AISM (2007) 59:3–25
DOI 10.1007/s10463-006-0097-x

A modified EM algorithm for mixture models based on Bregman divergence

Yu Fujimoto · Noboru Murata

Received: 7 April 2006 / Revised: 4 August 2006 /
Published online: 16 December 2006

Abstract The EM algorithm is a sophisticated method for estimating statistical models with hidden variables based on the Kullback–Leibler divergence. A natural extension of the Kullback–Leibler divergence is given by a class of Bregman divergences, which in general enjoy robustness to contamination data in statistical inference. In this paper, a modification of the EM algorithm based on the Bregman divergence is proposed for estimating finite mixture models. The proposed algorithm is geometrically interpreted as a sequence of projections induced from the Bregman divergence. Since a rigorous algorithm includes a nonlinear optimization procedure, two simplification methods for reducing computational difficulty are also discussed from a geometrical viewpoint. Numerical experiments on a toy problem are carried out to confirm appropriateness of the simplifications.

Keywords Bregman divergence · EM algorithm · Finite mixture models

1 Introduction

In information geometry, discrepancy measures between distributions play a key role. There are many candidates for such measures, and the Bregman divergence is one of them. The Bregman divergence is a class of divergences derived from an arbitrary convex functions and includes the Kullback–Leibler (KL) divergence as a special case.

Y. Fujimoto (✉) · N. Murata
School of Science and Engineering, Waseda University,
3-4-1 Ohkubo, Shinjuku, Tokyo 169-8555, Japan
e-mail: fyu@toki.waseda.jp

The Bregman divergence has some interesting properties when it is applied to statistical inferences. One of those properties is its duality. The Bregman divergence is associated with two representations of distributions: the mixture representation, and a specific representation derived from the convex function. Like the α-divergences (Amari and Nagaoka, 2000), these two representations are related to geometrical duality, and one of its interesting and important characteristics is simply expressed by Pythagorean relation (Murata et al. 2004).

Another interesting property is its robustness. Estimators based on the Bregman divergence is not Fisher efficient in general, but it shows good robust properties to outliers and noises (see for example, Minami and Eguchi, 2002; Takenouchi and Eguchi, 2004; Fujisawa and Eguchi, 2005; Takenouchi, 2005). Robustness of estimators can be discussed by using influence functions (Hampel et al., 1986), and certain estimators derived from some Bregman divergences are shown to have bounded influence functions, and therefore they are tolerant of outliers and noises.

In this paper, we propose a modification of the EM algorithm as an application of the Bregman divergence, which can be used for robust estimation of mixture models. For example, random variables with many categories are dealt with in Bayesian networks or graphical models, and for describing such random variables with many categories, statistical models with numerous parameters are required. Also, the number of collectable samples is sometimes insufficient compared with the number of parameters, and some of cells in contingency tables lack data. As a consequence, estimates of the parameters become less reliable. To overcome these problems, restricted mixture models such as aspect models (Hoffmann, 1999) or latent class models (Agresti, 2002, Chap. 13) are used, in which certain constraints among cells are imposed and the number of modifiable parameters are reduced. The EM algorithm is utilized to estimate the parameters of such mixture models, however, it sometimes fails to give a good estimate due to lacks of data in contingency tables. To avoid such an instability, we focus on the robust property of the Bregman divergence and geometrical flat structure of mixture models, and propose a variation of the EM algorithm based on the Bregman divergence.

This manuscript is organized as follows. In Sect. 2, the definition of the Bregman divergence is stated as a preliminary, and in Sect. 3, some geometrical properties of the Bregman divergence are discussed, then two projections and mixture models associated with the Bregman divergence are defined. The definition of the UM algorithm, which is a modified EM algorithm proposed in this paper, is given in Sect. 4, and two approximations of the optimizing procedure for practical implementations are discussed. Some illustrative examples with numerical experiments are shown in Sect. 5, and Sect. 6 is devoted to concluding remarks.

2 Bregman divergence

The Bregman divergence is a pseudo-distance for measuring discrepancy between two functions. Let U be a strictly convex function on R, then

discrepancy between two values f and g is defined by

$$\mathrm{d}(f,g) = U(g) - U(f) - U'(f)(g-f),$$

where U' is the derivative of U. Note that d is non-negative due to the convexity of U and not symmetric with respect to f and g (see Fig. 1a). The integral of d under a certain measure μ over the domain of x gives total discrepancy between two functions $f(x)$ and $g(x)$,

$$D(f,g) = \int d(f(x), g(x))\mathrm{d}\mu(x).$$

If x takes a discrete value on a certain space $\mathcal{X}$, total discrepancy is given by the weighted sum over $\mathcal{X}$ as

$$D(f,g) = \sum_{x\in\mathcal{X}} d(f(x), g(x))\mu(x).$$

Note that D is non-negative, asymmetric, and equal to zero if and only if $f = g$ (a.e.). In this paper, we deal with the case that x is continuous, but it can be extended to the discrete case in a straightforward way.

To use the empirical distribution, we consider a slightly modified version of the Bregman divergence as follows. Let us consider the space of positive finite measures over $x \in \mathcal{X}$ under a carrier measure $\mu(x)$

$$\mathcal{F} = \left\{ m(x) \;\middle|\; m : \mathcal{X} \to R_+, \int_{x\in\mathcal{X}} m(x)\mathrm{d}\mu(x) < \infty \right\}. \tag{1}$$

Note that probability densities belong to a subspace of $\mathcal{F}$,

$$\mathcal{P} = \left\{ m(x) \;\middle|\; m : \mathcal{X} \to R_+, \int_{x\in\mathcal{X}} m(x)\mathrm{d}\mu(x) = 1 \right\} \subset \mathcal{F}. \tag{2}$$

We define the Bregman divergence as follows (see Fig. 1b).

Definition 1 (Bregman divergence) *Let U be a strictly convex function on R, and $u = U'$ be the derivative of U, which has the inverse function $\xi = u^{-1}$. For $p(x)$ and $q(x)$ in $\mathcal{F}$, the Bregman divergence is defined as*

$$\begin{aligned} D_U(p,q) &= \int \mathrm{d}_U(p(x), q(x))\mathrm{d}\mu(x), \\ \text{where } \mathrm{d}_U(p,q) &= U(\xi(q)) - U(\xi(p)) - p\left[\xi(q) - \xi(p)\right]. \end{aligned} \tag{3}$$

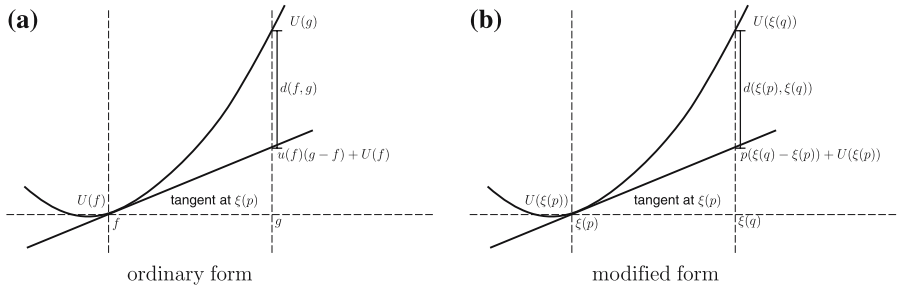


Fig. 1 An intuitive interpretation of the Bregman divergence. **a** Ordinary form. **b** Modified form

The above form is convenient when the empirical distribution is plugged into the divergence directly. To see this, let us define the cross entropy associated with U as

$$H_U(p,q) = \int \left[U(\xi(q(x))) - p(x)\xi(q(x))\right] \mathrm{d}\mu(x), \tag{4}$$

then the Bregman divergence is written with two cross entropies as

$$D_U(p,q) = H_U(p,q) - H_U(p,p). \tag{5}$$

This relation leads the equivalence of minimizing the Bregman divergence and minimizing the cross entropy for fixed p:

$$\operatorname*{argmin}_q D_U(p,q) = \operatorname*{argmin}_q H_U(p,q).$$

For given samples $\{x_i; i = 1, \ldots, N\}$, let

$$\tilde{p}(x) = \frac{1}{N}\sum_{i=1}^{N} \delta(x - x_i) \tag{6}$$

be the empirical probability density of x, where δ is Dirac's delta function. The minimizer of the Bregman divergence for given samples is obtained by plugging the empirical distribution $\tilde{p}$ directly into the cross entropy as

$$\operatorname*{argmin}_q H_U(\tilde{p},q) = \operatorname*{argmin}_q \left[\int U(\xi(q(x)))\mathrm{d}\mu(x) - \frac{1}{N}\sum_{i=1}^{N} \xi(q(x_i))\right].$$

Some typical examples of the convex function U are listed in Table 1 and their shapes are shown in Fig. 2.

