Generalization Bounds for Transfer Learning under Model Shift

Xuezhi Wang
Computer Science Dept.
Carnegie Mellon University
Pittsburgh, PA 15213

Jeff Schneider
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Transfer learning (sometimes also referred to as domain-adaptation) algorithms are often used when one tries to apply a model learned from a fully labeled source domain, to an unlabeled target domain, that is similar but not identical to the source. Previous work on covariate shift focuses on matching the marginal distributions on observations X across domains while assuming the conditional distribution $P(Y|X)$ stays the same. Relevant theory focusing on covariate shift has also been developed. Recent work on transfer learning under model shift deals with different conditional distributions $P(Y|X)$ across domains with a few target labels, while assuming the changes are smooth. However, no analysis has been provided to say when these algorithms work. In this paper, we analyze transfer learning algorithms under the model shift assumption. Our analysis shows that when the conditional distribution changes, we are able to obtain a generalization error bound of $O(\frac{1}{\lambda_* \sqrt{n_l}})$ with respect to the labeled target sample size n_l, modified by the smoothness of the change (λ_*) across domains. Our analysis also sheds light on conditions when transfer learning works better than no-transfer learning (learning by labeled target data only). Furthermore, we extend the transfer learning algorithm from a single source to multiple sources.

1 INTRODUCTION

In a classical transfer learning setting (see Fig. 1), we have a source domain with sufficient fully labeled data, and a target domain with data that has little or no labels. These two domains are related but not identical, and the usual assumption is that there is some knowledge that can be transferred from the source domain to the target domain. Examples of transfer learning applied in the real-world include, adapting classification models for different products, and transferring across diseases on medical data (Pan et al. (2009)). A number of different transfer learning techniques have been introduced in the past, e.g., algorithms dealing with covariate shift (Shimodaira (2000), Huang et al. (2007), Gretton et al. (2007)). Related theoretical analyses on covariate shift have also been developed, e.g., for sample size m in the source domain and sample size n in the target domain, the analysis of Mansour et al. (2009) achieves a rate of $O(m^{-1/2} + n^{-1/2})$, and convergence of reweighted means in feature space achieves rate $O((1/m + 1/n)^{1/2})$ (Huang et al. (2007)).

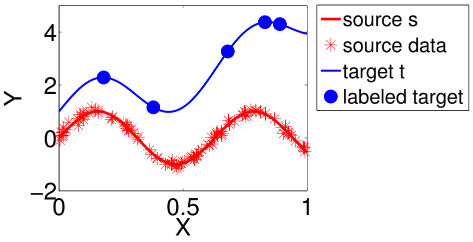

Figure 1: Transfer learning example: m source data points $\{X^s, Y^s\}$ (red), n target data points $\{X^t, Y^t\}$ (blue), and n_l labeled target points (solid blue circles). Here X denotes the input features and Y denotes the output labels.

However, not much work on transfer learning has considered the case when a few labels in the target domain are available. Also little work has been done when conditional distributions are allowed to change (defined as **model shift**). Recently, algorithms dealing with transfer learning under model shift have been proposed, where the changes on conditional distributions are assumed to be smooth (Wang et al. (2014)). However, no theoretical analysis has been provided for these approaches.

In this paper, we develop theoretical analysis for transfer learning algorithms under the model shift assumption. Our analysis shows that even when the conditional distributions are allowed to change across domains, we are still able to obtain a generalization bound of $O(\frac{1}{\lambda_* \sqrt{n_l}})$ with respect to the labeled target sample size n_l, modified by the smoothness of the transformation parameters (λ_*) across domains. Our analysis also sheds light on conditions when transfer learning works better than no-transfer learning. We show that under certain smoothness assumptions it is possible to obtain a favorable convergence rate with transfer learning compared to no transfer at all. Furthermore, using the generalization bounds we derived in this paper, we are able to extend the transfer learning algorithm from a single source to multiple sources, where each source is assigned a weight that indicates how helpful it is for transferring to the target.

We illustrate our theoretical results by empirical comparisons on both synthetic data and real-world data. Our results demonstrate cases where we obtain the same rate as no-transfer learning, and cases where we obtain a favorable rate with transfer learning under certain smoothness assumptions, which coincide with our theoretical analysis. In addition, experiments on the real data show that our algorithm for reweighting multiple sources yields better results than existing state-of-the-art algorithms.

2 RELATED WORK

Traditional methods for transfer learning use relatively restrictive assumptions, where specific parts of the learning model are assumed to be carried over between tasks. For example, Mihalkova et al. (2007) transfers relational knowledge across domains using Markov logic networks. Niculescu-Mizil & Caruana (2007) learns Bayes Net structures by biasing learning toward similar structures for each task. Do & Ng (2005) and Raina et al. (2006) assume that models for related tasks share same parameters or prior distributions of hyperparameters.

A large part of transfer learning work is devoted to the problem of covariate shift (Shimodaira (2000), Huang et al. (2007), Gretton et al. (2007)), where the assumption is that only the marginal distribution $P(X)$ differs across domains but the conditional distribution $P(Y|X)$ stays the same. The kernel mean matching (KMM) method (Huang et al. (2007), Gretton et al. (2007)), is one of the algorithms that deal with covariate shift. Huang et al. (2007) proved the convergence of reweighted means in the feature space, and showed that their method results in almost unbiased risk estimates. More recent research (Zhang et al. (2013)) focused on modeling target shift ($P(Y)$ changes), conditional shift ($P(X|Y)$ changes), and a combination of both. The assumption for target shift is that X depends causally on Y, thus $P(Y)$ can be re-weighted to match the distributions on X across domains. The authors also provided some theoretical analysis of the conditions when $P(X|Y)$ is identifiable. Both covariate shift and target/conditional shift make no use of target labels Y^t, even if some are available. For transfer learning under model shift, there could be a difference in $P(Y|X)$ that can not simply be captured by the differences in $P(X)$, hence neither covariate shift nor target/conditional shift will work well under the model shift assumption.

A number of theoretical analyses on domain adaptation have also been developed. Ben-David et al. (2006) presented VC-dimension-based generalization bounds for adaptation in classification tasks. Later Blitzer et al. (2007) extended the work with a bound on the error rate under a weighted combination of the source data. Mansour et al. (2009) introduced a discrepancy distance suitable for arbitrary loss functions and derived new generalization bounds for domain adaptation for a wide family of loss functions. However, most of the work mentioned above deals with domain adaptation under the covariate shift assumption, which means they still assume the conditional distribution stays the same across domains, or the labeling functions in the two domains share strong proximity in order for adaptation to be possible. For example, one of the bounds derived in Mansour et al. (2009) has a term $\mathcal{L}(h_Q^*, h_P^*)$ related to the average loss between the minimizer h_Q^* in the source domain and the minimizer h_P^* in the target domain, which could be fairly large when there exists a constant offset between the two labeling functions.

In Wang et al. (2014), the authors proposed a transfer learning algorithm to handle the general case where $P(Y|X)$ changes smoothly across domains. However, the authors fail to make explicit connections between the smoothness assumption and the generalization bounds for transfer learning. They do not show whether the performance will degrade when the smoothness assumption is relaxed, and whether the smoothness assumption yields a lower generalization error for transfer learning than no-transfer learning.

