SEQUENTIAL MONTE CARLO METHODS FOR STATISTICAL ANALYSIS OF TABLES

by

Arnab Chakraborty
Persi Diaconis
Susan P. Holmes
Department of Statistics
Stanford University

Yuguo Chen
Institute of Statistics and Decision Sciences
Duke University

Jun S. Liu
Department of Statistics
Harvard University

Technical Report No. 2003-24
August 2003

This research was supported in part by
National Science Foundation grant DMS-0072360

Department of Statistics
STANFORD UNIVERSITY
Stanford, California 94305-4065

http://www-stat.stanford.edu

Sequential Monte Carlo Methods for Statistical Analysis of Tables

Arnab Chakraborty[†], Yuguo Chen[*], Persi Diaconis[†], Susan P. Holmes[†], and Jun S. Liu[‡]

[†] Department of Statistics, Stanford University, Stanford, CA 94305.

[*] Institute of Statistics and Decision Sciences, Duke University, Durham, NC 27708.

[‡] Department of Statistics, Harvard University, Cambridge, MA 02138.

Abstract

This paper describes a sequential importance sampling (SIS) procedure for simulating two-way zero-one or contingency tables with fixed marginal sums. An essential feature of the new method is that it samples the columns of the table progressively according to certain special distributions. For the zero-one tables, the new method produces Monte Carlo samples that are remarkably close to the uniform distribution, enabling one to obtain an accurate estimate of the total number of zero-one tables with fixed margins and to approximate closely the null distributions of a number of test statistics involved in testing hypotheses about such tables. Our method compares favorably with other existing Monte Carlo-based algorithms, sometimes our approach can be a few orders of magnitude more efficient than others.

1 Introduction

1.1 Darwin's finch data

In ecology, researchers are often interested in testing theories about evolution and the competition among species. The zero-one table shown in Table 1 is called an *occurrence matrix* in ecological studies. The rows of the matrix correspond to species, the columns to geological locations. A "1" or "0" in cell (i, j) represents the presence or absence, respectively, of species i at location j. The occurrence matrix in Table 1 represents 13 species of finches inhabiting 17 islands of the Galápagos Islands (an archipelago in the East Pacific). The data is known as "Darwin's finches" because Charles Darwin collected some of these species when he visited the Galápagos. Darwin suggests in *The Voyage of the Beagle* that his observation of the striking diversity in these species of finches started a train of thought which culminated in his theory of evolution (however, Sullaway (1982) shows that Darwin did not realize the significance of the finches until years after he visited the Galápagos). Cook and Quinn (1995) catalog many other occurrence matrices that have been collected. The ecological importance of the distribution of species over islands was described in Sanderson (2000) as follows: "Birds with differing beaks may live side by side because they can eat different things, whereas similarly endowed animals may not occupy the same territory because they compete with one another for the same kinds of food. Ecologists have long debated whether such competition between similar species controls their distribution on island groups or whether the patterns found simply reflect chance events in the distant past."

From a statistical point of view, the null hypothesis that the pattern of finches on islands is the result of chance rather than competitive pressures can be translated to the statement that the observed zero-one table is a "typical" sample drawn uniformly from the set of all tables with the observed row and column marginal sums. The number of islands each species inhabits (the row sums) and the number of species on each island (the column sums) are kept fixed under the null hypothesis to reflect the fact that some species are naturally more widespread than others and some islands are naturally more accommodating to a wide variety of species than others (Manly, 1995; Connor and Simberloff, 1979). For testing whether there is competition between species, Roberts and Stone (1990) suggested the test statistic

$$\bar{S}^2 = \frac{1}{m(m-1)} \sum_{i \neq j} s_{ij}^2, \tag{1}$$

where m is the number of species, $S = (s_{ij}) = AA^T$ and $A = (a_{ij})$ is the occurrence matrix. The null hypothesis is rejected if $\bar{S}^2$ is too large. Sanderson (2000) used the number of instances of two

specific species living on the same island as the test statistic, which corresponds to focusing on two rows and counting the number of columns in which both of these rows contain a one. More test statistics are discussed in Connor and Simberloff (1979), Wilson (1987), Manly (1995), Sanderson, Moulton and Selfridge (1998), Sanderson (2000), and Cobb (2003). Our methods apply to all of these.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
large ground finch	0	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
medium ground finch	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	0	0
small ground finch	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	0	0
sharp-beaked ground finch	0	0	1	1	1	0	0	1	0	1	0	1	1	0	1	1	1
cactus ground finch	1	1	1	0	1	1	1	1	1	1	0	1	0	1	1	0	0
large cactus ground finch	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
large tree finch	0	0	1	1	1	1	1	1	1	0	0	1	0	1	1	0	0
medium tree finch	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
small tree finch	0	0	1	1	1	1	1	1	1	1	0	1	0	0	1	0	0
vegetarian finch	0	0	1	1	1	1	1	1	1	1	0	1	0	1	1	0	0
woodpecker finch	0	0	1	1	1	0	1	1	0	1	0	0	0	0	0	0	0
mangrove finch	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
warbler finch	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Table 1: The occurrence matrix for the Darwin finch data. Island name code: A=Seymour, B=Baltra, C=Isabella, D=Fernandina, E=Santiago, F=Rábida, G=Pinzón, H=Santa Cruz, I=Santa Fe, J=San Cristóbal, K=Española, L=Floreana, M=Genovesa, N=Marchena, O=Pinta, P=Darwin, Q=Wolf.

A difficult challenge in carrying out these tests is that there are no good analytic approximations to the null distributions of the corresponding test statistics. We show below how to simulate the zero-one tables nearly uniformly and then adjust the samples using importance weights. We can thus obtain a good approximation to the null distribution of any test statistic. Although several methods for generating tables from the uniform distribution conditional on marginal sums have been proposed in the literature, most of them are inefficient and some of them are incorrect (see Section 6.2).

1.2 Problem formulation

From magic squares to Darwin's theory of evolution, problems of counting the number of and testing hypotheses about zero-one or contingency tables arise in many fields including mathematics, statistics, ecology, education and sociology. Although fixing the marginal sums makes these problems much more difficult, it is important to do so for many problems. For statistical applications in which the subjects are not obtained by a sampling scheme but are the only ones available to the researcher, conditioning on the marginal sums of the table creates a probabilistic basis for a test (see Chapter 4.7 of Lehmann (1986)). In some other applications such as those related to the Rasch (1960) model, the marginal sums are sufficient statistics under the null hypothesis. Conditioning on the marginal sums is a way to remove the effect of nuisance parameters on tests (see Chapter 4 of Lehmann (1986) and Snijders (1991)).

Because the interactions between the requirements on the row sums and the column sums are complicated, no truly satisfactory combinatorial methods or analytical approximations are available for deriving the distributions of the test statistics (Snijders, 1991). The table-counting problem is slightly more approachable. Several asymptotic methods have been developed for approximating the count of zero-one or contingency tables with fixed marginal sums (Békéssy, Békéssy, and Komlos, 1972; Gail and Mantel, 1977; Good and Crook, 1977). However, these formulas are usually not very accurate for tables of moderate size. Wang (1988) provided an exact formula for counting zero-one tables, which is further improved in Wang and Zhang (1998). However, their exact formula is very complicated and both authors of the papers (by personal communication) think that the formula would take too long to compute for Table 1, which is only of moderate size among our examples.

From a practical point of view, if we can simulate tables from the uniform distribution, or some distribution close to it, we can both estimate the total count of the tables and approximate the distribution of any test statistic that is a function of the table. Several algorithms for generating uniform zero-one tables have been proposed (Connor and Simberloff, 1979; Wilson, 1987; Besag and Clifford, 1989; Rao, Jana and Bandyopadhyay, 1996; Sanderson, Moulton and Selfridge, 1998; Sanderson, 2000). Snijders (1991) used the importance sampling idea to construct tables. Algorithms for generating contingency tables from the uniform distribution have also been proposed (Balmer, 1988; Boyett, 1979; Patefield, 1981). Diaconis and Gangolli (1995) developed a Markov chain Monte Carlo (MCMC) algorithm. Holmes and Jones (1996) used the rejection method both to sample contingency tables from the uniform distribution and to approximately count the number of such tables with fixed margins. In our experience, all of these methods encounter difficulties for large, sparse tables.

