Estimation of the Mixed Logit Likelihood Function by Randomized Quasi-Monte Carlo

D. Munger[a], P. L'Ecuyer[a], F. Bastin[a], C. Cirillo[b], B. Tuffin[c]

[a]*Département d'informatique et de Recherche Opérationnelle, Université de Montréal, C.P. 6128 Succ. Centre-Ville, Montréal (QC), H3C 3J7, Canada. Members of CIRRELT.*
[b]*Department of Civil & Environmental Engineering, University of Maryland, College Park, MD 20742, USA.*
[c]*INRIA Rennes Bretagne-Atlantique, Campus de Beaulieu, 35042 Rennes cedex, France*

Abstract

We examine the effectiveness of randomized quasi-Monte Carlo (RQMC) techniques to estimate the integrals that express the discrete choice probabilities in a mixed logit model, for which no closed form formula is available. These models are used extensively in travel behavior research. We consider popular RQMC constructions such as randomized Sobol', Faure, and Halton points, but our main emphasis is on randomly-shifted lattice rules, for which we study how to select the parameters as a function of the considered class of integrands. We compare the effectiveness of all these methods and of standard Monte Carlo (MC) to reduce both the variance and the bias when estimating the log-likelihood function at a given parameter value. In our numerical experiments, randomized lattice rules (with carefully selected parameters) and digital nets are the best performers and they reduce the bias as much as the variance. With panel data, in our examples, the performance of all RQMC methods degrades rapidly when we simultaneously increase the dimension and the number of observations per individual.

1. Introduction

Travel behavior analysis makes heavy use of discrete choice models. Recent modeling frameworks account for a large number of (random) effects: heterogeneity in preferences (Hess et al., 2005; Cirillo and Axhausen, 2006) and/or in scale factor (Hess et al., 2009), variability in willingness to pay (Bastin et al., 2010), correlation across alternatives (Brownstone et al., 2000), in space (Bhat and Sener, 2009), etc. Advanced discrete choice models are often associated with

choice probabilities that can be written as multivariate integrals, but do not admit a closed-form formula. In mixed logit models, for example, we have an integral with respect to the mixing density, which depends on unknown parameters that we want to estimate. These integrals are typically estimated by Monte Carlo (MC) or quasi-Monte Carlo (QMC) methods (McFadden and Train, 2000; Train, 2003). In particular, Halton sequences have found widespread application for mixed logit model estimation in transportation (Train, 2000; Bhat, 2001, 2003). Sándor and Train (2004) experimented with digital nets and sequences such as those of Sobol'. Sivakumar et al. (2005) observed that Faure sequences performed better empirically than Halton sequences in the evaluation of the multidimensional integrals they considered.

In other areas of applications, for example in finance (L'Ecuyer, 2009), the best empirical results are usually obtained by Sobol' nets with certain types of scrambles and by randomly-shifted lattice rules with a baker's transformation. These methods have not been studied for estimating the integrals involved in mixed logit models and our primary aim in this paper is to fill this gap. A crucial issue in the application of lattice rules is the selection of their parameters, which should depend in principle on the considered class of integrands. Ideally, the parameters are selected to minimize a measure of discrepancy adapted to the problem, which gives a weight to each subset of coordinates, and those weights are chosen based on a functional analysis of variance (ANOVA) decomposition of the integrand (Dick et al., 2004; L'Ecuyer, 2009). We show in this paper how this methodology can be adapted to the setting of mixed logit log-likelihood estimation. This is not direct, because the log-likelihood is not expressed as an integral and we do not have an unbiased estimator of it, so we are not in a standard RQMC framework. We study both the bias and the variance of the log-likelihood estimator, and we examine and compare different ways of selecting the weights in the discrepancy. Given that computing the ANOVA for each application would be unpractical because it is too time consuming, we also compare the performance obtained with simplified weight-selection procedures. Fortunately, we find (empirically) that the rules obtained with simplified weight selections perform practically as well as those constructed from an extensive and costly ANOVA estimation, and are robust to changes in the model.

We focus in this paper on the estimation of the log-likelihood function at a given parameter value, for a given data set. This is the basic building block and the main source of error when estimating the parameter value that maximizes the log-likelihood function. Just examining the log-likelihood estimator permits one to avoid additional error sources from the maximization and to distinguish and

study separately the bias and the variance. Besides lattice rules and standard MC, we consider RQMC point sets constructed from the Sobol', Faure, and Halton sequences. We find that RQMC reduces both the variance and the simulation bias, and we explain why.

The remainder is organized as follows. In Section 2, we present basic definitions and properties of the mixed logit model considered in this paper, and we analyze the bias and variance of an MC estimator of the log-likelihood. In Section 3, we review briefly RQMC methods, with special focus on randomly-shifted lattice rules and the choice of their parameters, and we develop a methodology of parameter selection for the present setting, based on an ANOVA and a weighted discrepancy. In Section 4, we report numerical experiments that examine and compare the convergence of the bias and variance for various RQMC estimators, first for synthetic data (with 1 to 10 observations per individual), then for a real-life data set. We also report an experiment on parameter estimation (log-likelihood maximization) with the real-life data. Section 5 concludes the paper.

2. The mixed logit model and log-likelihood estimation

2.1. The model and the likelihood function

In a popular form of *multinomial mixed logit* model (McFadden and Train, 2000; Train, 2003) the utility of alternative j for individual q is

$$U_{q,j} = \boldsymbol{\beta}_q^{\mathsf{t}} \mathbf{x}_{q,j} + \epsilon_{q,j} = \sum_{\ell=1}^{s} \beta_{q,\ell} x_{q,j,\ell} + \epsilon_{q,j}$$

where $\boldsymbol{\beta}_q = (\beta_{q,1}, \ldots, \beta_{q,s})^{\mathsf{t}}$ is an unobserved random vector of taste parameters (or coefficients) for each individual q, $\mathbf{x}_{q,j} = (x_{q,j,1}, \ldots, x_{q,j,s})^{\mathsf{t}}$ gives the observed attributes for choice j and individual q, and the $\epsilon_{q,j}$ are independent Gumbel random variables with mean 0 and scale factor of 1, which represent unobserved random noise. For a random individual q, the (random) vector $\boldsymbol{\beta}_q$ has a multivariate density $f_{\boldsymbol{\theta}}$ that depends on a parameter vector $\boldsymbol{\theta}$. Individual q always selects the alternative j having the largest utility $U_{q,j}$.

Conditional on $\boldsymbol{\beta}_q$, the individual q selects alternative j with probability

$$L_q(j, \boldsymbol{\beta}_q) = \frac{\exp[\boldsymbol{\beta}_q^{\mathsf{t}} \mathbf{x}_{q,j}]}{\sum_{a \in \mathcal{A}(q)} \exp[\boldsymbol{\beta}_q^{\mathsf{t}} \mathbf{x}_{q,a}]}, \tag{1}$$

independently of other individuals, where $\mathcal{A}(q)$ is the set of his alternatives. The unconditional probability that a random individual selects alternative j is then

$$p_q(j, \boldsymbol{\theta}) = \mathbb{E}[L_q(j, \boldsymbol{\beta}_q)] = \int_{\mathbb{R}^s} L_q(j, \boldsymbol{\beta}) f_{\boldsymbol{\theta}}(\boldsymbol{\beta}) \, \mathrm{d}\boldsymbol{\beta}. \tag{2}$$

We assume that the random vector $\boldsymbol{\beta}_q$ can be written as $\boldsymbol{\beta}_q = h(\boldsymbol{\theta}, \mathbf{U})$ for some explicit function h, where $\mathbf{U}$ is a vector of independent uniform random variables over $(0, 1)$. This assumption is standard and is required to simulate realizations of $\boldsymbol{\beta}_q$ from independent uniform random numbers. In the examples considered in this paper, $\mathbf{U}$ has dimension s. Then we have

$$p_q(j, \boldsymbol{\theta}) = \mathbb{E}\left[L_q(j, h(\boldsymbol{\theta}, \mathbf{U}))\right] = \int_{(0,1)^s} L_q(j, h(\boldsymbol{\theta}, \mathbf{u})) \, \mathrm{d}\mathbf{u}. \tag{3}$$

To estimate the integral (2) or (3) by MC, for a given $\boldsymbol{\theta}$ and q, we generate n_q independent random points $\mathbf{U}_q^{(1)}, \ldots, \mathbf{U}_q^{(n_q)}$ in $(0, 1)^s$, put $\boldsymbol{\beta}_q^{(i)}(\boldsymbol{\theta}) = h(\boldsymbol{\theta}, \mathbf{U}_q^{(i)})$ for $i = 1, \ldots, n_q$, and compute the unbiased estimator

$$\hat{p}_q^{n_q}(j, \boldsymbol{\theta}) = \frac{1}{n_q} \sum_{i=1}^{n_q} L_q(j, \boldsymbol{\beta}_q^{(i)}(\boldsymbol{\theta})) = \frac{1}{n_q} \sum_{i=1}^{n_q} L_q(j, h(\boldsymbol{\theta}, \mathbf{U}_q^{(i)})), \tag{4}$$

where each $L_q(j, \boldsymbol{\beta}_q^{(i)}(\boldsymbol{\theta}))$ is computed via (1).

When the same individual (with the same $\boldsymbol{\beta}_q$) delivers $T_q > 1$ observations and selects alternative j_t for his tth decision, assuming that the selections are independent conditionally on $\boldsymbol{\beta}_q$, the joint likelihood of this choice sequence is

$$L_q^{T_q}(j_1, \ldots, j_{T_q}, \boldsymbol{\beta}_q) = \prod_{t=1}^{T_q} L_q(j_t, \boldsymbol{\beta}_q),$$

where each $L_q(j_t, \boldsymbol{\beta}_q)$ is computed as in (1), and its unconditional probability is

$$p_q(j_1, \ldots, j_{T_q}, \boldsymbol{\theta}) = \mathbb{E}\left[L_q^{T_q}(j_1, \ldots, j_{T_q}, h(\boldsymbol{\theta}, \mathbf{U}))\right].$$

This expectation can be estimated by MC in a manner similar to (4), with each $L_q(\cdot)$ replaced by a product $L_q^{T_q}(\cdot)$.

