Ann Inst Stat Math (2013) 65:763–783
DOI 10.1007/s10463-012-0392-7

Estimating the number of zero-one multi-way tables via sequential importance sampling

Jing Xi · Ruriko Yoshida · David Haws

Received: 1 May 2012 / Revised: 22 October 2012 / Published online: 30 December 2012

Abstract In 2005, Chen et al. introduced a sequential importance sampling (SIS) procedure to analyze zero-one two-way tables with given fixed marginal sums (row and column sums) via the conditional Poisson (CP) distribution. They showed that compared with Monte Carlo Markov chain (MCMC)-based approaches, their importance sampling method is more efficient in terms of running time and also provides an easy and accurate estimate of the total number of contingency tables with fixed marginal sums. In this paper, we extend their result to zero-one multi-way (d-way, $d \geq 2$) contingency tables under the no d-way interaction model, i.e., with fixed $d-1$ marginal sums. Also, we show by simulations that the SIS procedure with CP distribution to estimate the number of zero-one three-way tables under the no three-way interaction model given marginal sums works very well even with some rejections. We also applied our method to Samson's monks data set.

Keywords Categorical data analysis · Conditional Poisson · Counting problem · No three-way interaction

J. Xi
Statistics Department, University of Kentucky, 325 Multidisplinary
Science Building, Lexington, KY 40506-0082, USA
e-mail: jing.xi@uky.edu

R. Yoshida (✉)
Statistics Department, University of Kentucky, 325D Multidisplinary
Science Building, Lexington, KY 40506-0082, USA
e-mail: ruriko.yoshida@uky.edu

D. Haws
Computational Genetics, IBM, Thomas J. Watson Research Center,
Yorktown Heights, NY 10598, USA
e-mail: dchaws@gmail.com

1 Introduction

Sampling zero-one constrained contingency tables find applications in combinatorics (Huber 2006), statistics of social networks (Chen 2007; Snijders 1991), and regulatory networks Dinwoodie (2008). In 2005, Chen et al. (2005) introduced a sequential importance sampling (SIS) procedure to analyze zero-one two-way tables with given fixed marginal sums (row and column sums) via the conditional Poisson (CP) distribution. It proceeds by simply sampling cell entries of the zero-one contingency table sequentially for each row such that the final distribution approximates the target distribution. This method will terminate at the last column and sample independently and identically distributed (iid) tables from the proposal distribution. Thus, the SIS procedure does not require expensive or prohibitive pre-computations, as is the case of computing Markov bases for the Monte Carlo Markov Chain (MCMC) approach. Also, when attempting to sample a single table, if there is no rejection, the SIS procedure is guaranteed to sample a table from the distribution, whereas in an MCMC approach the chain may require a long time to run in order to satisfy the independent condition.

For sampling multi-way contingency tables without zero-one constraints using SIS procedures, i.e., each cell of the table is not bounded by an upper bound, which is equal to one, much work has been done. Chen et al. (2005) introduced also an SIS procedure for sampling multi-way contingency tables without zero-one constraints and Chen et al. (2006) gave an excellent algebraic interpretation of precisely when an interval will equal the support of the marginal distribution using *Markov bases*. Dinwoodie and Chen (2011) used linear programming and sequential normal sampling to develop a new SIS procedure to sample a multi-way contingency table. However, one cannot simply apply these methods to sampling zero-one multi-way contingency tables. In order to apply these methods directly to sampling zero-one multi-way contingency tables, we have to introduce the "slack" variables in the system of the linear equations and, if we are forced to do so, we have to double the number of variables so that the problem can become exponentially harder. For example, to sample one $10 \times 10 \times 10$ zero-one table in Example 19 with $s = 2$, the original SIS procedure with slack variables has to solve $2 \cdot (9 \cdot 9 \cdot 9)$ many integer programming problems with $O(2000)$ many variables. Note that solving an integer programming problem is NP-complete if we vary the number of variables (Garey and Johnson 1979). With our software written in R (R-Project-Team 2011), we have sampled $1,000$ many tables in about 450 s, while the original SIS procedure with slack variables (written in C++) sampled one table in 1,994 s (so in order to sample 1,000 tables, it will take about 23 days). This is the reason why Chen et al. (2005) developed an SIS procedure with the CP specifically for sampling zero-one two-way contingency tables. Therefore, we have to think of the problem for sampling zero-one multi-way contingency tables separately without applying the existing methods for sampling contingency tables without zero-one constraints.

Chen (2007) extended their SIS procedure to sample zero-one two-way tables with given fixed row and column sums with structures, i.e., some cells are constrained to be zero or one.

In this paper, we also extended the results from (Chen et al. 2005; Chen 2007) to zero-one multi-way (d-way, $d \geq 2$) contingency tables under the no d-way interaction model, i.e., with fixed $d - 1$ marginal sums.

This paper is organized as follows: in Sect. 2, we outline basics of the SIS procedure. In Sect. 3, we focus on the SIS procedure with CP distribution on three-way tables under no three-way interaction model. This model is particularly important, since if we are able to count or estimate the number of tables under this model, then this is equivalent to estimating the number of *lattice points* in any *polytope* (De Loera and Onn 2006). This means that if we can estimate the number of three-way zero-one tables under this model, then we can estimate the number of any zero-one tables by using De Loera and Onn's bijection mapping.

Let $\mathbf{X} = (X_{ijk})$ of size (m, n, l), where $m, n, l \in \mathbb{N}$ and $\mathbb{N} = \{1, 2, \ldots, \}$, are a table of counts whose entries are independent Poisson random variables with canonical parameters $\{\theta_{ijk}\}$. Here, $X_{ijk} \in \{0, 1\}$. Consider the generalized linear model,

$$\theta_{ijk} = \lambda + \lambda_i^M + \lambda_j^N + \lambda_k^L + \lambda_{ij}^{MN} + \lambda_{ik}^{ML} + \lambda_{jk}^{NL} \tag{1}$$

for $i = 1, \ldots, m$, $j = 1, \ldots, n$, and $k = 1, \ldots, l$ where M, N, and L denote the nominal-scale factors. This model is called the *no three-way interaction model.*

Notice that the sufficient statistics under the model in (1) are the *two-way marginals*, that is:

$$\begin{aligned} X_{+jk} &:= \sum_{i=1}^{m} X_{ijk}, (j = 1, 2, \ldots, n, k = 1, 2, \ldots, l), \\ X_{i+k} &:= \sum_{j=1}^{n} X_{ijk}, (i = 1, 2, \ldots, m, k = 1, 2, \ldots, l), \\ X_{ij+} &:= \sum_{k=1}^{l} X_{ijk}, (i = 1, 2, \ldots, m, j = 1, 2, \ldots, n). \end{aligned} \tag{2}$$

Hence, the conditional distribution of the table counts given the margins is the same regardless of the values of the parameters in the model.

In Sect. 4, we generalize the SIS procedure on zero-one two-way tables in (Chen et al. 2005; Chen 2007) to zero-one multi-way (d-way, $d \geq 2$) contingency tables under the no d-way interaction model, i.e., with fixed $d - 1$ marginal sums. In Sects. 5 and 6, we show some simulation results with our software which is available at http://www.polytopes.net/code/CP. Finally, we end with some discussions.

