Asymmetric Gaussian and Its Application to Pattern Recognition

Tsuyoshi Kato, Shinichiro Omachi, and Hirotomo Aso

Graduate School of Engineering, Tohoku University, Sendai-shi, 980-8579 Japan
{kato,machi,aso}@aso.ecei.tohoku.ac.jp

Abstract. In this paper, we propose a new probability model, 'asymmetric Gaussian(AG),' which can capture spatially asymmetric distributions. It is also extended to mixture of AGs. The values of its parameters can be determined by Expectation-Conditional Maximization algorithm. We apply the AGs to a pattern classification problem and show that the AGs outperform Gaussian models.

1 Introduction

Estimation of a probability density function(pdf) of the patterns in given data set is a very important task for pattern recognition [1], data mining and so on. Single Gaussian and mixtures of Gaussians are most popular probability models, and they are used for many applications [2]. However, they do not always fit any distribution of patterns, so it is meaningful to provide another probability model which can be chosen instead of single/mixture Gaussian model.

In this paper, we propose a new probability model, 'asymmetric Gaussian(AG),' which is an extension of Gaussian. The AG can capture spatially asymmetric distributions. In the past, 'Asymmetric Mahalanobis Distance(AMD), ' was introduced [3] and it was applied to handwritten Chinese and Japanese character recognition. The AMD can measure a spatially asymmetrical distance between an unknown pattern and the mean vector of a class and shows excellent classification performance. However, the AMD is suitable only for an unimodal distribution, so the range of its application is necessarily somewhat limited. Meanwhile, since our model is formulated by a density function, it is easy to be extended to mixture model, which can capture multi-modal distributions. Moreover, due to its probabilistic formulation, we can develop a wide variety of extensions in a theoretically well-appointed setting.

The remainder of the paper is organized as follows. In the next section, we introduce the concept of latent variable model of single Gaussian model. In section 3 we then propose the AG model by extending the framework to the asymmetric version. Next we extend the AG to mixture models in Section 4. Section 5 presents its maximum likelihood estimation algorithm. In section 6 we show empirically that the mixture of AGs captures clusters of patterns, each of which are distributed asymmetrically. In section 7 we apply AG models to pattern recognition and present results using a real-world data sets. The final section presents our conclusions.

T. Caelli et al. (Eds.): SSPR&SPR 2002, LNCS 2396, pp. 405-413, 2002.

2 A View of Single Gaussian

In this section we introduce a view of single Gaussian by a latent variable model. The goal of the latent variable model is to extend the representation for asymmetric distribution. We consider that single Gaussian has a d-dimensional latent variable $\boldsymbol{z}$ related to an observed data $\boldsymbol{x}$ in d-dimensional space. The i-th element of the latent variable, z_i, is distributed according to the following normal distribution $\mathcal{N}(z_i; \mu_i^z, \sigma_i^2)$ with mean μ_i^z and variance σ_i^2:

$$\mathcal{N}(z_i; \mu_i^z, \sigma_i^2) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(z_i - \mu_i^z)^2}{2\sigma_i^2}\right). \tag{1}$$

The Gaussian-distributed observed data vector $\boldsymbol{x}$ is generated by rotating $\boldsymbol{z}$ via an orthonormal matrix $\Phi = [\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_d] \in \mathcal{R}^{d\times d}$ as follows:

$$\boldsymbol{x} = \Phi \boldsymbol{z}. \tag{2}$$

The pdf of the observed variable $\boldsymbol{x}$ is consequently given by:

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}|\boldsymbol{z}) \prod_{i=1}^{d} \mathcal{N}(z_i; \mu_i^z, \sigma_i^2) dz_i \tag{3}$$

$$= \prod_{i=1}^{d} \mathcal{N}(\boldsymbol{\phi}_i^T \boldsymbol{x}; \mu_i^z, \sigma_i^2). \tag{4}$$

The last equality follows because the conditional density of $\boldsymbol{x}$ given $\boldsymbol{z}$ is $p(\boldsymbol{x}|\boldsymbol{z}) = \prod_{i=1}^{d} \delta(\boldsymbol{\phi}_i^T \boldsymbol{x} - z_i)$ where $\delta(\cdot)$ is the Dirac's delta function.

