# Minimum Volume Multi-Task Learning

Bo Liu* Ji Liu† Hao Men‡ Sridhar Mahadevan§ Yong Ge¶ Deguang Kong‖

**Abstract**

Multi-Task Learning (MTL) utilizes the intrinsic relationship among multiple related tasks to reach better generalization and improved performance. This paper studies the problem of multiple related supervised learning tasks with task relatedness in the predictor space. Given the assumption that the predictor relatedness can be interpreted with a *low-dimensional shared structure* in the intrinsic space and a *minimum-volume clusteredness* in the ambient space, the problem formulations with a low-rank regularizer and a non-convex volume constraint is introduced to depict the low-dimensional shared structure and ambient space clusteredness respectively. The objective function is solved via alternating direction method of multipliers(ADMM), which facilitates distributed computation. Theoretical analysis is also conducted on the solution properties with error boundary analysis. Experimental results proove the effectiveness and efficiency of this proposed framework.

## 1 Introduction

Complex real-world problems can usually be decomposed by multiple related tasks and resolved afterward. An intuitive approach is to apply single task learning on each task independently, which will lead to the obvious drawback of not utilizing task relatedness. Growing interest in Multi-Task Learning (MTL) or "Learning to Learn" appeared in recent decades, where multiple related tasks are learned simultaneously by extracting appropriate shared information across tasks. In MTL, multiple tasks are assumed to be related to each other, thus learning them simultaneously is expected to conduct both improved prediction performance and better generalization capability. Applications fields of multi-task learning include bioinformatics, computer vision, natural language processing and finance. There are two major approaches to model relatedness among tasks based on making task related assumptions in predictor space or feature space. The first approach is to assume there exists task relatedness in the predictor space, i.e., the task predictors either share a low-dimensional subspace, or can be clustered in the predictor space. The second is to model task relatedness based on the assumption that tasks share a common feature space instead of assuming task relations in the predictor space [2, 3]. These algorithms are referred as "Multi-Task Feature Learning" (MTFL). In this paper, research is focused on algorithms of the first kind with assumptions on task relatedness in the predictor space.

There are several important topics in MTL including low-dimensional shared structure, task clusteredness, regularization, hierarchy and convex formulation, etc. Low-dimensional shared structure is one of the most critical problems to resolve. Different approaches have been proposed to interpret low-dimensional task relatedness. In many cases, a low rank assumption is made, and then introduced the trace norm regularization in [9, 11]. Another approach assumes that the MTL models lie in a common low-dimensional subspace [5, 7, 2]. In [2], related tasks are assumed to share a latent common basis structure, then this non-convex problem formulation is solved via alternating structure optimization (ASO), which belongs to one of alternating optimization technique. The ASO formulation can also be relaxed to a convex problem in [5].

Besides the low-dimensional shared structure issue, the second important issue is the task clusteredness. A large family of algorithms assume that tasks can be clustered in task space, as investigated in [14, 10, 8, 16]. One approach assumes that all predictors are necessarily close to each other in task space, this can either be measured by various distance metrics, or sharing a common prior in the Bayesian framework [12, 13]. Both assumptions, including low-dimensional shared structure and task clusteredness in the ambient space, are common and realistic assumptions widely utilized in MTL research. However, few research has been conducted on integrating the low-dimensional shared structure with ambient space clusteredness till present.

This paper introduces research to explore the integration of low-dimensional shared structure with minimum volume task clusteredness. In real-world multi-task learning problems, the tasks can usually be composed of components which share common structure in

*School of Computer Science, University of Massachusetts, boliu@cs.umass.edu

†Department of Computer Sciences, University of Rochester, jliu@cs.rochester.edu

‡Bloomberg L.P. R & D, New York City, NY, 10022, hao.men@gmail.com

§School of Computer Science, University of Massachusetts, mahadeva@cs.umass.edu

¶Department of Computer Science, University of North Carolina at Charlotte, yong.ge@uncc.edu

‖Samsung Research America, San Jose, CA, 95134, doogkong@gmail.com

the low-dimensional intrinsic space, and clusteredness in high dimensional ambient space. To depict both task clusteredness and low-dimensional shared structure, the geometric interpretation of task relatedness is explored, i.e., the predictors are clustered and share latent low-dimensional structure in the predictor space. Therefore, the minimum volume constraint and low-rank constraint are introduced to depict the clusteredness and low-dimensional structure respectively, which are both non-convex and non-smooth. The low rank constraint is utilized to model the low-dimensional structure of the latent space embedded in the high-dimensional ambient predictor space. The minimum volume constraint helps to enforce the clusteredness of the predictors. To ensure the problem solvable, convex relaxation is enforced to convert the low rank regularization to trace norm regularization and to convert the minimum volume constraint to the group norm constraint. The algorithm is well designed to be suitable for parallel computation in several perspectives. The objective function is developed to be task-separable by introducing an $l_{2,\infty}$ constraint, and the alternative direction multiplier method (ADMM) is applied to facilitate parallel computation.

Section 2 covers definitions and notations, backgrounds on matrix volume and ADMM are also given. Section 3 presents motivations, problem formulations and algorithms. Section 4 conducts theoretical analysis including the theoretical guarantee of the solution and error boundary analysis. Empirical experimental studies are illustrated in Section 5 to validate the efficacy of the proposed algorithms and conclusions are presented in Section 6.

## 2 Background

Notations in this paper are introduced at the beginning, a brief overview of ADMM is given afterwards, ADMM is adopted as the major solver in this paper.

