Numerical Implementation of Minimum Relative Entropy in Discrete Multivariate Probability Estimation

Yupeng Li and Clayton V. Deutsch

Although the Maximum Entropy(ME) principle has been successfully explored and applied in many disciplines, using it in discrete multivariate probability meets some special challenges. These challenges and its solutions are addressed in this paper. The minimum relative entropy principle which is more general form of ME principle is used in discrete multivariate probability estimation. When the constraints are the full set of second order marginals of the desired probability distribution,it is impossible to accomplish the probability estimation using the traditional Lagrange multiplier solution approach to MRE. In this paper, a kind of iterative scaling solution to the minimum relative entropy principle is introduced. Some numerical examples are used to illustrated of its simplicity comparing with the traditional Lagrange multiplier approach.

1 Introduction

Using the Maximum Entropy(ME) principle in discrete probability estimation, the traditional solution is to construct an objective function and try to solve it using the the Lagrange multipliers approach[1]. For all but the most simple case, the Lagrange multipliers $\{\lambda_1, \lambda_1, \cdots, \lambda_M\}$ that minimize the objective function $L(p_\ell, \pi_\ell, \lambda)$ cannot be found analytically. Instead, one must resort to the numerical methods. A variety of numerical methods can be used to calculate all the λs such as such as the works of Balestrino[2], Mead and Papanicolau [3] and Woodbury[4]. All of them are based on the traditional optimal techniques such as the gradient assent, conjugate gradient or Newton-Raphson approaches.

When the constraints for the desired discrete multivariate probability are the lower order marginal probability, it will be a big challenge using the traditional Lagrange multiplier to get the final solution. None of all the proposed methods such as conjugate gradient method or other similar process could be safely extended into multivariate probability distribution from the full sets of lower order marginals probabilities. The main challenge would be discussed in this paper.

One alternative solution, iterative scaling approach is gave based on the principle of minimum relative entropy(MRE), a more general ME principle. Iterative scaling based on MRE is straightforward. The theory and the numerical details are given in this paper. From the numerical examples presented in this paper, it is shown that the discrete MP are well reproduced.

2 Constraints in Discrete MP Inference

Generally, any lower order marginals probability $b_m, m = 1, \cdots, M$ which could be any order moments of the desired multivariate probability, are a linear combination of the subset of the multivariate probability $\{p_\ell, \ell = 1, \cdots, N\}$ as:

$$\sum a_{m\ell} p_\ell = b_m; \quad m = 1, \cdots, M \tag{1}$$

where $a_{m\ell}$ are the function related to the m and ℓ which will be set later, b_m are the marginal probabilities.

In practice, the multivariate probability is unknown and must be estimated from the different orders of marginal probabilities. The high order multivariate probability is difficult to obtain directly from the limited sampled locations. Assuming the lower order marginal probability distributions are given, our goal is to construct a multivariate probability distribution to satisfy all the given marginal probabilities. That is inferring a multivariate probability $\{p_\ell, \ell = 1, \cdots, N\}$ which will satisfy the constraints:

$$\sum p_\ell = 1 \tag{2}$$

$$\sum a_{m\ell} p_\ell = b_m; m = 1, 2, \cdots, M \tag{3}$$

Practically, the univariate marginal and bivariate marginal are always used as they can be reliably inferred from the data. In the later section of this paper, the bivariate marginal is used as an example to illustrate the multivariate probability estimation procedure. Note, the methodology can be extended to any order marginal probability.

3 Traditional Numerical Implementation of MRE and ME

In the long history of implementing ME or MRE in engineering or artificial intelligence, some numerical solutions have been proposed trying to obtain the Lagrange multipliers from the optimal equation such as Mead and Papanicolau [3] and Woodbury, [4]. Here is a short review of the approach proposed by Mead and Papanicolau[3]. In their approach, solving the Lagrange multiplier need to define a new optimal function:

$$\Gamma(\lambda_1, \lambda_2, \cdots, \lambda_m) = \log \Lambda + \sum_{m=1}^{M} \lambda_m b_m \tag{4}$$

