Vector quantization based on $\varepsilon$-insensitive mixture models

Kazuho Watanabe

Department of Computer Science and Engineering, Toyohashi University of Technology, 1-1, Higashi, Tempaku-cho, Toyohashi, Aichi 441-8580, Japan

1. Introduction

Mixture models are widely used for clustering, quantization, and density estimation. In particular, Laplacian mixture models (LMMs) have been proposed and applied for the purposes of robust clustering and overcomplete source separation \[6,14\]. Among robust clustering methods \[11,10\], those based on LMMs provide simple learning algorithms similar to the learning of Gaussian mixture models (GMMs). However, there are two drawbacks in LMMs: (1) the degree of the robustness is uncontrollable, and (2) a cluster mean vector can inappropriately converge to a data sample, which is caused by the nature of the absolute-loss function. Mitianoudis and Stathaki \[14\] introduced a threshold to the distance to the sample points as a common solution to the second drawback.

In this paper, we consider an extension of LMMs to the mixture of $\varepsilon$-insensitive component distributions. The $\varepsilon$-insensitive distribution is defined by an $\varepsilon$-insensitive loss function which, when $\varepsilon = 0$, corresponds to the absolute loss function appearing in the Laplace distribution. The $\varepsilon$-insensitive loss function has been used in the support vector regression and other related methods to provide a sparsity inducing mechanism \[4,7,16,17,19\]. In a previous work, upper and lower bounds were obtained for the rate-distortion function associated with the $\varepsilon$-insensitive loss function \[20\]. Although the rate-distortion function shows the theoretically optimal performance of quantization schemes using the $\varepsilon$-insensitive loss function as a distortion measure, its explicit evaluation has yet to be obtained, and the optimal reconstruction distribution achieving the rate-distortion function is still unknown.

In this paper, we derive an Expectation-Maximization (EM)-type learning algorithm for the maximum likelihood estimation of mixtures of $\varepsilon$-insensitive component distributions, which provides an extension of the learning algorithm for LMMs \[14\]. The introduced $\varepsilon$, controlling the robustness of the method, partly solves the first drawback of LMMs. The maximization step (M-step) of the EM algorithm requires to minimize a function involving the $\varepsilon$-insensitive loss function. Examining the dual problem of this minimization problem, we derive a simple learning algorithm, and demonstrate that it naturally solves the second drawback of LMMs.

The $\varepsilon$-insensitive loss function has been used for a fuzzy clustering algorithm \[13\]. However, it has not been related to a probabilistic model. The proposed $\varepsilon$-insensitive mixture model is a generalization of LMMs as a probabilistic model, and is directly connected to the rate-distortion problem associated with the $\varepsilon$-insensitive loss function.

As the $k$-means algorithm is derived from the maximum likelihood estimation of GMMs in the small variance limit, we derive what we call the ei-means algorithm by taking the similar limit of the EM-type learning algorithm. We apply the EM-type algorithm to 1-dimensional problems where the rate-distortion functions associated with the $\varepsilon$-insensitive distortion measure are approximately computed. Then we apply it to a multi-dimensional synthetic data set to demonstrate the robustness-enhancing feature of the $\varepsilon$-insensitive component distribution. Finally, we compare the ei-means algorithm having different $\varepsilon$ and the k-means algorithm using a high-dimensional real data set. It is demonstrated that clustering performance is improved by adjusting $\varepsilon$ in the ei-means algorithm.
The paper is organized as follows. Section 2 defines the \( \varepsilon \)-insensitive component distribution and its mixture model. Section 3 describes the framework of the EM-type learning algorithm, introduces the dual problem of the M-step required for it, and derives update rules of parameters. As a limit of the EM-type algorithm the \( \varepsilon \)-means algorithm is also derived. Section 4 applies the EM-type algorithm to the approximate computation of the rate-distortion function under the \( \varepsilon \)-insensitive loss function. Section 5 examines the robustness property of the mixture of \( \varepsilon \)-insensitive component distributions and the clustering performance of the \( \varepsilon \)-means algorithm by numerical experiments using synthetic and real data. Section 6 concludes the paper and discusses future research directions.