**2.1 Definitions and Notations** First of all, the fundamental concepts of this paper is the predictor space, a predictor $p$ is a function that maps an input vector $x \in X$ to the corresponding output $y \in Y$. The set $\mathcal{P}$ containing all $p$ comprises a functional space, termed the predictor hypothesis space, or predictor space for short. The objective of MTL is to discover predictor $p$ from $\mathcal{P}$, which minimizes the empirical error of the sample set drawn from a probability distribution $\Xi$. In a linear predictor space $\mathcal{P} \subseteq \mathbb{R}^d$, where $d$ is the number of features, each predictor function is a point that lies within the $\mathbb{R}^d$ space. The loss function of MTL is formulated as [4]

$$L(W) = \frac{1}{2T}\sum_{i=1}^{T}\frac{1}{n_i}||Y_i - X_i W_i||_2^2 \tag{2.1}$$

where $X_i \in \mathbb{R}^{n_i \times d}, Y_i \in \mathbb{R}^{n_i \times 1}$ are the (data, label) pair for the $i$-th task, and $W_i$ is the $i$-th column of model matrix $W$ as defined above, $n_i$ is the number of samples of the $i$-th task, and there are $T$ tasks. Other definitions and notations are summarized in Figure 1.

- $T$: number of tasks; $d$: number of features
- For the $i$-th task $(X_i, Y_i)$, $n_i$ is the number of samples. $Y \in \mathbb{R}^{n_i \times 1}$ is the label matrix, $X_i \in \mathbb{R}^{n_i \times d}$ is the data matrix. $N = \sum_{i=1}^{T} n_i$ is the total number of samples.
- $W \in \mathbb{R}^{d \times T}$: the predictor matrix. We use $W_i := W_{(:,i)} \in \mathbb{R}^{d \times 1}$ to represent the predictor parameter for the $i$-th task. Also, for any predictor component matrix $U, V$, $(\cdot)_i = (\cdot)_{(:,i)}$ stands for the $i$-th column of the matrix for $U, V \in \mathbb{R}^{d \times T}$.
- Frobenius norm: $||A||_F^2 = \sum_{i,j} |a_{ij}|^2$
- Group norm: Let $X \in R^{m \times n}$, Given two vector norms $||\cdot||_r$ and $||\cdot||_p$, the group norm $||X||_{r,p}$ is defined as

  $$||X||_{r,p} = ||(||X_1||_r, ||X_2||_r, \cdots, ||X_n||_r)^T||_p$$

  namely, $||X||_{r,p}$ is to apply $||\cdot||_r$ to each column of $X$, and to apply $||\cdot||_p$ to the resulting vector.
- Trace norm $||X||_*$ is the sum of all the singular values of $X$, $||X||_* = \sum_i \sigma_i$.
- $\langle X, Y \rangle = \text{Trace}(X^T Y)$, $X, Y \in \mathbb{R}^{d \times T}$
- Vol($S$): volume of a matrix $S \in R^{m \times n}$, with non-zero singular values $\sigma_1, \sigma_2, \cdots, \sigma_t$, and Vol($S$) $= 0$ if $t = 0$, otherwise Vol($S$) $= \prod_{i=1}^{t} \sigma_i$

Figure 1: Notation used in this paper.

**2.2 ADMM** ADMM is an approximation of the dual ascent approach by applying Douglas-Rachford operator splitting to the dual problem with Augmented Lagrangian multipliers. A typical application of ADMM is the following problem with separable objective func-

tions (w.r.t variables) and coupled constraint,

$$\min_{x,y} f(x) + g(y), \quad \text{s.t. } Ax + By = c \tag{2.2}$$

The Augmented Lagrangian Multiplier (ALM) formulation $\Psi(x, y, \lambda)$ is as follows

$$\begin{aligned} \Psi(x, y, \lambda) &= f(x) + g(y) + \langle \lambda, Ax + By - c \rangle \\ &\quad + \frac{\rho}{2} ||Ax + By - c||_2^2 \end{aligned} \tag{2.3}$$

ADMM solves the problem by computing the following update at each iteration,

(2.4)

$$\begin{aligned} x^{k+1} &= \arg\min_x f(x) + \langle \lambda, Ax \rangle + \tfrac{\rho}{2} ||Ax + By^k - c||_2^2 \\ y^{k+1} &= \arg\min_y g(y) + \langle \lambda, By \rangle + \tfrac{\rho}{2} ||Ax^{k+1} + By - c||_2^2 \\ \lambda^{k+1} &= \lambda^k + \rho(Ax^{k+1} + By^{k+1} - c) \end{aligned}$$

## 3 Problem Formulations and Algorithms

In this section, different problem formulations have been introduced, then propose the corresponding algorithms. The first formulation assumes that the predictor satisfies both the task clusteredness and the low-dimensional shared structure. The second problem formulation assumes that the predictor is a sum of two components which satisfy the task clusteredness and the low-dimensional relatedness, respectively. ADMM is utilized as the inner solver of the algorithms. It is noted that both problem formulations can be solved through accelerated gradient [9], and using ADMM facilitates large scale parallel computation.

**3.1 Task Relatedness: Low-dimensional Shared Structure and Euclidean Clusteredness** This section explains the motivation of the methods proposed in this paper. We assume that multi-task model includes both the low-dimensional shared structure and the (ambient space) clusteredness structure. The low-dimensional shared structure and clusteredness can be considered as two geometrical properties of task relatedness demonstrated in *low-dimensional intrinsic space* and *high-dimensional Euclidean ambient space*, respectively. One common geometrical interpretation of task relatedness is that the predictor space is a low-dimensional embedding of the ambient $\mathbb{R}^d$ space, which is widely explored in previous MTL research [16, 2]. The low-dimensional shared structure could either be a low-dimensional manifold [1], a low-rank structure [9], or a shared linear subspace [2]. The low-dimensional structure in the predictor space is often captured by either trace norm regularization, or by alternating structure optimization (ASO) [2].

