Journal of Machine Learning Research 17 (2016) 1-30 Submitted 5/15; Revised 10/15; Published 4/16

Multi-task Sparse Structure Learning with Gaussian Copula Models

André R. Gonçalves ANDRERIC@DCA.FEE.UNICAMP.BR
Fernando J. Von Zuben VONZUBEN@DCA.FEE.UNICAMP.BR
School of Electrical and Computer Engineering
University of Campinas
São Paulo, Brazil

Arindam Banerjee BANERJEE@CS.UMN.EDU
Computer Science Department
University of Minnesota - Twin Cities
Minneapolis, USA

Editor: Urun Dogan, Marius Kloft, Francesco Orabona, and Tatiana Tommasi

Abstract

Multi-task learning (MTL) aims to improve generalization performance by learning multiple related tasks simultaneously. While sometimes the underlying task relationship structure is known, often the structure needs to be estimated from data at hand. In this paper, we present a novel family of models for MTL, applicable to regression and classification problems, capable of learning the structure of tasks relationship. In particular, we consider a joint estimation problem of the tasks relationship structure and the individual task parameters, which is solved using alternating minimization. The task relationship revealed by structure learning is founded on recent advances in Gaussian graphical models endowed with sparse estimators of the precision (inverse covariance) matrix. An extension to include flexible Gaussian copula models that relaxes the Gaussian marginal assumption is also proposed. We illustrate the effectiveness of the proposed model on a variety of synthetic and benchmark data sets for regression and classification. We also consider the problem of combining Earth System Model (ESM) outputs for better projections of future climate, with focus on projections of temperature by combining ESMs in South and North America, and show that the proposed model outperforms several existing methods for the problem.

Keywords: multi-task learning, structure learning, Gaussian copula, probabilistic graphical model, sparse modeling

1. Introduction

In multi-task learning (MTL) one can benefit from the knowledge of the underlying structure relating the learning tasks while carrying them out simultaneously. In situations where some tasks might be highly dependent on each other, the strategy of isolating each task will not be helpful in exploiting the potential information one might acquire from other related tasks. The last few years experienced an increase of activity in this area where new methods and applications have been proposed. From the methods perspective, there have been contributions devoted to novel formulations to describe task structure and to incorporate them into the learning framework (Evgeniou and Pontil, 2004; Ji and Ye, 2009;

Kim and Xing, 2010; Kumar and Daume III, 2012; Yang et al., 2013). Meanwhile MTL has been applied to problems ranging from object detection in computer vision, going through web image and video search (Wang et al., 2009), and achieving multiple microarray data set integration in computational biology (Widmer and Rätsch, 2012).

Much of the existing work in MTL assumes the existence of a priori knowledge about the task relationship structure (see Section 2). However, in many problems there is only a high level understanding of those relationships, and hence the structure of the task relationship needs to be estimated from the data. Recently, there have been attempts to explicitly model the relationship and incorporate it into the learning process (Zhang and Yeung, 2010; Zhang and Schneider, 2010; Yang et al., 2013). In the majority of these methods, the tasks dependencies are represented as unknown hyper-parameters in hierarchical Bayesian models and are estimated from the data. As will be discussed in Section 2, many of these methods are either computationally expensive or restrictive on dependence structure complexity.

In *structure learning*, we estimate the (conditional) dependence structures between random variables in a high-dimensional distribution, and major advances have been achieved in the past few years (Banerjee et al., 2008; Friedman et al., 2008; Cai et al., 2011; Wang et al., 2013). In particular, assuming sparsity in the conditional dependence structure, i.e., each variable is dependent only on a few others, there are estimators based on convex (sparse) optimization which are guaranteed to recover the correct dependence structure with high probability, even when the number of samples is small compared to the number of variables.

In this paper, we present a family of models for MTL, for regression and classification problems, which are capable of learning the structure of task relationships and parameters for individual tasks. The problem is posed as a joint estimation where parameters of the tasks and relationship structure are learned using alternating minimization. This paper is an extension of our early work (Gonçalves et al., 2014), as it further includes improvements on the task relationship modeling and can now handle a wider spectrum of problems.

