Exploiting Task-Feature Co-Clusters in Multi-Task Learning

Linli Xu[†], **Aiqing Huang**[†], **Jianhui Chen**[‡] and **Enhong Chen**[†]
[†] School of Computer Science and Technology
[†] University of Science and Technology of China, Hefei, Anhui 230027, China
[‡] Yahoo Labs, Sunnyvale, CA 94089, USA
linlixu@ustc.edu.cn, cassie89@mail.ustc.edu.cn, jianhui@yahoo-inc.com, cheneh@ustc.edu.cn

Abstract

In multi-task learning, multiple related tasks are considered simultaneously, with the goal to improve the generalization performance by utilizing the intrinsic sharing of information across tasks. This paper presents a multi-task learning approach by modeling the task-feature relationships. Specifically, instead of assuming that similar tasks have similar weights on all the features, we start with the motivation that the tasks should be related in terms of subsets of features, which implies a co-cluster structure. We design a novel regularization term to capture this task-feature co-cluster structure. A proximal algorithm is adopted to solve the optimization problem. Convincing experimental results demonstrate the effectiveness of the proposed algorithm and justify the idea of exploiting the task-feature relationships.

Introduction

Multi-task learning (Caruana 1997) has emerged as a promising discipline with empirical success and theoretical justification in the past decades. In multi-task learning, a number of related tasks are considered simultaneously, with the goal to improve the generalization performance by utilizing the intrinsic sharing of information across tasks.

In most of the existing work on multi-task learning, a key assumption is that there is some certain structure of how the tasks are related to each other, which includes hidden units in neural networks (Caruana 1997), a common prior in hierarchical Bayesian models (Bakker and Heskes 2003), an underlying model shared across tasks (Evgeniou and Pontil 2004), a low-dimensional subspace in tasks (Argyriou, Evgeniou, and Pontil 2007) or a low rank structure of the parameters (Ji and Ye 2009).

The above methods are restricted in the sense that they assume all the tasks are close to each other, or share a common underlying representation, which may not be the case in real problems. When this assumption does not hold, outlier tasks can impair the overall generalization predictive performance, or negative information transfer would occur among dissimilar tasks.

To address this issue, various methods have further been proposed along different directions. For example, task clustering approaches (Thrun and O'Sullivan 1996; Jacob, Bach, and Vert 2008; Kang, Grauman, and Sha 2011) assume that tasks are clustered into groups and tasks within a group are similar to each other, while robust multi-task learning (Chen, Zhou, and Ye 2011; Gong, Ye, and Zhang 2012) groups relevant tasks such that they share a common representation and identifies irrelevant or outlier tasks. On the other hand, the multi-task relationship learning approach (Zhang and Yeung 2010) is able to model negative task correlations in addition to positive task relationships, which is a generalization of the regularization methods in multi-task learning.

In this paper, we consider the multi-task learning problem with the assumption that tasks can be clustered into different groups which are not known a priori. With this task clustering structure, the problem of negative information transfer among dissimilar and outlier tasks can be avoided. Therefore a natural advantage of the task clustering approaches is the improved robustness when learning from multiple tasks. Existing task clustering methods include the clustered multi-task learning formulation (Jacob, Bach, and Vert 2008) that enforces the grouping structure with a regularization term; and the mixed integer programming approach (Kang, Grauman, and Sha 2011) that incorporates the integer cluster indicators into the multi-task feature learning framework; while in an earlier piece of work (Thrun and O'Sullivan 1996), task similarities are measured by how well the model for one task performs in the other task.

The task clustering methods discussed above consider the grouping structure at a general task-level, which assumes that tasks within a group are close to each other on all the features. This assumption could be restrictive in practice. For example, in a document classification problem, different tasks may be relevant to different sets of words; or in a recommender system, two users with similar tastes on one feature subset may have totally different preference on another subset. To address the above issue, in recent work of (Zhong and Kwok 2012) an additional regularization term on task clusters regarding each feature is introduced, which essentially results in clustering of tasks in a feature-by-feature manner. However, this method only considers features individually and neglects the relationship between features. On the other hand, recent work on feature grouping (Shen and Huang 2010; Yang, Yuan, and Lai 2012) exploits relationships of features in the learning procedure, which motivates us to incorporate feature grouping with task clustering in multi-task learning.

To model the task-feature relationship, we follow a more intuitive co-clustering methodology in this paper, where instead of assuming that similar tasks have similar weights on *all* the features, the tasks should be related in terms of *subsets* of features. Co-clustering works in two-way by clustering rows and columns simultaneously, and is preferred especially when there is association between the rows and columns (Li and Ding 2006). Various co-clustering methods (Ding et al. 2006) have been proposed with successful applications in text classification (Dhillon 2001; Dhillon, Mallela, and Modha 2003) and recommender systems (Xu et al. 2012). As far as we know, this is the first piece of work that incorporates co-clustering in multi-task learning.

