Algorithm-Dependent Generalization Bounds for Multi-Task Learning

Tongliang Liu[1], Dacheng Tao[1], Mingli Song[2], and Stephen J. Maybank[3]

[1] the Centre for Quantum Computation & Intelligent Systems,

University of Technology Sydney, Australia.

E-mail: tliang.liu@gmail.com, dacheng.tao@uts.edu.au.

[2] the College of Computer Science, Zhejiang University, China.

E-mail: mingls@uw.edu.

[3] the Department of Computer Science and Information Systems,

Birkbeck College, UK. E-mail: sjmaybank@dcs.bbk.ac.uk.

Abstract

Often, tasks are collected for multi-task learning (MTL) because they share similar feature structures. Based on this observation, in this paper, we present novel algorithm-dependent generalization bounds for MTL by exploiting the notion of algorithmic stability. We focus on the performance of one particular task and the average performance over multiple tasks by analyzing the generalization ability of a common parameter that is shared in MTL. When focusing on one particular task, with the help of a mild assumption on the feature structures, we interpret the function of the other tasks as a regularizer that produces a specific inductive bias. The algorithm for learning the common parameter, as well as the predictor, is thereby uniformly stable with respect to the domain of the particular task and has a generalization bound with a fast convergence rate of order $\mathcal{O}(1/n)$, where n is the sample size of the particular task. When focusing on the average performance over multiple tasks, we prove that a similar inductive bias exists under certain conditions on the feature structures. Thus, the corresponding algorithm for learning the common parameter is also uniformly stable with respect to the domains of the multiple tasks, and its generalization bound is of the order $\mathcal{O}(1/T)$, where T is the number of tasks. These theoretical analyses naturally show that the similarity of feature structures in MTL will lead to specific regularizations for predicting, which enables the learning algorithms to generalize fast and correctly from a few examples.

1 Introduction

Multi-task learning (MTL) has been proposed by Caruna (Caruna, 1993) to more efficiently learn several related tasks simultaneously by using the domain information of the related tasks as an inductive bias; therefore, it is superior to the traditional single-task learning because it more efficiently learns the shared information among the multiple tasks. Multi-task learning has achieved great success in machine learning for its appealing performances on a broad spectrum of applications (Wang et al., 2009; Y. Zhang &

Yeung, 2010; T. Zhang et al., 2012; Pillonetto et al., 2010; Collobert & Weston, 2008; Argyriou et al., 2008; Gong et al., 2014; X.-L. Zhang, 2015; Chen et al., 2013).

There have been some notable theoretical justifications (Baxter, 2000; Ben-David & Schuller, 2003; Ando & Zhang, 2005; Maurer, 2006, 2009; Maurer et al., 2013; Micchelli & Pontil, 2004) for the success of MTL. All of the theoretical justifications are focused on the average performance over all of the multiple tasks and are independent of the algorithms. However, as stated by Ando and Zhang (Ando & Zhang, 2005), in practice, we are often very interested in the performance of some particular task in an MTL problem. Comparing the performance of one particular task in the MTL setting with that of the traditional single-task learning is both necessary and highly important. However, such a comparison has remained elusive. In this paper, by providing novel algorithm-dependent generalization bounds, we analyze the performance of one particular task as well as the average performance over all of the multiple tasks for the MTL algorithms, which employ learning parameters to model the shared information among tasks.

Although some of the previous results state that the tasks in MTL that are learned jointly are "algorithmically related" because the tasks share a common optimal hypothesis class (see, for examples, (Baxter, 2000; Ben-David & Schuller, 2003)), most of the existing proof methods have been based on some measurements of the complexities of the whole hypothesis class and are independent of any of the algorithms. Such measurements include the VC-dimension (Shawe-Taylor et al., 1998; Vapnik, 2000), covering number (P. Bartlett et al., 1997; T. Zhang, 2002; D.-X. Zhou, 2003; Guo et al., 2002) and Rademacher complexity (Koltchinskii, 2001; P. L. Bartlett & Mendelson, 2003).

When the complexity measures of the VC-dimension or covering number were used, as discussed in (Baxter, 2000; Ben-David & Schuller, 2003; Ando & Zhang, 2005), the convergence rate obtained for the generalization bounds is of the order $\mathcal{O}(\sqrt{\log n/n})$ with respect to n, which is the training sample size, and of the order $\mathcal{O}(\sqrt{\log T/T})$ with respect to T, which is the number of related tasks. If the Rademacher complexity was used to measure the hypothesis class (see, for examples, (Maurer, 2006; Maurer et al., 2013; Maurer, 2009)), the obtained convergence rate is of order $\mathcal{O}(\sqrt{1/n})$ with respect to n and of order $\mathcal{O}(\sqrt{1/T})$ with respect to T. These convergence rates are slow because they are derived in such a way as to be dependent on the complexities of the whole set of the predefined hypothesis classes and independent of any of the algorithms. However, in this paper, we investigate the advantages of MTL by exploiting the notion of algorithmic stability, and we take a step forward from previous studies, where theoretical analyses are algorithm-independent, to derive algorithm-dependent generalization bounds that have fast convergence rates of order $\mathcal{O}(1/n)$ with respect to n, and of order $\mathcal{O}(1/T)$ with respect to T.

We show that stability analysis (Bousquet & Elisseeff, 2002) is more suitable for analyzing MTL. It can be used to illustrate that tasks in MTL can produce regularization. Based on the observation that tasks in MTL are usually chosen because the corresponding feature structures are similar, we prove that the algorithms of either one particular task or the overall multiple tasks for learning a common parameter that is shared in MTL are uniformly stable under certain conditions. Specifically, if a mild assumption is made on the structure of the data matrix and if the loss function ℓ is strongly convex, the common parameter that is shared by the related tasks will be learned with a fast convergence rate.

When we focus on the performance of one particular task instead of the average performance over all of the tasks, the other tasks can act as a regularizer. Specifically, if any feature vector of the focused task can be (approximately) reconstructed by the observations of the other tasks, the algorithm for learning the common parameter, as well as the predictor, will be uniformly stable with respect to the domain of the particular task and have an algorithm-dependent generalization bound with a fast convergence rate with respect to the sample size of the particular task. When we focus on the average performance over multiple tasks, the multiple tasks can also generate an inductive bias that will not vanish as the number of tasks goes to infinity. Such an inductive bias will act as a regularizer for the optimization procedure. The algorithm for learning the common parameter for multiple tasks is thereby uniformly stable with respect to the domains of the multiple tasks, and the corresponding generalization bound has a fast convergence rate with respect to the number of tasks. These analyses naturally show that if the related tasks are chosen carefully, the tasks in MTL will produce biased regularizers that are based on feature structures. MTL is therefore superior to the traditional single-task learning.

We illustrate the uniform stability property for MTL algorithms by exploiting their feature structures, which also provides a new insight into deriving the stability property for learning algorithms. Previous methods have shown that many learning algorithms exhibit a uniform stability relying on L_2 regularization (Y. Zhang, 2015; Audiffren & Kadri, 2013). However, our approach relies on more meaningful regularization, which can be reformulated to have a specific regularization matrix. Such a meaningful regularization matrix is based on feature structures and is conveyed from carefully collected tasks. Consequently, the analysis is easy to extend to many existing learning algorithms, such as the learning to learn (LTL) algorithms (Baxter, 2000; Maurer, 2009; Maurer et al., 2013). We provide the extension in the supplementary material.

By interpreting the function of some tasks in MTL as biased regularizers and by proving that the learning algorithm is thereby uniformly stable, this work also attempts to address an open question that was asked by Elisseeff and Pontil (Elisseeff & Pontil, 2003): "*Is there a way to incorporate prior knowledge via stability?*" Kuzborskij and Orabona (Kuzborskij & Orabona, 2013) tried to address this open question by posing a connection between hypothesis stability and hypothesis transfer learning, but they failed to improve the convergence rate by exploiting the uniform stability. In this paper, we illustrate that MTL has successfully incorporated prior knowledge into learning algorithms and that the learning algorithms are therefore uniformly stable. Moreover, the learning algorithms have fast convergence rates for generalization.

1.1 Related Work

There have been many results on MTL. We briefly summarize the related theoretical studies. Baxter (Baxter, 2000) proposed the model of inductive bias learning and extended it to MTL problems. He provided generalization bounds by analyzing the VC-dimension and covering number of the hypothesis class. MTL will benefit if the tasks share a common optimal hypothesis class. To define a common optimal hypothesis class produces an inductive bias. Many theoretical justifications for MTL have then been followed by exploiting specific inductive biases.

Ben-David and Schuller (Ben-David & Schuller, 2003) offered a data generating mechanism through which the relationships between the tasks are measured. Based on the notion of task-relatedness, they provided a tighter generalization bound than that provided in (Baxter, 2000) for MTL by also analyzing the VC-dimension. Ando and Zhang (Ando & Zhang, 2005) assumed that there was a common structure parameter that is shared by all of the tasks. They then proved that the shared parameter can be reliably estimated when the task number T is large by using a covering number definition different from that in (Baxter, 2000). The analysis is closely related to that of Baxter (Baxter, 2000).

Maurer (Maurer, 2006) studied the linear MTL problem where a common linear operator is chosen to preprocess the data matrices before learning multiple related tasks. He provided a generalization bound by exploiting the Rademacher complexity and illustrated the advantages of MTL by employing a proper common linear operator. Maurer et al. (Maurer et al., 2013) investigated the use of sparse coding and dictionary learning in the context of MTL. They assumed that the task parameters are well approximated by sparse linear combinations of the atoms in a dictionary and provided a generalization bound by exploiting the Rademacher complexity to measure the hypothesis complexity.

Micchelli and Pontil (Micchelli & Pontil, 2004) provided a framework of vector-valued functions and discussed their use in MTL. This approach can be theoretically justified using the notion of task-relatedness discussed in (Ben-David & Schuller, 2003). Recently, trace norm regularization has been proposed and has become popular for MTL (Argyriou et al., 2007; Pong et al., 2010). Maurer and Pontil (Pontil & Maurer, 2013) exploited the Rademacher complexity method to provide excess risk bounds for MTL problems that were regularized by the trace norm. Lounici (Lounici et al., 2009) considered the group lasso regularization for MTL and showed that under certain restricted eigenvalue conditions, the effect of the number of predictor variables in the upper bound of sparsity oracle inequalities could be negligible with respect to the number of tasks.

1.2 Main Contributions

The main results and contributions of this paper are summarized below:

1. We prove that the sample average stability is upper bounded by the Rademacher complexity. Previous results have shown that the Rademacher complexity is upper bounded by functions of the VC-dimension or covering number. Thus, algorithmic stability can be used to derive tighter upper generalization bounds than the VC-dimension, covering number and Rademacher complexity.

2. To the best of our knowledge, we are the first to analyze the performance of individual tasks in MTL. We thereby illustrate the superiority of MTL to traditional single-task learning.

3. We prove that the algorithm of MTL is uniformly stable under mild conditions and thereby provide algorithm-dependent generalization bounds. The generalization bound of the algorithm for learning one particular task (or the focused task) has a fast convergence rate of order $\mathcal{O}(1/n)$, where n is the sample size of the particular task. In addition, the generalization bound of MTL has a fast convergence rate of order $\mathcal{O}(1/T)$, where T is the number of the overall multiple tasks.

This paper is organized as follows. In Section II, we describe MTL, introduce the algorithm stability and upper bound the sample average stability using the Rademacher complexity. In Section III, we present algorithm-dependent generalization bounds for MTL. The proofs of our results are presented in Section IV. Finally, Section V concludes the paper.

2 Preliminaries

In this section, we first set up an MTL problem in which the different tasks share a common parameter, which can be viewed as an inductive bias. Then, we present the notion of algorithm stability and show that, for the MTL problem, the stability is more suitable for deriving generalization bounds than the complexity measures of the VC-dimension, covering number and Rademacher complexity.