2 Sequential importance sampling

Let Σ be the set of all tables satisfying marginal conditions. In this paper, we assume that $\Sigma \neq \emptyset$. Let $P(\mathbf{X})$ for any $\mathbf{X} \in \Sigma$ be the uniform distribution over Σ, so $p(\mathbf{X}) = 1/|\Sigma|$. Let $q(\cdot)$ be a trial distribution such that $q(\mathbf{X}) > 0$ for all $\mathbf{X} \in \Sigma$. Then we have

$$\mathbb{E}\left[\frac{1}{q(\mathbf{X})}\right] = \sum_{\mathbf{X} \in \Sigma} \frac{1}{q(\mathbf{X})} q(\mathbf{X}) = |\Sigma|.$$

Thus, we can estimate $|\Sigma|$ by

$$\widehat{|\Sigma|} = \frac{1}{\text{Num}} \sum_{i=1}^{\text{Num}} \frac{1}{q(\mathbf{X_i})},$$

where $\mathbf{X_1}, \ldots, \mathbf{X}_{\text{Num}}$ are tables drawn iid from $q(\mathbf{X})$. Here, this proposed distribution $q(\mathbf{X})$ is the distribution (approximate) to sample tables via the SIS procedure.

We vectorized the table $\mathbf{X} = (x_1, \ldots, x_t)$ and, by the multiplication rule, we have

$$q(\mathbf{X} = (x_1, \ldots, x_t)) = q(x_1)q(x_2|x_1)q(x_3|x_2, x_1) \ldots q(x_t|x_{t-1}, \ldots, x_1).$$

Since we sample each cell count of a table from an interval, we can easily compute $q(x_i|x_{i-1}, \ldots, x_1)$ for $i = 2, 3, \ldots, t$.

When we have rejections, this means that we are sampling tables from a bigger set Σ^* such that $\Sigma \subset \Sigma^*$. In this case, as long as the conditional probability $q(x_i|x_{i-1}, \ldots, x_1)$ for $i = 2, 3, \ldots$ and $q(x_1)$ are normalized, $q(\mathbf{X})$ is normalized over Σ^* since

$$\begin{aligned}\sum_{\mathbf{X}\in\Sigma^*} q(\mathbf{X}) &= \sum_{x_1,\ldots,x_t} q(x_1)q(x_2|x_1)q(x_3|x_2, x_1) \cdots q(x_t|x_{t-1}, \ldots, x_1)\\ &= \sum_{x_1} q(x_1)\left[\sum_{x_2} q(x_1|x_2)\left[\cdots\left[\sum_{x_t} q(x_t|x_{t-1}, \ldots, x_1)\right]\right]\right]\\ &= 1.\end{aligned}$$

Thus, we have

$$\mathbb{E}\left[\frac{\mathbb{I}_{\mathbf{X}\in\Sigma}}{q(\mathbf{X})}\right] = \sum_{\mathbf{X}\in\Sigma^*} \frac{\mathbb{I}_{\mathbf{X}\in\Sigma}}{q(\mathbf{X})} q(\mathbf{X}) = |\Sigma|,$$

where $\mathbb{I}_{\mathbf{X}\in\Sigma}$ is an indicator function for the set Σ. By the law of large numbers, $\widehat{|\Sigma|}$ is unbiased as the sample size gets large (Blitzstein and Diaconis 2010).

3 Sampling from the conditional Poisson distribution

Let

$$Z = (Z_1, \ldots, Z_l)$$

be independent Bernoulli trials with probability of successes $p = (p_1, \ldots, p_l)$. Then the random variable

$$S_Z = Z_1 + \cdots + Z_l$$

is a Poisson-binomial distribution.

We say the column of entries for the marginal $X_{i_0,j_0,+}$ of $\mathbf{X}$ is the (i_0, j_0)th column of $\mathbf{X}$ (equivalently, we say (i_0, k_0)th column for the marginal $X_{i_0+k_0}$ and (j_0, k_0)th column for the marginal $X_{+j_0k_0}$). Consider the (i_0, j_0)th column of the table $\mathbf{X}$ for some $i_0 \in \{1, \ldots, m\}$, $j_0 \in \{1, \ldots, n\}$ with the marginal $l_0 = X_{i_0j_0+}$. Also, we let $r_k = X_{i_0+k}$ and $c_k = X_{+j_ok}$. Now let $w_k = p_k/(1 - p_k)$ where $p_k \in (0,\ 1)$. Then,

$$P(Z_1 = z_1, \ldots, Z_l = z_l | S_Z = l_0) \propto \prod_{k=1}^{l} w_k^{z_k}. \tag{3}$$

Thus, for sampling a zero-one table with fixed marginals X_{+jk}, X_{i+k} for $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$, and $k = 1, 2, \ldots, l$, for $X_{i_0j_0+}$ for each $i_0 \in \{1, \ldots, m\}$ and $j_0 \in \{1, \ldots, n\}$, (or one can do each $X_{i_0+k_0}$ or $X_{+j_0k_0}$ instead by similar way), one decides which entries are ones (basically, there are $\binom{l}{l_0}$ many choices) using the conditional Poisson distribution above. We sample these cell entries with ones (say l_0 many entries with ones) in the (i_0, j_0)th column for the L factor with the following probability. Let A_k, for $k = 1, \ldots, l_0$, be the set of selected entries. Thus $A_0 = \emptyset$, and A_{l_0} is the final sample that we obtain. At the kth step of the drafting sampling ($k = 1, \ldots, l_0$), a unit $j \in A_{k-1}^c$ is selected into the sample with probability

$$P(j, A_{k-1}^c) = \frac{w_j R(l_0 - k, A_{k-1}^c - j)}{(l_0 - k + 1) R(l_0 - k + 1, A_{k-1}^c)},$$

where

$$R(s, A) = \sum_{B \subset A, |B| = s} \left(\prod_{i \in B} w_i \right).$$

For sampling a zero-one three-way table $\mathbf{X}$ with given two-way marginals, X_{ij+}, X_{i+k}, and X_{+jk} for $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$, and $k = 1, 2, \ldots, l$, we sample for the (i_0, j_0)th column of the table $\mathbf{X}$ for each $i_0 \in \{1, \ldots, m\}$, $j_0 \in \{1, \ldots, n\}$. We set

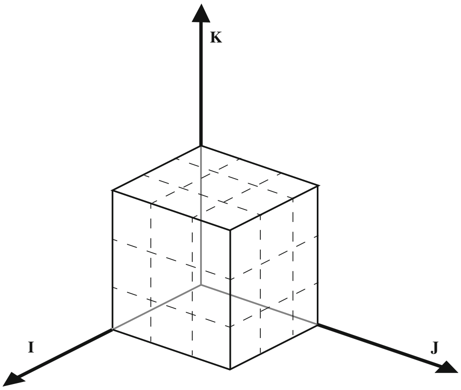
$$p_k = \frac{r_k \cdot c_k}{r_k \cdot c_k + (n - r_k)(m - c_k)}. \tag{4}$$

Thus we have

$$w_k = \frac{r_k \cdot c_k}{(n - r_k)(m - c_k)}. \tag{5}$$

Remark 1 We assume that we do not have the trivial cases, namely, $1 \leq r_k \leq n - 1$ and $1 \leq c_k \leq m - 1$.

Fig. 1 An example of a $3 \times 3 \times 3$ table



Theorem 2 *For the uniform distribution over all $m \times n \times l$ zero-one tables with given marginals $r_k = X_{i_0+k}, c_k = X_{+j_0k}$ for $k = 1, 2, \ldots, l$, and a fixed marginal for the factor L, l_0* (Fig. 1)*, the marginal distribution of the fixed marginal l_0 is the same as the conditional distribution of Z defined by* (3) *given $S_Z = l_0$ with*

$$p_k = \frac{r_k \cdot c_k}{r_k \cdot c_k + (n - r_k)(m - c_k)}.$$

Proof We start by giving an algorithm for generating tables uniformly from all $m \times n \times l$ zero-one tables with given marginals $r_k,\ c_k$ for $k = 1, 2, \ldots, l$, and a fixed marginal for the factor $L,\ l_0$.