In this paper, we propose a multi-task learning approach that models a common representation across tasks as well as the task-feature co-cluster structure. Specifically, we design a novel regularization term to capture the co-cluster structure which implies an implicit feature selection for each task. An alternating algorithm is adopted to solve this optimization problem. The proposed method is evaluated experimentally in a synthetic setting as well as on real benchmark datasets with convincing results.

The rest of the paper is organized as follows. We first formulate the problem of multi-task learning with task-feature co-clusters as a risk minimization problem with regularization to enforce the co-cluster structure of the task-feature relationships. The technique to optimize the model is then discussed. Next we present the experimental results to demonstrate the effectiveness of the proposed approach, followed by the conclusion.

Multi-Task Learning with Task-Feature Co-Clusters (CoCMTL)

Suppose there are m tasks where the training set for the i-th task is

$$\mathcal{D}_i = \{X^i, Y^i\} : i = 1, 2, \ldots, m,$$

where $X^i \in \mathbb{R}^{n_i \times d}$ is the input matrix for the i-th task with n_i instances and d features, while $Y^i \in \mathbb{R}^{n_i \times 1}$ is the corresponding target vector. We consider learning a linear predictive model for each task, where $\mathbf{w}^i$ is the weight vector for the i-th task. If we adopt the commonly used least squares loss function, the empirical loss of the set of m linear classifiers given in W can then be written as

$$\ell(W) = \sum_{i=1}^{m} \|Y^i - X^i\mathbf{w}^i\|_2^2 \tag{1}$$

where $W = [\mathbf{w}^1, \mathbf{w}^2, ..., \mathbf{w}^m]$ is the $d \times m$ weight matrix. Following a decomposition scheme, W is divided into two parts, $W = P + Q$, where P reflects global similarities among tasks, while Q captures the task-feature relationships.

Global Similarities

To capture the global similarities among tasks given $P = [\mathbf{p}^1, \mathbf{p}^2, ..., \mathbf{p}^m]$, it is natural to assume that the parameter vectors $\mathbf{p}^i, i \in 1, 2, \ldots, m$ are close to their mean. To encourage that, the following penalty function can be designed:

$$\Omega_1(P) = \sum_{i=1}^{m} \|\mathbf{p}^i - \frac{1}{m}\sum_{j=1}^{m} \mathbf{p}^j\|_2^2 = \mathrm{tr}(PLP^\top) \tag{2}$$

where $L = I - \frac{1}{m}\mathbf{1}\mathbf{1}^\top$. Notice that Eq.(2) also represents the variance of $\mathbf{p}^1, \mathbf{p}^2, \ldots, \mathbf{p}^m$.

Task-Feature Co-Clustering

Besides global similarities, we would like to include a second regularization term $\Omega_2(Q)$ that encodes a clustering of tasks. Instead of assuming that similar tasks have similar weights on all the features as in previous work (Figure 1, left), we take the fact into consideration that two tasks can be related only on a subset of features. That is, the clustering process should be conducted on tasks and features, or columns and rows of the Q matrix simultaneously, and the goal is to divide the tasks and features into k groups as illustrated in (Figure 1, right). It is clear that the co-clustering procedure produces subgroups of tasks and features, which would lead to a block-structure in our framework.

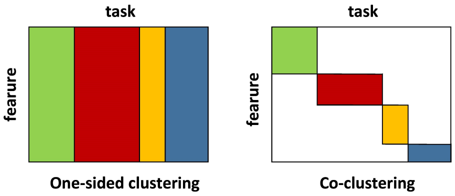

Figure 1: Comparison between one-sided clustering and co-clustering.

To model the task-feature relationships, we can introduce a directed bipartite graph $\mathcal{G} = (\mathcal{U}, \mathcal{V}, E)$ as shown in (Figure 2, left), where $\mathcal{U} = \{u_1, u_2, ..., u_d\}$ and $\mathcal{V} = \{v_1, v_2, ..., v_m\}$ correspond to the sets of feature vertices and task vertices respectively, while E is the set of edges connecting the features with the tasks. The weight on the edge E_{ij} corresponds to the relevance of the i-th feature with the j-th task and is encoded by Q_{ij}.

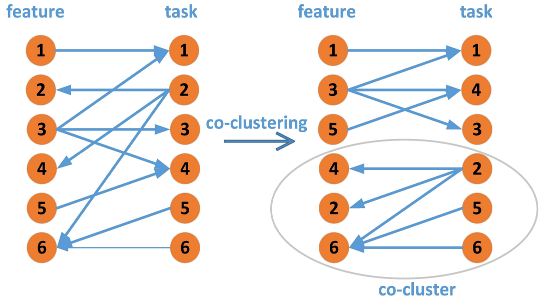

Figure 2: Left: directed bipartite graph of features and tasks; Right: co-clusters of features and tasks.

