Mixture Modeling with Compact Support Distributions for Unsupervised Learning

Ambedkar Dukkipati, Debarghya Ghoshdastidar and Jinu Krishnan
Department of Computer Science and Automation
Indian Institute of Science
Bangalore, 560012, India
Email:{ad, debarghya.g, jinu.krishnan}@csa.iisc.ernet.in

***Abstract*—The importance of the q-Gaussian distributions is attributed to their power law nature and the fact that they generalize the Gaussian distributions ($q \to 1$ retrieves the Gaussian distributions). While for $q > 1$, a q-Gaussian distribution is nothing but a Student's t-distribution, which is a long tailed distribution, for $q < 1$ it is a distribution with a compact support. Though mixture modeling with t-distributions has been studied, mixture modeling with compact support distributions has not been explored in the literature. The main aim of this paper is to study mixture modeling using q-Gaussian distributions that have a compact support. We study estimation of the parameters of this model using Maximum Likelihood Estimator (MLE) via Expectation Maximization (EM) algorithm. We further study applications of these compact support distributions to clustering and anomaly detection. As far as our knowledge, this is the first work that studies compact support distributions in statistical modeling for unsupervised learning problems.**

I. Introduction

Gaussian Mixture Models (GMM) are widely used in various applications including modeling the distributions of different kinds of random phenomena, intrinsic to classification and clustering. However, the major disadvantage of GMM is due to exponentially decaying tail of Gaussian mixture model. Firstly, it gives poor performance if the class conditional density originally follows power-law nature. Moreover, the exponential tail of the Gaussian distribution makes it sensitive to outliers which affects the robustness of such models [11] specially when the dimension of the data is high.

In the presence of noise one models this by either using only Student's-t or using Gaussian and uniform in addition to it for modeling noise. But one can note that noise need not be uniformly distributed nor can be modeled using uniform distribution. Mixture models based on the heavy tailed Student's-t distribution has been studied in [12], [14]. Some other variants of mixture of Gaussians has been studied in statistics [9].

The Bayesian treatment of the fat tailed t-distributions has been studied in [14]. In their work, they have clearly shown the limitations of GMM in the presence of outliers, and the robustness of fat tailed distributions in the presence of atypical components in the data.

This work is supported by DST-SERB through the project DST:SB/S3/EECE/093/2014.

In this work, we study mixture modeling q-Gaussian distributions that include Students-t distributions for $q > 1$. There is a co-relation between the degree of freedom in Student's-t distribution and q of q-Gaussian distribution for the case $q > 1$. q-Gaussian distributions are generalization of the Gaussian distribution as one can retrieve Gaussian distribution by setting $q \to 1$. q-Gaussian distributions are also maximum entropy distributions as one can also derive q-Gaussian distributions by maximizing Tsallis entropy [15] (a generalization of Shannon's entropy) with respect to appropriate mean and variance moment constraints. The significance of q-Gaussian distributions is due to its ability to encompass both long-tailed (for $q > 1$) and compact support (for $q < 1$) properties.

Now the interesting question is whether one can use compact support distributions, for mixture modeling? In this paper we try to answer this question by deriving a maximum likelihood (ML) formulation for this case and present a modified expectation maximization (EM) algorithm to explicitly compute the parameters of the model. We study applicability of these models for model based clustering and outlier detection.

The rest of the paper is organized as follows. In Section II we briefly discuss q-Gaussian distributions and their properties. EM algorithm for the proposed model, mixture of q-Gaussians is developed in Section III. We discuss two applications of mixture of q-Gaussians in this paper: (i) clustering, and (ii) anomaly detection; these are discussed in Section IV. In Section V we give experimental results and finally we provide some concluding remarks in Section VI.

II. q-Gaussian Distribution

The q-Gaussian distributions are obtained by maximizing Tsallis entropy with respect to appropriate moment constraints. Tsallis entropy is defined as [4], [15]

$$H_q(x) = \frac{1 - \int_{\mathcal{X}} [p(x)]^q \mathrm{d}x}{q-1}, \quad q \in \mathbb{R}. \tag{1}$$

When $q \to 1$ one can retrieve Shannon entropy. Given mean μ and covariance matrix Σ of a random variable X the multivariate q-Gaussian distribution can thus be defined as

$$\mathcal{G}_q(X|\mu_q, \Sigma_q) = \frac{1}{\Lambda_q |\Sigma_q|^{\frac{1}{2}}} \exp_q\left(-\frac{\Delta^2}{d+2-dq}\right), \tag{2}$$

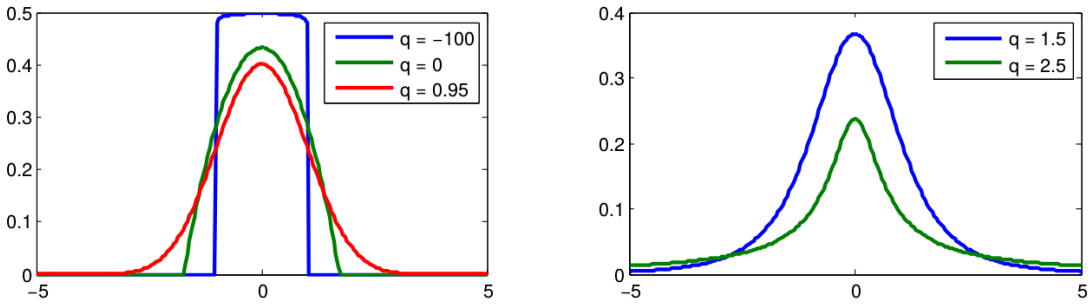


Fig. 1: Nature of q-Gaussian distributions. The left figure shows the distribution for $q < 1$, while the right one considers $q > 1$.