1. For $k = 1, \ldots, l$ consider the kth layer of $m \times n$ tables. We randomly choose r_k positions in the (i_0, k)th column and c_k positions in the (j_0, k)th column, and put 1s in those positions. The choices of positions are independent across different layers.
2. Accept those tables with given column sum l_0.

It is easy to see that tables generated by this algorithm are uniformly distributed over all $m \times n \times l$ zero-one tables with given marginals $r_k,\ c_k$ for $k = 1, 2, \ldots, l$, and a fixed marginal for the factor $L,\ l_0$ for the (i_0, j_0)th column of the table $\mathbf{X}$. We can derive the marginal distribution of the (i_0, j_0)th column of $\mathbf{X}$ based on this algorithm. In Step 1, we choose the cell at position $(i_0, j_0, 1)$ to put 1 in with the probability:

$$\frac{\binom{n-1}{r_1-1}\binom{m-1}{c_1-1}}{\binom{n-1}{r_1-1}\binom{m-1}{c_1-1} + \binom{n-1}{r_1}\binom{m-1}{c_1}} = \frac{r_1 \cdot c_1}{r_1 \cdot c_1 + (n - r_1)(m - c_1)}.$$

Because the choices of positions are independent across different layers, after Step 1 the marginal distribution of the (i_0, j_0)th column is the same as the distribution of Z defined by (3) with

$$p_k = \frac{\binom{n-1}{r_k-1}\binom{m-1}{c_k-1}}{\binom{n-1}{r_k-1}\binom{m-1}{c_k-1} + \binom{n-1}{r_k}\binom{m-1}{c_k}} = \frac{r_k \cdot c_k}{r_k \cdot c_k + (n - r_k)(m - c_k)}.$$

Step 2 rejects the tables whose (i_0, j_0)th column sum is not l_0. This implies that after Step 2, the marginal distribution of the (i_0, j_0)th column is the same as the conditional distribution of Z defined by (3) with

$$p_k = \frac{r_k \cdot c_k}{r_k \cdot c_k + (n - r_k)(m - c_k)}.$$

□

Remark 3 The sequential importance sampling via CP for sampling a two-way zero-one table defined in (Chen et al. 2005) is a special case of our SIS procedure. We can induce p_k defined in (4) and the weights defined in (5) to the weights for two-way zero-one contingency tables defined in (Chen et al. 2005). Note that when we consider two-way zero-one contingency tables, we have $c_k = 1$ for all $k = 1, \ldots, l$ and for all $j_0 = 1, \ldots, n$ (or $r_k = 1$ for all $k = 1, \ldots, l$ and for all $i_0 = 1, \ldots, m$), and $m = 2$ (or $n = 2$, respectively). Therefore, when we consider the two-way zero-one tables, we get

$$p_k = \frac{r_k}{n}, \quad w_k = \frac{r_k}{n - r_k},$$

or respectively

$$p_k = \frac{c_k}{m}, \quad w_k = \frac{c_k}{m - c_k}.$$

During the intermediary steps of our SIS procedure via CP on a three-way zero-one table, there will be some columns for the L factor with trivial cases. In that case, we have to treat them as structural zeros in the kth slice for some $k \in \{1, \ldots, l\}$. In that case, we have to use the probabilities for the distribution in (3) as follows:

$$p_k = \frac{r_k \cdot c_k}{r_k \cdot c_k + (n - r_k - g_k^{r_0})(m - c_k - g_k^{c_0})}, \tag{6}$$

where $g_k^{r_0}$ is the number of structural zeros in the (r_0, k)th column and $g_k^{c_0}$ is the number of structural zeros in the (c_0, k)th column. Thus, we have weights:

$$w_k = \frac{r_k \cdot c_k}{(n - r_k - g_k^{r_0})(m - c_k - g_k^{c_0})}. \tag{7}$$

Theorem 4 *For the uniform distribution over all $m \times n \times l$ zero-one tables with structural zeros with given marginals $r_k = X_{i_0+k}$, $c_k = X_{+j_0k}$ for $k = 1, 2, \ldots, l$, and a fixed marginal for the factor L, l_0, the marginal distribution of the fixed marginal l_0 is the same as the conditional distribution of Z defined by* (3) *given $S_Z = l_0$ with*

$$p_k = \frac{r_k \cdot c_k}{r_k \cdot c_k + (n - r_k - g_k^{r_0})(m - c_k - g_k^{c_0})},$$

where g_k^{r0} is the number of structural zeros in the (r_0, k)th column and g_k^{c0} is the number of structural zeros in the (c_0, k)th column.

Proof The proof is similar to the proof for Theorem 2; we replace the probability p_k with

$$p_k = \frac{\binom{n-1-g_k^{r0}}{r_k-1}\binom{m-1-g_k^{c0}}{c_k-1}}{\binom{n-1-g_k^{r0}}{r_k-1}\binom{m-1-g_k^{c0}}{c_k-1} + \binom{n-1-g_k^{r0}}{r_k}\binom{m-1-g_k^{c0}}{c_k}}$$
$$= \frac{r_k \cdot c_k}{r_k \cdot c_k + (n - r_k - g_k^{r0})(m - c_k - g_k^{c0})}.$$

□

Remark 5 The sequential importance sampling via CP for sampling a two-way zero-one table with structural zeros defined in Theorem 1 in (Chen 2007) is a special case of our SIS. We can induce p_k defined in (6) and the weights defined in (7) to the weights for two-way zero-one contingency tables defined in (Chen 2007). Note that when we consider two-way zero-one contingency tables, we have $c_k = 1$ for all $k = 1, \ldots, l$ and for all $j_0 = 1, \ldots, n$ (or $r_k = 1$ for all $k = 1, \ldots, l$ and for all $i_0 = 1, \ldots, m$), $m = 2$ (or $n = 2$, respectively), and $g_k^{c0} = 0$ (or g_k^{r0}, respectively). Therefore, when we consider the two-way zero-one tables we get

$$p_k = \frac{r_k}{n - g_k^{r0}}, \; w_k = \frac{r_k}{n - r_k - g_k^{r0}},$$

or respectively

$$p_k = \frac{c_k}{m - g_k^{c0}}, \; w_k = \frac{c_k}{m - c_k - g_k^{c0}}.$$

Algorithm 6 (*Store structures in the zero-one table*) This algorithm stores the structures, including zeros and ones, in the observed table $\mathbf{x}_0$. The output will be used to avoid trivial cases in sampling. The output A and B matrices both have the same dimension with $\mathbf{x}_0$, so the cell value in A will be 1 if the position is structured and 0 if not. The matrix B is only for structure 1s. We consider sampling a table without structure 1s, that is, a table with new marginals: $X_{ij+}^* = X_{ij+} - \sum_{k=1}^{l} B_{ijk} = X_{ij+} - B_{ij+}$, $X_{i+k}^* = X_{i+k} - \sum_{j=1}^{n} B_{ijk} = X_{i+k} - B_{i+k}$, and $X_{+jk}^* = X_{+jk} - \sum_{i=1}^{m} B_{ijk} = X_{+jk} - B_{+jk}$ for $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$, and $k = 1, 2, \ldots, l$.

```
Input The observed marginals $X_{ij+}$, $X_{i+k}$, and $X_{+jk}$ for $i = 1, 2, \ldots, m$,
    $j = 1, 2, \ldots, n$, and $k = 1, 2, \ldots, l$.
Output Matrix $A$ and $B$, new marginals $X_{ij+}^*$, $X_{i+k}^*$, and $X_{+jk}^*$ for
    $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$, and $k = 1, 2, \ldots, l$.
Algorithm
  1. Check all marginals in direction $I$. For $i = 1, 2, \ldots, m$:
     If $X_{+jk} = 0$, $A_{i'jk} = 1$, for all $i' = 1, 2, \ldots, m$ and $A_{i'jk} = 0$;
     If $X_{+jk} = 1$, $A_{i'jk} = 1$ and $B_{i'jk} = 1$, for all $i' = 1, 2, \ldots, m$ and $A_{i'jk} = 0$.
```