Next, we show an arbitrary Gaussian can be represented by the latent variable model. The observation variable $\boldsymbol{x}$ is assumed to be distributed according to a Gaussian $\mathcal{N}(\boldsymbol{\mu}^x, \Sigma^x)$ (the mean is $\boldsymbol{\mu}^x$ and the covariance matrix is Σ^x). The pdf of the Gaussian can be rewritten as

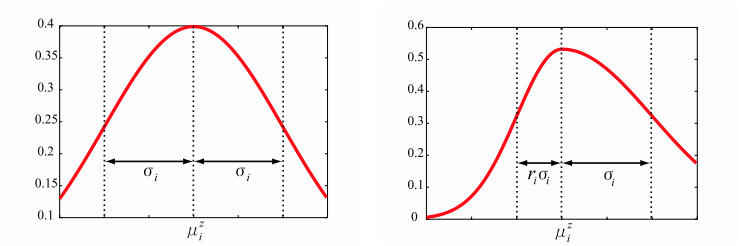
$$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}^x, \Sigma^x) = \prod_{i=1}^{d} \mathcal{N}(\boldsymbol{\psi}_i^T \boldsymbol{x}; \boldsymbol{\psi}_i^T \boldsymbol{\mu}^x, \lambda_i) \tag{5}$$

where λ_i and $\boldsymbol{\psi}_i$ denote i-th eigenvalue of the covariance matrix Σ^x and the corresponding eigenvector, respectively. By comparison between the formulae (4) and (5), it is shown that the above-mentioned latent variable model represents any Gaussian distribution by letting $\boldsymbol{\phi}_i = \boldsymbol{\psi}_i$, $\mu_i^z = \boldsymbol{\psi}_i^T \boldsymbol{\mu}^x$, $\sigma_i^2 = \lambda_i$.

3 Asymmetric Gaussian

We now introduce an asymmetric Gaussian(AG) model by extending the latent variable model.

In the same manner as Gaussian, the d-dimensional AG has a latent variable $\boldsymbol{z} \in \mathcal{R}^d$ and the observation variable $\boldsymbol{x}$ is modeled using $\boldsymbol{z}$ and an orthonormal



(a)Univariate Gaussian (b)Univariate asymmetric Gaussian

Fig. 1. Univariate Gaussian and univariate asymmetric Gaussian.

matrix $\Phi \in \mathcal{R}^{d\times d}$: $\boldsymbol{x} = \Phi \boldsymbol{z}$. The different point between the AG and the Gaussian is the distribution of the latent variable $\boldsymbol{z}$. We choose the following distribution of each element of $\boldsymbol{z}$:

$$\mathcal{A}(z_i; \mu_i^z, \sigma_i^2, r_i) \equiv \frac{2}{\sqrt{2\pi}} \frac{1}{\sqrt{\sigma_i^2}(r_i+1)} \begin{cases} \exp\left(-\frac{(z_i-\mu_i^z)^2}{2\sigma_i^2}\right) & \text{if } z_i > \mu_i^z, \\ \exp\left(-\frac{(z_i-\mu_i^z)^2}{2r_i^2\sigma_i^2}\right) & \text{otherwise,} \end{cases} \tag{6}$$

where μ_i^z,σ_i^2 and r_i are parameters of $\mathcal{A}(z_i; \mu_i^z, \sigma_i^2, r_i)$. We term the density model (6) 'univariate asymmetric Gaussian(UAG).' It is shown that UAG have an asymmetric distribution by the Figure 1(b) where the density function is plotted. In addition, UAG is an extension of Gaussian since UAG with $r_i = 1$ is equivalent to Gaussian.

