Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Doubly Robust Covariate Shift Correction

Sashank J. Reddi
Machine Learning Department
Carnegie Mellon University
sjakkamr@cs.cmu.edu

Barnabás Póczos
Machine Learning Department
Carnegie Mellon University
bapoczos@cs.cmu.edu

Alex Smola
Machine Learning Department
Carnegie Mellon University
alex@smola.org

Abstract

Covariate shift correction allows one to perform supervised learning even when the distribution of the covariates on the training set does not match that on the test set. This is achieved by re-weighting observations. Such a strategy removes bias, potentially at the expense of greatly increased variance. We propose a simple strategy for removing bias while retaining small variance. It uses a biased, low variance estimate as a prior and corrects the final estimate relative to the prior. We prove that this yields an efficient estimator and demonstrate good experimental performance.

Introduction

Covariate shift is a common problem when dealing with real data. Quite often the experimental conditions under which a training set is generated are subtly different from the situation in which the system is deployed. For instance, in cancer diagnosis the training set may have an overabundance of diseased patients, often of a specific subtype endemic in the location where the data was gathered. Likewise, due to temporal changes in user interest the distribution of covariates in advertising systems is nonstationary. This requires increasing the weight of data related to, e.g., 'Gangnam style', when processing historic data logs.

A common approach to addressing covariate shift is to reweight data such that the reweighted distribution matches the target distribution. Briefly, suppose we observe $X := \{x_1, \ldots, x_m\}$ drawn iid from $q(x)$, typically with associated labels $Y := \{y_1, \ldots, y_m\}$ drawn from $p(y|x)$. This constitutes the 'training set'. However, we need to find a minimizer f_p^* of risk R_p— defined in Equation (1) — with regard to $p(y|x)p(x)$, for which we only have iid draws of the covariates $X' := \{x'_1, \ldots x'_{m'}\}$. Note that simply minimizing the empirical risk on the training data leads to a biased estimate (since training set corresponds to samples from $q(x)p(y|x)$). If p and q are known, this problem can be addressed via importance sampling in the following manner:

$$\begin{aligned} R_p[f] &= \mathbf{E}_{x\sim p(x)}\mathbf{E}_{y|x}(\ell(y, f(x))) \\ &= \int \frac{p(x)}{q(x)} q(x)\mathbf{E}_{y|x}\ell(y, f(x))dx \\ &= \mathbf{E}_{x\sim q(x)}\mathbf{E}_{y|x}\left[\beta(x)\ell(y, f(x))\right], \end{aligned} \tag{1}$$

where $\beta(x) := \frac{p(x)}{q(x)}$ and ℓ is a loss function. Correspondingly, empirical averages with respect to X and X' can be reweighted, see, e.g., (Quiñonero-Candela et al. 2008; Cortes et al. 2008) and the references therein for further details. While estimator based on Equation (1) is unbiased, it tends to increase the *variance* of the empirical averages considerably by weighting the observations by β.

This issue is particularly exacerbated when the weights are large. As a rule of thumb the *effective* sample size of a reweighted dataset is $m_{\text{eff}} := \|\beta(X)\|_1^2 / \|\beta(X)\|_2^2$ where $\beta(X)$ is the *vector* of weights $\beta(x_1), \ldots, \beta(x_m)$. This quantity naturally arises, e.g., for a weighted average of Gaussian random variables, while deriving Chernoff bounds using the weights $\beta(X)$ (Gretton et al. 2008), or in the particle filtering context (Doucet, de Freitas, and Gordon 2001). To gain better intuition for m_{eff}, consider the case where $p = q$. In this case, we have high effective sample size ($m_{\text{eff}} = m$). Whereas in the undesirable case of a single observation having very high weight, $m_{\text{eff}} \approx 1$. Hence, m_{eff} is a good indicator of the effect of $\beta(x)$ on variance of the weighted empirical averages.

Thus, while one might obtain an unbiased estimator via Equation (1), it becomes nearly useless when the effective sample size is small relative to the original sample size. This situation is frequently observed in practice insofar as we encounter cases where simple covariate shift correction not only fails to improve generalization performance on the test set but, in fact, leads to estimates that perform worse than simply minimizing the empirical risk on the training data (i.e., unweighted estimation). Moreover, in many cases the *solutions* of the biased and the unbiased risk estimates are closer than what the distributions p and q would suggest. Figure 1 shows an example of such a scenario.

The situation described above is often encountered in practice — covariate shift correction fails to improve matters due to high variance while the unweighted solution performs reasonably well. This raises the question of how we

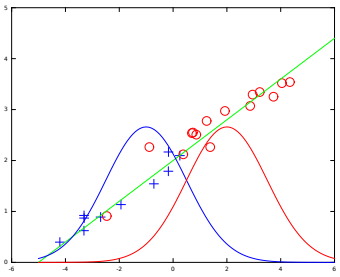

Figure 1: Assume that the dependence $y|x$ is linear in x, as indicated by the green line. In this case, inferring $y|x$ using the blue distribution q, as depicted by the blue crosses (with matching density), would lead to a perfectly accurate estimate, even if the test set is drawn according to red distribution p. On the other hand, reweighting with $\frac{p(x)}{q(x)}$ would lead to a very small effective sample size since p and q are very different. While this example is obviously somewhat artificial, there exist many situations where the minimizer of the biased risk is very good.

could benefit from the low variance of the biased estimate found by using q while removing bias via weighting with β. This is precisely what doubly robust estimators address — see, e.g., (Kang and Schafer 2007) for an overview. They provide us with two opportunities to obtain a good estimate. If the unweighted estimate solves the problem, the estimate will be very good and minimizing the unbiased risk will not change the final outcome significantly. Conversely, if the unweighted estimate is useless, we still have the opportunity to amend things in the context of estimating f_p^* by reweighting the dataset. This work focuses on tackling the problem of covariate shift correction from a doubly robust viewpoint by effectively utilizing the unweighted estimate.

