# Sign-Regularized Multi-Task Learning

Guangji Bai* Johnny Torres† Junxiang Wang* Liang Zhao* Cristina Abad†
Carmen Vaca†

## Abstract

Multi-task learning is a framework that enforces different tasks to share their knowledge to improve the generalization performance. It is a long-standing active domain that strives to handle several core issues including which tasks are correlated and similar and how to share the knowledge among correlated tasks. Existing works usually do not distinguish the polarity and magnitude of feature weights and commonly rely on linear correlation, due to three major technical challenges in: 1) optimizing the models that regularize feature weight polarity, 2) deciding whether to regularize sign or magnitude, 3) identifying which tasks should share their sign and/or magnitude patterns. To address them, this paper proposes a new multi-task learning framework that can regularize feature weight signs across tasks, beyond the conventional framework for feature weight regularization. We innovatively formulate such sign-regularization problem as a biconvex inequality constrained optimization upon the multiplications among feature weights with slacks. We then propose a new efficient algorithm for the optimization with theoretical guarantees on generalization performance and convergence. Extensive experiments on multiple datasets show the proposed methods' effectiveness, efficiency, and reasonableness of the regularized feature weighted patterns.

## 1 Introduction

In the real world, many learning tasks are correlated and have shared knowledge and patterns. For example, the sentiment analysis models built for the texts in different domains (e.g., sports, movie, and politics) exhibit shared patterns (e.g., emotion, icons) and exclusive patterns (e.g., domain-specific terminologies). Multi-task learning is a machine learning framework that makes it possible for different learning tasks to share their common knowledge yet preserve their exclusive characteristics and eventually improve the generalization performance of all the tasks. Multi-task learning has been applied into many types of learning tasks such as supervised, unsupervised, semi-supervised, and reinforcement learning tasks as well as numerous applications such as natural language understanding [25], computer vision [4], event forecasting [13, 14, 37], etc.

Multi-task learning is an active domain that has attracted much attention and research efforts. The key challenge in multi-task learning is how to selectively transfer information among the related tasks while preventing sharing knowledge between unrelated tasks, also known as *negative transfer* [29]. To achieve this, we must identify: 1) which tasks are correlated, and 2) which types of knowledge can be shared among the correlated tasks. Many research efforts have been recently devoted to addressing these two core issues. While most of the methods assume that all the tasks are correlated, fast-increase amount of methods are devoted to automatically identify which tasks are correlated and which are not, usually with some assumptions on the correlation patterns such as tree-structured [15], clustered [38, 40], graph-structured [18, 38], and deep-learning based [26, 3]. The price is the increase of computational complexity, risk of over-fitting, and difficulty of optimization [5] (e.g., involving discrete optimization). To attack the second core issue, different types of shared knowledge have been proposed in terms of model parameters (i.e., feature weights), by assuming that different tasks should share similar typically in terms of magnitude and linear correlation, such as *magnitude of feature weights* (e.g., via $\ell_{2,1}$ norm [24, 19]), latent topics (e.g., via enforcing *low-rank structure of feature weights* [11, 2]), and *value of feature weights* (e.g., squared loss among the feature weights). Existing works typically focus on regularizing the magnitude or the similarity among the weights instead of merely their signs.

Despite the large amount of existing work, many types of real-world tasks correlations cannot be extensively covered, especially those without tight and linear correlations. Frequently, it is appropriate to merely enforce similar *polarity* but not magnitude of feature weights across different tasks. For example, we may assume the term "happy" to contribute positively to both the tasks of sentiment classification on each "movie review blog" and each "sport-related tweet", but we do not

[*] Emory University, Atlanta, GA. {guangji.bai, junxiang.wang, liang.zhao}@emory.edu.

[†] Escuela Superior Politecnica del Litoral, ESPOL. jomatorr@espol.edu.ec, {cabad, cvaca}@fiec.espol.edu.ec.

Guangji Bai and Johnny Torres contribute equally to this work.

further expect it contributes with similar strengths in determining the overall text sentiments in these two tasks, due to their differences in topic and text length. In other cases, a feature $A$ that is more important than feature $B$ in one task might not necessarily indicate that it should be more important than $B$ in another task. However, the above issues have not been well explored due to several technical challenges including **1). Difficulties in optimizing the models that regularize feature weight polarity.** Feature weight signs involve discrete functions, which makes it difficult to jointly optimize with those continuous optimization problems in current multitask learning frameworks. **2) Incapability in deciding whether to regularize sign or magnitude.** It is difficult for existing methods to automatically learn and distinguish when the features' weights should share the same signs and when to further share similar importance across tasks. **3) Challenges in identifying which tasks should share their sign and/or magnitude patterns.** Not all tasks may satisfy the regularization of the sign and magnitude of weights. We need an efficient algorithm with theoretical guarantees for identifying task relations that satisfy sign regularization.

To address the above challenges, we propose a new **S**ign-**R**egularized **M**ulti-task **L**earning (**SRML**) framework that adaptively regularizes weight signs across different tasks. The proposed regularization framework has the potential to be applied in different models (e.g., ranging from linear regression models to deep neural networks) and task relations, though exploring different combinations of them is not the focus of this paper. Our main contributions include:

- **A new robust multi-task learning framework.** Our framework can regularize different tasks to share the same weight signs. It can automatically identify which tasks and features can share weight signs.
- **A new algorithm for parameter optimization.** The learning model has been innovatively formulated as a biconvex inequality constrained problem. New efficient optimization algorithm has been proposed based on non-convex alternating optimization.
- **Theoretical properties and guarantee of the proposed algorithm.** Theoretical merits of the proposed algorithm including convergence, convergence rate, generalization error and time complexity have been analyzed.
- **Extensive experiments have been conducted.** We demonstrated the effectiveness and efficiency on 6 real-world and 2 synthetic dataset, under the comparison with other multi-task learning methods. Further analyses on the learned feature weight patterns reveal the effectiveness of our method.

