THREE EXAMPLES OF MONTE-CARLO MARKOV CHAINS: AT THE INTERFACE BETWEEN STATISTICAL COMPUTING, COMPUTER SCIENCE, AND STATISTICAL MECHANICS

PERSI DIACONIS* AND SUSAN HOLMES†

Abstract. The revival of interest in Markov chains is based in part on their recent applicability in solving real world problems and in part on their ability to resolve issues in theoretical computer science. This paper presents three examples which are used to illustrate both parts: a Markov chain algorithm for estimating the tails of the bootstrap also illustrates the Jerrum-Sinclair theory of approximate counting. The Geyer-Thompson work on Monte-Carlo evaluation of maximum likelihood is compared with work on evaluation of the partition function. Finally, work of Diaconis-Sturmfels on conditional inference is complemented by the work of theoretical computer scientists on approximate computation of the volume of convex polyhedra.

Introduction

This paper presents three examples of what has come to be called the Markov chain simulation method. The examples blend together ideas from statistics, computer science, and statistical mechanics. The problems presented are set in statistical contexts of assessing variability, maximizing likelihoods, and carrying out goodness of fit tests. All of the examples involve reversible Markov chains on discrete sample spaces. In each case, the chains were actually run for a problem of real world interest. Each example is paired with a healthy theoretical development. As always, there are tensions and trade offs between practice and theory. This area brings them closer than usual.

1. The bootstrap and approximate counting

A. The bootstrap

Efron's bootstrap is a fundamental advance in statistical practice. It allows accurate estimation of variability without parametric assumptions. One begins with data $x_1, x_2, \cdots, x_n$ in a space $\mathcal{X}$. Let $T(x_1, x_2, \cdots, x_n)$ be a statistic of interest (e.g., a mean, median, correlation matrix,$\cdots$). We are interested in estimating the variability of T, assuming that x_i are independent and identically chosen from an unknown distribution F on $\mathcal{X}$.

The bootstrap draws resample observations $x_1^*, x_2^*, \cdots, x_n^*$ from $\{x_1, x_2, \cdots, x_n\}$ with replacement and calculates $T(x_1^*, x_2^*, \cdots, x_n^*)$. Doing this repeatedly gives a set of values which can be proved to give a good indication of the distribution of T. A splendid up-to-date account of the bootstrap appears in Efron and Tibshirani (1993). Hall (1992) gives a solid

* Dept. of Mathematics, Harvard University, Cambridge, MA 02138.

† INRA, Unité de Biométrie, 2, Place Pierre Viala, 34060 Montpellier, France. (Visiting Stanford University).

theoretical development.

The present section explains a Monte Carlo method for deriving large deviations estimates of $P\{T \geq t\}$ when this probability is small. The idea is to run a Markov chain on the set of values $\{\underset{\sim}{x}^* = (x_1^*, \cdots, x_n^*)\colon T(\underset{\sim}{x}^*) \geq t\}$.

A step of the chain picks I, $1 \leq I \leq n$, uniformly at random and replaces x_I^* with a fresh value chosen uniformly from the original set of values $\{x_1, x_2, \cdots, x_n\}$. If the new sample vector satisfies $T(\underset{\sim}{x}) \geq t$, the change is made. Otherwise the chain stays at the previous sample vector.

This generates a reversible Markov chain on $\{\underset{\sim}{x} : T(\underset{\sim}{x}) \geq t\}$. It has a uniform stationary distribution. Assuming the chain is connected (see below) this gives an easy to implement method of sampling from the uniform distribution.

To estimate $P\{T \geq t\}$ we choose a grid $t_0 < t_1 < \cdots < t_\ell < t$ with t_0 chosen in the middle of the distribution of T and t_i chosen so that $P\{T \geq t_{i+1} | T \geq t_i\}$ is not too small. Here, all probabilities refer to the uniform distribution on the set of n-tuples chosen with repetition from $\{x_1, x_2, \cdots, x_n\}$.

With t_i chosen, first estimate $P\{T \geq t_0\}$ by ordinary Monte Carlo.

Then, estimate $P\{T \geq t_1 | T \geq t_0\}$ by running the Markov chain on $\{\underset{\sim}{x}^*\colon T(\underset{\sim}{x}^*) \geq t_0\}$

and counting what proportion of values satisfy the constraint $T \geq t_1$.

Continue, estimating $P\{T \geq t_2 | T \geq t_1\} \cdots P\{T \geq t | T \geq t_\ell\}$. Multiplying these estimates gives an estimate for $P\{T \geq t\}$:

$$\widehat{P}\{T \geq t\} = \widehat{P}\{T \geq t_0\}\widehat{P}\{T \geq t_1 | T \geq t_0\} \cdots \widehat{P}\{T \geq t | T \geq t_\ell\}.$$

Example 1.1. The following list of 10 pairs gives the average test score (LSAT) and grade point average (GPA) of 10 American law schools

LSAT	576	635	558	578	666	580	555	661	652	605
GPA	3.39	3.30	2.81	3.03	3.44	3.07	3.00	3.43	3.36	3.13

The scatter plot of these numbers in Figure 1.1 suggests a fair amount of association between LSAT and GPA. The correlation coefficient is $T = .81$. The bootstrap can be used to set confidence intervals for the true population correlation coefficient as suggested by Efron (1979). Figure 1.2 shows the result of 1000 repetitions of the basic bootstrap sampling procedure. This yields a 90% confidence interval [0.51, 0.99] for the population correlation coefficient. It provides a simple example of the use of the bootstrap. See Efron and Tibshirani (1993) for more detail.

