Monte Carlo and Quasi-Monte Carlo for Statistics

Art B. Owen

Abstract This article reports on the contents of a tutorial session at MCQMC 2008. The tutorial explored various places in statistics where Monte Carlo methods can be used. There was a special emphasis on areas where Quasi-Monte Carlo ideas have been or could be applied, as well as areas that look like they need more research.

1 Introduction

This survey is aimed at exposing good problems in statistics to researchers in Quasi-Monte Carlo. It has a mix of well known and not so well known topics, which both have their place in a research context. The selection of topics is tilted in the direction of problems that I have looked at. That enables me to use real examples, and examples are crucial to understanding statistics.

Monte Carlo methods are ubiquitous in statistics. Section 2 presents the bootstrap. It is a method of resampling the observed data to judge the uncertainty in a quantity. The bootstrap makes minimal assumptions about how the data were obtained. Some efforts at bringing balance to the resampling process have brought improvements, but they are not large enough to have made much impact on how the bootstrap is used. Permutation tests, considered in Section 3 have a similar flavor to the bootstrap, but there, efforts to impose balance can distort the results.

Markov chain Monte Carlo (Section 4) is used when we cannot directly sample the quantity of interest, but are at least able to find a Markov chain from whose stationary distribution the desired quantity can be sampled. Space limitations make it impossible to cover all of the topics from a three hour tutorial in depth. The work

Department of Statistics
Stanford University
Stanford CA, 94305
http://stat.stanford.edu/~owen

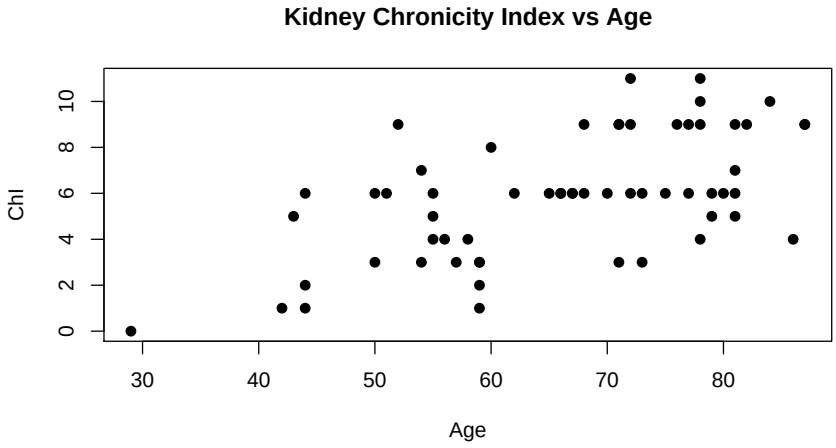

Fig. 1 A measure of kidney damage is plotted versus age for 60 subjects in [34].

on QMC for MCMC has appeared in [31], [43] and in Tribble's dissertation [42], and so it is just sketched here.

Monte Carlo methods are used for search as well as for integration. Section 5 presents the method of least trimmed squares (LTS). This is the most effective method known for highly robust regression model fitting. It uses an ad hoc Monte Carlo search strategy. Quasi-Monte Carlo methods have been used in search problems [26, Chapter 6], usually to search over the unit cube [27]. The space to search over in LTS is of a combinatorial nature. Finally, Section 6 points to two more important problems from statistics where QMC may be useful.

2 The bootstrap

Figure 1 plots a measure Y_i of kidney damage against age X_i, for 60 subjects studied in Rodwell et al. [34]. There is a clear tendency for older subjects to have greater kidney damage. This may be quantified through the correlation coefficient, which on this data takes the value 0.59. Since only 60 subjects were measured, we very much doubt that the true correlation taken over all people is exactly 0.59. Using ρ to denote a hypothetical true correlation and $\hat{\rho}$ to denote the measured one, we may just want to know the variance $V(\hat{\rho})$ of our estimate.

The variance of the sample correlation is not hard to find in closed form, so long as the data are a sample from the bivariate normal distribution. But we have no reason to expect that assumption is good enough to use. The bootstrap, introduced by Efron [5] provides a way out of that assumption. Operationally one does the following:

1. For $b = 1, \dots, B$
2. Draw (X_i^{*b}, Y_i^{*b}), $1 \le i \le 60$ with replacement from the original data.
3. Compute $\hat\rho^{*b} = \hat\rho\big((X_1^{*b}, Y_1^{*b}), \dots, (X_{60}^{*b}, Y_{60}^{*})\big)$.
4. Return the variance of $\hat\rho^{*1}, \dots \hat\rho^{*B}$.

Using $B = 9999$ the result came out to be 0.0081, so that the standard deviation of the $\hat\rho^*$ values is 0.090. What we actually got was a Monte Carlo estimate of the variance of the bootstrapped correlations $\hat\rho^*$ when resampling from the data. Even if we let $B \to \infty$ this would not be the same as the variance we want, which is that of $\hat\rho$ when sampling from the unknown true distribution of (X, Y) pairs. But bootstrap theory shows that the two variances become close quickly as the number n of sample values increases [7].

First impressions of the bootstrap are either that it is obviously ok, or that it is somehow too good to be true, like pulling oneself up by the bootstraps. The formal justification of the bootstrap begins with a statistical functional T defined on distributions F. In this case $T(F)$ gives the variance of the correlation measured on 60 pairs of data drawn from the distribution F on $\mathbb{R}^2$. Let F_0 be the unknown true distribution and $\widehat{F}_n$ be the distribution that puts equal probability $1/n$ on all n data points. As n increases $\widehat{F}_n$ approaches F_0. Then a continuity argument gives $T(\widehat{F}_n)$ approaching $T(F_0)$. The continuity argument holds in great generality but there are exceptions, as well as remedies in some of those cases [32].