## 2 Related Work

**Multi-task Learning (MTL):** Multi-task learning is a long-standing active research domain [36]. Two key research threads in multi-task learning strive to respectively explore: 1) what are the relations among the tasks, and 2) how to encompass the (known) task relations for jointly learning multiple tasks toward better generalizability. For the first research thread, in the earlier study on multi-task learning, the task relations are assumed to be known as prior knowledge. For example, [11] assumes that each task is similar to any other task. Hence, the model parameters of any single task are pushed to approach the average model parameters of all the tasks. More recent research works on multi-task learning aim at learning the relationship between tasks automatically [38, 22] based on certain assumptions over the cross-task relationship patterns. More recently, ever-increasing amount of work utilizes deep neural networks to learn the task relationship from the data, in the absence of explicit knowledge on task relations [31, 30]. Differently, the second thread focuses on leveraging different inductive biases to jointly learn the models of the related tasks under various ways of regularization. Over decades, different types of regularization has been proposed. For example, some works, including those based on deep learning, assume identical parameter values across tasks by sharing (part of) model parameters across tasks. Some works relax the assumptions by only enforcing similar parameters among different tasks [39], while some others only encouraging similar magnitude of parameter values [16].

This paper belongs to the second research thread in multitask learning mentioned above. Here we instead regularize the polarity of parameter values across tasks. Like many other methods in this thread, it could be generic to different types of predictive models (e.g., linear models or deep learning models) and task relations, though testing our method on all of their combinations is not the focus of this paper.

**Non-convex ADMM:** Though non-convex ADMM has achieved great performance in many problems, there is limited result on the theoretical side of the algorithm due to the complexity of both multiple coupled variables and various (e.g., inequality) constraints. Specifically, [17] proposed a majorized ADMM and gave convergence guarantee when the step length was either small or large. [12] discussed the convergence properties when the coupled objective function was jointly convex. [35] presented their convergence conditions when the coupled objective function was non-convex and non-smooth. [7] discussed the quadratic coupled terms. More recent work has extended the convergence result to deep learning, which are highly non-convex [33].

## 3 Proposed Method

This section introduces the sign-regularized multi-task learning (SRML) problem which encourages same weight polarity across multiple tasks during multi-task learning.

Define a multi-task learning problem with $T$ tasks, where the set of tasks $t \in \{1, \dots, T\}$ associated with a set of instances, $X_t \in \mathbb{R}^{m_t \times d}$ represent the input data, while $y_t \in \mathbb{R}^{m_t}$ is the target variable. Here $m_t$ denotes the number of instances for task $t$ and $d$ denotes the number of features.

Definition 3.1. (Sign-regularized MTL) *Given tasks $t \in \{1, \cdots, T\}$, we want to learn $T$ predictive mappings, where for each task $t$ the mapping is $f : \mathbb{R}^{m_t \times d} | w_t \to \mathbb{R}^{1 \times m_t}$ where the mapping function $f$ is parameterized by $w_t \in \mathbb{R}^{d \times 1}$, such that $\forall\ i \neq j$ we have* $\text{sign}(w_i) = \text{sign}(w_j)$.

Unlike most of the MTL frameworks that regularize the magnitude of the weights, the problem defined in Definition 3.1 is a new type of multi-task problem that regularizes over the *signs* of the weights. Such assumption is usually weaker and easier to satisfy in many types of applications, where tasks only need to share their knowledge on whether each feature should contribute positively or negatively to the prediction. The learning objective for SRML problem is formulated as

$$\min_{w_1,\cdots,w_T} \sum_{t=1}^{T} \mathcal{L}_t(w_t) + \lambda \cdot \Omega(w_{1:T}), \quad \text{s.t.} \quad w_{t,j} w_{t+1,j} \geq 0, \ \ \forall\, t = 1, \cdots, T-1, \ j = 1, \cdots, d, \tag{3.1}$$

where the inequality constraint enforces the same signs of each feature $j$ across different tasks. $\mathcal{L}_t(\cdot)$ denotes each task's loss function and $\Omega(\cdot)$ is the regularization over all the parameters. Eq 3.1 assumes all the tasks must completely share their polarity of weights, which may be too strict considering the possible noise and negative transfer among tasks. To enhance robustness and in the meanwhile allow automatic identification of those tasks who cannot share their weight signs, we add relaxations to it, leading to the following:

$$\begin{aligned} &\min_{w_1,\cdots,w_T,\xi} \sum\nolimits_{t=1}^{T} \mathcal{L}_t(w_t) + \lambda \sum\nolimits_{t=1}^{T} \Omega_t(w_t) + c \sum\nolimits_{t=1}^{T-1} \xi_t \\ &\quad \text{s.t.} \quad w_{t,j} w_{t+1,j} + \xi_{t,j} \geq 0, \quad \xi_{t,j} \geq 0, \\ &\qquad\quad\ \forall\, t = 1, 2, \dots, T-1, \quad j = 1, 2, \dots, d, \end{aligned} \tag{3.2}$$

where each $\xi_t \in \mathbb{R}^d$ is a *slack* variable, and $c$ is a hyperparameter for controlling the level of slacking.

## 4 Optimization Method

The optimization objective in Eq 3.2 is nonconvex with biconvex terms in inequality constraints. Moreover, there will be a huge number of constraints when the numbers of tasks and features are huge. There is no existing efficient methods that can handle this challenging new problem with theoretical guarantee. Although efficiency-enhanced methods such as Alternating Direction Methods of Multipliers (ADMM) [6] are demonstrated to accelerate classical Lagrangian methods, extending them to handle nonconvex-inequality constraints are highly nontrivial. To address such issues, this paper proposes a new efficient algorithm based on ADMM for handling such nonconvex-inequality-constrained problem. Theoretical analyses on convergence properties, generalization error, and complexity analysis are provided.