The relationship structure is modeled by either imposing a prior over the features across tasks (Section 3.3) or assuming correlated residuals (Section 3.7). We can use of a variety of methods from the structure learning literature to estimate the relationships. The formulation can be extended to Gaussian copula models (Liu et al., 2009; Xue and Zou, 2012), which are more flexible as it does not rely on strict Gaussian assumptions and has shown to be more robust to outliers. The resulting estimation problems are solved using suitable first order methods, including proximal updates (Beck and Teboulle, 2009) and alternating direction method of multipliers (Boyd et al., 2011). Based on our modeling, we show that MTL can benefit from advances in the structure learning area. Moreover, any future development in the area can be readily used in the context of MTL.

The proposed Multi-task Sparse Structure Learning (MSSL) approach has important practical implications: given a set of tasks, one can just feed the data from all the tasks without any knowledge or guidance on task relationship, and MSSL will figure out which tasks are related and will also estimate task specific parameters. Through experiments on a wide variety of data sets for multi-task regression and classification, we illustrate that MSSL is competitive with and usually outperforms several baselines from the existing MTL literature. Furthermore, the task relationships learned by MSSL are found to be accurate and consistent with domain knowledge on the problem.

In addition to evaluation on synthetic and benchmark data sets, we consider the problem of predicting air surface temperature in South and North America. The goal here is to combine outputs from Earth System Models (ESMs) reported by various countries to the Intergovernmental Panel on Climate Change (IPCC), where the regression problem at each geographical location forms a task. The weight on each model at each location forms the "skill" of that model, and the hope is that outputs from skillful models in each region can be more reliable for future projections of temperature. MSSL is able to identify geographically nearby regions as related tasks, which is meaningful for temperature prediction, without any previous knowledge of the spatial location of the tasks, and outperforms baseline approaches.

The remainder of the paper is structured as follows. Section 2 briefly discusses the related work in multi-task learning. Section 3 presents an overview and gentle introduction to the proposed multi-task sparse structure learning (MSSL) approach. Section 3.3 discusses a specific form of MSSL where the task structure dependence is learned based on the task coefficients. The MSSL is extended to the Gaussian copula MSSL in Section 3.6. Section 3.7 discusses another specific form of MSSL where the task structure dependence is learned based on the task residuals. Section 4 presents experimental results on regression and classification using synthetic, benchmark, and climate data sets. We conclude in Section 5.

Notation. We denote by m the number of tasks, d the problem dimension, supposed to be the same for all learning tasks, and n_k the number of samples for the k-th task. $\mathbf{X}_k \in \mathbb{R}^{n_k \times d}$ and $\mathbf{y}_k \in \mathbb{R}^{n_k \times 1}$ are the input and output data for the k-th task. Let $\mathbf{W} \in \mathbb{R}^{d \times m}$ be the parameter matrix, where columns are vector parameters $\mathbf{w}_k \in \mathbb{R}^d$, $k = 1, ..., m$, for the tasks. $(x)_+ = \max(0, x)$. Let $\mathcal{S}_+^p$ be the set of $p \times p$ positive semidefinite matrices. For any matrix $\mathbf{A}$, $\mathrm{tr}(\mathbf{A})$ is the trace operator, $\|\mathbf{A}\|_1$ and $\|\mathbf{A}\|_F$ are the ℓ_1-norm and Frobenius norm of $\mathbf{A}$, respectively. $\mathbf{A} \circ \mathbf{B}$ denotes the Hadamard (element-wise) product of the matrices $\mathbf{A}$ and $\mathbf{B}$. $\mathbf{I}_p$ is the $p \times p$ identity matrix and $\mathbf{0}_{p \times p}$ is a matrix full of zeros. For an m-variate random variable $V = (V_1, ..., V_m)$, we denote by $V_{\setminus\{i,j\}}$ the set of marginals except i and j.

2. Related Work

MTL has attracted a great deal of attention in the past few years and consequently many algorithms have been proposed (Evgeniou and Pontil, 2004; Argyriou et al., 2007; Xue et al., 2007; Jacob et al., 2008; Obozinski et al., 2010; Zhou et al., 2011b; Zhang and Yeung, 2010; Yang et al., 2013; Gonçalves et al., 2014). We will present a general view of the methods and discuss in more details those that are more related to ours.

The majority of the proposed methods fall into the class of regularized multi-task learning, which has the form

$$\min_{\mathbf{W}} \sum_{k=1}^{m} \left(\sum_{i=1}^{n_k} \ell\left(f(\mathbf{x}_k^i, \mathbf{w}_k), y_k^i\right) \right) + \mathcal{R}(\mathbf{W}),$$

where $\ell(\cdot)$ is the loss function such as squared, logistic, and hinge loss; $\mathcal{R}(\mathbf{W})$ is a regularization function for $\mathbf{W}$ that can be designed to enforce some sharing of information between tasks. In such a context, the goal of MTL is to estimate the tasks parameters $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_m]$, while taking into account the underlying relationship among tasks.