We now turn to an example of the Monte Carlo Markov chain method. Suppose we want to estimate the proportion of the 10^{10} bootstrap samples with $T \geq .99$. We begin by choosing $t_0 = .9$ and estimating $P\{T > .9\} \doteq .4613$ based on 3000 Monte Carlo samples. Following this, take $t_i = .9i, 1 \leq i \leq 9$. The following table gives the Markov chain estimates of

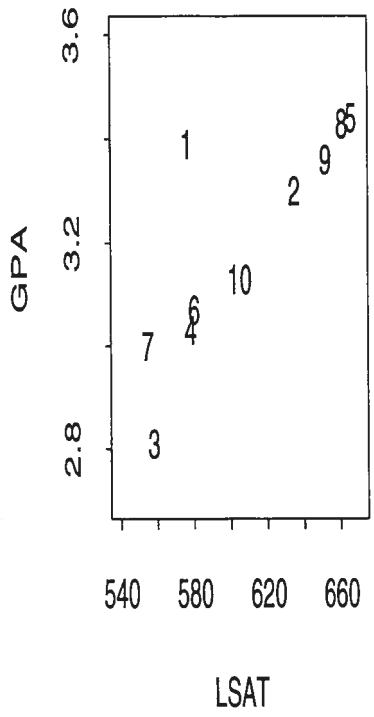

FIG. 1.1. *LSAT vs. GPA for 10 law schools*

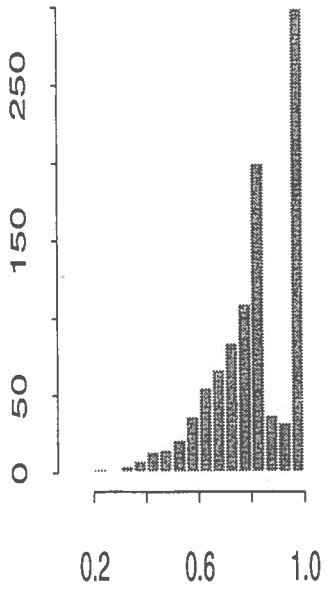

FIG. 1.2. *1,000 bootstraps for correlations*

$P\{T \geq t_i | T \geq t_{i-1}\}$. These are each based on running the Markov chain described above 3000 steps.

t_i	.91	.92	.93	.94	.95	.96	.97	.98	.99
$P\{T \geq t_i \mid T \geq t_{i-1}\}$	.997	.972	.989	.961	.984	.918	.897	.766	.710

These result in the estimate $\widehat{P}\{T > .99\} = .177$. For this example, exact enumeration of the bootstrap distribution of T is possible (see Diaconis and Holmes (1994a)) and gives $P\{T \geq .99\} = .1769$.

Remark 1.2. (1) In general, t_i can be chosen sequentially as an approximate median of T on $\{\underset{\sim}{x}^* : T \geq t_{i-1}\}$. This can be estimated by a sequence of preliminary walks. In fact, all that is needed is that $P\{T \geq t_i | T \geq t_{i-1}\}$ is not exponentially small. In our experience, an intuitive choice of grid values does fine. We derive the optimal choice in Diaconis and Holmes (1994b).

(2) Rates of convergence for this type of Markov chain which can be used as a guide to sample size choice are starting to become available. See

Diaconis and Saloff-Coste (1993) and Sinclair (1993) for recent surveys.

(3) Confidence intervals for $\widehat{P}$ can be based on central limit and large deviations theorems for Markov chains. See Höglund (1974), Gilman (1993) and the references cited there.

(4) The Markov chain described above is one of a host of chains described in Diaconis and Holmes (1994b). For example, there is no need to change only one value at a time. Any set of values may be chosen. This can be important for insuring connectivity of the underlying chain.

(5) One advantage of changing only a single value: for many statistics (including the correlation coefficient) fast updating algorithms can be used to avoid complete recomputation. See Section 3 in Diaconis and Holmes (1994a) for further discussion and references.

B. Approximate counting

The algorithm described above is derived from 15 years of development in the computer science literature. They consider the problem of approximate counting. Let $\mathcal{X}$ be a finite set. The problem is to approximate $|\mathcal{X}|$. We are given the ability to choose at random from $\mathcal{X}$. Without further restrictions, the best one could do would be to wait for repeated values in the sample. This takes order $|\mathcal{X}|^{1/2}$ steps and is virtually useless for the large $\#$-P complete problems of theoretical computer science.