The pdf of AG is given by:

$$p(\boldsymbol{x}) = \mathcal{A}(\boldsymbol{x}; \Theta) \equiv \int p(\boldsymbol{x}|\boldsymbol{z}) \prod_{i=1}^{d} \mathcal{A}(z_i; \mu_i^z, \sigma_i^2, r_i) dz_i \tag{7}$$

$$= \prod_{i=1}^{d} \mathcal{A}(\boldsymbol{\phi}_i^T \boldsymbol{x}; \mu_i^z, \sigma_i^2, r_i), \tag{8}$$

where $\Theta = \{\boldsymbol{\phi}_i, \mu_i^z, \sigma_i^2, r_i\}_{i=1}^d$ is the set of the adaptive parameters.

4 Mixture of Asymmetric Gaussians

Due to the definition of the density model, it is straightforward to consider a mixture of AG, which is able to model complex data structures with a linear combination of local AGs. The overall density of the K-component mixture model is written by

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{A}(\boldsymbol{x}; \Theta^{(k)}) \tag{9}$$

where $\mathcal{A}(\boldsymbol{x};\Theta^{(k)})$ is the kth local AG, with its own set of independent parameters, $\Theta^{(k)} = \{\boldsymbol{\phi}_{i,k}, \mu^z_{i,k}, \sigma^2_{i,k}, r_{i,k}\}_{i=1}^d$, and $\{\pi_k\}_{k=1}^K$ are mixing proportions satisfying $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^K \pi_k = 1$.

5 The EM Algorithm for Maximum Likelihood Estimation

Optimal values of the parameters of each local AG, $\{\Theta^{(k)}\}$, and mixing proportions $\{\pi_k\}$ are unable to be obtained in the closed form, and here we describe the formulae using Expectation-Maximization(EM) algorithm [4], [5] which provides a numerical method for estimating these maximum likelihood parameters. Given a data set $\{\boldsymbol{x}^n\}_{n=1}^N$, the log-likelihood function is given by

$$\mathcal{L} = \sum_{n=1}^{N} \log \sum_{k=1}^{K} \left(\pi_k \mathcal{A}(\boldsymbol{x}^n; \Theta^{(k)}) \right). \tag{10}$$

The maximization of the log-likelihood can be regarded as a missing-data problem in which the identity k of the component that has generated each pattern $\boldsymbol{x}^n$ is unknown.

In the E-step, we compute the posterior probability h_{nk}, called responsibility, of each local AG component k for generating pattern $\boldsymbol{x}^n$ using the current values of $\Theta^{(k)}$ and π_k:

$$h_{nk} = \hat{P}(k|\boldsymbol{x}^n) = \frac{\pi_k \mathcal{A}(\boldsymbol{x}^n; \Theta^{(k)})}{\sum_{k'} \pi_{k'} \mathcal{A}(\boldsymbol{x}^n; \Theta^{(k')})}. \tag{11}$$

In the M-step, the quantity of the expected complete-data log-likelihood which is given by

$$\langle \mathcal{L}_{\text{comp}} \rangle = \sum_{n=1}^{N} \sum_{k=1}^{K} h_{nk} \left(\log \mathcal{A}(\boldsymbol{x}^n; \Theta^{(k)}) + \log \pi_k \right) \tag{12}$$

is maximized with respect to $\{\Theta^{(k)}, \pi_k\}_{k=1}^K$. The following updates of $\{\pi_k\}$ maximize the quantity of the term containing $\{\pi_k\}$ in (12) with subject to the constraint $\sum_{k=1}^K \pi_k = 1$:

$$\pi_k = \frac{1}{N} \sum_{n=1}^{N} h_{nk}. \tag{13}$$

Although the parameter set of each local AG, $\Theta^{(k)} = \{\boldsymbol{\phi}_{i,k}, \mu^z_{i,k}, \sigma^2_{i,k}, r_{i,k}\}_{i=1}^d$, must also be found so that it maximizes the expected complete-data log-likelihood in the standard EM algorithm, it is not tractable to compute both Φ^k and the other parameters simultaneously. We therefore use a two-stage procedure. In the first stage of the M-step, $\bigcup_{i,k} \{\mu^z_{i,k}, \sigma^2_{i,k}, r_{i,k}\}$ is held constant, and the orthonormal matrix $\Phi^k = \{\phi_{i,k}\}$ is updated so as to increase $\langle \mathcal{L}_{\text{comp}} \rangle$ in (12). In

