Interactive Multi-Task Relationship Learning

Kaixiang Lin and Jiayu Zhou
Department of Computer Science and Engineering, Michigan State University, East Lansing, MI 48824
Email: {linkaixi, jiayuz}@msu.edu

***Abstract*—Multi-task learning (MTL) is a learning paradigm that provides a principled way to improve the generalization performance of a set of related machine learning tasks by transferring knowledge among the tasks. The past decade has witnessed many successful applications of MTL in different domains. In the center of MTL algorithms is how the relatedness of tasks are modeled and encoded in learning formulations to facilitate knowledge transfer. Among the MTL algorithms, the multi-task relationship learning (MTRL) attracted much attention in the community because it learns task relationship from data to guide knowledge transfer, instead of imposing a prior task relatedness assumption. However, this method heavily depends on the quality of training data. When there is insufficient training data or the data is too noisy, the algorithm could learn an inaccurate task relationship that misleads the learning towards suboptimal models. To address the aforementioned challenge, in this paper we propose a novel interactive multi-task relationship learning (iMTRL) framework that efficiently solicits partial order knowledge of task relationship from human experts, effectively incorporates the knowledge in a proposed knowledge-aware MTRL formulation. We propose an efficient optimization algorithm for kMTRL and comprehensively study query strategies that identify the critical pairs that are most influential to the learning. We present extensive empirical studies on both synthetic and real datasets to demonstrate the effectiveness of proposed framework.**

I. Introduction

Supervised learning has been a well studied area of machine learning and there are many efficient algorithms to learn from data and generate predictive models to infer labels for unseen data points. As extensively studied in the statistical learning theory, the quantity and quality of the labeled training data is the key to high-performance models. Unfortunately, even in the big data era, obtaining labeled instances in many real world domains such as biology and healthcare still incurs substantial cost. For example, the National Institute of Aging funded over $60 million to Alzheimer's disease neuroimaging initiative to study the disease and data are collected from less than 1000 patients. The limited sample size largely restricted the study of disease progression with many possible biomarkers.

Interestingly, while machine learning demands a large set of training samples to learn simple concepts, the learning process of human beings allows us link a learning task with what we have learned before and thus we are able to learn complicated cognitive concepts with much less training samples. Motivated by this human learning, the multi-task learning (MTL) paradigm learns related machine learning tasks simultaneously and performs inductive knowledge transfer among the tasks to improve their the generalization performance. MTL has many successful applications in board fields such as data mining, computer vision, text mining, bioinformatics and healthcare analytics [22, 33, 16]. For example, capturing temporal relatedness among multiple learning tasks allows researchers to build high performance disease progression models for Alzheimer's disease by transfer knowledge among time points [36].

One approach to learning multiple tasks is based on the regularized MTL framework [13]. The regularized MTL is extensively studied because of its flexibility to incorporate various learning objectives such as least squares, logistic regression and hinge loss, and to extend them with different kinds of assumptions on how tasks are related. Examples of such task relatedness regularizations include shared sets of features via sparsity inducing norms [23], shared low-dimensional subspace via the nuclear norm [5], and clustering structures via spectral k-means [34]. The same framework can accommodate more complicated assumptions such as dirty models [20] and robust models [10, 15]. Moreover, efficient implementations have been developed for regularized MTL, which can be easily extended to new regularization terms [35].

Many of the regularized MTL methods heavily depend on the prior knowledge of task relatedness. In [12, 21, 25], for example, the prior knowledge of task relatedness are assumed to be known and is then transfered to regularization terms to guide the learning. However, the relationship for all tasks may not always be available. To address this problem, the multi-task relationship learning (MTRL) approaches [32, 14, 31] are studied to *learn the task relationship* in the form of a *task covariance matrix* from the data, representing how similar are

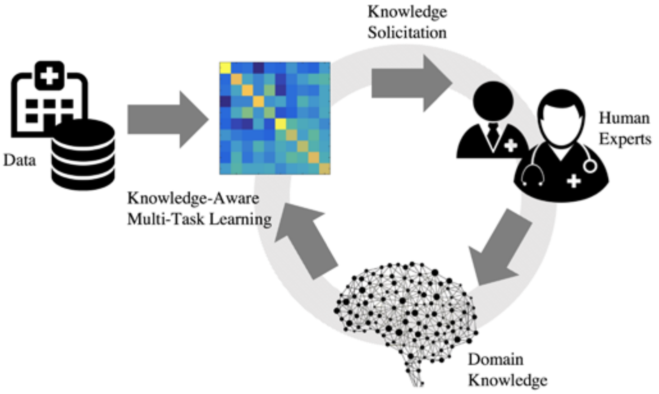


Fig. 1. Overview of the proposed iMTRL framework, which involves human experts in the loop of multi-task learning. The framework consists of three phases: (1) *Knowledge-aware multi-task learning*: learning multi-task learning models from knowledge and data, (2) *Solicitation*: soliciting most informative knowledge from human experts using active learning based query strategy, (3) *Encoding*: encoding the domain knowledge to facilitate inductive transfer.

the two tasks. These methods have been shown to be more effective than others in some learning problems. However, recall that in MTL the training samples are typically insufficient, and thus we may not always be able to infer reliable task relationship from the training data. If misleading task covariance matrix is learned from insufficient and noisy data, the subsequent knowledge transfer guided by such covariance information will not be performed towards the right direction as we expected, and lead to suboptimal models.

In many applications the human experts may have some domain knowledge about how some of tasks are related. For example, the physicians may indicate the predictive models of two disease models should be very similar due to the similarity in the their pathological pathways or dynamics in physiology. In those situations, soliciting and incorporating these domain knowledge in the learning could dramatically improve the generalization performance of learning models. Unfortunately, to the best of our knowledge, little research has been done on this area. We identified a few key questions in area: (1) What type of domain knowledge is suitable for guiding MTL? (2) How the solicited domain knowledge can be effectively incorporated into the MTL formulations; and (3) How the domain knowledge can be efficiently solicited?

To address the aforementioned challenges in MTL, this paper systematically investigated the above questions and propose a novel interactive multi-task relationship learning (iMTRL) framework. Specifically, in the iMTRL framework we propose to solicit the domain knowledge in the form of partial order between two pairs of tasks, which is equivalent to a pairwise relationship between two elements in the task covariance matrix. To effectively incorporate the partial order knowledge, we propose a knowledge aware MTRL (kMTRL) formulation, which learns a task covariance matrix constrained by the partial order relationships in the domain knowledge. We develop an efficient optimization algorithm for the proposed kMTRL. Moreover, since human labeling is very expensive even for weak supervision like tasks relationship, we propose an efficient query strategy for knowledge solicitation. We evaluate the proposed iMTRL framework on both synthetic and real datasets and demonstrate its efficiency and effectiveness.

The reminder of this paper is organized as follows: Section II reviews related works of MTL and active learning involving pairwise constraints. Section III introduces the framework of kMTRL and advanced algorithms. Section IV presents the experimental results on both synthetic and real datasets. Section V concludes the paper.

Notation: We use lowercase letters to denote scalars, lowercase bold letters to denote vectors (e.g. $\mathbf{x}$), uppercase bold letters to denote matrices (e.g. $\mathbf{\Omega}$). We use $\mathbb{R}$ to denote the set of real numbers and $\mathbb{R}_+(\mathbb{R}_{++})$ to denote the subset of non-negative (positive) ones. If $\mathbf{x} \in \mathbb{R}^d$, the p-norm of vector $\mathbf{x}$ is given by $\|\mathbf{x}\|_p = (\sum_{i=1}^d \|x_i\|^p)^{\frac{1}{p}}$. If $\mathbf{A} \in \mathbb{R}^{d\times K}$, we use $\mathbf{a}_j \in \mathbb{R}^d$ to denote the jth column of $\mathbf{A}$ and $\widetilde{\mathbf{a}}_i \in \mathbb{R}^T$ to denote the ith row of $\mathbf{A}$. For all $r, p > 1$, we define the $l_{p,q}$ norm of $\mathbf{A}$ as $\|\mathbf{A}\|_{p,q} = (\sum_{i=1}^d \|\widetilde{\mathbf{a}}_i\|_p^q)^{\frac{1}{p}}$. The set of K integers is denoted as $\mathbb{N}_K = [1, ..., K]$. We use $\mathbf{I}_d$ to denote a $d \times d$ identity matrix, and $\mathbf{1}_d$ to denote a d dimension vector with all elements are 1. Unless stated otherwise, all vectors are column vectors.

