KYBERNETIKA — VOLUME 39 (2003), NUMBER 3, PAGES 307–315

REVERSIBLE JUMP MCMC FOR TWO-STATE MULTIVARIATE POISSON MIXTURES[1]

JANI LAHTINEN AND JOUKO LAMPINEN

The problem of identifying the source from observations from a Poisson process can be encountered in fault diagnostics systems based on event counters. The identification of the inner state of the system must be made based on observations of counters which entail only information on the total sum of some events from a dual process which has made a transition from an intact to a broken state at some unknown time. Here we demonstrate the general identifiability of this problem in presence of multiple counters.

Keywords: Bayesian inference, fault diagnostics, Poisson processes, reversible-jump MCMC
AMS Subject Classification: 62P30, 62F15, 62M05

1. INTRODUCTION

Our framework is a set of counters each of which is Poisson distributed and we observe the final value of the counters at the end of some period of time. During this time the process has changed from the initial state to the final state at unknown time. For both the initial and final periods there are unknown number of possible substates (i.e. event occurrence rates).

This type of situation arises, e.g. in fault diagnosis, when occurrences of some event during the operation of the device are monitored, and only the total number of the events is stored. From these counter values the goal is to decide, for example, whether the device is operating in normal states (intact device), or whether some fault has occurred during the operation and the device is malfunctioning. The estimation task is complicated due to the fact, that we have no prior knowledge on the event rate for either intact of faulty devices, hence we need to estimate the event rates for the states and the state transitions simultaneously. Collecting the data during actual operation from a large number of devices causes additional complication in the model as the devices may not be exactly similar. In a paper machine, for example, both the sensors and production line hardware are continuously updated. Similarly, in mass production devices, like a portable computers, the same model may contain various different hardware configurations and operating system

[1]Presented at the Workshop "Perspectives in Modern Statistical Inference II" held in Brno on August 14–17, 2002.

versions, possibly affecting the rates of the monitored events. To account for this variation we model all states as mixtures of Poisson processes, with unknown number of substates. We assume that the substates are constant during the operation, so that each device has zero or one unknown state transitions to be estimated.

In this paper we show that the two states of the process can be recognized by the final values of the counters when the dimension (the number of counters) is large. The one dimensional case is not invertible as the observed phenomena can be explained by varying the distribution of the transition point of which we have no direct information.

We apply the Metropolis–Hastings–Green method [3, 6] to estimate the Bayesian posterior of the parameters. This is then used to classify test cases based on multiple counters generated by the Poisson process. Previously similar approach for mixture distributions has been studied by Viallefont et al in [7] for Poisson mixtures and by Marrs in [4] for Gaussian mixtures. Our work has the feature of two state history, which to the authors' knowledge has had little attention so far.

An important application, for which the presented method was designed, is in fault diagnostic systems where the fault type of a device can be assessed based on the number of some counted events recorded during the age of the device. A non–faulty device would then have been in the same state through the whole time, whereas a broken would have shifted from the initial intact state to a broken state of some sort. The class of intact devices can also constitute of many inner states issuing from differences in the devices or the usage environment of the particular devices.

2. THE COUNTER GENERATION MODEL

We assume that the process can be sampled in two ways, so that some of the observations have only gone through a single state (an intact device), and some had two states (an initially intact device which then broke down). Also there may be several inner states in which the device may be broken or intact.

We will mark the values of n counters by $x \in \mathbb{Z}^n$. The latent, unobservable, variables determining the states of the process are denoted by $z_1 \in \{1, \dots, k_1\}$ for the initial state and $z_2 \in \{k_1 + 1, \dots, k\}$ for the broken state, with k_1 and k_2 the number of initial and broken states, respectively. The (unobserved) value of the counter i during the initial state of length τ is denoted by y_i, and the final observed value during time t is denoted by x_i.

The matrix of Poisson rates for $k = k_1 + k_2$ states is $\Lambda \in \mathbb{R}^{k \times n}$. The probabilities of the k_1 initial states are denoted by $\omega \in \mathbb{R}^{k_1}$ and the matrix of the transition probabilities from the initial to broken states by $T \in \mathbb{R}^{k_1 \times k_2}$.

Let us assume that there are n counters, the device is initially in one of k_1 intact states, z_1, with probability ω_{z_1}. At time τ the counter i will have the value y_i drawn from Poisson distribution with mean $\Lambda_{z_1,i}$, see the example in Section 5. Then at time τ it makes a transition to a broken state z_2, of which there are k_2 possibilities, with a probability T_{z_1,z_2}. In this state the counter i is again generated at a different rate $\Lambda_{z_2,i}$. The total values of the counter i is then x_i.