Main Contributions: In summary, the paper makes the following contributions. (1) We develop a simple, yet powerful, framework for doubly robust estimation in the context of covariate shift correction, which to the best of our knowledge, has not been previously explored. (2) We demonstrate the generality of the framework by providing several concrete examples. (3) We present a general theory for the framework and provide a detailed analysis in the case of kernel methods. (4) Finally, we show good experimental performance on several UCI datasets.

Related Work

There has been extensive research in covariate shift correction problem. Most of the work is directed towards estimating the weights β. Several methods have been proposed to estimate these weights by optimization and statistical techniques (Gretton et al. 2008; Agarwal, Li, and Smola 2011; Sugiyama et al. 2008; Wen, nam Yu, and Greiner 2014). Likewise, there has been considerable work in developing doubly robust estimators for many statistical and machine learning problems, particularly in the problems involving missing data and reinforcement learning (Kang and Schafer 2007; Dudík, Langford, and Li 2011; Bang and Robins 2005). But none of these works address the problem of our concern, namely doubly robust estimation for covariate shift correction. While a few works, e.g., (Shimodaira 2000), attempt to reduce the variance by adjusting the weights and thereby, balancing the bias-variance tradeoff, they do not tackle the problem from doubly robust estimation point of view. In fact, these methods can be used in conjunction with our approach.

The most relevant to our work are (Kuzborskij and Orabona 2013), (Li and Bilmes 2007) and (Daume III 2007). All these works use similar ideas for addressing related problems in domain adaptation. However, none of these works address the problem of covariate shift correction. Moreover, our methodology and framework are much more general.

Doubly Robust Covariate Shift Correction

We first give a formal description of our problem, and then proceed to the algorithm and its theoretical analysis. Our language will be that of risk minimization. For this purpose denote by $\mathcal{X}$, with $x_i \in \mathcal{X}$, the space of covariates, and by $\mathcal{Y}$, with $y_i \in \mathcal{Y}$, the space of associated labels. For any function $f : \mathcal{X} \to \mathbb{R}$, we use f_i to denote the function evaluated at point x_i. The distributions $p(x)$ and $q(x)$ are defined on $\mathcal{X}$. Moreover, $y \sim p(y|x)$. As stated in the introduction, we assume that $x_i \sim q(x)$ and $x_i' \sim p(x)$ and $y_i \sim p(y|x_i)$. For simplicity, we assume $m = m'$ in this paper. Finally, we denote by $\ell : \mathcal{Y}^2 \to \mathbb{R}_0^+$ a loss function. We assume that the loss function is $L-$Lipschitz and bounded above by L.[1] Our goal is to minimize the expected risk with regard to distribution p $R_p[f] := \mathbf{E}_{(x,y)\sim p}[\ell(y, f(x))]$. Let $R_q[f] := \mathbf{E}_{(x,y)\sim q}[\ell(y, f(x))]$ denote expected risk with regard to distribution q. Quite often we will deal with empirical averages, often weighted. We define

$$\widehat{R}[f|X, Y, \alpha] := \frac{1}{m} \sum_i \alpha_i \ell(y_i, f(x_i))$$

The risks for X' are defined analogously. The unweighted empirical risk is $\widehat{R}[f|X, Y] = \widehat{R}[f|X, Y, 1_m]$ where 1_m is ones vector of size m. Given a class $\mathcal{F}$ of functions $\mathcal{X} \to \mathcal{Y}$ we aim to find some f_p^* that minimizes $R_p[f]$. Unfortunately, $R_p[f]$ is not directly accessible, hence we can only approximate it via the empirical risk $\widehat{R}[f|X, Y]$, or its reweighted variant $\widehat{R}[f|X, Y, \beta]$.

Furthermore, we use a regularizer Ω to ensure that we do not overfit to the data. This regularizer plays a rather critical role in our doubly robust approach. It quantifies the notion of 'simple' function. More specifically, we use $\Omega[f, f']$ to measure complexity of f relative to f'. By default we set $f' = 0$ with the corresponding shorthand $\Omega[f] := \Omega[f, 0]$. This views the constant null function as the simplest in the entire set. For instance, in kernel methods we have $\Omega[f, f'] := \frac{1}{2} \|f - f'\|^2$, where the norm is evaluated in a Reproducing Kernel Hilbert Space.

Finally, we introduce minimizers of expected and empirical risk, as is common in statistical learning theory (Vapnik 1998). We use f_p^* and f_q^* to denote the minimizers of risks

[1] We use the same constant L, without loss of generality.