The bootstrap can also be used to estimate and correct for biases in statistics. Let $\mathbb{E}(\hat\rho \mid F)$ denote the expected value of $\hat\rho$ when sampling n data pairs from F. Typically $\mathbb{E}(\hat\rho) \ne \rho$, so that the sample correlation is biased. The bootstrap estimate of the bias $B(F) \equiv \mathbb{E}(\hat\rho \mid F) - \rho(F)$ is $B(\widehat{F}_n) \equiv \mathbb{E}(\hat\rho^* \mid \widehat{F}_n) - \rho(\hat{F}_n)$. We can estimate this bias by resampling. In the present example we find that the average value of $\hat\rho^* - \hat\rho$ in resampling is -0.0047. If we are worried that $\hat\rho$ is too small by 0.0047 we can add 0.0047 (i.e. subtract the estimated bias) and get 0.594 instead of 0.589. Here the bias adjustment is very small. That is typical unless the method used has a large number of parameters relative to the sample size.

Figure 2 shows a histogram of the 9,999 bootstrap samples used in this analysis. The histogram is skewed, centered near the original correlation, and is quite wide. The resampled correlations cut the real line into 10,000 intervals. A bootstrap 95% confidence interval is formed by taking the central 9500 of those intervals. If the values are sorted $\hat\rho^{*(1)} \le \hat\rho^{*(2)} \le \cdots \le \hat\rho^{*(9999)}$, then the 95% confidence interval goes from $\hat\rho^{*(250)}$ to $\hat\rho^{*(9750)}$. In this example we have 95% confidence that $0.391 \le \rho \le 0.755$. Similarly there is 99% confidence that $\hat\rho^{*(100)} \le \rho \le \hat\rho^{*(9900)}$, or $0.346 \le \rho \le 0.765$. Bootstrap confidence levels are not exact. They are approximate confidence intervals. Typically they have coverage probability equal to their nominal level plus $O(n^{-1})$. See [13]. The intervals presented here are known as percentile intervals. They are the simplest but not the only bootstrap confidence interval. See [7] for other choices.

The balanced bootstrap [4] is an attempt to improve on bootstrap re-sampling. Instead of sampling n observations with replacement B times, it forms a large pool of nB observations, with B copies of each original data point. Then it randomly par-

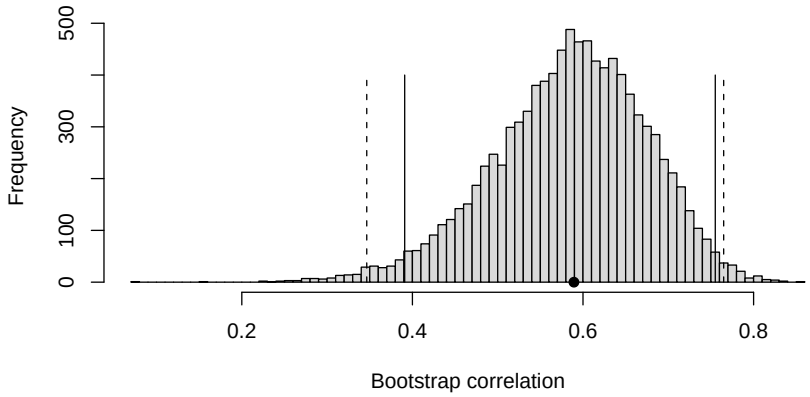

Fig. 2 This figure shows 9999 bootstrap resampled correlations for the kidney data. There is a reference point at the sample correlation. Solid vertical lines enclose the central 95% of the histogram. Dashed lines enclose the central 99%.

titions them into B subsets of equal size n. Those groups are treated as the bootstrap samples. Now each original observation appears the same number B of times among the sampled data. The balanced bootstrap is similar to QMC. Higher order balancing, based on orthogonal arrays, was proposed by [12], but that proposal requires the number n of observations to be a prime power.

To apply QMC, it helps to frame the bootstrap problem as an integral over $[0,1]^n$, as discussed in [28] and thoroughly investigated by Liu [21]. We may write X_i^* as $X_{\lceil nU_i \rceil}$ where $U_i \sim \mathbb{U}(0,1)$ are independent. Then $X^* = (X_1^*, \ldots, X_n^*)$ is a function of $U \sim \mathbb{U}(0,1)^n$. The bootstrap estimate of bias is a Monte Carlo estimate on B samples of $\int_{[0,1]^n} Q(U)\, dU$ for

$$Q(U) = T(X^*(U)) - T(X), \tag{1}$$

where $T(X)$ is a shorthand for $T(n^{-1}\sum_{i=1}^n \delta_{X_i})$ with δ_x the point mass distribution at x. The bootstrap estimate of variance has integrand

$$Q(U) = \left(T(X^*(U)) - \int_{[0,1]^n} T(X^*(U))\, dU \right)^2. \tag{2}$$

The upper end of a bootstrap 95% confidence interval is the solution $T^{0.975}$ to $\int_{[0,1]^n} Q(U)\, dU = 0.975$ where

$$Q(U) = 1_{T(X^*(U)) \le T^{0.975}}, \tag{3}$$

while the lower end uses 0.025 instead of 0.975.

To implement the ordinary bootstrap we take points $U_b = (U_{b1}, \ldots, U_{bn}) \sim \mathbb{U}(0,1)^n$ for $b = 1, \ldots, B$ and use ordinary Monte Carlo estimates of the integrals in (1), (2) and (3). To get a QMC version, we replace these points by a B point QMC rule in $[0,1]^n$.

