A Divergence Prior for Adaptive Learning

Xiao Li* and **Jeff Bilmes***
Dept. of Electrical Engineering.
University of Washington, Seattle WA 98195-2500

1 Introduction

We assume that $(\mathbf{x}, y) \in \mathcal{X} \times \{\pm 1\}$ is a pair of (input, label) variables with a joint distribution $p(\mathbf{x}, y)$. A fundamental problem in inductive learning is to learn a decision function $f \in \mathcal{F} : \mathcal{X} \rightarrow \mathcal{Y}$ that not only correctly classifies observed samples drawn from $p(\mathbf{x}, y)$, but also generalizes to unseen samples drawn from the *same* distribution. In other words, we desire to learn an f that minimizes the true risk $R_p(f) = E_{(\mathbf{x},y)\sim p(\mathbf{x},y)}[Q(f(\mathbf{x}), y)]$ under certain loss function $Q(\cdot)$. In practice, this is often approached by minimizing the empirical risk $R_{\text{emp}}(f) = \frac{1}{m}\sum_{i=1}^{m} Q(f(\mathbf{x}_i), y_i)$ on a training set, while utilizing certain regularization strategy to guarantee good generalization performance [1]. The target (or test-time) distribution, however, is often *different* from the training distribution. Sometimes, the difference only resides in the input distribution $p(\mathbf{x})$, while the conditional relation $p(y|\mathbf{x})$ remains the same. In several learning paradigms, this type of difference has been partially accounted for by explicitly taking into account the test input distribution [2, 3]. A learning setting that has not received as much theoretical attention is that of "adaptive learning", which studies a more general case where *both* $p(\mathbf{x})$ and $p(y|\mathbf{x})$ at test time vary from their training counterparts. Another distinctive assumption of adaptive learning is that while there may be essentially an unlimited amount of labeled training distribution data, only a small amount of labeled adaptation data drawn from the target distribution is available.

To formally define the adaptive learning paradigm, we let $p^{tr}(\mathbf{x}, y)$ and $p^{ad}(\mathbf{x}, y)$ denote the training and target distributions respectively, and we assume that two sources of information are given in a priori:

1. an "unadapted classifier" f^{tr}, trained on a sufficiently large amount of training data (but this data is not preserved), such that $f^{tr} \in \operatorname{argmin}_{f\in\mathcal{F}} R_{p^{tr}}(f)$;
2. "adaptation data" $\mathcal{D}_m^{ad} = \{(\mathbf{x}_i, y_i)|(\mathbf{x}_i, y_i) \sim p^{ad}(\mathbf{x}, y)\}_{i=1}^{m}$.

The goal of adaptation is to produce an "adapted classifier" that is as close as possible to our desired classifier $f^{ad} \in \operatorname{argmin}_{f\in\mathcal{F}} R_{p^{ad}}(f)$. In this setting, adaptation is supervised, as both training and adaptation data are labeled; it is also inductive, as the adapted classifier is desired to generalize to unseen data drawn from $p^{ad}(\mathbf{x}, y)$. Note that adaptation can be unsupervised or i transductive with modified assumptions. In fact, there has already been a vast amount of practical work on speaker/writer/domain adaptation, involving a variety of classifiers [4, 5, 6, 7]. We have, however, found little principled study that unifies adaptation algorithms for different classifiers. Moreover, a more fundamental question would be whether we can relate the adaptation sample complexity to the divergence between training and target distributions. We investigate these two problems in detail in Section 2 and Section 3 respectively, and present empirical classification experiments in the last section.

