LETTER **Communicated by Shun-ichi Amari**

Information Geometry of U-Boost and Bregman Divergence

Noboru Murata
noboru.murata@eb.waseda.ac.jp
School of Science and Engineering, Waseda University, Shinjuku, Tokyo 169-8555, Japan

Takashi Takenouchi
ttakashi@ism.ac.jp
Department of Statistical Science, Graduate University of Advanced Studies, Minato, Tokyo 106-8569, Japan

Takafumi Kanamori
kanamori@is.titech.ac.jp
Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Meguro, Tokyo 152-8552, Japan

Shinto Eguchi
eguchi@ism.ac.jp
Institute of Statistical Mathematics, Japan, and Department of Statistical Science, Graduate University of Advanced Studies, Minato, Tokyo 106-8569, Japan

We aim at an extension of AdaBoost to U-Boost, in the paradigm to build a stronger classification machine from a set of weak learning machines. A geometric understanding of the Bregman divergence defined by a generic convex function U leads to the U-Boost method in the framework of information geometry extended to the space of the finite measures over a label set. We propose two versions of U-Boost learning algorithms by taking account of whether the domain is restricted to the space of probability functions. In the sequential step, we observe that the two adjacent and the initial classifiers are associated 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 expectation-maximization algorithm. Statistical discussions for consistency and robustness elucidate the properties of the U-Boost methods based on a stochastic assumption for training data.

1 Introduction

In the past decade, several novel developments for classification and pattern recognition have been done, mainly along statistical learning theory

(see, e.g., McLachlan, 1992; Bishop, 1995; Vapnik, 1995; Hastie, Tibshirani, & Friedman, 2001). Several important approaches have been proposed and implemented in feasible computational algorithms. One promising direction is boosting, which is a method of combining many learning machines trained by simple learning algorithms. Theoretical research on boosting began with a 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 was given by Schapire (1990), who 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 the boosting algorithm is to sort important and unimportant examples according to whether machines are good at or weak in predicting those examples. The procedures are summarized in the following three types:

- **Filtering:** New examples are sampled and filtered by the previous trained machines so that as many difficult examples as easy examples are collected (Schapire, 1990; Domingo & Watanabe, 2000).
- **Resampling:** Examples are resampled from given examples repeatedly so that difficult examples are chosen with a high probability (Freund, 1995).
- **Reweighting:** Given examples are weighted so that difficult examples severely affect the error (Freund & Schapire, 1997; Friedman, Hastie, & Tibshirani, 2000).

In this letter, we focus on the reweighting method including AdaBoost (Freund & Schapire, 1997). Lebanon and Lafferty (2001) give a geometric consideration of the extended Kullback-Leibler divergence, which leads to a close relation between AdaBoost and logistic discrimination. The main objective in this letter is to propose a class of boosting algorithms, U-Boost, which is naturally derived from the Bregman divergence. This proposal provides an extension of the geometry discussed by Lebanon and Lafferty and elucidates the fact that the Bregman divergence associates the normalized with the unnormalized U-Boost from the viewpoint of information geometry. There are several different purposes for classification problems. The extension from AdaBoost to U-Boost can correspond to these different requirements, including robustness for mislabeling and outliers in a feature vectors. In this context MadaBoost and Eta-Boost in the class of U-Boost are introduced and discussed.

This letter is organized as follows. In section 2, we give the structure of the U-Boost algorithm proposed here. 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 information geometry, and we give a geometrical understanding of the U-Boost algorithm. In section 4, we

Table 1: Relationship Between Classifiers and Decision Functions.

Classifier	Decision Function
$h : \boldsymbol{x} \in \mathcal{X} \mapsto \mathcal{C} \subset \mathcal{Y}$	$f : (\boldsymbol{x}, y) \in \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$
	$\downarrow$ combine
$H : \boldsymbol{x} \in \mathcal{X} \mapsto y \in \mathcal{Y}$	$F : (\boldsymbol{x}, y) \in \mathcal{X} \times \mathcal{Y} \to R$
$\xleftarrow[\text{argmax}]{}$	
voting classifier	combined decision function

discuss the consistency, efficiency, and robustness of the algorithm based on the statistical consideration given in the previous section and provide some illustrative examples with numerical simulations. The last section sets out our concluding remarks and future work.

2 U-Boost Algorithm

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 the problem is mainly written in the case of multiclass discrimination; however we consider the case where y is the binary label with values -1 and 1 for the intuitive examples in this article.

Let h be a classifier, and h is a set-valued function, that is, for a given feature vector $\boldsymbol{x}$, h returns a subset $\mathcal{C}$ of class labels,

$$h : \boldsymbol{x} \in \mathcal{X} \mapsto \mathcal{C} \subset \mathcal{Y}. \tag{2.1}$$

Possible class labels are a, b, and c in $\mathcal{Y}$ for a feature vector $\boldsymbol{x}$; then h returns a set of labels,

$$h(\boldsymbol{x}) = \{a, b, c\}.$$

Next, we define a decision function on $\mathcal{X} \times \mathcal{Y}$ from the classifier h as

$$f(\boldsymbol{x}, y) = \begin{cases} 1, & \text{if } y \in h(\boldsymbol{x}), \\ 0, & \text{otherwise}, \end{cases} \tag{2.2}$$

namely,

$$f(\boldsymbol{x}, y) = I(y \in h(\boldsymbol{x})), \tag{2.3}$$

where I is the indicator function defined by

$$I(A) = \begin{cases} 1, & \text{if } A \text{ is true}, \\ 0, & \text{otherwise}. \end{cases} \tag{2.4}$$

In the following discussion, we identify a classifier h with a decision function f with the above correspondence.

For a set of decision functions $f_t(\boldsymbol{x}, y)$ $(t = 1, \ldots, T)$, let us define a combined decision function with a confidence rate $\alpha_t \in R$ $(t = 1, \ldots, T)$ as

$$F(\boldsymbol{x}, y) = \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y). \tag{2.5}$$

Note that F is a real-valued function; it is no longer an indicator function. By using the combined decision function, a voting classifier of a set of classifiers $h_t(\boldsymbol{x})$ $(t = 1, \ldots, T)$ is defined as

$$H(\boldsymbol{x}) = \operatorname*{argmax}_{y \in \mathcal{Y}} F(\boldsymbol{x}, y). \tag{2.6}$$

Hence, we follow the diagram in Table 1 for the general discussion on classification problems.

Let U be a convex function on R with the positive derivative U', and let us fix a set $\mathcal{F}$ of decision functions. Here, $\mathcal{F}$ is usually taken by a set of boosting stamps, decision trees, or neural networks. In this formulation of classifiers and decision functions, the U-Boost algorithm over $\mathcal{F}$ is described as follows.

U-Boost Algorithm.

Input: Given n examples $\{(\boldsymbol{x}_i, y_i); \boldsymbol{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}, i = 1, \ldots, n\}$.

Initialize: $D_1(i, y) = 1/n(|\mathcal{Y}| - 1)$, $i = 1, \ldots, n$, and $F_0(\boldsymbol{x}, y) = 0$, where $|\mathcal{Y}|$ is the cardinality of $\mathcal{Y}$.

Do for $t = 1, \ldots, T$

Step 1: Define an error of decision function f (classifier h) under the distribution D_t as

$$\epsilon_t(f) = \sum_{i=1}^{n} \sum_{y \neq y_i} \frac{f(\boldsymbol{x}_i, y) - f(\boldsymbol{x}_i, y_i) + 1}{2} D_t(i, y).$$

Then select a decision function such that

$$f_t(\boldsymbol{x}, y) = \operatorname*{argmin}_{f \in \mathcal{F}} \epsilon_t(f).$$

Step 2: Calculate the confidence rate α_t as

$$\alpha_t = \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} U\,(F_{t-1}(\boldsymbol{x}_i, y) - F_{t-1}(\boldsymbol{x}_i, y_i) + \alpha(f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i))).$$

Step 3: Update the combined decision function by

$$F_t(\boldsymbol{x}, y) = F_{t-1}(\boldsymbol{x}, y) + \alpha_t f_t(\boldsymbol{x}, y)$$

and the distribution D_t by

$$D_{t+1}(i, y) \propto U'\left(F_t(\boldsymbol{x}_i, y) - F_t(\boldsymbol{x}_i, y_i)\right),$$

where $D_{t+1}(i, y)$ is normalized as

$$\sum_{i=1}^{n} \sum_{y \neq y_i} D_{t+1}(i, y) = 1.$$

Output: The final decision by the majority vote as follows:

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

In the U-Boost algorithm, the dependency on U appears only in steps 2 and 3. When the U-function is equal to the exponential function, the above algorithm is reduced to the AdaBoost algorithm. Thus, steps 2 and 3 are simplified:

step 2′:

$$\alpha_t = \frac{1}{2} \log \frac{1 - \epsilon_t(f_t)}{\epsilon_t(f_t)}.$$

step 3′:

$$D_{t+1}(i, y) \propto \exp\{F_t(\boldsymbol{x}_i, y) - F_t(\boldsymbol{x}_i, y_i)\}.$$

In the above description, the best choice of classifier f_t at time t is given. In practice, we do not have to use the optimally chosen classifier, and in general we can adopt a classifier chosen by some weak learning algorithm like the common setup of the boosting method. Also, we can assume $\epsilon_t(f) \leq \frac{1}{2}$ without loss of generality, because when $\epsilon_t(f) > \frac{1}{2}$, we can use the reversed output $1 - f$ instead; then the error becomes $\epsilon_t(1 - f) < \frac{1}{2}$.

We have a fundamental property of the U-Boost algorithm,

$$\epsilon_{t+1}(f_t) = \frac{1}{2} \quad (\forall t = 1, 2, \ldots, T - 1),$$

for any convex function U. It will be discussed in detail and proved in a subsequent section. This means that the reweighting of U-Boost is commonly organized to assign the best selected decision function f_t in the tth stage to the least favorable weight distribution to the example set in the $t + 1$th stage. It is remarkable that this error property is common for all U-Boost algorithms.

We will present a geometric perspective for the U-Boost algorithm by introducing the Bregman U-divergence. In the discussion, the U-Boost al-

gorithm is seen as giving the sequential minimization algorithm of an empirical loss defined by

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

as observed in step 2.

3 Geometrical Structure of U-Boost

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

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

$$\text{where } F(\boldsymbol{x}, y) = \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y).$$

By adopting different loss functions, several variations of AdaBoost are proposed, such as LogitBoost (Friedman et al., 2000) and MadaBoost (Domingo & Watanabe, 2000), where the loss functions,

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

$$\text{LogitBoost:} \qquad \phi(z) = \log(1 + \exp(z))$$

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

are used instead of the exponential loss. In a subsequent discussion, we will focus on the robustness to outliers of the feature vector and class label. It will be shown that MadaBoost is the most robust in the class of logistic consistent methods. The U-Boost algorithm is a natural extension of those based on loss functions.

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, Freund, Bartlett, & Lee, 1998). However, the statistical properties such as consistency and efficiency are not apparent, because the relationship between loss functions and the distributions realized by voting classifiers has been unclear so far.

Lebanon and Lafferty (2001) first pointed out the duality of the AdaBoost algorithm in the space of distributions and discussed its geometrical structure from the viewpoint of linear programming.

In this section, we extend their geometrical consideration by introducing the Bregman divergence. The relationship between the Bregman divergence and boosting has been discussed by Kivinen and Warmuth (1999) and Collins, Schapire, and Singer (2000). Here we define a form of the Bregman divergence that is suited for statistical inferences. Then we discuss special subspaces associated with Bregman divergences. Based on the geometrical framework of the Bregman divergence introduced in this section, we will consider the roles of loss functions in boosting algorithms and discuss some statistical properties of algorithms.

3.1 Space of Finite Measures and Bregman Divergence. Let us consider the space of all the positive finite measures over $\mathcal{Y}$ conditioned by $\boldsymbol{x} \in \mathcal{X}$,

$$\mathcal{M} = \left\{ m(y|\boldsymbol{x}) \;\Big|\; \sum_{y\in\mathcal{Y}} m(y|\boldsymbol{x}) < \infty \text{ (a.e. } \boldsymbol{x}) \right\}, \tag{3.1}$$

and the conditional probability density as its subspace,

$$\mathcal{P} = \left\{ m(y|\boldsymbol{x}) \;\Big|\; \sum_{y\in\mathcal{Y}} m(y|\boldsymbol{x}) = 1 \text{ (a.e. } \boldsymbol{x}) \right\}. \tag{3.2}$$

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

$$\tilde{p}(y|\boldsymbol{x}) = \begin{cases} I(y = y_i), & \text{if } \boldsymbol{x} = \boldsymbol{x}_i, \\ \dfrac{1}{|\mathcal{Y}|}, & \text{otherwise} \end{cases} \tag{3.3}$$

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

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

and the discussion in this article can be extended in a straightforward manner. We also define the empirical marginal density of $\boldsymbol{x}$ with

$$\tilde{\mu}(\boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^n \delta(\boldsymbol{x} - \boldsymbol{x}_i), \tag{3.4}$$

where $\delta(\boldsymbol{x})$ is the delta function that represents a point mass on the origin.

