Information Geometry of U-Boost and Bregman Divergence

Noboru Murata
School of Science and Engineering, Waseda University

Takashi Takenouchi
Department of Statistical Science, Graduate University of Advanced Studies

Takafumi Kanamori
Department of Mathematical and Computing Sciences, Tokyo Institute of Technology

Shinto Eguchi
Institute of Statistical Mathematics, Japan and
Department of Statistical Science, Graduate University of Advanced Studies

Abstract

We aim to extend from AdaBoost to U-Boost in the paradigm to build up a stronger classification machine in a set of weak learning machines. A geometric understanding for the Bregman divergence defined by a generic function U being convex leads to U-Boost method in the framework of information geometry for the finite measure functions over the label set. We propose two versions of U-Boost learning algorithms by taking whether the domain is restricted to the space of probability functions or not. In the sequential step we observe that the two adjacent and the initial classifiers associate with a right triangle in the scale via the Bregman divergence, called the Pythagorean relation. This leads to a mild convergence property of the U-Boost algorithm as seen in the EM algorithm. Statistical discussion for consistency and robustness elucidates the properties of U-Boost methods based on a probabilistic assumption for a training data.

1 Introduction

In the last decade, several novel developments for classification and pattern recognition have been done mainly along statistical learning theory (see for example, MacLachlan, 1992; Bishop, 1995; Vapnik, 1995; Hastie et al., 2001). Several important approaches have been proposed and implemented into feasible computational algorithms. One promising direction is "boosting" which is a method of combining many learning machines trained by simple learning algorithms. Theoretical researches on "boosting" have been started from the question by Kearns and Valiant (1988):

> "Can a *weak learner* which is a bit better than random guessing be *boosted* into an arbitrarily accurate *strong learner*?"

The first interesting answer is given by Schapire (1990), who has proved that it is possible to construct an accurate machine by combining three machines trained by different examples, which are sequentially sampled and filtered by previous trained machines. Intuitively speaking, the key idea of boosting algorithm is to assort important and unimportant examples according whether machines are good at or weak in learning those examples. The procedures for sieving examples are summarized as following three types:

- **filtering:** new examples are sampled and filtered by the previous trained machines so that difficult examples are collected as many as easy examples (Schapire, 1990).
- **resampling:** examples are sampled from given examples repeatedly so that difficult examples are chosen with high probability (Freund, 1995; Domingo and Watanabe, 2000).
- **reweighting:** given examples are weighted so that difficult examples severely affect the error (Freund and Schapire, 1997; Friedman et al., 2000).

In this paper, we focus on the reweighting method including AdaBoost (Freund and Schapire, 1997). Lebanon and Lafferty (2001) give a geometric consideration of the extended Kullback-Leibler divergence which lead to a close relation between AdaBoost and logistic discrimination. We propose a class of boosting algorithms, U-Boost, which is naturally derived from the Bregman divergence. This proposal gives an extension of the geometry discussed by Lebanon and Lafferty, and elucidates that the Bregman divergence associates with a pair of the normalized and unnormalized U-Boost from the viewpoint of information geometry.

This paper is organized as follows. In section 2, we briefly review the AdaBoost algorithm and its geometrical understanding by Lebanon and Lafferty (2001), In section 3, we introduce the Bregman divergence in order to give a statistical framework of boosting algorithms, and discuss some properties in the sense of the information geometry. Then we propose the U-Boost algorithm based on the Bregman divergence and discuss its consistency, efficiency and robustness in section 4. We will give some illustrative examples with numerical simulations in section 5, and the last section is devoted for concluding remarks and future works.

2 Geometrical Structure of AdaBoost

Lebanon and Lafferty (2001) firstly pointed out the duality of the AdaBoost algorithm in the space of distributions and discussed its geometrical structure from the view point of linear programming. In this section, we briefly review their result with the notion of the information geometry.

Let us consider a classification problem where for a given feature vector $\boldsymbol{x}$, the corresponding label y is predicted. Hereafter we assume that the feature vector $\boldsymbol{x}$ belongs to some space $\mathcal{X}$, and the corresponding label y to a finite set $\mathcal{Y}$. We note that for intuitive examples in this paper, we consider the case where y is the binary label with values -1 and 1, however, the problem can be extended to the multi-class case in straightforward way.

Let us consider the space of all the positive finite measures over $\mathcal{Y}$ conditioned by $\boldsymbol{x} \in \mathcal{X}$

$$\mathcal{M} = \left\{ m(y|\boldsymbol{x}) \;\middle|\; \sum_{y \in \mathcal{Y}} m(y|\boldsymbol{x}) < \infty \text{ (a.e. } \boldsymbol{x}) \right\}, \tag{1}$$

and the conditional probability density as its subspace

$$\mathcal{P} = \left\{ m(y|\boldsymbol{x}) \;\middle|\; \sum_{y \in \mathcal{Y}} m(y|\boldsymbol{x}) = 1 \text{ (a.e. } \boldsymbol{x}) \right\}. \tag{2}$$

For given examples $\{(\boldsymbol{x}_i, y_i); i = 1, \ldots, n\}$, let

$$\tilde{p}(y|\boldsymbol{x}) = \begin{cases} \delta(y_i, y), & \boldsymbol{x} = \boldsymbol{x}_i, \\ \dfrac{1}{|\mathcal{Y}|}, & \text{otherwise} \end{cases} \tag{3}$$

be the empirical conditional probability density of y for given $\boldsymbol{x}$. Here we assume the consistent data assumption (Lebanon and Lafferty, 2001) where a unique label y_i is given for each input $\boldsymbol{x}_i$. If multiple labels are given for an input $\boldsymbol{x}_i$, we can use

$$\tilde{p}(y|\boldsymbol{x}) = \begin{cases} \dfrac{\sum_{i=1}^n I(\boldsymbol{x}_i = \boldsymbol{x})\delta(y_i, y)}{\sum_{i=1}^n I(\boldsymbol{x}_i = \boldsymbol{x})}, & \sum_{i=1}^n I(\boldsymbol{x}_i = \boldsymbol{x}) \neq 0, \\ \dfrac{1}{|\mathcal{Y}|}, & \text{otherwise}, \end{cases}$$

where I is the indicator function defined by

$$I(A) = \begin{cases} 1, & A \text{ is true}, \\ 0, & \text{otherwise}, \end{cases}$$

and $|\mathcal{Y}|$ is the cardinality of $\mathcal{Y}$. The discussion in this paper can be extended straightforwardly.

For two points p, q in $\mathcal{M}$, the Kullback-Leibler divergence (KL divergence) extended over $\mathcal{M}$ is defined by

$$D(p, q) = \int_{\mathcal{X}} \mu(\boldsymbol{x}) \sum_{y \in \mathcal{Y}} \left(p(y|\boldsymbol{x}) \log \frac{p(y|\boldsymbol{x})}{q(y|\boldsymbol{x})} - p(y|\boldsymbol{x}) + q(y|\boldsymbol{x}) \right) d\boldsymbol{x}, \tag{4}$$

where $\mu(\boldsymbol{x})$ is the marginal distribution of $\boldsymbol{x}$, and in the most cases of the following discussion, we fix $\mu(\boldsymbol{x})$ with the empirical distribution

$$\mu(\boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^n \delta(\boldsymbol{x}_i, \boldsymbol{x}).$$

Note that for probability densities $p, q \in \mathcal{P}$, the above definition is reduced to the conventional KL divergence, because the second and third terms are summed up to 1 and vanish by negating each other.

Next, we consider two subspaces in $\mathcal{M}$ depending on a certain fixed measure $q_0 \in \mathcal{M}$, the empirical distribution $\tilde{p}$ and a set of functions $\boldsymbol{f} = \{f_t(\boldsymbol{x}, y); t = 1, \ldots, T\}$ on $\mathcal{X} \times \mathcal{Y}$.

e-flat subspace

$$\begin{aligned}\mathcal{Q} &= \mathcal{Q}(q_0, \tilde{p}, \boldsymbol{f}) \\ &= \left\{ q \in \mathcal{M} \;\middle|\; q(y|\boldsymbol{x}) = q_0(y|\boldsymbol{x}) \exp\Big(\sum_{t=1}^{T} \alpha_t (f_t(\boldsymbol{x}, y) - \tilde{f}_t(\boldsymbol{x}))\Big) \right\},\end{aligned}$$

where $\{\alpha_i \in R; i = 1, \ldots, T\}$ and $\tilde{f}(\boldsymbol{x})$ is the conditional expectation of function f with respect to the empirical distribution

$$\tilde{f}(\boldsymbol{x}) = E_{\tilde{p}}(f|\boldsymbol{x}) = \sum_{y \in \mathcal{Y}} \tilde{p}(y|\boldsymbol{x}) f(\boldsymbol{x}, y).$$

m-flat subspace

$$\begin{aligned}\mathcal{F} &= \mathcal{F}(\tilde{p}, \boldsymbol{f}) \\ &= \left\{ q \in \mathcal{M} \;\middle|\; \int_{\mathcal{X}} \mu(\boldsymbol{x}) \sum_{y \in \mathcal{Y}} q(y|\boldsymbol{x})(f_t(\boldsymbol{x}, y) - \tilde{f}_t(\boldsymbol{x})) d\boldsymbol{x} = 0; \forall t \right\}.\end{aligned}$$

Although the subspaces $\mathcal{Q}(q_0, \tilde{p}, \boldsymbol{f})$ and $\mathcal{F}(\tilde{p}, \boldsymbol{f})$ depend on $\tilde{p}$, q_0 and $\boldsymbol{f}$, but in the absence of ambiguity, we just denote $\mathcal{Q}$ and $\mathcal{F}$.

The subspace $\mathcal{Q}$ is a sort of exponential family of the conditional measures, which includes q_0 and is spanned by $\{f_t\}$ as sufficient statistic, therefore the dimension of the subspace $\mathcal{Q}$ is T in the space $\mathcal{M}$. The difference from the conventional statistical exponential family is that the normalization term is missing.

An important property of $\mathcal{Q}$ is its flatness. Let q_1 and q_2 be in $\mathcal{Q}$, then for any positive numbers β_1 and β_2,

$$q_0 \exp\left(\beta_1 \log \frac{q_1}{q_0} + \beta_2 \log \frac{q_2}{q_0} \right) \in \mathcal{Q},$$

holds, therefore its structure is called exponential flat (e-flat) in terms of the information geometry.

On the other hand, the meaning of $\mathcal{F}$ is slightly complicated. Intuitively speaking, $\mathcal{F}$ is a set of measures which preserve the moments of the features f_t. From the geometrical point of view, the condition is rewritten as

$$\int_{\mathcal{X}} \mu(\boldsymbol{x}) \sum_{y \in \mathcal{Y}} \Big(q(y|\boldsymbol{x}) - \tilde{p}(y|\boldsymbol{x}) \Big) \Big(f_j(\boldsymbol{x}, y) - \tilde{f}_j(\boldsymbol{x}) \Big) d\boldsymbol{x} = 0$$

and it means that among the orthogonal subsets to $\mathcal{Q}$, $\mathcal{F}$ is the set including the empirical distribution $\tilde{p}$. This geometrical interpretation is minutely discussed in the succeeding sections.

Also $\mathcal{F}$ has a flat structure. For any p_1 and p_2 in $\mathcal{F}$, and for any positive numbers β_1 and β_2, we observe

$$\beta_1 p_1 + \beta_2 p_2 \in \mathcal{F},$$

hence $\mathcal{F}$ is a convex cone in $\mathcal{M}$ and its structure is called mixture flat (m-flat). Note that the codimension of the subspace $\mathcal{F}$ is T in the space $\mathcal{M}$ by its definition.