II. Related Work

A. Multi-task learning

MTL has been successfully applied to solve many challenging machine learning problems involving multiple related tasks. Recently the regularization based MTL approach has received a lot of attention because of its flexibility and efficient implementations. One major research direction in regularized MTL is to encode the relationship among tasks [12, 21, 14, 32, 25, 8]. The regularized MTL algorithms can be roughly classified into two types: the first involves assumptions about task relatedness, which are then "translated" into proper regularization terms in the regularization to infer a shared representation, that serves as the media of knowledge transfer. An example is the low-rank MTL [12, 21, 25], which seeks a shared low-dimensional subspace in task models, and the tasks are related through the shared subspace. One potential issue in such methods is that the prior knowledge may not always accurate and the assumption may not be suitable for all tasks. Later on some studies focus on infer the task relationship from the dataset [32, 14, 8], e.g, by learning a "covariance matrix" over tasks. Since the learned covariance matrix governing the knowledge transfer is also learned from data, these methods is heavily dependent on the quality and quantity of the training samples available. When an inaccurate task relationship is learned, it will lead to point the knowledge transfer in a wrong direction and lead to suboptimal models, as will be shown in our empirical studies. To alleviate the problem of existing models, we propose an active learning framework which can interactively label the ground truth of task relationship into learning model and guide correct knowledge transfer.

B. Active Learning

There are two common categories of active learning: the pool based and the batch mode. The pool based active learning approaches select the most informative unlabeled instance iteratively, which is then labeled by user, with the goal of learning a better model with less efforts [26]. The selection process is often referred as a *query*. However, such sequential query selection strategy is inefficient in many cases, i.e. adding one labeled data point at a time is typically insufficient to substantially improve the performance of model, and thus the training procedure is very slow. In contrast, the batch-mode active learning approaches select a set of most informative query instances simultaneously. To the best of our knowledge, all previous active learning focus on how to select a group of most informative instances or training samples. In this paper, we instead propose a novel query strategy to query another type of supervision: task relationship. This supervision is intuitive but comes with a significant challenge, i.e., most previous active learning strategies cannot be directly applied.

In our study the task supervision is represented by partial orders which lead to pairwise constraints. There are a few previous studies on the effectiveness of the pairwise constraints [30, 18] under active learning framework. In [18], a clustering algorithm named Active-PCCA was proposed to consider whether two data points should be assigned to the same cluster or not, by which it biases the categorization towards the one expected. The most informative pairwise constraints are selected using the data points on the frontier of those least well-defined clusters. In [30], the authors studied a semi-supervised clustering algorithm with a query strategy to choose pairwise constraints by selecting the most informative instance, as well as data points in its neighborhoods. The pairwise constraints are in the form of Must-link and Cannot-link, which restrict two data points should be in the same class or not. However, those methods are developed for clustering algorithms. How to select pairwise constraints on task relationship that are suitable for the MTL framework remains to be an open problem. In this paper, we study query strategies for task relationship supervision, including one novel strategy based on the inconsistency of learning model.

C. Interactive Machine Learning

Interactive machine learning (IML) is a systematic way to include human in the learning loop, observing the results of learning and providing feedback to improve the generalization performance of learning model [1]. It has provided a natural way to integrate background knowledge into the learning procedure [2, 4, 29, 3]. For example, the system called "perception-based classication" (PBC) [4] has been pioneered to offer an interactive way to construct decision. The PBC is able to construct a smaller decision tree but the accuracy achieved doesn't has significant improve compared to other decision tree methods such as C4.5. The decision construction has been further extend in [29]. They also found out that users can build good models only when the visualization are apparent in two dimension. Manual classifier construction is not successful for large data set involving high dimension interaction. In [2], an end-user IML system (ReGroup) are proposed to be able to help people create customized groups in social networks. In [3], the authors developed an IML system named as (CueT) to learn the triaging decision about network alarm in a highly dynamic environment. In this paper, iMTRL is proposed to combine the domain knowledge in terms of task relationship to build learning models. Our work is exploring a completely novel problem compared to the previous studies in interactive machine learning.

III. Interactive Multi-Task Relationship Learning

In this section, we first review the strengths and potential issues of the multi-task relationship learning in Subsection III-A, which motivate the overarching framework of the proposed interactive multi-task relationship learning (iMTRL) in Subsection III-B. Subsection III-C presents the knowledge-aware MTRL (kMTRL) formulation and algorithm. Subsection III-E introduces the novel batch mode knowledge query strategy based on active learning.

A. Revisit the Multi-task Relationship Learning

Before discussing the iMTRL framework, we revisit the multi-task relationship learning (MTRL) [32], one popular MTL model that learns not only the prediction models but also task relationship. The MTRL framework has a well founded Bayesian background. Assume we have K related learning tasks, and in each task we are given a data matrix and their corresponding responses. Let d be the number of features. For the task k, we are given m samples and their corresponding responses, collectively denoted by $\mathbf{X}^k = [(\mathbf{x}_1^k)^T; (\mathbf{x}_2^k)^T; ...; (\mathbf{x}_m^k)^T] \in \mathbb{R}^{m \times d}$ and $\mathbf{y}^k \in \mathbb{R}^m$. We assume that the responses come from a linear combination of features with a Gaussian noise, so that for sample j from task i, we have $y_j^i = \mathbf{w}_i^T \mathbf{x}_j^i + b_i + \epsilon_i$, where distribution of the noise is given by $\epsilon_i \sim \mathcal{N}(0, \epsilon_i^2)$. The goal of the learning is to estimate the task parameters $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_K]$ and bias term $\mathbf{b} = [b_1, ..., b_K]$ for all K tasks from data.

Based on the assumption we can write the likelihood of y_j^i given $\mathbf{x}_j^i, \mathbf{w}_i, b_i$, and ϵ_i is given by:

$$p(y_j^i | \mathbf{x}_j^i, \mathbf{w}_i, b_i, \epsilon_i) \sim \mathcal{N}(\mathbf{w}_i^T \mathbf{x}_j^i + b_i, \epsilon_i^2),$$

where $\mathcal{N}(\mathbf{m}, \Sigma)$ represents the multivariate distribution with mean $\mathbf{m}$ and covariance matrix Σ [7]. The prior on $\mathbf{W} = (\mathbf{w}_1, ..., \mathbf{w}_K)$ is given by:

$$p(\mathbf{W}|\epsilon_i) \sim (\prod_{i=1}^{K} \mathcal{N}(\mathbf{w}_i | 0_d, \sigma_i^2 \mathbf{I}_d)) q(\mathbf{W}),$$

where $\mathbf{I}_d \in \mathbb{R}^{d \times d}$ is the identity matrix. The first term is the extension of ridge prior to the multi-task learning setting, which controls the model complexity of each task $\mathbf{w}_i$. The second term refers to the task relationship, in which MTRL tries to learn the covariance of $\mathbf{W}$ using a matrix-variate normal distribution for $q(\mathbf{W})$

$$q(\mathbf{W}) = \mathcal{MN}_{d \times K}(\mathbf{W} | 0_{d \times K}, \mathbf{I}_d \otimes \mathbf{\Omega}),$$

where $\mathcal{MN}_{d \times K}(\mathbf{M}, \mathbf{A} \otimes \mathbf{B})$ denotes matrix-variate normal distribution with mean $\mathbf{M} \in \mathbb{R}^{d \times K}$, row covariance matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$ and column covariance matrix $\mathbf{B} \in \mathbb{R}^{K \times K}$. According to the Bayes's theorem, the posterior distribution for $\mathbf{W}$ is proportional to the product of the prior distribution and the likelihood function [7]:

$$p(\mathbf{W} | \mathbf{X}, \mathbf{y}, \mathbf{b}, \epsilon, \sigma, \mathbf{\Omega}) \propto p(\mathbf{y} | \mathbf{X}, \mathbf{W}, \mathbf{b}, \epsilon) p(\mathbf{W} | \mathbf{\Omega}, \sigma), \quad (1)$$

where $\mathbf{X}$ collectively denotes the data matrix for K tasks and $\mathbf{y} = [\mathbf{y}^1, ..., \mathbf{y}^k]$ denotes labels for all data points.

