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. In contrast with previous efforts that have been made on maximum likelihood estimation, Asymmetric Gaussian mixture (AGM) model has been proved more flexible than the classic mixture model from many aspects. This paper introduces a fully Bayesian learning method using Metropolis-Hastings within Gibbs sampling method to learn AGM model. We show the metrics of the model using synthetic dataset and a challenging intrusion detection applications.

Keywords - Asymmetric Gaussian Mixture, Metropolis-Hastings within Gibbs, MCMC, Intrusion detection

1 Introduction

Along with the development of the information industries, the network security problems are becoming more and more important today. In order to address this challenge, many data mining methodologies were proposed including both classification-based [1] and clustering-based [2] ones. However, classification-based solutions are always perform ineffectively for dynamic and variate attacking methods because changes of the intrusion patterns cannot be automatically adapted by the supervised learning algorithms. Consequently, unsupervised clustering-based approaches are more favorable for modern intrusion-detection system (IDS).

As one of the cluster analysis model, Gaussian mixtures model (GMM) [3] is widely deployed because of its outstanding suitability in several domains such as computer vision, real-time surveillance and intrusion detection etc,. At the meanwhile, GMM is based on an assumption that data observations are always following Gaussian distributions which is not always the case. To better adapt the non-Gaussian observations, the generalized Gaussian mixture (GGM) (Bouguila, 2008) [4] was introduced with one more parameter λ that controls the flatness of the mixture model which offering more flexibility for dataset fitting. In this paper, we choose asymmetric Gaussian mixture (AGM) model [5] for modeling because it uses two variance parameters for left and right parts of each distribution in the mixture which is able to accurately model non-Gaussian datasets including asymmetric ones.

Besides of the choose of AGM model, mixture parameter learning method is also performing a critical rule for clustering. The estimation of the parameters of

mixture distributions can be accomplished by introducing maximum-likelihood-based expectation maximization (EM) [6] algorithm. However, EM has some drawbacks such as estimation in higher dimensions, optimization [7] and overfitting problems [8] etc,. Therefore, an alternative is the fully Bayesian approach such as Markov chain Monte Carlo (MCMC) method which has been found to be useful in many applications without given priors of parameters which can avoids overfitting problems. As a sampling-based learning method, the main difficulty of MCMC method is that, under some circumstances, direct sampling is not always straightfoward. As widely deployed implementations of MCMC method, Metropolis-Hastings (Hastings, 1970) [9] and Gibbs sampling (Geman and Geman, 1984) [10] methods can be introduced to solve this problem by applying proposal priors and posteriors and sampling one parameter by giving the others. By combining the advantages of both methods together, the Metopolis-Hastings within Gibbs method [7] is selected as our learning algorithm for AGM model.

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 (network intrusion detection) will be tested against our AGM model and the results will be analyzed.

2 Bayesian Model

2.1 Asymmetric Gaussian Mixture Model

Assuming that the AGM model has M components then the likelihood function (Elguebaly and Bouguila, 2013) [5] is defined as follows:

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

where $\mathcal{X} = (X_1, ..., X_N)$ is the set of N observations, $\Theta = \{p_1, ..., p_M, \xi_1, ..., \xi_M\}$ represents the parameter set 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. Giving $X = (x_1, ..., x_d)$, the probability density function (Elguebaly and Bouguila, 2013) [5] can be defined as follows:

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

here $\xi_j = (\mu_j, \sigma_{lj}, \sigma_{rj})$ 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.

In order to simplify the Bayesian learning process, we introduce a M-dimensional membership vector Z. For each observation $X_i, 1 < i < N, Z_i = (Z_{i1}, ..., Z_{iM})$

which indicates to which specific component X_i belongs to (Bouguila, Ziou and Monga, 2006) [11], such that:

$$Z_{ij} = \begin{cases} 1 & \text{if } X_i \text{ belongs to component } j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

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

By combining the Eq. (1) and Eq. (3) together we derive the complete likelihood function:

$$p(\mathcal{X}, Z|\Theta) = \prod_{i=1}^{N} \prod_{j=1}^{M} (p_j p(X_i|\xi_j))^{Z_{ij}} \tag{4}$$

2.2 Learning Algorithm

Before describing MH-within-Gibbs learning steps, the priors and posteriors need to be clarified. Frist, we denote the postorior probability of membership vector Z as $\pi(Z|\Theta, \mathcal{X})$ (Elguebaly and Bouguila, 2011) [12]:

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

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 follows [13]:

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

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

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

Direct sampling of mixture parameters $\xi \sim p(\xi|Z, \mathcal{X})$ could be difficult so Metropolis-Hastings method should be deployed using 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:

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

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

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

the proposal distributions are d-dimensional Gaussian distributions with Σ as d x d identity matrix which makes 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. The acceptance ratio r is given by:

$$r = \frac{p(\mathcal{X}|\Theta^{(t)})\pi(\Theta^{(t)})q(\Theta^{(t-1)}|\Theta^{(t)})}{p(\mathcal{X}|\Theta^{(t-1)})\pi(\Theta^{(t-1)})q(\Theta^{(t)}|\Theta^{(t-1)})} \tag{12}$$

where $\pi(\Theta)$ is the proposed prior distribution which can be decomposed to d-dimensional Gaussian distributions such that $\mu \sim \mathcal{N}_d(\eta, \Sigma)$ and $\sigma_l, \sigma_r \sim \mathcal{N}_d(\tau, \Sigma)$ 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 $\alpha = min[1, r]$ [14]. 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 $\mathcal{X}$ and component number M
Output: AGM mixture parameter set Θ

1. Initialization
2. Step t: For $t = 1, \ldots$
   Gibbs sampling 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 [15] is selected for intrusion detection part.

