THE GENERALIZED GAUSSIAN MIXTURE MODEL USING ICA

Te-Won Lee

Institute for Neural Computation,
University of California, San Diego,
9500 Gilman Dr.
La Jolla, California 92093-0523, USA
tewon@inc.ucsd.edu

Michael S. Lewicki

Computer Science Department &
Center for the Neural Basis of Cognition
Carnegie Mellon University
4400 Fifth Ave.
Pittsburgh, PA 15213
lewicki@cs.cmu.edu

ABSTRACT

An extension of the Gaussian mixture model is presented using Independent Component Analysis (ICA) and the generalized Gaussian density model. The mixture model assumes that the observed data can be categorized into mutually exclusive classes whose components are generated by a linear combination of independent sources. The source densities are modeled by generalized Gaussians (Box and Tiao, 1973) that provide a general method for modeling non-Gaussian statistical structure of univariate distributions that have the form $p(x) \propto \exp(-|x|^q)$. By inferring q, a wide class of statistical distributions can be characterized including uniform, Gaussian, Laplacian, and other sub- and super-Gaussian densities. The generalized Gaussian mixture model using ICA infers for each class the source parameters, the basis functions and bias vectors. The new method can improve classification accuracy compared with standard Gaussian mixture models and shows promise for accurately modeling structure in high-dimensional data.

1. INTRODUCTION

In pattern classification, the performance of a method is often determined by how well it can model the underlying statistical distribution of the data. One recent example of this is independent component analysis (ICA). The success of ICA on problems such as blind source separation and signal analysis results directly from its ability to model non-Gaussian statistical structure. If the source distributions are assumed to be Gaussian, this technique is equivalent to principal component analysis (PCA). PCA assumes the data to be distributed according to a multivariate Gaussian. In contrast, ICA assumes that the source distributions are non-Gaussian allowing modeling non-Gaussian structure, e.g., platykurtic or leptokurtic probability density functions. In many applications of ICA, the form of the source distribution (or equivalently the "nonlinearity") is fixed. More recent work has extended these results so that the form of distribution can also be inferred from the data for example, using Gaussian mixtures (Attias, 1999) or mixtures of sub-Gaussian and super-Gaussian densities (Lee et al., 1999a).

In many pattern recognition problems there are data clusters in which each cluster can be fitted by non-Gaussian distributions. To model the ensemble of data classes a mixture model (Duda and Hart, 1973) is considered where the observed data can be categorized into several mutually exclusive classes. When the class variables are modeled as multivariate Gaussian densities, it is called a Gaussian mixture model. This model can be generalized by assuming that the sources in each class are independent and non-Gaussian. In Lee et al. (1999b) we modeled the underlying source density by two predefined non-Gaussian densities (super and sub-Gaussian) as used in the extended infomax algorithm. A binary parameter switched from a sub- or super-Gaussian density.

In this paper, we are interested in modeling a continuously defined parametric form of the underlying density for each class. The exponential power distribution (Box and Tiao, 1973)[1] is used to model distributions that deviate from normality. They provide a general method for modeling non-Gaussian statistical structure of univariate distributions that have the form $p(x) \propto \exp(-|x|^q)$. By inferring q, a wide class of statistical distributions can be characterized including uniform, Gaussian, Laplacian, and other sub- and super-Gaussian densities. This formulation of a mixture model using the generalized Gaussian contains as special the Gaussian mixture model when all the source

[1] also called a generalized Laplacian or generalized Gaussian.

densities are restricted to be Gaussian. Using this distribution in ICA, we show that the generalized Gaussian mixture model can be used to infer the degree of non-Gaussian statistical structure for classes of multivariate densities. This can be applied to situations where multiple classes exist with unknown source densities.

This paper is organized as follows: We present the generalized Gaussian model and show how to infer the parameters for this density model. This model is used for capturing the densities of the unknown sources in ICA. Finally, this ICA model is extended to multiple classes yielding in the generalized Gaussian mixture model using ICA. We demonstrate in simulations that this model can improve classification accuracy compared with standard Gaussian mixture models and shows promise for accurately modeling structure in high dimensional data.