By taking negative logarithm of Eq. (1), the maximum a posteriori estimation of $\mathbf{W}$ and maximum likelihood estimation of $\mathbf{\Omega}$ is given by:

$$\min_{\mathbf{W}, \Omega} \sum_{k=1}^{K} \frac{1}{\epsilon_k^2} \|\mathbf{y} - \mathbf{X}^k \mathbf{w}_k - b_k \mathbf{1}_{n_k}\|_F^2 + \frac{1}{\sigma_k^2} \mathrm{tr}(\mathbf{W}\mathbf{W}^T) + \mathrm{tr}(\mathbf{W} \mathbf{\Omega}^{-1} \mathbf{W}^T) + d \ln(\mathbf{\Omega}). \quad (2)$$

In the above formulation, the last term $d \ln(\mathbf{\Omega})$ controls the complexity of $\mathbf{\Omega}$ and is a concave function. In order to obtain a convex objective function, the MTRL proposed to use $\mathrm{tr}(\mathbf{\Omega}) = 1$ instead to control the complexity and project $\mathbf{\Omega}$ to be a positive semi-definite matrix. As such, the objective function of MTRL is derived as follows:

$$\min_{\mathbf{W},\Omega} \sum_{k=1}^{K} \frac{1}{n_k}\|\mathbf{y}^k - \mathbf{X}^k\mathbf{w}_k - b_k\mathbf{1}_{n_k}\|_F^2 + \frac{\lambda_1}{2}\mathrm{tr}(\mathbf{W}\mathbf{W}^T) \quad (3)$$

$$+ \frac{\lambda_2}{2}\mathrm{tr}(\mathbf{W}\mathbf{\Omega}^{-1}\mathbf{W}^T). \text{ s.t. } \mathbf{\Omega} \succeq 0, \mathrm{tr}(\mathbf{\Omega}) = 1$$

An alternating algorithm is proposed in [32] to solve this formulation. The algorithm iteratively solves two steps: first it optimizes Eq (3) with respect to $\mathbf{W}$ and $\mathbf{b}$ when $\mathbf{\Omega}$ is fixed; it then optimizes the objective function with respective to $\mathbf{\Omega}$, which admits a closed-form solution:

$$\mathbf{\Omega} = (\mathbf{W}^T\mathbf{W})^{1/2}/\mathrm{tr}((\mathbf{W}^T\mathbf{W})^{1/2}). \quad (4)$$

We note that there is a feedback loop in the learning of MTRL as illustrated above. The MTRL achieves knowledge transfer among task models via the task relation matrix $\mathbf{\Omega}$, and the task models will be used to estimate $\mathbf{\Omega}$. If the $\mathbf{\Omega}$ can be learned correctly or can closely represent the true tasks relationship, it will benefit learning on the tasks parameters $\mathbf{W}$ by guiding the knowledge transfer in a good direction. In turn, the better tasks parameters will help the algorithm to identify a more accurate estimation of $\mathbf{\Omega}$. The positive feedback loop is the key to help building a good MTRL model. On the contrary, the training procedure will be biased to wrong direction once we keep getting misleading feedbacks in the loop. To be more specific, once data is either low-quality or insufficient-quantity, the $\mathbf{\Omega}$ will indicate an inaccurate direction to transfer the knowledge among tasks, which leads to a negative feedback in the loop. This will end up learning a model with poor generalization performance, examples of which will be elaborated in the empirical studies.

Another remark is that in Eq. (3), due to the relaxation, the solution of $\mathbf{\Omega}$ is no longer the extract solution from the maximum likelihood estimation of column covariance matrix derived from Eq. (2). The advantages of the objective function in Eq. (3) compared to Eq. (2) have been discussed in details in [32]. We would like to further point out that the learned $\mathbf{\Omega}$ is actually a better representation of tasks relationship than the column covariance matrix. Recall that the covariance suggests the extent that elements in two vectors move to the same direction. Suppose we have tasks parameters $\mathbf{W} \in \mathbb{R}^{d\times K}$, the unbiased sample covariance can be computed by $\mathbf{C} = \mathbf{W}_c^T\mathbf{W}_c/(d-1)$, where $\mathbf{W}_c = \mathbf{W} - \mathbf{1}_d^T\mathbf{1}_d\mathbf{W}/d$ is the centralized tasks models. This measure is only meaningful when there are enough number of dimension d and the variance contains in tasks parameters. If $\mathbf{W} = [1, -2; 1, -2]$, the covariance matrix will return an all-zero matrix which will not indicate a correct relationship. Instead, an accurate estimation can be inferred by using Eq. (4). We can obtain a correlation matrix $\mathbf{Corr} = [1 - 1; -1, 1]$ from $\mathbf{\Omega}$.

The above discussions lead to two important conclusions: (1) The $\mathbf{\Omega}$ can indicate a genuine task relationship. (2) Maintaining an accurate $\mathbf{\Omega}$ is the key in this learning procedure.

B. The iMTRL Framework

In MTL scenarios, the quality and quantity of training data usually impose significant challenges to the learning algorithms. The task covariance matrix $\mathbf{\Omega}$ inferred from the data may not always give an accurate description of the true task relationship, which in turn would prevent effective knowledge transfer. Fortunately, in many real-world applications, human experts possess indispensable domain knowledge about relatedness among some tasks. For example, when building models predicting different regions of the brain from clinical features, neuroscientist and medical researcher can reveal important relationship among the regions. As such, solicit feedback from human experts on task relationship and encode them as supervision is especially attractive. To achieve this goal we need to answer the following problems:

1) What type of knowledge representation can be efficiently solicited from human experts, and also can be used to effectively guide the learning algorithms?
2) How to design MTL algorithm that combines the domain knowledge and data-driven insights?
3) How to effectively solicit knowledge, reducing the workload of the human experts by supplying only the most important knowledge that affects the learning system?

In this paper we propose a framework of interactive multi-task Machine learning (iMTRL), which provides an integrated solution to address the above challenging questions. The framework is illustrated in Fig 1. The iMTRL is an iterative learning procedure that involves human experts in the loop. In each iteration, the learning procedure involves the following:

1) *Encoding*. The domain knowledge of task relationship is represented as partial orders, and can be encoded in the learning as pairwise constraints.
2) *Knowledge-Aware Multi-Task Learning*. We propose a novel MTL algorithm that infers models and task relationship from data and conform the solicited knowledge.
3) *Active Learning based Knowledge Query*. To maximize the usefulness of solicited knowledge, we propose a knowledge query strategy based on active learning.

It is natural and intuitive to use partial orders as the knowledge presentation for task relationship. Query a question that whether the task i and j are more related than task i and k is much easier than asking to which extent the task i and j are related to each other. For example, ith task and jth tasks has positive relationship while the ith task and kth task has negative relationship, then this relationship is represented by a partial order $\mathbf{\Omega}_{i,j} \geq \mathbf{\Omega}_{i,k}$. The focus of this paper is the algorithm development for iMTRL and we make a few assumptions to alleviate common issues in using this presentation and simply our discussions:

Assumption 1: The domain knowledge acquired from human expert is accurate. The expert may choose not to label if he/she is not confident.

Assumption 2: The acquired partial orders are compatible, i.e. when $\mathbf{\Omega}_{i,j} > \mathbf{\Omega}_{i,k}$ and $\mathbf{\Omega}_{i,k} > \mathbf{\Omega}_{k,p}$ are established, the $\mathbf{\Omega}_{i,j} < \mathbf{\Omega}_{k,p}$ cannot be included.

If this situation happens, we can discard the less important constraints and make the remain constraints be compatible. The importance of constraints can be measured by the Inconsistency which we will introduced in Definition 2.

C. A knowledge-aware extension of MTRL

Assume in the current iteration of iMTRL, our domain knowledge is stored in a set $\mathcal{T}$ defined by:

$$\mathcal{T} = \{\mathbf{\Omega} : \mathbf{\Omega}_{i_1,j_1} \geq \mathbf{\Omega}_{i_2,j_2} \ \forall (i_1, j_1, i_2, j_2) \in S\}, \tag{5}$$

where each pairwise constraint has specified a preferred half-space that an ideal solution $\mathbf{\Omega}$ should belong to, and the set S contains the indexes of tasks selected by our query strategy. The partial order information is more important than the magnitude of $\mathbf{\Omega}$. The reason is that if we multiply each element in $\mathbf{\Omega}$ with a scalar a, it's equal to solve the Eq. (7) replacing λ_2 with $a\lambda_2$ [11]. Hence, the magnitude of elements in $\mathbf{\Omega}$ can be adjusted simultaneously without changing the results. But the order of pairs in $\mathbf{\Omega}$ is a more important structure to encode. These algorithmic advantages reinforced our choice of using pairwise constraints to represent domain knowledge.