3. BAYESIAN ESTIMATION

We use the Bayes formula

$$p(\theta|\mathcal{D}, M) = \frac{p(\mathcal{D}|\theta, M)p(\theta|M)}{p(\mathcal{D}|M)} \tag{1}$$

to obtain the posterior distribution of the parameters Λ and τ, given the model M, which then can be used by a maximum probability classifier.

The prior distributions of the variables were

$$\begin{aligned}
&k_1 \sim \text{Uniform}\{1, \dots, k_{\max}\} && k_2 \sim \text{Uniform}\{1, \dots, k_{\max}\} \\
&\omega \sim \text{Dirichlet}(\underbrace{1, \dots, 1}_{k_1 \text{ times}}) && T_{i,:} \sim \text{Dirichlet}(\underbrace{1, \dots, 1}_{k_2 \text{ times}}) \\
&\Lambda_{i,j} \sim \text{Gamma}(\alpha, \beta) && \tau \sim \text{Uniform}[0, t] \\
&z_1 \sim \text{Bernoulli}(\omega) && z_2 \sim \text{Bernoulli}(T_{z_1,:}) \\
&y_i \sim \text{Poisson}(\tau \Lambda_{z_1,i})
\end{aligned} \tag{2}$$

where $T_{z_1,:}$ marks the z_1th row of the transition probability matrix T. The Bernoulli distribution here is a discrete distribution where each of the k values have the corresponding probabilities in the parameter vector.

The likelihood of n observed counter values x when the latent variables and parameters are given is then:

$$p(x|t, \tau, \Lambda, z) = \prod_{i=1}^{n} \text{Poisson}(x_i|\Lambda_{z_1,i}\tau + \Lambda_{z_2,i}(t - \tau)). \tag{3}$$

The Poisson rates of the initial states are estimated first using the common Poisson mixture Gibbs sampling [2] with Reversible Jump [3]. These are then kept fixed for the estimation of the second state rates. The number of latent states was chosen in both cases according to the most likely values based on the MCMC sampling. Note that in full Bayesian analysis no fixed values for any intermediate variables are estimated, but instead the posterior distribution of the variables is propagated throughout the analysis. The sequential estimation of the parameters was done for practical reasons, to simplify the analysis and to make the sampling faster.

The parameters are sampled by a Metropolis–Hastings–Green sampling with *Split–Merge* type reversible jump moves with the following procedure (here we use the upper index to enumerate through the data samples):

1. Draw each $\Lambda'_{i,j}$, $k_1 < i \leq k$, from $\Gamma(\alpha + \sum_{l:\{z_2^l = i\}}(x_i^l - y_i^l), \beta + \sum_{l:\{z_2^l = i\}}(t^l - \tau^l)$.
2. Draw each $T_{i,:}$ from Dirichlet(A), where $A \in \mathbb{R}^{k_2}$, $A_j = 1 + \sum_l I\{z_1^l = i \wedge z_2^l = j\}$, where $I\{\cdot\}$ is the characteristic function of the set $\{\cdot\}$.
3. Draw each z_1^i from Bernoulli(B), where $B \in \mathbb{R}^{k_1}$, $B_j = \omega_j \prod_l \text{Poisson}(y_l^i|\tau^i \Lambda_{z_1^i,l})$.
4. Draw each z_2^i from Bernoulli(C), where $C \in \mathbb{R}^{k_2}$, $C_j = \omega_{z_1^i} T_{z_1^i,j} \prod_l \text{Poisson}(x_l^i - y_l^i|(t^i - \tau^i)\Lambda_{z_2^i,l})$.

5. Draw each τ^i and y^i from their posterior by Metropolis–Hastings procedure.
6. In the Reversible Jump step either decide to try a split or merge a random kernel (the Poisson rate parameters of some latent state) with probability 1/2.
7. Use the split (merge) map (see below) to a kernel κ chosen at random.
8. Reallocate the latent states $z_2^i = \kappa$, (or while merging $z_2^i = \kappa \vee z_2^i = \kappa + 1$) by drawing from Bernoulli(D), where $D \in \mathbb{R}^{k_2}$,
   $D_j = \omega_{z_1^i} T_{z_1^i, j} \prod_l \text{Poisson}(x_l^i - y_l^i | \tau^i \Lambda_{z_2^i, l})$.
9. Accept the split proposal with probability
   $$\min\left\{1, \frac{p(\mathcal{D}|\theta')}{p(\mathcal{D}|\theta)} \frac{|J|}{P_{\text{alloc}}}\right\}, \tag{4}$$
   where P_{alloc} is the reallocation probability of the latent z and $|J|$ is the Jacobian determinant of the split map (see below). In case of merge the acceptance probability is
   $$\min\left\{1, \frac{p(\mathcal{D}|\theta')}{p(\mathcal{D}|\theta)} \frac{P_{\text{alloc}}}{|J|}\right\}, \tag{5}$$
   in here P_{alloc} is the reallocation probability of the the latent z if splitting from the new state back to the original with the map whose Jacobian is $|J|$.