2.2. Estimating the log-likelihood of a sample: bias and variance

With (4), we can estimate the likelihood of $\boldsymbol{\theta}$ for a single individual. But to estimate $\boldsymbol{\theta}$ from a data set, one would usually maximize the *log-likelihood* $\ln L(\boldsymbol{\theta})$ (the logarithm of the likelihood) of the entire data set, as a function of $\boldsymbol{\theta}$, and we do not have an unbiased estimator for this function. Specifically, suppose we have a data set of one observation per individual for m individuals, in which individual q was given the vector of attributes $\mathbf{x}_{q,j}$ for each alternative j and made the choice y_q, for $q = 1, \ldots, m$. The log-likelihood, which we divide by m to obtain an average over individuals and prevent the expression from blowing up when $m \to \infty$, is

$$\frac{\ln L(\boldsymbol{\theta})}{m} = \frac{1}{m} \ln \prod_{q=1}^{m} p_q(y_q, \boldsymbol{\theta}) = \frac{1}{m} \sum_{q=1}^{m} \ln p_q(y_q, \boldsymbol{\theta}). \tag{5}$$

Replacing the $p_q(y_q, \boldsymbol{\theta})$ by their estimators $\hat{p}_q^{n_q}(y_q, \boldsymbol{\theta})$ defined in (4) yields the following estimator of $\ln L(\boldsymbol{\theta})/m$:

$$\frac{\ln(\hat{L}(\boldsymbol{\theta}))}{m} = \frac{1}{m} \sum_{q=1}^{m} \ln\left(\hat{p}_q^{n_q}(y_q, \boldsymbol{\theta})\right) = \frac{1}{m} \sum_{q=1}^{m} \ln\left(\frac{1}{n_q} \sum_{i=1}^{n_q} L_q\left(y_q, h(\boldsymbol{\theta}, \mathbf{U}_q^{(i)})\right)\right). \tag{6}$$

When $T_q > 1$, the $L_q(\cdot)$ inside the sum is replaced by the product $L_q^{T_q}(\cdot)$. This estimator is biased, because $\ln$ is not a linear function, and the bias is negative because $\ln$ is concave. To find the dominant terms of the bias and variance, let

$$R_q = \frac{\hat{p}_q^{n_q}(y_q, \boldsymbol{\theta}) - p_q(y_q, \boldsymbol{\theta})}{p_q(y_q, \boldsymbol{\theta})}, \tag{7}$$

the relative estimation error in $p_q(y_q, \boldsymbol{\theta})$. A Taylor expansion of $\ln(\hat{p}_q^{n_q}(y_q, \boldsymbol{\theta}))$ around $\ln(p_q(y_q, \boldsymbol{\theta}))$ gives

$$\ln(\hat{p}_q^{n_q}(y_q, \boldsymbol{\theta})) - \ln(p_q(y_q, \boldsymbol{\theta})) \quad = \quad R_q - R_q^2/2 + \mathcal{O}(|R_q|^3). \tag{8}$$

The total bias in (6) can then be written (using that $\mathbb{E}[R_q] = 0$) as

$$\mathbb{E}\left[\frac{\ln(\hat{L}(\boldsymbol{\theta})) - \ln(L(\boldsymbol{\theta}))}{m}\right] = \frac{1}{m} \sum_{q=1}^{m} \mathbb{E}\left[-\frac{R_q^2}{2} + \mathcal{O}(|R_q|^3)\right] \approx -\frac{1}{2m} \sum_{q=1}^{m} \mathbb{E}[R_q^2], \tag{9}$$

and, since $\mathrm{Var}[R_q] = \mathbb{E}[R_q^2]$ and the R_q's are independent, the variance is

$$\mathrm{Var}\left[\frac{\ln(\hat{L}(\boldsymbol{\theta}))}{m}\right] \approx \mathrm{Var}\left[\frac{1}{m}\sum_{q=1}^{m} R_q\right] = \frac{1}{m^2}\sum_{q=1}^{m}\mathbb{E}[R_q^2]. \tag{10}$$

With MC, $\mathbb{E}[R_q^2] = \mathcal{O}(1/n_q)$, so if $n_q = n$ for all q, the bias is $\mathcal{O}(1/n)$ and the variance is $\mathcal{O}(1/(mn))$. Thus, for fixed m, the contribution of the square bias to the mean square error (MSE) becomes negligible compared to that of the variance when n is large enough: $\mathcal{O}(n^{-2})$ compared with $\mathcal{O}((mn)^{-1})$. But in practice, n is not always very large, so the bias can be significant, and it is not reduced when we increase m, in contrast to the variance. For this reason, it is important to study the convergence of the bias in addition to that of the variance when we compare MC and RQMC methods. With RQMC, the terms $\mathbb{E}[R_q^2]$ and the convergence rate as a function of n are not the same as with MC. One may expect from (7) that $\mathbb{E}[R_q^2]$ is larger when p_q is smaller, and this is typically what we have observed empirically: the individuals with small p_q's tend to contribute more to the sums in (9) and (10).

3. RQMC Methods

3.1. RQMC and variance bounds

RQMC methods (Owen, 1998; L'Ecuyer, 2009; Lemieux, 2009) estimate the integral $\mu = \int_{(0,1)^s} f(\mathbf{u})\,\mathrm{d}\mathbf{u}$ of a function f over the s-dimensional unit cube $(0,1)^s$ by averaging evaluations of f over a set of n points $P_n = \{\mathbf{U}_0, \ldots, \mathbf{U}_{n-1}\}$:

$$\hat{\mu}_{n,\mathrm{rqmc}} = \frac{1}{n}\sum_{i=0}^{n-1} f(\mathbf{U}_i). \tag{11}$$

These points must form an RQMC point set, which means that

(a) P_n covers $(0,1)^s$ very uniformly when taken as a set and
(b) each individual point $\mathbf{U}_i$ has the uniform distribution over $(0,1)^s$.

Condition (b) ensures that the average $\hat{\mu}_{n,\mathrm{rqmc}}$ is an unbiased estimator of μ. With condition (a), we expect the RQMC estimator (11) to have smaller variance than the ordinary MC estimator, if f is sufficiently smooth. Different ways of measuring the uniformity of P_n in (a) are used for different types of constructions and classes of integrands, to obtain error and variance bounds (Niederreiter, 1992; Hickernell, 1998, 2000; L'Ecuyer, 2009). For this, one usually specifies a class

$\mathcal{H}$ of functions f, often a reproducing kernel Hilbert space, and one derives a worst-case bound on the integration error of the form

$$|\hat{\mu}_{n,\text{rqmc}} - \mu| \leq D(P_n)V(f) \tag{12}$$

for all $f \in \mathcal{H}$ and any point set $P_n \subset (0,1)^s$, where $D(P_n)$ measures the discrepancy of P_n from the uniform distribution and $V(f) = \|f - \mu\|_{\mathcal{H}}$ measures the variability of f in $\mathcal{H}$. The definitions of $D(P_n)$ and $V(f)$ depend on each other, and a definition that makes $V(f)$ smaller will generally make $D(P_n)$ larger, and vice-versa. One special case of (12) is the classical Koksma-Hlawka inequality often associated with QMC methods, where P_n is deterministic (Niederreiter, 1992). When P_n is randomized, we obtain the variance bound

$$\text{Var}[\hat{\mu}_{n,\text{rqmc}}] = \mathbb{E}[(\hat{\mu}_{n,\text{rqmc}} - \mu)^2] \leq \mathbb{E}[D^2(P_n)]\, V^2(f). \tag{13}$$

Then, if $V(f) < \infty$, the variance converges at the same rate as $\mathbb{E}[D^2(P_n)]$ as a function of n. RQMC point sets used in this paper include randomized lattices, digital nets, and Halton points. They are discussed in the remainder of this section.

3.2. Functional ANOVA decomposition

Covering the s-dimensional unit hypercube very uniformly requires a number of points n that increases exponentially with s, so accurate high-dimensional integration by RQMC looks hopeless at first sight (L'Ecuyer, 2009). Yet empirically, the method works well even in hundreds of dimensions in some cases. The usual explanation is that in those cases, the integrand f can be well approximated by a sum of low-dimensional functions that are accurately integrated by RQMC, and the residual has small variance (Owen, 1998; L'Ecuyer, 2009). This can be formalized via the *functional ANOVA decomposition* of f, defined as follows. If

$$\sigma^2 = \text{Var}[f(\mathbf{U})] = \int_{(0,1)^s} f(\mathbf{u})\, \mathrm{d}\mathbf{u} - \mu^2 < \infty$$

for $\mathbf{U}$ uniformly distributed over $(0,1)^s$, one can write

$$f(\mathbf{u}) = \mu + \sum_{\emptyset \neq \mathfrak{u} \subseteq \{1,\ldots,s\}} f_{\mathfrak{u}}(\mathbf{u}) \tag{14}$$

where $\mathfrak{u}$ denotes an arbitrary subset of coordinates (this is standard notation, not to be confused with $\mathbf{u}$), each $f_{\mathfrak{u}} : (0,1)^s \to \mathbb{R}$ depends only on $\{u_i,\ i \in \mathfrak{u}\}$, the $f_{\mathfrak{u}}$'s

integrate to zero and are orthogonal, and the variance admits the corresponding decomposition $\sigma^2 = \sum_{\mathfrak{u}\subseteq\{1,\dots,s\}} \sigma_{\mathfrak{u}}^2$ where $\sigma_{\mathfrak{u}}^2 = \mathrm{Var}[f_{\mathfrak{u}}(\mathbf{U})]$.

If $\sum_{\mathfrak{u}\in\mathcal{U}} \sigma_{\mathfrak{u}}^2/\sigma^2$ is very close to 1 for a relatively small set $\mathcal{U}$ of subsets of $\{1,\dots,s\}$, that is, the approximation of f by $\sum_{\mathfrak{u}\in\mathcal{U}} f_{\mathfrak{u}}$ accounts for most of the MC variance, then we can construct the RQMC point set P_n by focusing on the uniformity of its projections over the subsets of coordinates $\mathfrak{u} \in \mathcal{U}$, by giving them an importance in relation with $\sigma_{\mathfrak{u}}^2$, and neglect the other projections. This is how most good RQMC point sets are constructed, either by assuming a priori a certain behavior for the $\sigma_{\mathfrak{u}}^2$ or by trying to estimate them.

3.3. Randomly-shifted lattice rules

A *rank-1 lattice rule* with n points in s dimensions is defined as follows (Niederreiter, 1992; Sloan and Joe, 1994). Select a vector $\mathbf{a}_1 = (a_1,\dots,a_s)$ whose coordinates belong to $\mathbb{Z}_n = \{0,\dots,n-1\}$, let $\mathbf{v}_1 = (v_1,\dots,v_s) = \mathbf{a}_1/n$, and define $P_n^0 = \{\mathbf{v} = i\mathbf{v}_1 \bmod 1,\ i = 0,1,\dots,n-1\}$, where the division and the modulo operation are coordinate-wise. This point set is the intersection of a lattice with the unit hypercube in s dimensions. The a_j are usually taken relatively prime to n, so that the projection of P_n^0 over any of its coordinates contains the n distinct points $\{0, 1/n, \dots, (n-1)/n\}$. Thus, there is no need to measure the uniformity of the one-dimensional projections. Here we randomize P_n^0 by applying a random shift modulo 1, which consists in generating a single point $\mathbf{U}$ uniformly over $(0,1)^s$ and adding it to each point of P_n^0, modulo 1, coordinate-wise (Cranley and Patterson, 1976; L'Ecuyer and Lemieux, 2000), to obtain P_n. We follow this by a baker's transformation, which replaces each coordinate u of each point by $2u$ if $u < 1/2$ and by $2-2u$ otherwise (Hickernell, 2002; L'Ecuyer, 2009). To summarize, the following pseudocode enumerates the randomized points:

Generate $\mathbf{U} = (U_1,\dots,U_s)$ from the uniform distribution over $(0,1)^s$;
for $i = 0$ **to** $n-1$ **do**
    for $j = 1$ **to** s **do**
        $U_{i,j} \leftarrow 2((iv_j + U_j) \bmod 1)$;
        if $U_{i,j} \geq 1$ **then** $U_{i,j} \leftarrow 2 - U_{i,j}$;
    $\mathbf{U}_i = (U_{i,1},\dots,U_{i,s})$.

