Bayesian Learning of Asymmetric Gaussian-Based Statistical Models using Markov Chain Monte Carlo Techniques

Shuai Fu

A Thesis

in

The Concordia Institute

for

Information Systems Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Applied Science (Information Systems Security) at

Concordia University

Montréal, Québec, Canada

July 2018

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Shuai Fu**

Entitled: **Bayesian Learning of Asymmetric Gaussian-Based Statistical Models using Markov Chain Monte Carlo Techniques**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Information Systems Security)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

______________________ Chair
Dr. Amr Youssef

______________________ External Examiner
Dr. Joonhee Lee

______________________ Examiner
Dr. Jamal Bentahar

______________________ Supervisor
Dr. Nizar Bouguila

Approved by ______________________
Amr Youssef, Chair
Department of Information Systems Engineering

__________ 2018

____
Amir Asif, Dean
Faculty of Engineering and Computer Science

Abstract

Bayesian Learning of Asymmetric Gaussian-Based Statistical Models using Markov Chain Monte Carlo Techniques

Shuai Fu

A novel unsupervised Bayesian learning framework based on asymmetric Gaussian mixture (AGM) statistical model is proposed since AGM is shown to be more effective compared to the classic Gaussian mixture. The Bayesian learning framework is developed by adopting sampling-based Markov chain Monte Carlo (MCMC) methodology. More precisely, the fundamental learning algorithm is a hybrid Metropolis-Hastings within Gibbs sampling solution which is integrated within a reversible jump MCMC (RJMCMC) learning framework, a self-adapted sampling-based MCMC implementation, that enables model transfer throughout the mixture parameters learning process, therefore, automatically converges to the optimal number of data groups. Furthermore, a feature selection technique is included to tackle the irrelevant and unneeded information from datasets. The performance comparison between AGM and other popular solutions is given and both synthetic and real data sets extracted from challenging applications such as intrusion detection, spam filtering and image categorization are evaluated to show the merits of the proposed approach.

Acknowledgments

I would like to firstly express my gratitude to my supervisor Dr. Nizar Bouguila not only for his important academic guidance and suggestions throughout the study of my Master program but also for his personal concerns and continuous supports to me and my family. I could not image that I could graduate from the program without his endless patience, warm encourage and financial supports. What I am learning from him will definitely influences my entire life.

I also want to extend my gratefulness to all the professors and colleagues of Concordia Institute for Information Systems Engineering to providing this great opportunity to learning with the most talented supervisors and students in Concordia University.

Finally, I would like to thank my parents and my wife for their persistent and unconditional love and support, and my daughter, for the happiness she brought to my family.

Thank you all!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Over past decades, many statistical data mining approaches have been proposed to address challenging data modeling analysis problems given the fact that the volume of data is dramatically increasing due to the usage of Internet. Meanwhile, modern machine-learning-based techniques perform in both generative and discriminative ways which can be divided into two main streams, classification-based supervised and clustering-based unsupervised ones. Compared to supervised solutions, unsupervised approach has no assumption on the number of groups, therefore, friendly to newly added data and patterns which makes it more suitable for increasing database analysis. Moreover, it also immunizes against learning biases and overfitting problems that commonly exist in most supervised approaches if model training is inappropriate. Consequently, there has been an increasing trend of applying finite mixtures into different domains involving statistical modeling of data, such as astronomy, ecology, bioinformatics, pattern recognition, computer vision and machine learning [1]. Our work is based on asymmetric Gaussian mixture (AGM) model [2] and reversible jump Markov chain Monte Carlo (RJMCMC) learning algorithm [3]. Previous efforts reveal the fact that AGM outperforms classic Gaussian mixture model (GMM) by taking asymmetric datasets into consideration which provides more flexibility [4]. Our RJMCMC implementation is based on a hybrid sampling-based approach which takes advantages of both Metropolis-Hastings (MH) and Gibbs sampling methods [5], therefore, simplifies mathematical complexity and extends adaptability of the model. Moreover, without giving a fixed components number in advance, RJMCMC applies a dynamic data-based strategy to identify the optimal components number throughout iterations which makes the model learning a self-adaptive process. To achieve better fitting outcomes, feature selection process is involved to handle high-dimensional vectors of features and the analysis and discussion of deploying AGM to both synthetic datasets and real applications is given in the later chapters.

1.1.1 Finite Mixture Models

As upgrade of single-mathematical-model-based methodologies, mixture models [6, 7, 8] can be seen as a superimposition of certain mixture components sharing dependencies with each other, therefore, lead to outstanding performance especially for high-dimensional and multi-cluster datasets. Finite mixture models can be described by

$$p(X|\Theta) = \sum_{j=1}^{M} p_j p(X|\Theta_j) \tag{1}$$

where X reprensents a vector in a given dataset and Θ defines the mixture parameters set (for each mixture compoent, the sub-parameter set is described by $\Theta_j, j = 1, \dots, M$) as well as component weight p_j ($0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j$ = 1).

1.1.2 Probability Density Function Selection

Probability density function (PDF) selection has an important role in finite mixture model because it significantly affects the capability of representing the data. Improper PDF selection will cause incorrect outcomes such as wrong components number and poor data fitting. Gaussian mixture model (GMM) [3] demonstrated satisfactory fitting abilities on most real applications whose datasets are Gaussian-like. However, under more general circumstances regarding to non-Gaussian or asymmetric datasets, asymmetric Gaussian mixture (AGM) model [2] leads to a better accuracy by introducing two variance parameters for both left and right parts of asymmetric Gaussian distribution, providing more flexibility for variant real applications. Therefore, the justification of choosing AGM model and its merits will be discussed in the following chapters.

1.1.3 Bayesian Learning Framework

Estimating the parameters of mixture models could be a challenging task. The maximum-likelihood-based expectation maximization (EM) [9] algorithm is one of the most popular parameter learning approaches. However, the disadvantages of EM algorithm are also obvious. Given the fact that EM approximates values of mixture parameters in a deterministic way this could cause slow convergence and compromise the usability of the algorithm. Furthermore, bad initialization and overfitting problems [10, 11] will also significantly affect its accuracy. Therefore, fully Bayesian learning algorithms, such as Markov Chain Monte Carlo (MCMC) based implementations, are found to be useful to eliminate overfitting problems in mixture parameter learning by introducing prior and posterior distributions for mixture parameters. In our work, the learning process is accomplished by a hybrid MCMC algorithm, which is well known as Metropolis-Hastings within Gibbs sampling [10, 12], based on both Metropolis-Hastings [13] and Gibbs sampling [14] methods because the main difficulty of classic MCMC method is that, under some circumstances, direct sampling is not always straightforward. Moreover, we reinforce the learning algorithm by introducing reversible jump MCMC (RJMCMC) [3] methodology to increase the flexibility of AGM model by allowing model transfer throughout iterations via increasing (component birth/split step) and decreasing (component death/merge step) mixture components. Because of the stochastic sampling-based learning process, learning iterations could end up with different number of components so we choose marginal likelihood [10] to perform model selection in order to evaluate fitting results between models.

1.1.4 Dimensionality Reduction

One of the most important tasks in data mining, pattern recognition, computer vision and machine learning applications is that, the existence of outliers and irrelevant features severely compromises the clustering outcomes. Therefore, many dimensionality reduction methodologies have been proposed [15, 16] such as feature extraction and selection which try to remove these unneeded features in order to improve the performance of the modeling [17, 18] while feature extraction is based on transformations or combinations of the original features [19]. Indeed, feature selection methods identify relevant features in the original representation space. Recently, a volume of literature

[20, 21] has shown that selecting relevant features leads to more accurate modeling results. However, this problem is not trivial especially in the unsupervised context dealing with labelless data sets. For this reason, previous researches [22, 4, 23] were devoted to extend unsupervised feature selection to mixture-based clustering. In this thesis, we extended the RJMCMC-based simultaneous Bayesian clustering and feature selection approach proposed in [4] to asymmetric Gaussian mixture model in order to improve the modeling performance on a challenging image categorization application.

1.2 Contributions

The contributions of this thesis are as follows:

☞ **A Novel Bayesian Framework for Asymmetric Gaussian Mixture via Markov Chain Monte Carlo Method:** We chose an advanced MCMC implementation called reversible jump MCMC (RJMCMC) [11] which is based on a hybrid Metropolis-Hastings within Gibbs sampling [10] solution, combining both Metropolis-Hastings [13] and Gibbs sampling [14] methods because the main difficulty of applying traditional MCMC method is that, under some circumstances, direct sampling is not always straightforward that distributions of mixture parameters are latent and dependencies between parameters are unknown. By integrating the merits of both methods, mixture parameters will be evaluated iteratively and, eventually, the optimal parameter values will be identified after convergence. Furthermore the self-adapted learning process [11] treats components number as an extra parameter and adjusts it throughout iterations by automatically increasing (component birth/death step) and decreasing (component merge/split step) according to current status, therefore, enables model transfer which significantly improves the learning performance. This contribution has been published in [24].

☞ **Intrusion Detection and Spam Filtering by Applying Proposed Approach via Reversible Jump MCMC:** We apply and adapt the proposed Bayesian learning framework to two challenging applications namely intrusion detection and spam filtering ([25] and [26]).