Let us consider two optimization problem

$$\begin{aligned} &\text{minimize } D(p, q_0) \\ &\text{subject to } p \in \mathcal{F}(\tilde{p}, \boldsymbol{f}), \end{aligned} \tag{5}$$

and

$$\begin{aligned} &\text{minimize } D(\tilde{p}, q) \\ &\text{subject to } q \in \mathcal{Q}(q_0, \tilde{p}, \boldsymbol{f}). \end{aligned} \tag{6}$$

First of all, we should note that two subspaces $\mathcal{F}$ and $\mathcal{Q}$ intersect each other at one point q^*

$$\{q^*\} = \mathcal{F} \cap \mathcal{Q},$$

because of the relationship between $\dim \mathcal{Q}$ and $\operatorname{codim} \mathcal{F}$. From a fundamental property of the KL divergence and the definitions of $\mathcal{Q}$ and $\mathcal{F}$, we can prove the following lemma.

Lemma 1. *For any $p \in \mathcal{F}(\tilde{p}, \boldsymbol{f})$ and $q \in \mathcal{Q}(q_0, \tilde{p}, \boldsymbol{f})$, the Pythagorean relation*

$$D(p, q) = D(p, q^*) + D(q^*, q) \tag{7}$$

holds.

The proof is given later in more general form for the Bregman divergence. This lemma shows that from a fixed point $q_0 \in \mathcal{Q}$, q^* is the closest point in $\mathcal{F}$. Since

$$D(p, q_0) = D(p, q^*) + D(q^*, q_0), \text{ for any } p \in \mathcal{F}$$

holds, therefore we observe

$$D(p, q_0) \geq D(q^*, q_0)$$

and this means the point $q^* \in \mathcal{F}$ is the closest from q_0, and vice versa. As we discuss in the later, the one-dimensional e-flat subspace from q_0 to q_* is orthogonal to the m-flat subspace $\mathcal{F}$, hence q^* is said to be the e-projection of q_0 to $\mathcal{F}$. Simultaneously q^* is the m-projection of $\tilde{p}$ to $\mathcal{Q}$ (Amari and Nagaoka, 2000). As a natural consequence, we can conclude the following.

Theorem 1 (Lebanon and Lafferty (2001)). *Two optimization problems* (5) *and* (6) *give the same solution:*

$$q^* = \operatorname*{argmin}_{p \in \mathcal{F}(\tilde{p}, \boldsymbol{f})} D(p, q_0) = \operatorname*{argmin}_{q \in \mathcal{Q}(q_0, \tilde{p}, \boldsymbol{f})} D(\tilde{p}, q). \tag{8}$$

From the above theorem, the sequential update of AdaBoost can be naturally understood as follows. Let us define two subspaces for the sequential update.

e-flat subspace determined by q_{t-1}, $\tilde{p}$ and f_t:

$$\begin{aligned} \mathcal{Q}_t &= \mathcal{Q}(q_{t-1}, \tilde{p}, f_t) \\ &= \left\{ q \in \mathcal{M} \;\middle|\; q(y|\boldsymbol{x}) = q_{t-1}(y|\boldsymbol{x}) \exp\Big(\alpha_t \big(f_t(\boldsymbol{x}, y) - \tilde{f}_t(\boldsymbol{x})\big)\Big) \right\}. \end{aligned}$$

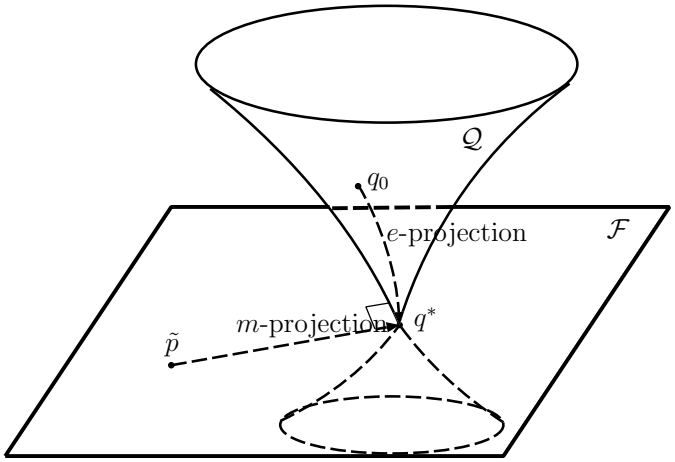


Figure 1: A geometrical interpretation of the dual optimization problems in the AdaBoost algorithm.

m-flat subspace determined by $\tilde{p}$ and f_t:

$$\begin{aligned}\mathcal{F}_t &= \mathcal{F}(\tilde{p}, f_t) \\ &= \left\{ p \in \mathcal{M} \;\middle|\; \int_{\mathcal{X}} \mu(\boldsymbol{x}) \sum_{y \in \mathcal{Y}} p(y|\boldsymbol{x})(f_t(\boldsymbol{x}, y) - \tilde{f}_t(\boldsymbol{x}))d\boldsymbol{x} = 0 \right\}.\end{aligned}$$

Note that because $\dim \mathcal{Q}_t = 1$ and $\operatorname{codim} \mathcal{F}_t = 1$, $\mathcal{Q}_t$ and $\mathcal{F}_t$ intersect at one point q_t.

Let us consider a learning machine $h(\boldsymbol{x})$ which predicts labels for an input $\boldsymbol{x}$. The machine can either output a unique label or output a set of labels. Obviously the latter case include the former as a special case, hence here we describe the algorithm with the latter. The AdaBoost algorithm is written as following way.

step 1: Initialize $q_0(y|\boldsymbol{x}) = 1$.

step 2: For $t = 1, \ldots, T$

- Select a machine h_t so that

$$\sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} q_{t-1}(y|\boldsymbol{x}_i)(f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i)) \neq 0,$$

  where

$$f_t(\boldsymbol{x}, y) = \begin{cases} \frac{1}{2}, & y \in h_t(\boldsymbol{x}), \\ -\frac{1}{2}, & \text{otherwise.} \end{cases}$$

- Construct $\mathcal{Q}_t$ and $\mathcal{F}_t$ with $\tilde{p}$, q_{t-1} and f_t.
- Find q_t and corresponding α_t which is the intersection of $\mathcal{Q}_t$ and $\mathcal{F}_t$.

step 3: Output the final decision as the majority vote of $\{h_t; t = 1, \ldots, T\}$ with weights $\{\alpha_t; t = 1, \ldots, T\}$

$$H(\boldsymbol{x}) = \operatorname*{argmax}_{y \in \mathcal{Y}} \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y) \left(= \operatorname*{argmax}_{y \in \mathcal{Y}} \sum_{t=1}^{T} \alpha_t I(y \in h_t(\boldsymbol{x})) \right).$$

Geometrical understanding is schematically shown in Fig. 2. The best choice of the machine h_t in step 2 is realized by

$$h_t(\boldsymbol{x}) = \operatorname*{argmax}_{h} \sum_{i=1}^{n} \Big[I(y_i \in h(\boldsymbol{x}_i)) \sum_{y \notin h(\boldsymbol{x}_i)} q_{t-1}(y|\boldsymbol{x}_i) - I(y_i \notin h(\boldsymbol{x}_i)) \sum_{y \in h(\boldsymbol{x}_i)} q_{t-1}(y|\boldsymbol{x}_i) \Big],$$

however we do not necessarily use this optimal h_t. Since the relation

$$D(\tilde{p}, q_{t-1}) = D(\tilde{p}, q_t) + D(q_t, q_{t-1}) \tag{9}$$

holds, as the step t increases q_t approaches to the empirical distribution $\tilde{p}$ as long as $D(q_t, q_{t-1}) > 0$. The selection policy of h_t in step 2 is to guarantee $\mathcal{Q}_t$ and $\mathcal{Q}_{t-1}$ to differ and $D(q_t, q_{t-1})$ to be positive. When $\mathcal{Q}_t$ coincides with $\mathcal{Q}_{t-1}$, the algorithm stops.

For the binary case, α_t is given by

$$\alpha_t = \frac{1}{2} \log \frac{1 - \epsilon_t}{\epsilon_t},$$

where ϵ_t is the weighted error defined by

$$\epsilon_t = \sum_{i=1}^{n} I(y_i \neq h_t(\boldsymbol{x}_i)) D_t(i),$$
$$D_t(i) = \frac{q_{t-1}(y \neq y_i|\boldsymbol{x}_i)}{Z_t},$$

and Z_t is a normalization constant to ensure $\sum_{i=1}^{n} D_t(i) = 1$. In section 4, we discuss the meaning of α_t in detail for the U-Boost algorithm.

3 Bregman Divergence and U-functions

AdaBoost can be regarded as a procedure of optimizing an exponential loss with an additive model (Friedman et al., 2000)

$$L(F) = \frac{1}{n} \sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} \exp(F(\boldsymbol{x}_i, y) - F(\boldsymbol{x}_i, y_i))$$
$$\text{where } F(\boldsymbol{x}, y) = \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y).$$

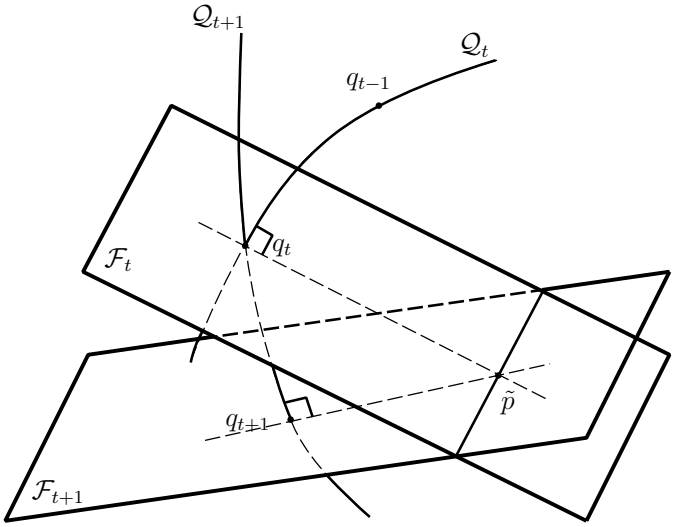


Figure 2: A geometrical interpretation of the sequential update of AdaBoost. Since $D(\tilde{p}, q_t) \geq D(\tilde{p}, q_{t+1})$ holds, q_t approaches to the empirical distribution $\tilde{p}$.

By adopting different loss functions, several variations of AdaBoost are proposed, such as MadaBoost (Domingo and Watanabe, 2000), where the loss function

$$L(F) = \frac{1}{n} \sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} \phi(F(\boldsymbol{x}_i, y) - F(\boldsymbol{x}_i, y_i))$$

$$\text{where } \phi(z) = \begin{cases} z + \frac{1}{2} & z \geq 0, \\ \frac{1}{2} \exp(2z) & \text{otherwise}, \end{cases}$$

is used instead of the exponential loss.

For constructing algorithms, the notion of the loss function is useful, because the various algorithms are derived based on the gradient descent and line search methods. Also the loss function controls the confidence of the decision, which is characterized by the margin (Schapire et al., 1998). However, the statistical properties such as consistency and efficiency are not apparent, because the relationship between loss functions and the distributions realized by combined machines is unclear so far.

In this section, we consider a form of the Bregman divergence which is suited for statistical inferences, and consider some of its properties.