2. THE GENERALIZED GAUSSIAN MODEL

The generalized Gaussian is used to model distributions that deviate from normality. In its simplest form, this distribution is

$$p(x) \propto \exp(-\frac{1}{2}|x|^q)\,. \tag{1}$$

By varying the exponent q, it is possible to describe Gaussian, platykurtic, and leptokurtic distributions. Using $q = 2/(1+\beta)$, Box and Tiao (1973) expressed this distribution in the following general form

$$p(x|\mu,\sigma,\beta) = \frac{\omega(\beta)}{\sigma}\exp\left[-c(\beta)\left|\frac{x-\mu}{\sigma}\right|^{2/(1+\beta)}\right], \tag{2}$$

where

$$c(\beta) = \left[\frac{\Gamma[\frac{3}{2}(1+\beta)]}{\Gamma[\frac{1}{2}(1+\beta)]}\right]^{1/(1+\beta)} \tag{3}$$

and

$$\omega(\beta) = \frac{\Gamma[\frac{3}{2}(1+\beta)]^{1/2}}{(1+\beta)\Gamma[\frac{1}{2}(1+\beta)]^{3/2}}, \quad \sigma > 0, \tag{4}$$

In this form, the data's mean and standard deviation are given by μ and σ, respectively. The parameter β is a measure of kurtosis and controls the distribution's deviation from normality.[2] When $\beta = 0$, the distribution is the standard normal; it is a Laplacian (or double exponential) for $\beta = 1$. As $\beta \to -1$, the distribution becomes uniform over the unit interval. As $\beta \to \infty$, the distribution a delta function at zero. The parameter β can also be converted to the standard kurtosis measure $\gamma_2 = E(x-\mu)^4/\sigma^4 - 3$. For the exponential power distribution, this relation is

$$\gamma_2 = \frac{\Gamma[\frac{5}{2}(1+\beta)]\Gamma[\frac{1}{2}(1+\beta)]}{\Gamma[\frac{5}{2}(1+\beta)]^2} - 3. \tag{5}$$

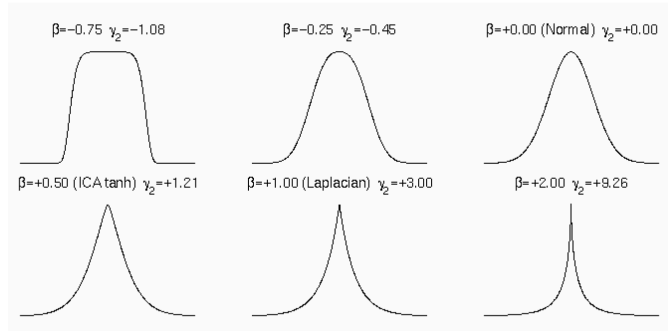

Figure 1: Exponential power distributions for various values β. The parameter β is also a measure of the distribution's kurtosis, and varies with the standard kurtosis measure, γ_2.

Figure 2 shows examples of the exponential power distribution for various values of β and the corresponding values of γ_2. In addition to the standard normal and the Laplacian, also shown is the distribution labeled "ICA tanh" which shows the best-fitting exponential power distribution to the implied prior distribution under the widely-used tanh non-linearity in ICA. The form of this prior is $p(x) = \cosh(bx)^{a/b}/Z$ where $Z = \pi^{-1/2}\Gamma(a/2b)/\Gamma((a+b)/2b)$. The fit by the exponential power distribution (for $a = b = 1$) is almost exact, differing by having a slightly shaper peak, and yielding a Kullback-Leibler divergence of 0.0007 for the optimum $\beta = 0.495$ and $\sigma = 1.525$.

