Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)

Nonparametric Risk and Stability Analysis for Multi-Task Learning Problems

Xuezhi Wang, Junier B. Oliva, Jeff Schneider, Barnabás Póczos
Carnegie Mellon University, Pittsburgh, PA, USA
{xuezhiw, joliva, schneide, bapoczos}@cs.cmu.edu

Abstract

Multi-task learning attempts to simultaneously leverage data from multiple domains in order to estimate related functions on each domain. For example, a special case of multi-task learning, transfer learning, is often employed when one has a good estimate of a function on a source domain, but is unable to estimate a related function well on a target domain using only target data. Multi-task/transfer learning problems are usually solved by imposing some kind of "smooth" relationship among/between tasks. In this paper, we study how different smoothness assumptions on task relations affect the upper bounds of algorithms proposed for these problems under different settings. For general multi-task learning, we study a family of algorithms which utilize a reweighting matrix on task weights to capture the smooth relationship among tasks, which has many instantiations in existing literature. Furthermore, for multi-task learning in a transfer learning framework, we study the recently proposed algorithms for the "model shift", where the conditional distribution $P(Y|X)$ is allowed to change across tasks but the change is assumed to be smooth. In addition, we illustrate our results with experiments on both simulated and real data.

1 Introduction

As machine learning is applied to a growing number of domains, many researchers have looked to exploit similarities in machine learning tasks in order to increase performance. For example, one may suspect that data for the classification of one commodity as profitable or not may help in classifying a different commodity. Similarly, it is likely that data for spam classification in one language can help spam classification in another language. A common technique for leveraging data from different domains for machine learning tasks is multi-task learning. Multi-task learning pools multiple domains together and couples the learning of several tasks by regularizing separate estimators jointly and dependently. For instance, in transfer learning, a special case of multi-task learning, one uses data (or an estimator) from a well understood source domain with plentiful data to aid the learning of a target domain with scarce data. Although multi-task learning algorithms are becoming prevalent in machine learning, there are gaps in our understanding of their properties, especially in nonparametric settings. This paper looks to increase our understanding of fundamental questions such as: What can one say about the true risk of a multi-task estimator given its empirical risk? How do relative sample sizes affect learning among different domains? How does the similarity between two functions affect one's ability to transfer learning between them?

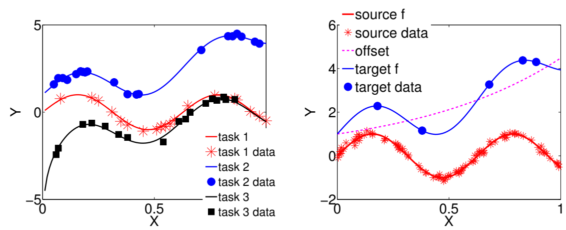


Figure 1: Toy example illustrating general multi-task learning (left) and transfer learning (right).

Although there are many regularization techniques proposed for multi-task learning (Figure 1 (left)), there are very few studies on how the learning bounds change under different parameter regularizations. In this paper, we analyze the stability bounds under a general framework of multi-task learning using kernel ridge regression. Our formulation places a reweighting matrix Λ on task weights to capture the relationship among tasks. Our analysis shows that most existing work can be cast into this framework by changing the reweighting matrix Λ. More importantly, we show that the stability bounds under this framework depend on the diagonal blocks of Λ^{-1}, thus providing insights on how much we can gain from regularizing the relationship among tasks by using different reweighting matrices.

Moreover, recently a general framework on transfer learning under model shift has been proposed [Wang *et al.*, 2014], where the conditional distribution between the source and target domains is allowed to differ, unlike in many previous methods, but the change is assumed to be smooth (Figure 1 (right)). However, it is not clear how performance varies with the type of smoothness assumption. Furthermore, it is unclear under what conditions transfer learning improves estimation over typical one data-set learning. In this paper, we provide analysis that connects the smoothness of the offset function to the learning bounds for this kind of transfer. We obtain tighter learning bounds for transfer learning when we assume a smooth change across domains, given that the data from the source domain is sufficiently large.

Contribution We provide a stability analysis for multi-task learning that allows one to understand the gap between the true risk and the empirical risk for many popular estimators. Also, we analyze the risk of multi-task learning under a nonparametric function transfer learning framework. In our analysis we derive an upperbound for the L_2 risk that elucidates previously unknown question such as the relationship between sample sizes and loss, as well as conditions for outperforming one data-set estimation with transfer learning.

2 Related Work

A large part of multi-task learning work is formulated using kernel ridge regression (KRR) with various regularizations on task relations. The ℓ_2 penalty is used on a shared mean function and on the variations specific to each task [Evgeniou and Pontil, 2004; Evgeniou *et al.*, 2005]. In [Solnon *et al.*, 2013] a pairwise ℓ_2 penalty is placed between each pair of tasks. In [Zhou *et al.*, 2011] an ℓ_2 penalty is proposed on each pair of consecutive tasks that controls the temporal smoothness. By regularizing the shared clustering structure among task parameters, task clusters are constructed for different features [Zhong and Kwok, 2012]. A multi-linear multi-task learning algorithm is proposed in [Romera-Paredes *et al.*, 2013] by placing a trace norm penalty. However, there are very few literature dealing with the stability of multi-task KRR algorithms. The stability bounds for transductive regression algorithms are analyzed in [Cortes *et al.*, 2008]. In [Audiffren and Kadri, 2013], the authors study the stability properties of multi-task KRR by considering each single task separately, thus failing to reveal the advantage of regularizing task relations. In [Maurer and Pontil, 2013], a regularized trace-norm multi-task learning algorithm is studied where the task relations are modeled implicitly, while we study a general family of multi-task learning algorithms where the task relations are modeled explicitly, reflected by a reweighting matrix Λ on task weights. More recently, the algorithmic stability for multi-task algorithms with linear models $f(x) = w^\top x$ is studied in [Zhang, 2015]. While in this paper we consider the more challenging **nonlinear** models with feature map $\phi(x)$, where the new multi-task kernel between x_i, x_j is defined as $\phi(x_i)\Lambda^{-1}\phi^\top(x_j)$ by absorbing the reweighing matrix Λ on task weights. Our theory is also developed on the more general nonlinear models.