In the bipartite graph, $Q_{ij} > 0$ corresponds to an edge from the i-th feature to the j-th task and $Q_{ij} < 0$ corresponds to an edge from the j-th task to the i-th feature. $|Q_{ij}|$

encodes how strong the relationship between the i-th feature and the j-th task is.

Before detailing the co-clustering procedure, we first revisit the one-sided clustering approach. Given $Z \in \mathbb{R}^{a\times b}$ where each column of Z represents a sample, according to (Ding, He, and Simon 2005) the traditional K-means clustering with spectral relaxation can be formulated as

$$\min_{H:H^\top H=I} \left\{\mathrm{tr}(Z^\top Z) - \mathrm{tr}((H^\top Z^\top ZH)\right\} \tag{3}$$

where H is a relaxation of the indicator matrix, denoting the assignment of the b samples to k clusters.

In our co-clustering procedure, we first define the data matrix as $Z = \begin{pmatrix} 0 & Q \\ Q^\top & 0 \end{pmatrix}$, where the first d columns index the features while the last m columns correspond to the tasks, then the clustering procedure on Q's rows and columns can be achieved naturally. Here we redefine $H = \begin{pmatrix} F \\ G \end{pmatrix}$ which indicates k clusters among both features and tasks, similarly we have $F^\top F = I$ and $G^\top G = I$. The problem (3) can then be further rewritten as

$$\min_{\substack{F:F^\top F=I,\\ G:G^\top G=I}} \left\{2\|Q\|_F^2 - \mathrm{tr}(F^\top QQ^\top F) - \mathrm{tr}(G^\top Q^\top QG)\right\} \tag{4}$$

In fact, the similarity matrix can be computed by $Z^\top Z = \begin{pmatrix} QQ^\top & 0 \\ 0 & Q^\top Q \end{pmatrix}$, in which the matrices $QQ^\top$ and $Q^\top Q$ are similar to the bibliographic coupling matrix and co-citation matrix in the field of bibliometrics (Satuluri and Parthasarathy 2011). Specifically, $QQ^\top$ and $Q^\top Q$ encode the feature similarities and task similarities respectively, based on which we can group similar features and tasks together and achieve co-clustering (Figure 2, right).

Theorem 1. *For any given matrix $Q \in \mathbb{R}^{d\times m}$, any matrices $F \in \mathbb{R}^{d\times k}$, $G \in \mathbb{R}^{m\times k}$and any nonnegative integer k, $k \leq \min(d,m)$, Problem (4) reaches its minimum value at $F = (\mathbf{u}_1, \dots, \mathbf{u}_k)$, $G = (\mathbf{v}_1, \dots, \mathbf{v}_k)$, where $\mathbf{u}_i$ and $\mathbf{v}_i$ are the i-th left and right singular vectors of Q respectively. The minimum value is $2\sum_{i=k+1}^{\min(d,m)} \sigma_i^2(Q)$, where $\sigma_1(Q) \geq \sigma_2(Q) \geq \cdots \geq \sigma_{\min(d,m)}(Q) \geq 0$ are the singular values of Q.*

Proof. Suppose the singular value decomposition of Q is: $Q = U\Sigma V^\top$, the singular value decomposition of $QQ^\top$ can be written as $QQ^\top = U\Sigma V^\top \cdot V\Sigma U^\top = U\Sigma^2 U^\top$, which essentially implies $\sigma_i(QQ^\top) = \sigma_i^2(Q)$. On the other hand, given $\sigma_i(FF^\top) = \sigma_i(F^\top F)$, it holds that $\sigma_i(FF^\top) = 1$ for $i = 1, 2, \dots, k$ while the remaining singular values are equal to 0. Therefore

$$\begin{aligned} \mathrm{tr}(F^\top QQ^\top F) \leq & \sum_{i=1}^{\min(d,m)} \sigma_i(QQ^\top)\sigma_i(FF^\top) \\ &= \sum_{i=1}^{k} \sigma_i^2(Q)\cdot 1 + \sum_{i=k+1}^{\min(d,m)} \sigma_i^2(Q)\cdot 0 \\ &= \sum_{i=1}^{k} \sigma_i^2(Q), \end{aligned}$$

when $F = U$, we have $\mathrm{tr}(F^\top QQ^\top F) = \sum_{i=1}^{k} \sigma_i^2(Q)$. Similarly, we get $\mathrm{tr}(G^\top Q^\top QG) = \sum_{i=1}^{k} \sigma_i^2(Q)$ when $G = V$. Then the problem (4) reaches its minimum value:

$$\begin{aligned} & 2\|Q\|_F^2 - \mathrm{tr}(F^\top QQ^\top F) - \mathrm{tr}(G^\top Q^\top QG) \\ =& 2\left\{\|Q\|_F^2 - \sum_{i=1}^{k} \sigma_i^2(Q)\right\} = 2\sum_{i=k+1}^{\min(d,m)} \sigma_i^2(Q). \end{aligned}$$