3.1 Statistical Form of Bregman Divergence

The Bregman divergence is a pseudo-distance for measuring the discrepancy between two functions. We define the Bregman divergence between two conditional measures as follows.

Definition 1 (Bregman divergence). Let U be a strictly convex function on R, then its derivative $u = U'$ is a monotone function, which has the inverse

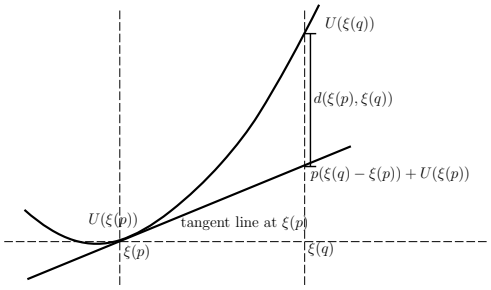
function $\xi = (u)^{-1}$. For $p(y|\boldsymbol{x})$ and $q(y|\boldsymbol{x})$ in $\mathcal{M}$, the Bregman divergence from p to q is defined by

$$D_U(p,q) = \int_{\mathcal{X}} \sum_{y\in\mathcal{Y}} \Big[\big\{ U(\xi(q(y|\boldsymbol{x}))) - U(\xi(p(y|\boldsymbol{x}))) \big\} - p(y|\boldsymbol{x}) \big\{ \xi(q(y|\boldsymbol{x})) - \xi(p(y|\boldsymbol{x})) \big\} \Big] \mu(\boldsymbol{x}) d\boldsymbol{x}. \quad (10)$$

In the following, if the context is clear, we omit $\boldsymbol{x}$ and y from functions for notational simplicity, such as

$$D_U(p,q) = \int \sum \Big[\big\{ U(\xi(q)) - U(\xi(p)) \big\} - p \big\{ \xi(q) - \xi(p) \big\} \Big] d\mu.$$

As easily seen, the Bregman divergence is not symmetric with respect to p and q in general, therefore it is not a distance.



$U : R \to R$, convex
$\xi : u^{-1} = (U')^{-1}$

Note: $u(\xi(p)) = p$

Figure 3: Bregman divergence.

A popular form of the Bregman divergence is

$$D_U(f,g) = \int d(f(z), g(z)) d\nu(z)$$

where f, g are one dimensional real-valued functions of z, and $\nu(z)$ is a certain measure on z, and d is the difference at g between U and tangent line at $(f, U(f))$

$$d(f,g) = U(g) - \{u(f)(g-f) + U(f)\}. \quad (11)$$

In the definition (10), densities are mapped by ξ first, then the form (11) is applied, and the meaning of d is easily understood from Fig. 3.

It is also closely related with the potential duality. Let us define the dual function of U by Legendre transformation

$$U^*(\eta) = \sup_{\theta} \{\eta\theta - U(\theta)\},$$

then d is written with U and U^* simply

$$d(f,g) = U^*(\eta_f) + U(g) - \eta_f g,$$

where

$$\eta_f = u(f).$$

The advantage of the form (10) is allowing us to plug in the empirical distribution directly. To see this, let us decompose the Bregman divergence into

$$D_U(p,q) = L_U(p,q) - L_U(p,p), \tag{12}$$

where

$$L_U(p,q) = \int \sum \{U(\xi(q)) - p\xi(q)\}\, d\mu. \tag{13}$$

Note that L_U can be regarded as a loss function, and since the Bregman divergence is non-negative, that is $D_U(p,q) \geq 0$, the loss is bounded below by

$$L_U(p,q) \geq L_U(p,p).$$

Now we consider a problem in which q is optimized with respect to $D_U(p,q)$ for fixed p. Picking out the terms which depend on q, the problem is simplified as

$$\operatorname*{argmin}_q D_U(p,q) = \operatorname*{argmin}_q L_U(p,q). \tag{14}$$

In $L_U(p,q)$, the distribution p appears only for taking the expectation of $\xi(q)$, therefore the empirical distribution is used without any difficulty as

$$L_U(\tilde{p},q) = \frac{1}{n}\sum_{i=1}^{n}\Big\{\sum_{y\in\mathcal{Y}} U(\xi(q(y|\boldsymbol{x}_i))) - \xi(q(y_i|\boldsymbol{x}_i))\Big\}, \tag{15}$$

which we refer as the empirical U-loss, and the optimal distribution for given examples is defined by

$$\tilde{q} = \operatorname*{argmin}_q L_U(\tilde{p},q).$$

This is equivalent with the well-known relationship between the maximum likelihood estimation and the minimization of the Kullback-Leibler divergence. Related discussions can be found in Eguchi and Kano (2001), in which the divergences are derived based on the pseudo-likelihood.

The followings are examples of the convex function U.

Example 1 (U-functions).

Kullback-Leibler:

$$U(z) = \exp(z), \qquad u(z) = \exp(z), \qquad \xi(z) = \log(z).$$

β-type:

$$U(z) = \frac{1}{\beta+1}(\beta z+1)^{\frac{\beta+1}{\beta}}, \qquad u(z) = (\beta z+1)^{\frac{1}{\beta}}, \qquad \xi(z) = \frac{z^\beta - 1}{\beta}.$$

β-type ($\beta = 1$):

$$U(z) = \frac{1}{2}(z+1)^2, \qquad u(z) = z+1, \qquad \xi(z) = z-1.$$

η-type:

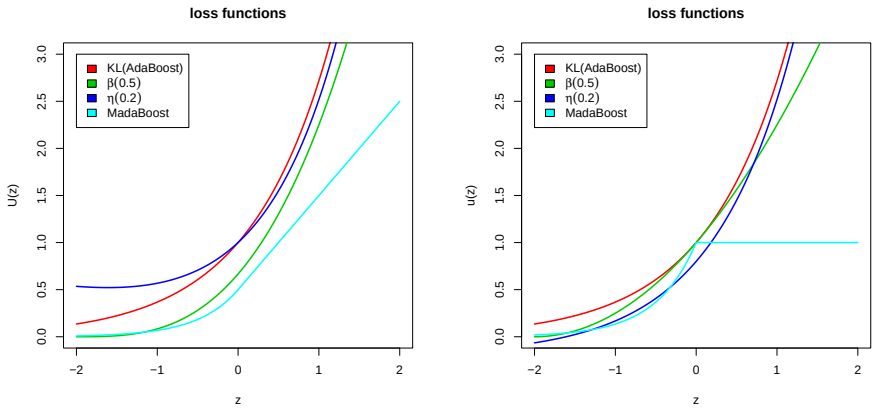
$$U(z) = \exp(z) - \eta z, \qquad u(z) = \exp(z) - \eta, \qquad \xi(z) = \log(z + \eta).$$

MadaBoost:

$$U(z) = \begin{cases} z + \frac{1}{2} & z \geq 0, \\ \frac{1}{2}\exp(2z) & z < 0, \end{cases} \quad u(z) = \begin{cases} 1 & z \geq 0, \\ \exp(2z) & z < 0, \end{cases} \quad \xi(z) = \frac{1}{2}\log(z)(z \leq 1).$$

Note that the MadaBoost U function is not strictly convex, hence $\xi(z)$ is not well defined for $z > 1$. Although it is peculiar as a loss function, it performs an important role to consider the robustness.

The divergence of β-type has been employed into for the independent component analysis from the viewpoint of robustness, while the divergence of η-type is shown to improve AdaBoost for the case with mislabelling (see Minami and Eguchi, 2002; Takenouchi and Eguchi, 2002).



(a) (b)

Figure 4: Examples of U-functions. (a) Shapes of U-functions. (b) Derivatives of U-functions.

3.2 Pythagorean Relation and Orthogonal Foliation

Let us define the inner product of functions of $\boldsymbol{x} \in \mathcal{X}$ and $y \in \mathcal{Y}$ by

$$\langle f, g \rangle = \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} f(\boldsymbol{x}, y) g(\boldsymbol{x}, y) d\mu(\boldsymbol{x})$$

and define that f and g are orthogonal if $\langle f, g \rangle = 0$. Then the Pythagorean relation for the Bregman divergence is stated as follows.

Lemma 2 (Pythagorean relation). *Let p, q and r be in $\mathcal{M}$. If $p - q$ and $\xi(r) - \xi(q)$ are orthogonal, the relation*

$$D_U(p, r) = D_U(p, q) + D_U(q, r) \tag{16}$$

holds.

Proof. For any conditional measures p, q and r,

$$\begin{aligned} & D_U(p,r) - D_U(p,q) - D_U(q,r) \\ &= \int_{\mathcal{X}} \sum_{y\in\mathcal{Y}} \left(q(y|\boldsymbol{x}) - p(y|\boldsymbol{x})\right) \left(\xi(r(y|\boldsymbol{x})) - \xi(q(y|\boldsymbol{x}))\right) d\mu(\boldsymbol{x}) \\ &= -\langle p - q, \xi(r) - \xi(q)\rangle \end{aligned} \tag{17}$$

holds by definition. From the orthogonality of $p-q$ and $\xi(r)-\xi(q)$, the right-hand side of (17) vanishes, and it proves the relation. □

Lemma 1 in the previous section is a special case of Lemma 2 associated with the Kullback-Leibler divergence. Note that in the above lemma, the orthogonality is defined between $p-q$ and $\xi(r)-\xi(q)$. The form $\xi(q)$ is rewritten as

$$q = u(\xi(q))$$

and $\xi(q)$ is called U-representation of q. In the following discussion, U-representation plays a key part.

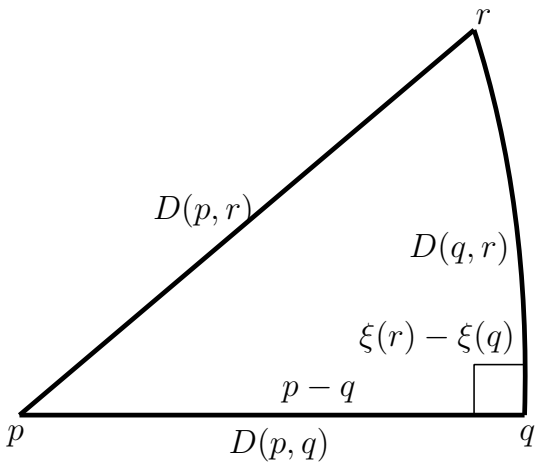


Figure 5: Pythagorean relation for Bregman divergence.

Now we consider subspaces feasible for the nature of the Bregman divergence. First, we start from the simplest case. Let us consider a set of conditional measures with a fixed $q_0 \in \mathcal{M}$ and a set of functions $\boldsymbol{f} = \{f_t(\boldsymbol{x}, y); t = 1, \ldots, T\}$, written in the form of

$$\begin{aligned} \mathcal{Q}_U &= \mathcal{Q}_U(q_0, \boldsymbol{f}) \\ &= \left\{ q \in \mathcal{M} \;\middle|\; q = u\big(\xi(q_0) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y)\big) \right\}, \end{aligned} \tag{18}$$

where $\boldsymbol{\alpha} = \{\alpha_t \in R; t = 1, \ldots, T\}$. In other words, $\mathcal{Q}_U$ consists of functions such that

$$\xi(q) - \xi(q_0) = \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y),$$

which means $\mathcal{Q}_U$ is a subspace including q_0 and spanned by $\boldsymbol{f}$. In this relation ξ plays the same role with logarithm in the e-flat subspace, and $\mathcal{Q}_U$ is called a U-flat subspace.