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, and 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 $\mathcal{M}$, let us define the Bregman cross-entropy under the marginal density $\mu(\boldsymbol{x})$ with respect to U by*

$$H_U(p, q; \mu) = \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} \{U(\xi(q(y|\boldsymbol{x}))) - p(y|\boldsymbol{x})\xi(q(y|\boldsymbol{x}))\}\mu(\boldsymbol{x})d\boldsymbol{x}, \tag{3.5}$$

and the Bregman divergence from p to q is defined by

$$D_U(p, q; \mu) = H_U(p, q; \mu) - H_U(p, p; \mu). \tag{3.6}$$

In the following, the Bregman divergence $D_U(p, q; \mu)$ associated with the convex function U is called U-divergence for short, and the entropy functions $H_U(p, q; \mu)$ and $H_U(p, p; \mu)$ are referred to as U-cross-entropy and U-entropy, respectively.

For notational simplicity, if the context is clear, we omit $\boldsymbol{x}$ and y from functions and abbreviate $\mu(\boldsymbol{x})d\boldsymbol{x}$ to $d\mu$ in the following, for example,

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

Also we drop μ from H_U and D_U if there is no ambiguity, such as $H_U(p, q)$ and $D_U(p, q)$.

A popular form of the Bregman divergence (see, e.g., Kivinen & Warmuth, 1999; Collins et al., 2000) 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, $\nu(z)$ is a certain measure on z, and d is the difference at g between the function U and the tangent line at $(f, U(f))$:

$$d(f, g) = U(g) - \{u(f)(g - f) + U(f)\}. \tag{3.7}$$

In the definition (see equation 3.6), densities are first mapped by ξ; then the form 3.7 is applied, and the meaning of d is easily understood from Figure 1. Note that from the convexity of U, d is obviously nonnegative; therefore, the Bregman divergence is nonnegative, and $D_U(p, q) = 0$ holds if and only if $p(y|\boldsymbol{x}) = q(y|\boldsymbol{x})$ (a.e. $\boldsymbol{x}$). Also note that the Bregman divergence is in general not symmetric with respect to p and q, as is easily seen; therefore, it is not a distance.

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

$$U^*(\zeta) = \sup_{\theta} \{\zeta\theta - U(\theta)\}.$$

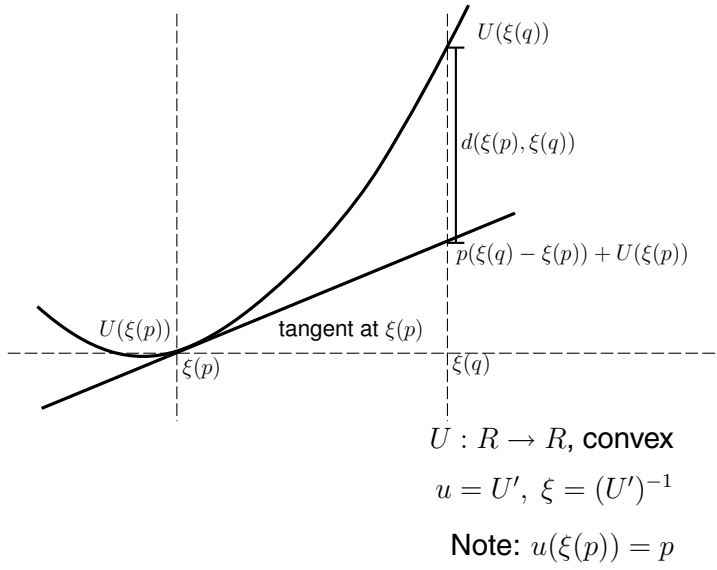


Figure 1: Bregman divergence.

Then d is written with U and U^* simply as

$$d(f, g) = U^*(u(f)) + U(g) - u(f)g.$$

The advantage of the form 3.6 is that it allows us to plug in the empirical distribution directly. Since the Bregman divergence is nonnegative, that is, $D_U(p, q) \geq 0$, the U-cross-entropy is bounded below by U-entropy as

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

Now let us consider a problem in which q is optimized with respect to $D_U(p, q)$ for fixed p. Neglecting the terms that do not depend on q, the problem is simplified as

$$\operatorname*{argmin}_q D_U(p, q) = \operatorname*{argmin}_q H_U(p, q). \tag{3.8}$$

In $H_U(p, q)$, the distribution p appears only for taking the expectation of $\xi(q)$, and therefore the empirical distributions $\tilde{p}$ and $\tilde{\mu}$ are used without any difficulty as

$$H_U(\tilde{p}, q; \tilde{\mu}) = \frac{1}{n} \sum_{i=1}^{n} \left[\sum_{y \in \mathcal{Y}} U(\xi(q(y|\boldsymbol{x}_i))) - \xi(q(y_i|\boldsymbol{x}_i)) \right]. \tag{3.9}$$

Hence, the optimal distribution for this example with respect to U-divergence is written as

$$\tilde{q} = \operatorname*{argmin}_{q} H_U(\tilde{p}, q; \tilde{\mu}).$$

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

Example 1 (U-functions). The following are important examples of the convex function U (see Figure 2):

Exponential (Kullback-Leibler divergence):

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

β-type (β-divergence):

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

Square-type ($\beta = 1$):

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

η-type (η-divergence):

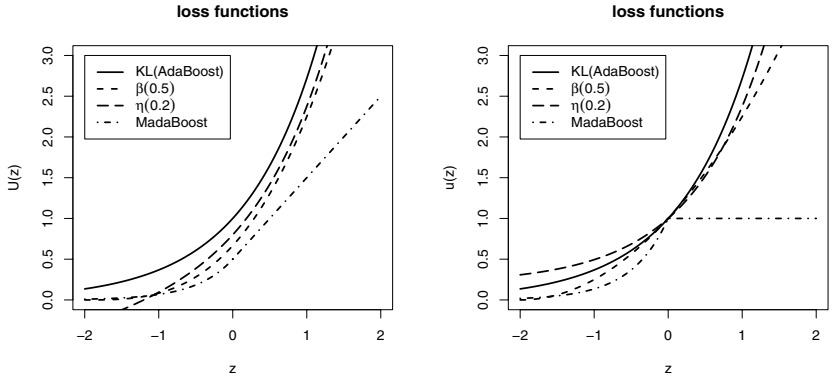
$$U(z) = (1-\eta)\exp(z) + \eta z, \quad u(z) = (1-\eta)\exp(z) + \eta,$$
$$\xi(z) = \log\frac{z-\eta}{1-\eta}.$$

MadaBoost type:

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

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

The divergence of β-type has been employed for independent component analysis from the viewpoint of robustness, while the divergence of



(a) (b)

Figure 2: Examples of U-functions. (a) Shapes of U-functions. (b) Derivatives of U-functions, u.

η-type is shown to improve AdaBoost in the case of mislabeling (see Minami & Eguchi, 2002; Takenouchi & Eguchi, 2004). Note that β-divergence and η-divergence coincide with KL divergence when $\beta \to 0$ and $\eta \to 0$, respectively. There are several proposals for statistical divergence, such as f-divergence and α-divergence (Amari, 1985). The class of f-divergence is totally different from that of U-divergence except for the KL divergence. See Figure 3 for a schematic map. Also note that U-entropy H_U is a generalization of the Shannon entropy. For example, the Tsallis entropy is derived from the β-type U-function (Amari & Nagaoka, 2000).

3.2 Pythagorean Relation and Orthogonal Foliation. Let us define the inner product of functions of $\boldsymbol{x} \in \mathcal{X}$ and $y \in \mathcal{Y}$ under $\mu(\boldsymbol{x})$ by

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

and define that f and g are orthogonal if $\langle f, g \rangle_\mu = 0$. Then the Pythagorean relation for the Bregman divergence is stated as follows (see Figure 4):

Theorem 1 (Pythagorean relation). *Let p, q, and r be in $\mathcal{M}$. If $p - q$ and $\xi(r) - \xi(q)$ are orthogonal under μ, the relation*

$$D_U(p, r; \mu) = D_U(p, q; \mu) + D_U(q, r; \mu) \tag{3.10}$$

holds.

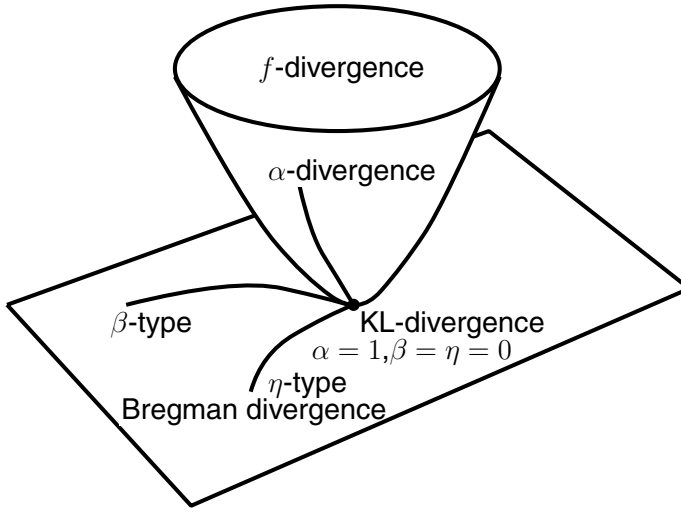


Figure 3: Schematic picture of statistical divergences.

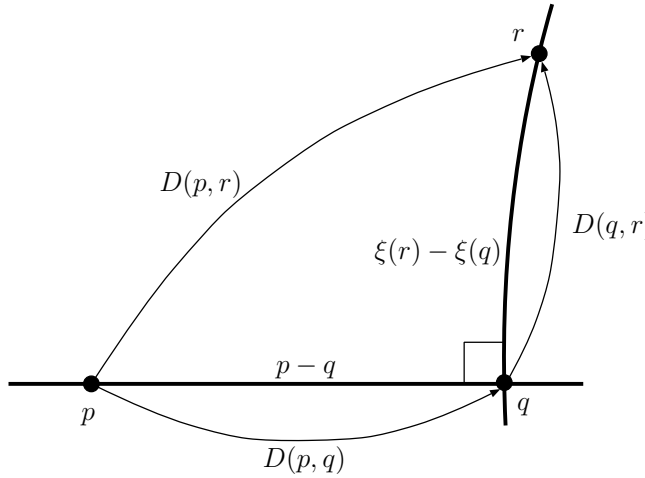


Figure 4: Pythagorean relation for Bregman divergence.

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

$$\begin{aligned} & D_U(p, r; \mu) - D_U(p, q; \mu) - D_U(q, r; \mu) \\ &= \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} (q(y|\boldsymbol{x}) - p(y|\boldsymbol{x}))(\xi(r(y|\boldsymbol{x})) - \xi(q(y|\boldsymbol{x})))\mu(\boldsymbol{x})d\boldsymbol{x} \\ &= \langle p - q, \xi(q) - \xi(r) \rangle_\mu \end{aligned} \tag{3.11}$$

holds by definition. From the orthogonality of $p - q$ and $\xi(r) - \xi(q)$, the right-hand side of equation 3.11 vanishes, and it proves the relation.

A special case of theorem 1 associated with the KL divergence is given in Amari (1985). Note that in theorem 1, the orthogonality is defined between $p - q$ and $\xi(r) - \xi(q)$. The form $\xi(q)$ is rewritten as

$$q(y|\boldsymbol{x}) = u(\xi(q(y|\boldsymbol{x}))),$$

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

Now we consider subspaces feasible for the nature of the Bregman divergence.

Definition 2 (U-flat subspace). *With a fixed $q_0 \in \mathcal{M}$ and a set of decision functions $\mathcal{F} = \{f_\lambda(\boldsymbol{x}, y); \lambda \in \Lambda\}$, where Λ is a certain finite set of indices, a set of conditional measures written in the form of*

$$\mathcal{Q}_U(q_0, \mathcal{F}) = \left\{ q \in \mathcal{M} \;\middle|\; q(y|\boldsymbol{x}) = u\left(\xi(q_0(y|\boldsymbol{x})) + \sum_{\lambda \in \Lambda} \alpha_\lambda f_\lambda(\boldsymbol{x}, y) \right),\ \alpha_\lambda \in R \right\}, \quad (3.12)$$

is called a U-flat subspace.

