Bayesian Learning of Finite Asymmetric Gaussian Mixtures by MH-within-Gibbs Method

Shuai Fu[1] and Nizar Bouguila[2]

[1] Concordia University, Montreal, Canada. **Email:** f_shuai@encs.concordia.ca
[2] Concordia University, Montreal, Canada. **Email:** bouguila@ciise.concordia.ca

Abstract. Asymmetric Gaussian mixture (AGM) model has been proved more flexible than the classic symmetric Gaussian mixture model because no constrain will be given that the distribution of observations should be symmetric from the proposed mean of every dimension. This paper introduces a fully Bayesian learning method using Metropolis-Hastings within Gibbs sampling method and its applications.

Keywords - Asymmetric Gaussian mixture, AGM, Asymmetric Gaussian distribution, AGD, Metropolis-Hastings method, Gibbs sampling, Markov chain Monte Carlo, MCMC, MH-within-Gibbs, Bayesian analysis

1 Introduction

This paper is trying to introduce a new Bayesian model which combines AGM model (Elguebaly and Bouguila, 2013) [1] with random sampling based Markov chain Monte Carlo (MCMC) method for mixture parameter learning. Distinguished to traditional expectation-maximization (Wu, 1983) [2] learning method, sampling-based Markov chain Monte Carlo method considers not only posterior distribution (likelihood) but also involves proposed prior and posterior distributions for every mixture parameter which is not deterministic and brings randomness into the learning procedure. As a variation of classic MCMC method, the Metropolis-Hastings-within-Gibbs sampling method (Bouguila, Ziou and Hammoud, 2008) [3] takes the advantages of both Gibbs sampling and Metropolis-Hastings method while sampling from proposed distributions of mixture parameters could be flexible and adjustable depending on specific applications and datasets.

Compared to classic Gaussian distribution, the mean of asymmetric Gaussian distribution (AGD) is as the same as symmetric one, however, AGD is using two independent standard deviation vectors to define 'left' and 'right' parts of the model. Therefore, AGD can be seen as a combination of multiple symmetric Gaussian distributions depending on the dimension d thus it can be easily extended as a multi-dimensional model.

Monte Carlo method is well known as an effective random sampling method for estimation purpose and Markov chain defines rules how model transfers from

one to another. Based on the characteristics of the AGM model, MCMC method is selected to be the learning algorithm for the estimation of mixture parameters.In fact,Metropolis-Hastings method (Hastings, 1970) [4] and Gibbs sampling (Geman and Geman, 1984) [5] are two most widely used implementations of MCMC method.Although the acceptance ratio of Gibbs sampling is always be 1 means every move will be accepted, when direct sampling from the distribution is not easy, Metropolis-Hastings method provides an alternative.

According to the organization of this paper, Section 2 will be the illustration of AGM model and Bayesian learning processes. In Section 3, both experimental and real applications will be applied to the mixture model and the results will be analyzed.

2 Bayesian Model

2.1 Asymmetric Gaussian Mixture Model

Assuming that the AGM model has M components and the likelihood function (Elguebaly and Bouguila, 2013) [1] is defined as follows:

$$P(\chi|\Theta) = \prod_{i=i}^{N} \sum_{j=1}^{M} p_j P(X_i|\xi_j) \tag{1}$$

where $\chi = (X_1, ..., X_N)$ is the set of observations with total amount of N, $\Theta = \{p_1, ..., p_M, \xi_1, ..., \xi_M\}$ represents the parameter sets of M mixture components for the AGM model, p_j ($0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1$) is the weight for each component in the mixture model and ξ_j is the AGD parameters of mixture component j.