The integrands Q are generally not smooth, because $X_{\lceil nU_i \rceil}$ is discontinuous in U_i, apart from trivial settings. Smoothness is important for QMC to be effective. Fortunately, there is a version of the bootstrap that yields smooth integrands, at least for estimating bias and variance. The weighted likelihood bootstrap (WLB), proposed by Newton and Raftery [24] uses continuous reweighting of the data points instead of discrete resampling. It is a special case of the Bayesian bootstrap of Rubin [38]. Where the ordinary bootstrap uses $T(n^{-1}\sum_{i=1}^n \delta_{X_i^*})$, the WLB uses $T(\sum_{i=1}^n w_i \delta_{X_i})$ for certain random weights w_i. To generate these weights put $v_i = -\log(U_i)$ and $w_i = v_i / \sum_{j=1}^n v_j$. Then for uniform U_i we find that v_i has the standard exponential distribution, while $w = (w_1, \ldots, w_n) = w(U)$ has a Dirichlet distribution. We substitute smooth reweighting for resampling, by using $T(\sum_{i=1}^n w_i(U)\delta_{X_i})$ in place of $T(X^*(U))$ in equations (1), (2), and (3). In a QMC version of the WLB, we take B points with low discrepancy in $[0,1]^n$ and use them as the uniform numbers that drive the reweighting.

The WLB does not give the same answers as the original bootstrap. But the original bootstrap is not exact, only asymptotically correct as $n \to \infty$. The same is true of the WLB.

In her dissertation, Ruixue Liu [21] compared various QMC methods for bootstrap problems. The underlying problem was to measure the correlation over law schools, of the average LSAT score and grade point average of newly admitted students.

In addition to the methods described above, she considered several QMC and QMC-like constructions, as follows. Latin hypercube sampling (LHS)[22] provides a sample in which each of the sample coordinates $U_{1i}, \ldots, U_{Bi}$ for $i = 1, \ldots, n$ is simultaneously stratified into intervals of width $1/B$. Randomized orthogonal array sampling [29] stratifies the bivariate or trivariate margins of the distribution of U_b. When the array has strength t, then all $\binom{n}{t}$ of the t–dimensional margins are stratified, typically into cubical regions of width $B^{-1/t}$. Orthogonal array based LHS [41] has both the LHS and the orthogonal array stratifications. Scrambled nets [30] are a randomization of a QMC method (digital nets).

Some numerical results from Liu [21] are shown in Tables 1 and 2. Bootstrap estimates of the bias, variance and 95'th percentile were repeatedly computed and their variance was found. Those variances are based on 2000 replications of each method, except for scrambled nets for which only 100 replications were used. The variances are presented as variance reduction factors relative to the variance of the plain resampling bootstrap. For example, we see that the LHS version of resampling is about 10 times as efficient as the ordinary bootstrap on the bias estimation problems.

Several trends are apparent in these results. Better variance reductions are obtained via reweighting than resampling, as one would expect because the former

	Resampling			Reweighting		
Method	Bias	Var	Perc	Bias	Var	Perc
Plain bootstrap	1.0	1.0	1.0	1.4	1.4	1.2
Balanced bootstrap	10.8	1.2	1.2			
Latin hypercube sampling	10.7	1.2	1.3	16.4	2.3	1.6
Randomized orthogonal array	15.7	3.1	1.7	105.2	8.3	2.9
OA-based LHS	36.0	3.2	1.6	126.1	9.0	2.7
Scrambled $(0,61^2,15)$–net in base 61	30.0	3.1	1.8	116.9	8.8	3.3

Table 1 This table shows variance reduction factors attained from applying QMC methods to the three bootstrap problems described in the text. The values are normalized so that the ordinary bootstrap gets a value of 1.0 in each problem. Each bootstrap method used $B = 61^2 = 3721$ resamples. The quantity being bootstrapped was a sample correlation. Both reweighting and resampling bootstraps were used to estimate bias, variance and 95'th percentile. The orthogonal arrays had strength 2. The balanced bootstrap is only defined for the resampling approach.

has smoother integrands. It is easiest to improve on bootstrap bias estimates, harder for variance estimates, and hardest for confidence intervals. Again we would expect this. In the von Mises expansion (e.g. [8]) the statistic T is approximated by a sum of functions of one observation at a time. When that approximation is accurate, then the bias integrands are more nearly additive in the n inputs while the variance integrands are nearly the square of an additive approximation and we might expect the resulting statistical function to be of low effective dimension in the superposition sense [2]. Unfortunately, the interest in usual applied settings is in the reverse order, percentiles, then variance then bias. Finally orthogonal array methods with high strength and few levels do poorly

3 Permutation tests

Newborn babies have a walking reflex, in which their feet start a walking motion when placed in contact with a surface. Zelazo et al. [44] conducted an experiment to test whether regular daily encouragement of this reflex would result in babies

	Resampling			Reweighting		
Method	Bias	Var	Perc	Bias	Var	Perc
Plain bootstrap	1.0	1.0	1.0	1.4	1.4	1.1
Balanced bootstrap	9.9	1.2	1.1			
Latin hypercube sampling	10.2	1.2	1.3	15.7	2.5	1.5
Randomized orthogonal array	3.3	0.6	0.4	8.5	0.7	0.3
OA-based LHS	7.5	0.6	0.4	8.8	0.7	0.3
Scrambled $(0,17^3,15)$–net in base 17	60.7	6.5	2.2	756.0	33.6	2.9

Table 2 This table shows the same quantities as Table 1, except that now $B = 17^3 = 4913$ and the orthogonal arrays had strength 3.