In other words, $\mathcal{Q}_U$ consists of functions such that

$$\xi(q) - \xi(q_0) = \sum_{\lambda \in \Lambda} \alpha_\lambda f_\lambda(\boldsymbol{x}, y),$$

which means $\mathcal{Q}_U$ is a subspace in $\mathcal{M}$ including q_0 and spanned by $\mathcal{F}$. Therefore, the dimension number of the subspace $\mathcal{Q}_U$ is $|\Lambda|$, the cardinality of Λ, in the space $\mathcal{M}$. We remark that $\mathcal{F}$ can be not only decision functions but also arbitrary functions on $\mathcal{X} \times \mathcal{Y}$ in general. The flat structure and geometrical properties discussed later hold for those general functions.

Because of the linear structure of U-representation, an important property of $\mathcal{Q}_U$ is introduced. Let q_1 and q_2 be in $\mathcal{Q}_U$; then for any numbers β_1 and β_2, any linear combination of U-representation of q_1 and q_2 belongs to $\mathcal{Q}_U$ again; that is,

$$u\left(\xi(q_0) + \beta_1(\xi(q_1) - \xi(q_0)) + \beta_2(\xi(q_2) - \xi(q_0))\right) \in \mathcal{Q}_U,$$

holds; therefore, we call $\mathcal{Q}_U(q_0, \mathcal{F})$ a U-flat subspace.

In this relation, ξ plays the same role with the logarithm for the exponential family in statistics, which is called an e-flat subspace in information geometry.

Next let us consider another flat subspace in $\mathcal{M}$.

Definition 3 (*m*-flat subspace). *A subspace in $\mathcal{M}$ that passes a point $p_0 \in \mathcal{M}$ and orthogonal to $\mathcal{Q}_U$ defined by*

$$\mathcal{T}(p_0, \mu, \mathcal{F}) = \left\{ p \in \mathcal{M} \;\middle|\; \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} (p(y|\boldsymbol{x}) - p_0(y|\boldsymbol{x})) f_\lambda(\boldsymbol{x}, y) \mu(\boldsymbol{x}) d\boldsymbol{x} = 0, \ \forall \lambda \right\}$$
$$= \{ p \in \mathcal{M} \mid \langle p - p_0, f_\lambda \rangle_\mu = 0, \ \forall \lambda \} \tag{3.13}$$

is called an m-flat subspace.

In the following, we omit p_0, μ, and $\mathcal{F}$ from $\mathcal{T}$ depending on the context. An important property of $\mathcal{T}$ is also its flat structure. For any p_1 and p_2 in $\mathcal{T}(p_0)$, and for any positive numbers β_1 and β_2, we observe

$$p_0 + \beta_1(p_1 - p_0) + \beta_2(p_2 - p_0) \in \mathcal{T}(p_0),$$

because of the linearity of the inner product,

$$\beta_1 \langle p_1 - p_0, f_\lambda \rangle_\mu + \beta_2 \langle p_2 - p_0, f_\lambda \rangle_\mu = 0, \ \forall \lambda.$$

Hence, its structure is called mixture flat (*m*-flat). Note that the codimension number of the subspace $\mathcal{T}$ is $|\Lambda|$ in the space $\mathcal{M}$ by its definition.

As a special case, let us take $q^* \in \mathcal{Q}_U$ and consider an *m*-flat subspace $\mathcal{T}(q^*)$ at $q^* \in \mathcal{Q}_U$. By the above definitions, $\mathcal{Q}_U$ and $\mathcal{T}(q^*)$ are orthogonal at q^*, and for any $p \in \mathcal{T}(q^*)$ and $q \in \mathcal{Q}_U$,

$$\langle p - q^*, \xi(q) - \xi(q^*) \rangle_\mu = 0, \ \forall p \in \mathcal{T}(q^*), \ \forall q \in \mathcal{Q}_U$$

holds (see Figure 5). A set of *m*-flat subspaces at all the points in $\mathcal{Q}_U$, $\{\mathcal{T}(q); q \in \mathcal{Q}_U\}$, covers the whole space $\mathcal{M}$ as

$$\bigcup_{q \in \mathcal{Q}_U} \mathcal{T}(q) = \mathcal{M},$$
$$\mathcal{T}(q) \cap \mathcal{T}(q') = \phi, \ \text{if } q \neq q'.$$

The set $\{\mathcal{T}(q); q \in \mathcal{Q}_U\}$ is called an orthogonal foliation of $\mathcal{Q}_U$ in $\mathcal{M}$, and each subspace $\mathcal{T}(q)$ is called a leaf—that is, $\mathcal{M}$ is decomposed into leaves orthogonal to $\mathcal{Q}_U$ (see Figure 6).

Based on the notion of the orthogonal foliation, we can prove the following theorem (see Figure 7).

Theorem 2. *Two optimization problems,*

$$\begin{aligned} &\textit{minimize } D_U(p, q_0; \mu) \\ &\textit{with respect to } p \in \mathcal{T}(p_0) \textit{ for fixed } q_0, \end{aligned} \tag{3.14}$$

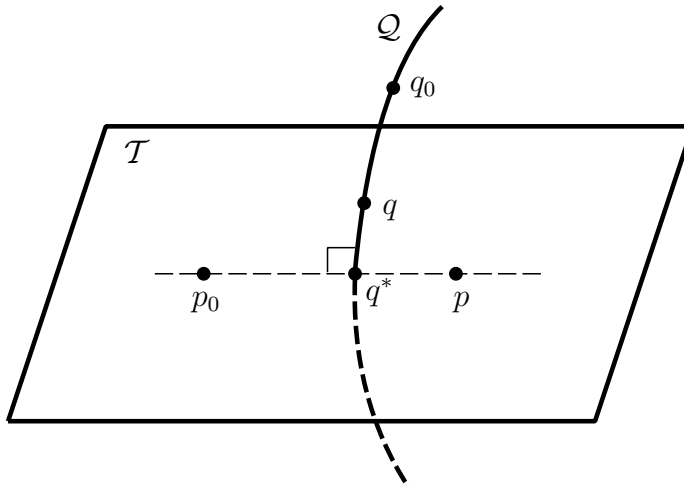


Figure 5: Geometrical structure of $\mathcal{Q}$ and $\mathcal{T}$.

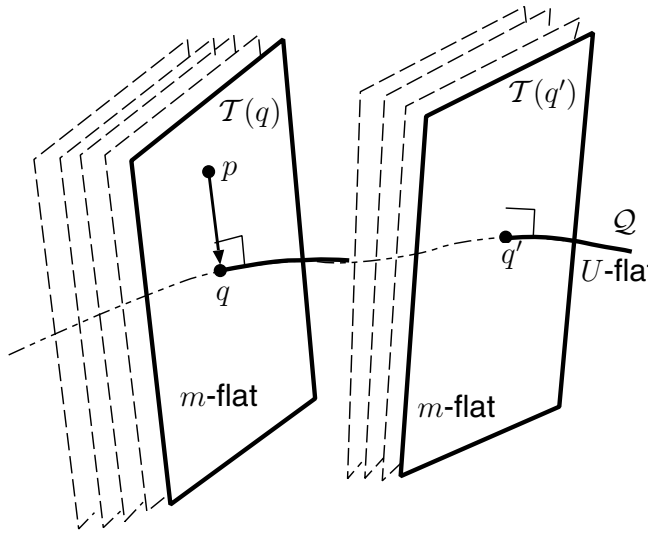


Figure 6: Orthogonal foliation derived from the Bregman divergence.

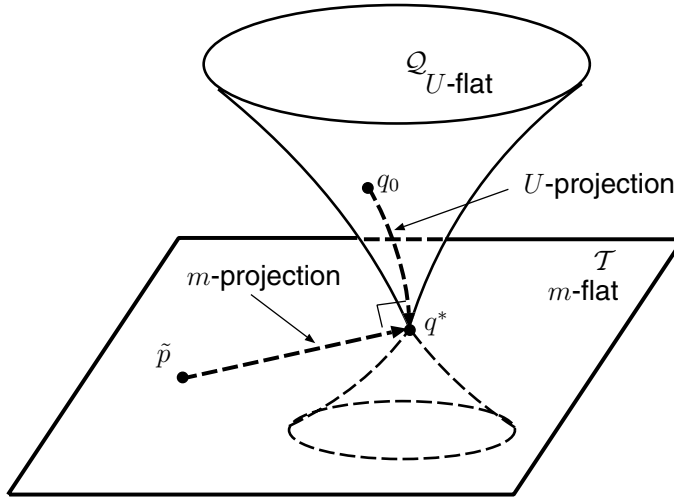


Figure 7: A geometrical interpretation of the dual optimization problems in the U-Boost algorithm.

and

$$\begin{array}{l} \textit{minimize } D_U(p_0, q; \mu) \\ \textit{with respect to } q \in \mathcal{Q}_U(q_0) \textit{ for fixed } p_0, \end{array} \tag{3.15}$$

give the same solution:

$$q^* = \operatorname*{argmin}_{p \in \mathcal{T}} D_U(p, q_0; \mu) = \operatorname*{argmin}_{q \in \mathcal{Q}_U} D_U(p_0, q; \mu), \tag{3.16}$$

which is the intersection of $\mathcal{Q}_U$ and $\mathcal{T}$.

Proof. First, we should note that two subspaces $\mathcal{Q}_U(q_0)$ and $\mathcal{T}(p_0)$ intersect each other at one point q^*,

$$\{q^*\} = \mathcal{T}(p_0) \cap \mathcal{Q}_U(q_0),$$

because of the relationship between $\dim \mathcal{Q}_U$ and $\operatorname{codim} \mathcal{T}$. From the Pythagorean relation of U-divergence, for any $p \in \mathcal{T}(p_0)$ and $q \in \mathcal{Q}_U(q_0)$,

$$D_U(p, q) = D_U(p, q^*) + D_U(q^*, q) \tag{3.17}$$

holds. As a consequence,

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

holds; therefore, we observe

$$D_U(p, q_0) \geq D_U(q^*, q_0), \text{ for any } p \in \mathcal{T},$$

because of the positivity of U-divergence, $D_U(p, q^*) \geq 0$. This means the point $q^* \in \mathcal{T}$ is the closest to q_0 in $\mathcal{T}$. And in the same way, we observe

$$D_U(p_0, q) \geq D_U(p_0, q^*), \text{ for any } q \in \mathcal{Q}_U,$$

and it shows q^* is the closest to p_0 in $\mathcal{Q}_U$. These prove the statement of the theorem.

As we discuss later, the one-dimensional U-flat subspace from q_0 to q_* is orthogonal to the m-flat subspace $\mathcal{T}(p_0)$, and hence q^* is said to be the U-projection of q_0 to $\mathcal{T}(p_0)$. Simultaneously q^* is the m-projection of p_0 to $\mathcal{Q}$ (cf. Amari & Nagaoka, 2000).

3.3 U-models for Classification. Our purpose of the classification problem is to estimate an element in a certain set of conditional measures, which minimizes U-divergence from the empirical distribution constructed from given examples,

$$q^* = \operatorname*{argmin}_q H_U(\tilde{p}, q; \tilde{\mu}).$$

In this section, we consider a special subspace, U-model, produced from a set of decision functions $\mathcal{F} = \{f_t(\boldsymbol{x}, y); t = 1, \ldots, T\}$ in the form

$$\mathcal{Q}_U(q_0, \mathcal{F}; b) = \left\{ q \in \mathcal{M} \;\middle|\; q(y|\boldsymbol{x}) = u\left(\xi(q_0(y|\boldsymbol{x})) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y) - b(\boldsymbol{x}, \alpha)\right)\right\}, \quad (3.18)$$

where $\alpha = (\alpha_t \in R; t = 1, \ldots, T)$ and b is an auxiliary function of $\boldsymbol{x}$ and α, which imposes an additional condition on $\mathcal{Q}_U$. Note that in the following, α is used as a parameter for specifying an element in $\mathcal{Q}_U$. Intuitively speaking, $\mathcal{Q}_U$ is spanned by a basis $\mathcal{F}$ in terms of U-representation, and b is introduced to overcome the inconvenience caused by the fact that the basis is restricted within decision functions.