By adding auxiliary variable $u$ and apply Lagrangian method to drag the constraint onto the objective, the problem in Eq 3.2 can be transformed into Eq 4.3:

$$\begin{aligned} &\min_{w_t,u_t} \sum\nolimits_{t=1}^{T} \mathcal{L}_t(w_t) + \lambda \sum_{t=1}^{T} \Omega_t(w_t) \\ &\qquad + c \sum\nolimits_{t=1}^{T-1} \sum\nolimits_{j=1}^{d} RELU\big(-u_{t,j} u_{t+1,j}\big) \\ &\quad \text{s.t.} \quad [w_1; \cdots; w_T] - [u_1; \cdots; u_T] = 0, \end{aligned} \tag{4.3}$$

where the new optimization problem has simpler constraints. Moreover, the smooth part, nonsmooth part, nonconvex part, and constraints have been separated, which allows to be formed into independent subproblems each of which is much easier to solve. Further, the augmented Lagrangian can be given as

$$\begin{aligned} L_\rho = &\sum\nolimits_{t=1}^{T} \Big[\mathcal{L}_t(w_t) + \lambda \left\|w_t\right\|_2^2\Big] \\ &+ c \sum\nolimits_{t=1}^{T-1} \sum\nolimits_{j=1}^{d} RELU\big(-u_{t,j} u_{t+1,j}\big) \\ &+ y^\intercal \Big([w_1; \cdots; w_T] - [u_1; \cdots; u_T]\Big) \\ &+ \frac{\rho}{2} \Big\|[w_1; \cdots; w_T] - [u_1; \cdots; u_T]\Big\|_2^2, \end{aligned} \tag{4.4}$$

where $y$ is the dual variable, $\rho$ is the penalty parameter that controls the trade-off between primal and dual residual. Now we can perform alternating optimization upon Eq 4.4 to alternately optimize all the variables until convergence, which is detailed in Section ”Optimization Method” in the appendix due to limited space.

## 5 Theoretical Analyses

In this section, we will present the theoretical properties of our SRML model and algorithms.

### 5.1 Generalization Error Bound.

We first provide an equivalent transformation on our original problem and then give the generalization error bound for it. Our original slacked multi-task learning problem with $\ell_1$-

norm regularization can be written as:

$$
\begin{aligned}
&\min_{w,\xi}\ \frac{1}{T}\sum_{t=1}^{T}\frac{1}{m}\sum_{i=1}^{m}\mathcal{L}\big(\langle w_t, x_{ti}\rangle, y_{ti}\big)\\
&\quad+\lambda\sum_{t=1}^{T}\|w_t\|_1 + c\sum_{t=1}^{T-1}\|\xi_t\|_1\,,\quad \text{s.t.}\\
&\quad w_t\otimes w_{t+1}+\xi_t\geq 0,\ \xi_t\geq 0;\ t=1,2,\ldots,T-1,
\end{aligned}
\tag{5.5}
$$

where $\otimes$ over any two vectors $a, b \in \mathbb{R}^n$ is defined as: $a\otimes b := (a_1b_1, a_2b_2, \cdots, a_nb_n)^T$. By combining the constraints with respect to $\xi$, we can simplify the constraints and get:

$$
\begin{aligned}
&\min_{w}\ \frac{1}{T}\sum_{t=1}^{T}\frac{1}{m}\sum_{i=1}^{m}\mathcal{L}\big(\langle w_t, x_{ti}\rangle, y_{ti}\big)\\
&+\lambda\sum_{t=1}^{T}\|w_t\|_1 + c\cdot\sum_{t=1}^{T-1}\big\|RELU(-w_t\otimes w_{t+1})\big\|_1.
\end{aligned}
\tag{5.6}
$$

We can prove by simply using the Lagrangian that Eq 5.6 could be equivalently transformed into the following one with a new set of parameters:

$$
\begin{aligned}
&\min_{w}\ \frac{1}{T}\sum_{t=1}^{T}\frac{1}{m}\sum_{i=1}^{m}\mathcal{L}\big(\langle w_t, x_{ti}\rangle, y_{ti}\big),\quad \text{s.t.}\\
&\sum_{t=1}^{T}\|w_t\|_1\leq\alpha,\ \sum_{t=1}^{T-1}\big\|RELU(-w_t\otimes w_{t+1})\big\|_1\leq\beta.
\end{aligned}
$$

Assumption 5.1. *The loss function $\mathcal{L}$ in this paper has values in $[0,1]$ and has Lipschitz constant $L$ in the first argument for any value of the second argument, i.e.: 1. $\mathcal{L}(\langle w,x\rangle,y)\in[0,1]$ 2. $\mathcal{L}(\langle w,x\rangle,y)\leq L\langle w_t,x\rangle,\forall y$.*

Definition 5.1. *(Expected risk, Empirical risk). Given any weights $w$, we denote the expected risk as:*

$$
\mathbb{E}(w) := \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{(x,y)\sim\mu_t}[\mathcal{L}(\langle w_t,x\rangle,y)]
\tag{5.7}
$$

*Given data $Z=(X,Y)$, the empirical risk is defined as:*

$$
\hat{\mathbb{E}}(w|Z) := \frac{1}{T}\sum_{t=1}^{T}\frac{1}{m}\sum_{i=1}^{m}\mathcal{L}(\langle w_t, X_{ti}\rangle, Y_{ti})
\tag{5.8}
$$