2. Mixture of \( \varepsilon \)-insensitive component distributions

In this section, we define the mixture model consisting of the noise model corresponding to the \( \varepsilon \)-insensitive loss function. The \( K \)-component mixture model of the distribution \( c_i(x; \theta_i) \) for \( x = (x^1, \ldots, x^d) \in \mathbb{R}^d \) is defined by

\[
p(x|w) = \sum_{k=1}^{K} a_k c_i(x; \theta_i),
\]

where \( w = ([a_1], [\theta_1]) \) denotes the parameter vector consisting of the parameter \( \theta_k \in \mathbb{R}^d \) for each component and the mixing proportions \( [a_k] \) satisfying \( a_k \geq 0 \) for \( k = 1, 2, \ldots, K \) and \( \sum_{k=1}^{K} a_k = 1 \).

In this paper, we focus on the following component distribution:

\[
c_i(x; \theta) = \frac{1}{c_i} \exp\{-sp_i(||x - \theta||)\},
\]

where \( P_i(z) = \max\{||z|-\varepsilon, 0\} \) is the \( \varepsilon \)-insensitive loss function, and \( ||x - \theta|| = \sqrt{\sum_{d=1}^{d} (x^d - \theta^d)^2} \) denotes the Euclidean distance between \( x \) and \( \theta \). The constant \( \varepsilon > 0 \) in Eq. (2) is called the slope parameter, which shows the (negated) slope of the tangent of the rate-distortion function in Section 4. The slope parameter can also be included in the model parameter and estimated from data. A straightforward extension of the model (1) is obtained by separately introducing the parameter \( s_k > 0 \) in the \( k \)-th component.

The normalization constant \( C_i \) in Eq. (2) is explicitly obtained as

\[
C_i = \int_{|x|} \exp\{-sp_i(||x||)\} \, dx = I(d) \int_0^{\infty} e^{-sp_i(r^2)} \, dr,
\]

where \( I(d) = \sqrt{\pi^d} / \Gamma(d/2 + 1) \) is the area of the \( d \)-dimensional unit hypersphere. Here, \( \Gamma(u) = \int_0^{\infty} t^{u-1}e^{-t} \, dt \) and \( \Gamma(u, \alpha) = \int_0^{\infty} t^{u-1} e^{-t} \, dt \) denote the gamma and the upper incomplete gamma functions respectively.

The component (2) for \( \varepsilon = 0 \) is the (isotropic) Laplace distribution, \( c_i(x; \theta) \propto \exp\{-||x - \theta||\} \), and the mixture (1) reduces to the LMM [6,14]. Although we restrict ourselves to the isotropic (spherical) component distribution in Eq. (2), it may be generalized to a distribution with arbitrary covariance structure in the same manner as for the Laplace distribution (\( \varepsilon = 0 \)) given in [9]. Hereafter, the mixture model in Eq. (1) is referred to as the \( \varepsilon \)-insensitive mixture model (EIMM).

3. EM algorithm for EIMMs

We derive a learning algorithm for the maximum likelihood estimation of the EIMM based on the EM algorithm [8,2]. The overall framework and the E-step are formulated as in usual mixture models (Section 3.1) while the M-step is formulated through the dual optimization problem (Section 3.2).

3.1. \( \varepsilon \) and M steps

The log-likelihood of the EIMM for the training samples \( x^0 = \{x_1, \ldots, x_n\} \) is lower bounded as follows:

\[
\sum_{i=1}^{n} \log p(x_i|w) = \sum_{i=1}^{n} \sum_{k=1}^{K} a_k c_i(x_i; \theta_k) \geq \sum_{i=1}^{n} \sum_{k=1}^{K} \eta_{ik} \left[ \log a_k - \log C_i - sp_i(||x_i - \theta_k||) - \log \eta_{ik} \right] = Q(w|\tilde{w}).
\]

Here,

\[
\eta_{ik} = \frac{\tilde{a}_k c_i(x_i; \tilde{\theta}_k)}{\sum_{j=1}^{K} \tilde{a}_j c_j(x_i; \tilde{\theta}_j)}
\]

satisfying \( \sum_{k=1}^{K} \eta_{ik} = 1 \) for \( i = 1, \ldots, n \) is the posterior probability that \( x_i \) is assigned to the \( k \)-th component under the current estimate of the model parameter \( \tilde{w} = ([\tilde{a}_1], [\tilde{\theta}_1]) \).