We note that the constraints in Eq. (5) would lead to a trivial solution that $\mathbf{\Omega}_{i_1,j_1} = \mathbf{\Omega}_{i_2,j_2} \ \forall (i_1, j_1, i_2, j_2) \in S$, which is apparently not the effect we seek. To overcome this problem, we add a positive parameter c so that we can assure the elements in $\mathbf{\Omega}$ preserve the true pair wise order. Hence, the convex set $\mathcal{T}$ is changed to:

$$\mathcal{T} = \{\mathbf{\Omega} : \mathbf{\Omega}_{i_1,j_1} \geq \mathbf{\Omega}_{i_2,j_2} + c, \ \ \forall (i_1, j_1, i_2, j_2) \in S\}. \tag{6}$$

The proposed knowledge-aware multi-task relationship (kMTRL) learning extends the MTRL by enforcing a feasible space for $\mathbf{\Omega}$ specified by $\mathcal{T}$. To this end, the kMTRL formulation is given by the following optimization problem:

$$\begin{aligned} \min_{\mathbf{W},\mathbf{b},\mathbf{\Omega}} \mathcal{F}(\mathbf{W}, \mathbf{b}, \mathbf{\Omega}) = \sum_{k=1}^{K} \frac{1}{n_k} \|\mathbf{y}^k - \mathbf{X}^k \mathbf{w}_k - b_k \mathbf{1}_{n_k}\|_F^2 \\ + \frac{\lambda_1}{2}\mathrm{tr}(\mathbf{W}\mathbf{W}^T) + \frac{\lambda_2}{2}\mathrm{tr}(\mathbf{W}\mathbf{\Omega}^{-1}\mathbf{W}^T) \\ \text{s.t.} \qquad \mathbf{\Omega} \succeq 0, \ \mathrm{tr}(\mathbf{\Omega}) = 1, \ \mathbf{\Omega} \in \mathcal{T} \end{aligned} \tag{7}$$

We note that even though the problem of kMTRL is considered to be more challenging to solve than MTRL because of additional constraints introduced in $\mathcal{T}$, the solution space of kMTRL is much smaller because each constraint cuts the solution space in half, and the optimization algorithms may converge faster in this case.

D. Efficient Optimization for kMTRL

The proposed kMTRL is a convex optimization problem, and we propose to solve it using an alternating algorithm:

Step 1: We first optimize the objective function with respect to $\mathbf{W}$ and $\mathbf{b}$ given a fixed $\mathbf{\Omega}$. This step can either be solved using the linear system [32] or off-the-shelf solvers such as CVX [17] and FISTA [6]. Different solvers can be applied depending on the nature of the data: first order solvers such as FISTA is more scalable when there are many samples, while solving linear system can be more efficient as feature dimension is high.

Step 2: Given $\mathbf{W}$ and $\mathbf{b}$, the objective function with respect to $\mathbf{\Omega}$ is given by an analytical solution using Eq. (4).

Step 3: The $\mathbf{\Omega}$ is projected to the convex set:

$$\mathbf{T} = \{\mathbf{\Omega} | \mathbf{\Omega} \in \mathcal{T}, \mathbf{\Omega} \succeq 0, \mathrm{tr}(\mathbf{\Omega}) = 1\}$$

by solving the Euclidean projection problem below:

$$\min_{\mathbf{\Omega}} \|\mathbf{\Omega} - \hat{\mathbf{\Omega}}\|_F^2, \quad s.t. \ \mathbf{\Omega} \in \mathbf{T}$$

where the $\hat{\mathbf{\Omega}}$ is the analytical solution we obtained from the Eq. (4). This objective function can be solved efficiently using a successive projection algorithm [19] that iteratively projects the solution to each constraint in the set.

The KKT analysis [9] of the above optimization problem leads to the property summarized in Theorem 1, and leads to Algorithm 2. To simplify the discussion, we requires the true pair orders are in the form of $\mathbf{\Omega}_{i1,j1} \geq \mathbf{\Omega}_{i2,j2}$.

Theorem 1: Suppose that $\mathcal{T} = \{\mathbf{\Omega} : \mathbf{\Omega}_{i1,j1} \geq \mathbf{\Omega}_{i2,j2} + c\}$, then, for any $\mathbf{\Omega} \in \mathbb{R}^{K \times K}$, the projection of $\mathbf{\Omega}$ to the convex set $\mathcal{T}$ is given by:

$$\mathrm{Proj}(\mathbf{\Omega}) = \mathbf{\Omega} \text{ if } \mathbf{\Omega} \in \mathcal{T},$$

otherwise

$$\mathrm{Proj}(\mathbf{\Omega}) = \mathbf{\Omega}^* = \begin{cases} \mathbf{\Omega}^*_{i1,j1} = \frac{1}{2}(\mathbf{\Omega}_{i1,j1} + \mathbf{\Omega}_{i2,j2} + c) \\ \mathbf{\Omega}^*_{i2,j2} = \frac{1}{2}(\mathbf{\Omega}_{i1,j1} + \mathbf{\Omega}_{i2,j2} - c) \\ \mathbf{\Omega}^*_{p,q} = \mathbf{\Omega}_{p,q}, \ \forall (p,q) \neq (i1, j1) \text{ and } (i2, j2) \end{cases}$$

In practice, the term $\mathbf{W}^T\mathbf{W}$ is not guaranteed to be a full rank matrix. In fact, in a typical MTL setting $\mathbf{W}$ is a low rank matrix and thus the $\mathbf{\Omega}$ calculated by Eq. (4) is also a rank deficiency matrix. Moreover, recall that the operation that projects $\mathbf{\Omega}$ to a convex set has a very high chance lead to a singular matrix. The numerical problems during the inversion of the singular matrix $\mathbf{\Omega}$ will lead to a meaningless inverse of task relation matrix and corrupt the training procedure. Therefore, we propose to solve a perturbed version of our original objective function Eq. (7) as follows:

$$\begin{aligned} \min_{\mathbf{W},\mathbf{b},\mathbf{\Omega}} \mathcal{F}(\mathbf{W}, \mathbf{b}, \mathbf{\Omega}) = \sum_{k=1}^{K} \frac{1}{n_k} \|\mathbf{y}^k - \mathbf{X}^k \mathbf{w}_k - b_k \mathbf{1}_{n_k}\|_F^2 \\ + \frac{\lambda_1}{2}\mathrm{tr}(\mathbf{W}\mathbf{W}^T) + \frac{\lambda_2}{2}\mathrm{tr}(\mathbf{\Omega}^{-1}(\mathbf{W}\mathbf{W}^T + \epsilon\mathbf{I})), \\ \text{s.t.} \qquad \mathbf{\Omega} \succeq 0, \ \mathrm{tr}(\mathbf{\Omega}) = 1, \ \mathbf{\Omega} \in \mathcal{T} \end{aligned} \tag{8}$$

where $\mathcal{T}$ follows the definition in Eq. (6). As a result, the analytical solution of $\mathbf{\Omega}$ in **Step 2.** is thus replaced by the following:

$$\mathbf{\Omega} = (\mathbf{W}^T\mathbf{W} + \epsilon\mathbf{I})^{1/2} / \mathrm{tr}((\mathbf{W}^T\mathbf{W} + \epsilon\mathbf{I})^{1/2}. \tag{9}$$

The algorithm to solve the objective function Eq. (8) is presented in Algorithm 1. This algorithm can be interpreted as alternately performing supervised and unsupervised steps. In the supervised step we learn the task specific parameters ($\mathbf{W}$ and $\mathbf{b}$). In unsupervised step we get the task relationship

Algorithm 1 ($\boldsymbol{\Omega}$, $\mathbf{W}$, $\mathbf{b}$) = kMTIL($\{\mathbf{X}^k, \mathbf{y}^k\}_k^K$, S, λ_1, λ_2, c)

```
Require: Training data $\{\mathbf{X}^k, \mathbf{y}^k\}_k^K$, constraints set $S$, regularization
    parameters $\lambda_1$, $\lambda_2$, a positive number $c$. Randomly initialize $\mathbf{W}^0$.
    $\boldsymbol{\Omega}^0 = \mathbf{I}/d$.
while $\mathbf{W}$ and $\boldsymbol{\Omega}$ are not converge do
    Compute $\{\mathbf{W}, \mathbf{b}\} = \arg\min_{\mathbf{W},\mathbf{b}} \mathcal{F}(\mathbf{W}, \mathbf{b}, \boldsymbol{\Omega})$
    Compute $\boldsymbol{\Omega}$ using Eq. (4)
    $\boldsymbol{\Omega}$ = Proj($\boldsymbol{\Omega}$, $S$, $n$, $c$)
end while
return $\mathbf{W}$, $\mathbf{b}$, $\boldsymbol{\Omega}$
```