R_p and R_q respectively. Throughout this paper, we use the following equivalent formulations interchangeably:

$$\hat{f}_{q,\lambda} := \operatorname*{argmin}_{f \in \mathcal{F}} \widehat{R}[f|X,Y] + \lambda \Omega[f]$$

$$\hat{f}_{q,\nu} := \operatorname*{argmin}_{f \in \mathcal{F}} \widehat{R}[f|X,Y] \text{ s.t. } \Omega[f] \leq \nu$$

The corresponding pair (λ, ν) and associated problem will be clear from the context. The equivalence follows from the fact that for any λ, there exists a ν such that the solution of the two problems is same. This is done merely for reasons of simplifying our theoretical analysis. This yields the following risk functionals with associated minimizers.

$$\hat{f}_{q,\nu_q} := \operatorname*{argmin}_{f \in \mathcal{F}} \widehat{R}[f|X,Y] \text{ s.t. } \Omega[f] \leq \nu_q \tag{2}$$

Here the risk functional, as defined in Equation (2) (referred to as unweighted estimator or minimizer) corresponds to the empirical risk minimizer when solving the inference problem with respect to the distribution $q(x)p(y|x)$. Let $\hat{\beta}$ be the estimated covariate shift weights. The next empirical risk functional is X, Y reweighted by $\hat{\beta}$ such that we obtain an unbiased estimate from p (referred to as weighted estimator or minimizer).

$$\hat{f}_{p,\nu_p} := \operatorname*{argmin}_{f \in \mathcal{F}} \widehat{R}[f|X,Y,\hat{\beta}] \text{ s.t. } \Omega[f] \leq \nu_p \tag{3}$$

Finally, let $\hat{f}_{\mathrm{DR}}$ denote doubly robust estimator which is risk minimizer, albeit with a prior around $\hat{f}_{q,\lambda}$ rather than 0.

$$\hat{f}_{\mathrm{DR}} := \operatorname*{argmin}_{f \in \mathcal{F}} \widehat{R}[f|X,Y,\hat{\beta}] \text{ s.t. } \Omega[f, \hat{f}_{q,\lambda}] \leq \nu' \tag{4}$$

Lastly, we define f^*_{q,λ_q} and f^*_{p,λ_p} to be the *penalized* minimizers of the expected risk. i.e.,

$$f^*_{q,\lambda_q} := \operatorname*{argmin}_{f \in \mathcal{F}} R_q[f] + \lambda_q \Omega[f]$$

$$f^*_{p,\lambda_p} := \operatorname*{argmin}_{f \in \mathcal{F}} R_p[f] + \lambda_p \Omega[f] \tag{5}$$

The above quantities are needed since f^*_p and f^*_q might not necessarily have bounded norm in function classes that we study. Briefly, our algorithm outline is the following.

Step 1: Unweighted estimate Solve the unweighted inference problem using (X, Y) as training data to obtain $\hat{f}_{q,\lambda_q}$ (see Equation (2)).

Step 2: Covariate shift correction weights Using X and X' estimate the covariate shift correction weights. This can be done by any off-the-shelf (e.g. kernel mean matching) covariate shift procedure (Gretton et al. 2008; Agarwal et al. 2011).

Step 3: Doubly robust estimate If m_{eff} is much smaller than m, use unweighted estimate in Step 1 and covariate shift weights in Step 2 to obtain $\hat{f}_{DR}$ (see Equation 4).

Intuitively, while $\hat{f}_{q,\lambda_q}$ will not minimize the expected risk, it is often a very good proxy. Given that no reweighting was carried out, the variance for $\hat{f}_{q,\lambda_q}$ is comparatively low. That is, we are using the large unweighted sample size to obtain a good starting point with high confidence.

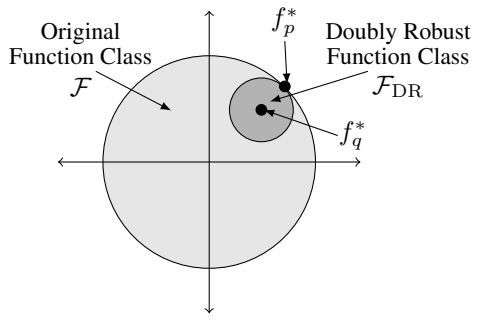

Figure 2: Pictorial representation of DR estimation procedure. Assumption 3 implies that f^*_q is close to f^*_p than origin (as shown in the figure). While the generic covariate shift finds the weighted empirical risk minimizer over the large function class $\mathcal{F}$, doubly robust procedure optimizes over a much smaller function class $\mathcal{F}_{\mathrm{DR}}$. This leads to small variance in doubly robust procedure as compared to generic covariate shift procedure when the effective sample size m_{eff} is small.

Assumptions

It is worth mentioning the assumptions required for the application of doubly robust estimation, since they motivate our design choices.

Assumption 1 *The conditional training and test distributions are identical i.e* $p(y|x) = q(y|x)$.

This is implicit in the definition of covariate shift — if $p(y|x) \neq q(y|x)$ it would be trivial to construct counterexamples for any algorithm attempting to solve covariate shift. For instance, setting $p(y|x) = q(-y|x)$ for binary classification would lead to a maximally bad solution.