In this paper we describe a sequential importance sampling (SIS) approach for approximating statistics related to the uniform distribution on zero-one and contingency tables with fixed margins. The distinctive feature of the SIS approach is that the generation of each table proceeds *sequentially* column by column and the partial importance weight is monitored along the way. Section 2 introduces the basic SIS methodology and the rules for evaluating the accuracy and efficiency of our estimates. Section 3 describes how we apply conditional-Poisson sampling together with the SIS for generating zero-one tables. Section 4 proposes a more delicate SIS method that is guaranteed to always generate proper tables. Section 5 generalizes the SIS method from zero-one tables to contingency tables. Section 6 shows some applications and numerical examples, including statistical evaluation of Table 1 and a count of the number of tables with the same row and column sums as Table 1, and Section 7 concludes with a brief discussion on the method.

2 Sequential Importance Sampling

Given the row sums $\boldsymbol{r} = (r_1, r_2, \cdots, r_m)$ and the column sums $\boldsymbol{c} = (c_1, c_2, \cdots, c_n)$, we let $\Sigma_{\boldsymbol{rc}}$ denote the set of all $m \times n$ (zero-one or contingency) tables with row sums $\boldsymbol{r}$ and column sums $\boldsymbol{c}$ (assuming that $\Sigma_{\boldsymbol{rc}}$ is nonempty). Let $p(T) = 1/|\Sigma_{\boldsymbol{rc}}|$ be the uniform distribution over $\Sigma_{\boldsymbol{rc}}$. If we can simulate a table $T \in \Sigma_{\boldsymbol{rc}}$ from a "trial distribution" $q(\cdot)$, where $q(T) > 0$ for all $T \in \Sigma_{\boldsymbol{rc}}$, then we have

$$E_q\left[\frac{1}{q(T)}\right] = \sum_{T \in \Sigma_{\boldsymbol{rc}}} \frac{1}{q(T)} q(T) = |\Sigma_{\boldsymbol{rc}}|.$$

Hence, we can estimate $|\Sigma_{\boldsymbol{rc}}|$ by

$$\widehat{|\Sigma_{\boldsymbol{rc}}|} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{q(T_i)}$$

from N i.i.d. samples $T_1, \ldots, T_N$ drawn from $q(T)$. Furthermore, if we are interested in evaluating $\mu = E_p f(T)$, we can use the weighted average

$$\hat{\mu} = \frac{\sum_{i=1}^{N} f(T_i) \frac{p(T_i)}{q(T_i)}}{\sum_{i=1}^{N} \frac{p(T_i)}{q(T_i)}} = \frac{\sum_{i=1}^{N} f(T_i) \frac{1}{q(T_i)}}{\sum_{i=1}^{N} \frac{1}{q(T_i)}} \tag{2}$$

as an estimate of μ. For example, if we let $f(T) = 1_{\{\chi^2\text{-statistic of } T \geq s\}}$, formula (2) estimates the p-value of the observed χ^2-statistics s.

The standard error of $\hat{\mu}$ can be simply estimated by further repeated sampling. A more analytical method is to approximate the denominator of $\hat{\mu}$ by the δ-method so that

$$\text{std}(\hat{\mu}) \approx \sqrt{\frac{\text{var}_q\{f(T)\frac{p(T)}{q(T)}\} + \mu^2 \text{var}_q\{\frac{p(T)}{q(T)}\} - 2\mu \text{cov}_q\{f(T)\frac{p(T)}{q(T)}, \frac{p(T)}{q(T)}\}}{N}}. \tag{3}$$

However, since this standard deviation is dependent on the particular function of interest, it is also useful to consider a "function-free" criterion, the *effective sample size* (Kong, Liu and Wong 1994) to measure the overall efficiency of an importance sampling algorithm:

$$\mathrm{ESS} = \frac{N}{1 + cv^2},$$

where the *coefficient of variation* is defined as

$$cv^2 = \frac{\mathrm{var}_q\{p(T_i)/q(T_i)\}}{E_q^2\{p(T_i)/q(T_i)\}}.$$

The ESS is simply the L^2-distance between the two distributions p and q; the smaller it is, the closer the two distributions are. Heuristically the ESS measures how many i.i.d. samples are equivalent to the N weighted samples. Throughout the paper, we use cv^2 as a measure of efficiency for an importance sampling scheme.

A central problem in importance sampling is the construction of a good trial distribution $q(\cdot)$. Because the target space Σ_{rc} is rather complicated, it is not immediately clear what proposal distribution $q(T)$ can be employed. Note that

$$q(T = (t_1, \ldots, t_n)) = q(t_1)q(t_2|t_1)q(t_3|t_2, t_1)\cdots q(t_n|t_{n-1}, \ldots, t_1), \tag{4}$$

where $t_1, \ldots, t_n$ denote the configurations of the columns of T. This factorization suggests that it is perhaps a fruitful strategy to generate the table sequentially, column by column, and use the partially sampled table to guide the sampling of the next column. More precisely, the first column of the table is sampled conditional on its marginal sum c_1. Conditional on the realization of the first column, the row sums are updated and the second column is sampled in a similar manner. This procedure is repeated until all the columns are sampled. The recursive nature of (4) gives rise to the name *sequential importance sampling*. A general theoretical framework for SIS is given in Liu and Chen (1998).

3 Sampling Zero-One Tables: Theory and Implementation

To avoid triviality, we assume throughout the paper that none of the column or row sums is zero. For the first column, we need to choose c_1 of the m possible positions to put ones in. Suppose the c_1 rows we choose are $i_1, \cdots, i_{c_1}$. Then we only need to consider the new $m \times (n-1)$ subtable. The row sums of the new table are updated by subtracting the respective numbers in the first column from the original row sums. Then the same procedure can be applied to sample the second column.

For convenience, we let $r_j^{(l)}$, $j = 1, \ldots, m$ denote the updated row sums after the first $l-1$ columns have been sampled. For example, $r_j^{(1)} = r_j$ and, after sampling the positions $i_1, \cdots, i_{c_1}$ for the first column, we have

$$r_j^{(2)} = \begin{cases} r_j^{(1)} - 1, & \text{if } j = i_k \text{ for some } 1 \le k \le c_1, \\ r_j^{(1)}, & \text{otherwise.} \end{cases} \tag{5}$$

Let $c_j^{(l)}$, $j = 1, \ldots, n-(l-1)$, $l = 1, \ldots, n$, denote the updated column sums after we have sampled the first $l-1$ columns. That is, after sampling the first $l-1$ columns the l-th column in the original table is updated to the first "current column" so that $(c_1^{(l)}, \ldots, c_{n-(l-1)}^{(l)}) = (c_l, \ldots, c_n)$.

A naive way to sample the c_1 nonzero positions for the first column (and subsequently the other columns) is from the uniform distribution, which can be rapidly executed. However, this method turns out to be very inefficient: the cv^2 routinely exceeds 10,000, making the effective sample size very small. Although it is perhaps helpful to apply the resampling idea (Liu and Chen, 1998), a more direct way to improve efficiency is to design a better sampling distribution. Intuitively, we want to put a "one" in position i if the ith row sum is very large, and vice versa. To achieve this goal, we adopt here the "conditional-Poisson (CP)" sampling method described in Brewer and Hanif (1983) and Chen, Dempster and Liu (1994).

3.1 Conditional Poisson sampling

Suppose we want to sample c units from a population $\Omega = \{u_1, \ldots, u_m\}$ of size m without replacement ($0 \le c \le m$). Let $\mathbf{p} = (p_1, \ldots, p_m)$ be a vector of probabilities, and let

$$\mathbf{Z} = (Z_1, \ldots, Z_m) \tag{6}$$

be independent Bernoulli trials with probability of successes $p_1, \ldots, p_m$, respectively. Then,

$$S_{\mathbf{Z}} = Z_1 + \cdots + Z_m$$

has the *Poisson-Binomial* distribution. Here $Z_i = 1$ corresponds to unit u_i being selected in the sample. The conditional distribution of $\mathbf{Z}$ given $S_{\mathbf{Z}}$ is called the conditional-Poisson distribution. If we let

$$w_i = \frac{p_i}{1 - p_i},$$

it is easy to see that the probability of obtaining a sample $S = \{u_{i_1}, \ldots, u_{i_c}\}$ is

$$P(S) \propto \prod_{k=1}^{c} w_{i_k}.$$

Chen et al. (1994) and Chen and Liu (1997) provide five schemes to sample from the CP distribution, and we adopt their drafting sampling method here. Let A_k $(k = 0, \ldots, c)$ denote the set of selected units after k draws. Thus, $A_0 = \emptyset$ and A_c is the final sample we obtain. At the kth step of the drafting sampling $(k = 1, \ldots, c)$, a unit $u_j \in A_{k-1}^c$ is selected into the sample with probability

$$P(j, A_{k-1}^c) = \frac{w_j R(c-k, A_{k-1}^c \backslash j)}{(c-k+1)R(c-k+1, A_{k-1}^c)},$$

where

$$R(s, A) = \sum_{B \subset A, |B|=s} \Big(\prod_{u_i \in B} w_i \Big).$$

Most of the computing time required by this sampling procedure is spent on the calculation of $R(s, A)$ through the recursive formula $R(s, A) = R(s, A \setminus \{u_s\}) + w_s R(s-1, A \setminus \{u_s\})$, and the whole process is of order $O(s|A|)$. See Chen et al. (1994) and Chen and Liu (1997) for more details regarding the CP sampling and its applications.

