A novel boosting algorithm for multi-task learning based on the Itakuda-Saito divergence

Takashi Takenouchi*, Osamu Komori† and Shinto Eguchi†

*Future university Hakodate, ttakashi@fun.ac.jp.
†Institute of Statistical Mathematics, Japan

Abstract. In this paper, we propose a novel multi-task learning algorithm based on an ensemble learning method. We consider a specific setting of the multi-task learning for binary classification problems, in which features are shared among all tasks and all tasks are targets of performance improvement. We focus on a situation that the shared structures among dataset are represented by divergence between underlying distributions associated with multiple tasks. We discuss properties of the proposed method and investigate validity of the proposed method with numerical experiments.

Keywords: Multi-task learning, Itakura-Saito distance, Un-normalized pseudo model

INTRODUCTION

In the framework of multi-task learning problems, we assume that there are multiple related tasks (datasets) sharing common structures, and can utilize the shared structures to improve generalization performance of predictors for multiple tasks [7]. This framework has been successfully employed to various kind of applications such as medical diagnosis. Most methods utilize similarity among tasks to improve performance by representing the shared structure as a regularization term [1, 3]. We tackle with this problem using a boosting method which makes it possible to adaptively learn complicated problems with low computational cost. The boosting methods are notable implementations of the ensemble learning and tries to construct a better classifier by combining weak classifiers without heavy computational cost, and AdaBoost is the most popular boosting method and a lot of variation including TrAdaBoost for the multi-task learning [2] have been developed. In this paper, we firstly reveal that AdaBoost can be derived by a sequential minimization of the Itakura-Saito (IS) distance between an empirical distribution and an un-normalized pseudo measure model constructed by a classifier. The IS distance is a special case of the Bregman divergence between positive measures and is frequently used in the region of signal processing. Secondly, we propose a novel boosting algorithm for the multi-task learning based on the IS distance. We utilize the IS distance as a discrepancy measure between pseudo models associated with tasks, and incorporate the IS distance as a regularizer into the AdaBoost. The proposed method can capture the shared structure, i.e., relationship between underlying distribution by considering divergence between pseudo models estimated by constructed classifiers. We discuss statistical properties of the proposed method and investigate validity of the regularization by the IS distance with small experiments using a synthetic dataset.
SETTINGS

In this study, we focus on binary classification problems. Let \( x \) be an input and \( y \in \mathcal{Y} = \{\pm 1\} \) be a class label. Let us assume that \( J \) datasets \( \mathcal{D}_j = \{x_i^{(j)}, y_i^{(j)}\}_{i=1}^{n_j} \) \( (j = 1, \ldots, J) \) are given, and let \( p_j(x|y) \) and \( \bar{p}_j(y|x) \) be an underlying distribution and an empirical distribution associated with the dataset \( \mathcal{D}_j \), respectively. Here we assume that each conditional distribution of \( y \) given \( x \) is written as

\[
p_k(y|x) = p_0(y|x) + \delta_k(x) \quad (1)
\]

where \( p_0(y|x) \) is a common conditional distribution for all datasets and \( \delta_k(x) \) is a perturbation term specific to the dataset \( \mathcal{D}_k \). Note that \( \sum_{y \in \mathcal{Y}} \delta_k(x) = 0 \) because \( p_k(y|x) \) is a probability distribution. While a discriminant function \( F_k \) is usually constructed using only the dataset \( \mathcal{D}_k \), the multi-task learning aims to improve performance of the discriminant function for each dataset \( \mathcal{D}_k \) with the help of datasets \( \mathcal{D}_j \) \( (j \neq k) \). For the purpose, we consider a risk minimization problem defined with an un-normalized pseudo model and the Itakura-Saito (IS) distance which is a discrepancy measure frequently used in a region of signal processing.

ITAKURA-SAITO DISTANCE AND PSEUDO MODEL

Let \( \mathcal{M} = \left\{ m(y) \mid 0 \leq \sum_{y \in \mathcal{Y}} m(y) < \infty \right\} \) be a space of all the positive finite measures over \( \mathcal{Y} \). The Itakura-Saito distance between \( p, q \in \mathcal{M} \) is defined as

\[
\text{IS}(p, q, r) = \int r(x) \sum_{y \in \mathcal{Y}} \left\{ \log \frac{q(y|x)}{p(y|x)} - 1 + \frac{p(y|x)}{q(y|x)} \right\} \, dx \quad (2)
\]

where \( r(x) \) is a marginal distribution of \( x \) shared by \( p, q \in \mathcal{M} \). Note that the IS distance is a kind of statistical version of the Bregman divergence [6] and we observe that \( \text{IS}(p, q; r) \geq 0 \) and \( \text{IS}(p, q; r) = 0 \) if and only if \( p = q \).