Most work on transfer learning assumes that specific parts of the model can be carried over between tasks. Recent work on covariate shift [Shimodaira, 2000; Huang *et al.*, 2007; Gretton *et al.*, 2007; Yu and Szepesvri, 2012; Wen *et al.*, 2014; Reddi *et al.*, 2015] considers the case where only $P(X)$ differs across domains, while $P(Y|X)$ stays the same (here X denotes the input feature space and Y denotes the output label space). In [Zhang *et al.*, 2013], target and conditional shift are modeled by matching the marginal distributions on X. 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 distribution $P(X)$, hence neither covariate shift or target/conditional shift will work well under the model shift assumption. This problem is also demonstrated in [Wang *et al.*, 2014]. In the same paper, the authors propose a transfer learning algorithm to handle the general case where $P(Y|X)$ changes smoothly across domains.

We focus our analysis to the nonparametric setting. In particular, we consider orthogonal series regression, where one attempts to model functions using a finite collection of orthonormal basis functions [Tsybakov, 2009; Wasserman, 2006]. Moreover, we also consider kernel ridge regression, a natural generalization of ridge regression [Hoerl and Kennard, 1970] to the nonparametric setting [Györfi, 2002].

3 Stability Analysis on Multi-Task Kernel Ridge Regression

In this section, we analyze the stability bounds for multi-task kernel ridge regression (MT-KRR). Our analysis shows that, MT-KRR achieves tighter stability bounds than independent task learning by regularizing task relations. In addition, different regularization techniques yield different stability bounds that are closely related to the diagonal blocks of the inversed reweighting matrix. Due to space constraints please refer to the appendix[1] for all the proofs.

3.1 Multi-task KRR Algorithm: Formulation and Objective

Assume we have T tasks, each task t has data matrix $X_t \in \mathcal{R}^{n_t \times d}, Y_t \in \mathcal{R}^{n_t}$, where $x_{t,i} \in \mathcal{X}$ is the i-th row of X_t, and $y_{t,i} \in \mathcal{Y}$ is the i-th scalar of Y_t. n_t is the number of data points for each task, and d is the dimension of features. Denote the total number of data points as $m = \sum_{t=1}^{T} n_t$.

Let ϕ be the feature mapping on x associated to kernel k with dimension q, and $\Phi(X_t)$ denote the matrix in $\mathcal{R}^{n_t \times q}$ whose rows are the vectors $\phi(x_{t,i})$. Let $\Phi(X) \in \mathcal{R}^{m \times Tq}$ represent the diagonalized data matrix $\Phi(X) = \text{diag}[\Phi(X_1)\ \Phi(X_2)\ \cdots\ \Phi(X_T)]$ for all tasks, $Y \in \mathcal{R}^{m\times 1}$ be the stacked label vector $Y = [Y_1\ Y_2\ \ldots\ Y_T]^\top$, and $w \in \mathcal{R}^{Tq\times 1}$ be the stacked weight vector $w = [w_1\ w_2\ \ldots\ w_T]^\top$. Throughout the paper we use ℓ_2 loss as the loss function for a hypothesis h, i.e., $l(h(x), y) = (h(x) - y)^2$. Note that $l(h(x), y)$ is a σ-admissible loss function, i.e., $\forall x, y, \forall h, h', |l(h(x), y) - l(h'(x), y)| \le \sigma |h(x) - h'(x)|$. For ℓ_2 loss $\sigma = 4B$, assuming $|h(x)| \le B, |y| \le B$ for some $B > 0$. Define the MT-KRR objective as:

$$\min_w \frac{1}{m} ||Y - \Phi(X)w||_F^2 + w^\top \Lambda w,$$

where Λ is a $Tq \times Tq$ reweighting matrix on task weights w. Let $\tilde{\phi}(x_{t,j}) = [0\ \cdots\ 0\ \phi(x_{t,j})\ 0\ \cdots\ 0]$ be a row of $\Phi(X)$ for task t. Let $\mathcal{H}$ be a reproducing kernel Hilbert space with kernel $k_{\Lambda^{-1}}(x_{s,i}, x_{t,j}) = \tilde{\phi}(x_{s,i})\Lambda^{-1}\tilde{\phi}^\top(x_{t,j})$ (s, t are

[1] Available at http://www.autonlab.org/autonweb/24058.html

indices for tasks), the objective becomes:

$$\min_{g \in \mathcal{H}} \frac{1}{m} \sum_{t=1}^{T} \sum_{j=1}^{n_t} (y_{t,j} - g(x_{t,j}))^2 + ||g||^2_{k_{\Lambda^{-1}}} \quad (1)$$

where $g(x) = \langle g, k_{\Lambda^{-1}}(x,.)\rangle_{\mathcal{H}}$, and $||.||_{K_{\Lambda^{-1}}}$ is the norm in $\mathcal{H}$. This generalizes to the case where $q = \infty$. The solution to MT-KRR is (assuming nonsingular Λ): $w = \Lambda^{-1}\Phi^\top(X)[\Phi(X)\Lambda^{-1}\Phi^\top(X) + mI]^{-1}Y$. Note in multi-task learning setting, we have $\Lambda = \Omega \otimes \mathbf{I_q}$ (for some $\Omega \in \mathcal{R}^{T\times T}$), where $\mathbf{I_q}$ is the $q \times q$ identity matrix and $\otimes$ is the Kronecker product. By the property of the inverse of a Kronecker product, $\Lambda^{-1} = M \otimes \mathbf{I_q}$ where $M = \Omega^{-1}$, and it can be easily shown that $k_{\Lambda^{-1}}(x_{s,i}, x_{t,j}) = M_{s,t}k(x_{s,i}, x_{t,j})$. Most existing multi-task algorithms can be cast into the above framework, see Table 1 for a few examples.

