Rates of Convergence of Nonparametric Estimators for Model Shift

Simon S. Du **Jayanth Koushik** **Aarti Singh** **Barnabás Póczos**
Machine Learning Department, Carnegie Mellon University

Abstract

Transfer learning techniques are often used when one tries to adapt a model learned from a source domain with abundant labeled samples to the target domain with limited labeled samples. In this paper, we consider the regression problem under *model shift*, i.e., regression functions are different but related in the source and target domains, when some small amount of labeled data is available from the target domain. We propose a class of models together with estimators for this problem, which includes some previous works as special cases. Theoretically, we show that by using source domain data, the proposed estimators have better statistical rate for excess risk than traditional non-transfer learning methods under the model shift assumption. Lastly, experiments on robotics data demonstrate the effectiveness of our framework.

1 Introduction

In a classical transfer learning setting, we have sufficient data from a source domain and a small amount of data from a target domain. These two domains are related but not identical, and the usual assumption is that there is some knowledge learned from the source domain that can be transferred to the target domain.

In this paper, we focus on the regression problem under the *model shift* setting: regression functions of source and target domains are different. Many real world problems can be formulated as model shift problems like distance estimation in robotics (Sec. 5). In this paper, we propose and analyze a class of algorithms for model shift problems. Our main contributions are summarized below:

- In Section 3, we formally define our proposed class of algorithms for the model shift problem, which is flexible enough to take any standard estimator for regression problem as a subroutine. Our framework includes some previously proposed algorithms as special cases but our class is significantly richer.
- In Section 4 we develop explicit excess risk bounds for this framework using kernel smoothing and RKHS regression as non-parametric regression subroutines. Theories show algorithms in this class are able to transform the problem of directly learning the target domain regression function f^{ta} into estimating a simpler (smoother) function w_G and thus achieve better statistical rate.

1.1 Related work

For *model shift* problems, a line of research has been established based on distribution discrepancy, a loss induced metric for the source and target distributions [3, 1]. However, these works are different from our setting since they assume there is no labeled data from the target domain available. Recently, Wang and Schneider proposed a method based on kernel mean embedding to match the conditional probability in the kernel space and derived generalization bounds for method problem [6]. Kuzborskij and Orabona [2] also provided agnostic bounds for model shift problems when

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

the linear estimators are used. However, all these works lack excess risk convergence guarantees. *In this paper, we provide excess risk guarantees for the model shift setting with some labels from the target domain and formally shows when transfer learning can help.*

2 Preliminaries

Let $X \in \mathbb{R}^d$ denote a feature vector and $Y \in \mathbb{R}$ be the corresponding label. We assume both X and Y lie in compact subsets. Throughout the paper, we let $\mathcal{T} = \{(X_i, Y_i)\}_{i=1}^n$ denote a set of samples. In the model shift setting, there are n_{so} samples from the source distribution: $\mathcal{T}^{so} = \{(X_i^{so}, Y_i^{so})\}_{i=1}^{n_{so}}$, and n_{ta} samples from the target distribution: $\mathcal{T}^{ta} = \{(X_i^{ta}, Y_i^{ta})\}_{i=1}^{n_{ta}}$. Let X^{so} and Y^{so} denote the features and the label drawn from the source distribution, and X^{ta} and Y^{ta} denote the features and the label from the target distribution. We model the relation between features and labels as:

$$Y^{so} = f^{so}(X) + \epsilon^{so} \text{ and } Y^{ta} = f^{ta}(X) + \epsilon^{ta},$$

where we assume the noise $\mathbf{E}[\epsilon^{so}] = \mathbf{E}[\epsilon^{ta}] = 0$, i.i.d, and bounded. We want to minimize L_2 loss: $R(\hat{f}^{ta}) = \mathbf{E}\left[\int |\hat{f}^{ta}(X) - Y|^2 dP_{XY}\right]$, where $\hat{f}^{ta}$ is an estimator for the target domain regression function. To estimate f^{so} and f^{ta}, we consider the following two non-parametric estimators.

We say a function f is in the (λ, α) Hölder class, if for any $x, x' \in \mathbb{R}^d$, f satisfies $|f(x) - f(x')| \leq \lambda ||x - x'||^{\alpha}$, for some $\alpha \in (0, 1)$. The kernel smoothing (KS) method uses a positive kernel K on $[0, 1]$, highest at 0, decreasing on $[0, 1]$, 0 outside $[0, 1]$, and $\int_{\mathbb{R}^d} u^2 K(u) < \infty$. The kernel smoothing estimator is defined as follows: $\hat{f}(x) = \sum_{i=1}^n w_i(x) Y_i$, where $w_i(x) = K(||x - X_i|| / h) / \left(\sum_{j=1}^n K(||x - X_j|| / h)\right)$.