The vector $\mathbf{v}_1$ is selected to try to minimize a given discrepancy measure of P_n^0. In this paper, we use the weighted $\mathcal{P}_{2\alpha}$ criterion (Dick et al., 2004)

$$\mathcal{P}_{2\alpha}(P_n^0) = \sum_{\emptyset\neq\mathfrak{u}\subseteq\{1,\dots,s\}} \gamma_{\mathfrak{u}}\mathcal{P}_{2\alpha,\mathfrak{u}}(P_n^0), \tag{15}$$

where α is a positive integer, the *projection-dependent weights* $\gamma_{\mathfrak{u}}$ are arbitrary positive real numbers, and

$$\mathcal{P}_{2\alpha,\mathfrak{u}}(P_n^0) = \frac{1}{n}\sum_{i=0}^{n-1}\left[\frac{-(-4\pi^2)^\alpha}{(2\alpha)!}\right]^{|\mathfrak{u}|}\prod_{j\in\mathfrak{u}} B_{2\alpha}(u_{i,j}) \tag{16}$$

is the discrepancy for the projection over $\mathfrak{u}$, in which $\mathbf{u}_i = (u_{i,1},\ldots,u_{i,s}) = i\mathbf{v}_1 \bmod 1$ is the ith lattice point before the shift, $|\mathfrak{u}|$ is the cardinality of $\mathfrak{u}$, and $B_{2\alpha}$ is the Bernoulli polynomial of order 2α. This criterion can be motivated as follows. Consider the class $\mathcal{F}_\alpha$ of functions f for which for each subset $\mathfrak{u}$ of coordinates, the partial derivative of order α with respect to these coordinates is square integrable, and the partial derivatives of orders 0 to $\alpha-2$ of the periodic continuation of f over $\mathbb{R}^s$ are continuous. For $\alpha = 1$, this continuity condition just disappears, but for $\alpha = 2$ the periodic continuation of f must be continuous. It turns out that the square variation of $f \in \mathcal{F}_\alpha$ defined as

$$V^2(f) = \sum_{\mathfrak{u}\subseteq\{1,\ldots,s\}} V^2(f_{\mathfrak{u}}) = \sum_{\emptyset\neq\mathfrak{u}\subseteq\{1,\ldots,s\}} \frac{1}{\gamma_{\mathfrak{u}}(4\pi^2)^{\alpha|\mathfrak{u}|}}\int_{[0,1]^{|\mathfrak{u}|}}\left|\frac{\partial^{\alpha|\mathfrak{u}|}}{\partial\mathbf{u}^\alpha} f_{\mathfrak{u}}(\mathbf{u})\right|^2 d\mathbf{u}, \tag{17}$$

corresponds to a square discrepancy for which $\mathbb{E}[D^2(P_n)] = \mathcal{P}_{2\alpha}(P_n^0)$, and the variance bound (13) holds for this pair (Dick et al., 2004; L'Ecuyer, 2009). It is also known that for any $\alpha > 1$, any $\delta > 0$, and any choices of weights $\gamma_{\mathfrak{u}}$, there exists a sequence of rank-1 lattices for which $\mathcal{P}_{2\alpha}(P_n^0) = \mathcal{O}(n^{-2\alpha+\delta})$, and the corresponding vectors $\mathbf{v}_1$ can be constructed explicitly one coordinate at a time, by a so-called *component-by-component* (CBC) construction method (Dick et al., 2004). This means that for any $f \in \mathcal{F}_\alpha$, it is possible to construct lattice rules for which $\mathrm{Var}[\hat{\mu}_{n,\mathrm{rqmc}}] = \mathcal{O}(n^{-2\alpha+\delta})$. The role of the baker's transformation mentioned earlier is to make the periodic condition of f continuous, so we can have $\alpha = 2$ instead of $\alpha = 1$ when f is smooth enough; see Hickernell (2002) and L'Ecuyer (2009) for detailed explanations.

3.4. Selecting the weights for lattice rules

Ideally, we would like to select the weights $\gamma_{\mathfrak{u}}$ in (15) so that minimizing this expression also minimizes the MSE of the log-likelihood estimator. In principle, we could either select a different set of weights and a different lattice for each individual, trying to minimize the MSE of the log-likelihood for that individual, or construct a single lattice and use it for all individuals, with independent random

shifts across individuals. We adopt the second option because it is much more practical and convenient. We nevertheless explored the potential MSE improvement that could be achieved with the first option, by some empirical experiments, and the gain was very small (see Section 4.3).

Note that for classical deterministic QMC methods, it is customary to use a different point set for each individual, otherwise, the "positive dependence" across individuals typically increases the error on the average. With RQMC, we can also do it, but we do not have to, because the independent randomizations remove the dependence (L'Ecuyer and Lemieux, 2000).

Both the bias and the variance in (9) and (10) depend on the sum $\sum_{q=1}^{m} \mathrm{Var}[R_q]$, which we would like to minimize. Under the assumption that the integrands L_q are sufficiently smooth to belong to $\mathcal{F}_\alpha$, from the end of Subsection 3.3 and (13), we have the bound

$$\mathrm{Var}[R_q] \le \sum_{\emptyset \ne \mathfrak{u} \subseteq \{1,\dots,s\}} \gamma_{q,\mathfrak{u}} \mathcal{P}_{2\alpha,\mathfrak{u}}(P_n^0) \tag{18}$$

for some per-individual projection-dependent weights $\gamma_{q,\mathfrak{u}}$, where $\mathcal{P}_{2\alpha,\mathfrak{u}}(P_n^0)$ is given by (16). Summing up over q gives

$$\sum_{q=1}^{m} \mathrm{Var}[R_q] \le \sum_{\emptyset \ne \mathfrak{u} \subseteq \{1,\dots,s\}} \gamma_{\mathfrak{u}} \mathcal{P}_{2\alpha,\mathfrak{u}}(P_n^0), \tag{19}$$

where

$$\gamma_{\mathfrak{u}} = \sum_{q=1}^{m} \gamma_{q,\mathfrak{u}}. \tag{20}$$

Our criterion will be the right sum in (19). It remains to estimate the appropriate $\gamma_{q,\mathfrak{u}}$'s. For that, we will use the fact (L'Ecuyer and Munger, 2011) that the sum in (18) is equal to the RQMC variance for a worst-case function $f^*_{\alpha,q}$ defined by

$$f^*_{\alpha,q}(\mathbf{u}) = \sum_{\emptyset \ne \mathfrak{u} \subseteq \{1,\dots,s\}} \sqrt{\gamma_{q,\mathfrak{u}}} \prod_{j \in \mathfrak{u}} \frac{(2\pi)^\alpha}{\alpha!} B_\alpha(u_j),$$

whose ANOVA variance components are

$$\sigma^2_{q,\mathfrak{u}} = \gamma_{q,\mathfrak{u}} \left[\mathrm{Var}[B_\alpha(U)] \frac{(4\pi^2)^\alpha}{(\alpha!)^2} \right]^{|\mathfrak{u}|} = \gamma_{q,\mathfrak{u}} \left[|B_{2\alpha}(0)| \frac{(4\pi^2)^\alpha}{(2\alpha)!} \right]^{|\mathfrak{u}|} \stackrel{\text{def}}{=} \gamma_{q,\mathfrak{u}} (\kappa(\alpha))^{-|\mathfrak{u}|} \tag{21}$$

where $\kappa(\alpha)$ is a constant that depends on α. In particular, we have $\kappa(1) = 3/\pi^2 \approx 0.30396$ and $\kappa(2) = 45/\pi^4 \approx 0.46197$. Acting as if we were integrating this $f^*_{\alpha,q}$, we adopt weights $\gamma_{q,\mathfrak{u}}$ given by these formulas, in which the $\sigma^2_{q,\mathfrak{u}}$ are replaced by estimates of the ANOVA components of $\mathrm{Var}[R_q]$. This amounts to taking the weights $\gamma_{\mathfrak{u}} = (\kappa(\alpha))^{|\mathfrak{u}|}\bar{\sigma}^2_{\mathfrak{u}}$ where $\bar{\sigma}^2_{\mathfrak{u}} = (1/m)\sum_{q=1}^{m}\sigma^2_{q,\mathfrak{u}}$. In our experiments, we estimated those $\sigma^2_{q,\mathfrak{u}}$ using the algorithm of Sobol' and Myshetskaya (2007) and we used $\alpha = 2$ (because we apply the baker's transformation) to compute the weights $\gamma_{q,\mathfrak{u}}$, then $\gamma_{\mathfrak{u}}$. In our results, we denote the lattices constructed based on $\mathcal{P}_{2\alpha}(P_n^0)$ with those weights by *lattice-*$\gamma_{\mathfrak{u}}$.

The lattice-$\gamma_{\mathfrak{u}}$ are our best shot at building lattices adapted to our problems. However, estimating all those weights becomes impractical when s increases, because there are $2^s - 1$ variance components to estimate. Note that the weights of one-dimensional projections (for which $|\mathfrak{u}| = 1$) are irrelevant for selecting $\mathbf{v}_1$, because all one-dimensional projections are the same under our assumptions, so there is no need to specify them. This gives s fewer parameters to estimate. Also, multiplying all weights by a given constant has no impact on the selection of $\mathbf{v}_1$, since it does not change the relative importance of the projections. To further reduce the number of parameters to estimate (and at the same time reduce the likelihood of overfitting), we may bundle the projections $\mathfrak{u}$ in subgroups, and attach one weight to each subgroup. For example, we can have *order-dependent weights*, where all projections of cardinality (or order) r are given the weight γ_r, for $r = 2, \ldots, s$. For this, we can estimate $\sigma_r^2 = \sum_{\{\mathfrak{u}:|\mathfrak{u}|=r\}}\bar{\sigma}^2_{\mathfrak{u}}$, and plug it in the formula $\gamma_r = C(\kappa(\alpha))^r\sigma_r^2/\binom{s}{r}$, where C is an arbitrary positive constant and $\binom{s}{r}$ is the number of projections of order r. This gives $s-1$ parameters to estimate. In our results, we will use *lattice-order* to refer to these rules.