☞ **Feature Selection for Image Categorization:** While deploying the proposed approach for image categorization, in order to better identify visual features from the challenging UIUC sport events dataset, the image representative data is generated by adopting scale-invariant feature transform (SIFT), bag-of-visual-words (BOVW) and probabilistic latent semantic analysis (pLSA) techniques. However, previous approaches assume all the features of observations have the same weight of importance and carry pertinent information which is not always the case and many of those features can be irrelevant for clustering purpose. In order to tackle this problem and define relevance and importance of features, feature selection techniques [4, 23] should be taken into consideration. Eventually, irrelevant and unneeded information will be filtered by feature selection.

1.3 Thesis Overview

The rest of this thesis is organized as follows:

- ❏ Chapter 2 introduces the Asymmetric Gaussian mixture model and its sampling based Bayesian learning framework. In particular, a self-adapted reversible jump MCMC implementation which has no assumption concerning the number of components and, therefore, the AGM model itself could be transferred between iterations. Furthermore the self-adapted learning process treats components number as an extra parameter and adjusts it throughout iterations by automatically increasing (component birth/death step) and decreasing (component merge/split step) according to current status, therefore, enables model transfer which significantly improves the learning performance.

- ❏ Chapter 3 is devoted to feature selection since the AGM model assumes that all the features of observations have the same weight of importance and carry pertinent information which is not always the case and many of those features can be irrelevant for clustering purpose. In order to tackle this problem and define relevance and importance of features, feature selection techniques should be taken into consideration. A challenging UIUC sports event database is selected for validation of the proposed approach.

- ❏ Chapter 4 concludes and summarizes the thesis and points out future research directions.

Chapter 2

Asymmetric Gaussian Mixtures with Reversible Jump MCMC and Applications

This chapter presents a novel intrusion detection classifier based on asymmetric Gaussian mixture (AGM) model and reversible jump Markov chain Monte Carlo (RJMCMC) learning algorithm. Previous efforts reveal the fact that AGM outperforms classic Gaussian mixture model (GMM) by taking asymmetric datasets into consideration which provides more flexibility. Our RJMCMC implementation is based on a hybrid sampling-based approach which takes advantages of both Metropolis-Hastings (MH) and Gibbs sampling methods, therefore, simplifies mathematical complexity and extends adaptability of the model. Moreover, without giving a fixed components number in advance, RJMCMC applies a dynamic data-based strategy to identify the optimal components number throughout iterations which makes the model learning a self-adaptive process. Since the model is nondeterministic, Laplace approximation based marginal likelihood is calculated for multiple runs as model selection procedure to improve the correctness and fitting accuracy. Both synthetic and real datasets are applied to our model to discover its merits and the test results will be evaluated and compared with other popular solutions.

2.1 Asymmetric Gaussian Mixture Model

The likelihood function of AGM model [2] with M mixture components can be illustrated as follows:

$$p(\mathcal{X}|\Theta) = \prod_{i=1}^{N} \sum_{j=1}^{M} p_j p(X_i|\xi_j) \tag{2}$$

where $\mathcal{X} = (X_1, ..., X_N)$ reprensents the dataset with N observations, $\Theta = \{p_1, ..., p_M, \xi_1, ..., \xi_M\}$ defines the mixture parameters set of AGM mixture model including component weight p_j (0 $<$ p_j ≤ 1 and $\sum_{j=1}^{M} p_j$ = 1) and asymmetric Gaussian distribution (AGD) parameters set ξ_j for mixture component j. Assuming the dataset $\mathcal{X}$ is d-dimensional, for each observation $X_n = (x_{n1}, ..., x_{nd}) \in \mathcal{X}$, the probability density function [2] for j-th component of the model can be defined as follows:

$$p(X|\xi_j) \propto \prod_{k=1}^{d} \frac{1}{(\sigma_{l_{jk}} + \sigma_{r_{jk}})} \times \begin{cases} \exp\left[-\frac{(x_k-\mu_{jk})^2}{2(\sigma_{l_{jk}})^2}\right] & if\ x_k\ <\ \mu_{jk} \\ \exp\left[-\frac{(x_k-\mu_{jk})^2}{2(\sigma_{r_{jk}})^2}\right] & if\ x_k\ \geqslant\ \mu_{jk} \end{cases} \tag{3}$$

parameters set of component j is $\xi_j = (\mu_j, \sigma_{lj}, \sigma_{rj})$ where $\mu_j = (\mu_{j1}, ..., \mu_{jd})$ is the mean, $\sigma_{lj} = (\sigma_{lj1}, ..., \sigma_{ljd})$ and $\sigma_{rj} = (\sigma_{rj1}, ..., \sigma_{rjd})$ represents the left and right standard deviation vectors of AGD .

We bring a M-dimensional membership vector Z to each observation $X_i \in \mathcal{X}$, $Z_i = (Z_{i1}, ..., Z_{iM})$, indicating which specific component X_i belongs to [1], such that:

$$Z_{ij} = \begin{cases} 1 & \text{if } X_i \text{ belongs to component } j \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

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

Hence, the complete likelihood function can be obtained by combining Eq. (2) and Eq. (4) as follows:

$$p(\mathcal{X}, Z|\Theta) = \prod_{i=1}^{N}\prod_{j=1}^{M}(p_j p(X_i|\xi_j))^{Z_{ij}} \tag{5}$$

2.2 Bayesian Learning Algorithm

Before describing MH-within-Gibbs learning steps, the priors and posteriors need to be specified. First, we denote the postorior probability of membership vector Z as $\pi(Z|\Theta, \mathcal{X})$ [5]:

$$Z^{(t)} \sim \pi(Z|\Theta^{(t-1)}, \mathcal{X}) \tag{6}$$

the number of observations belonging to a specific component j can be calculated using $Z^{(t)}$ as follows:

$$n_j^{(t)} = \sum_{i=1}^{N} Z_{ij}\ (j = 1, ..., M) \tag{7}$$

thus $n^{(t)} = (n_1^{(t)}, ..., n_M^{(t)})$ represents the number of observations belonging to each mixture component.

Since the mixture weight p_j satisfies the following conditions ($0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1$), a natural choice of the prior is Dirichlet distribution as follows [27, 28]

$$\pi(p_1, \ldots, p_M) \sim \mathcal{D}(\gamma_1, ..., \gamma_M) \tag{8}$$

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

$$p(p_1, \ldots, p_M|Z^{(t)}) \sim \mathcal{D}(\gamma_1 + n_1^{(t)}, ..., \gamma_M + n_M^{(t)}) \tag{9}$$

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

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

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

$$\sigma_{rj}^{(t)} \sim \mathcal{N}_d(\sigma_{rj}^{(t-1)}, \Sigma) \tag{12}$$

the proposal distributions are d-dimensional Gaussian distributions with Σ as d x d identity matrix which makes the sampling a 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:

$$r = \frac{p(\mathcal{X}|\Theta^{(t)})\pi(\Theta^{(t)})q(\Theta^{(t-1)}|\Theta^{(t)})}{p(\mathcal{X}|\Theta^{(t-1)})\pi(\Theta^{(t-1)})q(\Theta^{(t)}|\Theta^{(t-1)})} \tag{13}$$

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

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

in Eq. (14), since the mixture weigh p is generated following Gibbs sampling method whose 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:

$$\begin{aligned} q(\Theta^{(t)}|\Theta^{(t-1)}) &= q(\xi^{(t)}|\xi^{(t-1)}) \\ &= \prod_{j=1}^{M} \mathcal{N}_d(\mu_j^{(t)}|\mu_j^{(t-1)}, \Sigma)\mathcal{N}_d(\sigma_{lj}^{(t)}|\sigma_{lj}^{(t-1)}, \Sigma)\mathcal{N}_d(\sigma_{rj}^{(t)}|\sigma_{rj}^{(t-1)}, \Sigma) \end{aligned} \tag{15}$$

by combining Eqs. (3) (5) (10) (11) (12) (14) and (15), equation (13) can be written as follows:

$$\begin{aligned} r &= \frac{p(\mathcal{X}|\Theta^{(t)})\pi(\Theta^{(t)})q(\Theta^{(t-1)}|\Theta^{(t)})}{p(\mathcal{X}|\Theta^{(t-1)})\pi(\Theta^{(t-1)})q(\Theta^{(t)}|\Theta^{(t-1)})} \\ &= \prod_{i=i}^{N}\prod_{j=1}^{M}(\frac{p(X_i|\mu_j^{(t)}, \sigma_{lj}^{(t)}, \sigma_{rj}^{(t)})}{p(X_i|\mu_j^{(t-1)}, \sigma_{lj}^{(t-1)}, \sigma_{rj}^{(t-1)})} \\ &\quad \times \frac{\mathcal{N}_d(\mu_j^{(t)}|\eta, \Sigma)\mathcal{N}_d(\sigma_{lj}^{(t)}|\tau, \Sigma)\mathcal{N}_d(\sigma_{rj}^{(t)}|\tau, \Sigma)}{\mathcal{N}_d(\mu_j^{(t-1)}|\eta, \Sigma)\mathcal{N}_d(\sigma_{lj}^{(t-1)}|\tau, \Sigma)\mathcal{N}_d(\sigma_{rj}^{(t-1)}|\tau, \Sigma)} \\ &\quad \times \frac{\mathcal{N}_d(\mu_j^{(t-1)}|\mu_j^{(t)}, \Sigma)\mathcal{N}_d(\sigma_{lj}^{(t-1)}|\sigma_{lj}^{(t)}, \Sigma)\mathcal{N}_d(\sigma_{rj}^{(t-1)}|\sigma_{rj}^{(t)}, \Sigma)}{\mathcal{N}_d(\mu_j^{(t)}|\mu_j^{(t-1)}, \Sigma)\mathcal{N}_d(\sigma_{lj}^{(t)}|\sigma_{lj}^{(t-1)}, \Sigma)\mathcal{N}_d(\sigma_{rj}^{(t)}|\sigma_{rj}^{(t-1)}, \Sigma)} \end{aligned} \tag{16}$$