where d is the dimension, q is a real number,

$$\Delta^2 = (X - \mu_q)^T {\Sigma_q}^{-1} (X - \mu_q),$$

μ_q termed as q-mean, Σ_q is termed as q-variance which is defined as

$$\Sigma_q = \left(\frac{(d+4) - (d+2)q}{(d+2) - dq} \right) \Sigma$$

Here $\exp_q$ is the q-exponential and is defined as

$$\exp_q = [1 + (1-q)x]_+^{\frac{1}{1-q}} \tag{3}$$

where

$$[x]_+ = \begin{cases} 0, & \text{if } x \leq 0 \\ x, & \text{otherwise.} \end{cases}$$

This constraint is known as the Tsallis condition [7]. Further Λ_q is defined as

$$\Lambda_q = \begin{cases} \left(\pi \frac{d+2-dq}{1-q} \right)^{\frac{d}{2}} \frac{\Gamma\left[\frac{2-q}{1-q}\right]}{\Gamma\left[\frac{2-q}{1-q} + \frac{d}{2}\right]}, & \text{for } -\infty < q < 1 \\ (2\pi)^{\frac{d}{2}}, & q = 1 \\ \left(\pi \frac{d+2-dq}{q-1`} \right)^{\frac{d}{2}} \frac{\Gamma\left[\frac{1}{q-1} - \frac{d}{2}\right]}{\Gamma\left[\frac{1}{q-1}\right]}, & \text{for } 1 < q < 1 + \frac{2}{d} \end{cases} \tag{4}$$

Figure 1 shows q-Gaussian distributions for various q values ranging from -100 to 2.5, $\mu_q = 0.0$ and $\sigma_q = 1.0$. An interesting aspect here is that for $q < 1$, q-Gaussians have finite support. In Figure 1 it can be seen that the probability becomes zero for different values of x depending on the q value, which emphasize the fact of finite support distributions when $q < 1$. For a fixed μ_q and Σ_q an interesting observation is that as $q \to -\infty$ the distribution resembles a finite support uniform distribution. Figure 1 exhibits this observation. Conversely the support set becomes infinity when $q \to 1$ which is the Gaussian distribution. The case of $q > 1$ can be mapped to Student's-t distribution as $q = 1 + \frac{2}{d+\nu}$.

III. EM Algorithm for Mixture of q-Gaussians

A. Mixture of q-Gaussians

A mixture of q-Gaussians can be defined as

$$p(x|\Theta) = \sum_{n=1}^{K} w_k \mathcal{G}_{q_k}(x|\mu_{q_k}, \Sigma_{q_k}) \tag{5}$$

where w_k is the mixing proportion of the k^{th} component while μ_{q_k}, Σ_{q_k} are the respective q-mean and q-variance with $q = q_k$, $k = 1, \ldots, K$.

Given a set of independent samples, $X = (x_1, \ldots, x_N) \subset \mathbb{R}^d$, the standard practice [1] is to consider component labels $(z_{nk})_{n=1,\ldots,N,k=1,\ldots,K}$ such that $z_{nk} = 1$ only when $x_n \sim \mathcal{G}_{q_k}(.|\mu_{q_k}, \Sigma_{q_k})$ defined in (4). The complete-data likelihood, including the latent variables $Z = (z_{nk})_{n,k}$ can be written as

$$\begin{aligned} &\log p(X, Z|\Theta) \\ &= \log \left(\prod_{n=1}^{N} p(x_n|z_{n1}, \ldots, z_{nK}, \Theta) p(z_{n1}, \ldots, z_{nK}|\Theta) \right) \\ &= \sum_{n=1}^{N} \log \left(\prod_{k=1}^{K} \mathcal{G}_{q_k}(x_n|\mu_{q_k}, \Sigma_{q_k})^{z_{nk}} \right) \\ &\qquad + \sum_{n=1}^{N} \log \left(\prod_{k=1}^{K} w_k^{z_{nk}} \right). \end{aligned} \tag{6}$$

We express the above relation as

$$\log p(X, Z|\Theta) = L_1 + L_2 \ , \tag{7}$$

where we use (2) to express the first term

$$L_1 = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \Bigg[-\log \Lambda_{q_k} - \frac{1}{2} \log |\Sigma_{q_k}| + \frac{\log \left(d + 2 - dq_k - (1-q_k)(x_n - \mu_{q_k})^T \Sigma_{q_k}^{-1} (x_n - \mu_{q_k}) \right)}{1 - q_k} \Bigg]$$

as a function of q_k, μ_{q_k} and Σ_{q_k}. The second term involves only the mixing coefficients as

$$L_2 = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \log w_k \ .$$

B. EM algorithm for ML estimation

The basic idea of employing EM algorithm for maximum likelihood estimation is to compute, at each iteration t, the expectation of complete-data log likelihood, conditioned on the data and current estimate of parameters given by

$$\Theta^{(t)} = \left\{ (w_k^{(t)}, q_k^{(t)}, \mu_{q_k}^{(t)}, \Sigma_{q_k}^{(t)}), k = 1, \ldots, K \right\} \ .$$