Suppose further that there is a nested decreasing sequence of subsets $\mathcal{X} \supset \mathcal{X}_1 \supset \mathcal{X}_2 \cdots \supset \mathcal{X}_n$ with $k_i = |\mathcal{X}_i|/|\mathcal{X}_{i+1}|$ not too small, $k_0 = |\mathcal{X}|/|\mathcal{X}_1|$ and $|\mathcal{X}_n|$ small enough to be easily enumerated. We must also suppose the ability to sample uniformly from each $\mathcal{X}_i$ (at least approximately) one can then estimate $|\mathcal{X}_i|/|\mathcal{X}_{i+1}|$ by random sampling, providing an estimate denoted $\widehat{k_i}$ and then finally

$$\widehat{|\mathcal{X}|} = \widehat{k_0}\widehat{k_1}\ldots\widehat{k_{n-1}}\widehat{|\mathcal{X}_n|}$$

This is just the technique employed in Section A.

These ideas were introduced by Jerrum, Valiant, and Vazirani (1986) who showed that approximate counting and random generation are equivalent for self-reducible problems. Broder (1986) introduced the Markov chain aspect in his work on approximation of the permanent. If A is an $n \times n$ matrix $Per(A) = \sum_\pi \prod_{i=1}^n A_{i\pi(i)}$. The sum is over the symmetric group. If A is the adjacency matrix of a bipartite graph, $Per(A)$ counts the number of perfect matchings. Valiant (1979) has shown that evaluation of the number of matchings is $\#$-P complete. Broder introduced a random walk on the space of matchings and proved that one could get good approximations to the number of perfect matchings in polynomial time if this walk was rapidly mixing by using just self-reducilibity. Rapid mixing of Broder's walk for dense graphs was proved by Jerrum and Sinclair (1989) who introduced several new ideas (conductance arguments and the use of paths). This necessarily brief history omits mention of several contributors (Alon, Mihail and others). It also omits a description of the sizable body of

theory that has developed based on this work (Dyer, Frieze, Kannan, Lovasz, Simonovits and others). Fortunately, Sinclair (1993) gives a readable treatment of all of these issues.

One point is worth noting. The development in computer science took place in a theoretical context. The example in Section A may be close to the first implementation of these ideas. Evidently, applications present a field of further problems. One can only hope that the conversation continues.

2. Monte Carlo maximum likelihood and evaluation of partition functions

A. Maximum likelihood by Monte Carlo

In DNA fingerprinting problems one has samples of DNA from several individuals from which one would like to infer similarities. In one cleaned up version of the problem due to Geyer and Thompson (1992), the data consists of a binary matrix Y_{ij}, $1 \le i \le I$, $1 \le j \le J$, with each row representing an individual and the columns representing the lengths of DNA fragments. Geyer and Thompson used a model combining genetic insight with statistical mechanical simplicity. Let

$$\begin{aligned} U_i &= \textstyle\sum_j Y_{ij} = \#\text{ ones for } i^{\text{th}} \text{ individual} \\ V_j &= \textstyle\sum_i Y_{ij} = \#\text{ ones at } j^{\text{th}} \text{ level} \\ S_{ih} &= \textstyle\sum_j Y_{ij} Y_{hj} = \#\text{ common ones for } i^{\text{th}} \text{ and } h^{\text{th}} \text{ individual.} \end{aligned}$$

The model postulates

$$(2.1)\quad P\{Y_{ij} \,|\, 1 \le i \le I 1 \le j \le J\} = c(\theta)^{-1} \exp\left\{ \sum_i U_i \alpha_i + \sum_j V_j \beta_j + \sum_{i<k} S_{ih} \gamma_{ih} \right\}$$

Here, $\theta = \{\alpha_i, \beta_j, \gamma_{ih}\}$ has dimension $I + J + J(J-1)/2$ and $c(\theta)$ is a normalising factor. Genetic considerations force the constraint $\gamma_{ih} \ge 0$.

One problem now is, given a realization Y_{ij}, $1 \le i \le I$, $1 \le j \le J$, find the vector of parameters $\widehat{\theta}$ that maximizes the likelihood. In one of their examples, $I = 79, J = 32$. This leads to 607 parameters to be estimated. The practical version of the problem has additional complications: some binary values were missing and extensive prior knowledge of relatedness was applicable.

One contribution of the Geyer-Thompson work is a novel method for maximizing likelihoods. This is generally applicable and will be explained now. We begin by defining a general exponential family. Let $\mathcal{X}$ be a set and $\mu(dx)$ a measure defined on a suitable class of subsets of $\mathcal{X}$. Let $T : \mathcal{X} \to \mathbb{R}^d$. The exponential family through T, μ is the family of probability measures

$$P_\theta(dx) = c(\theta)^{-1} e^{T(x)\cdot\theta} \mu(dx), \quad \theta \in \Theta.$$

In (2.2) $c(\theta)$ is a normalizing constant and Θ is a prespecified subset of $\mathbb{R}^d$ such that

$$c(\theta) = \int e^{T(x)\cdot\theta}\mu(dx) < \infty \text{ for } \theta \in \Theta.$$

For fixed x, the maximum likelihood estimator (MLE) $\hat{\theta} = \theta(x)$ is the value minimizing $-\log P_\theta(dx) = -\log(c(\theta)) - T(x)\cdot\theta$. Thus computation of the MLE requires knowledge of the normalizing constant. Typically, this is not available in closed form.