Similarly, most work in transfer learning with multiple sources focuses only on $P(X)$. For example, Mansour et al. (2008) proposed a distribution weighted combining rule of source hypotheses using the input distribution $P(X)$ for both source and target. This approach requires estimating the distribution $D_i(x)$ of source i on a target point x from large amounts of unlabeled points typically available from the source, which might be difficult in real applications with high-dimensional features. Other existing work focuses on finding the set of sources that are closely related to the target (Crammer et al. (2008)), or a reweighting of sources based on prediction errors (Yao and Doretto, (2010)). Chattopadhyay et al. (2011) proposed a conditional probability based weighting scheme under a joint optimization framework, which leads to a reweighting of sources that prefers more consistent predictions on the target. However, these existing approaches do not consider the problem that there might exist shifts in the conditional distribution from source to the target, and how the smoothness of this shift can help in learning the target, which is the main issue addressed in this paper.

3 TRANSFER LEARNING UNDER MODEL SHIFT: A REVIEW OF THE ALGORITHMS

Notation: Let $\mathcal{X} \in \mathcal{R}^d$ and $\mathcal{Y} \in \mathcal{R}$ be the input and output space for both the source and the target domain. We are given a set of m labeled data points, $(x_i^s, y_i^s) \in (X^s, Y^s), i = 1, \ldots, m$, from the source domain. We are also given a set of n target data points, X^t, from the target domain. Among these we have n_l labeled target data points, denoted as (X^{tL}, Y^{tL}). The unlabeled part of X^t is denoted as X^{tU}, with unknown labels Y^{tU}. For simplicity let $z \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ denote the pair of (x, y), and we use z^s, z^t, z^{tL} for the source, target, and labeled target, correspondingly. We assume X^s, X^t are drawn from the same $P(X)$ throughout the paper since we focus more on $P(Y|X)$[1]. If necessary $P(X)$ can be easily matched by various methods dealing with covariate shift (e.g. Kernel Mean Matching) without the use of Y.

Let $\mathcal{H}$ be a reproducing kernel Hilbert space with kernel K such that $K(x, x) \leq \kappa^2 < \infty$ for all $x \in X$. Let $||.||_k$ denote the corresponding RKHS norm. Let ϕ denote the feature mapping on x associated with kernel K, and $\Phi(X)$ denote the matrix where the i-th column is $\phi(x_i)$. Denote $K_{XX'}$ as the kernel computed between matrix X and X', i.e., $K_{ij} = k(x_i, x'_j)$. When necessary, we use ψ to denote the feature map on y, and the corresponding matrix as $\Psi(Y)$. For a hypothesis $h \in \mathcal{H}$, assume that $|h(x)| \leq M$ for some $M > 0$. Also assume bounded label set $|y| \leq M$. We use $\ell 2$ loss as the loss function $l(h(x), y)$ throughout this paper, which is σ-admissible, i.e.,

$$\forall x, y, \forall h, h', |l(h(x), y) - l(h'(x), y)| \leq \sigma |h(x) - h'(x)|. \tag{1}$$

It is easy to see that $\sigma = 4M$ for bounded $h(x)$ and y. Note the loss function is also bounded, $l(h(x), y) \leq 4M^2$.

Next we will briefly review two algorithms introduced in Wang et al. (2014) that handle transfer learning under model shift: the first is conditional distribution matching, and the second is two-stage offset estimation.

(1) Conditional Distribution Matching (CDM).

The basic idea of CDM is to match the conditional distributions $P(Y|X)$ for the source and the target domain. Since there is a difference in $P(Y|X)$ across domains, these two conditional distributions cannot be matched directly. Therefore, the authors propose to make a parameterized-location-scale transform on the source labels Y^s:

$$Y^{new} = Y^s \odot \mathbf{w}(X^s) + \mathbf{b}(X^s),$$

where $\mathbf{w}$ denotes the scale transform, $\mathbf{b}$ denotes the location transform, and $\odot$ denotes the Hadamard (elementwise) product. $\mathbf{w}$ and $\mathbf{b}$ are non-linear functions of X which allows a non-linear transform from Y^s to Y^{new}.

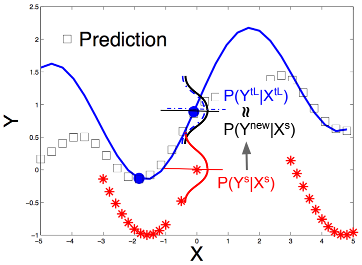

Figure 2: Illustration of the conditional distribution matching algorithm: red (source), blue (target).

The objective is to use the transformed conditional distribution in the source domain $P(Y^{new}|X^s)$, to match the conditional distribution in the target domain, $P(Y^{tL}|X^{tL})$, such that the transformation parameter $\mathbf{w}$ and $\mathbf{b}$ can be learned through optimization. The matching on $P(Y|X)$ is achieved by minimizing the discrepancy of the mean embedding of $P(Y|X)$ with a regularization term:

$$\begin{aligned} &\min_{\mathbf{w},\mathbf{b}} L + L_{reg}, \text{where} \\ &\quad L = ||\hat{\mathcal{U}}[P_{Y^{new}|X^s}] - \hat{\mathcal{U}}[P_{Y^{tL}|X^{tL}}]||_k^2, \\ &\quad L_{reg} = \lambda_{reg}(||\mathbf{w} - \mathbf{1}||^2 + ||\mathbf{b}||^2), \end{aligned} \tag{2}$$

where $\mathcal{U}[P_{Y|X}]$ is the mean embedding of the conditional distribution $P(Y|X)$ (Song et al. (2009)), and $\hat{\mathcal{U}}[P_{Y|X}]$ is the empirical estimation of $\mathcal{U}[P_{Y|X}]$ based on samples X, Y. Further the authors make a smoothness assumption on the transformation, i.e., $\mathbf{w}, \mathbf{b}$ are parameterized using: $\mathbf{w} = R\mathbf{g}, \mathbf{b} = R\mathbf{h}$, where $R = K_{X^sX^s}(K_{X^sX^s} + \lambda_R I)^{-1}$, and $\mathbf{g}, \mathbf{h} \in \mathbb{R}^{m \times 1}$ are the new parameters to optimize in the objective. After obtaining $\mathbf{g}, \mathbf{h}$ (or equivalently $\mathbf{w}, \mathbf{b}$), Y^{new} is computed based on the transformation. Finally the prediction on X^{tU} is based on the merged data: $(X^s, Y^{new}) \cup (X^{tL}, Y^{tL})$.

Fig 2 shows an illustration of the conditional distribution matching algorithm. As we can see from the figure, Y^s is transformed to Y^{new} such that $P(Y^{new}|X^s)$ and $P(Y^{tL}|X^{tL})$ can be approximately matched together.

Remark. Here we analyze what happens when the smoothness assumption is relaxed. It is easy to derive that, when setting $\mathbf{w} = \mathbf{1}, \mathbf{b} = \mathbf{0}$, we can directly solve for Y^{new} by taking the derivative of L with respect to Y^{new}, and we get:

$$\begin{aligned} &K_{X^sX^s}(K_{X^sX^s} + \lambda I)^{-1} Y^{new} \\ &= K_{X^sX^{tL}}(K_{X^{tL}X^{tL}} + \lambda I)^{-1} Y^{tL}, \end{aligned} \tag{3}$$

where λ is some regularization parameter to make sure the kernel matrix is invertible. In other words, the smoothed Y^{new} is given by the prediction on the source using only labeled target data. Hence Y^{new} provides no extra information for prediction on the target, compared with using the labeled target data alone.