Table 1 Examples of U functions

	$U(z)$	$u(z)$	$\xi(z) = u^{-1}(z)$
KL	$\exp(z)$	$\exp(z)$	$\log(z)$
β-type	$\frac{(\beta z+1)^{(\beta+1)/\beta}}{\beta+1}$	$(\beta z+1)^{1/\beta}$	$\frac{z^{\beta}-1}{\beta}$
η-type	$\exp(z)+\eta z$	$\exp(z)+\eta$	$\log(z-\eta)$

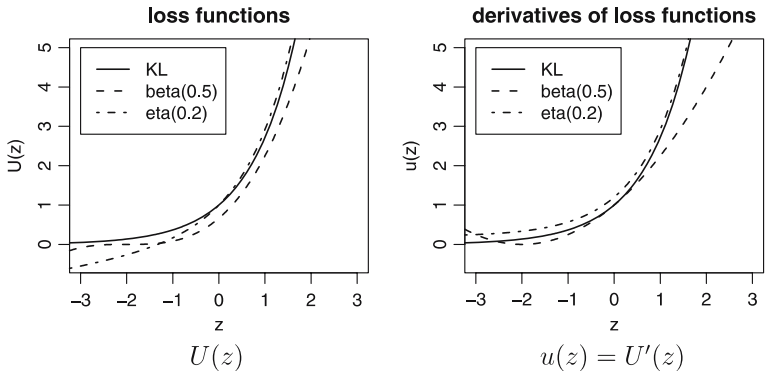


Fig. 2 Examples of U functions

3 Geometrical properties and mixture models

The Bregman divergence is closely related with the potential duality. Let us consider the Legendre transformation of U

$$U^*(\zeta) = \sup_z \{\zeta z - U(z)\}, \tag{7}$$

and define two representations as

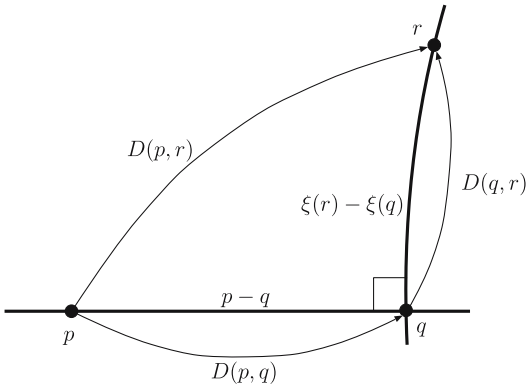
$$\begin{aligned} m\text{-representation:} &\quad p, \\ u\text{-representation:} &\quad p^* = \xi(p)\ (\xi = u^{-1} = U'^{-1}), \end{aligned}$$

then the Bregman divergence is written with a potential form as

$$\begin{aligned} \mathrm{d}_U(p,q) &= U^*(p) + U(q^*) - pq^*, \\ D_U(p,q) &= \int \mathrm{d}_U(p(x), q(x))\mathrm{d}\mu(x). \end{aligned} \tag{8}$$

It is possible to discuss detailed differential geometrical aspects of parametric models, however, here we focus on the following simple property of so-called Pythagorean relation. For any three points $p, q, r \in \mathcal{F}$, and for any function U,

Fig. 3 Pythagorean relation



it is easy to check the following equation holds:

$$D_U(p,r) - D_U(p,q) - D_U(q,r) = \int \{p(x) - q(x)\}\{\xi(r(x)) - \xi(q(x))\}\,\mathrm{d}\mu(x).$$

The right-hand side is regarded as an inner product of $p - q$ and $\xi(r) - \xi(q)$, therefore, Pythagorean relation for the Bregman divergence is stated as follows (see Fig. 3).

Theorem 1 (Pythagorean relation) (Murata et al., 2004) *Let p, q and r be in $\mathcal{F}$. If $p - q$ and $\xi(r) - \xi(q)$ are orthogonal at q, the relation*

$$D_U(p,r) = D_U(p,q) + D_U(q,r) \tag{9}$$

holds.

From above-mentioned Pythagorean relation, we can derive the dualistic structure of two different optimization problems. First we define two flat subspaces associated with *m*- and *u*-representations.

Definition 2 (flatness) *Let p and q be in $\mathcal{F}$, then the m-geodesic between p and q is defined as a set of interior divisions of p and q with m-representation*

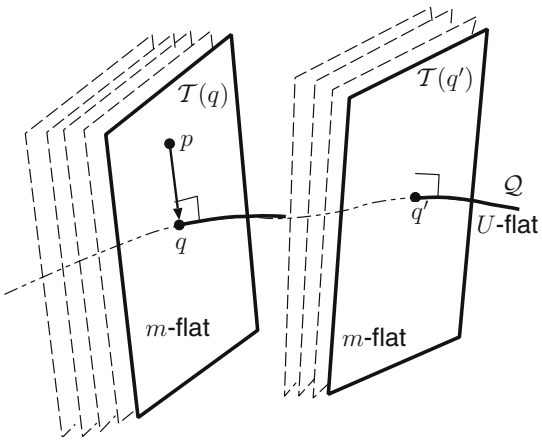
$$r(x;t) = (1-t)\cdot p(x) + t\cdot q(x), \quad 0 \le t \le 1,$$

and the u-geodesic is a set of interior divisions of p and q with u-representation

$$\xi(r(x;t)) = (1-t)\cdot \xi(p(x)) + t\cdot \xi(q(x)), \quad 0 \le t \le 1.$$

Subspaces are called m-flat or u-flat if m-geodesics or u-geodesics of any two points in the subspaces are included in the subspaces themselves.

Fig. 4 Orthogonal foliation



As shown in Fig. 4, with the notion of m- and u-flat subspaces, $\mathcal{F}$ is sliced into a set of disjoint m-flat subspaces $\mathcal{T}$'s which are orthogonal to a certain u-flat subspace $\mathcal{Q}$,

$$\bigcup_{q \in \mathcal{Q}} \mathcal{T}(q) = \mathcal{F},$$

where $\mathcal{T}(q)$ is an m-flat subspace which includes a point q. This sliced structure is called the orthogonal foliation of $\mathcal{F}$ (for more detailed definitions, see Murata et al., 2004). In the following, we mainly consider a specific u-flat parametric model as a base space of the orthogonal foliation, which is called a u-model defined by

$$\mathcal{Q}_U = \left\{ q(x;\theta) = u(\theta \cdot \boldsymbol{t}(x) + s(x) - b(\theta)),\ \theta \in \Theta \subset R^d \right\},$$

where $s(x)$ is a function of x which does not depend on θ. Especially a u-flat subspace which includes a point q is denoted by $\mathcal{Q}_U(q)$.

On $\mathcal{T}$ and $\mathcal{Q}_U$, two kinds of projections are defined as follows (see Fig. 5).