4. THE REVERSIBLE JUMP STEP

We use a *Split–Merge* procedure to simulate the jump between dimensions. This is done by randomly choosing either split or merge with equal probability. In splitting a kernel the center of mass of the two new kernels is preserved:

$$\omega_1'\lambda_1' + \omega_2'\lambda_2' = \omega\lambda. \tag{6}$$

The other parameter values are copied from the original. In order the merge–step (inverse map of split) to be reversible the other parameters are copied from one of the two components chosen at random. The new values for ω_1, ω_2, λ_1 and λ_2 are then mapped such that all possible positive values of λs satisfying equation (6) are equally probable. This is the following map, $(\lambda_i, \omega_i, u, v) \mapsto (\lambda_i', \lambda_{i+1}', \omega_i', \omega_{i+1}')$, in which the latent state i is split, where $u, v \in [0, 1]$ are drawn from the uniform distribution:

$$\begin{array}{rcl} \omega_i' & = & u\omega_i \\ \omega_{i+1}' & = & (1-u)\omega_i \\ \lambda_i' & = & v\lambda_i \\ \lambda_{i+1}' & = & \frac{\omega_i\lambda_i - \omega_i'\lambda_i'}{\omega_{i+1}'}. \end{array} \tag{7}$$

The Jacobian determinant is then

$$J = \frac{\omega_i \lambda_i}{u-1}. \tag{8}$$

5. EXAMPLES

5.1. A 2–dimensional model

Take for an example the following: the initial (intact) states are labeled as $\{1,2\}$ and the final (broken) states are $\{3,4,5,6\}$, where we divide the broken states to two groups, this would present that the device has two different categories of malfunctions, with the $\{3,4,5\}$ as one category (*class 1*) and the state $\{6\}$ alone (*class 2*), see Figure 1. The initial states were equally probable with Poisson parameters (event rates):

$$\Lambda = \begin{pmatrix} 1 & 2 \\ 2 & 4 \\ \hline 2 & 8 \\ 7 & 7 \\ 9 & 3 \\ 15 & 15 \end{pmatrix}, \quad T = \begin{pmatrix} 0 & 1/2 & 1/4 & 1/4 \\ 1/2 & 1/4 & 1/4 & 0 \end{pmatrix}. \tag{9}$$

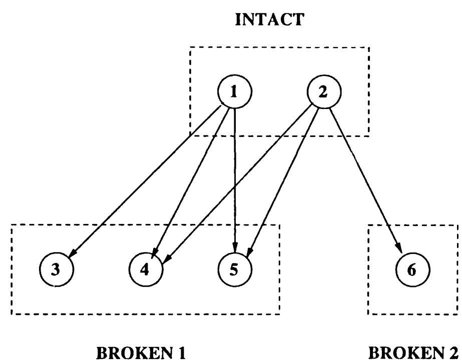

Fig. 1. The state diagram of the example system.

The simulated data had 25 samples from the initial model, representing intact devices and 100 samples from the two state model, representing broken devices (see Figure 2). The transition time was uniformly distributed.

The parameters of the intact devices were simulated for 1000 rounds and the parameters of broken devices were simulated for 3000 rounds, with priors $\alpha = 2$ and $\beta = 1/8$ in equation (2). The simulation of the functions in Matlab with Compaq AlphaStation XP1000 processor lasted for 10 min. The convergence of the simulation was tested using the Kolmogorov–Smirnov test [6] after we selected a proper subset of the datasamples based on the autocorrelation time to avoid the dependence of consecutive samples [5]. The estimated probabilities, taken as the mean of the simulated samples, with the most likely number of kernels of the initial