[1] This assumption is only required in our analysis for simplicity. It can be relaxed when applying the algorithms.

(2) Two-stage Offset Estimation (Offset).

The idea of Offset is to model the target function f^t using the source function f^s and an offset, $f^o = f^t - f^s$, while assuming that the offset function is smoother than the target function. Specifically, using kernel ridge regression (KRR) to estimate all three functions, the algorithm works as follows:
(1) Model the source function using the source data, i.e., $f^s(x) = K_{xX^s}(K_{X^sX^s} + \lambda I)^{-1}Y^s$.
(2) Model the offset function by the difference between the true target labels and the predicted target labels, i.e., $f^o(X^{tL}) = Y^{tL} - f^s(X^{tL})$.
(3) Transform Y^s to Y^{new} by adding the offset, i.e., $Y^{new} = Y^s + f^o(X^s)$, where $f^o(X^s) = K_{X^sX^{tL}}(K_{X^{tL}X^{tL}} + \lambda I)^{-1}f^o(X^{tL})$.
(4) Train a model on $\{X^s, Y^{new}\} \cup \{X^{tL}, Y^{tL}\}$, and use the model to make predictions on X^{tU}.

We would like to answer: under what conditions these transfer learning algorithms will work better than no-transfer learning, and how the smoothness assumption affects the generalization bounds for these algorithms.

4 ANALYSIS OF CONDITIONAL DISTRIBUTION MATCHING

In this section, we analyze the generalization bound for the conditional distribution matching (**CDM**) approach.

4.1 RISK ESTIMATES FOR CDM

We use stability analysis on the algorithm to estimate the generalization error. First we have:

Theorem 1. *(Bousquet & Elisseeff (2002), Theorem 12 and Example 3) Consider a training set $S = \{z_1 = (x_1, y_1), ..., z_m = (x_m, y_m)\}$ drawn i.i.d. from an unknown distribution D. Let l be the $\ell 2$ loss function which is σ-admissible with respect to $\mathcal{H}$, and $l \leq 4M^2$. The Kernel Ridge Regression algorithm defined by:*

$$A_S = \arg\min_{h \in \mathcal{H}} \frac{1}{m}\sum_{i=1}^{m} l(h, z_i) + \lambda||h||_k^2$$

has uniform stability β with respect to l with $\beta \leq \frac{\sigma^2\kappa^2}{2\lambda m}$.

In addition, let $R = \mathbb{E}_z[l(A_S, z)]$ be the generalization error, and $R_{emp} = \frac{1}{m}\sum_{i=1}^{m} l(A_S, z_i)$ be the empirical error, then the following holds with probability at least $1 - \delta$,

$$R \leq R_{emp} + \frac{\sigma^2\kappa^2}{\lambda m} + (\frac{2\sigma^2\kappa^2}{\lambda} + 4M^2)\sqrt{\frac{\ln(1/\delta)}{2m}}.$$

In CDM, the prediction on the unlabeled target data points is given by merging the transformed source data and the labeled target data, i.e., $(X^s, Y^{new}) \cup (X^{tL}, Y^{tL})$. Hence we need to bound the difference between the empirical error on the merged data and the generalization error (risk) in the target domain.

Denote $\tilde{z}_i = (\tilde{x}_i, \tilde{y}_i) \in (\tilde{X}, \tilde{Y})$, where $\tilde{X}, \tilde{Y}$ represents the merged data: $\tilde{X} = X^s \cup X^{tL}, \tilde{Y} = Y^{new} \cup Y^{tL}$. Let $h^* \in H$ be the minimizer on the merged data, i.e.,

$$h^* = \arg\min_{h \in \mathcal{H}} \frac{1}{m + n_l}\sum_{i=1}^{m+n_l} l(h, \tilde{z}_i) + \lambda||h||_k^2.$$

Then the following theorem holds:

Theorem 2. *Assume the conditions in Theorem 1 hold. Also assume $||\hat{\mathcal{U}}[P_{Y^{new}|X^s}] - \hat{\mathcal{U}}[P_{Y^{tL}|X^{tL}}]||_k \leq \epsilon$ after we optimize objective Eq. 2. The following holds with probability at least $1 - \delta$:*

$$\begin{aligned}&|\frac{1}{m + n_l}\sum_{i=1}^{m+n_l} l(h^*, \tilde{z}_i) - E_{z^t}[l(h^*, z^t)]| \\ &\quad \leq 4M(\epsilon\kappa + C(\lambda_c^{1/2} + (n_l\lambda_c)^{-1/2})) + \\ &\qquad \frac{\sigma^2\kappa^2}{\lambda_t(m + n_l)} + (\frac{2\sigma^2\kappa^2}{\lambda_t} + 4M^2)\sqrt{\frac{\ln(1/\delta)}{2(m + n_l)}},\end{aligned}$$

where λ_c is the regularization parameter used in estimating $\hat{\mathcal{U}}[P_{Y^{tL}|X^{tL}}] = \Psi(Y^{tL})(K_{X^{tL}X^{tL}} + \lambda_c n_l I)^{-1}\Phi^\top(X^{tL})$, and λ_t is the regularization parameter when estimating the target function. $C > 0$ is some constant.

Proof. Let $\bar{z}_i = (\bar{x}_i, \bar{y}_i) \in (\bar{X}, \bar{Y})$, where $\bar{X}, \bar{Y}$ are the auxiliary samples with $\bar{X} = X^s \cup X^{tL}, \bar{Y} = \bar{Y}_s^t \cup Y^{tL}$, where $\bar{Y}_s^t$ are pseudo labels in the target domain for the source data points X^s. Using triangle inequality we can decompose the LHS by:

$$\begin{aligned}&|\frac{1}{m + n_l}\sum_{i=1}^{m+n_l} l(h^*, \tilde{z}_i) - E_{z^t}[l(h^*, z^t)]| \\ &\leq |\frac{1}{m + n_l}\sum_{i=1}^{m+n_l} l(h^*, \tilde{z}_i) - \frac{1}{m + n_l}\sum_{i=1}^{m+n_l} l(h^*, \bar{z}_i)| \\ &+ |\frac{1}{m + n_l}\sum_{i=1}^{m+n_l} l(h^*, \bar{z}_i) - E_{z^t}[l(h^*, z^t)]|\end{aligned}$$

The second term is easy to bound since it is simply the difference between the empirical error and the generalization error in the target domain with effective sample size $n_l + m$, thus using Theorem 1, we have

$$\begin{aligned}&|\frac{1}{m + n_l}\sum_{i=1}^{m+n_l} l(h^*, \bar{z}_i) - E_{z^t}[l(h, z^t)]| \\ &\leq \frac{\sigma^2\kappa^2}{\lambda_t(m + n_l)} + (\frac{2\sigma^2\kappa^2}{\lambda_t} + 4M^2)\sqrt{\frac{\ln(1/\delta)}{2(m + n_l)}}.\end{aligned} \tag{4}$$