2 A divergence prior for regularized adaptation

We approach the adaptation problem from a Bayesian perspective by assuming that f itself is a random variable with a "standard" prior distribution $\pi(f)$ (which is chosen before seeing any training or test data, usually based on domain knowledge). In adaptation, we utilize the concept of "accuracy-regularization" where we seek a classifier that, on one hand, attains low empirical risk on adaptation

*This material is based on work supported by the National Science Foundation under grant IIS-0326382.

data, and on the other hand, has good generalization ability as measured by a regularizer. We propose to use a "divergence prior" $p_{\text{div}}(f)$ as the regularizer (to be defined shortly). Note that both $\pi(f)$ and $p_{\text{div}}(f)$ are Bayesian priors; the difference is that the former is chosen *before* training the unadapted model, whereas the latter is chosen *after* the unadapted model is obtained. Specifically, the divergence prior is defined as

$$\ln p_{\text{div}}(f) = \mathrm{E}_{p^{tr}(\mathbf{x},y)}[\ln p(f|\mathbf{x},y)] + \gamma \tag{1}$$

where $p^{tr}(\mathbf{x},y)$ again is the training distribution, $p(f|\mathbf{x},y)$ is the posterior probability of a classifier given a sample and γ is a normalization constant such that $p_{\text{div}}(f)$ sums to unity. Intuitively this prior tells how likely a classifier is given the training distribution. The reason we choose such a prior is that, as will be discussed shortly, $p_{\text{div}}(f)$ incorporates information from both the standard prior $\pi(f)$ and the unadapted classifier f^{tr}, and that it assigns higher probabilities to classifiers "closer" to f^{tr} in terms of KL-divergence. Using this prior amounts to an adaptation objective $\min R_{\text{emp}}(f) - \lambda \ln p_{\text{div}}(f)$, where λ balances the tradeoff between fitting D_m^{ad} and staying close to f^{tr}. Intuitively, the more adaptation data we have, or the more different the training and target distributions are, the smaller λ we should use.

This divergence prior leads to a unified adaptation strategy applicable to a variety of classifiers. We first explore the instantiation of $p_{\text{div}}(f)$ for classifiers using generative models. In such a case, the function space $\mathcal{F}$ consists of generative models f that describe the sample distribution $p(\mathbf{x},y|f) = p(\mathbf{x}|y,f)p(y|f)$ (here we slightly abuse notation by letting f denote a generative model instead of a decision function). If we use $Q(\cdot) = -\ln p(\mathbf{x},y|f)$, it is easy to prove that f^{tr} is the *true* model generating the training distribution, *i.e.*, $p(\mathbf{x},y|f^{tr}) = p^{tr}(\mathbf{x},y)$. Similarly, we have $p(\mathbf{x},y|f^{ad}) = p^{ad}(\mathbf{x},y)$. Note that by doing this, we implicitly assume that $\mathcal{F}$ contains the true generative models in both cases. Furthermore, applying Bayes rule, the posterior probability in Equation (1) can be factorized as $p(f|\mathbf{x},y) \propto p(\mathbf{x},y|f)\pi(f)$, leading to the following result.

$$-\ln p_{\text{div}}(f) = D(p(\mathbf{x},y|f^{tr})||p(\mathbf{x},y|f)) - \ln \pi(f) - \beta \tag{2}$$

where β is a normalization constant, and it can be proved that $\beta > 0$. Our adaptation objective for generative classifiers consequently becomes

$$\min_{f\in\mathcal{F}} R_{\text{emp}}(f) + \lambda D(p(\mathbf{x},y|f^{tr})||p(\mathbf{x},y|f)) - \lambda \ln \pi(f), \tag{3}$$

Given a uniform $\pi(f)$, this objective asks to minimize the KL-divergence between the sample distribution generated by f^{tr} and that generated from the model of interest.

