoids the IIA assumption as well as offers heterogeneity among the decision makers. However, when the choice set faced by the decision makers becomes very large, estimation of a Logit Mixture model from the full choice set can be very expensive or even impossible. Sampling of alternatives becomes in these case necessary for estimation. McFadden (1978) showed that sampling of alternatives in a conditional logit model, while avoiding computational burden, leads to a consistent estimation. This is a strong and neat theoretical result which is associated with the IIA property of MNL models. Building on an idea of Ben-Akiva (2009) to achieve sampling of alternatives in non-logit models, Guevara and Ben-Akiva (2010) extended McFadden's results into the Multivariate Extreme Value (MEV) models and showed that the estimation of such models under sampling of alternatives recovers, under certain conditions, the true parameters of the model.
However, there are no theoretical support examining how sampling of alternatives in Logit Mixture models for which IIA property does not hold, affects the empirical accuracy and unbiasedness of the estimated parameters.

Does sampling of Alternatives introduce a significant bias in the parameters' estimation of a Logit Mixture model? Do we need to correct the log-likelihood equation, as done in the case of the MEV models, in order to achieve consistency of the parameters' estimation? The main purpose of this project was to respond to these questions and investigate the impact of sampling of alternatives for a Logit Mixture model. The main conclusion we reached is that sampling of alternatives does not alter, in general, the parameters' estimates for a Logit Mixture model. We also found that correcting the log-likelihood equation introduces a significant bias in the parameters' estimation.

This paper is organized as follows: In chapter 2, we describe briefly the Logit Mixture model and present what has been done concerning sampling of alternatives for discrete choice models. In chapter 3, we present a literature overview on sampling of alternatives for mixture logit models which is mainly based on empirical testing and results. In chapter 4, we describe the different specifications of the Logit Mixture model we considered in our study. Chapter 5 presents our empirical results based on Monte Carlo experiments and discusses our findings. Chapter 6 presents some theoretical ideas which might constitute a start for explaining why sampling of alternative for a Logit Mixture model does not alter the parameters' estimation. Chapter 7 concludes this project. In the appendix, we present and explain part of the code which we implemented for our study.

Chapter 2

Backround

2.1 Logit Mixture Models

The Logit Mixture model structure can be motivated from two different but equivalent manners:

2.1.1 Random coefficients

The random parameter structure capture heterogeneity among the decision makers in sensitivity to exogenous variables.
A decision maker n makes a choice among J alternatives and the utility from alternative $j \in \{1, \dots J\}$ is given by

$$U_{nj} = \beta_n x_{nj} + \epsilon_{nj} \tag{2.1}$$

where x_{nj} is a vector of exogenous attributes, β_n is a vector of random coefficients which varies across individuals with respect to the density $f(\beta|\theta)$ representing hence the variation in taste or heterogeneity among the populations and ϵ_{nj} are $i.i.d$ Extreme Value type I error terms and are independent form the random vector of parameters, β.
The vector β varies over the decision makers with density $f(\beta|\theta)$ where θ represents the parameters of the distribution. For instance, considering the one dimensional case, if the random parameter β is distributed according to a normal distribution, $\beta \sim \mathcal{N}(\mu, \sigma^2)$, then θ represents the mean and the standard deviation of the distribution, $i.e.$ $\theta = (\mu, \sigma)$.

The decision maker chooses the alternative with the highest utility. The researcher observes only the attributes x_{nj} but not the values of the random parameters neither the error terms. Hence, conditioning on the value of the β and since the error term ϵ_{nj} are $i.i.d.$ Extreme Value, the conditional probability that the decision maker n chooses alternative j is the standard logit (McFadden, 1973):

$$\mathcal{P}(\text{individual } n \text{ choosing alt. } j \ |\beta_n) = L_{nj}(\beta_n) = \frac{e^{\beta_n x_{nj}}}{\sum_i e^{\beta x_{ni}}} \tag{2.2}$$

As mentioned before, the researcher does not observe the true values of the parameters β and hence these conditional probabilities are integrated over β according to a mixing distribution $f(\beta|\theta)$. Hence, we have the Logit Mixture probabilities which are given as follows

$$P_{nj} = \int L_{nj}(\beta) f(\beta|\theta) d\beta = \int \left(\frac{e^{\beta x_{nj}}}{\sum_i e^{\beta x_{ni}}}\right) f(\beta|\theta) d\beta \tag{2.3}$$

The researcher chooses a distribution for the random coefficients and the parameters of this mixing distribution given by θ are to be estimated.

This random parameter model does not impose the independence from the irrelevant alternatives property, IIA. Indeed, the ratio of the probability of choosing alternative j to the probability of choosing alternative k for and individual n is

$$\frac{P_{nj}}{P_{nk}} = \frac{\int \left(\frac{e^{\beta x_{nj}}}{\sum_i e^{\beta x_{ni}}}\right) f(\beta|\theta)\beta}{\int \left(\frac{e^{\beta x_{nk}}}{\sum_i e^{\beta x_{ni}}}\right) f(\beta|\theta) d\beta} \tag{2.4}$$

Equation (2.4) clearly does not factor and hence the ratio of probabilities between any two alternatives j and k depends also on the characteristics and attributes of other alternatives. Hence, the IIA property does not hold for Logit Mixture models which have a greater generality in representing heterogeneity among the decision makers.

2.1.2 Error Component

A second motivation of considering Logit Mixture models is to allow an error component that create correlations among the utilities of different alternatives.
We already know that the *i.i.d.* assumption on the error term in the MNL model is restrictive and hence, we would want the error components of the different alternatives to be correlated. One way to do this is to partition the stochastic component into two uncorrelated parts: The first part allows the unobserved error term to be heteroscedastic and correlated over alternatives. The second part is independent and identically distributed, type I Extreme Value, across the alternatives. Specifically, we consider the following utility function for individual n and alternative j

$$U_{nj} = \beta x_{nj} + \eta_{nj} = \beta x_{nj} + \mu_n z_{nj} + \epsilon_{nj} \tag{2.5}$$

where βx_{nj} and η_{nj} are respectively the systematic and stochastic component of the utility, and η_{nj} is further partitioned into two subcomponents, $\mu_n z_{nj}$ and ϵ_{nj}. The vectors x_{nj} and z_{nj} represent observed variables associated to alternative j, β is a vector of fixed coefficients, μ is a vector of random terms with zero mean and ϵ_{nj} is *i.i.d.* Extreme Value.

The component $\mu_n z_{nj}$ induces heteroscedasticity and correlation across the stochastic part of the utility of the different alternatives.

We notice that the random component specification for Logit Mixture model given in equation (2.1) and the error-component specification given in equation (2.5) are equivalent. Indeed, if the random parameters, β_n, are distributed with mean β and deviation μ_n and setting $x_{nj} = z_{nj}$ then the random parameter specification implies the error component specification. Conversely, under the error component specification, the utility given in equation (2.5) can be viewed as a random parameter model with fixed coefficients for the variables x_{nj} and random coefficients with zero mean for the variables z_{nj}. We refer to [2] for more details.
In this paper, we will use the random coefficient specification of the Logit Mixture structure.

2.1.3 Simulation

Simulated Logit Mixture probabilities

The Logit Mixture probabilities take the form of a multidimensional integral over a mixing distribution. This integral does not have a closed form in general and has to be approximated by simulation using random draws from the mixing distribution.
The Monte Carlo simulation method to evaluate the multi-dimensional integrals consists in computing the integrand at a sequence of random points drawn from the mixing distribution and computing the average of the integrand values. A large number of draws is usually needed to ensure reasonably low simulation error and by the strong law of large numbers, convergence is almost sure in this method.

From a mathematical point of view, we recall that the Logit Mixture probability is

$$P_{nj} = \int L_{nj}(\beta) f(\beta|\theta) d\beta \tag{2.6}$$

where

$$L_{nj} = \frac{e^{\beta x_{nj}}}{\sum_{i \in C} e^{\beta x_{ni}}} \tag{2.7}$$

where C is the full choice set. The researcher chooses a mixing distribution $f(\cdot|\theta)$ and wants to estimate the parameters in θ. The integral given by equation (2.6) is approximated by simulation as follows:

For each decision maker, any given value of θ and for $r \in 1, \ldots, R$

1. Draw a value $\beta \sim f(\beta|\theta)$ and label it β^r
2. Calculate the logit formula given by equation (2.7) using the draws: $L_{nj}(\beta^r)$

The Logit Mixture probabilities given in equation (2.6) are estimated by repeating step 1. and step 2. R times, R being the total number of draws, and then averaging out the logit probabilities in equation (2.7) over the draws. The average is the simulated probability and is given as follows:

$$\hat{P}_{nj} = \frac{1}{R} \sum_{r=1}^{R} L_{nj}(\beta^r) \tag{2.8}$$

We notice that, by construction, $\hat{P}_{nj}$ is an unbiased estimator of P_{nj} whose variance decreases as R, the total number of draws, rises.

Simulated Log-Likelihood

The log-likelihood function is given by equation (2.9) as shown below

$$LL(\theta) = \sum_{n=1}^{N} \ln P_{nj} \tag{2.9}$$

where j denotes the chosen alternative by each decision maker n.
Let $\hat{P}_{nj}$ be the simulated Logit Mixture probabilities as given in equation (2.8), then the simulated log-likelihood function is given as follows

$$SLL(\theta) = \sum_{n=1}^{N} \ln \hat{P}_{nj} \tag{2.10}$$

where, as before, j denotes the chosen alternative by each decision maker n.
The maximum simulated likelihood estimator (MSLE) is the value of the parameters in θ that maximizes SLL:

$$\tilde{\theta}_{MSLE} = \arg\max_{\theta} SLL(\theta) \tag{2.11}$$

We notice that, even though $\hat{P}_{nj}$ is an unbiased estimator of P_{nj}, the simulated log-likelihood, SLL, is a biased estimator of the log-likelihood function, LL, due to the nonlinearity introduced by the natural log transformation of the likelihood function. Hence, we have no guarantee on the unbiasedness of the $MSLE$ estimator, *i.e*, in general, we have

$$\mathbb{E}(\arg\max_{\theta} SLL(\theta)) \neq \arg\max_{\theta} LL(\theta)$$

However, the bias of the $MSLE$ estimator decreases as the number of draws rises, and when the number of draws rises faster or as fast as the square root of the number of observations, the $MSLE$ estimator is consistent and has the same property as the classical

maximum likelihood estimator (efficiency & consistency).
These properties of the $MSLE$ estimator have been studied previously in the literature (Lee 1997, reference [4]):
The $MSLE$ estimator is consistent as n and R go both to infinity and is asymptotically efficient when $\lim_{n\to\infty} \sqrt{n} \cdot R^{-1} = 0$.

2.1.4 Identification

It has been proved that there are no identification issues when estimating a Logit Mixture model with continuous attributes of the Alternatives.
When random parameters are specified for continuous attributes of the alternatives, there are no identification issues per se. Data willing, the full covariance structure (i.e. variance for each parameter as well as covariances across parameters) can be estimated (Walker, reference [3]).

2.2 Sampling of alternatives

In large choice set situations, the computational burden and the substantial effort of assembling the dataset, make it difficult to fit a discrete choice model with considering all the alternatives. Additionally, in some cases, the measurement and the identification of each potential alternatives is impossible. These issues have led researchers to explore and apply methods to enable consistent estimation with only a subset of alternatives.
The next section describes McFadden' results of sampling and estimation of a logit model with sampling of alternatives.

2.2.1 Estimation and Sampling of Alternatives in Logit Models

In order to describe Mcfadden's result on sampling of alternatives for logit models, we consider first the random utility U_{nj} of the decision maker n and the alternative j which can be decomposed into a systematic part and a stochastic part as follows

$$U_{nj} = V_{nj} + \epsilon_{nj}$$

where ϵ_{nj} are distributed *i.i.d* Extreme Value with location parameter $\eta = 0$ and scale parameter $\mu = 1$.
The probability that an individual n chooses alternative j from the set of all alternatives C_n available to him/her is given by the logit formula

$$P(\text{ individual } n \text{ chooses alt. } j\ ) = P_n(j) = \frac{e^{V_{nj}}}{\sum_{i \in C} e^{V_{ni}}}$$

We consider now that the researcher, instead of using the whole choice set C_n, will only consider a subset D_n with $\tilde{J}_n$ alternatives.

Let $\pi(D_n|j)$ be the probability under a certain sampling protocol of choosing subset D_n given that alternative j is chosen by individual n. For estimation purpose, D_n should include the chosen alternative and hence, is such that $\pi(D_n|j) = 0$ for any D_n that does not include alternative j.
The joint probability of choosing alternative j and constructing a particular set D_n is $\pi(j, D_n)$. Using Bayes theorem, we can write the last joint probability as follows

$$\pi(D_n, j) = \pi(D_n|j)P_n(j) = \pi(j|D_n)\pi(D_n) \tag{2.12}$$

Using equation (2.12) and the total probability theorem, we can write the conditional probability of choosing alternative j given that the set D_n is constructed, as follows

$$\begin{aligned} \pi(j|D_n) &= \frac{\pi(D_n|j) \cdot P_n(j)}{\pi(D_n)} \\ &= \frac{\pi(D_n|j) \cdot P_n(j)}{\sum_{i \in C_n} \pi(D_n|i) P_n(i)} \\ &= \frac{\pi(D_n|j) \cdot P_n(j)}{\sum_{i \in D_n} \pi(D_n|i) P_n(i)} \end{aligned} \tag{2.13}$$

where the simplification in the denominator in equation (2.13) is based on the fact that $\pi(D_n|j) = 0$ for $j \notin D_n$.
Plugging the logit formula into equation (2.13), we obtain by canceling and re-arranging terms

$$\pi(j|D_n) = \frac{\pi(D_n|j) \cdot e^{V_{nj}} / \sum_{i \in C_n} e^{V_{ni}}}{\sum_{i \in D_n} \pi(D_n|i) \cdot e^{V_{nj}} / \sum_{i \in C_n} e^{V_{ni}}} \tag{2.14}$$

$$= \frac{\pi(D_n|j) \cdot e^{V_{nj}}}{\sum_{i \in D_n} \pi(D_n|i) \cdot e^{V_{nj}}} \tag{2.15}$$

$$= \frac{e^{V_{nj} + \ln \pi(D_n|j)}}{\sum_{i \in D_n} e^{V_{ni} + \ln \pi(D_n|i)}} \tag{2.16}$$

The simplification in going from equation (2.14) to equation (2.15) is the cancellation of the denominators when dividing the probabilities of two alternatives in the logit model which is known as IIA property.
Estimation is based on maximizing the conditional log-likelihood function, CL, given by

$$CL(\beta) = \sum_{n=1}^{N} \ln \pi(j|D_n) = \sum_{n=1}^{N} \ln \frac{e^{V_{nj}(\beta, x_{nj}) + \ln \pi(D_n|j)}}{\sum_{i \in D_n} e^{V_{ni}(\beta, x_{ni}) + \ln \pi(D_n|i)}} \tag{2.17}$$

where j is the chosen alternatives by the decision maker n.

McFadden (1978, see reference [10]) proved that if the true model is Logit with full choice set C_n and if $\pi(D_n|j)$ satisfies the positive conditioning property that is

$$\pi(D_n|j) > 0 \qquad \text{for all} \quad j \in D_n$$

then maximizing the conditional likelihood function given in equation (2.17) provides consistent estimates of the model coefficients. Hence, if the model underlying the choice process is logit, estimation in the case of sampling of alternatives is adressed by adding a corrective constant to the systematic part of the utility of each alternative and leads to consistent parameters' estimation.
When $\pi(D_n|j)$ satisfies the uniform conditioning property, that is

$$\pi(D_n|j) = \pi(D_n|i) \qquad \text{for all} \quad i, j \in D_n \subset C_n$$

then equation (2.17) reduces to the standard likelihood function.

2.2.2 Estimation and Sampling of Alternatives in MEV Models

The logit model is saddeld with the IIA property resulting from the assumption that the error terms of the random utilities are uncorrelated. This assumption may be unrealistic in many choice situations.
This fact suggests extending McFadden's result on sampling and estimation for logit models to a more general class of models allowing a more complex error structure of the utilities. These models are know as Multivariate Extreme Value (MEV) models.
Guevara and Ben-Akiva (2010) studied the problem of sampling of alternatives for the MEV models and proved that under certain conditions, consistent estimates of the model's parameters can be obtained.