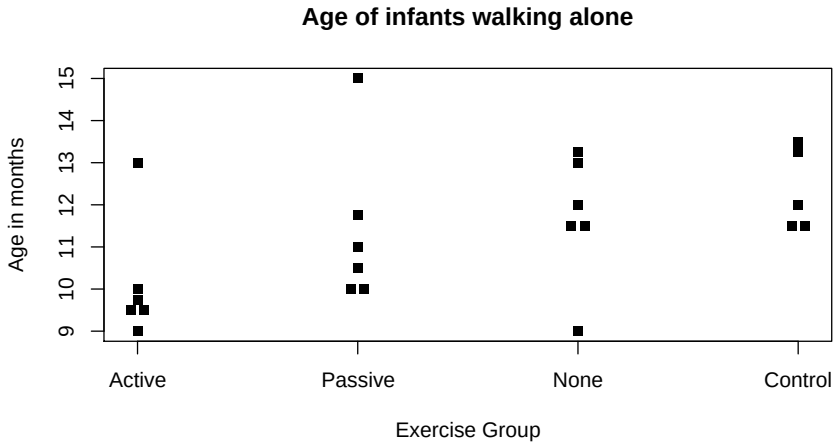

Fig. 3 Shown are ages at which babies learned to walk in each of 4 subject groups.

learning to walk at an earlier age, than without the encouragement. Figure 3 shows the results of this experiment.

There were twenty-four babies in the experiment. The age at which they could walk alone was reported for twenty-three of them. One group (active) had their reflex stimulated daily for eight weeks. Another (passive) had gross-motor stimulation of a different kind. A third group (none) got neither stimulation but was regularly tested for walking ability. The fourth group (control) was not even tested until a greater age.

For illustration, we will focus on the simple problem of testing whether the active group learned to walk sooner than the 'none' group. For differing statistical analyses of all four groups see [1, Chapter 10] and [44].

The active group learned to walk at an average age of $\bar{X} = 10.125$ months compared to $\bar{Y} = 11.7$ months for the none group. They were faster on average by 1.58 months, but perhaps this difference can be attributed to chance.

A permutation test can be used to judge the difference. The two groups combined have 12 babies. We could select 6 of them to be the active group in any of $\binom{12}{6} = 924$ ways. If there really is no difference in the distribution of walking age between the groups, then we could permute the active versus none labels without changing the distribution of the outcome. After such a permutation of the labels, we get new averages $\bar{X}^*$ and $\bar{Y}^*$. The original permutation has a 1 in 924 chance of giving the largest mean difference of them all, if the two groups are identical.

Figure 4 shows a histogram of all 924 permuted group mean differences $\bar{X}^* - \bar{Y}^*$. There is a vertical reference line at the observed value of $\bar{X} - \bar{Y} = -1.58$. There were only 49 values less than or equal to -1.58. By symmetry there were 49 values at least $+1.58$ and so there were 98 values as or more extreme than that observed.

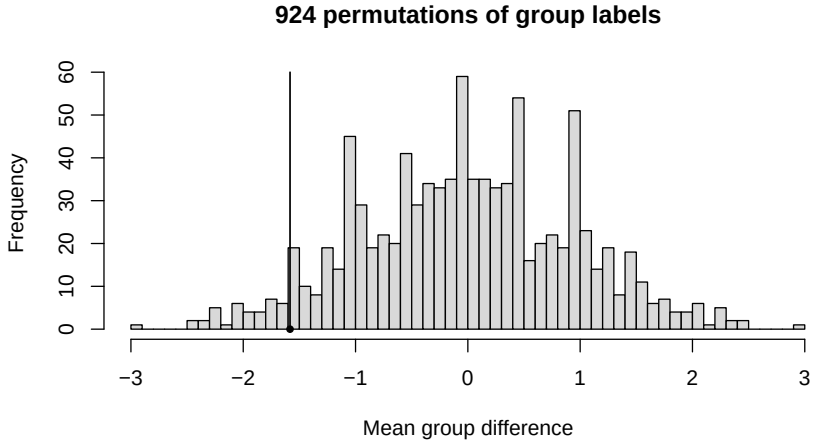

Fig. 4 The histogram shows all 924 values of $\bar{X}^* - \bar{Y}^*$ obtainable by permuting the labels (control versus none) for 12 of the babies learning to walk. The vertical reference line is at the observed value $\bar{X} - \bar{Y}$.

This permutation test allows one to claim a two-tailed p-value of $98/924 \doteq 0.106$ for the difference. A difference this large could arise by chance with probability about 10.6%. The observed difference is not at all unusual. Some statisticians (not including the author) would claim a p-value of $49/924$ here, which is closer to 5%, the conventional line at which results begin to be taken seriously.

In balanced permutations [6] we compare the two groups in a more carefully controlled way. Each time the labels are reassigned, we ensure that three members of the new treatment group come from the old treatment group and that three come from the old control group. The other six babies are similarly balanced, and they become the relabeled control group. There are $\binom{6}{3}^2 = 400$ balanced permutations for this data. It is customary to include the original sample, from an identity permutation, in an MC based permutation analysis. This avoids the possibility of $p = 0$. The identity permutation is not a balanced permutation, and so adding it in to the reference set here gives a histogram of 401 values.

