Multi-Task Feature Interaction Learning

Kaixiang Lin[1], Jianpeng Xu[1], Inci M. Baytas[1], Shuiwang Ji[2], Jiayu Zhou[1]
[1]Computer Science and Engineering, Michigan State University, East Lansing, MI 48824
[2]Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99164
{linkaixi, xujianpe, baytasin, jiayuz}@msu.edu, sji@eecs.wsu.edu

ABSTRACT

Linear models are widely used in various data mining and machine learning algorithms. One major limitation of such models is the lack of capability to capture predictive information from interactions between features. While introducing high-order feature interaction terms can overcome this limitation, this approach dramatically increases the model complexity and imposes significant challenges in the learning against overfitting. When there are multiple related learning tasks, feature interactions from these tasks are usually related and modeling such relatedness is the key to improve their generalization. In this paper, we propose a novel Multi-Task feature Interaction Learning (MTIL) framework to exploit the task relatedness from high-order feature interactions. Specifically, we collectively represent the feature interactions from multiple tasks as a tensor, and prior knowledge of task relatedness can be incorporated into different structured regularizations on this tensor. We formulate two concrete approaches under this framework, namely the shared interaction approach and the embedded interaction approach. The former assumes tasks share the same set of interactions, and the latter assumes feature interactions from multiple tasks share a common subspace. We have provided efficient algorithms for solving the two formulations. Extensive empirical studies on both synthetic and real datasets have demonstrated the effectiveness of the proposed framework.

CCS Concepts

•**Computing methodologies** → **Multi-task learning;**
•**Information systems** → *Data mining;*

Keywords

muti-task learning; feature interaction; structured regularization; tensor norm

KDD '16, August 13-17, 2016, San Francisco, CA, USA
 ISBN 978-1-4503-4232-2/16/08…$15.00
DOI: http://dx.doi.org/10.1145/2939672.2939834

1. INTRODUCTION

Linear models are simple yet powerful machine learning and data mining models that are widely used in many applications. Due to the additive nature of the linear models, it can fully unleash the power of feature engineering, allowing crafted features to be easily integrated into the learning system. This is a desired property in many practical applications, in which high-quality features are the key to predictive performance. Moreover, efficient parallel algorithms are readily available to learn linear models from large-scale datasets. Despite its attractive properties, one apparent limitation of such models is that they can only learn a set of individual effects of features contributing to the response, due to its linear additive property. Thus when a part of the response is derived from interactions between features, such models would not be able to detect such non-linear predictive information, thereby leading to poor predictive performance.

In practice, high-order feature interactions are common in many domains. For example, in genetics studies, environmental effects and genetic-environmental interaction are found to have strong relationship with the variability in adoptee aggressivity, conduct disorder and adult antisocial behavior [7]. Similarly, the interaction effects between continuance commitment and affective commitment was found in predicting annexed absences [28]. Also, a recent study of depression found that genotype, sex, environmental risk and their interaction have combined influence on depression symptoms [12]. It is also reported that the interaction of brain-derived neurotrophic factor and early life stress exposure are identified in predicting syndromal depression and anxiety, and associated alterations in cognition [16]. In biomedical studies, many human diseases are a result of complicated interactions among genetic variants and environmental factors [19]. One intuitive solution to overcome this limitation is to augment interaction terms into linear models, explicitly modeling the effects from the interactions. However, this will dramatically increase the model complexity and lead to poor generalization performance when there is limited amount of data [9, 11, 23, 26, 35].

On the other hand, when there are multiple related learning tasks, the multi-task learning (MTL) paradigm [1, 4, 8] has offered a principled way to improve the generalization performance of such learning tasks by leveraging the relatedness among tasks and performing inductive transfer among them. The past decade has witnessed a great amount of success in applying MTL to tackle problems where large amount of labeled data are not available or creating such

datasets incurs prohibitive cost. Such problems are especially prevalent in biological and medical domains, where MTL has achieved significant success, including data analysis on genotype and gene expression [21], breast cancer diagnosis [37] and progression modeling of Alzheimer's Disease [18], etc. The MTL improves generalization performance by learning a shared representation from all tasks, which serves as the agent for knowledge transfer. Structured regularization has provided an effective means of modeling such shared representation and encoding various types of domain knowledge on tasks [1, 20, 24, 33]. The attractive benefits provided by MTL make it an ideal scheme when learning problems involve multiple related tasks with feature interactions, because tasks may be related with each other by shared structures on feature interactions. For example, predicting various cognitive functions may involve a shared set of interactions among brain regions.

However, many existing MTL frameworks are based on linear models [1] in the original input space. Thus they cannot be directly applied to explore task relatedness in the form of high-order feature interactions. On the other hand, although traditional nonlinear MTL methods based on neural networks (e.g., [2]) can exploit non-linear feature interactions to some extends, it is generally difficult to encode prior knowledge on task relatedness to such models. In this paper, we propose a novel multi-task feature interaction learning framework, which learns a set of related tasks by exploiting task relatedness in the form of shared representations in both the original input space and the interaction space among features. We study two concrete approaches under this framework, according to different prior knowledge about the relatedness via feature interactions. The *shared interaction approach* assumes that there are only a small number of interactions that are relevant to the predictions, and all tasks share the same set of interactions; the *embedded interaction approach* assumes that, for each task, the feature interactions are derived from a low-dimensional subspace that is shared across different tasks. We have provided formulations and efficient algorithms for both approaches. We conduct empirical studies on both synthetic and real datasets to demonstrate the effectiveness of the proposed framework on leveraging feature interactions from tasks. The contributions of this paper are three folds:

- Our novel framework has extended the MTL paradigm, for the first time, to allow high-order representations to be shared among tasks, by exploiting predictive information from feature interactions.
- We proposed two novel approaches under our framework to model different task relatedness over feature interactions.
- Our comprehensive empirical studies on both synthetic and real data have led to practical insights of the proposed framework.

The remainder of this paper is organized as follows: Section 2 reviews related work of MTL and models involving feature interactions. Section 3 introduces the framework for MTIL. The two approaches under MTIL have been given in 4. Section 5 presents the experimental results on both synthetic and real datasets. Section 6 concludes the paper.

2. RELATED WORK

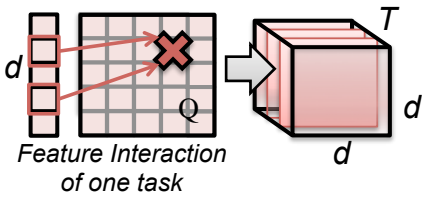
The proposed research is related to existing work on MTL and feature interaction learning. In this section, we briefly summarize the these related work and show how our work advances these areas.

2.1 Multi-Task Learning

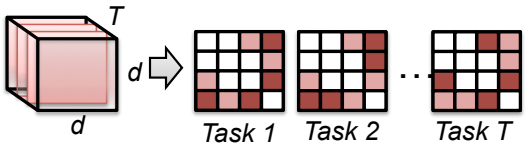
MTL has been extensive studied over the last two decades. In the center of most MTL algorithms is how task relationships are assumed and encoded into the learning formulations. The concept of learning multiple related tasks in parallel was first introduced in [8]. It was demonstrated in multiple real-world applications that adding a shared representation in neural network tasks can help others get better models. Such discovery had inspired many subsequent research efforts in the community and applications in diverse application domains. Among these studies, the regularized MTL framework has been pioneered by [13]. The regularization scheme can easily integrate various task relationship into existing learning formulations to couple MTL, thus providing a flexible multi-task extension to existing algorithms. It is well adopted and is soon generalized to a rich family of MTL algorithms.