Geyer and Thompson (1992) suggest a Monte Carlo approach for this: fix a value $\theta_0 \in \Theta$ and write

$$\begin{aligned} c(\theta) &= \int e^{T(x)\cdot\theta}\mu(dx) = c(\theta_0)\int e^{T(x)\cdot(\theta-\theta_0)}\frac{e^{T(x)\cdot\theta_0}}{c(\theta_0)}\mu(dx) \\ &= c(\theta_0)\int e^{T(x)(\theta-\theta_0)}P_{\theta_0}(dx). \end{aligned}$$

Run a Markov chain $X_0, X_1, X_2, \cdots$ with stationary distribution P_{θ_0}. Then,

$$d_n = \frac{1}{n}\sum_{n=1}^{n} e^{T(x_i)(\theta-\theta_0)} \longrightarrow c(\theta)/c(\theta_0)$$

for large n. So

$$-\widehat{\ell}_n = -\log d_n - T(x)\cdot\theta \longrightarrow \log c(\theta_0) - \log c(\theta) - T(x)\cdot\theta.$$

So the minimizer of $\widehat{\ell}_n$ approximates $\widehat{\theta}$. Practically, θ_0 should be well chosen else the chain will take a long time to converge. The following sequential scheme is used: after a few iterations with θ_0, replace θ_0 by $\widehat{\theta}$ and iterate this a few times.

In the DNA fingerprinting example, Geyer and Thompson carried out such an analysis for the model (2.1). They used the standard Metropolis algorithm changing single sites of $\{Y_{ij}\}$ and thinning down as usual to run a Markov chain with stationary distribution P_{θ_0}. The estimated parameters were used to compute correlations between observable quantities. The resulting correlation matrices were used to cluster the data. The results made sense in the context of the original problem and seemed useful. We refer to Geyer and Thompson (1992,1993) for examples and further details.

B. Computing partition functions

In statistical mechanics, the normalizing constant $c(\theta)$ is called the partition function. There has been a good deal of careful mathematical work which allows one to prove that exact evaluation or even good approximation of $c(\theta)$ *even at a fixed value of* θ is intractable in most cases of interest. Welsh (1993) and Jerrum-Sinclair (1991) review this work. These last authors also give an approximation scheme for the widely-studied case of the ferro-magnetic Ising model which uses randomness.

The Ising model is a standard exponential family for variables $Y_i \in \{-1, +1\}$, $1 \leq i \leq n$, with

$$P_\theta(Y_i; 1 \leq i \leq n) = c(\theta)^{-1} e^{\Sigma_{i,j} \gamma_{ij} Y_i Y_j - \beta \Sigma_i Y_i}.$$

Here γ_{ij} are constrained to be zero unless (i, j) is in E, the set of edges of a prespecified graph (often a rectangular lattice). If γ_{ij} are non-negative, one is said to be in the ferromagnetic case: configurations with high probability tend to be all the same (all ones or all minus ones). Jerrum and Sinclair (1991) gave a novel algorithm for estimating $c(\theta)$ in the ferromagnetic case. Their algorithm approximates $c(\theta)$ to relative error $(1 \pm \epsilon)$. They prove that their algorithm works with probability $1 - \delta$ and requires only a polynomial number of steps in $|E|, n, \log \frac{1}{\delta}$ and $1/\epsilon$, uniformly in θ.

It is worth remarking that any use of the Metropolis algorithm for estimating $c(\theta)$ is doomed to fail. Indeed, in dimension two or higher, Ising models have phase transitions which means that there are (at least) two high energy wells and the chain will spend an exponential time in one of them and so fail to mix rapidly.

Jerrum and Sinclair overcome this difficulty by re-expressing the partition function in a radically different way derived by physicists.

$$c(\theta) = A \sum_{X < E} W(X)$$

where the sum is over all subgraphs X of the underlying graph having all vertices of odd degree, A is a simple, easy to compute function of θ, and

$$W(X) = \mu^{|X|} \cdot \prod_{(i,j) \in X} \lambda_{ij}$$

with $\lambda_{ij} = \tanh(\gamma_{ij})$, $\mu = \tanh(\beta)$ and $|X|$ the number of vertices in X. The weights $W(X)$ are all positive and this allows Jerrum and Sinclair to run a Markov chain on the set of subgraphs X with stationary distribution proportional to $W(X)$. This is not the end of the story; there is a clever new idea that allows a good estimate $\widehat{c}(\theta)$ of $c(\theta)$ from this chain. For now, we will stop and state their main result as

THEOREM 2.1. *(Jerrum and Sinclair) Let E be a graph with $|E|$ edges and n vertices. For all $\gamma_{ij} \geq 0$ and β real.*

$$P\{1 - \epsilon \leq \left| \frac{\widehat{c}(\theta)}{c(\theta)} \right| \leq 1 + \epsilon\} > 1 - \delta.$$

Here $\widehat{c}$ is an estimator of c based on

$$O(\epsilon^{-2} |E|^2 \mu^{-4} \{\log(\delta^{-1}) + |E|\}) \text{ operations.}$$

The implicit constants are uniform in θ.