□

Based on the above analysis, we design the task-feature co-clustering regularization term $\Omega_2(Q)$ as

$$\Omega_2(Q) = \|Q\|_K^2 = \sum_{i=k+1}^{\min(d,m)} \sigma_i^2(Q).$$

Notice $\|Q\|_K^2$ is a continuous function of Q, and it changes to zero if $k \geq \mathrm{rank}(Q)$.

Based on the above discussion, we combine global similarities and task-feature group-specific similarities, and the model of multi-task learning with task-feature co-clusters can be formulated as

$$\min_{W,P,Q} \ell(W) + \lambda_1\Omega_1(P) + \lambda_2\Omega_2(Q) \tag{5}$$

where λ_1 and λ_2 control the tradeoff between global similarities and group specific similarities.

Algorithm

In this section, we consider solving the CoCMTL formulation in (5) with proximal methods. Proximal methods, which can be regarded as a natural extension of gradient-based techniques when the objective function consists of a non-smooth convex part, have received significant success in various problems.

However, the regularization term $\Omega_2(Q)$ in (5) is non-convex which makes the traditional proximal method not directly applicable here. Fortunately, we are able to solve the CoCMTL formulation with the Proximal Alternating Linearized Minimization (PALM) scheme (Bolte, Sabach, and Teboulle 2013) which is designed for a general class of problems with non-convex terms.

Proximal Method

Proximal Alternating Linearized Minimization (PALM) discussed in (Bolte, Sabach, and Teboulle 2013) solves a general class of problems in the form below:

$$\min_{P,Q}\{h(P,Q) + g(P) + f(Q)\} \tag{6}$$

where $h(P, Q)$ is a convex function and $g(P)$ and $f(Q)$ are proper lower semi continuous. In our case, $h(P, Q) = \ell(W) + \lambda_1 \mathrm{tr}(PLP^\top)$, $g(P) = 0$ and $f(Q) = \lambda_2 \|Q\|_K^2$. Note that $h(P, Q)$ is differentiable and jointly convex in P and Q whereas $f(Q)$ is non-smooth and non-convex. Denote $R = \begin{pmatrix} P \\ Q \end{pmatrix}$, consider a linear approximation of the convex function $h(R)$ at the previous estimate R_{r-1} regularized by a quadratic proximal term, the current value R_r can be updated as the solution of the proximal problem:

$$R_r = \arg\min_R h(R_{r-1}) + \frac{\gamma_r}{2}\|R - R_{r-1}\|_F^2 + \langle R - R_{r-1}, \bigtriangledown h_R(R_{r-1})\rangle \quad (7)$$

where γ_r is a positive real number and $\langle A, B\rangle = \mathrm{tr}(A^\top B)$ denotes the matrix inner product. $\bigtriangledown h_R(C)$ represents the gradient of function $h(R)$ with regard to R at point C. Next one can add the regularization term $f(Q)$ to (7), decouple P and Q, remove the constant terms and then get the following subproblems:

$$P_r = \arg\min_P \frac{\gamma_r}{2}\|P - C_P(P_{r-1})\|_F^2 \quad (8)$$

$$Q_r = \arg\min_Q \frac{\gamma_r}{2}\|Q - C_Q(Q_{r-1})\|_F^2 + \lambda_2\|Q\|_K^2 \quad (9)$$

where $C_P(P_{r-1}) = P_{r-1} - \bigtriangledown h_P(R_{r-1})/\gamma_r$ and $C_Q(Q_{r-1}) = Q_{r-1} - \bigtriangledown h_Q(R_{r-1})/\gamma_r$ are both constants of the previous solution points P_{r-1} and Q_{r-1}. Notice that P_r can be easily obtained as $P_r = C_P(P_{r-1})$. However, the problem (9), which is also known as the proximal operator for Q, involves a non-convex term. Next, we will discuss how to compute this proximal operator, which is crucial for solving CoCMTL.