The divergence prior for classifiers using conditional models can be obtained in a similar fashion. In this case, the function space $\mathcal{F}$ consists of conditional models f. Using log conditional likelihood loss, we have $p(y|\mathbf{x},f^{tr}) = p^{tr}(y|\mathbf{x})$ and $p(y|\mathbf{x},f^{ad}) = p^{ad}(y|\mathbf{x})$. Furthermore, the posterior probability can be factorized as $p(f|\mathbf{x},y) \propto p(y|\mathbf{x},f)\pi(f)$ where f and $\mathbf{x}$ are assumed to be independent variables. This leads to a result analogous to Equation (2), where $D(p(\mathbf{x},y|f^{tr})||p(\mathbf{x},y|f))$ is replaced by $D(p(y|\mathbf{x},f^{tr})||p(y|\mathbf{x},f))$. Computing the KL-divergence, however, requires the knowledge of $p^{tr}(\mathbf{x})$. In the case we do not know $p^{tr}(\mathbf{x})$, we seek an upper bound on the KL-divergence independent of the input distribution. To this end, we need to specify the form of $p(y|\mathbf{x},f)$. one class of discriminative classifiers, including MLPs, SVMs, CRFs and conditional maximum entropy models, can be viewed as generalized log linear models, *i.e.* $p(y|\mathbf{x},f) = \sigma(y(\mathbf{w}^T\phi(\mathbf{x}) + b))$ (thus f is represented by $(\mathbf{w},b)$), where $\sigma(\cdot)$ is a sigmoid function[1], and $\phi(\mathbf{x})$ is a nonlinear transformation in the input space. For consistency, we use $\mathbf{x}$ to represent features, but $\mathbf{x}$ can be readily replaced by $\phi(\mathbf{x})$ for nonlinear cases. In this setting, we can prove that the divergence prior for generalized log linear models satisfies

$$\ln p_{\text{div}}(f) \geq \alpha\|\mathbf{w} - \mathbf{w}^{tr}\| + |b - b^{tr}| - \ln \pi(f) - \beta \tag{4}$$

where $\alpha = \mathrm{E}_{p^{tr}(x)}[\|\mathbf{x}\|]$. The adaptation objective for this type of classifiers hence becomes

$$\min_f R_{\text{emp}}(f) + \frac{\lambda_1}{2}\|\mathbf{w} - \mathbf{w}^{tr}\| + \frac{\lambda_2}{2}|b - b^{tr}| - \lambda_2 \ln \pi(f) \tag{5}$$

where λ_1 and λ_2 are regularization coefficients. Given a uniform $\pi(f)$, this objective asks to minimize the euclidian distance of the weight vectors. Computationally, it is often easier to minimize squared ℓ_2-norms instead. Before we evaluate these algorithms in Section 4, we derive generalization error bounds for adaptation in the PAC-Bayesian framework.

[1]Note that although SVMs in general do not explicitly model $p(y|\mathbf{x},f)$, there have been methods to fit SVM outputs to a probability function using a sigmoid function [8].

3 PAC-Bayes Error Bound Analysis

A fundamental problem in machine learning is to study the generalization performance of a classifier in terms of an error bound or, equivalently, a sample complexity bound. A PAC-Bayesian approach [9, 10] incorporates domain knowledge in the form of a Bayesian prior and provides a guarantee on generalization error regardless of the truth of the prior. This work is particularly interested in how well an adapted classifier generalizes to unseen data drawn from the target distribution. We derive the error bounds by using our proposed prior in the PAC-Bayesian setting. It is important to note that, although we may apply different loss functions $Q(\cdot)$ (which are often surrogates of the 0-1 loss for computational tractability) in actually *training* a classifier, we use the 0-1 loss in *evaluating* error bounds in all cases below. In other words, we have $R_{p^{ad}}(f) = \mathrm{E}_{p^{ad}(\mathbf{x},y)}[I(f(\mathbf{x}) \neq y)]$, and $R_{\text{emp}}(f) = \frac{1}{m}\sum_{i=1}^{m} I(f(\mathbf{x}_i) \neq y_i)$, $(\mathbf{x}_i, y_i) \in \mathcal{D}_m^{ad}$, in the following text.