Algorithm 2 Projection $\boldsymbol{\Omega}$ = Proj($\boldsymbol{\Omega}$, S, n, c)

```
Require: Task correlation matrix $\boldsymbol{\Omega}$, constraints set $S$, max iteration
    $n$, a positive number $c$.
for $i = 1, ..., n$ do
    while $\forall(i_1, j_1, i_2, j_2) \in S$ do
        if $\boldsymbol{\Omega}_{i_1,j_1} < \boldsymbol{\Omega}_{i_2,j_2}$ then
            $\boldsymbol{\Omega}_{i_1,j_1} = \frac{1}{2}(\boldsymbol{\Omega}_{i_1,j_1} + \boldsymbol{\Omega}_{i_2,j_2} + c)$
            $\boldsymbol{\Omega}_{i_1,j_1} = \frac{1}{2}(\boldsymbol{\Omega}_{i_1,j_1} + \boldsymbol{\Omega}_{i_2,j_2} - c)$
        end if
    end while
    Dynamic update $c = c \times 0.9$
    Project $\boldsymbol{\Omega}$ to be a positive semi-definite matrix
    if All constraints are satisfied then
        break
    end if
end for
return $\boldsymbol{\Omega}$
```

Algorithm 3 Obtain queries $\mathcal{T}$ = query($\mathbf{W}$, $\boldsymbol{\Omega}$, n)

```
Require: The task correlation matrix $\boldsymbol{\Omega}$, the model parameter matrix
    $\mathbf{W}$ for all tasks, the number of pairwise constraints $n$ selected
    to be query
Compute $\hat{\boldsymbol{\Omega}} = (\mathbf{W}^T\mathbf{W})^{1/2}/\mathrm{tr}((\mathbf{W}^T\mathbf{W})^{1/2})$
while $\forall(i_1, j_1, i_2, j_2)$ do
    Compute $\boldsymbol{\Omega}_{(i_1,j_1,i_2,j_2)}$ and $\hat{\boldsymbol{\Omega}}_{(i_1,j_1,i_2,j_2)}$
end while
while $\forall(i_1, j_1, i_2, j_2)$ do
    Compute Inc$_{(i_1,j_1,i_2,j_2)}$
end while
Select $n$ pairs with highest scores into the set $\mathcal{T}$
return $\mathcal{T}$
```

Algorithm 4 The proposed iMTRL framework

```
Require: Training sets $\{\mathbf{X}^k, \mathbf{y}^k\}_k^K$, number of selected queries $\mathbf{q}$.
    regularization parameters $\lambda_1$, $\lambda_2$, positive number $c$, $\mathcal{T}^0 = \emptyset$
1: for $i = 1, ..., n$ do
2:     ($\boldsymbol{\Omega}^i$, $\mathbf{W}^i$, $\mathbf{b}^i$) = kMTIL($\{\mathbf{X}^k, \mathbf{y}^k\}_k^K$, $\mathcal{T}^{i-1}$, $\lambda_1$, $\lambda_2$, $c$)
3:     $\mathcal{T}^i$ = query($\mathbf{W}^i$, $\boldsymbol{\Omega}^i$, $\mathbf{q}_i$)
4:     $\mathcal{T}^i$ = $\mathcal{T}^i \cup \mathcal{T}^{i-1}$
5: end for
6: $\boldsymbol{\Omega} = \boldsymbol{\Omega}^i$, $\mathbf{W} = \mathbf{W}^i$, $\mathbf{b} = \mathbf{b}^i$
7: return $\boldsymbol{\Omega}$, $\mathbf{W}$, $\mathbf{b}$
```

matrix from the task parameters. Finally, the last supervised step we encode prior knowledge to the task relationship matrix $\boldsymbol{\Omega}$. We repeat the steps iteratively until converge.

E. Batch Mode Pairwise Constraints Active learning

There are too many possible pairs for human experts to label them all, and thus the efficiency of iMTRL framework heavily relies on the quality of the pairs selected by the system. In this subsection, we discuss the important question of how to efficiently solicit the domain knowledge. Specifically, we would like to select the pairs that are most informative to the learning process. We propose an efficient heuristic query strategy as elaborated as follows.

We first design a score function for pairwise constraints based on the *inconsistency* in the model. To explain the inconsistency, we denote the analytical solution calculated by $\mathbf{W}$ as $\hat{\boldsymbol{\Omega}} = (\mathbf{W}^T\mathbf{W})^{1/2}/\mathrm{tr}((\mathbf{W}^T\mathbf{W})^{1/2})$ and the difference between elements $\boldsymbol{\Omega}_{i_1,j_1}$ and $\boldsymbol{\Omega}_{i_2,j_2}$ in the learned $\boldsymbol{\Omega}$ as $\boldsymbol{\Omega}_{(i_1,j_1,i_2,j_2)} = \boldsymbol{\Omega}_{i_1,j_1} - \boldsymbol{\Omega}_{i_2,j_2}$. Then inconsistency in the model is defined as follows:

Definition 2: Inconsistency is defined as:

$$\mathrm{Inc}_{(i_1,j_1,i_2,j_2)} = \mathrm{sign}(i_1, j_1, i_2, j_2)|\boldsymbol{\Omega}_{(i_1,j_1,i_2,j_2)} - \hat{\boldsymbol{\Omega}}_{(i_1,j_1,i_2,j_2)}|,$$

where $\mathrm{sign}(i_1, j_1, i_2, j_2) = \frac{\boldsymbol{\Omega}_{(i_1,j_1,i_2,j_2)}\hat{\boldsymbol{\Omega}}_{(i_1,j_1,i_2,j_2)}}{|\boldsymbol{\Omega}_{(i_1,j_1,i_2,j_2)}\hat{\boldsymbol{\Omega}}_{(i_1,j_1,i_2,j_2)}|}$.

The Inc$_{(i_1,j_1,i_2,j_2)}$ represents two types of inconsistency:

Negative inconsistency: Given that the pairwise orders of two relationship matrices ($\boldsymbol{\Omega}$ and $\hat{\boldsymbol{\Omega}}$) are not consistent, i.e. $\boldsymbol{\Omega}_{i_1,j_1} > \boldsymbol{\Omega}_{i_2,j_2}$, but $\hat{\boldsymbol{\Omega}}_{i_1,j_1} < \hat{\boldsymbol{\Omega}}_{i_2,j_2}$ or vice versa, the Inc$_{(i_1,j_1,i_2,j_2)}$ is always negative. The smaller the Inc$_{(i_1,j_1,i_2,j_2)}$ is, the higher is the heuristic score.

Positive inconsistency: Given that the pairwise orders of two relationship matrices are consistent, then the inconsistency comes from $\|\boldsymbol{\Omega}_{(i_1,j_1,i_2,j_2)} - \hat{\boldsymbol{\Omega}}_{(i_1,j_1,i_2,j_2)}\|$. The larger the Inc$_{(i_1,j_1,i_2,j_2)}$ is, the higher is the heuristic score .

Note that the disorder of two pairs are more important that the difference of two pairs, and all pairs with negative inconsistency has the priority to be selected over those with positive inconsistency. At the first iteration, before adding any pairwise constraints into the training procedure, the learned $\boldsymbol{\Omega}$ is very close to the analytical solution calculated from $\mathbf{W}$, i.e. $\boldsymbol{\Omega}_{(i_1,j_1,i_2,j_2)} = \hat{\boldsymbol{\Omega}}_{(i_1,j_1,i_2,j_2)}$, except for the disturb of numerical term $\epsilon\mathbf{I}$. Therefore, the inconsistency is caused by some numerical issues in the first round. Therefore at the first training iteration, there is no negative inconsistency. As the number of constraints added into the model, the inconsistency will appear and the query strategy will become more effective in this situation. The Algorithm 3 describes the query strategy.

Finally, we summarize all procedures of iMTRL in Algorithm 4. The line 1 means there are n iterations learning procedures need to be conducted. The line 2 corresponds to the knowledge-aware MTL step in our iMTRL framework. The line 3 is to solicit the domain knowledge and line 4 is to answer the query and encoding the knowledge into the model.

IV. Experiments

A. Importance of High-Quality Task Relationship

In this subsection, we conduct experiments to show that encoding an accurate task relationship will significantly enhance the performance of MTRL. The effectiveness of MTRL has already been demonstrated in [32], in which the authors showed that MTRL can infer an accurate task relationship from a relatively clean dataset with sufficient training samples.