Once acceptance ratio r is derived by Eq. (16), we compute acceptance probability $\alpha = min[1, r]$ [29]. 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, we 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 in the following steps:

Input: Data observations $\mathcal{X}$ and components 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. (6)

(b) Compute $n_j^{(t)}$ from Eq. (7)

(c) Generate $p_j^{(t)}$ from Eq. (9)

Metropolis-Hastings part

(d) Sample $\xi_j^{(t)}$ $(\mu_j^{(t)}, \sigma_{lj}^{(t)}, \sigma_{rj}^{(t)})$ from Eqs. (10) (11) (12)

(e) Compute acceptance ratio r from Eq. (13)

(f) Generate $\alpha = min[1, r]$ and $u \sim U_{[0,1]}$

(g) If $u \geq \alpha$ then $\xi^{(t)} = \xi^{(t-1)}$

2.3 Reversible Jump Markov Chain Monte Carlo

We reinforce the learning algorithm by introducing reversible jump MCMC (RJMCMC) [11] methodology to increase the flexibility of AGM model because traditional MH-within-Gibbs algorithm assumes that the component number M is given and persistent throughout the learning process. However, because of bad initialization or just information leakage, M could be inaccurate or unknown. Under these circumstances, RJMCMC algorithm presents its merits by providing extra four independent steps (birth/death steps and merge/split steps) into learning process which could change component number M, therefore, brings more generalities.

In practice, within every RJMCMC learning iteration, the current component number m is considered as an extra parameter which has a proposed Poisson prior $\mathcal{P}(\lambda)$ with $\lambda = 4$ particularly in our case [3]. Accordingly, let M_{min} and M_{max} denote the minimum and maximum number of components M, and assume the probabilities of performing birth/split and death/merge steps are b_m and $d_m = 1 - b_m$ for $m = M_{min}, \ldots, M_{max}$ respectively. Obviously, $b_{M_{max}} = 0$ and $d_{M_{min}} = 0$. Correspondingly, $d_{M_{max}} = 1 - b_{M_{max}} = 1$ and $b_{M_{min}} = 1 - d_{M_{min}} = 1$. For $m = M_{min} + 1, \ldots, M_{max} - 1$, for simplification purpose, we choose the same value for both b_m and d_m as $b_m = d_m = 0.5$. Within every iteration, we generate a random value $u' \sim U_{[0,1]}$ respectively for the four RJMCMC steps. If $b_m >= u'$ or $d_m >= u'$, birth/split or death/merge steps should be performed correspondingly [3].

Merge and Split Steps: Randomly choose two components (j_1, j_2) satisfying that $\mu_{j_1} < \mu_{j_2}$ with no other μ_j in the interval $[\mu_{j_1}, \mu_{j_2}]$. The newly merged component j' will contain the observations that previously belonged to both component j_1 and j_2. Meanwhile, reduce current value of component number m to $m-1$, then calculate mixture weight and parameters for j' as follows:

$$\begin{aligned}
p_{j'} &= p_{j_1} + p_{j_2} \\
p_{j'}\mu_{j'} &= p_{j_1}\mu_{j_1} + p_{j_2}\mu_{j_2} \\
p_{j'}(\mu^2_{j'} + \sigma^2_{j'l}) &= p_{j_1}(\mu^2_{j_1} + \sigma^2_{j_1 l}) + p_{j_1}(\mu^2_{j_1} + \sigma^2_{j_1 l}) \\
p_{j'}(\mu^2_{j'} + \sigma^2_{j'r}) &= p_{j_1}(\mu^2_{j_1} + \sigma^2_{j_1 r}) + p_{j_1}(\mu^2_{j_1} + \sigma^2_{j_1 r})
\end{aligned} \tag{17}$$

As a reverse of merge step, we split component j' into two (j_1 and j_2) with 3 degrees of freedom $(u_1 \sim Beta(2,2), u_2 \sim Beta(2,2), u_3 \sim Beta(1,1))$ and, accordingly, increase m to $m+1$. Therefore, mixture parameters for split components can be calculated as follows:

$$\begin{aligned}
p_{j_1} &= p_{j'}u_1, p_{j_2} = p_{j'}u_2 \\
\mu_{j_1} &= \mu_{j'} - \frac{u_2(\sigma_{j'l} + \sigma_{j'r})}{2}\sqrt{\frac{p_{j_2}}{p_{j_1}}} \\
\mu_{j_2} &= \mu_{j'} + \frac{u_2(\sigma_{j'l} + \sigma_{j'r})}{2}\sqrt{\frac{p_{j_1}}{p_{j_2}}} \\
\sigma^2_{j_1 l} &= u_3(1 - u_2^2)\sigma^2_{j'l}\frac{p_{j'}}{p_{j_1}} \\
\sigma^2_{j_1 r} &= u_3(1 - u_2^2)\sigma^2_{j'r}\frac{p_{j'}}{p_{j_1}} \\
\sigma^2_{j_2 l} &= (1 - u_3)(1 - u_2^2)\sigma^2_{j'l}\frac{p_{j'}}{p_{j_2}} \\
\sigma^2_{j_2 r} &= (1 - u_3)(1 - u_2^2)\sigma^2_{j'r}\frac{p_{j'}}{p_{j_2}}
\end{aligned} \tag{18}$$

In order to decide whether the merge and split steps should be accepted or not, the acceptance probability [3] can be derived as follows:

$$\begin{aligned}\mathcal{A} = {} & \frac{p(\mathcal{X}, Z|\Theta')}{p(\mathcal{X}, Z|\Theta)} \frac{m'\mathcal{P}(m'|\lambda)}{\mathcal{P}(m|\lambda)} \frac{p_{j_1}^{\gamma-1+n_1} p_{j_2}^{\gamma-1+n_2}}{p_{j'}^{\gamma-1+n_1+n_2} Beta(\gamma, m\gamma)} \\ & \times \sqrt{\frac{\kappa}{2\pi}} \exp[-\frac{1}{2}\kappa(\mu_{j_1} - \xi) + (\mu_{j_2} - \xi) + (\mu_{j'} - \xi)] \\ & \times \frac{\beta^\alpha}{\Gamma(\alpha)} (\frac{\sigma^2_{j_1 l}\sigma^2_{j_1 r}\sigma^2_{j_2 l}\sigma^2_{j_2 r}}{\sigma^2_{j' l}\sigma^2_{j' r}})^{-\alpha-1} \\ & \times \exp[-\beta(\sigma^2_{j_1 l} + \sigma^2_{j_1 r} + \sigma^2_{j_2 l} + \sigma^2_{j_2 r} - \sigma^2_{j' l} - \sigma^2_{j' r})] \\ & \times \frac{d_{m'}}{b_m P_{alloc}} [Beta(\mu_1|2,2) Beta(\mu_2|2,2) Beta(\mu_3|1,1)]^{-1} \\ & \times \frac{p_{j'}|\mu_{j_1} - \mu_{j_2}|\sigma^2_{j_1 l}\sigma^2_{j_1 r}\sigma^2_{j_2 l}\sigma^2_{j_2 r}}{\mu_2(1-\mu_2^2)\mu_3(1-\mu_3)\sigma^2_{j' l}\sigma^2_{j' r}} \end{aligned} \tag{19}$$

where Θ' and $m' = m + 1$ denote the mixture parameters set and the component number respectively before merge or after split steps. κ is a known hyperparameter and ξ is the midpoint of the variation interval of the involved data observations. Besides, P_{alloc} is the probability of which this particular allocation is made. Therefore, the acceptance probability for merge step is $\min(1, \mathcal{A})$ and, correspondingly, for split step is $\min(1, \mathcal{A}^{-1})$.

Birth and Death Steps: Compared to merge and split steps, birth and death steps are relatively straightforward because the newborn and dead components are empty ones which means parameter re-calculation is not needed. Mixture weight p_{new} in birth step can be obtained by sampling from Beta distribution $p_{new} \sim Beta(1, m)$ and mixture parameters can be derived from the priors as follows [30]:

$$\mu \sim \mathcal{N}(\xi, \kappa^{-1}), \quad \sigma_l^{-2}, \sigma_r^{-2} \sim \Gamma(\alpha, \beta), \quad \beta \sim \Gamma(g, h) \tag{20}$$

where hyperparameters κ, α, g and h are chosen according to the data. For death step, an empty component should be randomly selected and deleted among the existing components if there is any. Otherwise, this step will be skipped. After birth and death steps, mixture weights p_j should be rescaled so that all weights sum to 1. Acceptance probability for birth and death steps is also required as the one for merge and split steps whose definition is as follows:

$$\mathcal{A}' = \frac{\mathcal{P}(m'|\lambda)}{\mathcal{P}(m|\lambda)} \frac{1}{Beta(m\gamma,\gamma)} p_{j'}^{\gamma-1}(1-p_{j'})^{N+m\gamma-m} m' \frac{d_{m'}}{(m_0+1)b_m} \frac{1}{Beta(p_{j'}|1,m)}(1-p_{j'})^{m} \tag{21}$$

where m_0 is the amount of empty components. Thus, the probabilities of occurrence of birth and death steps are $\min(1,\mathcal{A}')$ and $\min(1,\mathcal{A}'^{-1})$ [3].