2. Check all marginals in direction J. For $j = 1, 2, \dots, n$:
   If $X_{i+k} = 0,\ A_{ij'k} = 1$, for all $j' = 1, 2, \dots, n$ and $A_{ij'k} = 0$;
   If $X_{i+k} = 1,\ A_{ij'k} = 1$ and $B_{ij'k} = 1$, for all $j' = 1, 2, \dots, n$ and $A_{ij'k} = 0$.
3. Check all marginals in direction K. For $k = 1, 2, \dots, l$:
   If $X_{ij+} = 0,\ A_{ijk'} = 1$, for all $k' = 1, 2, \dots, l$ and $A_{ijk'} = 0$;
   If $X_{ij+} = 1,\ A_{ijk'} = 1$ and $B_{ijk'} = 1$, for all $k' = 1, 2, \dots, l$ and $A_{ijk'} = 0$.
4. If any changes made in step (1), (2) or (3), then come back to (1), else stop.
5. Compute new marginals:
   $X^*_{ij+} = X_{ij+} - B_{ij+},\ X^*_{i+k} = X_{i+k} - B_{i+k}$, and $X^*_{+jk} = X_{+jk} - B_{+jk}$ for $i = 1, 2, \dots, m,\ j = 1, 2, \dots, n$, and $k = 1, 2, \dots, l$.

Algorithm 7 (*Generate a two-way table with the given marginals*) This algorithm is used to generate a layer (fixed i) of the three-way table, with the probability of the sampled layer.

Input Row sums r^*_j and column sums $c^*_k,\ j = 1, 2, \dots, n$, and $k = 1, 2, \dots, l$; structures A; marginals on direction I: X_{+jk} for $i = 1, 2, \dots, m$.
Output A sampled table and its probability. Return 0 if the process fails.
Algorithm

1. Order all columns with decreasing sums.
2. Generate the column (along the direction K) with the largest sum, and the weights used in CP are shown in Eq. (7). Notice that k relates to each specific cell in the column, r_k and c_k which are the row sums in the direction J and I, respectively. g^{r0}_k and g^{c0}_k are the number of structures in the rows of the direction J and I, respectively. The probability of the generated column will be returned if the process succeeds, while 0 may be returned in this step if it does not exist.
3. Delete the generated column in (2), and for the remaining subtable, do the following:
   (a) If only one column is left, fill it with fixed marginals and go to (4).
   (b) If (a) is not true, check all marginals to see if there are any new structures caused by step (2). We need to avoid trivial cases by doing this. Go back to (1) with new marginals and structures.
4. Return generated matrix as the new layer and its CP probability. If failed, return 0.

Algorithm 8 (*SIS with CP for sampling a three-way zero-one table*) We describe an algorithm to sample a three-way zero-one table $\mathbf{X}$ with given marginals $X_{ij+},\ X_{i+k}$, and X_{+jk} for $i = 1, 2, \dots, m,\ j = 1, 2, \dots, n$, and $k = 1, 2, \dots, l$ via SIS with CP.

Input The observed table $\mathbf{x}_0$.
Output The sampled table $\mathbf{x}$.
Algorithm

1. Compute the marginals $X_{ij+},\ X_{i+k}$, and X_{+jk} for $i = 1, 2, \dots, m$, $j = 1, 2, \dots, n$, and $k = 1, 2, \dots, l$.
2. Use Algorithm 6 to compute the structure tables A and B. Consider the new marginals in the output as the sampling marginals.

3. For the sampling marginals, do SIS:
   (a) Delete the layers filled by structures; consider the leftover subtable.
   (b) Consider the layers in direction I (i varies). Sum within all layers and order them from the largest to smallest.
   (c) Consider the layer with the largest sum and plug in the structure table A from Algorithm 7 to generate a sample for this layer. The algorithm may return 0 if the sampling fails.
   (d) Delete the generated layer in (c), and for the remaining subtable, do the following:
      (i) If only one layer is left, fill it with fixed marginals and go to (e).
      (ii) else, go back to (2) with new marginals.
   (e) Add the sampled table with table B (the structure 1s table).
4. Return the table in (e) and the same probability with the sampled table. Return 0 if failed.

4 Four or higher dimensional zero-one tables

In this section, we consider a d-way zero-one table under the no d-way interaction model for $d \in \mathbb{N}$ and $d > 3$. Let $\mathbf{X} = (X_{i_1 \dots i_d})$ be a zero-one contingency table of size $(n_1 \times \cdots \times n_d)$, where $n_i \in \mathbb{N}$ for $i = 1, \dots, d$. The sufficient statistics under the no d-way interaction model are

$$\begin{aligned} &X_{+i_2 \dots i_d},\ X_{i_1 + i_3 \dots i_d},\ \dots,\ X_{i_1 \dots i_{d-1}+}, \\ &\text{for}\, i_1 = 1, \dots, n_1,\ i_2 = 1, \dots, n_2, \dots, i_d = 1, \dots, n_d. \end{aligned} \tag{8}$$

For each $i_1^0 \in \{1, \dots, n_1\}, \dots, i_{d-1}^0 \in \{1, \dots, n_d\}$, we say the column of the entries for a marginal $X_{i_1 \dots i_{j-1} + i_{j+1} \dots i_d}$ the $(i_0, \dots, i_{j-1}, i_{j+1}, \dots, i_d)$th column of $\mathbf{X}$. For each $i_1^0 \in \{1, \dots, n_1\}, \dots, i_{d-1}^0 \in \{1, \dots, n_{d-1}\}$, we consider the $(i_1^0, \dots, i_{d-1}^0)$th column for the dth factor. Let $l_0 = X_{i_1^0, \dots, i_{d-1}^0 +}$. Let $r_k^j = X_{i_1^0 \dots i_{j-1}^0 + i_{j+1}^0 \dots i_{d-1}^0 k}$ for fixed $k \in \{1, \dots, n_d\}$. For sampling a zero-one d-way table $\mathbf{X}$, we set

$$p_k = \frac{\prod_{j=1}^{d-1} r_k^j}{\prod_{j=1}^{d-1} r_k^j + \prod_{j=1}^{d-1} (n_j - r_k^j)}. \tag{9}$$

Remark 9 We assume that we do not have trivial cases, namely, $1 \le r_k^j \le n_j - 1$ for $j = 1, \dots, d$.