2.1. Estimating β

For the purposes of finding the basis functions and the β parameter in ICA, zero mean and unit variance is assumed. The problem then becomes to estimate the value of β from the data. This can be accomplished by simply finding the maximum posteriori value of β. The posterior distribution of β given the observations $\mathbf{x} = \{x_1, \ldots, x_N\}$ is

$$p(\beta|\mathbf{x}) \propto p(\mathbf{x}|\beta)p(\beta)\,, \tag{6}$$

[2]Box and Tiao (1973) considered the case for β over the range $[-1, 1]$, but the distributions are also valid for the more general case for $\beta > 1$.

where the data likelihood is

$$p(\mathbf{x}|\beta) = \prod_n \omega(\beta) \exp\left[-c(\beta)|x_n|^{2/(1+\beta)}\right], \qquad (7)$$

and $p(\beta)$ defines the prior distribution for β. Because $\beta > -1$, it is convenient to use $p(\beta) \sim \text{Gamma}(1+\beta|a,b)$. Choosing the values $a = 2$ and $b = 2$ gives a broad prior distribution with a 95% density range of $[-0.5, 10.5]$, which is sufficient for our purposes here. See Box and Tiao (1973) for further discussion on inference with the exponential power distribution.

3. THE GENERALIZED GAUSSIAN ICA

The objective of ICA is to infer both the unknown sources and the unknown basis functions (mixing matrix) from the data signal (Jutten and Herault, 1991; Comon, 1994; Bell and Sejnowski, 1995). This problem can be formulated explicitly as one of density estimation (Pearlmutter and Parra, 1996; MacKay, 1996; Cardoso, 1997). The data likelihood is derived by marginalizing over the sources

$$p(\mathbf{x}|\mathbf{A}) = \int p(\mathbf{x}|\mathbf{s}, \mathbf{A}) p(\mathbf{s})\, d\mathbf{s}. \qquad (8)$$

Because there is a unique expression for the data in terms of the sources, $\mathbf{s} = \mathbf{A}^{-1}\mathbf{x}$, the conditional likelihood is a delta function

$$p(\mathbf{x}|\mathbf{s}, \mathbf{A}) = \delta(\mathbf{x} - \mathbf{A}\mathbf{s}). \qquad (9)$$

In this case, the expression for the data likelihood is

$$p(\mathbf{x}|\mathbf{A}) = \frac{p(\mathbf{s})}{|\det \mathbf{A}|}. \qquad (10)$$

Performing gradient ascent on this expression gives a rule for learning the mixing matrix, $\mathbf{A}$

$$\Delta\mathbf{A} \propto \mathbf{A}\mathbf{A}^T \frac{\partial}{\partial \mathbf{A}} \log p(\mathbf{x}|\mathbf{A}) = -\mathbf{A}(\varphi(\mathbf{s})\mathbf{s}^T - \mathbf{I}). \qquad (11)$$

where the prefactor $\mathbf{A}\mathbf{A}^T$ is used to obtain the natural gradient solution (Amari et al., 1996) which gives an ascent direction that is insensitive to rescalings of the data. The vector $\varphi(\mathbf{s})$ is a function of the prior and is defined by $\varphi(\mathbf{s}) = \frac{\partial \log p(\mathbf{s})}{\partial \mathbf{s}}$. In the case of the exponential power distribution (eq.2), for $p(\mathbf{s})$ we have

$$\varphi_i(s_i) = -\eta |s_i - \mu_i|^{q-1} q c \sigma_i^{-q}, \qquad (12)$$

where $\eta = \text{sign}(s_i - \mu_i)$, $q = 2/(1+\beta_i)$, and $c = [\Gamma(3/q)/\Gamma(1/q)]^{q/2}$. Details of the learning rule derivation are given in Lewicki (2000). Figure 2 shows examples of the fitting two dimensional distributions with the ICA-exponential power model. The values of β were estimated periodically during learning by maximizing the posterior (eq.6).