The intuitive reason for balancing the permutations is as follows. If there is no difference between the groups, then the balanced permutations still give rise to the same distribution of the treatment difference as the real sample has. But when there is a difference, for example the treatment group learn to walk two months earlier on average, then things change. In relabeling we get $\bar{X}^*$ for treatment and $\bar{Y}^*$ for control. In some unbalanced permutations all or most of $\bar{X}^*$ came from the original control group while in others few or none came from there. In balanced permutations exactly half of the values contributing to $\bar{X}^*$ are at the high level and half at the low level. A mean difference between the two original groups will mostly cancel for the relabeled groups. As a result, the histogram of $\bar{X}^* - \bar{Y}^*$ for balanced permutations

can be expected to be narrower than the one for full permutations, when there is a treatment difference. Narrower histograms lead to smaller reported p-values. In the baby walking data, we find that $|\bar{X}^* - \bar{Y}^*| \geq |\bar{X} - \bar{Y}|$ holds for only 27 points in the reference set, yielding a p-value of $27/401 = 0.067$, which is smaller than for the full permutation set.

While balanced permutations have the potential to sharpen inferences, they have been applied without theoretical support. In simulations they have been found to give p–values that are too permissive. The problem with balanced permutations is that they do not form a group under composition. The group property is a key ingredient in the permutation argument [17]. Some results in [40] show that the chance of the original permutation beating all $\binom{n}{2}^2$ balanced permutations is much larger than $(1+\binom{n}{2}^2)^{-1}$ even when two groups of size n have identical distributions. The p values are too small by a factor that grows quickly with n and is already over 100 for $n = 10$.

It may be possible to repair balanced permutations, although this looks difficult at present. One approach is to try to compensate by adjusting the reported p values. Another is to search for a suitable subgroup of permutations to use.

4 MCMC

Usually in QMC problems, we write the desired answer as an integral $\mu = \int_{[0,1]^d} f(x)\,dx$. The function f takes independent uniformly distributed quantities, transforms them into the desired ones, such as dependent non-uniform values, and then computes whatever it is we want to average as a function of those values. As is well known [26], QMC methods achieve better rates of convergence than MC on such problems, making only modest smoothness assumptions on f.

In some applications however, we seek a value $\mu = \int_{\mathbb{X}} f(x)\pi(x)\,dx$ where there is no practical way to turn uniform random variables into the desired ones from π on the state space $\mathbb{X}$. We might be able to get $x \sim \pi$ by rejection sampling but with such an unfavorable acceptance rate that the method would be useless. This issue arises commonly in Bayesian computations, for statistics [10] and machine learning, as well as in the physical sciences [23].

In MCMC we generate $x_i = \phi(x_{i-1}, v_i)$ where $v_i \sim \mathbb{U}(0,1)^d$. The distribution of x_i depends on $x_{i-1}, x_{i-2}, \ldots$ only through the immediate predecessor x_{i-1}, and so it has the Markov property. With some skill and care, one can often choose ϕ so that the Markov chain has π as its stationary distribution. Sometimes it is also important to choose a good starting point x_0. Under reasonable conditions there is a law of large numbers for MCMC, so that

$$\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^{n} f(x_i) \to \mu$$

and there are also central limit theorems for MCMC. The theory and practice of MCMC in statistics is presented in several books, including Liu [20] and Robert and Casella [33].

The quantity $\hat{\mu}_n$ depends on n values $v_i \sim \mathbb{U}(0,1)^d$. Its expectation is thus an nd dimensional integral which approximates, but does not equal, the desired one. Unlike crude Monte Carlo, there is a bias in MCMC. Under reasonable conditions it decays exponentially with n and so is often very small. Some other times the exponential decay is still too slow for practically useful problems, and so the constant in the exponent matters. For some studies of the bias, see Rosenthal [35].

There have been some efforts to replace n vectors v_i by quasi-Monte Carlo points. The key idea is to open up the vectors v_i into one long sequence $u_1, u_2, u_3, \ldots, u_{nd}$ where $v_i = (u_{d(i-1)+1}, u_{d(i-1)+2}, \ldots, u_{di})$. Then one replaces the independent and identically distributed (IID) points u_i for $i = 1, \ldots, nd$ by some alternative points with good equidistribution properties. Naive substitution of QMC points v_i into MCMC can fail very badly.

An early effort by Liao [19] has proven effective. Liao's approach is to take a QMC point set $a_1, \ldots, a_n \in [0,1]^d$ and randomly reorder it getting $v_i = a_{\tau(i)}$ where τ is a random permutation of $\{1, \ldots, n\}$. Then the reordered points $v_1, v_2, \ldots, v_n$ are concatenated into a single vector of nd values in $[0,1]$ with which to drive the MCMC.

The thesis of Tribble [42] has an up to date account, extending the work published in [31] and [43], and giving methods that do even better than Liao's, in numerical examples.

What is known so far is that a completely uniformly distributed (CUD) sequence $u_1, u_2, \ldots$ can be used in place of IID points. In that case the QMC answer converges to the correct result, at least for Markov chains with finite state spaces. The main theoretical technique is a coupling argument first made by Chentsov [3] for sampling Markov chains by inversion but then extended by Owen and Tribble [31] to handle Metropolis-Hastings sampling. As of 2008 the continuous state space case had not been handled. (It is now covered in a technical report by Chen, Dick and Owen.)

A CUD sequence is one that can be grouped into overlapping d-tuples $z_i = (u_i, \ldots, u_{i+d-1})$ such that the d dimensional star–discrepancy of $z_1, \ldots, z_{n-d+1}$ tends to 0. This must hold for all d. Extensions for random sequences and for limits of finite length sequences are given in [43]. If points u_i are independent $\mathbb{U}(0,1)$ then z_i have discrepancy that converges to 0 with probability one. But specially constructed sequences can have smaller discrepancies and hence may be more accurate.