In order to simplify the Bayesian learning process, we introduced the membership vector $Z_i = (Z_{i1}, ..., Z_{iM})$. For each observation $X_i, 1 < i < N, Z_i = (Z_{i1}, ..., Z_{iM})$ is a M-dimensional membership vector which indicates X_i belongs to a specific component (Bouguila, Ziou and Monga, 2006) [6], such that:

$$Z_{ij} = \begin{cases} 1 & \text{if } X_i \text{ belongs to component } j \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

in another word, $Z_{ij} = 1$ only if observation X_i has the highest probability of belonging to component j and accordingly, for other components, $Z_{ij} = 0$.

Combine the Eq. (1) and Eq. (2) together we derived the new density function:

$$P(\chi, Z|\Theta) = \prod_{i=1}^{N} \prod_{j=1}^{M} (p_j P(X_i|\xi_j))^{Z_{ij}} \tag{3}$$

Specifically for AGM model, concerning observation X belongs to mixture component j and $X = (x_1, ..., x_d)$, the probability density function (Elguebaly

and Bouguila, 2013) [1] can be defined as following:

$$P(X|\xi_j) \propto \prod_{k=1}^{d} \frac{1}{(\sigma_{l_{jk}} + \sigma_{r_{jk}})} \times \begin{cases} \exp\left[-\frac{(x_k-\mu_{jk})^2}{2(\sigma_{l_{jk}})^2}\right] & if\ x_k\ <\ \mu_{jk} \\ \exp\left[-\frac{(x_k-\mu_{jk})^2}{2(\sigma_{r_{jk}})^2}\right] & if\ x_k\ \geqslant\ \mu_{jk} \end{cases} \tag{4}$$

where $\xi_j = (\mu_j, \sigma_{lj}, \sigma_{rj})$ is the set of parameters of component j and $\mu_j = (\mu_{j1}, ..., \mu_{jd})$ is the mean, $\sigma_{lj} = (\sigma_{lj1}, ..., \sigma_{ljd})$ and $\sigma_{rj} = (\sigma_{rj1}, ..., \sigma_{rjd})$ are the left and right standard deviations for AGD. To be more specific, $x_k \sim N(\mu_{jk}, \sigma_{ljk})$ $(x_k < \mu_{jk})$ and $x_k \sim N(\mu_{jk}, \sigma_{rjk})$ $(x_k \geqslant \mu_{jk})$ for each dimension.

2.2 Learning Algorithm

Before describing detailed MH-within-Gibbs learning steps, some priors and posteriors need to be clarified that having the membership vector in hand, we denote the postorior probability of membership vector Z as $\pi(Z|\Theta, \chi)$ (Elguebaly and Bouguila, 2011) [7]. Therefore, during the iteration t of learning process:

$$Z^{(t)} \sim \pi(Z|\Theta^{(t-1)}, \chi) \tag{5}$$

then derive the number of observations belonging to a specific component j according to $Z^{(t)}$ as follows:

$$n_j^{(t)} = \sum_{i=1}^{N} Z_{ij}\ (j = 1, ..., M) \tag{6}$$

thus $n^{(t)} = (n_i^{(t)}, ..., n_M^{(t)})$ represents the number of observations belonging to each mixture component.

Since the mixture weight p_j $(0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1)$, a nature choice of the prior is Dirichlet distribution as following (Marin, Mengersen and Robert, 2005) [8]:

$$\pi(p_j^{(t)}) \sim D(\gamma_1, ..., \gamma_M) \tag{7}$$

where γ_j) is known hyperparameter. Consequently, the posterior of the mixture weight p_j is:

$$P(p_j^{(t)}|Z^{(t)}) \sim D(\gamma_1 + n_1^{(t)}, ..., \gamma_M + n_M^{(t)}) \tag{8}$$

As mentioned before, direct sampling of mixture parameters $\xi \sim P(\xi|Z, \chi)$ could be difficult so Metropolis-Hastings method should be involved to propose proposal distributions for $\xi^{(t)} \sim q(\xi|\xi^{(t-1)})$. To be more specific, for parameters of AGM model which are μ , σ_l and σ_r, we choose proposal distributions as follows:

$$\mu_j^{(t)} \sim N_d(\mu_j^{(t-1)}, \Sigma) \tag{9}$$

$$\sigma_{lj}^{(t)} \sim N_d(\sigma_{lj}^{(t-1)}, \Sigma) \tag{10}$$

$$\sigma_{rj}^{(t)} \sim N_d(\sigma_{rj}^{(t-1)}, \Sigma) \tag{11}$$

the proposal distributions are d-dimensional Gaussian distributions with Σ as d x d identity matrices which make the sampling as random walk MCMC process.