3.2 Synthetic Data

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

Observation number is set to 300 spliting into two groups ($M = 2$). Hyper-parameters are set accordingly that $\gamma_j = 1$ [16] for sampling mixture weight p_j from Eq. (8). η and τ are d-dimensional zero vectors in prior distributions of mixture parameter set ξ.

Different proposed component numbers ($M' = 1, \ldots, 5$) are applied during the AGM learning process and the statistics are summarized in Table 1. In order to distinguish the best value of the proposed component number M', we introduce marginal likelihood [7] as a measurement that the best fit M' should have the maximum marginal likelihood value. Obviously, the best result is derived when proposed component number M' equals to original component number M ($M' = M = 2$) which is shown as Fig. 1. The probability density diagrams is plotted for both original and estimated AGM components and the polylines show the trace of accepted moves for each component.

Then we focus on the best fit result , the accuracy is evaluated by calculating the Euclidean distance between original and estimated mixture parameter sets ξ and $\hat{\xi}$ (Table 2). In summary, the estimation of mean is accurate because the Euclidean distance between μ_j and $\hat{\mu}_j$ are small but the distance between standard deviation σ_{lj}, σ_{rj} and $\hat{\sigma}_{lj}, \hat{\sigma}_{rj}$ are significant. Since membership vector Z is involved for hard-clustering so this difference will not affect the clustering result too much.

Table 1. AGM Learning Statistics ($M = 2,\ M' = 1, \ldots, 5,\ iterations = 300$)

Component Number M'	***Moves Accepted***	***Acceptance Ratio***	***ML***[a]
1	22	7.33%	-1596.143
2	11	3.67%	-1500.370
3	14	4.67%	-1684.518
4	63	21.00%	-1522.148
5	39	13.00%	-1517.533

[a]Marginal likelihood.

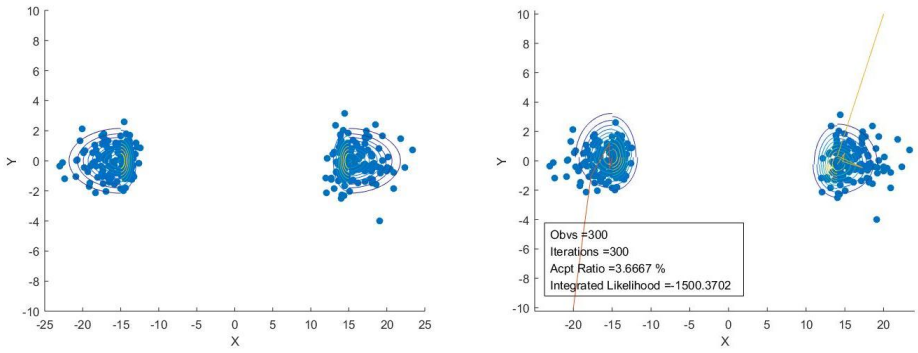

Fig. 1. Original synthetic observations and learning result ($M' = M = 2$)

Table 2. Accuracy Analysis ($M' = M = 2$)

Component Number $j = 1$	**Mean (μ_j)**	**Left Standard deviation (σ_{lj})**	**Right Standard deviation (σ_{rj})**
ξ	[-15.00, 0.00]	[10.00, 1.00]	[1.00, 1.00]
$\hat{\xi}$	[-14.99, 0.25]	[4.77, 1.13]	[2.31, 1.88]
Euclidean Distance	0.246	5.236	1.581
Component Number $j = 2$	**Mean (μ_j)**	**Left Standard deviation (σ_{lj})**	**Right Standard deviation (σ_{rj})**
ξ	[15.00, 0.00]	[1.00, 1.00]	[10.00, 1.00]
$\hat{\xi}$	[14.02, -0.24]	[2.04, 1.04]	[5.70, 1.59]
Euclidean Distance	1.010	1.036	4.338

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:

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

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:

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

by combining Eqs. (2) (4) (9) (10) (11) (14) and (15), equation (12) can be written as follows:

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

References

1. Puttini, R.S., Marrakchi, Z. and M, L., 2003, March. A Bayesian Classification Model for RealTime Intrusion Detection. In AIP Conference Proceedings (Vol. 659, No. 1, pp. 150-162). AIP.

2. Zhong, S., Khoshgoftaar, T.M. and Seliya, N., 2007. Clustering-based network intrusion detection. International Journal of reliability, Quality and safety Engineering, 14(02), pp.169-187.
3. Rasmussen, C.E., 2000. The infinite Gaussian mixture model. In Advances in neural information processing systems (pp. 554-560).
4. Bouguila, N. (2008). Finite general Gaussian mixture modeling and application to image and video foreground segmentation. Journal of Electronic Imaging, 17(1), p.013005.
5. 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.
6. Dempster, A.P., Laird, N.M. and Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the royal statistical society. Series B (methodological), pp.1-38.
7. 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.
8. Bouguila, N. and Elguebaly, T. (2012). A fully Bayesian model based on reversible jump MCMC and finite Beta mixtures for clustering. Expert Systems with Applications, 39(5), pp.5946-5959.
9. Hastings, W. (1970). Monte Carlo Sampling Methods Using Markov Chains and Their Applications. Biometrika, 57(1), p.97.
10. 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.
11. 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.
12. Elguebaly, T. and Bouguila, N. (2011). Bayesian learning of finite generalized Gaussian mixture models on images. Signal Processing, 91(4), pp.801-820.
13. Marin, J.M. and Mengersen, K.R.C., 2005. Handbook of Statistics: Bayesian modelling and inference on mixtures of distributions, Vol. 25.
14. 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.
15. 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
16. 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.