This step (E-step) requires the expected values of the latent variables Z given by

$$\gamma^{(t)}(z_{nk}) = \mathsf{E}[z_{nk}|X,\Theta] = \frac{w_k^{(t)}\mathcal{G}_{q_k^{(t)}}(x_n|\mu_k^{(t)},\Sigma_k^{(t)})}{\sum_{j=1}^{K} w_j^{(t)}\mathcal{G}_{q_j^{(t)}}(x_n|\mu_j^{(t)},\Sigma_j^{(t)})} \ , \tag{8}$$

which can be used to compute the conditional expectation of complete log-likelihood function as

$$\begin{aligned} Q\left(\Theta,\Theta^{(t)}\right) &= \mathsf{E}_Z\left[\log p(X,Z|\Theta)\,\middle|\,X,\Theta^{(t)}\right] \\ &= Q_1\left((q_k,\mu_{q_k},\Sigma_{q_k})_{k=1,\ldots,K},\Theta^{(t)}\right) \\ &\quad + Q_2\left((w_k)_{k=1,\ldots,K},\Theta^{(t)}\right) \end{aligned} \tag{9}$$

with each of the terms being

$$\begin{aligned} &Q_1\Big((q_k,\mu_{q_k},\Sigma_{q_k})_{k=1,\ldots,K},\Theta^{(t)}\Big) \\ &= \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma^{(t)}(z_{nk})\Bigg[-\log\Lambda_{q_k} - \frac{1}{2}\log|\Sigma_{q_k}| \\ &+ \frac{\log\left(d+2-dq_k(1-q_k)(x_n-\mu_{q_k})^T\Sigma_{q_k}^{-1}(x_n-\mu_{q_k})\right)}{(1-q_k)}\Bigg] \end{aligned} \tag{10}$$

and

$$Q_2\Big((w_k)_{k=1,\ldots,K},\Theta^{(t)}\Big) = \sum_{n=1}^{N}\sum_{k=1}^{K}\gamma^{(t)}(z_{nk})\log w_k \ . \tag{11}$$

In the next step (M-step), the set of parameters $\Theta^{(t)}$ is updated to maximize $Q\left(\Theta,\Theta^{(t)}\right)$ in (9). We can observe that the mixing coefficients w_k can be updated independently, but the remaining parameters pertaining to each of the q-Gaussian components are all associated with maximization of (10). The updated value of mixing coefficient is obtained by maximizing

$$Q_2\Big((w_k)_{k=1,\ldots,K},\Theta^{(t)}\Big) + \lambda\left(\sum_{k=1}^{K} w_k - 1\right) \ ,$$

where λ is a Lagrange multiplier corresponding to the constraints that mixing coefficients form a probability mass function. The maximizer is obtained at

$$w_k^{(t+1)} = \frac{\sum_{n=1}^{N}\gamma^{(t)}(z_{nk})}{\sum_{n=1}^{N}\sum_{k=1}^{K}\gamma^{(t)}(z_{nk})} = \frac{1}{N}\sum_{n=1}^{N}\gamma^{(t)}(z_{nk}) \tag{12}$$

since the sum of z_{nk} over all samples and components equals the total number of samples. For the other parameters, we equate the partial derivatives of (10), with respect to μ_{q_k}, Σ_{q_k} and q_k, to zero. We observe that equating the derivatives to zero does not provide a closed form solution for μ_{q_k} and Σ_{q_k}. Hence, we use the previous estimates in form of

$$\begin{aligned} &\beta_k\left(\mu_{q_k},\Sigma_{q_k},q_k\right) \\ &= \left(d+2-dq_k-(1-q_k)(x_n-\mu_{q_k})^T\Sigma_{q_k}^{-1}(x_n-\mu_{q_k})\right) \end{aligned}$$

Using the above term, the updates for μ_{q_k} and Σ_{q_k} are

$$\mu_{q_k}^{(t+1)} = \frac{\sum_{n=1}^{N}\frac{\gamma^{(t)}(z_{nk})}{\beta_k\left(\mu_{q_k}^{(t)},\Sigma_{q_k}^{(t)},q_k^{(t)}\right)}x_n}{\sum_{n=1}^{N}\frac{\gamma^{(t)}(z_{nk})}{\beta_k\left(\mu_{q_k}^{(t)},\Sigma_{q_k}^{(t)},q_k^{(t)}\right)}} \tag{13}$$

and

$$\begin{aligned} &\Sigma_{q_k}^{(t+1)} \\ &= \frac{2\sum_{n=1}^{N}\frac{\gamma^{(t)}(z_{nk})}{\beta_k\left(\mu_{q_k}^{(t+1)},\Sigma_{q_k}^{(t)},q_k^{(t)}\right)}\left(x_n-\mu_{q_k}^{(t)}\right)\left(x_n-\mu_{q_k}^{(t)}\right)^T}{\sum_{n=1}^{N}\gamma^{(t)}(z_{nk})} \ . \end{aligned} \tag{14}$$