Remark. Eq. 1 assumes same weight $1/m$ on the loss for $(x_{t,j}, y_{t,j})$ for all tasks. Alternatively, we can put different weights on the loss for different tasks, i.e, $\min_w \sum_{t=1}^{T} \frac{1}{n_t} \sum_{j=1}^{n_t} (\phi(x_{t,j})w_t - y_{t,j})^2 + w^\top \Lambda w$. The solution becomes $w = \Lambda^{-1}\Phi^\top(X)(\Phi(X)\Lambda^{-1}\Phi^\top(X) + C^{-1}I)^{-1}Y$, where C is the loss-reweighting matrix with $1/n_t$'s as the diagonal elements. As C is the same under different Λ's, it is not the focus of this paper. A study on the effect of C can be found in [Cortes *et al.*, 2008].

3.2 Uniform Stability for MT-KRR

We study the uniform stability [Bousquet and Elisseeff, 2002], which is usually used to bound true risk in terms of empirical risk, for the MT-KRR algorithm.

Definition 3.1. **([Bousquet and Elisseeff, 2002])**. *The uniform stability β for an algorithm A w.r.t. the loss function l is defined as: $\forall S \in \mathcal{Z}^m, \forall i \in \{1, ..., m\}, ||l(A_S, .) - l(A_{S\setminus i}, .)||_\infty \leq \beta$, where $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ drawn i.i.d from an unknown distribution D, and $S \setminus i$ is formed by removing the i-th element from S.*

Definition 3.2. **(Uniform stability w.r.t a task t)**. *Let i be a data index for task t. The uniform stability β_t of a learning algorithm A w.r.t a task t, w.r.t. loss l is: $\forall S \in \mathcal{Z}^m, \forall i \in \{1, ..., n_t\}, ||l(A_S, .) - l(A_{S\setminus i}, .)||_\infty \leq \beta_t$.*

Let the risk or generalization error be defined as $R(A, S) = \mathbb{E}_z[l(A_S, z)], z \in \mathcal{Z}$, and the empirical error be defined as $R_{emp} = \frac{1}{m}\sum_{i=1}^{m} l(A_S, z_i), z_i \in \mathcal{Z}^m$. Then we have the following generalization error bound (Theorem 12, [Bousquet and Elisseeff, 2002]) with probability at least $1 - \delta$: $R \leq R_{emp} + 2\beta + (4m\beta + 4B^2)\sqrt{\frac{\ln 1/\delta}{2m}}$. This theorem gives tight bounds when the stability β scales as $1/m$. For the MT-KRR algorithm, we have the following theorem hold with respect to the uniform stability:

Theorem 3.3. *Denote $\Lambda^{-1} = M \otimes \mathbf{I_q}$, and $M_1, \ldots, M_T$ are the diagonal elements of M. Assuming the kernel values are bounded: $\forall x \in \mathcal{X}, k(x, x) \leq \kappa^2 < \infty$. The learning algorithm defined by the minimizer of Eq. 1 has uniform stability β w.r.t. σ-admissible loss l with:*

$$\beta \leq \frac{\sigma^2\kappa^2}{2m} \max_t M_t.$$

The proof is similar to the proof of Thm. 22 in [Bousquet and Elisseeff, 2002], except that in the multi-task learning setting, for the tth task $\tilde{\phi}(x_{t,i}) = [0 \cdots 0 \ \phi(x_{t,i}) \ 0 \cdots 0]$, by the standard bounds for Rayleigh quotient, we have

$$\tilde{\phi}(x_{t,i})\Lambda^{-1}\tilde{\phi}^\top(x_{t,i}) \leq \kappa^2\lambda_{max}(M_t\mathbf{I_q}) = \kappa^2 M_t. \quad (2)$$

Remark. The above theorem provides a more direct stability bound by taking the maximum over the diagonal elements of M, instead of computing the largest eigenvalue as in [Zhang, 2015]. Also, for a specific task t, if $M_t < \max_s\{M_s\}$, then it is possible to obtain tighter stability β_t using only M_t, which yields tighter bounds than the one in [Zhang, 2015] where they consider the worst case for all tasks.

Lemma 3.4. *The learning algorithm defined by the minimizer of Eq. 1 has uniform stability β_t w.r.t a task t, w.r.t. σ-admissible loss l with: $\beta_t \leq \frac{\sigma^2\kappa^2}{2n_t}M_t$.*

The proof is a straightforward adaptation of proof of Thm 3.3, with x constrained to be $x_{t,i}$. The reason we care about β_t is that, it leads to a tighter generalization error bound for a task t, given $\beta_t < \beta$. We have, with probability at least $1-\delta$, $R^t \leq R^t_{emp} + 2\beta_t + (4n_t\beta_t + 4B^2)\sqrt{\frac{\ln 1/\delta}{2n_t}}$, where $R^t = \mathbb{E}_z[l(A_S(x_{t,i}), y_{t,i})]$, $R^t_{emp} = \frac{1}{n_t}\sum_{i=1}^{n_t} l(A_S(x_{t,i}), y_{t,i})$. In the following section, we study the stability bounds under a few special cases (Table 1), where it can be shown that we have tighter stability bounds for MT-KRR than learning each task independently.