MTL via Regularization. Among the work in the regularization based MTL scheme, there are many different assumptions about how tasks are related, leading to different regularization terms in the formulation. For example, one common assumption is that the tasks share a subset of features, and the task relatedness can be captured by imposing a group sparsity penalty on the models to achieve simultaneous feature selection across tasks [33, 24]. Another common assumption is that the models of tasks come from the same subspace, leading to a low-rank structure within the model matrix. Directly penalizing the rank function leads to NP-hard problems, and one convex alternative is to penalize the convex envelop of the rank function, i.e., trace norm. This encourages low-rank by introducing sparsity to the singular values of the model matrix [20]. In [1], the authors studied a MTL formulation that learns a common feature mapping for the tasks and assumed all tasks share the same features after the mapping. The authors have shown that this assumption can also be equivalently expressed by a low- rank regularization on the model. There are many more formulations that fall into this category of formulation to capture task relatedness by designing different shared representation and regularization terms, such as cluster structures [38], tree/graph structures [21, 10], etc. However, to the best of our knowledge, all of these formulations do not consider feature interactions in the model, and extensions to consider interactions are not straightforward. In this work, we will extend the MTL framework to enable knowledge transfer not only in the original input space, but also in higher- order feature interaction space.

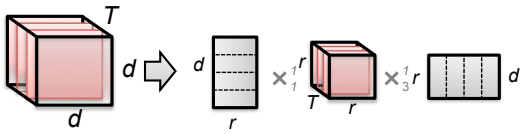
Multilinear MTL. The use of tensor in MTL has shown to be very effective in representing structural information underlying in MTL problems. In [27], Romera-Paredes *et al.* proposed a multilinear multitask (MLMTL) framework that arranges parameters of linear effects from all tasks into a tensor $\mathcal{W}$, by which they are able to represent the multi-modal relationships among tasks. In a dataset containing multi-modal relationships, tasks can be referenced by multiple indices. In MLMTL, the authors employed a regularizer on $\mathcal{W}$ to induce a low-rank structure to transfer knowledge among



a) tensor representation of feature interactions



b) structured sparsity of an interaction tensor



c) low-rank structure of an interaction tensor

Figure 1: Illustration of MTL with feature interactions. (a) the feature interactions from multiple tasks can be collectively represented as a tensor $\mathcal{Q}$; group sparse structures (c) and low-rank structures (b) in feature interactions can be used to facilitate multi-task models.

tasks. The optimization problem contains the minimization of tensor's rank, which leads to solving a non-convex problem. Thus the authors develop an alternating algorithm, employing the Tucker decomposition and convex relaxation using tensor trace norm. Although the authors also used a tensor representation in MTL, the learning formulations, implications, as well as the meaning of such the tensor is fundamentally different from those in our work. The proposed MTIL framework utilizes tensor to capture the relatedness among tasks and transfer knowledge through high-order feature interactions, which cannot be achieved by any existing MTL formulations. Note that the tensor in MLMTL is indexed by multi-modal tasks. In MTIL, the tensor is indexed by features and tasks, which is clearly different from the aforementioned work. In the proposed embedded interaction approach for MTIL, however, we face a similar challenge in MLMTL to seek a solution involving a low-rank tensor.

2.2 Feature Interaction

In many machine learning tasks, we are interested in learning a linear predictive model. Given the input feature vector of a sample, the response is given by a linear combination of these features, i.e., a weighted sum of the features. Because of this reason we call them linear effects. There are strong evidences found in many complex applications that, in addition to the linear effects, there are also effects from high-order interactions between such features. As a result, there are considerable efforts from both academia and industry aiming at addressing this limitation by removing the additive assumption and including interaction effects.

To overcome the dimensionality issues introduced by interaction effects, two types of heredity constraints have been studied [5]; namely strong hierarchy in which an interaction effect can be selected into the model only if both of its corresponding linear effects have been selected, and weak hierarchy, in which an interaction effect can be selected if at least one of its corresponding linear effects has been selected. In [11], the authors proposed an approach known as SHIM to identify the important interaction effects. SHIM extends the classical Lasso [29] and enforces a strong hierarchy. An iterative algorithm was proposed based on Lasso, which may not scale to problems with high dimensional feature space. Radchenko *et. al* proposed the VANISH method to address the problem [26]. They developed a convex formulation with a refined penalty that can not only learn the sparse solution, but also treat the linear and interaction effects using different weights. This way, the main effect could have more influence on the prediction. In [5], a hierarchical lasso was proposed to search for interactions with large main effects instead of considering all possible interactions. The authors proposed an algorithm based on ADMM for strong hierarchy lasso and a generalized gradient descent for weak hierarchical lasso. More recently, Liu *et al.* [23] proposed an efficient algorithm for solving the non-convex weak hierarchical Lasso directly, based on the framework of general iterative shrinkage and thresholding (GIST) [17]. The authors proposed a closed form solution of proximal operator and further improved the efficiency of solving the subproblem of proximal operator from quadratic to linearithmic time complexity.

In many real work applications there are multiple related tasks. When those these tasks involve interaction effects, the tasks could be related via the high order feature interactions. In our paper, we propose to address the model complexity issue from interaction effects using a new perspective, by leveraging such relatedness.

3. TASK RELATEDNESS IN HIGH ORDER FEATURE INTERACTIONS

In this section, we present the framework of Multi-Task feature Interaction Learning (MTIL). For completeness, we give a self-contained introduction of our work. We will derive concrete learning algorithms under this framework in Section 4.

Linear and Interaction Effects. Consider the traditional linear models. For an input feature vector $\mathbf{x} \in \mathbb{R}^d$ and a scalar response y, we have assumed the following underlying linear generative model:

$$y = \sum_{i=1}^{d} x_i w_i + \epsilon,$$

where $\mathbf{w} \in \mathbb{R}^d$ is the weight vector for linear effects, and $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is a Gaussian noise. A linear model $f(\mathbf{x}; \mathbf{w}) = \mathbf{x}^T\mathbf{w}$ can be a quite effective prediction function. However, if the underlying generative model includes effects from feature interactions, i.e.,

$$y = \sum_{i=1}^{d} x_i w_i + \sum_{i=1}^{d}\sum_{j=1}^{d} x_i x_j Q_{i,j} + \epsilon,$$

where $x_i x_j Q_{i,j}$ is the joint effect between the ith feature and the jth feature, and $Q_{i,j}$ is the weight for this joint effect. This type of feature interactions have been commonly found in many applications. If the training data follow this distribution then the linear model is not enough to capture the relationship between input features and output responses.

One of the approaches is to introduce non-linear feature interaction terms into the linear model. That is, we can denote it as a quadratic function:

$$f(\mathbf{x}; \mathbf{w}, \mathbf{Q}) = \mathbf{x}^T\mathbf{w} + \mathbf{x}^T\mathbf{Q}\mathbf{x}, \quad (1)$$

where $\mathbf{w} \in \mathbb{R}^d$ and $\mathbf{Q} \in \mathbb{R}^{d\times d}$ collectively represent the parameters for linear effects and interaction effects, respectively. We note that $\mathbf{Q}$ is typically symmetric because this representation includes two terms involving feature i and j: $x_ix_j(Q_{i,j} + Q_{j,i})$ and it also includes second-order feature transformations of the original features $x_i^2Q_{i,i}$.