3.2 Justification of the CP sampling

The following lemma (personal communication with Charles Stein) sheds some insight on why the CP distribution is desirable in the sequential sampling of zero-one tables.

THEOREM 1 *For an $m \times n$ zero-one table, the distribution of the first column conditional on its sum c_1 and the row sums $r_1, \ldots, r_m$ is the same as the conditional distribution of $\mathbf{Z}$ (defined by (6)) given $S_{\mathbf{Z}} = c_1$ with $p_i = r_i/n$.*

Thus, it is natural to define the sampling distribution $q(t_1)$ for the first column to be the CP distribution with $p_i = r_i/n$, i.e., with weights $w_i = r_i/(n - r_i)$. Suppose we have sampled the first $l-1$ columns during the process; then we update the current number of columns left, $n - (l-1)$, and the current row sums $r_i^{(l)}$ and then generate column l with the CP sampling method using the weights $\frac{r_i^{(l)}}{n-(l-1)-r_i^{(l)}}$. Since the CP distribution $q(t_1)$ is not exactly the same as the target distribution $p(t_1)$ (the marginal distribution of the first column), we may want to adjust the weights in order to make $q(t_1)$ closer to $p(t_1)$. One easy adjustment is to use the set of weights $\left(\frac{r_i^{(l)}}{n-(l-1)-r_i^{(l)}}\right)^u$, where u can be chosen by the user. In our experience, however, $u = 1$ gave close to the best performances (see the examples in Section 6).

Asymptotic analysis of Good and Crook (1977) provides another intuition for the use of CP sampling. In particular, they give the following approximation to the number of zero-one matrices

with fixed row sums $\boldsymbol{r} = (r_1, r_2, \ldots, r_m)$ and column sums $\boldsymbol{c} = (c_1, c_2, \ldots, c_n)$:

$$|\Sigma_{\boldsymbol{rc}}| \sim \Delta_{\boldsymbol{rc}} \equiv \frac{\prod_{i=1}^{m} \binom{n}{r_i} \prod_{j=1}^{n} \binom{m}{c_j}}{\binom{mn}{M}}, \tag{7}$$

where $M = \sum_{i=1}^{m} r_i = \sum_{j=1}^{n} c_j$. Let $v(i_1, \ldots, i_{c_1})$ be the zero-one vector of length m which has i_k-th component equal to 1 for $1 \leq k \leq c_1$ and all other components equal to 0. For a particular configuration of the first column, $t_1 = v(i_1, \ldots, i_{c_1})$, we let $\boldsymbol{r}^{(2)} = (r_1^{(2)}, \ldots, r_m^{(2)})$ and $\boldsymbol{c}^{(2)} = (c_2, \ldots, c_n)$ be the updated row and column sums as defined in (5), respectively. Then, by approximation (7), we have

$$p(t_1 = v(i_1, \ldots, i_{c1})) \approx \frac{\Delta_{\boldsymbol{r}^{(2)}\boldsymbol{c}^{(2)}}}{\Delta_{\boldsymbol{rc}}} \propto \prod_{k=1}^{c_1} \frac{r_{i_k}}{n - r_{i_k}}.$$

Thus this approximation also suggests that we should sample the first column according to the CP distribution with weights proportional to $r_i/(n - r_i)$.

Békéssy, Békéssy, and Komlos (1972) gave another asymptotic result for $|\Sigma_{\boldsymbol{rc}}|$:

$$|\Sigma_{\boldsymbol{rc}}| \sim \Delta^*_{\boldsymbol{rc}} \equiv \frac{M!}{\prod_{i=1}^{m} r_i! \prod_{j=1}^{n} c_j!} e^{-\alpha(\boldsymbol{r},\boldsymbol{c})}, \tag{8}$$

where

$$\alpha(\boldsymbol{r}, \boldsymbol{c}) = 2\frac{[\sum_{i=1}^{m} \binom{r_i}{2}][\sum_{j=1}^{n} \binom{c_j}{2}]}{(\sum_{i=1}^{m} r_i)(\sum_{j=1}^{n} c_j)} = \frac{1}{2M^2} \sum_{i=1}^{m} (r_i^2 - r_i) \sum_{j=1}^{n} (c_j^2 - c_j).$$

This approximation is proved to work well for large and sparse zero-one matrices. By (8), we have

$$p(t_1 = v(i_1, \ldots, i_{c1})) \approx \frac{\Delta^*_{\boldsymbol{r}^{(2)}\boldsymbol{c}^{(2)}}}{\Delta^*_{\boldsymbol{rc}}} \propto \prod_{k=1}^{c_1} r_{i_k} e^{-\alpha(\boldsymbol{r}^{(2)}, \boldsymbol{c}^{(2)})}.$$

We note that

$$\alpha(\boldsymbol{r}^{(2)}, \boldsymbol{c}^{(2)}) = \frac{\sum_{j=2}^{n} (c_j^2 - c_j)}{(M - c_1)^2} \sum_{i=1}^{m} \binom{r_i^{(2)}}{2},$$

and $\sum_{i=1}^{m} \binom{r_i^{(2)}}{2} = \sum_{i=1}^{m} (r_i^2 - r_i)/2 + c_1 - \sum_{k=1}^{c_1} r_{i_k}$. Hence,

$$\frac{\Delta^*_{\boldsymbol{r}^{(2)}\boldsymbol{c}^{(2)}}}{\Delta^*_{\boldsymbol{rc}}} \propto \prod_{k=1}^{c_1} (r_{i_k} e^{d r_{i_k}}),$$

where $d = \sum_{j=2}^{n} (c_j^2 - c_j)/(M - c_1)^2$. Thus, another CP sampling distribution can be conducted with the weights proportional to $r_i e^{d r_i}$.

Although it is not clear if the approximations (7) or (8) are accurate for a given table, we observed that these two CP-based SIS methods performed well in all the settings we have tested and were extremely accurate when the marginal sums do not vary much.

3.3 Successive sampling

Another possible strategy for filling in the first column is *successive sampling* (Hájek, 1981), which draws the first unit without replacement with probability proportional to $w_1, w_2, \cdots, w_m$, then draws the second unit with probability proportional to the remaining weights, etc. The procedure continues until c units have been drawn. It is easy to see that the probability of obtaining a sample $S = \{u_{i_1}, \cdots, u_{i_c}\}$, without regard of ordering, is the following:

$$P(S) = \Big\{ \prod_{k=1}^{c} w_k \Big\} \sum_{(j_1,\cdots,j_c)} [W(W - w_{j_1})(W - w_{j_1} - w_{j_2}) \cdots (W - w_{j_1} - \cdots - w_{j_{c-1}})]^{-1},$$

where $W = \sum_{k=1}^{m} w_k$ and $\sum_{(j_1,\cdots,j_c)}$ denotes the summation over all permutations of $\{i_1, \cdots, i_c\}$. When applied to the examples in Section 6, the successive sampling method (combined with the analytic SIS method described in Section 4) resulted in small cv^2's for all the cases. For example, the cv^2 was around 0.7 for the finch data; and $cv^2 = 0.02$ for the "magic square" (the 12 by 12 table with row and column sums equal to 2). However, the computation of the normalizing constant W is very time-consuming because we have to sum over all the permutations of the elements in the sample. For the finch data, the SIS with successive sampling was nearly 2,000 times slower than that with CP sampling. See Chen (2001) for a fast method approximating the successive sampling, which performs well for sampling zero-one tables, and more details on other sampling strategies for zero-one tables.