Remark 2.2. We find a comparison between the Geyer-Thompson and Jerrum-Sinclair work instructive. First, although problems and techniques are clearly related, these authors worked without knowledge of each other or associated literature. Geyer-Thompson tried their ideas out in several examples and proved nothing beyond convergence. Jerrum and Sinclair's work strongly suggests that what Geyer-Thompson were trying to do was impossible!

Jerrum and Sinclair proved convergence in polynomial time, but did not implement it. In a personal communication they suggest that although slow, their algorithm should be implementable, perhaps running with the same efficiency as the volume algorithm studied in the next section.

Their work suggests that there is much more proving and comparing to be done before the algorithms of Geyer-Thompson are accepted: there are versions of the Ising model where the partition function is known or accurately computable (e.g., for planar graphs). At the least the algorithms should be tried out here. In the other direction, we would like to encourage computer scientists to get implementable versions of their algorithms. It often leads to fascinating insights and seems like a simple embarrassment which the rest of the world has really noticed.

3. Contingency tables and approximate volumes

A. Random walks on contingency tables

A contingency table is an $I \times J$ array of non-negative integers. These arise in statistical analysis of cross-classified data (e.g., a 4×7 table might represent a classification of students by class (freshman, sophomore, junior, senior) amd seven categories of extra-curricular activity. We will work with tables having fixed row sums $(r_1, r_2, \cdots, r_I)$ and fixed column sums $(c_1, c_2, \cdots, c_J)$. Thus define

$$\mathcal{X}(\underset{\sim}{r}, \underset{\sim}{c}) = \{x_{ij} : \sum_j x_{ij} = r_i, \ \sum_i x_{ij} = c_j, \ 1 \le i \le I, \ \ 1 \le j \le J\}.$$

The problem considered is choosing a table uniformly at random from $\mathcal{X}(\underset{\sim}{r}, \underset{\sim}{c})$.

This problem arises in statistical work and in many areas of combinatorics. A survey of applications is given in the article of Diaconis and Gangolli in this volume. For even moderate size tables (e.g., 4×7) the enumeration problem gets wildly out of hand. The following simple random walk algorithm gives a satisfactory method of proceeding.

Pick a pair of rows (i, i') at random and a pair of columns (j, j') at random. These intersect in 4 entries:

$$\begin{array}{c} \\ i \\ i' \end{array} \begin{array}{c} \begin{array}{cc} j & j' \end{array} \\ \begin{pmatrix} \cdot & \cdot \\ \cdot & \cdot \end{pmatrix} \end{array}$$

These entries are changed by adding and subtracting 1 from the 4 entries in the pattern $\begin{matrix}+ & - \\ - & +\end{matrix}$ or $\begin{matrix}- & + \\ + & -\end{matrix}$ with probability 1/2. This doesn't change the row or column sums. If it forces negative entries, the step is not taken (the walk stays where it was last). It is clear that this walk is symmetric: the chance of going from x to y in one step is the same as the chance of going from y to x: one must pick the same rows and columns and the opposite pattern. It is easy to see that this walk is connected. A connected symmetric walk on a finite set converges to the uniform distribution (there is some holding probability so there are no periodicity problems).

There has been extensive practical work based on this algorithm and many variations (e.g., one needn't move one each time and more complicated patterns can be used). These algorithms appear to converge rapidly and seem to make a previously intractable problem easy.

Here is an example: in testing for statistical independence of row and column effects one often uses the chi-squared statistic

$$T(x) = \sum_{i,j} \frac{\{x_{ij} - r_i c_j/n\}^2}{r_i c_j/n}.$$

Here $n = r_1 + \cdots + r_I = c_1 + \cdots + c_J$. Diaconis and Efron (1986) wanted to know the distribution on $\mathcal{X}(\underset{\sim}{r}, \underset{\sim}{c})$. As an example, consider the 5×3 table

5	2	3	10
50	7	5	62
3	6	4	13
5	3	3	11
2	7	30	39
65	25	45	135

This table has $T(x) = 72.18$. Gangolli (1991) ran the Markov chain algorithm and recorded the values of $T(x)$ as the chain progressed. A histogram of these values is shown in Figure 3.1 below , it is to be noted that these computations took about 1/30th of the time of the exhaustive method (20 minutes) . In practice, one would use this histogram to estimate the proportion of tables with $T(x) \leq 72.18$ which is about .76086.

Gangolli (1991) also gave a complete enumeration: there are 239,382,173 tables in $\mathcal{X}(\underset{\sim}{r}, \underset{\sim}{c})$, this enumeration took 15 hours and 21 minutes of real time . Based on these, a complete enumeration of the distribution of T is shown in Figure 3.2. The simulation and the truth match closely.