3.3 Stability Bounds under Different Penalties

(a) Independent tasks. It is easy to derive that $\forall t, M_t = 1/\lambda_s$, and $\beta_{ind} \leq \frac{\sigma^2\kappa^2}{2\lambda_s m}$.

Remark. In [Audiffren and Kadri, 2013], the stability of multi-task KRR is analyzed by considering each task separately, which corresponds to the above analysis. In the following, we will show that different regularizations on task relations help tighten the stability bounds of MTL algorithms.

(b) Central function+offset. Applying blockwise matrix inversion we have $\forall t$, $M_t = \frac{\lambda_p/T + \lambda_s}{\lambda_s(\lambda_p + \lambda_s)}$. We achieve tighter stability bounds than β_{ind} for $T \geq 2$ and $\lambda_p > 0$ since:

$$\max_t M_t = \frac{\lambda_p/T + \lambda_s}{\lambda_s(\lambda_p + \lambda_s)} < \frac{1}{\lambda_s}. \quad (3)$$

(c) Pairwise penalty. Similarly to (b), we can derive that $\forall t$, $M_t = \frac{\lambda_p + \lambda_s}{\lambda_s(\lambda_p T + \lambda_s)}$. For $T \geq 2$ and $\lambda_p > 0$, again we obtain tighter bounds than β_{ind}:

$$\max_t M_t = \frac{\lambda_p + \lambda_s}{\lambda_s(\lambda_p T + \lambda_s)} < \frac{1}{\lambda_s}. \quad (4)$$

(d) Temporal penalty. We have the following lemma:

Lemma 3.5. *Let Λ be defined as in Table 1 under temporal penalty and M be defined as in theorem 3.3. Let $M_{t_{mid}}$ be the middle element(s) of $M_1, M_2, \ldots, M_T$, i.e., $t_{mid} = T/2, T/2 + 1$ if T is even, and $t_{mid} = (T + 1)/2$ if T is odd. Then the following hold: $M_t < M_{t-1}, t = 2, \ldots, t_{mid}; M_t < M_{t+1}, t = t_{mid}, \ldots, T; \max_t M_t = M_1 = M_T < \frac{1}{\lambda_s}; \min_t M_t = M_{t_{mid}} \geq \frac{\lambda_p + \lambda_s}{\lambda_s(\lambda_p T + \lambda_s)}$.*

Methods	Penalty $P = w^\top \Lambda w$	$\Lambda = \Omega \otimes \mathbf{I_q}$
Independent tasks	$\lambda_s \sum_{t=1}^{T} \|\|w_t\|\|^2$	$\Omega = \lambda_s \mathbf{I_T}$
Central+offset [Evgeniou and Pontil, 2004]	$\lambda_s \sum_{t=1}^{T} \|\|w_t\|\|^2 + \lambda_p \sum_{t=1}^{T} \|\|w_t - \frac{1}{T}\sum_{s=1}^{T} w_s\|\|^2$	$\begin{cases} \Omega_{t,t} = \lambda_s + \lambda_p(1-\frac{1}{T}) \\ \Omega_{s,t} = -\frac{\lambda_p}{T}, s \neq t \end{cases}$
Pairwise [Solnon *et al.*, 2013]	$\lambda_p \sum_{s \neq t} \|\|w_s - w_t\|\|^2 + \lambda_s \sum_{t=1}^{T} \|\|w_t\|\|^2$	$\begin{cases} \Omega_{t,t} = \lambda_p(T-1) + \lambda_s \\ \Omega_{s,t} = -\lambda_p, s \neq t \end{cases}$
Temporal [Zhou *et al.*, 2011]	$\lambda_p \sum_{t=1}^{T-1} \|\|w_t - w_{t+1}\|\|^2 + \lambda_s \sum_{t=1}^{T} \|\|w_t\|\|^2$	$\begin{cases} \Omega_{t,t} = 2\lambda_p + \lambda_s, t = 2, \ldots, T-1; \\ \Omega_{t,t+1} = \Omega_{t+1,t} = -\lambda_p, t = 1, \ldots, T-1; \\ \Omega_{1,1} = \Omega_{T,T} = \lambda_p + \lambda_s; \text{ zero otherwise.} \end{cases}$

Table 1: Examples of multi-task learning algorithms with different Λ's as penalty

Combining Lemma 3.5 and Lemma 3.4 we can see that, with temporal penalty we have tightest stability bounds β_t for $t = t_{mid}$. Also, we achieve tighter stability bounds β_t for the tth task than the $t-1$th task, if $t < t_{mid}$; and tighter β_t for the tth task than the $t+1$th task, if $t > t_{mid}$. However, since $M_{t_{mid}} \geq (\lambda_p + \lambda_s)/(\lambda_s(\lambda_p T + \lambda_s))$, we achieve a looser bound with temporal penalty than Eq. 3 or Eq. 4. It indicates that we might lose some algorithmic stability due to the relatively restricted temporal smoothness assumption, compared to assuming pairwise smoothness. Nonetheless, the stability bound with temporal penalty is tighter than learning each task independently: $\max_t M_t < 1/\lambda_s$, for $T \geq 2$.

4 Upper Bounds on Transfer Learning

A special case of multi-task learning, transfer learning, also assumes that one can benefit from task relations, but focuses mainly on two tasks. While general multi-task learning assumes a comparable number of samples for each task, transfer learning usually assumes a sufficiently labeled source task and a very limited labeled target task. In this section, we analyze the L_2 risk for transfer learning with respect to the source and target sample size, and smoothness assumptions made between the tasks.

4.1 Model

We consider a densely sampled function f_0, which one uses to aid in the regression of a sparsely sampled function f_1. The relationship between functions is defined through a smoothness assumption on the difference of the two functions: $g(x) \equiv f_1(x) - f_0(x)$.