Next let us consider an m-flat subspace in $\mathcal{M}$ which passes a point $q \in \mathcal{Q}_U$ by

$$\begin{aligned}\mathcal{F}_U(q) &= \mathcal{F}_U(q, \boldsymbol{f}) \\ &= \left\{ p \in \mathcal{M} \;\middle|\; \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} (p(y|\boldsymbol{x}) - q(y|\boldsymbol{x})) f_t(\boldsymbol{x}, y) d\mu(\boldsymbol{x}) = 0,\ \forall t \right\} \\ &= \left\{ p \in \mathcal{M} \;\middle|\; \langle p - q, f_t \rangle = 0,\ \forall t \right\}. \end{aligned} \tag{19}$$

By these definitions, $\mathcal{Q}_U$ and $\mathcal{F}_U(q)$ are orthogonal at q, that is,

$$\langle p - q, \xi(r) - \xi(q) \rangle = 0,\ \forall p \in \mathcal{F}_U(q),\ \forall r \in \mathcal{Q}_U.$$

A set $\{\mathcal{F}_U(q); q \in \mathcal{Q}_U\}$ is called a foliation of $\mathcal{M}$, which covers the whole space $\mathcal{M}$ as

$$\begin{aligned}&\bigcup_{q \in \mathcal{Q}} \mathcal{F}_U(q) = \mathcal{M}, \\ &\mathcal{F}_U(q) \cap \mathcal{F}_U(q') = \phi, \text{ if } q \neq q'.\end{aligned}$$

To put it in other words, $\mathcal{M}$ is decomposed into an orthogonal foliation by giving $\mathcal{Q}_U$.

Second, we consider a general version. Let $b(\boldsymbol{x}, \boldsymbol{\alpha})$ be a function of $\boldsymbol{x}$ and $\boldsymbol{\alpha}$. A U-flat subspace, which we refer the U-model in the following, is defined by

$$\begin{aligned}\mathcal{Q}_U &= \mathcal{Q}_U(q_0, \boldsymbol{f}, b) \\ &= \left\{ q \in \mathcal{M} \;\middle|\; q_{\boldsymbol{\alpha}} = u\big(\xi(q_0) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y) - b(\boldsymbol{x}, \boldsymbol{\alpha})\big) \right\}, \end{aligned} \tag{20}$$

and an m-flat subspace which passes a point $q = q_{\boldsymbol{\alpha}} \in \mathcal{Q}_U$ by

$$\begin{aligned}\mathcal{F}_U(q) &= \mathcal{F}_U(q, \boldsymbol{f}, b) \\ &= \left\{ p \in \mathcal{M} \;\middle|\; \langle p - q, f_t - b'_t(\boldsymbol{\alpha}) \rangle = 0,\ \forall t \right\}, \end{aligned} \tag{21}$$

where

$$b'_t(\boldsymbol{x}, \boldsymbol{\alpha}) = \frac{\partial b(\boldsymbol{x}, \boldsymbol{\alpha})}{\partial \alpha_t}.$$

In this case, the orthogonality of $\mathcal{Q}_U$ and $\mathcal{F}_U(q)$ is defined with the tangent of $\mathcal{Q}_U$ at q,

$$\left\langle p - q, \frac{\partial}{\partial \alpha_t} \xi(q) \right\rangle = \langle p - q, f_t - b'_t(\boldsymbol{\alpha}) \rangle = 0,\ \forall p \in \mathcal{F}_U(q),\ \forall t.$$

The function b must be determined by the constraint on the U-model such as from the statistical viewpoint or computational convenience. From a statistical

point of view, we consider following two specific cases, normalized models and unnormalized models.

Let $p(y|\boldsymbol{x})$ be a true distribution of y given $\boldsymbol{x}$, and for denoting a distribution let us use the U-representation as

$$q_F(y|\boldsymbol{x}) = u(F(\boldsymbol{x}, y)), \text{ i.e. } F(\boldsymbol{x}, y) = \xi(q_F(y|\boldsymbol{x}))$$

For the classification task, we adopt the rule that the corresponding label for a given input $\boldsymbol{x}$ is estimated by the maximum value of $q_F(y|\boldsymbol{x})$ which is realized by

$$\hat{y} = \operatorname*{argmax}_{y\in\mathcal{Y}} F(\boldsymbol{x}, y) = \operatorname*{argmax}_{y\in\mathcal{Y}} \xi(q_F(y|\boldsymbol{x})), \tag{22}$$

because ξ is monotonic. Hereafter we focus on

$$\begin{aligned}\Delta(F, F^*) &= D_U(p, q_F) - D_U(p, q_{F^*}) - D_U(q_{F^*}, q_F) \\ &= \int_{\mathcal{X}} \sum_{y\in\mathcal{Y}} \big(F(\boldsymbol{x}, y) - F^*(\boldsymbol{x}, y)\big)\big(q_{F^*}(y|\boldsymbol{x}) - p(y|\boldsymbol{x})\big) d\mu(\boldsymbol{x}) \\ &= \big\langle F - F^*, q_{F^*} - p\big\rangle, \end{aligned} \tag{23}$$

and consider the conditions where $\Delta(F, F^*)$ vanishes to utilize the Pythagorean relation.

3.2.1 Normalized U-model

First let us consider a set

$$\mathcal{F} = \{F | F(\boldsymbol{x}, y) = F_0(\boldsymbol{x}, y) - b(\boldsymbol{x})\},$$

where $F_0(\boldsymbol{x}, y)$ is fixed and $b(\boldsymbol{x})$'s are arbitrary functions of $\boldsymbol{x}$. We note that the classification rule associated with any $F(\boldsymbol{x}, y) \in \mathcal{F}$ is equivalent to that with $F_0(\boldsymbol{x}, y)$ because

$$\operatorname*{argmax}_{y\in\mathcal{Y}} F(\boldsymbol{x}, y) = \operatorname*{argmax}_{y\in\mathcal{Y}} F_0(\boldsymbol{x}, y) - b(\boldsymbol{x}) = \operatorname*{argmax}_{y\in\mathcal{Y}} F_0(\boldsymbol{x}, y),$$

and for any $F = F_0 - b$ and $F^* = F_0 - b^*$,

$$\Delta(F, F^*) = \int_{\mathcal{X}} \big(b(\boldsymbol{x}) - b^*(\boldsymbol{x})\big) \sum_{y\in\mathcal{Y}} \big(q_{F^*}(y|\boldsymbol{x}) - p(y|\boldsymbol{x})\big) d\mu(\boldsymbol{x})$$

holds. Suppose $\sum_{y\in\mathcal{Y}} q_{F^*}(y|\boldsymbol{x}) = 1$ (a.e. $\boldsymbol{x}$), then

$$D_U(p, q_F) = D_U(p, q_{F^*}) + D_U(q_{F^*}, q_F)$$

holds because $\Delta = 0$, and this means that F^* is the closest from p among the functions in $\mathcal{F}$ which give the same classification rule, that is to say,

$$F^* = \operatorname*{argmin}_{F\in\mathcal{F}} D_U(p, q_F).$$

Therefore the minimization $D_U(p, q_F)$ in $\mathcal{F}$ is equivalent to introducing the normalizing factor $b^*(\boldsymbol{x})$ so that

$$\sum_{y\in\mathcal{Y}} q_{F^*}(y|\boldsymbol{x}) = \sum_{y\in\mathcal{Y}} u(F_0(\boldsymbol{x}, y) - b^*(\boldsymbol{x})) = 1, \tag{24}$$

namely q_{F^*} is restricted on a conditional probability density.

For the U-model $\mathcal{Q}_U$, let F^* be written as

$$F^*(\boldsymbol{x}, y) = \xi(q_0(y|\boldsymbol{x})) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y) - b^*(\boldsymbol{x}, \boldsymbol{\alpha}),$$

then the empirical loss for the normalized U-model is led to

$$L_U(\tilde{p}, q) = \frac{1}{n} \sum_{i=1}^{n} \Big[\sum_{y \in \mathcal{Y}} U\big(\xi(q_0(y|\boldsymbol{x})) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}_i, y) - b^*(\boldsymbol{x}_i, \boldsymbol{\alpha})\big) - \xi(q_0(y|\boldsymbol{x})) - \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}_i, y_i) + b^*(\boldsymbol{x}_i, \boldsymbol{\alpha}) \Big]. \quad (25)$$

3.2.2 Unnormalized U-model

Secondly, we consider the case that $q_{F^*}(y|\boldsymbol{x}) = c(\boldsymbol{x})p(y|\boldsymbol{x})$ or equivalently $F^*(\boldsymbol{x}, y) = \xi(c(\boldsymbol{x})p(y|\boldsymbol{x}))$, which implies the rule associated with $F^*(\boldsymbol{x}, y)$ is equivalent to the Bayes rule for p

$$\hat{y} = \operatorname*{argmax}_{y \in \mathcal{Y}} F^*(\boldsymbol{x}, y) = \operatorname*{argmax}_{y \in \mathcal{Y}} \xi(c(\boldsymbol{x})p(y|\boldsymbol{x})) = \operatorname*{argmax}_{y \in \mathcal{Y}} p(y|\boldsymbol{x}).$$

In this case,

$$\Delta(F, F^*) = \int_{\mathcal{X}} \big(c(\boldsymbol{x}) - 1\big) \sum_{y \in \mathcal{Y}} \big(F(\boldsymbol{x}, y) - F^*(\boldsymbol{x}, y)\big) p(y|\boldsymbol{x}) d\mu(\boldsymbol{x})$$

holds for any F. Let us define γ by

$$\gamma(\boldsymbol{x}) = \sum_{y \in \mathcal{Y}} F^*(\boldsymbol{x}, y) p(y|\boldsymbol{x}),$$

and let $\mathcal{F}$ be

$$\mathcal{F} = \left\{ F \;\middle|\; \sum_{y \in \mathcal{Y}} F(\boldsymbol{x}, y) p(y|\boldsymbol{x}) = \gamma(\boldsymbol{x}) \text{ (a.e. } \boldsymbol{x}) \right\}.$$

Then $\Delta(F, F^*) = 0$ for any $F \in \mathcal{F}$ and

$$D_U(p, q_F) = D_U(p, q_{F^*}) + D_U(q_{F^*}, q_F)$$

holds, therefore F^* gives the minimum of $D_U(p, q_F)$ among $F \in \mathcal{F}$, that is,

$$F^* = \operatorname*{argmin}_{F \in \mathcal{F}} D_U(p, q_F),$$

in other word, D_U chooses the Bayes optimal rule in $\mathcal{F}$. With the same discussion of the normalized model, $F - b$ gives the same classification rule for any $b(\boldsymbol{x})$, therefore by change F into $F - \gamma$, we can introduce a simple constraint for $\mathcal{F}$ as

$$\sum_{y \in \mathcal{Y}} F(\boldsymbol{x}, y) p(y|\boldsymbol{x}) = \sum_{y \in \mathcal{Y}} \xi(q_F(y|\boldsymbol{x})) p(y|\boldsymbol{x}) = 0.$$

Under this constraint, the U-loss is reduced to

$$L_U(p,q) = \int_{\mathcal{X}} \sum_{y\in\mathcal{Y}} U(\xi(q))d\mu,$$

and the empirical loss for the U-model $\mathcal{Q}_U$ is simply written as

$$L_U(\tilde{p},q) = \frac{1}{n}\sum_{i=1}^{n}\sum_{y\in\mathcal{Y}} U\big(\xi(q_0(y|\boldsymbol{x}_i)) + \sum_{t=1}^{T}\alpha_t\big(f_t(\boldsymbol{x}_i,y) - f_t(\boldsymbol{x}_i,y_i)\big)\big). \quad (26)$$

4 U-Boost

Using the Bregman divergence instead of the Kullback-Leibler divergence, a class of loss functions are introduced. In this section, we consider boosting algorithms which are naturally derived from these loss functions and discuss some properties from the statistical point of view.