Another popular non-parametric estimator is kernel ridge regression (KRR) which uses the theory of reproducing kernel Hilbert space (RKHS) for regression [5]. Any symmetric positive semidefinite kernel function $K : \mathcal{R}^d \times \mathcal{R}^d \to \mathcal{R}$ defines a RKHS $\mathcal{H}$. Given the inner product, the $\mathcal{H}$ norm of a function is defined as $||g||_{\mathcal{H}} \triangleq \sqrt{\langle g, g \rangle_{\mathcal{H}}}$ and similarly the L_2 norm, $||g||_2 \triangleq \left(\int_{\mathbb{R}^d} g(x)^2 dP_x\right)^{1/2}$ for a given P_X. Also, the kernel induces an integral operator $T_K : L_2(P_X) \to L_2(P_X)$: $T_K[f](x) = \int_{\mathbb{R}^d} K(x', x)(x') \, dP_x(x')$. with eigenvalues: $\{\mu_i\}_{i \geq 1}$ where $\mu_i \geq \mu_{i+1}$. For a given function f, the approximation error is defined as: $A(\lambda) \triangleq \inf_{h \in \mathcal{H}} \left(||h - f||^2_{L_2(P_x)} + \lambda ||h||^2_{\mathcal{H}}\right)$. The KRR estimator is defined by $\hat{f} \triangleq \operatorname{argmin}_{h \in \mathcal{H}} \left\{\frac{1}{n} \sum_{i=1}^n (h(X_i) - Y_i)^2 + \lambda ||h||^2_{\mathcal{H}}\right\}$.

3 Transformation Function for Model Shift

Our models are based on the idea that transfer learning is helpful when one transforms the target domain regression problem into a simpler regression problem using source domain knowledge. The following simple example illustrates this concept:

Example: Offset Transfer. Suppose

$$f^{so}(x) = \sqrt{x(1-x)} \sin\left(\frac{2.1\pi}{x + 0.05}\right), \qquad f^{ta}(x) = f^{so}(x) + x. \tag{1}$$

f^{so} is the so called Doppler function. It requires a large number of samples to estimate well because of its lack of smoothness [8]. For the same reason, f^{ta} is also difficult to estimate if we only have limited data. However, if we have enough data from the source domain, we can have a fairly good estimate of f^{so}, denoted by $\hat{f}^{so}$. Further, notice that the offset function $w(x) = f^{ta}(x) - f^{so}(x) = x$, is just a linear function. Thus, instead of directly using $\mathcal{T}^{ta}$ to estimate f^{ta}, we can use the target domain samples to find an estimate of $w(x)$, denoted by $\hat{w}(x)$, and our estimator for the target domain is just: $\hat{f}^{ta}(x) = \hat{f}^{so}(x) + \hat{w}(x)$.

Now we generalize this idea by consider the following function: $G(x, y) : \mathbb{R}^2 \to \mathbb{R}$, which satisfies that given $x \in \mathbb{R}$, $G(x, \cdot)$ is invertible. Let $G_x^{-1}(\cdot)$ denote the inverse of $G(x, \cdot)$ such that $G\left(x, G_x^{-1}(y)\right) = y$. Let $\mathcal{G}$ denote a set of such functions given by

$$\mathcal{G} = \left\{G : \mathbb{R}^2 \to \mathbb{R} | f^{ta}(X) = G\left(f^{so}(X), w_G(X)\right)\right\},$$

Algorithm 1 Hypothesis based Transfer Learning

Inputs: Source domain data: $\mathcal{T}^{so} = \{(X_i^{so}, Y_i^{so})\}_{i=1}^{n_{so}}$, target domain data: $\mathcal{T}^{ta} = \{(X_i^{ta}, Y_i^{ta})\}_{i=1}^{n_{ta}}$, transformation function: G, algorithm to train f^{so}: $\mathcal{A}_{so}$, algorithm to train w_G: $\mathcal{A}_{w_G}$.

Outputs: Regression function for the target domain: $\hat{f}^{ta}$.

1: Train the source domain regression function $\hat{f}^{so} = \mathcal{A}_{so}(\mathcal{T}^{so})$.

2: Construct new data using $\hat{f}^{so}$ and $\mathcal{T}^{ta}$: $\mathcal{T}^{w_G} = \{(X_i^{ta}, W_i)\}_{i=1}^{n_{ta}}$, where $W_i = \widehat{W}_i\left(f^{so}(X_i), Y_i^{ta}\right)$, an unbiased estimate of $w_G(X_i^{ta})$.

3: Train the auxiliary function: $\hat{w}_G = \mathcal{A}_{W_G}(\mathcal{T}^{w_G})$.