These contingency table problems arise in many variations and extensions (e.g., 3-dimensional arrays). A calculus for deriving basic moves that uses algebraic geometry is given by Diaconis and Sturmfels (1993) who also discuss the statistical literature and competing algorithms. Diaconis, Graham

and Sturmfels (1993) give further extensions. A rigorous analysis of the running times of the suggested walks that applies to small tables (the case of most interest in statistical work) appears in Diaconis and Saloff-Coste (1993). Chung, Graham, and Yau (1994) have announced more powerful results. Gangolli (1991) has developed tempting conjectures about "the right answer" for running times. Mann (1994) gives exact formulae for $2 \times J$ and $3 \times J$ tables which permit checks of validity.

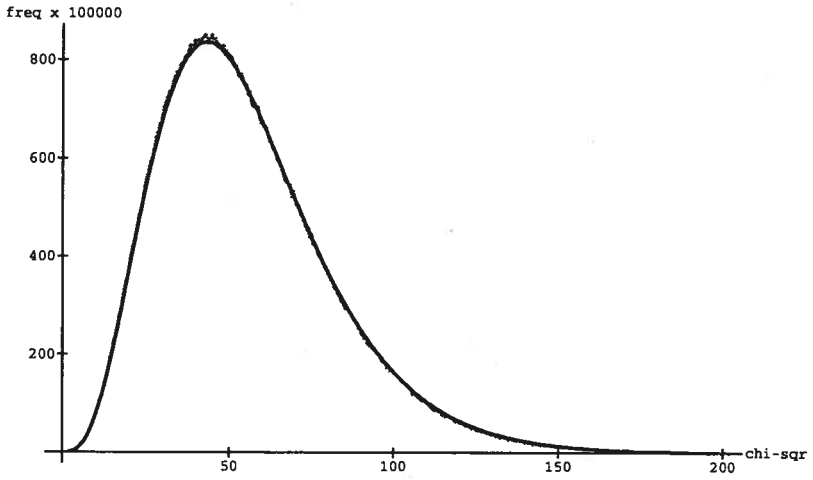

FIG. 3.1. *The exact distribution of $T(x)$ (in black)*

B. Approximating the volume

The work described above has close connections to a very healthy development in theoretical computer science. This asks for ways of computing the volume of a convex polyhedra in $\mathbb{R}^d$. There is a clear intuitive connection; the set $\mathcal{X}(\underset{\sim}{r}, \underset{\sim}{c})$ consists of the integer points in the convex polyhedron of all non-negative real arrays with a given set of row and column sums.

The volume problem is clearly basic. The problem is also hard: technically #-P complete. In fact, if use of randomness is forbidden, it can be proved that it is hard to get an approximate answer to within a factor of 2. Careful description and references are in the readable survey paper of Dyer and Frieze (1991).

Recent work shows that it is possible to get good approximations to the volume accurate to within a factor of $1 \pm \epsilon$, in a polynomial number of operations. Here, the parameters governing the problem may be taken as N – the number of hyperplanes specifying the polyhedron. The current best result, due to Lovasz-Kannan-Shimonovitz (1994) is a polynomial of degree 5 in N and $\log(1/\epsilon)$. The procedure uses a rapidly mixing Markov

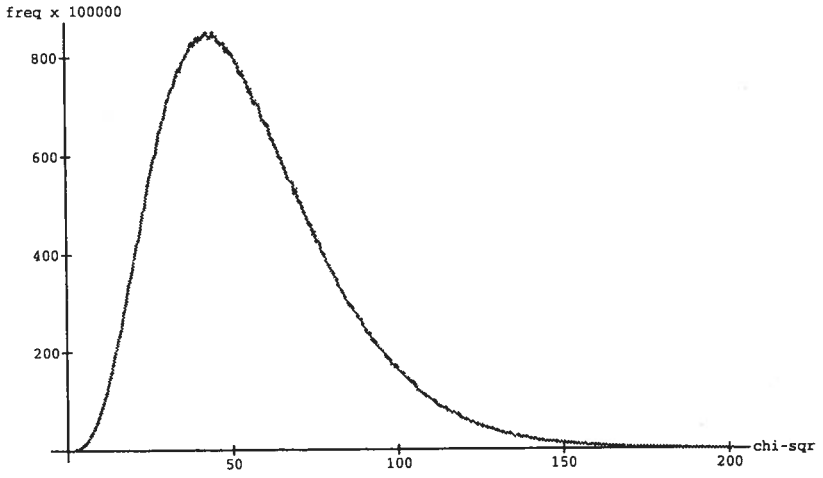

FIG. 3.2. *A Monte-Carlo approximation to Figure 3.1*

chain to sample from the polyhedron. As this result is of independent interest in statistical applications, we will spell it out.

Let C be a convex region in $\mathbb{R}^d$. Let $f(x)$ be a log concave probability density on C. The problem is to sample from $f(x)$. As an example, C might be the orthant

$$C = \{x_1 < x_2 < \cdots < x_d\}$$

and $f(x)$ might be the d-dimensional normal density (with mean vector μ) restricted to C. This is the problem of simulating normal vectors with a given order structure.