Let $\mathcal{H}$ denote a finite or infinite dimensional separable real Hilbert space with the inner product $\langle \cdot, \cdot \rangle$ and norm $\| \cdot \|$, let $\mathbb{R}$ be the Euclidean space and let $z = (x, y) \in \mathcal{H} \times \{-1, +1\}$ be a training example, where x denotes a feature vector (or observation) and y represents the corresponding real-valued label. We denote $S = \{z_1, \ldots, z_n\} = \{(x_1, y_1), \ldots, (x_n, y_n)\} \in (\mathcal{H} \times \{-1, +1\})^n$ as a training sample.

Let $S_t = \{z_{t,1}, \ldots, z_{t,n_t}\} = \{(x_{t,1}, y_{t,1}), \ldots, (x_{t,n_t}, y_{t,n_t})\}$ denote the training sample for the t-th task, and let $\ell(y, h(x))$ measure the loss that is incurred by predicting $h(x)$ when the true label is y, where $h \in H$ and H is a linear function class[1] (also called hypothesis class). We analyze the following setting for MTL:

$$\min_{w_1, \ldots, w_T, \theta} \frac{1}{T} \sum_{t=1}^{T} \frac{1}{n_t} \sum_{i=1}^{n_t} \ell \left(y_{t,i}, \langle w_t + \theta, x_{t,i} \rangle \right). \tag{1}$$

where $\theta \in H$ is a common parameter that is shared by all of the tasks in MTL, with

[1] In this paper, H can also be a reproducing kernel Hilbert space.

which the tasks are related, and $w_t \in H, t = 1, \ldots, T$, are the predictors which are specific for different tasks. Note that the predictor for the t-th task is $h_t = w_t + \theta \in H$ and that some constraints should be further placed on w_t, or θ, or both, to make the MTL in (1) well-posed because there is a trade-off between w_t and θ. Many different constraints based on the inherent properties of MTL or the prior knowledge about its specific applications have been therefore proposed. For example, the regularized MTL (Evgeniou & Pontil, 2004) models $\sum_{t=1}^{T} \|w_t\|^2$ to be small and θ to be smooth; the trace norm regularized MTL (Pong et al., 2010) also essentially forces w_t to be small; the multi-task feature learning algorithm (Argyriou et al., 2008) employs a group sparse constraint on w_t. However, in this paper, we do not consider explicit constraints and simply assume that w_t and θ in (1) can be learned. The obtained results apply to all the constrained MTL problems because the employed constraints will only shrink the search space of the parameters to be learned[2].

There are many interesting MTL problems that can be treated directly within our setting (see, for examples, (Ando & Zhang, 2005; Evgeniou & Pontil, 2004; Chen et al., 2009; Rai & Daume, 2010; J. Zhou et al., 2011; Kumar & Daume, 2012; Lin et al., 2012)). Not surprisingly, some potential MTL scenarios are outside of our setting, such as (Liu et al., 2009). However, our analyses can be easily extended to justify a much more general form of problem (1), where some but not all of the tasks share common parameters. Then, we can discuss the models involving outlier tasks. And many other

[2] Note that for case where the constraints on $w_t, t = 1, \ldots, T$, and θ are positive and convex, due to the additive and non-negative properties of Bregman divergence as shown in Lemma 1, the proof methods provided in this paper can be easily extended to the constrained MTL problems to imporve the results.

MTL scenarios are within the scope of our discussion. One example is MTL based on coding schemes (Maurer et al., 2013).

Existing generalization bounds for MTL have relied on complexity measures such as the VC-dimension, covering number and Rademacher complexity. The obtained bounds are therefore dependent on the complexities of the predefined hypothesis classes and are independent of the learning algorithms. However, in this paper, we will use the notion of algorithmic stability other than the notions of VC-dimension, covering number and Rademacher complexity to derive algorithm-dependent generalization bounds for MTL. In particular, stability is a property of an learning algorithm, i.e., if two training samples are close to each other, a stable algorithm will output close predictors. There are many versions of stability, such as the hypothesis stability (Kearns & Ron, 1999), sample average stability (Shalev-Shwartz et al., 2010) and uniform stability (Bousquet & Elisseeff, 2002). We will focus on the uniform stability, to which the other types of stability are closely related.

Definition 1 (Uniform stability) *An algorithm is uniformly stable (or β uniformly stable) with respect to the loss function ℓ and a specific domain $\mathcal{Z} \subset \mathcal{H} \times \{-1, +1\}$ if the following holds:*

$$\forall S \in \mathcal{Z}^n,\ \forall i \in \{1, ..., n\}, \forall z = (x, y), z_i' \in \mathcal{Z},$$

$$|\ell(y, h_S(x)) - \ell(y, h_{S^i}(x))| \leq \beta,$$

where h_S is the hypothesis function that is returned by the learning algorithm when the input training sample is S, and S^i denotes the training sample S with the i-th example z_i replaced by an independent and identically distributed example z_i'.

Note that, in this paper, we say an algorithm is uniform stable if the minimum value of β converges to zero as the training sample size increases without limit.

To illustrate that stability is a more subtle notion than the VC-dimension, covering number and Rademacher complexity for deriving upper generalization bounds, we will employ the sample average stability defined in (Shalev-Shwartz et al., 2010) and show that it is upper bounded by the VC-dimension, covering number and Rademacher complexity.

Definition 2 (Sample average stability) *An algorithm is sample average stable (or γ sample average stable) with respect to the loss function ℓ and a specific domain $\mathcal{Z} \subset \mathcal{H} \times \{-1, +1\}$ if the following inequality holds:*

$$\left| \frac{1}{n} \sum_{i=1}^{n} E_{S \sim D^n, \{z'_1=(x'_1,y'_1),\ldots,z'_n=(x'_n,y'_n)\} \sim D^n} [\ell(y'_i, h_{S^i}(x'_i)) - \ell(y'_i, h_S(x'_i))] \right| \leq \gamma,$$

where D denotes the distribution over the domain $\mathcal{Z}$ for generating the training example z.

Remark 1 *For any $S, S' = \{z'_1, \ldots, z'_n\} \in \mathcal{Z}^n$, the relationship between the uniform stability and sample average stability is shown by the following inequality:*

$$\begin{aligned} &\left| \frac{1}{n} \sum_{i=1}^{n} E_{S \sim D^n, \{z'_1,\ldots,z'_n\} \sim D^n} [\ell(y'_i, h_{S^i}(x'_i)) - \ell(y'_i, h_S(x'_i))] \right| \\ &\leq \max_{z=(x,y) \in \mathcal{Z}, i \in \{1,\ldots,n\}} |\ell(y, h_S(x)) - \ell(y, h_{S^i}(x))| \leq \beta. \end{aligned}$$

Previous results show that the Rademacher complexity could be upper bounded with respect to the VC-dimension and covering number, respectively. For example, by combining Massart's Lemma (Massart, 2000) and Sauer's Lemma (Sauer, 1972), it can be proven that the Rademacher complexity is upper bounded by a function of the VC-dimension; Dudley (Dudley, 1967) showed that the Rademacher complexity is upper bounded by a function of the covering number. Thus, the Rademacher complexity method has the potential to derive tighter upper generalization bounds than approaches based on the VC-dimension and covering number, and therefore, it has been widely used to analyze generalization bounds.

We show that the sample average stability is upper bounded by the Rademacher complexity, as follows.

Theorem 1 *The sample average stability is upper bounded by Rademacher complexity*

$$\left|\frac{1}{n}\sum_{i=1}^{n} E_{S\sim D^n,\{z'_1,\dots,z'_n\}\sim D^n}[\ell(y'_i, h_{S^i}(x'_i)) - \ell(y'_i, h_S(x'_i))]\right| \leq 2\mathfrak{R}(\ell \circ H).$$

The Rademacher complexity $\mathfrak{R}(\ell \circ H)$ is defined by

$$\mathfrak{R}(\ell \circ H) = E_{S,\sigma} \sup_{h_s \in H} \frac{1}{n}\left|\sum_{i=1}^{n} \sigma_i \ell(y_i, h_s(x_i))\right|,$$

where H is the predefined linear hypothesis class, $\ell \circ H$ denotes the set of compositions of functions ℓ and $h \in H$, and $\sigma_1, \dots, \sigma_n$ are the independent Rademacher variables that are uniformly distributed on $\{-1, 1\}$.

See the proof in Section 4.2.

Theorem 1 shows that the algorithm stability has the potential for deriving tighter generalization bounds than the VC-dimension, covering number and Rademacher complexity.

In contrast to the previous theoretical analyses of uniform stability derived from L_2 regularization, we propose a new method for proving uniform stability based on the feature structures. We show that the learning algorithms for MTL are uniformly stable. Details are presented in the next section.

3 Main Results

In this section, we provide algorithm-dependent generalization bounds for MTL. These results are based on the idea that when related tasks are learned at the same time, tasks can function as regularizers for predicting.

Note that the related tasks for MTL are usually chosen because of the similarity in their feature structures. We formulate the prior as the following assumption:

Assumption 1 *If we focus on the j-th task, $j \in \{1, \dots, T\}$, there exists a subset $B = \{b_1, \dots, b_N\} \subset \{x_{1,1}, \dots, x_{T,n_T}\} - \{x_{j,1}, \dots, x_{j,n_j}\}$ such that for any feature vector x distributed from the j-th task, x can be reconstructed by B with a small reconstruction error, i.e., $x = \sum_{j=1}^{N} \alpha_j b_j + \eta$, where $\alpha_j \in \mathbb{R}, \|\alpha\| \leq r$, and η is a small error that satisfies $\|\eta\| \leq \epsilon$. If we focus on the whole multiple tasks, for any task $t, t = 1, \dots, T$, there also exists a subset $B_t = \{b_{t,1}, \dots, b_{t,N_t}\} \subset \{x_{t,1}, \dots, x_{t,n_t}\}$ such that for any feature vector x distributed from any task, x can be reconstructed by B with a small reconstruction error.*

We note that Assumption 1 is very mild. If the feature space is of low-rank or the data lies on a manifold, the assumption can be easily satisfied even for the traditional single task learning problem. If the feature vectors are randomized, the assumption will also hold if the sample size reaches the dimension of the feature vector.

Before presenting our main results, we introduce strongly convex loss functions:

Definition 3 (Strongly convex) *A differentiable loss function $\ell(y, h(x))$ is c-strongly convex if the following inequality holds for any two hypotheses $h, h' \in H$:*

$$(\nabla \ell(y, h(x)) - \nabla \ell(y, h'(x)))^T (h(x) - h'(x)) \geq c\|h(x) - h'(x)\|^2,$$

where $c \in \mathbb{R}_+$ and $\nabla\ell(y, h(x))$ denotes the gradient of the loss function $\ell(y, h(x))$ with respect to $h(x)$.

Remark 2 *The quadratic loss function $\ell(y, h(x)) = (y - h(x))^2$ is 2-strongly convex and has been widely used in many scientific fields. Many other frequently used loss functions, such as hinge loss and logistic loss, are only convex but not strongly convex. However, in statistical learning theory, we often assume that $h(x)$ is bounded, e.g., $h(x) \in [-U, U]$, where U is a positive constant. In this case, the loss functions may be strongly convex. For example, the logistic loss $\ell(y, h(x)) = \log(1 + \exp(-yh(x)))$ is $\exp(-U)/4$-strongly convex when $h(x)$ is restricted to the interval $[-U, U]$, because $d^2\ell(y, h(x))/d^2h(x) = \exp(yh(x)) \,/(\exp(yh(x)) + 1)^2 \geq \exp(-U)/4$. Note that the strong convexity (Hazan & Kale, 2011) and strong smoothness are dual properties, strongly convex programming algorithms have many benign properties both on the speed of optimization and the quality of generalization; see, for examples, (Hazan & Kale, 2011; Rakhlin et al., 2012; Tsianos & Rabbat, 2012; Kakade & Tewari, 2009).*

3.1 Algorithm-Dependent Generalization Bounds

Instead of providing the most general analysis with the tightest possible generalization bounds for the task predictors in MTL, we present the tightest possible generalization bounds for learning the shared parameter θ in (1). Our purpose is to illustrate the main benefit of MTL, which is that the shared parameter can be more accurately estimated. Moreover, we focus on both the performance of one particular task and the average performance over all of the multiple tasks.