Definition 5.2. *(Global optimal solution, Optimized solution). Define $\mathcal{F}_{\alpha,\beta}=\{w\in\mathbb{R}^{d\times T}:\sum_{t=1}^{T}\|w_t\|_1\leq\alpha,\sum_{t=1}^{T-1}\|max(0,-w_t\otimes w_{t+1})\|_1\leq\beta\}$. Denote $w^*$ as the global optimal solution of the expected risk:*

$$
\begin{aligned}
w^* &:= \arg\min_{w\in\mathcal{F}_{\alpha,\beta}}\mathbb{E}(w)\\
&= \arg\min_{w\in\mathcal{F}_{\alpha,\beta}}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{(x,y)\sim\mu_t}[\mathcal{L}(\langle w_t,x_{ti}\rangle,y_{ti})].
\end{aligned}
\tag{5.9}
$$

*Denote $w^*_{(Z)}$ as the optimized solution by minimizing the empirical risk:*

$$
\begin{aligned}
w^*_{(Z)} &:= \arg\min_{w\in\mathcal{F}_{\alpha,\beta}}\hat{\mathbb{E}}(w|Z)\\
&= \arg\min_{w\in\mathcal{F}_{\alpha,\beta}}\frac{1}{T}\sum_{t=1}^{T}\frac{1}{m}\sum_{i=1}^{m}\mathcal{L}(\langle w_t, X_{ti}\rangle, Y_{ti}).
\end{aligned}
\tag{5.10}
$$

Finally, the following theorem shows the upper-bounded generalization error of our SRML model.

Theorem 5.1. (Generalization error bound) *Let $\epsilon>0$ and $\mu_1,\mu_2,\ldots,\mu_T$ be the probability measure on $\mathbb{R}^d\times\mathbb{R}$. With probability of at least $1-\epsilon$ in the draw of $Z=(X,Y)\sim\prod_{t=1}^{T}\mu_t^m$, we have:*

$$
\begin{aligned}
\mathbb{E}(w^*_{(Z)})-\mathbb{E}(w^*) =&\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{(x,y)\sim\mu_t}[\mathcal{L}(\langle w^*_{(Z)t},x\rangle,y)]\\
&-\inf_{w\in\mathcal{F}_{\alpha,\beta}}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{(x,y)\sim\mu_t}[\mathcal{L}(\langle w_t,x\rangle,y)]\\
\leq&\frac{2L\alpha}{mT}\max_{1\leq t\leq T}\|x_t\|_{1,\infty}+2\sqrt{\frac{2\ln 2/\epsilon}{mT}}
\end{aligned}
$$

Our Theorem 5.1 provides important insights into the proposed model: **1)** The more training samples available, the lower generalization error it will be; **2)** The generalization error converges to 0 when the training sample size approaches infinity; **3)** The smaller value of $\max_{1\leq t\leq T}\|x_t\|_{1,\infty}$ is, the faster convergence rate of the error bound will be.

**5.2 Convergence Analysis** In this section, we analyze the conditions and properties of our non-convex ADMM algorithm's convergence. For simpler notation, we will use $F$ to denote the objective function in Eq 4.3. We prove the convergence by first introducing the following definitions:

Definition 5.3. *Given any input data $X\in(\mathbb{R}^d)^{mT}$, define constant $H_{reg}$ and $H_{class}$ as:*

$$
H_r := \max_t\Big\{2\,\|X_t^{\intercal}X_t\|\Big\};\ H_c := \max_t\Big\{\frac{1}{m}\sum_{j=1}^{m}\|X_{tj}\|^2\Big\}
$$

The following theorem guarantees the global convergence of our ADMM. Set $H=\max\{H_r,H_c\}$, we have:

Theorem 5.2. *If $\rho>2H$, then for the variables $(w_1,\cdots,w_t,u_1,\cdots,u_t,y)$ in Eq 4.4, starting from any $(w_1^0,\cdots,w_t^0,u_1^0,\cdots,u_t^0,y^0)$, this sequence generated by our ADMM has the following properties:*

1. *Residual convergence: $r^k\to 0$ as $k\to\infty$, where $r$ is the primal residual.*

2. *Objective convergence: the objective function defined in Eq 4.3 converges as $k \to \infty$.*

Eq 5.2 only guarantees the convergence of nonconvex ADMM, but $w$ and $u$ are not necessarily converging. The next theorem shows that they will converge to a Nash point:

Theorem 5.3. (Convergence to a Nash point) *For $w$ and $u$ defined in Eq 4.3, the vector of variables $(w_1^k, \cdots, w_T^k, u_1^k, \cdots, u_T^k)$ will converge to a feasible Nash point $(w_1^*, \cdots, w_T^*, u_1^*, \cdots, u_T^*)$ of the objective function in Eq 4.3.*

The last theorem shows that non-convex ADMM can achieve a sublinear convergence rate despite the nonconvex and nonlinear nature of Eq 4.3.

Theorem 5.4. (Convergence Rate Analysis) *For a sequence $(w_1^k, \cdots, w_T^k, u_1^k, \cdots, u_T^k, y^k)$, define*

$$v^k := \min_{0 \le l \le k} \Big( \sum_{t=1}^{T} \big\| w_t^{l+1} - w_t^l \big\|_2^2 + \sum_{t=1}^{T} \big\| u_t^{l+1} - u_t^l \big\|_2^2 \Big),$$

*then the convergence rate of $v_k$ is $o(1/k)$.*

All formal proofs can be found in the appendix, which can be found at `https://drive.google.com/file/d/15iiw-Wh3S4WjeSOXpeSbT68-Kg_iOCgW/view?usp=sharing`