We first recall that the joint distribution of the error terms of the utilities in the case of MEV models has the following form

$$F(\epsilon_{n1}, \ldots, \epsilon_{nJ}) = e^{-G(e^{-\epsilon_{n1}}, \ldots, e^{-\epsilon_{nJ}}; \gamma)}$$

where γ is a set of parameters related to the distribution and G is a generating function which has certain properties so that F is a CDF function. This result is due to McFadden (1978) and is discussed in depth in reference [1].
The choice probability in the MEV models can be written in a logit form as follows

$$P_n(i) = \frac{e^{V_{ni} + \ln G_i(e^{V_{n1}}, e^{V_{n2}}, \ldots, e^{V_{nJ}}; \gamma)}}{\sum_{j \in C_n} e^{V_{nj} + \ln G_j(e^{V_{n1}}, e^{V_{n2}}, \ldots, e^{V_{nJ}}; \gamma)}}$$

where we used the notation

$$G_i(y_1, \ldots, y_n) = \frac{\partial G}{\partial y_i}(y_1, \ldots, y_n)$$

As in the logit model case, we can do the same steps to derive the conditional probability of choosing alternatives i given that the set D_n is sampled. This probability is given by equation (2.18) below

$$\pi(i|D_n) = \frac{e^{V_{ni}+\ln G_i(e^{V_{n1}},e^{V_{n2}},\ldots,e^{V_{nJ}};\gamma)+\ln \pi(D_n|i)}}{\sum_{j\in D_n} e^{V_{nj}+\underbrace{\ln G_j(e^{V_{n1}},e^{V_{n2}},\ldots,e^{V_{nJ}};\gamma)}_{\star}+\ln \pi(D_n|j)}} \tag{2.18}$$

We notice that even though the sum in the denominator of equation (2.18) is only over the sampled set D_n, the summand still contains a term, marked by the $(\star)$ in the equation above, which depends on the whole choice set. Hence, equation (2.18) is not practical since we can not anymore apply the theory on maximizing the conditional likelihood as in the logit case. However, consistent estimators of the model parameters can be recovered by estimating the quasi-log-likelihood function as shown in the following result developed by Guevara and Ben-Akiva (2010).

Theorem 2.2.1. Let C_n be a choice set of cardinality J_n and D_n a subset of cardinality $\tilde{J}_n$. If

1. The choice model is MEV
2. $\pi(D_n|j) > 0 \ \forall j \in D_n$ and $\pi(D_n|j) = 0 \ \forall j \notin D_n$
3. $\hat{G}_{in}(D_n)$ is a consistent estimator of $G_{in}(C_n) = G_i(e^{V_{n1}}, e^{V_{n2}}, \ldots e^{V_{nJ}}; \gamma)$ as $\tilde{J}_n$ grows, which depends only on the set D_n

Then, the maximization of the quasi-log-likelihood function

$$QLL(\beta) = \sum_{n=1}^{N} \ln \frac{e^{V_{ni}(\beta)+\ln \hat{G}_{in}(D_n)+\ln \pi(D_n|i)}}{\sum_{j\in D_n} e^{V_{nj}(\beta)+\ln \hat{G}_{jn}(D_n)+\ln \pi(D_n|j)}}$$

leads, under general regularity conditions, to consistent estimates of the model parameters $\beta^\star$, as N and $\tilde{J}_n$ rise at any rate.

Ben-Akiva and Guevara studied in particular the case of Nested Logit model with $\mathcal{M}$ nests where the function G is

$$G(y_1, \ldots, y_n, \gamma) = \sum_{m=1}^{\mathcal{M}} \Big(\sum_{i\in C_{mn}} y_i^{\mu_m} \Big)^{\frac{\mu}{\mu_m}-1}$$

where γ is the set of scales μ_m of the nests, and C_{mn} are the sets of alternatives which belong to the nest m. In this case, the term marked by a star in equation (2.18) becomes

$$\ln G_i(e^{V_{n1}}, e^{V_{n2}}, \ldots e^{V_{nJ}}; \gamma) = (\frac{\mu}{\mu_{m(i)}} - 1) \underbrace{\Big(\ln \sum_{j\in C_{m(i)n}} e^{\mu_{m(i)} V_{nj}} \Big)}_{\star\star} + \ln \mu + (\mu_{m(i)} - 1) V_{ni} \tag{2.19}$$

where $m(i)$ is the nest to which alternative i belongs. It has been shown that the expanded sum of the elements in $D_{m(i)n}$ as shown in equation (2.20) below, satisfies the conditions of theorem (2.2.1) above

$$\ln \sum_{j \in C_{m(i)n}} w_{nj} e^{\mu_m V_{nj}} \tag{2.20}$$

where $w_{nj} = 1/\mathbb{E}(1_{j \in D_{m(i)n}})$. The denominator is the probability of drawing alternative j when the protocol is sampling without replacement.
Hence, using the expanded sum given in equation (2.20) along with the theorem mentioned previously, consistent estimators of the parameters of the MEV models can be recovered.

Chapter 3

Literature Overview

Some studies have been done concerning the impact of sampling of alternatives in logit mixture models. These studies gave only empirical results on the effects of sampling of alternatives when IIA property does not hold and there are still no theoretical support for sampling of alternatives in this case.

McConnel and Tseng (2000, see [6]) studied the behavior of the random parameters logit (RPL) models under sampling of alternatives using the Chesapeake beach visit data set. This data set contains 10 alternatives (beach sites), 388 observations (trips) and 4 welfare attributes. They considered an RPL model with only one random coefficient distributed with respect to the log-normal distribution. They fitted the model both with the full choice set and with a reduced choice set after sampling 4 alternatives out of the 10 available alternatives. They repeated the fit experiment 15 times in order to avoid the sample variation and took the mean of the estimated parameters. They reached the conclusion that *sampling of alternatives does not systematically or substantially alter the RPL results. For all the welfare attributes, the mean estimates for the sampled RPL is no more than* 14% *different from the full set RPL ... Consequently, there is no evidence of systematic inconsistency in sampling of alternatives with RPL model.*

Nerella and Bhat (2004, see [7]) reached the same conclusion. They used Monte Carlo data drawn from a normal distribution. They considered five independent variables as attributes and the data set included 750 observations and 200 alternatives. The true model had two random coefficients which are independent and normally distributed. Their experiments showed that Logit Mixture model has a poor performance when the number of sampled alternatives is small but the model were able to recover the true parameters and achieve good numerical performances in term of bias, variance and mean square error when the number of sampled alternatives increases. They suggested to use at minimum one fourth of the full choice set.

However, Chen et al. (2005, see [8]) used Monte Carlo data to show that for Logit Mixture models which introduce correlation among the alternatives, sampling of alternative causes

a significant bias in the estimation of the model's parameters. The model they considered was slightly different from the models considered in the two previous studies. In fact, Chen et al. considered a Logit Mixture models with 3 nests whereas the other authors considered a Logit Mixture models with generic specification.
Their experiment had 1000 observations and each observation had 30 alternatives available. The attributes were generated from a normal distribution and were independent across alternatives and individuals. The model used was a Logit Mixture which had three nests. For each nest, a correlation coefficient was estimated. The stochastic part of the utility of each alternative had a normal part responsible for the correlation between alternatives belonging to the same nest and an error term generated from independent Type I Extreme value distribution with scale parameter of one.
They fitted several models by sampling each time $\tilde{J} = 5, 10, 15, \ldots, 25$ alternatives out of the 30 alternatives available for each observation. The results they obtained were that sampling of alternatives introduces a significant bias in the parameters' estimation.

Chapter 4

Models Specification

4.1 Main Problematic

Let N be the number of observations, C_n the choice set of cardinality J_n and D_n a subset of cardinality $\tilde{J}_n$.
When we fit a Logit Mixture model, the optimized log-likelihood function involves multidimensional integrals. The integrand is the logit probability multiplied by a mixing distribution. From the previous sections, the logit probability has a denominator which depends on the whole choice set C_n. When we fit the model using only the reduced choice set, this denominator will only depend on the sampled alternatives. Thus, optimizing the log-likelihood function with this new integrand without any correction might introduce a bias in the parameters' estimation.
The empirical approach we had in this project consists mainly in understanding whether there is a significant bias while fitting the model with the reduced choice set. In this case and following the same idea as for the MEV model, we try to correct the log-likelihood equation with an appropriate expanded sum of the denominator in order to obtain unbiased or at least consistent estimates.

4.2 General Description

In our studies, we consider a Logit Mixture model with two random coefficients. We characterize each alternative by two attributes and we adopt a generic specification. The way we form the data set, is to sample for each decision maker J independent draws from a chosen distribution. Each of these J draws correspond to the values of one attribute specifically for each of the J alternatives. Hence, we follow this process to generate two matrices of data, a and b which contain, for each decision maker, the values of the attributes per alternatives. More specifically, we have

$$a_{nj},\ b_{nj}\ \sim g(\cdot), \quad \text{for}\ \ j = 1, \ldots, J \text{ and } n = 1, \ldots, N$$

where $g(\cdot)$ is a chosen distribution to generate the data. In our study, we mainly used normal and uniform distributions.

Once we have the data, we can construct the utilities for each alternative j and decision maker n as follows

$$U_{nj} = \beta_n^a a_{nj} + \beta_n^b b_{nj} + \epsilon_{nj} \tag{4.1}$$

where the random coefficients β^a and β^b vary over decision makers in the population and are generated from a normal distribution as follows $\beta^a \sim \mathcal{N}(\mu_a, \sigma_a^2)$ and $\beta^b \sim \mathcal{N}(\mu_b, \sigma_b^2)$. We suppose that β_a and β_b are independent random variables and $f(\beta_a, \beta_b|\theta)$ is their joint distribution. It is the bivariate normal distribution with independent Gaussian marginals which true parameters are given by $\theta = (\mu_a, \mu_b, \sigma_a, \sigma_b)$. The random terms ϵ_{in} are iid Extreme Value type I with location parameter $\eta = 0$ and scale parameter $\mu = 1$.
Hence, we have a generic specification and for each decision maker n, the utility equations are

$$\begin{cases} U_{n1} &= \beta_n^a a_{n1} + \beta_n^b b_{n1} + \epsilon_{n1} \\ \vdots & \vdots \qquad\qquad \vdots \\ U_{nJ} &= \beta_n^a a_{nJ} + \beta_n^b b_{nJ} + \epsilon_{nJ} \end{cases}$$

After generating the random parameters and constructing the J utilities, random utility theory is used to specify the chosen alternatives for each decision maker. That is, given an individual n, we compute his/her utilities $U_{n1}, \ldots, U_{nJ}$ and the chosen alternative is the one which has the maximum utility, *i.e.* for an individual n

$$\text{if } i^* \text{ is such that } \; U_{ni^*} = \max_{j \in 1,\ldots,J} U_{nj} \quad \Longrightarrow \quad i^* \text{ is the chosen alternative}$$

Once we generate the data and obtain the choice variables, we fit three models in order to deal with the issue of sampling of alternatives for Logit Mixture models:

(M.1) : It is the Original Model. No sampling of alternatives is done. Only a Logit Mixture model is fitted and the parameters of the two random coefficients are estimated.

(M.2) : It is the Model with Sampling of alternatives. The parameters of the random coefficients are re-estimated using the reduced set of alternatives.

(M.3) : It is the Model with Sampling of alternatives and correction. As in the case of sampling of alternatives in the MEV models, we introduce some weights in order to correct the log likelihood equation so as to have unbiased estimates.

4.3 Models Specification and Estimation

4.3.1 The Original Model (M.1)

For the particular two random parameters Logit Mixture model we are studying, the choice probabilities are given by

$$P_{nj} = \int L_{nj}(\beta) f(\beta|\theta) d\beta \tag{4.2}$$

where

$$L_{ni}(\beta) = \frac{e^{\beta^a a_{ni} + \beta^b b_{ni}}}{\sum_{j \in C_n} e^{\beta^a a_{nj} + \beta^b b_{nj}}} \tag{4.3}$$

The log-likelihood function is given by

$$LL_{M.1}(\theta) = \sum_{n=1}^{N} \ln(P_{ni}) = \sum_{n=1}^{N} \ln\Big(\int \frac{e^{\beta^a a_{ni} + \beta^b b_{ni}}}{\sum_{j \in C_n} e^{\beta^a a_{nj} + \beta^b b_{nj}}} f(\beta|\theta)\ d\beta\ \Big) \tag{4.4}$$

where the alternative i is the chosen alternative by the decision maker n, β is the vector of random parameters (β^a, β^b) and $f(\cdot|\beta)$ is the bivariate normal distribution with independent marginals which parameters are given by $\theta = (\mu_a, \mu_b, \sigma_a, \sigma_b)$.

The Logit Mixture probabilities given in equation (4.2) take the form of a bidimensional integral over a mixing bivariate normal distribution. This integral does not have a closed form and has to be approximated by Monte Carlo simulation using random draws from the mixing distribution as explained Section (2.1.3). The integrand is computed at this sequence of random draws and the simulated Logit Mixture probabilities are given by the average of these integrand values as follows

$$\hat{P}_{ni} = \frac{1}{R} \sum_{r=1}^{R} L_{ni}(\beta_r) = \frac{1}{R} \sum_{r=1}^{R} \frac{e^{\beta_r^a a_{ni} + \beta_r^b b_{ni}}}{\sum_{j \in C_n} e^{\beta_r^a a_{nj} + \beta_r^b b_{nj}}} \tag{4.5}$$

The log-likelihood function $LL_{M.1}(\theta) = \sum_{n=1}^{N} \ln(P_{ni})$ is estimated by the simulated log-likelihood given by

$$SLL_{M.1}(\theta) = \sum_{n=1}^{N} \ln(\hat{P}_{ni}) = -N \ln(R) + \sum_{n=1}^{N} \ln\Big(\sum_{r=1}^{R} \underbrace{\frac{e^{\beta_r^a a_{ni} + \beta_r^b b_{ni}}}{\sum_{j \in C_n} e^{\beta_r^a a_{nj} + \beta_r^b b_{nj}}}}_{\star}\Big) \tag{4.6}$$

We maximize the simulated log-likelihood equation over θ in order to recover the true values of the parameters $(\mu_a, \mu_b, \sigma_a, \sigma_b)$. Models estimation is done using the BIOGEME software.
The related BIOGEME code for this model in given in the appendix, section (8.2).

4.3.2 The Model with Sampling of Alternatives (M.2)

Suppose now we use only a subset of alternatives $D_n \subset C_n$ which has $\tilde{J}_n$ alternatives. We use a sampling without replacement protocol to choose for each decision maker n, a subset of alternatives D_n which includes the chosen alternative. First, the chosen alternative is drawn. Afterwards, the non chosen alternatives are randomly sampled up to complete a total of $\tilde{J}_n$ alternatives.
Since we are only considering a subset of alternatives, equation (4.3) is substituted by

$$L_{ni}(\beta) = \frac{e^{\beta^a a_{ni} + \beta^b b_{ni}}}{\sum_{j \in D_n} e^{\beta^a a_{nj} + \beta^b b_{nj}}} \tag{4.7}$$

The maximized log-likelihood function becomes

$$LL_{M.2}(\theta) = \sum_{n=1}^{N} \ln(P_{ni}) = \sum_{n=1}^{N} \ln\Big(\int \frac{e^{\beta^a a_{ni} + \beta^b b_{ni}}}{\sum_{j \in D_n} e^{\beta^a a_{nj} + \beta^b b_{nj}}} f(\beta|\theta)\ d\beta\ \Big) \tag{4.8}$$

where, as before, the alternative i is the chosen alternative by the decision maker n, β is the vector of random parameters (β^a, β^b) and $f(\cdot|\beta)$ is the bivariate normal distribution with independent marginals which parameters are given by $\theta = (\mu_a, \mu_b, \sigma_a, \sigma_b)$.
Optimization in done using the simulated version of equation (4.8) given as follows

$$SLL_{M.2}(\theta) = \sum_{n=1}^{N} \ln(\hat{P}_{ni}) = -N \ln(R) + \sum_{n=1}^{N} \ln\Big(\sum_{r=1}^{R} \frac{e^{\beta_r^a a_{ni} + \beta_r^b b_{ni}}}{\sum_{j \in D_n} e^{\beta_r^a a_{nj} + \beta_r^b b_{nj}}} \Big) \tag{4.9}$$

where for $r \in 1, \ldots, R$, $\beta_r^a \sim \mathcal{N}(\mu_a, \sigma_a)$ and $\beta_r^b \sim \mathcal{N}(\mu_b, \sigma_b)$, R being the total number of draws.
The simulated log-likelihood equation is maximized over θ in order to recover the true values of the parameters $(\mu_a, \mu_b, \sigma_a, \sigma_b)$.
As for the previous model, estimation is done using the BIOGEME software. The related code for this model is given in the appendix, section (8.2).

4.3.3 The Model with Sampling and Corrections (M.3)

We might expect that fitting a Logit Mixture model with only a subset of the whole choice set might introduce a significant bias in the parameters' estimation. Hence, in order to recover the true values of the parameters, we suggest to correct the log-likelihood function by introducing appropriate weights as done for the MEV models.
In more details, under sampling of alternatives for Logit Mixture models, the only part of the likelihood equation given by (4.6) that has to be estimated is the term marked by $(\star)$ which involves the whole set of alternatives, C_n. Hence, we try to find an appropriate estimator of $(\star)$ so that to recover the true values of the parameters $\theta = ((\mu_a, \mu_b, \sigma_a, \sigma_b))$.