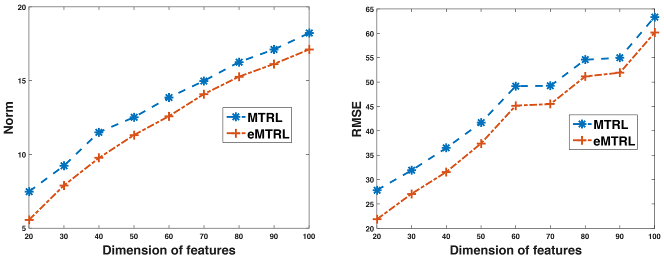
Here we use a toy example to show that MTRL would infer a misleading relationship when noise presents and there are insufficient training samples. The toy dataset is generated as follows. There are three tasks with data sampled from $y = 3x + 10$, $y = -2x + 5$ and $y = 10x + 1$, respectively. For each tasks we generate 5 samples from a uniformly distribution in $[0, 10]$. The function outputs for three tasks are corrupted by a Gaussian noise with zero mean and standard variance equal to 30, 10 and 10, respectively. According to the generative regression functions, we expect that the correlation between the first task and third task is close to 1 and for the rest of pairs is close to -1. We use the linear kernel of MTRL with $\lambda_1 = 0.01$ and $\lambda_2 = 0.05$. The learned $\mathbf{\Omega}$ gives a correlation matrix as follows:

$$\begin{bmatrix} 1 & 0.9999 & -0.9999 \\ 0.9999 & 1 & -1 \\ -0.9999 & -1 & 1 \end{bmatrix}$$

From the above matrix we see that the learned relationship for task 1 is opposite to the supposed relationship, because of the highly noised data. This will leads to suboptimal solution for $\mathbf{W} = [-3.7283, -2.6605, 3.0105]$, as compared to the ground truth $\mathbf{W} = [3, -2, 10]$. On the other hand, if we encode the true tasks relationship by fixing the $\mathbf{\Omega}$ to be the ground truth during the learning process, with the exactly same parameters setting as above. We can then learn a model $\mathbf{W} = [0.6850, -0.3878, 2.5840]$ that is closer to the ground truth in terms of l_2 norm and keeps the correct tasks relationship. This procedure is denoted as truth- encoded multi-task relationship learning (eMTRL) in this subsection.

This observation motivates us to further explore the effectiveness of eMTRL. We created synthetic dataset by generating $K = 10$ tasks parameters $\mathbf{w}_i$ and b_i from a uniform distribution between 0 and 1. Each task contains 25 samples drawn from a Gaussian distribution with zero means and the variance equals to 10. The function response is also corrupted by a Gaussian noise with zero mean and has a variance of 5. We split this synthetic dataset to training, validation and testing set. Out of the 25 samples for each tasks, 20% are for training, 30% for validation and 50% for testing. We fix the number of samples and the number of tasks, vary the number of features from 20 to 100. The parameters λ_1 and λ_2 have been tuned in $[1\text{e}{-3}, 1\text{e}{-2}, 1\text{e}{-1}]$ and $[0, 1\text{e}{-3}, 1\text{e}{-2}, 1\text{e}{-1}, 1, 10, 1\text{e}2, 1\text{e}3]$, respectively.

The performance has been evaluated using Root Mean Square Error (RMSE) and Frobenius norm between learned task model and the ground truth task model. The results shown in Figure 2 indicate that encoding the knowledge about task relationship will significantly benefit the prediction. Even though eMTRL is not a practical model because we can never know the true task relationship, the experimental results confirm that there is a huge potential to improve predictive performance if we can take advantage of domain knowledge. The experimental results in next section will show how to efficiently solicit and incorporate the domain knowledge about tasks relationship into the learning.



(a) (b)

Fig. 2. Performance of MTRL and eMTRL as the number of features changing, in terms of (a) Frobenius norm and (b) RMSE. MTRL [32] learns both task models and task relationship at the same time, while eMTRL here learns the task models while the task relationship $\mathbf{\Omega}$ is fixed to ground truth, i.e. encoding the correct domain knowledge about the task relationship.

B. *Effectiveness of Query Strategy*

In this subsection, we conduct the experiments to show that encoding the domain knowledge in the form of partial order is useful. We follow the same synthetic data set with 20 feature dimension generated above. The same setting of splitting training, testing and validation dataset, and 5 fold random split validation are applied. The parameters λ_1 and λ_2 have been tuned in $[1\text{e}{-3}, 1\text{e}{-2}, 1\text{e}{-1}]$ and $[0, 1\text{e}{-3}, 1\text{e}{-2}, 1\text{e}{-1}, 1, 10, 1\text{e}2, 1\text{e}3]$, respectively. After the learning algorithm converges, we compare the the pairwise constraints are chosen by the proposed query strategy and the randomly selected strategy. The results of two strategies are reported in Table I. We see the trend that both of the proposed query strategy and the random selection reach better generalization performance as the number of incorporated pairwise constraints increases. To be more specific, the results in first column is worse than all the results using query strategy and most of the results using random selection. This show that solicit the domain knowledge in terms of pairwise constraints is effective. On the other hand, when comparing the results of the proposed query strategy and random selection, we see that our query strategy selects important pairwise constraints, leading to a better model than the random query. When the number of pairwise constraints is larger than 5, the proposed query strategy works consistently better than random selection.

C. *Interactive Scheme for Query Strategy*

To further analysis our query strategy, we also explore different interactive schemes in our query strategy. There are multiple ways to query a certain amount of partial orders. We can either query many times and each time with less labeling efforts, or vice versa. We use *kMTRL-a-b* to denote a total b constraints and each time we query a constraints (the human expert needs to interact with the system b/a times). The different interactive scheme will highly impact the user experience. For example, kMTRL-10-100 needs to query experts 10 times and experts need to label 10 constraints at each time. Also, it takes 10 training iterations which is much more expensive than other schemes. In contrast, kMTRL-100-100 only needs to query experts once, which is the most efficient scheme. However, this scheme cannot benefit from the iterative process of iMTRL. The pairwise constraints added in

TABLE I
THE AVERAGE RMSE OF QUERY AND RANDOM STRATEGY ON TESTING DATASET OVER 5 RANDOM SPLITTING OF TRAINING AND VALIDATION SAMPLES.

number of constraints	0	5	10	15	20	25	30	35	40
Query Strategy	1.1387	1.1267	1.1224	1.1117	1.1125	1.1101	1.1102	1.1137	1.1168
Random Selection	1.1387	1.1255	1.1390	1.1284	1.1165	1.1285	1.1379	1.1382	1.1364

TABLE II
THE AVERAGE RMSE COMPARISON OF COMPETING METHODS ON THE SCHOOL DATASET AND MMSE DATASET. THE FIRST COLUMN IS THE PERCENTAGE OF TRAINING SAMPLES IN EACH TASK. THE kMTRL METHODS OUTPERFORMS ALL OTHER METHODS

School	RR	MTL-L	MTL-l21	MTRL	kMTRL-20	kMTRL-40	kMTRL-60	kMTRL-80
5%	1.1737±0.0041	1.1799± 0.0047	1.176± 0.0043	1.0615 ± 0.0167	1.0584 ± 0.0128	1.0553 ± 0.0155	**1.0551** ± 0.0158	1.0551 ± 0.0159
10%	1.1428±0.0306	1.1485 ± 0.0293	1.1477 ± 0.0282	0.9872 ± 0.0057	0.9823 ± 0.0030	0.9805 ± 0.0014	**0.9803** ± 0.0018	0.9803 ± 0.0018
15%	1.0665±0.0395	1.0699 ± 0.0405	1.0700 ± 0.0399	0.9491 ± 0.0060	0.9334 ± 0.0057	**0.9321** ± 0.0081	0.9322 ± 0.0083	0.9323 ± 0.0082
20%	0.9756±0.0157	0.9774 ± 0.0153	0.9776 ± 0.0149	0.9047 ± 0.0031	0.8966 ± 0.0123	0.8906 ± 0.0123	0.8844 ± 0.0022	**0.8843** ± 0.0019
MMSE	RR	MTL-L	MTL-l21	MTRL	kMTRL-5	kMTRL-10	kMTRL-15	kMTRL-20
2%	0.9503± 0.1467	0.9319±0.1497	0.9314±0.1693	0.9106 ± 0.0976	0.9113 ± 0.0982	**0.9058** ± 0.0926	0.9058 ± 0.0926	0.9058 ± 0.0926

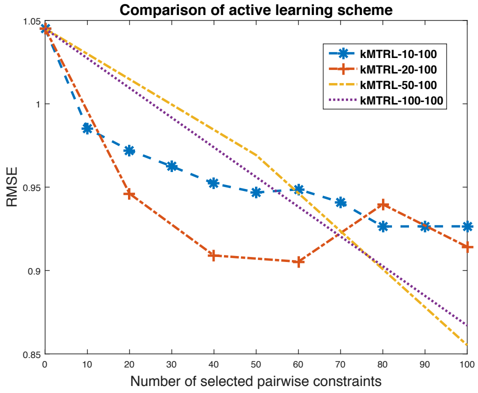


Fig. 3. The averaged RMSE of kMTRL using different setting of query strategy. The kMTRL-10-100 means selecting 10 pairwise constraints at the end of each iteration, start from zero, add 10 pairwise constraints at a time, until 100 constraints. For all 4 schemes, kMTRL with zero constraints is equivalent to MTRL. Results are the average over 5 fold random splitting.

previous iterations will affect the model and won't be selected again. This will reveal other important constraints. Taking a one iteration scheme cannot utilize this information. The results are summarized in Figure 3. We see that kMTRL-50-100 achieves the best performance. Therefore, the best scheme indicate that our query strategy is mostly effective when we balance the two parameters, and thus it does not require intensively interaction with experts and meanwhile utilizes the previous information effectively[1].