For the first time in the literature, we provide algorithm-dependent theoretical analysis for MTL by showing the property of uniform stability, which is upper bounded by $\mathcal{O}(1/n)$ or $\mathcal{O}(1/T)$, for the learning algorithms of MTL. To upper bound the uniform stability, we assume that the loss function ℓ satisfies the following Lipschitz-like condition, which has been widely used (see, for examples, (Bousquet & Elisseeff, 2002; Mohri et al., 2012)):

Definition 4 *A loss function ℓ is σ-admissible with respect to the hypothesis class H if there exists $\sigma \in \mathbb{R}_+$ such that for any two hypotheses $h, h' \in H$ and any example $z \in \mathcal{H} \times \{-1, +1\}$, the following inequality holds:*

$$|\ell(y, h(x)) - \ell(y, h'(x))| \leq \sigma |h(x) - h'(x)|.$$

Proposition 1 *Let focus on the j-th task in the multi-task learning problem (1). Let Assumption 1 hold that there exists a subset $B \subseteq \{x_{1,1}, \ldots, x_{T,n_T}\} - \{x_{j,1}, \ldots, x_{j,n_t}\}$ such that for any feature vector x distributed from the j-th task, it holds that $x = \sum_{j=1}^{N} \alpha_j b_j + \eta$, where $\alpha_j \in \mathbb{R}, \|\alpha\| \leq r$, and η is a small error that satisfies $\|\eta\| \leq \epsilon$. Let the loss function ℓ be c-strongly convex and σ-admissible. Then, the algorithm for learning θ is uniformly stable with respect to the domain of the j-th task. That is, for any $z_j = (x_j, y_j)$ distributed from the j-th task, any $\theta_{S_j^i}$ and θ_{S_j} learned by algorithm (1), given w_j, the following inequality holds:*

$$\begin{aligned} &\left|\ell\left(y_j, \left\langle w_j + \theta_{S_j^i}, x_j \right\rangle\right) - \ell\left(y_j, \left\langle w_j + \theta_{S_j}, x_j \right\rangle\right)\right| \leq \max_{z_j = (x_j, y_j) \in \mathcal{Z}_j} \sigma \left|\left\langle \theta_{S_j^i} - \theta_{S_j}, x_j \right\rangle\right| \\ &\leq \frac{\sigma r \max\{n_t : t \neq j\}}{2c} \left(\sqrt{\left(\frac{2\sigma r}{n_j}\right)^2 + \frac{4c\mathcal{O}(\epsilon)}{n_j \max\{n_t : t \neq j\}}} + \frac{2\sigma r}{n_j} \right), \end{aligned}$$

where S_j is the training sample for the j-th task, S_j^i is the training sample of the j-th task with the i-th example $z_i, i \in \{1, \ldots, n_j\}$, replaced by another independent and *identically distributed example z_i', θ_{S_j} denotes the shared parameter θ learned by algorithm (1) when the j-th task has the training sample S_j, $\mathcal{Z}_j$ denotes the domain of the j-th task, and n_j represents the training sample size of the j-th task.*

For simplicity, let $\epsilon = 0$, and we have

$$\begin{aligned}
&\left|\ell\left(y_j, \left\langle w_j + \theta_{S_j^i}, x_j\right\rangle\right) - \ell\left(y_j, \left\langle w_j + \theta_{S_j}, x_j\right\rangle\right)\right| \\
&\leq \max_{z_j=(x_j,y_j)\in\mathcal{Z}_j} \sigma \left|\left\langle \theta_{S_1^i} - \theta_{S_1}, x_j\right\rangle\right| \\
&\leq \frac{2\sigma^2 r^2 \max\{n_t : t \neq j\}}{n_j c}.
\end{aligned}$$

See the proof in Section 4.3.

Note that n_j denotes the training sample size of the j-th task. When we focus on the j-th tasks, the sample sizes of the other tasks, which are $\{n_t : t \neq j\}$, should be fixed, then the upper bounds in Proposition 1 will decrease quickly as the training sample size of the j-th task is increased.

Remark 3 *For simplicity, we will consider $\eta = 0$ (or $\epsilon = 0$) in the remainder of the paper. However, our results could be easily extended to the case of $\eta \neq 0$, as in the case shown in Proposition 1. We note that the upper bounds are independent of the number N of representative observations B. Thus, we could increase N to obtain a small $\|\eta\|$.*

Remark 4 *According to the proof method of Proposition 1, when we focus on one particular task, we interpret the function of the other tasks as regularizers. The proof of Proposition 1 can be interpreted to rely on regularization $\lambda\|\Gamma\theta\|_2^2$, where λ is a regularization parameter that is dependent on the training samples of the unfocused tasks, and Γ, which is dependent on the representative observations B (defined in Assumption 1), is referred to as the regularization matrix. More details are illustrated in the proof* *of Proposition 1 and Remark 16. The superiority of MTL could therefore be explained by the fact that a proper inductive bias, the regularization $\lambda\|\Gamma\theta\|_2^2$, has been carefully collected for the focused task. The algorithm for learning the parameter θ is therefore uniformly stable with respect to the domain of the focused task. As shown in Proposition 1, when we increase the training sample size n_j of the j-th task, the upper bound will decease fast with order $\mathcal{O}(1/n_j)$.*

Remark 5 *Our analyses are different from the idea of regularizing a projected version of the shared parameter in a new space, because the regularization matrix Γ, which is constructed from the representative observations B is not necessarily a projection matrix. Moreover, the regularization matrix Γ (or the representative observations B) could be over-complete, for the construction of the observations $x \in \mathcal{H}$.*

A generalization bound for learning θ can be easily derived using the upper bound of uniform stability presented in Proposition 1.

Proposition 2 *Let focus on the j-th task in the MTL problem (1). Let Assumption 1 hold and $\eta = 0$. Let the loss function ℓ be c-strongly convex, σ-admissible and upper bounded by M. Let $\mu_1, \ldots, \mu_T$ be probability measures on the domains of T different tasks. Let the set $\{n_t : t \neq j\}$ be fixed. For any learned θ and any $\delta > 0$, given w_j, with probability at least $1 - \delta$, the following inequality holds:*

$$\begin{aligned}
&E_{z_j=(x_j,y_j)\sim\mu_j}\ell\left(y_j, \langle w_j+\theta, x_j\rangle\right) - \frac{1}{n_j}\sum_{i=1}^{n_j}\ell\left(y_{j,i}, \langle w_j+\theta, x_{j,i}\rangle\right)\\
&\leq \frac{2\sigma^2 r^2 \max\{n_t : t \neq j\}}{n_j c} + \left(\frac{4\sigma^2 r^2 \max\{n_t : t \neq j\}}{c} + M\right)\sqrt{\frac{\log 1/\delta}{2n_j}}.
\end{aligned}$$

See the proof in Section 4.3.

Remark 6 *Using the Rademacher complexity method, we can prove that the algorithm for learning the shared parameter θ has a generalization bound of order $\mathcal{O}\left(\sqrt{1/\sum_{t=1}^{T} n_t}\right)$, which could be tighter than the order*[3] *$\mathcal{O}(1/n_j)$ presented in Proposition 2 when $n_1 = \ldots = n_T = n$ and $T > n$. However, such a generalization bound is of order $\mathcal{O}(\sqrt{1/n_j})$ with respect to the sample size n_j of the j-the task and decreases far more slowly than our bound presented in Proposition 2 when increasing the sample size n_j.*

Remark 7 *Bousquet and Elisseeff (Bousquet & Elisseeff, 2002) and Shalev-Shwartz (Shalev-Shwartz et al., 2010) have proven that the generalization bound for learning the predictor of a single task can be of order $\mathcal{O}(1/n)$, where n is the sample size of the task. Their results apply to the focused task in MTL. However, their bounds strictly rely on L_2 regularization on the predictor, or θ, while our bound does not and thus is more general.*

Remark 8 *The generalization bound shown in Proposition 2 is algorithm-dependent, because it has been derived by interpreting some of the tasks in the MTL problem (1) as regularizers, which greatly shrinks the search space of the parameters to be learned.*

Since there is an trade-off between w_j and θ, our results in Propositions 1 and 2 can be easily extended to learn the predictor of the focused task by simply setting $w_j = 0$ in the proof. Using the same proof method, we have the following theorem.

[3] It is accepted in the machine learning community that the convergence rate of a generalization bound is calculated according to the terms related to the hypothesis complexity, but not according to the terms involving the confidence interval parameter δ introduced by employing concentration ineqaulities. This is why we claim that the convergence rate in Proposition 2 is of order $\mathcal{O}(1/n_j)$.

Theorem 2 (Main result one) *Under the conditions of Propositions 1 and 2, for any $z_j = (x_j, y_j)$ distributed from the j-th task, any predictor h_{j,S_j} and h_{j,S_j^i} learned by algorithm (1) for the j-th task, the following inequality holds:*

$$\begin{aligned}|\ell(y_j, h_{j,S_j^i}(x_j)) - \ell(y_j, h_{j,S_j}(x_j))| &\leq \max\nolimits_{z_j=(x_j,y_j)\in\mathcal{Z}_j} \sigma \left|\left\langle h_{j,S_j^i} - h_{j,S_j}, x_j\right\rangle\right| \\ &\leq \tfrac{\sigma r \max\{n_t : t\neq j\}}{2c}\left(\sqrt{\left(\tfrac{2\sigma r}{n_j}\right)^2 + \tfrac{4c\mathcal{O}(\epsilon)}{n_j \max\{n_t:t\neq j\}}} + \tfrac{2\sigma r}{n_j}\right).\end{aligned}$$

Let $\epsilon = 0$, and we have

$$\begin{aligned}&|\ell(y_j, h_{j,S_j}(x_j)) - \ell(y_j, h_{j,S_j^i}(x_j))| \\ &\leq \max_{(x_j,y_j)\in\mathcal{Z}_j} \sigma \left|\left\langle h_{j,S_j^i} - h_{j,S_j}, x_j\right\rangle\right| \\ &\leq \frac{2\sigma^2 r^2 \max\{n_t : t \neq j\}}{n_j c}.\end{aligned}$$

Moreover, for any predictor h_j learned for the j-th task and any $\delta > 0$, with probability at least $1-\delta$, the following inequality holds:

$$\begin{aligned}&E_{z_j=(x_j,y_j)\sim\mu_j}\ell(y_j, h_j(x_j)) - \frac{1}{n_j}\sum_{i=1}^{n_j}\ell(y_{j,i}, h_j(x_{j,i})) \\ &\leq \frac{2\sigma^2 r^2 \max\{n_t : t\neq j\}}{n_j c} + \left(\frac{4\sigma^2 r^2 \max\{n_t : t \neq j\}}{c} + M\right)\sqrt{\frac{\log 1/\delta}{2n_j}}.\end{aligned}$$

Remark 9 *When focused on a specific task, we are interested in the problem of how increasing its sample size affects its learning performance. Since the existing generalization bound of the empirical risk minimization (ERM) algorithm for single-task learning has the fastest convergence rate of order $\mathcal{O}(\sqrt{1/n})$, Theorem 2 shows the benefit of MTL over the traditional single-task learning by illustrating that the generalization bound for learning the focused task in MTL is of order $\mathcal{O}(1/n)$ with respect to its sample size n.*

Remark 10 *The obtained generalization bounds in Proposition 2 and Theorem 2 have fast convergence rates with respect to the sample size of the focused task. To derive the fast convergence rates, we have concentrated on feature structures and ignored the labeling information of the unfocused tasks, so no benefit is shown for any increase in the other training sample sizes $n_t, t \neq j$. However, in practice, large $n_t, t \neq j$, will provide more labeling information that is useful to the focused task. We refer the readers to the related work for more details.*

When learning the shared parameter θ, we have shown that the algorithm can be uniformly stable with respect to the domain of one particular task in MTL and that the convergence rate with respect to the corresponding training sample size is fast. We now show that the shared parameter θ can be estimated with a fast convergence rate with respect to the number of the multiple tasks.