To reduce this number even further, we can simply assume that $\gamma_r = C\gamma^r$ for all r, for some constant γ, and estimate a γ that best fits this model, for instance by fitting a linear regression model of the form

$$r \ln \kappa(\alpha) + 2 \ln \sigma_r = \ln C + r \ln \gamma + \epsilon_r$$

by finding C and γ that minimize $\sum_{r=2}^{\infty}\epsilon_r^2$. We call the resulting weights *geometric order-dependent weights*. In our experiments, we will end up taking either a best fit for γ, or the fixed values $\gamma = 0.1$ or $\gamma = 0.5$, and we refer to the corresponding rules by *lattice-*γ, *lattice-0.1* and *lattice-0.5*. Note that multiplying γ by some factor here is equivalent to multiplying $\kappa(\alpha)$ in (21) by the same factor.

We also considered for comparison some Korobov lattices tabulated by L'Ecuyer and Lemieux (2000), whose parameters were selected based on the $M_{32,24,16,12}$

criterion defined in that paper, and which accounts for projections in up to 32 dimensions over successive coordinates, and a selected set of projections of order 2, 3, and 4 only. This criterion also takes the worst case over the selected projections instead of a weighted average as in (15). These point sets were constructed with no particular application in mind. We refer to them by *lattice-*$M32$.

3.5. Sobol' and Faure nets

Randomized digital nets, which include Sobol' and Faure nets among others, form another important class of RQMC constructions. A Sobol' net with $n = 2^k$ points in s dimensions, for a positive integer k, contains the first n points of a Sobol' sequence in s dimensions (Sobol', 1967; Lemieux, 2009). These points are defined by $w \times k$ binary *generator matrices* $\mathbf{C}_1, \ldots, \mathbf{C}_s$. To define the ith point $\mathbf{u}_i$, for $i = 0, \ldots, 2^k - 1$, we write the digital expansion of i in base 2 and multiply the vector of its digits by $\mathbf{C}_j$, modulo 2, to obtain the first w digits of the binary expansion of the jth coordinate of $\mathbf{u}_i$. The columns of the matrices $\mathbf{C}_j$ are determined by direction numbers that must be chosen. For our experiments here, we have used the default direction numbers of SSJ (L'Ecuyer, 2008), taken from Lemieux et al. (2004), which improve upon previous proposals. To randomize our Sobol' nets, we first apply a left matrix scramble, which multiplies each $\mathbf{C}_j$ (modulo 2) by a lower triangular binary matrix $\mathbf{M}_j$ with 1's on the diagonal and random bits below the diagonal (Owen, 2003). This is followed by a digital random shift, which generates a single point $\mathbf{U}$ uniformly over $(0, 1)^s$ and adds each digit of the binary expansion of each of its coordinate to the corresponding digit of each point, modulo 2 (Owen, 2003).

The randomized Faure nets are defined in a similar way, except that all binary operations are replaced by operations modulo a prime integer b (the base), and the matrices $\mathbf{C}_j$ are defined in a specific way, with elements in $\{0, \ldots, b - 1\}$ (Lemieux, 2009). One must have $b \geq s - 1$ and n a power of b for the method to be effective. These requirements are a problem when s is large.

3.6. Halton points

We also try P_n defined as the first n points of randomized versions of the Halton sequence in s dimensions, using again the same points with independent randomizations across individuals. These Halton points are popular in discrete choice modeling and analysis. Faure and Lemieux (2009) provide a recent extensive study of various scrambles and randomizations for the Halton sequence, and propose a new scrambling that compares favorably with all the previous ones when combined either with a digital random shift, or with a random starting point

as proposed in Struckmeier (1995) and discussed in Ökten (2009). These two randomizations yield very similar performances in their results. We implemented these scrambled Halton points with a random starting point, and we refer to them as *Halton-FL points*. We also try a simpler randomization of the Halton points, by random shifts modulo 1, and we refer to these points as *Halton-shift points*. We also experimented with the more traditional practice of defining P_n for individual q as the points $(q-1)n-1$ through qn of the Halton sequence, i.e., individual 1 has the first n points, individual 2 has the next n points, and so on (Train, 2000; Bhat, 2001, 2003). This strategy is appropriate when using deterministic QMC sequences, but not needed when a random shift is applied to obtain an unbiased RQMC estimator, as we do here. In all our experiments, the variance and bias are almost undistinguishable from the case where we randomize the same points for all individuals, so we do not report the detailed results.

4. Numerical experiments

4.1. General experimental setting

We report experimental results for two examples, for which we estimate and compare the bias and variance of various RQMC estimators of the log-likelihood averaged over individuals. The first example uses artificially generated data and has multiple variants. The second one is based on real-life data. For each example, we select a parameter value $\boldsymbol{\theta}$ not far from the optimizer of the log-likelihood. To select the lattice rule parameters based on our methodology, we first estimate the relative variance components $\sigma^2_{q,\mathfrak{u}}$ in the ANOVA decompositions of the integrand $f_q(\mathbf{u}) = L_q(y_q, h(\boldsymbol{\theta}, \mathbf{u}))$ associated with (4), for each q and each $\mathfrak{u}$, or its extension to multiple choices per individual, using 100 independent RQMC runs with 65537 lattice points, for each $\mathfrak{u}$. Then we compute the weights $\gamma_{q,\mathfrak{u}}$, $\gamma_{\mathfrak{u}}$, γ_r, and γ^r based on those estimates of the $\bar{\sigma}^2_{\mathfrak{u}}$'s as explained earlier, and we construct lattices having a small value of the weighted $\mathcal{P}_{2\alpha}$ criterion (15), for each form of weights. These lattices are used for all individuals. For some examples, we also construct a specialized lattice based on the specific weights $\gamma_{q,\mathfrak{u}}$ for each individual, just to see if this could bring a significant gain in performance.

For MC and each RQMC method, for each n considered, we compute 1000 independent realizations of the estimator (6) of (5), then compute the empirical mean and variance of these realizations. We estimate the bias using the approximation in (9), based on the 1000 independent realizations of (4) for each q. For MC, the lattice rules, and the Halton points, we consider 28 prime values of n ranging from 31 to 16381 (including values close to a power of two).

For the Sobol' nets, we consider for n all powers of two from $n = 2^5 = 32$ to $n = 2^{14} = 16384$, and match each one with its nearest prime n for the lattice rules. For the Faure nets, for $s = 5$ we take $b = 5$ and $n = 25$, 125, 625, 3125, and 15625, while for $s = 10$ we take $b = 11$ and $n = 121$, 1331 and 14641.

Based on (9) and (10), the variance and the square bias of (6) with RQMC should behave as $\mathrm{Var}[\ln(\hat{L}(\boldsymbol{\theta}))/m] \approx V_0 m^{-1} n^{-\nu}$ and $\mathrm{Bias}^2[\ln(\hat{L}(\boldsymbol{\theta}))/m] \approx (V_0/2)^2 n^{-2\nu}$, for large enough n, for constants V_0 and ν that depend on the RQMC method. Thus, the square bias dominates the MSE for small n and large m, and is negligible compared with the variance when n is large enough. Thus, the MSE decreases roughly as $\mathcal{O}(n^{-2\nu})$ for small n and $\mathcal{O}(n^{-\nu})$ for n large.

In fact, adding another term to the variance expansion adds another term whose convergence order is comparable to that of the square bias in the MSE approximation. To take this into account, we replace the constant $(V_0/2)^2$ in front of the square bias by another arbitrary constant $B_0 > 0$. This gives

$$\mathrm{MSE}[\ln(\hat{L}(\boldsymbol{\theta}))/m] \approx V_0 m^{-1} n^{-\nu} + B_0 n^{-2\nu}. \tag{22}$$

For MC, we know that $\nu = 1$. For RQMC, we want to fit this model in the range of values of n of practical interest, say from 2^8 to 2^{14}, instead of in the limit when $n \to \infty$ (where the parameters might differ). We do this by applying linear regression to the logarithm of the observations for $n \geq 2^8 = 256$. We discard the smaller values of n because the exponent ν sometimes changes in that range and keeping these values would distort the results.

For any fixed n, we define the *MSE reduction factor* of an RQMC estimator with n points, with respect to an MC estimator based on n independent simulation runs, as the MSE of the MC estimator divided by that of the RQMC estimator. We estimate this factor using the fitted versions of (22). When n is large, it increases approximately as $\mathcal{O}(n^{-1}/n^{-\nu}) = \mathcal{O}(n^{\nu-1})$. But for small or moderate n, it is not always increasing in n, because of the effect of the bias in (22).

4.2. Examples with synthetic data

Our first set of experiments are with an artificial data set generated from a known model, very similar to Sivakumar et al. (2005). We consider both cross-sectional data ($T_q = 1$) and panel data (with $T_q = 3$ and 10), and $s = 5$, 10, and 15. For each s and T_q, we have $|\mathcal{A}(q)| = 4$ for every q, for $m = 500$ individuals. The coordinates $x_{q,j,\ell}$ of the attribute vectors are independent $N(1, 1)$ random variables (normal with mean 1 and variance 1) for alternatives $j = 1, 2$ and $N(0.5, 1)$ for alternatives $j = 3, 4$. The s coordinates of $\boldsymbol{\beta}_q = (\beta_{q,1}, \ldots, \beta_{q,s})$

are also independent $N(1,1)$ random variables. Then we repeated the experiments for $\boldsymbol{\beta}_q$ multinormal with the same $N(1,1)$ marginals, but with correlations of 0.3 across its s components. That is, the covariance matrix has 1's on the diagonal and 0.3 everywhere else. We refer to these two distributions as the *independent* and *correlated* cases. To generate realizations of $\boldsymbol{\beta}_q$, we generate the standard normal variates by inversion. In the correlated case we decompose the covariance matrix via PCA, as explained in the appendix and in L'Ecuyer (2009) (this gives slightly smaller RQMC variances than the more familiar Cholesky decomposition). After generating an artificial data set from the model in a first stage, in the second stage we estimate the log-likelihood function for this data set, at the value $\boldsymbol{\theta}^0$ of $\boldsymbol{\theta}$ used to generate the data, by simulation. To see what happens when $\boldsymbol{\theta}$ is farther away from $\boldsymbol{\theta}^0$, for the independent case with $s = 5$ and 10, we also estimate the log-likelihood function at 5 values of $\boldsymbol{\theta}$ generated randomly on the surface of the ball $\|\boldsymbol{\theta} - \boldsymbol{\theta}^0\| = \rho$, for $\rho = 0.1$ and 0.3. Here we give a representative subset and a summary of our results. Detailed results are in the online appendix.