As in the case of MEV models, we suggest to expand the sum over all alternatives in C_n by the sum over the alternatives in D_n where the summand is multiplied by a certain weight:

$$\sum_{j\in D_n} w_{nj} e^{\beta_r^a a_{nj}+\beta_r^b b_{nj}} \quad \text{as a consistent and unbiased estimation of} \quad \sum_{j\in C_n} e^{\beta_r^a a_{nj}+\beta_r^b b_{nj}}$$

where the weights w_{nj} depend on the sampling protocol. When we sample without replacement, these expansion weights are given by the same formula as in equation (2.20). That is

$$w_{nj} = \frac{1}{\mathbb{P}(\text{sampling alt. } j)}$$

Using Bayes formula and conditioning on the event of whether alternative j is the chosen alternative or not, the probability of choosing alternative j is equal to

$$\begin{aligned}\mathbb{P}(\text{sampling alt. } j) &= \mathbb{P}(\text{sampling alt. } j | j \text{ is the chosen alt.}) \cdot \mathbb{P}(j \text{ is the chosen alt.}) \\ &+ \mathbb{P}(\text{sampling alt. } j | j \text{ is not the chosen alt.}) \cdot \mathbb{P}(j \text{ is not the chosen alt.}) \\ &= 1 \cdot p_{nj}^* + \frac{\tilde{J}_n - 1}{J_n - 1} \cdot (1 - p_{nj}^*)\end{aligned}$$

where p_{nj}^* are the true probabilities. Hence, the expansion weights become

$$w_{nj} = \frac{1}{1 \cdot p_{nj}^* + \frac{\tilde{J}_n - 1}{J_n - 1} \cdot (1 - p_{nj}^*)}$$

The true probabilities p_{nj}^* can be viewed from two perspectives

1. The probabilities p_{nj}^* are equal to the probabilities generated by BIOSIM after fitting the model with the full choice set, M.1.

2. When we generate the synthetic data and form the systematic part of the utility of each decision maker, we draw β from a chosen mixing distribution. So, we know the values of the random parameters and hence, the Logit Mixture model turns out to be a logit model. For this reason, we can consider that the true probabilities p_{nj}^* are given by the logit formula as given in equation (2.2).

Substituting the sum over the whole choice set in equation (4.4) by the expanded sum over the reduced choice set, the log-likelihood equation becomes

$$LL_{M.3}(\theta) = \sum_{n=1}^{N} \ln(P_{ni}) = \sum_{n=1}^{N} \ln\Big(\int \frac{e^{\beta^a a_{ni}+\beta^b b_{ni}}}{\sum_{j\in D_n} w_{nj} \cdot e^{\beta^a a_{nj}+\beta^b b_{nj}}} f(\beta|\theta)\ d\beta\ \Big) \tag{4.10}$$

To optimize the previous equation using BIOGEME, we need first to specify the utility equations which correspond to this model. For this purpose, let us first write the integral inside equation (4.10) in this following form

$$\int \frac{e^{\beta^a a_{ni}+\beta^b b_{ni}+\ln}}{\sum_{j\in D_n} w_{nj} \cdot e^{\beta^a a_{nj}+\beta^b b_{nj}}} f(\beta|\theta)\ d\beta = \frac{1}{w_{ni}} \int \frac{e^{\beta^a a_{ni}+\beta^b b_{ni}+\ln w_{ni}}}{\sum_{j\in D_n} e^{\beta^a a_{nj}+\beta^b b_{nj}+\ln w_{nj}}} f(\beta|\theta)\ d\beta$$

Plugging the previous formula into equation (4.10), we obtain

$$LL_{M.3}(\theta) = \sum_{n=1}^{N} \ln(P_{ni}) = -\sum_{n=1}^{N} \ln w_{ni} + \sum_{n=1}^{N} \ln\Big(\int \frac{e^{\beta^a a_{ni} + \beta^b b_{ni} + \ln w_{ni}}}{\sum_{j \in D_n} e^{\beta^a a_{nj} + \beta^b b_{nj} + \ln w_{nj}}} f(\beta|\theta)\ d\beta\ \Big) \quad (4.11)$$

Since the term '$-\sum_{n=1}^{N} \ln w_{ni}$' in equation (4.11) does not depend on the θs, maximizing the log-likelihood over the model's parameters given by θ is equivalent to maximizing the following log-likelihoood

$$\tilde{LL}_{M.3}(\theta) = \sum_{n=1}^{N} \ln\Big(\int \frac{e^{\beta^a a_{ni} + \beta^b b_{ni} + \ln w_{ni}}}{\sum_{j \in D_n} e^{\beta^a a_{nj} + \beta^b b_{nj} + \ln w_{nj}}} f(\beta|\theta)\ d\beta\ \Big) \quad (4.12)$$

For each decision makers n, the utility equations of the model which log-likelihood equation is given by the previous equation are given by the following generic specification

$$\begin{cases} U_{n1} &= \beta_n^a a_{n1} + \beta_n^b b_{n1} + \alpha \ln w_{n1} + \epsilon_{n1} \\ \vdots & \vdots \qquad\qquad \vdots \qquad\quad \vdots \\ U_{nJ} &= \beta_n^a a_{nJ} + \beta_n^b b_{nJ} + \alpha \ln w_{nJ} + \epsilon_{nJ} \end{cases}$$

where α is fixed to 1. The only coefficients to be estimated are the parameters of the mixing distribution.
As for the previous two models, BIOGEME software is used for estimation and the related code for this model is given in the appendix, section (8.2).

Chapter 5

Simulations and Results

In this chapter, we present the fits' results of the three models, M.1, M.2 and M.3, introduced in the last Chapter. We performed several Monte Carlo experiments using different set of synthetic data drawn from the uniform and normal distribution. The choice of the model parameters particularly, the number of observations and the number of draws needed, along with the results of the models' fits are explained and presented in the next sections of this chapter.

5.1 Choice of the Models' Parameters

Before fitting any model, we had to figure out the appropriate parameters, in particular the number of observations while generating the data, N, and the number of draws for the simulations, R. Since any consistency results are obtained with a large number of observations, we first started with a sample of $N = 1000$ decision makers. As mentioned previously in section (2.1.3), the number of draws R, needs to be large enough to ensure reasonably low simulation error. We first set $R = 1000$ and fit with BIOGEME a Logit Mixture model using the whole choice set. The results of this first fit were not satisfactory since we obtained parameters' estimates which were significantly different from their true values. We increased the number of observations to $N = 10'000$ and fitted the same model using BIOGEME. The bias was not significant, however the simulation took a long time to generate the results. With a larger number of observations, we ran out of computer's physical memory. For this reason a trade off between the feasibility of the simulation (time and memory needed) and how large should the sample size be to give unbiased estimations, is very important. After trying different combinations of N and R, we found out that setting $N = 2000$ and $R = 1000$ gives satisfactory results. Additionally, the final log-likelihood and the values of the parameters seemed to be stable enough and increasing the number of draws or the number of observations did not improve significantly their values.
Once we could estimate the first model, M.1, with no sampling of alternatives and have satisfactory estimates, we fitted the two other models M.2 and M.3 using the same data set,

the same number of observations, $N = 2000$, and the same number of simulation draws, $R = 1000$. The next sections present the results of all the experiments we did.

5.2 Results: Uniform Data

In this section, the attributes of the model are Monte Carlo data drawn from the uniform distribution. The specifications of the experiment are the following

5.2.1 Experiment Specifications

- Number of observations: $N = 2000$
- Number of alternatives: $J = 100$
- Number of Draws: $R = 1000$
- Utility: For each observation n and each alternative j

$$U_{nj} = \beta_n^a a_{nj} + \beta_n^b b_{nj} + \epsilon_{nj}$$

  where

  - The data are draws from the UNIFORM distribution *i.e.* for each observation n and each alternative j

$$a_{nj},\ b_{nj} \sim Uniform[-5,\ 5]$$

  - The random coefficients β_a and β_b are distributed over the population such as

$$\beta^a \sim \mathcal{N}(\mu_a, \sigma_a^2) \qquad \text{and} \qquad \beta^b \sim \mathcal{N}(\mu_b, \sigma_b^2)$$

	μ_a	μ_b	σ_a	σ_b
True Values	1	2	1	2

Table 5.1: True Parameters

  - The random term ϵ_{nj} are iid Extreme Value distributed with location parameter $\eta = 0$ and scale parameter $\mu = 1$

- Number of Sampled alternatives: depends on the experiment: $\tilde{J} = 5, 10, 20, \ldots 90$
- The sampling protocol: Let D denotes the set of sampled alternatives, then
  - The chosen alternative for each individual n is included in the set D
  - The non chosen-alternatives are randomly sampled without replacement up to complete a total of $\tilde{J}$ alternatives.

We perform two different kinds of experiments. In the first experiment, the number of sampled alternatives is $\tilde{J} = 5$ whereas in the second experiment $\tilde{J} = 50$.

5.2.2 Number of sampled alternatives: $\tilde{J} = 5$

Experiment Description and Results

We fit the three models introduced in chapter (4), using BIOGEME

1. Model M.1: Using the whole choice set which contains a total number of alternatives equal to $J = 100$
2. Model M.2: Using a reduced choice set. Then number of Sampled alternatives is $\tilde{J} = 5$
3. Model M.3: It is the same model as Model M.2 but with correction of the simulated log-likelihood equation as specified in section (4.3.3).

We repeat this experiment 10 times with different sets of data drawn from the UNIFORM distribution. This is done to test the stability of the parameters' estimates with respect to the Monte Carlo data. The estimation results are given in tables (5.2), (5.3) and (5.4) below and each table contains the final log-likelihood and the values of the parameters of the random coefficients along with their standard deviation between parenthesis as estimated by BIOGEME.
We perform a t-test with a significance level of 5% to test the significance of the parameter's estimates against their true values. The star mark ($\star$) denotes the parameters which are significantly different form the true values. We recall that the true values are given in table (5.1) above.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-5679.556	1.03 (0.0434)	1.96 (0.0802)	1.03 (0.0469)	2.04 (0.0838)
Experiment.2	-5629.86	0.99 (0.0412)	2.06 (0.0833)	0.972 (0.0442)	2.08 (0.0843)
Experiment.3	-5693.032	0.959 (0.0399)	1.98 (0.0785)	0.954 (0.0431)	1.89 (0.0774)
Experiment.4	-5754.125	0.949 (0.0402)	1.96 (0.0781)	0.997 (0.0439)	1.92 (0.0781)
Experiment.5	-5731.268	0.997 (0.042)	1.94 (0.0783)	1.01 (0.0449)	1.95 (0.0793)
Experiment.6	-5714.539	1.07 (0.0441)	1.83 (0.0753) *	1.05 (0.0466)	1.9 (0.0783)
Experiment.7	-5614.255	1.01 (0.0422)	2.06 (0.0832)	1 (0.0453)	2.05 (0.0837)
Experiment.8	-5597.678	1.04 (0.0431)	2.08 (0.0835)	1.03 (0.0456)	2.05 (0.0843)
Experiment.9	-5677.779	0.988 (0.0413)	1.99 (0.0791)	1.02 (0.0451)	1.96 (0.0807)
Experiment.10	-5664.415	0.973 (0.0395)	1.99 (0.0795)	0.931 (0.0416)	1.9 (0.0776)

Table 5.2: Model M.1: Original Model.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-1649.533	1.16 (0.129)	2.29 (0.241)	1.07 (0.136)	2.24 (0.246)
Experiment.2	-1645.629	0.964 (0.103)	2.23 (0.223)	0.828 (0.109)	2.2 (0.232)
Experiment.3	-1636.417	0.966 (0.108)	2.11 (0.211)	0.915 (0.124)	1.93 (0.206)
Experiment.4	-1643.349	1.2 (0.129)	2.3 (0.224)	1.12 (0.138)	2.17 (0.223)
Experiment.5	-1670.43	0.885 (0.0952)	2.19 (0.213)	0.754 (0.104) *	2.19 (0.218)
Experiment.6	-1665.679	1.1 (0.115)	1.95 (0.187)	0.952 (0.122)	1.94 (0.195)
Experiment.7	-1641.828	1.05 (0.116)	2.15 (0.22)	0.97 (0.129)	2.04 (0.221)
Experiment.8	-1598.718	1.12 (0.117)	2.25 (0.223)	0.953 (0.12)	2.08 (0.214)
Experiment.9	-1657.823	1.05 (0.116)	2.12 (0.215)	0.981 (0.126)	1.98 (0.215)
Experiment.10	-1585.316	1.31 (0.148) *	2.59 (0.284) *	1.16 (0.149)	2.42 (0.278)

Table 5.3: Model M.2: Model with Sampling of alternatives.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-1759.712	4.92 (0.601) *	9.91 (1.11) *	4.93 (0.585) *	10.2 (1.17) *
Experiment.2	-1765.971	4.21 (0.452) *	9.84 (1.06) *	4.15 (0.472) *	10 (1.11) *
Experiment.3	-1755.54	4.38 (0.504) *	9.16 (1) *	4.61 (0.539) *	8.65 (0.94) *
Experiment.4	-1762.618	4.81 (0.528) *	8.65 (0.892) *	4.92 (0.544) *	8.5 (0.932) *
Experiment.5	-1802.979	3.54 (0.342) *	8.55 (0.809) *	3.51 (0.352) *	8.58 (0.812) *
Experiment.6	-1795.133	4.45 (0.475) *	7.73 (0.792) *	4.45 (0.545) *	7.84 (0.793) *
Experiment.7	-1757.585	4.7 (0.596) *	9.53 (1.11) *	5.13 (0.708) *	9.58 (1.16) *
Experiment.8	-1718.685	4.98 (0.566) *	9.68 (1.05) *	4.67 (0.535) *	8.93 (0.961) *
Experiment.9	-1774.997	4.77 (0.56) *	9.04 (0.954) *	4.99 (0.613) *	8.95 (0.993) *
Experiment.10	-1685.061	5.09 (0.573) *	9.97 (1.1) *	4.84 (0.571) *	10 (1.14) *

Table 5.4: Model M.3: Model with the corrections.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

Tables (5.2) and (5.3) report the estimation results of model M.1 and M.2. We notice that only few parameters are significantly different from their true values. However, table (5.4) shows that all the parameters are significantly different from their true values. This result suggests that fitting a logit mixture model with a reduced choice set does not affect, in general, the parameters' estimation. However, correcting the log-likelihood equation and estimating the model introduces a significant bias in the parameters' estimation.

5.2.3 Number of sampled alternatives: $\tilde{J} = 50$

Experiment Description and Results

We repeat the same Experiments as before. The only difference is that Model M.2 and model M.3 are estimated after sampling $\tilde{J} = 50$ alternatives out of a total of $J = 100$. The results of the parameters' estimation are given in tables (5.5), (5.6) and (5.7) below

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-5774.604	0.972 (0.0404)	1.92 (0.0798)	0.964 (0.043)	2.04 (0.0825)
Experiment.2	-5760.063	0.977 (0.0394)	1.89 (0.0774)	0.934 (0.0416)	1.93 (0.0781)
Experiment.3	-5700.025	0.95 (0.0399)	2 (0.0797)	0.949 (0.0432)	1.98 (0.0816)
Experiment.4	-5608.663	1.01 (0.0427)	2.12 (0.0863)	1.03 (0.0464)	2.16 (0.0883)
Experiment.5	-5625.785	1.07 (0.0439)	1.97 (0.0791)	1.04 (0.0461)	1.99 (0.0804)
Experiment.6	-5656.908	0.981 (0.0407)	2.05 (0.0835)	0.982 (0.044)	2.06 (0.08383)
Experiment.7	-5635.529	0.973 (0.0402)	2.05 (0.0817)	0.968 (0.0433)	1.97 (0.0796)
Experiment.8	-5684.985	0.966 (0.0394)	2 (0.0794)	0.938 (0.0421)	1.96 (0.0793)
Experiment.9	-5639.938	0.952 (0.0401)	2.05 (0.0822)	0.967 (0.0434)	2.03 (0.0834)
Experiment.10	-5650.369	0.973 (0.0411)	2.05 (0.0821)	0.995 (0.045)	2.03 (0.0826)

Table 5.5: Model M.1: Original Model.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-4625.297	0.962 (0.044)	2.01 (0.0913)	0.946 (0.0477)	2.13 (0.0953)
Experiment.2	-4615.061	0.988 (0.0438)	1.89 (0.0835)	0.928 (0.0462)	1.92 (0.0849)
Experiment.3	-4575.647	0.918 (0.0423)	1.98 (0.0259)	0.902 (0.0466) *	1.96 (0.0879)
Experiment.4	-4472.744	1.02 (0.047)	2.2 (0.0983) *	1.02 (0.0513)	2.22 (0.101) *
Experiment.5	-4477.757	1.08 (0.049)	2.06 (0.0913)	1.03 (0.0517)	2.09 (0.0937)
Experiment.6	-4504.92	0.992 (0.0452)	2.12 (0.0944)	0.973 (0.049)	2.12 (0.0948)
Experiment.7	-4498.136	0.998 (0.045)	2.07 (0.0902)	0.993 (0.0496)	1.99 (0.0882)
Experiment.8	-4537.088	0.976 (0.0439)	2.02 (0.087)	0.937 (0.0472)	1.97 (0.0867)
Experiment.9	-4497.031	0.954 (0.0443)	2.11 (0.0927)	0.959 (0.0492)	2.09 (0.0945)
Experiment.10	-4506.42	0.991 (0.0461)	2.11 (0.0927)	1 (0.051)	2.09 (0.0938)

Table 5.6: Model M.2: Model with sampling of alternatives,
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-4707.427	1.12 (0.0495) *	3.32 (0.103) *	1.12 (0.0527) *	2.47 (0.108) *
Experiment.2	-4705.954	1.15 (0.0489) *	2.19 (0.0945) *	1.09 (0.0509)	2.23 (0.0955) *
Experiment.3	-4662.96	1.07 (0.0477)	2.28 (0.0971) *	1.07 (0.0517)	2.27 (0.0997) *
Experiment.4	-4565.032	1.19 (0.0532) *	2.56 (0.112) *	1.22 (0.0575) *	2.6 (0.115) *
Experiment.5	-4525.386	1.25 (0.0548) *	2.38 (0.103) *	1.21 (0.0574) *	2.42 (0.106) *
Experiment.6	-4592.631	1.16 (0.0509) *	2.46 (0.107) *	1.15 (0.0546) *	2.47 (0.107) *
Experiment.7	-4579.178	1.16 (0.0511) *	2.38 (0.102) *	1.17 (0.0547) *	2.3 (0.099) *
Experiment.8	-4627.18	1.14 (0.0495) *	2.34 (0.0983) *	1.12 (0.0526) *	2.29 (0.0979) *
Experiment.9	-4581.984	1.12 (0.0501) *	2.44 (0.105) *	1.14 (0.0548) *	2.43 (0.107) *
Experiment.10	-4587.658	1.16 (0.0519) *	2.44 (0.105) *	1.18 (0.0566) *	2.42 (0.105) *

Table 5.7: Model M.3: Model with the correction.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

The same conclusion as in the previous section is reached. Sampling of alternatives does not seem to significantly affect the parameters' estimation in general. However, using the same weights as in MEV models to correct the log-likelihood equation seems to lead to a biased models' estimates.