The existing methods basically differ in the way the regularization $\mathcal{R}(\mathbf{W})$ is designed, including the structural constraints imposed to matrix $\mathbf{W}$ and the relationship among the

tasks. Some methods assume a fixed structure a priori, while others try to estimate it from the data. In the following we present a representative set of methods from these categories.

2.1 MTL with All Tasks Related

One class of MTL methods assumes that all tasks are related and the information about tasks are selectively shared among all tasks, with the hypothesized structure of the parameter matrix $\mathbf{W}$ controlling how the information is shared.

Evgeniou and Pontil (2004) considered the scenario that all tasks are related in a way that the model parameters are close to some mean model. Motivated by the sparsity inducing property of the ℓ_1-norm (Tibshirani, 1996), the idea of structured sparsity has been widely explored in MTL algorithms. Argyriou et al. (2007) assumed that there exists a subset of features that is shared for all the tasks and imposed an $\ell_{2,1}-$norm penalization on the matrix $\mathbf{W}$ to select such set of features. In the dirty-model proposed in Jalali et al. (2010) the matrix $\mathbf{W}$ is modeled as the sum of a group sparse and an element-wise sparse matrix. The sparsity pattern is imposed by ℓ_q and ℓ_1-norm regularizations. Similar decomposition was assumed in Chen et al. (2010), but there $\mathbf{W}$ is a sum of an element-wise sparse (ℓ_1) and a low-rank (nuclear norm) matrix. The assumption that a low-dimensional subspace is shared by all tasks is explored in Ando et al. (2005), Chen et al. (2009), and Obozinski et al. (2010). For example, in Obozinski et al. (2010) a trace norm regularization on $\mathbf{W}$ was used to select the common low-dimensional subspace.

2.2 MTL with Cluster Assumption

Another class of MTL methods assumes that not all tasks are related, but instead the relatedness is in a group (cluster) structure, that is, mutually related tasks are in the same cluster, while unrelated tasks belong to different clusters. Information is shared only by those tasks belonging to the same cluster. The problem then involves estimating the number of clusters and the matrix encoding the assignment cluster information.

In Bakker and Heskes (2003) task clustering was enforced by considering a mixture of Gaussians as a prior over task parameters. Evgeniou et al. (2005) proposed a task clustering regularization to encode cluster information in the MTL formulation. Xue et al. (2007) employed a Dirichlet process prior over the task coefficients to encourage task clustering and the number of clusters was someway automatically determined by the prior.

2.3 MTL with Dependence Structure Learning

Recently, there have been some proposals to estimate and incorporate the dependence among the tasks into the learning process. These methods are the most related to ours.

A matrix-variate normal distribution was used as a prior for $\mathbf{W}$ matrix in Zhang and Yeung (2010). The hyper-parameter for such a prior distribution captures the covariance matrix ($\mathbf{\Sigma}$) among all task coefficients. The resulting non-convex maximum a posteriori problem is relaxed by restricting the model complexity. It has a positive side of making the whole problem convex, but has the downside of significantly restricting the flexibility of the task relatedness structure. Also, in Zhang and Yeung (2010), the task relationship is modeled by the covariance among tasks, but uses the inverse (precision matrix, $\mathbf{\Sigma}^{-1} = \mathbf{\Omega}$)

in the task parameter learning step, therefore, the inverse of the covariance matrix needed to be computed at every iteration. We, on the other hand, do not constrain the complexity of our model and also learn the inverse of the covariance matrix directly, which tends to be more stable than computing covariance and then inverting it.

Zhang and Schneider (2010) also used a matrix-variate normal prior over $\mathbf{W}$. The two matrix hyper-parameters explicitly represent the covariance among the features (assuming the same feature relationships in all tasks) and covariance among the tasks, respectively. Sparse inducing penalization on the inverse covariance $\mathbf{\Omega}$ of both is added into the formulation. Unlike Zhang and Yeung (2010), both matrices are learned in an alternating minimization algorithm and can be computationally prohibitive in high dimensional problems due to the cost of modeling and estimating the feature covariance.