### 5.3 Time Complexity Analysis.

Denote the number of iterations for non-convex ADMM as $l_1$ and the number of iterations for (projected) gradient descent as $l_2$. The time complexity per iteration of gradient descent for subproblem of $w_t$ is the complexity for calculating $w_t$'s gradient. For example, in regression problem with $\ell_2$ regularization, the gradient for $w_t$ is $2(X_t^\intercal X_t + (\lambda+\rho)I)w_t - (2X_t^\intercal Y_t + \rho(u_t^k - y_t^k/\rho))$. The time complexity for calculating this gradient is $\mathcal{O}(dm)$. In addition, we analytically solve the subproblem for a single $u_{tj}$, where we find a minima for a univariate piece-wise quadratic function. Therefore, the time complexity for solving all subproblems of $u$ should be $\mathcal{O}(Tdm)$. Hence, the total time complexity of our non-convex ADMM algorithm is $\mathcal{O}(l_1(l_2T(dm) + Tdm))$ and can be simplified as $\mathcal{O}(l_1 l_2 Tdm)$, which is linear to the sample size.

## 6 Experiments

In this section, we evaluate the performance of our SRML using several synthetic and real-world datasets against the state-of-the-art, on various aspects including accuracy, efficiency, convergence, sensitivity, scalability, and qualitative analyses. The experiments were performed on a 64-bit machine with a 8-core processor (i9, 2.4GHz), 64GB memory. **Code** available at `https://github.com/BaiTheBest/SRML`.

Table 1: Performance on real-world regression datasets (MSE). The reported results are taken average on 10 random seeds. Our proposed SRML outperformed comparison methods on all three datasets while achieving competitive computational efficiency.

| Model | School | Computers | TrafficSP |
|---|---|---|---|
| CASO | 107.39 ±1.65 | 31.91 ±5.21 | 9.83 ±1.32 |
| L21 | 107.78 ±1.66 | 31.91 ±5.21 | 10.46 ±1.21 |
| LASSO | 108.30 ±1.65 | 31.91 ±5.21 | 10.46 ±1.21 |
| RMTL | 108.16 ±1.65 | 39.89 ±7.11 | 10.24 ±1.38 |
| Multi-Lasso | 108.23 ±1.75 | 31.26 ±5.91 | 13.64 ±1.44 |
| MWT | 107.16 ±1.23 | 31.77 ±4.21 | 12.03 ±1.26 |
| SSML | 107.89 ±1.56 | 31.93 ±5.21 | 9.70 ±1.23 |
| SRML | **106.65 ±1.90** | **30.63 ±5.78** | **9.52 ±1.10** |

### 6.1 Experimental Settings Synthetic Datasets:

There are 2 synthetic datasets whose generation process is elaborated in the appendix due to limited space. We manually design the generation process to make sure the feature weight sign follows some patterns. **Synthetic Dataset 1** is for regression, with 20 tasks, 100 instances per task, and 25 features. **Synthetic Dataset 2** is for classification, with 5 tasks, 100 instances per task, and 25 features.

**Real-World Datasets:** Six real-world datasets were used to evaluate the proposed methods and the comparison methods, including: **1).** School Dataset, **2).** Computer Dataset, **3).** Traffic SP Dataset **4).** CIFAR-10, **5).** CelebA, **6).** COCO. For image dataset, we followed standard protocol [23] to extract semantic features by a pre-trained CNN and use PCA to reduce the dimension of the features. The detailed descriptions and download links are elaborated in Section " Real-world Datasets" of our appendix.

**Comparison Methods and Baseline.** We compared our SRML with several recent state-of-the-art multi-task leraning methods. (1) Lasso (2) Join Feature Learning (L21) [1]. (3) Convex Alternating Structure Optimization (CASO) [8]. (4) Robust Multi-task Learning (RMTL) [9]. (5) Trace-norm Regularized Learning (LowRank) [21]. (6) Sparse Structure-Regularized Multi–task Learning (SRMTL) [39]. (7) Multi-level Lasso [27]. (8) MTW [20]. (9) Strict Sign-regularized Multi-task Learning (SSML) The implementation of all the multi-task learning methods is based on [39], where either linear regression or logistic regression model is employed depending on the actual problem type.

**Evaluation Metrics:** To evaluate the performance of the methods on regression tasks, we employ the mean absolute error (MAE), mean square error (MSE), and the mean square logarithmic error (MSLE). For classification tasks, the accuracy (ACC) and area under the curve score (AUC) are used to evaluate the performance, where a larger value denotes better performance.

Table 2: Performance on real-world large-scale image classification datasets. Our proposed SRML can outperform comparison methods in most cases, demonstrating our model's potential ability in handling deep semantic features.

| Model | CIFAR-10 | | CelebA | | COCO | |
|---|---|---|---|---|---|---|
| | ACC | AUC | ACC | AUC | ACC | AUC |
| CASO | 91.96 | 51.47 | 56.96 | 59.89 | 53.66 | 59.02 |
| L21 | 91.50 | 87.58 | 76.09 | 66.12 | 75.07 | **65.02** |
| LASSO | 91.48 | 86.64 | 75.55 | 66.69 | 76.50 | 64.40 |
| SRMTL | 92.76 | 91.13 | 75.79 | 65.16 | 78.36 | 62.46 |
| LowRank | 92.28 | 85.65 | 75.52 | 66.99 | 76.87 | 65.01 |
| SSML | 92.98 | 91.76 | 76.82 | 66.09 | 77.49 | 62.58 |
| SRML | **93.53** | **93.35** | **77.85** | **67.60** | **80.45** | 63.25 |

Table 3: Performance on Synthetic Dataset 1 (regression task). Our SRML outperformed all baselines by a great margin thanks to capturing the sign relation between tasks as designed in the synthetic datasets.

| Model | MAE | MSE | MSLE |
|---|---|---|---|
| CASO | 6.96E+01 | 8.55E+03 | 9.21E-03 |
| L21 | 7.12E+01 | 8.96E+03 | 6.23E-03 |
| LASSO | 7.12E+01 | 8.96E+03 | 6.23E-03 |
| RMTL | 6.80E+01 | 8.14E+03 | 1.13E-02 |
| MWT | 4.73E+01 | 6.25E+03 | 5.79E-03 |
| SSML | 2.73E+03 | 1.28E+07 | 1.20E+00 |
| SRML | **2.02E+00** | **1.05E+01** | **1.36E-06** |

**6.2 Experimental Results. Effectiveness Evaluation in Synthetic Dataset:** The empirical results (Table 3) show that our SRML model achieves the best performance on synthetic datasets for regression task (left) and classification task (right).