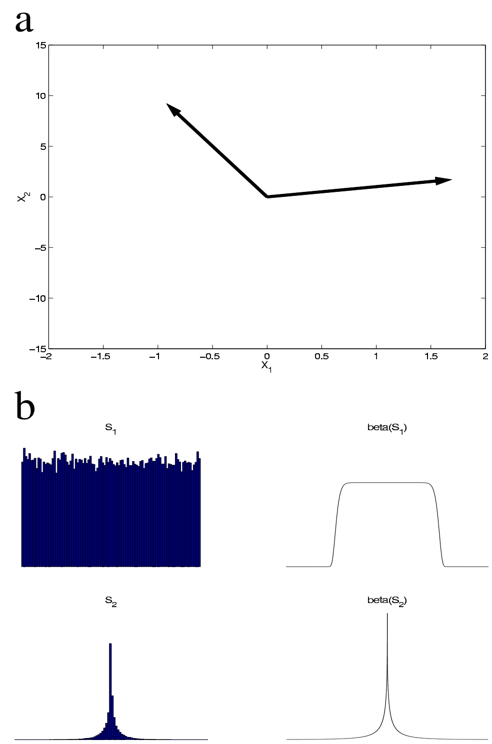

Figure 2: Fitting independent components of two dimensional distributions using generalized Gaussian source models. The scatter plot (a) shows the data distributions. The arrows indicate the learned basis functions (rescaled). The histograms of the distribution of coefficients and the inferred values of the exponential power parameter β are shown in (b). The example shows a mixture of super- and sub-Gaussian sources in which the true values of β were -1 and +4. The inferred β values were -0.89 and 3.78 respectively.

4. THE GENERALIZED MIXTURE MODEL USING ICA

A mixture density is defined as (Duda and Hart, 1973):

$$p(\mathbf{x}_n|\Theta) = \sum_{k=1}^{K} p(\mathbf{x}_n|C_k, \theta_k) p(C_k), \qquad (13)$$

where $\Theta = (\theta_1, \cdots, \theta_K)$ are the unknown parameters $(\mathbf{A}_k, \mathbf{b}_k, \beta_k)$ for the component densities $p(\mathbf{x}_n|C_k, \theta_k)$. The likelihood of the data is the joint density

$$p(\mathbf{X}|\Theta) = \prod_{n=1}^{N} p(\mathbf{x}_n|\Theta). \qquad (14)$$

We assume that $p(\mathbf{X}|\Theta)$ is a differentiable function of Θ. The log-likelihood L is then

$$L = \sum_{n=1}^{N} \log p(\mathbf{x}_n|\Theta) \tag{15}$$

and the gradient for the parameters of each class k is

$$\begin{aligned}
\nabla_{\theta_k} L &= \sum_{n=1}^{N} \frac{1}{p(\mathbf{x}_n|\Theta)} \nabla_{\theta_k} p(\mathbf{x}_n|\Theta) \\
&= \sum_{n=1}^{N} \frac{\nabla_{\theta_k} \left[\sum_{k=1}^{K} p(\mathbf{x}_n|C_k,\theta_k) p(C_k)\right]}{p(\mathbf{x}_n|\Theta)} \\
&= \sum_{n=1}^{N} \frac{\nabla_{\theta_k} p(\mathbf{x}_n|C_k,\theta_k) p(C_k)}{p(\mathbf{x}_n|\Theta)}. \qquad (16)
\end{aligned}$$

Using the Bayes relation, the class probability for a given data vector $\mathbf{x}_n$ is

$$p(C_k|\mathbf{x}_n,\Theta) = \frac{p(\mathbf{x}_n|\theta_k,C_k)p(C_k)}{\sum_k p(\mathbf{x}_n|\theta_k,C_k)p(C_k)}. \tag{17}$$

Substituting eq.17 in eq.16 leads to

$$\begin{aligned}
\nabla_{\theta_k} L &= \sum_{n=1}^{N} p(C_k|\mathbf{x}_n,\Theta) \frac{\nabla_{\theta_k} p(\mathbf{x}_n|\theta_k,C_k)p(C_k)}{p(\mathbf{x}_n|\theta_k,C_k)p(C_k)} \\
&= \sum_{n=1}^{N} p(C_k|\mathbf{x}_n,\Theta) \nabla_{\theta_k} \log p(\mathbf{x}_n|C_k,\theta_k). (18)
\end{aligned}$$

The log likelihood function in eq.18 is the log likelihood for each class. For the present model, the class log likelihood is given by the log likelihood for the standard ICA model:

$$\begin{aligned}
\log p(\mathbf{x}_n|\theta_k,C_k) &= \log \frac{p(\mathbf{s}_t)}{|\det \mathbf{A}_k|} \qquad (19) \\
&= \log p(\mathbf{A}_k^{-1}(\mathbf{x}_n - \mathbf{b}_k)) - \log|\det \mathbf{A}_k|.
\end{aligned}$$

Gradient ascent is used to estimate the parameters that maximize the log Likelihood. The gradient parameters for each class are the gradient of the basis functions and the gradient of the bias vector $\nabla_{\theta_k} L = \{\nabla_{\mathbf{A}_k} L, \nabla_{\mathbf{b}_k} L, \nabla_{\beta_k} L\}$. We consider each in turn.