Theorem 10 *For the uniform distribution over all d-way zero-one contingency tables $\mathbf{X} = (X_{i_1, \dots, i_d})$ of size $(n_1 \times \cdots \times n_d)$, where $n_i \in \mathbb{N}$ for $i = 1, \dots, d$ with marginals $l_0 = X_{i_1^0, \dots, i_{d-1}^0 +}$, and $r_k^j = X_{i_1^0 \dots i_{j-1}^0 + i_{j+1}^0 \dots i_{d-1}^0 k}$ for $k \in \{1, \dots, n_d\}$, the marginal distribution of the fixed marginal l_0 is the same as the conditional distribution of Z defined by (3) given $S_Z = l_0$ with*

$$p_k = \frac{\prod_{j=1}^{d-1} r_k^j}{\prod_{j=1}^{d-1} r_k^j + \prod_{j=1}^{d-1}(n_j - r_k^j)}.$$

Proof The proof is similar to the proof for Theorem 2; we extend the same argument to a d-way zero-one table under the no d-way interaction model with the probability

$$p_k = \frac{\prod_{j=1}^{d-1} \binom{n_j-1}{r_k^j-1}}{\prod_{j=1}^{d-1} \binom{n_j-1}{r_k^j-1} + \prod_{j=1}^{d-1} \binom{n_j-1}{r_k^j}} = \frac{\prod_{j=1}^{d-1} r_k^j}{\prod_{j=1}^{d-1} r_k^j + \prod_{j=1}^{d-1}(n_j - r_k^j)}.$$

□

During the intermediary steps of our SIS procedure via CP on a three-way zero-one table, there will be some columns for the dth factor with trivial cases. In that case, we have to treat them as structural zeros in the kth slice for some $k \in \{1, \dots, l\}$. In that case, we have to use the probabilities for the distribution in (3) as follows:

$$p_k = \frac{\prod_{j=1}^{d-1} r_k^j}{\prod_{j=1}^{d-1} r_k^j + \prod_{j=1}^{d-1}(n_j - r_k^j - g_k^j)}. \tag{10}$$

where g_k^j is the number of structural zeros in the $(i_1^0, \dots, i_{j-1}^0, i_{j+1}^0, \dots, i_{d-1}^0 k)$th column of $\mathbf{X}$. Thus, we have weights:

$$w_k = \frac{\prod_{j=1}^{d-1} r_k^j}{\prod_{j=1}^{d-1}(n_j - r_k^j - g_k^j)}. \tag{11}$$

Theorem 11 *For the uniform distribution over all d-way zero-one contingency tables $\mathbf{X} = (X_{i_1,\dots,i_d})$ of size $(n_1 \times \cdots \times n_d)$, where $n_i \in \mathbb{N}$ for $i = 1, \dots, d$ with marginals $l_0 = X_{i_1^0,\dots,i_{d-1}^0+}$, and $r_k^j = X_{i_1^0 \dots i_{j-1}^0 + i_{j+1}^0 \dots i_{d-1}^0 k}$ for $k \in \{1, \dots, n_d\}$, the marginal distribution of the fixed marginal l_0 is the same as the conditional distribution of Z defined by (3) given $S_Z = l_0$ with*

$$p_k = \frac{\prod_{j=1}^{d-1} r_k^j}{\prod_{j=1}^{d-1} r_k^j + \prod_{j=1}^{d-1}(n_j - r_k^j - g_k^j)}$$

where g_k^j is the number of structural zeros in the $(i_1^0, \dots, i_{j-1}^0, i_{j+1}^0, \dots, i_{d-1}^0 k)$th column of $\mathbf{X}$.

Proof The proof is similar to the proof for Theorem 4; we extend the same argument to a d-way zero-one table under the no d-way interaction model with the probability

$$p_k = \frac{\prod_{j=1}^{d-1}\binom{n_j-1-g_k^j}{r_k^j-1}}{\prod_{j=1}^{d-1}\binom{n_j-1-g_k^j}{r_k^j-1} + \prod_{j=1}^{d-1}\binom{n_j-1-g_k^j}{r_k^j}} = \frac{\prod_{j=1}^{d-1} r_k^j}{\prod_{j=1}^{d-1} r_k^j + \prod_{j=1}^{d-1}(n_j - r_k^j - g_k^j)}.$$

□

5 Computational examples

For our simulation study, we used the software package R (R-Project-Team 2011). We count the *exact* numbers of tables via the software LattE (De Loera et al. 2005) for small examples in this section. When the contingency tables are large and/or the models are complicated, it is very difficult to obtain the exact number of tables. Thus, we need a good measurement of accuracy in the estimated number of tables. In Chen et al. (2005), the coefficient of variation (cv^2) was used:

$$cv^2 = \frac{\text{var}_q\{p(\mathbf{X})/q(\mathbf{X})\}}{\mathbb{E}_q^2\{p(\mathbf{X})/q(\mathbf{X})\}}$$

which is equal to $\text{var}_q\{1/q(\mathbf{X})\}/\mathbb{E}_q^2\{1/q(\mathbf{X})\}$ for the problem of estimating the number of tables. The value of cv^2 is simply the chi-square distance between the two distributions p' and q, which means the smaller it is, the closer the two distributions are. In Chen et al. (2005), cv^2 was estimated by:

$$cv^2 \approx \frac{\sum_{i=1}^{N}\{1/q(\mathbf{X_i}) - \left[\sum_{j=1}^{N} 1/q(\mathbf{X_j})\right]/N\}^2/(N-1)}{\left\{\left[\sum_{j=1}^{N} 1/q(\mathbf{X_j})\right]/N\right\}^2},$$

where $\mathbf{X_1}, \ldots, \mathbf{X_N}$ are tables drawn iid from $q(\mathbf{X})$. When we have rejections, we compute the variance using only accepted tables. In this paper, we also investigated relations with the exact numbers of tables and cv^2 when we have rejections.

In this section, we define the three two-way marginal matrices as follows: Suppose we have an observed table $\mathbf{x} = (x_{ijk})_{m\times n\times l}$, $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$, and $k = 1, 2, \ldots, l$;

Define: $si = (X_{+jk})_{n\times l}$, $sj = (X_{i+k})_{m\times l}$, and $sk = (X_{ij+})_{m\times n}$.

Example 12 (The three-dimensional semimagic cube) Suppose si, sj, and sk are all 3×3 matrices with all 1s inside, that is:

$$si = sj = sk = \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

The real number of tables is 12. We took 114.7 s to run 10,000 samples in the SIS procedure; the estimator was 12 and acceptance rate was 100 %. Actually, we found that if the acceptance rate was 100 %, then sample size did not matter in the estimation.

We used R to produce more examples. Examples below are constructed with the same code, but with different values for parameters. We used the R package "Rlab" for the following code.

```
seed=6; m=3; n=3; l=4; prob=0.8; N=1000; k=200
set.seed(seed)
A=array(rbern(m*n*l,prob),c(m,n,l))
outinfo=tabinfo(A)
numtable(N,outinfo,k)
```

Here, prob is the probability of getting 1 for every Bernoulli variable, and N is the sample size (the total number of tables sampled, including both acceptances and rejections). Notice that cv^2 is defined as $\frac{\text{Var}}{\text{Mean}^2}$.

Example 13 (seed = 6; $m = 3$; $n = 3$; $l = 4$; prob = 0.8) Suppose si, sj, and sk are as following, respectively:

$$\begin{array}{|c|c|c|c|}\hline 2&2&2&2\\\hline 1&3&2&2\\\hline 2&3&3&2\\\hline\end{array}, \quad \begin{array}{|c|c|c|c|}\hline 2&3&2&2\\\hline 1&3&3&3\\\hline 2&2&2&1\\\hline\end{array}, \quad \begin{array}{|c|c|c|}\hline 3&3&3\\\hline 3&3&4\\\hline 2&2&3\\\hline\end{array}.$$

The real number of tables is 3. The estimator was 3.00762 with $cv^2 = 0.0708$. The whole process took 13.216 s (in R) with a 100 % acceptance rate.