Finally, Figure 2.1 describes the dependencies between constants and variables involved in the Bayesian network of RJMCMC mixture parameter learning, and then, a typical learning procedure of AGM can be summarized as follows:

```
Input: Data observations $\mathcal{X}$ and component number $M$
Output: AGM mixture parameter set $\Theta$
  (1) Initialization
  (2) Step $t$: For $t = 1, \ldots$
        Gibbs sampling part
        (a) Generate $Z^{(t)}$ from Eq. (4)
        (b) Compute $n_j^{(t)}$ from Eq. (7)
        (c) Generate $p_j^{(t)}$ from Eq. (9)
        Metropolis-Hastings part
        (d) Sample $\xi_j^{(t)}$ $(\mu_j^{(t)}, \sigma_{lj}^{(t)}, \sigma_{rj}^{(t)})$ from Eqs. (10) (11) (12)
        (e) Compute acceptance ratio $r$ from Eq. (13)
        (f) Generate $\alpha = min[1, r]$ and $u \sim U_{[0,1]}$
        (g) If $u \geq \alpha$ then $\xi^{(t)} = \xi^{(t-1)}$
        RJMCMC part
        (h) Generate $u' \sim U_{[0,1]}$. If $b_m >= u'$, perform split or birth step, then calculate acceptance
            probability $\mathcal{A}$. If the step is accepted, set $m = m + 1$.
        (i) Generate $u' \sim U_{[0,1]}$. If $d_m >= u'$, perform merge or death step, then calculate acceptance
            probability $\mathcal{A}'$. If the step is accepted, set $m = m - 1$.
```

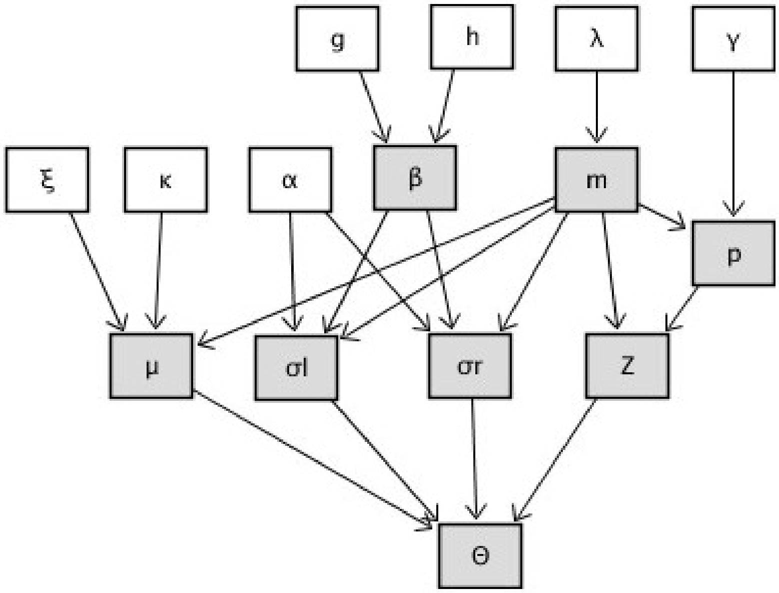


Figure 2.1: DAG of RJMCMC parameter learning Bayesian network

2.3.1 Model Selection

Theoretically, RJMCMC learning process should always be able to derive the optimal components number M. However, because of the stochastic sampling, improper proposal distributions or bad initialization parameters, learning result based on a single estimation run is not always satisfactory. In order to establish a robust parameter estimation algorithm, we evaluate the estimation outputs derived from multiple RJMCMC runs with different initial values of components number by calculating their marginal likelihood with the Laplace approximation [10] on the logarithm scale which is defined as follows:

$$\log(p(\mathcal{X}|M)) \;=\; \log(p(\mathcal{X}|\hat{\Theta}, M)) + \log(\pi(\hat{\Theta}|M)) + \frac{N_p}{2}\log(2\pi) + \frac{1}{2}\log(|H(\hat{\Theta})|) \quad (22)$$

where $\hat{\Theta}$ denotes the proposed optimal parameter set derived from a specific learning process and $\pi(\hat{\Theta}|M)$ is the prior density of mixture parameters as well as its Hessian matrix $H(\hat{\Theta})$ which is asymptotically equal to the posterior covariance matrix.

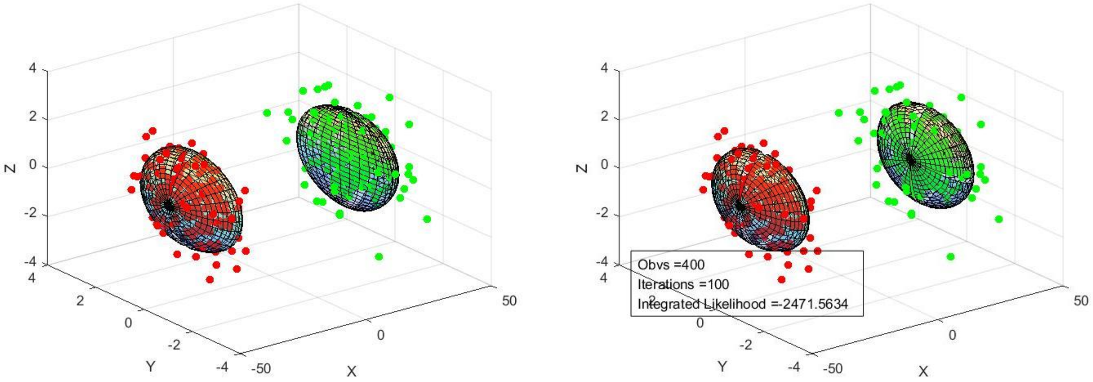


Figure 2.2: Original synthetic data grouping and learning results

2.4 Experimental Results

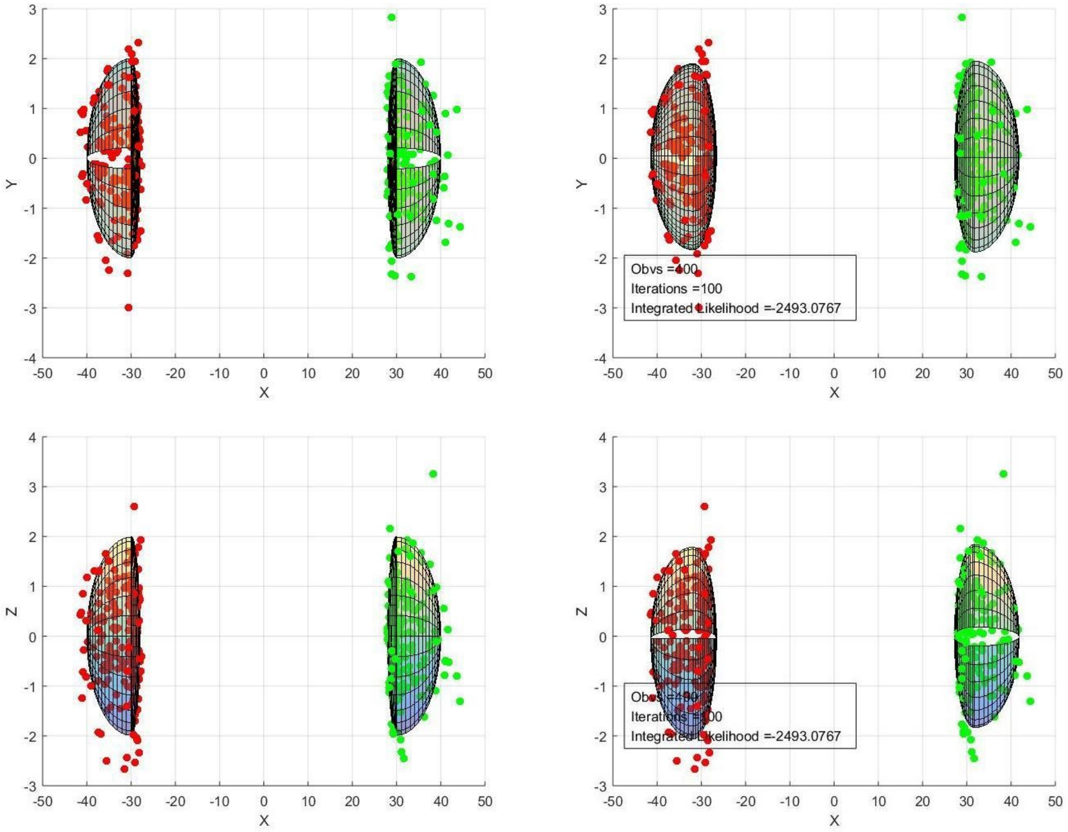
2.4.1 Design of Experiments

Firstly, we apply the AGM model to both synthetic data and intrusion detection. For synthetic data validation, testing observations will be generated from AGM with known components number M and experimental results will be evaluated by comparing the estimated and actual mixture parameters. In intrusion detection application, we select NSL-KDD dataset [31] as testing database. K-means algorithm is used for initialization and the results analysis will be based on statistics derived from confusion matrix. Then, the proposed approach will be deployed to the Spambase spam filtering database contains multiple spam textual features including spam word/character dictionaries and profiles of uninterrupted capital letter sequences.

2.4.2 Synthetic Data

The main goals of this section are feasibility analysis and efficiency evaluation of the AGM learning algorithm. The number of observations is set to 300 grouped into two clusters (M = 2). Hyperparameters are set to $\gamma_j = 1$ [32] for sampling mixture weight p_j from Eq. (9). η and τ are considered as d-dimensional zero vectors in prior distributions of mixture parameter ξ.