One can easily see that for $q_k \to 1$ (Gaussian), $\beta_k \to 2$ and hence, the updates are independent of the previous estimates. In fact, the updates are identical to the ones for ML estimation of GMM. Similar to the degree of freedom estimation in [12], we provide the equation that needs to be solved to obtain estimates for $q_k^{(t+1)}$ for $k = 1,\ldots,K$. However, these updates involve the normalizing constant, Λ_{q_k}, and hence, we have to deal with the cases $q_k > 1$ and $q_k < 1$ separately. For $q_k > 1$, we solve

$$\begin{aligned} &\Big(\sum_{n=1}^{N}\gamma^{(t)}(z_{nk})\Big)\Bigg(\frac{d\left(q_k^{(t+1)}-1\right)}{\left(d+2-dq_k^{(t+1)}\right)} + \psi\Bigg(\frac{1}{q_k^{(t+1)}-1}-\frac{d}{2}\Bigg) \\ &\qquad - \psi\Bigg(\frac{1}{q_k^{(t+1)}-1}\Bigg) - 1\Bigg) \\ &+ \sum_{n=1}^{N}\Bigg(\gamma^{(t)}(z_{nk})\log\beta_k\left(\mu_{q_k}^{(t+1)},\Sigma_{q_k}^{(t+1)},q_k^{(t+1)}\right) \\ &\qquad + \frac{2}{\beta_k\left(\mu_{q_k}^{(t+1)},\Sigma_{q_k}^{(t+1)},q_k^{(t+1)}\right)}\Bigg) = 0 \ , \end{aligned} \tag{15}$$

where $\psi(y) = \frac{\mathrm{d}}{\mathrm{d}y}\log\Gamma(y)$ is the digamma function.

On the other hand, for $q_k < 1$, we solve

$$\begin{aligned} &\Big(\sum_{n=1}^{N}\gamma^{(t)}(z_{nk})\Big)\Bigg(\frac{d\left(q_k^{(t+1)}-1\right)}{\left(d+2-dq_k^{(t+1)}\right)} \\ &+ \psi\Bigg(1+\frac{1}{1-q_k^{(t+1)}}+\frac{d}{2}\Bigg) - \psi\Bigg(1+\frac{1}{1-q_k^{(t+1)}}\Bigg) - 1\Bigg) \\ &+ \sum_{n=1}^{N}\Bigg(\gamma^{(t)}(z_{nk})\log\beta_k\left(\mu_{q_k}^{(t+1)},\Sigma_{q_k}^{(t+1)},q_k^{(t+1)}\right) \\ &\qquad + \frac{2}{\beta_k\left(\mu_{q_k}^{(t+1)},\Sigma_{q_k}^{(t+1)},q_k^{(t+1)}\right)}\Bigg) = 0. \end{aligned} \tag{16}$$

The above two equations are similar only in particular cases, where d is even and $\frac{1}{|q_k^{(t+1)}-1|}$ is not an integer. It is easy to see that analytical solution of the above relations is not possible, and hence, one has to resort to numerical schemes. We further note that since the estimation is different for either cases, we need to specify, a priori, whether a component has $q_k > 1$ or $q_k < 1$.

C. Optimization using Simulated Annealing

Solution of the optimization problem of maximizing the log likelihood function was not possible analytically for q_k, μ_{q_k} and Σ_{q_k}, $k = 1, \ldots, K$. In the case of μ_{q_k} and Σ_{q_k} an iterative method using previous estimates is possible, but for the parameters q_k the derivative equations (15) and (16) are intractable rendering any analytic method useless. In this paper we use simulated annealing to solve this optimization problem.

To demonstrate this, at the outset, we have performed the following experiment. 5000 samples were drawn from a predefined unimodal q-Gaussian distribution, *i.e.*, $K = 1$. Simulated annealing was used to estimate the parameter q_1. The results are shown in Table I. The results clearly indicate the ability of Simulated annealing to find the global optimum.

	μ	σ	q of distribution	q from SA
case 1	0.0	1.0	0.9	0.8901
case 2	0.0	1.0	0.5	0.4898
case 3	0.0	1.0	0.1	0.0652

TABLE I: Estimated q using simulated Annealing from 5000 samples drawn from a pre-defined q-distribution.

Even though iterative procedures are in place for estimating μ_{q_k}, it is not a closed form solution. We use simulated annealing to estimate this as well. Following the same experiment described above, we studied the problem of estimating μ_{q_1}, and the results are summarized in Table II. The results show that the estimated μ_{q_1} is close to the distribution's mean from which the samples were drawn warranting the use of simulated annealing in the estimation of μ_{q_1}.

	q_{actual}	q_{comp}	μ_q *(actual)*	μ_q *(estimated)*
case 1	0.9	0.8967	0.0	-0.0061
case 2	0.8	0.7816	0.0	-0.0088
case 3	0.5	0.4831	0.0	-0.0022
case 4	0.1	0.0927	0.0	-0.0078

TABLE II: Estimation of q and μ_q using simulated annealing.

However the estimation of Σ_{q_k} is not taken up. Though not to be neglected, the sensitivity of Σ_{q_k} is much less compared to parameters q_k and μ_{q_k} in terms of clustering accuracy. The main bottleneck in the estimation of the covariance matrix is the explosion in the number of parameters. For instance in case of d dimensional data, the number of parameters in Σ_{q_k} is $\frac{d^2-d}{2}$ for each k. This results in the formation of a very high dimensional search space for the probabilistic estimator slowing down the estimation process considerably.

In summary, we use a modified form of EM algorithm to solve the ML optimization problem. The posterior probabilities and mixing coefficients are computed using the EM closed form formula (8) and (12). The log likelihood is computed as in (7). Σ_{q_k} is computed using the iterative procedure given in (14), while q_k and μ_{q_k} for each component is estimated using simulated annealing.

IV. Applications of Mixture of q-Gaussians

In this section, we discuss the use of q-GMM in clustering and anomaly detection. We also propose algorithms for these applications.

A. Clustering using q-GMM

We now describe the clustering algorithm using q-Gaussian Mixture Model. The algorithm is described below, where we specify the number of mixture components K as an input. We also fix the maximum number of iterations to be T.