4 A More Delicate SIS Method

Although the SIS procedure described in the previous section is already very effective, we found that sometimes the sampling cannot proceed after a few columns have been generated because no valid zero-one table can be produced. For example, suppose we want to sample tables with row sums 4, 4, 2, 1 and column sums 3, 3, 3, 1, 1. If we happen to draw the first column as $(1, 0, 1, 1)^T$ and the second column as $(1, 1, 1, 0)^T$, then we would have no way to sample the third column. In the following, we show that there exists an easy-to-check condition that guarantees the existence of subtables with the updated row and column sums. This condition helps us develop a more delicate SIS procedure for sampling more efficiently from $\Sigma_{\boldsymbol{rc}}$. Before we describe the procedure, we first provide some background. See Marshall and Olkin (1979) for more details and any unproven assertions.

DEFINITION 1 *For any* $\mathbf{x} = (x_1, \ldots, x_n) \in \mathcal{R}^n$, *we let* $x_{[1]} \geq \cdots \geq x_{[n]}$ *denote the components of* x

in decreasing order. For $\mathbf{x}, \mathbf{y} \in \mathcal{R}^n$, *we define*

$$\mathbf{x} \prec \mathbf{y} \quad \textit{if} \quad \begin{cases} \sum_{i=1}^{k} x_{[i]} \le \sum_{i=1}^{k} y_{[i]}, & k = 1, \ldots, n-1, \\ \sum_{i=1}^{n} x_{[i]} = \sum_{i=1}^{n} y_{[i]}. \end{cases}$$

When $\mathbf{x} \prec \mathbf{y}$, $\mathbf{x}$ *is said to be majorized by* $\mathbf{y}$ *(*$\mathbf{y}$ *majorizes* $\mathbf{x}$*).*

LEMMA 1 *Suppose* $(j_1, \ldots, j_n)$ *is a permutation of* $(1, \ldots, n)$. *Then* $\mathbf{x} \prec \mathbf{y}$ *implies that*

$$\begin{cases} \sum_{i=1}^{k} x_{j_i} \le \sum_{i=1}^{k} y_{[i]}, & k = 1, \ldots, n-1, \\ \sum_{i=1}^{n} x_{j_i} = \sum_{i=1}^{n} y_{[i]}. \end{cases} \tag{9}$$

DEFINITION 2 *Let* $a_1, a_2, \ldots, a_n$ *be nonnegative integers, and define*

$$a_j^* = \#\{a_i : a_i \ge j\}, \quad j = 1, 2, \ldots.$$

The sequence $a_1^*, a_2^*, a_3^*, \ldots$ *is said to be conjugate* to $a_1, a_2, \ldots, a_n$. Note that the conjugate sequence $\{a_i^*\}$ is always non-increasing and is independent of the order of the a_i.

LEMMA 2 *(Gale, 1957; Ryser, 1957) Let* $r_1, \ldots, r_m$ *be nonnegative integers not exceeding* n, *and* $c_1, \ldots, c_n$ *be nonnegative integers not exceeding* m. *A necessary and sufficient condition for the existence of an* $m \times n$ *zero-one table with row sums* $r_1, \ldots, r_m$ *and column sums* $c_1, \ldots, c_n$ *is that*

$$\boldsymbol{c} \equiv (c_1, \ldots, c_n) \prec (r_1^*, \ldots, r_n^*) \equiv \boldsymbol{r}^*;$$

or, equivalently, $\boldsymbol{r} \equiv (r_1, \ldots, r_m) \prec (c_1^*, \ldots, c_m^*) \equiv \boldsymbol{c}^*$.

Since the size of $\Sigma_{\boldsymbol{rc}}$ does not depend on the order of the row sums, we can arrange that $r_1 \ge \cdots \ge r_m$ without loss of any generality. Let the conjugate of $(c_1^{(1)}, \ldots, c_n^{(1)}) = (c_1, \ldots, c_n)$ be $(c_1^{(1)*}, \ldots, c_n^{(1)*})$. The conjugate of $(c_1^{(2)}, \ldots, c_{n-1}^{(2)})$, denoted by $(c_1^{(2)*}, \ldots, c_{n-1}^{(2)*})$ is

$$c_j^{(2)*} = \begin{cases} c_j^{(1)*} - 1, & 1 \le j \le c_1, \\ c_j^{(1)*}, & j > c_1. \end{cases}$$

From Lemma 2, we know that a necessary and sufficient condition for the existence of an $m \times (n-1)$ zero-one table with row sums $r_1^{(2)}, \ldots, r_m^{(2)}$ and column sums $c_1^{(2)}, \ldots, c_{n-1}^{(2)}$ is that

$$\boldsymbol{r}^{(2)} \equiv (r_1^{(2)}, \ldots, r_m^{(2)}) \prec (c_1^{(2)*}, \ldots, c_m^{(2)*}) \equiv \boldsymbol{c}^{(2)*},$$

i.e.,

$$\begin{cases} \sum_{i=1}^{k} r_{[i]}^{(2)} \le \sum_{i=1}^{k} c_i^{(2)*}, & k = 1, \ldots, m-1, \\ \sum_{i=1}^{m} r_{[i]}^{(2)} = \sum_{i=1}^{m} c_i^{(2)*}, \end{cases}$$

where recall that $r_{[i]}$ denotes the components of $\boldsymbol{r}$ in decreasing order. From Lemma 1, we know that $\boldsymbol{r}^{(2)} \prec \boldsymbol{c}^{(2)*}$ implies that

$$\begin{cases} \sum_{i=1}^{k} r_i^{(2)} \le \sum_{i=1}^{k} c_i^{(2)*}, & k = 1, \ldots, m-1, \\ \sum_{i=1}^{m} r_i^{(2)} = \sum_{i=1}^{m} c_i^{(2)*}. \end{cases} \tag{10}$$

Thus, (10) is clearly a necessary condition for the existence of the subtable with new row sums and column sums. We prove in the following theorem that it is also a sufficient condition.

THEOREM 2 *Let* $\boldsymbol{a}' = (a_1', \ldots, a_n')$, $\boldsymbol{b} = (b_1, \ldots, b_n)$. *Suppose* $a_1' \ge \cdots \ge a_n'$ *and* $b_1 \ge \cdots \ge b_n$ *are all nonnegative integers and there are* $d \ge 1$ *nonzero components in* $\boldsymbol{b}$. *Pick any* d', $1 \le d' \le d$, *nonzero components from* $\boldsymbol{b}$, *say* $b_{k_1}, \ldots, b_{k_{d'}}$. *Define* $\boldsymbol{b}' = (b_1', \ldots, b_n')$ *as*

$$b_j' = \begin{cases} b_j - 1, & \text{if } j = k_i \text{ for some } 1 \le i \le d', \\ b_j, & \text{otherwise} \end{cases}$$

and suppose $\boldsymbol{b}'$ *satisfies*

$$\begin{cases} \sum_{i=1}^{k} b_i' \le \sum_{i=1}^{k} a_i', & k = 1, \ldots, n-1, \\ \sum_{i=1}^{n} b_i' = \sum_{i=1}^{n} a_i'. \end{cases} \tag{11}$$

Then $\boldsymbol{b}'$ *is majorized by* $\boldsymbol{a}'$.

The proof of Theorem 2 is given in the Appendix. The reason this result is not entirely trivial is that $\boldsymbol{b}'$ is not necessarily ordered. For example, if $\boldsymbol{a}' = (4,4,2,1)$, $\boldsymbol{b} = (4,4,3,1)$ and $d' = 1$, then $\boldsymbol{b}'$ might be $(3,4,3,1)$. To see that the theorem implies that condition (10) is necessary and sufficient, we let $\boldsymbol{a}' = \boldsymbol{c}^{(2)*}$, $\boldsymbol{b} = \boldsymbol{r}^{(1)}$ $(= \boldsymbol{r})$, and $\boldsymbol{b}' = \boldsymbol{r}^{(2)}$, and let condition (10) hold. Theorem 2 implies that $\boldsymbol{b}' \prec \boldsymbol{a}'$, or equivalently, $\boldsymbol{r}^{(2)} \prec \boldsymbol{c}^{(2)*}$, which, according to Lemma 2, guarantees that there exists some zero-one subtable having the new row sums $\boldsymbol{r}^{(2)}$ and column sums $\boldsymbol{c}^{(2)}$.