Different proposed component numbers ($M' = 1, \ldots, 5$) are tested during the AGM learning process and the statistics are summarized in Table 2.1. In order to select the best number of components, we consider marginal likelihood as described in [10]. The probability density functions are plotted for both original and estimated AGM components and the polylines show the trace of



(a) (b)

Figure 2.3: (a) Original synthetic data grouping; (b) AGM clustering results

accepted moves for each component.

In terms of 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 2.2). In summary, the estimation of mean is accurate because the Euclidean distance between μ_j and $\hat{\mu}_j$ is small but the distance between standard deviation σ_{lj}, σ_{rj} and $\hat{\sigma}_{lj}, \hat{\sigma}_{rj}$ is slightly significant. However, this difference has not affected the clustering result.

2.4.3 Intrusion Detection

Along with the rapid growth of information technologies, personal and commercial behaviors tend to rely on computer network and Internet environments. However, based on the characteristics of networking, exposing sensitive privacy and valuable business secret online is extremely dangerous because accessibility and anonymity make network intrusions hard to be detected and

Table 2.1: AGM Learning Statistics

Component number M'	***Moves accepted***	***Acceptance ratio***	***Marginal likelihood***
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

Table 2.2: 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

traced, therefore, compromise network security. Cisco 2017 Annual Cybersecurity Report (ACR) [33] pointed out a crucial fact that more than one-third of organizations that experienced a breach in 2016 reported more than 20 percent of customer, opportunity and revenue loss. As a consequence, more than 90 percent of these organizations are improving threat defense technologies and processes by enhancing IT and security functions, increasing security training of employees and implementing risk mitigation techniques. Recently, machine learning-based intrusion detection solutions [34, 35] are drawing more attention because of their efficiency and flexibility.

Earlier intrusion prevention approaches, such as authentication, avoiding programming errors and encryption, were proven as insufficient because along with the increasing of the complexity of network-based software systems, exploitable weaknesses are inevitable due to programming issues. Moreover, authentication and encryption are not always reliable since credentials could be leaked and encryption algorithm could also be compromised by applying powerful hacking techniques to make the attack feasible. In consequence, once intrusion happens, detection will be harder than prevention and sometimes victims could not be even aware of it. Therefore, many supervised data mining solutions were proposed in terms of misuse and anomaly detection systems by establishing known intrusion scenarios, normal usage patterns and the sequential interrelations between user operations to identify intrusion behaviors [36]. However, the disadvantages of supervised intrusion detection systems are significant since predefined patterns and interrelations are inconsistent concerning the system upgrades and newly-founded intrusions which could lead to incessant intrusion detection system adjustment and affect its performance. Furthermore, inductive bias and overfitting problems caused by poor training datasets will also affect the accuracy of the systems. Therefore, researchers are paying more attention to unsupervised solution [37, 38] for seeking flexibility and robustness.

Therefore, we select NSL-KDD [31] (Table 2.3), an improved KDDCUP'99 intrusion-detection data-set, as the testing target since redundant records have been removed from original dataset to avoid potential learning bias. Before applying the testing models onto the dataset, the data preprocessing is needed since discrete enumerated values must be translated to numerical ones and be normalized properly to lead an accurate result. Therefore, we substitute enumerated values with their numbers of occurrences which could reflect the density distribution of discrete values. Having all numerical data in hand, we apply feature scaling method to normalize numerical values between 0 to 1 as follows:

Table 2.3: Original NSL-KDD data records

No	***Value***
1	0,tcp,ftp_data,SF,491,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,150,25,0.17,0.03,0.17,0.00,0.00,0.00,0.05,0.00,normal
2	0,udp,other,SF,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,13,1,0.00,0.00,0.00,0.00,0.08,0.15,0.00,255,1,0.00,0.60,0.88,0.00,0.00,0.00,0.00,0.00,normal
3	0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,123,6,1.00,1.00,0.00,0.00,0.05,0.07,0.00,255,26,0.10,0.05,0.00,0.00,1.00,1.00,0.00,0.00,neptune
4	0,tcp,http,SF,232,8153,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,5,5,0.20,0.20,0.00,0.00,1.00,0.00,0.00,30,255,1.00,0.00,0.03,0.04,0.03,0.01,0.00,0.01,normal
5	0,tcp,http,SF,199,420,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,30,32,0.00,0.00,0.00,0.00,1.00,0.00,0.09,255,255,1.00,0.00,0.00,0.00,0.00,0.00,0.00,0.00,normal
6	0,icmp,eco_i,SF,18,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,1,16,1.00,0.00,1.00,1.00,0.00,0.00,0.00,0.00,ipsweep

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{23}$$

where x and x' denote original and normalized values. In this way we could use unified proposal distribution for every dimension with the same value of hyperparameter Σ during random walk MCMC sampling step (Table 2.4).

K-means clustering algorithm [39] is chosen for the comparison of accuracy. Testing data records with total amount of 25192 (20% of NSL-KDD dataset) are clustered into two groups with 11743 intrusions and 13449 normal behaviors indicating components number $M' = 2$. In order to better evaluate the pros and cons of models, results derived from Gaussian mixture model (GMM) will also be taken into consideration. The comparison based on confusion matrices resulted from K-means, GMM and AGM model (Table 2.5) reveals the fact that based on a less accurate initialization given by K-means (60.85%), GMM performs almost the same way as K-means and the difference between these two models is trivial. In contrast, AGM model makes a significant improvement with much higher accuracy rate (80.47%) and precision percentage (96.86%), while much lower false positive rate (4.26%) illustrating AGM model is capable of effectively detecting intrusions from background noises. Compared with K-means and GMM, AGM model has a higher false negative rate (28.58%) which means it tends to strictly identify normal behaviors as intrusions which could be mitigated by reducing dimensions of dataset using feature selection methodologies.

Table 2.4: Translation and Normalization of Internet Protocols (Enumerated Values)

Internet Protocols	***Number of Occurrences***	***Normalized Values***
ICMP	1655	0
UDP	3011	0.071867
TCP	20526	1

Table 2.5: Confusion Matrices and Statistics of K-means, GMM and AGM Models

K-means

	NF [a]	***F*** [b]
NF	2445	9298
F	565	12884

GMM

	NF	***F***
NF	2464	9279
F	584	12865

AGM

	NF	***F***
NF	11484	259
F	5621	7828

	K-means	***GMM***	***AGM***
Accuracy	60.85%	60.85%	76.66%
Precision	20.82%	20.98%	97.79%
False Positive Rate	41.92%	41.90%	3.20%
False Negative Rate	18.77%	19.16%	32.86%

[a] Non fault-prone, [b] Fault-prone.

Table 2.6: AGM Statistics

Init. Comp. Number m	***Accuracy***	***Integrated Likelihood***
$m = 1$	55.64%	5.7074e5
$m = 2$	51.21%	4.0543e5
$m = 3$	58.99%	8.4238e5

2.4.4 Spam Filtering

Statistics reveal a crucial fact that more than 59% of worldwide e-mail traffic is considered as unsolicited messages, also well known as spams, in 2017 [40]. Most spams are irritating and resource-consuming, and some of them are extremely dangerous in terms of phishing scam, fee fraud, job offer scam, etc,. Since the damages of spam are persistent and significant not only for individuals but also for governments, companies and organizations, many spam filtering technologies have been proposed to address this issue and eliminate unwanted e-mails automatically over recent decades.

Consequently, a well organized Spambase dataset [41] is selected with attributes related to multiple spam textual features including spam word/character dictionaries and profiles of uninterrupted capital letter sequences. Data pre-processing includes Scaling-based data normalization which rescales numerical values within the range between 0 and 1 and label extraction for generating confusion matrix. To better evaluate the performance and accuracy of AGM model under different initial number of components, the integrated likelihood [10] values are given in Table 2.6 to identify the best-fit result. Obviously, the result with initial component number $m = 3$ has the largest integrated likelihood value (8.4238e5). Therefore, we select it as the best-fit result and make horizontal comparison with GMM. Statistics in Table 2.7 reveal the fact that comparing to GMM, AGM provides higher accuracy and precision, additionally, lower false positive rate and false negative rate indicate that AGM outperforms GMM. However, because of the nature of spambase, the performance of both mixture models is not satisfactory since most of spams cannot be identified. Therefore, data-based adjustment of the model might lead to a better result in the future.

Table 2.7: Confusion Matrices and Statistics of GMM and AGM

GMM

	NF [a]	***F*** [b]
NF	35	1778
F	295	2493

AGM

	NF	***F***
NF	249	1564
F	323	2465

	GMM	***AGM***
Accuracy	54.94%	58.99%
Precision	1.93%	13.81%
False Positive Rate	41.63%	38.81%
False Negative Rate	89.39%	56.46%

[a] Non fault-prone, [b] Fault-prone.

2.4.5 Conclusion

This chapter firstly illustrated a new intrusion detection approach by applying asymmetric Gaussian mixtures with a fully Bayesian learning process which is achieved by applying a hybrid sampling-based MH-within-Gibbs learning algorithm. According to the experiment results, the AGM model is proved as an effective approach for clustering. In spite of the advantages of AGM we mentioned above, some improvements are still needed to promote the accuracy and flexibility and mitigate the drawbacks. Therefore, we shall extend the Bayesian learning process and introduce model selection and feature selection methodologies to improve the performance in the case of high-dimensional datasets.