To bound the first term, we have

$$
\begin{aligned}
&|\frac{1}{m+n_l}\sum_{i=1}^{m+n_l} l(h^*,\tilde{z}_i) - \frac{1}{m+n_l}\sum_{i=1}^{m+n_l} l(h^*,\bar{z}_i)| \\
&\leq \frac{1}{m+n_l}\sum_{i=1}^{m+n_l} |l(h^*,\tilde{z}_i) - l(h^*,\bar{z}_i)| \\
&\leq \frac{1}{m+n_l}\sum_{i=1}^{m} 4M|y_i^{new} - \mathcal{U}[P_{Y^t|X^t}]\phi(x_i^s)| \\
&\leq \frac{4M}{m+n_l}\sum_{i=1}^{m}(|\hat{\mathcal{U}}[P_{Y^{new}|X^s}]\phi(x_i^s) - \hat{\mathcal{U}}[P_{Y^{tL}|X^{tL}}]\phi(x_i^s)| \\
&\qquad + |\hat{\mathcal{U}}[P_{Y^{tL}|X^{tL}}]\phi(x_i^s) - \mathcal{U}[P_{Y^t|X^t}]\phi(x_i^s)|) \\
&\leq \frac{4M}{m+n_l}\sum_{i=1}^{m}(||\hat{\mathcal{U}}[P_{Y^{new}|X^s}] - \hat{\mathcal{U}}[P_{Y^{tL}|X^{tL}}]||_k\sqrt{k(x,x)} \\
&\qquad + |\hat{\mathcal{U}}[P_{Y^{tL}|X^{tL}}]\phi(x_i^s) - \mathcal{U}[P_{Y^t|X^t}]\phi(x_i^s)|) \\
&\leq 4M(\epsilon\kappa + C(\lambda_c^{1/2} + (n_l\lambda_c)^{-1/2})),
\end{aligned}
\tag{5}
$$

where in the last inequality, the second term is bounded using Theorem 6, Song et al. (2009).

Now combining Eq. 5 and Eq. 4 concludes the proof. □

4.2 TIGHTER BOUNDS UNDER SMOOTH PARAMETERIZATION

Theorem 2 suggests that using CDM, the empirical risk converges to the expected risk at a rate of

$$O(\lambda_c^{1/2} + (n_l\lambda_c)^{-1/2} + \lambda_t^{-1}(m+n_l)^{-1/2}). \tag{6}$$

In the following, we show how the smoothness parameterization in CDM helps us obtain faster convergence rates.

Under the smoothness assumption on the transformation, $\mathbf{w}, \mathbf{b}$ are parameterized using: $\mathbf{w} = R\mathbf{g}, \mathbf{b} = R\mathbf{h}$, where $R = K_{X^sX^s}(K_{X^sX^s} + \lambda_R I)^{-1}$. For simplicity we assume the same λ_R for both $\mathbf{w}$ and $\mathbf{b}$. Similar to the derivation in Eq. 5, we have

$$
\begin{aligned}
&|y_i^{new} - \mathcal{U}[P_{Y^t|X^t}]\phi(x_i^s)| \\
&= |\hat{\mathcal{U}}[P_{Y^{new}|X^s}]\phi(x_i^s) - \hat{\mathcal{U}}[P_{Y^{tL}|X^{tL}}]\phi(x_i^s)| \\
&\quad + |\hat{\mathcal{U}}[P_{Y^{tL}|X^{tL}}]\phi(x_i^s) - \mathcal{U}[P_{Y^t|X^t}]\phi(x_i^s)|) \\
&\leq \epsilon\kappa + |\hat{\mathcal{U}}[P_{w^{tL}|X^{tL}}]\phi(x_i^s) - \mathcal{U}[P_{w^t|X^t}]\phi(x_i^s)| \cdot |y_i^s| \\
&\quad + |\hat{\mathcal{U}}[P_{b^{tL}|X^{tL}}]\phi(x_i^s) - \mathcal{U}[P_{b^t|X^t}]\phi(x_i^s)| \\
&\leq \epsilon\kappa + C_1(\lambda_R^{1/2} + (n_l\lambda_R)^{-1/2})M + C_2(\lambda_R^{1/2} + (n_l\lambda_R)^{-1/2}) \\
&\leq \epsilon\kappa + C'(\lambda_R^{1/2} + (n_l\lambda_R)^{-1/2}).
\end{aligned}
\tag{7}
$$

Hence we can update the bound in Eq. 5 by:

$$
\begin{aligned}
&|\frac{1}{m+n_l}\sum_{i=1}^{m+n_l} l(h^*,\tilde{z}_i) - \frac{1}{m+n_l}\sum_{i=1}^{m+n_l} l(h^*,\bar{z}_i)| \\
&\leq 4M(\epsilon\kappa + C'(\lambda_R^{1/2} + (n_l\lambda_R)^{-1/2})).
\end{aligned}
\tag{8}
$$

It is easy to see that Eq. 4 remains the same. Hence, the rate for CDM under the smooth parametrization is:

$$O(\lambda_R^{1/2} + (n_l\lambda_R)^{-1/2} + \lambda_t^{-1}(m+n_l)^{-1/2}). \tag{9}$$

In transfer learning we usually assume the number of source data is sufficient, i.e., $m \to \infty$. Comparing Eq. 9 with Eq. 6 we can see that, when the number of labeled points n_l is small, the term $(n_l\lambda_c)^{-1/2}$ in Eq. 6 and the term $(n_l\lambda_R)^{-1/2}$ in Eq. 9 take over. If we further assume that the transformation $\mathbf{w}$ and $\mathbf{b}$ are smoother functions with respect to X than the target function with respect to X, i.e., $\lambda_R > \lambda_c$, then Eq. 9 is more favorable. On the other hand, when the number of labeled target points n_l is large enough for the first term $\lambda_c^{1/2}$ in Eq. 6 and the first term $\lambda_R^{1/2}$ in Eq. 9 to take over, then it is reasonable to use a λ_R closer to λ_c to get a similar convergence rate as in Eq. 6. Intuitively, when the number of labeled target points is large enough, it is not very helpful to transfer from the source for target prediction.

Remark. Note that in Eq. 6 and Eq. 9, an ideal choice of λ close to $1/\sqrt{n_l}$ can minimize $\lambda^{1/2} + (n_l\lambda)^{-1/2}$. However, note that the generalization bound is the difference between the expected risk R and the empirical risk R_{emp}, and a λ that minimizes the generalization bound does not necessarily minimize the expected risk R, since the empirical risk R_{emp} (which is also affected by λ) can still be large. To obtain a relatively small empirical risk, λ should be determined by the smoothness of the offset/target function, since it is the regularization parameter when estimating the offset/target. In practice λ is chosen by cross validation on the labeled data, and is not necessarily close to $1/\sqrt{n_l}$. For example, on real data we find that λ is usually chosen to be in the range of $1e-2$ to $1e-4$ to accommodate a fairly wide range of functions, which makes the second term $1/\sqrt{n_l\lambda}$ dominate the risk if n_l is much smaller than $1e4$.