Our estimator works as follows: first, we use a sample of noisy f_0 values to produce an estimate $\tilde{f}_0$; second, we use $\tilde{f}_0$ to generate noisy samples of g by subtracting $\tilde{f}_0$ from noisy samples of f_1, and we produce an estimate $\hat{g}$; lastly, we define our estimator of f_1 as $\hat{f}_1(x) \equiv \tilde{f}_0(x) + \hat{g}(x)$. Specifically we consider the following data:

$$\{u_{0i}\}_{i=1}^{n_0}, \{u_{1i}\}_{i=1}^{n_1} \overset{iid}{\sim} \text{Unif}([0,1]^d), \text{ and} \tag{5}$$

$$Y_0 \equiv \{y_{0i} = f_0(u_{0i}) + \epsilon_{0i}\}_{i=1}^{n_0}, \tag{6}$$

$$Y_1 \equiv \{y_{1i} = f_1(u_{1i}) + \epsilon_{1i}\}_{i=1}^{n_1}, \text{ where} \tag{7}$$

$$\epsilon_{ij} \overset{iid}{\sim} \Xi, \ \mathbb{E}[\epsilon_{ij}] = 0, \ \text{Var}[\epsilon_{ij}] \leq \sigma^2 < \infty. \tag{8}$$

Note, Ξ is an error distribution with moment constraints. Furthermore, we shall take $n_1 = O(n_0)$, although this is not necessary for the bounds derived below.

4.2 Basis Functions and Projections

We describe the estimation of functions using orthonormal basis functions. Let $\{\varphi_i\}_{i \in \mathbb{Z}}$ be an orthonormal basis for $L_2([0,1])$, where $L_2(\Omega) = \{f : \Omega \mapsto \mathbb{R} : \int_\Omega f^2 < \infty\}$. Then, the tensor product of $\{\varphi_i\}_{i \in \mathbb{Z}}$ serves as an orthonormal basis for $L_2([0,1]^d)$; that is, the following is an orthonormal basis for $L_2([0,1]^d)$: $\{\varphi_\alpha\}_{\alpha \in \mathbb{Z}^d}$ where $\varphi_\alpha(x) = \prod_{i=1}^d \varphi_{\alpha_i}(x_i)$, $x \in [0,1]^d$. So we have that $\forall \alpha, \zeta \in \mathbb{Z}^d$, $\langle \varphi_\alpha, \varphi_\zeta \rangle = I\{\alpha = \zeta\}$. Let $f \in L_2([0,1]^d)$, then $f(x) = \sum_{\alpha \in \mathbb{Z}^d} a_\alpha(f) \varphi_\alpha(x)$ where $a_\alpha(f) = \langle \varphi_\alpha, f \rangle = \int_{[0,1]^d} \varphi_\alpha(z) f(z) \mathrm{d}z \in \mathbb{R}$.

Suppose function f has a corresponding set of evaluations $Y = \{y_j = f(u_j) + \epsilon_j\}_{j=1}^n$ where $u_j \overset{iid}{\sim} \text{Unif}([0,1]^d)$ and $\mathbb{E}[\epsilon_j] = 0$, $\mathbb{E}[\epsilon_j^2] < \infty$. Then, $\tilde{f}$, the estimate of f, will be:

$$\tilde{f}(x) = \sum_{\alpha \in M} a_\alpha(Y) \varphi_\alpha(x) \quad \text{where} \quad a_\alpha(Y) = \frac{1}{n} \sum_{j=1}^n y_j \varphi_\alpha(u_j), \tag{9}$$

and M is a finite set of indices for basis functions.

4.3 Theory

We bound the L_2 risk of a transfer learning based estimate of f_1: $\mathbb{E}\left[\|f_1 - \hat{f}_1\|_2\right]$. First, we state our assumptions on functions f_0, and f_1:

(a) Sobolev Ellipsoid Function Class Assumptions. We shall make a Sobolev ellipsoid assumption for $f_0, f_1 \in \mathcal{F}$. Let $a(f) \equiv \{a_\alpha(f)\}_{\alpha \in \mathbb{Z}^d}$. Suppose that: $\mathcal{F}_{\gamma,A} = \{f : a(f) \in \Theta_{\gamma,A}, \|f\|_\infty \leq f_{\max}\}$, where $\Theta_{\gamma,A} = \left\{\{\theta_\alpha\}_{\alpha \in \mathbb{Z}^d} : \sum_{\alpha \in \mathbb{Z}^d} \theta_\alpha^2 \kappa_\gamma^2(\alpha) < A^2\right\}$, and $\kappa_\gamma^2(\alpha) = \sum_{i=1}^d |\alpha_i|^{2\gamma_i}$ for $\gamma \in \mathbb{R}_{++}^d$, $f_{\max}, A \in \mathbb{R}_{++}$, $\mathbb{R}_{++} = (0, \infty)$. This assumption will control the tail-behavior of projection coefficients and allow one to effectively estimate $f \in \mathcal{F}$ using a finite number of projection coefficients on the empirical functional observation.

(b) Smooth Difference Assumption. We shall make an additional assumption on the difference between f_1 and f_0, $g(x) \equiv f_1(x) - f_0(x)$: $g = f_1 - f_0 \in \mathcal{F}_{\rho,B}$. Namely, we are imposing a smoothness constraint on the difference between our functions f_0 and f_1, which we will show controls the effectiveness of transfer learning.