For a conditional measure q in this subspace $\mathcal{Q}_U$, the empirical U-cross-entropy is simply written as

$$L_U^{\text{emp}}(q) = H_U(\tilde{p}, q; \tilde{\mu}) = \frac{1}{n} \sum_{i=1}^{n} \left[\sum_{y \in \mathcal{Y}} U(\xi(q(y|\boldsymbol{x}_i)) - \xi(q(y_i|\boldsymbol{x}_i)) \right],$$

where

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

which we call the empirical U-loss, or simply U-loss.

Taking account of the classification rule invariance discussed in the next section, function b for constraint on the U-model must be determined based on computational convenience or statistical validity. From a statistical point of view, we consider following two specific cases, an empirical model and a normalized model, in the following sections.

3.3.1 Invariance of Classification Rule. Let us use U-representation for denoting a conditional measure as

$$\xi(q(y|\boldsymbol{x})) \text{ equivalently } q(y|\boldsymbol{x}) = u(\xi(q(y|\boldsymbol{x}))),$$

and $\xi(q)$ is referred to later as a discriminate function. For the classification task, we use Bayes rule, where the estimate of the corresponding label for a given input $\boldsymbol{x}$ is given by the maximum value of $q(y|\boldsymbol{x})$, which is realized by using the discriminate function as

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

because ξ is monotonic. There are two equivalent conditions for discriminate functions for Bayes rule.

Shift invariance. Let $b(\boldsymbol{x})$ be an arbitrary function of $\boldsymbol{x}$. The classification rule associated with $u(\xi(q) - b)$ is equivalent to that with q,

$$\operatorname*{argmax}_{y \in \mathcal{Y}} \xi(q(y|\boldsymbol{x})) = \operatorname*{argmax}_{y \in \mathcal{Y}} \{\xi(q(y|\boldsymbol{x})) - b(\boldsymbol{x})\}.$$

Scale invariance. Let $c(\boldsymbol{x})$ be an arbitrary positive function of $\boldsymbol{x}$. The classification rule associated with $c(\boldsymbol{x})q(y|\boldsymbol{x})$ is equivalent to that with $q(y|\boldsymbol{x})$,

$$\operatorname*{argmax}_{y \in \mathcal{Y}} \xi(c(\boldsymbol{x})q(y|\boldsymbol{x})) = \operatorname*{argmax}_{y \in \mathcal{Y}} \xi(q(y|\boldsymbol{x})),$$

because of the monotonicity of ξ.

These invariant properties are closely related to obtaining a probability density $p \in \mathcal{P}$ from a measure $m \in \mathcal{M}$ by U-projection and m-projection, respectively.

In the next two sections, we focus on the quantity defined by

$$\begin{aligned}\Delta(q, q^*) &= D_U(p, q) - D_U(p, q^*) - D_U(q^*, q) \\ &= \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} (\xi(q(y|\boldsymbol{x})) - \xi(q^*(y|\boldsymbol{x})))(q^*(y|\boldsymbol{x}) - p(y|\boldsymbol{x}))\mu(\boldsymbol{x})d\boldsymbol{x} \\ &= \langle \xi(q) - \xi(q^*), q^* - p \rangle_\mu, \end{aligned} \tag{3.19}$$

for a fixed conditional density $p(y|\boldsymbol{x}) \in \mathcal{P}$ and a density $\mu(\boldsymbol{x})$. By using the above invariance, we consider the conditions where $\Delta(q, q^*)$ vanishes in order to utilize the Pythagorean relation.

3.3.2 Empirical U-Model. Suppose $q^*(y|\boldsymbol{x}) = c(\boldsymbol{x})p(y|\boldsymbol{x})$ or, equivalently, $\xi(q^*(y|\boldsymbol{x})) = \xi(c(\boldsymbol{x})p(y|\boldsymbol{x}))$, which implies that the classification rule associated with $\xi(q^*)$ is equivalent to the Bayes rule for p by scale invariance. Then for any measure q, we observe

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

Let $\mathcal{D}_m$ be a set of discriminate functions with the same conditional expectation under p,

$$\mathcal{D}_m = \left\{ q \in \mathcal{M} \;\middle|\; \sum_{y \in \mathcal{Y}} \xi(q(y|\boldsymbol{x}))p(y|\boldsymbol{x}) = \sum_{y \in \mathcal{Y}} \xi(q^*(y|\boldsymbol{x}))p(y|\boldsymbol{x}) \text{ (a.e. } \boldsymbol{x}) \right\}.$$

Then $\Delta(q, q^*) = 0$ for any $q \in \mathcal{D}_m$ and hence the Pythagorean relation

$$D_U(p, q) = D_U(p, q^*) + D_U(q^*, q)$$

holds. This means that q^* gives the minimum of $D_U(p, q)$ among $q \in \mathcal{D}_m$,

$$q^* = \operatorname*{argmin}_{q \in \mathcal{D}_m} D_U(p, q).$$

In other words, D_U chooses the Bayes optimal rule for p in $\mathcal{D}_m$, which consists of discriminate functions of the same conditional expectation.

From the shift invariant property, $\xi(q) - b$ gives the same classification rule with $\xi(q)$; therefore, by adopting an appropriate b, we can introduce a simple constraint on $\mathcal{D}_m$ as

$$\mathcal{D}_m = \left\{ q \in \mathcal{M} \;\middle|\; \sum_{y \in \mathcal{Y}} \xi(q(y|\boldsymbol{x}))p(y|\boldsymbol{x}) = \sum_{y \in \mathcal{Y}} \xi(q_0(y|\boldsymbol{x}))p(y|\boldsymbol{x}) \right\},$$

where

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

In this case, b is simply given by

$$b(\boldsymbol{x}, \alpha) = \sum_{y \in \mathcal{Y}} \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y)p(y|\boldsymbol{x}).$$

Under empirical distributions $\tilde{\mu}$ and $\tilde{p}$, the above condition for $\mathcal{D}_m$ is satisfied by taking

$$b(\boldsymbol{x}, \alpha) = \sum_{t=1}^{T} \alpha_t \tilde{f}_t(\boldsymbol{x}),$$

where $\tilde{f}$ is the conditional expectation under $\tilde{p}$:

$$\tilde{f}_t(\boldsymbol{x}) = \sum_{y\in\mathcal{Y}} \tilde{p}(y|\boldsymbol{x}) f_t(\boldsymbol{x}, y)$$
$$= \begin{cases} f_t(\boldsymbol{x}_i, y_i), & \text{if } \boldsymbol{x} = \boldsymbol{x}_i, \\ \frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}} f_t(\boldsymbol{x}, y), & \text{otherwise.} \end{cases}$$

Then we define the empirical U-model depending on the empirical conditional density $\tilde{p}$ by

$$\mathcal{Q}_U^{\mathrm{emp}}(q_0, \mathcal{F})$$
$$= \left\{ q \in \mathcal{M} \;\middle|\; \xi(q(y|\boldsymbol{x})) = \xi(q_0(y|\boldsymbol{x})) + \sum_{t=1}^{T} \alpha_t (f_t(\boldsymbol{x}, y) - \tilde{f}_t(\boldsymbol{x})) \right\}. \quad (3.20)$$

Note that the empirical U-model depends on $\tilde{p}$ and has a U-flat structure by its definition,

$$\mathcal{Q}_U^{\mathrm{emp}} = \mathcal{Q}_U(q_0, \{f_t - \tilde{f}_t, t = 1, \ldots, T\}).$$

Hence, the geometrical properties discussed in the previous section hold between $\mathcal{Q}_U^{\mathrm{emp}}$ and its orthogonal m-flat subspaces,

$$\mathcal{T}(q) = \left\{ p \in \mathcal{M} \;\middle|\; \langle p - q, f_t - \tilde{f}_t \rangle_{\tilde{\mu}} = 0, \ \forall t \right\}, \ q \in \mathcal{Q}_U^{\mathrm{emp}}, \quad (3.21)$$

under the empirical distribution $\tilde{\mu}(\boldsymbol{x})$.

For the empirical U-model, the U-loss for $q \in \mathcal{Q}_U^{\mathrm{emp}}$ under $\tilde{p}$ and $\tilde{\mu}$ is written as

$$L_U(q) = \frac{1}{n}\sum_{i=1}^{n} \left[\sum_{y\in\mathcal{Y}} U\left(\xi(q_0(y|\boldsymbol{x}_i)) + \sum_{t=1}^{T} \alpha_t (f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i)) \right) \right.$$
$$\left. - \xi(q_0(y_i|\boldsymbol{x}_i)) \right]. \quad (3.22)$$

As a special case that $\xi(q_0) = 0$, $L_U(q)$ depends on only the combined decision function F,

$$F(\boldsymbol{x}, y) = \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y),$$

and in this case, as stated in section 2, the U-loss is denoted by

$$L_U^{\mathrm{emp}}(F) = \frac{1}{n}\sum_{i=1}^{n}\sum_{y\in\mathcal{Y}} U(F(\boldsymbol{x}_i, y) - F(\boldsymbol{x}_i, y_i)). \quad (3.23)$$

3.3.3 Normalized U-Model. Next let us consider a set of discriminate functions, which are shifted from a fixed function $\xi(q_0(y|\boldsymbol{x}))$ by arbitrary functions $b(\boldsymbol{x})$,

$$\mathcal{D}_s = \{q \in \mathcal{M} \mid \xi(q(y|\boldsymbol{x})) = \xi(q_0(y|\boldsymbol{x})) - b(\boldsymbol{x})\}.$$

By shift invariance, the classification rule associated with any $q \in \mathcal{D}_s$ is equivalent to that with q_0. Also, for any $\xi(q) = \xi(q_0) - b$ and $\xi(q^*) = \xi(q_0) - b^*$,

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

holds. Therefore, if $q^* \in \mathcal{P}$, that is, $\sum_{y \in \mathcal{Y}} q^*(y|\boldsymbol{x}) = 1$ (a.e. $\boldsymbol{x}$), then $\Delta = 0$ and the Pythagorean relation,

$$D_U(p, q) = D_U(p, q^*) + D_U(q^*, q),$$

holds. This means that q^* is the closest point from p in $\mathcal{D}_s$, all the elements in which give the same classification rule:

$$q^* = \operatorname*{argmin}_{q \in \mathcal{D}_s} D_U(p, q).$$

As a consequence, the minimization $D_U(p, q)$ in $\mathcal{D}_s$ is equivalent to introducing the normalizing factor, so that

$$\sum_{y \in \mathcal{Y}} q^*(y|\boldsymbol{x}) = 1.$$

Namely, q^* is restricted within a set of conditional probability densities $\mathcal{P}$.

We define the normalized U-model by

$$\mathcal{Q}_U^{\mathrm{norm}}(q_0, \mathcal{F}) = \left\{ q \in \mathcal{P} \;\middle|\; \xi(q(y|\boldsymbol{x})) = \xi(q_0(y|\boldsymbol{x})) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y) - \phi(\boldsymbol{x}, \alpha) \right\}, \quad (3.24)$$

where ϕ is the normalization term chosen so that $\sum_{y \in \mathcal{Y}} q(y|\boldsymbol{x}) = 1$ is satisfied, which might depend on q_0 and $\mathcal{F}$ implicitly.

Note that because of the normalization term ϕ, $\mathcal{Q}_U^{\mathrm{norm}}$ is not a U-flat subspace in $\mathcal{M}$. However, the Pythagorean relation for the normalized U-model holds by restricting within $\mathcal{P}$ as follows.