Theorem 3 (Main result two) *Let $\mathbf{S} = \{S_1, \ldots, S_T\}$ denote the training sample set for MTL. Let Assumption 1 hold that for any task $t, t = 1, \ldots, T$, there exists $B_t = \{b_{t,1}, \ldots, b_{t,N_t}\} \subset \{x_{1,1}, \ldots, x_{T,n_T}\} - \{x_{t,1}, \ldots, x_{t,n_t}\}$ such that for any feature vector x distributed from any task, x can be reconstructed by B with a small reconstruction error, i.e., $x = \sum_{j=1}^{N_t} \alpha_j b_{t,j} + \eta$, $\|\alpha\| \leq r$, and η is a small error that satisfies $\|\eta\| \leq \epsilon$. Let also assume that $\eta = 0$. Let the loss function ℓ be c-strongly convex and σ-admissible, and let $n_1 = n_2 = \ldots = n_T = n$, where $n \geq 2$. Then, the algorithm for learning θ by multi-task learning is uniformly stable with respect to the domains of all the multiple tasks. Thus, we have*

$$\max_{z=(x,y)\in\{\mathcal{Z}_1\cup\ldots\cup\mathcal{Z}_T\}} |\langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, x\rangle| \leq \frac{2\sigma r^2}{cT},$$

where $\theta_{\mathbf{S}}$ denotes the θ that is learned using sample $\mathbf{S}$*,* $\mathbf{S}^i$ *represents the training sample* $\mathbf{S}$ *with the i-th training example $z_i, i = 1, \ldots, Tn$, replaced by an independent and identically distributed one z_i', and $\{\mathcal{Z}_1 \cup ... \cup \mathcal{Z}_T\}$ denotes the joint domains of all the multiple tasks.*

See the proof in Section 4.4.

Remark 11 *Theorem 3 is based on the idea that multiple tasks can provide an inductive bias that does not vanish when T goes to infinity. The inductive bias makes the learning algorithm uniformly stable with respect to the domains of all the multiple tasks.*

When focusing on the average performance over all of the multiple tasks, we show that the generalization bound decreases fast as the number of tasks increases.

Proposition 3 *When focusing on all of the multiple tasks, let Assumption 1 hold and let η therein be 0. Let the loss function ℓ be c-strongly convex and σ-admissible, let $\mu_1, \ldots, \mu_T$ be probability measures on the domains of the multiple tasks and let $n_1 = n_2 = \ldots = n_T = n$ be fixed. Then, for any θ that is learned using (1) for MTL, and any $\delta > 0$, given $w_t, t = 1, \ldots, T$, with probability at least $1 - 2\delta$, the following inequality holds:*

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T} E_{z_t=(x_t,y_t)\sim\mu_t}\ell\left(y_t, \langle w_t + \theta, x_t\rangle\right) \\
&\leq \frac{1}{T}\sum_{t=1}^{T}\frac{1}{n}\sum_{j=1}^{n}\ell\left(y_{t,j}, \langle w_t + \theta, x_{t,j}\rangle\right) + \frac{2\sigma^2 r^2}{cT}\sqrt{2\ln(2/\delta)} + M\sqrt{\frac{2\ln\left(1/\delta\right)}{T}}.
\end{aligned}$$

See the proof in Section 4.4.

Remark 12 *The predictor $h_t = w_t + \theta, t = 1, \ldots, T$, has a two-part structure: one part corresponds to the specific task in the set of multiple tasks, and the other part is for the shared parameter θ. When focused on the multiple tasks, our statements (i.e., Propositions 2 and 3) concern the shared parameter θ but not the parameters that are specific for different tasks. This strategy is natural because MTL has not shown any benefit for learning knowledge that is specific to one particular task and is irrelevant to the other tasks. The fast convergence rate for learning the shared parameter θ is a strong theoretical justification for the good performance of MTL, even though the generalization ability of the overall predictor $h_t, t = 1, \ldots, T$, has not been exploited explicitly.*

Remark 13 *When focusing on the performance of one particular task, we have derived an algorithm-dependent generalization bound with a fast covergence rate with respect to the predictor $h_j = w_j + \theta$. When focusing on the average performance over multiple tasks, we have only dervied an algorithm-dependent generalization bound with a fast covergece rate with respect to the common parameter θ. This is because it is unreasonable to claim that the MTL algorithm for learning $w_j, j = 1, \ldots, T$ is uniformly stable with respect to the domains of all the multiple tasks.*

Remark 14 *Increasing the number of tasks can be helpful for multi-task learning. To obtain an intuitive understanding, consider an extreme case in which all of the tasks are related and each task has an independently drawn sample of size one. Increasing the number of related tasks is equal to increasing the number of the independently drawn examples and will definitely help learn the related information. Proposition 3 provides a theoretical guarantee for this intuition with a fast convergence rate of order $\mathcal{O}(1/T)$.*

Remark 15 *To model the relationship between tasks, we have used a parameter θ that is shared by all of the tasks, as shown in (1). However, based on the proof methods, our analyses could be easily extended to a more general setting where only a few tasks share some common parameters. For example, Maurer et al. (Maurer et al., 2013) proposed a coding schemes model for MTL, where the task parameters are linear combinations of the atoms in a dictionary (also called an implementation in coding schemes). In this model, multi-task learning benefits when at least one atom is shared by some of the task parameters. Our analyses show that the shared atom can be efficiently learned with a fast convergence rate with respect to the training sample size or the number of tasks that share the atom.*

Different from previous results showing that most learning algorithms exhibit a uniform stability relying on L_2 regularization, we have illustrated the uniformly stable property for MTL algorithms by exploiting feature structures. In our analyses, carefully collected tasks could provide biased regularization. Consequently, our approach is easy to extend to many existing learning algorithms. We present the extensions to learning to learn (LTL), as an example, in the supplementary material.

4 Proof

In this section, we present detailed proofs of the assertions that were made in previous sections. We begin by introducing the concentration inequalities, which play an important role in proving generalization bounds.

4.1 Used Tools

McDiarmid's inequality (McDiarmid, 1998), which is known as the bounded difference inequality, is widely used for deriving generalization bounds.

Theorem 4 (McDiarmid's inequality) *Let $X = (x_1, \ldots, x_n)$ be a sample set of independent random variables and let X^i be a new sample set with the i-th example x_i in X replaced by a new one x_i'. If there exists $c_1, \ldots, c_n > 0$ such that $f : \mathcal{X}^n \to \mathbb{R}$ satisfies the following conditions:*

$$|f(X) - f(X^i)| \leq c_i,$$

for all $i \in \{1, \ldots, n\}$ and any points $x_1, \ldots, x_n, x_i' \in \mathcal{X}$. Then for any $X \in \mathcal{X}^n$ and $\epsilon > 0$, the following inequality holds:

$$Pr\{Ef(X) - f(X) \geq \epsilon\} \leq \exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^n c_i^2}\right),$$

where $Pr\{A\}$ denotes the probability that event A occurs.

Note that McDiarmid's inequality holds for independent random variables, which are not required to be identically distributed. Combined with McDiarmid's inequality, the uniform stability of learning algorithms is used to develop generalization bounds with fast convergence rates. The generalization bound derived using uniform stability is as follows (Mohri et al., 2012):

Theorem 5 *Assume that the loss function ℓ is bounded by M. Let $\mathcal{A}$ be a β-stable learning algorithm, S be a sample set with n i.i.d. random variables, and h_S be the hypothesized function that is output by the learning algorithm $\mathcal{A}$ when the input training* *sample is S. Then, for any $\delta > 0$, with probability at least $1-\delta$, the following holds:*

$$\begin{aligned}
&E_{z=(x,y)}\ell(y, h_S(x)) - \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h_S(x_i)) \\
&\leq \beta + (2n\beta + M)\sqrt{\frac{\log 1/\delta}{2n}}.
\end{aligned}$$

Proof sketch of Theorem 5: Let

$$\Phi(S) = E_{(x,y)}\ell(y, h_S(x)) - \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h_S(x_i)).$$

It can be proven that $|\Phi(S) - \Phi(S^i)| \leq 2\beta + M/n$ and that $E_S\Phi(S) \leq \beta$. Then, Theorem 5 can be obtained using McDiarmid's inequality.

To upper bound the uniform stability, we need to introduce the notion of Bregman divergence (Mohri et al., 2012).

Definition 5 (Bregman divergence) *Let $F : \mathcal{H} \to \mathbb{R}$ be a convex function. For all $f, g \in \mathcal{H}$, we have*

$$B_F(f\|g) = F(f) - F(g) - \langle f - g, \delta F(g) \rangle,$$

where $\delta F(g)$ denotes the subgradient of F at g.

Detailed discussions about Bregman divergence can be found in (Mohri et al., 2012).

Lemma 1 *Bregman divergence is additive and non-negative. If $F = F_1 + F_2$ and both F_1 and F_2 are convex, for any $f, g \in \mathcal{H}$, we have*

$$B_F(f\|g) = B_{F_1}(f\|g) + B_{F_2}(f\|g)$$

and

$$B_F(f\|g) \geq 0.$$

To prove that the learning algorithm of MTL has a fast generalization rate of order $\mathcal{O}(1/T)$, we will use Hoeffding's inequality (Hoeffding, 1963).

Theorem 6 (Hoeffding's inequality) *Let $x_1, \dots, x_n$ be independent random variables with x_i taking values in $[a_i, b_i]$ for all $i \in \{1, \dots, n\}$. Then for any $\epsilon > 0$, the following inequality holds:*

$$\mathit{Pr}\left\{ E\sum_{i=1}^{n} x_i - \sum_{i=1}^{n} x_i \geq \epsilon \right\} \leq \exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^{n}(b_i - a_i)^2} \right).$$

4.2 Proof of Theorem 1

We show that the Rademacher complexity and algorithmic stability are closely related by proving that the sample average stability is upper bounded by the Rademacher complexity.