Another common assumption is that tasks are clustered, which is also a geometrical interpretation of task relatedness. Different from the low-dimensional structure sharing assumption, predictor clusteredness assumes that the predictors are not far away from each other in the Euclidean predictor space. One intuitive idea of depicting this "predictor closeness/clusteredness" is to minimize the volume of the convex hull containing all these predictors. However, it is not to resolve due to its non-convexity. An alternative is to minimize the volume of the ball that contains the convex hull. To this end, instead of enforcing the non-convex constraint $V(W) \leq \nu$, controlling the radius of the minimum $l_2$ ball containing all the predictors, wherein the centroid is $C$, and the radius $r$ of the $l_2$ ball is $r = \max_i ||W_i - C||_2$, this is equivalent to the group norm formulation of $\|W - C\mathbf{1}^T\|_{2,\infty} \leq r$, where $\mathbf{1} \in \mathbb{R}^{T \times 1}$ with all entries equal to 1. In the case of a single centroid, the Euclidean mean can be applied to approximate this centroid $C$, i.e., $C \approx W\mathbf{1}/T$. Then the constraint is

$$\|W - C\mathbf{1}^T\|_{2,\infty} \approx \|WD\|_{2,\infty} \leq r, \tag{3.5}$$

where $D = (I - \mathbf{1}\mathbf{1}^T/T)$. Note that $D$ is both symmetric and idem-potent, namely, $D = D^T, D^2 = D$, rank($D$)=$T - 1$.

**3.2 Problem Formulation 1** Given the problem formulation where the model $W$ is assumed to be both low-dimensional and clustered. To depict the low-dimensional latent space structure, low-rank regularization rank($W$) is introduced. On the other hand, a minimum volume constraint is introduced to enforce the predictors to be clustered together. Therefore the problem can be formulated as follows

$$\min_W L(W) + \alpha \text{rank}(W), \quad \text{s.t. } \text{Vol}(W) \leq v \tag{3.6}$$

where the regularizer $\alpha$rank($W$) and the constraint Vol($W$) $\leq v$ are both non-convex and non-smooth. Next to consider the convex relaxations. The rank norm regularization is relaxed to the trace norm regularization [9], and the minimum volume constraint is relaxed as discussed above. Thus (3.6) is relaxed as

$$\min_W L(W) + \alpha ||W||_*, \quad \text{s.t. } ||WD||_{2,\infty} \leq r \tag{3.7}$$

Now the problem can be formulated as a convex smooth objective function with a convex non-smooth trace norm penalty and a convex non-smooth group norm constraint.

**3.3 Algorithm 1** Algorithm 1 aims to solve the objective function of Equation (3.7). The ALM formula-

tion is

$$\begin{aligned}
&\Psi(W,S,U,V,\lambda_1,\lambda_2,\lambda_3)\\
=\ & L(W)+\alpha\|U\|_*+\langle\lambda_1,S-VD\rangle+\frac{\rho_1}{2}||S-VD||^2\\
&+\langle\lambda_2,U-W\rangle+\frac{\rho_2}{2}||U-W||^2\\
&+\langle\lambda_3,V-W\rangle+\frac{\rho_3}{2}||V-W||^2 \quad \text{s.t. } ||S||_{2,\infty}\le r
\end{aligned}$$

The update rule is

$$\begin{aligned}
S^{k+1} &= \mathbf{CC}(V^kD-\frac{\lambda_1^k}{\rho_1},r)\\
U^{k+1} &= \mathbf{SVT}(W^k-\frac{\lambda_2^k}{\rho_2},\frac{\alpha}{\rho_2})\\
V^{k+1} &= \left[\lambda_1^kD-\lambda_3^k+\rho_1S^{k+1}D+\rho_3W^k\right]\cdot\\
&\quad \left(\rho_1D^2+\rho_3\mathrm{I}\right)^{-1}
\end{aligned} \tag{3.8}$$

For each task, $W_i^{k+1}$ is updated as

$$\begin{aligned}
&W_i^{k+1}=(\tfrac{1}{Tn_i}{X_i}^TX_i+(\rho_2+\rho_3)I)^{-1}\cdot\\
&\left(\tfrac{1}{Tn_i}{X_i}^TY_i+\lambda_{2,i}^k+\lambda_{3,i}^k+\rho_2U_i^{k+1}+\rho_3V_i^{k+1}\right)
\end{aligned} \tag{3.9}$$

where **SVT** is the singular value thresholding operator and **CC** is the column-wise group norm constraint operator, which are both detailed in the Appendix. To cache the factorization result, the matrix inversion lemma can be used, which is described in detail in the supplementary material due to space considerations. $\lambda_1^{k+1},\lambda_2^{k+1},\lambda_3^{k+1}$ are updated as

$$\begin{aligned}
\lambda_1^{k+1}&=\lambda_1^k+\rho_1(S^{k+1}-V^{k+1}D)\\
\lambda_2^{k+1}&=\lambda_2^k+\rho_2(U^{k+1}-W^{k+1})\\
\lambda_3^{k+1}&=\lambda_3^k+\rho_3(V^{k+1}-W^{k+1})
\end{aligned} \tag{3.10}$$

The illustration is shown in Figure 2, where model $W$ is assumed to be both low-rank and all the predictors stay within the $l_2$ ball with radius $r$.

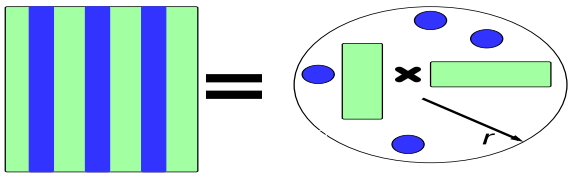


Figure 2: Illustration of Problem Formulation 1

**3.4 Problem Formulation 2** Another problem formulation assumes that the model $W$ is a sum of a low-dimensional shared structure $U$ and a Euclidean clustered structure $V$. To this end, the problem formulation

**Algorithm 1** Minimum Volume MTL 1 (MVMTL1)

**INPUT**: $\{X,Y\}=\{X_i,Y_i\}_{i=1}^T$ for MTL learning
**OUTPUT**: $W$

1: **repeat**
2: Update $S^{k+1},U^{k+1},V^{k+1},W^{k+1}$ via (3.8, 3.9)
3: Update $\lambda_1^{k+1},\lambda_2^{k+1},\lambda_3^{k+1}$ via (3.10)
4: **until** Some stopping criteria is met;

is presented as follows,

$$\min_{U,V} L(U+V)+\alpha\mathrm{rank}(U),\quad \text{s.t. Vol}(V)\le v, \tag{3.11}$$

which can be relaxed as

$$\min_{U,V} L(U+V)+\alpha||U||_*,\quad \text{s.t. } ||VD||_{2,\infty}\le r \tag{3.12}$$

The illustration can be seen in Figure 3, wherein the model $W$ is decomposed to a low-rank component $U$ and a group norm constrained component $V$.