4.2.1 Connection with Domain Adaptation Learning Bounds

In Mansour et al. (2009), the authors provided several bounds on the pointwise difference of the loss for two different hypothesis (Theorem 11, 12 and 13). It is worth noting that in order to bound the pointwise loss, the authors make the following assumptions when the labeling function f_S (source) and f_T (target) are potentially different:

$$\delta^2 = L_{\hat{S}}(f_S(x), f_T(x)) \ll 1,$$

where $L_{\hat{S}}(f_S(x), f_T(x)) = \mathbb{E}_{\hat{S}(x)} l(f_S(x), f_T(x))$. This condition is easily violated under the model shift assumption, where the two labeling functions can differ by a large margin. However, with our transformation from Y^s to Y^{new}, we can translate the above assumption to the following equivalent condition:

$$\delta^2 = L_{\hat{S}}(Y^{new}, f_T(x)) = \frac{1}{m}\sum_{i=1}^{m}(y_i^{new} - \mathcal{U}[P_{Y^t|X^t}]\phi(x_i^s))^2$$
$$\leq (\epsilon\kappa + C'(\lambda_R^{1/2} + (n_l\lambda_R)^{-1/2}))^2,$$

using the results in Eq. 7. Hence we can bound δ^2 to be small under reasonable assumptions on n_l and λ_R.

4.2.2 Comparing with No-transfer Learning

Without transfer, which means we predict on the unlabeled target set based merely on the labeled target set, the generalization error bound is simply: $|\frac{1}{n_l}\sum_{i=1}^{n_l} l(h^{tL}, z_i^{tL}) - E_{z^t}[l(h^{tL}, z^t)]| \leq \frac{\sigma^2\kappa^2}{\lambda_t n_l} + (\frac{2\sigma^2\kappa^2}{\lambda_t} + 4M^2)\sqrt{\frac{\ln(1/\delta)}{2n_l}}$, where h^{tL} is the KRR minimizer on $\{X^{tL}, Y^{tL}\}$. Then

$$E_{z^t}[l(h^{tL}, z^t)] - \frac{1}{n_l}\sum_{i=1}^{n_l} l(h^{tL}, z_i^{tL}) = O(\frac{1}{\lambda_t\sqrt{n_l}}). \quad (10)$$

We can see that with transfer learning, first we obtain a faster rate $O(\lambda_t^{-1}(m + n_l)^{-1/2})$ in Eq. 9 with effective sample size $n_l + m$ than $O(\lambda_t^{-1}n_l^{-1/2})$ in Eq. 10 with effective sample size n_l. However, the transfer-rate Eq. 9 comes with a penalty term $O(\lambda_R^{1/2} + (n_l\lambda_R)^{-1/2})$ which captures the estimation error between the transformed labels and the true target labels. Again, in transfer learning usually we assume $m \to \infty$, and n_l is relatively small, then the transfer-rate becomes $O((n_l\lambda_R)^{-1/2})$. Further if we assume that the smoothness parameter λ_R for the transformation is larger than the smoothness parameter λ_t for the target function ($\lambda_R > \lambda_t$ will be sufficient if $\lambda_R < 1$, otherwise we need to set $\lambda_R > \lambda_t^2$ if $\lambda_R \geq 1$), then we obtain a faster convergence rate with transfer than no-transfer. We will further illustrate the results by empirical comparisons on synthetic and real data in the experimental section.

5 ANALYSIS ON THE OFFSET METHOD

In this section, we analyze the generalization error on the two-stage offset estimation (**Offset**) approach. Interestingly, our analysis shows that the generalization bounds for offset and CDM have the same dependency on n_l.

5.1 RISK ESTIMATES FOR OFFSET

(1) First, we learn a model from the source domain by minimizing the squared loss on the source data, i.e.,

$$h^s = \arg\min_{h\in\mathcal{H}} \frac{1}{m}\sum_{i=1}^{m} l(h, z_i^s) + \lambda_s||h||_k^2.$$

Using Theorem 1, we have with probability at least $1-\delta$,

$$R^s \leq R^s_{emp} + \frac{\sigma^2\kappa^2}{\lambda_s m} + (\frac{2\sigma^2\kappa^2}{\lambda_s} + 4M^2)\sqrt{\frac{\ln(1/\delta)}{2m}},$$

where $R^s = E_{z^s}[l(h^s, z^s)], R^s_{emp} = \frac{1}{m}\sum_{i=1}^{m} l(h^s, z_i^s)$. Hence

$$R^s - R^s_{emp} = O(\frac{1}{\lambda_s\sqrt{m}}), \quad (11)$$

(2) Second, we learn the offset by KRR on $\{X^{tL}, \hat{y}^o\}$, where $\hat{y}^o = Y^{tL} - f^s(X^{tL})$, i.e., $\hat{y}^o$ is the estimated offset on labeled target points X^{tL}, and $f^s(X^{tL})$ is the prediction on X^{tL} using source data.

Denote $\hat{h}^o$ as the minimizer on $\hat{z}^o = \{X^{tL}, \hat{y}^o\}$, i.e.,

$$\hat{h}^o = \arg\min_{h\in\mathcal{H}} \frac{1}{n_l}\sum_{i=1}^{n_l} l(h, \hat{z}_i^o) + \lambda_o||h||_k^2$$
$$= \arg\min_{h\in\mathcal{H}} R(h) + N(h). \quad (12)$$

Denote h^o as the minimizer on $z^o = \{X^{tL}, y^o\}$, where y^o is the unknown true offset:

$$h^o = \arg\min_{h\in\mathcal{H}} \frac{1}{n_l}\sum_{i=1}^{n_l} l(h, z_i^o) + \lambda_o||h||_k^2$$
$$= \arg\min_{h\in\mathcal{H}} R'(h) + N(h), \quad (13)$$

Using Theorem 1, we have with probability at least $1-\delta$,

$$R^o \leq R^o_{emp} + \frac{\sigma^2\kappa^2}{\lambda^o n_l} + (\frac{2\sigma^2\kappa^2}{\lambda^o} + 4M^2)\sqrt{\frac{\ln(1/\delta)}{2n_l}} \quad (14)$$

where $R^o = E_{z^o}[l(h^o, z^o)], R^o_{emp} = \frac{1}{n_l}\sum_{i=1}^{n_l} l(h^o, z_i^o)$. In our estimation we use $\hat{y}^o$ instead of y^o, hence we need to account for this estimation error.

Lemma 1. *The generalization error R^o is bounded by:*

$$R^o = \bar{R}^o_{emp} + O(\frac{1}{\lambda_o\sqrt{n_l}}), \quad (15)$$

as $m \to \infty$. Here $\bar{R}^o_{emp} = \frac{1}{n_l}\sum_{i=1}^{n_l} l(\hat{h}^o, \hat{z}_i^o)$ is the empirical error of our estimator $\hat{h}^o$ on $\{X^{tL}, \hat{y}^o\}$.