4.1. Estimating the basis functions

Adapt the basis functions for each class $\mathbf{A}_k$ with eq.18.

$$\nabla_{\mathbf{A}_k} L = \sum_{n=1}^{N} p(C_k|\mathbf{x}_n,\Theta) \nabla_{\mathbf{A}_k} \log p(\mathbf{x}_n|C_k,\theta_k). \tag{20}$$

The adaptation is performed by using gradient ascent with the gradient of the component density with respect to the basis functions giving

$$\Delta \mathbf{A}_k \propto p(C_k|\mathbf{x}_n,\Theta) \frac{\partial}{\partial \mathbf{A}_k} \log p(\mathbf{x}_n|C_k,\theta_k). \tag{21}$$

In the basis functions adaptation, the gradient of the component density with respect to the basis functions $\mathbf{A}_k$ is weighted by $p(C_k|\mathbf{x}_n,\Theta)$.

The section 2 describes the learning rules for the adaptation of $\frac{\partial}{\partial \mathbf{A}_k} \log p(\mathbf{x}_n|C_k,\theta_k)$ using the generalized Gaussian ICA (eq.11 and eq.12).

4.2. Estimating bias vectors

We can use eq.18 to adapt the bias vectors for each class $\mathbf{A}_k$.

$$\nabla_{\mathbf{b}_k} L = \sum_{n=1}^{N} p(C_k|\mathbf{x}_n,\Theta) \nabla_{\mathbf{b}_k} \log p(\mathbf{x}_n|C_k,\theta_k). \tag{22}$$

The adaptation is performed by using gradient ascent with the gradient of the component density with respect to the bias vector $\mathbf{b}_k$ giving

$$\Delta \mathbf{b}_k \propto p(C_k|\mathbf{x}_n,\Theta) \frac{\partial}{\partial \mathbf{b}_k} \log p(\mathbf{x}_n|C_k,\theta_k). \tag{23}$$

Using eq.20 in eq.23 we can adapt $\mathbf{b}_k$ as follows

$$\begin{aligned}
\Delta \mathbf{b}_k \quad \propto \quad & p(C_k|\mathbf{x}_n,\Theta) \frac{\partial}{\partial \mathbf{b}_k} \left[\log p(\mathbf{A}_k^{-1}(\mathbf{x}_n - \mathbf{b}_k))\right. \\
& \left. - \log|\det \mathbf{A}_k|\right] \qquad (24)
\end{aligned}$$

Instead of using the gradient we may also use an approximate method for the adaptation of the bias vectors. The maximum likelihood estimate $\hat{\Theta}$ must satisfy the condition

$$\sum_{n=1}^{N} p(C_k|\mathbf{x}_n,\hat{\Theta}) \nabla_{\theta_k} \log p(\mathbf{x}_n|C_k,\hat{\theta}_k) = 0, \tag{25}$$

We can use eq.25 to adapt the bias vector or mean vector $\mathbf{b}_k$.

$$\begin{aligned}
\nabla_{\mathbf{b}_k} L &= 0 \\
\sum_{n=1}^{N} p(C_k|\mathbf{x}_n,\Theta) \nabla_{\mathbf{b}_k} \log p(\mathbf{x}_n|\theta_k,C_k) &= 0. \quad (26)
\end{aligned}$$

Substituting eq.20 into eq.26 shows that the gradient of the first term in eq.20 must be zero. From this it follows that

$$\nabla_{\mathbf{b}_k} \log p(\mathbf{A}_k^{-1}(\mathbf{x}_n - \mathbf{b}_k)) = 0. \tag{27}$$

Assuming that we observe a large amount of data x_n and the probability density function (p.d.f.) of the prior $p(\mathbf{s}_t)$ is symmetric and differentiable, then $\log p(\mathbf{s}_t)$ will be symmetric as well and the bias vector can be approximated by the weighted average of the data samples

$$\mathbf{b}_k = \frac{\sum_n \mathbf{x}_n \mathbf{h}_k p(C_k|\mathbf{x}_n, \Theta)}{\sum_n \mathbf{h}_k p(C_k|\mathbf{x}_n, \Theta)}. \tag{28}$$

where

$$\mathbf{h}_k = |\mathbf{x}_n - \mathbf{b}_k|^{\frac{-2\beta_k}{(1+\beta_k)}} \tag{29}$$

is an additional term from the generalized Gaussian model consideration (Box and Tiao, 1973).