Parameter estimation with pseudo model

Let \( q_F(y|x) \) be an un-normalized pseudo model associated with a function \( F(x) \),

\[
q_F(y|x) = \exp(F(x)y) \quad (3)
\]

Note that \( q_F(y|x) \) is not a probability function, \( i.e., \sum_{y \in \mathcal{Y}} q_F(y|x) \neq 1 \) in general. If \( q_F(y|x) \) is normalized, the model reduces to the classical logistic model

\[
\bar{q}_F(y|x) = \frac{\exp(F(x)y)}{\exp(F(x)) + \exp(-F(x))} \quad (4)
\]

When the function \( F \) is parametrized by \( \theta \), the maximum likelihood estimation (MLE) \( \text{argmax}_\theta \sum_{i=1}^{n} \log \bar{q}_F(y_i|x_i) \) or equivalently minimization of the (extended) Kullback-Leibler (KL) divergence, is a powerful tool for estimation of \( \theta \), and the MLE has
Proposition 1 Assume that the underlying distribution is written as \( p(y|x) = \bar{q}_{F_0}(y|x) \) where \( F_0 \) is a function. Then we observe

\[
\min_{F} \text{IS}(p,q_F;r) = F_0, \quad \text{and} \quad \min_{F} \text{IS}(q_F,p;r) = F_0. \tag{5}
\]

On the other hand, when we consider estimation based on an extended KL divergence, \( \text{argmin}_F \text{KL}(p,q_F;r) \) where \( \text{KL}(p,q;r) = \int r(x) \sum_{y \in \mathcal{Y}} \{ p(y|x) \log \frac{p(y|x)}{q(y|x)} - p(y|x) + q(y|x) \} dx \), we observe the following.

Remark 1 Assume that the underlying distribution is written as \( p(y|x) = \bar{q}_{F_0}(y|x) \) where \( F_0 \) is a function (\( F_0 \neq 0 \)). Then we observe

\[
\min_{F} \text{KL}(p,q_F;r) \neq F_0, \quad \text{and} \quad \min_{F} \text{KL}(q_F,p;r) \neq F_0. \tag{6}
\]

This shows that the extended KL divergence is not appropriate for estimation with the un-normalized pseudo model.

**Relationship with AdaBoost**

The IS distance between the underlying conditional distribution \( p(y|x) \) and the un-normalized pseudo model \( q_F(y|x) \) is written as

\[
\text{IS}(p,q_F;r) = C + \int r(x) \sum_{y \in \mathcal{Y}} \left\{ F(x)y + \frac{p(y|x)}{q_F(y|x)} \right\} dx = C + \int r(x) \sum_{y \in \mathcal{Y}} p(y|x)e^{-F(x)y} dx, \tag{7}
\]

where \( C \) is a constant and (7) is equivalent to an abstract loss of AdaBoost except for the constant term. Then sequential minimization of an empirical version of (7) is equivalent to the algorithm of AdaBoost. Also \([4, 6]\) discussed that a gradient-based boosting algorithm can be derived from minimization of the KL divergence or the Bregman divergence between the underlying distribution and an un-normalized pseudo model. An important difference between these frameworks and our framework (7) is employed pseudo model. The pseudo model employed by the previous frameworks assumes a condition called “Consistent data assumption” and is defined with the empirical distribution implying that the pseudo model varies depending on dataset. On the other hand, the pseudo model (3) employed in (7) is fixed against dataset as usual statistical models.

Also, the IS distance between two pseudo models \( q_F(y|x) \) and \( q_{F'}(y|x) \) is written as,

\[
\text{IS}(q_F,q_{F'};r) = \text{Const.} + \int r(x) \left\{ \exp(F(x) - F'(x)) + \exp(F'(x) - F(x)) \right\} dx. \tag{8}
\]

Note that \( \text{IS}(q_{F'},q_F;r) = \text{IS}(q_F,q_{F'};r) \) holds for arbitrary \( q_F \) and \( q_{F'} \) while the IS distance itself is not necessarily symmetric. Also note that the symmetric property does not hold for normalized models \( \bar{q}_F \) and \( \bar{q}_{F'} \).
PROPOSED METHOD
There are two main types of frameworks for multi-task learning. One is that there is a target dataset $\mathcal{D}_k$ and our interest is to construct a discriminant function $F_k$ utilizing remaining datasets $\mathcal{D}_j (j \neq k)$ or a priori constructed discriminant functions $F_j (j \neq k)$. The other is that our interest is to simultaneously construct better discriminant functions $F_1, \ldots, F_J$ using all $J$ datasets $\mathcal{D}_1, \ldots, \mathcal{D}_J$ by utilizing shared information among datasets.