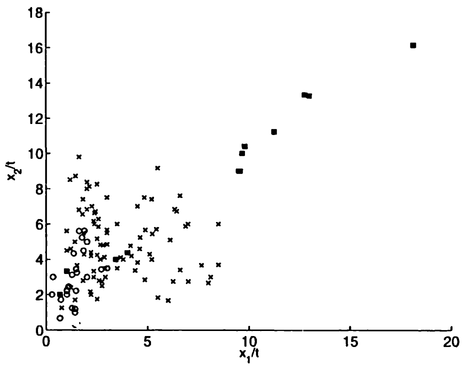

Fig. 2. Samples of Example 1, intact devices marked with circles, broken class 1 with crosses and broken class 2 with boxes.

latent states were:

$$\hat{\Lambda} = \begin{pmatrix} 1.2 & 2.0 \\ 2.0 & 4.1 \\ \hline 2.0 & 8.6 \\ 7.3 & 7.2 \\ 9.1 & 3.1 \\ 15.6 & 15.4 \end{pmatrix} . \tag{10}$$

The transition probabilities to the first broken class were:

$$\hat{T} = \begin{pmatrix} 0.25 & 0.47 & 0.28 \\ 0.29 & 0.49 & 0.22 \end{pmatrix} . \tag{11}$$

The estimated distribution of the initial states was:

$$\hat{\omega} = \begin{pmatrix} 0.55 \\ 0.45 \end{pmatrix} . \tag{12}$$

The estimated parameters were tested in a simulated classification task and the confusion matrix C (for 100 samples from initial process, representing intact device, and 500 samples from the two-state process, representing broken devices) compared to the 3-Nearest Neighbor classifier is:

$$C = \begin{pmatrix} 0.92 & 0.080 & 0 \\ 0.22 & 0.72 & 0.064 \\ 0.13 & 0.14 & 0.75 \end{pmatrix} , \quad C_{3\text{-NN}} = \begin{pmatrix} 0.57 & 0.43 & 0 \\ 0.13 & 0.85 & 0.011 \\ 0.085 & 0.64 & 0.27 \end{pmatrix} . \tag{13}$$

From these we can see that neither method mistakes an intact device with broke in class 2 (the first row), but that the Bayesian classifier confuses much less an intact with the broken class 1.

The estimation of the τ parameters as the median of the samples for each data sample is plotted in Figure 3. The lines are the *90 % Highest Probability Density intervals* (90 % HPD intervals) [1]. The uncertainty of the estimates comes from the facts, that the data does not contain direct information of the transition point, and there is only one observation related to estimation of each transition point, and thus the estimates tend to come from the uniform prior. In classification estimation we marginalized out the τ dependence by summing over the estimated τ values of the dataset.

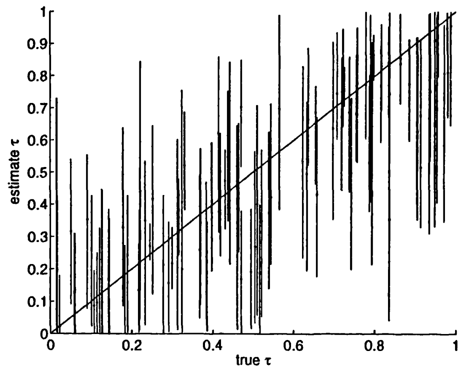

Fig. 3. The true relative values of τ for the 2–D example compared against the median estimates with the 90 % HPD intervals.

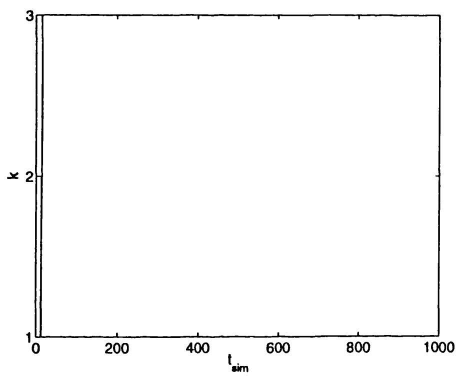

Fig. 4. The true relative values of τ for the 2–D example compared against the median estimates with the 90 % HPD intervals.

The classification based on maximal probability borders compared with the 3–Nearest Neighbor classifier can be seen in Figure 5.