Proof of Theorem 1. We have

$$
\begin{aligned}
&\left|\frac{1}{n}\sum_{i=1}^{n} E_{S\sim D^n,S'\sim D^n}[\ell(y_i', h_{S^i}(x_i')) - \ell(y_i', h_S(x_i'))]\right| \\
&= \left|\frac{1}{n}\sum_{i=1}^{n} E_{S\sim D^n,\{z_1',\ldots,z_n'\}\sim D^n}\ell(y_i', h_{S^i}(x_i'))\right. \\
&\quad \left. -\frac{1}{n}\sum_{i=1}^{n} E_{S\sim D^n,\{z_1',\ldots,z_n'\}\sim D^n}\ell(y_i', h_S(x_i'))\right| \\
&= \left|\frac{1}{n}\sum_{i=1}^{n} E_{S\sim D^n,z_i'\sim D}\ell(y_i', h_{S^i}(x_i'))\right. \\
&\quad \left. -\frac{1}{n}\sum_{i=1}^{n} E_{S\sim D^n,z_i'\sim D}\ell(y_i', h_S(x_i'))\right| \\
&= \left|\frac{1}{n}\sum_{i=1}^{n} E_{S\sim D^n}\ell(y_i, h_S(x_i)) - E_{S\sim D^n,z\sim D}\ell(y, h_S(x))\right| \\
&= \left|E_{S\sim D^n}\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, h_S(x_i)) - E_{S\sim D^n,z\sim D}\ell(y, h_S(x))\right| \\
&= \left|E_{S\sim D^n}\left(\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, h_S(x_i)) - E_{z\sim D}\ell(y, h_S(x))\right)\right| \\
&\le E_S\left|\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, h_S(x_i)) - E_{z\sim D}\ell(y, h_S(x))\right| \\
&\le E_S\sup_{h\in H}\left|\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, h(x_i)) - E_{z\sim D}\ell(y, h(x))\right| \\
&\le E_S\sup_{h\in H}\left|\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, h(x_i)) - E_{S'\sim D^n}\frac{1}{n}\sum_{i=1}^{n}\ell(y_i', h(x_i'))\right| \\
&\le E_S\sup_{h\in H}E_{S'}\left|\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, h(x_i)) - \frac{1}{n}\sum_{i=1}^{n}\ell(y_i', h(x_i'))\right| \\
&\le E_{S,S'}\sup_{h\in H}\left|\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, h(x_i)) - \frac{1}{n}\sum_{i=1}^{n}\ell(y_i', h(x_i'))\right| \\
&= E_{S,S',\sigma_1,\ldots,\sigma_n}\sup_{h\in H}\left|\frac{1}{n}\sum_{i=1}^{n}\sigma_i[\ell(y_i, h(x_i)) - \ell(y_i', h(x_i'))]\right| \\
&\le 2E_{S,\sigma_1,\ldots,\sigma_n}\sup_{h\in H}\left|\frac{1}{n}\sum_{i=1}^{n}\sigma_i\ell(y_i, h(x_i))\right| \\
&= 2\mathfrak{R}(\ell\circ H),
\end{aligned}
$$

where $S' = \{z'_1, \ldots, z'_n\} \sim D^n$. This completes the proof of Theorem 1. $\square$

4.3 Proofs of Propositions 1 and 2

We first prove that when the loss function ℓ is c-strongly convex, the corresponding ERM algorithm (1) for MTL will be uniformly stable with respect to the domain of a particular task if a mild assumption on the data structure holds. Note that in the proof, we will interpret the functions of some of the tasks as regularizers.

Proof Proposition 1. We prove that the algorithm for learning θ is uniformly stable with respect to the domain of the first task. For the other tasks, the same proof strategy applies.

Let $S_1 = (z_{1,1}, \ldots, z_{1,n_1})$ be the i.i.d. training sample for the first task. For any given w_1 and any $z_1 = (x_1, y_1)$ distributed from the first task, the following inequalities hold:

$$\begin{aligned}
&\left|\ell(y_1, \langle w_1 + \theta_{S_1,\ldots,S_T}, x_1\rangle) - \ell\left(y_1, \left\langle w_1 + \theta_{S_1^i,S_2,\ldots,S_T}, x_1\right\rangle\right)\right| \\
&\leq \max_{z_1=(x_1,y_1)\in\mathcal{Z}_1} \left|\ell(y_1, \langle w_1 + \theta_{S_1,\ldots,S_T}, x_1\rangle) - \ell\left(y_1, \left\langle w_1 + \theta_{S_1^i,S_2,\ldots,S_T}, x_1\right\rangle\right)\right| \\
&\leq \max_{(x_1,y_1)\in\mathcal{Z}_1} \sigma \left|\left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, x_1\right\rangle\right|, \qquad (2)
\end{aligned}$$

where $\theta_{S_1,\ldots,S_T}$ represents the parameter that corresponds to the related information among the tasks and is learned when the training samples are $S_1, \ldots, S_T$.

We will use the notion of Bregman divergence to derive an upper bound for

$$\max_{z_1=(x_1,y_1)\in\mathcal{Z}_1} \left|\langle \theta_{S_1,\ldots,S_T} \; - \theta_{S_1^i,S_2,\ldots,S_T}, x_1\right\rangle\Big| .$$

Let $B = (b_1, \ldots, b_N) \in \{x_{1,1}, \ldots, x_{T,n_T}\} - \{x_{1,1}, \ldots, x_{1,n_1}\}$ be the representative observations defined in Assumption 1 such that for any feature vector x distributed from the first task, x can be reconstructed by B with a small reconstruction error. Let

$$N(\theta) = \frac{1}{T}\sum_{j=1}^{N}\frac{1}{n_{t_j}}\ell\left(y_{b_j}, \langle w_{t_j} + \theta, b_j\rangle\right),$$

where n_{t_j} is the size of the training sample S_{t_j}, $t_j \in \{1, \dots, T\}$, to which the example (b_j, y_{b_j}) belongs. Let

$$F(\theta) = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{n_t}\sum_{i=1}^{n_t}\ell\left(y_{t,i}, \langle w_t + \theta, x_{t,i}\rangle\right). \tag{3}$$

Define $V(\theta)$ by

$$V(\theta) = F(\theta) - N(\theta).$$

Then, $V(\theta)$ is non-negative and convex, because $\{(b_1, y_{b_1}), \dots, (b_N, y_{b_N})\} \subseteq \{z_{1,1}, \dots, z_{T,n_T}\}$ and ℓ is convex.

Using the non-negative and additive properties of Bregman divergence, for any z_i' distributed from the domain of the first task, the following inequality holds:

$$\begin{aligned}&B_{F_{S_1,\dots,S_T}}(\theta_{S_1^i,S_2,\dots,S_T}\|\theta_{S_1,\dots,S_T}) + B_{F_{S_1^i,S_2,\dots,S_T}}(\theta_{S_1,\dots,S_T}\|\theta_{S_1^i,S_2,\dots,S_T})\\ &\geq B_N(\theta_{S_1^i,S_2,\dots,S_T}\|\theta_{S_1,\dots,S_T}) + B_N(\theta_{S_1,\dots,S_T}\|\theta_{S_1^i,S_2,\dots,S_T}),\end{aligned} \tag{4}$$

where $F_{S_1,\dots,S_T}$ denotes $F(\theta)$ in (3) computed using the training samples $S_1, \dots, S_T$.

To lower bound the right-hand side of inequality (4), we consider two different forms of loss function: (i) $\ell(y, h(x)) = \ell(y - h(x))$ and (ii) $\ell(y, h(x)) = \ell(yh(x))$, separately.

When the loss function is of form (i), the following inequalities hold:

$$
\begin{aligned}
& B_N(\theta_{S_1^i,S_2,\ldots,S_T} \| \theta_{S_1,\ldots,S_T}) + B_N(\theta_{S_1,\ldots,S_T} \| \theta_{S_1^i,S_2,\ldots,S_T}) \\
&= -\left\langle \theta_{S_1^i,S_2,\ldots,S_T} - \theta_{S_1,\ldots,S_T}, \frac{1}{T}\sum_{j=1}^{N} \frac{1}{n_{t_j}} \delta\ell\left(y_{b_j} - \left\langle w_{t_j} + \theta_{S_1,\ldots,S_T}, b_j\right\rangle\right) b_j \right\rangle \\
&\quad -\left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, \frac{1}{T}\sum_{j=1}^{N} \frac{1}{n_{t_j}} \delta\ell\left(y_{b_j} - \left\langle w_{t_j} + \theta_{S_1^i,S_2,\ldots,S_T}, b_j\right\rangle\right) b_j \right\rangle \\
&= \frac{1}{T}\sum_{j=1}^{N} \frac{1}{n_{t_j}} \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, \delta\ell\left(y_{b_j} - \left\langle w_{t_j} + \theta_{S_1,\ldots,S_T}, b_j\right\rangle\right) b_j \right\rangle \\
&\quad -\frac{1}{T}\sum_{j=1}^{N} \frac{1}{n_{t_j}} \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, \delta\ell\left(y_{b_j} - \left\langle w_{t_j} + \theta_{S_1^i,S_2,\ldots,S_T}, b_j\right\rangle\right) b_j \right\rangle \\
&= \frac{1}{T}\sum_{j=1}^{N} \frac{1}{n_{t_j}} \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, \delta\ell\left(y_{b_j}\right.\right. \\
&\quad \left.\left. - \left\langle w_{t_j} + \theta_{S_1,\ldots,S_T}, b_j\right\rangle\right) b_j - \delta\ell\left(y_{b_j} - \left\langle w_{t_j} + \theta_{S_1^i,S_2,\ldots,S_T}, b_j\right\rangle\right) b_j \right\rangle \\
&\geq \frac{1}{T}\sum_{j=1}^{N} \frac{c}{n_{t_j}} \left\langle \theta_{S_1^i,S_2,\ldots,S_T} - \theta_{S_1,\ldots,S_T}, b_j \right\rangle^2 \\
&\geq \frac{c}{\max\{n_t : t \neq 1\} T} \sum_{j=1}^{N} \left\langle \theta_{S_1^i,S_2,\ldots,S_T} - \theta_{S_1,\ldots,S_T}, b_j \right\rangle^2,
\end{aligned} \tag{5}
$$

where the first inequality holds because the loss function ℓ is c-strongly convex.

When the loss function is of form (ii), similar to (5), the following inequalities hold:

$$
\begin{aligned}
&B_N(\theta_{S_1^i,S_2,\dots,S_T} \| \theta_{S_1,\dots,S_T}) + B_N(\theta_{S_1,\dots,S_T} \| \theta_{S_1^i,S_2,\dots,S_T}) \\
&= -\left\langle \theta_{S_1^i,S_2,\dots,S_T} - \theta_{S_1,\dots,S_T}, \frac{1}{T}\sum_{j=1}^{N} \frac{1}{n_{t_j}} \delta\ell\left(y_{b_j}\left\langle w_{t_j} + \theta_{S_1,\dots,S_T}, b_j\right\rangle\right) b_j \right\rangle y_{b_j} \\
&\quad -\left\langle \theta_{S_1,\dots,S_T} - \theta_{S_1^i,S_2,\dots,S_T}, \frac{1}{T}\sum_{j=1}^{N} \frac{1}{n_{t_j}} \delta\ell\left(y_{b_j}\left\langle w_{t_j} + \theta_{S_1^i,S_2,\dots,S_T}, b_j\right\rangle\right) b_j \right\rangle y_{b_j} \\
&= \frac{1}{T}\sum_{j=1}^{N} \frac{y_{b_j}}{n_{t_j}} \left\langle \theta_{S_1,\dots,S_T} - \theta_{S_1^i,S_2,\dots,S_T}, \delta\ell\left(y_{b_j}\left\langle w_{t_j} + \theta_{S_1,\dots,S_T}, b_j\right\rangle\right) b_j \right\rangle \\
&\quad -\frac{1}{T}\sum_{j=1}^{N} \frac{y_{b_j}}{n_{t_j}} \left\langle \theta_{S_1,\dots,S_T} - \theta_{S_1^i,S_2,\dots,S_T}, \delta\ell\left(y_{b_j}\left\langle w_{t_j} + \theta_{S_1^i,S_2\dots,S_T}, b_j\right\rangle\right) b_j \right\rangle \\
&= \frac{1}{T}\sum_{j=1}^{N} \frac{1}{n_{t_j}} \left\langle \theta_{S_1,\dots,S_T} - \theta_{S_1^i,S_2,\dots,S_T}, \delta\ell\left(y_{b_j}\left\langle w_{t_j} + \theta_{S_1,\dots,S_T}, b_j\right\rangle\right) b_j y_{b_j} \right. \\
&\quad \left. -\delta\ell\left(y_{b_j}\left\langle w_{t_j} + \theta_{S_1^i,S_2,\dots,S_T}, b_j\right\rangle\right) b_j y_{b_j} \right\rangle \\
&\geq \frac{1}{T}\sum_{j=1}^{N} \frac{c}{n_{t_j}} \left\langle \theta_{S_1^i,S_2,\dots,S_T} - \theta_{S_1,\dots,S_T}, b_j y_{b_j} \right\rangle^2 \\
&\geq \frac{c}{\max\{n_t : t \neq 1\}T} \sum_{j=1}^{N} \left\langle \theta_{S_1^i,S_2,\dots,S_T} - \theta_{S_1,\dots,S_T}, b_j \right\rangle^2 y_{b_j}^2 \\
&= \frac{c}{\max\{n_t : t \neq 1\}T} \sum_{j=1}^{N} \left\langle \theta_{S_1^i,S_2,\dots,S_T} - \theta_{S_1,\dots,S_T}, b_j \right\rangle^2. \qquad (6)
\end{aligned}
$$