As the most important part of Metropolis-Hastings method, at the end of each iteration, for new generated mixture parameter set $\Theta^{(t)}$, an acceptance ratio r needs to be calculated in order to make a decision whether they should be accepted or discarded for the next iteration. For acceptance ration r, compute:

$$r = \frac{P(\chi|\Theta^{(t)})\pi(\Theta^{(t)})q(\Theta^{(t-1)}|\Theta^{(t)})}{P(\chi|\Theta^{(t-1)})\pi(\Theta^{(t-1)})q(\Theta^{(t)}|\Theta^{(t-1)})} \tag{12}$$

where $\pi(\Theta)$ is the proposed prior distribution which can be decomposed to d-dimensional Gaussian distributions that $\mu \sim N_d(\eta, \Sigma)$ and $\sigma_l, \sigma_r \sim N_d(\tau, \Sigma)$ given known hyperparameters η and τ. Since mixture weight p has been computed previously during the Gibbs sampling part, it should not be included in Eq. (12). Further information about the calculation of acceptance ratio r is explained in Appendix A.

Once acceptance ratio r is derived by Eq. (15), compute acceptance probability $\alpha = min[1, r]$ (Luengo, D. and Martino, L., 2013) [9]. Then $u \sim U_{[0,1]}$ is supposed to be generated randomly. If $\alpha < u$, the proposed move should be accepted and parameters should be updated by $p^{(t)}$ and $\xi^{(t)}$ for next iteration. Otherwise, discard $p^{(t)}$, $\xi^{(t)}$ and set $p^{(t)} = p^{(t-1)}$, $\xi^{(t)} = \xi^{(t-1)}$.

We summarize the MH-within-Gibbs learning process for AGM model as the following steps:

Input: Data observations χ and component number M
Output: AGM mixture parameter set Θ

1. Initialization
2. Step t: For $t = 1, \ldots$

   Gibbs part

   (a) Generate $Z^{(t)}$ from Eq. (5)
   (b) Compute $n_j^{(t)}$ from Eq. (6)
   (c) Generate $p_j^{(t)}$ from Eq. (8)

   Metropolis-Hastings part

   (d) Sample $\xi_j^{(t)}$ $(\mu_j^{(t)}, \sigma_{lj}^{(t)}, \sigma_{rj}^{(t)})$ from Eqs. (9) (10) (11)
   (e) Compute acceptance ratio r from Eq. (15)
   (f) Generate $\alpha = min[1, r]$ and $u \sim U_{[0,1]}$
   (g) If $\alpha \geq u$ then $\xi^{(t)} = \xi^{(t-1)}$

3 Experimental Results

3.1 Design of Experiments

We apply the AGM model to both synthetic data and intrusion detection application. For synthetic data validation part, testing observations will be generated from AGD with known component number M. NSL-KDD dataset (Tavallaee, M., Bagheri, E., Lu, W. and Ghorbani, A.A., 2009) [10] is selected for intrusion detection part.

3.2 Synthetic Data

The main goals of this section are feasibility analysis and evaluating the efficiency of the AGM learning algorithm.

Observation number is set to 300 spliting into two groups. Fig. 1 displayed observation distribution and its histograms for both dimensions. Hyperparameters are set accordingly as γ_j introduced in Eq. (7) is 1 (Stephens, M., 2000) [11] for sampling of proposed mixture weight p_j, η and τ are d-dimensional zero vectors for prior distributions of mixture parameter ξ.