5.3 Results: Normal Data

In this section, the attributes are generated from a normal distribution. The experiment's specifications and the results of the fit are given as follows

5.3.1 Experiment Specifications

- Number of observations: $N = 2000$
- Number of alternatives: $J = 100$
- Number of Draws: $R = 1000$
- Utility: For each observation n and each alternative j

$$U_{nj} = \beta_n^a a_{nj} + \beta_n^b b_{nj} + \epsilon_{nj}$$

where

  - The data are draws from the NORMAL distribution *i.e.* for each observation n and each alternative j

$$a_{nj} \sim \mathcal{N}(0,\ 1) \qquad \text{and} \qquad b_{nj} \sim \mathcal{N}(2,\ 3^2) \tag{5.1}$$

  - The random coefficients β_a and β_b are distributed over the population such as

$$\beta^a \sim \mathcal{N}(\mu_a, \sigma_a^2) \qquad \text{and} \qquad \beta^b \sim \mathcal{N}(\mu_b, \sigma_b^2)$$

	μ_a	μ_b	σ_a	σ_b
True Values	1	2	1	2

Table 5.8: True Parameters

  - The random term ϵ_{nj} are iid Extreme Value distributed with location parameter $\eta = 0$ and scale parameter $\mu = 1$

- Number of Sampled alternatives: $\tilde{J} = 50$
- The sampling protocol: Let D denotes the set of sampled alternatives, then
  - The chosen alternative for each individual n is included in the set D
  - The non chosen-alternatives are randomly sampled without replacement up to complete a total of $\tilde{J}$ alternatives.

5.3.2 Experiment Description and Results

As we have done previously using the uniform Monte Carlo data, we also fit the same three models using BIOGEME.

1. Model M.1: Using the whole choice set. Total number of alternatives $J = 100$
2. Model M.2: Using a reduced choice set. Number of Sampled alternatives $\tilde{J} = 50$
3. Model M.3: Same as Model M.2 but with correction of the simulated log-likelihood equation.

We repeat this experiment 10 times with different sets of data drawn from the NORMAL distributions as shown in equation (5.3). This is done to test the stability of the parameters estimation if we use data drawn from the Normal distribution instead from the Uniform distribution. The fits' results are given in tables (5.9), (5.10) and (5.11) below. As before, each table contains the final log-likelihood and the values of the parameters of the random coefficients along with their standard deviation between parenthesis as estimated by BIOGEME.
We perform a t-test for each estimate against its true value. As before, the significance level of the t-test is 5%.The star mark ($\star$) denotes the parameters which are significantly different form the true values. We recall that their true values are given in table (5.8) above.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-4150.446	0.996 (0.0509)	2.03 (0.0829)	0.95 (0.0737)	2.11 (0.0858)
Experiment.2	-4173.818	0.963 (0.0514)	1.94 (0.0757)	1.03 (0.0718)	1.89 (0.076)
Experiment.3	-4357	0.929 (0.0508)	1.84 (0.0743) *	1.03 (0.072)	1.91 (0.076)
Experiment.4	-4097.562	1.08 (0.0505)	2.11 (0.0859)	0.887 (0.0743)	2.12 (0.0876)
Experiment.5	-4071.471	1.1 (0.0545)	2.09 (0.083)	1.08 (0.0738)	1.95 (0.0795)
Experiment.6	-4249.702	0.971 (0.0543)	2.02 (0.0828)	1.15 (0.074) *	2.06 (0.0847)
Experiment.7	-4158.873	1.01 (0.0538)	2.02 (0.081)	1.08 (0.0735)	1.96 (0.081)
Experiment.8	-4185.165	1.08 (0.0528)	1.97 (0.0792)	1.01 (0.0737)	1.97 (0.0805)
Experiment.9	-4148.362	1.05 (0.0544)	2.09 (0.0841)	1.08 (0.0744)	1.95 (0.0805)
Experiment.10	-4009.956	0.979 (0.0532)	2.12 (0.088)	1.02 (0.074)	2.16 (0.0881)

Table 5.9: Model M.1: Original Model.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-3319.217	1.01 (0.0586)	2.07 (0.0951)	0.956 (0.0862)	2.16 (0.1)
Experiment.2	-3309.158	0.954 (0.0565)	1.99 (0.0873)	0.977 (0.0852)	1.95 (0.089)
Experiment.3	-3489.375	0.994 (0.0577)	1.91 (0.0867)	1.04 (0.0833)	1.97 (0.0882)
Experiment.4	-3205.621	1.13 (0.0589) *	2.25 (0.104) *	0.874 (0.0879)	2.26 (0.107) *
Experiment.5	-3212.925	1.13 (0.0618) *	2.15 (0.0963)	1.03 (0.0866)	2 (0.0929)
Experiment.6	-3373.897	1 (0.0599)	2.12 (0.0976)	1.06 (0.0862)	2.15 (0.0991)
Experiment.7	-3325.798	1.03 (0.0608)	2.07 (0.0935)	1.1 (0.0844)	2.01 (0.0936)
Experiment.8	-3352.099	1.1 (0.0589)	2.02 (0.0921)	0.941 (0.0845)	2.02 (0.0927)
Experiment.9	-3275.367	1.08 (0.0607)	2.13 (0.0959)	1.01 (0.0859)	1.97 (0.0914)
Experiment.10	-3234.035	0.98 (0.0589)	2.21 (0.103) *	1 (0.0832)	2.26 (0.103) *

Table 5.10: Model M.2: Model with sampling of alternatives.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-3395.341	1.16 (0.0661) *	2.47 (0.112) *	1.18 (0.112)	2.6 (0.118) *
Experiment.2	-3382.938	1.09 (0.0634)	2.36 (0.102) *	1.18 (0.0907) *	2.33 (0.105) *
Experiment.3	-3570.511	1.08 (0.0644)	2.28 (0.102) *	1.24 (0.0881) *	2.36 (0.104) *
Experiment.4	-3280.454	1.29 (0.0665) *	2.69 (0.123) *	1.09 (0.091)	2.72 (0.127) *
Experiment.5	-3285.414	1.3 (0.0695) *	2.57 (0.114) *	1.24 (0.0929) *	2.41 (0.109) *
Experiment.6	-3448.358	1.15 (0.0674) *	2.53 (0.115) *	1.27 (0.0926) *	2.57 (0.116) *
Experiment.7	-3402.493	1.18 (0.0683) *	2.49 (0.11) *	1.32 (0.0912) *	2.43 (0.111) *
Experiment.8	-3425.204	1.25 (0.0658) *	2.42 (0.108) *	1.14 (0.0885)	2.42 (0.11) *
Experiment.9	-3354.966	1.23 (0.0684) *	2.56 (0.114) *	1.22 (0.0919) *	2.37 (0.108) *
Experiment.10	-3303.729	1.12 (0.0668)	2.64 (0.121) *	1.2 (0.0885) *	2.67 (0.121) *

Table 5.11: Model M.3: Model with the correction.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

We notice that fitting a Logit Mixture model with a reduced choice set leads in the majority of the times to parameters which are not significantly different from their true values. Indeed, only 6 out of 40 parameters have a significant bias. However, as in the previous case, correcting the log-likelihood leads in the large majority of the cases, to parameters which are significantly different from their true values.

5.4 Results: Data Generated from two different Normal Distributions

In this experiment, the attributes a and b are generated each from two different normal distributions. Indeed, we separated the decision makers into two sets. The first and the second set of the decision makers have attributes a_{nj} and b_{nj} which are generated from two different normal distributions with different parameters. By generating the data in this manner, we want to explore whether introducing more heterogeneity among the population affects the results of the previous two sections where all the data were generated from the same distribution. The experiment specification and the results are given in the next two sections.

5.4.1 Experiment Specifications

- Number of observations: $N = 2000$
- Number of alternatives: $J = 100$
- Number of Draws: $R = 1000$
- Utility: For each observation n and each alternative j

$$U_{nj} = \beta_n^a a_{nj} + \beta_n^b b_{nj} + \epsilon_{nj}$$

where

- The data are draws from two different NORMAL distributions *i.e.* given an alternative j and for the first $\frac{N}{2} = 1000$ observations, we have

$$a_{nj} \sim \mathcal{N}(0,\ 1) \qquad \text{and} \qquad b_{nj} \sim \mathcal{N}(-1,\ 0.5^2) \tag{5.2}$$

and for the second $\frac{N}{2} = 1000$ observations, we have

$$a_{nj} \sim \mathcal{N}(5,\ 2^2) \qquad \text{and} \qquad b_{nj} \sim \mathcal{N}(0,\ 10^2) \tag{5.3}$$

- The random coefficients β_a and β_b are distributed over the population such as

$$\beta^a \sim \mathcal{N}(\mu_a, \sigma_a^2) \qquad \text{and} \qquad \beta^b \sim \mathcal{N}(\mu_b, \sigma_b^2)$$

	μ_a	μ_b	σ_a	σ_b
True Values	1	2	1	2

Table 5.12: True Parameters

  - The random term ϵ_{nj} are iid Extreme Value distributed with location parameter $\eta = 0$ and scale parameter $\mu = 1$

- Number of Sampled alternatives: $\tilde{J} \in \{5, 50\}$
- The sampling protocol: Let D denotes the set of sampled alternatives, then
  - The chosen alternative for each individual n is included in the set D
  - The non chosen-alternatives are randomly sampled without replacement up to complete a total of $\tilde{J}$ alternatives.

5.4.2 Experiment Description and Results

As we did before, the three models introduced in chapter (4) are fitted using BIOGEME. The estimation results are reported in table (5.13) below

Parameters	J	$\tilde{J}$	μ_a	μ_b	σ_a	σ_b
True Values	100	-	1	2	1	2
Model M.1	100	100	1.06 (0.0453)	1.96 (0.083)	1.03 (0.0551)	2.01 (0.0895)
Model M.2	100	5	1.3 (0.106) *	2.14 (0.179)	1.17 (0.148)	2.09 (0.194)
Model M.3	100	5	2.76 (0.214) *	5 (0.389) *	3.01 (0.269) *	5.19 (0.415) *
Model M.2	100	50	1.10 (0.05) *	1.99 (0.0909)	1.03 (0.063)	2.09 (0.107)
Model M.3	100	50	1.81 (0.0752) *	3.41 (0.133) *	1.85 (0.0833) *	3.8 (0.15) *

Table 5.13: Estimation results for models M.1, M.2 and M.3 where the data are drawn from two different normal distributions.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

We recall that model M.1 is a logit mixture model fitted with the whole choice set and M.2 is the same model fitted with a reduced choice set. Model M.3 is also fitted with a reduced choice set and a correction of the log-likelihood equation. We recall also that the star mark ($\star$) denotes the parameters which are significantly different form the true values. We performed a t-test with a significance level of 5% to test the significance of the parameters' estimates against their true values.

Table (5.13), shows the estimation's results of the two experiments done with those kind of data. In the first experiment, the number of sampled alternatives, $\tilde{J}$, is equal to 5 and in the second experiment $\tilde{J}$ is equal to 50 alternatives. In both experiments and for model M.2, only one parameter, μ_a, is significantly different from its true value. However, for model M.3, all parameters are significantly different from their true values. The same conclusion as in the previous sections is reached.

5.5 Varying the number of sampled alternatives

The experiment consists in fitting the model with the same set of uniform Monte Carlo data several times by varying the number of sampled alternatives each time, $\tilde{J} = 10, \ldots, 100$. The experiment's specification is the same as in section (5.2), in particular the attributes are drawn from a uniform distribution.
Through this experiment, we would like to check the effect of sampling different numbers of alternatives on the parameters' estimation. For each given number of sampled alternatives, we fit two models. The first one without correction of the simulated log-likelihood and the second one with correction. The experiment where $\tilde{J} = 100$ corresponds to the model fitted with the whole choice set *i.e.* without any sampling of alternatives.
The estimation results are given in tables (5.14) and (5.15) respectively. The first column of each table contains the number of sampled alternatives. The other columns contain the final log-likelihood and the values of the parameters of the random coefficients along with their standard deviation between parenthesis as estimated by BIOGEME.
As in the previous sections, the star mark ($\star$) denotes the parameters which are significantly different form their true values. We recall that the true values are given in table (5.1) above and that we performed a t-test with a 5% significance level.
From table (5.14), we notice that, while varying the number of sampled alternatives, the model estimation leads in general to parameters which are not significantly different from their true values. Only few parameters are significantly biased. We notice as well that in general, the parameters' estimates get closer to their true value and their standard deviation decreases when the number of sampled alternatives increases.
However, table (5.15) shows that fitting a logit mixture model with a reduced choice set and correction of the log-likelihood introduces a significant bias in the parameters' estimation. However, this bias decreases through the experiment when the number of sampled alternatives increases, but the parameters still remain significantly different from their true values.

$\tilde{J}$	loglikelihood	μ_a	μ_b	σ_a	σ_b
10	-2207.78	1.15 (0.0884)	2.44 (0.173) *	1.07 (0.0934)	2.27 (0.172)
20	-3063.681	1.09 (0.0622)	2.28 (0.121) *	1.01 (0.0656)	2.13 (0.12)
30	-3595.1	1.08 (0.0558)	2.25 (0.107) *	1.03 (0.0593)	2.1 (0.106)
40	-3997.749	1.11 (0.0531) *	2.2 (0.0988) *	1.06 (0.056)	2.06 (0.0974)
50	-4331.109	1.09 (0.0498)	2.17 (0.0937)	1.04 (0.0524)	2.02 (0.092)
60	-4623.328	1.09 (0.0482)	2.14 (0.0897)	1.05 (0.051)	2 (0.0879)
70	-4870.381	1.1 (0.0478) *	2.15 (0.0887)	1.06 (0.05)	2.03 (0.0879)
80	-5059.614	1.11 (0.0469) *	2.14 (0.0859)	1.07 (0.0487)	2.01 (0.0847)
90	-5266.331	1.1 (0.046) *	2.14 (0.0848)	1.07 (0.0482)	2.01 (0.0837)
100	-5447.022	1.1 (0.0454) *	2.12 (0.0831)	1.07 (0.0475)	2 (0.0822)

Table 5.14: Models with sampling of alternatives and without corrections

$\tilde{J}$	loglikelihood	μ_a	μ_b	σ_a	σ_b
10	-2359.46	2.6 (0.185) *	5.34 (0.361) *	2.56 (0.191) *	5.14 (0.368) *
20	-3217.646	1.71 (0.0902) *	3.49 (0.175) *	1.65 (0.0932) *	3.31 (0.174) *
30	-3726.845	1.48 (0.0715) *	3 (0.138) *	1.44 (0.0744) *	2.85 (0.136) *
40	-4106.367	1.38 (0.0626) *	2.69 (0.117) *	1.34 (0.0651) *	2.54 (0.116) *
50	-4416.196	1.27 (0.0561) *	2.51 (0.106) *	1.23 (0.0584) *	2.36 (0.104) *
60	-4687.054	1.21 (0.0523) *	2.37 (0.0978) *	1.18 (0.055) *	2.23 (0.0958) *
70	-4916.173	1.19 (0.0505) *	2.31 (0.094) *	1.15 (0.0526) *	2.19 (0.0932) *
80	-5087.415	1.16 (0.0485) *	2.24 (0.089) *	1.12 (0.0501) *	2.1 (0.0877)
90	-5279.576	1.13 (0.0468) *	2.18 (0.0862) *	1.09 (0.0488)	2.05 (0.0851)
100	-5447.022	1.1 (0.0454) *	2.12 (0.0831)	1.07 (0.0475)	2 (0.0822)

Table 5.15: Models with sampling of alternatives and corrections

The following plots show the evolution of the parameters estimates with respect to the number of sampled alternatives, $\tilde{J}$ for both models M.2 and M.3.