Discussions on Feature Interactions. In supervised learning, we seek a predictive function that maps an input vector $\mathbf{x} \in \mathbb{R}^d$ to a corresponding output $y \in \mathbb{R}$. Let $(\mathbf{X}, \mathbf{y}) = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ...(\mathbf{x}_n, y_n)\}$ be a training dataset, in which each data point is drawn from certain *i.i.d.* distribution μ. The goal of learning is to find the best predictor $\hat{f} \in \mathcal{H}$ so that the predicted value $\hat{\mathbf{y}}_i$ for the input data $\mathbf{x}_i$ is as close as possible to the ground truth $\mathbf{y}_i, \forall(\mathbf{x}_i, y_i) \in (\mathbf{X}, \mathbf{y})$, given a loss function $L(.,.)$. We hope that the predictor f learned in this way is close to the optimal model that minimizes the expected loss according to the μ:

$$R(f) = \mathbb{E}_{(\mathbf{X},\mathbf{y})\sim\mu}L(f(\mathbf{X}), \mathbf{y}). \quad (2)$$

Such predictor is given by the minimum of the empirical risk:

$$\hat{f} = \arg\min_{f\in\mathcal{H}} \sum_{i=1}^{n} L(f(\mathbf{x}_i), \mathbf{y}_i).$$

The error caused by learning the best predictor in the training dataset is called the estimation error. The error caused by using a restricted $\mathcal{H}$ is called the approximation error. For a fixed data size, the smaller the hypothesis space $\mathcal{H}$, the larger the approximation error, and vice versa. The trade-off between approximation error and estimation error is controlled by selecting the size of $\mathcal{H}$. By including feature interactions we would enlarge the hypothesis space, and we may be able to dramatically minimize the approximation error compared to the traditional hypothesis space for linear models. On the other hand, we note that given a limited amount of data, a large hypothesis space may result in models with poor generalization performance. We will need to either increase our training data, or provide effective regularizations to narrow down the hypothesis space.

Multi-task Feature Interactions. We consider the setting that there are multiple learning tasks which are related not only in the original feature space, but also in terms of feature interactions. The propose framework simultaneously learns all related tasks and provides an effective regularization on the hypothesis space using relatedness on the interactions.

Let $\mathcal{D} = (\mathbf{X}_1, \mathbf{y}_1), \ldots, (\mathbf{X}_T, \mathbf{y}_T)$ be the training data for the T learning tasks, and the *i.i.d.* training samples for task t is drawn from $(\mu_t)^{m_t}$, where m_t is the number of data points available for task t. We collectively denote the distribution as $\mathcal{D} \sim \boldsymbol{\mu} = \prod_{t=1}^{T}(\mu_t)^{m_t}$. All tasks have a d-dimensional feature space (i.e., $\mathbf{x}_i \in \mathbb{R}^d$). The corresponding features are homogeneous and have the same semantic meaning. The total training data points are:

$$(\mathbf{X}_t, \mathbf{y}_t) = \{(\mathbf{x}_{1t}, y_{1t}), (\mathbf{x}_{2t}, y_{2t}), \ldots, (\mathbf{x}_{mt}, y_{mt})\}, t = 1, \ldots, T,$$

The goal of MTL is to learn T functions for the tasks such that $f_t(\mathbf{x}_{it}) = y_{it}$, based on the assumption that all task functions are related to some extent.

In order to consider interactions for each task, we use the quadratic predictive function in Eq. 1 for all tasks. We collectively represent the linear effects from all tasks as a matrix $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_T] \in \mathbb{R}^{d\times T}$, $\mathbf{w}_i \in \mathbb{R}^d$ and the interaction effects as a tensor $\mathcal{Q} \in \mathbb{R}^{d\times d\times T}$, in which the t-th frontal slice $\mathbf{Q}_t \in \mathbb{R}^{d\times d}$ represents the interaction effects for task t. We illustrate this interaction tensor in Figure 1(a).

Given specific loss functions $\hat{\ell}$ for samples from one task, (e.g., square loss for regression and logistic loss for classification, see Table 1), the loss function for each task is $\ell_t(f, \mathbf{w}, \mathbf{Q}; \mathbf{X}, \mathbf{y}) = \sum_{i=1}^{m_t} \hat{\ell}(f(\mathbf{x}_i; \mathbf{w}, \mathbf{Q}), y_i)$. Our multi-task feature interaction loss function is given by:

$$L(\mathbf{W}, \mathcal{Q}; f, \mathbf{X}, \mathbf{Y}) = \sum_{t=1}^{T} \ell_t(f, \mathbf{w}_t, \mathcal{Q}_t; \mathbf{X}_t, \mathbf{Y}_t). \quad (3)$$

Note that it is not necessary for all tasks to have the same loss function. In MTL, the learning of each task benefits from the knowledge from other tasks, which effectively reduces the hypothesis space for all tasks. In order to achieve knowledge transfer among tasks, we would like to impose shared representations via designing regularization terms on both $\mathbf{W}$ and $\mathcal{Q}$, which specify how tasks are related in the original feature space and features interactions, respectively.

The MTIL Framework. The proposed Multi-Task feature Interaction Learning (MTIL) framework is then given by the following learning objective:

$$\min_{\mathbf{W},\mathcal{Q}} L(\mathbf{W}, \mathcal{Q}; f, \mathbf{X}, \mathbf{Y}) + \lambda_R R_F(\mathbf{W}) + \lambda_I R_I(\mathcal{Q}), \quad (4)$$

where $R_F(\mathbf{W})$ is the regularization providing task relatedness in the original feature space, $R_I(\mathcal{Q})$ is the regularization encoding our knowledge about how feature interactions are related among tasks, λ_R and λ_I are the corresponding regularization coefficients. For $\lambda_I \to \infty$, the problem reduces to traditional MTL, when R_I is chosen properly. In this paper, we formulate two concrete approaches to capture the feature interaction patterns:

- **Shared Interaction Approach.** In many applications, even though we have a large number of feature interactions, only a few interactions may be related to the response [5, 11]. When learning with multiple tasks, different tasks may share exactly the same set of feature interactions, but with different effects. As such, we can design MTIL formulations that learns a set of common feature interactions, which could effectively reduce the hypothesis space. During the learning process the selected feature interactions for one task will be task's knowledge, contributing to the share representation: a set of indices of common interactions. An analogy in traditional MTL is the joint feature learning approach [24, 33], in which tasks share the same set of features. One way to achieve this approach is by using the structured sparsity to induce the same sparsity patterns on the interaction effects. An illustration of this approach is given in Figure 1(b).

- **Embedded Interaction Approach.** When the response from one task is related to complicated feature interactions, the patterns of such interactions may be captured by a low-dimensional space, resulting in a

low-rank interaction matrix. When there are multiple related tasks, they could have a shared low-dimensional space, i.e., different interaction matrices may share the same set of rank-1 basis matrices, but have different weights associated with these basis matrices. When collectively represented by a tensor, we end up with a low-rank tensor. During the learning process, each task contributes their subspace information to facilitate learning of the share low-dimensional subspace, which in turn, improves the feature space. The analogy in traditional MTL is the low-rank based models [1, 20]. However, there are challenging questions such as: How to define a proper rank function for tensor? Are there tractable algorithms to induce low-rank structure in tensor? In the next section we will discuss these important questions and propose efficient algorithms. We illustrate this approach in Figure 1(c).

We note that even though we only provided two specific approaches in this paper, the proposed MTIL framework could offer broader class of formulations. The proposed framework allows many other possible ways to define task relatedness on feature interactions, leading to a brand-new research area of MTL.

4. FORMULATIONS AND ALGORITHGMS OF THE TWO MTIL APPROACHES

In this section, we will study how the formulations and algorithms of the shared interaction approach and embedded interaction approach under the proposed MITL framework. We note that extension of multi-task learning to feature interactions is not trivial because of the involvement of tensors. We start with formulating the shared interaction approach by incorporating a group Lasso penalty to introduce structured sparsity on the tensor, which would select only a set of common feature interactions across different tasks that are relevant to the prediction. For the embedded interaction approach, we propose both a convex formulation and a non-convex formulation. While the convex formulation leads to efficient optimization algorithms and global solutions, the non-convex formulation provides reduced storage complexity for large-scale problems.