Using a CUD sequence can be likened to using the entire period of a pseudo-random number generator. This is an old suggestion of Niederreiter [25]. Quite a different approach is to drive multiple copies of Markov chains by QMC, with reordering between steps. See for example L'Ecuyer, Lécot and Tuffin [16] and earlier work by Lécot [15].

The best numerical results for MCQMC so far used some small linear feedback shift registers, in Tribble [42]. He gets variance reduction factors well over 1,000 for the posterior means of parameters using the Gibbs sampler on the well known pump failure data set. For a higher dimensional vasorestriction data set of [9]. he

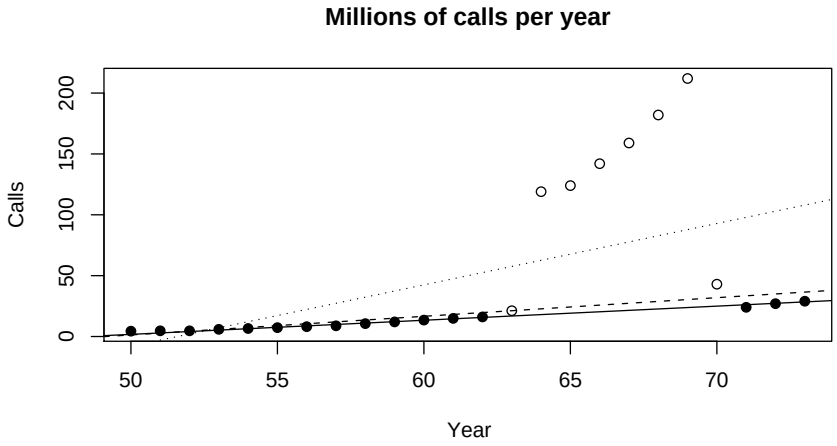

Fig. 5 This plot shows phone calls versus year. Some data values (plotted as open circles) were corrupted by counts of minutes instead of calls. From top to bottom at the right, the three fitted lines are least squares regression (dots), L_1 regression (dashes), and least trimmed squares (solid).

obtains variance reduction factors up to 100. In both cases the variance reduction factors increase with sample size. Switching from IID to CUD to randomized CUD points brings the best improvements when the function $\phi(x,v)$ is smooth. This occurs for the Gibbs sampler, which randomly updates components of x one at a time given the others. More general Metropolis-Hastings sampling methods typically have acceptance-rejection steps which make for discontinuous integrands and lessened improvements. Still, the Gibbs sampler is important enough that improvements of it are worth pursuing.

The theoretical results so far may be likened to the strong law of large numbers. They indicate that for large enough n, the answer converges. What is missing is an analogue of the central limit theorem, or of the Koksma-Hlawka inequality, to say how fast the convergence takes place. Furthermore, not enough is known about the speed with which discrepancies (for varying d) of CUD sequences can vanish. A survey of CUD constructions is given by [18].

5 Least trimmed squares

Figure 5 shows the Belgian telephone data of [37]. The data were supposed to portray the number of calls per year (in millions) as a function of the year (minus 1900). As it turned out minutes, and not calls, were counted for a period starting in late 1963 and ending in early 1970. The errors in the data make a big difference to the regression line, fit by least squares.

Errors or other data contamination of this nature are not as rare as we would like. In the present setting we can clearly see that something is amiss, but in problems where dozens or even thousands of explanatory variables are used to make predictions, gross errors in the data might not be easy to see.

A robust line would be better suited to this problem. Robust methods are those that are less affected by bad data. An old, but still very good reference on robust statistical methods is the book by Huber [14].

The least squares regression line shown Figure 5 was found by minimizing $\sum_{i=1}^{n}(Y_i-\beta_0-\beta_1X_i)^2$ over β_0 and β_1, where Y_i is the i'th phone measure and X_i is the i'th year. The largest errors dominate the sum of squares, and so least squares is far from robust. A natural alternative is to minimize the L_1 error $\sum_{i=1}^{n}|Y_i-\beta_0-\beta_1X_i|$ instead. It can be fit by quantile regression, where it generalizes the sample median to the regression context. As such, this choice is less affected by outliers. It brings a big improvement for this data, but it is still not robust, and gets fooled badly on other data.

The current state of the art in robust fitting is to sum most of the squared errors, but not the large ones. This is the Least Trimmed Squares (LTS) method of [37]. For any $\beta=(\beta_0,\beta_1)$ let $e_i(\beta)=|Y_i-\beta_0-\beta_1X_i|$, and then sort these absolute errors: $e_{(1)}(\beta)\le e_{(2)}(\beta)\le\cdots\le e_{(n)}(\beta)$. We choose β to minimize

$$f(\beta)=\sum_{i=1}^{\lfloor\alpha n\rfloor}e_{(i)}(\beta)^2.$$

If we were confident that fewer than 20% of the data were bad, we could take $\alpha=0.8$. The smallest workable value for α is $(n+p+1)/(2n)$, which allows for just under half the data to be bad.

Figure 5 shows the least trimmed squares fit. It goes through the good points and is oblivious to the bad ones.

Figure 6 shows residuals $Y_i-\hat\beta_0-\hat\beta_1X_i$ for estimates $\hat\beta$ fit by least squares and by least trimmed squares. In least squares, the good observations have residuals of about the same size as the bad ones. For least trimmed squares, the residuals from bad points are much farther from zero than those from good ones. A data analyst could then easily decide that those points need further investigation.

When there is only one explanatory variable, then there are fast algorithms to find the least trimmed squares estimates. But when there are many such variables then the best known fitting strategy is a Monte Carlo search of Rousseeuw and van Driessen [36]. Their search is guided by some theory.