Definition 3 (projections) *When the m-geodesic from a point p_0 is orthogonal to the u-flat subspace $\mathcal{Q}_U$ at the point $q^* \in \mathcal{Q}_U$, that is, the m-geodesic is orthogonal to any u-geodesic in $\mathcal{Q}_U$, q^* is called the m-projection from p_0 onto $\mathcal{Q}_U$. On the other hand, when the u-geodesic from a point q_0 is orthogonal to the m-flat subspace $\mathcal{T}$ at the point $q^* \in \mathcal{T}$, q^* is called the u-projection from q_0 onto $\mathcal{T}$.*

In Fig. 5, two kinds of projections are given as follows:

$$\operatorname*{argmin}_{q \in \mathcal{Q}_U} D_U(p_0, q) \quad (m\text{-projection}),$$
$$\operatorname*{argmin}_{p \in \mathcal{T}} D_U(p, q_0) \quad (u\text{-projection}).$$

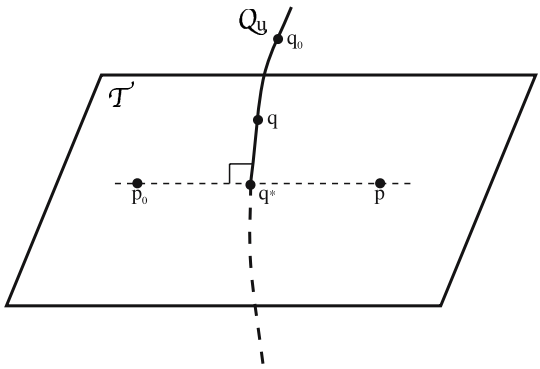


Fig. 5 Equivalence of two optimization problems

Particularly, in the case of the KL divergence, the *u*-projection is called the *e*-projection.

The following important duality is naturally derived from the orthogonal foliation and two kinds of projections (Fig. 5).

Theorem 2 (Murata et al., 2004) *Two optimization problems*

- *minimize* $D_U(p, q_0)$ *with respect to* $p \in \mathcal{T}(p_0)$ *for fixed* q_0, *(find an optimum in the m-flat subspace),*
- *minimize* $D_U(p_0, q)$ *with respect to* $q \in \mathcal{Q}_U(q_0)$ *for fixed* p_0, *(find an optimum in the u-flat subspace),*

give the same solution

$$q^* = \operatorname*{argmin}_{p \in \mathcal{T}(p_0)} D_U(p, q_0) = \operatorname*{argmin}_{q \in \mathcal{Q}_U(q_0)} D_U(p_0, q).$$

Based on the notion of flatness, two important mixture models are introduced as follows.

Definition 4 (finite mixture models) *Let* $\mathrm{P}_k(x; \phi_k)$ *be a point in* $\mathcal{P}$ *where* ϕ_k *is the parameter set of* P_k. *Then the mixture of K components* $\mathrm{P}_1, \ldots, \mathrm{P}_K$ *is called a finite mixture model. In the case that components are in m-representation, that is*

$$p(x; \theta) = \sum_{k=1}^{K} \pi_k \mathrm{P}_k(x; \phi_k), \tag{10}$$

where $\pi_k \geq 0$ *and* $\sum_{k=1}^{K} \pi_k = 1$, *the model is called an m-mixture, and in the case that components are in u-representation, that is*

$$p(x; \theta) = u\left(\sum_{k=1}^{K} \pi_k \xi(\mathrm{P}_k(x; \phi_k)) - c \right),$$

where c is a normalization constant, the model is called a u-mixture.

4 UM algorithm

4.1 EM algorithm with Bregman divergence

The EM algorithm is a popular method to estimate parameters of finite m-mixture models (McLachlan and Krishnan, 1997). In the finite m-mixture model with K components, given by Eq. (10), the observed data x is generated from one of the components. We introduce a hidden variable z in a space Z, that is

$$Z = \{z \in \{1, \ldots, K\}\},$$

where z indicates from which component of the mixture the observation arose. Let $h_k(z)$ be an indicator function, that is

$$h_k(z) = \begin{cases} 0 & \text{if } z \neq k, \\ 1 & \text{if } z = k. \end{cases} \tag{11}$$

Then, the joint distribution of x and z is given by

$$q(x, z; \theta) = \sum_{k=1}^{K} h_k(z) \pi_k \mathrm{P}_k(x; \phi_k), \tag{12}$$

where $\theta = \{\pi_1, \ldots, \pi_K, \phi_1, \ldots, \phi_K\}$, and on the condition that is $z = k$ holds, $q(x, k; \theta)$ is especially given by,

$$q(x, k; \theta) = \pi_k \mathrm{P}_k(x; \phi_k). \tag{13}$$

In the EM estimation of finite mixture models, we define the Q-function with an estimate $\theta^{(t)}$ at the t-th step, which is given by

$$\begin{aligned} Q(\theta; \theta^{(t)}) &= \frac{1}{N} \sum_{i=1}^{N} \sum_{z \in Z} q(z|x_i; \theta^{(t)}) \log q(x_i, z; \theta) \\ &= \frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{K} q(k|x_i; \theta^{(t)}) \log(\pi_k \mathrm{P}_k(x_i; \phi_k)), \end{aligned}$$

where $q(k|x_i; \theta^{(t)})$ is a conditional probability given by

$$q(k|x_i; \theta^{(t)}) = \frac{\pi_k^{(t)} \mathrm{P}_k(x_i; \phi_k^{(t)})}{\sum_{m=1}^{K} \pi_m^{(t)} \mathrm{P}_m(x_i; \phi_m^{(t)})}. \tag{14}$$

The E-step and the M-step are defined as follows.

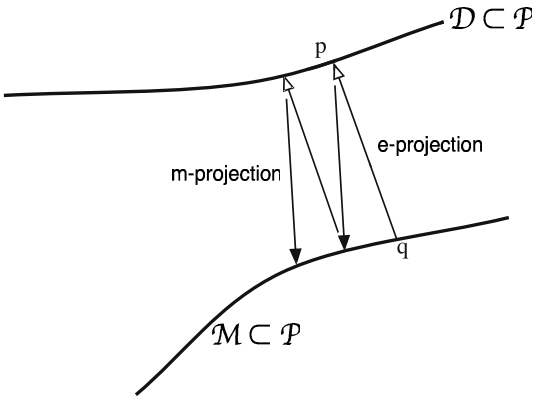


Fig. 6 A geometrical interpretation of the EM algorithm

EM algorithm

input data set $\{x_i; i = 1, \ldots, N\}$
initialize choose an initial parameter $\theta^{(0)}$
repeat from $t = 0$, until some conditions are satisfied

E-step calculate the Q-function from the previous estimate $\theta^{(t)}$:

$$Q(\theta; \theta^{(t)}) = \frac{1}{N} \sum_{i=1}^{N} \sum_{z \in Z} q(z|x_i; \theta^{(t)}) \log q(x_i, z; \theta).$$

M-step maximize the Q-function with respect to θ:

$$\theta^{(t+1)} = \operatorname*{argmax}_{\theta} Q(\theta, \theta^{(t)}).$$

output converged parameter vector θ

From a geometrical viewpoint, the dynamics of the EM algorithm is interpreted as shown in Fig. 6.