Although we do not know $\boldsymbol{r}^{(2)}$ before we sample the first column, we can restate condition (10) from the current $\boldsymbol{r}^{(1)}$ and $\boldsymbol{c}^{(2)*}$. For each $1 \le k \le m$, we compare $\sum_{i=1}^{k} r_i$ and $\sum_{i=1}^{k} c_i^{(2)*}$:

- If $\sum_{i=1}^{k} r_i > \sum_{i=1}^{k} c_i^{(2)*}$, then we require that we must put at least $\sum_{i=1}^{k} r_i - \sum_{i=1}^{k} c_i^{(2)*}$ ones at or before the k-th row in the first column. For convenience, we may call k a knot;
- If $\sum_{i=1}^{k} r_i \le \sum_{i=1}^{k} c_i^{(2)*}$, then there is no restriction at the k-th row.

These two conditions can be summarized by two vectors: one vector records the positions of the knots, denoted by $(k[1], k[2], \ldots)$; the other vector records how many ones we must put before those knots, denoted by $(v[1], v[2], \ldots)$. In order to make the conditions easier to implement, we eliminate some redundant knots:

(i) If $v[j] \leq v[i]$ for some $j > i$, we ignore knot $k[j]$.

(ii) If $v[j] - v[i] \geq k[j] - k[i]$ for some $j > i$, then we ignore knot $k[i]$. If the restriction on knot $k[j]$ is satisfied, it will guarantee that the restriction on knot $k[i]$ is also satisfied.

Using the above conditions, we design the following sampling strategy.

- We are required to place at least $v[1]$ but no more than $\min(k[1], c_1)$ ones before row $k[1]$. So we assign equal probability to these choices, i.e.

$$q_1\{(\text{number of ones before row } \ k[1]) = i\} = \frac{1}{\min(k[1], c_1) - v[1] + 1}$$

  for $v[1] \leq i \leq \min(k[1], c_1)$.

- After the number of ones o_1 before row $k[1]$ is chosen according to the above distribution, we pick the o_1 positions between row 1 and row $k[1]$ using the CP sampling with weights $(r_i/(n - r_i))^u$, where u is usually chosen to be 1 (see Section 7). Sampling uniformly instead of using the CP distribution for this step can reduce the algorithm's efficiency by several orders of magnitude.

- After o_1 positions have been chosen for knot 1, we consider knot 2 conditional on the ones we have already placed before knot 1. Since we are required to place at least $v[2]$ ones before row $k[2]$, the number of ones o_2 we could put between knot 1 and knot 2 ranges from $\max(v[2] - o_1, 0)$ to $\min(k[2] - k[1], c_1 - o_1)$. We assign equal probability to all these choices for o_2. Then we pick the o_2 positions between row $k[1]$ and $k[2]$ using the CP sampling again.

- The procedure is continued until all the knots in column 1 have been considered.

- After we have completed the first column, we record the probability $q(t_1)$ of getting such a sample for the first column, update the row sums, rearrange the updated row sums in decreasing order, and repeat the procedure with the second column.

The reader may want to look ahead to Section 6 for examples.

5 Sampling from Contingency Tables

Sampling from contingency tables is much easier to implement than sampling from zero-one tables because there are fewer "restrictions." For a contingency table, given positive row sums $r_1, \ldots, r_m$

and column sums $c_1, \ldots, c_n$, the necessary and sufficient condition for the existence of a contingency table with such row and column sums is

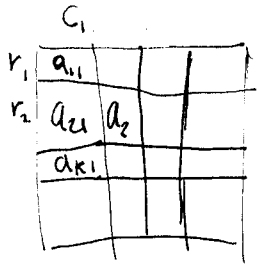
$$r_1 + r_2 + \cdots + r_m = c_1 + c_2 + \cdots + c_n \equiv M.$$

This is much simpler than the Gale-Ryser condition which makes the whole procedure much simpler to implement.

We still sample column by column as we did for zero-one tables. Suppose the element at the i-th row and the j-th column is a_{ij}. We start from the first column. We have that a_{11} must satisfy:

$$0 \le a_{11} \le r_1,$$

$$c_1 - \sum_{i=2}^{m} r_i = c_1 + r_1 - M \le a_{11} \le c_1.$$



So combining the two equations we have

$$\max(0, c_1 + r_1 - M) \le a_{11} \le \min(r_1, c_1).$$

It is also easy to see that this is the only condition a_{11} needs to satisfy. Recursively, suppose we have chosen $a_{i1} = a'_{i1}$ for $1 \le i \le k-1$. Then the only restriction on a_{k1} is

$$\max\left(0, \left(c_1 - \sum_{i=1}^{k-1} a'_{i1}\right) - \sum_{i=k+1}^{m} r_i\right) \le a_{k1} \le \min\left(r_k, c_1 - \sum_{i=1}^{k-1} a'_{i1}\right).$$

Thus, we need consider only the strategy for sampling a_{11} and the same strategy can be applied recursively to sample other cells.

If we collapse columns 2 to m and rows 2 to n to form a 2×2 table with a_{11} as the only variable, the uniform distribution on all tables implies that a_{11} is uniform in its range $[\max(0, c_1 + r_1 - M),\ \min(r_1, c_1)]$. However, if we consider both a_{11} and a_{21} simultaneously (the original table is collapsed into a 3×2 table), then for each $a_{11} = x$, the choices of a_{21} ranges from $\max(0, c_1 + r_1 + r_2 - M - x)$ to $\min(r_2, c_1 - x)$. Thus, if our goal is to sample a 3×2 table uniformly, we should have

$$P(a_{11} = x) \propto \min(r_2, c_1 - x) - \max(0, c_1 + r_1 + r_2 - M - x) + 1.$$

An analog of conditional Poisson sampling could be developed. Our examples in Section 6 show, however, that the simple uniform sampling of a_{11} seems to have already worked very well.

6 Applications and Simulations

6.1 Counting zero-one tables

Here we apply the SIS procedure described in Section 4 to estimate the number of zero-one tables with given row sums $r_1, r_2, \ldots, r_m$ and column sums $c_1, c_2, \ldots, c_n$. Since the ordering of the column or row sums does not affect the total number of tables, in the following examples we attempt to arrange the rows and columns in such a way that the cv^2 is made small. Some heuristic rules for arranging the rows and columns to achieve a low cv^2 will be discussed in Section 7. All examples were coded in C language and run on a Sun Ultra 60 workstation with a 359 MHz UltraSPARC-II processor.

We first tested our method on counting the number of 12×12 zero-one tables with all marginal sums equal to 2, which is a subset of the 12×12 "magic squares." For CP sampling with weights proportional to $\left(\frac{r_i^{(l)}}{n-(l-1)-r_i^{(l)}}\right)^{0.88}$, the cv^2 of the weights was 0.005. It took about one second to obtain 100 tables and their weights using the delicate SIS procedure in Section 4. The average of the weights gives rise to an estimate of $(2.195 \pm 0.005) \times 10^{16}$, where the number after the "$\pm$" sign is the standard error. For this table, the exact answer of $21,959,547,410,077,200$ is given in Wang and Zhang (1998). Although Wang and Zhang's formula provides a fast answer to this problem, it is often difficult to quickly compute their formula for larger zero-one tables.

Counting the number of tables with the same marginal sums as the finch data (Table 1) is a more challenging exercise. The last row of the original table is removed since it consists of all 1's and will not affect the counting. We ordered the 17 column sums from the largest to the smallest and applied the CP sampling with weights proportional to $\left(\frac{r_i^{(l)}}{n-(l-1)-r_i^{(l)}}\right)^{1.11}$, which gives a cv^2 of around 0.7. It took about 7.2 seconds to generate 1000 tables. With $100,000$ samples, which took about 12 minutes, we estimated the total number of zero-one tables to be $(6.72 \pm 0.02) \times 10^{16}$. As a verification, we obtained a more accurate estimate of 6.715061×10^{16} based on 10^8 samples. Here the exact answer was computed for us by David desJardins using a clever divide-and-conquer algorithm. His program (confirmed by an independent check) gives exactly $67,149,106,137,567,626$ tables. We see that the SIS algorithm gives a very accurate approximation. Figure 1 is the histogram of $1,000$ importance weights. It is seen that the weights are tightly distributed in a relatively small range. The ratio of the maximum weight over the median weight is about 10. If we use CP sampling with weights proportional to $\frac{r_i^{(l)}}{n-(l-1)-r_i^{(l)}}$, i.e., $u = 1$, the cv^2 is around 1.2, which is still very small.