Case 1
In this subsection, we focus on the above first framework. Let us assume that discriminant functions $F_j(x) (j \neq k)$ are given, i.e., constructed by an arbitrary binary classification method. Then let us consider a risk function

$$L_k(F_k) = IS(p_k, q_{F_k}; r_k) + \sum_{j \neq k} \lambda_j IS(q_{F_k}, q_{F_j}; r_k)$$

$$= \int r_k(x) \left\{ \sum_{y \in \mathcal{Y}} p_k(y|x) e^{-F_k(x)y} + \sum_{j \neq k} \lambda_j \left( e^{F_k(x) - F_j(x)} + e^{F_j(x) - F_k(x)} \right) \right\} dx,$$

where $\lambda_j (j \neq k)$ are regularization constants. Note that the risk function depends on functions $F_j (j \neq k)$. The second term becomes small when the target discriminant function $F_k$ is similar to functions $F_j (j \neq k)$ in the sense of the IS distance and corresponds to a regularizer incorporating the shared information among datasets into the target function $F_k$. Note that the marginal distribution $r_k$ is shared in the second term for easiness of implementation and simplicity of theoretical analysis. The minimizer $F_k^*$ of the risk function satisfies the following

$$\frac{\delta L_k(F_k)}{\delta F_k(x)} \bigg|_{F_k=F_k^*} = -p_k(+1|x)e^{-F_k^*(x)} + p_k(-1|x)e^{F_k^*(x)} + \sum_{j \neq k} \lambda_j \left( e^{F_k^*(x) - F_j(x)} - e^{F_j^*(x) - F_k(x)} \right) = 0,$$

which implies

$$F_k^*(x) = \frac{1}{2} \log \frac{p_k(+1|x) + \sum_{j \neq k} \lambda_j \exp(F_j(x))}{p_k(-1|x) + \sum_{j \neq k} \lambda_j \exp(-F_j(x))},$$

(11)

or equivalently

$$p_k(y|x) = p_{0,k}(y|x) \left( 1 + \sum_{j \neq k} \lambda_j \exp(-F_j(x)y) \right) - p_{0,k}(-y|x) \sum_{j \neq k} \lambda_j \exp(F_j(x)y),$$

(12)

where $p_{0,k}(y|x) = \frac{\exp(F_k(x)y)}{\exp(F_k(x)y) + \exp(-F_k(x)y)}$. This can be interpreted as a probabilistic model of asymmetric mislabeling [8, 9]. In (11), confidence of classification is discounted by results of remaining discriminant functions when the classifier $\text{sgn}(F_k^*(x))$ makes
a different decision from these of \( \text{sgn}(F_j(x)) \) \((j \neq k)\). Note that \( F_k^\ast(x) \geq 0 \) does not mean \( p_k(+1|x) = \frac{1}{2} \) unless \( F_j(x) = \frac{1}{2} \log \frac{p_k(+1|x)}{p_k(-1|x)} \) holds.

**Proposition 2** Let us assume that \( F_j(x) \) satisfies
\[
\frac{\exp(F_j(x)y)}{\exp(F_j(x)) + \exp(-F_j(x))} = p_0(y|x) + \varepsilon_j(x)y, \quad ||\varepsilon_j(x)|| \ll 1. \tag{13}
\]

Then (11) can be approximated as
\[
F_k^\ast(x) \approx \frac{1}{2} \log p_0(+1|x) + \frac{1}{2} \frac{1}{p_0(-1|x) + \frac{1}{2} \frac{P \delta_k(x) + \sum_{j \neq k} \lambda_j \varepsilon_j(x)}{P^3 + \sum_{j \neq k} \lambda_j \varepsilon_j(x)} \tag{14}
\]

where \( P = \sqrt{p_0(+1|x)p_0(-1|x)} \) and \( \Lambda_k = \sum_{j \neq k} \lambda_j \).

We observe that a discrepancy \( \delta_k \) of \( F_k^\ast \) from the optimal discriminant function associated with the common structure \( p_0 \) is moderated by the mixture of \( \varepsilon_j \) when perturbations \( \varepsilon_j \) are independently and identically distributed.