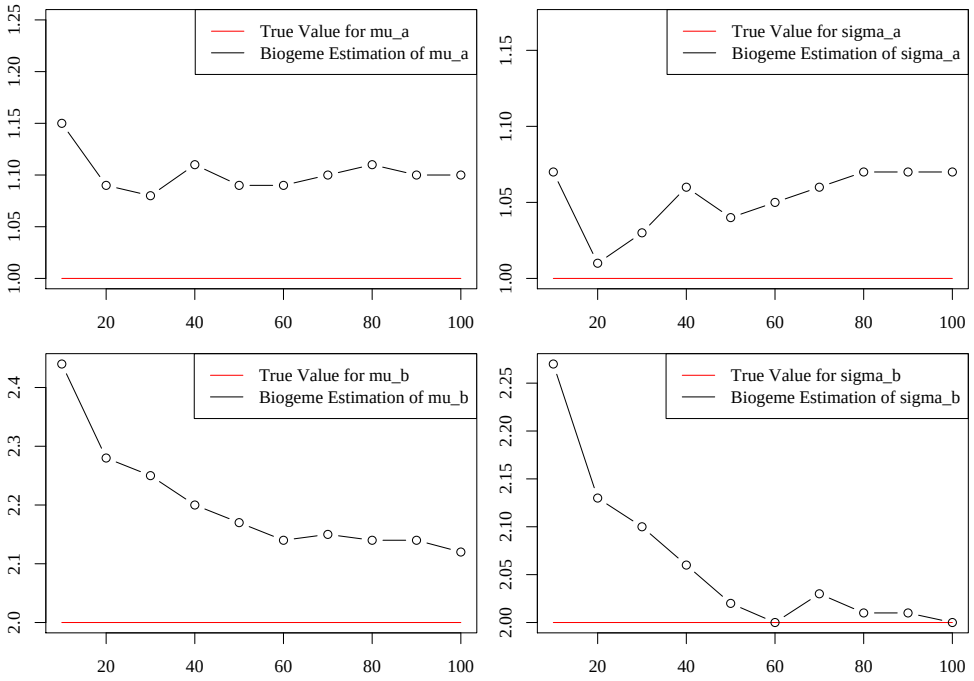

Figure 5.1: Evolution of the parameters estimations with the number of Alternatives for Model M.2: Sampling without Correction.
X-axis: Number of Sampled Alternatives
Y-axis: Parameters estimates

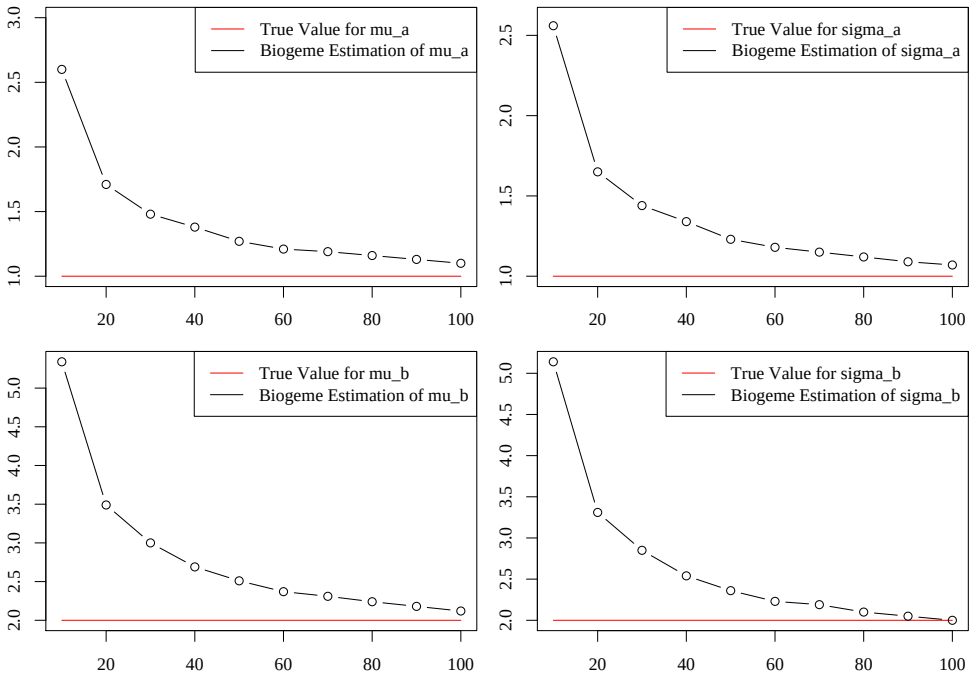

Figure 5.2: Evolution of the parameters estimations with the number of Alternatives for Model M.3: Sampling with Correction.
X-axis: Number of Sampled Alternatives
Y-axis: Parameters estimates

5.6 Different Sampling protocol

In this section, we present the fits' results of the model with the same specifications as presented in section (5.2). The attributes are generated from a uniform distribution and the random parameters from normal distributions. The only difference remains in the sampling protocol. In fact, instead of sampling the same number of alternatives, $\tilde{J}_n$, for all decision makers, we sample randomly different number of alternatives. That is, for each decision maker $n = 1, \ldots, N$, we pick up randomly a number of sampled alternatives, $\tilde{J}_n \in \{2, \ldots, J\}$. Once we assigned J_n to the decision maker n, we use a sampling without replacement protocol to choose for this decision maker n, a subset of alternatives D_n which includes the chosen alternative. First, the chosen alternative is drawn. Afterwards, the non chosen alternatives are randomly sampled up to complete a total of J_n alternatives. We repeat this experiment 10 times. Each time, we generate new set of data from the uniform distribution, construct the different set of sampled alternatives for each decision maker and fit the three models M.1, M.2 and M.3 as described previously. The results are given in tables (5.16), (5.17) and (5.18) below.

The same conclusion is reached as in the previous experiments. Indeed, fitting a Logit Mixture model with a reduced choice set leads in general to parameters' estimates which are not significantly different from their true values. However, fitting the same model with sampling of alternatives and correction of the log-likelihood equation leads to a significant bias in the parameters' estimates.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-5825.858	0.871 (0.0367) *	1.91 (0.0775)	0.905 (0.0409) *	1.97 (0.0788)
Experiment.2	-5749.012	1.01 (0.0421)	1.87 (0.0775)	1 (0.0454)	1.99 (0.08)
Experiment.3	-5697.115	0.985 (0.041)	1.96 (0.0793)	0.972 (0.044)	1.97 (0.0806)
Experiment.4	-5655.496	1.03 (0.0427)	2.04 (0.0838)	0.99 (0.0454)	2.09 (0.0853)
Experiment.5	-5700.591	1.05 (0.044)	1.96 (0.0832)	1.05 (0.0468)	2.13 (0.0861)
Experiment.6	-5592.81	1.03 (0.0416)	1.97 (0.0792)	0.935 (0.0427)	1.97 (0.0798)
Experiment.7	-5723.572	0.975 (0.0401)	1.92 (0.079)	0.932 (0.0425)	2.03 (0.0818)
Experiment.8	-5483.084	1.06 (0.0443)	2.19 (0.0864) *	1.07 (0.0475)	2.06 (0.0847)
Experiment.9	-5745.799	0.946 (0.0397)	1.96 (0.0791)	0.967 (0.0434)	1.98 (0.0809)
Experiment.10	-5648.202	0.998 (0.0411)	2 (0.0798)	0.988 (0.0443)	1.93 (0.078)

Table 5.16: Model M.1: Original Model.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-4408.2	0.883 (0.0424) *	1.97 (0.09)	0.908 (0.0479)	2.02 (0.0925)
Experiment.2	-4340.202	1.03 (0.049)	1.95 (0.0917)	0.985 (0.053)	2.07 (0.095)
Experiment.3	-4319.225	0.988 (0.0471)	1.96 (0.0899)	0.975 (0.0514)	1.98 (0.0928)
Experiment.4	-4274.792	1.02 (0.049)	2.03 (0.095)	0.985 (0.0526)	2.07 (0.0975)
Experiment.5	-4284.674	1.09 (0.0527)	2.07 (0.0991)	1.09 (0.057)	2.26 (0.106) *
Experiment.6	-4205.542	1.05 (0.0491)	2 (0.0919)	0.944 (0.0505)	2 (0.0935)
Experiment.7	-4319.411	0.977 (0.0458)	1.95 (0.0909)	0.912 (0.0487)	2.06 (0.096)
Experiment.8	-4158.825	1.05 (0.0512)	2.29 (0.104) *	1.06 (0.056)	2.14 (0.101)
Experiment.9	-4328.547	0.964 (0.047)	2.05 (0.0937)	0.991 (0.0522)	2.07 (0.0972)
Experiment.10	-4300.105	0.985 (0.0464)	2.01 (0.0916)	0.964 (0.0505)	1.95 (0.0902)

Table 5.17: Model M.2: Model with sampling of alternatives. For each decision maker n, we assign a reduced choice set D_n with different number of sampled alternatives J_n.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

	loglikelihood	μ_a	μ_b	σ_a	σ_b
Experiment.1	-4738.804	1.59 (0.0671) *	3.44 (0.142) *	1.72 (0.0723) *	3.62 (0.149) *
Experiment.2	-4655.01	1.84 (0.0769) *	3.45 (0.146) *	1.84 (0.0812) *	3.68 (0.151) *
Experiment.3	-4640.685	1.74 (0.074) *	3.42 (0.143) *	1.8 (0.0767) *	3.55 (0.15) *
Experiment.4	-4588.734	1.85 (0.0786) *	3.62 (0.154) *	1.87 (0.0809) *	3.74 (0.156) *
Experiment.5	-4565.322	1.92 (0.0823) *	3.61 (0.162) *	1.98 (0.0864) *	3.95 (0.167) *
Experiment.6	-4512.054	1.88 (0.0778) *	3.56 (0.15) *	1.76 (0.0782) *	3.61 (0.15) *
Experiment.7	-4628.111	1.74 (0.0715) *	3.44 (0.147) *	1.69 (0.0733) *	3.65 (0.15) *
Experiment.8	-4457.679	1.91 (0.0828) *	4.05 (0.167) *	1.99 (0.0862) *	3.87 (0.162) *
Experiment.9	-4635.125	1.72 (0.0738) *	3.55 (0.147) *	1.84 (0.0798) *	3.69 (0.155) *
Experiment.10	-4617.565	1.75 (0.0732) *	3.57 (0.147) *	1.79 (0.0762) *	3.51 (0.143) *

Table 5.18: Model M.3: Model with sampling and correction.
The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

5.7 Results: Models with Larger Choice Set $J = 500$

In this section, we present the fits' results for the Logit Mixture model with a larger choice set, $J = 500$. For this purpose, we used the new version of BIOGEME to write the utilities equations. In fact, in the new version of BIOGEME, the .mod file is substituted by a python script which automates the process of writing the utilities. We fit a Logit Mixture model with the same specifications as presented in section (5.2), only the total number of alternatives is now set to $J = 500$. We needed also to run the simulations on special computers with 46 GB of memory.
We fit different models by varying each time the number of sampled alternatives, $\tilde{J} = 10, 20, \ldots, 100$. The fits' results are given in tables (5.19) below. The first column of the table contains the number of sampled alternatives. The other columns contain the final log-likelihood and the values of the parameters of the random coefficients along with their standard deviation between parenthesis. The models' estimation was also performed using BIOGEME software. As in the previous sections, the star mark ($\star$) denotes the parameters which are significantly different form their true values. We recall that the true values are given by table (5.1) in section (5.2) and that we performed a t-test with a 5% significance level.
From table (5.19), we notice that, while varying the number of sampled alternatives, the models' estimation leads, in general, to parameters which are not significantly different from their true values. Only few parameters are significantly biased.

$\tilde{J}$	loglikelihood	μ_a	μ_b	σ_a	σ_b
10	-2300.803	0.982 (0.0769)	2.38 (0.172) *	0.902 (0.0824)	2.4 (0.177) *
20	-3132.145	0.934 (0.0556)	2.2 (0.12)	0.854 (0.0601) *	2.2 (0.122)
30	-3661.034	0.955 (0.0502)	2.15 (0.107)	0.885 (0.054) *	2.15 (0.11)
40	-4087.359	0.949 (0.0466)	2.09 (0.0981)	0.885 (0.05) *	2.09 (0.0996)
50	-4373.617	1.02 (0.0486)	2.11 (0.0972)	0.965 (0.0516)	2.13 (0.0989)
60	-4684.929	0.965 (0.0438)	2.12 (0.0945)	0.909 (0.0468)	2.12 (0.0955)
70	-4942.719	1.01 (0.0452)	2.07 (0.0907)	0.969 (0.0482)	2.08 (0.0915)
80	-5134.566	0.968 (0.0422)	2.07 (0.0881)	0.916 (0.045)	2.06 (0.0885)
90	-5309.373	0.996 (0.0431)	2.13 (0.0902)	0.948 (0.0456)	2.13 (0.0909)
100	-5490.959	0.988 (0.0419)	2.09 (0.0871)	0.946 (0.0442)	2.10 (0.0877)
500	-8443.666	0.976 (0.0373)	2.03 (0.0771)	0.942 (0.0393)	2.03 (0.0769)

Table 5.19: Models with sampling of alternatives and without correction. The number of sampled alternatives for each model is $\tilde{J} = 10, 20, \ldots, 100$ out of $J = 500$ alternatives. The table contains the values of the parameters and their standard deviation between parenthesis. The star ($\star$) denotes the parameters which are significantly different from their true value.

Chapter 6

Sampling of Alternatives for Logit Mixture Models

In this chapter, we try to present some ideas which, after further research, might constitute a starting point to justify the results obtained empirically in the previous chapter and understand why sampling of alternatives seems not to affect the parameters' estimation for a Logit Mixture model. These ideas are mainly based on the proof of McFadden concerning sampling of alternatives for Logit models.

Conditioning on the value of β, a Logit Mixture Model is simply a Logit Model. Hence, to deal with the issue of sampling of alternatives, we can follow the steps of Mcfadden for sampling of alternatives in the logit case:
It has been shown that the conditional probability of choosing alternative i given that a particular choice set D is constructed is given as follows

$$\pi(i|D_n, \beta, x) = \frac{e^{V_{ni}(x_{ni},\beta)+\ln \pi(D_n|i)}}{\sum_{j\in D_n} e^{V_{nj}(x_{nj},\beta)+\ln \pi(D_n|j)}} \tag{6.1}$$

However, the researcher does not observe the value of β and therefore cannot condition on the value of β. The unconditional version of the probability given in equation (6.1) is given by its integral over all possible values of β

$$\pi(i|D_n, x) = \int \frac{e^{V_{ni}(x_{ni},\beta)+\ln \pi(D_n|i)}}{\sum_{j\in D_n} e^{V_{nj}(x_{nj},\beta)+\ln \pi(D_n|j)}} f(\beta|\theta)d\beta \tag{6.2}$$

The modified log-likelihood function is given by

$$LL_N(\theta) = \sum_{n=1}^{N} \ln \int \frac{e^{V_{ni}(x_{ni},\beta)+\ln \pi(D_n|i)}}{\sum_{j\in D_n} e^{V_{nj}(x_{nj},\beta)+\ln \pi(D_n|j)}} f(\beta|\theta)d\beta \tag{6.3}$$

where alternative i is the chosen alternatives by individual n, $f(\cdot|\theta)$ is the mixing distribution of β and θ are the parameters of the distribution to be estimated.

Dividing and multiplying equation (6.3) by N, we can see it as an estimator for an expectation which depends on the true parameters θ^*, the sampling protocol used to draw the set D and the density of the data $g(\cdot)$. Hence, using the weak law of large number, the probability limit of $\frac{LL_N(\theta)}{N}$ is given as follows

$$LL = \int_x \sum_{i\in C}\sum_{D\subset C} \mathbb{P}(i|\theta^*, x, C)\pi(D|i,x) \times \Big[\ln \int \frac{e^{V_i(x,\beta)+\ln \pi(D|i)}}{\sum_{j\in D} e^{V_j(x,\beta)+\ln \pi(D|j)}} f(\beta|\theta)d\beta\Big]\ g(x)dx \quad (6.4)$$

Let's denote by $\pi(i|D,x,\theta) = \int \frac{e^{V_i(x,\beta)+\ln \pi(D|i)}}{\sum_{j\in D} e^{V_j(x,\beta)+\ln \pi(D|j)}} f(\beta|\theta)d\beta$. The previous equation becomes

$$LL = \int_x \sum_{i\in C}\sum_{D\subset C} \Big[\int \frac{e^{V_i(x,\beta)}}{\sum_{j\in C} e^{V_j(x,\beta)}} f(\beta|\theta^*)d\beta\Big]\pi(D|i,x) \times \ln \pi(i|D,x,\theta)g(x)dx \quad (6.5)$$

Using the same trick as in McFadden proof for sampling of alternatives in logit models, we try to make the term $\pi(i|D,x,\theta)$ appears inside the brackets in the previous equation by multiplying and dividing by the term $\sum_{j\in D} e^{V_j+\ln \pi(D|x,j)}$ and putting the term $\pi(D|i,x)$ inside the brakets. The right-hand side of equation (6.5) becomes

$$\int_x \sum_{i\in C}\sum_{D\subset C} \Big[\int \frac{e^{V_i(x,\beta)+\ln \pi(D|i,x)}}{\sum_{j\in D} e^{V_j+\ln \pi(D|j,x)}} \underbrace{\frac{\sum_{j\in D} \pi(D|j,x)\ e^{V_j(x,\beta)}}{\sum_{j\in C} e^{V_j(x,\beta)}}}_{\star} f(\beta|\theta^*)d\beta\Big] \ln \pi(i|D,x,\theta)g(x)dx \quad (6.6)$$

Let us set the term denoted by $\star$ as

$$\mathcal{C}(\beta,x,D) = \frac{\sum_{j\in D} \pi(D|j,x)\ e^{V_j(x,\beta)}}{\sum_{j\in C} e^{V_j(x,\beta)}} \quad (6.7)$$

We suppose now that this term does not depend on β, that is $\mathcal{C}(\beta,x,D) = \mathcal{C}(x,D)$. We can take it out from the integral over the β and equation (6.6) becomes

$$\begin{aligned} LL &= \int_x \sum_{i\in C}\sum_{D\subset C} \mathcal{C}(x,D) \underbrace{\int \frac{e^{V_i(x,\beta)+\ln \pi(D|i,x)}}{\sum_{j\in D} e^{V_j+\ln \pi(D|j,x)}} f(\beta|\theta^*)d\beta}_{=\pi(i|D,x,\theta^\star)} \times \ln \pi(i|D,x,\theta)\ g(x)dx \quad (6.8)\\ &= \int_x \sum_{D\subset C} \mathcal{C}(x,D) \sum_{i\in C} \pi(i|D,x,\theta^\star) \ln \pi(i|D,x,\theta)\ g(x)dx \quad (6.9)\\ &= \int_x \sum_{D\subset C} \mathcal{C}(x,D) \sum_{i\in C} \psi_i(\theta^\star) \ln \psi_i(\theta)\ dx \quad (6.10) \end{aligned}$$

where we set $\psi_i(\theta) = \pi(i|D,x,\theta)$.