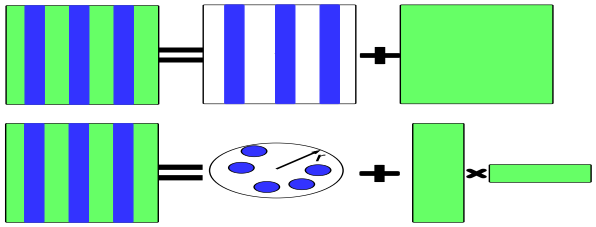


Figure 3: Illustration of Problem Formulation 2

**3.5 Algorithm 2** The loss function of Algorithm 2 is Equation (3.12). The ALM formulation is

$$\begin{aligned}
&\Psi(W,S,U,V,\lambda_1,\lambda_2)\\
=\ & L(W)+\alpha\|U\|_*+\langle\lambda_1,S-VD\rangle+\frac{\rho_1}{2}||S-VD||^2\\
&+\langle\lambda_2,U+V-W\rangle+\frac{\rho_2}{2}||U+V-W||^2,\\
&\text{s.t. } ||S||_{2,\infty}\le r
\end{aligned} \tag{3.13}$$

The update rule is

$$\begin{aligned}
S^{k+1} &= \mathbf{CC}(V^kD-\frac{\lambda_1^k}{\rho_1},r)\\
U^{k+1} &= \mathbf{SVT}(W^k-V^k-\frac{\lambda_2^k}{\rho_2},\frac{\alpha}{\rho_2})\\
V^{k+1} &= \left[\lambda_1^kD-\lambda_2^k+\rho_1S^{k+1}D+\rho_2(W^k-U^{k+1})\right]\cdot\\
&\quad \left(\rho_1D^2+\rho_2\mathrm{I}\right)^{-1}
\end{aligned} \tag{3.14}$$

For each task, $W_i^{k+1}$ is updated as

$$W_i^{k+1} = (\frac{1}{Tn_i}X_i^T X_i + \rho_2 I)^{-1} \cdot \left(\frac{1}{Tn_i}X_i^T Y_i + \lambda_{2,i}^k + \rho_2(U_i^{k+1} + V_i^{k+1})\right) \tag{3.15}$$

And $\lambda_1^{k+1}, \lambda_2^{k+1}$ are updated as

$$\begin{aligned} \lambda_1^{k+1} &= \lambda_1^k + \rho_1(S^{k+1} - V^{k+1}D) \\ \lambda_2^{k+1} &= \lambda_2^k + \rho_2(U^{k+1} + V^{k+1} - W^{k+1}) \end{aligned} \tag{3.16}$$

**3.6 Clustered MVMTL** Instead of assuming the data are centred around its column mean, next problem assume there exists a point $c_1$ in the predictor space which minimizes the max of $l_2$ distance of all the predictors with respect to (w.r.t.) $c_1$. Therefore the objective function is reformulated as

$$\min_{U,V,c_1} L(U+V), \quad \text{s.t. } ||V - c_1 1^T||_{2,\infty} \leq r \tag{3.17}$$

Note this formulation considers the centroid $c_1$ as a variable, and in the Euclidean space $c_1$ is expected to be the mean of columns of $V$.

The aforementioned MTL framework can be easily extended to clustered multi-task learning problems. If there are $k$ task clusters such that

$$\min_{U,V,C} L(U+V) + \alpha||U||_*, \quad \text{s.t. } ||V - CI_c||_{2,\infty} \leq r \tag{3.18}$$

Where $C = [\ c_1 \quad c_2 \quad \cdots \quad c_k\ ]$, and $c_i$ is the centroid of the $i$-th task cluster and $I_c \in \mathbb{R}^{k \times T}$ is the cluster indicator function defined as

$$I_c[i,j] = \begin{cases} 1 & \text{task } j \in \text{cluster } i \\ 0 & \text{task } j \notin \text{cluster } i \end{cases} \tag{3.19}$$

The ALM formulation is

$$\begin{aligned} & \Psi(W,S,U,V,\lambda_1,\lambda_2) \\ = \ & L(W) + \alpha||U||_* \\ & + \langle \lambda_1, S - V + CI_c \rangle + \frac{\rho_1}{2}||S - V + CI_c||^2 \\ & + \langle \lambda_2, U + V - W \rangle + \frac{\rho_2}{2}||U + V - W||^2, \\ & \text{s.t. } ||S||_{2,\infty} \leq r \end{aligned} \tag{3.20}$$

The update rule is

$$\begin{aligned} S^{k+1} &= \mathbf{CC}(V^k - C^k I_c, r) \\ U^{k+1} &= \mathbf{SVT}(W^k - V^k - \frac{\lambda_2^k}{\rho_2}, \frac{\alpha}{\rho_2}) \\ V^{k+1} &= \frac{1}{\rho_1 + \rho_2} \cdot \\ & \quad \left(\lambda_1^k + \rho_1 S^{k+1} + \rho_1 C^k I_c - \lambda_2^k - \rho_2 U^{k+1} - \rho_2 W^k\right) \\ C^{k+1} &= \left[V^{k+1} - S^{k+1} + \frac{\lambda_1^k}{\rho_1}\right] I_c^+ \\ & \quad \left(\rho_1 D^2 + \rho_2 \mathrm{I}\right)^{-1} \end{aligned} \tag{3.21}$$

$W^{k+1}$ is updated as in Equation (3.15),and $\lambda_1^{k+1}, \lambda_2^{k+1}$ are updated as

$$\begin{aligned} \lambda_1^{k+1} &= \lambda_1^k + \rho_1(S^{k+1} - V^{k+1} + C^{k+1}I_c) \\ \lambda_2^{k+1} &= \lambda_2^k + \rho_2(U^{k+1} + V^{k+1} - W^{k+1}) \end{aligned} \tag{3.22}$$

where $I_c^+ = I_c^T(I_c^T I_c^T)^{-1}$ is the right pseudo-inverse The illustration is shown in Figure 4.