First, let us define the orthogonal m-flat subspace in $\mathcal{M}$ at $q_\alpha \in \mathcal{Q}_U^{\mathrm{norm}}$ by

$$\mathcal{T}(q_\alpha) = \{p \in \mathcal{M} \mid \langle p - q_\alpha, f_t - \phi_t'(\alpha) \rangle = 0, \ \forall t\},$$

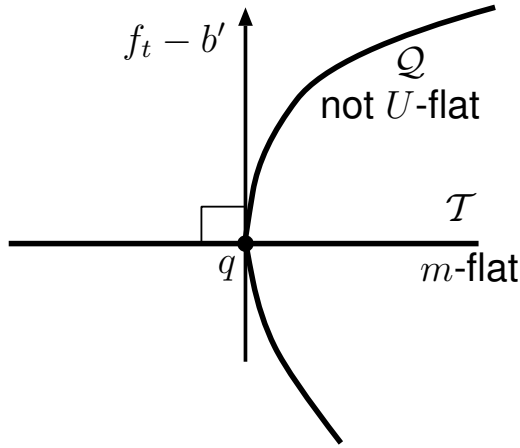


Figure 8: Orthogonal condition of the normalized U-model.

where the subscript α of q_α is a parameter for specifying an element in $\mathcal{Q}_U$ and

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

In this case, the orthogonality of $\mathcal{Q}_U^{\mathrm{norm}}$ and $\mathcal{T}(q_\alpha)$ is locally defined with the tangent of $\mathcal{Q}_U^{\mathrm{norm}}$ at q_α (see Figure 8):

$$\left\langle p - q_\alpha, \frac{\partial}{\partial \alpha_t} \xi(q_\alpha) \right\rangle_\mu = \langle p - q_\alpha, f_t - \phi_t'(\alpha) \rangle_\mu = 0.$$

Knowing that

$$\sum_{y \in \mathcal{Y}} (p(y|\boldsymbol{x}) - q(y|\boldsymbol{x})) = 1 - 1 = 0, \ \forall p, q \in \mathcal{P},$$

and for any function g that does not depend on y,

$$\langle p - q, g \rangle_\mu = \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} (p(y|\boldsymbol{x}) - q(y|\boldsymbol{x})) g(\boldsymbol{x}) \mu(\boldsymbol{x}) d\boldsymbol{x} = 0, \ \forall p, q \in \mathcal{P},$$

we observe that for $p \in \mathcal{P}$,

$$\langle p - q_\alpha, f_t - \phi_t'(\alpha) \rangle_\mu = \langle p - q_\alpha, f_t \rangle_\mu \ \forall t,$$

and hence the orthogonal m-flat subspace in $\mathcal{P}$ is written as

$$\mathcal{T}(q) \cap \mathcal{P} = \{ p \in \mathcal{P} \mid \langle p - q, f_t \rangle_\mu = 0, \ \forall t \}.$$

Let $q, r \in \mathcal{Q}_U^{\mathrm{norm}}$ be

$$\xi(q) = \xi(q_0) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}, y) - \phi(\boldsymbol{x}, \alpha),$$
$$\xi(r) = \xi(q_0) + \sum_{t=1}^{T} \alpha'_t f_t(\boldsymbol{x}, y) - \phi(\boldsymbol{x}, \alpha').$$

Then for any $p \in \mathcal{T}(q) \cap \mathcal{P}$, we observe

$$\langle p - q, \xi(q) - \xi(r) \rangle_\mu = \langle p - q, \sum_{t=1}^{T} (\alpha'_t - \alpha_t) f_t - (\phi(\alpha) - \phi(\alpha')) \rangle_\mu = 0.$$

Therefore, the Pythagorean relation

$$D_U(p, r) = D_U(p, q) + D_U(q, r)$$

holds for the normalized U-model in $\mathcal{P}$.

In this case, the empirical U-loss is led to

$$L_U(q) = \frac{1}{n} \sum_{i=1}^{n} \left[\sum_{y \in \mathcal{Y}} U\left(\xi(q_0(y|\boldsymbol{x}_i)) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}_i, y) - \phi(\boldsymbol{x}_i, \alpha) \right) - \left\{ \xi(q_0(y_i|\boldsymbol{x}_i)) + \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x}_i, y_i) - \phi(\boldsymbol{x}_i, \alpha) \right\} \right]. \quad (3.25)$$

In the case of $\xi(q_0) = 0$, $L_U(q)$ is written in terms of F and ϕ, and ϕ is determined from F. Therefore, we denote it by

$$L_U^{\mathrm{norm}}(F) = \frac{1}{n} \sum_{i=1}^{n} \left[\sum_{y \in \mathcal{Y}} U(F(\boldsymbol{x}_i, y) - \phi(\boldsymbol{x}_i, \alpha)) - \{F(\boldsymbol{x}_i, y_i) - \phi(\boldsymbol{x}_i, \alpha)\} \right]. \quad (3.26)$$

3.4 Geometrical Understanding of U-Boost. In the following, we treat the algorithm for constructing the combined decision function sequentially.

Here we give a unified description of the U-Boost algorithm from a geometrical point of view. In the following, we consider a one-dimensional expansion of U-model given by

$$Q_U(q_{t-1}, f_t; b_t) = \{q \mid \xi(q(y|\boldsymbol{x})) = \xi(q_{t-1}(y|\boldsymbol{x})) + \alpha f_t(\boldsymbol{x}, y) - b_t(\boldsymbol{x}, \alpha),\ \alpha \in R\}, \quad (3.27)$$

where b_t is the auxiliary function for the U-model,

$$b_t(\boldsymbol{x}, \alpha) = \begin{cases} \alpha \tilde{f}_t(\boldsymbol{x}), & \text{(empirical model)}, \\ \phi_t(\boldsymbol{x}, \alpha), & \text{(normalized model)}, \end{cases} \tag{3.28}$$

and its derivative at $\alpha = 0$,

$$b'_t(\boldsymbol{x}) = \begin{cases} \tilde{f}_t(\boldsymbol{x}), & \text{(empirical model)}, \\ \phi'_t(\boldsymbol{x}, \alpha)|_{\alpha=0}, & \text{(normalized model)}. \end{cases} \tag{3.29}$$

Geometrical Description of U-Boost Algorithm.

Input: Given n examples $\{(\boldsymbol{x}_i, y_i); \boldsymbol{x}_i \in \mathcal{X}, y \in \mathcal{Y}, i = 1, \ldots, n\}$.

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

Do for $t = 1, \ldots, T$

Step 1: Select a classifier h_t based on the corresponding decision function f_t so that $f_t - b'_t$ points in the same direction of $q_{t-1} - \tilde{p}$ as much as possible, that is,

$$\text{maximize } \langle q_{t-1} - \tilde{p}, f_t - b'_t \rangle_{\tilde{\mu}}.$$

Step 2: Construct a one-dimensional U-model $\mathcal{Q}_t = Q_U(q_{t-1}, f_t; b_t)$ by equation 3.27. Then find α_t, which minimize $D_U(\tilde{p}, q; \tilde{\mu})$, namely,

$$\begin{aligned} \alpha_t &= \operatorname*{argmin}_{\alpha} L_U(q) \\ &= \operatorname*{argmin}_{q \in \mathcal{Q}_t} \sum_{i=1}^{n} \left[\sum_{y \in \mathcal{Y}} U(\xi(q(y|\boldsymbol{x}_i))) - \xi(q(y_i|\boldsymbol{x}_i)) \right]. \end{aligned}$$

Step 3: Update

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

equivalently,

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

Output: The final decision by the majority vote as

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

A geometrical interpretation of the algorithm is shown in Figure 9.

In step 1, a strategy for selecting a decision function is described. Since f_t is a base of the next one-dimensional U-model that originates from the

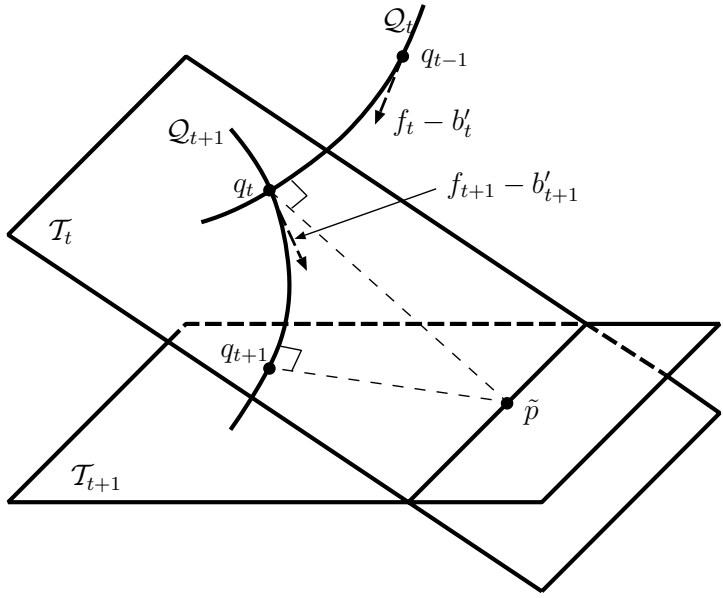


Figure 9: Geometrical interpretation of the optimization procedure of the U-Boost algorithm.

current estimate q_{t-1}, the tangent $f_t - b'_t$ should parallel $\tilde{p}$ as much as possible for the next estimate q_t to come close to $\tilde{p}$. For the empirical U-model, $\langle \tilde{p} - q_{t-1}, f_t - b'_t \rangle_{\tilde{\mu}}$ is rewritten as

$$\langle \tilde{p} - q_{t-1}, f_t - \tilde{f}_t \rangle_{\tilde{\mu}} = -\langle q_{t-1}, f_t - \tilde{f}_t \rangle_{\tilde{\mu}},$$

and for the normalized U-model,

$$\begin{aligned} \langle \tilde{p} - q_{t-1}, f_t - \phi_t \rangle_{\tilde{\mu}} &= \langle \tilde{p} - q_{t-1}, f_t \rangle_{\tilde{\mu}} \\ &= \tilde{f}_t - \langle q_{t-1}, f_t \rangle_{\tilde{\mu}} \quad (\text{because } \tilde{f}_t = \langle q_{t-1}, \tilde{f}_t \rangle_{\tilde{\mu}}) \\ &= -\langle q_{t-1}, f_t - \tilde{f}_t \rangle_{\tilde{\mu}}. \end{aligned}$$

Therefore, for both models, step 1 is equivalent to

$$\text{minimize } \langle q_{t-1}, f_t - \tilde{f}_t \rangle_{\tilde{\mu}}. \tag{3.30}$$

This translation is related to the error rate property discussed later.

The optimization procedure for obtaining q_t in step 2 is geometrically interpreted as m-projection from $\tilde{p}$ onto $\mathcal{Q}_t$ as shown in Figure 10. For a U-model $\mathcal{Q}_t$, we can consider an orthogonal foliation $\{\mathcal{T}_t(q)\}$ as

$$\mathcal{T}_t(q) = \{p \in \mathcal{M} \text{ or } \mathcal{P} \mid \langle p - q, f_t - b'_t \rangle_{\tilde{\mu}} = 0\}, \ q \in \mathcal{Q}_t, \tag{3.31}$$

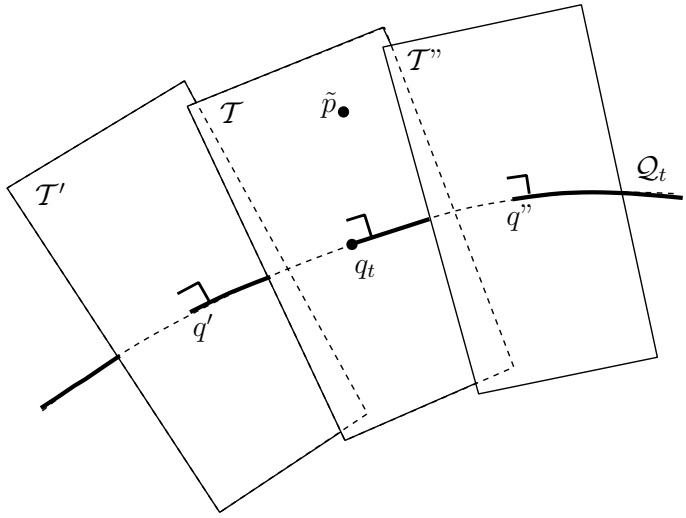


Figure 10: Relationship between the m-projection and the foliation of U-model in the U-Boost algorithm.

where the foliation is considered in $\mathcal{M}$ for the empirical model and in $\mathcal{P}$ for the normalized model. Then we can find a leaf $\mathcal{T}_t(q_*)$ that passes the empirical distribution $\tilde{p}$, and the optimal model is determined by $q_t = q_*$.