The EM algorithm maximizes \( Q(w|\tilde{w}) \) with respect to \( \tilde{w} \) at each iteration, which is guaranteed to increase the log-likelihood. More specifically, setting an initial value for \( \tilde{w} \), we iterate the following E- and M-steps until convergence:

**E-step:** Compute \( \eta_{ik} \) for \( i = 1, \ldots, n \) and \( k = 1, \ldots, K \) by Eq. (3).

**M-step:** For \( k = 1, \ldots, K \),

\[
\tilde{a}_k \leftarrow \frac{1}{n} \sum_{n} \eta_{ik}, \quad \tilde{\theta}_k \leftarrow \operatorname{argmin}_{\theta} \sum_{i=1}^{n} \eta_{ik} p_i(||x_i - \theta||) + \frac{\varepsilon}{d}.
\]

The updating rule of \( \theta_k \) in the M-step is not explicitly solved unlike for usual GMMs. We focus on the minimization problem in Eq. (4) in the next subsection.

In order to estimate the parameter \( s \), we can use the first order approximation, \( \log C_i \approx \log I(d) \Gamma(d)/s^d + s \varepsilon \), and include the update rule,

\[
\frac{1}{s} = \frac{1}{d} \sum_{d=1}^{d} \sum_{k=1}^{K} \eta_{ik} p_i(||x_i - \theta_k||) + \frac{\varepsilon}{d}.
\]

If we further introduce the parameter \( s_k \) to each component, its update rule is given by

\[
\frac{1}{s_k} = \frac{1}{d} \sum_{n} \eta_{ik} p_i(||x_i - \theta_k||) + \frac{\varepsilon}{d},
\]

where \( n_k = \sum_{i=1}^{n} \eta_{ik} \).

3.2. Dual problem for M-step and partial M-step

In the M-step of the EM algorithm, it is required to minimize a convex function of the form,

\[
L(\theta) = \sum_{i=1}^{n} \nu_i p_i(||x_i - \theta||),
\]

where \( 0 \leq \nu_i \leq 1 \) for \( i = 1, \ldots, n \). By introducing slack variables \( \xi_i \), we reformulate the minimization of the function (5) as the following minimization problem with inequality constraints:

\[
\min \sum_{i=1}^{n} \nu_i \xi_i \quad \text{subject to} \quad ||x_i - \theta|| - \varepsilon \leq \xi_i \quad \text{and} \quad \xi_i \geq 0 \quad (i = 1, \ldots, n).
\]

Let \( \alpha = (\alpha_1, \ldots, \alpha_n) \), \( L(\alpha, \theta) = \sum_{i=1}^{n} \alpha_i (||x_i - \theta|| - \varepsilon) \), and \( B = (\alpha : 0 \leq \alpha_i \leq \alpha_i, i = 1, \ldots, n) \). Examining the Lagrange dual problem of the above minimization problem, we have

\[
L(\alpha) = \max_{\theta} L(\alpha, \theta).
\]
In fact, the maximum with respect to \( \alpha_i \) is achieved when

\[
\alpha_i = \begin{cases} 1 & (||x_i - \theta|| > \epsilon) \\ 0 & (||x_i - \theta|| \leq \epsilon) \end{cases}
\]  

(7)

for \( i = 1, \ldots, n \). Putting this back into \( \hat{L}(\alpha, \theta) \) in Eq. (6) yields the original form of \( L(\theta) \) in Eq. (5).

To derive a simple update rule, instead of maximizing with respect to \( \alpha \) in Eq. (6), we first minimize \( \hat{L}(\alpha, \theta) \) with respect to \( \theta \) for fixed \( \alpha \). We set the derivative of \( \hat{L}(\alpha, \theta) \) to zero,

\[
\frac{\partial L}{\partial \theta} = \sum_{i=1}^{n} \alpha_i \frac{\theta - x_i}{||\theta - x_i||} = 0 \quad \Rightarrow \quad \theta = \frac{\sum_{i=1}^{n} \alpha_i x_i}{\sum_{i=1}^{n} \alpha_i},
\]

which implies the update rule of \( \theta \),

\[
\theta \leftarrow \frac{\sum_{i=1}^{n} \alpha_i \frac{x_i}{||x_i - \theta||}}{\sum_{i=1}^{n} \alpha_i / ||x_i - \theta||},
\]

(8)

Hence, we can think of the fixed-point optimization approach that iterates the updating rules (7) and (8) to minimize \( L(\theta) \). However, this approach does not always minimize \( L(\theta) \) although it does fully minimize \( \hat{L}(\alpha, \theta) \) in some cases [21]. Instead, we propose to iterate the updating rules (7) and (8) once at each M-step, which is an example of the so-called “partial M-step” [2], if the updating rule (8) decreases \( L(\theta) \) even a little.