Probability density diagrams with different component values of k ($k = 1, \ldots, 5$) are shown in Fig. 2. Polylines ploted the trace of accepted moves of each component. Obviously, the best result is derived when component number k equals to 2 which is the same as the number of observation groups. Furthermore, it has the maximum marginal likelihood value which also indicates the best component number.

AGM learning statistics summarized in Table 1 shows that the algorithm leads to the convergence within 300 iterations and accepted moves are less than 35. From the best test result ($k = 2$), algorithm was converged after 21 accepted moves without abnormal shaking and drifting.

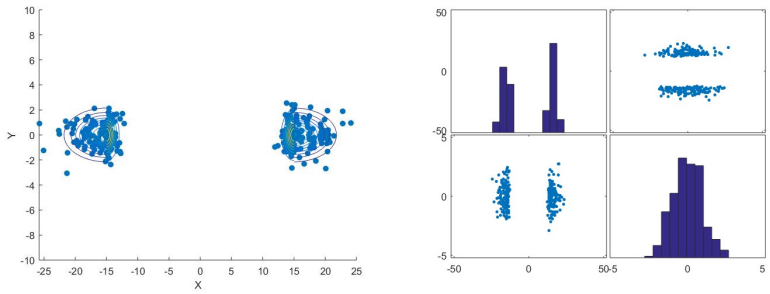

Fig. 1. Original synthetic observations

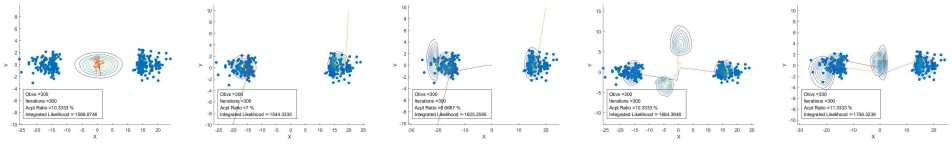

Fig. 2. AGM learning results (component number $k = 1, \ldots, 5$, iteration=300)

Table 1. AGM learning statistics

Comp. Num. k	***Moves Accepted***	***Acceptance Ratio***	***ML***[a]
1	31	-1588.8746	10.3%
2	21	-1544.3335	7.0%
3	26	-1625.2595	8.7%
4	31	-1664.3948	10.3%
5	34	-1756.3239	11.3%

[a]Marginal likelihood.

3.3 Intrusion Detection (TBD)

TBD

4 Conclusion (TBD)

TBD

Acknowledgment (TBD)

TBD

Appendix A

4.1 Derivation of Acceptance Ratio r by Eq. (12)

The derivation of acceptance ratio r is based on the assumption that mixture parameters are independent from each other which means that:

$$\begin{aligned}\pi(\Theta) &= \pi(p, \xi) = \pi(\xi) \\ &= \prod_{j=1}^{M} \pi(\mu_j)\pi(\sigma_{lj})\pi(\sigma_{rj}) \\ &= \prod_{j=1}^{M} N_d(\mu_j|\eta, \Sigma)N_d(\sigma_{lj}|\tau, \Sigma)N_d(\sigma_{rj}|\tau, \Sigma) \end{aligned} \tag{13}$$

in Eq. (14), since the mixture weigh p is generated following Gibbs sampling method which acceptance ratio is always 1, it should be excluded from Metropolis-Hastings estimation step. Accordingly, apply the same rule to the proposal distribution as well:

$$\begin{aligned} q(\Theta^{(t)}|\Theta^{(t-1)}) &= q(\xi^{(t)}|\xi^{(t-1)}) \\ &= \prod_{j=1}^{M} N_d(\mu_j^{(t)}|\mu_j^{(t-1)}, \Sigma) N_d(\sigma_{lj}^{(t)}|\sigma_{lj}^{(t-1)}, \Sigma) N_d(\sigma_{rj}^{(t)}|\sigma_{rj}^{(t-1)}, \Sigma) \end{aligned} \quad (14)$$