D. Performance on Real Datasets

The school dataset is a widely used benchmark dataset for multi-task regression problem. It contains 15372 students with 28 features from 139 secondary schools in the year of 1985, 1986 and 1987, provided by the Inner London Education Authority(ILEA). The task is to predict the score for students in 139 schools. The experimental settings are explained as follows. We first split the dataset into training, validation and testing datasets. The percentage of testing samples varies from 10% to 25% of all samples each tasks in original dataset. Taking the 10% testing dataset as an example, we perform 3-fold random split on the rest 90% data. Each fold has 20% samples for training and 70% for testing. The same random splitting are applied to the three datasets.

Another real dataset we used in this paper is Alzheimer's Disease Neuroimaging Initiative (ADNI) database[2]. The experimental setup is same as described in the paper [37]. The goal is to predict the successive cognition status of patients based on the measurements at the screening or the baseline visit. We use 2% samples for training, 10% for testing and the rest for validation. We also perform 3-fold random split on this dataset. The predictive performance of the competing methods listed below are reported on the real datasets:

- RR: This approach refers to ridge regression.
- MTL-L: This approach refers to the low-rank multi-task learning with trace norm regularization [5].
- MTL-L21: This approach refers to multi-task joint feature learning using $l_{2,1}$ norm that selects a subset of features shared by all tasks [24].
- MTRL: This approach refers to the multi-task relationship learning as we described in Section III [32].
- kMTRL-N: This approach refers to the proposed kMTRL method with N pairwise encoded into the model.

We tune the regularization parameters on $\mathbf{W}$ in $[1e{-}3, 1e{-}2, 1e{-}1]$ for RR, MTL-L and MTL-L21. The regularization parameters λ_1 and λ_2 in Eq.(8) are tuned in [1e−3, 1e−2, 1e−1] and [0, 1e−3, 1e−2, 1e−1, 1, 10, 1e2, 1e3] respectively. The best parameters are selected based on the performance on the validation set. The performance of learned models are measured by RMSE on the testing dataset. The experimental results are shown in Table II, from which we see that kMTRL achieves the best results. In this experiment, we adopt the scheme kMTRL-20-80 for school dataset and kMTRL-5-20 for MMSE dataset as described in previous subsection.

E. Case Study: Brain Atrophy and Alzheimer's Disease

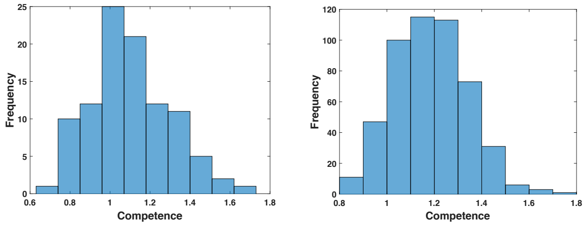
In this section we apply the proposed iMTRL framework to study the brain atrophy patterns and how the changes in the brain is associated to different clinical dementia scores and symptoms that are related to Alzheimer's disease (AD). It is estimated that there are currently 5 million Americans have AD, and AD has become one of the leading causes of death in the United States. Since AD is characterized by structural atrophy in the brain, there is a pressing demand

[1] Code is publicly available at https://github.com/illidanlab/iMTL

[2] Data is publicly available at http://adni.loni.usc.edu/

TABLE III
THE NAME OF THE BRAIN REGIONS IN FIGURE 5, WHERE (C) DENOTES CORTICAL PARCELLATION AND (W) DENOTES WHITE MATTER PARCELLATION.

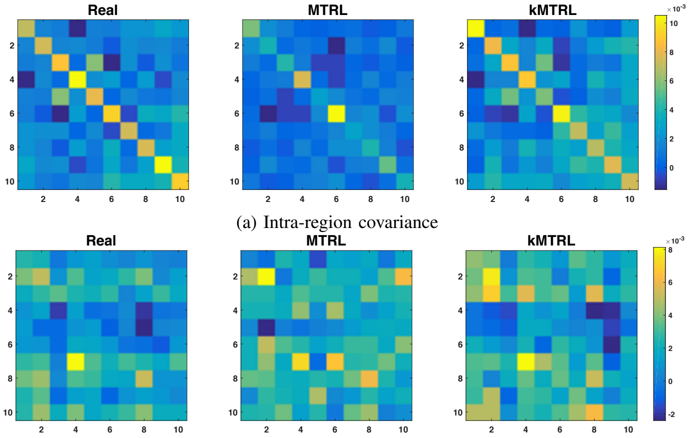
#	Intra-region	Inter-region Row	Inter-region Column
1	(C) Right Caudal Middle Frontal	(W) Right Putamen	(C) Right Inferior Temporal
2	(C) Right Pericalcarine	(W) Left Cerebral Cortex	(C) Left Rostral Middle Frontal
3	(W) Corpus Callosum Mid Anterior	(W) Right Ventral Diencephalon	(C) Right Pars Triangularis
4	(W) Right Cerebellum Cortex	(C) Right Caudal Anterior Cingulate	(C) Right Precentral
5	(W) Corpus Callosum Central	(C) Left Temporal Pole	(C) Right Medial Orbitofrontal
6	(C) Left Bank ssts	(C) Right Postcentral	(C) Left Pars Triangularis
7	(C) Right Pars Opercularis	(C) Right Precentral	(C) Right Superior Parietal
8	(C) Left Isthmus Cingulate	(W) Right Cerebral Cortex	(C) Right Inferior Parietal
9	(C) Left Supramarginal	(C) Left Isthmus Cingulate	(C) Left Pars Orbitalis
10	(C) Right Inferior Temporal	(C) Left Superior Frontal	(W) Corpus Callosum Central



(a)

(b)

Fig. 4. The distribution of competence on (a) intra-region covariance and (b) inter-region covariance. kMTRL performs better than MTRL when competence> 1. Higher competence indicates better performance achieved by kMTRL as compared to MTRL. We see in a majority of regions the kMTRL outperforms the MTRL.



(a) Intra-region covariance

(b) Inter-region covariance

Fig. 5. Comparison of sub-matrices of covariance among (left) task covariance using 90% all data points that is considered as “ground truth”, (middle) the covariance matrix learned via MTRL on 20% data and (right) the covariance matrix learned via kMTRL on 20% data with 0.8% pair-wise constraints queried by the proposed query scheme.

of understanding how the brain atrophy is related to the progression of the disease.

In this paper we study how the structural features of brain regions can be related to 51 cognitive markers such as, Alzheimers Disease Assessment Scale (ADAS), clinical dementia rating (CDR), Global Deterioration Scale (GDS), Hachinski, Neuropsychological Battery, WMS-R Logic, and other neuropsychological assessment scores. We are interested in predicting the volume of brain areas extracted from the structural magnetic resonance imaging (MRI). We use the ADNI cohort consisting 648 subjects whose baseline MRI images passed quality control. We used the FreeSurfer tool to extract the 99 brain volumes from regions of interest (ROIs) of the baseline MRI images. Considering the prediction of the volume of each ROI as a learning task, we thus have a collection of 99 learning tasks, with each task having 648 samples and 51 features. Since the brain regions are related during the aging process and Alzheimer’s progression, the MTL approach can be used to improve the performance by considering such relatedness among brain regions.