A simple algorithm runs as follows: starting from $x \in C$, pick a point on a small ball of radius δ uniformly. Now use the Metropolis algorithm to thin down this uniform walk to have density $f(x)$. This produces a reversible Markov chain with stationaary distribution $f(x)$ on C. The actual algorithm has some further ideas: The body C is first rounded by an affine transformation and there is some art in choosing a suitable δ. The proof of rapid mixing uses conductance which is bounded by a version of the Payne-Weinberger theorem of differential geometry. This coming together of different fields: statistics, convex geometry, differential geometry, linear programming and the theory of algorithms seems truly exciting.

At the workshop, Ravi Kannan some spectacular work with Dyer and Mount. They have found a way to adapt the continuous convex set problem to generate contingency tables! Briefly, their idea is this: consider the tables as lattice points in a convex polyhedra of dimension $(I-1)(J-1)$. Each

table is in a small "box" of the same size. Adding such boxes for tables near the edge makes a non-convex figure. They take the convex hull of this. Now, the continuous algorithm is used to sample from the uniform distribution on the augmented figure. For each sample point, one associates the table labeling the box containing the point.

Kannan and Mount prove that this generates approximately uniform tables in a polynomial number of steps.

The Kannan and Mount work requires a mild restriction on the row and column sums to guarantee success. Here is the essence of their results: Suppose $r_i > J(J-1)(I-1)$ for each i and $c_j > I(I-1)(J-1)$ for each j. Then, there is an algorithm for generating a random table distribution within ϵ of uniform in variation distance which runs in time bounded by a polynomial in I, J, $\max_{i,j}(\log r_i, c_j)$ and $\log(1/\epsilon)$.

For I and J small, the restrictions on r_i and c_j allow tables of practical interest. For example with $I = J = 4$ the restrictions are $r_i, c_j \geq 36$. The above is a simplified version of their work which is actually more general.

Moreover, they have programmed versions of their algorithm which at the time of this writing produces one "clean" table per second. It has given an independent check on other procedures, which revealed an embarrassing error had been made. One can hope that it can be adopted for other Monte Carlo procedures proposed by Diaconis and Sturmfels (1993).

4. Related literature

The use of Markov chains is in a phase of seemingly exponential growth. We have pointed to some of the pieces above. A completely independent development is occuring in the area of image processing. The recent survey volume by Barone, Frigessi and Piccioni (1992) gives a good set of pointers to this literature. There is an equally healthy development in the language of statistical computing. Volume 55, *no.*1 of the Journal of the Royal Statistical Society has several surveys and discussions by leading workers in this field. Finally we mention that workers in the computational side of statistical mechanics have not stopped with the 'Metropolis' algorithm! There has been a steady development over the years. One way to access this is to browse through the last few years of the *Journal of Statistical Mechanics.*

The work described leaves a rich legacy of problems for probablists and statisticians. Usually, no hint of rates of convergence are available (aside from essentially meaningless statements about "exponential convergence" with unspecified constants in and in front of the exponent). The statistical problems involved when making Markov chain runs also seem wide open. For instance, should one use all the data afterr an inital inhibiting time, or even after the initial period only use widely spaced instances. The first case provides better accuracy, but the correlation involved is difficult to estimate. The latter pays by lack of efficiency for providing a standard estimate forr the correlation structure. Another aspect yet to be addressed

is how to take into account any a priori knowledge.

REFERENCES

Barone,P., Frigessi A. and Piccioni M. (editors), 1992, Stochastic Models , Statistical Methods, and Algorithms in Image Analysis, Lecture Notes in Statistics, Springer–Verlag.

Broder, A. (1986). How Hard is it to Marry at Random? (On the approximation of the permanent.) *Proc. 18th ACM Symp. Th. Comp.*, 50–58. Erratum in *Proc. 20th ACM Symp. Th. Comp.*, 551.

Chung, F., Graham, R., and Yau, S.T. (1994). Unpublished manuscript.

Diaconis, P. and Efron, B. (1986). Testing for independence in a two-way table: new interpretations of the chi-square statistic (with discussion). *Ann. Statist.* **13**, 845–905.

Diaconis, P. and Gangolli, A. (1994). The number of arrays with given row and column sums. In this volume.

Diaconis, P., Graham, R.L., and Sturmfels, B. (1994). Primitive Partition Identities. Technical Report No. 9, Dept. of Statistics, Stanford University.

Diaconis, P. and Holmes, S. (1994a). Gray codes for randomization procedures. Technical Report No. 10, Dept. of Statistics, Stanford University.

Diaconis, P. and Holmes, S. (1994b). Random walks for bootstrap tails. Technical Report, Dept. of Mathematics, Harvard University.

Diaconis, P. and Saloff-Coste, L. (1993). Comparison theorems for reversible Markov chains. *Ann. Appl. Prob.* **3**, 696–730.