the second stage, we find the optimal parameters of each UAG in each local AG, $\mu^z_{i,k}$,$\sigma^2_{i,k}$ and $r_{i,k}$, keeping the orthonormal matrix Φ^k constant. This procedure performs only partial maximization, however, the partial maximization of $\langle \mathcal{L}_{\rm comp} \rangle$ also guarantees the log-likelihood not to decrease during each iteration. Such a strategy is called generalized Expectation-Maximization(GEM) algorithm [4], [6]. The proposed maximum likelihood(ML) estimation scheme is an example of Expectation-Conditional Maximization(ECM) algorithm [7], which is a subclass of GEM algorithms. Further details concerning the two-stage procedure can be seen in Appendix.

The ML estimation algorithm is summarized as follows:

begin
  repeat
  { E-step }
  Evaluate responsibilities (11);
  { M-step }
  Update mixing proportions using (13);
  foreach $\forall k$ **begin**
    Update the orthonormal matrix Φ^k with $\{\mu^z_{i,k}, \sigma^2_{i,k}, r_{i,k}\}^d_{i=1}$ fixed;
    Find the optimal values of $\{\mu^z_{i,k}, \sigma^2_{i,k}, r_{i,k}\}^d_{i=1}$ with Φ^k fixed
  end;
  until the convergence of $\mathcal{L}$
end.

6 Simulations

We applied the ML estimation algorithm mentioned in the previous section for AG model to a problem involving 229 hand-crafted data points in the 2-dimensional space shown in Figure 2.

Figure 2(b) shows the results using three components. We also fitted the mixture of (standard) Gaussians for comparison (Figure 2(a)). The ellipse in (a) denotes the set of points that have the same Mahalanobis distance from the mean of each component, and the cross point in each ellipse lies on the mean. Similarly the loop in (b) denotes the set of points satisfying the values of the exponent of each local AG equal to one, and the cross point in each ellipse lies on the point $(\mu^z_{1,k}, \mu^z_{0,k})$. The AG captures the asymmetric distribution, which cannot be done by the Gaussian intrinsically. Although it might seem that AG tends to over-fit to the data set, we expect that this problem could be overcome by evidence framework [8].

7 Application to Pattern Recognition

In this section, we first present how to apply mixture of AGs to pattern recognition, and then show the experimental results on character recognition problem.

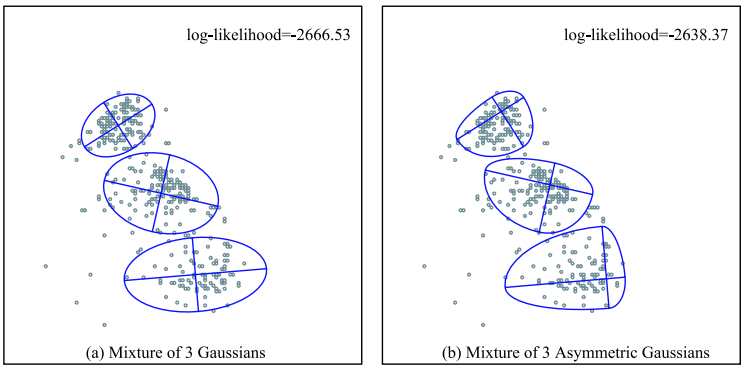


Fig. 2. Comparison between Mixture of Gaussians and Mixture of Asymmetric Gaussians.

In the training stage, we estimate the density function of each class w, $p(\boldsymbol{x}|w)$, using the ML estimation algorithm. In the classification stage, we find the class which has the largest posterior class probability:

$$P(w|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|w)P(w)}{\sum_{w'} p(\boldsymbol{x}|w')P(w')} \tag{14}$$

where the prior class probability $P(w)$ is assumed to be non-informative.

We have tested the method in the public database 'Letter' [9] obtained from the UCI Machine Learning repository. The data contain 20,000 instances extracted from character images. Each of them has 16 features. The number of classes is 26. The database is partitioned into five almost equal subsets. In rotation, four subsets are used to train the AG parameters of each class and the trained AGs are tested on the remaining subset. In this experiment, we choose $K = 1$ for each class, that is, non-mixture AG models are used. For comparison, we also test Gaussians.