When \( \epsilon = 0 \), the overall learning procedure reduces to the learning algorithm for LMMs proposed in [14].

For \( \epsilon > 0 \), we can prove the monotonic decrease of \( L(\theta) \) by the above update rule. The proof is given in the Appendix. Note, however, for \( \epsilon = 0 \), once the updated \( \theta \) comes very close to a data point \( x_i \), \( \theta \) converges to \( x_i \) because the weight for \( x_i \) approaches 1 in Eq. (8). Mitianoudis and Stathaki [14] solved this problem by introducing a threshold \( \delta \). That is, \( x_i \) is ignored when \( ||x_i - \theta|| < \delta \).

Eq. (7) shows that this solution is naturally implemented by the \( \epsilon \)-insensitive loss function with \( \epsilon > 0 \). We can optionally switch to the (sub)gradient method with a learning rate \( \mu \),

\[
\theta \leftarrow \theta - \mu \left( \theta - \frac{\sum_{i=1}^{n} \alpha_i \frac{x_i}{||x_i - \theta||}}{\sum_{i=1}^{n} \alpha_i / ||x_i - \theta||} \right),
\]

to guarantee the convergence. Effective optimization methods of \( L(\theta) \) for \( \epsilon > 0 \) with convergence guarantee are to be explored further.

3.3. \( \epsilon \)-Insensitive-means algorithm

The famous k-means algorithm is derived from the small variance limit of the EM algorithm for GMMs. The similar limit, \( \epsilon \rightarrow \infty \), of the derived EM-type algorithm leads to a clustering algorithm (Algorithm 1), which we name as the \( \epsilon \)-insensitive-means (ei-means) algorithm. Note that the ei-means algorithm with \( \epsilon > 0 \) is not guaranteed to converge as discussed in Section 3.2. The ei-means algorithm has the objective function,

\[
\min_{\theta} \sum_{i=1}^{n} \rho_{\epsilon}(||x_i - \theta||),
\]

where \( c(i) \) denotes the cluster label of the \( i \)th data point \( x_i \), although we cannot compare the objective values to choose \( c \).

**Algorithm 1.** ei-means.

**Input:** Data points \( \{x_i\}_{i=1}^{n} \) on \( \mathbb{R}^{d} \)

**Output:** Cluster labels \( \{c(i)\}_{i=1}^{n} = \{1, 2, \ldots, K\} \)

Initialize \( \theta_k \) for \( k = 1, 2, \ldots, K \)

repeat

\[
c(i) \leftarrow \arg\min_{k} ||x_i - \theta_k|| \quad \text{for} \quad i = 1, \ldots, n
\]

\[
\theta_k \leftarrow \frac{\sum_{j=1}^{n} \alpha_i \frac{x_j}{||x_j - \theta_k||}}{\sum_{j=1}^{n} \alpha_i / ||x_j - \theta_k||}
\]

for \( k = 1, \ldots, K \)

until convergence or maximum number of iterations reached.

4. Application to rate-distortion computation

In this section, applying the learning algorithm developed in the previous sections, we approximately compute the rate-distortion function for the \( \epsilon \)-insensitive loss function [20]. The rate-distortion function, \( R(D) \), shows the minimum possible rate (logarithm of the codebook size) required for reconstructing the original information with average distortion not exceeding \( D \) [3]. It also shows how close the optimal mixture model is to the distribution of an information source in the sense of the Kullback–Leibler (KL) divergence as described below.