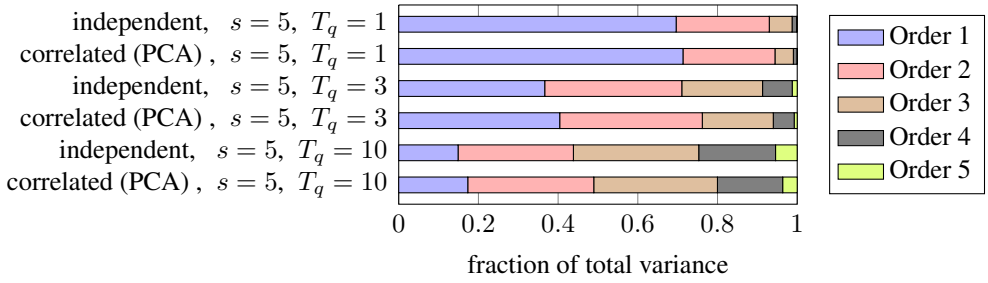

Figure 1: Fraction of average relative variance for all individuals, per projection order r, for $s = 5$, with $T_q = 1$, 3 and 10, for the independent and correlated cases.

Figure 1 shows the estimated relative variances per order r, σ_r^2, averaged over individuals, for $s = 5$. There is not much difference between the independent and correlated cases, except that the lower-order projections have a slightly larger share of the variance for the correlated case, so we expect RQMC to work slightly better in that case. Increasing the correlation would amplify this effect. The share of lower-order projections also decreases (so we expect RQMC to be less effective) when T_q increases. A closer observation (see the appendix) reveals that all projections of the same order contribute almost the same proportion of MC variance. This is explained by the homogeneity of the synthetic population. It suggests that order-dependent weights are appropriate for this situation. For $s = 5$ and $T_q = 1$, the fractions of relative variance σ_r^2/σ^2 for $r = 1, \ldots, 5$ are 0.70,

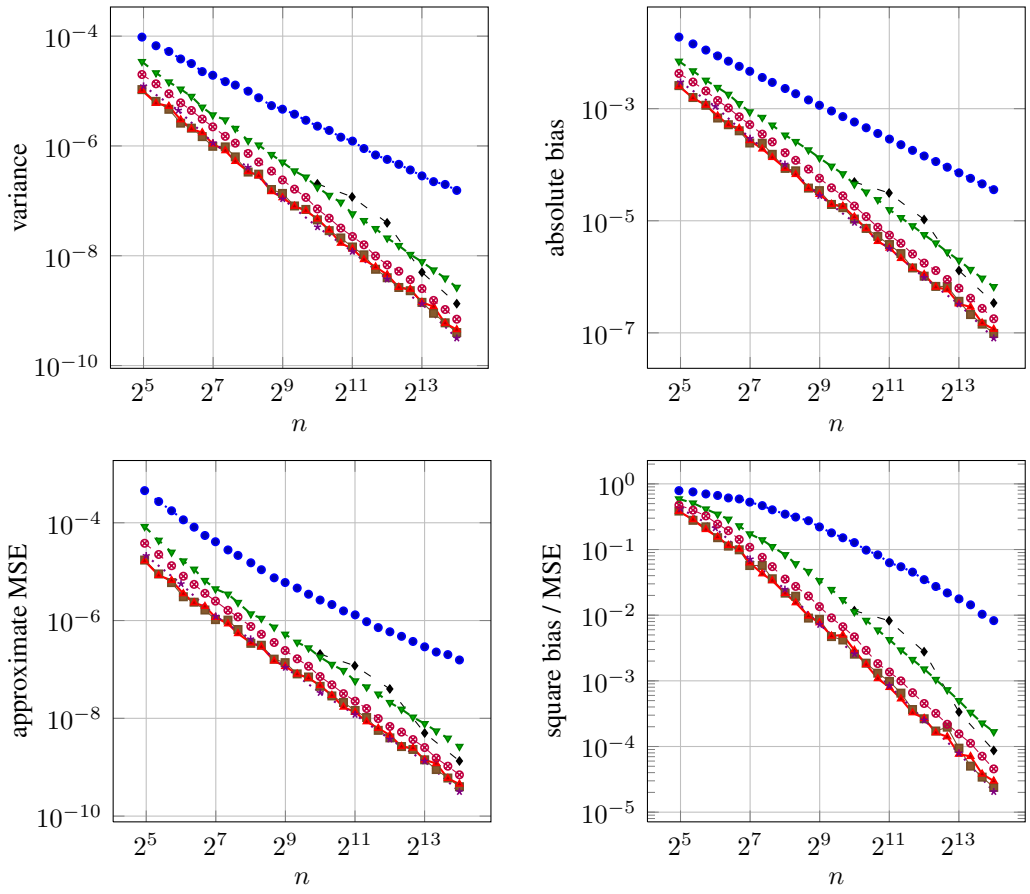

Figure 2: Estimated variance (top left), bias (top right), MSE (bottom left), and fraction of the MSE contributed by the square bias (bottom right) of the MC and RQMC estimators of the log-likelihood function for the independent case with $s = 5$ and $T_q = 1$, using MC (·●·), lattice-γ_u (—■—), lattice-0.1 (—▲—), lattice-$M32$ (-♦-), Sobol' nets (···★···) Halton-shift points (-▼-), and Halton-FL points (-⊗-). The results for lattice-order and lattice-0.5 (not shown here) are very similar to those of lattice-γ_u and lattice-0.1.

0.23, 0.057, 0.011, and 0.0014. The associated order-dependent weights (normalized so that $\gamma_2 = 1$) are $\gamma_3 = 0.11$, $\gamma_4 = 0.021$, and $\gamma_5 = 0.0057$ (recall that γ_1 is not used). If we fit the parameter γ for exponential order-dependent weights, in the independent case with $s = 5$, we obtain $\gamma = 0.18$, 0.33 and 0.57 for $T_q = 1$, 3 and 10, respectively. This provides some support for adopting the lattice-0.1 rules for $T_q = 1$ and the lattice-0.5 rules for $T_q = 10$.

The variance, bias, and MSE on the log-likelihood function, obtained with

the lattices constructed based on those weights and for other RQMC point sets, are given in Figure 2, in logarithmic scale. We find that all the lattice rules (except lattice-$M32$) as well as the randomized Sobol' and Faure nets have comparable performances. They perform a little better than Halton-FL, much better than Halton-shift, and they provide a large MSE reduction over MC. The lattice-$M32$ rules have an erratic behavior; they come close to the good ones for some values of n and they do much worse for other values. The plausible explanation is that for certain values of n, we are more lucky with the uniformity of the most important projections not considered in the $M_{32,24,16,12}$ criterion. For the other rank-1 lattices, no choice of weights seems to offer a solid advantage over the other choices and their differences in MSE reductions fluctuate somewhat randomly (but not too much) across values of n. For example, the ratio of variances for lattice-0.1 and lattice-γ_{u} is larger than 1 on average but ranges from 0.5 to 3.5 for the different values of n, and is not monotone in n. These ratios are similar for the other choices of weights (compared with lattice-γ_{u}). When we do the same comparison with a larger s, the main difference is that lattice-0.1 rules become slightly better than the lattice-0.5 and the Sobol' and Faure nets for $s = 10$ or 15; they reduce the variance by up to 30 % more. The bottom right plot in the figure confirms that the share of MSE contributed by the square bias decreases faster with RQMC than with MC when n increases. For small n (say less than about 200), the MSE reduction is mostly due to the bias reduction.

Estimates of the exponents ν for various RQMC methods and values of s and T_q are given in Table B.1 of the appendix. These exponents are smaller when s or T_q is larger. They are approximately between 1.5 and 1.7 for $s = 5$ and $T_q = 1$, between 1.4 and 1.5 for $s = 5$ and $T_q = 10$, between 1.2 and 1.3 for $s = 15$ and $T_q = 1$, and cannot be distinguished from 1 (so we do not report them) when $s \geq 10$ and $T_q = 10$. In the latter case, for $s \geq 10$ and $T_q = 10$, we could in fact observe no significant MSE reduction at all for any of the RQMC methods! That is, RQMC methods are no better than MC for this example, at least in the range of values of n that we have observed. Table 1 shows the (significant) MSE reduction factors interpolated from (22) (with estimated parameters) at $n = 10^4$. The Sobol' nets generally give the best MSE reduction factors in low dimensions and when $T_q = 1$, but the rank-1 lattices win in higher dimension and when $T_q > 1$. In general, the MSE reduction factors are much higher in situations where the low-order projections have a large share of the variance (smaller s, smaller T_q, and higher correlation), as expected. In fact, the Sobol' nets generally do a bit better in situations where the projections of order two have a larger share of the variance (smaller s and higher correlation). This is consistent with the fact

that their parameters were selected mainly based of the uniformity of their two-dimensional projections. The restriction on the number of points in Faure nets prevented us from simulating for enough values of n to fit the bias and variance as with other RQMC constructions. We do not show the results in the figures to reduce the number of superpositions; their performance is comparable to Sobol' nets for $s = 5$ and to lattice rules for $s = 10$ and 15. In additional experiments, we tried replacing the random starting point in Halton-FL with a random shift modulo 1, and this reduced their performance to a level almost comparable to the Halton-shift points. Reciprocally, the Halton points without the Faure-Lemieux scramble but with a random starting point perform almost as well as Halton-FL. This suggests that a random starting point is the key ingredient for improvement, and that it significantly improves the projections on average.

Figure 3 illustrates the fact that the MSE reduction factor does not always increase in n, in the range of values of n used in practice. It shows the fitted MSE (top) and MSE reduction factors (middle) with MC and RQMC, for the independent case with $s = 10$, for $T_q = 1$ and 3. For small n, the square bias dominates the MSE for MC, and it decreases faster than for RQMC, which may cause a decrease in the MSE reduction factor as a function of n. This behavior depends in particular on the proportion of variance in projections of high order and on the population size m.

Independent case

s	5			10		15	
T_q	1	3	10	1	3	1	3
Halton-shift	43	9.0	10	7.5	2.6	3.4	1.5
Halton-FL	150	21	9.8	10	3.5	5.5	1.8
Sobol' nets	**300**	**32**	**15**	14	3.4	5.0	1.9
lattice-0.1	230	30	11	**18**	**4.4**	**7.9**	**2.4**

Correlated (PCA) case

s	5			10		15	
T_q	1	3	10	1	3	1	3
Halton-shift	59	15	12	13	3.1	6.2	2.1
Halton-FL	200	32	13	18	4.2	9.1	2.6
Sobol' nets	**400**	**60**	**20**	24	4.9	9.9	2.8
lattice-0.1	350	47	13	**27**	**5.4**	**11**	**3.3**

Table 1: MSE reduction factors with respect to MC, approximated by (22), evaluated at $n = 10^4$, for the independent case (top table) and the correlated case (bottom table).