Proximal Operator Computation

Here we design a simple but effective alternating method following the similar scheme in (Hu, Zhang, and Ye 2012). Recall that $\|Q\|_K^2 = \mathrm{tr}(F^\top QQ^\top F)$ where $F = (\mathbf{u}_{k+1}, \ldots, \mathbf{u}_{\min(d,m)})$. In the s-th iteration for solving the subproblem (9), one can first compute F_s based on the singular value decomposition of the present point $\tilde{Q}_{s-1}$, then fix F_s and solve the following problem:

$$\tilde{Q}_s = \arg\min_Q \frac{\gamma_r}{2}\|Q - C_Q(Q_{r-1})\|_F^2 + \lambda_2 \mathrm{tr}(F_s^\top QQ^\top F_s).$$

Since both of the two terms are convex and differentiable regarding Q, the solution $\tilde{Q}_s$ can be obtained by simply setting the corresponding derivative to zero:

$$\tilde{Q}_s = \gamma_r(\gamma_r I + 2\lambda_2 F_s F_s^\top)^{-1} C_Q(Q_{r-1}). \quad (10)$$

The iterative procedure will converge to the solution of the subproblem (9).

In addition, when $k = 0$, (9) reduces to

$$Q_r = \arg\min_Q \frac{\gamma_r}{2}\|Q - C_Q(Q_{r-1})\|_F^2 + \lambda_2\|Q\|_F^2 \quad (11)$$

which provides an upper bound of the objective of problem (9). Therefore we can set the initial value $\tilde{Q}_0$ to $\frac{\gamma_r}{\gamma_r + 2\lambda_2} C_Q(Q_{r-1})$ by solving problem (11). The iterative procedure is summarized in Algorithm 1.

Algorithm 1 Iterative algorithm for computing Q_r

```
Input: $X^1, X^2, ..., X^m$, $Y^1, Y^2, ..., Y^m$, $\lambda_1$, $\lambda_2$, $P_{r-1}$, $Q_{r-1}$, $\gamma_r$.
Output: $Q_r$.
Initialize: $\tilde{Q}_0 = \frac{\gamma_r}{\gamma_r+2\lambda_2} C_Q(Q_{r-1})$, $s = 0$;
repeat
    $s = s + 1$;
    Obtain the smallest $(\min(d, m) - k)$ left singular vectors of $\tilde{Q}_{s-1}$ to get $F_s$;
    Obtain $\tilde{Q}_s$ according to (10);
until convergence;
$Q_r = \tilde{Q}_s$;
```

Moreover, PALM provides a scheme for step size estimation, which iteratively increases γ_r until the inequality

$$h(R_r) \le h(R_{r-1}) + \frac{\gamma_r}{2}\|R_r - R_{r-1}\|_F^2 + \langle R_r - R_{r-1}, \bigtriangledown h_R(R_{r-1})\rangle \quad (12)$$

is not satisfied. Specifically, given a multiplicative factor $L > 1$, γ_r is increased repeatedly by multiplying L until (12) is not satisfied.

The proximal method for CoCMTL is summarized in Algorithm 2. Convergence of the PALM scheme is analyzed in (Bolte, Sabach, and Teboulle 2013) (Lemma 3), where it is proved that the sequence $(R_1, \ldots, R_r, R_{r+1})$ is non-increasing.

Algorithm 2 Proximal algorithm for CoCMTL

```
Input: $X^1, X^2, ..., X^m$, $Y^1, Y^2, ..., Y^m$, $\lambda_1$, $\lambda_2$.
Output: $W$.
Initialize: $P_0$, $Q_0$, $\gamma_0$, $L > 1$, $r = 1$;
repeat
    $\gamma_r = \gamma_{r-1}$;
    while (12) is satisfied do
        $P_r = C_P(P_{r-1})$;
        Compute $Q_r$ by Algorithm 1;
        $\gamma_r = \gamma_r L$;
    end while;
    $r = r + 1$;
until convergence;
$W = P_r + Q_r$;
```

Experiments

In this section, we evaluate the proposed approach of multi-task learning with task-feature co-clusters (CoCMTL) in comparison with single task learning methods as well as representative multi-task learning algorithms. The experiments are first conducted in a synthetic setting, and then on two real-world data sets.

Table 1: Performance of the various algorithms in terms of nMSE on the Synthetic data. Methods with the best and comparable performance (measured by paired t-tests at 95% significance level) are bolded.

	Training Ratio	Ridge	L21	Low Rank	rMTL	rMTFL	Dirty	Flex-Clus	CMTL	CoCMTL
S1	10%	0.3620	0.4701	0.2654	0.2824	0.3608	0.4709	**0.2213**	0.2766	0.2306
	20%	0.2510	0.2280	0.1869	0.2341	0.2652	0.2353	**0.1536**	0.1758	**0.1541**
	30%	0.1528	0.1521	0.1378	0.1542	0.1623	0.1465	**0.1235**	0.1338	**0.1229**
S2	10%	0.3554	0.3563	0.1361	0.1577	0.3333	0.2414	0.1242	0.2157	**0.1048**
	20%	0.1946	0.2057	0.1442	0.1960	0.2729	0.1750	0.1174	0.1662	**0.1031**
	30%	0.1426	0.1511	0.1267	0.1520	0.1737	0.1364	0.1097	0.1344	**0.0991**
S3	10%	0.4231	0.5197	0.2139	0.2457	0.4239	0.5637	0.3218	0.3132	**0.1688**
	20%	0.2525	0.3218	0.2003	0.2705	0.3133	0.4963	0.2207	0.2332	**0.1480**
	30%	0.1875	0.2152	0.1730	0.2061	0.2155	0.2437	0.1751	0.1868	**0.1383**
S4	10%	0.4032	0.4980	0.1912	0.2186	0.3891	0.5004	0.3290	0.2929	**0.1516**
	20%	0.2280	0.2883	0.1806	0.2465	0.2859	0.4509	0.2094	0.2080	**0.1320**
	30%	0.1675	0.1897	0.1529	0.1809	0.1900	0.2109	0.1587	0.1636	**0.1223**