First, for a countable function space, we apply Occam's Razor bound (Lemma 1 in [9]) which bounds the true error of a single, deterministic classifier. In particular, for generative models, we replace the standard prior $\pi(f)$ in the Occam's Razor bound by the divergence prior $p_{\text{div}}(f)$ in Equation (2). As a result, the following bound holds true with probability of at least $1 - \delta$,

$$R_{p^{ad}}(f) \leq R_{\text{emp}}(f) + \sqrt{\frac{D(p(\mathbf{x}, y|f^{tr})||p(\mathbf{x}, y|f)) - \ln \pi(f) - \beta - \ln \delta}{2m}} \tag{6}$$

This result implies that for the set of classifiers $\mathcal{G} = \{f : D((p(\mathbf{x}, y|f^{tr})||p(\mathbf{x}, y|f)) < \beta\}$, their error bounds in Equation (6) are tighter than the Occam's Razor bounds using the standard prior. Since $\beta > 0$, $\mathcal{G}$ is always nonempty. For classifiers in the complementary set $\bar{\mathcal{G}}$, however, we reach the opposite argument. An important question to ask is: to which set does our estimated classifier belongs? We are particularly interested in the error bound at f^{ad}, since we desire that our estimated classifier is as close to f^{ad} as possible. If $D(p^{tr}(\mathbf{x}, y)||p^{ad}(\mathbf{x}, y)) < \beta$, we have $f^{ad} \in \mathcal{G}$ and we achieve better generalization performance at f^{ad} by using the divergence prior. Practically speaking, this implies that it is better to utilize adaptation unless the training and target distributions are quite different.

McAllester's PAC-Bayesian bound on stochastic errors for Gibbs classifiers [9, 10] is applicable to both countable and uncountable function spaces. As a ramification of this theorem, PAC-Bayesian margin bounds have been developed which provide theoretical foundations for SVMs [11]. We study hyperplane classifiers $f(\mathbf{x}) = \text{sgn}\,(\mathbf{w}^T\mathbf{x} + b)$ in a similar setting as in [11]. We use a Gaussian prior $p(f)$ centered at $(\mathbf{w}^{tr}, b^{tr})$ with the assumption that $\mathbf{w}$ and b are independent variables. Note that this prior relates to previous work on margin bounds [11] where the Gaussian prior is centered at zero. Furthermore, we choose a posterior $q(f)$ to be also a Gaussian for mathematical convenience. For simplicity we let $p(\mathbf{w}, b) = \mathcal{N}(\mathbf{w}; \mathbf{w}^{tr}, I) \cdot \mathcal{N}(b; b^{tr}, 1)$ and $q(\mathbf{w}, b) = \mathcal{N}(\mathbf{w}; \mathbf{w}', I) \cdot \mathcal{N}(b; b', 1)$, although we are allowed to use any other covariance matrices. Applying the PAC-Bayesian theorem (Theorem 1 in [9]), we arrive at the result that, for any choice of $(\mathbf{w}', b')$, the following bound holds true with probability at least $1 - \delta$,

$$\mathrm{E}_{q(f)}[R_{p^{ad}}(f)] \leq \frac{1}{m}\sum_{i=1}^{m} F(\frac{y_i(\mathbf{x}_i^T\mathbf{w}' + b')}{\sqrt{\|\mathbf{x}_i\|^2 + 1}}) + \sqrt{\frac{\frac{\|\mathbf{w}' - \mathbf{w}^{tr}\|^2 + |b' - b^{tr}|^2}{2} - \ln \delta + \ln m + 2}{2m - 1}} \tag{7}$$

where $F(t) = \int_t^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}\, dt$, and $(\mathbf{x}_i, y_i) \in \mathcal{D}_m^{ad}$. The derivation of the stochastic error, which is a key component of the proving the above result, follows the fact that any affine transformation of a Gaussian variable is still a Gaussian variable.

4 Empirical Experiments

We evaluate our adaptation algorithms on two datasets: a vowel classification dataset [5] where we conduct speaker adaptation, and a subset of the NORB database [12] for objective recognition where we conduct lighting condition adaptation. Specifically, we apply Equation (5) to MLP and SVM adaptation, where we use a uniform $\pi(f)$ and the squared ℓ_2-norm in both cases to simplify optimization. Our goal here is to compare different adaptation algorithms for a given classifier, rather than to compare different classifiers since the features we use may favor one classifier against the other.