Proof. Define the Bregman Divergence associated to F of f to g by $B_F(f||g) = F(f) - F(g) - < f - g, \nabla F(g) >$. Let $F(h) = R(h) + N(h), F'(h) = R'(h) + N(h)$. Since $h^o, \hat{h}^o$ are the minimizers, we have $B_{F'}(\hat{h}^o||h^o) + B_F(h^o||\hat{h}^o) = F'(\hat{h}^o) - F'(h^o) + F(h^o) - F(\hat{h}^o) = R'(\hat{h}^o) - R'(h^o) + R(h^o) - R(\hat{h}^o)$. In addition, using the nonnegativity of B and $B_F = B_R + B_N$, $B_{F'} = B_{R'} + B_N$, we have $B_N(\hat{h}^o||h^o) + B_N(h^o||\hat{h}^o) \leq B_F(h^o||\hat{h}^o) + B_{F'}(\hat{h}^o||h^o)$. Combining the two we have $B_N(\hat{h}^o||h^o) + B_N(h^o||\hat{h}^o) \leq R'(\hat{h}^o) - R'(h^o) +$

$R(h^o) - R(\hat{h}^o) = \frac{1}{n_l}\sum_{i=1}^{n_l} l(\hat{h}^o, z_i^o) - \frac{1}{n_l}\sum_{i=1}^{n_l} l(h^o, z_i^o) + \frac{1}{n_l}\sum_{i=1}^{n_l} l(h^o, \hat{z}_i^o) - \frac{1}{n_l}\sum_{i=1}^{n_l} l(\hat{h}^o, \hat{z}_i^o) \leq \frac{2}{n_l}\sum_{i=1}^{n_l} \sigma|y_i^o - \hat{y}_i^o|$, using $|l(\hat{h}^o, z_i^o) - l(\hat{h}^o, \hat{z}_i^o)| \leq |2\hat{h}^o(x_i) - y_i^o - \hat{y}_i^o| \cdot |y_i^o - \hat{y}_i^o| \leq \sigma|y_i^o - \hat{y}_i^o|, \sigma = 4M$.

Since for RKHS norm $B_N(f||g) = ||f - g||_k^2$, we have $B_N(\hat{h}^o||h^o) + B_N(h^o||\hat{h}^o) = 2||h^o - \hat{h}^o||_k^2$. Combined with the above inequality, we have $2||h^o - \hat{h}^o||_k^2 \leq \frac{2}{n_l}\sum_{i=1}^{n_l} \sigma|y_i^o - \hat{y}_i^o|$. Then we have $|l(h^o, z_i^o) - l(\hat{h}^o, z_i^o)| \leq \sigma|h^o(x_i) - \hat{h}^o(x_i)| \leq \sigma||h^o - \hat{h}^o||_k \kappa \leq \sigma\kappa\sqrt{\frac{1}{n_l}\sum_{i=1}^{n_l} \sigma|y_i^o - \hat{y}_i^o|}$. Hence $|\frac{1}{n_l}\sum_{i=1}^{n_l} l(h^o, z_i^o) - \frac{1}{n_l}\sum_{i=1}^{n_l} l(\hat{h}^o, \hat{z}_i^o)| \leq \frac{1}{n_l}\sum_{i=1}^{n_l}[|l(h^o, z_i^o) - l(\hat{h}^o, z_i^o)| + |l(\hat{h}^o, z_i^o) - l(\hat{h}^o, \hat{z}_i^o)|] \leq \sigma\kappa\sqrt{\frac{1}{n_l}\sum_{i=1}^{n_l} \sigma|y_i^o - \hat{y}_i^o|} + \frac{1}{n_l}\sum_{i=1}^{n_l} \sigma|y_i^o - \hat{y}_i^o|$. Now we can conclude that

$$R_{emp}^o = \frac{1}{n_l}\sum_{i=1}^{n_l} l(h^o, z_i^o) \leq \bar{R}_{emp}^o + P, \tag{16}$$

where $\bar{R}_{emp}^o = \frac{1}{n_l}\sum_{i=1}^{n_l} l(\hat{h}^o, \hat{z}_i^o)$, and $P = \sigma\kappa\sqrt{\frac{1}{n_l}\sum_{i=1}^{n_l} \sigma|y_i^o - \hat{y}_i^o|} + \frac{1}{n_l}\sum_{i=1}^{n_l} \sigma|y_i^o - \hat{y}_i^o|$. To bound P, first we have $\frac{1}{n_l}\sum_{i=1}^{n_l} |y_i^o - \hat{y}_i^o| = \frac{1}{n_l}\sum_{i=1}^{n_l} |(y_i^{tL} - y_i^s) - (y_i^{tL} - \hat{y}_i^s)| = \frac{1}{n_l}\sum_{i=1}^{n_l} |y_i^s - \hat{y}_i^s| \leq \sqrt{\frac{1}{n_l}\sum_{i=1}^{n_l}(y_i^s - \hat{y}_i^s)^2}$. Using Eq. 11, $\frac{1}{n_l}\sum_{i=1}^{n_l}(y_i^s - \hat{y}_i^s)^2$ is bounded by $R_{emp}^s + O(\frac{1}{\lambda_s\sqrt{m}})$. We can see that the penalty term P diminishes as $m \to \infty$. Plugging Eq. 16 into Eq. 14 concludes the proof. □

(3) Now we analyze the generalization error in the target domain. Using the assumption that the target labels y^t can also be decomposed by $y^o + y^s$, we have:

$$\begin{aligned} &\mathbb{E}_{z^t}[l(h, z^t)] = \mathbb{E}_{z^t}[(h(x^t) - y^t)^2] \\ &= \mathbb{E}[(h^o(x^t) + h^s(x^t) - y^o - y^s)^2] \\ &\leq 2\mathbb{E}(h^o(x^t) - y^o)^2 + 2\mathbb{E}(h^s(x^t) - y^s)^2. \end{aligned} \tag{17}$$

Plugging in Eq. 11 and Eq. 15, we have

$$R^t = \mathbb{E}_{z^t}[l(h, z^t)] = 2R_{emp}^s + 2\bar{R}_{emp}^o + O(\frac{1}{\lambda_o\sqrt{n_l}} + \frac{1}{\lambda_s\sqrt{m}})$$

In transfer learning usually we assume that the number of source data is sufficient, i.e., $m \to \infty$, hence

$$R^t - 2(R_{emp}^s + \bar{R}_{emp}^o) = O(\frac{1}{\lambda_o\sqrt{n_l}}). \tag{18}$$

5.1.1 Comparing with No-transfer Learning

As with the no-transfer-rate in Sec. 4.2.2, we have

$$R^t - R_{emp}^{tL} = O(\frac{1}{\lambda_t\sqrt{n_l}}), \tag{19}$$

where λ_t is the regularization parameter when estimating the target function. Comparing this rate with Eq. 18, and using our assumption that we have a smoother offset than the target function, i.e., $\lambda_o > \lambda_t$, we can see that we obtain a faster convergence rate with transfer than no-transfer.

6 MULTI-SOURCE TRANSFER LEARNING

In this section, we show that we can easily adapt the transfer learning algorithm from a single source to transfer learning with multiple-sources, by utilizing the generalization bounds we derived in earlier sections. Transfer learning with multiple sources is similar to multi-task learning, where we learn the target and multiple sources jointly.

A closer look at Eq. 9 for CDM, and Eq. 18 for Offset reveals that, when n_l is small and $m \to \infty$, we have a convergence rate of $O(\frac{1}{\lambda_*\sqrt{n_l}})$ for both algorithms, where λ_* is some parameter that controls the smoothness of the source-to-target transfer (for Eq. 9 we can set $\lambda_R = \lambda_*^2$). This observation motivates our reweighting scheme on the source hypotheses to achieve transfer learning under multiple sources, described as the following.