Example 14 (seed = 60; $m = 3$; $n = 4$; $l = 4$; prob = 0.5) Suppose si, sj, and sk are as follows, respectively:

$$\begin{array}{|c|c|c|c|}\hline 2&2&2&1\\\hline 1&1&1&0\\\hline 1&1&1&2\\\hline 1&1&2&3\\\hline\end{array}, \quad \begin{array}{|c|c|c|c|}\hline 3&3&2&1\\\hline 1&0&2&2\\\hline 1&2&2&3\\\hline\end{array}, \quad \begin{array}{|c|c|c|c|}\hline 3&2&2&2\\\hline 1&0&2&2\\\hline 3&1&1&3\\\hline\end{array}.$$

The real number of tables is 5. The estimator was 4.991026 with $cv^2 = 0.1335$. The whole process took 17.016 s (in R) with a 100 % acceptance rate.

Example 15 (seed = 240; $m = 4$; $n = 4$; $l = 4$; prob = 0.5) Suppose si, sj, and sk are as follows, respectively:

$$\begin{array}{|c|c|c|c|}\hline 2&3&3&2\\\hline 1&3&2&1\\\hline 1&2&3&0\\\hline 4&2&2&2\\\hline\end{array}, \quad \begin{array}{|c|c|c|c|}\hline 2&2&4&1\\\hline 3&2&2&2\\\hline 2&3&3&1\\\hline 1&3&1&1\\\hline\end{array}, \quad \begin{array}{|c|c|c|c|}\hline 2&2&3&2\\\hline 3&2&1&3\\\hline 3&2&2&2\\\hline 2&1&0&3\\\hline\end{array}.$$

The real number of tables is 8. The estimator was 8.039938 with $cv^2 = 0.2857$. The whole process took 23.612 s (in R) with a 100 % acceptance rate.

Example 16 (seed = 5,440; $m = 4$; $n = 4$; $l = 4$; prob = 0.5) Suppose si, sj, and sk are as follows, respectively:

Table 1 Summary of computational results on $m \times n \times l$ tables for $m = n = l = 4, \ldots, 10$. All marginal sums are equal to one in this example. We counted the exact number for $m \times m \times m$ semimagic cube with marginal sums equal to one using the number of all Latin squares of size m

Dimension m	# Tables	N	CPU time (s)	Estimation	cv^2	Acceptance rate (%)
4	576	1,000	32.44	568.944	0.26	100
		10,000	324.18	571.1472	0.27	100
5	161,280	1,000	60.39	161,603.5	0.18	99
		10,000	605.45	161,439.3	0.18	99.2
6	812,851,200	1,000	102.66	801,634,023	0.58	98.3
		10,000	1,038.46	819,177,227	0.45	98.8
7	6.14794e+13	1,000	158.55	6.08928e+13	0.60	97
		10,000	1,590.84	6.146227e+13	0.64	97.7
8	1.08776e+20	1,000	234.53	1.080208e+20	1.07	95.6
		10,000	2,300.91	1.099627e+20	1.00	96.5
9	5.52475e+27	1,000	329.17	5.845308e+27	1.46	94
		10,000	3,238.1	5.684428e+27	1.59	95.3
10	9.98244e+36	1,000	451.24	9.648942e+36	1.44	93.3
		10,000	4,425.12	9.73486e+36	1.73	93.3

$$\begin{array}{|c|c|c|c|}\hline 2&1&0&1\\\hline 2&3&1&2\\\hline 3&1&2&1\\\hline 1&3&2&2\\\hline\end{array}, \quad \begin{array}{|c|c|c|c|}\hline 2&3&2&1\\\hline 2&1&2&3\\\hline 2&1&0&1\\\hline 2&3&1&1\\\hline\end{array}, \quad \begin{array}{|c|c|c|c|}\hline 1&2&2&3\\\hline 1&1&3&3\\\hline 1&3&0&0\\\hline 1&2&2&2\\\hline\end{array}.$$

The real number of tables is 9. The estimator was 8.882672 with $cv^2 = 0.7701368$. The whole process took 30.171 s (in R) with a 100 % acceptance rate. Another result for the same sample size is: an estimator is 8.521734, $cv^2 = 0.6695902$. You can find that the latter has a slightly better cv^2, but a slightly worse estimator. We will discuss more in Sect. 7.

Example 17 (seed = 222; $m = 4$; $n = 4$; $l = 5$; prob = 0.2) Suppose si, sj, and sk are as follows, respectively:

$$\begin{array}{|c|c|c|c|c|}\hline 1&0&1&1&1\\\hline 2&1&0&1&2\\\hline 0&1&1&1&0\\\hline 1&1&1&1&1\\\hline\end{array}, \quad \begin{array}{|c|c|c|c|c|}\hline 2&1&0&0&2\\\hline 1&2&1&2&1\\\hline 1&0&1&1&1\\\hline 0&0&1&1&0\\\hline\end{array}, \quad \begin{array}{|c|c|c|c|}\hline 2&3&0&0\\\hline 1&3&2&1\\\hline 0&0&1&3\\\hline 1&0&0&1\\\hline\end{array}.$$

The real number of tables is 2. The estimator was 2 with $cv^2 = 0$. The whole process took 19.064 s (in R) with a 100 % acceptance rate.

Example 18 (High-dimension semimagic cubes) In this example, we consider $m \times n \times l$ tables for $m = n = l = 4, \ldots, 10$ such that each marginal sum equals to 1. The results are summarized in Table 1.

Table 2 Summary of computational results on $m \times n \times l$ tables for $m = n = l = 4, \ldots, 10$. All marginal sums are equal to s in this example. The sample $N = 1,000$ in this example

Dimension m	s	CPU time (s)	Estimation	cv^2	Acceptance rate (%)
4	2	27.1	51,810.36	0.66	97.7
5	2	58.1	25,196,288,574	1.69	97.5
6	2	97.1	6.339628e+18	2.56	94.8
	3	99.3	1.269398e+22	2.83	96.5
7	2	150.85	1.437412e+30	4.76	93.1
	3	166.68	2.365389e+38	25.33	96.7
8	2	229.85	5.369437e+44	6.68	89.8
	3	256.70	3.236556e+59	7.05	94.5
	4	328.52	2.448923e+64	11.98	94.3
9	2	319.32	4.416787e+62	8.93	85.7
	3	376.67	7.871387e+85	15.23	91.6
	4	549.73	2.422237e+97	14.00	93.4
10	2	429.19	2.166449e+84	10.46	83.3
	3	527.14	6.861123e+117	26.62	90
	4	883.34	3.652694e+137	33.33	93.8
	5	1439.50	1.315069e+144	46.2	91.3

Example 19 (High-dimension semimagic cubes continues) In this example, we consider $m \times n \times l$ tables for $m = n = l = 4, \ldots, 10$ such that each marginal sum equals to s. The results are summarized in Table 2. In this example, we set the sample size as $N = 1,000$.

Example 20 (Bootstrap t confidence interval of semimagic cubes) As we can see in Table 2, generally cv^2 is larger when the number of tables is larger, and in this case, the estimator we get via the SIS procedure might vary greatly in different iterations. Therefore, we propose computing a $(1-\alpha)100$ % confidence interval for each estimator via a nonparametric bootstrap method (see Appendix 8) for a pseudo code for a nonparametric bootstrap method to get the $(1-\alpha)100$ % confidence interval for $|\Sigma|$). See Table 3 for some results of bootstrap t 4 95 % confidence intervals ($\alpha = 0.05$).