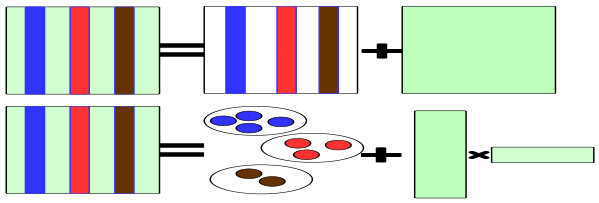

Figure 4: Illustration of Clustered MTL Formulation

**Algorithm 2** Minimum Volume MTL 2 (MVMTL2)

**INPUT**: $\{X, Y\} = \{X_i, Y_i\}_{i=1}^T$ for MTL learning
**OUTPUT**: $W = U + V$

1: **repeat**
2: Update $S^{k+1}, U^{k+1}, V^{k+1}, W^{k+1}$ via (3.14,3.15)
3: Update $\lambda_1^{k+1}, \lambda_2^{k+1}$ via (3.16)
4: **until** Some stopping criteria is met;

## 4 Theoretical Guarantee

In this section, both theoretical analysis for Algorithm 1 and Algorithm 2 are illustrated. Solution properties are derived and the choice of regularization parameters is presented also. Error boundary analysis is also conducted. The theoretical analysis and proof follow the spirit of the analysis in [6]. Due to space limitation, we will focus on the analysis on Algorithm 2, and the analysis for Algorithm 1 will be elaborated in a longer technical report.

**4.1 Solution Property** To obtain the solution property, first convert the $l_{2,\infty}$ constraint to $l_{2,1}$ regularizers. It can be derived that due to the dual norm relation between $l_{2,1}$ norm and $l_{2,\infty}$ norm, for a given radius $r$, there exists a $\beta$ such that the solution of Equation (3.7) is the solution of the following formulation,

$$\min_W L(W) + \alpha\|W\|_* + \beta\|WD\|_{2,1} \tag{4.23}$$

And likewise, there is the corresponding formulation of Equation (3.12)

$$\min_{U,V} L(U+V) + \alpha\|U\|_* + \beta\|VD\|_{2,1} \tag{4.24}$$

Assume the linear predictor for the $i$-th task is formulated as $Y_i = f_i(X_i) + \delta_i$, where the stochastic Gaussian noise vector $\delta_i \in \mathbb{R}^{n_i \times 1}$, and each noise entry $\delta_{ij} \sim N(0, \sigma^2), \forall j \leq n_i$.

**4.2 Error Boundary** Lemma 1 is presented for the choice of the regularization parameter pair $(\alpha, \beta)$.

**Lemma 1**:(Regularization parameters) With high probability

$$\text{prob} \geq 1 - \frac{1}{T}\exp\left(-\frac{1}{2}(t - d\log(1+\frac{t}{d}))\right), \tag{4.25}$$

there exists a pair $(\alpha, \beta)$ such that if $\frac{\alpha}{\sqrt{T}}, \beta \geq \frac{2\sigma}{N}\sqrt{d+t}$, $t$ is a pre-chosen constant, then the global minimizer $(U^*, V^*)$ of problem (4.24), and an arbitrary pair $U, V \in \mathbb{R}^{d\times T}$, the following holds,

$$\begin{aligned} &\frac{1}{2T}\sum_{i=1}^{T}\frac{1}{n_i}||f_i - X_i(U^*+V^*)_i)||_2^2 \\ \leq\ &\frac{1}{2T}\sum_{i=1}^{T}\frac{1}{n_i}||f_i - X_i(U+V)_i)||_F^2 \\ &+\alpha||\mathrm{P}(U-U^*)||_* + \beta||\mathrm{Q}(V-V^*)||_{2,1} \end{aligned} \tag{4.26}$$

where $(\cdot)_i$ denotes the $i$-th column of the matrix. Lemma 1 implies that $(\alpha, \beta)$ should be proportional to $\sqrt{d}$. Given the choice of $(\alpha, \beta)$, we present the error boundary analysis as Theorem 1.

**Theorem 1** (Performance of Algorithm 2): Given $(\alpha, \beta)$ chosen following Lemma 1 ,with high probability

$$\text{prob} \geq 1 - \frac{1}{T}\exp\left(-\frac{1}{2}(t - d\log(1+\frac{t}{d}))\right), \tag{4.27}$$

there is a global optimal solution $(U^*, V^*)$ of problem (3.12) satisfying the following,

$$\begin{aligned} &\frac{1}{2T}\sum_{i=1}^{T}\frac{1}{n_i}||f_i - X_i(U^*+V^*)_i)||_2^2 \\ \leq\ &(1+\varepsilon)\inf_{U,V\in\mathcal{R}(p,q)}\frac{1}{2T}\sum_{i=1}^{T}\frac{1}{n_i}||f_i - X_i(U+V)_i)||_2^2 \\ &+\xi(\varepsilon)\left(\frac{\alpha^2}{\kappa^2(2p)} + \frac{\beta^2}{\tau^2(q)}\right) \end{aligned} \tag{4.28}$$

where $t, \varepsilon > 0$ are two pre-chosen constants, constants $p, q$, real functions $\kappa(\cdot), \tau(\cdot)$, and the restricted set $\mathcal{R}(p,q)$ are all defined in the supplementary material, and $\xi(\varepsilon) = \frac{(\varepsilon+2)^2}{2\varepsilon}$. Theorem 1 implies that error can be decomposed into two terms, where one depends on the formulation of $L(W)$, and the other is controlled by the regularization terms, and both bounds go to zero given the cardinality of the sample set goes to infinity. Similar analysis can be derived for Algorithm 1, it is not elaborated in detail due to space limitations.