4.3. Estimating β_k

Adapt the generalized Gaussian parameters β_k for each class k with eq.7.

$$\nabla_{\beta_k} L = \sum_{n=1}^{N} p(C_k|\mathbf{x}_n, \Theta) \nabla_{\beta_k} \log p(\mathbf{x}_n|C_k, \theta_k). \tag{30}$$

The adaptation is performed by using gradient ascent with the gradient of the component density with respect to β_k. Note that β_k is a vector containing the β parameter for each source in the class.

$$\Delta\beta_k \propto p(C_k|\mathbf{x}_n, \Theta) \frac{\partial}{\partial \beta_k} \log p(\mathbf{x}_n|C_k, \theta_k). \tag{31}$$

In the beta adaptation, the gradient of the component density with respect to β_k is weighted by $p(C_k|\mathbf{x}_n, \Theta)$. Section 2.1 describes the learning rules for the adaptation of $\beta_{i,k}$ for all sources in each class using the generalized Gaussian density model.

5. UNSUPERVISED CLASSIFICATION

To demonstrate the performance of the learning algorithm, we generated random data drawn from different distributions in each class and used the proposed method to learn the parameters and to classify the data. Figure 3 shows an example of two dimensional data and four classes. The data in each class was generated by random choices for the parameters (β_k, $\mathbf{A}_k$, $\mathbf{b}_k$). The β parameters were chosen as follows in the range from -1 to +2, resulting in uniform, Gaussian and heavy Laplacian densities. The task for the algorithm was to learn the four basis vectors, bias vectors and β_k parameters given only the unlabeled two dimensional data set. The parameters were randomly initialized.

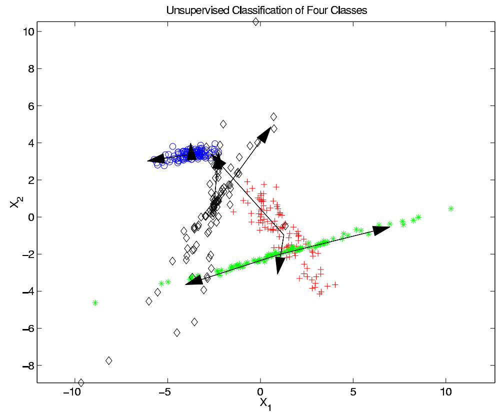

Figure 3: An example of classification of a mixture of independent components. There are 4 different classes, each generated by two independent sources and bias vectors. The algorithm is able to infer the β_k parameters, estimate the basis vectors and bias vectors for each class.

Table 1: Estimated β_k and KL-divergence

β & (KL)	C_1	C_2	C_3	C_4
$\beta(\mathbf{s}_k)$	-0.3	-0.2	1.6	2
$\mathrm{KL}(p(s_k)\|q(\beta_k))$	0.003	0.005	0.005	0.007

The algorithm always converged after 300 to 500 iterations depending on the initial conditions. During the adaptation process, the data log likelihood increased with the number of iterations. The arrows in Figure 3 indicate the basis vectors of $\mathbf{A}_k$. Table 1 shows the inferred parameters for β_k and the Kullback-Leibler divergence measure between the inferred density model and the actual source density. The classification performance was tested by processing each data instance with the learned parameters β_k, $\mathbf{A}_k$ and $\mathbf{b}_k$. The probability of the class $p(C_k|\mathbf{x}_n, \theta_k)$ was computed and the corresponding instance label was compared to the highest class probability. For this example, in which the classes had several overlapping areas, the algorithm was run 10 times with random initial conditions, in which it converged all times. The difference between the inferred β_k and the true β_k were less than 10%. The classification error on the whole data set averaged over 10 trials was 4.0% $\pm$ 0.5%. The Gaussian mixture model used in AutoClass (Stutz and Cheeseman, 1994) gave an error of 5.5% $\pm$ 0.3% and converged in all 10 trials. For k-means (Euclidean distance measure) clustering algorithm, the error was 18.3%. The classification error with the original parameters was 3.3%.