The rate-distortion function is defined by the mutual information minimized under the constraint that the average distortion is at most \( D \). This problem is equivalent to minimizing the following functional over the reconstruction density \( q(\theta) \) [3,20]:

\[
F(q) = - \int p(x) \log \left( \int e^{-d(x, \theta)q(\theta)} d\theta \right) dx,
\]

(9)

where \( p(x) \) is the source density, and \( d(x, \theta) \) is the distortion measure between \( x \) and \( \theta \). If we find the optimal reconstruction density \( q(\theta) \), we have the optimal conditional density of reconstruction, which is given by \( q_i(x|\theta) \propto q(\theta) \exp(-sd(x, \theta)) \). Then, the parametric form of the rate-distortion function is obtained as follows:

\[
R(D_s) = \int p(x)q_i(x|\theta) \log \frac{q_i(x|\theta)}{p(x)q(\theta)} d\theta dx,
\]

(10)

\[
D_s = \int p(x)q_i(x|\theta)d(x, \theta) dx d\theta,
\]

(11)

where the negated slope parameter \( s \) is the slope of the tangent of the rate-distortion function \( R(D) \) at \( (D_s, R(D_s)) \). The rate \( R \) is measured by the unit “nat” instead of “bit” since we use the natural logarithms in this paper.

The above problem is also equivalent to minimizing the KL-divergence from \( p(x) \) to the mixture of \( \epsilon \)-insensitive distributions (2) mixed by \( q(\theta) \) if we take \( d(x, \theta) = \rho_{\epsilon}(||x - \theta||) \). Hence, the maximum likelihood estimation of the model \( q(\theta)q_i(x|\theta) \) \( d\theta \) approximately solves the rate-distortion problem if we approximate the source \( p(x) \) by the empirical distribution, \( p(x) = \frac{1}{n} \delta(x - x_i) \) of the samples \( \{x_1, \ldots, x_n\} \) drawn i.i.d. from \( p(x) \), where \( \delta \) is Dirac’s delta function. Then, the rate-distortion function is approximately computed by obtaining the maximum likelihood estimate \( \hat{w} \) for the parameter of the mixture of \( \epsilon \)-insensitive distributions (1) for each slope parameter \( s \) if the reconstruction distribution is restricted to be a \( K \)-component discrete distribution, \( q(\theta) = \sum_{k=1}^{K} q_i(x|\theta) \).

To examine the accuracy of the bounds for the rate-distortion function obtained in the 1-dimensional case [20], we focused on the case of \( d = 1 \). We fixed \( \epsilon = 0.1 \) throughout the experiment since the relative behavior of the bounds does not change with the value of \( \epsilon \), and the bounds simply get looser as \( \epsilon \) grows. We generated two data sets of size \( n = 10^{5} \) according to the standard normal
distribution and the Laplace distribution with the density $l_p(x) = (β/2)e^{-β|x|} (β = 1/\sqrt{2}$) respectively. The golden section search [15, Section 10.2] was applied for solving the minimization of $l(θ)$ in Eq. (5) exactly. By using the approximation of the source by the empirical distribution and the discrete approximation to the reconstruction distribution, we approximately calculated the 6 points on the rate-distortion curve corresponding to $s = 1.25, 2.5, 5, 10, 20, 40$. For each $s$, we applied the EIMMs with $K = 2, 4, \ldots, 48, 50$ and adopted the number of components $K$ when the increase in the likelihood was saturated. The resulting rate (10)

$$R(θ) = \sum_{i=1}^{N} \log p(x_i | θ)$$

discrete distribution. We find that the rate-distortion pairs of $s = 40$ for the Laplacian data set and of $s = 20, 40$ for the Gaussian data set are above the upper bounds. This may be due to the limited number of mixture components (up to 50) and the limited number of EM iterations (up to 500 iterations).

5. Application to multi-dimensional problems

In this section, we apply EIMMs to multi-dimensional problems. In the first experiment, we examine the robustness property of EIMMs using synthetic data containing outliers. In the second experiment, we apply the ei-means algorithm (Section 3.3) to real e-mail filtering data set to compare it with the k-means algorithm.

5.1. Synthetic data

It was demonstrated for the support vector regression that the $ε$-insensitive loss function promotes robustness to outlying observations [4,7,13,16,17,19]. We investigate the robustness property of EIMMs by using 10-dimensional synthetic data set contaminated with outliers.

As a true data-generating distribution, we fixed a 5-component isotropic LMM with equal weights in 10-dimensional space and generated 500 samples $X_i \sim \mathcal{N}(0, \Sigma)$. The mean parameters of the true LMM were randomly generated from the uniform distribution on $[-5, 5]^5$ and we set $s = 5$. As a contamination, we replaced $C = 0, 2.5$, and 5% of data by random points uniformly distributed on $[-5, 5]^5$ to make 3 data sets with different contamination levels.