4.1 Preliminary

In this paper, we use the following basic definition of tensor:
Mode-n fiber is a vector defined by fixing every index but one. We may see it as the higher order analogue of matrix rows (mode-2 fibers) and columns (mode-1 fibers). For example, in a three-way tensor $\mathcal{Q} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, the mode-3 fiber is $\mathcal{Q}_{i,j,:} \in \mathbb{R}^{n_3}$.
Mode-n unfolding is the process of reordering the elements of an N-way tensor $\mathcal{Q} \in \mathbb{R}^{n_1 \times n_2 \times ,..,\times n_N}$ into a matrix. The mode-k unfolding of tensor $\mathcal{Q}$ is denoted by $\mathcal{Q}_{(k)} \in \mathbb{R}^{n_k \times J_k}$, where $J_k = \prod_{i=1, i\neq k}^{N}$. The matrix is arranged by concatenating all mode-k fibers of the tensor.
Rank-n in our paper denotes the rank of tensor's mode-n unfolding. It's actually the dimension of the space spanned by the mode-n fibers of tensor. Specifically, $\text{rank}_n(\mathcal{Q}) = \text{rank}(\mathcal{Q}_{(n)})$. When $\mathcal{Q}$ is a matrix (i.e. 2-way tensor), this becomes the regular definition of rank, since $\text{rank}_1(\mathcal{Q}) = \text{rank}_2(\mathcal{Q}) = \text{rank}(\mathcal{Q})$.

4.2 Shared Interaction Approach

The goal of the shared interaction approach is to identify a set of common and relevant feature interactions across different tasks. The interaction tensor $\mathcal{Q}$ in our framework has provided a convenient representation to encode such information, and we are able to incorporating a group Lasso penalty [14] to induce a special type of structured sparsity on the tensor, coupling the same interactions for all tasks. Recall that the sparsity implies that only the significant interaction effects are captured in the model. For the purpose of shared interaction, a *sparse tensor norm* is defined as:

$$||\mathcal{Q}||_{\text{GL-Sym}} \equiv \sum\nolimits_{i=1}^{d} \sum\nolimits_{j\geq i}^{d} \sqrt{\sum\nolimits_{k=1}^{K} \left(Q_{i,j,k}^2 + Q_{j,i,k}^2\right)}. \quad (5)$$

Note that this norm enforces a symmetric sparsity by over the tensor, so that the one group is defined to include coefficients of one interaction between feature i and feature j, from all tasks. Penalizing the tensor sparse norm leads to the following formulation:

$$\min_{\mathbf{w},\mathcal{Q}} L(\mathbf{W}, \mathcal{Q}; f, \mathbf{X}, \mathbf{Y}) + \lambda_F R_F(\mathbf{W}) + \lambda_I ||\mathcal{Q}||_{\text{GL-Sym}}, \quad (6)$$

where the parameter λ_I control the sparsity of tensor $\mathcal{Q}$, a larger λ_I will end up with a more sparse $\mathcal{Q}$. The solution to formulation delivers a tensor such that the mode-3 fibers are either all zeros vectors or non zero vectors, i.e., interaction effects between 2 features x_i, x_j either exists on all tasks, or irrelevant for all tasks. Note that even the sparsity patterns is same for all tasks, their interactions may have different weights. It is easy to see that, this approach subsumes the traditional multi- task learning as a special case: when $\lambda_I \to \infty$ by setting regularization parameter on tensor $\mathcal{Q}$ to infinity, all the elements in of $\mathcal{Q}$ in the solution will be zeros, and the model only considers linear effects.

When the loss function L chosen is convex and continuously differentiable with Lipschitz continuous gradient [26], then we can use proximal based gradient methods, such as first order FISTA [3], SpaRSA [34] or second order Proximal Newton [22] to solve it efficiently. Because that the linear effects and interaction effects are decoupled in the predictive function, a major class of loss functions belong to this category, and we give a few examples of common loss functions in Table 1. Note that even when L is non-convex, a local optimal solution can be efficiently obtained using the GIST framework [17]. The key to apply these algorithms is to efficiently compute the proximal operator that associates to the problem (refer to [25] for more details about proximal):

$$\min_{\mathbf{W},\mathcal{Q}} \frac{1}{2}(\|\mathbf{W} - \hat{\mathbf{W}}\|_F^2 + \|\mathcal{Q} - \hat{\mathcal{Q}}\|_F^2) + \rho_1 R_F(\mathbf{W}) + \rho_2 ||\mathcal{Q}||_{\text{GL-Sym}},$$

where $\hat{\mathbf{W}}$ and $\hat{\mathcal{Q}}$ are intermediate solutions at each step, ρ_1 and ρ_2 are regularization parameters augmented with step size. Note that we have extend the Forbenius norm from matrix to tensor. We see that the problem is decoupled for $\mathbf{W}$ and $\mathcal{Q}$. And the tensor proximal:

$$\min_{\mathcal{Q}} \frac{1}{2}\|\mathcal{Q} - \hat{\mathcal{Q}}\|_F^2 + \rho_2 ||\mathcal{Q}||_{\text{GL-Sym}},$$

can be solved in the same way as the group Lasso proximal operator [36]. Moreover, we find that when the gradient is symmetric, we don't need to enforce a symmetric tensor

Table 1: Examples of three common smooth loss functions and their gradients with the interaction augmented predictive function given in Eq. (1).

Loss with Interaction	Loss function L_i	Gradient | Linear Eff. $\nabla_{\mathbf{W}} L_i$	Gradient | Interaction Eff. $\nabla_{\mathbf{Q}_t} L_i$
Logistic Loss*	$-[log(g(\mathbf{x}_i))y_{ti} + (1-y_{ti})(log(1-g(\mathbf{x}_i)))]$	$(g(\mathbf{x}_i) - y_{ti})\mathbf{x}_i$	$(g(\mathbf{x}_i) - y_{ti})\mathbf{x}_i\mathbf{x}_i^T$
Squared Loss	$\frac{1}{2}\|\|\mathbf{x}_i^T\mathbf{w}_t + \mathbf{x}_i^T\mathbf{Q}_t\mathbf{x}_i - y_{ti}\|\|_2^2$	$\mathbf{x}_i(\mathbf{x}_i^T\mathbf{w}_t + \mathbf{x}_i^T\mathbf{Q}_t\mathbf{x}_i - y_{ti})$	$\mathbf{x}_i(\mathbf{x}_i^T\mathbf{w}_t + \mathbf{x}_i^T\mathbf{Q}_t\mathbf{x}_i - y_{ti})\mathbf{x}_i^T$
Squared Hinge†	$h(y_{ti}(\mathbf{x}_i^T\mathbf{w}_t + \mathbf{x}_i^T\mathbf{Q}_t\mathbf{x}_i))$	$y_{ti}\mathbf{x}_i h'(\mathbf{x}_i^T\mathbf{w}_t + \mathbf{x}_i^T\mathbf{Q}_t\mathbf{x}_i)$	$y_{ti}\mathbf{x}_i\mathbf{x}_i^T h'(\mathbf{x}_i^T\mathbf{w}_t + \mathbf{x}_i^T\mathbf{Q}_t\mathbf{x}_i)$

*$g(\mathbf{x})$ is the sigmoid function defined as $g(\mathbf{x}_i) = 1/\left\{1 + exp(-(\mathbf{x}_i^T\mathbf{w}_t + \mathbf{x}_i^T\mathbf{Q}_t\mathbf{x}_i))\right\}$

†$h'(z) = \{-1 \quad \text{for } z \le 0, \quad z-1 \text{ for } 0 < z < 1, \quad 0 \text{ for } z \ge 1\}$

sparse norm, and we could simply use a simple alternative:

$$||\mathcal{Q}||_{GL} = \sum\nolimits_{i,j} \sqrt{\sum\nolimits_{k=1}^{K} \mathcal{Q}_{i,j,k}^2},$$

and initialize the algorithm with a symmetric tensor as the starting point. The reason that symmetry holds can be explained by two parts. First, the gradient of $\mathcal{Q}$ is symmetric, therefore the gradient descent step won't change the symmetry of tensor $\mathcal{Q}$. Second, the proximal operator associated to sparse tensor norm won't change the symmetry of matrix. To see this, the proximal operation is performed by vectorizing the matrix into a vector and shrink each element of the vector with respect to a input vector, which is obtained by the last gradient descent step. Since the input vector represents an symmetric matrix, the element and its symmetric element will always shrink to the same new value. Therefore, the symmetry of $\mathcal{Q}$ holds. The sparse tensor norm is equivalent to perform the l_1 projection of vectors where each element is the l_2 norm of mode-3 fiber in tensor $\mathcal{Q}$.