For regression tasks, our SRML outperforms the baseline models by a large margin for all the metrics. For the MAE, it achieves an order of magnitude better score w.r.t the best baseline model RMTL. The MSE and MSLE metrics show similar improvements (several orders of magnitude w.r.t the baseline model). Although SSML uses a similar approach, it enforces a strict polarity regularization compared to our model. The hard constraints in the SSML model fail to capture changes in features' polarity between tasks and achieve the worst performance compared to other baseline models.

For the binary classification task, our model achieves the best score in every metric. The AUC metric shows a significant margin (8%) compared to the baseline, which indicates that our model will perform better at different thresholds for labels. However, the margin of improvement is small for ACC w.r.t. the best comparison method. The reason for this is because the dependent variable is less sensitive to the variation of the magnitude of weights of the model parameters in the dataset generation. The SSML has been excluded from classification experiments as the reference paper only provide their implementation for regression tasks.

**Effectiveness Evaluation in Real-world Dataset:** Table 1 shows the results of our method SRML and the baselines on the shallow real-world datasets in the MSE metric (average over 10 runs and the standard deviation). SRML outperforms the comparison methods on ALL three datasets, by a clear margin. We found our model performs well on datasets (e.g., Computers) with mixed features types (categorical and real values), since in these types of datasets we can exploit the features' sign correlation between different tasks. In addition, the training runtime on TrafficSP Dataset is also presented, where we can see that the fastest method is LASSO due to its relative simplicity. Our method, though is slower than simple methods such as LASSO and L21, is still highly efficient comparing with other complex methods such as CASO and RMTL. The runtime on other datasets follow similar trend.

We report the performance of our method and the baselines on the image dataset in Table 2. As can be seen, SRML significantly outperforms the baselines on all three dataset and the strict version SSML achieves the second-best performance on two dataset (CIFAR-10 and CelebA). SRML exhibited outstanding performance on CelebA and COCO where the number of tasks is relatively large and there exist many similar tasks in the sense that the semantic features being important in some tasks are likely to be important in the related tasks as well. For example, in CelebA we have two tasks to classify whether a celebrity's face is smiling and whether his/her mouth is slightly open, respectively. The semantic features around the mouth area are important and those semantic features are likely to share the same polarity of feature weights on both tasks (smiling and having mouth open are highly likely to co-occur in face photos). The strict version SSML was slightly outperformed by SRMTL baseline on COCO in accuracy, which is possibly due to the fact that COCO has the most number of tasks (> 80) and without the slack variable SSML's performance will be harmed by some noisy tasks. SRMTL achieved the best performance among all the baselines on both CIFAR-10 and COCO, showing that exploiting task relationship could be more beneficial in practice than learning the shared feature subspace (e.g., $L_{21}$), especially when the number of tasks is relatively large and there exist many similar tasks.. In general, for all methods AUC is lower than ACC, which could be due to each task is a one-versus-all binary classification problem so the dataset is more imbalanced on CelebA and COCO where the number of classes is bigger.

**Qualitative Analysis:** In order to investigate whether

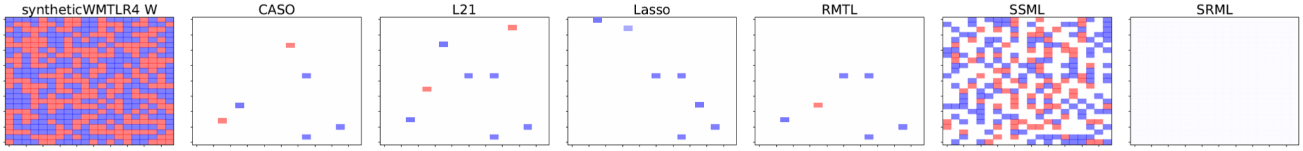
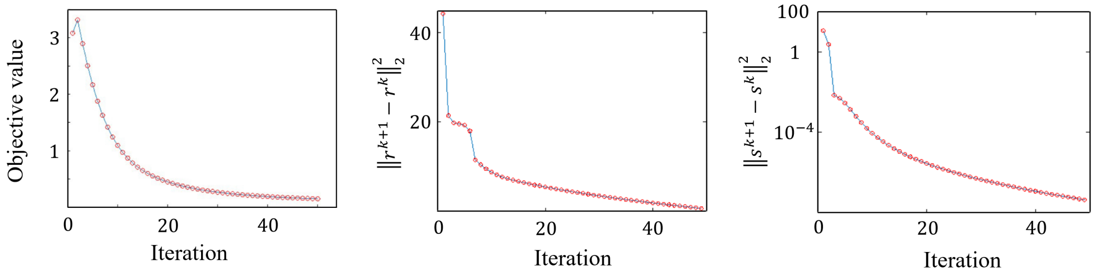
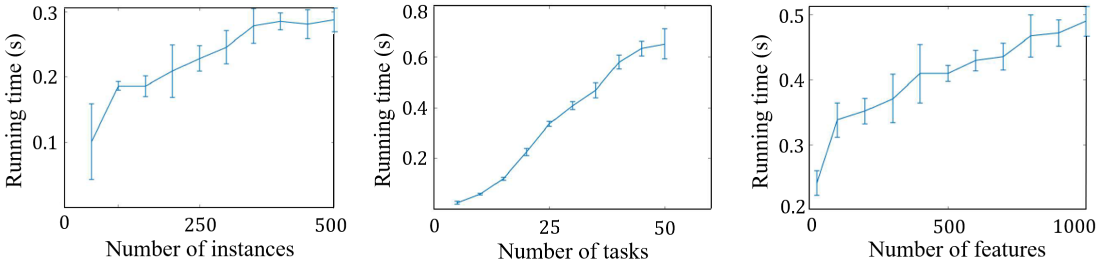