Chapter 3

Unsupervised Learning with Feature Selection: Application to Image Categorization

3.1 Introduction

Recently, as the consequence of frequent usage of mobile phone, social media and cloud storage, digitalized visual data such as photos and pictures brings difficulties for management and analysis. Unlike those within text-based documents, indexing and comparison among images could be challenging. Therefore, image categorization is becoming one of the most interesting research topics in computer vision community. Indeed, finding relevant images from a rapidly growing unannotated image database is challenging which makes the previous time-consuming manual categorization methods infeasible. Automated methods such as machine-learning-based approaches [42] introduce image representation methodologies for visual feature extraction and both generative and discriminative classifiers for categorization. Meanwhile, modern machine-learning-based solutions can be divided into two main streams, classification-based supervised and clustering-based unsupervised ones. Compared to supervised solutions, unsupervised approach has no assumption on the number of groups, therefore, friendly to new added images and categories which makes it more suitable for increasing datasets. Moreover, it also immunizes against learning biases and overfitting problems that commonly exist in most supervised approaches if model training is inappropriate.

Consequently, unsupervised categorization of images or image parts [43, 37, 38] has an important role for image and video summarization, human action recognition, and image search etc,. It can also be seen as a pre-processing step for supervised methodologies for classification or segmentation. As upgrade of single-mathematical-model-based methodologies, mixture models [6, 7, 8] can be seen as a superimposition of certain mixture components sharing dependencies with each other, therefore, lead to outstanding performance especially for high-dimensional and multi-cluster datasets.

Before deploying UIUC sports event database [44] to validate AGM framework, image processing is needed because images should be represented as visual features and, eventually, be translated into numerical data. As we discussed in Chapter 2, parameter estimation could be challenging and highly affects the performance of mixture models especially for high-dimensional data sets. For this reason, we decided to adopt scale-invariant feature transform (SIFT) [45] to detect and describe image features even under changes in image scale, noise and illumination. However, generated SIFT features have to be categorized and features that belong to each image should be considered as histograms to be the input of AGM model. Therefore, bag-of-visual-words [46] and probabilistic latent semantic analysis (pLSA) [47] are responsible for the generation of features histogram. Meanwhile, high-dimensionality will also bring difficulties to classifiers since previous approaches assume all the features of observations have the same weight of importance and carry pertinent information which is not always the case and many of those features can be irrelevant for clustering purpose. In order to tackle this problem and define relevance and importance of features, feature selection techniques [4, 23] should be taken into consideration. Eventually, irrelevant and unneeded information will be filtered by feature selection.

3.2 Dimensionality Reduction for AGM

The AGM model defined in Eq. (2) assumes that all the d features of observations have the same weight of importance and carry pertinent information which is not always the case and many of those features can be irrelevant for clustering purpose. In order to tackle this problem and define relevance and importance of features, feature selection techniques [4, 23] should be taken into consideration. By denoting background Gaussian distributions for all the d features with parameter set $\Psi = \{\mu'_1, ..., \mu'_d, \sigma'_1, ..., \sigma'_d\}$, where μ' and σ' represent the mean and standard deviation of the

Gaussian distribution, respectively. Then, Eq. (2) can be reformulated with the feature relevancy approach suggested in [22] as follows:

$$p(\mathcal{X}|\Theta,\Psi,\Phi) = \prod_{i=1}^{N}\sum_{j=1}^{M} p_j \prod_{k=1}^{d} p(X_{ik}|\xi_{jk})^{\phi_k}\mathcal{N}(X_{ik}|\psi_k)^{1-\phi_k} \tag{24}$$

where $\mathcal{N}(X_{ik}|\psi_k)$ denotes the likelihood that k-th feature of i-th observation is irrelevant where $\psi_k = (\mu'_k, \sigma'_k)$ is the parameters set of background Gaussian distribution. $\Phi = (\phi_1, \ldots, \phi_d)$ is a binary relevancy vector where $\phi_k = 1$ if k-th feature is relevant or $\phi_k = 0$ otherwise. If we consider the relevancy vector Φ as a latent variable, the complete likelihood function of AGM model with full parameter set will be given as follows:

$$p(\mathcal{X}|\Theta') = \prod_{i=1}^{N}\sum_{j=1}^{M} p_j \prod_{k=1}^{d}[\omega_k p(X_{ik}|\xi_{jk}) + (1-\omega_k)\mathcal{N}(X_{ik}|\psi_k)] \tag{25}$$

where $\Theta' = (\Theta, \Psi, \Omega)$ and $\Omega = (\omega_1, \ldots, \omega_d)$ is the relevancy weight with value range of $0 \leq \omega_d \leq 1$ which represents the probability that k-th feature is relevant. Finally, the calculation of relevancy weight ω_k is given as follows:

$$\omega_k = \frac{\prod_{i=1}^{N}\sum_{j=1}^{M} p_j p(X_{ik}|\xi_{jk})}{\prod_{i=1}^{N}\sum_{j=1}^{M} p_j p(X_{ik}|\xi_{jk}) + \prod_{i=1}^{N}\mathcal{N}(X_{ik}|\psi_k)} \tag{26}$$

Therefore, irrelevant features only have small contribution for the clustering process, thus the usability of AGM model is extended to more common and complicated cases such as high-dimensional noisy applications. The parameter learning algorithm with feature selection can be described as follows:

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. (6) and (25)

(b) Compute $n_j^{(t)}$ from Eq. (7)

(c) Generate $p_j^{(t)}$ from Eq. (9)

Metropolis-Hastings part

(d) Sample $\xi_j^{(t)}$ $(\mu_j^{(t)}, \sigma_{lj}^{(t)}, \sigma_{rj}^{(t)})$ from Eqs. (10) (11) (12)

(e) Calculate relevancy weight $\omega_k^{(t)}$ from Eq. (26)

(f) Generate background Gaussian parameters $\psi_k^{(t)}$ by random walk

(g) Compute acceptance ratio r from Eq. (13)

(h) Generate $\alpha = min[1, r]$ and $u \sim U_{[0,1]}$

(i) If $u \geq \alpha$ then $\xi^{(t)} = \xi^{(t-1)}, \psi_k^{(t)} = \psi_k^{(t-1)}, \omega_k^{(t)} = \omega_k^{(t-1)}$

RJMCMC part

(j) Generate $u' \sim U_{[0,1]}$. If $b_m >= u'$, perform split or birth step, then calculate acceptance probability $\mathcal{A}$. If the step is accepted, set $m = m + 1$.

(k) Generate $u' \sim U_{[0,1]}$. If $d_m >= u'$, perform merge or death step, then calculate acceptance probability $\mathcal{A}'$. If the step is accepted, set $m = m - 1$.

Figure 3.1 illustrates updated DAG parameter dependency figure with feature selection related parameters added.

3.3 Image Categorization

A challenging UIUC sports event database [44] is selected for our target application which has been evaluated by previous researches [48, 43]. It has 1579 images in total and consists of 8 sports event categories: rowing (250 images), badminton (200 images), polo (182 images), bocce (137 images), snowboarding (190 images), croquet (236 images), sailing (190 images), and rock climbing (194 images). The first step of image pre-processing is applying scale-invariant feature transform (SIFT) on the original image files using difference of Gaussian (DoG) [45] (Figure 3.2) as interest point detector and then, visual feature will be translated into 128-dimensional feature descriptor vectors. Next, bag-of-visual-words (BOVW) [49] approach is used to cluster feature vectors into a visual vocabulary $\mathcal{W}$ with variant vocabulary size using K-means algorithm. Consequently, each

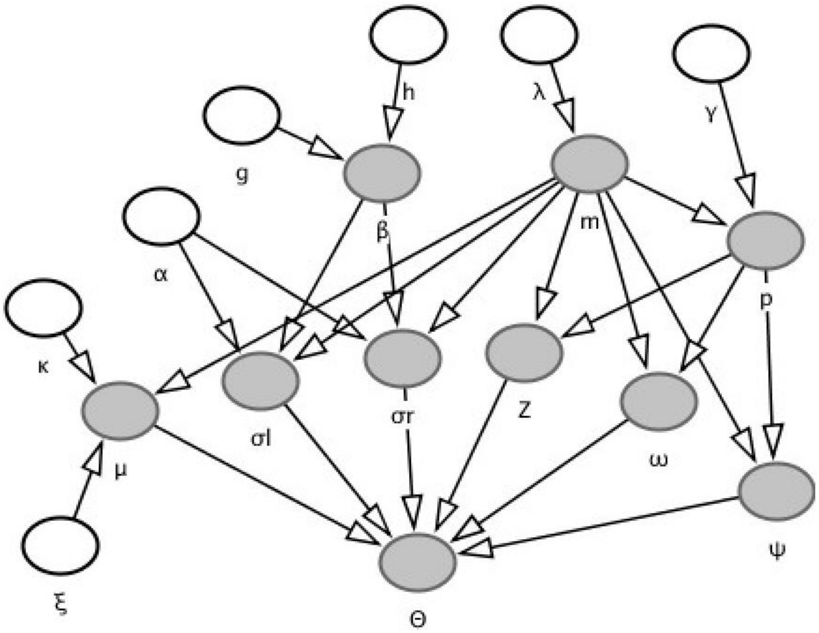
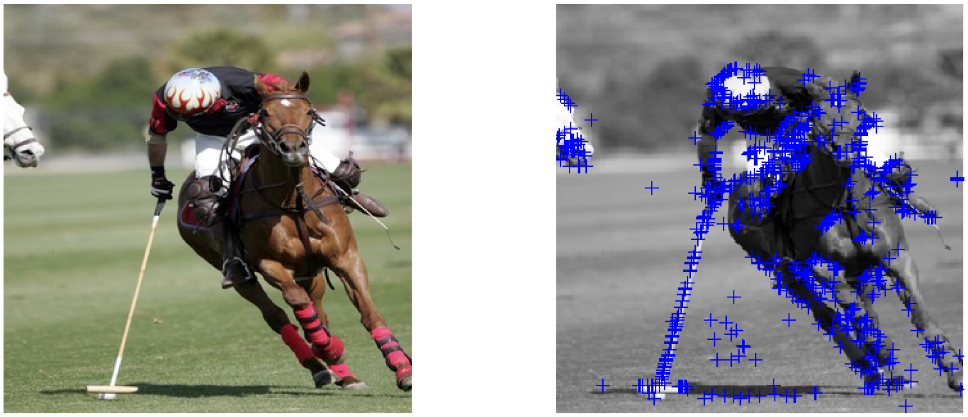