Where Λ is the partition function. The desired set of Lagrange multipliers are the stationary point of the optimal function Γ, being the solution of the linear equation:

$$\frac{\partial \Gamma}{\lambda_m} = 0 \Longrightarrow \sum_{\ell=1}^{N} a_{m\ell} p_\ell = b_m, \quad m = 1, \cdots, M \tag{5}$$

Denote the vector of Lagrange multiplier as λ, and the gradient of Γ by $\mathbf{r}$, one can write the iteration equation for Newton's method as

$$\lambda^{\mathbf{n+1}} = \lambda^{\mathbf{n}} - \mathbf{H^{-1}r} \tag{6}$$

Where the matrix $\mathbf{H}$ is the Hessian matrix of the partition function. Each component of $\mathbf{r}$ in equation (6) is given by

$$r_m = b_m - \sum_{\ell=1}^{N} a_{m\ell} p_\ell^{(n)} \tag{7}$$

which is the residual between the input sample mean and the corresponding expected value over the estimated mean at n^{th} iteration. It is shown above that solving of iteration process of (6) and get all the λs will get the maximum entropy solution from the constraints.

4 Numerical Implementation of IS Solution

But for discrete multivariate probability in facies modeling, there are some special challenges. The first challenge comes from the multivariate data event space which are defined by the facies number and conditioning data number. It is very often that three facies are the minimum facies number that should be defined in facies modeling. Also, the conditioning data number for the unsampled location could be easily more than ten during conditioning simulation. As shown in Table 1, the multivariate data events number increases fast

Table 1: Data event space dimension

Total data locations	Two facies	Three facies	Four facies	Five facies
3	8	27	64	125
4	16	81	256	625
5	32	243	1,024	3,125
6	64	729	4,096	15,625
7	128	2,187	16,384	78,125
8	256	6,561	65,536	390,625
9	512	19,683	262,144	1,953,125
10	1,024	59,049	1,048,576	9,765,625

$$\begin{bmatrix}
1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0\\
0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0\\
0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1&0&0&1\\
1&1&1&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0\\
0&0&0&1&1&1&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&1&1&1&0&0&0\\
0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&1&1&1\\
1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1&1&1&1&1&1
\end{bmatrix}
\times
\begin{bmatrix} p_1\\ p_2\\ p_3\\ p_4\\ \vdots\\ p_{24}\\ p_{25}\\ p_{26}\\ p_{27} \end{bmatrix}
=
\begin{bmatrix} b_1\\ b_2\\ b_3\\ b_4\\ b_5\\ b_6\\ b_7\\ b_8\\ b_9 \end{bmatrix}$$

Figure 1: Example of univariate marginalization from trivariate probability

as the conditioning data locations increase. Thus, indexing and tracing all the multivariate probability for each of the multivariate data event would be the first challenge.

The solution to the index tracing challenge is ordering all the multivariate data events $\{\omega_\ell, \ell = 1, \cdot, N\}$ as a one dimension array[5]. The index ℓ for each data event $(\mathbf{u}_1 = k_1, \cdots, \mathbf{u}_n = k_n)$ is calculated from the outcomes itself as:

$$\ell = f(k_1, \cdots, k_n) = 1 + \sum_{\alpha=1}^{n} (k_\alpha - 1) \times K^{\alpha-1}, \quad k_\alpha = 1, \cdots, K \tag{8}$$

where the outcome k_α for each location is coming from an integer set $\{1, 2, ..., K\}$ which is obtained by ordering and coding all the categories in the set $\{e_1, \cdots, e_K\}$ into an integer, but the order of the categories does not matter. By assigning the unique index to each multivariate event and its probability state, in the marginalization, all the multivariate probability states will be easily traced by this index.

The second challenge comes from the marginalization operation that is building and saving the marginalization matrix $a_{m\ell}$ efficiently in the iteration process. In a very simple case, assuming one unsampled location is needed to estimated from two sampled locations and there are three possible categories. In this case, for univariate marginalization, each univariate marginal would comes from 9 trivariate probabilities. Taking one of the univariate marginal probability as an example, the probability of data event that facies one is found at location one $p(\mathbf{u}_1 = 1)$ can be calculated from the multivariate probability as:

$$p(\mathbf{u}_1 = 1) = \sum_{k_2=1}^{3} \sum_{k_3=1}^{3} p(k_1 = 1, k_2, k_3)$$

The indices of the multivariate probabilities that contribute to the univariate probability are: 1, 4, 7, 10, 13, 16, 19, 22 and 25 calculated from index function of equation (8). There could be 9 possible univariate probabilities. The calculation of these univariate probabilities is shown in Figure 1.