Yang et al. (2013) also assumed a matrix normal prior for $\mathbf{W}$. However, the row and column covariance hyperparameters have a Matrix Generalized Inverse Gaussian (MGIG) prior distribution. The mean of matrix $\mathbf{W}$ is factorized as the product of two matrices that also has matrix-variate normal distribution as a prior. The model inference is done via a variational Expectation Maximization (EM) algorithm. Due to the lack of a closed form expression to compute statistics of the MGIG distribution, the method resort to the use of sampling techniques, which can be slow for high-dimensional problems.

Rothman et al. (2010) also enforced sparsity on both $\mathbf{W}$ and $\mathbf{\Omega}$. Similar to our residual-based MSSL formulation, it differs in two aspects: (i) our formulation allows a richer class of conditional distribution $p(y|\mathbf{x})$, namely distributions in the exponential family, rather than simply Gaussian; and (ii) we employ a semiparametric Gaussian copula model to capture task relationship, which does not rely of Gaussian assumption on the marginals and have shown to be more robust to outliers (Liu et al., 2012), then traditional Gaussian model used in Rothman et al. (2010). As will be seen in the experiments, the MSSL method with copula models produced more accurate predictions. Rai et al. (2012) extended the formulation in Rothman et al. (2010) to model feature dependence, additionally to the task dependence modeling. However, it is computationally prohibitive for high-dimensional problems, due to the cost of estimating another precision matrix for feature dependence.

Zhou and Tao (2014) used copula as a richer class of conditional marginal distributions $p(y_k|\mathbf{x})$. As copula models express the joint distribution $p(\mathbf{y}|\mathbf{x})$ from the set of marginal distributions, this formulation allows marginals to have arbitrary continuous distributions. Output correlation is exploited via the sparse inverse covariance in the copula function, which is estimated by a procedure based on proximal algorithms. Our method also covers a rich class of conditional distributions, the exponential family that includes Gaussian, Bernoulli, Multinomial, Poisson, and Dirichlet, among others. We use Gaussian copula models to capture tasks dependence, instead of explicitly modeling marginal distributions.

3. Multi-task Sparse Structure Learning

In this section we describe our multi-task Sparse Structure Learning (MSSL) method. As our modeling is founded on structure estimation in Gaussian graphical models, we first introduce the associated problem before presenting the proposed method.

3.1 Structure Estimation in Gaussian Graphical Models

Here we describe the undirected graphical model used to capture the underlying linear dependence structure of our multi-task learning framework.

Let $V = (V_1, \ldots, V_m)$ be an m-variate random vector with joint distribution $p(V)$. Such distribution can be characterized by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the vertex set $\mathcal{V}$ represents the m covariates of V and edge set $\mathcal{E}$ represents the conditional dependence relations between the covariates of V. If V_i is conditionally independent of V_j given the other variables, then the edge (i, j) is not in $\mathcal{E}$. Assuming $V \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, the missing edges correspond to zeros in the inverse covariance matrix or *precision* matrix given by $\mathbf{\Sigma}^{-1} = \mathbf{\Omega}$, i.e., $(\mathbf{\Sigma}^{-1})_{ij} = 0 \; \forall (i, j) \notin E$ (Lauritzen, 1996).

Classical estimation approaches (Dempster, 1972) work well when m is small. Given, that we have n i.i.d. samples $v_1, \ldots, v_n$ from the distribution, the empirical covariance matrix is $\hat{\mathbf{\Sigma}} = \frac{1}{n} \sum_{i=1}^{n} (v_i - \bar{v})^\top (v_i - \bar{v})$, where $\bar{v} = \frac{1}{n} \sum_{i=1}^{n} v_i$. However, when $m > n$, $\hat{\mathbf{\Sigma}}$ is rank-deficient and its inverse cannot be used to estimate the precision matrix $\mathbf{\Omega}$. Nonetheless, for a sparse graph, i.e. most of the entries in the precision matrix are zero, several methods exist to estimate $\mathbf{\Omega}$ (Friedman et al., 2008; Boyd et al., 2011).

3.2 MSSL Formulation

For ease of exposition, let us consider a simple linear model for each task: $\mathbf{y}_k = \mathbf{X}_k \mathbf{w}_k + \boldsymbol{\xi}_k$ where $\mathbf{w}_k$ is the parameter vector for task k and $\boldsymbol{\xi}_k$ denotes the residual error. The proposed MSSL method estimates both the task parameters $\mathbf{w}_k$ for all tasks and the structure dependence, based on some information from each task. Further, the dependence structure is used as inductive bias in the $\mathbf{w}_k$ learning process, aiming at improving the generalization capability of the tasks.