Assume we have S sources and a target. First, we apply the transfer learning algorithm from a single source to obtain a model M_s from each source s to target t, where the parameter λ_*^s is determined by cross-validation, $s = 1, ..., S$. Second, we compute the weight for each source s by:

$$\begin{aligned} w_s &= p(D|M_s)p(M_s), \text{ where} \\ p(D|M_s) &= \exp\{-\sum_{i=1}^{m_s}(y_i^{tL} - \hat{f}^s(x_i^{tL}))^2\}, \\ p(M_s) &\propto \exp\{-\alpha\frac{1}{\lambda_*^s}\}, \end{aligned}$$

where $\hat{f}^s(x_i^{tL})$ is the prediction given by M_s.

The idea is similar to Bayesian Model Averaging (Hoeting et al. (1999)), where the first term $p(D|M_s)$ serves as the data likelihood of the predictive model M_s from source s, and the second term $p(M_s)$ is the prior probability on model M_s. In our case, $p(M_s)$ is chosen to indicate how similar each source to the target is, where the similarity is measured by how smooth the change is from source s to target t. It is easy to see that, the weights coincide with our analysis of the generalization bounds for transfer learning, and the choice of α should be in the order of $O(1/\sqrt{n_l})$. Intuitively, when the number of labeled target points n_l is small, $p(M_s)$ has a larger effect on w_s, which means we prefer the source that has a smoother change (larger λ_*^s) for the transfer. On the other hand, when n_l is large, then $p(D|M_s)$ takes over, i.e., we prefer the source that results in a larger data likelihood (smaller prediction errors). Finally, we combine the predictions by:

$$\hat{f}(x_i^{tU}) = \sum_{s=1}^{S} \frac{w_s}{\sum_{s=1}^{S} w_s} \hat{f}^s(x_i^{tU})$$

This weighted combination of source hypotheses gives us the following generalization bound in the target domain:

$$\begin{aligned}\mathbb{E}_{z^t}[l(h, z^t)] &= \mathbb{E}_{z^t}[(\sum_s \frac{w_s}{\sum_{s=1}^{S} w_s} h_s(x^t) - y^t)^2] \\ &= \mathbb{E}_{z^t}[(\sum_s \frac{w_s}{\sum_{s=1}^{S} w_s} (h_s(x^t) - y^t))^2] \\ &\leq \sum_s \frac{w_s}{\sum_{s=1}^{S} w_s} \mathbb{E}_{z^t}[(h_s(x^t) - y^t)^2] \\ &= \sum_s \frac{w_s}{\sum_{s=1}^{S} w_s} [\tilde{R}^s_{emp} + O(\frac{1}{\lambda_s \sqrt{n_l}})],\end{aligned}$$

where the third inequality uses Jensen's inequality, and the last equality uses the bounds we derived. Here $\tilde{R}^s_{emp}$ refers to the empirical error for source s when transferring from s to t (Thm. 2 for CDM and Eq. 18 for Offset).

7 EXPERIMENTS

7.1 SYNTHETIC EXPERIMENTS

In this section, we empirically compare the generalization error of transfer learning algorithms to that of no-transfer learning (learning by labeled target data only), on synthetic datasets simulating different conditions.

We generate the synthetic dataset in this way: X^s, X^t are drawn uniformly at random from $[0, 4]$, $Y^s = \sin(2X^s) + \sin(3X^s)$ with additive Gaussian noise. Y^t is the same function with a smoother location-scale transformation/offset. In each of the following comparisons, we plot the mean squared error (MSE) on the unlabeled target points (as an estimation of the generalization error) with respect to different number of labeled target points. The labeled target points are chosen uniformly at random, and we average the error over 10 experiments. The parameters are chosen using cross validation.

In Fig. 3, we compare transfer learning using CDM with no-transfer learning. The results show that with the additional smoothness assumption, we are able to achieve a much lower generalization error for transfer learning than no-transfer learning. In Fig. 4 and 5, we compare transfer learning using the Offset approach with no-transfer learning. The two figures show different generalization error curves when the smoothness of the offset is different. We can see that with a smoother offset (Fig. 4) we are able to achieve a much lower generalization error than no-transfer learning. With a less smooth offset (Fig. 5) we can still achieve a lower generalization error than no-transfer learning, but the rate is slower compared to Fig. 4. Further we analyze the case when the smoothness assumption does not hold, by setting the source function to be $\sin(X^s) + \epsilon$ such that the target changes faster than the source. In this case, transfer learning with CDM/Offset yield almost the same generalization error as no-transfer learning (Fig. 6), i.e., the source data does not help in learning the target.

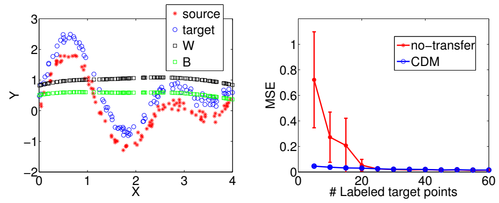

Figure 3: No-transfer learning vs. transfer learning (CDM)

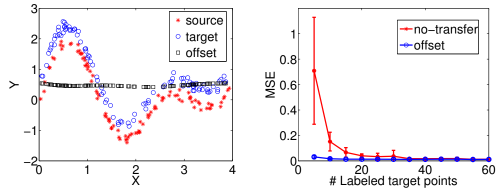

Figure 4: No-transfer learning vs. transfer learning using the Offset approach (smoother offset, $\lambda_R = 0.1$)

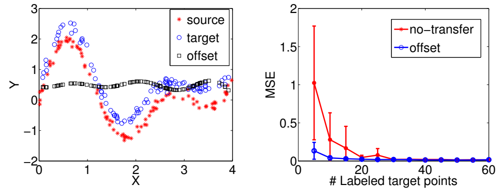

Figure 5: No-transfer learning vs. transfer learning using the Offset approach (less smooth offset, $\lambda_R = 0.001$)

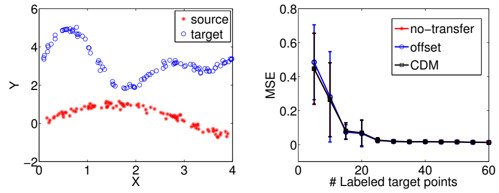

Figure 6: No-transfer learning vs. transfer learning, when the smoothness assumption does not hold

7.2 EXPERIMENTS ON THE REAL DATA

7.2.1 Comparing Transfer Learning to No-transfer Learning, Using Different Sources

The real-world dataset is an Air Quality Index (AQI) dataset (Mei et al. (2014)) during a 31-day period from Chinese cities. For each city, the input feature x_i is a bag-of-words vector extracted from Weibo posts of each day, with $100,395$ dimensions as the dictionary size. The output label y_i is a real number which is the AQI of that day.

Fig. 7 shows a comparison of MSE on the unlabeled target points, with respect to different number of labeled target points, when transferring from a nearby city (Ningbo) and a faraway city (Xi'an), to a target city (Hangzhou). The data is shown in the left figure of Fig. 7, where the x-axis is each day. The results are averaged over 20 experiments with uniformly randomly chosen labeled target points. First we observe that we obtain a lower generalization error by transferring from other cities than learning by the target city data alone (no-transfer). In addition, the generalization error are much lower if we transfer from nearby cities where the difference between source and target is smoother.