4.1 Algorithm

In the following, we treat the sequential updates of the algorithms mainly. The parallel updates are derived by replacing the subspaces $\mathcal{Q}$ and $\mathcal{F}$ in the same way as discussed in section 2.

A generic form of the U-Boost algorithm is given as follows.

generic U-Boost

step 1: Initialize $q_0(y|\boldsymbol{x})$. (In usual case, set $\xi(q_0) = 0$ for simplicity.)

step 2: For $t = 1,\ldots,T$

- Select a machine h_t so that

  $$\langle \tilde{p} - q_{t-1}, f_t - b_t'(\alpha = 0)\rangle \neq 0,$$

  where

  $$f_t(\boldsymbol{x},y) = \begin{cases} \frac{1}{2}, & y \in h_t(\boldsymbol{x}), \\ -\frac{1}{2}, & \text{otherwise}, \end{cases}$$

  and $b_t(\boldsymbol{x},\alpha)$ is an auxiliary function to satisfy an imposed constraint.
- Construct $\mathcal{Q}_t$,

  $$\mathcal{Q}_t = \left\{ q\in\mathcal{M} \;\middle|\; q = u\big(\xi(q_{t-1}) + \alpha f_t(\boldsymbol{x},y) - b_t(\boldsymbol{x},\alpha)\big)\right\}$$

- Find q_t and corresponding α_t which minimize $D_U(\tilde{p},q)$,

  $$\begin{aligned} q_t &= \operatorname*{argmin}_{q\in\mathcal{Q}_t} D_U(\tilde{p},q) \\ &= \operatorname*{argmin}_{q\in\mathcal{Q}_t} \sum_{i=1}^{n}\left\{\sum_{y\in\mathcal{Y}} U(\xi(q(y|\boldsymbol{x}_i))) - \xi(q(y_i|\boldsymbol{x}_i))\right\}. \end{aligned}$$

step 3: Output the final decision as the majority vote,

$$H(\boldsymbol{x}) = \underset{y\in\mathcal{Y}}{\operatorname{argmax}} \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y).$$

The optimization procedure in step 2 is geometrically interpreted as shown in Fig. 6. For a U-model $\mathcal{Q}_t$, we can consider an orthogonal foliation $\mathcal{F}_t(q)$ as

$$\mathcal{F}_t(q) = \left\{ p \in \mathcal{M} \;\middle|\; \langle p - q, f_t - b_t'(\alpha) \rangle = 0 \right\},\ q = q_\alpha \in \mathcal{Q}_t. \tag{27}$$

Then we can find a leaf $\mathcal{F}_t(q_*)$ which passes the empirical distribution $\tilde{p}$, and the optimal model is determined by $q_t = q_*$.

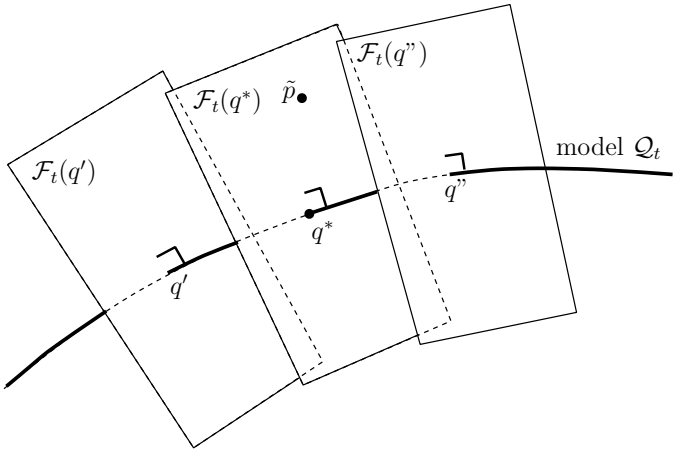


Figure 6: A geometrical interpretation of the U-Boost algorithm.

In general, $b_t(\boldsymbol{x}, \alpha)$ is chosen according to the discussion about constraints of U-models as follows.

4.1.1 Normalized U-Boost

The first constraint for $\mathcal{Q}_t$ is restricting the model on the conditional probability densities,

$$\sum_{y\in\mathcal{Y}} q(y|\boldsymbol{x}) = \sum_{y\in\mathcal{Y}} u\big(\xi(q_{t-1}) + \alpha f_t(\boldsymbol{x}, y) - b_t(\boldsymbol{x}, \alpha)\big) = 1. \tag{28}$$

As previously discussed, this constraint gives the solution which is the closest to the true distribution in the U-model giving the same classification rule.

The optimization procedure is defined by

$$q_t = \operatorname*{argmin}_{q \in \mathcal{Q}_t} L_U(\tilde{p}, q),$$

$$\alpha_t = \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \Big\{ \sum_{y \in \mathcal{Y}} U(\xi(q_{t-1}(y|\boldsymbol{x}_i)) + \alpha f_t(\boldsymbol{x}_i, y) - b_t(\boldsymbol{x}_i, \alpha))) \tag{29}$$

$$- \xi(q_{t-1}(y_i|\boldsymbol{x}_i)) - \alpha f_t(\boldsymbol{x}_i, y_i) + b_t(\boldsymbol{x}_i, \alpha)) \Big\}. \tag{30}$$

For $U(z) = \exp(z)$, which introduces the KL divergence, the constraint is

$$\sum_{y \in \mathcal{Y}} q(y|\boldsymbol{x}) = \sum_{y \in \mathcal{Y}} \exp\big(\log(q_{t-1}(y|\boldsymbol{x})) + \alpha f_t(\boldsymbol{x}, y) - b_t(\boldsymbol{x}, \alpha)\big) = 1,$$

therefore, the normalization term b_t is written as

$$b_t(\boldsymbol{x}, \alpha) = \log\Big(\sum_{y \in \mathcal{Y}} q_{t-1}(y|\boldsymbol{x}) \exp(\alpha f_t(\boldsymbol{x}, y)) \Big).$$

In this case, α_t is given by

$$\begin{aligned}
\alpha_t &= \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \log \frac{\sum_{y \in \mathcal{Y}} q_{t-1}(y|\boldsymbol{x}_i) \exp(\alpha f_t(\boldsymbol{x}_i, y))}{q_{t-1}(y_i|\boldsymbol{x}_i) \exp(\alpha f_t(\boldsymbol{x}_i, y_i))} \\
&= \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \log \frac{\sum_{y \in \mathcal{Y}} \exp(F_{t-1}(\boldsymbol{x}_i, y) + \alpha f_t(\boldsymbol{x}_i, y))}{\exp(F_{t-1}(\boldsymbol{x}_i, y_i) + \alpha f_t(\boldsymbol{x}_i, y_i))},
\end{aligned}$$

where F_{t-1} is the U-representation of q_{t-1}

$$F_{t-1}(\boldsymbol{x}, y) = \xi(q_0) + \sum_{k=1}^{t-1} \alpha_k f_k(\boldsymbol{x}, y).$$

Especially, for the binary case where a machine $h(\boldsymbol{x})$ outputs either $+1$ or -1, f is written as

$$f(\boldsymbol{x}, y) = \frac{1}{2} y h(\boldsymbol{x}), \tag{31}$$

and $\xi(q_0) = 0$ is employed, then the above equation is drastically simplified

$$\begin{aligned}
\alpha_t &= \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \log \frac{\sum_{y \in \{\pm 1\}} \exp(F_{t-1}(\boldsymbol{x}_i, y) + \alpha f_t(\boldsymbol{x}_i, y))}{\exp(F_{t-1}(\boldsymbol{x}_i, y_i) + \alpha f_t(\boldsymbol{x}_i, y_i))} \\
&= \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \log \frac{\exp(F_{t-1}(\boldsymbol{x}_i, y_i) + \alpha f_t(\boldsymbol{x}_i, y_i)) + \exp(-F_{t-1}(\boldsymbol{x}_i, y_i) - \alpha f_t(\boldsymbol{x}_i, y_i))}{\exp(F_{t-1}(\boldsymbol{x}_i, y_i) + \alpha f_t(\boldsymbol{x}_i, y_i))} \\
&= \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \log\Big(1 + \exp\big(-2(F_{t-1}(\boldsymbol{x}_i, y_i) + \alpha f_t(\boldsymbol{x}_i, y_i))\big) \Big),
\end{aligned} \tag{32}$$

where we use the fact $f(\boldsymbol{x}, +1) + f(\boldsymbol{x}, -1) = (1-1)h(\boldsymbol{x}) = 0$. This representation is equivalent to the unnormalized U-Boost which will be discussed in the next

section, and $U(z) = \log(1 + \exp(2z))$ is used as the U-loss function. Moreover, the above equation is written as

$$\alpha_t = \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \log\Big(1 + \exp\big(-y_i(\textstyle\sum_{k=1}^{t-1} \alpha_k h_k(\boldsymbol{x}) + \alpha h_t(\boldsymbol{x}_i))\big)\Big),$$

and this is equivalent to LogitBoost (Friedman et al., 2000). Namely, the normalized U-Boost associated with $U(z) = \exp(z)$ conducts the same procedure of LogitBoost.

4.1.2 Unnormalized U-Boost

Next, let us consider the constraint for $\mathcal{Q}_t$

$$\sum_{y \in \mathcal{Y}} \tilde{p}(y|\boldsymbol{x}) \xi(q(y|\boldsymbol{x})) = 0. \tag{33}$$

As discussed in the previous section, it is guaranteed that the minimizer of the U-loss becomes Bayes optimal in the constrained subspace of $\mathcal{M}$.

The optimization procedure is reduced to

$$\begin{aligned} q_t &= \operatorname*{argmin}_{q \in \mathcal{Q}_t} L_U(\tilde{p}, q), \\ \alpha_t &= \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} U\big(\xi(q_{t-1}(y|\boldsymbol{x}_i)) + \alpha(f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i))\big). \end{aligned} \tag{34}$$

As in the previous section, the empirical U-loss is simplified for the binary classification case as

$$L_U(\tilde{p}, q) = \sum_{i=1}^{n} U\big(-y_i(\textstyle\sum_{k=1}^{t-1} \alpha_k h_k(\boldsymbol{x}_i) + \alpha h_t(\boldsymbol{x}_i))\big), \tag{35}$$

where $\xi(q_0) = 0$ is adopted, and in this case the optimal α_t is given by

$$\alpha_t = \operatorname*{argmin}_{\alpha} L_U(\tilde{p}, q).$$

In the case of the KL divergence, where $U(z) = \exp(z)$, this procedure is equivalent to AdaBoost (cf. Lebanon and Lafferty, 2001).

We observe that for the sequence of densities $\{q_t; t = 0, 1, \dots\}$ defined by the normalized or unnormalized U-Boost algorithm, the relation

$$L_U(\tilde{p}, q_{t+1}) - L_U(\tilde{p}, q_t) = D_U(q_t, q_{t+1}) \tag{36}$$

holds. This property is closely related with that in the EM algorithm for obtaining the maximum likelihood estimator. See Amari (1995) for the geometric considerations of the EM algorithm.