Let p be a probability density from which the data set is generated. We introduce a hidden variable z to p as well as the model q, that is

$$p(x, z; \psi) = \tilde{p}(x) p(z|x; \psi),$$

where $\psi = \{\check{\pi}, \check{\phi}\}$ gives the conditional probability $p(z|x; \psi)$. Especially $p(k|x; \psi)$ is given by

$$p(k|x; \psi) = \frac{\check{\pi}_k \mathrm{P}_k(x; \check{\phi}_k)}{\sum_{m=1}^{K} \check{\pi}_m \mathrm{P}_m(x; \check{\phi}_m)}.$$

For a geometrical interpretation, let us introduce a model manifold

$$\mathcal{M} = \{q(x, z; \theta)\} \subset \mathcal{P},$$

and a data manifold

$$\mathcal{D} = \{p(x, z; \psi)\} \subset \mathcal{P}.$$

The parameter θ specifies the point on the model manifold $\mathcal{M}$, and ψ specifies the point on the data manifold $\mathcal{D}$. Then the E-step and the M-step are interpreted as the *e*-projection and the *m*-projection between $\mathcal{D}$ and $\mathcal{M}$ in the function space, and the sequence of projections is called the *em* algorithm (In general, the EM algorithm and the *em* algorithm are not equivalent: an example where the EM and *em* algorithms do not coincide and the condition for their equivalence are presented in Amari (1995), Watanabe and Yamaguchi (2004)). Based on this geometrical interpretation, we generalize the EM algorithm with the Bregman divergence instead of the KL divergence as follows.

UM algorithm

input data set $\{x_i; i = 1, \ldots, N\}$
initialize choose an initial parameter $\theta^{(0)}$
repeat from $t = 0$, until some conditions are satisfied

u-**step** apply the u-projection from the previous estimate $\theta^{(t)}$ to $\mathcal{D}$, and obtain $\psi^{(t+1)}$:

$$\psi^{(t+1)} = \operatorname*{argmin}_{\psi} D_U(p(\psi), q(\theta^{(t)})).$$

m-**step** apply the m-projection from $\psi^{(t+1)}$ to $\mathcal{M}$, and obtain $\theta^{(t+1)}$:

$$\begin{aligned}\theta^{(t+1)} &= \operatorname*{argmin}_{\theta} D_U(p(\psi^{(t+1)}), q(\theta)) \\ &= \operatorname*{argmin}_{\theta} H_U(p(\psi^{(t+1)}), q(\theta)).\end{aligned}$$

output converged parameter vector θ

The algorithm is composed by a sequence of the u- and the m-projections as

$$\begin{aligned} q^{(t)} \in \mathcal{M} &\longrightarrow p^{(t+1)} \in \mathcal{D} \quad (u\text{-projection}), \\ p^{(t+1)} \in \mathcal{D} &\longrightarrow q^{(t+1)} \in \mathcal{M} \quad (m\text{-projection}). \end{aligned}$$

Due to the alternative optimization by the u-projection and the m-projection associated with the Bregman divergence, we call this algorithm *UM algorithm*.

With Theorem 2, every u-step satisfies the following condition

$$D_U(p(\psi^{(t+1)}), q(\theta^{(t)})) \leq D_U(p(\psi^{(t)}), q(\theta^{(t)})),$$

and every m-step satisfies the following condition

$$D_U(p(\psi^{(t+1)}), q(\theta^{(t+1)})) \leq D_U(p(\psi^{(t+1)}), q(\theta^{(t)})),$$

then $D_U(p(\psi), q(\theta))$ decreases monotonously as t increases. A monotonously decreasing sequence with a lower bound is guaranteed its convergence, therefore $D_U(p(\psi), q(\theta))$ converges to its minimum, or at least its local minima, with the UM algorithm.

At the u-step, the u-projection is given by

$$\psi^{(t+1)} = \operatorname*{argmin}_{\psi} \int \sum_{z \in Z} \Big(U(\xi(q(x,z;\theta^{(t)}))) - U(\xi(p(x,z;\psi))) - p(x,z;\psi) \left(\xi(q(x,z;\theta^{(t)})) - \xi(p(x,z;\psi)) \right) \Big) \mathrm{d}\mu(x).$$

The minimizer $\psi^{(t+1)}$ is the solution of

$$\begin{aligned} &\frac{\partial D_U(p(\psi), q(\theta^{(t)}))}{\partial \psi} \\ &= \sum_{i=1}^{N} \tilde{p}(x_i) \sum_{z \in Z} \frac{\partial p(z|x_i;\psi)}{\partial \psi} \left(\xi(q(x_i;\theta^{(t)}) q(z|x_i;\theta^{(t)})) - \xi(\tilde{p}(x_i) p(z|x_i;\psi)) \right) \\ &= 0, \end{aligned}$$

and a sufficient condition is given by

$$\xi(\tilde{p}(x_i) p(z|x_i;\psi^{(t+1)})) = \xi(q(x_i;\theta^{(t)}) q(z|x_i;\theta^{(t)})) \quad \text{where } \tilde{p}(x_i) > 0, \tag{15}$$

which leads to the condition

$$\begin{aligned} p(z|x_i;\psi^{(t+1)}) &= \frac{q(x_i;\theta^{(t)})}{\tilde{p}(x_i)} q(z|x_i;\theta^{(t)}) \\ &\propto q(z|x_i;\theta^{(t)}). \end{aligned}$$

Knowing that conditional probabilities p and q satisfy the condition

$$\sum_{z \in Z} p(z|x_i;\psi^{(t+1)}) = \sum_{z \in Z} q(z|x_i;\theta^{(t)}) = 1,$$

if p and q are described with the same model, then the u-projection in the u-step results in

$$p(z|x_i;\psi^{(t+1)}) = q(z|x_i;\theta^{(t)}) \tag{16}$$

where $q(z|x_i;\theta^{(t)})$ is given by Eq. (14), which is the same as $\psi^{(t+1)} = \theta^{(t)}$ in this case. Note that in the case that the minimizer $\psi^{(t+1)}$ is not lead from the condition Eq. (15), that is conditional probabilities p and q are different models, then Eq. (16) is not appropriate.

On the other hand, the m-step for the estimation of the finite mixture model is written as

$$\begin{aligned}\theta^{(t+1)} &= \operatorname*{argmin}_{\theta} \int \sum_{z\in Z}\Big(U(\xi(q(x,z;\theta))) \\ &\quad - p(x,z;\psi^{(t+1)})\xi(q(x,z;\theta))\Big)\mathrm{d}\mu(x) \\ &= \operatorname*{argmin}_{\pi,\phi} \sum_{k=1}^{K} \int \Big(U(\xi(\pi_k \mathrm{P}_k(x;\phi_k))) \\ &\quad - p(x,k;\psi^{(t+1)})\xi(\pi_k \mathrm{P}_k(x;\phi_k))\Big)\mathrm{d}\mu(x), \end{aligned} \tag{17}$$

by using Eq. (13). Since simultaneous optimization of π and ϕ is highly nonlinear and generally difficult, we adopt the following two-step optimization as an approximation of Eq. (17)

$$\begin{aligned}\phi_k^{(t+1)} &= \operatorname*{argmin}_{\phi_k} \int \Big(U(\xi(\pi_k^{(t)} \mathrm{P}_k(x;\phi_k))) \\ &\quad - p(x,k;\psi^{(t+1)})\xi(\pi_k^{(t)} \mathrm{P}_k(x;\phi_k))\Big)\mathrm{d}\mu(x), \end{aligned} \tag{18}$$

$$\begin{aligned}\pi^{(t+1)} &= \operatorname*{argmin}_{\pi} \sum_{k=1}^{K} \int \Big(U(\xi(\pi_k \mathrm{P}_k(x;\phi_k^{(t+1)}))) \\ &\quad - p(x,k;\psi^{(t+1)})\xi(\pi_k \mathrm{P}_k(x;\phi_k^{(t+1)}))\Big)\mathrm{d}\mu(x). \end{aligned} \tag{19}$$

This approximation is the same as the Expectation/Conditional Maximization algorithm (see Meng and Rubin 1993). Moreover, the estimation of $\pi^{(t+1)}$ in Eq. (19) is approximated and simplified based on the geometrical structure of models as discussed in the next subsection.