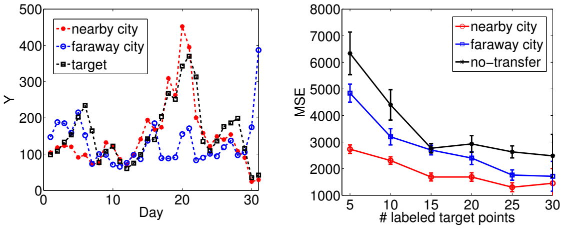

Figure 7: Comparison of MSE on unlabeled target points

7.2.2 Transfer Learning with Multiple Sources

The results in Sec. 7.2.1 indicate that, when transferring from multiple sources to a target, it is important to choose which source to transfer, in order to obtain a larger gain. In this section, we show the results on the same air quality index data by reweighting different sources (Sec. 6).

Fig. 8 shows a comparison of MSE on the unlabeled target data (data shown in the left figure) with respect to different number of labeled target points ($n_l \in \{2, 5, 10, 15, 20\}$), where the prediction is based on each source independently (labeled as **source** i, $i \in \{1, 2, 3\}$), and based on multiple sources (labeled as **posterior**). Since CDM and Offset give similar bounds, we use two-stage offset estimation as the prediction algorithm from each source s to target t. The weighting on the sources is as described in Sec. 6. As can be seen from the results, using posterior reweighting on different sources, we obtain results that are very close to the results using the best source.

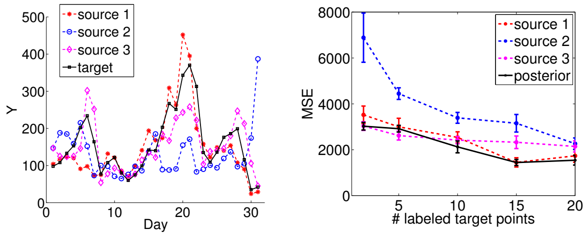

Figure 8: Comparison of MSE on unlabeled target points, with multiple sources

Further in Figure. 9, we show a comparison of MSE on the unlabeled target data between the proposed approach and two baselines, with respect to different number of labeled target points. The results are averaged over 20 experiments. The first baseline **wDA** is a weighted multi-source domain adaptation approach proposed in Mansour et al. (2008), where the distribution $D_i(x)$ for source i on a target point x is estimated using kernel density estimation with a Gaussian kernel. Note that the original algorithm proposed in Mansour et al. (2008) does not assume the existence of a few labeled target points, thus the hypothesis $h_i(x)$ from each source i is computed by using the source data only. To ensure a fair comparison, we augment $h_i(x)$ by using the prediction of the Offset approach given n_l labeled target points. The second baseline **optDA** is a multi-source domain adaptation algorithm under an optimization framework, as proposed in Chattopadhyay et al. (2011), where the parameters γ_A, γ_I are set as described in the paper, and θ is chosen using cross-validation on the set $\{0.1, 0.2, ..., 0.9\}$ (the final choice of θ is 0.1). Note that our proposed algorithm gives the best performance. In addition, our algorithm does not require density estimation as in **wDA**, which can be difficult in real-world applications with high-dimensional features. Further note **posterior** considers the change in $P(Y|X)$ while **wDA** focuses on the change of $P(X)$. A potential improvement can be achieved by combining these two in the reweighting scheme, which should be an interesting future direction.

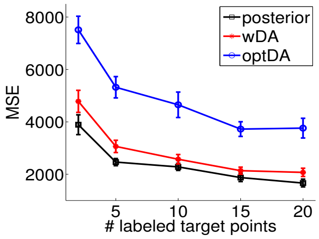

Figure 9: Multi-source transfer learning: comparison of MSE on the proposed approach (**posterior**) and baselines

8 CONCLUSION

In this paper, we provide theoretical analysis for algorithms proposed for transfer learning under the model shift assumption. Unlike previous work on covariate shift, the model shift poses a harder problem for transfer learning, and our analysis shows that we are still able to achieve a similar rate as in covariate shift/domain adaptation, modified by the smoothness of the transformation parameters. We also show conditions when transfer learning works better than no-transfer learning. Finally we extend the algorithms to transfer learning with multiple sources.

Acknowledgements

This work is supported in part by the US Department of Agriculture under grant number 20126702119958.

References

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain Adaptation with multiple sources. *NIPS* 2008.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain Adaptation: Learning Bounds and Algorithms. *COLT* 2009.

Olivier Bousquet and Andre Elisseeff. Stability and Generalization. *JMLR* 2002.

Jiayuan Huang, Alex Smola, Arthur Gretton, Karsten Borgwardt, and Bernhard Schölkopf. Correcting sample selection bias by unlabeled data. *NIPS 2007*, 2007.

Arthur Gretton, Karsten M. Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. A kernel method for the two-sample-problem. *NIPS 2007*, 2007.

Xuezhi Wang, Tzu-Kuo Huang, and Jeff Schneider. Active transfer learning under model shift. *ICML*, 2014.

S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. *NIPS 2006*.

J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman. Learning bounds for domain adaptation. *NIPS 2007*.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *TKDE 2009*, 2009.

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

Lilyana Mihalkova, Tuyen Huynh, and Raymond J. Mooney. Mapping and revising markov logic networks for transfer learning. *Proceedings of the 22nd AAAI Conference on Artificial Intelligence (AAAI-2007)*, 2007.

Chuong B. Do and Andrew Y. Ng. Transfer learning for text classification. *Neural Information Processing Systems Foundation*, 2005.

Rajat Raina, Andrew Y. Ng, and Daphne Koller. Constructing informative priors using transfer learning. *Proceedings of the Twenty-third International Conference on Machine Learning*, 2006.

Alexandru Niculescu-Mizil and Rich Caruana. Inductive transfer for bayesian network structure learning. *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2007.

J. Jiang and C. Zhai. Instance weighting for domain adaptation in nlp. *Proc. 45th Ann. Meeting of the Assoc. Computational Linguistics*, pp. 264-271, 2007.

X. Liao, Y. Xue, and L. Carin. Logistic regression with an auxiliary data source. *Proc. 21st Intl Conf. Machine Learning*, 2005.

Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. Domian adaptation under target and conditional shift. *ICML 2013*, 2013.

Le Song, Jonathan Huang, Alex Smola, and Kenji Fukumizu. Hilbert space embeddings of conditional distributions with applications to dynamical systems. *ICML 2009*, 2009.

Rita Chattopadhyay, Jieping Ye, Sethuraman Panchanathan, Wei Fan, and Ian Davidson. Multi-Source Domain Adaptation and Its Application to Early Detection of Fatigue. *KDD'11*, 2011.

Yi Yao and Gianfranco Doretto. Boosting for transfer learning with multiple sources. *CVPR 2010*, 2010.

Koby Crammer, Michael Kearns, and Jennifer Wortman. Learning from Multiple Sources. *JMLR 2008*, 2008.

Shike Mei, Han Li, Jing Fan, Xiaojin Zhu, and Charles R. Dyer. Inferring Air Pollution by Sniffing Social Media. *The International Conference on Advances in Social Network Analysis and Mining (ASONAM)*, 2014.

Jennifer A. Hoeting, David Madigan, Adrian E. Raftery, and Chris T. Volinsky. Bayesian Model Averaging: A Tutorial. *Statistical Science, Vol. 14, No. 4, 382-417*, 1999.