When we moved $\boldsymbol{\theta}$ randomly at distance ρ from $\boldsymbol{\theta}^0$, we observed variations

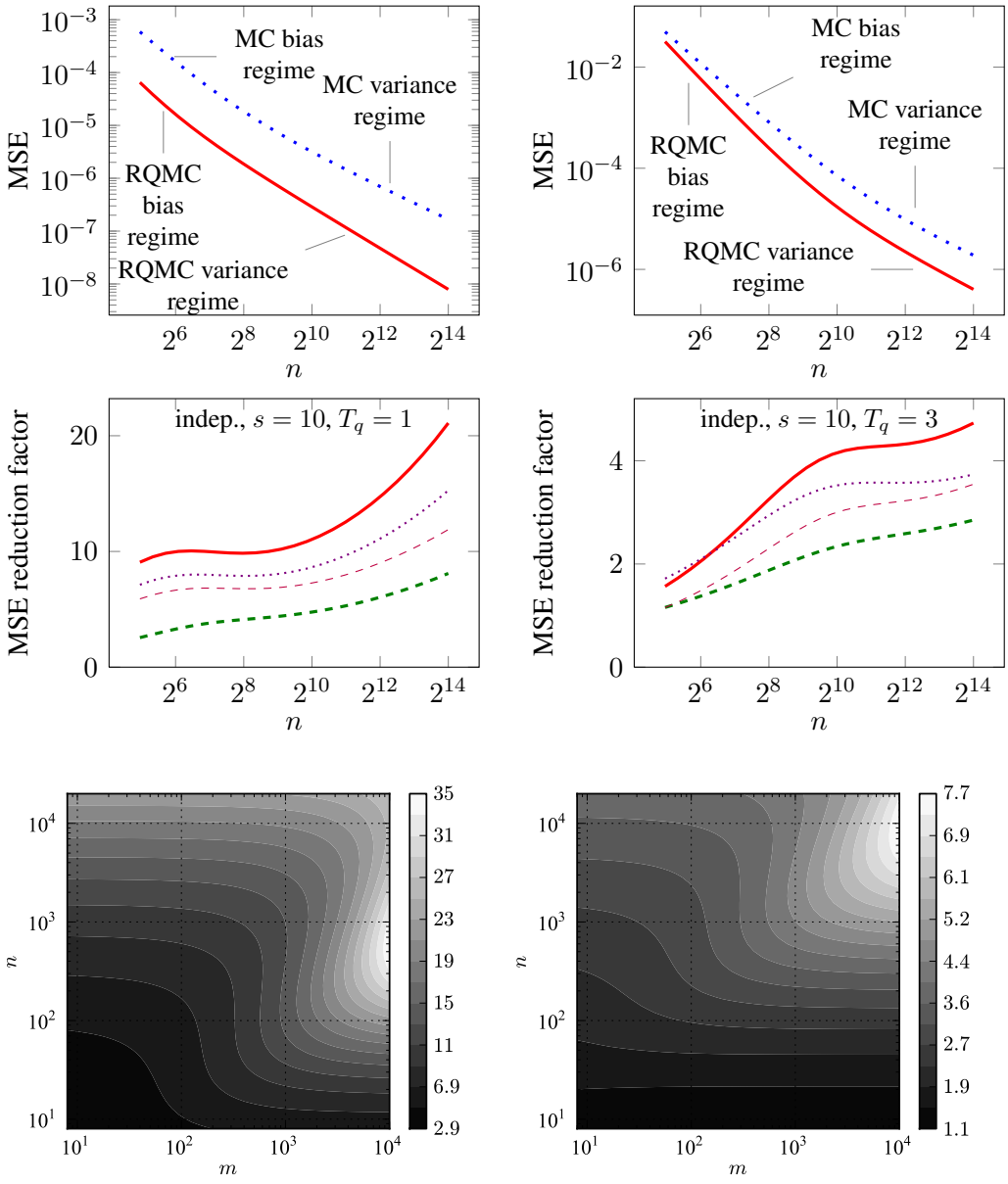

Figure 3: Top: fitted MSE reduction factor showing the bias and variance regimes, for the independent case with $s = 10$, $T_q = 1$ (left) and $T_q = 3$ (right), using MC and RQMC (lattice-0.1 rules). Middle: MSE reduction based on the fitted MSE, for the same cases as above, using lattice-0.1 (——), Sobol' nets (·····) Halton-shift points (- - -). and Halton-FL points (- - -). Bottom: fitted MSE reduction factor as a function of m and n, for the same cases as above, using lattice-0.1 rules. Notice the different scales used on the left and on the right.

(both increases and decreases) of the MSE with RQMC, almost always within 10 % of its value at $\boldsymbol{\theta} = \boldsymbol{\theta}^0$ for $\rho = 0.1$, and within 30 % for $\rho = 0.3$, regardless of s and T_q. This gives an idea of the robustness with respect to this type of variation.

The distribution of the ANOVA terms $\sigma^2_{q,\mathrm{u}}$ among the different projections varies significantly across individuals, as can be seen from Figure 4, which gives the total variance per projection order for selected individuals, for the independent case with $s = 5$ and $T_q = 1$. (More detailed illustrations are given in the appendix.) For example, the proportion of variance contributed by the projections of order 3 and higher is less than 2% for individual $q = 1$ and more than 9% for individual $q = 4$. We thus expect RQMC to be more effective for the first individual than for the fourth one. Figure 5 confirms this; it shows the variance of the log-likelihood estimator for these two individuals, with a specialized lattice for each individual (lattice-γ_{u}) and for other RQMC point sets, in logarithmic scale. The behavior for each individual is similar to that of the average given in Figure 2, although somewhat more erratic than the average. Further experiments reported in the appendix agree with these observations, but also show that using a specialized lattice for each individual (lattice-γ_{u}) is hardly better than using the same rule for all individuals, with well-chosen weights.

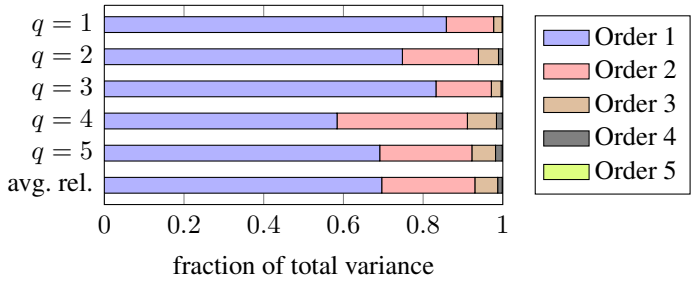

Figure 4: Fraction of relative variance per projection order, for selected individuals, and on average over individuals, for the independent case with $s = 5$ and $T_q = 1$.

Besides the MSE, another important aspect in comparing the performances of RQMC estimators is the CPU time required for their computation. These CPU times depend on the computing platform, compiler, and software implementation. With our Java implementation, on Intel® Xeon® E5462 processors clocked at 2.8 GHz, the simulation times per randomization at $n = 8191$, for $s = 5$ and $T_q = 1$, are approximately 2.9 seconds with MC and with Sobol nets, 4.3 seconds with lattice rules and Faure nets, and 5.5 and 16.6 seconds with Halton-shift and

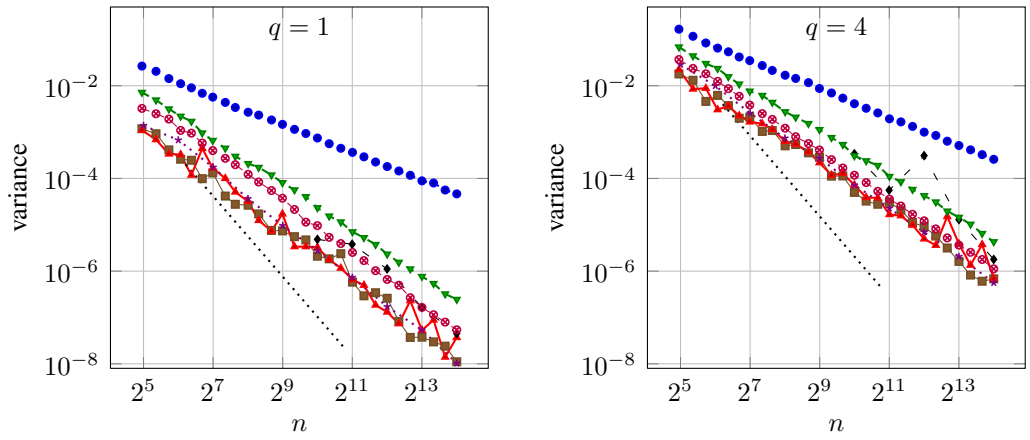

Figure 5: Estimated variance of the MC and RQMC estimators of the log-likelihood of a single individual for the independent case with $s = 5$ and $T_q = 1$, for individuals $q = 1$ (left) and 4 (right), using MC (·●·), lattice-γ_u (—■—), lattice-0.1 (—▲—), lattice-$M32$ (-♦-), Sobol' nets (···✶···), Halton-shift points (-▼-), and Halton-FL points (-⊗-). For lattice-order and lattice-0.5, the variances are very similar to those of lattice-γ_u and lattice-0.1, and we do not show them to reduce the number of curve superpositions. The dotted line indicates the n^{-2} slope, for reference.

Halton-FL points. For $s = 10$ and $T_q = 10$, these values increase to approximately 17 seconds with MC and with Sobol nets, 21 seconds with lattice rules, Faure nets, and Halton-shift points, and 37 seconds with Halton-FL. Thus, while performing better than Halton-shift points in terms of MSE, the Halton-FL points have a computational disadvantage, so the overall performance of both variants of Halton points is inferior to that of the other RQMC methods, for which the small increase in CPU time is more than compensated by the MSE reduction. Note that precomputing and storing the (unrandomized) points once for all, and retrieving them when needed, does not help in general, because the time for memory access and randomization dominates the time to compute the points.