The concrete expression of step 2 is written as follows for each model:

Empirical U-model

$$\alpha_t = \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \sum_{y \neq y_i} U(F_{t-1}(\boldsymbol{x}_i, y) - F_{t-1}(\boldsymbol{x}_i, y_i) + \alpha\{f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i)\}), \tag{3.32}$$

Normalized U-model

$$\alpha_t = \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} \left[\sum_{y \in \mathcal{Y}} U(F_{t-1}(\boldsymbol{x}_i, y) + \alpha f_t(\boldsymbol{x}_i, y) - \phi_t(\boldsymbol{x}_i, \alpha)) - \{F_{t-1}(\boldsymbol{x}_i, y_i) + \alpha f_t(\boldsymbol{x}_i, y_i) - \phi_t(\boldsymbol{x}_i, \alpha)\} \right]. \tag{3.33}$$

The above sequential procedure from f_1 through f_T without recursion is not in general equivalent to the parallel procedure, which directly gives the optimal point q^* for given $\mathcal{F} = \{f_t; t = 1, \ldots, T\}$, that is, the intersection

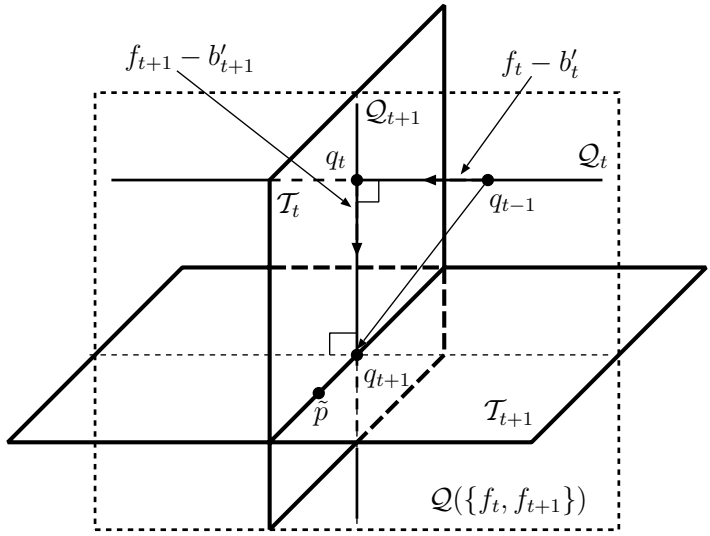


Figure 11: the sequential and parallel procedures coincide when U-models $\{\mathcal{Q}_t\}$ are orthogonal to each other.

of $\mathcal{Q}(\mathcal{F})$ and $\mathcal{T}(\mathcal{F})$. As intuitively shown in Figure 11, the sequential procedure without recursion coincides with the parallel procedure when the sequences of $\mathcal{Q}_t$'s are orthogonal to each other as a special case. However, if we use $\{f_t; t = 1, \ldots, T\}$ repeatedly, the following property suggests that the solution of the sequential update with recursion approaches the optimal solution of the parallel procedure. By simple calculation, we observe that for the sequence of densities $\{q_t; t = 0, 1, \ldots\}$ defined by the normalized or unnormalized U-Boost algorithm, the relation

$$L_U(q_{t+1}) - L_U(q_t) = -D_U(q_{t+1}, q_t) \tag{3.34}$$

holds. This shows that as far as q_t and q_{t+1} are different, namely, $D_U(q_t, q_{t+1}) > 0$, U-loss decreases monotonically. This property is closely related with that in the expectation-maximization (EM) algorithm for obtaining the maximum likelihood estimator. (See Amari, 1995, for the geometric considerations of the EM algorithm.)

3.4.1 Note on Binary Classification. In this section, we summarize some characteristics of binary classification problems, and in the following, we suppose the initial measure is simplified as $\xi(q_0) = 0$.

A classifier $h(\boldsymbol{x})$ outputs either $\{+1\}$ or $\{-1\}$ in the binary case; therefore, h can be regarded as a real-valued function, and the corresponding decision

function f is written as

$$f(\boldsymbol{x}, y) = \frac{yh(\boldsymbol{x}) + 1}{2}. \tag{3.35}$$

In this case, the combined decision function satisfies

$$F_t(\boldsymbol{x}_i, y) - F_t(\boldsymbol{x}_i, y_i) = \begin{cases} 0, & \text{if } y = y_i, \\ -y_i \sum_{k=1}^{t} \alpha_k h_k(\boldsymbol{x}_i), & \text{if } y \neq y_i. \end{cases} \tag{3.36}$$

The U-loss for the empirical model is written as

$$\begin{aligned} L_U(q) &= \frac{1}{n} \sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} U(F_{t-1}(\boldsymbol{x}_i, y) - F_{t-1}(\boldsymbol{x}_i, y_i) + \alpha(f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i))) \\ &= \frac{1}{n} \sum_{i=1}^{n} \left[U(0) + U\left(-y_i \left(\sum_{k=1}^{t-1} \alpha_k h_k(\boldsymbol{x}_i) + \alpha h_t(\boldsymbol{x}_i) \right) \right) \right]. \end{aligned}$$

Therefore, α_t in step 2 is given by

$$\begin{aligned} \alpha_t &= \operatorname*{argmin}_{\alpha} L_U(q) \\ &= \operatorname*{argmin}_{\alpha} \sum_{i=1}^{n} U\left(-y_i \left(\sum_{k=1}^{t-1} \alpha_k h_k(\boldsymbol{x}_i) + \alpha h_t(\boldsymbol{x}_i) \right) \right). \end{aligned} \tag{3.37}$$

In the case of the KL divergence, where $U(z) = \exp(z)$, this procedure is equivalent to AdaBoost:

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

(cf. Lebanon & Lafferty, 2001).

Also, for $U(z) = \exp(z)$, the constraint for the normalized model is described as

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

The normalization term ϕ_t is solved as

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

and α_t is given by

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

This representation is equivalent to the U-Boost for the empirical model with $U(z) = \log(1 + \exp(z))$, and this shows that the normalized U-Boost associated with $U(z) = \exp(z)$ conducts the same procedure of LogitBoost (Friedman et al., 2000). Also, we note that the above discussion is an example that the U-Boost algorithm with the empirical and normalized models with the same U-function results in different solutions.

4 Statistical Properties of U-Boost

4.1 Error Rate Property. One of the important characteristics of the AdaBoost algorithm is the evolution of its weighted error rates, that is, the classifier h_t at step t shows the worst performance, which is equivalent to random guess, under the distribution at the next step $t + 1$.

Let us define the weight at step $t + 1$ by

$$D_{t+1}(i, y) = \frac{q_t(y|\boldsymbol{x}_i)}{Z_{t+1}}, \tag{4.1}$$

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

$$Z_{t+1} = \sum_{i=1}^{n} \sum_{y\neq y_i} q_t(y|\boldsymbol{x}_i). \tag{4.2}$$

Then define the weighted error of the classifier h with its decision function f by

$$\epsilon_{t+1}(f) = \sum_{i=1}^{n} \sum_{y\neq y_i} \frac{f(\boldsymbol{x}_i, y) - f(\boldsymbol{x}_i, y_i) + 1}{2} D_{t+1}(i, y), \tag{4.3}$$

that is,

$$2\epsilon_{t+1}(f) - 1 = \sum_{i=1}^{n} \sum_{y \neq y_i} \big(f(\boldsymbol{x}_i, y) - f(\boldsymbol{x}_i, y_i)\big) D_{t+1}(i, y),$$
$$\propto \langle q_t, f - \tilde{f} \rangle_{\tilde{\mu}}, \tag{4.4}$$

where the meaning of the factor of the weight is interpreted by considering the following four cases:

$$\frac{f(\boldsymbol{x}_i, y) - f(\boldsymbol{x}_i, y_i) + 1}{2} = \begin{cases} 0, & \text{if } y_i \in h(\boldsymbol{x}_i) \text{ and } y \notin h(\boldsymbol{x}_i), \\ 1/2, & \text{if } y_i \in h(\boldsymbol{x}_i) \text{ and } y \in h(\boldsymbol{x}_i), \\ 1/2, & \text{if } y_i \notin h(\boldsymbol{x}_i) \text{ and } y \notin h(\boldsymbol{x}_i), \\ 1, & \text{if } y_i \notin h(\boldsymbol{x}_i) \text{ and } y \in h(\boldsymbol{x}_i). \end{cases} \tag{4.5}$$

Intuitively speaking, the first case is that 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 $\{y_i\}$:

$$\epsilon_{t+1}(h) = \sum_{\substack{1 \le i \le n \\ y_i \notin h(x_i) \\ y \in h(x_i)}} D_{t+1}(i, y) + \sum_{\substack{1 \le i \le n \\ y_i \notin h(x_i) \\ y \notin h(x_i) \\ y \neq y_i}} \frac{1}{2} D_{t+1}(i, y)$$
$$+ \sum_{\substack{1 \le i \le n \\ y_i \in h(x_i) \\ y \in h(x_i) \\ y \neq y_i}} \frac{1}{2} D_{t+1}(i, y). \tag{4.6}$$

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; hence, we omit $q_t(y_i|\boldsymbol{x}_i)$ from the weighted error, that is, the weights of labels that are different from given examples are defined only as in AdaBoost.M2 (Freund & Schapire, 1996). Also note that the correct rate is written as

$$1 - \epsilon_{t+1}(f) = \sum_{\substack{1 \le i \le n \\ y_i \in h(x_i) \\ y \notin h(x_i)}} D_{t+1}(i, y) + \sum_{\substack{1 \le i \le n \\ y_i \notin h(x_i) \\ y \notin h(x_i) \\ y \neq y_i}} \frac{1}{2} D_{t+1}(i, y)$$
$$+ \sum_{\substack{1 \le i \le n \\ y_i \in h(x_i) \\ y \in h(x_i) \\ y \neq y_i}} \frac{1}{2} D_{t+1}(i, y), \tag{4.7}$$

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

Then we can prove the following interesting property of the error rate of normalized and empirical U-Boost algorithms:

Theorem 3. *The U-Boost algorithm updates the distribution into the least favorable at each step, which means*

$$\epsilon_{t+1}(f_t) = \frac{1}{2}. \tag{4.8}$$

Proof. By differentiating the U-loss for the empirical model with respect to α, we know that α_t satisfies

$$\sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} (f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i)) u(\xi(q_{t-1}(y|\boldsymbol{x}_i)) + \alpha_t (f_t(\boldsymbol{x}_i, y) - f_t(\boldsymbol{x}_i, y_i))) = 0; \tag{4.9}$$

by using the definition

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

the above equation is rewritten as

$$\langle f_t - \tilde{f}_t, q_t \rangle_{\tilde{\mu}} = 0.$$

Similarly, by differentiating the U-loss for the normalized model, α_t satisfies

$$\begin{aligned} &- \sum_{i=1}^{n} (f_t(\boldsymbol{x}_i, y_i) - \phi_t'(\boldsymbol{x}_i, \alpha_t)) \\ &\quad + \sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} (f_t(\boldsymbol{x}_i, y) - \phi_t'(\boldsymbol{x}_i, \alpha_t)) u(\xi(q_{t-1}(y|\boldsymbol{x}_i)) \\ &\qquad + \alpha_t f_t(\boldsymbol{x}_i, y) - \phi_t(\boldsymbol{x}_i, \alpha_t)) = 0. \end{aligned} \tag{4.10}$$

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

$$\begin{aligned} &- \sum_{i=1}^{n} (f_t(\boldsymbol{x}_i, y_i) - \phi_t'(\boldsymbol{x}_i, \alpha_t)) + \sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} (f_t(\boldsymbol{x}_i, y) - \phi_t'(\boldsymbol{x}_i, \alpha_t)) q_t(y|\boldsymbol{x}_i) \\ &= -\langle f_t - \phi_t', \tilde{p} \rangle_{\tilde{\mu}} + \langle f_t - \phi_t', q_t \rangle_{\tilde{\mu}} \\ &= \langle f_t - \phi_t', q_t - \tilde{p} \rangle_{\tilde{\mu}} \end{aligned}$$

$$
\begin{aligned}
&= \langle f_t, q_t - \tilde{p} \rangle_{\tilde{\mu}} \qquad (\phi_t' \text{ does not depend on } y) \\
&= \langle f_t, q_t \rangle_{\tilde{\mu}} - \tilde{f}_t \\
&= \langle f_t - \tilde{f}_t, q_t \rangle_{\tilde{\mu}} \qquad (\langle 1, q_t \rangle_{\tilde{\mu}} = 1) \\
&= 0.
\end{aligned}
$$

Therefore, for both models, the optimal condition for q_t is written as the decision function and its conditional expectation as

$$\langle f_t - \tilde{f}_t, q_t \rangle_{\tilde{\mu}} = 0.$$

This condition is interpreted as

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

that is,

$$\sum_{\substack{1 \le i \le n \\ y_i \notin h_t(x_i) \\ y \in h_t(x_i)}} q_t(y|\boldsymbol{x}_i) = \sum_{\substack{1 \le i \le n \\ y_i \in h_t(x_i) \\ y \notin h_t(x_i)}} q_t(y|\boldsymbol{x}_i). \tag{4.11}$$

This means that the accumulated probability of correctly classified examples and wrongly classified examples is balanced under q_t. By imposing the above relation into error rate 4.6 and correct rate 4.7, we observe

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

which concludes

$$\epsilon_{t+1}(h_t) = \frac{1}{2}. \tag{4.12}$$

In this way, the U-Boost algorithm is constructed as updating the distribution into the least favorable for the present step. A set of decision functions defined by

$$\mathcal{R}(\tilde{p}, q_t) = \left\{ f \;\middle|\; \sum_{i=1}^{n} \sum_{y \in \mathcal{Y}} (f(\boldsymbol{x}_i, y) - f(\boldsymbol{x}_i, y_i)) q_t(y|\boldsymbol{x}_i) = 0 \right\} \tag{4.13}$$

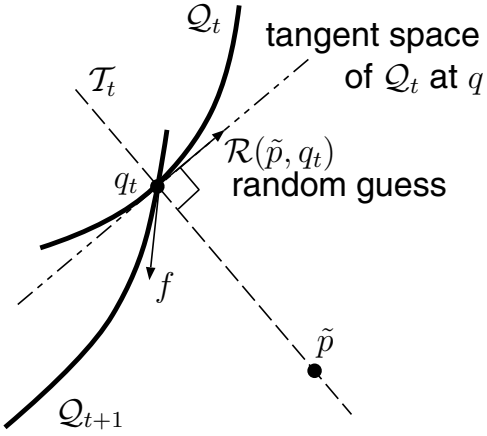


Figure 12: Relationship between "random guess" and the tangent space of $\mathcal{Q}$.

can be regarded as "random guess" associated with the weighted error. As seen in the above proof, the condition on $\mathcal{R}$ is equivalent to

$$\langle \tilde{p} - q_t, f - b'|_{\alpha=0} \rangle_{\tilde{\mu}} = 0, \tag{4.14}$$

and hence the set of "random guess," which is a set of the "worst" classifiers, is included in the tangent space of $\mathcal{Q}_t$ at q_t. Therefore, step 1 of the U-Boost claims that the next classifier must be chosen from outside of random guess (see Figure 12).