Note that for any $z'_{1,i}, i = 1, \dots, n_1$, distributed from the first task, using (4), (5) and (6), we have the following inequalities:

$$
\begin{aligned}
&\frac{c}{\max\{n_t : t \neq 1\}T} \sum_{j=1}^{N} \left\langle \theta_{S_1^i,S_2,\dots,S_T} - \theta_{S_1,\dots,S_T}, b_j \right\rangle^2 \\
&\leq B_{F_{S_1,\dots,S_T}}(\theta_{S_1^i,S_2,\dots,S_T} \| \theta_{S_1,\dots,S_T}) + B_{F_{S_1^i,S_2,\dots,S_T}}(\theta_{S_1,\dots,S_T} \| \theta_{S_1^i,S_2,\dots,S_T}) \\
&(\because \delta F_{S_1,\dots,S_T}(\theta_{S_1,\dots,S_T}) = 0 \text{ and } \delta F_{S_1^i,S_2,\dots,S_T}(\theta_{S_1^i,S_2,\dots,S_T}) = 0) \\
&= \frac{1}{n_1 T}\left\{ \ell\left(y_{1,i}, \left\langle w_1 + \theta_{S_1^i,S_2,\dots,S_T}, x_{1,i}\right\rangle\right) - \ell\left(y_{1,i}, \langle w_1 + \theta_{S_1,\dots,S_T}, x_{1,i}\rangle\right) \right. \\
&\quad \left. +\ell\left(y'_{1,i}, \langle w_1 + \theta_{S_1,\dots,S_T}, x'_{1,i}\rangle\right) - \ell\left(y'_{1,i}, \left\langle w_1 + \theta_{S_1^i,S_2,\dots,S_T}, x'_{1,i}\right\rangle\right) \right\} \\
&\leq \frac{\sigma}{n_1 T}\left(\left|\left\langle \theta_{S_1^i,S_2,\dots,S_T} - \theta_{S_1,\dots,S_T}, x_{1,i}\right\rangle\right| + \left|\left\langle \theta_{S_1,\dots,S_T} - \theta_{S_1^i,S_2,\dots,S_T}, {x'}_{1,i}\right\rangle\right| \right). (7)
\end{aligned}
$$

By Assumption 1, for any x_1, we have $x_1 = \sum_{j=1}^{N} \alpha_j b_j + \eta$, where $\alpha_j \in \mathbb{R}, j = 1, \ldots, N, \|\alpha\| \leq r$ and $\|\eta\| \leq \epsilon$. Thus,

$$
\begin{aligned}
&\left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, x_1 \right\rangle \\
&= \sum_{j=1}^{N} \alpha_j \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, b_j \right\rangle \\
&\quad + \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, \eta \right\rangle \\
&\text{(Using Cauchy-Schwarz inequality)} \\
&\leq \sqrt{\sum_{j=1}^{N} \alpha_j^2} \sqrt{\sum_{j=1}^{N} \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, b_j \right\rangle^2} + \mathcal{O}(\epsilon) \\
&\leq r \sqrt{\sum_{j=1}^{N} \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, b_j \right\rangle^2} + \mathcal{O}(\epsilon). \qquad (8)
\end{aligned}
$$

Combining (7) and (8), we have

$$
\begin{aligned}
&\frac{c}{\max\{n_t : t \neq 1\} T} \sum_{j=1}^{N} \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, b_j \right\rangle^2 \\
&\leq \frac{2\sigma r}{n_1 T} \sqrt{\sum_{j=1}^{N} \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, b_j \right\rangle^2} + \frac{\mathcal{O}(\epsilon)}{n_1 T}.
\end{aligned}
$$

This gives

$$
\begin{aligned}
&\sqrt{\sum_{j=1}^{N} \left\langle \theta_{S_1,\ldots,S_T} - \theta_{S_1^i,S_2,\ldots,S_T}, b_j \right\rangle^2} \\
&\leq \tfrac{\max\{n_t : t \neq 1\}}{2c} \left(\sqrt{\left(\tfrac{2\sigma r}{n_1}\right)^2 + \tfrac{4c\mathcal{O}(\epsilon)}{n_1 \max\{n_t : t \neq 1\}}} + \tfrac{2\sigma r}{n_1} \right).
\end{aligned}
$$

We are now ready to upper bound $\max_{z_1=(x_1,y_1)\in\mathcal{Z}_1}\left|\left\langle \theta_{S_1,\ldots,S_T}-\theta_{S_1^i,S_2,\ldots,S_T},x_1\right\rangle\right|$:

$$\begin{aligned}
&\max\nolimits_{(x_1,y_1)\in\mathcal{Z}_1}\left|\left\langle \theta_{S_1,\ldots,S_T}-\theta_{S_1^i,S_2,\ldots,S_T},x_1\right\rangle\right|\\
&=\max\nolimits_{\alpha}\left|\sum\nolimits_{j=1}^{N}\alpha_j\left\langle \theta_{S_1,\ldots,S_T}-\theta_{S_1^i,S_2,\ldots,S_T}b_j\right\rangle\right|\\
&\quad\text{(Using Cauchy-Schwarz inequality)}\\
&\leq r\sqrt{\sum\nolimits_{j=1}^{N}\left\langle \theta_{S_1,\ldots,S_T}-\theta_{S_1^i,S_2,\ldots,S_T},b_j\right\rangle^2}\\
&\leq \tfrac{r\max\{n_t:t\neq 1\}}{2c}\left(\sqrt{\left(\tfrac{2\sigma r}{n_1}\right)^2+\tfrac{4c\mathcal{O}(\epsilon)}{n_1\max\{n_t:t\neq 1\}}}+\tfrac{2\sigma r}{n_1}\right).
\end{aligned}$$

Thus, the inequalities hold:

$$\begin{aligned}
&\max_{z_1=(x_1,y_1)\in\mathcal{Z}}\left|\ell\left(y_1,\langle w_1+\theta_{S_1,\ldots,S_T},x_1\rangle\right)-\ell\left(y_1,\left\langle w_1+\theta_{S_1^i,S_2,\ldots,S_T},x_1\right\rangle\right)\right|\\
&\leq \max_{z_1=(x_1,y_1)\in\mathcal{Z}}\sigma\left|\left\langle \theta_{S_1,\ldots,S_T}-\theta_{S_1^i,S_2,\ldots,S_T},x_1\right\rangle\right|\\
&\leq \frac{\sigma r\max\{n_t:t\neq 1\}}{2c}\left(\sqrt{\left(\frac{2\sigma r}{n_1}\right)^2+\frac{4c\mathcal{O}(\epsilon)}{n_1\max\{n_t:t\neq 1\}}}+\frac{2\sigma r}{n_1}\right).
\end{aligned}$$

This statement concludes the proof of Proposition 1. □

Remark 16 *Comparing the proof method of Proposition 11.1 in (Mohri et al., 2012) with our above proof method of Proposition 1, the term $N(\theta)$ in the above proof intrinsically functions as a regularizer for optimizing $F(\theta)$. When we focus on the first task, $F(\theta)$ in the above proof can be rewritten as*

$$F(\theta)=\frac{1}{n_1T}\sum_{i=1}^{n_1}\ell\left(y_{1,i},\langle w_1+\theta,x_{1,i}\rangle\right)+R_1(\theta)+R_2(\theta),$$

where

$$R_1(\theta)=N(\theta)=\frac{1}{T}\sum_{j=1}^{N}\frac{1}{n_{t_j}}\ell\left(y_{b_j},\langle w_{t_j}+\theta,b_j\rangle\right),$$

$$R_2=V(\theta)-\frac{1}{n_1T}\sum_{i=1}^{n_1}\ell\left(y_{1,i},\langle w_1+\theta,x_{1,i}\rangle\right)$$

and both of them are positive and convex. Thus, we can interpret the function of the unfocused tasks as regularizers. In the proof of Proposition 1, we used $R_1(\theta) = N(\theta)$ as regularization to obtain the upper bound of uniform stability. If we replace $R_1(\theta)$ by $\frac{1}{T}\sum_{j=1}^{N}\frac{1}{n_{t_j}}\langle\theta, b_j\rangle^2$, the proof procedure and result of Proposition 1 remain the same, which means that we have not used the labeling information of the unfocused tasks in the proof (as discussed in Remark 10) and that Proposition 1 relies on the simple form regularizer $\frac{1}{T}\sum_{j=1}^{N}\frac{1}{n_{t_j}}\langle\theta, b_j\rangle^2$. Note that we have written $\frac{1}{T}\sum_{j=1}^{N}\frac{1}{n_{t_j}}\langle\theta, b_j\rangle^2 = \lambda\|\Gamma\theta\|_2^2$ in Remark 4.

Proof of Proposition 2. According to Proposition 1, we have proven that the algorithm for learning θ is uniformly stable with respect to the domain of the first task and that

$$\beta \leq \frac{2\sigma^2 r^2 \max\{n_t : t \neq j\}}{n_j c}.$$

Thus, Proposition 2 is proven by combining Proposition 1 and Theorem 5. □

4.4 Proofs of Theorem 3 and Proposition 3

The proof method of Theorem 3 is similar to that of Proposition 1. The key idea is that every training sample $S_t, t = 1, \ldots, T$, independently contributes to an inductive bias.

Proof of Theorem 3. Similar to (2), for any $t \in \{1, \ldots, T\}$ and any $z = (x, y)$ distributed from any of the multiple tasks, we have

$$\begin{aligned}&|\ell\left(y, \langle w_t, \theta_{\mathbf{S}}, x\rangle\right) - \ell\left(y, \langle w_t + \theta_{\mathbf{S}^i}, x\rangle\right)| \\ &\leq \max_{z=(x,y)\in\{\mathcal{Z}_1\cup\ldots\cup\mathcal{Z}_T\}} \sigma\left|\langle\theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, x\rangle\right|,\end{aligned}$$

where $\theta_{\mathbf{S}}$ represents the parameter that is corresponding to the related information among tasks and is learned for MTL using the training sample set $\mathbf{S} = \{S_1, \ldots, S_T\}$, and $\mathbf{S}^i$ represents the training sample $\mathbf{S}$ with the i-th training example $z_i, i = 1, \ldots, Tn$, replaced by an independent and identically distributed one z_i'.