Diaconis, P. and Sturmfels, B. (1993). Algebraic algorithms for sampling from conditional distributions. Technical Report, Dept. of Mathematics, Harvard University.

Dyer, M. and Frieze, A. (1991). Computing the volume of convex bodies: a case where randomness proveably helps. *Probabilistic Combinatorics and its Applications*, B. Bollobàs, (ed.), *Proc. Symp. Appl. Math.* **44**, 123–170, Amer. Math. Soc., Providence.

Efron, B. (1979). Bootstrap methods: another look at the jackknife. *Ann. Statist.*, **7**, 1–26.

Efron, B. and Tibshirani, R. (1993). *An Introduction to the Bootstrap.* Chapman and Hall.

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

Geyer, C. and Thompson, E. (1992). Constrained Monte Carlo maximum likelihood for dependent data. *Jour. Roy. Statist. Soc.* B **54**, 657–699.

Geyer, C. and Thompson, E. (1993). Analysis of relatedness in the California condors, from DNA fingerprints. *Mol. Biol. Evol.* **10**, 571-589.

Gillman, D. (1993). A Chernoff bound for random walks on expander graphs. Preprint.

Hall, P. (1992). *The Bootstrap and Edgeworth Expansions*, Springer–Verlag.

Höglund, T. (1974). Central limit theorems and statistical inference for finite Markov chains. *Zeit. Wahr. Verw. Gebeits.* **29**, 123–151.

Jerrum, M. and Sinclair, A. (1989). Approximating the permanent. *Siam J. Comput.* **18**, 1149–1178.

Jerrum, M. and Sinclair, A. (1993). Polynomial time approximation algorithms for the Ising model, *Siam J. Comp.*, **22**, 1087-1116.

Jerrum, M., Valiant, L. and Vazirani, V. (1986). Random generation of combinatorial structures from a uniform distribution, *Theor. Comput. Sci.* **43**, 169–188.

Kannan, R., Lovasz, L. and Simonovitz, M. (1994). Unpublished manuscript.

Kannan, R. and Mount, J. (1994). Unpublished manuscript.

Mann, B. (1994). Some formulae for enumeration of contingency tables. Technical Report, Harvard University.

Sinclair, A. (1993). *Algorithms for random generation and counting*, Birkhäuser, Boston.

Valiant, L. (1979). The complexity of computing the permanent. *Theor. Comput. Sci.* **8**, 189–201.

Welsh, D. (1993). *Complexity : Knots, colourings and counting*, Cambridge University Press.

THE MOVE-TO-FRONT RULE FOR SELF-ORGANIZING LISTS WITH MARKOV DEPENDENT REQUESTS*

ROBERT P. DOBROW† AND JAMES ALLEN FILL‡

Abstract. We consider the move-to-front self-organizing linear search heuristic where the sequence of record requests is a Markov chain. Formulas are derived for the transition probabilities and stationary distribution of the permutation chain. The spectral structure of the chain is presented explicitly. Bounds on the discrepancy from stationarity for the permutation chain are computed in terms of the corresponding discrepancy for the request chain, both for separation and for total variation distance.

AMS(MOS) subject classifications. Primary 60J10; secondary 68P10, 68P05.

Key words. Markov chains, self-organizing search, move-to-front rule, convergence to stationarity, separation, total variation distance, coupling.

1. Introduction and summary. A collection of n records is arranged in a sequential list. Associated with the ith record is a weight r_i measuring the long-run frequency of its use. We assume that each $r_i > 0$ and normalize so that $\sum r_i = 1$. At each unit of time, item i is removed from the list with probability r_i and replaced at the front of the list. This gives a Markov chain on the permutation group S_n.

If we assume that items are requested independently of all other requests, this model for dynamically organizing a sequential file is known as the move-to-front (MTF) heuristic and has been studied extensively for over 20 years. Background references include Rivest (1976), Bitner (1979), Hendricks (1989), Diaconis (1993), and Fill (1995). In the case when all the weights are equal the model corresponds to a card-shuffling scheme known as the random-1-to-top shuffle; see Diaconis et al. (1992) for a thorough analysis in this case.

One objection to this model is that in practice record requests tend to exhibit "locality of reference." That is, frequencies of access over the short run may differ quite substantially from those over the long run. Knuth (1973) and Bentley and McGeoch (1985), among others, have noted that MTF tends to work even better in practice than predicted from the i.i.d. model. Knuth cites computational experiments involving compiler symbol tables and notes that typically "successive searches are not independent (small groups of keys tend to occur in bunches)."

Konnecker and Varol (1981) proposed modeling the request sequence along Markovian or autocorrelative lines. Lam et al. (1984) formally set up

* Research for both authors supported by NSF grant DMS–93–11367.

† NIST, Administration Building, A337, Gaithersburg, MD 20899–0001. E-mail Address: dobrow@cam.nist.gov

‡ The Department of Mathematical Sciences, Johns Hopkins University, 34th and Charles Streets, Baltimore, MD 21218-2689. E-mail Address: jimfill@jhuvm.hcf.jhu.edu