4.2 Consistency and Bayes Rule Equivalence. Using the basic property of the Bregman divergence, we can show the consistency of the U-loss as follows:

Theorem 4. *Let $p(y|\boldsymbol{x})$ be the true conditional distribution and $F(\boldsymbol{x}, y)$ be the minimizer of the U-loss $H_U(q)$ with $q = u(F)$. The classification rule given by F becomes Bayes optimal:*

$$\hat{y}(\boldsymbol{x}) = \operatorname*{argmin}_{y \in \mathcal{Y}} F(\boldsymbol{x}, y) = \operatorname*{argmin}_{y \in \mathcal{Y}} p(y|\boldsymbol{x}). \tag{4.15}$$

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 3.8, the minimizer of the U-cross-entropy $H_U(p, q; \mu)$ with respect to q, is given by

$$F(\boldsymbol{x}, y) = \underset{F}{\operatorname{argmin}} H_U(p, u_F) = \xi(p(y|\boldsymbol{x})).$$

The statement comes from the monotonicity of ξ:

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

In the U-Boost algorithm, $F(\boldsymbol{x}, y)$ is chosen from a class of functions that are a linear combination of $f_t(\boldsymbol{x}, y)$, $(t = 1, \ldots, T)$ with some bias function q_0 and b. In the case that the true distribution is not always in the considered U-model, which happens in practical cases, U-Boost cannot achieve Bayes optimality, and the closest point in the model is chosen in the sense of U-loss. If the number of functions T is sufficiently large and the functions $f_t; t = 1, \ldots, T$ are chosen from sufficiently various decision functions, the U-model can well approximate the true distribution. It depends on the richness of the decision functions, which are basis of discriminate function F. For a discussion about the richness of the linear combination of simple functions, see, for example, Barron (1993) and Murata (1996).

For the binary case in particular, we can show the following interesting relationship between the U-function and the log-likelihood ratio. In a binary classification problem, as shown in equation 3.37, the objective function of the U-loss is simplified as

$$\int_{\mathcal{X}} \sum_{y\in\{\pm 1\}} U(-yF(\boldsymbol{x}))p(y|\boldsymbol{x})\mu(\boldsymbol{x})d\boldsymbol{x}, \tag{4.16}$$

where q_t and F are linked as

$$q = u(yF(\boldsymbol{x})),$$

and the discriminate function F gives the classification rule as

$$y = \begin{cases} +1, & F(\boldsymbol{x}) > 0, \\ -1, & F(\boldsymbol{x}) < 0. \end{cases}$$

Theorem 5. *The minimizer of the U-cross-entropy gives the Bayes optimal rule, 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

$$\log \frac{u(z)}{u(-z)} = 2z \tag{4.17}$$

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 equation 4.16 satisfies

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

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 theorem and by imposing $\rho(z) = 2z$, the second part is proved.

The last part of the theorem agrees with the result in Eguchi and Copas (2001, 2002). U-functions for AdaBoost, LogitBoost, and MadaBoost satisfy the condition 4.17.

4.3 Asymptotic Covariance. To see the efficiency of the U-model, we investigate the asymptotic covariance of α in this section.

Let us consider the generic U-model in the form of

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

parameterized by $\alpha = (\alpha_t; t = 1, \ldots, T)$, and consider the case that the auxiliary function b does not depend on the data. Let $p(y|\boldsymbol{x})$ be the true

conditional distribution. The optimal point q^* in the U-model is given by

$$\alpha^* = \operatorname*{argmin}_{\alpha} H_U(p, q; \mu) = \operatorname*{argmin}_{\alpha} \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} \left\{ U(\xi(q)) - p\xi(q) \right\} d\mu, \quad (4.18)$$

and for given n examples, the estimate of α is given by

$$\begin{aligned} \hat{\alpha} &= \operatorname*{argmin}_{\alpha} L_U(q) \\ &= \operatorname*{argmin}_{\alpha} H_U(\tilde{p}, q; \tilde{\mu}) = \operatorname*{argmin}_{\alpha} \int_{\boldsymbol{x} \in \mathcal{X}} \sum_{y \in \mathcal{Y}} \left\{ U(\xi(q)) - \tilde{p}\xi(q) \right\} d\tilde{\mu}, \end{aligned} \quad (4.19)$$

in abstract form. When n is sufficiently large, the covariance of $\hat{\alpha}$ with respect to all the possible sample sets is given as follows:

Theorem 6. *The asymptotic covariance of $\hat{\alpha}$ is given by*

$$\mathrm{Cov}(\hat{\alpha}) = \frac{1}{n} H^{-1} G H^{-1} + o\left(\frac{1}{n}\right), \quad (4.20)$$

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

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

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

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

and $\boldsymbol{f} = (f_1, f_2, \ldots, f_T)^T$.

The proof is given by usual asymptotic arguments (see Murata, Yoshizawa, and Amari, 1994, for example).

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

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

where I is the Fisher information matrix of the logistic model, which means LogitBoost attains the Cramér-Rao bound asymptotically, that is, LogitBoost is asymptotic efficient. In general, the asymptotic covariance of U-Boost algorithms is inferior to the Cramér-Rao bound; hence, from this point of view, U-Boost is not efficient. However, instead of efficiency, some 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

$$E(L_U(q_t)) = E(H_U(\tilde{p}, q_t; \tilde{\mu})) = H_U(p, q^*; \mu) + \frac{1}{2n} \operatorname{tr} H^{-1} G + o\left(\frac{1}{n}\right), \quad (4.21)$$

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

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

4.4.1 Most B-Robust U-Function. Let us consider the statistical model with one parameter α,

$$\begin{aligned}&\mathcal{Q}^{\mathrm{norm}}(q_0, \{yh(\boldsymbol{x})\})\\ &\quad = \{q \in \mathcal{P} \mid \log q(y|\boldsymbol{x}, \alpha) = \log q_0(y|\boldsymbol{x}) + \alpha y h(\boldsymbol{x}) - b(\boldsymbol{x}, \alpha),\ \alpha \in R\},\end{aligned} \quad (4.22)$$

where $q_0 \in \mathcal{P}$ and $h(\boldsymbol{x})$ takes $+1$ or -1. Let us define the likelihood ratio of q_0 by

$$F(\boldsymbol{x}) = \frac{1}{2} \log \frac{q_0(+1|\boldsymbol{x})}{q_0(-1|\boldsymbol{x})}.$$

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

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

where $q\mu$ is the joint distribution of $\boldsymbol{x}$ and y. As considered in the previous section when the U-function satisfies condition 4.17, the estimator by the

U-function satisfies

$$\alpha_U(q_\alpha \mu) = \alpha$$

for any $q_\alpha \in \mathcal{Q}^{\mathrm{norm}}(q_0, \{yh(x)\})$.

We measure the robustness of the estimator by the gross error sensitivity (Hampel, Rousseeuw, Roncheti, & Stahel, 1986)

$$\gamma(U, q_0) = \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} \lim_{\epsilon \to +0} \frac{(\alpha_U(\tilde{p}_\epsilon) - \alpha_U(q_0\mu))^2}{\epsilon^2}, \tag{4.23}$$

where

$$\tilde{p}_\epsilon = (1-\epsilon)\, q_0 \mu + \epsilon\, \delta(\tilde{\boldsymbol{x}}, \tilde{y})$$

and $\delta(\tilde{\boldsymbol{x}}, \tilde{y})$ is the probability distribution with a point mass at $(\tilde{\boldsymbol{x}}, \tilde{y})$. The gross error sensitivity measures the worst influence that a small amount of contamination can have on the estimate. 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 7. *The U-function that derives the MadaBoost algorithm minimizes the gross error sensitivity among the U-function with the property of 4.17.*

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

$$\gamma(U, q_0) = \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} u(\tilde{y}F(\tilde{\boldsymbol{x}}))^2 \left(\int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} u'(-F(\boldsymbol{x})y) q_0(y|\boldsymbol{x}) \mu(\boldsymbol{x}) d\boldsymbol{x} \right)^{-2}. \tag{4.24}$$

Knowing that the conditional probability is written with F as

$$\begin{aligned} q_0(y|\boldsymbol{x}) &= \frac{1}{1 + \exp(-2yF(\boldsymbol{x}))} \\ &= \frac{u(yF(\boldsymbol{x}))}{u(F(\boldsymbol{x})) + u(-F(\boldsymbol{x}))} \end{aligned}$$

and u satisfies

$$\frac{u'(z)}{u(z)} + \frac{u'(-z)}{u(-z)} = 2$$

from the consistent condition on u

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

we get

$$\begin{aligned}
&u'(-F(\boldsymbol{x}))q_0(1|\boldsymbol{x}) + u'(F(\boldsymbol{x}))q_0(-1|\boldsymbol{x}) \\
&\quad = \frac{u'(-F(\boldsymbol{x}))u(F(\boldsymbol{x}))}{u(F(\boldsymbol{x})) + u(-F(\boldsymbol{x}))} + \frac{u'(F(\boldsymbol{x}))u(-F(\boldsymbol{x}))}{u(F(\boldsymbol{x})) + u(-F(\boldsymbol{x}))} \\
&\quad = \frac{2u(F(\boldsymbol{x}))u(-F(\boldsymbol{x}))}{u(F(\boldsymbol{x})) + u(-F(\boldsymbol{x}))} \\
&\quad = 2u(F(\boldsymbol{x}))q_0(-1|\boldsymbol{x}).
\end{aligned}$$

By imposing the above relation, we obtain the expression of the gross error sensitivity only with function u as

$$\gamma(U, q_0) = \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} u(\tilde{y}F(\tilde{\boldsymbol{x}}))^2 \left(2\int_{\mathcal{X}} u(F(\boldsymbol{x}))q_0(-1|\boldsymbol{x})\mu(\boldsymbol{x})d\boldsymbol{x}\right)^{-2}. \tag{4.25}$$

From the above formula, we find that the gross error sensitivity diverges if $u(yF(\boldsymbol{x}))$ is not bounded and the integration of $u(F(\boldsymbol{x}))$ by $q_0(-1|\boldsymbol{x})\mu(\boldsymbol{x})$ is bounded. 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 that maximizes

$$\int_{\mathcal{X}} u(F(\boldsymbol{x}))q_0(-1|\boldsymbol{x})\mu(\boldsymbol{x})d\boldsymbol{x}.$$

The point-wise 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.

4.4.2 Robustness of Boosting Algorithm. Next, we study the robustness of the U-Boost. Let us define the normalized U-model $\mathcal{Q}_U^{\mathrm{norm}}(q_0, \mathcal{F})$ with the set of binary decision functions,

$$\mathcal{F} = \{yh_1(\boldsymbol{x}), \ldots, yh_T(\boldsymbol{x})\},$$

where $h_t(\boldsymbol{x})$ takes $+1$ or -1.