We adopt the same experimental setting as in the previous experiments, where we compare the MTRL with the proposed kMTRL by querying and adding pair-wise expert knowledge and inspecting the effectiveness of the queried task relationship supervision. We show the differences among the (1) task covariance using 90% all data points that is considered as “ground truth”, (2) the covariance matrix learned via MTRL on 10% data and (3) the covariance matrix learned via kMTRL on 10% data with 0.8% pair-wise constraints queried by the proposed query scheme. Since the complete 99×99 covariance matrices are hard to visualize, we choose investigate two types of subregions of the covariance matrices: (a) a random intra region of the covariance of the size 10×10 (row regions and column regions are the same) and (b) a random inter region of the covariance of the size 10×10 (row regions and column regions are different). We define the *competence* metric to quantify how the quality of the sub-covariance:

$$\|\Omega_{\text{MTRL}} - \Omega_{\text{real}}\|_F / \|\Omega_{\text{kMTRL}} - \Omega_{\text{real}}\|_F, \qquad (10)$$

where the kMTRL performs better than MTRL when competence > 1, and the higher the better. We repeatedly choose random sub-covariances and the distribution of the competence is shown in the Figure 4, indicating that in a majority of cases knowledge can improve relationship estimation.

We visualize two sub-covariance matrices in Figure 5, whose regions are shown in Table III. In Figure 5(a), we see that the covariances from both the ground truth and the kMTRL discourage the positive knowledge transfer from *Right Cerebellum Cortex*, which agrees with the pathological characteristics of AD [27], where cerebellum does not correlate with the progression of AD. Also the positive correlation between *Corpus Callosum Mid Anterior* and *Corpus Callosum Central* is identified in both the ground truth and the kMTRL, and ignored by MTRL. The significant reduced corpus callosum size was previously reported in AD studies [28], and the progression patterns of the two regions can be similar because of the physical distance between the two regions. Figure 5(b), we see that the unsubstantiated strong correlation between *Right Precentral* and *Left Pars Triangularis* as found in MTRL has been largely suppressed by the domain knowledge.

However, since we only specified partial order relationship, there are chances the proposed kMTRL algorithm may "overutilize" the supervision, as we notice that some unsubstantiated positive correlations involving *Right Ventral Diencephalon* are introduced to the covariance. We plan to further elaborate the findings and clinical insights of AD and dementia in the journal extension of this paper.

V. Conclusion

The multi-task relationship learning (MTRL) could learn an inaccurate task relationship when there are insufficient training data or the data is too noisy, and would mislead the learning towards suboptimal models. In this paper, we proposed a novel interactive multi-task relationship learning (iMTRL) framework that efficiently solicits partial order knowledge of task relationship from human experts, effectively incorporates the knowledge in a proposed knowledge-aware MTRL formulation. We proposed efficient optimization algorithm for kMTRL and comprehensively study query strategies that identify the critical pairs that are most influential to the learning. Extensive empirical studies on both synthetic and real datasets demonstrated the effectiveness of proposed framework.

Acknowledgement

This material is based in part upon work supported by the National Science Foundation under Grant IIS-1565596, IIS-1615597 and Office of Naval Research N00014-14-1-0631.

References

[1] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza. Power to the people: The role of humans in interactive machine learning. AAAI, 2014.

[2] S. Amershi, J. Fogarty, and D. Weld. Regroup: Interactive machine learning for on-demand group creation in social networks. In *Proceedings of the SIGCHI*, pages 21–30. ACM, 2012.

[3] S. Amershi, B. Lee, A. Kapoor, R. Mahajan, and B. Christian. Cuet: human-guided fast and accurate network alarm triage. In *Proceedings of the SIGCHI*, pages 157–166. ACM, 2011.

[4] M. Ankerst, C. Elsen, M. Ester, and H.-P. Kriegel. Visual classification: an interactive approach to decision tree construction. In *SIGKDD*, pages 392–396. ACM, 1999.

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

[6] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[7] C. M. Bishop. Pattern recognition. *Machine Learning*, 2006.

[8] E. V. Bonilla, K. M. Chai, and C. Williams. Multi-task gaussian process prediction. In *NIPS*, pages 153–160, 2007.

[9] S. Chang, G.-J. Qi, C. C. Aggarwal, J. Zhou, M. Wang, and T. S. Huang. Factorized similarity learning in networks. In *2014 IEEE ICDM*, pages 60–69. IEEE, 2014.

[10] J. Chen, J. Zhou, and J. Ye. Integrating low-rank and group-sparse structures for robust multi-task learning. In *ACM SIGKDD*. ACM, 2011.

[11] P. Dutilleul. The mle algorithm for the matrix normal distribution. *J STAT COMPUT SIM*, 64(2):105–123, 1999.

[12] T. Evgeniou, C. A. Micchelli, and M. Pontil. Learning multiple tasks with kernel methods. In *JMLR*, pages 615–637, 2005.

[13] T. Evgeniou and M. Pontil. Regularized multi–task learning. In *SIGKDD*, pages 109–117. ACM, 2004.

[14] H. Fei and J. Huan. Structured feature selection and task relationship inference for multi-task learning. *Knowledge and information systems*, 35(2):345–364, 2013.

[15] P. Gong, J. Ye, and C. Zhang. Robust multi-task feature learning. In *SIGKDD*, pages 895–903. ACM, 2012.

[16] P. Gong, J. Zhou, W. Fan, and J. Ye. Efficient multi-task feature learning with calibration. In *SIGKDD*, pages 761–770. ACM, 2014.

[17] M. Grant, S. Boyd, and Y. Ye. Cvx: Matlab software for disciplined convex programming, 2008.

[18] N. Grira, M. Crucianu, and N. Boujemaa. Active semi-supervised fuzzy clustering for image database categorization. In *Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, pages 9–16. ACM, 2005.

[19] S.-P. Han. A successive projection method. *Mathematical Programming*, 40(1-3):1–14, 1988.

[20] A. Jalali, S. Sanghavi, C. Ruan, and P. K. Ravikumar. A dirty model for multi-task learning. In *NIPS*.

[21] T. Kato, H. Kashima, M. Sugiyama, and K. Asai. Multi-task learning via conic programming. In *NIPS*, pages 737–744, 2008.

[22] S. Kim and E. P. Xing. Tree-guided group lasso for multi-task regression with structured sparsity. 2010.

[23] J. Liu, S. Ji, and J. Ye. Multi-task feature learning via efficient $\ell_{2,1}$-norm minimization. In *Proceedings of the 25th conference on UAI*, pages 339–348. AUAI Press, 2009.

[24] J. Liu and J. Ye. Efficient l1/lq norm regularization. *arXiv:1009.4766*, 2010.

[25] A. Schwaighofer, V. Tresp, and K. Yu. Learning gaussian process kernels via hierarchical bayes. In *NIPS*, pages 1209–1216, 2004.

[26] B. Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11, 2010.

[27] R. Sultana, D. Boyd-Kimball, H. F. Poon, J. Cai, W. M. Pierce, J. B. Klein, M. Merchant, W. R. Markesbery, and D. A. Butterfield. Redox proteomics identification of oxidized proteins in alzheimer's disease hippocampus and cerebellum: an approach to understand pathological and biochemical alterations in ad. *Neurobiology of aging*, 27(11):1564–1576, 2006.

[28] S. J. Teipel, W. Bayer, G. E. Alexander, Y. Zebuhr, D. Teichberg, L. Kulic, M. B. Schapiro, H.-J. Möller, S. I. Rapoport, and H. Hampel. Progression of corpus callosum atrophy in alzheimer disease. *Archives of Neurology*, 59(2):243–248, 2002.

[29] M. Ware, E. Frank, G. Holmes, M. Hall, and I. H. Witten. Interactive machine learning: letting users build classifiers. *INT J HUM-COMPUT ST*, 55(3):281–292, 2001.

[30] S. Xiong, J. Azimi, and X. Z. Fern. Active learning of constraints for semi-supervised clustering. *TKDE*, 26(1):43–54, 2014.

[31] Y. Zhang and J. G. Schneider. Learning multiple tasks with a sparse matrix-normal penalty. In *NIPS*, pages 2550–2558, 2010.

[32] Y. Zhang and D.-Y. Yeung. A convex formulation for learning task relationships in multi-task learning. *arXiv preprint arXiv:1203.3536*, 2012.

[33] Y. Zhang, D.-Y. Yeung, and Q. Xu. Probabilistic multi-task feature selection. In *NIPS*, pages 2559–2567, 2010.

[34] J. Zhou, J. Chen, and J. Ye. Clustered multi-task learning via alternating structure optimization. In *NIPS*, 2011.

[35] J. Zhou, J. Chen, and J. Ye. Malsar: Multi-task learning via structural regularization. *Arizona State University*, 2011.

[36] J. Zhou, J. Liu, V. A. Narayan, J. Ye, A. D. N. Initiative, et al. Modeling disease progression via multi-task learning. *NeuroImage*, 78:233–248, 2013.

[37] J. Zhou, L. Yuan, J. Liu, and J. Ye. A multi-task learning formulation for predicting disease progression. In *Proceedings of the 17th ACM SIGKDD*, pages 814–822. ACM, 2011.