4.2 Simplification in m-step

4.2.1 Simplification with $\mathcal{P}$ method

In Eq. (19) at the m-step, $\pi_k^{(t+1)}$ is given by the solution of the following Lagrange equation,

$$\begin{aligned}
&\frac{\partial(H_U(p(\psi^{(t+1)}), q(\theta)) + \lambda_\pi(1 - \sum_{m=1}^K \pi_m))}{\partial \pi_k} \\
&= \int \pi_k \mathrm{P}_k(x; \phi_k^{(t+1)})^2 \xi'(\pi_k \mathrm{P}_k(x; \phi_k^{(t+1)})) \mathrm{d}\mu(x) \\
&\quad - \int p(x, k; \psi^{(t+1)}) \xi'(\pi_k \mathrm{P}_k(x; \phi_k^{(t+1)})) \mathrm{P}_k(x; \phi_k^{(t+1)}) \mathrm{d}\mu(x) - \lambda_\pi \\
&= 0,
\end{aligned} \tag{20}$$

where λ_π is a Lagrange multiplier. From Eq. (20), $\pi_k^{(t+1)}$ is given by

$$\begin{aligned}
\pi_k^{(t+1)} &= \frac{\int \mathrm{P}_k(x; \phi_k^{(t+1)}) \xi'(\pi_k^{(t+1)} \mathrm{P}_k(x; \phi_k^{(t+1)})) p(x, k; \psi^{(t+1)}) \mathrm{d}\mu(x) + \lambda_\pi}{\int \mathrm{P}_k(x; \phi_k^{(t+1)})^2 \xi'(\pi_k^{(t+1)} \mathrm{P}_k(x; \phi_k^{(t+1)})) \mathrm{d}\mu(x)} \\
\sum_{m=1}^K \pi_m^{(t+1)} &= 1.
\end{aligned} \tag{21}$$

On the right-hand side of Eq. (21), $\pi_k^{(t+1)}$ is required to derive itself, then to solve Eq. (21), a recursive procedure is needed. A simple approximation to obtain $\pi_k^{(t+1)}$ is made by replacing $\pi_k^{(t+1)}$ with $\pi_k^{(t)}$ on the right-hand side, that is given by

$$\begin{aligned}
\pi_k^{(t+1)} &= \frac{\int \mathrm{P}_k(x; \phi_k^{(t+1)}) \xi'(\pi_k^{(t)} \mathrm{P}_k(x; \phi_k^{(t+1)})) p(x, k; \psi^{(t+1)}) \mathrm{d}\mu(x) + \lambda_\pi}{\int \mathrm{P}_k(x; \phi_k^{(t+1)})^2 \xi'(\pi_k^{(t)} \mathrm{P}_k(x; \phi_k^{(t+1)})) \mathrm{d}\mu(x)} \\
\sum_{m=1}^K \pi_m^{(t+1)} &= 1.
\end{aligned} \tag{22}$$

In general, we have to solve Eq. (22) with respect to both π and λ_π, here we furthermore propose the following simplification instead of the complete optimization. Let $\mathcal{M}_\mathcal{F}$ be a subspace in $\mathcal{F}$, defined as

$$\mathcal{M}_\mathcal{F} = \left\{ q_\mathcal{F}(x, z; \theta_\mathcal{F}) = \sum_{k=1}^K h_k(z) w_k \mathrm{P}_k(x; \phi_k) \right\} \subset \mathcal{F},$$

where $\theta_\mathcal{F} = \{w, \phi\}$ and $w = \{w_1, \ldots, w_K\}$ be the mixture weight instead of π which satisfies $w_k \geq 0$ for all k. In general, $\mathcal{M}_\mathcal{F}$ is not in $\mathcal{P}$ because $\sum_{k=1}^K w_k = 1$ is not assumed. As mentioned before, the Bregman divergence is defined as a pseudo-distance to measure discrepancy between two functions in $\mathcal{F}$, therefore the calculation in the m-step can be modified by applying the m-projection from p on $\mathcal{D} \subset \mathcal{P}$ to $q_\mathcal{F}$ on $\mathcal{M}_\mathcal{F} \subset \mathcal{F}$. By this modification, the calculation in Eq. (22)

is replaced by

$$w_k^{(t+1)} = \frac{\int \mathrm{P}_k(x;\phi_k^{(t+1)})\xi'(w_k^{(t)}\mathrm{P}_k(x;\phi_k^{(t+1)}))p(x,k;\psi^{(t+1)})\mathrm{d}\mu(x)}{\int \mathrm{P}_k(x;\phi_k^{(t+1)})^2\xi'(w_k^{(t)}\mathrm{P}_k(x;\phi_k^{(t+1)}))\mathrm{d}\mu(x)},$$

which does not include λ_π. Therefore the weight parameter w is obtained without solving the minimization problem. To proceed to the next u-step, $q_{\mathcal{F}}$ on $\mathcal{M}_{\mathcal{F}} \subset \mathcal{F}$ should be projected to an appropriate point on $\mathcal{M} \subset \mathcal{P}$. This is simply achieved by applying a projection from $q_{\mathcal{F}} \in \mathcal{F}$ to $q \in \mathcal{P}$.

For example, $\pi^{(t+1)}$ is estimated from $w^{(t+1)}$ by applying the u-projection, given by

$$\begin{aligned}\pi^{(t+1)} &= \operatorname*{argmin}_{\pi} D_U(\pi, w^{(t+1)}) \\ &= \operatorname*{argmin}_{\pi} \left[\sum_{k=1}^{K}\left(U(\xi(w_k^{(t+1)})) - U(\xi(\pi_k)) - \pi_k\left\{\xi(w_k^{(t+1)}) - \xi(\pi_k)\right\}\right)\right. \\ &\quad \left. +\lambda\left(1 - \sum_{k=1}^{K}\pi_k\right)\right],\end{aligned}$$

and $\pi^{(t+1)}$ is given by the solution of

$$\frac{\partial D_U(\pi, w^{(t+1)})}{\partial \pi_k} = \xi(\pi_k) - \xi(w_k^{(t+1)}) - \lambda = 0,$$

therefore

$$\pi_k^{(t+1)} = u(\xi(w_k^{(t+1)}) + \lambda), \tag{23}$$

and λ is determined so as to satisfy the normalization condition

$$\sum_{m=1}^{K} \pi_m^{(t+1)} = 1.$$

In this procedure, we only have to solve the one-dimensional minimization problem with respect to λ.

Another way of estimating $\pi^{(t+1)}$ is applying the m-projection, that is

$$\begin{aligned}\pi^{(t+1)} &= \operatorname*{argmin}_{\pi} D_U(w^{(t+1)}, \pi) = \operatorname*{argmin}_{\pi} H_U(w^{(t+1)}, \pi) \\ &= \operatorname*{argmin}_{\pi} \left[\sum_{k=1}^{K}\left(U(\xi(\pi_k)) - w_k^{(t+1)}\xi(\pi_k)\right) + \lambda\left(1 - \sum_{k=1}^{K}\pi_k\right)\right],\end{aligned}$$

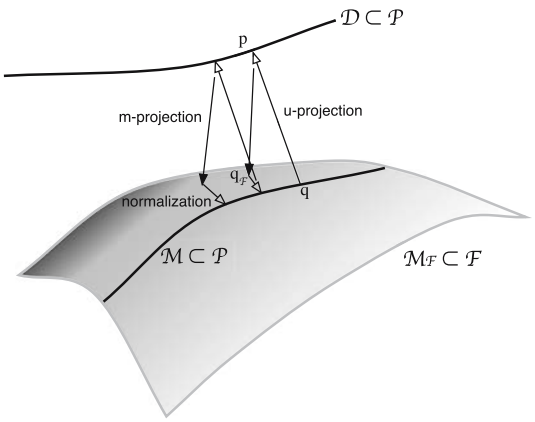


Fig. 7 A geometrical interpretation of the UM algorithm with $\mathcal{P}$ method. Note that $\mathcal{M} \subset \mathcal{M}_{\mathcal{F}}$ holds

and $\pi^{(t+1)}$ is given by the solution of

$$\frac{\partial H_U(w^{(t+1)}, \pi)}{\partial \pi_k} = \pi_k \xi'(\pi_k) - w_k^{(t+1)} \xi'(\pi_k) - \lambda = 0,$$

therefore

$$\pi_k^{(t+1)} = w_k^{(t+1)} + \frac{\lambda}{\xi'(\pi_k^{(t+1)})}. \tag{24}$$

In this case, a recursive procedure is required to obtain $\pi_k^{(t+1)}$, because the equation has $\pi_k^{(t+1)}$ on the right-hand side.