Algorithm 1 The q-GMM Clustering Algorithm

1: Initialize $q_k^{(0)}$ and $\Sigma_{q_k}^{(0)}$ for $k = 1 \ldots K$ clusters
2: Run K-means and initialize $\mu_{q_k}^{(0)}$ and mixing coefficients $w_k^{(0)}$ for $k = 1, \ldots, K$
3: **for** $t = 1$ to T **do**
4: Compute $\gamma^{(t)}$ using (8)
5: Estimate $L^{(t)}$ the maximum log likelihood
6: Estimate $q_k^{(t)}$ and $\mu_{q_k}^{(t)}$ using Simulated Annealing
7: Compute $\Sigma_{q_k}^{(t)}$ using (14)
8: Compute $w_k^{(t)}$ using (12)
9: **end for**
10: $I = \underset{t=1\ldots T}{\text{argmax}} \quad L^{(t)}$
11: Set $\mu_{q_k} = \mu_{q_k}^{(I)}$, $\Sigma_{q_k} = \Sigma_{q_k}^{(I)}$ and $w_k = w_k^{(I)}$ for each $k = 1, \ldots, K$
12: **return** $\gamma = \gamma^{(I)}$

K-means initializes the cluster center variable $\mu_{q_k}^{(0)}$, and the mixing coefficient $w_k^{(0)}$ for each component is initialized based on the data assignments done by K-means. Since the Σ_{q_k} are not estimated, convergence to a global maximum is sensitive to the initialization of the algorithm. This dependence is not as sensitive as the case with normal EM algorithm, nevertheless to attain a stable and conclusive result, we recommend multiple runs of K-means random seeding. During each such iteration, the $q_k^{(0)}$ and $\Sigma_{q_k}^{(0)}$ are initialized. In our experiments, we have used $\Sigma_{q_k}^{(0)} = I$, whereas we set $q_k^{(0)} = 0.99$ for $q_k < 1$ and 1.01 for $q_k > 1$.

We now briefly discuss the Simulated Annealing technique used for estimating q_k and μ_{q_k}. While there is no constraint on the μ_{q_k}, one should note that the search space for q_k is constrained as either $q_k < 1$ for the compact support q-Gaussian, or $1 < q < 1 + \frac{2}{d}$ for the unbounded support q-Gaussian. The optimization aims at minimizing the negative

of the log likelihood as given in (7). The algorithm is listed below, which iterates till the change in likelihood, L is below some pre-specified threshold ϵ.

Algorithm 2 Simulated Annealing Optimization

while $\Delta L < \epsilon$ **do**
  Use standard annealing procedure to determine a feasible point for q_k and μ_{q_k}, $k = 1 \dots K$
  Set $L = 0$
  for $i = 1$ to N **do**
    if $q_k < 1$ and $x_i \notin$ support of $\mathcal{G}_{q_k}(\cdot|\mu_{q_k}, \Sigma_{q_k})$ for some k **then**
      $L = \infty$
      Go to step 2
    else
      Update L using (7)
    end if
  end for
end while
return L, q_k and μ_{q_k} for $k = 1, \dots, K$

We remark on the computational time of the simulated annealing algorithm, which depends on the size of the search space. Since, $q_k < 1$ provides compact support distributions, the annealing process is faster. Furthermore, one can further reduce the search space by eliminating the values for q_k that makes certain data points behave as outliers. However, the same argument does not hold when $q_k > 1$, and hence, one cannot avail such computational benefits in this case.

B. Anomaly detection using q-GMM

It is well known that generative models are suitable for anomaly detection [6], [10]. To this end, q-Gaussians with $q < 1$ are quite appropriate for the task due to their compact support. We propose to use q-GMM with $q < 1$ to model the normal behavior, whereas the anomalous instances are assigned as outliers that lie outside the support of the q-GMM.

Typically, any supervised outlier detection algorithm consists of two phases – a learning phase used for estimating the model parameters, and a subsequent detection phase. While mixture models provide superior performance as compared to single distributions [1], it is quite tricky to estimate the appropriate number of components in the mixture. An useful strategy is to use statistical measures such as Bayesian Information Criteria (BIC) [2]. Here, one starts with a single distribution model ($K = 1$) and gradually increases the number of components K. Finally, one selects the model $\mathcal{M}$ that achieves the highest BIC, where this quantity is given by

$$BIC(\mathcal{M}) = -2 \log p(X|\mathcal{M}) - m_{\mathcal{M}} \log N \ . \qquad (17)$$

Here, $p(X|\mathcal{M})$ denotes the maximum likelihood of the given data X obtained by estimating the optimal model parameters for $\mathcal{M}$ as discussed before. The quantity $m_{\mathcal{M}}$ denotes the number of parameters to be estimated for model $\mathcal{M}$, and N is the number of data points.