Estimator: Before writing our estimator for f_1, we define some terms. First, let $\tilde{f}_0$ be the standard estimator for f_0 based of Y_0, let $M_\gamma(t) \equiv \{\alpha \in \mathbb{Z}^d : \kappa_\gamma(\alpha) \leq t\}$:

$$\tilde{f}_0(x) = \sum_{\alpha \in M_\gamma(t)} a_\alpha(Y_0)\varphi_\alpha(x) \quad \text{where} \tag{10}$$

$$a_\alpha(Y_0) = \frac{1}{n}\sum_{j=1}^{n} y_{0j}\varphi_\alpha(u_{0j}). \tag{11}$$

We will take $\hat{g}$ to be the estimate of g based on Z, where

$$Z \equiv \{z_j = y_{1j} - \tilde{f}_0(u_{1j})\}_{j=1}^{n_1}, \tag{12}$$

$$z_j = f_1(u_{1j}) - \tilde{f}_0(u_{1j}) + \epsilon_{1j} = g(u_{1j}) + r(u_{1j}) + \epsilon_{1j}, \tag{13}$$

and $g(x) = f_1(x) - f_0(x)$, $r(x) = f_0(x) - \tilde{f}_0(x)$. Our estimator for f_1 will then be: $\hat{f}_1(x) = \tilde{f}_0(x) + \hat{g}(x)$, where $\hat{g}$ is the estimate of g based on Z, $\hat{g}(x) = \sum_{\alpha \in M_\rho(v)} a_\alpha(Z)\varphi_\alpha(x)$.

Risk Analysis: We analyze the L_2 risk of our estimator below. Note that: $\mathbb{E}\left[\|f_1 - \hat{f}_1\|_2\right] = \mathbb{E}\left[\|f_0 + g - (\tilde{f}_0 + \hat{g})\|_2\right] \leq \sqrt{\mathbb{E}\left[\|f_0 - \tilde{f}_0\|_2^2\right]} + \sqrt{\mathbb{E}\left[\|g - \hat{g}\|_2^2\right]}$, thus we first upper-bound the risk for typical function estimation $\mathbb{E}\left[\|f_0 - \tilde{f}_0\|_2^2\right]$ then that for the smooth transfer $\mathbb{E}\left[\|g - \hat{g}\|_2^2\right]$. First we analyze the risk of standard functions regression one a single data-set for the estimation of the source function.

Lemma 4.1. *Let* $f_0 \in \mathcal{F}_{\gamma,A}$, *then* $\mathbb{E}\left[\|f_0 - \tilde{f}_0\|_2^2\right] = O\left(n_0^{-\frac{2}{2+\gamma^{-1}}}\right)$, where $\gamma^{-1} = \sum_{i=1}^{d} \gamma_i^{-1}$.

Next we analyze the risk of estimating g from Z (13). Note that Z is not a set of noisy observations from g as Y_0 is to f_0; we, instead have biased observations (from using $\tilde{f}_0$), thus the rate will vary a bit.

Lemma 4.2. *Let* $g \in \mathcal{F}_{\rho,B}$, *then* $\mathbb{E}\left[\|g - \hat{g}\|_2^2\right] = O\left(n_1^{-\frac{2}{2+\rho^{-1}}}\left(1 + \frac{n_1}{n_0}\right)^{\frac{2}{2+\rho^{-1}}}\right)$.

One can see that we pay a penalty of $(1 + n_1/n_0)^{2/(2+\rho^{-1})}$ for using a biased sample to approximate g. As one would expect the penalty diminishes as $n_0 \to \infty$. Note furthermore that if $n_0 \geq n_1$ then this penalty is no more than $2^{\frac{2}{2+\rho^{-1}}} = O(1)$. Hence, the risk of $\hat{g}$ is asymptotically upper-bounded with the same rate as that of the unbiased sample estimator $\tilde{g}$.

Transfer Estimator Risk: Below we state this section's main theorem and discuss some insights gained from it.

Theorem 4.3. *Let* $f_1 \in \mathcal{F}$ *and* $\hat{f}_1(x) \equiv \tilde{f}_0(x) + \hat{g}(x)$, *then:* $\mathbb{E}\left[\|f_1 - \hat{f}_1\|_2\right] = O\left(n_0^{-1/2+\gamma^{-1}} + n_1^{-1/2+\rho^{-1}}(1 + n_1/n_0)^{1/2+\rho^{-1}}\right)$.

For simplification, consider the case where smoothness parameters are $\gamma^\top = (\tau, \ldots, \tau)^\top$ and $\rho^\top = (\nu, \ldots, \nu)^\top$, and $n_0 = n_1^\lambda$ for $\lambda \geq 1$. One then has that: $\mathbb{E}\left[\|f_1 - \hat{f}_1\|_2\right] =$
$= O\left(n_1^{-\frac{\lambda\tau}{2\tau+d}} + n_1^{-\frac{\nu}{2\nu+d}}\left(1 + n_1^{1-\lambda}\right)^{\frac{\nu}{2\nu+d}}\right)$
$= O\left(n_1^{-\frac{\lambda\tau}{2\tau+d}} + n_1^{-\frac{\nu}{2\nu+d}}\right)$. If $\nu > \tau$ (i.e. the difference, g is smoother than each function) and $\lambda > 1$ (the densely sampled function has strictly more samples than the sparsely sampled function), then $\mathbb{E}\left[\|f_1 - \hat{f}_1\|_2\right] = o\left(n_1^{\frac{-\tau}{2\tau+d}}\right)$. That is, we have shown that transfer learning is asymptotically faster than single data-set regression on the target function for the typical case where the target function is similar to the source function and we have more samples from a source function. In fact, if $\nu > \tau$ and $\lambda > 1$, then $\mathbb{E}\left[\|f_1 - \hat{f}_1\|_2\right] = O\left(n_1^{\frac{-\nu}{2\nu+d}}\right)$. In other words, transfer learning has an asymptotic risk of regressing the smooth difference function g with the target sample of size n_1: $\mathbb{E}\left[\|f_1 - \hat{f}_1\|_2\right] = O\left(\mathbb{E}\left[\|g - \tilde{g}\|_2\right]\right)$, where $\tilde{g}$ is defined analogously to (9) with a sample size of n_1. Since functions f_1 and f_0 are similar the asymptotic reduction to the rate of estimation for g proves very beneficial.