Consider a general linear model, which predicts Y_i by $\sum_{j=1}^{p}\beta_jX_{ij}=\beta'X_i$ for $p\ge2$ and $n\ge p$. It is known that the LTS solution $\hat\beta$ solves $Y_i=\hat\beta'X_i$ for p of the points $i=1,\ldots,n$. Thus the solution can be found by checking $\binom{n}{p}$ interpolating models. The cost of checking all those models though is often too high. They use instead a Monte Carlo strategy.

One of their Monte Carlo search methods is presented in Figure 7. It is the one recommended when $n\le600$.

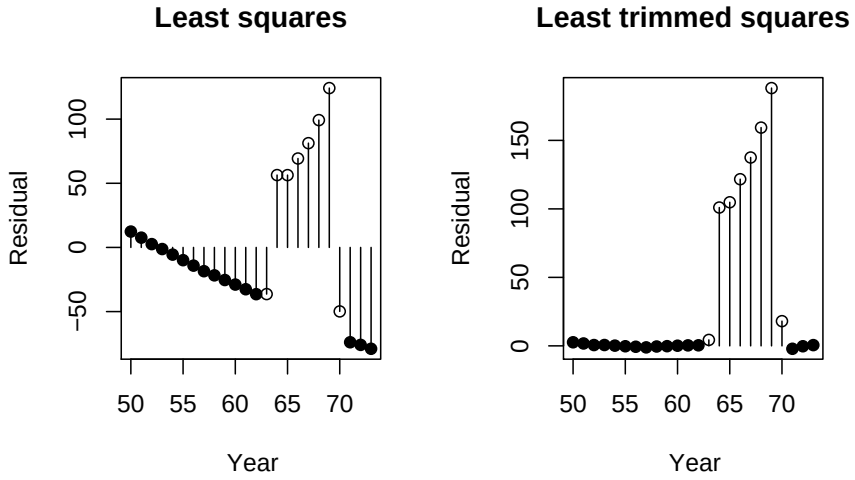

Fig. 6 The left panel shows the residuals from least squares. The good points get errors of about the same size as the bad ones. The right panel shows residuals from least trimmed squares. The bad points get very large residuals, making them numerically conspicuous.

The search in Figure 7 has clearly been tuned empirically, for speed and effectiveness. For example the *C*–step brings an improvement, but they found diminishing returns to fully iterated *C*-steps. It is better to generate many candidates and then follow up only the best ones.

The algorithm makes numerous choices that seem arbitrary. There is clearly room for a better understanding of how to search for an optimum.

1. Sample p points: $(Y_i, X_{i1}, \ldots, X_{ip}) \quad i \in \mathbb{I} \subset \{1, \ldots, n\} \quad |\mathbb{I}| = p$
2. While linear interpolation of Y_i to X_i is not unique, add one more sample point
3. Find $\hat{\beta}$ to interpolate $Y_i = X_i' \hat{\beta}$ on sampled points.
4. Find points with smallest $h = \lfloor \alpha n \rfloor$ absolute residuals (among all n points)
5. C-step: Fit LS to the h points and find newest h points with smallest absolute residuals
6. Do 2 more C-steps
7. Repeat steps 1 through 6, 500 times, keeping 10 best results
8. Run C-steps to convergence for these 10
9. Select best of those 10 end points

Fig. 7 This is an outline of the Monte Carlo search algorithm of [36] for solving the least trimmed squares regression problem, when $n \leq 600$.

6 Other methods

Some other uses of MC and QMC in statistics are very important but were not described here. Notable among the gaps is the problem of fitting generalized linear mixed models. Some efforts at this problem via QMC are reported in [39]. This problem is very important in statistical applications. It features integrands which can become very spiky in even moderately high dimensions and practical problems can involve quite high dimension. Another classical quadrature problem arising in statistics is that of integrating a probability density function of dependent random variables, (e.g. Gaussian or multivariate t) over a rectangular region. For recent work in this area see [11].

Acknowledgments

I thank Pierre L'Ecuyer and the other organizers of MCQMC 2008 for inviting this tutorial, and for organizing such a productive meeting. Thanks also to two anonymous reviewers and Pierre for helpful comments. This research was supported by grant DMS-0604939 of the U.S. National Science Foundation.

References

1. Brown, B.W., Hollander, M.: Statistics, A Biomedical Introduction. John Wiley & Sons, New York (1977)
2. Caflisch, R.E., Morokoff, W., Owen, A.B.: Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension. Journal of Computational Finance **1**, 27–46 (1997)
3. Chentsov, N.: Pseudorandom numbers for modelling Markov chains. Computational Mathematics and Mathematical Physics **7**, 218–2332 (1967)
4. Davison, A.C., Hinkley, D.V., Schechtman, E.: Efficient bootstrap simulation. Biometrika **73**(3), 555–566 (1986)
5. Efron, B.: Bootstrap methods: Another look at the jackknife. The Annals of Statistics **7**, 1–26 (1979)
6. Efron, B., Tibshirani, R., Storey, J.D., Tusher, V.: Empirical Bayes analysis of a microarray experiment. Journal of the American Statistical Association **96**, 1151–1160 (2001)
7. Efron, B.M., Tibshirani, R.J.: An Introduction to the Bootstrap. Chapman and Hall (1993)
8. Fernholz, L.T.: von Mises calculus for statistical functionals. Springer-Verlag, New York (1983)
9. Finney, D.J.: The estimation from individual records of the relationship between dose and quantal response. Biometrika **34**, 320–334 (1947)
10. Gelman, A., Carlin, J.B., Stern, H.S., Rubin, D.B.: Bayesian Data Analysis. Chapman & Hall, Boca Raton, FL (2003)
11. Genz, A., Bretz, F., Hochberg, Y.: Approximations to multivariate t integrals with application to multiple comparison procedures. In: Recent Developments in Multiple Comparison Procedures, vol. 47, pp. 24–32. Institute of Mathematical Statistics (2004)