## 5 Experiment

This section evaluates the effectiveness of the proposed two algorithms. The data set is introduced first, the performance of the algorithms is measured on two benchmark data sets, the School data and SARCOS data. The School data is composed of the exam scores of 15362 students from 139 schools, where the students are described with 21 features including gender and ethnic group, and each school's test score is a regression task, so altogether there are 139 regression tasks. The SARCOS data is an inverse dynamic prediction problem for a robot arm with 7 degrees-of-freedom. There are 48933 observations with 28 entries for each sample, where the initial 21 entries are features, and the rest 7 entries are target values for the 7 tasks. Several different measures are used to ensure fairness in the comparison study. Normalized mean squared error (nMSE), averaged mean squared error (aMSE), weighted mean squared error (WMSE) and weighted root of sum of squared error (WRSE) are used as evaluation measures, which are defined in the supplementary material.

**5.1 Volume Control Study** The volume control performance of the proposed algorithms is presented in this experiment. Set $\alpha = [0, 10, 20, \cdots, 50, 100, 200, \cdots, 500, 1000]$, and $r = [10, 20, \cdots, 50, 100, 200, \cdots, 500, 1000]$. Figure 5 shows the results of the volume of the learned model (For MVMTL1, it is $W$, for MVMTL2, it is the clustered component $V$). From the top subfigure of Figure 5 we can see that for MVMTL1, the radius of the model is affected by both $\alpha$ and $r$, although

when $\alpha$ is large (e.g., $\alpha > 50$), the impact becomes less obvious. For MVMTL2, we can see from the bottom subfigure of Figure 5 that $\alpha$ only has impact on the $U$ component, and merely has any impact on the radius of the $V$ model, which is consistent with the theoretical analysis.

We also test the relation between the pre-set radius $r$ and the real radius of the learned model, as shown in Figure 6. From the top subfigure, it is clear to observe that for MVMTL1, a large $\alpha$ helps to reduce the radius of the model, and the radius of the true model often goes beyond the pre-set radius when a small $\alpha$ is used. For MVMTL2, the true radius of the $V$ component is almost strictly consistent with the pre-set radius.

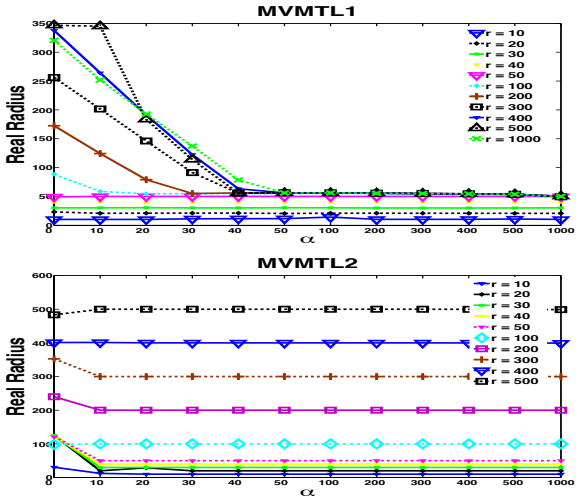


Figure 5: Radius Control Comparison w.r.t $\alpha$

**5.2 Sensitivity Studies on Parameters** This experiment conducts the sensitivity study of the proposed algorithm to the parameter $\alpha$. The result of MVMTL1 is shown in Figure 7. For each $\alpha$, cross-validation is utilized to find out the best $r$. We randomly split the School data into $10\% : 90\%$, where $10\%$ is used for training, and the rest $90\%$ is used for testing. The result is shown in Figure 7, where the elbow effect is shown in choosing parameter $\alpha$. The result for MVMTL2 is similar and is thus not shown due to space limitations.

**5.3 Sample Complexity** This experiment evaluates the performance of the proposed algorithms w.r.t different training ratios. The samples are randomly selected with a percentage of $\{10\%, 20\%, \cdots, 70\%\}$ from the School data as the training set and the rest of the data as the test set. We can observe from Table 1 that as the number of training samples increases, the measures (aMSE, nMSE, WMSE and WRSE) all decrease,

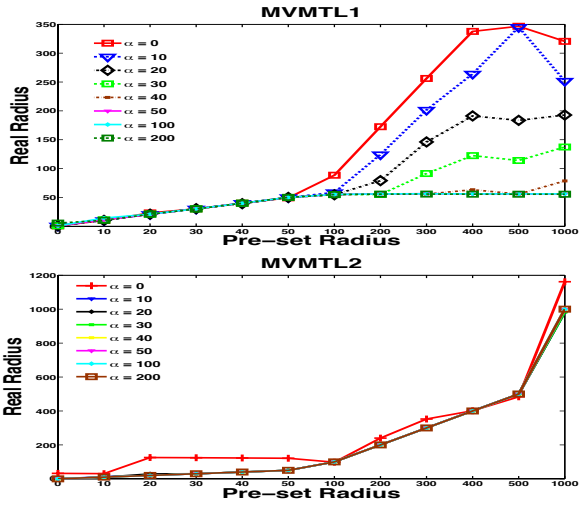


Figure 6: Radius Control Comparison w.r.t Different Pre-set Radiuses

which shows more accurate prediction can be archived by supplying more training samples.