6 Experiment with Sampson's data set

Sampson recorded the social interactions among a group of monks while he visited as an experimenter on vision. He collected numerous sociometric rankings (Breiger et al. 1975; Sampson 1969). The data are organized as a $18 \times 18 \times 10$ table and one can find the full data sets at http://vlado.fmf.uni-lj.si/pub/networks/data/ucinet/UciData.htm#sampson. Each layer of 18×18 table represents a social relation between 18 monks at some time point. Most of the present data are retrospective, collected after the breakup occurred. They concern a period during which a new cohort entered the monastery near

Table 3 Summary of confidence intervals

Dim	s	Estimation			cv^2			Acceptance rate (%)
		$\widehat{\lvert\Sigma\rvert}$	Lower 95 %	Upper 95 %	$\widehat{cv^2}$	Lower 95 %	Upper 95 %	
7	2	1.306480e+30	1.156686e+30	1.468754e+30	3.442306	2.678507	4.199513	93.3
	3	3.033551e+38	2.245910e+38	4.087225e+38	22.84399	8.651207	35.080408	96.2
8	2	5.010225e+44	4.200752e+44	5.902405e+44	6.712335	4.539368	8.590578	90.4
	3	2.902294e+59	2.389625e+59	3.484405e+59	9.047914	5.680128	12.797488	93.1
	4	2.474874e+64	1.847911e+64	3.295986e+64	21.53559	5.384647	32.166086	94.6
9	2	4.548401e+62	3.682882e+62	5.593370e+62	10.07973	4.886817	15.406899	87.1
	3	9.702672e+85	7.189849e+85	1.250875e+86	18.65302	11.33462	23.77980	92.5
	4	2.023034e+97	1.547951e+97	2.561084e+97	14.96126	10.20331	19.09515	92.2
10	2	2.570344e+84	1.908609e+84	3.339243e+84	17.83684	9.785778	24.231544	84.8
	3	8.68783e+117	5.92233e+117	1.22271e+118	29.67200	18.64549	37.64892	90.2
	4	4.12634e+137	2.94789e+137	5.52727e+137	23.36831	15.32719	31.02614	92
	5	1.54956e+144	9.85557e+143	2.24043e+144	39.06521	20.23674	53.60838	91.8

Dimensions and marginals = s are defined same with Table 2

$\widehat{\lvert\Sigma\rvert}$ means an estimator of $\lvert\Sigma\rvert$ and $\widehat{cv^2}$ is an estimator of cv^2. The sample size for the SIS procedure is $N = 1{,}000$ and the sample size for bootstraping is $B = 5{,}000$. Only cases with relatively large cv^2 are involved the end of the study but before the major conflict began. The exceptions are "liking" data gathered at three times: SAMPLK1 to SAMPLK3 that reflect changes in group sentiment over time (SAMPLK3 was collected in the same way as the data described below). In the data set, four relations are coded, with separate matrices for positive and negative ties on the ten relations: esteem (SAMPES) and disesteem (SAMPDES); liking (SAMPLK which are SAMPLK1 to SAMPLK3) and disliking (SAMPDLK); positive influence (SAMPIN) and negative influence (SAMPNIN); praise (SAMPPR) and blame (SAMPNPR). In the original data set, they listed the top three choices and recorded as ranks. However, we set these ranks as an indicator (i.e., if they are in the top three choices, then we set one, or else zero).

We ran the SIS procedure with $N = 100,000$ and a bootstrap sample size $B = 50,000$. The estimator was 1.704774e+117 with its 95 % confidence interval, [1.119321e+117 2.681264e+119] and $cv^2 = 621.4$ with its 95 % confidence interval, [324.29, 2,959.65]. The CPU time was 70,442 s. The acceptance rate was 3 %.

7 Discussion

In this paper, we do not have a sufficient and necessary condition for the existence of the three-way zero-one table, so we cannot avoid rejection. However, since the SIS procedure gives an unbiased estimator, we may only need a small sample size as long as it converges. Also, note that the acceptance rate does not depend on the sample size. Thus, it would be interesting to investigate the convergence rate of the SIS procedure with CP for zero-one three-way tables.

It seems that the convergence rate is slower when we have a "large" table (here "large" means in terms of $|\Sigma|$ rather than its dimension, i.e., the number of cells). A large estimator $\widehat{|\Sigma|}$ usually corresponds to a larger cv^2, and this often comes with large variations of $\widehat{|\Sigma|}$ and cv^2. This means that if we have a large $|\Sigma|$, more likely we get extremely larger $\widehat{|\Sigma|}$ and cv^2 and different iterations can give very different results. For example, we ran three iterations for the $8 \times 8 \times 8$ semimagic cube with all marginals equal to 3 and we got the following results: estimator =3.236556e+59 with $cv^2 = 7.049114$; estimator =2.902294e+59 with $cv^2 = 9.047914$; and estimator =3.880133e+59 with $cv^2 = 55.59179$. Fortunately, though we have a large $|\Sigma|$, our acceptance rate is still high and a computational time seems to still be attractive. Thus, when one finds a large estimation or a large cv^2, we recommend applying several iterations and picking the result with the smallest cv^2. We should always compare cv^2 in a large scale. However, a small improvement does not necessarily mean a better estimator (see Example 16).

For calculating the bootstrap t confidence intervals, we often have a larger confidence interval when we have a larger cv^2, and this confidence interval might be less informative and less reliable. Therefore, we suggest using the result with the smallest cv^2 for bootstraping procedure. In Table 3, we showed only confidence intervals for semimagic cubes with $m = n = l = 7, \ldots, 10$ in Example 20 because of the following reason: when cv^2 is very small, computing bootstrap t confidence interval does not make much sense, since the estimation has already converged.

For the experiment with Sampson's data set, we observed a very low acceptance rate compared with experimental studies on simulated data sets. We investigated why this happened and how to increase the acceptance rates. By simulations, we found two possible reasons: first, it seems that our sampling works better when the values of marginals are balanced (for example, consider a case that every cell in the table has a similar or the same probability to be 1 and one of extreme cases is a semimagic cube where all marginals are the same); second, a higher dimension may be unfavorable for acceptance rate. Simulations show that the acceptance rates can be very small when we have both cases: a simulation of a $10 \times 10 \times 10$ table with unbalanced marginals has only 40 % acceptance rate and decreases to only 1 % for a $18 \times 18 \times 10$ table. On the other hand, the large cv^2 also causes problem. It seems that we have a large cv^2 for Sampson's data set, because there are very few sampled tables which have very small probabilities. These "outliers" can make our result very unstable (see Table 4 in Appendix for results with and without seven "outliers"). A problem is that the values we took off are not theoretical "outliers"; hence, whether it is reasonable to delete them and which one is more reliable become issues. This is one of the open problems we need to deal with.

In Chen et al. 2005, the Gale–Ryser Theorem was used to obtain an SIS procedure without rejection for two-way zero-one tables. However, for three-way table cases, it seems very difficult because we naturally have structural zeros and trivial cases on a process of sampling one table. In (Chen 2007) Chen showed a version of Gale–Ryser Theorem for structural zero for two-way zero-one tables, but it assumes that there is at most one structural zero in each row and column. In general, there are usually more than one in each row and column.

In this paper, the target distribution is the uniform distribution. We sample a table from the set of all zero-one tables satisfying the given marginals as close and uniformly via the SIS procedure with CP.

8 Appendix: Nonparametric bootstrap method

In this section, we explain how to use a nonparametric bootstrap method to get the $(1-\alpha)100\,\%$ confidence interval for $|\Sigma|$. Notice that the bootstrap sample size is fixed as B, and notations here are consistent with Sect. 2.