The same to bivariate marginalization. In this small example, all the bivariate probabilities calculated from multivariate probability is shown in Figure 2. As the data event space increases, so does the dimension

$$\begin{bmatrix}
0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0\\
0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0\\
0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&1\\
1&0&0&1&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&1&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&0&1&0&0\\
0&1&0&0&1&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&0&1&0\\
0&0&1&0&0&1&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&1&0&0&1\\
1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0\\
0&0&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&0&0\\
0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1
\end{bmatrix}
\times
\begin{bmatrix}
p_1\\p_2\\p_3\\p_4\\p_5\\p_6\\p_7\\p_8\\p_9\\p_{10}\\p_{11}\\p_{12}\\p_{13}\\p_{14}\\p_{15}\\p_{16}\\p_{17}\\p_{18}\\p_{19}\\p_{20}\\p_{21}\\p_{22}\\p_{23}\\p_{24}\\p_{25}\\p_{26}\\p_{27}
\end{bmatrix}
=
\begin{bmatrix}
b_1\\b_2\\b_3\\b_4\\b_5\\b_6\\b_7\\b_8\\b_9\\b_{10}\\b_{11}\\b_{12}\\b_{13}\\b_{14}\\b_{15}\\b_{16}\\b_{17}\\b_{18}\\b_{19}\\b_{20}\\b_{21}\\b_{22}\\b_{23}\\b_{24}\\b_{25}\\b_{26}\\b_{27}
\end{bmatrix}$$

Figure 2: the example of bivariate marginalization from a trivariate probability

Category number K	random number n	marginal order m	Dimension of $a_{m\ell}$
3	3	2	$\binom{3}{2} \cdot 3^2 \times 3^3 = 27 \times 27$
3	5	2	$\binom{5}{2} \cdot 3^2 \times 3^5 = 90 \times 243$
3	10	3	$\binom{10}{3} \cdot 3^3 \times 3^{10} = 3240 \times 59049$

Table 2: The dimension of marginal construction matrix

of the marginalization matrix. Generally, the dimension will be defined from the marginal order m, the total location number n and the facies number K as:

$$K^m \cdot \binom{n}{m} \times K^n \tag{9}$$

For example, if there are 20 random variables and 3 categories, the dimension of marginal construction matrix will be $1710 \times 3,484,784,401$. Some of the marginal construction matrix are as listed in table 2.

Handling such a higher dimension of matrix efficiently is a challenge in numerical implementation of this method. The naive data structure for a matrix is an array. Each entry $a_{m\ell}$ can be accessed by the two indices m and ℓ. Huge memory is needed to store all the entries to represent the matrix.

Although the dimension of $a_{m\ell}$ increases dramatically as the random variable number increases, there are many zero values in the matrix and non-zero values are always equal to one. Because in the marginalization, only some of the multivariate probability values will be summed up. Most of the elements in matrix $a_{m\ell}$ will be zero, as it is shown in Figure 1 and Figure 2. The marginalization computation can take advantage of this sparse matrix character and can be proceeded with a more efficient computation.

There are many ways to represent a sparse matrix [6]. The way used in this research is List of Lists (LIL). Other expression approaches could be used. By LIL approach, only the non-zero column index are stored. In this research, all the non-zero column indices are calculated from the multivariate event index function as in equation(8). The sparse matrix is saved by a one dimensional array (only the nonzero elements column number) and two parameters: the total row number and the non-zero elements number in each row of the naive matrix $a_{m\ell}$. Substantial memory requirement reduction is obtained and yields huge savings in memory when compared to a naive approach.