6. DISCUSSION

We proposed a new algorithm for capturing the statistical structure in multivariate data. The mixture model allows the modeling of the data in mutually exclusive classes, allowing unsupervised classification and finding several clusters or class in data. In each class the data is assumed to be generated by a linear superposition of independent sources that have non-Gaussian densities. The deviation from Gaussianity is modeled using the generalized Gaussian density in which the exponent can be inferred from the data. Since this model requires only one parameter for modeling the source density it is less complex than ICA models that require a set of parameters for describing the density of the sources (Pearlmutter and Parra, 1996; Attias, 1999). This mixture model is an extension of the Gaussian mixture model where the source components have non-Gaussian densities. It differs from the ICA mixture model in Lee et al. (1999b) where the underlying density is described by a fixed sub or super-Gaussian distribution. Here, the distribution is modeled continuously which allows a more accurate density estimation and hence more accurate characterization of the underlying data structure.

Simulations demonstrate that this method can be used in unsupervised classification and the performance will be equal or superior to the Gaussian mixture model. The advantage is more evident in real data containing outliers and more non-Gaussian distributions.

In capturing the statistics of natural images Lewicki (2000) has shown that the use of the exponential power distribution in ICA provides even sparser codes than previous methods that used a fixed density model (Bell and Sejnowski, 1997; Lewicki and Sejnowski, 2000). For modeling a wide range of images we expect that the generalized Gaussian mixture model using ICA will provide very efficient image codes for compression algorithms, better blind separation of a greater variety of sources, and better performance for de-noising algorithms.

References

Amari, S., Cichocki, A., and Yang, H. (1996). A New Learning Algorithm for Blind Signal Separation. In *Advances in Neural Information Processing Systems 8*, pages 757–763.

Attias, H. (1999). Blind separation of noisy mixtures: An EM algorithm for independent factor analysis. *Neural Computation*, 11:803–851.

Bell, A. J. and Sejnowski, T. J. (1995). An Information-Maximization Approach to Blind Separation and Blind Deconvolution. *Neural Computation*, 7:1129–1159.

Bell, A. J. and Sejnowski, T. J. (1997). The 'independent components' of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338.

Box, G. and Tiao, G. (1973). *Baysian Inference in Statistical Analysis*. John Wiley and Sons.

Cardoso, J.-F. (1997). Infomax and maximum likelihood for blind source separation. *IEEE Signal Processing Letters*, 4(4):112–114.

Comon, P. (1994). Independent component analysis – a new concept? *Signal Processing*, 36(3):287–314.

Duda, R. and Hart, P. (1973). *Pattern classification and scene analysis*. Wiley, New York.

Jutten, C. and Herault, J. (1991). Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, 24:1–10.

Lee, T.-W., Girolami, M., and Sejnowski, T. J. (1999a). Independent component analysis using an extended infomax algorithm for mixed sub-gaussian and super-gaussian sources. *Neural Computation*, 11(2):409–433.

Lee, T.-W., Lewicki, M. S., and Sejnowski, T. J. (1999b). Unsupervised classification with non-Gaussian mixture models using ICA. In *Advances in Neural Information Processing Systems 11*, pages 508–514. MIT Press.

Lewicki, M. (2000). A flexible prior for independent component analysis. *Neural Computation*, submitted.

Lewicki, M. S. and Sejnowski, T. J. (2000). Learning overcomplete representations. *Neural Computation*, 12(2):337–365.

MacKay, D. (1996). Maximum likelihood and covariant algorithms for independent component analysis. Report, University of Cambridge, Cavendish Lab.

Pearlmutter, B. and Parra, L. (1996). A context-sensitive generalization of ICA. In *International Conference on Neural Information Processing*, pages 151–157.

Stutz, J. and Cheeseman, P. (1994). Autoclass - a Bayesian approach to classification. *Maximum Entropy and Bayesian Methods*, Kluwer Academic Publishers.