(1) Drawing pseudo data set

Concept In an SIS procedure with sample size N, we get a sequence of random tables $\mathbf{X_1}, \ldots, \mathbf{X_N}$. Define $\mathbf{Y_i} = \frac{\mathbb{I}_{\mathbf{X_i} \in \Sigma}}{q(\mathbf{X_i})}$, $i = 1, \ldots, N$ where $q(\mathbf{X})$ is the trial distribution, then $\mathbf{Y_1}, \ldots, \mathbf{Y_N}$ is a sequence of iid random variables. This means that it makes sense to consider the empirical distribution of $\mathbf{Y_i}$, which is nonparametric maximum likelihood estimator of the real distribution of $\mathbf{Y_i}$ (actually, as $\mathbf{Y_i}$ can only take finitely many values, the empirical distribution becomes the maximum likelihood estimator of the real distribution). Draw a pseudo sample $\mathbf{Y_1^*}, \ldots, \mathbf{Y_N^*}$ from the empirical distribution.

Algorithm Use the SIS procedure to get $\mathbf{Y_i} = \frac{\mathbb{I}_{\mathbf{X_i} \in \Sigma}}{q(\mathbf{X_i})}$, $i = 1, \ldots, N$, which should be just a sequence of numbers. Draw N elements from this sequence with replacement.

Table 4 Compare the Sampson results of with/without seven “outliers”

“Outliers”	Estimation			cv^2			Acceptance rate (%)
	$\widehat{\lvert\Sigma\rvert}$	Lower 95 %	Upper 95 %	$\widehat{cv^2}$	Lower 95 %	Upper 95 %	
With	1.313089e+117	4.771677e+116	2.391368e+117	392.6767	230.4170	711.2878	2.803
Without	1.932762e+116	9.973317e+115	3.154941e+116	226.8825	124.2770	336.6534	2.796

The sample size for SIS procedure is $N = 100,000$ and the sample size for bootstraping is $B = 50,000$. The cutoff of “outlier” is 4e+120, that is, values which are greater than 4e+120 are deleted. The seven “outliers” are: 2.83e+121, 1.93e+121, 3.00e+121, 1.07e+121, 9.08e+120, 4.66e+120, 1.00e+121

(2) One Bootstrap replication

Concept Consider the pseudo sample $\mathbf{Y}_\mathbf{1}^*, \ldots, \mathbf{Y}_\mathbf{N}^*$ as a "new" sample from the empirical distribution, then the cumulative distribution function (CDF) of $\widehat{\theta}^* = T(\mathbf{Y}_\mathbf{1}^*, \ldots, \mathbf{Y}_\mathbf{N}^*)$ is a consistent estimator of the CDF of $\widehat{\theta} = T(\mathbf{Y}_\mathbf{1}, \ldots, \mathbf{Y}_\mathbf{N})$. Here, we can consider our estimator of $|\Sigma|$:

$$\widehat{|\Sigma|} = \widehat{\theta_1} = T_1(\mathbf{Y}_\mathbf{1}, \ldots, \mathbf{Y}_\mathbf{N}) = \frac{1}{N}\sum_{i=1}^{N}\mathbf{Y}_\mathbf{i}$$

The cv^2:

$$\widehat{cv^2} = \widehat{\theta_2} = T_2(\mathbf{Y}_\mathbf{1}, \ldots, \mathbf{Y}_\mathbf{N}) = \frac{\sum_{i=1}^{N}\left\{\mathbf{Y}_\mathbf{i} - \left[\sum_{j=1}^{N}\mathbf{Y}_\mathbf{j}\right]/N\right\}^2/(N-1)}{\left\{\left[\sum_{j=1}^{N}\mathbf{Y}_\mathbf{j}\right]/N\right\}^2}$$

Algorithm Treat the pseudo sample as a sample from the SIS and compute the statistics based on it. That means, this bootstrap replication can be obtained by:

$$\widehat{|\Sigma|}^{*1} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{Y}_\mathbf{i}^*; \quad \widehat{cv^2}_{*1} = cv^2 \text{ of } (\mathbf{Y}_\mathbf{1}^*, \ldots, \mathbf{Y}_\mathbf{N}^*)$$

(3) Bootstrap t confidence interval

Concept Repeat the previous two steps until we get B Bootstrap replications: $\widehat{\theta_i}^{*1}, \ldots, \widehat{\theta_i}^{*B},\ i = 1, 2$. The empirical distribution of $\widehat{\theta_i}^*$ is the nonparametric maximum likelihood estimator of CDF of $\widehat{\theta_i}^*$, and the latter is a consistent estimator of the CDF of $\widehat{\theta_i}$. So, we can use $(\frac{\alpha}{2})100_{th}$ and $(1 - \frac{\alpha}{2})100_{th}$ percentiles of the empirical distribution as our confidence interval.

Algorithm Repeat the previous two steps B times. For $\{\widehat{|\Sigma|}^{*1}, \ldots, \widehat{|\Sigma|}^{*B}\}$, define $\widehat{|\Sigma|}^*_{(a)}$ as the $100a_{th}$ percentile of the list of values. Then bootstrap-t $(1-\alpha)100\,\%$ confidence interval of $\widehat{|\Sigma|}$ is $[\widehat{|\Sigma|}^*_{(\alpha/2)}, \widehat{|\Sigma|}^*_{(1-\alpha/2)}]$. Similarly, we can get confidence interval for $\widehat{cv^2}$.

Acknowledgments The authors would like to thank Drs. Stephen Fienberg and Yuguo Chen for useful conversations.

References

Blitzstein, J., Diaconis, P. (2010). A sequential importance sampling algorithm for generating random graphs with prescribed degrees. *Internet Mathematics*, *6*(4), 489–522.

Breiger, R., Boorman, S., Arabie, P. (1975). An algorithm for clustering relational data with applications to social network analysis and comparison with multidimensional scaling. *Journal of Mathematical Psychology*, *12*, 328–383.

Chen, Y. (2007). Conditional inference on tables with structural zeros. *Journal of Computational and Graphical Statistics*, *16*(2), 445–467.

Chen, Y., Diaconis, P., Holmes, S., Liu, J. S. (2005). Sequential monte carlo methods for statistical analysis of tables. *Journal of the American Statistical Association*, *100*, 109–120.

Chen, Y., Dinwoodie, I., Sullivant, S. (2006). Sequential importance sampling for multiway tables. *The Annals of Statistics*, *34*(1), 523–545.

De Loera, J., Haws, D., Hemmecke, R., Huggins, P., Tauzer, J., Yoshida, R. (2005). LattE, version 1.2. http://www.math.ucdavis.edu/~latte/.

De Loera, J., Onn, S. (2006). All linear and integer programs are slim 3-way transportation programs. *SIAM Journal on Optimization*, *17*, 806–821.

Dinwoodie, I. H. (2008). Polynomials for classification trees and applications. Statistical and Applied Mathematical Sciences Institute Technical, Report 2008-7.

Dinwoodie, I. H., Chen, Y. (2011). Sampling large tables with constraints. *Statistica Sinica*, *21*, 1591–1609.

Garey, M. R., Johnson, D. S. (1979). *Computers and intractabihty, a guide to the theory of NP-completeness*. San Francisco: Freeman & Co.

Huber, M. (2006). Fast perfect sampling from linear extensions. *Discrete Mathematics*, *306*, 420–428.

R-Project-Team. (2011). R project. GNU software. http://www.r-project.org/.

Sampson, S. (1969). Crisis in a cloister. Doctoral dissertation (unpublished).

Snijders, T. A. B. (1991). Enumeration and simulation methods for $0 - 1$ matriceswith given marginals. *Psychometrika*, *56*, 397–417.