We investigate and formalize two ways of learning the relationship structure (a graph indicating the relationship among the tasks), represented by $\mathbf{\Omega}$: (a) modeling $\mathbf{\Omega}$ from the task specific parameters $\mathbf{w}_k, \forall k = 1, ..., m$ and (b) modeling $\mathbf{\Omega}$ from the residual errors $\boldsymbol{\xi}_k, \forall k = 1, ..., m$. Based on how we model $\mathbf{\Omega}$, we propose p-MSSL (from tasks parameters) and r-MSSL (from residual error). Both models are discussed in the following sections.

At a high level, the estimation problem in such MSSL approaches takes the form:

$$\min_{\mathbf{W}, \mathbf{\Omega} \succ 0} \; \mathcal{L}((\mathbf{Y}, \mathbf{X}), \mathbf{W}) + \mathcal{B}(\mathbf{W}, \mathbf{\Omega}) + \mathcal{R}_1(\mathbf{W}) + \mathcal{R}_2(\mathbf{\Omega}), \tag{1}$$

where $\mathcal{L}(\cdot)$ denotes suitable task specific loss function, $\mathcal{B}(\cdot)$ is the inductive bias term, and $\mathcal{R}_1(\cdot)$ and $\mathcal{R}_2(\cdot)$ are suitable sparsity inducing regularization terms. The interaction between parameters $\mathbf{w}_k$ and the relationship matrix $\mathbf{\Omega}$ is captured by the $\mathcal{B}(\cdot)$ term. Notably, when $\mathbf{\Omega}_{k,k'} = 0$, the parameters $\mathbf{w}_k$ and $\mathbf{w}_{k'}$ have no influence on each other. Sections 3.3 to 3.7 delineate the modeling details behind MSSL algorithms and how it leads to the solution of the optimization problem in (1).

3.3 Parameter Precision Structure

If the tasks are unrelated, one can learn the columns of the coefficient matrix $\mathbf{W}$ independently for each of the m tasks. However, when there exist relationships among the m tasks, learning the columns of $\mathbf{W}$ independently fails to capture these dependencies. In such a

scenario, we propose to use the precision matrix $\mathbf{\Omega} \in \mathbb{R}^{m \times m}$ in order to capture pairwise partial correlations between tasks.

In the parameter precision structure based MSSL (p-MSSL) model we assume that features across tasks (*rows* $\hat{\mathbf{w}}_j$ of the matrix $\mathbf{W}$) follows a multivariate Gaussian distribution with zero mean and covariance matrix $\mathbf{\Sigma}$, i.e., $\hat{\mathbf{w}}_j \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})\ \forall j = 1, ..., d$, where $\mathbf{\Sigma}^{-1}$=$\mathbf{\Omega}$. The problem of interest is to estimate both the parameters $\mathbf{w}_1, \ldots, \mathbf{w}_m$ and the precision matrix $\mathbf{\Omega}$. By imposing such a prior over the *rows* of $\mathbf{W}$, we are capable of explicitly estimating the dependency structure among the tasks via the precision matrix $\mathbf{\Omega}$.

With a multivariate Gaussian prior over the *rows* of $\mathbf{W}$, its posterior can be written as

$$p\left(\mathbf{W}|(\mathbf{X}, \mathbf{Y}), \mathbf{\Omega}\right) \propto \prod_{k=1}^{m} \prod_{i=1}^{n_k} p\left(y_k^i | \mathbf{x}_k^i, \mathbf{w}_k^\top\right) \prod_{j=1}^{d} p\left(\hat{\mathbf{w}}_j | \mathbf{\Omega}\right), \tag{2}$$

where the first term in the right hand side denotes the conditional distribution of the response given the input and parameters, and the second term denotes the prior over *rows* of $\mathbf{W}$. In this paper, we consider the *penalized* maximization of (2), assuming that the parameter matrix $\mathbf{W}$ and the precision matrix $\mathbf{\Omega}$ are sparse, i.e., contain few non-zero elements. In the following, we provide two specific instantiations of this model. First, we consider a Gaussian conditional distribution, wherein we obtain the well known least squares regression problem (Section 3.3.1). Second, for discrete labeled data, choosing a Bernoulli conditional distribution leads to a logistic regression problem (Section 3.3.2).