The accuracy on each subset is plotted in Figure 3. The 'average' in the figure is obtained by the ratio of the sum of the numbers classified correctly on each subset to the number of all instances. AGs improve in classification performance on every subset and AGs obtain 88.14% 'average' accuracy while Gaussians obtain 87.71%. It can be considered that AGs capture the distribution of patterns more precisely than Gaussians.

8 Conclusion

In this paper, we proposed a new probability density model, asymmetric Gaussian, which can fit the spatially asymmetric distribution, and extended it to mixture model. We also developed an algorithm of the maximum likelihood estimation for mixture of AGs using the Expectation-Conditional Maximization technique and it was applied to a two-dimensional problem. We also applied the

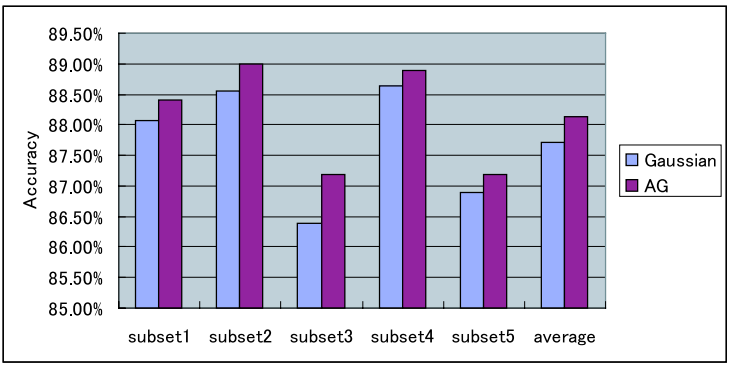


Fig. 3. Experimental results on the database 'Letter'.

AGs to character classification problem and showed that the AGs outperform Gaussian models.

Appendix: M-Step in the EM Algorithm

We now describe the details about how to update the parameters of mixture of AGs, $\Theta^{(k)}$, in the M-step. We use a two-stage procedure to update $\Theta^{(k)}$ which increases the expected complete-data log-likelihood function. The two-stage procedure runs as follows: (1) Update the orthonormal matrix Φ^k with remaining parameters $\{\mu^z_{i,k}, \sigma^2_{i,k}, r_{i,k}\}_{i=1}^d$ fixed. (2) Update $\{\mu^z_{i,k}, \sigma^2_{i,k}, r_{i,k}\}_{i=1}^d$ with Φ^k fixed.

(1) Update Φ^k with $\{\mu^z_{i,k}, \sigma^2_{i,k}, r_{i,k}\}_{i=1}^d$ fixed

We compute Φ^k_{new} as follow:

$$\Phi^k_{\text{new}} = \Phi^k_{\text{old}} + \eta \left. \frac{\partial \langle \mathcal{L}_{\text{comp}} \rangle}{\partial \Phi^k} \right|_{\Phi^k = \Phi^k_{\text{old}}} \tag{15}$$

where η is the learning constant and Φ^k_{old} denotes the old value of Φ^k. Note that there is no constraint to ensure that Φ^k_{new} in (15) will result in an orthonormal matrix. Therefore, after updating, we modify Φ^k_{new} by using Gram-Schmidt orthonormalization procedure. Then the log-likelihood $\mathcal{L}$ using Φ^k_{new} is evaluated. If $\mathcal{L}$ improves, Φ^k_{new} is chosen as the new value of Φ^k. If not, Φ^k is not updated.