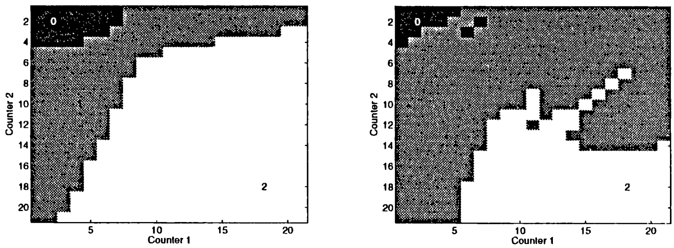

Fig. 5. On the left is the maximal probability classification borders for the 2-dimensional example, the value counters in unit time, and on the right the 3–Nearest Neighbor classifier for the same data. The black is the area of intact devices (class 0) and the gray broken class 1 and white broken class 2.

5.2. A 5–dimensional model

This is otherwise similar to the model above but with 5 counters, with 2 classes: intact and broken. The estimation with a similar dataset, 25 intact devices, 100 broken, with simulation length of 3000 results to:

$$\Lambda = \begin{pmatrix} 1 & 2 & 2 & 1 & 5 \\ 1 & 3 & 1 & 4 & 1 \\ \hline 2 & 8 & 7 & 5 & 1 \\ 7 & 7 & 1 & 2 & 1 \\ 9 & 1 & 2 & 1 & 6 \\ 5 & 5 & 4 & 8 & 8 \end{pmatrix}, \quad \hat{\Lambda} = \begin{pmatrix} 0.92 & 2.0 & 2.2 & 0.91 & 4.5 \\ 1.2 & 3.2 & 1.1 & 3.7 & 0.91 \\ \hline 2.2 & 7.6 & 6.7 & 4.9 & 0.82 \\ 7.1 & 6.6 & 1.0 & 2.0 & 1.1 \\ 9.2 & 0.68 & 1.9 & 1.0 & 6.1 \\ 5.8 & 5.4 & 3.5 & 8.9 & 8.5 \end{pmatrix}. \tag{14}$$

The confusion matrix C in this case compared to the 3-NN classifier is:

$$C = \begin{pmatrix} 0.92 & 0.070 & 0.010 \\ 0.13 & 0.86 & 0.016 \\ 0.016 & 0.063 & 0.92 \end{pmatrix}, \quad C_{3\text{-NN}} = \begin{pmatrix} 0.75 & 0.23 & 0.020 \\ 0.078 & 0.91 & 0.016 \\ 0.047 & 0.36 & 0.59 \end{pmatrix}. \tag{15}$$

6. CONCLUSIONS

We have demonstrated that it is possible to estimate the parameters and states of Poisson mixture processes containing transition between states at unknown time, using Reversible Jump MCMC method. The estimation becomes always more difficult when the transition time τ is accumulated more towards the end of total time, in which case the counters only exhibit behavior of the initial states. In this estimation the availability of data for purely intact devices, and presence of more than one counter to record events, is critical.

ACKNOWLEDGEMENTS

This work has been done in collaboration with the Nokia corporation. The authors would like to thank Dr. Aki Vehtari for his advice on the convergence diagnostics and the Reversible Jump algorithm.

(Received November 29, 2002.)

REFERENCES

[1] Ming-Hui Chen, Qi-Man Shao, and J. Q. Ibrahim: Monte Carlo Methods in Bayesian Computation. Springer, New York 2000.

[2] S. Geman and D. Geman: Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Trans. on Pattern Analysis and Machine Intelligence *64* (1984), 2, 721–741.

[3] P. Green: Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. Biometrika *82* (1995), 711–732, http://www.stats.bris.ac.uk/pub/reports/MCMC/revjump.ps.

[4] A. D. Marrs: An application of Reversible–Jump MCMC to multivariate spherical Gaussian mixtures. In: Advances in Neural Information Processing Systems 10 (M. I. Jordan, M. J. Kearns, and S. A. Solla, eds.), MIT Press, Cambridge, MA 1998.

[5] R. M. Neal: Probabilistic Inference Using Markov Chain Monte Carlo Methods. Technical Report No. CRG-TR-93-1, Dept. of Computer Science, University of Toronto, 1993, ftp://ftp.cs.toronto.edu/pub/radford/review.ps.Z.

[6] C. P. Robert and G. Casella: Monte Carlo Statistical Methods. Springer–Verlag, New York 1999.

[7] V. Viallefont, S. Richardson, and P. Green: Bayesian analysis of Poisson mixtures. J. Nonparametric Statistics *14* (2002), 1-2, 181–202.

Jani Lahtinen, M.Sc. and Dr. Jouko Lampinen, Laboratory of Computational Engineering, Helsinki University of Technology, P. O. Box 9400, FIN-02015 HUT. Finland.
e-mails: jani.lahtinen@hut.fi, jouko.lampinen@hut.fi