When the probability $q_0(y|\boldsymbol{x})$ changes to $\tilde{p}_\epsilon$, the U-Boost estimator is altered from $F(\boldsymbol{x})$ to

$$F(\boldsymbol{x}) + \alpha_U(\tilde{p}_\epsilon)\tilde{h}(\boldsymbol{x}),$$

where $\tilde{h}(\boldsymbol{x})$ is an element of $\{h_1(\boldsymbol{x}), \ldots, h_T(\boldsymbol{x})\}$, which depends on $\tilde{p}_\epsilon$. The probability $q_\epsilon(y|\boldsymbol{x}) \in \mathcal{Q}^{\mathrm{norm}}(q_0, \mathcal{F})$, which is specified by the above decision function, is written as

$$\begin{aligned}\log q_\epsilon(y|\boldsymbol{x}) &= \log q_0(y|\boldsymbol{x}) + \alpha_U(\tilde{p}_\epsilon)y\tilde{h}(\boldsymbol{x}) - b(\boldsymbol{x}, \alpha_U(\tilde{p}_\epsilon)) \\ &\in \mathcal{Q}^{\mathrm{norm}}(q_0, \{y\tilde{h}(\boldsymbol{x})\}).\end{aligned}$$

Let us measure the robustness of U-Boost by the gross error sensitivity of the distribution estimated by the algorithm, which is defined with the KL divergence as

$$\begin{aligned}&\gamma_{\mathrm{boost}}(U, q_0) \\ &\quad = \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} \lim_{\epsilon \to +0} \frac{2}{\epsilon^2} \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} q_0(y|\boldsymbol{x})(\log q_0(y|\boldsymbol{x}) - \log q_\epsilon(y|\boldsymbol{x}))\mu(\boldsymbol{x})d\boldsymbol{x}. \quad (4.26)\end{aligned}$$

Intuitively, this measures the sensitivity of the KL divergence between the true and the estimated distributions under the small contamination.

For a fixed $(\tilde{\boldsymbol{x}}, \tilde{y})$, the chosen classifier $\tilde{h}$ does not depend on the value of ϵ because

$$\langle \tilde{p}_\epsilon - q_0, yh_t(\boldsymbol{x}) - b_t' \rangle = \epsilon \cdot \langle \delta(\tilde{\boldsymbol{x}}, \tilde{y}) - q_0, yh_t(\boldsymbol{x}) - b_t' \rangle$$

holds. Therefore, we find that

$$\begin{aligned}&\lim_{\epsilon \to +0} \frac{2}{\epsilon^2} \int_{\mathcal{X}} \sum_{y \in \mathcal{Y}} q_0(y|\boldsymbol{x})(\log q_0(y|\boldsymbol{x}) - \log q_\epsilon(y|\boldsymbol{x}))\mu(\boldsymbol{x})d\boldsymbol{x} \\ &\quad = \lim_{\epsilon \to +0} I(\tilde{h}) \cdot \frac{(\alpha_U(\tilde{p}_\epsilon) - \alpha_U(q_0\mu))^2}{\epsilon^2}\end{aligned}$$

by the asymptotic expansion, where $I(\tilde{h})$ is the Fisher information of $\mathcal{Q}^{\mathrm{norm}}$ $(q_0, y\tilde{h}(\boldsymbol{x}))$ at $\alpha = 0$. From the property of $(yh(\boldsymbol{x}))^2 = 1$, we find that $I(h)$ does not depend on h, and let us define I_0 as the common value of $I(h_t)$ for $t = 1, \ldots, T$.

From the above argument, we find that

$$\begin{aligned}\gamma_{\mathrm{boost}}(U, q_0) &= \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} \lim_{\epsilon \to +0} I(\tilde{h}) \cdot \frac{(\alpha_U(\tilde{p}_\epsilon) - \alpha_U(q_0\mu))^2}{\epsilon^2} \\ &= I_0 \sup_{(\tilde{\boldsymbol{x}}, \tilde{y})} \lim_{\epsilon \to +0} \frac{(\alpha_U(\tilde{p}_\epsilon) - \alpha_U(q_0\mu))^2}{\epsilon^2} \\ &= I_0 \gamma(U, q_0).\end{aligned}$$

Hence, the U-function of MadaBoost also minimizes $\gamma_{\mathrm{boost}}(U, q_0)$. As a consequence, MadaBoost minimizes the influence of outliers around the true distribution.

4.5 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 Figure 13. The detailed setup is

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

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 the top $10a$% examples, a% are randomly chosen and flipped without replacement. That means that contamination is avoided around the boundary of classification. The plots are made by averaging 50 different runs. 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 Figure 14, 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 Figure 15, the difference between 1% contamination and noncontamination and between 2% contamination and noncontamination are plotted, respectively.

In fact, Figures 14b and 14c for the examples with outliers show that LogitBoost and MadaBoost stably provide an optimal performance around

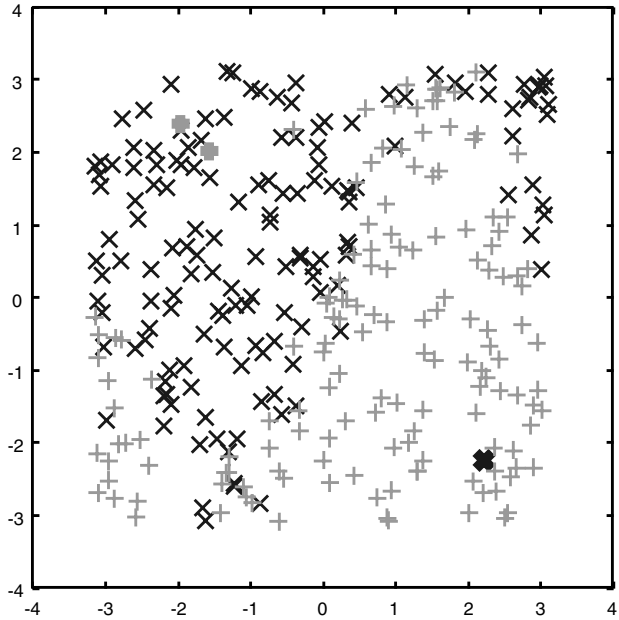


Figure 13: Typical examples with contamination.

the boosting number 30 to 50, while AdaBoost fails to attain such an optimal performance observed for the noncontaminated examples case, as observed in Figure 14a. Thus, we conclude that AdaBoost is more sensitive to outliers than MadaBoost in respect to learning curves.

5 Conclusion

In this article, we formulated boosting algorithms as sequential updates of conditional measures, and we introduced a class of boosting algorithms by considering the relation with the Bregman divergence. Using a statistical framework, we discuss properties of consistency, efficiency, and robustness.

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

Here we treated only 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 remains a subject for future work.

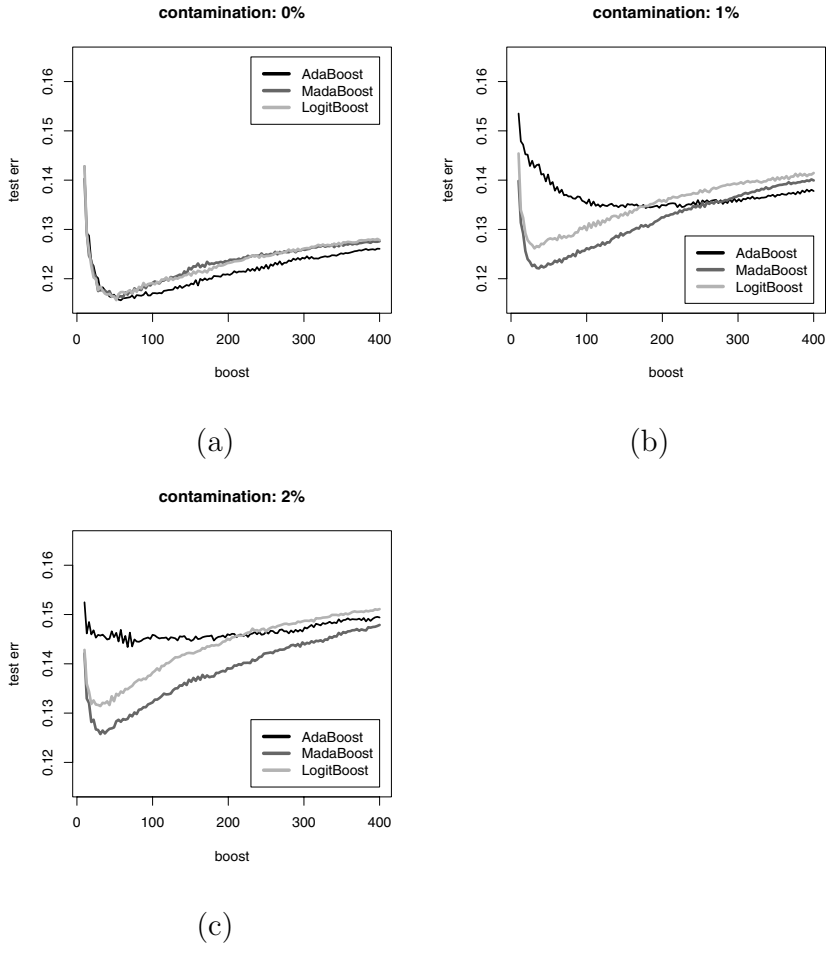


Figure 14: Test error of boosting algorithms. (a) Training data are not contaminated. (b) 1% contamination. (c) 2% contamination.

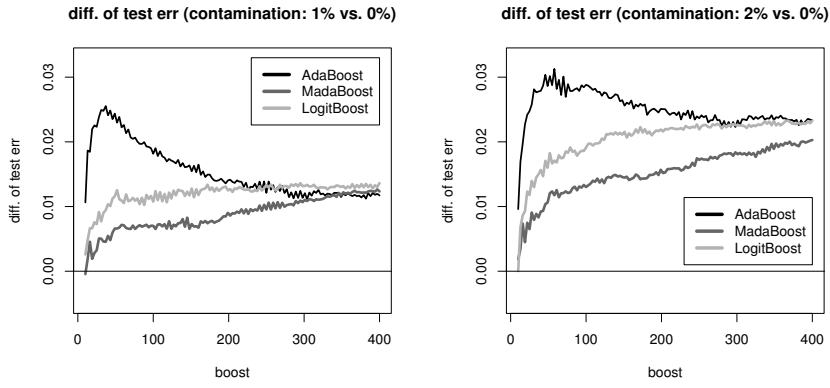
References

Amari, S. (1985). *Differential-geometrical methods in statistics*. Berlin: Springer-Verlag.

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*. New York: Oxford University Press.

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



(a) (b)

Figure 15: Difference of test errors of the original examples and that of the contaminated examples. (a) Difference between 1% contamination and noncontamination. (b) Difference between 2% contamination and noncontamination.

Bishop, C. (1995). *Neural networks for pattern recognition*. Oxford: Clarendon Press.

Collins, M., Schapire, R. E., & Singer, Y. (2000). Logistic regression, Adaboost and Bregman distances. In *Proceedings of the Thirteenth Annual Conference on Computational Learning Theory* (pp. 158–169). San Francisco: Morgan Kaufmann.

Domingo, C., & Watanabe, O. (2000). MadaBoost: A modification of AdaBoost. In *Proceedings of the Thirteenth Conference on Computational Learning Theory* (pp. 180–189). San Francisco: Morgan Kaufmann.

Eguchi, S., & Copas, J. B. (2001). Recent developments in discriminant analysis from an information geometric point of view. *Journal of the Korean Statistical Society, 30*, 247–264.

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

Eguchi, S., & Kano, Y. (2001). *Robustifing maximum likelihood estimation by psi-divergence* (ISM Research Memorandum 802). Tokyo: Institute of Statistical Mathematics.

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

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Machine Learning: Proceedings of the Thirteenth International Conference* (pp. 148–156). San Francisco: Morgan Kaufmann.

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

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

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

Hastie, T., Tibishirani, R., & Friedman, J. (2001). *The elements of statistical learning*. New York: Springer-Verlag.

Kearns, M., & Valiant, L. G. (1988). *Learning boolean formulae or finite automata is as hard as factoring* (Tech. Rep. TR-14-88). Cambridge, MA: Harvard University Aiken Computation Laboratry.

Kivinen, J., & Warmuth, M. K. (1999). Boosting as entropy projection. In *Proceedings of the Twelfth Annual Conference on Computational Learning Theory* (pp. 134–144). New York: ACM Press.

Lebanon, G., & Lafferty, J. (2001). *Boosting and maximum likelihood for exponential models* (Tech. Rep. CMU-CS-01-144). Pittsburgh, PA: School of Computer Science, Carnegie Mellon University.

McLachlan, G. J. (1992). *Discriminant analysis and statistical pattern recognition*. New York: Wiley.

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

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

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

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

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

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

Vapnik, V. (1995). *The nature of statistical learning theory*. Berlin: Springer-Verlag.

Received December 3, 2003; accepted January 8, 2004.