(2) Update $\{\mu^z_{i,k}, \sigma^2_{i,k}, r_{i,k}\}$ with Φ^k fixed

The expected complete-data log-likelihood function can be factorized by $Q_{i,k}$'s:

$$\langle \mathcal{L}_{\text{comp}} \rangle = \left(\sum_{k=1}^{K} \sum_{i=1}^{d} Q_{i,k} \right) + \left(\sum_{n=1}^{N} \sum_{k=1}^{K} h_{nk} \log \pi_k \right), \tag{16}$$

where

$$Q_{i,k} = \sum_{n=1}^{N} h_{nk} \log \mathcal{A}\left((\boldsymbol{\phi}_i^k)^T \boldsymbol{x}^n;\ \mu_{i,k}^z, \sigma_{i,k}^2, r_{i,k}\right). \tag{17}$$

Note that $Q_{i,k}$ depends only on three parameters, $\mu_{i,k}^z$, $\sigma_{i,k}^2$ and $r_{i,k}$. The above factorization permits us to find the optimal values of $\{\mu_{i,k}^z, \sigma_{i,k}^2, r_{i,k}\}$ separately so that $Q_{i,k}$ is maximized. However, it is intractable to maximize $Q_{i,k}$ with respect to the triple $\{\mu_{i,k}^z, \sigma_{i,k}^2, r_{i,k}\}$ simultaneously. So each of $Q_{i,k}$ is maximized sequentially with respect to each of parameters by the following iterative scheme:

begin

repeat

Find the optimal value of $\mu_{i,k}^z$ with $\sigma_{i,k}^2$ and $r_{i,k}$ fixed;

Find the optimal value of $r_{i,k}$ with $\sigma_{i,k}^2$ and $\mu_{i,k}^z$ fixed;

Find the optimal value of $\sigma_{i,k}^2$ with $\mu_{i,k}^z$ and $r_{i,k}$ fixed;

until the convergence of $Q_{i,k}$

end.

Each maximization step is performed by finding the value of $\mu_{i,k}^z, \sigma_{i,k}^2, r_{i,k}$ so that $\frac{\partial Q_{i,k}}{\partial \mu_{i,k}^z} = 0$ $\frac{\partial Q_{i,k}}{\partial r_{i,k}} = 0$ and $\frac{\partial Q_{i,k}}{\partial \sigma_{i,k}^2} = 0$ are satisfied, respectively. It is straightforward to maximize $Q_{i,k}$ with respect to $\mu_{i,k}^z, \sigma_{i,k}^2$ because the equations are linear. $r_{i,k}$ is optimized by Newton-Raphson method [10] since the equation $\frac{\partial Q_{i,k}}{\partial r_{i,k}} = 0$ is non-linear.

References

1. T. Kato, S. Omachi and H. Aso: "Precise hand-printed character recognition using elastic models via nonlinear transformation", Proc. 15th ICPR, Vol. 2, pp. 364–367 (2000).
2. Z. R. Yang and M. Zwolinski: "Mutual information theory for adaptive mixture models", IEEE Trans. PAMI, **23**, 4, pp. 396–403 (2001).
3. N. Kato, M. Suzuki, S. Omachi, H. Aso and Y. Nemoto: "A handwritten character recognition system using directional element feature and asymmetric Mahalanobis distance", IEEE Trans. PAMI, **21**, 3, pp. 258–262 (1999).
4. A. P. Dempster, N. M. Laird and D. B. Rubin: "Maximum likelihood from incomplete data via the EM algorithm", J.R. Statistical Society, Series B, **39**, pp. 1–38 (1977).
5. C. M. Bishop: "Neural network for pattern recognition", Oxford, England: Oxford University Press (1995).
6. R. M. Neal and G. E. Hinton: "A view of the EM algorithm that justifies incremental, sparse, and other variants", Learning in Graphical Models (Ed. by M. I. Jordan), Kluwer Academic Publishers, pp. 355–368 (1998).
7. X. L. Meng and D. B. Rubin: "Recent extensions of the EM algorithms", Bayesian Statistics (Eds. by J. M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith), Vol. 4, Oxford (1992).
8. D. J. C. MacKay: "Bayesian interpolation", Neural Computation, **4**, 3, pp. 415–447 (1992).

9. P. W. Frey and D. J. Slate: "Letter recognition using holland-style adaptive classifiers", Machine Learning, **6**, 2 (1991).
10. W. H. Press, S. A. Teukolski, W. T. Vetterling and B. P. Flannery: "Numerical Recipes in C", Cambridge University Press (1988).