**5.4 Performance Comparison** This experiment compares performance of the proposed algorithms with various peer methods. For school data, we randomly select $10\%, 20\%, 30\%$ of the samples from the training set and use all the rest of the samples for testing. For SARCOS data, $50, 100, 150$ samples are sampled randomly as the training data, and 5000 samples are randomly selected from the rest of the data as the testing data. The methods applied for comparison are multi-task learning with trace-norm regularization [9], $L_{2,1}$ [3] and robust MTL [6]. The code is adopted from MALSAR [15]. Lasso is used as a single task learning method. The result is shown in Table 2 and Table 3, where the numbers in bold represent the best performance among competing methods with the same training samples. From the result shown in Tables 2 and Table 3, it is easy to observe that MVMTL2 performs the best, and RMTL and MVMTL1 both perform the second best. The reason for the similar performance of MVMTL2 and RMTL may be due to the intrinsic relation between their modelling structure, wherein both models are composed of a low-rank component and a group-norm constrained/regularized component, and $l_{2,1}$, $l_{2,\infty}$ norms are dual norms. Although MVMTL1 seems not to have the best performance among tasks, the performances of MVMTL1 are very close to the best method (MVMTL2 or RMTL) in most tasks.

## 6 Conclusion

This paper proposes a novel minimum volume multi-task learning (MVMTL) framework based on the minimum volume assumption. The proposed MVMTL framework depicts the task relatedness with a low-rank structure to capture the low-dimensional shared structure component, and a $l_{2,\infty}$ group norm constraint structure to capture the component of task clusteredness. The low rank task relatedness is computed via a trace-norm regularization, and the non-convex minimum volume assumption is relaxed via task separable group norm constraint, which can be efficiently computed via ADMM. Experimental studies validate the effectiveness of the proposed algorithms.

## References


[1] Arvind Agarwal, Hal Daumé III, and Samuel Gerber. Learning multiple tasks using manifold regularization. volume 23, pages 46–54, 2010.

[2] Rie Kubota Ando and Tong Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *The Journal of Machine Learning Research*, 6:1817–1853, 2005.

[3] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, 2008.

[4] Jianhui Chen, Ji Liu, and Jieping Ye. Learning incoherent sparse and low-rank patterns from multiple tasks. In *KDD*, pages 1179–1188, 2010.

[5] Jianhui Chen, Lei Tang, Jun Liu, and Jieping Ye. A convex formulation for learning shared structures from multiple tasks. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 137–144. ACM, 2009.

[6] Jianhui Chen, Jiayu Zhou, and Jieping Ye. Integrating low-rank and group-sparse structures for robust multi-task learning. In *KDD*, pages 42–50, 2011.

[7] Pinghua Gong, Jieping Ye, and Changshui Zhang. Multi-stage multi-task feature learning. In *Advances in Neural Information Processing Systems 25*, pages 1997–2005. 2012.

[8] Laurent Jacob, Francis Bach, and Jean-Philippe Vert. Clustered multi-task learning: A convex formulation. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 745–752, 2009.

[9] Shuiwang Ji and Jieping Ye. An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 457–464. ACM, 2009.

[10] Abhishek Kumar and Hal Daume III. Learning task grouping and overlap in multi-task learning. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, ICML '12, pages 1383–1390, 2012.

[11] Ting Kei Pong, Paul Tseng, Shuiwang Ji, and Jieping Ye. Trace norm regularization: reformulations, algorithms, and multi-task learning. *SIAM Journal on Optimization*, 20(6):3465–3489, 2010.

[12] Kai Yu, Volker Tresp, and Anton Schwaighofer. Learning Gaussian processes from multiple tasks. In *Proceedings of the 22nd international conference on Machine learning*, pages 1012–1019, 2005.

[13] Jian Zhang, Zoubin Ghahramani, and Yiming Yang. Learning multiple related tasks using latent independent component analysis. *Advances in neural information processing systems*, 18:1585, 2005.

[14] Wenliang Zhong and James Kwok. Convex multitask learning with flexible task clusters. In *Proceedings of the 29th Annual International Conference on Machine Learning*, 2012.

[15] J. Zhou, J. Chen, and J. Ye. *MALSAR: Multi-tAsk Learning via StructurAl Regularization.* Arizona State University, 2011.

[16] Jiayu Zhou, Jianhui Chen, and Jieping Ye. Clustered multi-task learning via alternating structure optimization. *Advances in Neural Information Processing Systems*, 25, 2011.


## Appendix

**Singular Value Thresholding Operator and Column-wise Group Norm Constraint** The two building blocks of the algorithm are singular value thresholding(SVT) and the column-wise group norm constraint. We introduce SVT first.

(1) The standard formulation of $\mathbf{SVT}(Y, \alpha)$ is

$$W = \arg\min_X \frac{1}{2}||X - Y||_F^2 + \alpha||X||_* \tag{6.29}$$

And the closed form solution is to exert SVD on matrix $Y$ and denote $\text{rank}(Y) = t$. If $t > 0$ we do SVD on $Y = U \sum V^T$, wherein $\sum = \text{diag}\{\sigma_1, \sigma_2, \cdots, \sigma_t\}$.

$$\begin{aligned} Y &= U \sum V^T, \sum = \text{diag}\{\sigma_1, \sigma_2, \cdots, \sigma_t\} \\ \bar{\sum} &= \text{diag}\left\{\max\left(\sigma_i - \alpha, 0\right)\right\} \\ W &= U \bar{\sum} V \end{aligned} \tag{6.30}$$

(2) The standard formulation of the Column-wise $l_{2,\infty}$ group norm Constraint (CC) $\mathbf{CC}(Y, r)$ is

$$W = \arg\min_X \frac{1}{2}||X - Y||_F^2, \quad \text{s.t. } ||X||_{2,\infty} \leq r \tag{6.31}$$

Decompose it column-wise, we have

$$W_i = \arg\min_{x_i} \left(\frac{1}{2}||x_i - y_i||_2^2\right), \quad \text{s.t. } ||x_i||_2 \leq r \tag{6.32}$$