Figure 3.1: DAG of AGM Bayesian learning network with feature selection

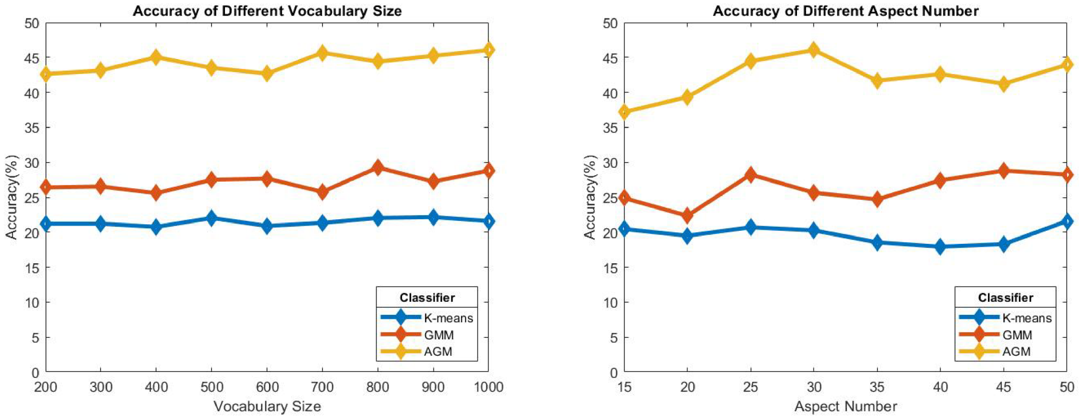
image will be represented by a frequency histogram of occurrence of visual words in $\mathcal{W}$. Since the vocabulary size in our test is between 200 to 1000, we also adopted probabilistic latent semantic analysis (pLSA) to describe each image as several latent topics (aspect) and therefore, reduce the dimension of image representation. Before applying AGM model for clustering, normalization based on feature scaling method is added to restrain the range of numerical attributes which will improve the Bayesian learning performance. Finally, we deploy the proposed AGM model as an unsupervised classifier to categorize the whole database. The proposed model is tested against image representation generated with different vocabulary sizes between 200 to 1000 with an interval of 100 and latent aspect number between 15 to 50 with an interval of 5 in order to identify the best accuracy.

According to a comparison between K-means, Gaussian mixture model (GMM) and proposed AGM model reveals the fact that applying AGM to the sports event database leads to a significant clustering accuracy boost due to the best accuracy numbers of all the 3 classifiers illustrated in figure 3.3(a) (K-means: 22.17%, GMM: 29.26% and AGM: 46.04%). Moreover, the impact of accuracy from different latent aspect numbers can be found in figure 3.3(b). Finally, the detailed clustering confusion matrix of AGM under the optimal vocabulary size (1000) and latent aspect number (30) is described in figure 3.4.



(a) (b)

Figure 3.2: a) Original UIUC sport image. b) Enhanced image with SIFT extracted visual features.

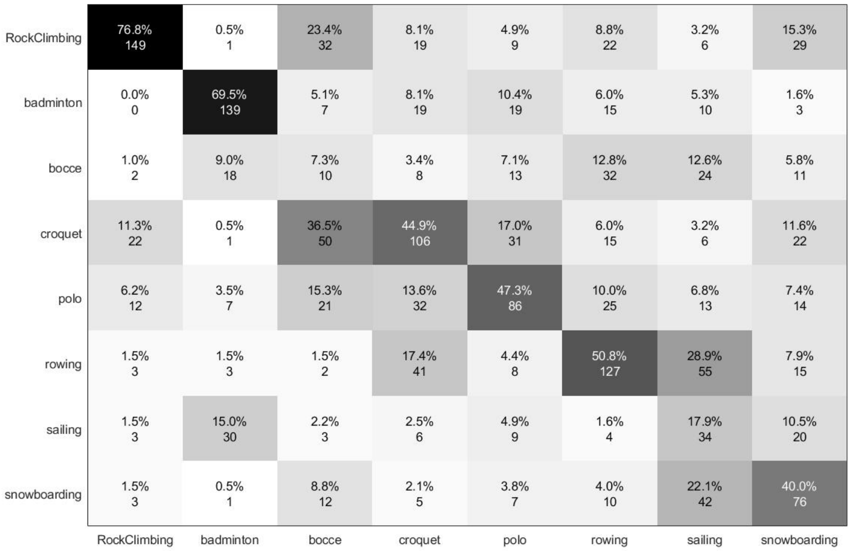


(a) (b)

Figure 3.3: a) Clustering accuracy by vocabulary size. b) Clustering accuracy by latent aspect number.

3.4 Conclusion

In this chapter, we integrated feature selection within the Bayesian framework proposed in previous chapter and we applied it to image categorization using the challenging UIUC sports event database. The proposed AGM model includes feature selection which can not only filters irrelevant and unneeded features but also weights relevant features based on the pertinent information they carry. For the parameter learning part, the proposed approach combines the merits of both Metropolis-Hastings and Gibbs sampling methods and allows model transfer throughout iterations by adopting RJMCMC methodology. A horizontal comparison between other unsupervised classifiers showing a significant accuracy improvement and future research directions will focus on model



	RockClimbing	badminton	bocce	croquet	polo	rowing	sailing	snowboarding
RockClimbing	76.8% 149	0.5% 1	23.4% 32	8.1% 19	4.9% 9	8.8% 22	3.2% 6	15.3% 29
badminton	0.0% 0	69.5% 139	5.1% 7	8.1% 19	10.4% 19	6.0% 15	5.3% 10	1.6% 3
bocce	1.0% 2	9.0% 18	7.3% 10	3.4% 8	7.1% 13	12.8% 32	12.6% 24	5.8% 11
croquet	11.3% 22	0.5% 1	36.5% 50	44.9% 106	17.0% 31	6.0% 15	3.2% 6	11.6% 22
polo	6.2% 12	3.5% 7	15.3% 21	13.6% 32	47.3% 86	10.0% 25	6.8% 13	7.4% 14
rowing	1.5% 3	1.5% 3	1.5% 2	17.4% 41	4.4% 8	50.8% 127	28.9% 55	7.9% 15
sailing	1.5% 3	15.0% 30	2.2% 3	2.5% 6	4.9% 9	1.6% 4	17.9% 34	10.5% 20
snowboarding	1.5% 3	0.5% 1	8.8% 12	2.1% 5	3.8% 7	4.0% 10	22.1% 42	40.0% 76

Figure 3.4: Confusion matrix of AGM model (Vocabulary size is 1000 and latent aspect number is 30)

adjustments and improvements to tackle high-dimensional datasets to achieve higher clustering accuracy.

Chapter 4

Conclusion

Our work is based on asymmetric Gaussian mixture (AGM) model and reversible jump Markov chain Monte Carlo (RJMCMC) learning algorithm. Previous efforts reveal the fact that AGM outperforms classic Gaussian mixture model (GMM) by taking asymmetric datasets into consideration which provides more flexibility. Our RJMCMC implementation is based on a hybrid sampling-based approach which takes advantages of both Metropolis-Hastings (MH) and Gibbs sampling methods, therefore, simplifies mathematical complexity and extends adaptability of the model. Moreover, without giving a fixed components number in advance, RJMCMC applies a dynamic data-based strategy to identify the optimal components number throughout iterations which makes the model learning a self-adaptive process. Since the model is nondeterministic, Laplace approximation based marginal likelihood is calculated for multiple runs as model selection procedure to improve the correctness and fitting accuracy. Moreover, the proposed AGM model includes feature selection which can not only filters irrelevant and unneeded features but also weights relevant features based on the pertinent information they carry.

In order to validate the performance and accuracy of the proposed approach, applications including intrusion detection, spam filtering and image categorization have been conducted and the results are analyzed and compared with popular machine learning models. Future research directions will focus on model adjustments and improvements to tackle high-dimensional datasets and achieve high clustering accuracy. The proposed work could be applied to other applications such as content-based images summarization [50], retrieval [51], and suggestion [52].

Bibliography

[1] N. Bouguila, D. Ziou, and E. Monga, “Practical bayesian estimation of a finite beta mixture through gibbs sampling and its applications,” *Statistics and Computing*, vol. 16, no. 2, pp. 215–225, 2006.

[2] T. Elguebaly and N. Bouguila, “Background subtraction using finite mixtures of asymmetric gaussian distributions and shadow detection,” *Mach. Vis. Appl.*, vol. 25, no. 5, pp. 1145–1162, 2014.