4.3 Embedded Interaction Approach

The share interaction approach has enforced a very restrictive form of how tasks are supposed to relate to each other. In many applications, the prediction may be a result of complicated feature interactions, instead only involves a few interactions. Even though the prediction may involve all feature interactions, it is usually a reasonable assumption that there are patterns among these interactions. Numerically, existence of patterns imply a low-dimensional subspace, which is reflected by a low-rank structure in the matrix. When there are multiple related learning tasks, one way for these tasks relate to others via a shared low-dimensional subspace, which gives us a low-rank tensor. As such, we may design a structured regularization to encourage the matrix $\mathcal{Q}$ to be a low-rank tensor. In this paper we describe one convex formulation that encourages low-rank structure by penalizing a tensor norm and one non-convex formulation that directly learns a low-rank representation.

4.3.1 Convex Formulation

One way to obtain a low-rank tensor is to augment our formulation with a rank penalty. One problem associates to tensor is that there is no consistent way to define the rank of a tensor. One way is to use the average rank of unfolding on different mode [15]:

$$\frac{1}{N}\sum_{n=1}^{N} \text{rank}_n(\mathcal{Q}) = \frac{1}{N}\sum_{n=1}^{N} \text{rank}(\mathcal{Q}_{(n)}),$$

where N is the total number of mode of the tensor ($N = 3$ when only pair-wise interactions), and $\mathcal{Q}_{(n)}$ is unfold on n mode. Since minimizing the rank function is proven to be NP-hard, we could penalize the trace norm instead, which is the convex envelope of the rank function. The trace norm is defined as the sum of singular values of the matrix variable [20]. We then obtain the following convex formulation:

$$\min_{\mathbf{W},\mathbf{Q}} L(\mathbf{W}, \mathcal{Q}; f, \mathbf{X}, \mathbf{Y}) + \lambda_R R_1(\mathbf{W}) + \frac{\lambda_I}{N}\sum_{n=1}^{3} ||\mathcal{Q}_{(n)}||_*, \quad (7)$$

where $\|.\|_*$ denotes the trace norm. However, this convex formulation penalizes every mode of tensor $\mathcal{Q}$ to be jointly low rank, which may be too restricted in practice, which may lead to suboptimal performance. Moreover, the practical way to solve the formulation in Eq. (7) is to use the alternating direction methods of multipliers (ADMM) [6], which introduces auxiliary variables and equality constraints, in order to decouple the three tensor trace norm terms. However, ADMM algorithm in practice is shown to have a slow convergence rate, and less preferred when composite proximal methods such as FISTA can be applied.

One alternative way to address these issues is to use the latent trace norm [30, 31], which is defined as following for a $N-$way tensor:

$$||\mathbf{Q}||_{\text{latent}} = \inf_{\mathcal{Q}^{(1)}+\mathcal{Q}^{(2)}+\ldots+\mathcal{Q}^{(N)}=\mathcal{Q}} \sum_{n=1}^{N} ||\mathcal{Q}_{(n)}^{(n)}||_*,$$

where $\mathcal{Q}^{(1)} \ldots \mathcal{Q}^{(N)}$ are a set of low-rank auxiliary tensors, which states that the original tensor can be decomposed into the sum of a set of tensors that are low-rank in different modes. Finally, we proposed to drop the equality constraint that each auxiliary tensor equal to the original one, but we directly use the mixture of tensors to represent the original tensor, so the problem becomes a unconstrained optimization problem. The predictive function of task t with such mixture is given by:

$$f_{\text{mix}}(\mathbf{x}; \mathbf{w}_t, \{\mathcal{Q}^{(i)}\}_{i=1}^3) = \mathbf{x}^T\mathbf{w}_t + \mathbf{x}^T(\sum\nolimits_{i=1}^{3} \mathcal{Q}_t^{(i)})\mathbf{x},$$

where $\mathcal{Q}^{(j)} \in \mathbb{R}^{d\times d\times K}$, $\forall j = 1, 2, 3$ are the auxiliary tensors for replacing the original tensor $\mathcal{Q}$, matrix $\mathcal{Q}_{(j)}^{(j)} \in \mathbb{R}^{(n_1 n_2 n_3/n_j)\times n_j}$ is the mode j unfolding of tensor $\mathcal{Q}^{(j)}$, $\mathcal{Q}_t^{(j)} \in \mathbb{R}^{d\times d}$ is the tth frontal slice of tensor $\mathcal{Q}^{(j)}$. Finally, our convex formulation under embedded interaction approach is given by:

$$\min_{\mathbf{W},\{\mathcal{Q}^{(i)}\}_{i=1}^3} L(\mathbf{W}, \{\mathcal{Q}^{(i)}\}_{i=1}^3; f_{\text{mix}}, \mathbf{X}, \mathbf{Y}) + \lambda_F R_F(\mathbf{W}) + \lambda_I \sum_{j=1}^{3} ||\mathcal{Q}_{(j)}^{(j)}||_*.$$

The convexity of this formulation holds since both the loss function and the penalty are convex. We note that this formulation can be solved in the same way as the formulation in Eq. (7), and the model is much more flexible to model the complicated interactions among the features, leveraging the advantages of such auxiliary tensors.

4.3.2 Non-Convex Formulation

Although using proximal gradient methods we are able to secure an optimal solution for the convex formulation, the time complexity and storage cost are unacceptable in practice as the dimension of data increase. To see this, we note that the proximal operator associated to a trace norm regularized objective requires singular projections [20], which requires cubic-complexity singular value decomposition. Recall in each iteration of the gradient methods could involve more than one computation of proximal operator [3], and thus the computation may be prohibitive when dimension grows larger. On the other hand, we have to maintain 3 dense tensors of size $d \times d \times T$ which means the storage cost is at $O(d^2)$, where T is the number of tasks and typically we have $T \ll d$. Also the mixture of three low-rank auxiliary tensors may lead to some difficulty when it comes to analyzing the predictive model itself.

To this end, we propose to use a tensor with a explicit low-rank structure. Consider the interaction effects matrix $\mathbf{Q} \in \mathbb{R}^{d\times d}$ for one task, we assume the low-rank decomposition $\mathbf{Q} = \mathbf{B}\tilde{\mathbf{Q}}\mathbf{B}^T$, where $\mathbf{B} \in \mathbb{R}^{d\times r}$ is a basis matrix, $\tilde{\mathbf{Q}} \in \mathbb{R}^{r\times r}$ is a small matrix, capturing the information of the original tensor under the set of bases (columns) in $\mathbf{B}$. To see this, we can expand $\mathbf{Q} = \sum_{i,j=1}^{r} \tilde{\mathbf{Q}}_{(i,j)}\mathbf{B}_i\mathbf{B}_j^T$, meaning the matrix $\mathbf{Q}$ is a result of interactions among bases in $\mathbf{B}$ and also spanned by the columns of $\mathbf{B}$. We thus can use a predictive function that explicitly considers this low-rank structure:

$$f_{\text{nvc}}(\mathbf{x}; \mathbf{w}, \mathbf{B}, \tilde{\mathbf{Q}}) = \mathbf{x}^T\mathbf{w} + \mathbf{x}^T\mathbf{B}\tilde{\mathbf{Q}}\mathbf{B}^T\mathbf{x}.$$