Assumption 2 *$\beta(x)$ is well defined and bounded by some constant η. This ensures that there cannot exist sets of nonzero measures with respect to P that have zero measure with respect to Q.*

Again, in the absence of this assumption we could design pessimal algorithms. In this case we could, e.g., set $y|x = 0$ for all $x \notin S$ and $y|x = C$ for $x \in S$, immediately implying substantial misprediction regardless of the sample size.

Assumption 3 *The risk minimizer f^*_{p,λ_p} is much closer to the unweighted risk minimizer f^*_{q,λ_q} rather than the origin, i.e.,* $\nu_{\mathrm{DR}} = \Omega[f^*_{p,\lambda_p}, f^*_{q,\lambda_q}] \ll \Omega[f^*_{p,\lambda_p}] = \nu_p$.

The above assumption indicates that the unweighted solution is beneficial for solving the weighted solution. This assumption is reminiscent of approaches used in previous literature on domain adaptation (see Kuzborskij and Orabona 2013; Li and Bilmes 2007). Also, note that the assumption is *only relative* to the origin and does not assume anything about the absolute closeness of the weighted and unweighted solutions.

We would also like to emphasize that Assumption 3 does not trivially mean improved result. Note that we additionally need to estimate the unweighted solution, which can degrade the performance of the algorithm. However, the critical point we exploit is that the unweighted estimator, although biased,

has low variance since it does not involve reweighting the dataset. We will revisit this issue later in Section .

Estimating Covariate Shift Weights

Before delving into a specific algorithm we need to discuss means of obtaining estimates of $\beta(X)$. A number of approaches have been proposed in the literature. We only give a brief outline of a few approaches here and refer interested readers to the appropriate references for further details.

Penalized Risk Minimization (PRM) The basic idea in this approach is to estimate covariate shift weights β by solving a particular regularized convex minimization problem over a function class (Nguyen, Wainwright, and Jordan 2008). The rationale for the approach stems from the fact that the optima to the variational representation of KL-divergence is attained at the point $\beta(x) = \frac{p(x)}{q(x)}\ \forall x \in \mathcal{X}$. More specifically, consider the following variational representation of KL-divergence: $D(p,q) = \sup_{g>0} \int \log g(x)p(x)dx - \int g(x)q(x)dx + 1$. This is obtained by a simple application of Legendre-Frenchel convex duality (see (Nguyen, Wainwright, and Jordan 2008) for more details). More importantly for us, the supremum is attained at $g(x) = \beta(x) = p(x)/q(x)$. Let us assume that the function β belongs to RKHS $\mathcal{G}$. Since the access to distributions p and q is through their corresponding samples, we solve the following regularized empirical version of the problem:

$$\hat{\beta} = \operatorname*{argmin}_{g \in \mathcal{G}} \frac{1}{m}\sum_{i=1}^{m} g(x_i) - \frac{1}{m}\sum_{i=1}^{m} \log g(x_i') + \frac{\gamma_m}{2} I^2(g)$$

where $I(g)$ is a non-negative measure of complexity for g such that $I(\beta) < \infty$. It is shown that the above estimator enjoys good statistical properties. A more detailed theoretical exposition of the estimator will follow in later sections.

Kernel Mean Matching (KMM) Another popular approach to obtain the covariate shift weights is by matching the mean embeddings in the feature space induced by a universal RKHS $\mathcal{K}$ on the domain $\mathcal{X}$ (Gretton et al. 2008). More specifically, we solve the following optimization problem

$$\min_{\hat{\beta}} \hat{L}(\hat{\beta}) := \left\| \frac{1}{m}\sum_{i=1}^{m} \hat{\beta}_i \Phi(x_i) - \frac{1}{m}\sum_{i=1}^{m} \Phi(x_i') \right\|$$
$$\text{s.t. } 0 \le \hat{\beta}_i \le \eta \text{ and } \frac{1}{m}\sum_{i=1}^{m} \hat{\beta} = 1,$$

where $\Phi : \mathcal{X} \to \mathcal{K}$. Intuitively, the above procedure tries to match the mean embeddings of weighted training and test distributions. Since the RKHS is universal, matching the embeddings provides estimates for covariate shift weights β. As above, we delay the theoretical details. Note that while the first estimation procedure gives the function β, the KMM approach computes the function evaluated only at the training points. See e.g. (Agarwal, Li, and Smola 2011) for a detailed comparison to other approaches.

Examples

To gain a better understanding of our approach, we now present our estimators in various algorithmic settings. Let us assume, we have estimated covariate shift weights $\hat{\beta}$ via PRM, KMM or in general, any other method.