4.2 Error Rate Property

One of the important characteristics of the AdaBoost algorithm is the evolution of its weighted error rates, that is, the machine h_t at step t shows the worst

performance, that is equivalent to random guess, under the distribution at the next step $t+1$. In the case of the binary classification problem, by defining the weighted error as

$$\epsilon_t(h) = \sum_{i=1}^{n} I(h(\boldsymbol{x}_i) \neq y_i) D_t(i),$$

where $D_t(i)$ is updated by

$$D_{t+1}(i) = \frac{D_t(i)\exp(-\alpha_t y_i h_t(\boldsymbol{x}_i))}{Z_{t+1}} \text{ with } D_0(i) = \frac{1}{n},$$

Z_{t+1} is a normalization constant to ensure $\sum_{i=1}^{n} D_{t+1}(i) = 1$, and α_t is given by

$$\alpha_t = \frac{1}{2}\ln\left(\frac{1-\epsilon_t(h_t)}{\epsilon_t(h_t)}\right),$$

this can be easily confirmed by

$$\begin{aligned}
&\epsilon_{t+1}(h_t) \\
&= \sum_{i=1}^{n} I(h_t(\boldsymbol{x}_i) \neq y_i) D_{t+1}(i) \\
&= \frac{\sum_{i=1}^{n} I(h_t(\boldsymbol{x}_i) \neq y_i) D_t(i) e^{\alpha_t}}{\sum_{i=1}^{n} I(h_t(\boldsymbol{x}_i) = y_i) D_t(i) e^{-\alpha_t} + \sum_{i=1}^{n} I(h_t(\boldsymbol{x}_i) \neq y_i) D_t(i) e^{\alpha_t}} \\
&= \frac{e^{\alpha_t}\epsilon_t}{e^{-\alpha_t}(1-\epsilon_t) + e^{\alpha_t}\epsilon_t} = \frac{1}{2}.
\end{aligned}$$

Similar disposition can be observed in the U-Boost algorithm as follows. First, we consider the unnormalized U-Boost algorithm. Let us focus on the value of $f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i)$. There are four different cases:

$$f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i) = \begin{cases} -1, & \text{if } y_i \in h_t(\boldsymbol{x}_i) \text{ and } y \notin h_t(\boldsymbol{x}_i), \\ 0, & \text{if } y_i \in h_t(\boldsymbol{x}_i) \text{ and } y \in h_t(\boldsymbol{x}_i), \\ 0, & \text{if } y_i \notin h_t(\boldsymbol{x}_i) \text{ and } y \notin h_t(\boldsymbol{x}_i), \\ 1, & \text{if } y_i \notin h_t(\boldsymbol{x}_i) \text{ and } y \in h_t(\boldsymbol{x}_i). \end{cases}$$

Intuitively speaking, the first is the case where h_t is correct for y and y_i, the second and third are the cases where h_t is partially correct or partially wrong, and the last is the case where h_t is wrong, because the correct classification rule for $\boldsymbol{x}_i$ is to output $\{y_i\}$. Now let us define the weight

$$D_{t+1}(i, y) = \frac{q_t(y|\boldsymbol{x}_i)}{Z_{t+1}}, \tag{37}$$

where Z_{t+1} is a normalization constant defined by

$$Z_{t+1} = \sum_{i=1}^{n} \sum_{y \neq y_i} q_t(y|\boldsymbol{x}_i), \tag{38}$$

and then define the weighted error by

$$\begin{aligned}\epsilon_{t+1}(h) &= \sum_{i=1}^{n}\sum_{y\neq y_i}\left(\frac{f(\boldsymbol{x}_i,y)-f(\boldsymbol{x}_i,y_i)+1}{2}\right)D_{t+1}(i,y)\\ &= \sum_{\substack{1\le i\le n\\ y_i\notin h(\boldsymbol{x}_i)\\ y\in h(\boldsymbol{x}_i)}} D_{t+1}(i,y)+\sum_{\substack{1\le i\le n\\ y_i\notin h(\boldsymbol{x}_i)\\ y\notin h(\boldsymbol{x}_i)\\ y\neq y_i}}\frac{1}{2}D_{t+1}(i,y)+\sum_{\substack{1\le i\le n\\ y_i\in h(\boldsymbol{x}_i)\\ y\in h(\boldsymbol{x}_i)\\ y\neq y_i}}\frac{1}{2}D_{t+1}(i,y).\end{aligned} \tag{39}$$

Note that $f(\boldsymbol{x}_i,y)-f(\boldsymbol{x}_i,y_i)$ vanishes when $y=y_i$ despite whether $h(\boldsymbol{x}_i)$ is correct or not, hence we omit $q_t(y_i|\boldsymbol{x}_i)$ from the weighted error, that is to say, the weights are defined only on incorrect labels as in AdaBoost.M2 (Freund and Schapire, 1996). Also note that the correct rate is written as

$$1-\epsilon_{t+1}(h)=\sum_{\substack{1\le i\le n\\ y_i\in h(\boldsymbol{x}_i)\\ y\notin h(\boldsymbol{x}_i)}} D_{t+1}(i,y)+\sum_{\substack{1\le i\le n\\ y_i\notin h(\boldsymbol{x}_i)\\ y\notin h(\boldsymbol{x}_i)\\ y\neq y_i}}\frac{1}{2}D_{t+1}(i,y)+\sum_{\substack{1\le i\le n\\ y_i\in h(\boldsymbol{x}_i)\\ y\in h(\boldsymbol{x}_i)\\ y\neq y_i}}\frac{1}{2}D_{t+1}(i,y), \tag{40}$$

and that the second and third terms in the right-hand side are the same in the error rate.

Then by differentiating the U-loss for the unnormalized U-Boost, we know that α_t satisfies

$$\sum_{i=1}^{n}\sum_{y\in\mathcal{Y}}(f_t(\boldsymbol{x}_i,y)-f_t(\boldsymbol{x}_i,y_i))u\big(\xi(q_{t-1}(y|\boldsymbol{x}_i))+\alpha_t(f_t(\boldsymbol{x}_i,y)-f_t(\boldsymbol{x}_i,y_i))\big)=0, \tag{41}$$

namely, by using the definition

$$q_t(y|\boldsymbol{x}_i)=u\big(\xi(q_{t-1}(y|\boldsymbol{x}_i))+\alpha_t(f_t(\boldsymbol{x}_i,y)-f_t(\boldsymbol{x}_i,y_i))\big),$$

the above equation is rewritten as

$$\begin{aligned}&\sum_{i=1}^{n}\sum_{y\in\mathcal{Y}}(f_t(\boldsymbol{x}_i,y)-f_t(\boldsymbol{x}_i,y_i))q_t(y|\boldsymbol{x}_i)\\ &=\sum_{\substack{1\le i\le n\\ y_i\notin h_t(\boldsymbol{x}_i)\\ y\in h_t(\boldsymbol{x}_i)}} q_t(y|\boldsymbol{x}_i)-\sum_{\substack{1\le i\le n\\ y_i\in h_t(\boldsymbol{x}_i)\\ y\notin h_t(\boldsymbol{x}_i)}} q_t(y|\boldsymbol{x}_i)\\ &=0,\end{aligned}$$

that is

$$\sum_{\substack{1\le i\le n\\ y_i\notin h_t(\boldsymbol{x}_i)\\ y\in h_t(\boldsymbol{x}_i)}} q_t(y|\boldsymbol{x}_i)=\sum_{\substack{1\le i\le n\\ y_i\in h_t(\boldsymbol{x}_i)\\ y\notin h_t(\boldsymbol{x}_i)}} q_t(y|\boldsymbol{x}_i). \tag{42}$$

By imposing the above relation into (39) and (40), we observe

$$\epsilon_{t+1}(h_t) = 1 - \epsilon_{t+1}(h_t),$$

which concludes

$$\epsilon_{t+1}(h_t) = \frac{1}{2}. \tag{43}$$

Similarly, in the case of the normalized U-Boost algorithm, the above relation is proved as follows. We use the same definitions of the weight (37) and the error (39). By differentiation the U-loss, α_t satisfies

$$\begin{aligned} &-\sum_{i=1}^{n}\Big(f_t(\boldsymbol{x}_i, y_i) - b_t'(\boldsymbol{x}_i, \alpha_t)\Big) \\ &+\sum_{i=1}^{n}\sum_{y\in\mathcal{Y}}\Big(f_t(\boldsymbol{x}_i, y) - b_t'(\boldsymbol{x}_i, \alpha_t)\Big) u\big(\xi(q_{t-1}(y|\boldsymbol{x}_i)) + \alpha_t f_t(\boldsymbol{x}_i, y) - b_t(\boldsymbol{x}_i, \alpha_t)\big) = 0. \end{aligned} \tag{44}$$

Using the definition of $q_t(y|\boldsymbol{x}_i)$ and the constraint $\sum_{y\in\mathcal{Y}} q_t(y|\boldsymbol{x}_i) = 1$, the above equation is rewritten as

$$\begin{aligned} &-\sum_{i=1}^{n}\Big(f_t(\boldsymbol{x}_i, y_i) - b_t'(\boldsymbol{x}_i, \alpha_t)\Big) + \sum_{i=1}^{n}\sum_{y\in\mathcal{Y}}\Big(f_t(\boldsymbol{x}_i, y) - b_t'(\boldsymbol{x}_i, \alpha_t)\Big) q_t(y|\boldsymbol{x}_i) \\ &= -\sum_{i=1}^{n} f_t(\boldsymbol{x}_i, y_i) + \sum_{i=1}^{n} b_t'(\boldsymbol{x}_i, \alpha_t) + \sum_{i=1}^{n}\sum_{y\in\mathcal{Y}} f_t(\boldsymbol{x}_i, y) q_t(y|\boldsymbol{x}_i) - \sum_{i=1}^{n} b_t'(\boldsymbol{x}_i, \alpha_t) \\ &= \sum_{i=1}^{n}\sum_{y\in\mathcal{Y}}\Big(f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i)\Big) q_t(y|\boldsymbol{x}_i) \\ &= 0 \end{aligned}$$

This is equivalent to (42) and it proves (39).

In this way, the U-Boost algorithm updates the distribution into the least favorable at each step.

4.3 Consistency and Bayes Optimality

Using the basic property of the Bregman divergence, we can show the consistency of the U-loss as follows.

Lemma 3. *Let $p(y|\boldsymbol{x})$ be the true conditional distribution and $F(\boldsymbol{x}, y)$ be the minimizer of the U-loss $L_U(p, q_F)$. The classification rule given by F becomes Bayes optimal*

$$\hat{y}(\boldsymbol{x}) = \operatorname*{argmin}_{y\in\mathcal{Y}} F(\boldsymbol{x}, y) = \operatorname*{argmin}_{y\in\mathcal{Y}} p(y|\boldsymbol{x}). \tag{45}$$

Proof. From the property of the Bregman divergence,

$$D_U(p, q) = 0 \;\Leftrightarrow\; p(y|\boldsymbol{x}) = q(y|\boldsymbol{x}) \text{ (a.e. } \boldsymbol{x})$$

and equivalence relation (14), the minimizer of the U-loss $L_U(p, q_F)$ over $F(\boldsymbol{x}, y)$ is given by

$$F(\boldsymbol{x}, y) = \operatorname*{argmin}_F L_U(p, q_F) = \xi(p(y|\boldsymbol{x})).$$

The statement comes from the monotonicity of ξ

$$\operatorname*{argmin}_{y\in\mathcal{Y}} p(y|\boldsymbol{x}) = \operatorname*{argmin}_{y\in\mathcal{Y}} \xi(p(y|\boldsymbol{x})).$$

□

In the U-Boost algorithm, $F(\boldsymbol{x}, y)$ is chosen from a class of functions which are linear combination of $f_t(\boldsymbol{x}, y); t = 1, \ldots, T$. In the case that the true distribution is not in the considered U-model, the closest point in the model is chosen in the sense of U-loss, however, if the number of boosting is sufficiently large and the functions $f_t; t = 1, \ldots, T$ are diverse, U-model can well approximate the true distribution. See for example Barron (1993); Murata (1996), for the discussion about the richness of the linear combination of simple functions.