[3] S. Richardson and P. J. Green, “On bayesian analysis of mixtures with an unknown number of components (with discussion),” *Journal of the Royal Statistical Society: series B (statistical methodology)*, vol. 59, no. 4, pp. 731–792, 1997.

[4] T. Elguebaly and N. Bouguila, “Simultaneous bayesian clustering and feature selection using rjmcmc-based learning of finite generalized dirichlet mixture models,” *Signal Processing*, vol. 93, no. 6, pp. 1531–1546, 2013.

[5] ——, “Bayesian learning of finite generalized gaussian mixture models on images,” *Signal Processing*, vol. 91, no. 4, pp. 801–820, 2011.

[6] J. Yang, X. Liao, X. Yuan, P. Llull, D. J. Brady, G. Sapiro, and L. Carin, “Compressive sensing by learning a gaussian mixture model from measurements,” *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 106–119, Jan 2015.

[7] C. K. Wen, S. Jin, K. K. Wong, J. C. Chen, and P. Ting, “Channel estimation for massive mimo using gaussian-mixture bayesian learning,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 3, pp. 1356–1368, March 2015.

[8] N. Bouguila, "Count data modeling and classification using finite mixtures of distributions," *IEEE Trans. Neural Networks*, vol. 22, no. 2, pp. 186–198, 2011.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[10] N. Bouguila, D. Ziou, and R. I. Hammoud, "On bayesian analysis of a finite generalized dirichlet mixture via a metropolis-within-gibbs sampling," *Pattern Anal. Appl.*, vol. 12, no. 2, pp. 151–166, 2009.

[11] N. Bouguila and T. Elguebaly, "A fully bayesian model based on reversible jump MCMC and finite beta mixtures for clustering," *Expert Syst. Appl.*, vol. 39, no. 5, pp. 5946–5959, 2012.

[12] S. Bourouis, M. A. Mashrgy, and N. Bouguila, "Bayesian learning of finite generalized inverted dirichlet mixtures: Application to object classification and forgery detection," *Expert Syst. Appl.*, vol. 41, no. 5, pp. 2329–2336, 2014.

[13] W. K. Hastings, "Monte carlo sampling methods using markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[14] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," in *Readings in Computer Vision*. Elsevier, 1987, pp. 564–584.

[15] N. Bouguila, D. Ziou, and S. Boutemedjet, "Simultaneous non-gaussian data clustering, feature selection and outliers rejection," in *Pattern Recognition and Machine Intelligence - 4th International Conference, PReMI 2011, Moscow, Russia, June 27 - July 1, 2011. Proceedings*, ser. Lecture Notes in Computer Science, S. O. Kuznetsov, D. P. Mandal, M. K. Kundu, and S. K. Pal, Eds., vol. 6744. Springer, 2011, pp. 364–369.

[16] S. Boutemedjet, N. Bouguila, and D. Ziou, "A hybrid feature extraction selection approach for high-dimensional non-gaussian data clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 8, pp. 1429–1443, 2009.

[17] S. Raudys and A. K. Jain, "Small sample size effects in statistical pattern recognition: Recommendations for practitioners," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 3, pp. 252–264, 1991.

[18] R. Kohavi and G. H. John, “Wrappers for feature subset selection,” *Artif. Intell.*, vol. 97, no. 1-2, pp. 273–324, 1997.

[19] K. Z. Mao, “Identifying critical variables of principal components for unsupervised feature selection,” *IEEE Trans. Systems, Man, and Cybernetics, Part B*, vol. 35, no. 2, pp. 339–344, 2005.

[20] C. Tsai and C. Chiu, “Developing a feature weight self-adjustment mechanism for a k-means clustering algorithm,” *Computational Statistics & Data Analysis*, vol. 52, no. 10, pp. 4658–4672, 2008.

[21] J. G. Dy and C. E. Brodley, “Feature selection for unsupervised learning,” *Journal of Machine Learning Research*, vol. 5, pp. 845–889, 2004.

[22] M. H. Law, M. A. Figueiredo, and A. K. Jain, “Simultaneous feature selection and clustering using mixture models,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 9, pp. 1154–1166, 2004.

[23] T. Elguebaly and N. Bouguila, “Simultaneous high-dimensional clustering and feature selection using asymmetric gaussian mixture models,” *Image and Vision Computing*, vol. 34, pp. 27–41, 2015.

[24] S. Fu and N. Bouguila, “Bayesian learning of finite asymmetric gaussian mixtures,” in *Proceedings of The 31st International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems Montreal, QC, CA, June 25-28, 2018*, 2018.

[25] ——, “Asymmetric gaussian mixtures with reversible jump MCMC,” in *2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE) (CCECE 2018)*, Quebec City, Canada, May 2018.

[26] ——, “A bayesian intrusion detection framework,” in *Cyber Science 2018, Glasgow, Scotland, UK, June 11-12, 2018*, 2018.

[27] N. Bouguila and D. Ziou, “A powreful finite mixture model based on the generalized dirichlet distribution: Unsupervised learning and applications,” in *17th International Conference on Pattern Recognition, ICPR 2004, Cambridge, UK, August 23-26, 2004.* IEEE Computer Society, 2004, pp. 280–283.

[28] ——, "Dirichlet-based probability model applied to human skin detection [image skin detection]," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2004, Montreal, Quebec, Canada, May 17-21, 2004*. IEEE, 2004, pp. 521–524.

[29] D. Luengo and L. Martino, "Fully adaptive gaussian mixture metropolis-hastings algorithm," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26-31, 2013*. IEEE, 2013, pp. 6148–6152.

[30] G. Casella, C. P. Robert, and M. T. Wells, "Mixture models, latent variables and partitioned importance sampling," *Statistical Methodology*, vol. 1, no. 1-2, pp. 1–18, 2004.

[31] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, CISDA 2009, Ottawa, Canada, July 8-10, 2009*. IEEE, 2009, pp. 1–6.

[32] M. Stephens, "Bayesian analysis of mixture models with an unknown number of components-an alternative to reversible jump methods," *Annals of statistics*, pp. 40–74, 2000.

[33] Investor.cisco.com. (2018) Cisco 2017 annual cybersecurity report. [Online]. Available: https://investor.cisco.com/investor-relations/news-and-events/news/news-details/2017/Cisco-2017-Annual-Cybersecurity-Report-Chief-Security-Officers-Reveal-True-Cost-of-Breaches-And-The-default.aspx

[34] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys Tutorials*, vol. 18, no. 2, pp. 1153–1176, Secondquarter 2016.

[35] R. A. R. Ashfaq, X.-Z. Wang, J. Z. Huang, H. Abbas, and Y.-L. He, "Fuzziness based semi-supervised learning approach for intrusion detection system," *Information Sciences*, vol. 378, pp. 484 – 497, 2017.

[36] W. Lee and S. J. Stolfo, "Data mining approaches for intrusion detection," in *Proceedings of the 7th USENIX Security Symposium, San Antonio, TX, USA, January 26-29, 1998*, A. D. Rubin, Ed. USENIX Association, 1998.

[37] N. Bouguila, "Bayesian hybrid generative discriminative learning based on finite liouville mixture models," *Pattern Recognition*, vol. 44, no. 6, pp. 1183–1200, 2011.

[38] M. Azam and N. Bouguila, "Unsupervised keyword spotting using bounded generalized gaussian mixture model with ICA," in *2015 IEEE Global Conference on Signal and Information Processing, GlobalSIP 2015, Orlando, FL, USA, December 14-16, 2015*. IEEE, 2015, pp. 1150–1154.

[39] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[40] K. Lab. (2018) Spam: share of global email traffic 2014-2017. [Online]. Available: https://www.statista.com/statistics/420391/spam-email-traffic-share/

[41] G. F. J. S. Mark Hopkins, Erik Reeber. (2018) Uci machine learning repository: Spambase data set. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Spambase?ref=datanews.io

[42] Y. Han and X. Qi, "Machine-learning-based image categorization," in *International Conference Image Analysis and Recognition*. Springer, 2005, pp. 585–592.

[43] W. Fan and N. Bouguila, "Infinite dirichlet mixture model and its application via variational bayes," in *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on*, vol. 1. IEEE, 2011, pp. 129–132.

[44] L.-J. Li and L. Fei-Fei, "What, where and who? classifying events by scene and object recognition," 2007.

[45] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.

[46] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 524–531.

[47] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine learning*, vol. 42, no. 1-2, pp. 177–196, 2001.

[48] F. Najar, S. Bourouis, A. Zaguia, N. Bouguila, and S. Belghith, "Unsupervised human action categorization using a riemannian averaged fixed-point learning of multivariate ggmm," in *International Conference Image Analysis and Recognition*. Springer, 2018, pp. 408–415.

[49] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Workshop on statistical learning in computer vision, ECCV*, vol. 1, no. 1-22. Prague, 2004, pp. 1–2.

[50] N. Bouguila, "Spatial color image databases summarization," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2007, Honolulu, Hawaii, USA, April 15-20, 2007.* IEEE, 2007, pp. 953–956.

[51] N. Bouguila and D. Ziou, "Improving content based image retrieval systems using finite multinomial dirichlet mixture," in *Machine Learning for Signal Processing, 2004. Proceedings of the 2004 14th IEEE Signal Processing Society Workshop*. IEEE, 2004, pp. 23–32.

[52] S. Boutemedjet, D. Ziou, and N. Bouguila, "Unsupervised feature selection for accurate recommendation of high-dimensional image data," in *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*, J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis, Eds. Curran Associates, Inc., 2007, pp. 177–184.