Going back to sampling of alternatives in the logit case, let's first recall the conditional log-likelihood equation where β are the parameters to be estimated

$$L_N(\beta) = \sum_{n=1}^{N} \ln \big(\frac{e^{V_{ni}(x_{ni},\beta)+\ln \pi(D_n|i,x_n)}}{\sum_{j\in D_n} e^{V_{nj}(x_{nj},\beta)+\ln \pi(D_n|j,x_n)}} \big)$$

and let's us consider its probability limit given by $L = \lim\limits_{N\to\infty} \frac{L_N}{N}$. Then, following the same steps as done for the Logit Mixture model, we have the following equality

$$L = \int_x \sum_{D\subset C} K(x,D) \sum_{i\in C} \phi_i(\beta^\star) \ln \phi_i(\beta)\ dx \tag{6.11}$$

where $K(x,D)$ is some constant which does not depend on the parameter β and

$$\phi_i(\beta) = \frac{e^{V_i(x,\beta)}\pi(D|i,x)}{\sum_{j\in D} e^{V_j(x,\beta)}\pi(D|j,x)}$$

From equation (6.11), we notice that, in the case of the logit model, the parameter β enter this equation in a term of the form

$$\sum_{i\in C} \phi_i(\beta^\star) \ln \phi_i(\beta) \quad \text{where} \quad \sum_{i\in C} \phi_i(\beta) = 1 \tag{6.12}$$

McFadden (1978, see [10]) showed that the term in the left hand side of equation (6.12) has a maximum at the true parameter, *i.e*, for $\beta = \beta^\star$. Hence, L given in equation (6.11), has also a maximum at $\beta = \beta^\star$. Under normal regularity conditions, this maximum is unique and it was shown that the maxima of $\frac{L_N}{N}$ converge in probability to the maximum of L. This fact, establish that maximization of $\frac{L_N}{N}$, and thus, the conditional log-likelihood, yields consistent parameters' estimators for the logit models.

Returning to the Logit Mixture case, the next steps of the proof are to study further these different points

1. We need to justify the fact that the term $\mathcal{C}(\beta, x, D)$ given in equation (6.7) does not depend on β.
2. We need to maximize the term $\sum_{i\in C} \psi_i(\theta^\star) \ln \psi_i(\theta)$ over the parameter of the mixing distribution θ and show that the maximum is given by the true parameter $\theta = \theta^\star$.
3. If previous steps don't work, we need to optimize directly the log-likelihood function given by equation (6.5) and show that this maximum is reached for $\theta = \theta^\star$
4. We need to show that the maxima of $\frac{LL_N(\theta)}{N}$ converge in probability to the maxima of LL which is tricky for the Logit Mixture model since we don't have anymore the nice concave feature of the log-likelihood function as it is the case for the logit model.

Chapter 7

Conclusion

The attraction of the Logit Mixture model is its generality. Indeed, the Logit Mixture model is not saddled by the IIA assumption allowing thus, a more complex error structure of the utility's equations. It also captures heterogeneity among the decision makers in sensitivity to exogenous variables. However, when the choice set is very large, estimation of a Logit Mixture model from the full choice set can be very expensive. In some cases, identifying and measuring the attributes of a large number of alternatives is impossible and thus estimation of a Logit Mixture model from the full choice set is impossible as well. Sampling of alternatives becomes in this case necessary for estimation.
After investigating the literature, it appears that sampling of alternatives for Logit Mixture model does not introduce a significant bias in the parameters' estimation. The previous studies were mainly empirical investigation examining how the sample size of alternatives affects the estimated parameters. These studies used both synthetic Monte Carlo data and real data and the same conclusion was reached in both cases.

In this project, we tried to assess empirically the impact of sampling of alternatives on the estimated parameters using the BIOGEME software. We used Monte Carlo data generated from different distributions, mainly uniform and normal distributions. Our experiments consisted in fitting three models. The first model is simply a Logit Model fitted with the whole choice set. The second model is a Logit Mixture model fitted with the reduced choice set. The third model is the same as the previous one but has a corrected log-likelihood equation. The weights used for correction are the same used to correct the log-likelihood equation for the MEV model. We performed a first set of experiments where we fitted the same model many times by generating each time new attributes from the same distribution. We did this procedure using data generated from the uniform and normal distribution. The purpose of doing so is to assess the stability of the experiments' results with respect to the data. The conclusion was the same for each experiment: Sampling of alternatives does not alter in general the parameters' estimation of the Logit Mixture model. However, fitting a Logit Mixture model with a reduced choice set and correction of the log-likelihood leads to a significant bias in the parameters' estimation.
We tried also different sets of experiments by fitting a Logit Mixture model several times

with the same data but each time, with varying the number of sampled alternatives. We reached the same conclusion as before. In fact, while varying the number of sampled alternatives, the estimation of the Logit Mixture model with a reduced choice set leads to parameters which are not significantly different from their true values. However, correcting the log-likelihood equation and fitting the same model with sampling of alternatives lead to a significant bias in the parameters' estimation.
Using the new version of BIOGEME which automates the process of writing the utility's equations, we fitted a Logit Mixture model with a larger choice set containing in total 500 alternatives. For this case, we needed to run the simulations on a computer with a large RAM memory. We had access to 46 GB RAM computer and fitted only the Logit Mixture model with full and reduced choice sets. We repeated the same experiments as described previously by fitting the same model several times with varying each times the size of the choice set. The same results were reached.
For testing purpose, we tried different sampling protocols. In all the previous experiments, we constructed the reduced choice set randomly by including first the chosen alternative and sampling without replacement the non chosen alternatives up to reach the required size of the reduced choice set. We tried another sampling protocol by making the size of the reduced choice set different for each decision maker. We repeated the same experiments using this sampling protocol and attributes generated from the uniform distribution. We fitted the three models several times by generating each time a new data set and constructing the different sets of sampled alternatives for each decision maker. The same results are reached: Fitting a Logit Mixture model with a reduced choice set leads, in general, to parameters' estimates which are not significantly different from their true values. However, fitting the same model with sampling of alternatives and correction of the log-likelihood equation leads to a significant bias in the parameters' estimates.
As we obtained the same results as in the case of sampling of alternatives for logit models, we tried to mimic McFadden's proof in order to give a theoretical support for our empirical investigation. The proof is not complete but might constitute a start for justifying the results obtained for sampling of alternatives in Logit Mixture models. More research to develop this proof might broaden our understanding of sampling of alternatives for Logit Mixture models.

Chapter 8

Appendix

8.1 Appendix A: R Code to create the .dat file

In Appendix A, we explain how we generated the .dat file for BIOGEME software to fit a Logit Mixture model. We used the software R, to generate the attributes, construct the reduced sets of sampled alternatives and calculate the weights to correct to log-likelihood function.
First, we need to load the following libraries in R

```
library(evd)
library(statmod)
library(adapt)
```

Afterward, we need to choose the total number of observations, the number of alternatives J and the number of sampled alternatives $\tilde{J}$. The next step, is to generate from the uniform distribution, the two attributes of the model. In the code, the two attributes are denoted by a and b. These are matrices which has N rows and J columns. Specifically, the n^{th} row of each matrix contains J values of each attribute corresponding to the J alternatives. Once we have the attributes, we generate from two different normal distributions, the values of the random coefficients. These values are given by two vectors of length N denoted in the code by 'beta_a' and 'beta_b'. The error part of the utility is denoted in the code by the matrix 'epsilon' which is a $N \times J$ matrix containing independent draws from the gumbel distribution.

```
N<-2000
J<- 100
J1<- 10

#--- Generating the attributes
Min <- -5
Max <- 5
```

```
a <- matrix(runif(N*J, Min, Max), N, J)
b <- matrix(runif(N*J, Min, Max), N, J)

#--- Generating the random coefficients
beta_a <- rnorm(N, 1, 1 )
beta_b <- rnorm(N, 2, 2)

#--- Generating the error part of the utility
epsilon <- matrix(rgumbel(N*J), N, J)

#--- Constructing the Utilities
V <- beta_a * a + beta_b * b
U <- V + epsilon
```

Once we constructed the J utilities for each observation n, we can get the chosen alternatives which has the maximum utility. This is done by R using the function 'Choice()' which take as input, the utilities 'U' and returns a $N \times J$ matrix. The nj^{th} element of this matrix is equal to either 0 or 1. If it is equal to 1, it means that alternative j is the chosen alternative. The logit probabilities are obtained by the function 'True.Probabilities()' and the weight used to correct the log-likelihood are computed by the function 'Weight()'. Sampling of alternatives is done by the function 'Sampling()' which return a list. One of the arguments of this list is 'sampled.alt' which is a vector containing the ordinal coordinate of the sampled alternatives. The function 'Availabilities()' assign to each alternatives the number 0 or 1 depending whether it is sampled or not. As described in section (5.6), we used also other sampling protocol which consists in constructing reduced choice sets with different number of sampled alternatives across the individuals. The R function 'Sampling.Diff.Nbre.Alt()' performs this kind of sampling of alternatives.

```
choice <- Choice(U)

#--- True Probabilities given by the logit Formula
true.prob <- matrix(NA, N, J)
true.prob <- True.Probabilities(V)

#--- Weights to correct the log-likelihood function
weights <- matrix(NA, N, J)
weights <- Weights(true.prob, J1)

#--- Sampling of Alternatives
sampling <- Sampling(choice,a,b,epsilon,J1)
availabilities <- Availabilities(sampling$sampled.alt, J)
```

The following code, is used to create the matrix X of the data which contains the ID of the decision maker, the chosen alternatives, the attributes, the weights and the availabilities.

```
#--- The BIOGEME DATA.
X <- matrix(NA, N, 2+4*J)

#---Column Names
cnames<-rep(NA,2+4*J)
cnames[1]<- "id"
cnames[2]<-"choice"
for(i in 1:J){
cnames[i+2]<- paste("a", i, sep="")
cnames[i+2+J]<- paste("b", i, sep="")
cnames[i+2+(2*J)] <- paste("w", i, sep="")
cnames[i+2+(3*J)] <- paste("av", i, sep="")
}
colnames(X)<- cnames
rownames(X) <- rownames(X, do.NULL = FALSE, prefix = " ")
for(i in 1:N){
X[i,1] <- i
X[i,2] <- which( choice[i,] == 1)
X[i,3:(3+J-1)] <- a[i,]
X[i,(3+J):(2*J+2)] <- b[i,]
X[i,(2*J+3):(2+3*J)] <- weights[i,]
X[i,(3+3*J):(2+4*J)] <- availabilities[i,]
}

#--- Creating the .dat file
write.table(X, file="/Users/ines/Desktop/epfl stuff 2009-2010/mit-project/
R_workspace/Unif-Data-J.100-J1.10.dat", quote=FALSE, sep=" ", row.names = FALSE)
```

The functions mentioned above are given by the following R code

```
#######################################################################
#-------------------------------- Functions ----------------------------
#######################################################################

#--- The choice function
"Choice" <- function(utility)
{
N<-nrow(utility)
J<-ncol(utility)
choice<-matrix(NA, N, J)
```

```
for(i in 1:N){
choice[i,] <- as.numeric(utility[i,] == max(utility[i,]))
}

return(choice)
}

#--- The True probabilities given by the Logit formula
"True.Probabilities" <- function(V){
N <- nrow(V)
J <- ncol(V)
true.prob <- matrix(NA, N, J)
true.prob <- exp(V) / apply(exp(V), 1, sum)

return(true.prob)
}

#--- The weights correcting the log-likelihood function
"Weights" <- function(true.prob, J1){
N <- nrow(true.prob)
J<- ncol(true.prob)
weights <- matrix(NA, N, J)
weights <- 1/(true.prob * 1 + (1-true.prob) * (J1-1)/(J-1))

return(weights)
}

"Availabilities" <- function(sampled.alt, J){
N <- nrow(sampled.alt)
availabilities <- matrix(0, N, J)
for(n in 1:N){
availabilities[n,sampled.alt[n,]] <-1
}

return(availabilities)
}

#--- Sampling of Alternatives
"Sampling" <- function(choice, a, b, epsilon, J1){
a<-as.matrix(a)
b<-as.matrix(b)
N<- nrow(choice)
J<- ncol(choice)
```

```
a1<- matrix(NA, N, J1)
b1<-matrix(NA, N, J1)
epsilon1 <- matrix(NA, N, J1)

alt.indice <- c(1:J)

sampled.alt <- matrix(NA, N, J1)
for(n in 1:N){
#Selecting The chosen Alternative
chosen.alt <- which(choice[n,]==1)
sampled.alt[n,1] <- chosen.alt
a1[n,1] <- a[n, chosen.alt]
b1[n,1] <- b[n, chosen.alt]
epsilon1[n, 1] <- epsilon[n, chosen.alt]


#Sampling J1-1 alternative among the rest
sampled.alt[n, 2:J1] <- sample(alt.indice[-chosen.alt], size= J1-1)
a1[n,2:J1] <- a[n, sampled.alt[n, 2:J1]]
b1[n,2:J1] <- b[n, sampled.alt[n, 2:J1]]
epsilon1[n, 2:J1] <- epsilon[n, sampled.alt[n, 2:J1]]
}

return(sampling=list(a=a1, b=b1, epsilon= epsilon1, sampled.alt = sampled.alt))
}

##---- DIFFERENT SAMPLING PROTOCOLS
##---- Sampling Different Number of Alternatives for different Observations
##---- The Chosen alternatives is automatically included
"Sampling.Diff.Nbre.Alt" <- function(N, J){
nbre.sampled.alt <- rep(NA, N)
sampled.alt <- matrix(NA, N, J)

for(n in 1:N){
#--- Sampling J1: for each observation we associate a different number of sampled
#--- Alternatives: Minimum number of Sampled Alternatives is 2
#Selecting The chosen Alternative
chosen.alt <- which(choice[n,]==1)
sampled.alt[n,1] <- chosen.alt

J1 <- sample(c(2:J), 1)
nbre.sampled.alt[n]<-J1
```

```
alt.indice <- c(1:J)
sampled.alt[n, 2:J1] <- sample(alt.indice[-chosen.alt], size= (J1-1))
}

return(sampling=list(nbre.sampled.alt = nbre.sampled.alt, sampled.alt = sampled.alt))
}
```

An example of the .dat file generated by the previous R code is given by figure (8.1) where the total number of decision makers is $N = 10$, the total number of alternatives is $J = 5$ and the number of sampled alternatives is $\tilde{J} = 2$.

id	choice	a1	a2	a3	a4	a5	b1	b2	b3	b4	b5	w1	w2	w3	w4	w5	av1	av2	av3	av4	av5
1	2	-2.03	-0.8	0.66	-4.02	-4.71	4.12	4.48	0.51	3.42	-0.25	3.74	1.17	2.66	4	4	1	1	0	0	0
2	3	-4.98	0.29	4.99	3.49	-1.35	3.26	-3.55	-3.85	1.66	0.33	4	4	1	4	4	0	1	1	0	0
3	1	2.17	-1.19	-2.02	-4.02	-0.15	3.74	-2.29	-3.54	4.7	4.93	1	4	4	4	3.97	1	0	0	0	1
4	2	4.71	4.49	-0.68	4.35	0.14	-4.52	-1.58	-4.88	-1.92	-0.78	3.98	1.31	4	2.16	3.71	0	1	0	1	0
5	5	-3.68	-1.42	-3.63	1.41	3.41	-0.77	-4.41	4.61	2.16	3.69	4	4	4	3.99	1	0	1	0	0	1
6	2	3.61	2.77	-3.57	-1.55	1.04	1.53	2.95	-2.86	-0.57	-2.01	1.87	1.86	3.62	3.22	2.94	1	1	0	0	0
7	5	-3.99	-3.97	1.06	0.1	-3.36	-1.46	4.97	2.74	-0.3	4.46	4	1.21	3.97	4	2.37	0	0	1	0	1
8	3	2.55	1.89	2.82	1.37	3.43	-0.75	-1.4	2.81	-4.2	-3.95	4	4	1	4	4	0	1	1	0	0
9	4	0.44	-4.9	-0.13	-2.74	-1.1	-1.72	-4.57	-2.57	2.5	0.45	4	4	4	1	4	0	0	0	1	1
10	3	-0.72	-3.43	0.93	2.32	-0.69	1.53	4.9	2.71	-2.81	-4.9	3.84	2.45	1.2	4	4	0	0	1	1	0

Figure 8.1: Example of a .dat file generated by the previous R code: $N = 10$, $J = 5$ and $\tilde{J} = 2$.