4: Output the estimated regression for the target domain: $\hat{f}^{ta}(X) = G\left(\hat{f}^{so}(X), \hat{w}_G(X)\right)$.

where $w_G(x) = G^{-1}_{f^{so}(x)}(f^{ta}(x))$. We call $G \in \mathcal{G}$ a transformation function, and the corresponding w_G the auxiliary function for G. Here G is a user-defined transformation of the regression function from the source domain to the target domain, which is chosen by users' prior knowledge on the relation between the source and target domains. Later, we will show that learning f^{ta} is reduced to learning w_G instead. In the previous example, $G(x, y) = x + y$ and $w_G(x) = x$.

Algorithm 1 shows the Pseudocode of our meta-algorithm. In Step 2 we estimate the auxiliary function and in Step 4 we recover the target regression function by our estimation of source function and auxiliary function. In this way, we may decrease the estimation error.

4 Theoretical analysis

In this section, we present theoretical analyses for the proposed class of models and estimators. We make the following two assumptions.

Assumption 1 *For all $G \in \mathcal{F}$, G is L-Lipschitz: $|G(x, y) - G(x', y')| \leq L\,||(x, y) - (x', y')||$, where the norm is the usual Euclidean distance.*

Assumption 2 *The unbiased estimator for $w_G(X_i^{ta})$ is bounded.*

The following theorem provides the guarantee for kernel smoothing:

Theorem 1 *Suppose $P_{X^{ta}}$ and $P_{X^{so}}$ have the same support $\mathcal{X}$and there are constant C_0 and C_1 such that $C_0 r^d \leq P_{X^{so}}(B(x, r)) \leq C_1 r^d$ and $C_0 r^d \leq P_{X^{ta}}(B(x, r)) \leq C_1 r^d$ for all $x \in \mathcal{X}$, where $B(x, r)$ denotes the ball centered at x with radius r. Further assume f^{so} is $(\lambda_{so}, \alpha_{so})$ Hölder and w_G is $(\lambda_{w_G}, \alpha_{w_G})$ Hölder . If we use kernel smoothing estimation for f^{so} and w_G with bandwidth $h_{so} \asymp n_{so}^{-1/(2\alpha_{so}+d)}$ and $h_{w_G} \asymp n_{ta}^{-1/(2\alpha_{w_G}+d)}$, with probability at least $1 - \delta$ the risk satisfies:*

$$R\left(\hat{f}^{ta}\right) - R\left(f^{ta}\right) = O\left(\left(\log^2(n_{ta}) \cdot n_{so}^{-2\alpha_{so}/(2\alpha_{so}+d)} + n_{ta}^{-2\alpha_{w_G}/(2\alpha_{w_G}+d)}\right)\log(1/\delta)\right).$$

Theorem 1 suggests that the risk depends on two sources, one from estimation of f^{so} and one from estimation of w_G. For the first term, even though it depends logarithmically on n_{ta}, since in the typical transfer learning scenarios $n_{so} >> n_{ta}$, it is relatively small in the setting we focus on. The second terms shows the power of transfer learning on transforming a possibly complex target regression function into a simpler auxiliary function. It is well known that learning f^{ta} only using target domain has risk of the order $\Omega\left(n_{ta}^{-2\alpha_{f^{ta}}/\left(2\alpha_{f^{ta}}+d\right)}\right)$. Thus, if the auxiliary function is smoother than the target regression function, i.e. $\alpha_{w_G} > \alpha_{f^{ta}}$, we obtain better statistical rate.

Next, we give an upper bound for KRR:

Theorem 2 *Suppose $P_{X^{so}} = P_{X^{ta}} = P_X$. Assume that the eigenvalues of the integral operator T_K satisfy $\mu_i \leq a i^{-1/p}, \quad i \geq 1$ where $a \geq 16\triangle_Y^4$ and $p \in (0, 1)$ and there exists a constant $C \geq 1$ such that for $f \in \mathcal{H}$, $||f||_\infty \leq C\,||f||_{\mathcal{H}}^p \cdot ||f||_{L_2(P_X)}^{1-p}$. Further suppose that $A^{f^{so}}(\lambda) \leq c\lambda^{\beta_{so}}$ and*