Remark. We see that the upper bounds derived for MT-KRR and transfer learning are both affected by the smoothness assumptions we make between/among tasks (the γ, ρ parameter in the above analysis, and the λ_p, λ_s parameter in Sec. 3.3).

5 Experiments

5.1 Synthetic Data

Multi-Task Learning Stability. To show the stability bounds under different penalties, we simulate data with T tasks. Each task t has $\{X_t, Y_t\}$: $Y_t = f_c + f_o + 0.1\epsilon$, where $f_c = \sin(20x) + \sin(10x)$ is the central function, and $f_o = \sin 5(1 + t_i)x$ is a smoother additive function, with $t_i \sim \text{Unif}(0, 1)$, plus $\epsilon \in \mathcal{N}(0, 1)$. Fig.2 (left) shows an example of the data with $T = 3$ and $n_t = 20$ per task.

In Fig.2, we also plot the risk difference $R - R_{emp}$ (Sec. 3.2) w.r.t different number of tasks (fixed 10 points per task), and different number of points per task (with fixed 5 tasks), averaged over 50 experiments. We also plot the theoretical bounds (fitted to the actual curve using regression) for each case. We see that the results are consistent with our analysis. Using central+offset (Eq. 3), pairwise-penalty (Eq. 4), or temporal-penalty (Lemma 3.5) we achieve tighter bounds than learning each task independently (denoted as Separate). In addition, central+offset and pairwise-penalty result in the same curve (red and blue) when we set λ_p/T in central+offset equal to λ_p in pairwise-penalty, which shows the equivalence of these two methods. Further we observe that temporal-penalty gives slightly larger $R - R_{emp}$ than central+offset and pairwise-penalty, which coincides with our analysis.

Transfer Learning Risk. We illustrate the risk of function transfer learning through an experiment with synthetic data. We randomly generate f_0, g, and f_1 and we draw data-sets Y_0, Y_1 with various configurations of n_1 and n_0. We consider the cosine basis: $\varphi_0(x) = 1$, $\varphi_k(x) = \sqrt{2}\cos(\pi k x)$, $\forall\ k \geq 1$.

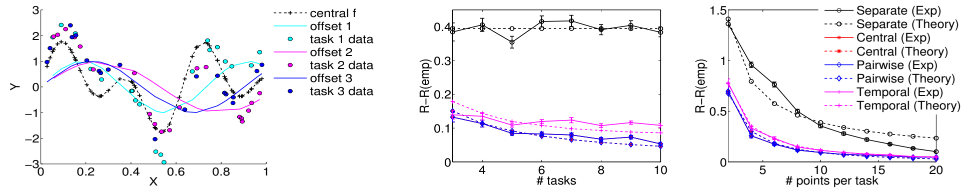
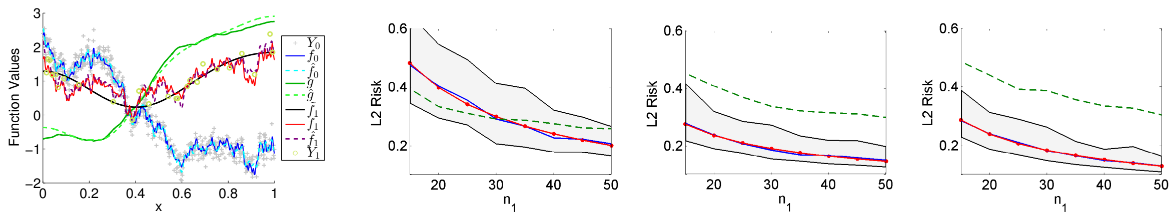


Figure 2: Left: Example data with $T = 3, n_t = 20$ (left); Center: $R - R_{emp}$ w.r.t # tasks (fixed $n_t = 10$ points per task); Right: $R - R_{emp}$ w.r.t # points per task (fixed $T = 5$ tasks)



(a) Synthetic data (b) $\gamma_g = 1.25$ (c) $\gamma_g = 1.75$ (d) $\gamma_g = 2.25$

Figure 3: (a) Example synthetic data for $n_1 = 25$. Note the better quality of the transfer learning estimate $\hat{f}_1$, to the estimate based only on Y_1, $\tilde{f}_1$. (b)-(d) Risk estimation for various γ_g values; the .1 to .9 percentiles are filled in gray, the mean of the data within this range is in blue, and the best fit curve (of constants) of our upperbounding rate is shown in red markers, the risk estimating on f_1 using only Y_1 is show in dashed green.

We define $f_0(x) \equiv \sum_{k=0}^{M} \theta_k^{(0)} \varphi_k(x)$, by generating the projection coefficients $\theta^{(0)}$, $M = 500$. See Fig.3 (a) for example of functions. Specifically we consider $n_1 \in \{15, 20, \ldots, 50\}$ and $n_0 = n_1^2$. For each configuration, we draw 100 instances of Y_0 and Y_1 and we estimate the risk at (n_0, n_1) by cross-validating the set of projection coefficients and calculating the loss of the estimate, and taking the mean over the 100 instances of Y_0 and Y_1. The risk estimation was performed for the values of $\gamma_g \in \{1.25, 1.75, 2.25\}$ and $\gamma_0 = 1$, keeping f_0 constant throughout and changing g per value of γ_g (see Fig.3 (b-d)). As one would expect given our analysis, as the smoothness of g increases (i.e. γ_g increases) so too does the efficacy of transfer learning. It is interesting to note that transfer learning outperforms typical regression in all scenarios except for when one does not have a smooth offset and many source data samples; this too is consistent with our analysis. Lastly, it is worth noting that we can achieve a good fit of our upperbound on risks of transfer learning (Theorem 4.3).