3.3.1 Least Squares Regression

Assume that

$$p\left(y_k^i \middle| \mathbf{x}_k^i, \mathbf{w}_k\right) = \mathcal{N}_1\left(y_k^i \middle| \mathbf{w}_k^\top \mathbf{x}_k^i, \sigma_k^2\right),$$

where it is considered for ease of exposition that the variance of the residuals $\sigma_k^2 = 1, \forall k = 1, ..., m$, though it can be incorporated in the model and learned from the data. We can write this optimization problem as minimization of the negative logarithm of (2), which corresponds to a regularized linear regression problem

$$\min_{\mathbf{W}, \mathbf{\Omega} \succ 0} \quad \frac{1}{2} \sum_{k=1}^{m} \sum_{i=1}^{n_k} \left(\mathbf{w}_k^\top \mathbf{x}_k^i - y_k^i\right)^2 - \frac{d}{2} \log |\mathbf{\Omega}| + \frac{1}{2} \mathrm{tr}(\mathbf{W} \mathbf{\Omega} \mathbf{W}^\top).$$

Further, assuming that $\mathbf{\Omega}$ and $\mathbf{W}$ are sparse, we add ℓ_1-norm regularizers over both parameters. In the case one task has a much larger number of samples compared to the others, it may dominate the empirical loss term. To avoid such bias we modify the cost function and compute the weighted average of the empirical losses of the form

$$\min_{\mathbf{W}, \mathbf{\Omega} \succ 0} \quad \sum_{k=1}^{m} \frac{1}{n_k} \sum_{i=1}^{n_k} \left(\mathbf{w}_k^\top \mathbf{x}_k^i - y_k^i\right)^2 - d \log |\mathbf{\Omega}| + \lambda_0 \mathrm{tr}(\mathbf{W} \mathbf{\Omega} \mathbf{W}^\top) + \lambda_1 \|\mathbf{W}\|_1 + \lambda_2 \|\mathbf{\Omega}\|_1, \tag{3}$$

where λ_0, λ_1, and $\lambda_2 > 0$ are penalty parameters. The sparsity assumption on $\mathbf{W}$ is motivated by the fact that maybe some features are not relevant for discriminative purposes and can then be dropped out from the model. Precision matrix $\mathbf{\Omega}$ plays an important role in Gaussian graphical models because its zero entries precisely capture the conditional

independence, that is, $\mathbf{\Omega}_{ij} = 0$ if and only if $\mathbf{w}_i \perp\!\!\!\perp \mathbf{w}_j | \mathbf{W}_{\setminus\{i,j\}}$. Then, enforcing sparsity on $\mathbf{\Omega}$ will highlight the conditional independence among tasks parameters.

In this formulation, the term involving the trace of the outer product $\text{tr}(\mathbf{W}\mathbf{\Omega}\mathbf{W}^\top)$ affects the *rows* of $\mathbf{W}$, such that if $\mathbf{\Omega}_{ij} \neq 0$, then $\mathbf{w}_i$ and $\mathbf{w}_j$ are constrained to be similar.

Although the problem is not jointly convex on $\mathbf{W}$ and $\mathbf{\Omega}$, it is in fact biconvex, that is, fixing $\mathbf{\Omega}$ the problem is convex on $\mathbf{W}$, and vice-versa. So, the associated biconvex function in problem (3) is split into two convex functions exhibited in (4a) and (4b). Then, one can use an alternating optimization procedure that updates $\mathbf{W}$ and $\mathbf{\Omega}$ by fixing one of them and solving the corresponding convex optimization problem (Gorski et al., 2007), given by

$$f_{\mathbf{\Omega}}(\mathbf{W}; \mathbf{X}, \mathbf{Y}, \lambda_0, \lambda_1) = \sum_{k=1}^{m} \frac{1}{n_k} \sum_{i=1}^{n_k} (\mathbf{w}_k^\top \mathbf{x}_k^i - y_k^i)^2 + \lambda_0 \text{tr}(\mathbf{W}\mathbf{\Omega}\mathbf{W}^\top) + \lambda_1 \|\mathbf{W}\|_1, \tag{4a}$$

$$f_{\mathbf{W}}(\mathbf{\Omega}; \mathbf{X}, \mathbf{Y}, \lambda_0, \lambda_2) = \lambda_0 \text{tr}(\mathbf{W}\mathbf{\Omega}\mathbf{W}^\top) - d \log |\mathbf{\Omega}| + \lambda_2 \|\mathbf{\Omega}\|_1. \tag{4b}$$

The alternating minimization algorithm proceeds as described in Algorithm 1. The procedure is guaranteed to converge to a *partial optimum* Gorski et al. (2007), since the original problem (3) is biconvex and convex in each argument $\mathbf{\Omega}$ and $\mathbf{W}$.