Figure 1: Illustration on how well the signs of the learned feature weights match the ground truth on Synthetic Dataset 1. For each model, we show the selected weights ($y$ axis) of each of the 20 tasks ($x$ axis), where each cell's color denote if the sign matches to the ground truth (*white* color), or there is a difference either positive (*red*) or negative (*blue*). Hence, our model's results completely match the ground truth. Zoom in for detail.



(a) Iteration v.s. objective value (b) Iteration v.s. primal residual (c) Iteration v.s. dual residual

Figure 2: Convergence on synthetic dataset.



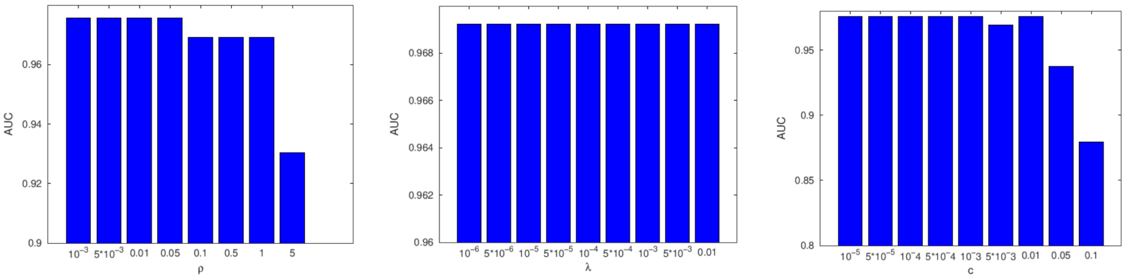
(a) Runtime v.s. #Instances (b) Runtime v.s. #Tasks (c) Runtime v.s. #Features

Figure 3: Scalability Analysis on synthetic dataset.

and how the proposed regularization in SRML impact and benefit the learning of feature weight signs, Figure 1 shows a comparison among different methods in terms of the difference between their learned signs and ground truth signs in all tasks and features. The first subplot labeled "syntheticWMTLR4W" shows the ground truth feature weights' signs, while the other subplots correspond to the differences between the weights' signs learned by different models and the ground truth signs shown in the first subplot. It can be clearly seen that our SRML achieves an exact match to the ground truth as the cells are all-white, meaning "no difference" to the ground truth. It hence outperforms the competing methods who do not leverage the sign-regularization for instructing multitask learning for this types of tasks. Moreover, as we expected, the baseline SSML has numerous cells different to the ground truth because it leverage strict constraints forcing each feature's weights to be the same across tasks. Therefore, the effectiveness of our "slack mechanism" is clearly shown by contrasting the performance between SSML and SRML.

**Convergence Analysis:** The trends of objective function value, primal and dual residual during the optimization of one training process are illustrated in Figures 2a, 2b, and 2c, respectively. The demonstrated convergence results is consistent with our convergence analysis in Section "Convergence Analysis".

**Scalability Analysis:** Figure 3 illustrates the scalability of the proposed SRML on regression synthetic dataset in the training running time when the size of the dataset varies. Specifically, Figure 3a shows that when the numbers of tasks and features are fixed, the runtime increases near-linearly when the number of instances increases. Similar story for other figures. All the observations above are consistent with our theoretical analysis on our proposed method's time complexity.



(a) Sensitivity of parameter $\rho$ (b) Sensitivity of parameter $\lambda$ (c) Sensitivity of parameter c

Figure 4: Sensitivity Analysis on synthetic dataset (Zoom in for detail.)

**Sensitivity Analysis:** For sensitivity analysis, Figure 4a illustrates that our model performs best with parameter $\rho$ smaller than 0.1. In addition, Figure 4b shows our model is barely sensitive to the coefficient regularization $\lambda$ Figure 4b. This is potentially reasonable because the synthetic dataset for this experiment has low dimensions in features and no sparsity. Last, Figure 4c shows our SRML model performs best when the parameter $c$ is smaller than 0.1. This makes sense because we added noise into the sign of ground truth weights and since smaller $c$ provides SRML more slacking our model could achieve better score.

## 7 Conclusions

Considering the assumption that in some real-world applications, the tasks share a similar polarity for features across tasks, we propose sign-regularized multi-task learning framework by enforcing the learning weights to share polarity information to neighbor tasks. Experiments on synthetic and real-world datasets demonstrate the effectiveness and efficiency of our methods in various metrics, compared with several baselines. Various analyses such as convergence analyses, scalability analyses have also been done theoretically and experimentally. Additional analyses on the learned parameters such as sensitivity and qualitative analyses on learned parameters have also been discussed.

## Acknowledgement

This work was supported by the National Science Foundation (NSF) Grant No. 1755850, No. 1841520, No. 2007716, No. 2007976, No. 1942594, No. 1907805, a Jeffress Memorial Trust Award, Amazon Research Award, NVIDIA GPU Grant, and Design Knowledge Company (subcontract number: 10827.002.120.04).

## References

[1] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. Multi-task feature learning. In *Advances in neural information processing systems*, pages 41–48, 2007.