Table 2: Performance of various algorithms in terms of nMSE, aMSE and rMSE on the School data. Methods with the best and comparable performance (measured by paired t-tests at 95% significance level) are bolded.

	Training Ratio	Ridge	L21	Low Rank	rMTL	rMTFL	Dirty	Flex-Clus	CMTL	CoCMTL
nMSE	10%	1.1031	1.0931	0.9693	0.9603	1.3838	1.1421	0.8862	0.9914	**0.8114**
	20%	0.9178	0.9045	0.8435	0.8198	1.0310	0.9436	0.7891	0.8462	**0.7688**
	30%	0.8511	0.8401	0.8002	0.7833	0.9103	0.8517	**0.7634**	0.8064	**0.7515**
aMSE	10%	0.2891	0.2867	0.2541	0.2515	0.3618	0.2983	0.2315	0.2593	**0.2118**
	20%	0.2385	0.2368	0.2207	0.2147	0.2702	0.2470	0.2062	0.2214	**0.2009**
	30%	0.2212	0.2197	0.2091	0.2049	0.2378	0.2225	**0.1992**	0.2107	**0.1961**
rMSE	10%	11.5321	11.5141	11.2000	11.1984	12.1233	11.6401	10.9991	11.2680	**10.7430**
	20%	10.7318	10.7011	10.5427	10.4866	10.9928	10.8033	10.3986	10.5500	**10.3110**
	30%	10.1831	10.1704	10.0663	10.0291	10.3338	10.1956	**9.9767**	10.0865	**9.9221**

Competing Algorithms and Measurement

In our experiments, we evaluate the proposed CoCMTL approach with other multi-task algorithms. Representative multi-task learning algorithms including the L21 formulation (Argyriou, Evgeniou, and Pontil 2007), low rank method (Ji and Ye 2009), robust multi-task learning (rMTL) (Chen, Zhou, and Ye 2011), robust multi-task feature learning (rMTFL) (Gong, Ye, and Zhang 2012), dirty model (Dirty) (Jalali, Ravikumar, and Sanghavi 2010) and convex multi-task learning with flexible task Clusters (Flex-Clus) (Zhong and Kwok 2012) are compared against. Moreover, least squares ridge regression provides single task learning baselines, whereas low rank and rMTL are both trace norm based methods, additionally rMTL, rMTFL and Flex-Clus are examples of decomposition models similar to the proposed algorithm. CMTL (Jacob, Bach, and Vert 2008) and Flex-Clus are the representatives of task clustering multi-task learning algorithms. For CMTL and the proposed CoCMTL, k is treated as a hyper-parameter. In the experiments, the hyper-parameters are tuned by 3-fold cross validation. All algorithms are implemented with MATLAB. The maximum number of iterations is set to 5000 for all algorithms, with tolerance of 10^{-5}.

To evaluate the performance, the normalized mean squared error (nMSE), the averaged mean squared error (aMSE) and the root mean squared error (rMSE) are employed. Note that nMSE is defined as the mean squared error (MSE) divided by the variance of the target vector; aMSE is defined as MSE divided by the squared norm of the target vector. A smaller value of nMSE, aMSE and rMSE represents better regression performance.

Synthetic Dataset

For the synthetic data set, all samples have 80 features. For the i-th task, we randomly generate 400 samples with $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, I)$ and $y^i \sim \mathbf{x}\mathbf{w}^i + \mathcal{N}(0, 10)$. 100 tasks and the corresponding weight vectors are generated according to the following scenarios:

S1: All tasks are independent – $\mathbf{w}^i \sim \mathcal{U}(\mathbf{0}, \mathbf{10})$, where $\mathcal{U}$ denotes the uniform distribution.

S2: All tasks are similar, which means they are grouped in a major cluster – $\mathbf{w}^i \sim \mathcal{N}(\boldsymbol{\mu}, I)$ and $\boldsymbol{\mu} \sim \mathcal{U}(\mathbf{0}, \mathbf{10})$. In this case k is 1 .