combines Eqs. (1) (4) (9) (10) (11) (14) and (15), equation (12) can be written as follows:

$$\begin{aligned} r &= \frac{P(\chi|\Theta^{(t)})\pi(\Theta^{(t)})q(\Theta^{(t-1)}|\Theta^{(t)})}{P(\chi|\Theta^{(t-1)})\pi(\Theta^{(t-1)})q(\Theta^{(t)}|\Theta^{(t-1)})} \\ &= \prod_{i=i}^{N}\prod_{j=1}^{M}\Big(\frac{P(X_i|\mu_j^{(t)}, \sigma_{lj}^{(t)}, \sigma_{rj}^{(t)})}{P(X_i|\mu_j^{(t-1)}, \sigma_{lj}^{(t-1)}, \sigma_{rj}^{(t-1)})} \\ &\times \frac{N_d(\mu_j^{(t)}|\eta, \Sigma) N_d(\sigma_{lj}^{(t)}|\tau, \Sigma) N_d(\sigma_{rj}^{(t)}|\tau, \Sigma)}{N_d(\mu_j^{(t-1)}|\eta, \Sigma) N_d(\sigma_{lj}^{(t-1)}|\tau, \Sigma) N_d(\sigma_{rj}^{(t-1)}|\tau, \Sigma)} \\ &\times \frac{N_d(\mu_j^{(t-1)}|\mu_j^{(t)}, \Sigma) N_d(\sigma_{lj}^{(t-1)}|\sigma_{lj}^{(t)}, \Sigma) N_d(\sigma_{rj}^{(t-1)}|\sigma_{rj}^{(t)}, \Sigma)}{N_d(\mu_j^{(t)}|\mu_j^{(t-1)}, \Sigma) N_d(\sigma_{lj}^{(t)}|\sigma_{lj}^{(t-1)}, \Sigma) N_d(\sigma_{rj}^{(t)}|\sigma_{rj}^{(t-1)}, \Sigma)} \end{aligned} \quad (15)$$

References

1. Elguebaly, T. and Bouguila, N. (2013). Background subtraction using finite mixtures of asymmetric Gaussian distributions and shadow detection. Machine Vision and Applications, 25(5), pp.1145-1162.
2. Wu, C. (1983). On the Convergence Properties of the EM Algorithm. The Annals of Statistics, 11(1), pp.95-103.
3. Bouguila, N., Ziou, D. and Hammoud, R. (2008). On Bayesian analysis of a finite generalized Dirichlet mixture via a Metropolis-within-Gibbs sampling. Pattern Analysis and Applications, 12(2), pp.151-166.
4. Hastings, W. (1970). Monte Carlo Sampling Methods Using Markov Chains and Their Applications. Biometrika, 57(1), p.97.
5. Geman, S. and Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-6(6), pp.721-741.
6. Bouguila, N., Ziou, D. and Monga, E. (2006). Practical Bayesian estimation of a finite beta mixture through gibbs sampling and its applications. Statistics and Computing, 16(2), pp.215-225.
7. Elguebaly, T. and Bouguila, N. (2011). Bayesian learning of finite generalized Gaussian mixture models on images. Signal Processing, 91(4), pp.801-820.
8. Marin, J.M. and Mengersen, K.R.C., 2005. Handbook of Statistics: Bayesian modelling and inference on mixtures of distributions, Vol. 25.

9. Luengo, D. and Martino, L., 2013, May. Fully adaptive gaussian mixture metropolis-hastings algorithm. In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on (pp. 6148-6152). IEEE.
10. Tavallaee, M., Bagheri, E., Lu, W. and Ghorbani, A.A., 2009, July. A detailed analysis of the KDD CUP 99 data set. In Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on (pp. 1-6). IEEE. Vancouver
11. Stephens, M., 2000. Bayesian analysis of mixture models with an unknown number of components-an alternative to reversible jump methods. Annals of statistics, pp.40-74.