An empirical version of (10) is written as
\[
\hat{L}_k(F_k) = \frac{1}{n_k} \sum_{i=1}^{n_k} \left( e^{-F_k(x_i^{(k)})y_i^{(k)}} + \sum_{j \neq k} \lambda_j \left( e^{F_k(x_i^{(k)}) - F_j(x_i^{(k)})} + e^{F_j(x_i^{(k)}) - F_k(x_i^{(k)})} \right) \right). \tag{15}
\]

An algorithm is derived by sequential minimization of (15) by updating \( F_k \) to \( F_k + \alpha f \), i.e., \((\alpha, f) = \arg\min_{\alpha, f} \hat{L}_k(F_k + \alpha f) \) where \( f \) is a weak classifier and \( \alpha \) is coefficient [5].

1. Initialize function to \( F_k^0 \) and define weights for \( i \)-th example with the function \( F \) as
\[
w_1(i; F) = \frac{e^{-F(x_i^{(k)})y_i^{(k)}}}{Z_1(F)}, \quad w_2(i; F) = \frac{\sum_{j \neq k} \lambda_j e^{-f(x_i^{(k)}) - F_j(x_i^{(k)})}}{Z_2(F)}, \quad \text{where}
\]
\[
Z_1(F) = \sum_{i=1}^{n_k} e^{-F(x_i^{(k)})y_i^{(k)}}, \quad Z_2(F) = \sum_{i=1}^{n_k} \sum_{j \neq k} \lambda_j (e^{F(x_i^{(k)}) - F_j(x_i^{(k)})} + e^{-F(x_i^{(k)}) + F_j(x_i^{(k))}}).
\]

2. For \( t = 1, \ldots, T \)
   (a) Select a weak classifier \( f_t^k \in \{ \pm 1 \} \) which minimizes the following quantity
\[
\varepsilon(f) = \frac{Z_1(F_k^{t-1})}{Z_1(F_k^{t-1}) + Z_2(F_k^{t-1})} \varepsilon_1(f) + \frac{Z_2(F_k^{t-1})}{Z_1(F_k^{t-1}) + Z_2(F_k^{t-1})} \varepsilon_2(f). \tag{16}
\]
   where \( \varepsilon_1(f) = \sum_{i=1}^{n_k} w_1(i; F_k^{t-1}) \mathbf{1}(f(x_i^{(k)}) \neq y_i^{(k)}) \) and \( \varepsilon_2(f) = \sum_{i=1}^{n_k} w_2(i; F_k^{t-1}) \).
(b) Calculate a coefficient of \( f_t^k \) by \( \alpha_t^k = \frac{1}{2} \log \frac{1 - \varepsilon(f_t^k)}{\varepsilon(f_t^k)} \).
(c) Update the discriminant function as \( F_t^k = F_t^{k-1} + \alpha_t^k f_t^k \).

3. Output \( F_T^k(x) = F_0^k(x) + \sum_{t=1}^{T} \alpha_t^k f_t^k(x) \).
In step 1, \( F^0_k \) is typically initialized as \( F^0_k(x) = 0 \). (16) is a mixture of two quantities: \( \varepsilon_1(f) \) is a weighted error rate of the classifier \( f \) and \( \varepsilon_2(f) \) is sum of weights \( w_2(f) \) which represents degree of discrepancy between \( f \) and \( F - F_j \). \( \varepsilon_2(f) \) becomes large when \( F \) is updated as departed from \( F_j \) by \( f \).

**Case 2**

In this subsection, we consider simultaneous construction of discriminant functions \( F_1, \ldots, F_J \) by minimizing the following risk function:

\[
L(F_1, \ldots, F_J) = \sum_{j=1}^{J} \pi_j L_j(F_j) \tag{17}
\]

where \( \pi_j(j = 1, \ldots, J) \) is a constant and \( L_k \) is defined in (10). While we can directly minimize the empirical version of (17), the derived algorithm is complicated and is computationally heavy. Then we derive a simplified algorithm utilizing the algorithm shown in Case 1 in which a target dataset is fixed.

1. Initialize functions \( F_1, \ldots, F_J \).
2. For \( t = 1, \ldots, T \)
   (a) Randomly choose a target index \( k \in \{1, \ldots, J\} \).
   (b) Update the function \( F_k \) using the algorithm in Case 1 by \( S \) steps with fixed functions \( F_j \) \((j \neq k)\).
3. Output learned functions \( F_1, \ldots, F_J \).