In either case, q is able to be estimated from $q_{\mathcal{F}}$. The simplified algorithm is composed of a sequence of projections as follows.

$$\begin{aligned} q^{(t)} \in \mathcal{M} &\longrightarrow p^{(t+1)} \in \mathcal{D} \quad (u\text{-projection to } \mathcal{P}) \\ p^{(t+1)} \in \mathcal{D} &\longrightarrow q_{\mathcal{F}}^{(t+1)} \in \mathcal{M}_{\mathcal{F}} \quad (m\text{-projection to } \mathcal{F}) \\ q_{\mathcal{F}}^{(t+1)} \in \mathcal{M}_{\mathcal{F}} &\longrightarrow q^{(t+1)} \in \mathcal{M} \quad (\text{normalization by projection to } \mathcal{P}) \end{aligned}$$

And this approximation is also interpreted geometrically as shown in Fig. 7.

In the following, this approximation is denoted by "$\mathcal{P}$ method".

4.2.2 Simplification with $\mathcal{F}$ method

In $\mathcal{P}$ method, a projection from $\mathcal{M}_{\mathcal{F}}$ to $\mathcal{M}$ is applied every step. When the drastic approximations which are

$$\operatorname*{argmin}_{p\in\mathcal{D}} D_U(p, q^{(t)}) \simeq \operatorname*{argmin}_{p\in\mathcal{D}} D_U(p, q^{(t)}_{\mathcal{F}}) \tag{25}$$

$$\operatorname*{argmin}_{q\in\mathcal{M}} D_U(p^{(t)}, q) \simeq \begin{cases} \operatorname*{argmin}_{q\in\mathcal{M}} D_U(q, q^{(t)}_{\mathcal{F}}) & (\mathcal{P} \text{ method w. } u\text{-proj.}), \\ \operatorname*{argmin}_{q\in\mathcal{M}} D_U(q^{(t)}_{\mathcal{F}}, q) & (\mathcal{P} \text{ method w. } m\text{-proj.}), \end{cases} \tag{26}$$

hold for q and $q_{\mathcal{F}}$, the algorithm can be simplified by ignoring a projection from $\mathcal{M}_{\mathcal{F}}$ to $\mathcal{M}$ in $\mathcal{P}$ method. Roughly speaking, the above conditions Eqs. (25) and (26) mean q and $q_{\mathcal{F}}$ are close each other,

At the u-step of this simplification, the posterior of the hidden variable z, that is denoted by $q_{\mathcal{F}}(z|x_i;\theta^{(t)}_{\mathcal{F}})$, is needed for obtaining the u-projection from $\mathcal{M}_{\mathcal{F}}$ to $\mathcal{P}$. It is defined by

$$q_{\mathcal{F}}(k|x_i;\theta^{(t)}_{\mathcal{F}}) = \frac{w^{(t)}_k \mathrm{P}_k(x_i;\phi^{(t)}_k)}{\sum_{m=1}^{K} w^{(t)}_m \mathrm{P}_m(x_i;\phi^{(t)}_m)} \quad \text{at the } u\text{-step}$$

and we can simply use

$$p(z|x_i;\psi^{(t+1)}) = q_{\mathcal{F}}(z|x_i;\theta^{(t)}_{\mathcal{F}}),$$

instead of Eq. (16).

In this approximation, a sequence of projections is written as follows:

$$q^{(t)}_{\mathcal{F}} \in \mathcal{M}_{\mathcal{F}} \longrightarrow p^{(t+1)} \in \mathcal{D} \quad (u\text{-projection to } \mathcal{P}),$$

$$p^{(t+1)} \in \mathcal{D} \longrightarrow q^{(t+1)}_{\mathcal{F}} \in \mathcal{M}_{\mathcal{F}} \quad (m\text{-projection to } \mathcal{F}),$$

and like the discussion of the convergence of the UM algorithm, $q_{\mathcal{F}}$ converges to local minima in $\mathcal{M}_{\mathcal{F}} \subset \mathcal{F}$ by this sequence of projections. This approximation is called "$\mathcal{F}$ method".

In $\mathcal{F}$ method, we do not have to normalize the model $q_{\mathcal{F}} \in \mathcal{M}_{\mathcal{F}} \subset \mathcal{F}$ to $\mathcal{M} \subset \mathcal{P}$ before u-steps every time. Once the converged $q_{\mathcal{F}} \in \mathcal{M}_{\mathcal{F}}$ is found by using $\mathcal{F}$ method, the convergence point $q \in \mathcal{M}$ can be found by the u-projection or the m-projection from $\mathcal{M}_{\mathcal{F}}$ to $\mathcal{M}$ like m-step in $\mathcal{P}$ method. And the convergence of this drastic approximation is expected to be faster than $\mathcal{P}$ method because the extended model manifold $\mathcal{M}_{\mathcal{F}}$ is less restricted than $\mathcal{M} \subset \mathcal{P}$.

The geometrical interpretation of this approximation is shown in Fig. 8.

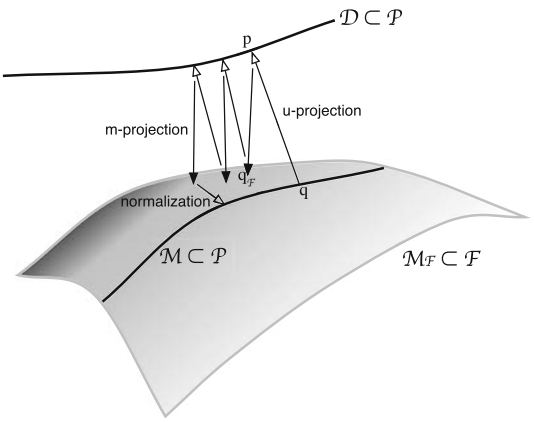


Fig. 8 A geometrical interpretation of the UM algorithm with $\mathcal{F}$ method. Note that $\mathcal{M} \subset \mathcal{M}_{\mathcal{F}}$ holds

4.2.3 Special case of simplification

As for the conventional EM algorithm, the approximation Eq. (22) is not required for calculating $\pi_k^{(t+1)}$, and $\pi_k^{(t+1)}$ is simply given by

$$\begin{aligned}\pi_k^{(t+1)} &= \int \tilde{p}(x) p(k|x; \psi^{(t+1)}) \mathrm{d}\mu(x) \\ &= \frac{1}{N} \sum_{i=1}^{N} p(k|x_i; \psi^{(t+1)}).\end{aligned}$$

This is because that $\xi'(\pi_k \mathrm{P}_k(x; \phi_k))$ is decomposed as follows,

$$\begin{aligned}\xi'(\pi_k \mathrm{P}_k(x; \phi_k)) &= \frac{1}{\pi_k \mathrm{P}_k(x; \phi_k)} \\ &= \frac{1}{\pi_k} \cdot \frac{1}{\mathrm{P}_k(x; \phi_k)},\end{aligned}$$

and all $\pi_k^{(t+1)}$ on the right-hand side in Eq. (21) is eliminated in the case of $\xi(\cdot) = \log(\cdot)$. With this fact, Eq. (21) can be calculated without recursive procedures when $\xi'(\pi_k \mathrm{P}_k(x; \phi_k))$ is decomposed into the product of two functions $f_1(\cdot)$ and $f_2(\cdot)$ as

$$\xi'(\pi_k \mathrm{P}_k(x; \phi_k)) = f_1(\pi_k) f_2(\mathrm{P}_k(x; \phi_k)).$$