After learning an appropriate model $\mathcal{M}$ for the normal behavior, one may assume that the data follows the distribution

$$\mathcal{D} = (1 - \lambda)\mathcal{M} + \lambda\mathcal{A},$$

where $\mathcal{A}$ is the distribution depicting the outliers, and $\lambda \ll 1$ denotes the probability of occurrence of outliers. The value of λ can be estimated experimentally or from past experiences or could be a carefully chosen value given by a domain expert. We follow the lines of [6] for the detection phase. Assume that the data for detection arrives in an online manner over time $t = 1, 2, \dots$. Let at some instant t, the set of normal data points and outlier be given by $X^{(t)}$ and $X^{(t)}_{out}$, respectively. Then the likelihood of the entire data at instant t is given by

$$\begin{aligned} L^{(t)} &= \log p\left(X^{(t)}, X^{(t)}_{out}|\mathcal{D}\right) \\ &= \left((1-\lambda)^{|X^{(t)}|} \prod_{x \in X^{(t)}} p(x|\mathcal{M})\right)\left(\lambda^{|X^{(t)}_{out}|} \prod_{x \in X^{(t)}_{out}} p(x|\mathcal{A})\right) \end{aligned} \qquad (18)$$

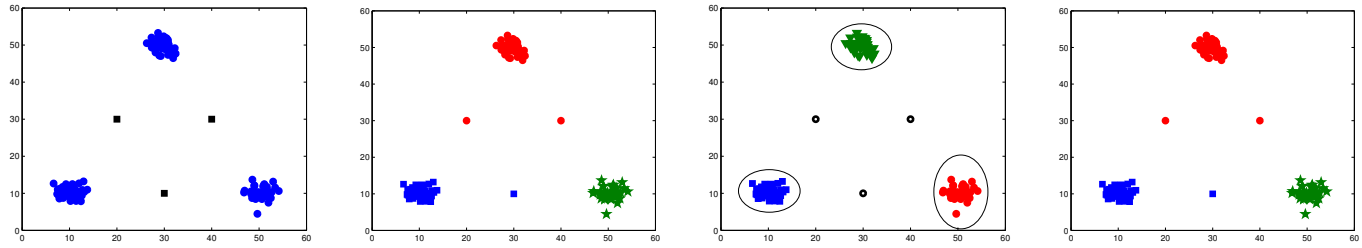
Any new data is labeled as an outlier if its addition to the set of normal data points reduces the likelihood below a certain threshold, say c.

The overall algorithm is presented below, where one specifies the maximum number of mixture components K_{max} and the threshold for likelihood ratio c. The algorithm initially learns an appropriate model from a given set of normal data points X, and based on this model it detects whether new instances are anomalous. The generative model is also updated based on the decision.

Algorithm 3 The q-GMM Anomaly Detection Algorithm

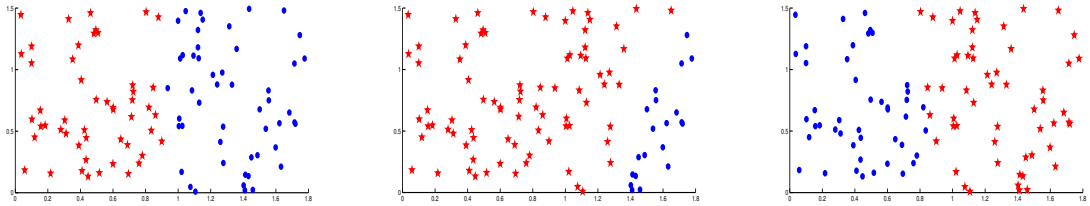
Learning phase
for $t = 1$ to K_{max} **do**
  Run Algorithm 1 to estimate q-GMM model, $\mathcal{M}_K$, with K mixture components
  Compute BIC($\mathcal{M}_K$) using (17)
end for
Set $\mathcal{M} = \underset{K}{\text{argmax}} \quad \text{BIC}(\mathcal{M}_K)$

Detection phase
Set $X^{(0)} = X$ (data for learned model $\mathcal{M}$
Initialize model $\mathcal{A}$, and fix $X^{(0)}_{out}$ to be empty set
for $t = 0, 1, 2, \dots$ **do**
  Compute $L^{(t)}$ using (18)
  For new data $\overline{x}$, compute $\overline{L}^{(t+1)}$ using (18) assuming that $X^{(t+1)} = X^{(t)} \cup \{\overline{x}\}$
  Compute the ratio $\eta = \overline{L}^{(t+1)}/L^{(t)}$
  if $\eta < c$ **then**
    Mark $\overline{x}$ as anomaly, and let $X^{(t+1)}_{out} = X^{(t)}_{out} \cup \{\overline{x}\}$
  else
    Mark $\overline{x}$ as normal, and let $X^{(t+1)} = X^{(t)} \cup \{\overline{x}\}$
  end if
  Re-estimate parameters for $\mathcal{M}$
end for



(a) Dataset (b) GMM clusters (c) q-GMM ($q < 1$) clusters (d) q-GMM ($q > 1$) clusters

Fig. 2: Figure 2a shows three clusters with three outliers, Figures 2b, 2c, 2d show the clustering result with GMM, q-GMM ($q < 1$) and q-GMM ($q > 1$), respectively. The outliers are assigned to clusters in GMM and q-GMM ($q > 1$), but not in q-GMM ($q < 1$).



(a) Dataset (b) Clusters from GMM (c) Clusters from q-GMM

Fig. 3: Evidence of superior performance of q-GMM as compared to GMM in separating two nearly rectangular clusters with low inter cluster distance.

V. Experimental Results

In this section, we demonstrate the empirical advantages of q-GMM, particularly for $q < 1$. We compare the performance of q-GMM with GMM in the context of clustering and anomaly detection. It was earlier noted that for $q > 1$, q-Gaussian distribution has a one-one association with Students-t distribution, and hence, our results with q-GMM ($q > 1$) provides an immediate comparison with mixture of t-distributions [12].

A. Synthetic Experiments

We first consider few synthetic datasets to demonstrate the contrast in behavior of q-GMM and GMM.