When there are multiple tasks, our assumption for embedded interaction approach is the shared basis, meaning $\mathbf{B}$ is restricted to be same as all other tasks. The multi-task loss function is thus given by:

$$L(\mathbf{W}, \{\mathbf{B}\}, \tilde{\mathcal{Q}}; f_{\text{nvc}}, \mathbf{X}, \mathbf{Y}) = \sum_{t=1}^{T} \ell_t(f_{\text{nvc}}, \mathbf{w}_t, \mathbf{B}, \tilde{\mathcal{Q}}_t; \mathbf{X}_t, \mathbf{Y}_t),$$

where $\tilde{\mathcal{Q}} \in \mathbb{R}^{r\times r\times T}$ collective denotes the set of matrices $\tilde{\mathcal{Q}}$ from all tasks. This loss function is not convex because of the multiplication of variables in $\mathbf{x}^T\mathbf{B}\tilde{\mathbf{Q}}\mathbf{B}^T\mathbf{x}$. This loss function leads to our final non-convex formulation for embedded:

$$\min_{\mathbf{W},\{\mathbf{B}\},\tilde{\mathcal{Q}}} L(\mathbf{W},\{\mathbf{B}\},\tilde{\mathcal{Q}}; f_{\text{nvc}}, \mathbf{X}, \mathbf{Y}) + \lambda_F R_F(\mathbf{W}) + \lambda_I R_I(\{\mathbf{B}\}, \tilde{\mathcal{Q}}),$$

where the regularization $R_I(\{\mathbf{B}\}, \tilde{\mathcal{Q}})$ can be Forbenius norm or other structural information (e.g. ℓ_1 norm). The dimension r of $\mathbf{B}$ can be chosen according to the need of specific application demands, and can be selected by cross-validation. In general, we choose $r \ll d$. We note that the storage complexity for the feature interaction effects (e.g., tensor $\mathcal{Q}$) is reduce from $O(d^2K)$ to $O(dr + r^2K)$, which is dramatically smaller than the full tensor, especially in the high dimensional settings. We could use the family of block coordinate descent algorithms [32] to alternatively solve the variables $\mathbf{W}, \{\mathbf{B}\}$, and $\tilde{\mathcal{Q}}$, to get a local optimal solution.

5. EXPERIMENTS

In this section, we perform experiments on both synthetic datasets and two real world datasets to evaluate the effectiveness of our proposed MTIL framework.

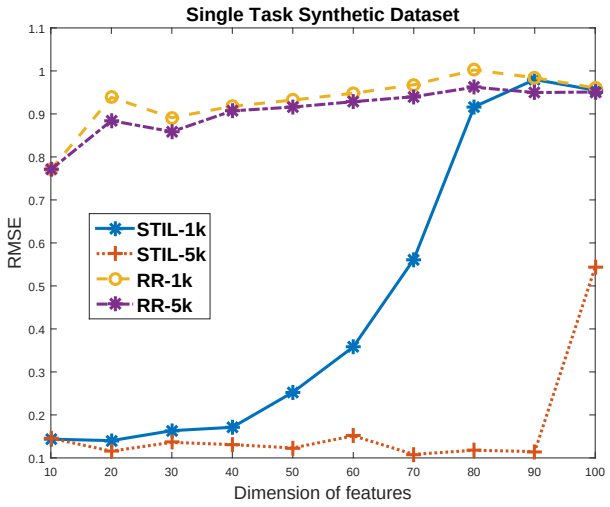


Figure 2: RMSE comparison between RR and STIL on two synthetic datasets with sample size of 1k and 5k, respectively.

5.1 Synthetic Dataset

In order to justify the effectiveness of modeling the feature interactions and MTIL framework, we test our methods on synthetic datasets.

5.1.1 Effectiveness of modeling feature interactions

In this subsection, we test whether the interactions between features can be properly handled by adding the interaction term $\mathbf{Q}$. To do so, we create a single task synthetic dataset by assuming:

$$\mathbf{y} = \mathbf{X}\mathbf{w} + \text{diag}(\mathbf{X}\mathbf{Q}\mathbf{X}') + \boldsymbol{\epsilon}, \tag{8}$$

where $\mathbf{X} \in \mathbb{R}^{n\times d}$ is the feature matrix, $\mathbf{y} \in \mathbb{R}^{n\times 1}$ is the responses, $\mathbf{w} \in \mathbb{R}^{d\times 1}$ is the weight vector, $\mathbf{Q} \in \mathbb{R}^{d\times d}$ is a symmetric, low-rank sparse matrix, which represents the feature interactions in the dataset, and $\boldsymbol{\epsilon} \sim \mathcal{N}(0, 0.01\mathbf{I}_n)$ is the additive noise term. We generate 20 synthetic datasets with different sizes (1000 or 1k and 5000 or 5k) and different feature dimensions (varying from 10 to 100, stepped by 10) by randomly selecting $\mathbf{X}$, $\mathbf{w}$, and $\mathbf{Q}$ and computing $\mathbf{y}$ according to Eq.(8).

We use single task feature interaction learning model (STIL) to evaluate the effectiveness of the interaction term $\mathbf{Q}$:

$$\min_{\mathbf{w},\mathbf{Q}} \sum_{i=1}^{n} \frac{1}{2}||\mathbf{x}_i^T\mathbf{w} + \mathbf{x}_i^T\mathbf{Q}\mathbf{x}_i - y_i||_2^2 + \frac{\lambda}{2}||\mathbf{w}||_2^2 + \mu||\mathbf{Q}||_{1,1},$$

where $\mathbf{w} \in \mathbb{R}^{d\times 1}$ is the weight vector, $\mathbf{Q} \in \mathbb{R}^{d\times d}$ is the feature interaction matrix, and $\|\mathbf{Q}\|_{1,1} = \sum_i \sum_j |\mathbf{Q}_{i,j}|$ denotes the $\ell_{1,1}$ norm.

We compared the Root Mean Square Error (RMSE) between the Ridge Regression(RR) and STIL on both of the synthetic datasets. As the results show in Figure 2, STIL outperforms RR on both of the datasets, which shows the effectiveness of modeling the feature interaction in the data. Besides, STIL-5k (RR-5k) performs better than STIL-1k (RR-1k), which demonstrates that the learning models will capture the underlining models of the data better with larger training size. Also note that with the number of dimensions increases, STIL will gradually overfit the data, because of the dramatic increase of the interactions between features.

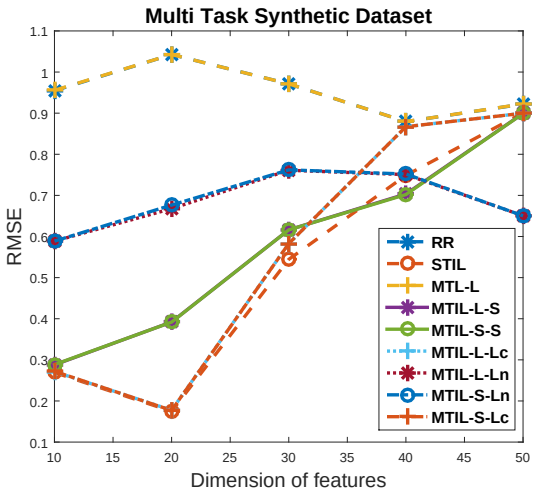


Figure 3: Synthetic dataset (Multi-task): Root Mean Square Error (RMSE) comparisons among all the methods. The Y-axis is RMSE, X-axis is dimension of features.