Regression The simplest setting is linear regression, possibly in a Reproducing Kernel Hilbert Space. Here the loss ℓ, the function f, and Ω are given by $f(x) = \langle w, \phi(x)\rangle$, $\ell(y, f(x)) = \frac{1}{2}(y - f(x))^2$ and $\Omega[f, f'] = \frac{1}{2}\|w - w'\|^2$, where $\phi(x)$ is a feature map. The three steps of doubly robust covariate shift correction are:

1. Solve the quadratic optimization problem below.
$$\hat{w}_{q,\lambda_q} = \arg\min_{w} \frac{1}{2}\sum_{i=1}^{m}(y_i - \langle \phi(x_i), w\rangle)^2 + \frac{\lambda_q}{2}\|w\|^2$$
2. Estimate the covariate shift correction weights $\hat{\beta}$.
3. Solve the centered weighted regression problem to obtain the doubly robust estimator $\hat{w}_{\mathrm{DR}}$.
$$\underset{w}{\text{minimize}} \frac{1}{2}\sum_{i=1}^{m} \hat{\beta}_i(y_i - \langle \phi(x_i), w\rangle)^2 + \frac{\lambda'}{2}\left\|w - \hat{w}_{q,\lambda_q}\right\|^2$$

The approach works whenever $\left\|w_p^* - w_q^*\right\| \ll \left\|w_p^*\right\|$, i.e. whenever the unbiased and the biased solutions are close compared to the overall complexity of the solutions.

SVM Classification The approach is quite analogous to the above approach, the main difference being a different loss function. This yields $f(x) = \langle w, \phi(x)\rangle$, $\ell(y, f(x)) = \max(0, 1 - yf(x))$, and $\Omega[f, f'] = \frac{1}{2}\|w - w'\|^2$. The associated algorithm is as follows:

1. Solve a standard SVM classification problem using X, Y to obtain $\hat{w}_{q,\lambda_q}$.
$$\min_{w} \sum_{i=1}^{m} \max(0, 1 - y_i f(x_i)) + \frac{\lambda_q}{2}\|w\|^2$$
2. Estimate the covariate shift correction weights $\hat{\beta}$.
3. Solve the centered weighted SVM classification problem to obtain the DR estimator $\hat{w}_{\mathrm{DR}}$.
$$\min_{w} \sum_{i=1}^{m} \hat{\beta}_i \max(0, 1 - y_i f(x_i)) + \frac{\lambda'}{2}\left\|w - \hat{w}_{q,\lambda_q}\right\|^2$$

Regression Tree The nontrivial challenge here is to define what it means to use an existing tree as a prior. We obtain the following algorithm:

1. Compute a Regression Tree $\hat{f}_{q,\lambda_q}$ using X, Y with suitable pruning strategy λ_q.
2. Estimate the covariate shift correction weights $\hat{\beta}$.
3. Compute the residuals $\epsilon_i := y_i - \hat{f}_{q,\lambda_q}(x_i)$. Train a second regression tree δf using $(x_i, \epsilon_i, \hat{\beta}_i)$ as covariates, labels, and sample weights. Output the corrected tree $\hat{f}_{\mathrm{DR}} := \hat{f}_{q,\lambda_q} + \delta f$.

Analogous modifications are possible for Gaussian Process estimates where we use stage 1 estimates as prior, or for neural networks. Given the generality, our analysis proceeds in two steps — we first derive a general metatheorem, followed by an application to kernel methods.

Theoretical Analysis

In this section we derive generalization bounds for the doubly robust estimation procedure and show that they are provably better than the standard covariate shift bounds under the conditions assumed in this paper. To this end, we develop a general framework for analyzing the doubly robust estimator and use it to prove generalization bounds for kernel methods. More precisely, we obtain upper bounds on risk R_p of functions, $\hat{f}_{p,\lambda_p}$ (standard covariate shift correction) and $\hat{f}_{DR}$ (doubly robust estimator).

Let $\mathcal{H}$ be a reproducing kernel Hilbert space associated with $\mathcal{X}$ and feature map $\phi(x) \in \mathcal{H}$. We use $\mathbf{K}$ denote the kernel matrix corresponding to the training points X.Let $\|\phi(x)\|_{\mathcal{H}} \leq \kappa$ for all $x \in \mathcal{X}$. Due to lack of space, we relegate the details of general framework and proofs to the appendix of the full version[2], and only state the result for kernel methods using covariate shift weights obtained through PRM in the main paper. The bounds for KMM can be obtained in a similar manner. We state the main results about generalization bounds for PRM, which follow as corollaries of our general framework.

Theorem 1 *Suppose $\hat{f}_{p,\lambda_p}$ and f^*_{p,λ_p} are as defined in Equations* (3) *and* (5) *respectively, and $\beta \in \mathcal{G}$. Let the regularization parameter for PRM be $\gamma_m = cm^{-2/(2+\tau)}$ for some $\tau > 0$ and a constant c. Then we have the following with probability at least $1 - \delta$.*

$$R_p[\hat{f}_{p,\lambda_p}] \leq R_p[f^*_{p,\lambda_p}] + \Delta_{W,S} + \Delta_{W,R}. \tag{6}$$

$\Delta_{W,S}$ and $\Delta_{W,R}$, representing the covariate shift and function complexity parts of the bound are:

$$\Delta_{W,S} = \frac{2\kappa^2 L^2}{\lambda}\left(\sqrt{\eta\gamma_m} + \eta\sqrt[4]{\frac{8}{m}\log\left(\frac{4}{\delta}\right)}\right)$$

$$\Delta_{W,R} = 2\eta L\left(\frac{2\nu}{m}\sqrt{\mathrm{tr}(\mathbf{K})} + 3\sqrt{\frac{1}{2m}\log\left(\frac{4}{\delta}\right)}\right)$$