Example 1. An illustrative example is constructed to visualize how the parameter q affects the cluster formation. The data set consists of 3 distinct clusters and 3 outlier points (see Figure 2(a)). The remaining plots in Figure 2 shows the clusters obtained from GMM, q-GMM ($q < 1$) and q-GMM ($q > 1$). Due to their unbounded support, both GMM and q-GMM accommodated the outliers as part of the cluster, which shows the susceptibility of these models to noise or outliers. Though not shown in Figure 2, we observed that model parameter of GMM with and without outliers are significantly different. The change in parameters for q-GMM ($q > 1$) is much less since it allows the occurrence of distant points.

However in the case of q-GMM with $q < 1$, the estimated model parameters are such that the support of the three mixture components lie around the clusters, and do not encompass the outliers. The boundaries for the supports of the mixture components are shown in Figure 2c. Thus, this model is truly capable of estimating the model parameters by getting rid of the outliers.

Example 2. A more realistic situation is considered in Figure 3a, the synthetic dataset consists of 100 two-dimensional vectors. These points were generated by two uniform distributions resulting in rectangular clusters.

We clustered the data using GMM and q-GMM, and results are shown in Figures 3b and 3c. It is clearly observed that performance of q-GMM is much better than GMM. We also measured the Rand index [13] of the obtained clusters, and found that the result using q-GMM achieves a high Rand index of 0.96 as compared to mere 0.58 in the case of GMM. However, one can note that if the separation between the clusters is increased, then both GMM and q-GMM correctly capture the clusters.

Example 3. We now demonstrate the robustness of q-GMM with $q < 1$ that motivates the use of this model in anomaly detection. We construct a synthetic dataset of 1020 points, out of which 1000 points depict normal behavior and 20 points constitute outliers. The outliers are derived from a uniform distribution and the normal points are drawn from a mixture of a normal distribution and an uniform distribution.

Figure 4a shows the generated data along with the outliers

(marked in black), while Figure 4b reports the model and outliers estimated by q-GMM. The accuracy of the model is quite high in spite of the complicated distribution of the data.

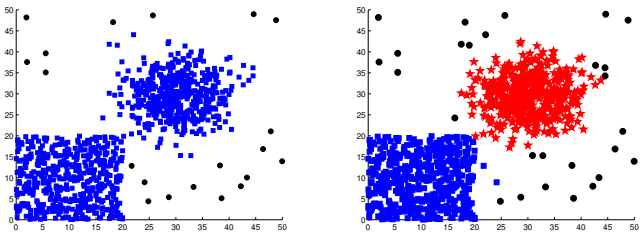


Fig. 4: Evidence of robustness of q-GMM for $q < 1$ in the presence of outliers.

B. Clustering benchmark datasets

We now focus on benchmark problems. To compare the performance of GMM and the proposed q-GMM in the context of clustering, we consider six popular datasets from UCI machine learning repository [8], and six gene expression datasets [3]. The datasets are listed in Table III, where one can see that the gene expression data sets are high dimensional. We also note that following the lines of [3], we normalize the gene expression data using z-score.

Name	# Instances	# Features	# classes
Parkinson	195	21	2
Breastcancer	699	9	2
Iris	150	4	3
Seeds	210	7	3
Wine	178	13	3
Haberman	306	3	2
chowdary-2006	104	182	2
shipp-2002-v1	77	798	2
singh-2002	102	339	2
chen-2002	179	85	2
nutt-v3	22	1152	2
alizadeh-v2	62	2093	3

TABLE III: Data sets used for clustering.

The results of clustering are presented in Table IV, where we report the corrected Rand index of the clusters obtained from GMM, q-GMM ($q < 1$) and q-GMM ($q > 1$). One can observe that in most cases, the best result (marked in bold) are achieved by q-GMM with $q < 1$, while the other two models provide inferior results. Though the performance of GMM and q-GMM ($q > 1$) are mostly similar, in certain cases q-GMM achieve higher Rand index due to the flexibility of the model.

C. Benchmark study on anomaly detection

We finally consider a benchmark for anomaly detection. In order to establish the accuracy of q-GMM anomaly detection algorithm, we study its performance on the KDD 1999 Classifier Learning Contest data set[1], which is a benchmark data set used for testing intrusion detection systems. We compare our results with the results of the winning entry.

Dataset	GMM	q-GMM $q < 1$	q-GMM $q > 1$
Parkinson	0.02	**0.21**	0.14
Breast Cancer	0.50	**0.77**	0.62
Iris	0.90	**0.92**	0.90
Seeds	0.72	**0.76**	0.71
Wine	0.40	**0.42**	0.40
Haberman	0.10	**0.15**	0.00
chowdary-2006	0.07	0.37	**0.93**
shipp-2002-v1	0.03	**0.05**	**0.05**
singh-2002	0.04	**0.05**	0.03
chen-2002	**0.67**	0.60	**0.67**
nutt-v3	**1.00**	**1.00**	**1.00**
alizadeh	**1.00**	0.95	0.96

TABLE IV: Corrected Rand index of clusters obtained from GMM, q-GMM ($q < 1$) and q-GMM ($q > 1$).

The raw training data was 4GB of compressed binary data (TCP dump) from 7 weeks of network traffic. This raw data was processed into 5 million connection entities which formed the training set. The test data was obtained similarly from 2 weeks of network traffic. Each network instance is a point in 41 dimensional space. There are about 38 attack types which are categorized into four sets

- DOS - Denial-of-Service
- R2L - Unauthorized access from remote machine
- U2R - Unauthorized access from local root user
- probing - surveillance and other probing

Furthermore, we note that the test data is not from the same probability distribution as the training data, which makes the scenario realistic. Another point of interest is that the training data contains only 24 attack types whereas the test data contains 38, which helps to assess the novelty detection ability of the classifiers.