For the binary case, where the U-loss is given by

$$L_U(p, q) = \int_{\mathcal{X}} \sum_{y\in\{\pm 1\}} p(y|\boldsymbol{x}) U(-yF(\boldsymbol{x})) d\mu(\boldsymbol{x}),$$

we can show the following theorem.

Theorem 2. *The minimizer of the U-loss gives the Bayes optimal, that is,*

$$\{\boldsymbol{x}|F(\boldsymbol{x}) > 0\} = \left\{\boldsymbol{x}\middle| \log\frac{p(+1|\boldsymbol{x})}{p(-1|\boldsymbol{x})} > 0\right\}.$$

Moreover, if

$$\log\frac{u(z)}{u(-z)} = 2z \tag{46}$$

holds, F coincides with the log likelihood ratio

$$F(x) = \frac{1}{2}\log\frac{p(+1|\boldsymbol{x})}{p(-1|\boldsymbol{x})}.$$

Proof. By usual variational arguments, the minimizer of the U-loss satisfies

$$\int_{\mathcal{X}} \Big(p(+1|\boldsymbol{x})u(-F(\boldsymbol{x})) - p(-1|\boldsymbol{x})u(F(\boldsymbol{x}))\Big)\Delta(\boldsymbol{x})d\mu(\boldsymbol{x}) = 0$$

for any function $\Delta(\boldsymbol{x})$, hence

$$\log\frac{p(+1|\boldsymbol{x})}{p(-1|\boldsymbol{x})} = \log\frac{u(F(\boldsymbol{x}))}{u(-F(\boldsymbol{x}))} \text{ (a.e. } \boldsymbol{x}).$$

Knowing that for any convex function U,

$$\rho(z) = \log\frac{u(z)}{u(-z)}$$

is monotonically increasing and satisfies $\rho(0) = 0$. This directly shows the first part of the Lemma and by imposing $\rho(z) = 2z$, the second part is proved. □

The last part of the theorem agree with the result in Eguchi and Copas (2001, 2002). U-functions for AdaBoost, LogitBoost, and MadaBoost satisfy the condition (46).

4.4 Asymptotic Covariance

To see the efficiency of the U-loss, we investigate the asymptotic variance of $\boldsymbol{\alpha}$ in this section.

Let us consider the U-model parameterized by $\boldsymbol{\alpha} = \{\alpha_t; t = 1, \ldots, T\}$

$$q(y|\boldsymbol{x}) = u\big(\xi(q_0(y|\boldsymbol{x})) + \sum_{i=1}^{T} \alpha_t f_t(\boldsymbol{x}, y) - b(\boldsymbol{x}, \boldsymbol{\alpha})\big),$$

and let $p(y|\boldsymbol{x})$ be the true conditional distribution. The optimal point q^* in the U-model is given by

$$\boldsymbol{\alpha}^* = \operatorname*{argmin}_{\boldsymbol{\alpha}} L_U(p, q) = \operatorname*{argmin}_{\boldsymbol{\alpha}} \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} \{U(\xi(q)) - p\xi(q)\}\, d\mu, \tag{47}$$

and for given n examples, the estimate of $\boldsymbol{\alpha}$ is given by

$$\hat{\boldsymbol{\alpha}} = \operatorname*{argmin}_{\boldsymbol{\alpha}} L_U(\tilde{p}, q) = \operatorname*{argmin}_{\boldsymbol{\alpha}} \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} \{U(\xi(q)) - \tilde{p}\xi(q)\}\, d\mu, \tag{48}$$

where $\tilde{p}$ is the empirical distribution of given examples. When n is sufficiently large, the covariance of $\hat{\boldsymbol{\alpha}}$ with respect to all the possible sample sets is given as follows.

Lemma 4. *The asymptotic covariance of $\hat{\boldsymbol{\alpha}}$ is given by*

$$\mathrm{Cov}(\hat{\boldsymbol{\alpha}}) = \frac{1}{n} H^{-1} G H^{-1} + o\left(\frac{1}{n}\right) \tag{49}$$

where H and G are $T \times T$ matrices defined by

$$\begin{aligned}
H &= \frac{\partial^2}{\partial \boldsymbol{\alpha} \partial \boldsymbol{\alpha}^\tau} L_U(p, q^*) \\
&= \int_{\mathcal{X}} \frac{\partial^2}{\partial \boldsymbol{\alpha} \partial \boldsymbol{\alpha}^\tau} r(\boldsymbol{x}, \boldsymbol{\alpha}^*) d\mu(\boldsymbol{x}), \\
G &= \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} p \frac{\partial}{\partial \boldsymbol{\alpha}} (U(\xi(q^*)) - \xi(q^*)) \frac{\partial}{\partial \boldsymbol{\alpha}^\tau} (U(\xi(q^*)) - \xi(q^*)) d\mu \\
&= \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} p(y|\boldsymbol{x}) \Big(\frac{\partial}{\partial \boldsymbol{\alpha}} r(\boldsymbol{x}, \boldsymbol{\alpha}^*) - \boldsymbol{f}(\boldsymbol{x}, y) \Big) \Big(\frac{\partial}{\partial \boldsymbol{\alpha}^\tau} r(\boldsymbol{x}, \boldsymbol{\alpha}^*) - \boldsymbol{f}(\boldsymbol{x}, y) \Big) d\mu(\boldsymbol{x}),
\end{aligned}$$

where r is the function of $\boldsymbol{x}$ defined by

$$r(\boldsymbol{x}, \boldsymbol{\alpha}) = \sum_{y \in \mathcal{Y}} U\big(\xi(q_0(y|\boldsymbol{x})) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y) - b(\boldsymbol{x}, \boldsymbol{\alpha})\big) + b(\boldsymbol{x}, \boldsymbol{\alpha}).$$

The proof is simply given by usual asymptotic arguments (see Murata et al., 1994, for example).

When the true distribution is included in the U-model, that is $p = q^*$, the asymptotic covariance of LogitBoost becomes

$$\mathrm{Cov}(\hat{\boldsymbol{\alpha}}) = \frac{1}{n} I^{-1} + o\left(\frac{1}{n}\right)$$

where I is the Fisher information matrix of the logistic model, which means LogitBoost attains the Cramer-Rao bound asymptotically, that is, LogitBoost is asymptotic efficient. In general, the asymptotic covariance of U-Boost algorithms are inferior to the Cramer-Rao bound, hence from this point of view, U-Boost is not efficient, however, instead of the efficiency, some of the U-Boost algorithms show robustness as discussed in the next section.

The expected U-loss of q_t estimated with given n examples is asymptotically bounded by

$$E(L_U(p, q_t)) = L_U(p, q^*) + \frac{1}{2n} \operatorname{tr} H^{-1} G + o\left(\frac{1}{n}\right), \tag{50}$$

where E is the expectation over all the possible sample sets (cf. Murata et al., 1994).

4.5 Robustness of U-Boost

In this section, we study the robustness of the U-Boost for the binary classification problem. First, we consider the robust condition for U-functions, then discuss the robustness of the algorithm.

4.5.1 Most B-robust U-function

Let us consider the statistical model with one parameter α

$$M = M(h) = \left\{ p_\alpha(y|\boldsymbol{x}) = \frac{1}{1 + \exp\{-2y(F(\boldsymbol{x}) + \alpha h(\boldsymbol{x}))\}}; \alpha \in R \right\}, \tag{51}$$

where $h(\boldsymbol{x})$ takes $+1$ or -1 and $F(\boldsymbol{x})$ is the log likelihood ratio of the true distribution $p(y|\boldsymbol{x})$

$$F(\boldsymbol{x}) = \frac{1}{2} \log \frac{p(+1|\boldsymbol{x})}{p(-1|\boldsymbol{x})},$$

that is the true parameter is $\alpha = 0$, namely $p = p_0$.

We define the estimator of α with the U-function as

$$\alpha_U(q\mu) = \operatorname*{argmin}_\alpha \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} q(y|\boldsymbol{x}) U\big(-y(F(\boldsymbol{x}) + \alpha h(\boldsymbol{x}))\big) d\mu(\boldsymbol{x}),$$

where $q\mu$ is the joint distribution of $\boldsymbol{x}$ and y. As considered in the previous section when the U-function satisfies the condition (46), which asserts the estimator to be log-likelihood consistent, the estimator by the U-function is Fisher consistent, that is,

$$\alpha_U(p_\alpha \mu) = \alpha.$$

The robustness of the estimator is measured by the gross error sensitivity (Hampel et al., 1986)

$$\gamma(U, p_0) = \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} \left\{ \lim_{\epsilon \to +0} \frac{1}{\epsilon} \Big[\alpha_U((1-\epsilon)\, p_0 \mu + \epsilon\, \delta(\tilde{\boldsymbol{x}}, \tilde{y})) - \alpha_U(p_0 \mu) \Big] \right\}^2, \tag{52}$$

where $\delta(\tilde{\boldsymbol{x}}, \tilde{y})$ is the probability distribution with a point mass at $(\tilde{\boldsymbol{x}}, \tilde{y})$. The gross error sensitivity measures the worst influence which a small amount of contamination can have on the value of the estimator. The estimator which minimizes the gross error sensitivity is called the most B-robust estimator. For a choice of a robust U-function, we show the following theorem.

Theorem 3. *The U-function which derives MadaBoost algorithm minimizes the gross error sensitivity among the U-function with the property of* (46).

Proof. By the brief calculation, the gross error sensitivity of the estimator is written as

$$\gamma(U, p_0) = \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} u(\tilde{y}F(\tilde{\boldsymbol{x}}))^2 \left(2 \int_{\mathcal{X}} u(F(\boldsymbol{x})) p_0(-1|\boldsymbol{x}) d\mu(\boldsymbol{x}) \right)^{-2}. \tag{53}$$

From this, if u is not bounded such as $u(z) = \exp(z)$, the gross error sensitivity diverges. Therefore we focus on the case that u is bounded. Without loss of generality, we can suppose

$$\sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} u(\tilde{y}F(\tilde{\boldsymbol{x}}))^2 = 1$$

because the multiplication of the positive value to the U-function does not change the estimator. To minimize the gross error sensitivity, we need to find a U-function which maximizes

$$\int_{\mathcal{X}} u(F(\boldsymbol{x})) p_0(-1|\boldsymbol{x}) d\mu(\boldsymbol{x}).$$

The pointwise maximization of $u(z)$ under the conditions

$$u(-z) = u(z)e^{-2z} \quad \text{and} \quad \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} u(\tilde{y}F(\tilde{\boldsymbol{x}})) = 1$$

leads to

$$u(z) = \begin{cases} 1, & z \geq 0, \\ \exp(2z), & z < 0, \end{cases}$$

and this coincides with the MadaBoost U-function. □

4.5.2 Robustness of boosting algorithm

Next, we study the robustness of the estimator by the U-boost. Let us consider the estimator $F(\boldsymbol{x}) + \alpha h(\boldsymbol{x})$ updated from $F(\boldsymbol{x})$ by the U-Boost procedure. The robustness of this updated estimator is measured by the gross error sensitivity with the Kullback-Leibler divergence D defined by

$$\gamma_{\text{boost}}(U, p) = \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} \lim_{\epsilon \to +0} \frac{2}{\epsilon^2} D(p_0, p_{\alpha_U, \epsilon, (\tilde{\boldsymbol{x}}, \tilde{y})}), \tag{54}$$

where $p_{\alpha_U, \epsilon, (\tilde{\boldsymbol{x}}, \tilde{y})}$ means the updated estimator which is calculated on the assumption that the true distribution is contaminated as $(1 - \epsilon)p(y|\boldsymbol{x})\mu(\boldsymbol{x}) + \epsilon\delta(\tilde{\boldsymbol{x}}, \tilde{y})$.