5.1.2 Effectiveness of MTIL

In order to test the effectiveness of MTIL, we generate a multi-task synthetic data by assuming:

$$\mathbf{y}_t = \mathbf{X}_t\mathbf{w}_t + \text{diag}(\mathbf{X}_t\mathbf{Q}_t\mathbf{X}_t^T),\ t = 1, 2, 3, .., T,$$

where $\mathbf{X}_t \in \mathbb{R}^{n\times d}$ is the feature matrix of task t, $\mathbf{y}_t \in \mathbb{R}^{n\times 1}$ is the responses of task t, $\mathbf{W} \in \mathbb{R}^{d\times T} = [\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3, ..., \mathbf{w}_T]$ is the weights for tasks. As described in Section 4.3, we generate feature interaction matrix $\mathbf{Q}_t = \mathbf{B}\mathbf{q}_t\mathbf{B}^T$ and project it into a sparse, symmetric space.

In this experiment, we generate 5 datasets with different feature dimensions from 10 to 50, stepped by 10, by randomly selecting $\mathbf{X}_t$, $\mathbf{w}_t$, $\mathbf{B}$ and $\mathbf{q}_t$.

The predictive performance of the methods outlined below are examined on the synthetic multi-task datasets:

- Ridge Regression (RR): We choose this model as the baseline and make neither assumptions of feature interaction nor the relation among all the tasks.
- STIL: We perform STIL on each of the task independently.
- MTL-L: This approach refers to the traditional MTL method regularized by the trace norm of the weight matrix $\mathbf{W}$[1]. It does not make assumptions on feature interactions.
- MTIL-L-S: This approach, refers to multi-task feature interaction learning regularized by the trace norm of the weight matrix $\mathbf{W}$ and the tensor group lasso norm of tensor $\mathcal{Q}$ (see section 4.2).
- MTIL-S-S: This approach is similar to MTIL-L-S except that the regularization term on $\mathbf{W}$ is $\ell_{2,1}$ norm.
- MTIL-L-Lc: This approach refers to multi-task feature interaction learning regularized by the trace norm of the weight matrix $\mathbf{W}$ and latent trace norm of tensor $\mathcal{Q}$ (see section 4.3).
- MTIL-S-Lc: This approach is similar to MTIL-L-Lc except for that the regularization term on $\mathbf{W}$ is $\ell_{2,1}$ norm.
- MTIL-L-Ln: This approach refer to multi-task feature interaction learning regularized by the low rank norm of tensor $\mathcal{Q}$ and the trace norm of the weight matrix $\mathbf{W}$ (see section 4.3.2).
- MTIL-S-Ln: This approach is similar to MTIL-L-Ln except for that the regularization term on $\mathbf{W}$ is $\ell_{2,1}$ norm.

Figure 3 compares the RMSE of the above methods on the 5 synthetic datasets. We can see that MTIL-L-Ln and MTIL-S-Ln are not that sensitive to the change of feature dimensions, thanks to the low-rank assumption on the feature interaction. Also, RR and MTL-L share a similar performance, which is consistent with the fact that we did not assume any low-rank structure in this synthetic dataset. Note that although STIL performs almost the best on low dimensional data, its performance deteriorates rapidly compared with other MTIL methods, due to the incapability of learning the feature interactions across tasks.

5.2 School Dataset

This dataset contains the examination records of 15362 students with 28 features from 139 schools in years of 1985, 1986 and 1987, provided by the Inner London Education Authority(ILEA). In this dataset, each task is to predict exam scores for students in one out of the 139 schools. We perform 4 sets of experiments by varying the amount of training size, from 20% to 50% of the total sample size. We test the approaches summarized in section 5.1.2 and tune the parameters on λ_R in set $[10^{-1}, 10^0, ..., 10^9, 10^{10}]$. For MTIL-L-Ln and MTIL-S-Ln methods, the rank of matrix r for each task are tuned in $[2, 3, ..., 19, 20]$. For MTIL-L-S and MTIL-L-Lc, we tune the regularization parameters λ_I in $[10^{-1}, 10^0, ..., 10^9, 10^{10}]$.

The experimental results are shown in Table 2. First, for most of the methods, RMSE will decrease when the training size increases. This means that providing more data in the training set will help overcome the overfitting problem. Also, we found that the performance of embedded feature approaches (i.e. MTIL-L-Lc, MTIL-L-Ln, MTIL-S-Ln) are worse than the single task learning approach. The reason behind this is that embedded feature approaches do not have sparse constraints on the interaction term, which will severely overfit the data when there is not sufficient training samples. Additionally, the MTL-L and MTIL-L-S obtain better performance than single task learning, which indicates that the low-rank structure shared by tasks are effectively captured by the low-rank assumption in these two methods. Moreover, MTIL-L-S method outperforms all other methods, which empirically proves the effectiveness of learning the shared interactions with sparse constraints.

5.3 Modeling Alzheimer's Disease

The Alzheimer's Disease Neuroimaging Initiative (ADNI) database(adni.loni.ucla.edu), which was launched in 2003 as a 5-year public-private partnership, is aimed to test whether the positron emission tomography (PET), serial magnetic resonance imaging (MRI), other biological markers, and clinical and neuropsychological assessments can be combined to measure the progression of mild cognitive impairment (MCI) and early Alzheimer's disease (AD). We follow the procedure of preprocessing mentioned in [39] and obtain 648 subjects and 305 MRI features. The parameters are tuned in the same way as we described in 5.2.

Table 2: Performance comparison of competing methods on the School dataset in terms of RMSE. The MTIL-L-S method consistently outperforms all other methods, showing the effectiveness of the shared interactions.

	Training 20%	Training 30%	Training 40%	Training 50%
RR	0.9149 ± 0.0031	0.9025 ± 0.0058	0.8885 ± 0.0067	0.8722 ± 0.0059
STIL	0.9149 ± 0.0031	0.9025 ± 0.0057	0.8885 ± 0.0067	0.8721 ± 0.0058
MTL-L	0.8998 ± 0.0044	0.8807 ± 0.0052	0.8657 ± 0.0032	0.8503 ± 0.0070
MTIL-L-S	**0.8623** ± 0.0048	**0.8506**± 0.0038	**0.8511**±0.0043	**0.8404** ± 0.0067
MTIL-S-S	0.8999 ± 0.0063	0.8907 ± 0.0049	0.8832 ± 0.0077	0.8686 ± 0.0046
MTIL-L-Lc	0.9252 ± 0.0090	0.8893 ± 0.0037	0.8859 ± 0.0037	0.8720 ± 0.0044
MTIL-S-Lc	0.9353 ± 0.0133	0.9139 ± 0.0053	0.8941 ± 0.0024	0.8761 ± 0.0062
MTIL-L-Ln	1.0084 ± 0.0180	0.9758 ± 0.0097	0.9328 ± 0.0267	0.9041 ± 0.0140
MTIL-S-Ln	1.0026 ± 0.0368	0.9585 ± 0.0059	0.9297 ± 0.0253	0.8965 ± 0.0066

Table 3: Performance comparison of different methods on the ADNI dataset in terms of RMSE. All of the MTLs outperform the single task learning approaches (RR and STIL) and MTIL-S-Lc method outperforms all other methods, which demonstrates the effectiveness of embedded feature interactions.

	RMSE ± standard deviation
RR	0.9418 ± 0.0023
STIL	0.9417 ± 0.0021
MTL-L	0.9031 ± 0.0007
MTIL-L-S	0.9030 ± 0.0007
MTIL-S-S	0.9162 ± 0.0017
MTIL-L-Lc	0.8941 ± 0.0050
MTIL-S-Lc	**0.8909** ± 0.0059
MTIL-L-Ln	0.8926 ± 0.0009
MTIL-S-Ln	0.9085 ± 0.0028

The RMSE comparison result is shown in Table 3. First, we found that all of the MTLs outperform the single task learning approaches (RR and STIL), which demonstrates the effectiveness of learning multiple tasks jointly by exploring the relatedness between tasks, as well as the existence of the underlying relatedness between tasks in the ADNI dataset. Second, the RMSE results of MTIL-L-S and MTL-L are comparable with each other, which indicates that the multiple tasks in this dataset do not share the same feature interaction structure. Finally, the result of MTIL-S-Lc method outperforms all other methods, which shows superiority of our feature interaction framework. Through a mixture of 3 low-rank tensor, we are able to learn the feature interaction pattern in this dataset.