To further challenge our method, we randomly generated a 50×50 table for which the probability

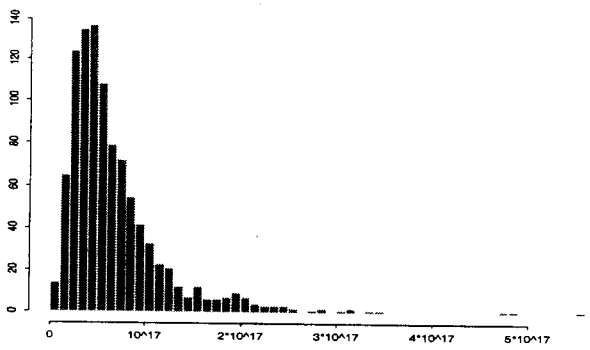


Figure 1: Histogram of 1,000 importance weights

for each cell to be 1 is 0.2. The row sums of the table are

$$10, 8, 11, 11, 13, 11, 10, 9, 7, 9, 10, 16, 11, 9, 12, 14, 12, 7, 9, 10, 10, 6, 11, 8, 9, 8, 14, 12, 5, 10, 10,$$
$$8, 7, 8, 10, 10, 14, 6, 10, 7, 13, 4, 6, 8, 9, 15, 11, 12, 10, 6$$

and the column sums are

$$9, 6, 12, 11, 9, 8, 8, 11, 9, 11, 13, 7, 10, 8, 9, 7, 8, 3, 10, 11, 13, 7, 5, 11, 10, 9, 10, 13, 9, 9, 7, 7, 6, 8,$$
$$10, 12, 8, 12, 16, 12, 15, 12, 13, 13, 10, 7, 12, 13, 6, 11.$$

We ordered the column sums from largest to smallest and used CP sampling with weights proportional to $\frac{r_i^{(l)}}{n-(l-1)-r_i^{(l)}}$ which gives a cv^2 of around 0.03. Based on 100 samples, which took a few minutes to generate, we estimate that the total number of zero-one tables with these marginal sums is $(7.7 \pm 0.1) \times 10^{432}$.

Since our method generally works well when the marginal sums do not vary much, we tried another example for which the marginal sums were forced to vary considerably. We generated a 50×50 table with cell (i, j) being 1 with probability $e^{-6.3(i+j-2)/(m+n-2)}$, which gave rise to the row sums:

$$14, 14, 19, 18, 11, 12, 12, 10, 13, 16, 8, 12, 6, 15, 6, 7, 12, 1, 12, 3, 8, 5, 9, 4, 2, 4, 1, 4, 4, 5,$$
$$2, 3, 3, 1, 1, 1, 2, 1, 1, 2, 1, 3, 3, 1, 3, 2, 1, 1, 1, 2$$

and the column sums

$$14, 13, 14, 13, 13, 12, 14, 8, 11, 9, 10, 8, 9, 8, 4, 7, 10, 9, 6, 7, 6, 5, 6, 8, 1, 6, 6, 3, 2, 3,$$
$$5, 4, 5, 2, 2, 2, 3, 2, 4, 3, 1, 1, 1, 3, 2, 2, 3, 5, 2, 5.$$

With the same SIS method as in the previous case, we had a cv^2 of 0.2. Based on 1,000 samples, we estimate the total number of zero-one tables with these margins as $(8.9 \pm 0.1) \times 10^{242}$. Based on 10,000 samples, the estimate is improved to $(8.78 \pm 0.05) \times 10^{242}$. Lastly, we estimated the total number of 100×100 zero-one tables with all marginal sums equal to two to be $(2.96 \pm 0.03) \times 10^{314}$ based on 100 Monte Carlo samples. The cv^2 in this case was 0.008, showing again that the SIS approach is extremely efficient.

6.2 Testing zero-one tables in ecology

We applied the SIS method to carry out the test suggested by Roberts and Stone (1990) for the finch data (see (1) in Section 1.1). The CP sampling used the weights proportional to $\left(\frac{r_i^{(l)}}{n-(l-1)-r_i^{(l)}}\right)^{1.11}$. The observed test statistic for the original finch data is 53.1. Based on 10,000 sampled tables, which took less than 2 minutes, we estimated that the p-value is 0.0004 with a standard error of 0.0002. Thus, we can reject the null hypothesis of a uniform distribution (conditional on the marginal sums) at a high level of significance. A histogram of the test statistic under the null distribution is given in Figure 2.

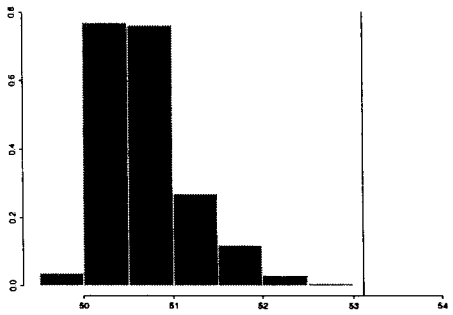


Figure 2: Histogram of the test statistic under null distribution. Vertical line is $\bar{S}^2(T_0)$.

Sanderson (2000) describes a method to generate zero-one tables with fixed margins, which he applies to the finch data, with an implied belief that the tables obtained are uniformly distributed. We note, however, that his method does not produce uniformly distributed tables. For example, for the set of 3×3 tables with marginal sums $(2, 2, 1)$ for both the columns and the rows, we found that the probability for Sanderson's method to generate one of the five possible configurations is 332/1512, but is 295/1512 to generate each of the remaining configurations. Because his sampling method does not generate tables uniformly, the conclusion of his statistical hypothesis testing is questionable.

6.3 Testing the Rasch model

Rasch (1960) proposed a simple linear logistic model to measure people's ability based on their answers to a dichotomous response test. Suppose n persons are asked to answer m questions (items). We can construct a zero-one matrix based on all the answers. A 1 in cell (i, j) means that the ith person answered the jth question correctly, and a 0 means otherwise. The Rasch model assumes that each person's ability is characterized by a parameter θ_i, each item's difficulty is characterized by a parameter β_j, and

$$P(x_{ij} = 1) = \frac{e^{\theta_i - \beta_j}}{1 + e^{\theta_i - \beta_j}}, \tag{12}$$

where x_{ij} is the ith person's answer to the jth question. The number of items answered correctly by each person (the column sums) are minimal sufficient statistics for the ability parameters and the number of people answering each item correctly (the row sums) are minimal sufficient statistics for the item difficulty parameters.

The Rasch model has numerous attractive features and is widely used for constructing and scoring educational and psychological tests (Fischer and Molenaar, 1995). There has been a considerable literature on testing the goodness of fit of the Rasch model (see Glas and Verhelst, 1995, for an overview). Most of the proposed tests rely on asymptotic theory for their validity, with which Rasch did not feel very comfortable (Andersen, 1995). In his seminal book, Rasch (1960) proposed a parameter-free "exact" test based on the conditional distribution of the zero-one matrix of responses with the observed marginal sums fixed. It is easy to see that under model (12), all the zero-one tables are uniformly distributed conditional on the row and column sums. Because of the difficulty of accurately approximating the distribution of test statistics under this uniform distribution, Rasch never implemented his approach. Besag and Clifford (1989) and Ponocny (2001) have studied the use of Monte Carlo methods to test the Rasch model. The conceptually simpler and more efficient SIS strategy developed in this article is also ideally suited for implementing Rasch's ideas. For example, Chen and Small (2003) show that in testing for item bias (Kelderman, 1989), the uniformly most powerful (UMP) unbiased test resulting from Rasch's idea (Ponocny 2001) is both "exact" and highly powerful. In a simulation study with 100 samples, it was shown that the SIS-based UMP unbiased test had a power of 0.90 at the 0.05 significance level, whereas the popular Mantel-Haenszel test proposed by Holland and Thayer (1988) only had power 0.40. Chen and Small (2003) also reported that the SIS approach is more efficient and accurate than the Monte Carlo method developed in Ponocny (2001).