12. Graham, R.L., Hinkley, D.V., John, P.W.M., Shi, S.: Balanced design of bootstrap simulations. Journal of the Royal Statistical Society, Series B **52**, 185–202 (1990)
13. Hall, P.G.: The Bootstrap and Edgeworth Expansion. Springer, New York (1992)
14. Huber, P.J.: Robust Statistics. Wiley, New York (1981)
15. Lécot, C.: Low discrepancy sequences for solving the Boltzmann equation. Journal of Computational and Applied Mathematics **25**, 237–249 (1989)
16. L'Ecuyer, P., Lécot, C., Tuffin, B.: A randomized Quasi-Monte Carlo simulation method for Markov chains. Operations Research **56**(4), 958–975 (2008)
17. Lehmann, E.L., Romano, J.P.: Testing Statistical Hypotheses, third edn. Springer, New York (2005)
18. Levin, M.B.: Discrepancy estimates of completely uniformly distributed and pseudo-random number sequences. International Mathematics Research Notices pp. 1231–1251 (1999)
19. Liao, L.G.: Variance reduction in Gibbs sampler using quasi random numbers. Journal of Computational and Graphical Statistics **7**, 253–266 (1998)
20. Liu, J.S.: Monte Carlo strategies in scientific computing. Springer, New York (2001)
21. Liu, R.: New findings of functional ANOVA with applications to computational finance and statistics. Ph.D. thesis, Stanford University (2005)
22. McKay, M.D., Beckman, R.J., Conover, W.J.: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics **21**(2), 239–45 (1979)
23. Newman, M.E.J., Barkema, G.T.: Monte Carlo Methods in Statistical Physics. Oxford University Press, New York (1999)
24. Newton, M.A., Raftery, A.E.: Approximate Bayesian inference with the weighted likelihood bootstrap (disc: P26-48). Journal of the Royal Statistical Society, Series B, Methodological **56**, 3–26 (1994)
25. Niederreiter, H.: Multidimensional integration using pseudo-random numbers. Mathematical Programming Study **27**, 17–38 (1986)
26. Niederreiter, H.: Random Number Generation and Quasi-Monte Carlo Methods. S.I.A.M., Philadelphia, PA (1992)
27. Niederreiter, H., Peart, P.: Quasi-Monte Carlo optimization in general domains. Caribbean Journal of Mathematics **4**(2), 67–85 (1985)
28. Owen, A.B.: Discussion of the paper by Newton and Raftery. Journal of the Royal Statistical Society, Series B **56**(1), 42–43 (1994)
29. Owen, A.B.: Lattice sampling revisited: Monte Carlo variance of means over randomized orthogonal arrays. The Annals of Statistics **22**, 930–945 (1994)
30. Owen, A.B.: Randomly permuted (t,m,s)-nets and (t,s)-sequences. In: H. Niederreiter, P.J.S. Shiue (eds.) Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing, pp. 299–317. Springer-Verlag, New York (1995)
31. Owen, A.B., Tribble, S.D.: A quasi-Monte Carlo Metropolis algorithm. Proceedings of the National Academy of Sciences **102**(25), 8844–8849 (2005)
32. Politis, D.N., Romano, J.P., Wolf, M.: Subsampling. Springer, New York (1999)
33. Robert, C., Casella, G.: Monte Carlo Statistical Methods, 2nd edn. Springer, New York (2004)
34. Rodwell, G., Sonu, R., Zahn, J.M., Lund, J., Wilhelmy, J., Wang, L., Xiao, W., Mindrinos, M., Crane, E., Segal, E., Myers, B., Davis, R., Higgins, J., Owen, A.B., Kim, S.K.: A transcriptional profile of aging in the human kidney. PLOS Biology **2**(12), 2191–2201 (2004)
35. Rosenthal, J.S.: Minorization conditions and convergence rates for Markov chain Monte Carlo. Journal of the American Statistical Association **90**, 558–566 (1995)
36. Rousseeuw, P.J., van Driessen, K.: Computing LTS regression for large data sets. Data Mining and Knowledge Discovery **12**, 29–45 (2006)
37. Rousseeuw, P.J., Leroy, A.M.: Robust Regression and Outlier Detection. Wiley, New York (1987)
38. Rubin, D.B.: The Bayesian bootstrap. The Annals of Statistics **9**, 130–134 (1981)
39. Sloan, I.H., Kuo, F.Y., Dunsmuir, W.T., Wand, M., Womersley, R.S.: Quasi-Monte Carlo for highly structured generalised response models. Tech. rep., University of Wollongong Faculty of Informatics (2007)

40. Southworth, L.K., Kim, S.K., Owen, A.B.: Properties of balanced permutations. Journal of Computational Biology **16** (2009). In press.
41. Tang, B.: Orthogonal array-based Latin hypercubes. Journal of the American Statistical Association **88**, 1392–1397 (1993)
42. Tribble, S.D.: Markov chain Monte Carlo algorithms using completely uniformly distributed driving sequences. Ph.D. thesis, Stanford University (2007)
43. Tribble, S.D., Owen, A.B.: Construction of weakly CUD sequences for MCMC sampling. Electronic Journal of Statistics **2**, 634–660 (2008)
44. Zelazo, P.R., Zelazo, N.A., Kolb, S.: Walking in the newborn. Science **176**, 314–315 (1972)