Algorithm 1: Multitask Sparse Structure Learning (MSSL) algorithm

Data: $\{\mathbf{X}_k, \mathbf{y}_k\}_{k=1}^m$. `// training data for all tasks`
Input: $\lambda_0, \lambda_1, \lambda_2 > 0$. `// penalty parameters chosen by cross-validation`
Result: $\mathbf{W}$, $\mathbf{\Omega}$. `// estimated parameters`
begin
  `/* ` $\mathbf{\Omega}^0$ `is initialized with identity matrix and */`
  `/* ` $\mathbf{W}^0$ `with random numbers in [-0.5,0.5]. */`
  Initialize $\mathbf{\Omega}^0$ and $\mathbf{W}^0$
  $t = 1$
  repeat
    $\mathbf{W}^{(t+1)} = \underset{\mathbf{W}}{\text{argmin}} f_{\mathbf{\Omega}^{(t)}}(\mathbf{W})$ `// optimize` $\mathbf{W}$ `with` $\mathbf{\Omega}$ `fixed`
    $\mathbf{\Omega}^{(t+1)} = \underset{\mathbf{\Omega}}{\text{argmin}} f_{\mathbf{W}^{(t+1)}}(\mathbf{\Omega})$ `// optimize` $\mathbf{\Omega}$ `with` $\mathbf{W}$ `fixed`
    $t = t + 1$
  until *stopping condition met*
end

Update for W: The update step involving (4a) is an ℓ_1-regularized quadratic problem. Thus the problem is an ℓ_1-penalized quadratic optimization program, which we solve using established proximal gradient descent methods such as FISTA (Beck and Teboulle, 2009). The $\mathbf{W}$-step can be seen as a general case of the formulation in Subbian and Banerjee (2013) in the context of climate model combination, where in our proposal $\mathbf{\Omega}$ is any positive definite precision matrix, rather than a fixed Laplacian matrix as in Subbian and Banerjee (2013).

In the class of proximal gradient methods the cost function $h(x)$ is decomposed as $h(x) = f(x) + g(x)$, where $f(x)$ is a convex and smooth function and $g(x)$ is convex and

typically non-smooth. The accelerated proximal gradient iterates as follows

$$\begin{aligned} \mathbf{z}^{t+1} &:= \mathbf{w}_k^t + \omega^t \left(\mathbf{w}_k^t - \mathbf{w}_k^{t-1}\right) \\ \mathbf{w}_k^{t+1} &:= \mathbf{prox}_{\rho^t g}\left(\mathbf{z}^{t+1} - \rho^t \nabla f\left(\mathbf{z}^{t+1}\right)\right), \end{aligned} \tag{5}$$

where $\omega^t \in [0, 1)$ is an extrapolation parameter and ρ^t is the step size. The ω^t parameter is chosen as $\omega^t = (\eta_t - 1)/\eta_{t+1}$, with $\eta_{t+1} = (1 + \sqrt{1 + 4\eta_t^2})/2$ as done in Beck and Teboulle (2009) and ρ^t can be computed by a line search. The proximal operator associated with the ℓ_1-norm is the soft-thresholding operator

$$\mathbf{prox}_{\rho^t}(\mathbf{x})_i = (|x_i| - \rho^t)_+ \text{sign}(x_i) \tag{6}$$

The convergence rate of the algorithm is $\mathcal{O}(1/t^2)$ (Beck and Teboulle, 2009). Considering the squared loss, the gradient for the weights of the k-th task is computed as

$$\nabla f\left(\mathbf{w}_k\right) = \frac{1}{n_k}(\mathbf{X}_k^\top \mathbf{X}_k \mathbf{w}_k - \mathbf{X}_k^\top \mathbf{y}_k) + \lambda_0 \boldsymbol{\psi}_k, \tag{7}$$

where $\boldsymbol{\psi}_k$ is the k-th column of matrix $\boldsymbol{\Psi} = 2\mathbf{W}\boldsymbol{\Omega} = \frac{\partial}{\partial \mathbf{W}} \text{tr}(\mathbf{W}\boldsymbol{\Omega}\mathbf{W}^\top)$. Note that the first two terms of the gradient, which come from the loss function, are independent for each task and then can be computed in parallel.