The .dat file contains $2 + 4 * J$ columns which are described as follows

- Column 'id': gives the id of the decision makers
- Column 'choice': gives the chosen alternatives for each decision maker
- Columns 'a': gives the values of the attribute 'a' corresponding to each decision maker and alternative.
- Columns 'b': gives the values of the attribute 'b' corresponding to each decision maker and alternative.
- Columns 'w': gives the weights corresponding to each decision maker and alternative.
- Columns 'av': indicates, for each decision maker, whether the alternative i is sampled or not.

8.2 Appendix B: BIOGEME .mod files and python scripts

Once we have the .dat file, we can fit the three Logit Mixture models using either the old version or the new version of BIOGEME. For both cases, we give the code to fit the three models M.1, M.2 and M.3 as described in chapter (4).

8.2.1 Code for the .mod file

Fitting model M.1

This is the .mod file for model M.1 to fit a Logit Mixture model. No sampling of alternatives is done. We only give the code for the 5 first utilities' equations.

```
[DataFile]
$COLUMNS = 402

[Choice]
choice

[Beta]
// Name Value     LowerBound UpperBound  status (0=variable, 1=fixed)
beta_a  0.0           -100.0      100.0          0
beta_b  0.0           -100.0      100.0          0

   sd_a 1.0           -100.0      100.0          0
   sd_b 1.0           -100.0      100.0          0

[Utilities]
// Id   Name   Avail   linear-in-parameter expression (beta1*x1 + beta2*x2 + ... )
   1    Alt1     one    beta_a [ sd_a ] * a1 + beta_b [ sd_b ] * b1
   2    Alt2     one    beta_a [ sd_a ] * a2 + beta_b [ sd_b ] * b2
   3    Alt3     one    beta_a [ sd_a ] * a3 + beta_b [ sd_b ] * b3
   4    Alt4     one    beta_a [ sd_a ] * a4 + beta_b [ sd_b ] * b4
   5    Alt5     one    beta_a [ sd_a ] * a5 + beta_b [ sd_b ] * b5

[Model]
$MNL

[Expressions]
one  = 1

[Draws]
1000
```

The first and the second columns correspond to the ID and the name of each alternatives. The third column, 'Avail', specifies which alternatives are available. In our models, putting 'one' in each line means that all the alternatives are considered while fitting the model. The last column gives the specifications of the model.

Fitting model M.2

This is the .mod file for model M.2 to fit a Logit Mixture model with sampling of alternatives. We only give the code for the 5 first utilities equations.

```
\begin{verbatim}
[DataFile]
$COLUMNS = 402

[Choice]
choice

[Beta]
// Name Value     LowerBound UpperBound  status (0=variable, 1=fixed)
beta_a  0.0           -100.0      100.0          0
beta_b  0.0           -100.0      100.0          0

   sd_a 1.0           -100.0      100.0          0
   sd_b 1.0           -100.0      100.0          0

[Utilities]
// Id   Name   Avail   linear-in-parameter expression (beta1*x1 + beta2*x2 + ... )
   1    Alt1     av1    beta_a [ sd_a ] * a1 + beta_b [ sd_b ] * b1
   2    Alt2     av2    beta_a [ sd_a ] * a2 + beta_b [ sd_b ] * b2
   3    Alt3     av3    beta_a [ sd_a ] * a3 + beta_b [ sd_b ] * b3
   4    Alt4     av4    beta_a [ sd_a ] * a4 + beta_b [ sd_b ] * b4
   5    Alt5     av5    beta_a [ sd_a ] * a5 + beta_b [ sd_b ] * b5

[Model]
$MNL

[Draws]
1000
```

The only difference with the previous BIOGEME code, is that the third column contains, for a given decision maker, a variable denoted by **'av_i'** which is equal to either 1 or 0. This variable specifies whether the corresponding alternative is available or not to that specific decision maker.

Fitting model M.3

This is the .mod file for model M.3 to fit a Logit Mixture model with sampling of alternatives and correction of the log-likelihood equation. We only give the code for the 5 first utilities' equations.

```
[DataFile]
$COLUMNS = 402

[Choice]
choice

[Beta]
// Name Value     LowerBound UpperBound  status (0=variable, 1=fixed)
 alpha  1.0           -100.0      100.0          1

beta_a  0.0           -100.0      100.0          0
beta_b  0.0           -100.0      100.0          0

   sd_a 1.0           -100.0      100.0          0
   sd_b 1.0           -100.0      100.0          0

[Utilities]
// Id   Name  Avail   linear-in-parameter expression
   1    Alt1    av1    beta_a [ sd_a ] * a1 + beta_b [ sd_b ] * b1 + alpha * ln_w1
   2    Alt2    av2    beta_a [ sd_a ] * a2 + beta_b [ sd_b ] * b2 + alpha * ln_w2
   3    Alt3    av3    beta_a [ sd_a ] * a3 + beta_b [ sd_b ] * b3 + alpha * ln_w3
   4    Alt4    av4    beta_a [ sd_a ] * a4 + beta_b [ sd_b ] * b4 + alpha * ln_w4
   5    Alt5    av5    beta_a [ sd_a ] * a5 + beta_b [ sd_b ] * b5 + alpha * ln_w5


[Model]
$MNL

[Expressions]
ln_w1 = log( w1 )
ln_w2 = log( w2 )
ln_w3 = log( w3 )
ln_w4 = log( w4 )
ln_w5 = log( w5 )

[Draws]
1000
```

We have the same code as for model M.2. The only difference is that the term **'alpha * ln_wi'** is added in each of the utilities' equations. The second part of this term is the log-transform of the weights which is specified under [Expressions].

8.2.2 Code for Python script

Fitting model M.1

This is the python script for model M.1 to fit a Logit Mixture model using the new version of BIOGEME. No sampling of alternatives is done.

```
from biogeme import *
from headers import *
from logit import *
from loglikelihood import *
from statistics import *

dataFile = "Data-J1000-J1.5.dat"

beta_a =  Beta( 'beta_a', 0.0, -10000, 10000, 0)
beta_b =  Beta( 'beta_b', 0.0, -10000, 10000, 0)
sd_a =  Beta( 'sd_a', 1.0, -10000, 10000, 0)
sd_b =  Beta( 'sd_b', 1.0, -10000, 10000, 0)

betaRnd_a = beta_a + bioNormal('a') * sd_a
betaRnd_b = beta_b + bioNormal('b') * sd_b

# Attributes
a={}
for i in range(1,501):
        a[i] = Variable('a' + str( i ))

b={}
for i in range(1,501):
        b[i] = Variable('b' + str( i ))

# Utilities
V={}
for i in range(1,501):
        V[i] = betaRnd_a * a[i] + betaRnd_b * b[i]

# Availabilities
av={}
for i in range(1,501):
```

```
        av[i] = 1

chosen = Elem(V,choice)
den = 0
for i,v in V.items() :
    den += av[i] * exp(v-chosen)

avail = Elem(av,choice)
P = avail / den

drawIterator('drawIter')
l = Sum(P,'drawIter')
rowIterator('obsIter', Datafile(dataFile))

BIOGEME_OBJECT.ESTIMATE = Sum(log(l),'obsIter')
BIOGEME_OBJECT.DRAWS = 1000
BIOGEME_OBJECT.PARAMETERS['optimizationAlgorithm'] = "CFSQP"
BIOGEME_OBJECT.PARAMETERS['numberOfThreads'] = "4"
```

Fitting model M.2

This is the python script for model M.2 to fit a Logit Mixture model with sampling of alternatives using the new version of BIOGEME.

```
from biogeme import *
from headers import *
from logit import *
from loglikelihood import *
from statistics import *

dataFile = "data-J500.dat"

beta_a =  Beta( 'beta_a', 0.0, -10000, 10000, 0)
beta_b =  Beta( 'beta_b', 0.0, -10000, 10000, 0)
sd_a =  Beta( 'sd_a', 1.0, -10000, 10000, 0)
sd_b =  Beta( 'sd_b', 1.0, -10000, 10000, 0)

betaRnd_a = beta_a + bioNormal('a') * sd_a
betaRnd_b = beta_b + bioNormal('b') * sd_b

# Attributes
a={}
```

```
for i in range(1,501):
        a[i] = Variable('a' + str( i ))

b={}
for i in range(1,501):
        b[i] = Variable('b' + str( i ))

# Utilities
V={}
for i in range(1,501):
        V[i] = betaRnd_a * a[i] + betaRnd_b * b[i]

# Availabilities
av={}
for i in range(1,501):
        av[i] = Variable('av' + str( i ))

chosen = Elem(V,choice)
den = 0
for i,v in V.items() :
    den += av[i] * exp(v-chosen)

avail = Elem(av,choice)
P = avail / den

drawIterator('drawIter')
l = Sum(P,'drawIter')

rowIterator('obsIter', Datafile(dataFile))
BIOGEME_OBJECT.ESTIMATE = Sum(log(l),'obsIter')
BIOGEME_OBJECT.DRAWS = 2000
BIOGEME_OBJECT.PARAMETERS['optimizationAlgorithm'] = "CFSQP"
BIOGEME_OBJECT.PARAMETERS['numberOfThreads'] = "4"
```

Fitting model M.3

This is the python script for model M.3 to fit a Logit Mixture model with sampling of alternatives and correction of the log-likelihood equation.

```
from biogeme import *
from headers import *
from logit import *
```

```
from loglikelihood import *
from statistics import *

dataFile = "data-J500.dat"

beta_a =  Beta( 'beta_a', 0.0, -10000, 10000, 0)
beta_b =  Beta( 'beta_b', 0.0, -10000, 10000, 0)
sd_a =  Beta( 'sd_a', 1.0, -10000, 10000, 0)
sd_b =  Beta( 'sd_b', 1.0, -10000, 10000, 0)

betaRnd_a = beta_a + bioNormal('a') * sd_a
betaRnd_b = beta_b + bioNormal('b') * sd_b

alpha = 1

# Attributes
a={}
for i in range(1,501):
        a[i] = Variable('a' + str( i ))

b={}
for i in range(1,501):
        b[i] = Variable('b' + str( i ))

lnW={}
for i in range(1,501):
        lnW[i] = log(Variable('w' + str( i )))


# Utilities
V={}
for i in range(1,501):
        V[i] = betaRnd_a * a[i] + betaRnd_b * b[i] + alpha * lnW[i]

# Availabilities
av={}
for i in range(1,501):
        av[i] = Variable('av' + str( i ))

chosen = Elem(V,choice)
den = 0
for i,v in V.items() :
    den += av[i] * exp(v-chosen)
```

```
avail = Elem(av,choice)
P = avail / den

drawIterator('drawIter')
l = Sum(P,'drawIter')

rowIterator('obsIter', Datafile(dataFile))
BIOGEME_OBJECT.ESTIMATE = Sum(log(l),'obsIter')
BIOGEME_OBJECT.DRAWS = 2000
BIOGEME_OBJECT.PARAMETERS['optimizationAlgorithm'] = "CFSQP"
BIOGEME_OBJECT.PARAMETERS['numberOfThreads'] = "4"
```

8.3 Appendix C: Data Processing

Within the scope of this project, we had to run many simulations and thus, processing the BIOGEME outputs, mainly the .html files containing the fits' results, was an important part of our work. We had to automate some tasks such as collecting some specfic results from the .html files (mainly the log-likelihood value, the parameters' estimation and their standard deviation), creating .xls files containing these results, performing a 5% t-test to figure out whether the estimates are significantly different from their true values or not and finally, generating the latex-code which contains the final fits' results as shown in the previous sections.
We mainly used Visual Basic programming (VBA) to perform the first task of collecting the necessary results from the .htlm files (log-likelihood value, parameters' estimates and their standard deviation). We used R, to perform the t-test and generate the latex code for the final fits' results. Both codes are given in the next sections.

8.3.1 Data Processing: VBA code

Using VBA, we first convert the .html files into .xls files. The function 'Main_Convert()' calls 3 sub-routines which effectuate the conversion depending on the name of the files.

```
Private Const original As String = "Original_Model_"
Private Const sampled As String = "Sampled_Model_"
Private Const corrected As String = "Corrected_Model_"
Private Const startconv As Integer = 1
Private Const finishconv As Integer = 10
Private Const liststart As Integer = 1
Private Const liststop As Integer = 10
Private Const oldExt As String = ".html"
Private Const saveExt As String = ".xls"
Private model As String

Private Const wd As String = "C:\Documents and Settings\ITS  Lab\Desktop\_
Ines workspace\new_uniform\html files\"
Private Const sd As String = "C:\Documents and Settings\ITS  Lab\Desktop\_
Ines workspace\new_uniform\xls files\"

Public L As String
Public j As Integer
Public i As Integer
Public colnum_likelihood As Integer
Public beta_a As Integer
Public beta_a_sdv As Integer
Public beta_b As Integer
```

```
Public beta_b_sdv As Integer
Public sigma_a As Integer
Public sigma_a_sdv As Integer
Public sigma_b As Integer
Public sigma_b_sdv As Integer

Sub convOriginal()
fname = wd & original & L & oldExt
fsavename = sd & original & L & saveExt
Workbooks.Open (fname)
ActiveWorkbook.SaveAs Filename:=fsavename, FileFormat:=xlNormal,_
Password:="", WriteResPassword:="", _
         ReadOnlyRecommended:=False, CreateBackup:=False
ActiveWorkbook.Close
    End Sub

Sub convSampled()
fname = wd & sampled & L & oldExt
fsavename = sd & sampled & L & saveExt
Workbooks.Open (fname)
ActiveWorkbook.SaveAs Filename:=fsavename, FileFormat:=xlNormal, _
Password:="", WriteResPassword:="", _
         ReadOnlyRecommended:=False, CreateBackup:=False
ActiveWorkbook.Close
End Sub

Sub convCorrected()
fname = wd & corrected & L & oldExt
fsavename = sd & corrected & L & saveExt
Workbooks.Open (fname)
ActiveWorkbook.SaveAs Filename:=fsavename, FileFormat:=xlNormal,_
Password:="", WriteResPassword:="", _
         ReadOnlyRecommended:=False, CreateBackup:=False
ActiveWorkbook.Close
End Sub

Sub Main_Convert()
For i = startconv To finishconv
   L = CStr(i)
   convOriginal
   convSampled
   convCorrected
Next i
```

```
End Sub
```

Once we have the .html files converted into .xls files, we copy the necessary fits' results into a new table which contains the log-likelihood function, the parameters' estimates and their standard deviation. The function 'Main_Fill' calls three subroutines which effectuate this task depending on the name of the files. These subroutines are called 'Sub Fill_from_original()', 'Sub Fill_from_sampled()' and 'Sub Fill_from_corrected()'. We only give here the code for the main routine and 'Sub Fill_from_original()'.

```
Sub Fill_from_original()
        '--- take the loglikelihood
        colnum_likelihood = 2
        Cells(j, colnum_likelihood) = "='" & sd & "[" & original & L &_
         saveExt & "]" & original & L & "'" & "!" & "$B$18"
        Cells(j, colnum_likelihood).Value = Cells(j, colnum_likelihood)

        '--- take the beta_a
        beta_a = 3
        Cells(j, beta_a) = "='" & sd & "[" & original & L & saveExt & "]" &_
         original & L & "'" & "!" & "$B$32"
        Cells(j, beta_a).Value = Cells(j, beta_a)

        '--- take the beta_a_sd
        beta_a_sdv = 4
        Cells(j, beta_a_sdv) = "='" & sd & "[" & original & L & saveExt & "]" &_
         original & L & "'" & "!" & "$C$32"
        Cells(j, beta_a_sdv).Value = Cells(j, beta_a_sdv)

        '--- take the beta_b
        beta_b = 5
        Cells(j, beta_b) = "='" & sd & "[" & original & L & saveExt & "]" &_
         original & L & "'" & "!" & "$B$33"
        Cells(j, beta_b).Value = Cells(j, beta_b)

        '--- take the beta_b_sd
        beta_b_sd = 6
        Cells(j, beta_b_sd) = "='" & sd & "[" & original & L & saveExt & "]" &_
         original & L & "'" & "!" & "$C$33"
        Cells(j, beta_b_sd).Value = Cells(j, beta_b_sd)

        '--- take the sigma_a
        sigma_a = 7
```

```
        Cells(j, sigma_a) = "='" & sd & "[" & original & L & saveExt & "]" &_
         original & L & "'" & "!" & "$B$34"
        Cells(j, sigma_a).Value = Cells(j, sigma_a)

        '--- take the sigma_a_sd
        sigma_a_sdv = 8
        Cells(j, sigma_a_sdv) = "='" & sd & "[" & original & L & saveExt & "]" &_
         original & L & "'" & "!" & "$C$34"
        Cells(j, sigma_a_sdv).Value = Cells(j, sigma_a_sdv)

        '--- take the sigma_b
        sigma_b = 9
        Cells(j, sigma_b) = "='" & sd & "[" & original & L & saveExt & "]" &_
         original & L & "'" & "!" & "$B$35"
        Cells(j, sigma_b).Value = Cells(j, sigma_b)

        '--- take the sigma_b_sd
        sigma_b_sdv = 10
        Cells(j, sigma_b_sdv) = "='" & sd & "[" & original & L & saveExt & "]" &_
         original & L & "'" & "!" & "$C$35"
        Cells(j, sigma_b_sdv).Value = Cells(j, sigma_b_sdv)
End Sub

Sub Main_Fill()
For j = liststart + 3 To liststop + 3
        i = j - 3
        L = CStr(i)

        Fill_from_original
        Fill_from_sampled
        Fill_from_corrected
Next j
End Sub
```

8.3.2 Data Processing: R code

This is the code of two R functions:

1. 'Significance()' function which performs a 5% the t-test on the parameters' estimates against their true values.