6.4 Contingency tables

To illustrate the SIS method described in Section 5 for counting the number of contingency tables, we consider the two examples discussed in Diaconis and Gangolli (1995). The first example is a 5×3 table with row sums 10, 62, 13, 11, 39, and column sums 65, 25, 45, respectively. We observed that the smallest cv^2 (1.07) was achieved when the column sums are arranged from the largest to the smallest and row sums from the smallest to the largest. We obtained 100,000 Monte Carlo samples, which took less than a second and provided us with the estimate of 2.393×10^8 and its standard error 0.007×10^8. The true value of 239,382,173 is given in Diaconis and Gangolli (1995).

Besides counting the number of tables, the SIS method is also useful for carrying out certain hypothesis tests for contingency tables. The conditional volume test proposed by Diaconis and Efron (1985) addresses the question of whether the Pearson χ^2-statistic of a contingency table is "atypical" when the observed table is regarded as a draw from the uniform distribution over tables with the given marginal sums. The observed chi-square statistic for the 5×3 table described above is 72.1821. With one million Monte Carlo samples produced by our SIS method, which took about 20 seconds, we estimated the p-value for the conditional volume test to be 0.7606 ± 0.0005. Using a random-walk based Markov chain Monte Carlo algorithm, Diaconis and Gangolli (1995) estimated the p-value to be 0.7638 based on five independent chains, each with more than 2,000,000 steps. Their procedure took 23.5 minutes. They also gave the true value as 0.76086 based on a 12-hour exhaustive enumeration.

The second example is a 4×4 table with row sums 220, 215, 93, 64 and column sums 108, 286, 71, 127, respectively. Ordering the row sums from largest to smallest and the column sums from smallest to largest works best which gave us a cv^2 around 3.7. The mean estimate based on 1 million samples, which took 10 seconds, was $(1.225 \pm 0.002) \times 10^{15}$. The true value of 1,225,914,276,768,514 was given in Diaconis and Gangolli (1995). Diaconis and Efron (1985) gave a formula to approximately count the number of tables with given row and column sums that estimates 1.235×10^{16} tables for this example. Holmes and Jones (1996) estimated 1.226×10^{16} tables by the rejection method. We also performed the conditional volume test for this example. With one million Monte Carlo samples, we estimated the p-value to be 0.1521 ± 0.0009. In contrast, it took 28 days to run Markov chain Monte Carlo to obtain a 95% confidence interval of (0.14, 0.16) (Gangolli, 1991). Even after we take into consideration the fact that the machine we used is about 35 times faster than the machine Gangolli used (personal communication with Anil Gangolli), the SIS approach is still much faster than the MCMC method for this problem.

Holmes and Jones (1996) gave another example, with five row sums 9, 49, 182, 478, and 551,

and five column sums 9, 309, 355, 596 and 1269, respectively, and showed that the approximation formula in Diaconis and Efron (1985) does not work well. A distinctive feature of their example is that both the row and column sums have a very small value. We tried SIS on this example, using the original order of the rows and ordering the column sums in decreasing order. The cv^2 was around 7 so that the effective sample size was about $N/(1+7) = 12.5\% \times N$. Holmes and Jones' first algorithm has an acceptance rate of 9.7% and the revised one, 12.5%. In terms of effective sample size, our algorithm is as efficient as their revised algorithm. However the SIS approach is simpler to implement and easier to understand than the revised algorithm of Holmes and Jones, which requires calculating the coefficients of a product of some very large polynomials.

For Holmes and Jones' example, we estimated the total number of tables to be $(3.40 \pm 0.03) \times 10^{16}$ based on 100,000 SIS samples. This took just a second to produce. With one million samples, which took 10 seconds, our estimate was improved to $(3.384 \pm 0.009) \times 10^{16}$. Several estimates based on 10^8 samples were all around 3.383×10^{16}. In contrast, the estimates given in Holmes and Jones (1996) are 3.346×10^{16} and 3.365×10^{16}, which we believe underestimate the true number of tables.

7 Discussion

In this paper, we have developed a set of sequential importance sampling strategies for computing with zero-one or contingency tables. Our results showed that these approaches are both very efficient and simple to implement. Two distinctive features of our approach to sampling zero-one tables are (i) it guarantees each sample produces a valid table, thus avoid wasting computational resources, and (ii) it uses the CP sampling as the trial distribution to greatly increase its efficiency.

For CP sampling, we used weights proportional to $\left(\frac{r_i^{(l)}}{n-(l-1)-r_i^{(l)}}\right)^u$ where $u > 0$ can be chosen by the user. For all the zero-one tables we have tested, the choice of $u = 1$ has worked very well. Although some small variation of u (e.g., ranging from 0.8 to 1.2) may improve the efficiency of the SIS a bit, we do not expect to see any dramatic effect. We suggest that the user starts with $u = 1$, and then tries out values 0.8, 0.9, 1.1, and 1.2, of which one will usually give close to the optimal performance. One may also use a small sample size to estimate the cv^2 and choose the u that gives the the lowest cv^2. This should not take more than a few seconds.

We used several different orderings of row sums and column sums. Our experience is that for zero-one tables, it is best to order the column sums from largest to smallest. This makes intuitive sense because when we start with columns with many 1's, we do not have many choices

and $q(t_l|t_{l-1},\ldots,t_1)$ must be close to $p(t_l|t_{l-1},\ldots,t_1)$. After such columns have been sampled, the updated row sums will be reduced greatly which will cause $q(t_l|t_{l-1},\ldots,t_1)$ to be closer to $p(t_l|t_{l-1},\ldots,t_1)$. Because of the way we do the sampling, we need to order the row sums from largest to smallest. Another option is to sample rows instead of columns. Our experience is that if the number of rows is greater than the number of columns, sampling rows gives better results.

For contingency tables, we found that listing the column sums in decreasing order and listing the row sums in increasing order works the best. The intuition is similar to that for zero-one tables. A surprising fact for contingency tables is that given a certain ordering of the row and column sums, sampling the columns is the same as sampling the rows. It is not difficult to check this fact by carrying out our sampling method. Thus, we do not need to worry about whether exchanging the roles of rows and columns provides better performance.

Since the tables produced by the SIS approach described here have a distribution very close to the target one (as evidenced by the low cv^2 values), the SIS method is markedly better than the available MCMC approach, which typically has a very long autocorrelation time especially for large tables. This advantage of the SIS is reflected not only by a more accurate Monte Carlo approximation, but also by a more reliable estimate of the standard error of this approximation. Furthermore, estimating the normalizing constant of the target distribution is a rather straightforward step for the SIS method, but is much more difficult for MCMC strategies.

8 Acknowledgment

The authors thank David desJardins, Dylan Small and Charles Stein for inspiring discussions and helpful suggestions. This work was partly supported by the National Science Foundation grants DMS-0203762 and DMS-0204674.

APPENDIX: PROOF OF THEOREM 2

Suppose there are l distinct values among $\{b_1,\ldots,b_n\}$ and we assume that $i_1 < \cdots < i_l$ are the jump points, i.e.,

$$b_{i_{k-1}+1} = \cdots = b_{i_k} > b_{i_k+1}, \quad k = 1,\ldots,l-1,$$
$$b_{i_{l-1}+1} = \cdots = b_{i_l},$$

where $i_0 = 0$, $i_l = n$. Because $b'_i = b_i$ or $b'_i = b_i - 1$ and $\boldsymbol{b}$ is ordered from largest to smallest, it is clear that if we have $i_{k-1} < i \le i_k$ and $i_k < j \le i_{k+1}$ for any i, j, k, then $b'_i \ge b'_j$. But within each

block from b'_{i_k} to $b'_{i_{k+1}}$, some of the b'_i's are equal to b_{i_k} and the others are equal to $b_{i_k}-1$. In other words, the b'_i's may not be ordered from largest to smallest in each block. If there is a j such that $i_{k-1} < j < i_k$ and

$$b'_j = b_{i_k} - 1, \quad b'_{j+1} = b_{i_k},$$

we will show that we can switch b'_j and b'_{j+1} and still maintain property (11). There are two different cases to consider:

Case (i): $\sum_{i=1}^{j} b'_i < \sum_{i=1}^{j} a'_i$.

In this case, of course property (11) still holds after we switch b'_j and b'_{j+1} and obtain

$$b'_j = b_{i_k}, \quad b'_{j+1} = b_{i_k} - 1.$$

Case (ii): $\sum_{i=1}^{j} b'_i = \sum_{i=1}^{j} a'_i$, which we will show can never happen.