From the commutativity of $f_1(\cdot)$ and $f_2(\cdot)$, and with taking the case of $\pi_k = 1$ into consideration, the condition that the right-hand side of Eq. (21) does not include $\pi_k^{(t+1)}$ is given by

$$\xi'(\pi_k \mathrm{P}_k(x; \phi_k)) = \xi'(\pi_k) \xi'(\mathrm{P}_k(x; \phi_k)). \tag{27}$$

The β-divergence is a concrete case that satisfies Eq. (27), that is

$$(\pi_k \mathrm{P}_k(x;\phi_k))^{\beta-1} = \pi_k^{\beta-1}\mathrm{P}_k(x;\phi_k)^{\beta-1}.$$

Additionally, if the extended model manifold $\mathcal{M}_{\mathcal{F}}$ is u-flat, then $\mathcal{P}$ method with the m-projection from $q_{\mathcal{F}}$ to q is equivalent to the complete minimization of Eq. (19). Because Pythagorean relation

$$D_U(p^{(t+1)}, q^{(t+1)}) = D_U(p^{(t+1)}, q_{\mathcal{F}}^{(t+1)}) + D_U(q_{\mathcal{F}}^{(t+1)}, q^{(t+1)})$$

holds from Theorem 1, and $p^{(t+1)}$ and $q^{(t+1)}$ satisfies the conditions

$$p^{(t+1)} = \operatorname*{argmin}_{p\in\mathcal{D}} D_U(p, q^{(t)}), \tag{28}$$

$$q^{(t+1)} = \operatorname*{argmin}_{q\in\mathcal{M}} D_U(q_{\mathcal{F}}^{(t+1)}, q) = \operatorname*{argmin}_{q\in\mathcal{M}} D_U(p^{(t+1)}, q), \tag{29}$$

$\mathcal{P}$ method with the m-projection from $q_{\mathcal{F}}$ to q is exactly equivalent to the complete optimization of Eq. (19).

However $\mathcal{P}$ method with the m-projection is as difficult as the complete optimization of Eq. (18) because Eq. (19) also requires the optimization with respect to both π and λ to obtain $\pi^{(t+1)}$. On the other hand, $\mathcal{P}$ method with the u-projection which is given by Eq. (23) requires the optimization with respect to λ only. Moreover, the u-projection is a natural way to normalize a u-model. Therefore, $\mathcal{P}$ method with the u-projection is applied in our experiments shown in the following section.

These exact cases in $\mathcal{P}$ method also holds for the β-divergence: $\mathcal{P}$ method with the m-projection is exactly equivalent to the complete optimization which is given by Eq. (19). We explain that the β-divergence satisfies the condition of Eq. (27) in the next section.

5 Example: UM algorithm with β-divergence

Let A and B be two categorical variables, A with I categories, and B with J categories. An independency between A and B is denoted by

$$p(a_i, b_j) = p(a_i)p(b_j), \tag{30}$$

where $p(a_i, b_j)$ is the joint probability of the event $(A, B) = (a_i, b_j)$ on a contingency table.

Let K be the number of prepared independent tables to express the joint probability distribution between A and B. The joint probability $p(a_i, b_j)$ is

described as

$$\begin{aligned} p(a_i, b_j) &= \sum_{k=1}^{K} \pi_k \mathrm{P}_k(a_i, b_j; \phi_k) \\ &= \sum_{k=1}^{K} \pi_k \mathrm{P}_k(a_i) \mathrm{P}_k(b_j), \end{aligned}$$

where $\mathrm{P}_k(a_i)$ and $\mathrm{P}_k(b_j)$ are marginal probability distributions on the k-th independent table. This mixture model is called the aspect model (Hoffmann, 1999), or the latent class model (Agresti, 2002, Chap. 13). To estimate the mixture of independent tables, we can apply the UM algorithm with the Bregman divergence. Here we apply the UM algorithm with the β-divergence, given by

$$D_\beta(p, q) = \sum_{i=1}^{I} \sum_{j=1}^{J} \left(\frac{q(a_i, b_j)^{\beta+1}}{\beta+1} - \frac{p(a_i, b_j) q(a_i, b_j)^{\beta}}{\beta} + \frac{p(a_i, b_j)^{\beta+1}}{\beta(\beta+1)} \right)$$

as an actual example of Eq. (3). With the β-divergence, all $\pi^{(t+1)}$ on the right-hand side in Eq. (21) is canceled, because

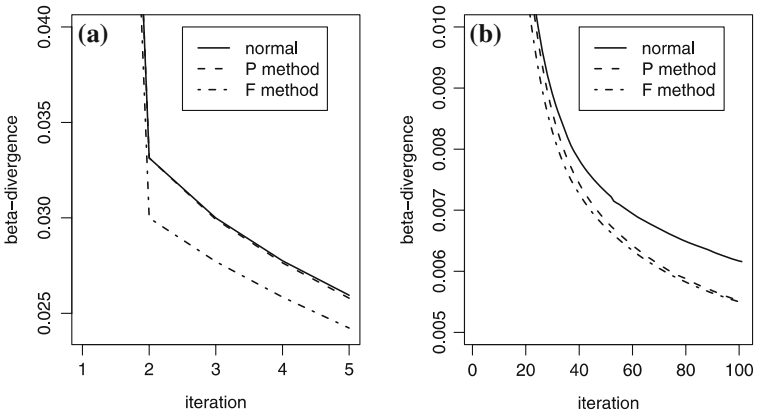
$$\begin{aligned} \xi'(\pi_k \mathrm{P_k}(\mathrm{x}; \phi_\mathrm{k})) &= \pi_k^{\beta-1} \mathrm{P_k}(\mathrm{x}; \phi_\mathrm{k})^{\beta-1} \\ &= \xi'(\pi_k) \xi'(\mathrm{P_k}(\mathrm{x}; \phi_\mathrm{k})), \end{aligned}$$

and Eq. (27) holds. Moreover, the extended model manifold with respect to w is given by

$$\begin{aligned} q_{\mathcal{F}}(x, z; w_k) &= u\left(\xi \left(\sum_{k=1}^{K} h_k(z) w_k \mathrm{P}_k(x) \right) \right) \\ &= u\left(\sum_{k=1}^{K} h_k(z) \left(\frac{w_k^\beta \mathrm{P}_k(x)^\beta}{\beta} - \frac{1}{\beta} \right) \right) \\ &= u\left(\sum_{k=1}^{K} \frac{w_k^\beta}{\beta} h_k(z) \mathrm{P}_k(x)^\beta - \frac{1}{\beta} \right) \\ &= u\left(\sum_{k=1}^{K} \acute{w}_k t_k(x, z) - b \right) \end{aligned}$$

where $\acute{w}_k = \frac{w_k^\beta}{\beta}$, $t_k(x, z) = h_k(z) \mathrm{P}_k(x)^\beta$ and $b = \frac{1}{\beta}$. Note that this model $q_{\mathcal{F}}$ is u-flat.

In this experiment, the distance between the mixture model $q(A, B)$ with $K = 10$ and the empirical probability distribution $\tilde{p}(A, B)$ where $I = 20$ and



first 5 steps. estimation results.

Fig. 9 The β-divergence $D_\beta(\tilde{p}, q)$ with $\beta = 0.1$ against the number of *um*-steps

Table 2 Average CPU time in m-step calculations

	Normal	$\mathcal{P}$ method	$\mathcal{F}$ method
CPU time (s)	0.6749	0.0091	0.0074

$J = 20$, was measured based on the β-divergence $D_\beta(\tilde{p}, q)$ with $\beta = 0.1$ in order to compare the estimating methods.