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

[3] Guangji Bai, Ling Chen, and Liang Zhao. Temporal domain generalization with drift-aware dynamic neural network. *arXiv preprint arXiv:2205.10664*, 2022.

[4] Guangji Bai and Liang Zhao. Saliency-regularized deep multi-task learning. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 15–25, 2022.

[5] Christopher M Bishop. *Pattern recognition and machine learning.* springer, 2006.

[6] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[7] Caihua Chen, Min Li, Xin Liu, and Yinyu Ye. Extended admm and bcd for nonseparable convex minimization models with quadratic coupling terms: convergence analysis and insights. *Mathematical Programming*, 173(1-2):37–77, 2019.

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

[9] Jianhui Chen, Jiayu Zhou, and Jieping Ye. Integrating low-rank and group-sparse structures for robust multi-task learning. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 42–50. ACM, 2011.

[10] Wei Deng, Ming-Jun Lai, Zhimin Peng, and Wotao Yin. Parallel multi-block admm with o (1/k) convergence. *Journal of Scientific Computing*, 71(2):712–736, 2017.

[11] Theodoros Evgeniou and Massimiliano Pontil. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117, 2004.

[12] Xiang Gao and Shu-Zhong Zhang. First-order algorithms for convex optimization with nonseparable objective and coupled constraints. *Journal of the Operations*

*Research Society of China*, 5(2):131–159, 2017.

[13] Yuyang Gao and Liang Zhao. Incomplete label multi-task ordinal regression for spatial event scale forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[14] Yuyang Gao, Liang Zhao, Lingfei Wu, Yanfang Ye, Hui Xiong, and Chaowei Yang. Incomplete label multi-task deep learning for spatio-temporal event subtype forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3638–3646, 2019.

[15] Nico Görnitz, Christian Widmer, Georg Zeller, André Kahles, Gunnar Rätsch, and Sören Sonnenburg. Hierarchical multitask structured output learning for large-scale sequence segmentation. In *Advances in Neural Information Processing Systems*, pages 2690–2698, 2011.

[16] Lei Han and Yu Zhang. Learning multi-level task groups in multi-task learning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[17] Mingyi Hong, Tsung-Hui Chang, Xiangfeng Wang, Meisam Razaviyayn, Shiqian Ma, and Zhi-Quan Luo. A block successive upper-bound minimization method of multipliers for linearly constrained convex optimization. *Mathematics of Operations Research*, 2020.

[18] Laurent Jacob, Jean-philippe Vert, and Francis R Bach. Clustered multi-task learning: A convex formulation. In *Advances in neural information processing systems*, pages 745–752, 2009.

[19] Ali Jalali, Sujay Sanghavi, Chao Ruan, and Pradeep K Ravikumar. A dirty model for multi-task learning. In *Advances in neural information processing systems*, pages 964–972, 2010.

[20] Hicham Janati, Marco Cuturi, and Alexandre Gramfort. Wasserstein regularization for sparse multi-task regression. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1407–1416. PMLR, 2019.

[21] Shuiwang Ji and Jieping Ye. An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th annual international conference on machine learning*, pages 457–464, 2009.

[22] Abhishek Kumar and Hal Daumé III. Learning task grouping and overlap in multi-task learning. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1723–1730, 2012.

[23] Giwoong Lee, Eunho Yang, and Sung Hwang. Asymmetric multi-task learning based on task relatedness and loss. In *International conference on machine learning*, pages 230–238. PMLR, 2016.

[24] Jun Liu, Shuiwang Ji, and Jieping Ye. Multi-task feature learning via efficient l2, 1-norm minimization. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 339–348, 2009.

[25] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy, July 2019. Association for Computational Linguistics.

[26] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and S Yu Philip. Learning multiple tasks with multilinear relationship networks. In *NIPS*, 2017.

[27] Aurelie C Lozano and Grzegorz Swirszcz. Multi-level lasso for sparse multi-task regression. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 595–602, 2012.

[28] Andreas Maurer, Massi Pontil, and Bernardino Romera-Paredes. Sparse coding for multitask and transfer learning. In *International conference on machine learning*, pages 343–351, 2013.

[29] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[30] Sebastian Ruder, Joachim Bingel, Isabelle Augenstein, and Anders Søgaard. Latent multi-task architecture learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4822–4829, 2019.

[31] Trevor Standley, Amir R Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? *arXiv preprint arXiv:1905.07553*, 2019.

[32] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[33] Junxiang Wang, Fuxun Yu, Xiang Chen, and Liang Zhao. Admm for efficient deep learning with global convergence. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 111–119, 2019.

[34] Junxiang Wang and Liang Zhao. Nonconvex generalization of alternating direction method of multipliers for nonlinear equality constrained problems. *Results in Control and Optimization*, 2:100009, 2021.

[35] Yu Wang, Wotao Yin, and Jinshan Zeng. Global convergence of admm in nonconvex nonsmooth optimization. *Journal of Scientific Computing*, 78(1):29–63, 2019.

[36] Yu Zhang and Qiang Yang. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*, 2017.

[37] Liang Zhao. Event prediction in the big data era: A systematic survey. *ACM Computing Surveys (CSUR)*, 54(5):1–37, 2021.

[38] Jiayu Zhou, Jianhui Chen, and Jieping Ye. Clustered multi-task learning via alternating structure optimization. In *Advances in neural information processing systems*, pages 702–710, 2011.

[39] Jiayu Zhou, Jianhui Chen, and Jieping Ye. Malsar: Multi-task learning via structural regularization. *Arizona State University*, 21, 2011.

[40] Qiang Zhou, Gang Wang, Kui Jia, and Qi Zhao. Learning to share latent tasks for action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2264–2271, 2013.