4.3. An example with real-life data

For our second example, we consider behavioral data collected in April 2008 on airport ground access with automated vehicle technology (called cybercars) (Cirillo and Xu, 2010). The respondents were intercepted in the waiting area of the airport and the responses recorded during a face-to-face interview. The final sample contains information from 274 respondents. Both Revealed Preference (RP) data and Stated Preferences (SP) information were collected. The SP experiment includes 2 parts: a between-mode experiment (SP game 1) and a within

mode experiment (SP game 2). In this paper we only use data from SP game 2, which proposes two different cybercar services over which the respondents are called to express their preferences. Each respondent was presented with 9 scenarios (i.e. $T_q = 9$), giving a total of 2466 observations. Attribute levels are based upon the respondents real trip to the airport as reported in the RP questionnaire. The attributes and their levels are described in Table 2.

attribute	possible levels
1. waiting time	5, 10, 15, 20 (in minutes)
2. travel cost	70% of taxi, 85% of taxi, same as taxi
3. dropping area	terminal building, parking lot
4. maneuvering system	fully automated, human driver with ITS, human driver
5. track structure	guideway, grade with rubber tire

Table 2: The variables and their admissible levels, for SP game 2

A number of parametric models for the distributions were estimated and compared. The retained model assumes one constant factor associated to each possible level of waiting time, except the level 5 minutes, taken as reference. The cost factor follows a lognormal, and the remaining service-level factors are normally distributed, with moreover one factor for automated maneuvering system and one factor for human driver maneuvering system attribute (using ITS or not). These distributions of the components of $\boldsymbol{\beta}_q$ are given in Table 3, where $N(\mu, \sigma^2)$ and $\ln N(\mu, \sigma^2)$ refer the the normal and lognormal distributions, respectively, with parameters μ and σ^2. The first three components of $\boldsymbol{\beta}_q$ have constant values, so we simulate only the last five, to which we assign the indices 1 through 5. Thus, $s = 5$, and the vector $\boldsymbol{\theta}$ here has 13 dimensions.

factor	coordinate index	distribution
Waiting time 10 minutes		constant $= -0.6141158$
Waiting time 15 minutes		constant $= -1.0036583$
Waiting time 20 minutes		constant $= -1.7356732$
Cost	1	$\ln N(-1.9953313, 1.8167162)$
Passenger dropped	2	$N(1.7088261, 1.5988351)$
Automated cybercar	3	$N(-0.23137212, 1.1745619)$
Human driven cybercar	4	$N(0.13015642, 0.71851975)$
Guided way cybercar	5	$N(-0.10063324, 1.042506)$

Table 3: Distributions of the components of $\boldsymbol{\beta}_q$ in the simulation experiments with the real data.

An examination of the relative ANOVA variances averaged over individuals (see Figure C.16 of the appendix) reveals that, unlike in the example with synthetic data, the variances are not uniform across projections of the same order and do not consistently decrease with projection order. This suggests that order-dependent or geometric weights are not ideal in this case, but we find that they nevertheless perform well empirically. The fractions of relative variance σ_r^2/σ^2 per projection order r, for $r = 1, \ldots, 5$, are 0.34, 0.37, 0.22, 0.061, and 0.0052. If we insist on exponentially-decreasing weights, the best fit for γ is $\gamma = 0.24$.

The variance, bias, and MSE on the log-likelihood function are plotted in Figure 6 for selected RQMC point sets. Lattice-order and lattice-γ rules, although less suited for this example than for the previous example with synthetic data, perform comparably to lattice-γ_u rules. Figure 7 shows plots of the fitted MSE reduction factors with respect to MC, as a function of n, for several RQMC point sets. These results point to the robustness of lattice rules constructed with criterion (15) with weights of a simple form, such as lattice-0.1 or lattice-0.5 rules.

Like for the example of Subsection 4.2, the distribution of the ANOVA variances among the different projections varies across individuals. For individual $q = 116$, less than 10 % of the total variance goes in projections of order $r \geq 3$, whereas this percentage is more than 45% for individual $q = 79$. Detailed results on the ANOVA variances can be found in the appendix. Figure 8 shows the variance of the log-likelihood estimator for the constructed lattices and other RQMC point sets for these two individuals. As expected, RQMC is more effective for individual $q = 116$ than for $q = 79$. The respective performances of the different point sets follow the same pattern as in the example with synthetic data, and here too, no choice of weights for lattices clearly stands out for all values of n. For example, the ratio of variances for lattice-0.1 to those for lattice-γ_u is larger than 1 on average but ranges from 0.5 to 2.3 for the different values of n, and is not monotone in n.

4.4. Optimization with RQMC for the real-life data

While our main target in this paper was to develop a better understanding of RQMC methods for the evaluation of choice probabilities, the ultimate goal is to improve parameters estimation. As an empirical test of the improvement provided by our randomized lattice rules for this estimation, we generated 50 independent realizations of the log-likelihood function estimator (6), using $n = 1021$ independent draws per individuals (standard MC) for each realization. Then we maximized each of those functions with respect to $\boldsymbol{\theta}$, using a modified version of AMLET (Bastin et al., 2006), and we computed the sample mean and variance

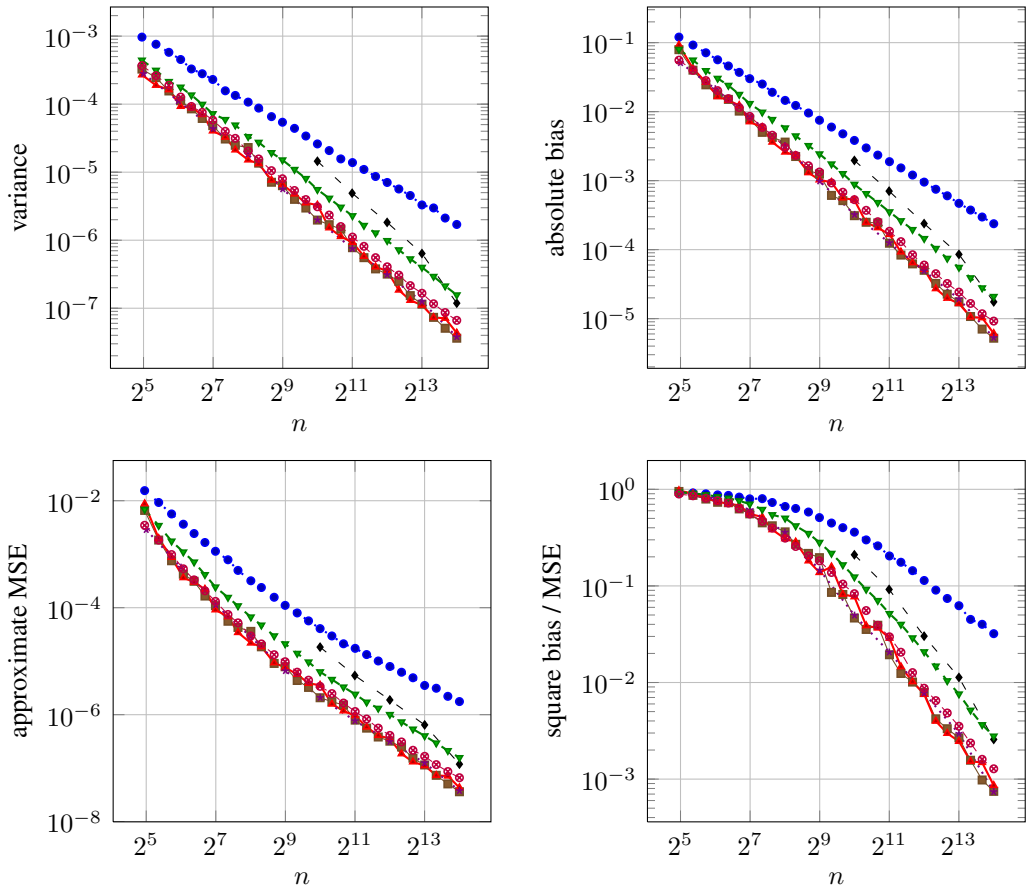

Figure 6: Estimated variance (top left), bias (top right), MSE (bottom left), and fraction of the MSE contributed by the square bias (bottom right) of the MC and RQMC estimators of the log-likelihood function for the example with real-life data, using MC (· ● ·), lattice-γ_u (—■—), lattice-0.1 (—▲—), lattice-$M32$ (- ♦ -), Sobol' nets (···✶···) Halton-shift points (- ▼ -), and Halton-FL points (- ⊗ -). The results for the lattice-order and lattice-0.5 rules are very similar to those of lattice-γ_u and lattice-0.1 rules and Sobol' nets.

of these 50 optimizers. We repeated the same experiment using Halton sequence and the lattice-0.5 rule with $n = 1021$ points for each individual, with independent random shifts across individuals and across the 50 runs in both cases. Table 4 reports the empirical means of the 13 parameter estimates for MC, Halton points, and lattice-0.5 (they agree quite well), and the variance reduction factor (VRF), defined as the MC variance divided by the RQMC variance. It also reports the means and the VRF for the 50 estimates of the log-likelihood maximum value

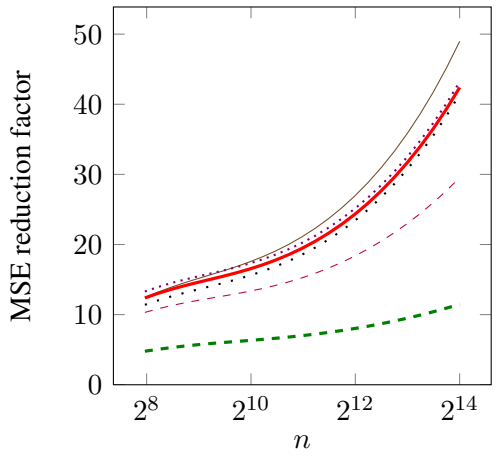

Figure 7: MSE reduction factors based on the fitted MSE, for the example with real-life data, using lattice-γ_u (——), lattice-0.1 (——), lattice-0.5 (· · · ·), Sobol' nets (·····), Halton-shift points (- - -), and Halton-FL points (- - -). The curve for lattice-γ, not shown here, almost coincides with that for lattice-0.1.

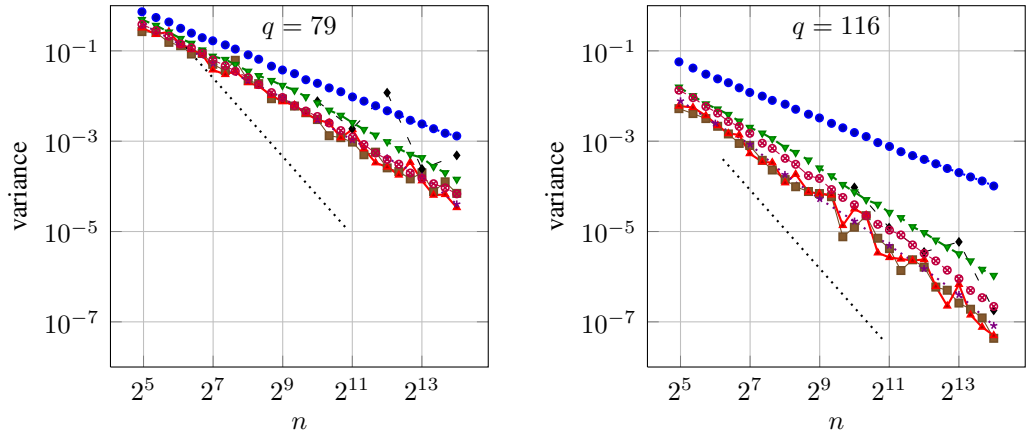

Figure 8: Estimated variance of the MC and RQMC estimators of the log-likelihood of a single individual for the example with real-life data, for individuals $q = 79$ (left) and 116 (right), using MC (· ·●·), lattice-γ_u (—■—), lattice-0.1 (—▲—), lattice-$M32$ (- ♦ -), Sobol' nets (···*···), and Halton and Halton-FL points (- ▼ - and - ⊗ -). The dotted line indicates the n^{-2} slope for reference.

returned by the algorithm, for comparison. This VRF is not the same as in the results of Figure 6, where $\boldsymbol{\theta}$ is fixed instead of being optimized. We see that the VRF for the parameter estimates is more modest than for the log-likelihood estimate, but it is still significant, with a slight advantage for the lattice rule. We

obtained similar results with other lattice rules, so we do not report them. Here we have no good way of estimating the overall bias on the parameter estimates, because various sources of bias (from the simulation and from the optimization) can interfere, as observed by Bastin and Cirillo (2010).