S3: Tasks are clustered into 4 groups, each of which contains 25 tasks. Weight vectors in the c-th group are generated by $\mathcal{N}(\boldsymbol{\mu}^c, I)$, and $\boldsymbol{\mu}^c \sim \mathcal{U}(\mathbf{0}, \mathbf{10})$. For CMTL and CoCMTL k is set to 4.

S4: Tasks are separated into 4 groups and each group contains a subset of tasks as well as the corresponding subset of features, which simulates the block structure of clustering. Specifically, $\mathbf{w}^i = \mathbf{p}^i + \mathbf{q}^i$ where $\mathbf{p}^i \sim \mathcal{N}(\boldsymbol{\mu}, I)$, $\boldsymbol{\mu} \sim \mathcal{U}(\mathbf{0}, \mathbf{5})$, and $\mathbf{q}^i$ in each group is generated according to the rule similar to that of $\mathbf{w}^i$ in **S3** but only using a subset of features. The ground truth of the structure of W is shown in Figure 3. Again, k for CMTL and CoCMTL is set to 4.

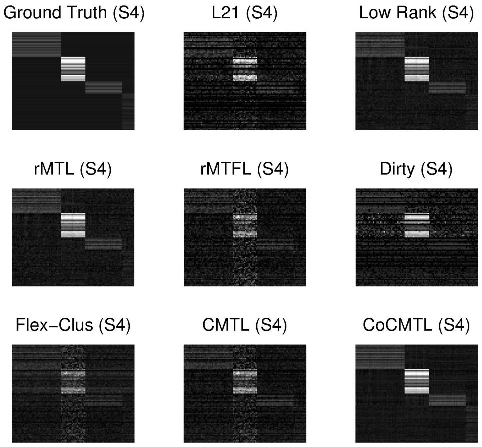

Figure 3: Recovery of W by different algorithms in **S4** which simulates the two-sided block structure. The training ratio is 10% here.

10%, 20%, and 30% of the samples from each task are randomly selected as training sets and the rest are used as test sets. We measure the performance on the synthetic data with nMSE and average the results after 20 repetitions which are reported in Table 1.

Table 1 shows that in **S1**, the proposed CoCMTL is comparable to the other methods, while being significantly superior in the cases of **S2** and **S3**, especially when the training examples are insufficient (with training ratio of 10%). More importantly, it can well capture the underlying structure in the scenario **S4** with task-feature co-cluster structure.

Figure 3 shows the comparison of the predicted weight matrices W produced by all the algorithms with the ground truth in scenario **S4**. One can observe that in the scenario **S4** where tasks are related on subsets of features, CoCMTL can recover the weight matrices very nicely.

School Data

The School data consists of the exam scores of 15362 students from 139 secondary schools in London during the years of 1985-1987; each student is described by 27 attributes including gender, ethnic group, etc. The exam score prediction of the students can be regarded as a multi-task regression problem with 139 tasks (schools), where each task has a different number of instances (students).

Here we follow the experimental setup as in (Chen, Zhou, and Ye 2011) and randomly select 10%, 20%, and 30% of the samples from each task as the training data and use the rest as the test data. The experimental results are averaged over 20 repetitions and summarized in Table 2.

From Table 2 it can be observed that the multi-task learning algorithms do improve the predictive performance significantly over the independent ridge regression algorithm, especially in the lack of training samples such as the cases with training ratio 10%. This justifies the motivation of learning multiple tasks simultaneously. Among all the algorithms, the proposed CoCMTL approach is superior to the other methods on School Data in terms of the three performance measures, namely nMse, aMSE and rMSE. More importantly, when comparing the three clustering based algorithms, one can observe the clear advantage of task-feature co-clustering (CoCMTL) over one-sided clustering (CMTL) and feature-wise clustering (Flex-Clus).

Computer Survey Data

We next experiment with another data set - Computer Survey Data (Argyriou, Evgeniou, and Pontil 2008), which consists of ratings of 201 students on 20 different personal computers. The input is represented with 13 binary attributes including telephone hot line, amount of memory, etc. Here the students correspond to tasks and the computer models correspond to instances. Invalid ratings and students with more than 8 zero ratings are removed, leaving the tasks of 172 students. Due to the lack of instances, we do not sample the training set with different ratios in this experiment, instead, we randomly split the 20 instances into training, validation and test sets with sizes of 8, 8, 4 respectively. Results are averaged over 20 random repetitions and presented in Table 3, which indicates the proposed CoCMTL approach outperforms other MTL methods again.

Table 3: Performance of various algorithms on Computer Survey Data. Methods with the best and comparable performance (measured by paired t-tests at 95% significance level) are bolded.