It has an analytical solution

$$W_i = \min\left(1, \frac{r}{||y_i||_2}\right) y_i \tag{6.33}$$

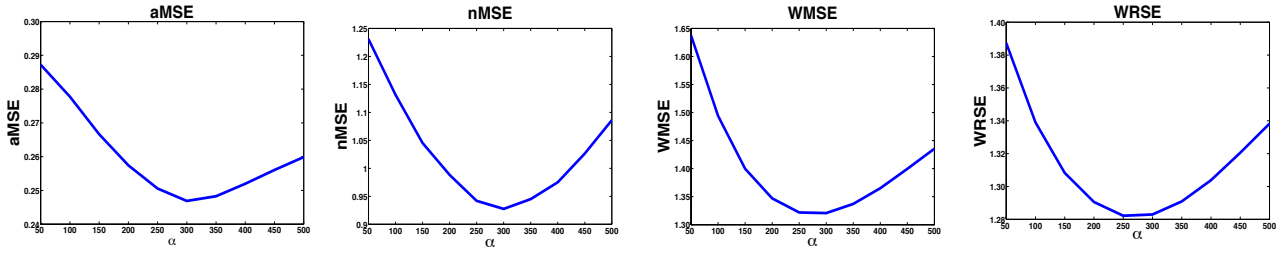


Figure 7: Sensitivity Performance w.r.t $\alpha$

| method | MVMTL1 | | | | MVMTL2 | | | |
|---|---|---|---|---|---|---|---|---|
| sample ratio | aMSE | nMSE | WMSE | WRSE | aMSE | nMSE | WMSE | WRSE |
| 10% | 0.2481 | 0.9216 | 1.3210 | 1.2835 | 0.2364 | 0.8784 | 1.3383 | 1.2177 |
| 20% | 0.2185 | 0.8156 | 1.2057 | 1.2184 | 0.2047 | 0.8056 | 1.2437 | 1.1824 |
| 30% | 0.1951 | 0.7669 | 1.1551 | 1.1640 | 0.1844 | 0.7542 | 1.1179 | 1.0608 |
| 40% | 0.1874 | 0.7534 | 1.1858 | 1.0843 | 0.1765 | 0.7364 | 1.0422 | 0.9624 |
| 50% | 0.1802 | 0.7349 | 1.0947 | 0.9951 | 0.1683 | 0.7240 | 0.9093 | 0.8638 |
| 60% | 0.1723 | 0.7267 | 0.9824 | 0.9218 | 0.1617 | 0.7165 | 0.8735 | 0.7541 |
| 70% | 0.1614 | 0.7126 | 0.9246 | 0.8115 | 0.1559 | 0.6943 | 0.8639 | 0.7102 |

Table 1: Sample Complexity Comparison on School Data

| School | aMSE | | | nMSE | | | WMSE | | | WRSE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| samples | 10% | 20% | 30% | 10% | 20% | 30% | 10% | 20% | 30% | 10% | 20% | 30% |
| Trace | 0.2504 | 0.2156 | 0.2089 | 0.9359 | 0.8211 | 0.7870 | **1.3241** | 1.1773 | 1.1726 | **1.2088** | 1.0859 | 1.0117 |
| Lasso | 0.2682 | 0.2289 | 0.2137 | 1.0261 | 0.8754 | 0.8144 | 1.4940 | 1.3079 | 1.2769 | 1.2759 | 1.1338 | 1.0543 |
| RMTL | 0.2330 | **0.2018** | 0.1844 | 0.9130 | **0.8055** | 0.7600 | 1.3267 | 1.1767 | 1.1201 | 1.2131 | **1.0844** | 1.0127 |
| L21 | 0.2735 | 0.2218 | 0.1903 | 1.0173 | 0.8549 | 0.8206 | 1.3986 | 1.2249 | 1.2206 | 1.2968 | 1.1089 | 1.0364 |
| MVMTL1 | 0.2510 | 0.2185 | 0.1951 | 0.9216 | 0.8156 | 0.7669 | 1.3360 | **1.2057** | 1.1551 | 1.2985 | 1.2184 | 1.0240 |
| MVMTL2 | **0.2324** | 0.2047 | **0.1827** | **0.8784** | 0.8056 | **0.7542** | 1.3383 | 1.2437 | **1.1179** | 1.2177 | 1.1824 | **1.0108** |

Table 2: Performance Comparison on School Data

| SARCOS | aMSE | | | nMSE | | | WMSE | | | WRSE | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| samples | 50 | 100 | 150 | 50 | 100 | 150 | 50 | 100 | 150 | 50 | 100 | 150 |
| Trace | 0.1122 | 0.0805 | 0.0772 | 0.2257 | 0.1531 | 0.1318 | 0.2989 | 0.1670 | 0.1517 | **0.3064** | 0.2362 | 0.2248 |
| Lasso | 0.1228 | 0.0907 | 0.0822 | 0.2337 | 0.1616 | 0.1469 | 0.2959 | 0.1713 | 0.1576 | 0.3072 | 0.2413 | 0.2239 |
| RMTL | **0.0982** | **0.0737** | 0.0674 | **0.2123** | 0.1456 | 0.1245 | 0.2972 | 0.1654 | 0.1414 | 0.3076 | 0.2345 | 0.2179 |
| L21 | 0.1276 | 0.0879 | 0.8115 | 0.2348 | 0.1574 | 0.1396 | 0.2990 | 0.1652 | 0.1429 | 0.3074 | 0.2348 | 0.2191 |
| MVMTL1 | 0.1097 | 0.0945 | 0.0764 | 0.2241 | 0.1496 | 0.1309 | 0.2849 | 0.1534 | 0.1419 | 0.3115 | 0.2283 | 0.2126 |
| MVMTL2 | 0.1078 | 0.0742 | **0.0667** | 0.2127 | **0.1447** | **0.1226** | **0.2797** | **0.1522** | **0.1393** | 0.3067 | **0.2249** | **0.2081** |

Table 3: Performance Comparison on SARCOS Data