Theorem 2 *Suppose $\hat{f}_{\mathrm{DR}}$ and f^*_{p,λ_p} are as defined in Equations* (4) *and* (5) *respectively, and $\beta \in \mathcal{G}$. Let the regularization parameter for PRM be $\gamma_m = cm^{-2/(2+\tau)}$ for some $\tau > 0$ and a constant c. Then we have the following with probability at least $1 - \delta$.*

$$R_p[\hat{f}_{\mathrm{DR}}] \leq R_p[f^*_{p,\lambda_p}] + \Delta_{\mathrm{DR},S} + \Delta_{\mathrm{DR},R}. \tag{7}$$

$\Delta_{\mathrm{DR},S}$ and $\Delta_{\mathrm{DR},R}$, denoting the covariate shift and function complexity parts of the bound are:

$$\begin{aligned}\nu' &= \nu_{DR} + \nu_\Delta \\ &= \nu_{DR} + \sqrt{\frac{4L}{\lambda_q}\left(\frac{2\nu_q\sqrt{\mathrm{tr}(\mathbf{K})}}{m} + 3\sqrt{\frac{\log(6/\delta)}{2m}}\right)}\end{aligned}$$

$$\Delta_{\mathrm{DR},S} = \frac{2\kappa^2 L^2}{\lambda'}\left(\sqrt{\eta\gamma_m} + \eta\sqrt[4]{\frac{8}{m}\log\left(\frac{6}{\delta}\right)}\right)$$

$$\Delta_{\mathrm{DR},R} = 2\eta L\left(\frac{2\nu'\sqrt{\mathrm{tr}(\mathbf{K})}}{m} + 3\sqrt{\frac{\log(6/\delta)}{2m}} + \frac{\|\hat{f}_{q,\lambda_q}\|_2}{m}\right)$$

[2]Full version of the paper can be found at www.cs.cmu.edu/~sjakkamr/dr.pdf.

Proof sketch for Theorem 2. The proof consists of two crucial components. First, we derive a uniform convergence result for $\hat{f}_{q,\lambda_q}$ relative to the expected risk minimizer f^*_{q,λ_q}. Second, we bound the error in risk caused due to estimation of covariate weights $\hat{\beta}$ and the complexity of the function class. Combining the bounds on $\hat{f}_{q,\lambda_q}$ relative to f^*_{q,λ_q} and error in estimation of covariate shift weights, we get the required result.

Discussion on the Generalization Bounds

In order to understand the benefit of our doubly robust estimator, we make a qualitative comparison of the various generalization bounds in this section. We only compare the bounds for PRM here, but analysis for KMM yields similar conclusions. From Assumption 3, we have $\nu' \ll \nu_p$ and expect $\lambda' \gg \lambda_p$, provided bound ν_Δ is small. When the variance of $\hat{f}_{q,\lambda_q}$ is small, it is easy to see that $\Delta_{\mathrm{DR},R} \ll \Delta_{W,R}$ and $\Delta_{\mathrm{DR},S} \ll \Delta_{W,S}$ (in Equations (6) and (7)). These bounds also clearly demonstrate the doubly robust nature of the algorithm. Before ending our discussion, we need to make it explicit that our analysis only compares the upper bounds and hence, needs to be interpreted with caution. Nonetheless, our empirical evaluation, in the next section, supports our theoretical analysis and provides a compelling case to use our estimators in practice.

Experiments

We present our empirical results in this section. We apply doubly robust covariate shift correction to a broad range of UCI datasets and a real-world dataset to demonstrate its performance. In particular, we show that it is effective both for classification and regression settings, and both for linear methods (by using a Support Vector Classifier) and nonlinear approaches (by using a Regression Tree).

For our experiments we compare the performance of unweighted (see Equation (2)) (referred to as UNWEIGHTED), weighted (see Equation (3))(referred to as WEIGHTED) and doubly robust (see Equation (4))(referred to as DOUBLYROBUST) empirical estimators. That is, UNWEIGHTED ignores the problem of covariate shift correction; WEIGHTED uses the weights computed by KLIEP (Sugiyama et al. 2008) with Gaussian kernel. For simplicity we use a reduced rank expansion with 100 basis functions in our experiments. The bandwidth of the kernel is chosen by cross-validation.

We would like to emphasize that while we only report results for KLIEP due to lack of space, using doubly robust estimation in conjunction with other popular approaches (e.g., (Gretton et al. 2008; Shimodaira 2000)) yields similar results.

Synthetic Data: This experiment is meant to provide a comparison of WEIGHTED and DOUBLYROBUST approaches when varying effective sample size m_{eff}. The data for this experiment is generated based on a polynomial objective $y = -x + x^3 + \epsilon$ where $\epsilon \sim \mathcal{N}(0, 0.3)$ (Gretton et al. 2008). We set $p(x) = \mathcal{N}(0, 1)$ and use as biasing distribution $p(x) = \mathcal{N}(\mu, 0.3)$ where μ is adjusted such that we obtain different effective samples sizes. 300

training and test samples are drawn. We use linear regression with standard ℓ_2 penalization. Figure 3 shows the root mean square error (RMSE) ratio of DOUBLYROBUST to WEIGHTED. It can be seen that DOUBLYROBUST outperforms WEIGHTED for lower values of m_{eff} and is marginally worse for higher values of m_{eff}. The latter is not surprising, since DOUBLYROBUST makes use of the data thrice rather than twice.