Algorithm	nMse	aMse	rMse
Ridge	2.4529	1.4893	2.1744
L21	2.1912	1.3706	2.0764
Low Rank	2.2095	1.3873	2.0630
rMTL	2.2118	1.3863	2.0655
rMTFL	2.2343	1.4004	2.0839
Dirty	2.4839	1.5910	2.1188
Flex-Clus	**2.0994**	**1.3197**	2.0774
CMTL	**2.0976**	**1.2840**	2.0775
CoCMTL	**1.8588**	**1.2047**	**1.9644**

Conclusion

In this paper, we study the task-feature relationships in multi-task learning. Based on the intuitive motivation that tasks should be related to subsets of features, we exploit the co-cluster structure of the task-feature relationships and present a novel co-clustered multi-task learning method (CoCMTL). The proposed approach is formulated as a decomposition model which separates the global similarities and group-specific similarities. To capture the group specific similarities, unlike the traditional task clustering approaches with only one-way clustering, we impose a novel regularization term which leads to a block structure. Experiments on both synthetic and real data have verified the effectiveness of CoCMTL, which offers consistently better performance than several state-of-the-art multi-task learning algorithms. In future work we are interested in investigations of other principles to enforce a co-cluster structure in the task-feature relationships, as well as the optimization techniques.

Acknowledgment

Research supported by the National Natural Science Foundation of China (No. 61375060), the Fundamental Research Funds for the Central Universities (WK0110000036), and the National Science Foundation for Distinguished Young Scholars of China (No. 61325010).

References

Argyriou, A.; Evgeniou, T.; and Pontil, M. 2007. Multitask feature learning. In *Advances in Neural Information Processing Systems 19*. MIT Press.

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

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

Bolte, J.; Sabach, S.; and Teboulle, M. 2013. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming* 4:1–36.

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

Chen, J.; Zhou, J.; and Ye, J. 2011. 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*, 42–50.

Dhillon, I. S.; Mallela, S.; and Modha, D. S. 2003. Information-theoretic co-clustering. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03, 89–98.

Dhillon, I. S. 2001. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, 269–274.

Ding, C.; Li, T.; Peng, W.; and Park, H. 2006. Orthogonal nonnegative matrix tri-factorizations for clustering. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 126–135.

Ding, C.; He, X.; and Simon, H. D. 2005. On the equivalence of nonnegative matrix factorization and spectral clustering. In *in SIAM International Conference on Data Mining*.

Evgeniou, T., and Pontil, M. 2004. Regularized multi-task learning. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, 109–117.

Gong, P.; Ye, J.; and Zhang, C. 2012. Robust multi-task feature learning. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, 895–903.

Hu, Y.; Zhang, D.; and Ye, J. 2012. Fast and accurate matrix completion via truncated nuclear norm regularization. In *in IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Jacob, L.; Bach, F.; and Vert, J.-P. 2008. Clustered multi-task learning: A convex formulation. In *Advances in Neural Information Processing Systems*, 745–752.

Jalali, A.; Ravikumar, P.; and Sanghavi, S. 2010. A dirty model for multi-task learning. In *Advances in Neural Information Processing Systems*.

Ji, S., and Ye, J. 2009. An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th Annual International Conference on Machine Learning*, 457–464.

Kang, Z.; Grauman, K.; and Sha, F. 2011. Learning with whom to share in multi-task feature learning. In *Proceedings of the 28th Annual International Conference on Machine Learning*, 521–528.

Li, T., and Ding, C. 2006. The relationships among various nonnegative matrix factorization methods for clustering. In *Proceedings of the 6th International Conference on Data Mining*, ICDM '06, 362–371.

Satuluri, V., and Parthasarathy, S. 2011. Symmetrizations for clustering directed graphs. In *Proceedings of the 14th International Conference on Extending Database Technology, ACM*, 362–371.

Shen, X., and Huang, H. 2010. Grouping pursuit through a regularization solution surface. *Journal of the American Statistical Association* 105(490).

Thrun, S., and O'Sullivan, J. 1996. Discovering structure in multiple learning tasks: The tc algorithm. In *Proceedings of the 13th International Conference on Machine Learning*, 489–497.

Xu, B.; Bu, J.; Chen, C.; and Cai, D. 2012. An exploration of improving collaborative recommender systems via user-item subgroups. In *Proceedings of the 21st International Conference on World Wide Web*, WWW'12, 21–30.

Yang, S.; Yuan, L.; and Lai, Y. 2012. Feature grouping and selection over an undirected graph. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM2012, 922–930.

Zhang, Y., and Yeung, D.-Y. 2010. A convex formulation for learning task relationships in multi-task learning. In *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence*, 733–442.

Zhong, W., and Kwok, J. T. 2012. Convex multitask learning with flexible task clusters. In *Proceedings of the 29th International Conference on Machine Learning*.