5.4 Discussion

The proposed multi-task feature interaction learning framework has provided us a way to bridge related tasks using interaction effects. By employing different types of regularizations on the interaction effects tensor, the formulations under this framework have very different characteristics.

For the shared interaction approach: we utilize Group Lasso on the interaction tensor to control the model complexity. The proximal operator admits a closed form solution, and thus the overall computational cost is very low. We are able to obtain interpretable results from the model, showing what are important interactions that are relevant to the prediction tasks. The main drawback is that we assume all tasks share the same set of interaction effects, which may not be the case for many data sets. One way to further improve the formulation is by extending the strong or weak heredity properties [5, 23] to the proposed MTIL framework.

For the embedded interaction approach: we can easily obtain the global optimal for the convex formulation. Though we are able to tune the regularization parameter on the trace norms to control the rank of the interaction tensor, it is usually very hard to decide the value unless cross-validation is used. A rank larger than necessary may lead to over-fitting when training samples are insufficient. On the other hand, the obtained mixture of 3 tensor is hard to interpret. The non-convex formulation provides a better model decomposition, from which we can see the combination of basis for different tasks and identify embedded bases that are shared among the set of tasks. The drawback of this formulation is that we may easily trapped in a bad local optimal unless we carefully choose the initial value (e.g., using the solution from the convex formulation).

In general, this framework can be generalized into many other possible relatedness on feature interactions by incorporating different regularization terms. Different approaches of this framework should be carefully chosen according to the application domain. In the future work we plan to study the statistical properties of the proposed model, which may lead to deeper understanding of these interaction models.

6. CONCLUSIONS

One major limitation of linear models is the lack of capability to capture predictive information from interactions between features. While introducing high-order feature interaction terms can overcome this limitation, this approach tremendously increases the model complexity and imposes significant challenges in the learning against overfitting. In this paper, we proposed a novel Multi-Task feature Interaction Learning (MTIL) framework to exploit the task relatedness from high-order feature interactions, which provides better generalization performance by inductive transfer among tasks via shared representations of feature interactions. We formulate two concrete approaches under this framework and provide efficient algorithms: the shared interaction approach and the embedded interaction approach. The former assumes tasks share the same set of interactions, and the latter assumes feature interactions from multiple tasks come from a shared subspace. We have provided efficient algorithms for solving the two approaches. Extensive empirical studies on both synthetic and real datasets have demonstrated the effectiveness of the proposed framework.

Acknowledgments

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

References

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

[2] B. Bakker and T. Heskes. Task clustering and gating for bayesian multitask learning. *The Journal of Machine Learning Research*, 4:83–99, 2003.

[3] 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.

[4] S. Ben-David and R. Schuller. Exploiting task relatedness for multiple task learning. In *Learning Theory and Kernel Machines*, pages 567–580. Springer, 2003.

[5] J. Bien, J. Taylor, and R. Tibshirani. A lasso for hierarchical interactions. *Annals of statistics*, 41(3):1111, 2013.

[6] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[7] R. J. Cadoret, W. R. Yates, G. Woodworth, and M. A. Stewart. Genetic-environmental interaction in the genesis of aggressivity and conduct disorders. *Archives of General Psychiatry*, 52(11):916–924, 1995.

[8] R. Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

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

[10] X. Chen, X. Shi, X. Xu, Z. Wang, R. Mills, C. Lee, and J. Xu. A two-graph guided multi-task lasso approach for eqtl mapping. In *AISTATS*, pages 208–217, 2012.

[11] N. H. Choi, W. Li, and J. Zhu. Variable selection with the strong heredity constraint and its oracle property. *JASA*, 105(489):354–364, 2010.

[12] T. C. Eley, K. Sugden, A. Corsico, A. M. Gregory, P. Sham, P. McGuffin, R. Plomin, and I. W. Craig. Gene–environment interaction analysis of serotonin system markers with adolescent depression. *Molecular psychiatry*, 9(10):908–915, 2004.

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

[14] J. Friedman, T. Hastie, and R. Tibshirani. A note on the group lasso and a sparse group lasso. *arXiv preprint arXiv:1001.0736*, 2010.

[15] S. Gandy, B. Recht, and I. Yamada. Tensor completion and low-n-rank tensor recovery via convex optimization. *Inverse Problems*, 27(2):025010, 2011.

[16] J. Gatt, C. Nemeroff, C. Dobson-Stone, R. Paul, R. Bryant, P. Schofield, E. Gordon, A. Kemp, and L. Williams. Interactions between bdnf val66met polymorphism and early life stress predict brain and arousal pathways to syndromal depression and anxiety. *Molecular psychiatry*, 14(7):681–695, 2009.

[17] P. Gong, C. Zhang, Z. Lu, J. Z. Huang, and J. Ye. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In *ICML*, volume 28, page 37, 2013.

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

[19] K. Hemminki, J. L. Bermejo, and A. Försti. The balance between heritable and environmental aetiology of human disease. *Nature Reviews Genetics*, 7(12):958–965, 2006.

[20] S. Ji and J. Ye. An accelerated gradient method for trace norm minimization. In *ICML*, pages 457–464. ACM, 2009.

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

[22] J. Lee, Y. Sun, and M. Saunders. Proximal newton-type methods for convex optimization. In *NIPS*, pages 836–844, 2012.

[23] Y. Liu, J. Wang, and J. Ye. An efficient algorithm for weak hierarchical lasso. In *SIGKDD*, pages 283–292. ACM, 2014.

[24] G. Obozinski, B. Taskar, and M. I. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252, 2010.

[25] N. Parikh and S. P. Boyd. Proximal algorithms. *Foundations and Trends in optimization*, 1(3):127–239, 2014.

[26] P. Radchenko and G. M. James. Variable selection using adaptive nonlinear interaction structures in high dimensions. *Journal of the American Statistical Association*, 105(492):1541–1553, 2010.

[27] B. Romera-Paredes, H. Aung, N. Bianchi-Berthouze, and M. Pontil. Multilinear multitask learning. In *ICML*, pages 1444–1452, 2013.

[28] M. J. Somers. Organizational commitment, turnover and absenteeism: An examination of direct and interaction effects. *Journal of Organizational Behavior*, 16(1):49–58, 1995.

[29] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[30] R. Tomioka, K. Hayashi, and H. Kashima. Estimation of low-rank tensors via convex optimization. *arXiv preprint arXiv:1010.0789*, 2010.

[31] R. Tomioka and T. Suzuki. Convex tensor decomposition via structured schatten norm regularization. In *NIPS*, pages 1331–1339, 2013.

[32] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001.

[33] B. A. Turlach, W. N. Venables, and S. J. Wright. Simultaneous variable selection. *Technometrics*, 47(3):349–363, 2005.

[34] S. J. Wright, R. D. Nowak, and M. A. Figueiredo. Sparse reconstruction by separable approximation. *Signal Processing, IEEE Transactions on*, 57(7):2479–2493, 2009.

[35] J. Xu, P.-N. Tan, and L. Luo. Orion: Online regularized multi-task regression and its application to ensemble forecasting. In *ICDM*, pages 1061–1066. IEEE, 2014.

[36] L. Yuan, J. Liu, and J. Ye. Efficient methods for overlapping group lasso. In *NIPS*, pages 352–360, 2011.

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

[38] J. Zhou, J. Chen, and J. Ye. Clustered multi-task learning via alternating structure optimization. In *NIPS*, pages 702–710, 2011.

[39] 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.