	Vowel set			Image set	
# adapt samples per speaker/lighting	0.8K	1.6K	2.4K	90	180
Unadapted (using f^{tr})	32.03	32.03	32.03	21.4	21.4
Retrained from zeros	14.21	11.20	9.09	41.9	32.5
Retrained from f^{tr}	12.15	9.64	7.88	**19.6**	**18.6**
Regularized	**11.56**	**8.16**	**7.30**	**19.2**	**18.0**

Table 1: Adaptation of MLP classifiers

	Vowel set			Image set	
# adapt samples per speaker/lighting	0.8K	1.6K	2.4K	90	180
Unadapted (using f^{tr})	38.21	38.21	38.21	12.5	12.5
Retrained	24.70	**18.94**	**14.00**	30.1	18.9
Boosted [6]	32.32	34.11	28.85	12.1	11.8
Regularized I	**23.28**	**19.01**	**15.00**	14.8	13.4
Regularized II	28.55	25.38	20.36	**11.0**	**10.4**

Table 2: Adaptation of SVM classifiers

In MLP adaptation we use the log loss and let λ_1=λ_2=λ, and we extend this to a multi-class two-layer MLP where we regularize the input-to-hidden and hidden-to-output weight matrices with separate trade-off coefficients determined on a development set (we regularize the *input-to-hidden* layer only because we have found it to be practically advantageous — this regularizer is not derived from our divergence prior). This objective is akin to training an MLP with weight decay [13] except that we penalize the deviation from the unadapted weights. The results are summarized in Table 4. When applying Equation (5) to SVM adaptation, we utilize the hinge loss and let $\lambda_2 = 0$. Using the "kernel trick" [1], we obtain the optimal decision function: $f(\mathbf{x}) = \text{sgn}\,(\sum_{i=1}^{m} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + \sum \alpha_j^{tr} y_j^{tr} k(\mathbf{x}_j^{tr}, \mathbf{x}))$, where $(\mathbf{x}_j^{tr}, y_j^{tr})$ are support vectors from the unadapted model with coefficients α_j^{tr}. Optimal α_i are solved in the dual space using D_m^{ad} only, where the number of new support vectors is controlled by λ_1 in (5). This algorithm corresponds to "Regularized I" in Table 4. Alternatively, since the old support vectors are available at adaptation time, we can update their coefficients as well by performing optimization on both the old support vectors and the adaptation data, corresponding to "Regularized II". As shown in the tables, the regularized adaptation algorithms demonstrated good performance in practice.

References

[1] B. Schölkopf and A. J. Smola, *Learning with kernels*, The MIT Press, 2001.

[2] D. Zhou, O. Bousquet, J. Weston T. N. Lal, and B. Schlkopf, "Learning with local and global consistency," in *NIPS*, 2003.

[3] M. Sugiyama and K.-R. Mller, "Input-dependent estimation of generalization error under covariate shift," *Statistics & Decisions*, vol. 23, no. 4, 2005.

[4] P.C. Woodland, "Speaker adaptation: Techniques and challenges," in *Proc. IEEE ASRU*, 1999.

[5] X. Li and J. Bilmes, "Regularized adaptation of discriminative classifiers," in *ICASSP*, 2006.

[6] N. Mati'c, I. Guyon, J. Denker, and V. Vapnik, "Writer adaptation for on-line handwritten character recognition," in *Proc. Intl. Conf. on Document Analysis and Recognition*, 1993.

[7] C. Chelba and A. Acero, "Adaptation of maximum entropy capitalizer: Little data can help a lot," in *Empirical Methods in Natural Language Processing*, July 2004.

[8] J. Platt, "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods," in *Advances in Large Margin Classifiers*, A.J. Smola et. al., Ed., 2000, pp. 61–74.

[9] D. A. McAllester, "PAC-Bayesian stochastic model selection," *Machine learning journal*, 2001.

[10] J. Langford, "Tutorial on practical prediction theory for classification," *Journal of Machine Learning Research*, pp. 273–306, March 2005.

[11] J. Langford and J. Shawe-Taylor, "PAC-Bayes and margins," in *NIPS*, 2002.

[12] Y. LeCun, F. J. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *CVPR*, 2004.

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