Let $B_t = (b_{t,1}, \ldots, b_{t,N_t}) \in \{x_{t,1}, \ldots, x_{t,n}\}, t = 1, \ldots, T$, be the representative observations for the t-th task defined in Assumption 1 such that for any feature vector x distributed from any task, x can be reconstructed by B_t with a small reconstruction error. Let

$$N_T(\theta) = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{n}\sum_{j=1}^{N_t}\ell\left(y_{b_{t,j}}, \langle w_t + \theta, b_{t,j}\rangle\right),$$

where $y_{b_{t,j}}$ denotes the label for the observation $b_{t,j}$. Let

$$F(\theta) = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{n}\sum_{i=1}^{n}\ell\left(y_{t,i}, \langle w_t + \theta, x_{t,i}\rangle\right).$$

Then, we have

$$F(\theta) = N_T(\theta) + V_T(\theta),$$

where $V_T(\theta)$ is the sum of some prediction losses of examples and therefore is non-negative and convex. Using the non-negative and additive properties of Bregman divergence again, for any $z_{t,i}'$ distributed from any of the multiple tasks, we have

$$\begin{aligned} & B_{F_{\mathbf{S}}}(\theta_{\mathbf{S}^i}\|\theta_{\mathbf{S}}) + B_{F_{\mathbf{S}^i}}(\theta_{\mathbf{S}_1}\|\theta_{\mathbf{S}_1^i}) \\ & \geq B_{N_T}(\theta_{\mathbf{S}^i}\|\theta_{\mathbf{S}}) + B_{N_T}(\theta_{\mathbf{S}}\|\theta_{\mathbf{S}^i}). \end{aligned} \tag{9}$$

Similar to the proof in (5) and (6), we have the following inequalities

$$\begin{aligned}
&B_{N_T}(\theta_{\mathbf{S}^i}\|\theta_{\mathbf{S}}) + B_{N_T}(\theta_{\mathbf{S}}\|\theta_{\mathbf{S}^i}) \\
&= -\left\langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, \frac{1}{T}\sum_{t=1}^{T}\frac{1}{n}\sum_{j=1}^{N_t}\delta\ell\left(y_{b_{t,j}}, \langle w_t + \theta_{\mathbf{S}}, b_{t,j}\rangle\right) b_{t,j} \right\rangle \\
&\quad - \left\langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, \frac{1}{T}\sum_{t=1}^{T}\frac{1}{n}\sum_{j=1}^{N_t}\delta\ell\left(y_{b_{t,j}}, \langle w_t + \theta_{\mathbf{S}^i}, b_{t,j}\rangle\right) b_{t,j} \right\rangle \\
&= \frac{1}{T}\sum_{t=1}^{T}\frac{1}{n}\sum_{j=1}^{N_t}\left\langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, \delta\ell\left(y_{b_{t,j}}, \langle w_t + \theta_{\mathbf{S}}, b_{t,j}\rangle\right) b_{t,j} - \delta\ell\left(y_{b_{t,j}}, \langle w_t + \theta_{\mathbf{S}^i}, b_{t,j}\rangle\right) b_{t,j} \right\rangle \\
&\geq \frac{1}{T}\sum_{t=1}^{T}\frac{c}{n}\sum_{j=1}^{N_t}\left\langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, b_{t,j}\right\rangle^2. \qquad (10)
\end{aligned}$$

Note that for any $z'_{t,i}, t \in \{1, \ldots, T\}, i \in \{1, \ldots, n\}$, distributed form the t-th task, according to (9) and (10), we have

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T}\frac{c}{n}\sum_{j=1}^{N_t}\left\langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, b_{t,j}\right\rangle^2 \\
&\leq B_{F_{\mathbf{S}}}(\theta_{\mathbf{S}^i}\|\theta_{\mathbf{S}}) + B_{F_{\mathbf{S}^i}}(\theta_{\mathbf{S}_1}\|\theta_{\mathbf{S}_1^i}) \\
&\quad (\because \delta F_{\mathbf{S}}(\theta_{\mathbf{S}}) = 0 \text{ and } \delta F_{\mathbf{S}^i}(\theta_{\mathbf{S}^i}) = 0) \\
&= \frac{1}{nT}\left\{\ell\left(y_{t,i}, \langle w_t + \theta_{\mathbf{S}^i}, x_{t,i}\rangle\right) - \ell\left(y_{t,i}, \langle w_t + \theta_{\mathbf{S}}, x_{t,i}\rangle\right)\right. \\
&\quad \left. + \ell\left(y'_{t,i}, \left\langle w_t + \theta_{\mathbf{S}}, x'_{t,i}\right\rangle\right) - \ell\left(y'_{t,i}, \left\langle w_t + \theta_{\mathbf{S}^i}, x'_{t,i}\right\rangle\right)\right\} \\
&\leq \frac{\sigma}{nT}\left(\left|\left\langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, x_{t,i}\right\rangle\right| + \left|\left\langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, {x'}_{t,i}\right\rangle\right|\right).
\end{aligned}$$

Then, we have

$$\begin{aligned}
&c\sum_{t=1}^{T}\sum_{j=1}^{N_t}\left\langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, b_{t,j}\right\rangle^2 \\
&\leq \sigma\left(\left|\left\langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, x_{t,i}\right\rangle\right| + \left|\left\langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, {x'}_{t,i}\right\rangle\right|\right).
\end{aligned}$$

According to Assumption 1 and the assumption that $\eta = 0$, for any x distributed from any of the multiple tasks, we have $x = \sum_{j=1}^{N_t}\alpha_{t,j}b_{t,j}, t = 1, \ldots, T$, where $\alpha_{t,j} \in$

$\mathbb{R}, j = 1, \ldots, N_t, \|\alpha_t\| = \sqrt{\sum_{j=1}^{N_t} \alpha_{t,j}^2} \leq r$. Thus, it holds that

$$\begin{aligned}
\langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, x \rangle &= \frac{1}{T} \sum_{t=1}^{T} \langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, x \rangle \\
&= \frac{1}{T} \sum_{t=1}^{T} \sum_{j=1}^{N_t} \alpha_{t,j} \langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, b_{t,j} \rangle \\
&\text{(Using Cauchy-Schwarz inequality)} \\
&\leq \frac{1}{T} \sqrt{\sum_{t=1}^{T} \sum_{j=1}^{N_t} \alpha_{t,j}^2} \sqrt{\sum_{t=1}^{T} \sum_{j=1}^{N_t} \langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, b_{t,j} \rangle^2} \\
&\leq \frac{r}{\sqrt{T}} \sqrt{\sum_{t=1}^{T} \sum_{j=1}^{N_t} \langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, b_{t,j} \rangle^2}.
\end{aligned}$$

Therefore, we have

$$c \sum_{t=1}^{T} \sum_{j=1}^{N_t} \langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, b_{t,j} \rangle^2 \leq \frac{2\sigma r}{\sqrt{T}} \sqrt{\sum_{t=1}^{T} \sum_{j=1}^{N_t} \langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, b_{t,j} \rangle^2}.$$

Thus,

$$\sqrt{\sum_{t=1}^{T} \sum_{j=1}^{N_t} \langle \theta_{\mathbf{S}^i} - \theta_{\mathbf{S}}, b_{t,j} \rangle^2} \leq \frac{2\sigma r}{c\sqrt{T}}.$$

Now, we are ready to upper bound $\max_{z=(x,y)\in\{\mathcal{Z}_1 \cup \ldots \cup \mathcal{Z}_T\}} |\langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, x \rangle|$:

$$\begin{aligned}
&\max_{z=(x,y)\in\{\mathcal{Z}_1 \cup \ldots \cup \mathcal{Z}_T\}} |\langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, x \rangle| \\
&= \max_{z_1=(x_1,y_1)\in\mathcal{Z}_1, \ldots, z_T=(x_T,y_T)\in\mathcal{Z}_T} \left| \frac{1}{T} \sum_{t=1}^{T} \langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, x_t \rangle \right| \\
&= \max_{\alpha_1,\ldots,\alpha_T} \left| \frac{1}{T} \sum_{t=1}^{T} \sum_{j=1}^{N_t} \alpha_{t,j} \langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, b_{t,j} \rangle \right| \\
&\leq \frac{r}{\sqrt{T}} \sqrt{\sum_{t=1}^{T} \sum_{j=1}^{N_t} \langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, b_{t,j} \rangle^2} \\
&\leq \frac{r}{\sqrt{T}} \times \frac{2\sigma r}{c\sqrt{T}} = \frac{2\sigma r^2}{cT}.
\end{aligned}$$

This concludes the proof of Theorem 3. □

Remark 17 *The term $N_T(\theta)$, which is defined according to the feature structures of T tasks, in the above proof intrinsically functions as a regularizer. Note that our proof method of Theorem 3 can be easily extended to the case where some but not all the tasks contribute to producing the regularizer (or the case where some but not all the tasks share common parameters). For example, let the reconstruction property described in Assumption 1 hold when focusing on T' tasks, let denote their indices by $\mathcal{T}$ and let*

$$N_{T'}(\theta) = \frac{1}{T'} \sum_{t \in \mathcal{T}} \frac{1}{n} \sum_{j=1}^{N_t} \ell\left(y_{b_{t,j}}, \langle w_t + \theta, b_{t,j} \rangle\right).$$

Then, we can prove that

$$\max_{z=(x,y), z'_{t,i}=(x'_{t,i}, y'_{t,i}) \in \cup_{s \in \mathcal{T}} \mathcal{Z}_s, t \in \mathcal{T}, i \in \{1,\ldots,n\}} |\langle \theta_{\mathbf{S}} - \theta_{\mathbf{S}^i}, x \rangle| \frac{2\sigma r^2}{cT'}.$$

Proof of Prposition 3. The proof method is similar to that of Proposition 2. However, there are some differences, e.g., the sample in Proposition 2 are i.i.d.; while the samples for multiple tasks are not i.i.d.. Note that the examples of one particular task are i.i.d..

Let

$$\begin{aligned} \Phi(\mathbf{S}) &= \sup_{\theta \in H'} \left(\frac{1}{T} \sum_{t=1}^{T} E_{z_t=(x_t,y_t) \sim \mu_t} \ell\left(y_t, \langle w_t + \theta, x_t \rangle\right) \right. \\ &\quad \left. - \frac{1}{T} \sum_{t=1}^{T} \frac{1}{n} \sum_{j=1}^{n} \ell\left(y_{t,j}, \langle w_t + \theta, x_{t,j} \rangle\right) \right), \qquad (11) \end{aligned}$$

where H' denotes the active hypothesis class of the learning algorithm, which is the set of all the possible outputs of the shared parameter θ, and

$$\begin{aligned} X_t(S_t) &= \sup_{\theta \in H'} \left(E_{z_t=(x_t,y_t) \sim \mu_t} \ell\left(y_t, \langle w_t + \theta, x_t \rangle\right). \right. \\ &\quad \left. - \frac{1}{n} \sum_{j=1}^{n} \ell\left(y_{t,j}, \langle w_t + \theta, x_{t,j} \rangle\right) \right). \end{aligned}$$

Then, $X_1(S_1), \dots, X_T(S_T)$ are independent random variables, and

$$\Phi(\mathbf{S}) \leq \frac{1}{T}\sum_{t=1}^{T} X_t(S_t).$$

We have that

$$|X_t(S_t)| \quad \leq \sup_{\theta} E_{z_t=(x_t,y_t)\sim\mu_t} \frac{1}{n}\sum_{j=1}^{n} |\ell\left(y_t, \langle w_t+\theta, x_t\rangle\right) - \ell\left(y_{t,j}, \langle w_t+\theta, x_{t,j}\rangle\right)| \leq M.$$

Using Hoeffding's inequality, we have

$$\begin{aligned} &\mathbf{Pr}\left\{\frac{1}{T}\sum_{t=1}^{T} X_t(S_t) - E_{\mathbf{S}}\frac{1}{T}\sum_{t=1}^{T} X_t(S_t) \geq \epsilon\right\} \\ &\leq \exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^{T}\frac{4M^2}{T^2}}\right). \end{aligned}$$