	$n_{ta}=10$	$n_{ta}=20$	$n_{ta}=40$	$n_{ta}=80$	$n_{ta}=160$	$n_{ta}=320$
Only Target KS	0.086 ± 0.022	0.076 ± 0.010	0.066 ± 0.008	0.064 ± 0.007	0.065 ± 0.006	0.063 ± 0.005
Only Target KRR	0.080 ± 0.017	0.078 ± 0.022	0.063 ± 0.013	0.050 ± 0.007	0.048 ± 0.006	$\mathbf{0.040 \pm 0.005}$
Only Source KRR	0.098 ± 0.017	0.098 ± 0.017	0.098 ± 0.017	0.098 ± 0.017	0.098 ± 0.017	0.098 ± 0.017
Combined KS	0.092 ± 0.011	0.084 ± 0.008	0.077 ± 0.009	0.075 ± 0.006	0.074 ± 0.006	0.067 ± 0.006
Combined KRR	0.087 ± 0.025	0.077 ± 0.015	0.062 ± 0.009	0.061 ± 0.005	0.047 ± 0.003	0.041 ± 0.004
CDM KRR	0.105 ± 0.023	0.074 ± 0.020	0.064 ± 0.008	0.060 ± 0.007	0.053 ± 0.009	0.056 ± 0.004
Offset KS	0.080 ± 0.026	0.066 ± 0.023	$\mathbf{0.052 \pm 0.006}$	0.054 ± 0.006	0.050 ± 0.003	0.052 ± 0.004
Offset KRR	0.146 ± 0.112	0.066 ± 0.017	0.053 ± 0.007	$\mathbf{0.048 \pm 0.006}$	$\mathbf{0.043 \pm 0.004}$	0.041 ± 0.003
Scale KS	$\mathbf{0.078 \pm 0.022}$	$\mathbf{0.065 \pm 0.013}$	0.056 ± 0.009	0.056 ± 0.005	0.054 ± 0.008	0.055 ± 0.004
Scale KRR	0.102 ± 0.033	0.095 ± 0.100	0.057 ± 0.014	0.052 ± 0.010	0.044 ± 0.004	0.042 ± 0.002

Table 1: 1 standard deviation intervals for the mean squared errors of various algorithms when transferring from kin-8fm to kin-8nh. 'Only Target' means only using target domain data, 'Only Source' means only using source domain data and 'Combined' means treating source and target domain data as one data set. CDM is the algorithm proposed by [7]. 'Offset' means we use $G(x,y)=x+y$ as our transformation function and 'Scale' means we use $G(x,y)=xy$ as our transformation function.

$A^{w_G}(\lambda) \leq c\lambda^{\beta_{w_G}}$. *Then if we use KRR for estimating* f^{so} *and* w_G *with regularization parameters* $\lambda_{so} \asymp n_{so}^{-1/(\beta_{so}+p)}$ *and* $\lambda_{w_G} \asymp n_{ta}^{-1/(\beta_{w_G}+p)}$*, with probability at least* $1-\delta$ *the excess risk satisfies:*

$$R\left(\hat{f}^{ta}\right) - R\left(f^{ta}\right) = O\left(\left(n_{ta}^{2/\left(\beta_{w_G}+p\right)} \log\left(n_{ta}\right) \cdot n_{so}^{-\beta_{so}/(\beta_{so}+p)} + n_{ta}^{-\beta_{w_G}/(\beta_{w_G}+p)}\right) \log\left(1/\delta\right)\right).$$

Similar to Theorem 1, Theorem 2 suggests that the estimation error comes two sources. Again, the error of estimating f^{so} is amplified by $O\left(n_{ta}^{2/\left(\beta_{w_G}+p\right)}\right)$, which is the price we need to pay for using transfer learning.

5 Experiments

In this section we use robotic data to demonstrate the effectiveness of the proposed framework. The task is predicting the distance of the end-effector of a robotic arm from a target; the inputs are various attributes of the ar [4]. The two datasets we use are 'kin-8fm' and 'kin-8nh'. We consider transferring task from kin-8fm to kin-8nh where kin-8fm has fairly linear output and low noise; kin-8nh on the other hand has non-linear output and high noise. In this experiment, We set $n_{so}=320$, and vary the size of the target domain. Table 1 shows results of different algorithms.

References

[1] C. Cortes and M. Mohri. Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science*, 519:103–126, 2014.

[2] I. Kuzborskij and F. Orabona. Stability and hypothesis transfer learning. In *ICML (3)*, pages 942–950, 2013.

[3] Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*, 2009.

[4] C. E. Rasmussen, R. M. Neal, G. Hinton, D. van Camp, M. Revow, Z. Ghahramani, R. Kustra, and R. Tibshirani. Delve data for evaluating learning in valid experiments. *URL http://www. cs. toronto. edu/ delve*, 1996.

[5] V. Vovk. Kernel ridge regression. In *Empirical Inference*, pages 105–116. Springer, 2013.

[6] X. Wang and J. Schneider. Generalization bounds for transfer learning under model shift.

[7] X. Wang and J. Schneider. Flexible transfer learning under support and model shift. In *Advances in Neural Information Processing Systems*, pages 1898–1906, 2014.

[8] L. Wasserman. *All of nonparametric statistics*. Springer Science & Business Media, 2006.