Note that the empirical risk function cannot be monotonically decreased because minimization of \( L_k(F_k) \) is a trade-off of the first term and the second regularization term, and decrease of \( L_k(F_k) \) does not mean decrease of the regularization term in general.

**COMPARISON OF REGULARIZATION TERMS**

The proposed method incorporates the regularization term defined by the IS distance into the AdaBoost. In this section, we discuss a property of the regularization term.

**Proposition 3** Let us assume that a perturbation function \( \varepsilon(x) \) satisfies \( |\varepsilon(x)| \ll 1 \).

\[
\text{KL}(\tilde{q}_F, \tilde{q}_F + \varepsilon; r) \approx \int 2r(x)\tilde{q}_F(+1|x)\tilde{q}_F(-1|x)\varepsilon(x)^2 dx \tag{18}
\]

\[
\text{KL}(q_F, q_F + \varepsilon; r) \approx \int \frac{r(x)}{2\sqrt{\tilde{q}_F(+1|x)\tilde{q}_F(-1|x)}} \varepsilon(x)^2 dx \tag{19}
\]

\[
\text{IS}(\tilde{q}_F, \tilde{q}_F + \varepsilon; r) \approx \int 2r(x) \sum_{y \in \mathcal{Y}} \tilde{q}_F(y|x)^2 \varepsilon(x)^2 dx \tag{20}
\]

\[
\text{IS}(q_F, q_F + \varepsilon; r) \approx \int r(x) \varepsilon(x)^2 dx \tag{21}
\]
FIGURE 1. Generated three datasets and decision boundaries.

Those relations implies that the KL-divergence (18) emphasizes a region of input $x$ whose conditional distribution is nearly equal to $\frac{1}{2}$, while the IS-distance (20) focuses on a region of $x$ whose conditional distribution is nearly equal to 0 or 1. The IS-distance between pseudo models (21) considers intermediate of (18) and (20).

EXPERIMENTS

In this section, we investigate performance of the proposed method using a synthetic dataset within the situation described in Case 2. We set the number $J$ of dataset to 3 and assume that a marginal distribution of $x$ is uniform distribution on $[-1,1]^2$, and a discriminant function $F_j$ ($j = 1, 2, 3$) associated with each dataset is generated by $F_j(x) = c_{j,2}(x_1 - c_{j,1}) - x_2$ where $c_1 \sim \mathcal{N}(0,0.2^2)$ and $c_2 \sim \mathcal{N}(1,0.1^2)$. In addition, we randomly added a contamination noise on label $y$. Under these settings, we generated training dataset including 400 examples and test dataset including 600 examples. Generated datasets are shown in figure 1. We observe that each discriminant function and noise structure are different from the other two. We compared the proposed method (A) with AdaBoost learned with an individual dataset (B), and AdaBoost learned with all datasets simultaneously (C). We employed the boosting stump $^1$ as the weak classifier and fixed as $\pi_j = 1/3$. A boosting-type method has a hyper-parameter $T$, step number of boosting, and the proposed method additionally has the hyper-parameter $\lambda_j$. In the experiment, we assume that $\lambda_j = \lambda$ for all $j$ and determined $\lambda$ and $T$ by the validation technique. We utilized 80% of the training dataset for training of classifiers and the remaining 20% for the validation. We repeated the above procedure 20 times and observed averaged performance of methods. A figure 2 shows boxplots of test errors of each method for datasets $\mathcal{D}_j$ ($j = 1, 2, 3$). We observe that the proposed method consistently outperforms individually learned AdaBoost, and AdaBoost learned with all datasets simultaneously. This implies that the proposed method can incorporate shared information among datasets into classifiers.

---

$^1$ Boosting stump is a decision tree with only one node.
FIGURE 2. Boxplots of the test error of each method (A: proposed method, B: AdaBoost learned with the individual dataset, and C: AdaBoost learned with all datasets simultaneously) for three datasets, over the 20 simulation trials.

CONCLUSIONS

In this paper, we proposed a novel binary classification method for the multi-task learning. The proposed method is based on the Itakura-Saito distance and the un-normalized pseudo model. The IS-divergence between pseudo models can be interpreted as the regularization term incorporating shared information among tasks into the binary classifier for the target dataset. We numerically investigated performance of the proposed method with the synthetic dataset, and verified the effectiveness of the proposed method.

Acknowledgements This study was partially supported by Grant-in-Aid for Young Scientists (B), 50403340, from MEXT, Japan.

REFERENCES