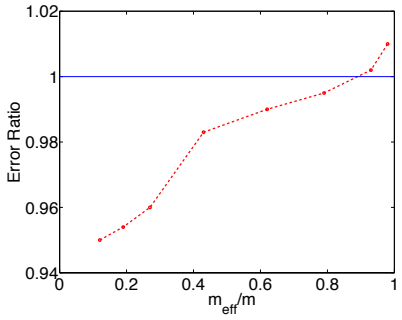

Figure 3: Comparison of WEIGHTED and DOUBLYROBUST using a synthetic dataset. We plot the error ratio as a function of the effective sample size. As can be seen, our method improves the most when the increase in variance is the highest. This is consistent with the fact that it acts as variance reducer.

Real Data: For a more realistic comparison we apply our method to several UCI[3] and benchmark[4] datasets. To control the amount of bias we use PCA to obtain the leading principal component. The projections onto the first principal component are then used to construct a subsampling distribution q. Let t_0 and t_1 be the minimum and the maximum of the projected values respectively. Let σ_{PC} be the standard deviation of the projected values. We then subsample using their projected values according to normal distribution $\mathcal{N}(t_0 + \alpha(t_1 - t_0), 0.5\sigma_{\text{PC}})$. Varying the value of α changes the m_{eff} of the training data by shifting q relative to p. The value $\alpha \in (0, 1)$ is independently set for each dataset in such a way that the effective sample size m_{eff} is less than 1/3 of the training data. This method of inducing covariate shift in the data set is often used in the covariate shift literature (see, e.g., (Gretton et al. 2008)).

For classification, we use support vector machines with a linear kernel. As mentioned earlier, $\Omega[f, f'] = \frac{1}{2}\|w - w'\|^2$, i.e., the correction is additive in feature space. The regularization parameters are chosen separately for each empirical estimator by cross validation. We report the classification error $\Pr\{yf(x) < 0\}$. We normalize the errors with the UNWEIGHTED error.

For regression we apply regression trees to several UCI datasets. We report the square error loss for these experiments. As explained earlier, we first train a regression tree on the unweighted dataset and then build a differential regression tree on the residual with restricted tree depth in order to train the doubly robust regression tree.

The results are reported in Figure 4. We report the average RMSE error and the standard deviation over 30 trials for each experiment. The errors in both the above cases are normalized by the error of UNWEIGHTED. In both the tasks, it can be clearly seen that DOUBLYROBUST outperforms both UNWEIGHTED and WEIGHTED on most of the datasets. Note that neither UNWEIGHTED nor WEIGHTED are significantly better than each other. On the other hand, our approach consistently outperforms both. This is in line with our intuition that the unweighted solution is an excellent variance reducer. Overall, we conclude that our method is promising for covariate shift correction problem.

Conclusion

In this paper we proposed an intuitive and easy-to-use strategy for improving covariate shift correction. It addresses a key issue that plagues many covariate shift correction algorithms, namely that the variance increases considerably whenever samples are reweighted. It achieves this goal by using the unweighted solution as a variance-reducing proxy for the unknown true weighted solution. This is a rather general strategy and has been used with great success, e.g. as control variate, in the context of reinforcement learning (Sutton and Barto 1998).

Our approach is particularly simple insofar as it requires essentially no additional code to use — all that is required in practice is to allow for reweighting and offset-correction in a linear model, a decision tree, or any other estimator that might be at hand. Of particular importance is the fact that we found our approach never to be worse than unweighted solution, something that cannot be said in general for covariate shift correction.

Acknowledgements

This work is supported in part by NSF Big Data grant IIS-1247658 and IIS-1250350.

References

Agarwal, D.; Li, L.; and Smola, A. J. 2011. Linear-time estimators for propensity scores. *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS-14)* 93–100.

Bang, H., and Robins, J. M. 2005. Doubly robust estimation in missing data and causal inference models. *Biometrics* 61:962–973.

Cortes, C.; Mohri, M.; Riley, M.; and Rostamizadeh., A. 2008. Sample selection bias correction theory. In *Proceedings of the 19th International Conference on Algorithmic Learning Theory*, 38–53.

Daume III, H. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 256–263. Association for Computational Linguistics.

Doucet, A.; de Freitas, N.; and Gordon, N. 2001. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag.