Since $\sum_{i=1}^{j+1} b'_i \le \sum_{i=1}^{j+1} a'_i$, we have $a'_{j+1} \ge b'_{j+1} = b_{i_k}$. But since the a'_i are monotone non-increasing, we have

$$a'_{i_{k-1}+1} \ge \cdots \ge a'_j \ge a'_{j+1} \ge b_{i_k}.$$

Since $b'_i \le b_{i_k}$ for $i_{k-1}+1 \le i < j$, and $b'_j = b_{i_k} - 1$, we must have

$$\sum_{i=i_{k-1}+1}^{j} b'_i < \sum_{i=i_{k-1}+1}^{j} a'_i. \tag{A.1}$$

Combining (A.1) with the fact that $\sum_{i=1}^{i_{k-1}} b'_i \le \sum_{i=1}^{i_{k-1}} a'_i$, we finally have

$$\sum_{i=1}^{j} b'_i < \sum_{i=1}^{j} a'_i,$$

which contradicts the assumption that $\sum_{i=1}^{j} b'_i = \sum_{i=1}^{j} a'_i$.

The preceding arguments imply that we can always switch b'_j and b'_{j+1} and maintain (11) if

$$b'_j = b_{i_k} - 1, \quad b'_{j+1} = b_{i_k}.$$

After a finite number of such switches, all the b'_i in this block must be ordered from largest to smallest: $b'_{i_{k-1}+1} \ge \cdots \ge b'_{i_k}$, which leads easily to the conclusion that $\boldsymbol{b}' \prec \boldsymbol{a}'$.

REFERENCES

Andersen, E. B. (1995). What Georg Rasch would have thought about this book. In *Rasch Models: Their Foundations, Recent Developments and Applications* (Edited by G.H. Fischer and I.W. Molenaar). Springer-Verlag, New York.

Békéssy, A., Békéssy, P., and Komlos, J. (1972). Asymptotic enumeration of regular matrices. *Studia Scientiarum Mathematicarum Hungarica.* **7** 343-353.

Balmer, D. (1988). Recursive enumeration of $r \times c$ tables for exact likelihood evaluation. *Applied Statistics.* **37** 290-301.

Besag, J. and Clifford, P. (1989). Generalized Monte-Carlo significance tests. *Biometrika.* **76** 633-642.

Boyett, J.M. (1979). Algorithm AS144. Random $R \times S$ tables with given row and column totals. *Applied Statistics.* **28** 329-332.

Brewer, K.R.W. and Hanif, M. (1983). *Sampling with unequal probabilities.* Lecture Notes in Statistics. Springer-Verlag, New York.

Chen, S.X. and Liu, J.S. (1997). Statistical applications of the Poisson-Binomial and conditional Bernoulli distributions. *Statistica Sinica* **7** 875-892.

Chen, X.H., Dempster, A.P. and Liu, J.S. (1994). Weighted finite population sampling to maximize entropy. *Biometrika* **81** 457-469.

Chen, Y. (2001). Sequential importance sampling with resampling: theory and applications. Ph.D. dissertation, Department of Statistics, Stanford University.

Chen, Y. and Small, D. (2003). Testing the Rasch model via sequential importance sampling. Tentatively accepted for *Psychometrika.*

Cobb, G.W. and Chen, Y.-P. (2003). An application of Markov chain Monte Carlo to community ecology. *American Mathematical Monthly.* **110**(4) 265-288.

Connor, E.F. and Simberloff, D. (1979). The assembly of species communities: chance or competition? *Ecology.* **60** 1132-1140.

Cook, R.R. and Quinn, J.F. (1995). The influence of colonization in nested species subsets. *Oecologia* **102** 413-424.

Diaconis, P. and Efron, B. (1985). Testing for independence in a two-way table: new interpretations of the chi-square statistic. *Ann. Statist.* **13** 845-874.

Diaconis, P. and Gangolli, A. (1995). Rectangular arrays with fixed margins. In *Discrete Probability and Algorithms* (D. Aldous, P. Diaconis, J. Spencer and J.M. Steele, eds) Springer, New York.

Fischer, G.H. and Molenaar, I.W. (1995). *Rasch Models: Foundations, Recent Developments, and Applications.* Springer-Verlag, New York.

Gail, M. and Mantel, N. (1977). Counting the number of $r \times c$ contingency tables with fixed margins. *Jour. Amer. Statist. Assoc.* **72** 859-862.

Gale, D.(1957). A theorem on flows in networks. *Pacific J. Math.* **7**, 1073-1082.

Gangolli, A. (1991). Convergence bounds for Markov chains and applications to sampling, Ph.D. dissertation, Computer Science Department, Stanford University.

Glas, C. A. W. and Verhelst, N. D. (1995). Testing the Rasch Model. In *Rasch Models: Their Foundations, Recent Developments and Applications* (Edited by G.H. Fischer and I.W. Molenaar). Springer-Verlag, New York.

Good, I.J. and Crook, J. (1977). The enumeration of arrays and a generalization related to contingency tables. *Discrete Math.* **19** 23-65.

Hájek, J. (1981). *Sampling from a finite population.* Marcel Dekker, Inc., New York and Basel.

Holland, P.W. and Thayer, D.T. (1988). Differential item performance and the Mantel-Haenszel procedure. In *Test Validity* (Edited by H. Wainer and H.I. Braun). Erlbaum, Hillsdale.

Holmes, R.B. and Jones, L.K. (1996). On uniform generation of two-way tables with fixed margins and the conditional volume test of Diaconis and Efron. *Ann. Statist.* **24** 64-68.

Kelderman, H. (1989). Item bias detection using loglinear IRT. *Psychometrika.* **54** 681-697.

Kong, A., Liu, J.S. and Wong, W.H. (1994). Sequential imputations and bayesian missing data problems. *Journal of the American Statistical Association* **89**(425) 278-288.

Lehmann, E. L. (1986). *Testing Statistical Hypotheses.* John Wiley & Sons, New York.

Liu, J.S. and Chen, R. (1998). Sequential monte-carlo methods for dynamic systems. *Journal of the American Statistical Association.* **93**(443) 1032-1044.

Marshall, A.W. and Olkin, I.(1979). *Inequalities: Theory of Majorization and Its Applications.* Academic Press, New York.

Manly, B.F. (1995). A note on the analysis of species co-occurrences. *Ecology.* **76**(4) 1109-1115.

Patefield, W.M. (1981). An efficient method of generating random $r \times c$ tables with given row and column totals. *Applied Statistics.* **30** 91-97.

Ponocny, I. (2001). Nonparameteric goodness-of-fit tests for the Rasch model. *Psychometrika.* **66** 437-460.

Rao, A.R., Jana, R. and Bandyopadhyay, S. (1996). A Markov chain Monte Carlo method for generating random (0,1)-matrices with given marginals. *Sankhya Ser. A* **58** 225-242.

Rasch, G. (1960). *Probabilistic Models for Some Intelligence and Attainment Tests.* Danish Institute for Educational Research, Copenhagen.

Roberts, A., Stone, L (1990). Island-sharing by archipelago species. *Oecologia* **83** 560-567.

Ryser, H.J.(1957). Combinatorial properties of matrices of zeros and ones. *Canad. J. Math.* **9,** 371-377.

Sanderson, J.G. (2000). Testing Ecological Patterns. *American Scientist.* **88** 332-339.

Sanderson, J.G., Moulton, M.P. and Selfridge, R.G. (1998). Null matrices and the analysis of species co-occurrences. *Oecologia.* **116** 275-283.

Snijders, T.A.B. (1991). Enumeration and simulation methods for 0-1 matrices with given marginals. *Psychometrika.* **56**, No. 3, 397-417.

Sullaway, F.J. (1982). Darwin and his finches: The evolution of a legend. *Journal of the History of Biology.* **15**(1) 1-53.

Wang, B.Y. (1988). Precise number of $(0,1)$-matrices in $\mathcal{U}(R,S)$. *scientia Sinica, Series A* No.1 1-6.

Wang, B.Y. and Zhang, F. (1998). On the precise number of $(0,1)$-matrices in $\mathcal{U}(R,S)$. *Discrete Math..* **187** 211-220.

Wilson, J.B. (1987). Methods for detecting non-randomness in species co-occurrences: a contribution. *Oecologia.* **73** 579-582.