5.2 Real Data

The real dataset is the Air Quality Index (AQI) dataset [Mei *et al.*, 2014]. We extract bag-of-words vectors (feature X with dimension $d = 100,395$) from social media posts to predict the AQI (label Y) across cities. The results are averaged over 20 experiments. In Fig. 4 (left), we show the prediction error of MT-KRR using pairwise penalty (or equivalently the central+offset penalty) with 4 cities as 4 different tasks. We see that the MT-KRR algorithm (mtl) outperforms independent-task-learning (ind). In addition, we plot the leave-one-out error for each task (loo-1 through 4), and the prediction error by MT-KRR for the best task (mtl-min), which outperforms learning that task by itself (loo-3). Fig. 4 (right) shows the prediction error using the transfer method analyzed in this paper, compared with state-of-the-art baselines. The transfer method benefits from modeling a smoother offset across domains compared to optDA [Chattopadhyay *et al.*, 2011] with single-source, and it also outperforms KMM [Huang *et al.*, 2007] by allowing changes in $P(Y|X)$.

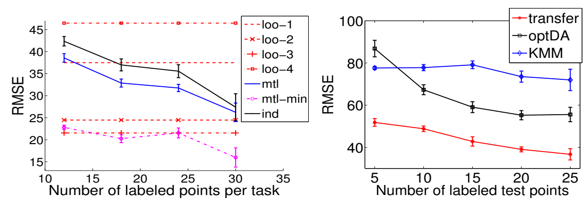


Figure 4: Results for multi-task learning (left) and transfer learning (right) on the AQI data

6 Conclusion

In this paper we provide theory that connects the risk bounds for both transfer and multi-task learning to the relation of tasks. We show that, by imposing a smooth relationship between/among tasks, we obtain favorable learning rates for both algorithms, compared to learning tasks independently.

Acknowledgments

This work is funded in part by DARPA grants FA87501220324, FA87501420244, NSF IIS-1247658 and DOE DE-SC0011114 grants.

References

[Audiffren and Kadri, 2013] Julien Audiffren and Hachen Kadri. Stability of multi-task kernel regression algorithms. In *ACML*, 2013.

[Bousquet and Elisseeff, 2002] Olivier Bousquet and Andre Elisseeff. Stability and generalization. In *JMLR*, 2002.

[Chattopadhyay *et al.*, 2011] Rita Chattopadhyay, Jieping Ye, Sethuraman Panchanathan, Wei Fan, and Ian Davidson. Multi-source domain adaptation and its application to early detection of fatigue. In *KDD*, 2011.

[Cortes *et al.*, 2008] Corinna Cortes, Mehryar Mohri, Dmitry Pechyony, and Ashish Rastogi. Stability of transductive regression algorithms. In *ICML*, 2008.

[Evgeniou and Pontil, 2004] T. Evgeniou and M. Pontil. Regularized multi-task learning. In *KDD*, 2004.

[Evgeniou *et al.*, 2005] Theodoros Evgeniou, Charles A. Micchelli, and Massimiliano Pontil. Learning multiple tasks with kernel methods. In *JMLR*, 2005.

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

[Györfi, 2002] László Györfi. *A distribution-free theory of nonparametric regression*. Springer Science & Business Media, 2002.

[Hoerl and Kennard, 1970] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

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

[Maurer and Pontil, 2013] Andreas Maurer and Massimiliano Pontil. Excess risk bounds for multitask learning with trace norm regularization. In *JMLR*, 2013.

[Mei *et al.*, 2014] Shike Mei, Han Li, Jing Fan, Xiaojin Zhu, and Charles R. Dyer. Inferring air pollution by sniffing social media. In *ASONAM*, 2014.

[Reddi *et al.*, 2015] Sashank J. Reddi, Barnabas Poczos, and Alex Smola. Doubly robust covariate shift correction. In *AAAI*, 2015.

[Romera-Paredes *et al.*, 2013] Bernardino Romera-Paredes, Min Hane Aung, Nadia Bianchi-Berthouze, and Massimiliano Pontil. Multilinear multitask learning. In *ICML*, 2013.

[Shimodaira, 2000] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. In *Journal of Statistical Planning and Inference*, 2000.

[Solnon *et al.*, 2013] Matthieu Solnon, Sylvain Arlot, and Francis Bach. Multi-task regression using minimal penalties. In *JMLR*, 2013.

[Tsybakov, 2009] AB Tsybakov. Introduction to nonparametric estimation. revised and extended from the 2004 french original. translated by vladimir zaiats, 2009.

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

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

[Wen *et al.*, 2014] Junfeng Wen, Chun-Nam Yu, and Russ Greiner. Robust learning under uncertain test distributions: Relating covariate shift to model misspecification. In *International Conference on Machine Learning (ICML)*, 2014.

[Yu and Szepesvri, 2012] Yaoliang Yu and Csaba Szepesvri. Analysis of kernel mean matching under covariate shift. In *In ICML*, 2012.

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

[Zhang, 2015] Yu Zhang. Multi-task learning and algorithmic stability. In *AAAI*, 2015.

[Zhong and Kwok, 2012] Leon Wenliang Zhong and James T. Kwok. Convex multitask learning with flexible task clusters. In *ICML*, 2012.

[Zhou *et al.*, 2011] Jiayu Zhou, Lei Yuan, Jun Liu, and Jieping Ye. A multi-task learning formulation for predicting disease progression. In *KDD*, 2011.