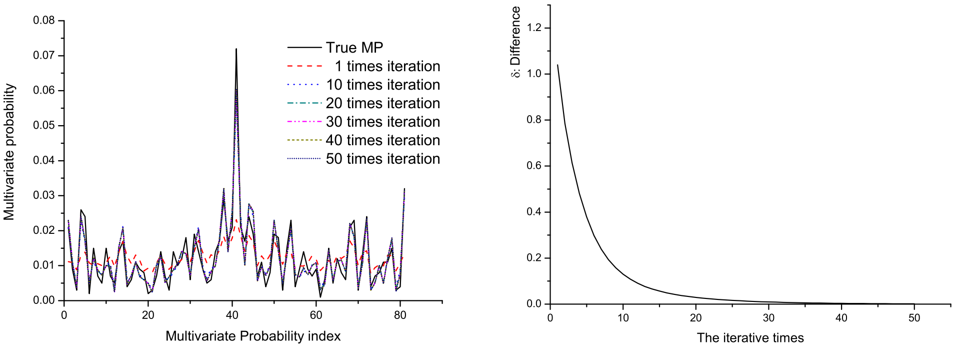

Figure 3: Left:The convergence of the iteration results to the true multivariate probability Right: The difference of from the estimated results and the requirement constraints

The marginalization computation can be done in a very fast linear operation style with a relative small storage requirement by taking advantage of the sparse matrix operation. More importantly, the sparse matrix is constant according to the order of the multivariate probability and only need to build one time after it is built in the first time. It saves a lot of CPU time when the marginalization is needed in every iteration.

5 Example Application

In this section, one example comes from estimating a true multivariate probability scanned from one training image. It will be shown that the multivariate probabilities are perfectly reproduced from the bivariate marginals.

Given a training image and one data configuration, the multivariate probability concerning the joint outcome for this group of locations will be scanned from the training image as discussed in the traditional multiple point geostatistics such as the work of Guardiano and Srivastava [7]. The bivariate probability b_m is calculated from the scanned multivariate probability using equation (1). The estimated multivariate probability $p^e$is obtained from the full set of bivariate probabilities using the iterative scaling approach as proposed in the paper of this volume. Then the estimated multivariate probability distribution p^e is compared with the original scanned true one p^a.

As shown in Figure (3), the multivariate probability sequence calculated from different iteration time converges to a feasible solution which is very close to the true multivariate probability.

Comparing the difference between the aiming bivariate probability b^a and the iterated bivariate probability b^e, as shown in Figure (3), the difference between them close to zero after 30 to 40 times iteration, that is $lim\delta = lim\|b^a - b^e\| \to 0$.

6 Discussion and Conclusion

The huge multivariate data event space which is decided from outcomes of each discrete random number and the number of discrete random variables brings special challenges to the numerical applications of the Minimum Relative Entropy approach in discrete multivariate probability estimation. Instead of using the traditional Lagrange multiplier solution approach to the minimum relative entropy principle, the iterative scaling approach is used in the multivariate probability estimation. Also, the sparse matrix numerical application is successfully implemented in the iterative scaling process where many times of discrete multivariate probability marginalization is calculated during the iterative process. Those two techniques makes the explicit multivariate probability estimation possible from its different lower order constraints for spatial discrete random variables.

References

[1] E. T. Jaynes. Information theory and statistical mechanics. *Physical Review*, 106(6):620–630, 1957.

[2] A. Balestrinoa, A. Caitia, and E. Crisostomia. Efficient numerical approximation of maximum entropy estimates. *International Journal of Control*, 79(9):1145–1155, 2006.

[3] Lawrence R. Mead and N. Papanicolaou. Maximum entropy in the problem of moments. *Journal of Mathematical Physics*, 25:2404–2417, 1984.

[4] Allan D. Woodbury. A fortran program to produce minimum relative entropy distributions. *Computer & Geosciences*, 30:131–138, 2004.

[5] C. V. Deutsch. *Annealing Techniques Applied to Reservoir Modeling and the Integration of Geological and Engineering (Well Test) Data*. PhD thesis, Stanford University, 1992.

[6] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press(Baltimore), 1996.

[7] F.Guardiano and R.M.Srivastava. *Multivariate geostatistics:Beyond bivariate moments*, volume 1 of *Geostatistics-Troia*. Kluwer Academic, 1993.