Let us define $I(h)$ as the Fisher information matrix of the model $M(h)$ at $\alpha = 0$. From the property of $(yh(\boldsymbol{x}))^2 = 1$ we find that $I(h)$ does not depend on h, thus it can be written as I. Then the gross error sensitivity for the boosting algorithm is written as

$$\gamma_{\text{boost}}(U, p_0) = I\gamma(U, p_0). \tag{55}$$

Hence the U-function of MadaBoost also minimizes $\gamma_{\text{boost}}(U, p_0)$. As a consequence, MadaBoost minimizes the influence of outliers when the estimator is close to the true distribution.

5 Illustrative Examples

In the following numerical experiments, we study the two-dimensional binary classification problem with "stumps" (Friedman et al., 2000). We generate labeled examples subject to a fixed probability and a few examples are flipped by the contamination as shown in Fig. 7. The detailed setup is

$$\begin{aligned}
&\boldsymbol{x} = (x_1, x_2) \in \mathcal{X} = [-\pi, \pi] \times [-\pi, \pi] \\
&y \in \mathcal{Y} = \{+1, -1\} \\
&\mu(\boldsymbol{x}) : \text{uniform on } \mathcal{X} \\
&p(y|\boldsymbol{x}) = \frac{1 + \tanh(F(\boldsymbol{x}))}{2} \\
&\qquad \text{where } F(\boldsymbol{x}) = x_2 - 3\sin(x_1)
\end{aligned}$$

and $a\%$ contaminated data are generated according to the following procedure. First, examples are sorted by descending order of $|F(\boldsymbol{x}_i)|$ and from top $10a\%$ examples, $a\%$ are randomly chosen and flipped without replacement. That means the contamination is avoided around the boundary of classification. The plots are made by averaging 50 different runs, and in each run 300 training data are produced and the classification error rate is calculated with 4000 test data.

The training results by three boosting methods, AdaBoost, LogitBoost and MadaBoost, are compared from the viewpoint of the robustness.

In Fig. 8 (a),(b) and (c), we show the test error evolution in regard to the number of boosting. All the boosting methods show overfit phenomena as the number of boosting increases. We can see that AdaBoost is quite sensitive to the contaminated data.

To show the robustness to the contamination, we plot the test error differences against the number of boosting. In Fig. 9 (a) and (b), the difference between 1%-contamination and non-contamination, and between 2%-contamination and non-contamination are plotted, respectively. In this classification problem, we observer AdaBoost is more sensitive to outliers than MadaBoost as shown in the previous section.

6 Conclusion

In this paper, we formulated boosting algorithms as sequential updates of conditional measures, and we introduced a class of boosting algorithms by considering

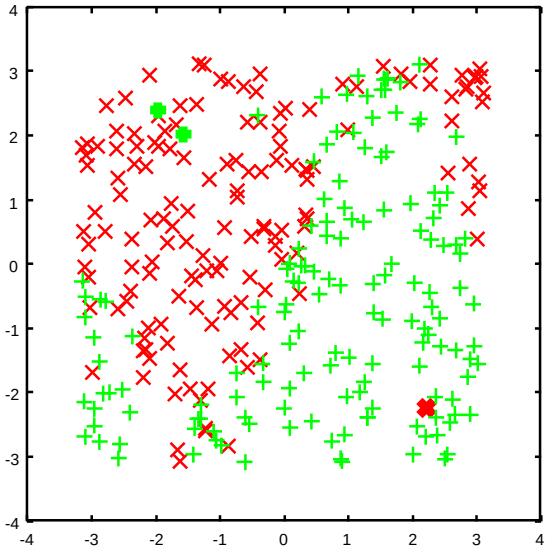


Figure 7: Typical examples with contamination.

the relation with the Bregman divergence. By dint of the statistical framework, properties of consistency, efficiency and robustness are discussed.

Still detailed studies on some properties such as the rate of convergence, and stopping criteria of boosting are needed to avoid overfitting problem and so on.

Here we only treated the classification problem, but the formulation can be extended to the case where y is in some continuous space, such as regression and density estimation. This is also remained as a future work.

References

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

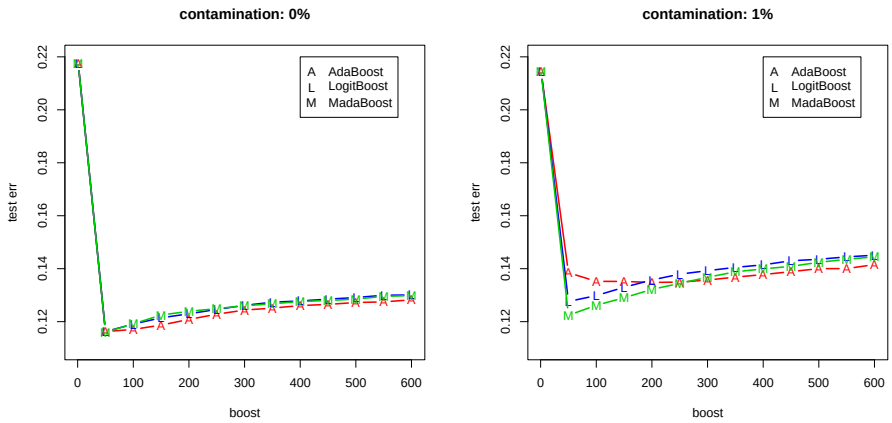
S. Amari and H. Nagaoka. *Methods of Information Geometry*, volume 191 of *Translations of Mathematical Monographs*. Oxford University Press, 2000.

A. R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Information Theory*, 39(3):930–945, May 1993.

C. Bishop. *Neural Networks for Pattern Recognition*. Clarendon Press, Oxford, 1995.

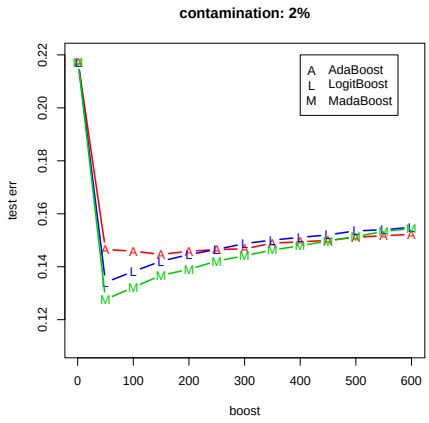
C. Domingo and O. Watanabe. MadaBoost: A modification of AdaBoost. In *Proc. of the 13th Conference on Computational Learning Theory, COLT'00*, 2000.

S. Eguchi and J. B. Copas. Recent developments in discriminant analysis from an information geometric point of view. *Journal of the Korean Statistical*



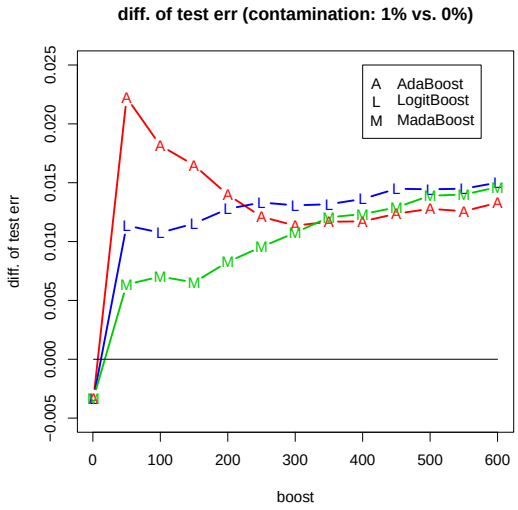
(a)

(b)

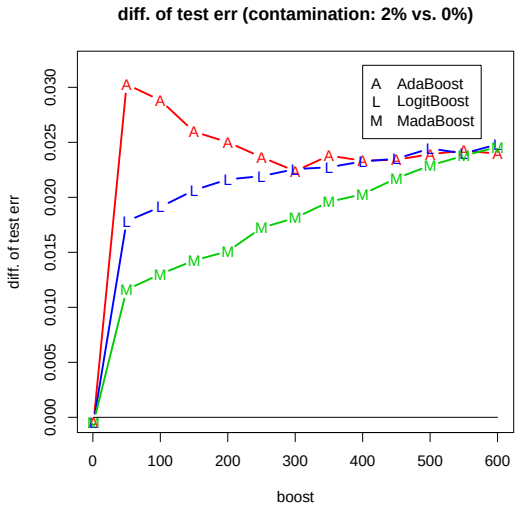


(c)

Figure 8: Test error of boosting algorithms. (a) training data is not contaminated. (b) 1% contamination. (c) 2% contamination.



(a)



(b)

Figure 9: Difference of test errors of the original examples and that of the contaminated examples. (a) difference between 1%-contamination and non-contamination. (b) difference between 2%-contamination and non-contamination.

Society, 30:247–264, 2001. (The special issue of the 30th anniversary of the Korean Statistical Society).

S. Eguchi and J. B. Copas. A class of logistic type discriminant functions. *Biometrika*, 89:1–22, 2002.

S. Eguchi and Y. Kano. Robustifing maximum likelihood estimation by psi-divergence. ISM Research Memo 802, The Institute of Statistical Mathematics, 2001.

Y. Freund. Boosting a weak learning algorithm by majority. *Information and Computation*, 121(2):256–285, 1995.

Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Machine Learning: Proceedings of the Thirteenth International Conference*, pages 148–156, 1996.

Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, aug 1997.

J. H. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: A statistical view of boosting. *Annals of Statistics*, 28:337–407, 2000.

F. R. Hampel, P. J. Rousseeuw, E. M. Ronchetti, and W. A. Stahel. *Robust Statistics*. John Wiley and Sons, Inc., 1986.

T. Hastie, R. Tibishirani, and J. Friedman. *The elements of statistical learning*. Springer, New York, 2001.

M. Kearns and L. G. Valiant. Learning boolean formulae or finte automata is as hard as foctoring. Technical Report TR-14-88, Harvard University Aiken Computation Laboratry, aug 1988.

G. Lebanon and J. Lafferty. Boosting and maximum likelihood for exponential models. Technical Report CMU-CS-01-144, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, aug 2001.

G. J. MacLachlan. *Discriminant analysis and statistical pattern recognition*. Wiley, New York, 1992.

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

N. Murata. An integral representation with ridge functions and approximation bounds of three-layered network. *Neural Networks*, 9(6):947–956, 1996.

N. Murata, S. Yoshizawa, and S. Amari. Network information criterion — determining the number of hidden units for an artificial neural network model. *IEEE Trans. Neural Networks*, 5(6):865–872, 1994.

R. E. Schapire. The strength of weak learnability. *Machine Learning*, 5:197–227, 1990.

R. E. Schapire, Y. Freund, P. Bartlett, and W. S. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26(5):1651–1686, 1998.

T. Takenouchi and S. Eguchi. Robustifying AdaBoost by adding the naive error rate. ISM Reserach Memorandam 859, Institute of Statistical Mathematics, Tokyo, Japan, 2002.

V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.