2. 'Latex.Format() which gives the latex code for the results' tables presented in the previous sections. Note that the xtable library need to be loaded in R.

```
#--- Significance of the Bigoeme Estimates -------
"Significance" <- function(data, nbre.estimates=4, true.parameters=c(1,2,1,2)){
data <- as.matrix(data)
sd<-matrix(NA, nrow(data), (3*nbre.estimates))
sd<-data[,which(colnames(data)=="sd")]
parameters<-matrix(NA, nrow(data), (3*nbre.estimates))
parameters<-data[,-which(colnames(data) %in% c("loglikelihood","sd"))]

significance <- matrix(NA,nrow(data), (3*nbre.estimates))
colnames(significance)<-colnames(data)
rownames(significance)<-rownames(parameters)

for(n in 1:nrow(data)){
t.statistic<-abs(parameters[n,]-c(true.parameters, true.parameters,
 true.parameters))/sd[n,]
significance[n,which(colnames(data)%in%c("beta_a", "beta_b",
"sigma_a", "sigma_b"))]<-1-as.numeric(t.statistic <=1.96)
}
significance<-significance[, -which(colnames(significance) == "sd")]
return(significance)

}
#---Latex table:
library(xtable)
"Latex.Format"<-function(data, significance){
data<-as.matrix(data)
nRow <-nrow(data)
nCol <-ncol(data[,-which(colnames(data)=="sd")])
latex.data<-matrix(NA, nRow, nCol)
colnames(latex.data)<-colnames(data[,-which(colnames(data)=="sd")])
rownames(latex.data)<-rownames(data)
for(n in 1:nRow){
latex.data[n,which(colnames(latex.data)=="loglikelihood")]
=data[n,which(colnames(data)=="loglikelihood")]
latex.data[n,-which(colnames(latex.data)=="loglikelihood")]
=paste(data[n,which(colnames(data)%in%c("beta_a", "beta_b",
"sigma_a", "sigma_b"))],"  (",data[n,which(colnames(data) =="sd")], ") ", sep="")
latex.data[n,which(significance[n,] ==1)]<-
paste(latex.data[n,which(significance[n,] ==1)], " * ", sep="")
}
return(latex.data)
}
```

8.4 Appendix D: Other R codes

Before using the BIOGEME software, we coded in R a function which implements and maximizes the log-likelihood function of a logit model and a Logit Mixture model. In order to calculate the integral of the Logit Mixture probabilities, we tried first Monte Carlo methods and second numerical Gaussian Quadrature methods. The code of the implemented functions is given in the next sections.

When we used the R function which uses Monte Carlo methods, we could not recover the true values of the parameters, in particular, the variance of the random coefficients was poorly estimated. However, running the simulations using the Gaussian Quadrature functions lead to good parameters' estimations.

8.4.1 Monte Carlo Methods

Logit Model

```
#########################################################################
#------------------------------- Functions -----------------------
#########################################################################
"Logit.Lik" <- function(beta, a, b, choice)
{
V <- beta[1] * a + beta[2] * b
log.likelihood<- sum( diag(choice %*% t(V)) - log(apply(exp(V), 1, sum)) )
return(-1*log.likelihood)
}

"Optimization.L" <- function(choice, a, b)
{
return(optim(c(1,1), Logit.Lik,gr=NULL, a, b, choice, method="L-BFGS-B",
hessian=T, control=list(maxit=6000), lower=c(-Inf, -Inf), upper=c(Inf, Inf)))
}
```

Logit Mixture Model

```
#########################################################################
#------------------------------- Functions -----------------------
#########################################################################
"Mixture.Logit.Lik" <- function(theta, a, b, choice,
unif.draws.a, unif.draws.b, R=500, weights)
{
N <- nrow(choice)
beta <- matrix(NA, N, 2)
#L has R columns representing the number of time the experience is made and
```

```
#N rows which is equal to the number of obs/individuals
L <- matrix(NA, N, R)
P <- rep(NA, N)

for(r in 1:R){
##Two random coefficients
beta[,1] <- qnorm(unif.draws.a[,r], theta[1], abs(theta[2]))
beta[,2] <- qnorm(unif.draws.b[,r], theta[3], abs(theta[4]))

V <- beta[,1] * a + beta[,2] * b
#For a fixed column, that is for each experience, the r^th column of L
#contains the logit formula page 144 FOR ONLY THE CHOSEN ALTERNATIVE

L[,r] <- exp(diag(choice %*% t(V))) / apply(weights * exp(V), 1, sum)
}

P <- apply(L, 1, sum) / R

log.likelihood <- sum(log(P))
LOG.LIKELIHOOD <<- append(LOG.LIKELIHOOD, log.likelihood)
print(paste(log.likelihood,"-", (length(LOG.LIKELIHOOD)-1), "-", theta[1],
 "-",theta[2], "-", theta[3], "-", theta[4]))
return(-1*log.likelihood)
}

"Optimization.ML" <- function(choice, a, b, unif.draws.a, unif.draws.b, R)
{
return(optim(c(1,1,2,2), Mixture.Logit.Lik,gr=NULL, a, b, choice,
 unif.draws.a, unif.draws.b, R, method="L-BFGS-B", hessian=T,
 control=list(maxit=6000), lower=c(-Inf, -Inf, -Inf, -Inf),
 upper=c(Inf, Inf, Inf, Inf)))
}

#######################################################################
#------------------ Script of the Simulation -----------------------
#######################################################################
N <- 100
J <- 10
J1 <- 5

beta_a <- rnorm(N, 1, 1)
beta_b <- rnorm(N, 2,2)
```

```
a <- matrix(runif(N*J), N, J)
b <- matrix(runif(N*J), N, J)
epsilon <- matrix(rgumbel(N*J), N, J)

V <- beta_a * a + beta_b * b
U <- V + epsilon

choice <- Choice(U)

##---Gobal Variable to Store the likelihood value
LOG.LIKELIHOOD <<- rep(NA)
R <- 500
unif.draws.a <- matrix(runif((N*R)), N, R)
unif.draws.b <- matrix(runif((N*R)), N, R)

system.time(opt <- Optimization.ML(choice,a,b, unif.draws.a, unif.draws.b, R))

##--- Sampling of Alternatives and fitting the model with a reduced choice set
sampling <- Sampling(choice, a, b,epsilon, J1)
V.choice <- beta_a * sampling$a + beta_b * sampling$b
U.choice <- V.choice + sampling$epsilon

sampling.choice <- Choice(U.choice)

system.time(opt.sampling <- Optimization.ML(sampling.choice,sampling$a,sampling$b,
 unif.draws.a, unif.draws.b, R))
```

8.4.2 Gaussian Quadrature Methods

```
##--- Numerical integration: one random parameter ----------

#--- This Function is an implementation of the function L of the logit
#--- probabilities.I did a change of variable in order to use the Gauss
#--- approximation quadrature which is an of the integrale of a
#--- function L * the standard Gaussian Kernel.
#--- Hence, this function is an implementation of the Logit probability
#--- formula + change of variable.
#--- This function returns a vector which the n^th component is the
#--- estimation of the integral for the n^th decision maker.

"L" <- function(mu, sigma, beta, a, choice){
V <- sqrt(2) * sigma * (beta * a)
l <- exp(-mu^2/(2*sigma^2)) * exp(diag(choice %*% t(V))) /
```

```
apply(exp(V), 1, sum) * exp(beta * sqrt(2) * mu/ sigma)/ sqrt(pi)
return(l)
}

"Mixture.logit.lik.NI1" <- function(theta, a, choice){
#--- mu and sigma are the parameters of the normal distribution
mu <- theta[1]
sigma <- theta[2]
N <- nrow(choice)
#--- Gaussian quadrature: gives the nodes and the weights (standard gaussian
#--- distribution evaluated at the nodes)
GI <- gauss.quad(100, kind="hermite")
nodes <- GI$nodes
mesh <- length(nodes)
weights <- GI$weights
#--- l is a matrix which has "N" rows and "mesh" columns
#--- Each row represents a decision maker and contains the value of
#--- the function L in the integral, evaluated at the nodes.
l<- matrix(NA,N, mesh)
for(n in 1:mesh){
l[,n] <- L(mu, sigma, nodes[n], a, choice)
}

#--- probability  is a vector which, for each person "n", contains the estimation
#---  of the integral by the Gaussian quadrature method: inner product of
#--- weight and L both evaluated at nodes
proba <- rep(NA, N)
for(n in 1:N){
proba[n] <- sum(l[n,] * weights)
}
log.likelihood <- sum(log(proba))
LOG.LIKELIHOOD <<- append(LOG.LIKELIHOOD, log.likelihood)
print(paste(log.likelihood,"-", (length(LOG.LIKELIHOOD)-1), "-",
theta[1], "-",theta[2]))
return(-1*log.likelihood)
}

"Optimization.ML.NI1" <- function(choice, a)
{
return(optim(c(1,1), Mixture.logit.lik.NI1, gr=NULL, a, choice,
method="L-BFGS-B", hessian=T, control=list(maxit=6000),
lower=c(-5, 0.2), upper=c(5, 5)))
}
```

8.5 Appendix E: Plots' generation

In this section, we give a part of the code that generates the plots we did in this project. Let's denote by 'different.alt.models' the table or matrix that contains the fit results, the the code for the plots is given as follows

```
#---Plot 1: Evolution of Estimated Value of the Parameter wrt Nbre of Alternative
pdf("/Users/ines/Desktop/epfl stuff 2009-2010/mit-project/Latex Stuff/
Model Specification for Bierlaire/evolution_param_Nbre_alt.pdf",width=10,height=7)

x1<-seq(10,100,10)
xlab <- "Number of Alternatives"
x2=x1
different.alt.models<-diff_nbre_sampled_alt[,10:27]

op <- par(mfrow=c(2,2),mar=c(2,1.8,1,1),family="serif",cex=1.1)

#plot1: Beta_a
y1=different.alt.models[,2]
y2=rep(1,10)
plot(x2, y2, type='l',xlab='Number of Alternatives', ylab='mu_a Estimates',
 xlim=range(x2,x1), ylim=range(y2, y1+0.1), col="red")
points(x1, y1, type='b')

#legend("topleft",legend = "portfolio P&L",lty=1)
legend("topright",
#title = "P&L volatility, MAVG 250 days",
legend = c("True Value for mu_a",
"Biogeme Estimation of mu_a"),
col = c("red","black"),
lty=c(1,1),
lwd=c(1,1))


#plot2: Sigma_a
y1=different.alt.models[,6]
y2=rep(1,10)
plot(x2, y2, type='l',xlab='Number of Alternatives', ylab='sigma_a Estimates',
xlim=range(x2,x1), ylim=range(y2, y1+0.1), col="red")
points(x1, y1, type='b')

#legend("topleft",legend = "portfolio P&L",lty=1)
legend("topright",
```

```
#title = "P&L volatility, MAVG 250 days",
legend = c("True Value for sigma_a",
"Biogeme Estimation of sigma_a"),
col = c("red","black"),
lty=c(1,1),
lwd=c(1,1))


#plot3: Beta_b
y1=different.alt.models[,4]
y2=rep(2,10)
plot(x2, y2, type='l',xlab='Number of Alternatives', ylab='mu_b Estimates',
 xlim=range(x2,x1), ylim=range(y2, y1), col="red")
points(x1, y1, type='b')

#legend("topleft",legend = "portfolio P&L",lty=1)
legend("topright",
#title = "P&L volatility, MAVG 250 days",
legend = c("True Value for mu_b",
"Biogeme Estimation of mu_b"),
col = c("red","black"),
lty=c(1,1),
lwd=c(1,1))


#plot4: Sigma_b
y1=different.alt.models[,8]
y2=rep(2,10)
plot(x2, y2, type='l',xlab='Number of Alternatives', ylab='sigma_b Estimates',
xlim=range(x2,x1), ylim=range(y2, y1), col="red")
points(x1, y1, type='b')

#legend("topleft",legend = "portfolio P&L",lty=1)
legend("topright",
#title = "P&L volatility, MAVG 250 days",
legend = c("True Value for sigma_b",
"Biogeme Estimation of sigma_b"),
col = c("red","black"),
lty=c(1,1),
lwd=c(1,1))

par(op)
dev.off()
```

8.6 Appendix F: PBS BATCH file

In order to fit the Logit Mixture models with a large number of alternatives, we needed to run the simulations on a computer with a large RAM memory. For this study, we had access to a computer with 46 GB of memory. We had to install on it the new version of BIOGEME along with python 3.1. Once we generated the python script for the model and the .dat file, the next step consisted in creating a PBS script which is just a simple text file. We recall that PBS is a queueing system for submitting serial and parallel jobs. Our PBS script has a '.sh' extension and is given as follows:

```
1- #PBS -S /bin/bash
2- #PBS -l walltime=720:00:00,nodes=1:ppn=8
3- #PBS -l mem=46gb
4- #PBS -N test-sub
5- #PBS -q batch
6- #Import all current environment variables from submitting shell
7- #PBS -V
8- cd /home/azaiez
9- runbiogeme original-model-J100  data-J100.dat
```

The items included in the PBS batch file are described as follows:

- The first line specifies the shell interpreter we wish to use. In our case, we used /bin/bash
- The second line specifies the maximum amount of time we believe our job will run. In our case, we put 720 hours.
  The general syntax is given as follows: <hours>:<minutes>:<seconds>.
  The second part of line specifies the number of nodes and the number of processors per node we required for the simulation. In our case, we used 1 node and 8 processors per node. We figured out after trying different combinations of nodes and processors that our job can not be parallelized. In fact, if we use two nodes and eight processors per node for a total of 16 processors, we run out memory.
- The third line specifies the amount of memory required for our job. In our case, we needed the maximum amount of memory available that is 46 GB.
- The fourth line specify the name of our job. In this case, it is called 'test-sub'
- The fifth line specifies the queue we are submitting to. In our case, it is 'batch'
- The seventh line allows the environment variables to be visible from within our PBS batch script. For more details, we refer to [11] and [12].
- The eighth line specifies the home directory where the output files will be generated.

- The ninth line specifies the linux command to run the simulation using the new version of BIOGEME. The general syntax is:

  ```
  runbiogeme python_file_without_extension data.dat
  ```

We refer to [13] for any information about using the PBS batch files. Here are the most commonly used commands:

1. qsub file.sh: The qsub command is used to submit a job to the queuing system.
2. qstat -u user_name: The qstat command is used to check the job status.
3. qdel job_number: The qdel command is used to kill or remove the jobs from the queuing system.

Bibliography

[1] Ben Akiva M., Bierlaire M., Bolduc D., Walker J., 2009, *Discrete Choice Analysis.*

[2] Kenneth E. Train, 2009, *Discrete Choice Methods with Simulation.*

[3] Joan Leslie Walker, 2001, *Extended Discrete Choice Models: Integrated Framework, Flexible Error Structures, and Latent Variables*, PhD thesis, Massachusetts Institute of Technology.

[4] Lung-Fei Lee, 1997, *A simulated likelihood estimator for qualitative response models with sufficient statistics*, Economics Letters 57 (1997) 23-32.

[5] Daniel McFadden, 1978, *Modeling the choice of residential location*, In Transportation Research Record 673, TRB, National Research Council, Washington D.C.

[6] Tseng W. C, Mcconnell K , 2000, *Some Preliminary Evidence on Sampling of Alternatives with the Random Parameters Logit*, Marine Resource Economics, page 317-332.

[7] Nerlla S, Bhat C. R , 2004, *A numerical analysis of the effect of sampling of alternatives in discrete choice models*, March 2004.

[8] Chen Y, et al, 2005, *The estimation of Discrete choice models with large choice set*, Journal of the Eastern Asia Society for Transportation Studies, page 1724-1739

[9] Domanski A, 2009 *Estimating Mixed Logit Recreation Demand Models with Large choice Sets*, Selected Paper prepared for presentation at the Agricultural and Applied Economics, July 26-28 2009.

[10] McFadden D, Train K, 2000 *Mixed MNL Models for Discrete Response*, Journal of applied econometrics, page 447-470, 2000.

[11] http://rcsg.rice.edu/rcsg/shared/env.html

[12] http://www.clusters.umaine.edu/wiki/index.php/Running_a_Batch_Job

[13] http://www.msi.umn.edu/cgl/info/pbs/index.html