We applied the EM algorithm using the partial M-step for the EIMMs with $ε = 0$ (LMM), 0.25, 0.5, 2.75, and 3 and obtained the estimate $\hat{θ} = \hat{a}_k, \hat{θ}_k$ for each EIMM. We generated the test data $\{\bar{x}_i\}_{i=1}^{T} (T = 25000)$ from the true LMM (without contamination) and calculated the test negative log-likelihood, also known as the log-perplexity:

$$E(x^ε) = -\frac{1}{T} \sum_{i=1}^{T} \log \sum_{k=1}^{K} \hat{a}_k \log p(x_i | \hat{θ}_k),$$

where we set $ε = 0$ to ignore the influence of model mismatch and compare the accuracy of estimates for different $ε$. We repeated the experiment 1000 times using different training data sets obtained from the same generation process and calculated the average of the log-perplexities (Fig. 2). Note here that although the value of

![Fig. 1](image1.png)

**Fig. 1.** Rate-distortion bounds (curves) and approximated values of rate-distortion pairs (crosses) for (a) the Laplacian data set and (b) the Gaussian data set. Only the lowest curve in each panel is a lower bound, while the remaining curves (or straight line) are upper bounds.

![Fig. 2](image2.png)

**Fig. 2.** Average test log-perplexities for different $ε$. The minimum for each contamination level is marked by a circle. The minimums for the contamination levels, 2.5% and 5% are significantly smaller than those of $ε = 0$ (paired t-test, $p < 0.05$).
the log-perplexity in Eq. (12) itself is meaningless, the smaller value of the log-perplexity means that the test data are better predicted by the learned model. It can be seen that introducing a positive \( \varepsilon \) reduces the average log-perplexity when there is a contamination if the value of \( \varepsilon \) is appropriately chosen. In fact, the minimum average log-perplexities for \( C=2.5 \) (\( \varepsilon=0.5 \)) and \( C=5 \) (\( \varepsilon=0.75 \)) are significantly smaller than the average log-perplexities for \( \varepsilon=0 \) according to the paired \( t \)-test (\( p<0.05 \)). For a given percentage of noisy data, selection of the \( \varepsilon \) value can be accomplished by applying the EM algorithm repeatedly with different \( \varepsilon \) values and comparing the likelihoods.

5.2. Spam data

To compare the clustering performance of the ei-means algorithm with that of the k-means algorithm, we applied them to Spambase data set [1], where each datum consists of 57 attributes representing an e-mail and a binary class label representing whether the e-mail was considered as spam or not. This data set was originally used for the classification problem, the details of which can be found in [12, Chapter 1]. Assuming that there are two clusters in the data set corresponding to the two classes (spam or non-spam), we modify the problem as a clustering task by removing class labels of training data.

We used 3600 data with equal numbers of spam or non-spam e-mail data (1800 data for each class). Dividing the data set into 5 blocks, we conducted 5-fold stratified cross validation. That is, in each fold, 2880 training data (1440 for each class) were clustered without their labels by clustering algorithms having the number of each fold, 2880 training data (1440 for each class) were clustered without their labels by clustering algorithms having the number of classes, 2880 training data (1440 for each class) were clustered without their labels by clustering algorithms having the number of clusters, \( K=2 \), to simplify the clustering task, selecting \( K \) is also an important problem in a practical scenario. Possible approaches to selection of \( K \) and \( \varepsilon \) include cross validation and Bayesian methods, which are to be explored in the future.

Fig. 3 shows the classification accuracy for each of the 5-fold cross validation. The ei-means algorithm with appropriate \( \varepsilon \) outperformed the k-means algorithm, which failed to obtain a clustering result predictive of the class label for 3 data sets. The k-means algorithm with \( \varepsilon=0.1 \) performed best on average. The accuracy of the ei-means with \( \varepsilon=0.05 \) is higher than that with \( \varepsilon=0 \) for all data sets. Hence, the nonparametric sign test (\( p=0.031 \)) implies that the clustering method based on the LMM (\( \varepsilon=0 \)) is improved by the ei-means with \( \varepsilon=0.05 \).