We report the performance of anomaly detection using q-GMM in Table V. It would be useful to understand the statistical measure for assessing the performance of a classification algorithm. We call a classification test output as positive if it is identified as anomaly, and negative otherwise. In this terminology, true positive TP denotes the set of anomalies which are correctly identified, and false negative FN is the set of anomalies incorrectly identified as normal behavior. The Sensitivity or True Positive Rate of an algorithm is defined as

$$\text{Sensitivity} = \frac{|TP|}{|TP| + |FN|}\ , \tag{19}$$

which measures the proportion of actual positives which are correctly identified by the classifier. Form the perspective of

[1] Available at: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

an anomaly detector, FN should be as small as possible, and hence, sensitivity should be close to one. We compared the performance of q-GMM with that of the winning entry of KDD-99 challenge, where the results for the latter have been computed from [5][2]. We find that q-GMM provides superior performance than reported resutls.

Attack type	q-GMM			KDD best
	TP	FN	Sensitivity	Sensitivity
probe	2721	215	0.930	0.877
DOS	226828	382	0.998	0.977
U2R	163	10	0.940	0.263
R2L	3655	10156	0.264	0.103

TABLE V: Sensitivity results for q-GMM for the KDD-99 dataset.

One also needs to verify whether q-GMM raises a high number of false alarms even for normal samples. To check this, one needs to study the Specificity or True Negative Rate, defined as

$$\text{Specificity} = \frac{|TN|}{|TN| + |FP|} \tag{20}$$

that measures the proportion of actual negatives which are correctly identified by the classifier. Here FP are the false positives, that is, the set of normal samples incorrectly identified as anomalies. On the overall dataset, we found that q-GMM achieves a Specificity score of **0.924**, whereas the same for the winning KDD entry is **0.995**. Indeed q-GMM, with lower Specificity, is prone to raising more false alarms. However, the difference in scores is not drastic, and hence, in view of the practical importance truly identifying anomalies, q-GMM may be viewed as the superior approach in this comparison.

VI. Conclusions

In this paper we studied compact support q-Gaussian mixture models and proposed a method for ML estimation of parameters using EM algorithm. We explored two applications of this models: clustering and anomaly detection.

In the clustering scenario, the mixture model of q-Gaussian performed well on normal real data sets as given by the UCI repository. The robustness of the heavy tailed $q > 1$ distributions on the statistical representation of the data could be made out from the results obtained. But they were not convincing for higher dimensional datasets like gene expression profiles. Suitable hierarchical methods may work but we have not attempted the same in this work. Furthermore, we considered only problems with two or three clusters due to the computational time required for studying larger problems. Study of this approach on larger problems will be done in future.

Significant breakthrough was achieved in the application of q-Gaussian to anomaly detection problem. The inherent finite support property when $q < 1$ helped the cause of outlier elimination. The comparable results obtained in the KDD 1999 challenge data set proves the claim. Though we considered only a simple and crude model for intrusion detection, the results were in par with the results obtained by the winning entry.

[2]Also available at `http://cseweb.ucsd.edu/ elkan/clresults.html`

References

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[2] A. R. C. Fraley. How many clusters? which clustering method? answers via model-based clustering analysis. *The Computer Journal*, 1998.

[3] M. C. P. de Suoto, I. G. Costa, D. S. A. de Araujo, T. B. Ludermir, and A. Schliep. Clustering cancer gene expression data a comparative study. *BMC Bioinformatics*, 9:497–511, 2008.

[4] A. Dukkipati, S. Bhatnagar, and M. N. Murty. On measure theoretic aspects of nonextensive entropy functionals and corresponding maximum entropy prescriptions. *Physica A*, 384:758–774, 2007.

[5] C. Elkan. Results of the KDD'99 Classifier Learning. *SIGKDD Explorations*, 1(2):63–64, 2000.

[6] E. Eskin. Anomaly detection over noisy data using learned probability distributions. In *In Proceedings of Seventeenth International Conference on Machine Learning*, pages 252–262. Morgan Kauffman Publishers Inc., 2000.

[7] D. Ghoshdastidar, A. Dukkipati, and S. Bhatnagar. q-gaussian based smoothed functional algorithms for stochastic optimization. In *In Proceedings of IEEE International Symposium on Information Theory (ISIT'2012)*, pages 1059–1063. IEEE press, 2012.

[8] M. Lichman. UCI machine learning repository. http://archive.ics.uci.edu/ml, 2013.

[9] T. I. Lin, J. C. Lee, and S. Y. Yen. Finite mixture modelling using the skew normal distribution. *Statistica Sinica*, 17(2007):909–927, 2005.

[10] A. K. J. M. A. T. Figueiredo. Unsupervised learning of finite mixture models. In *IEEE Transactions on Pattern Analysis and Machine Learning*, volume 24, pages 1–16. IEEE Press, 2002.

[11] G. J. McLachlan and D. Peel. *Finite Mixture Models*. Wiley, 2000.

[12] D. Peel and G. J. McLachlan. Robust mixture modelling using the t distribution. *Statistics and Computing*, 510:339–348, 2000.

[13] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[14] M. Svensén and C. M. Bishop. Robust bayesian mixture modelling. *Neurocomputing*, 64:235–252, 2007.

[15] C. Tsallis. Possible generalization of boltzmann-gibbs statistics. *T. Stat. Phys.*, 52(479), 1988.