[3] http://archive.ics.uci.edu/ml/datasets.html

[4] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

classification	UNWEIGHTED	WEIGHTED	DOUBLYROBUST
hill	1.00 (± 0.05)	1.03 (± 0.04)	**0.98** (± 0.03)
splice	1.00 (± 0.01)	0.98 (± 0.01)	**0.97** (± 0.01)
german	1.00 (± 0.04)	1.08 (± 0.05)	**0.96** (± 0.03)
diabetes	1.00 (± 0.05)	0.89 (± 0.02)	**0.85** (± 0.01)
ionosphere	1.00 (± 0.04)	0.82 (± 0.01)	**0.79** (± 0.01)
cod-rna	1.00 (± 0.05)	1.05 (± 0.03)	**0.94** (± 0.02)
ijcnn	1.00 (± 0.03)	0.99 (± 0.02)	**0.96** (± 0.02)
breast-cancer	1.00 (± 0.03)	**0.96** (± 0.02)	0.97 (± 0.02)
fourclass	**1.00** (± 0.03)	1.04 (± 0.02)	1.03 (± 0.02)
australian	1.00 (± 0.04)	1.02 (± 0.03)	**0.97** (± 0.03)
sonar	1.00 (± 0.05)	0.98 (± 0.04)	**0.97** (± 0.04)
spambase	1.00 (± 0.05)	0.99 (± 0.03)	**0.98** (± 0.03)

regression	UNWEIGHTED	WEIGHTED	DOUBLYROBUST
abalone	1.00 (± 0.01)	0.97 (± 0.03)	**0.95** (± 0.01)
mg	1.00 (± 0.04)	1.04 (± 0.03)	**0.97** (± 0.03)
enuite	1.00 (± 0.04)	0.95 (± 0.03)	**0.93** (± 0.02)
space	1.00 (± 0.05)	0.98 (± 0.04)	**0.94** (± 0.03)
mpg	1.00 (± 0.03)	**0.93** (± 0.02)	0.94 (± 0.03)
bodyfat	1.00 (± 0.04)	**0.96** (± 0.03)	0.97 (± 0.03)
cadata	**1.00** (± 0.03)	1.11 (± 0.04)	1.03 (± 0.04)
housing	1.00 (± 0.02)	0.99 (± 0.04)	**0.97** (± 0.03)

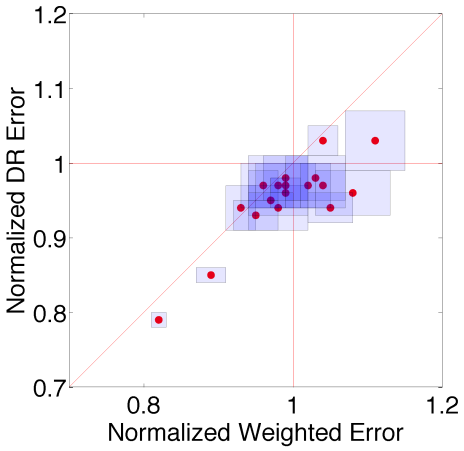

Figure 4: Relative performance of SVM classifiers and regression trees on UCI datasets. We normalize the unweighted performance to 1 and report relative variance. DOUBLYROBUST consistently outperforms other estimators. Error bars are obtained using 30 trials for each experiment. The graph on the RHS summarizes these results. We combine both regression and classification results since their behavior is entirely analogous. Boxes represent the extent of uncertainty, with a red solid dot in the middle. The points to the left of the vertical (resp. below the horizontal) line at 1 represent the cases where WEIGHTED (resp. DOUBLYROBUST) performs better than UNWEIGHTED. The points below straight diagonal line represent the cases where DOUBLYROBUST outperforms WEIGHTED. As can be seen, our method is much less susceptible to an increase in variance.

Dudík, M.; Langford, J.; and Li, L. 2011. Doubly robust policy evaluation and learning. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 1097–1104.

Gretton, A.; Smola, A. J.; Huang, J.; Schmittfull, M.; Borgwardt, K.; and Schölkopf, B. 2008. Dataset shift in machine learning. In *Covariate Shift and Local Learning by Distribution Matching*, 131–160.

Kang, J. D. Y., and Schafer, J. L. 2007. Demystifying double robustness: a comparison of alternative strategies for estimating a population mean from incomplete data. *Statistical science* 22(4):523–539.

Kuzborskij, I., and Orabona, F. 2013. Stability and hypothesis transfer learning. In *Proceedings of The 30th International Conference on Machine Learning (ICML-13)*, 942–950.

Li, X., and Bilmes, J. 2007. A Bayesian divergence prior for classifier adaptation. In *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics (AISTATS-2007)*, 275–282.

Nguyen, X. L.; Wainwright, M.; and Jordan, M. 2008. Estimating divergence functionals and the likelihood ratio by penalized convex risk minimization. In *Advances in Neural Information Processing Systems 20*, 1089–1096.

Quiñonero-Candela, J.; Sugiyama, M.; Schwaighofer, A.; and Lawrence, N. 2008. *Dataset Shift in Machine Learning*. MIT Press.

Shimodaira, H. 2000. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference* 90(2):227–244.

Sugiyama, M.; Nakajima, S.; Kashima, H.; von Bünau, P.; and Kawanabe, M. 2008. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems 20*, 1433–1440.

Sutton, R. S., and Barto, A. G. 1998. *Reinforcement Learning: An Introduction*. MIT Press.

Vapnik, V. 1998. *Statistical Learning Theory*. New York: John Wiley and Sons.

Wen, J.; nam Yu, C.; and Greiner, R. 2014. Robust learning under uncertain test distributions: Relating covariate shift to model misspecification. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 631–639.