Parameter	MC mean	Halton-shift mean	Lattice-0.5 mean
Waiting time 10 minutes	-0.614	-0.615 (2.8)	-0.618 (3.7)
Waiting time 15 minutes	-1.003	-1.006 (2.3)	-1.010 (3.6)
Waiting time 20 minutes	-1.736	-1.739 (2.6)	-1.744 (3.3)
Cost (μ)	-2.013	-2.012 (2.4)	-2.008 (3.1)
Cost (σ)	1.815	1.815 (1.6)	1.811 (2.1)
Passenger dropped (μ)	1.704	1.704 (2.7)	1.714 (4.0)
Passenger dropped (σ)	1.594	1.567 (3.1)	1.604 (2.6)
Automated cybercar (μ)	-0.230	-0.230 (4.2)	-0.231 (4.1)
Automated cybercar (σ)	1.168	1.174 (3.9)	1.179 (3.0)
Human driven cybercar (μ)	0.134	0.133 (1.4)	0.134 (1.7)
Human driven cybercar (σ)	0.715	0.721 (1.7)	0.731 (3.2)
Guided way cybercar (μ)	-0.098	-0.100 (1.8)	-0.099 (2.1)
Guided way cybercar (σ)	1.042	1.044 (2.6)	1.048 (3.2)
Loglikelihood	-4.413	-4.411 (4.9)	-4.410 (18.9)

Table 4: Parameter estimates with MC, Halton points, and lattice-0.5. for the example with real data; variance reduction factors (VRF) for Halton and lattice-0.5, compared with MC, are given in brackets next to the estimates.

5. Conclusion

We studied the application of RQMC methods to reduce both the bias and the variance when estimating the mixed-logit log-likelihood function. We showed that RQMC improves the convergence of the bias in exactly the same way as for the variance, compared with MC, and the reduction of the squared bias can dominate the MSE reduction when n is small. Our main emphasis was on randomly-shifted lattice rules, for which we studied how the parameters should be selected based on measures of discrepancy that take into account the class of integrands considered and give weights to the projections. Previous parameter selections for these rules (based on the $M_{32,24,16,12}$ criterion) performed very poorly for our examples, which means that the choice of parameters is very important.

We also found that our lattice rules constructed with CBC with the weighted $\mathcal{P}_{2\alpha}$ criterion are quite robust to the choice of weights among those we proposed,

which is very encouraging, because there is then no need to spend a huge effort to estimate the appropriate weights, and that there is no need to select different parameters for the different individuals.

The randomized Sobol' and Faure nets provided improvements comparable to the lattice rules, although for the Faure nets there are strong limitations on the number of points that can be selected. The randomly-shifted Halton points were not competitive, although using a random starting point in the Halton sequence improved their performance significantly. But they still remained slightly inferior to the other point sets and are also more expensive computationally.

As expected, the observed efficiency improvement of RQMC compared to MC decreased when we increased the dimension s of the integrals. It also decreased rapidly when we increased T_q, the number of selections per individual, even though this does not change the dimension s of the integral. Presumably, increasing T_q increases the variability of the function and/or pushes a larger proportion of the variance into higher-order projections.

In one example with real data, we found that RQMC also reduces the variance of the parameter estimates obtained by maximizing the sample log-likelihood. This improvement is less spectacular than for estimating the log-likelihood at a single point, but is still significant.

Acknowledgements

This work was supported by NSERC-Canada Discovery Grants to P. L'Ecuyer and F. Bastin, a Canada Research Chair to P. L'Ecuyer, a GERAD scholarship to D. Munger, the EuroNF Network of Excellence to B. Tuffin, and INRIA's associated team MOCQUASIN to all but the fourth author. This paper is an outgrow from a communication entitled "Estimation strategies for mixed logit models" (Paper 555), presented at the *12th Conference of the The International Association for Travel Behaviour Research*, in Jaipur, India, 2009.

References

Bastin, F., Cirillo, C., 2010. Reducing simulation bias in mixed logit model estimation. Journal of Choice Modelling 3 (2), 71–88.

Bastin, F., Cirillo, C., Toint, Ph. L., 2006. An adaptive Monte Carlo algorithm for computing mixed logit estimators. Computational Management Science 3 (1), 55–79.

Bastin, F., Cirillo, C., Toint, Ph. L., 2010. Estimating non-parametric random utility models, with an application to the value of time in heterogeneous populations. Transportation Science 44 (4), 537–549.

Bhat, C. R., 2001. Quasi-random maximum simulated likelihood estimation of the mixed multinomial logit model. Transportation Research 35B (7), 677–693.

Bhat, C. R., 2003. Simulation estimation of mixed discrete choice models using randomized and scrambled Halton sequences. Transportation Research B 37 (3), 837–855.

Bhat, C. R., Sener, I. N., 2009. A copula-based closed-form binary logit choice model for accommodating spatial correlation across observational units. Journal of Geographical Systems 11 (3), 243–272.

Brownstone, D., Bunch, D. S., Train, K., 2000. Joint mixed logit models of stated and revealed references for alternative-fuel vehicles. Transportation Research Part B: Methodological 34 (5), 315–338.

Cirillo, C., Axhausen, K. W., 2006. Evidence on the distribution of values of travel time savings from a six-week diary. Transportation Research A 40, 444–457.

Cirillo, C., Xu, R., 2010. Forecasting cybercar use for airport ground access: A case study at BWI (Baltimore Washington International Airport). Journal of Urban Planning and Development 136 (3), 186–194.

Cranley, R., Patterson, T. N. L., 1976. Randomization of number theoretic methods for multiple integration. SIAM Journal on Numerical Analysis 13 (6), 904–914.

Dick, J., Sloan, I. H., Wang, X., Wozniakowski, H., 2004. Liberating the weights. Journal of Complexity 20 (5), 593–623.

Faure, H., Lemieux, C., 2009. Generalized Halton sequences in 2008: A comparative study. ACM Transactions on Modeling and Computer Simulation 19 (4), Article 15.

Hess, S., Bierlaire, M., Polak, J. W., 2005. Estimation of value of travel-time savings using mixed logit models. Tranportation Research Part A 39 (3), 221–236.

Hess, S., Rose, J., Bain, S., 2009. Random scale heterogeneity in discrete choice models. In: Proceedings of the European Transport Conference. Leeuwenhorst, The Netherlands.

Hickernell, F. J., 1998. A generalized discrepancy and quadrature error bound. Mathematics of Computation 67, 299–322.

Hickernell, F. J., 2000. What affects the accuracy of quasi-Monte Carlo quadrature? In: Niederreiter, H., Spanier, J. (Eds.), Monte Carlo and Quasi-Monte Carlo Methods 1998. Springer-Verlag, Berlin, pp. 16–55.

Hickernell, F. J., 2002. Obtaining $O(N^{-2+\epsilon})$ convergence for lattice quadrature rules. In: Fang, K.-T., Hickernell, F. J., Niederreiter, H. (Eds.), Monte Carlo and Quasi-Monte Carlo Methods 2000. Springer-Verlag, Berlin, pp. 274–289.

L'Ecuyer, P., 2008. SSJ: A Java Library for Stochastic Simulation. Software user's guide, available at http://www.iro.umontreal.ca/~lecuyer.

L'Ecuyer, P., 2009. Quasi-Monte Carlo methods with applications in finance. Finance and Stochastics 13 (3), 307–349.

L'Ecuyer, P., Lemieux, C., 2000. Variance reduction via lattice rules. Management Science 46 (9), 1214–1235.

L'Ecuyer, P., Munger, D., 2011. On choices of discrepancy for randomly-shifted lattice rules. In: Monte Carlo and Quasi-Monte Carlo Methods 2010. Invited paper.

Lemieux, C., 2009. Monte Carlo and Quasi-Monte Carlo Sampling. Springer-Verlag, New York, NY.

Lemieux, C., Cieslak, M., Luttmer, K., 2004. RandQMC User's Guide: A Package for Randomized Quasi-Monte Carlo Methods in C. Software user's guide, available at http://www.math.uwaterloo.ca/~clemieux/randqmc.html.

McFadden, D. L., Train, K., 2000. Mixed MNL models for discrete response. Journal of Applied Econometrics 15 (5), 447–270.

Niederreiter, H., 1992. Random Number Generation and Quasi-Monte Carlo Methods. Vol. 63 of SIAM CBMS-NSF Regional Conference Series in Applied Mathematics. SIAM, Philadelphia, PA.

Ökten, G., 2009. Generalized von Neumann-Kakutani transformation and random-start scrambled Halton sequences. Journal of Complexity 25, 318–331.

Owen, A. B., 1998. Latin supercube sampling for very high-dimensional simulations. ACM Transactions on Modeling and Computer Simulation 8 (1), 71–102.

Owen, A. B., 2003. Variance with alternative scramblings of digital nets. ACM Transactions on Modeling and Computer Simulation 13 (4), 363–378.

Sándor, Z., Train, K., 2004. Quasi-random simulation of discrete choice models. Transportation Research Part B 38 (4), 313–327.

Sivakumar, A., Bhat, C. R., Ökten, G., 2005. Simulation estimation of mixed discrete choice models with the use of randomized quasi-Monte Carlo sequences: A comparative study. Transportation Research Record 1921, 112–122.

Sloan, I. H., Joe, S., 1994. Lattice Methods for Multiple Integration. Clarendon Press, Oxford.

Sobol', I. M., 1967. The distribution of points in a cube and the approximate evaluation of integrals. U.S.S.R. Comput. Math. and Math. Phys. 7, 86–112.

Sobol', I. M., Myshetskaya, E. E., 2007. Monte Carlo estimators for small sensitivity indices. Monte Carlo Methods and Applications 13 (5–6), 455–465.

Struckmeier, J., 1995. Fast generation of low-discrepancy sequences. Journal of Computational and Applied Mathematics 61, 29–41.

Train, K., 2000. Halton sequences for mixed logit. Working paper No. E00-278, Department of Economics, University of California, Berkeley, USA.

Train, K., 2003. Discrete Choice Methods with Simulation. Cambridge University Press, New York, USA.