At first, 20×20 contingency table was generated subject to a randomly generated distribution p, which is the mixture model with 10 independent tables.

The total sample size in a contingency table is 4000, and initial values of parameters to be estimated are chosen at random from the uniform distribution on $[0, 1]$, and normalized such as $\sum_{k=1}^{K} \pi_k = 1$, $\sum_{i=1}^{I} \mathrm{P}_k(a_i) = 1$ and $\sum_{j=1}^{J} \mathrm{P}_k(b_j) = 1$ respectively. Figure 9a compares $D_\beta(\tilde{p}, q)$ for the first 5 steps of the normal UM method with two methods. In $\mathcal{P}$ method, $q_{\mathcal{F}}$ was normalized by the u-projection. For $\mathcal{F}$ method, $q_{\mathcal{F}}$ was normalized to $q \in \mathcal{M}$ by the u-projection at every m-step in order to evaluate $D_\beta(\tilde{p}, q)$, while the algorithm is executed in $\mathcal{M}_{\mathcal{F}}$. The graph shows the decrease of $D_\beta(\tilde{p}, q)$ for all the methods. $\mathcal{P}$ method with the β-divergence shows almost the same result as the normal method. In addition, $\mathcal{F}$ method converges slightly faster than the other methods. This property comes from the fact that minimizers are searched in $\mathcal{F}$ with less restriction at each step than in $\mathcal{P}$. These results show that $\mathcal{P}$ and $\mathcal{F}$ methods are working appropriately. Figure 9b depicts the evolution of $D_\beta(\tilde{p}, q)$ for 100 steps, and it shows $\mathcal{P}$ and $\mathcal{F}$ methods converge almost equivalently after 100 steps.

Table 2 summarizes the average time for calculation of π (or w, in the case of $\mathcal{F}$ method) of 100 m-steps .

The models were estimated on a Power Mac G5 which has 2 GHz PowerPC G5 dual processors with 512 KB caches for each, and the time spent for I/O

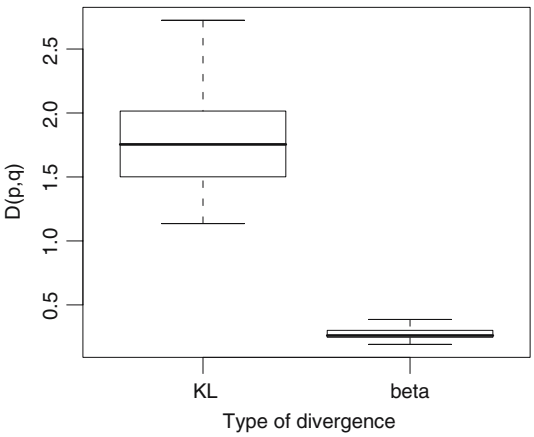


Fig. 10 Boxplots of the KL divergence $D_{\mathrm{KL}}(p, q)$ which evaluates discrepancy between the true distribution p and the model q estimated with small samples. The *left* boxplot is the result of the conventional EM estimation with the KL divergence. The *right* one is that of the UM estimation with the β-divergence ($\beta = 0.1$)

is excluded from CPU time. To obtain $\pi^{(t+1)}$ at the m-step in the normal UM estimation, we use the "L-BFGS-B" method by using "optim" routine equipped in R-language version 2.1.1 (see R Development Core Team, 2005) with lower bound $\pi_k \geq 0$, and minimize the objective function

$$D_\beta(p, q) + 10^4 \times \left(1 - \sum_{k=1}^{K} \pi_k\right)^2 .$$

From the table, $\mathcal{F}$ method is more effective than the normal method with the undevised optimization in view of computational cost. The average CPU time of $\mathcal{F}$ method is about 1.2 times faster than that of $\mathcal{P}$ method. This difference is important for estimation of probability tables especially in huge graphical models.

The β-divergence possesses robustness to outliers in several situations (Minami and Eguchi, 2002). In our experiments, the UM estimation with the β-divergence shows the robustness to the small sample data set (for detail, see Fujimoto and Murata, 2006). For example, on the same experimental setup with the sample size is 400 which is very small compared with the number of the parameters, the conventional EM algorithm (with the KL divergence) shows over-fits to the sample though the UM algorithm with the β-divergence ($\beta = 0.1$) does not show over-fits remarkably. We evaluated the same procedure 20 times with different small sample sets. In the UM algorithm, the m-step is also achieved by $\mathcal{P}$ method with the u-projection. Figure 10 shows boxplots of the KL divergence $D_{\mathrm{KL}}(p, q)$ which evaluates discrepancy between the true distribution p and the model q estimated with small samples. The estimation result with the β-divergence is much better than that with the KL divergence as shown in the figure.

6 Conclusion

In this paper, we have generalized the EM algorithm with the Bregman divergence from a geometrical viewpoint. In the UM estimation of finite m-mixture models, the m-step for estimating the mixture ratio parameter tends to be complicated with the Bregman divergence. We have proposed two methods to simplify the calculation at the m-step, and compared their computational costs and estimation results on a toy example, and the results show appropriateness of the proposed methods. We have also explained the relationship between $\mathcal{P}$ method and the undevised UM procedure in the particular situation. The β-divergence applied in our experiments is a particularly convenient case.

In this paper, we focused on the m-mixture model, but there is another model, the u-mixture model, which is closely related with the Bregman divergence. In the case of the u-mixture model which is the other mixture model derived from the Bregman divergence, the parameter set is not estimated simply with the UM algorithm, this is because data generated from u-mixtures are not simply interpreted by introducing hidden variables like m-mixtures. Instead of using the UM algorithm, we can estimate the u-mixture model with alternative approaches, though this remains as a future work.

References

Agresti, A. (2002). *Categorical data analysis* (pp. 538–575). New York: Wiley-Interscience.

Amari, S. (1995). Information geometry of the EM and em algorithms for neural networks. *Neural Networks, 8*(9), 1379–1408.

Amari, S., Nagaoka, H. (2000). *Methods of information geometry*. Providence, RI: American Mathematical Society.

Fujimoto, Y., Murata, N. (2006). Robust estimation for mixture of probability tables based on β-likelihood. In *Proceedings of the sixth SIAM conference on data mining* (pp. 519–523).

Fujisawa, H., Eguchi, S. (2005). Robust estimation in the normal mixture model. *Journal of Statistical Planning and Inference, 136*, 3989–4011.

Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., Stahel, W. A. (1986). *Robust statistics*. New York: Wiley.

Hofmann, T. (1999). Probabilistic latent semantic analysis. In *Proceedings of uncertainty in artificial intelligence, UAI '99* (pp. 289–296).

McLachlan, G. J., Krishnan, T. (1997). *The EM algorithm and extensions*. New York: John Wiley.

Meng, X. L., Rubin, D. B. (1993). Maximum likelihood estimation via the ECM algorithm: a general framework. *Biometrika, 80*(2), 267–278.

Minami, M., Eguchi, S. (2002). Robust blind source separation by beta divergence. *Neural Computation, 14*, 1859–1886.

Murata, N., Takenouchi, T., Kanamori, T., Eguchi, S. (2004). Information geometry of U-Boost and Bregman divergence. *Neural Computation, 16*(7), 1437–1481.

R Development Core Team (2005). R: A language and environment for statistical computing. http://www.R-project.org. R Foundation for Statistical Computing.

Takenouchi, T., Eguchi, S. (2004). Robustifying AdaBoost by adding the naive error rate. *Neural Computation, 16*(4), 767–787.

Takenouchi, T. (2005). Robust boosting algorithm for multiclass classification by eta-divergence. In *Proceedings of 2nd international symposium on information geometry and its applications* (pp. 12–16).

Watanabe, M., Yamaguchi, K. (2004). EM algorithm in neural network learning. In N. Murata, S. Ikeda (Eds.) *The EM algorithm and related statistical models* (pp. 95–125). New York: Marcel Dekker.