Fig. 3 also implies that we need to select a suitable value of \( \varepsilon \) to use the ei-means algorithm effectively. Although we fixed the number of clusters, \( K=2 \), to simplify the clustering task, selecting \( K \) is also an important problem in a practical scenario. Possible approaches to selection of \( K \) and \( \varepsilon \) include cross validation and Bayesian methods, which are to be explored in the future.

6. Conclusion

In this study, we derived an EM-type algorithm for EIMMs. As a limit of the EM-type algorithm, we also derived the ei-means algorithm. We applied these algorithms to approximate rate-distortion computation, density estimation, and clustering problems. It has been demonstrated that the EIMM with appropriate \( \varepsilon \) is robust against noisy data.

It is an important undertaking to investigate the convergence property of the EM-type algorithm and the ei-means algorithm in high-dimensional problems. High-dimensional extensions of the rate-distortion analysis are also to be addressed.

Acknowledgments

The author is grateful to the anonymous referees for their helpful comments and suggestions. This work was supported by JSPS KAKENHI Grant numbers 23700175 and 25120014.

Appendix A. Convergence property of the updating rule (8) for \( \varepsilon=0 \)

We prove that the updating rule in Eq. (8) monotonically decreases \( L(\hat{\Theta}) \) in Eq. (5) for \( \varepsilon=0 \). This is proved based on the fact that Eq. (8) for \( \varepsilon=0 \),

\[
\hat{\Theta} = \frac{\sum_{i=1}^{n} k_i \frac{x_i}{\|x_i - \Theta\|}}{\sum_{i=1}^{n} k_i \frac{1}{\|x_i - \Theta\|}}
\]

(A.1)

can be considered as the mean shift algorithm having the kernel function, \( k_i |x - \Theta| = C - (\sqrt{|x - \Theta|^2} - \frac{a}{b}) \), where \( C \) is a large constant [5]. While the convergence proof of the mean shift algorithm is given in [5] for general kernel functions, we present the proof for this particular case for the sake of completeness.

From the concavity of the square root, we have

\[
\sqrt{b} - \sqrt{a} \leq \frac{1}{2\sqrt{a}}(b - a),
\]

for \( a > 0 \) and \( b > 0 \). It follows that

\[
L(\Theta) - L(\hat{\Theta}) = \sum_{i=1}^{n} u_i \left( \sqrt{\|x_i - \Theta\|^2} - \sqrt{\|x_i - \hat{\Theta}\|^2} \right)
\]

\[1\] Although we also tried the multi-start method with 10 different initializations (common to all the clustering methods), the result changed little.
\[ L(\theta) \leq \sum_{i=1}^{n} \nu_i \frac{1}{2 \sqrt{\|X_i - \theta\|^2}} \left( \|X_i - \theta\|^2 - \|X_i - \hat{\theta}\|^2 \right) \]

\[ = \sum_{i=1}^{n} \nu_i \frac{1}{2 \sqrt{\|X_i - \theta\|^2}} \left( \|\theta\|^2 - 2\|X_i\| \cdot (\theta - \hat{\theta}) - \|\hat{\theta}\|^2 \right) \]

\[ = \left( \|\theta\|^2 - \|\hat{\theta}\|^2 \right) \sum_{i=1}^{n} \nu_i \frac{1}{2 \sqrt{\|X_i - \theta\|^2}} + 2 \theta \cdot \hat{\theta} \sum_{i=1}^{n} \nu_i \frac{1}{2 \sqrt{\|X_i - \theta\|^2}} \]

\[ = \sum_{i=1}^{n} \nu_i \frac{1}{2 \sqrt{\|X_i - \theta\|^2}} \|\theta - \hat{\theta}\|^2 \geq 0, \]

where the second to last equality follows from the updating rule (A.1). This means that \( L(\theta) \) is monotonically decreased by this updating rule.

References


Kazuo Watanabe was born in 1980. He received B.E., M.E. and Ph.D. degrees from the Tokyo Institute of Technology, Japan, in 2002, 2004 and 2006, respectively. From 2007 to 2008, he worked as a postdoctoral fellow and a research associate at the Department of Complexity Science and Engineering, the University of Tokyo. From 2009 to 2013, he worked as an assistant professor at the Graduate School of Information Science, Nara Institute of Science and Technology. He is currently a lecturer at the Department of Computer Science and Engineering, Toyohashi University of Technology. His research interests include statistical learning theory and its applications.