Let $\exp\left(\frac{-T\epsilon^2}{2M^2}\right) = \delta$, where $\delta > 0$. Then, we have

$$\epsilon = \sqrt{\frac{2M^2 \ln\left(1/\delta\right)}{T}}. \tag{12}$$

Thus, with probability at least $1-\delta$, the following holds

$$\begin{aligned} \Phi(\mathbf{S}) \quad &\leq \frac{1}{T}\sum_{t=1}^{T} X_t(S_t) \leq E_{\mathbf{S}}\frac{1}{T}\sum_{t=1}^{T} X_t(S_t) + \epsilon \\ &= E_{\mathbf{S}}\frac{1}{T}\sum_{t=1}^{T} X_t(S_t) + \sqrt{\frac{2M^2 \ln\left(1/\delta\right)}{T}}. \end{aligned}$$

According to (Pinelis, 1994), with probability at least $1-\delta$, we have $\|\theta - E_{S_t}\theta_{S_t}\| \leq \frac{2\sigma r}{cT}\sqrt{2n\ln(2/\delta)}$. We now upper bound $E_{\mathbf{S}}\frac{1}{T}\sum_{t=1}^{T} X_t(S_t)$. With probability at least

$1-\delta$, we have

$$\begin{aligned}
& E_{\mathbf{S}}\frac{1}{T}\sum_{t=1}^{T}X_t(S_t) \\
&= E_{\mathbf{S}}\frac{1}{T}\sum_{t=1}^{T}\sup_{\theta\in H'}\left(E_{z_t\sim\mu_t}\ell\left(y_t,\langle w_t+\theta,x_t\rangle\right)\right. \\
&\quad \left.-\frac{1}{n}\sum_{j=1}^{n}\ell\left(y_{t,j},\langle w_t+\theta,x_{t,j}\rangle\right)\right) \\
&\leq \frac{1}{T}\sum_{t=1}^{T}E_{S_t,\sigma}\sup_{\theta\in H'}\frac{1}{n}\sum_{j=1}^{n}\sigma_j\ell\left(y_{t,j},\langle w_t+\theta,x_{t,j}\rangle\right) \\
&= \frac{\sigma}{T}\sum_{t=1}^{T}E_{S_t,\sigma}\sup_{\theta\in H'}\frac{1}{n}\sum_{j=1}^{n}\sigma_j\left\langle\theta,x_{t,j}\right\rangle \\
&= \frac{\sigma}{T}\sum_{t=1}^{T}E_{S_t,\sigma}\sup_{\theta\in H'}\frac{1}{n}\sum_{j=1}^{n}\sigma_j\left\langle\theta-E_{S_t}\theta_{S_t},x_{t,j}\right\rangle \\
&\leq \frac{\sigma}{T}\sum_{t=1}^{T}E_{S_t,\sigma}\sup_{\theta\in H'}\frac{1}{n}\|\theta-E_{S_t}\theta_{S_t}\|\left\|\sum_{j=1}^{n}\sigma_j x_{t,j}\right\| \\
&\leq \frac{\sigma}{T}\sum_{t=1}^{T}\frac{2\sigma r}{cT}\sqrt{2n\ln(2/\delta)}\sqrt{n}r \\
&\leq \frac{2\sigma^2r^2}{cT}\sqrt{2\ln(2/\delta)}. \qquad (13)
\end{aligned}$$

Combining (11), (12) and (13), we can conclude that Proposition 3 holds. □

5 Conclusions

In this paper, we utilized two inductive biases for MTL to derive algorithm-dependent generalization bounds from a uniform stability point of view. One inductive bias is that the tasks share common parameters. The other one is that the feature structures of all tasks are similar. Our analyses justify the claim that the common parameter can be learned with a fast convergence rate. When focusing on one particular task in MTL, the algorithm for learning the shared parameter has a generalization bound with a fast convergence rate of order $\mathcal{O}(1/n)$, where n is the sample size of the particular task. When focusing on the average performance over multiple tasks, the corresponding algorithm has a generalization bound of order $\mathcal{O}(1/T)$, where T is the number of tasks. Moreover, our analyses offer an insight into the advantages of MTL over the traditional single-task learning by showing that tasks could function as regularization, which is a carefully chosen inductive bias and enables MTL to generalize efficiently from a few examples.

We conclude with an open question. It would be valuable to investigate the fast convergence rate of order $\mathcal{O}(1/nT)$ for learning the common parameter θ in MTL problem (1).

Acknowledgment

The work was supported in part by Australian Research Council Projects DP-140102164 and FT-130101457.

References

Ando, R. K., & Zhang, T. (2005). A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, *6*, 1817–1853.

Argyriou, A., Evgeniou, T., & Pontil, M. (2008). Convex multi-task feature learning. *Machine Learning*, *73*(3), 243–272.

Argyriou, A., Pontil, M., Ying, Y., & Micchelli, C. A. (2007). A spectral regularization framework for multi-task structure learning. In *Nips.*

Audiffren, J., & Kadri, H. (2013). Stability of multi-task kernel regression algorithms. In *Proceedings of acml* (pp. 1–16).

Bartlett, P., Kulkarni, S., & Posner, S. (1997). Covering numbers for real-valued function classes. *IEEE Transactions on Information Theory*, *43*(5), 1721–1724.

Bartlett, P. L., & Mendelson, S. (2003). Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, *3*, 463–482.

Baxter, J. (2000). A model of inductive bias learning. *Journal of Artificial Intelligence Research*, *12*(1), 149–198.

Ben-David, S., & Schuller, R. (2003). Exploiting task relatedness for multiple task learning. In *Colt.* Springer.

Bousquet, O., & Elisseeff, A. (2002). Stability and generalization. *Journal of Machine Learning Research*, *2*, 499–526.

Caruna, R. (1993). Multitask learning: A knowledge-based source of inductive bias. In *Icml.*

Chen, J., Tang, L., Liu, J., & Ye, J. (2009). A convex formulation for learning shared structures from multiple tasks. In *Icml.*

Chen, J., Tang, L., Liu, J., & Ye, J. (2013). A convex formulation for learning a shared predictive structure from multiple tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(5), 1025–1038.

Collobert, R., & Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Icml.*

Dudley, R. M. (1967). The sizes of compact subsets of hilbert space and continuity of Gaussian processes. *Journal of Functional Analysis*, *1*(3), 290–330.

Elisseeff, A., & Pontil, M. (2003). Leave-one-out error and stability of learning algorithms with applications. *NATO science series sub series iii computer and systems sciences*, *190*, 111–130.

Evgeniou, T., & Pontil, M. (2004). Regularized multi–task learning. In *Sigkdd* (pp. 109–117).

Gong, P., Zhou, J., Fan, W., & Ye, J. (2014). Efficient multi-task feature learning with calibration. In *Sigkdd.*

Guo, Y., Bartlett, P., Shawe-Taylor, J., & Williamson, R. (2002). Covering numbers for support vector machines. *IEEE Transactions on Information Theory*, *48*(1), 239–250.

Hazan, E., & Kale, S. (2011). Beyond the regret minimization barrier: an optimal algorithm for stochastic strongly-convex optimization. In *Colt.*

Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. *Journal of the American statistical association*, *58*(301), 13–30.

Kakade, S. M., & Tewari, A. (2009). On the generalization ability of online strongly convex programming algorithms. In *Nips.*

Kearns, M., & Ron, D. (1999). Algorithmic stability and sanity-check bounds for leave-one-out cross-validation. *Neural Computation*, *11*(6), 1427–1453.

Koltchinskii, V. (2001). Rademacher penalties and structural risk minimization. *IEEE Transactions on Information Theory*, *47*(5), 1902–1914.

Kumar, A., & Daume, H. (2012). Learning task grouping and overlap in multi-task learning. In *Icml.* ACM.

Kuzborskij, I., & Orabona, F. (2013). Stability and hypothesis transfer learning. In *Icml.*

Lin, B., Yang, S., Zhang, C., Ye, J., & He, X. (2012). Multi-task vector field learning. In *Nips.*

Liu, Q., Liao, X., Carin, H., Stack, J., & Carin, L. (2009). Semisupervised multitask learning. *IEEE transactions on pattern analysis and machine intelligence*, *31*(6), 1074.

Lounici, K., Pontil, M., Tsybakov, A. B., & van de Geer, S. (2009). Taking advantage of sparsity in multi-task learning. In *Colt.*

Massart, P. (2000). Some applications of concentration inequalities to statistics. In *Proceedings of annales de la faculté des sciences de toulouse.*

Maurer, A. (2006). Bounds for linear multi-task learning. *Journal of Machine Learning Research*, *7*, 117–139.

Maurer, A. (2009). Transfer bounds for linear feature learning. *Machine learning*, *75*(3), 327–350.

Maurer, A., Pontil, M., & Romera-paredes, B. (2013). Sparse coding for multitask and transfer learning. In *Icml.* ACM.

McDiarmid, C. (1998). Concentration. In *Probabilistic methods for algorithmic discrete mathematics* (pp. 195–248). Springer.

Micchelli, C. A., & Pontil, M. (2004). Kernels for multi–task learning. In *Nips.*

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2012). *Foundations of machine learning*. MIT press.

Pillonetto, G., Dinuzzo, F., & De Nicolao, G. (2010). Bayesian online multitask learning of gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(2), 193–205.

Pinelis, I. (1994). Optimum bounds for the distributions of martingales in banach spaces. *The Annals of Probability*, 1679–1706.

Pong, T. K., Tseng, P., Ji, S., & Ye, J. (2010). Trace norm regularization: reformulations, algorithms, and multi-task learning. *SIAM Journal on Optimization*, *20*(6), 3465–3489.

Pontil, M., & Maurer, A. (2013). Excess risk bounds for multitask learning with trace norm regularization. In *Colt.*

Rai, P., & Daume, H. (2010). Infinite predictor subspace models for multitask learning. In *Aistats.*

Rakhlin, A., Shamir, O., & Sridharan, K. (2012). Making gradient descent optimal for strongly convex stochastic optimization. In *Icml.* ACM.

Sauer, N. (1972). On the density of families of sets. *Journal of Combinatorial Theory, Series A*, *13*(1), 145–147.

Shalev-Shwartz, S., Shamir, O., Srebro, N., & Sridharan, K. (2010). Learnability, stability and uniform convergence. *Journal of Machine Learning Research*, *11*, 2635–2670.

Shawe-Taylor, J., Bartlett, P. L., Williamson, R. C., & Anthony, M. (1998). Structural risk minimization over data-dependent hierarchies. *IEEE Transactions on Information Theory*, *44*(5), 1926–1940.

Tsianos, K. I., & Rabbat, M. G. (2012). Distributed strongly convex optimization. In *Proceedings of annual allerton conference on communication, control, and computing (allerton).*

Vapnik, V. (2000). *The nature of statistical learning theory*. springer.

Wang, X., Zhang, C., & Zhang, Z. (2009). Boosted multi-task learning for face verification with applications to web image and video search. In *Proceedings of cvpr* (pp. 142–149).

Zhang, T. (2002). Covering number bounds of certain regularized linear function classes. *Journal of Machine Learning Research*, *2*, 527–550.

Zhang, T., Ghanem, B., Liu, S., & Ahuja, N. (2012). Robust visual tracking via multi-task sparse learning. In *Proceedings of cvpr* (pp. 2042–2049).

Zhang, X.-L. (2015). Convex discriminative multitask clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*(1), 28–40.

Zhang, Y. (2015). Multi-task learning and algorithmic stability. In *Proceedings of aaai* (pp. 3181–3187).

Zhang, Y., & Yeung, D.-Y. (2010). Multi-task warped gaussian process for personalized age estimation. In *Proceedings of cvpr* (pp. 2622–2629).

Zhou, D.-X. (2003). Capacity of reproducing kernel spaces in learning theory. *IEEE Transactions on Information Theory*, *49*(7), 1743–1752.

Zhou, J., Chen, J., & Ye, J. (2011). Clustered multi-task learning via alternating structure optimization. In *Nips*.