Update for $\boldsymbol{\Omega}$: The update step for $\boldsymbol{\Omega}$ involving (4b) is known as the *sparse inverse covariance selection problem* and efficient methods have been proposed recently (Banerjee et al., 2008; Friedman et al., 2008; Boyd et al., 2011; Cai et al., 2011; Wang et al., 2013). Re-writing (4b) in terms of the sample covariance matrix $\mathbf{S}$, the minimization problem is

$$\min_{\boldsymbol{\Omega} \succ 0} \quad \lambda_0 \text{tr}(\mathbf{S}\boldsymbol{\Omega}) - \log|\boldsymbol{\Omega}| + \frac{\lambda_2}{d}\|\boldsymbol{\Omega}\|_1, \tag{8}$$

where $\mathbf{S} = \frac{1}{d}\mathbf{W}^\top \mathbf{W}$. This formulation will be useful to connect to the Gaussian copula extension in the next section. As λ_2 is a user defined parameter, the factor $\frac{1}{d}$ can be incorporated into λ_2.

To solve the minimization problem (8) we use an efficient Alternating Direction Method of Multiplies (ADMM) algorithm (Boyd et al., 2011). ADMM is a strategy that is intended to blend the benefits of dual decomposition and augmented Lagrangian methods for constrained optimization. It takes the form of a *decomposition-coordination* procedure, in which the solutions to small local problems are coordinated to find a solution to a large global problem. We refer interested readers to Boyd et al. (2011) in its Section 6.5 for details on the derivation of the updates.

In ADMM, we start by forming the augmented Lagrangian function of the problem (8)

$$L_\rho(\boldsymbol{\Theta}, \mathbf{Z}, \mathbf{U}) = \lambda_0 \text{tr}(\mathbf{S}\boldsymbol{\Theta}) - \log|\boldsymbol{\Theta}| + \lambda_2\|\mathbf{Z}\|_1 + \frac{\rho}{2}\|\boldsymbol{\Theta} - \mathbf{Z} + \mathbf{U}\|_F^2 - \frac{\rho}{2}\|\mathbf{U}\|_F^2, \tag{9}$$

where $\mathbf{U}$ is the scaled dual variable. Note that the non-smooth convex function (8) is split in two functions by adding an auxiliary variable $\mathbf{Z}$, besides a linear constraint $\boldsymbol{\Theta} - \mathbf{Z} = 0$.

Given the matrix $\mathbf{S}^{(t+1)} = \frac{1}{d}(\mathbf{W}^{(t+1)})^\top \mathbf{W}^{(t+1)}$ and setting $\mathbf{\Theta}^0 = \mathbf{\Omega}^{(t)}$, $\mathbf{Z}^0 = \mathbf{0}_{m\times m}$, and $\mathbf{U}^0 = \mathbf{0}_{m\times m}$, the ADMM for the problem (8) consists of the iterations:

$$\mathbf{\Theta}^{l+1} = \underset{\mathbf{\Theta}\succ 0}{\operatorname{argmin}} \quad \lambda_0 \operatorname{tr}(\mathbf{S}^{l+1}\mathbf{\Theta}) - \log|\mathbf{\Theta}| + \frac{\rho}{2}\|\mathbf{\Theta} - \mathbf{Z}^l + \mathbf{U}^l\|_F^2 \tag{10a}$$

$$\mathbf{Z}^{l+1} = \underset{\mathbf{Z}}{\operatorname{argmin}} \quad \lambda_2\|\mathbf{Z}\|_1 + \frac{\rho}{2}\|\mathbf{\Theta}^{l+1} - \mathbf{Z} + \mathbf{U}^l\|_F^2 \tag{10b}$$

$$\mathbf{U}^{l+1} = \mathbf{U}^l + \mathbf{\Theta}^{l+1} - \mathbf{Z}^{l+1}. \tag{10c}$$

The output of the ADMM is $\mathbf{\Omega}^{t+1} = \mathbf{\Theta}^L$, where L is the number of steps for convergence.

Each ADMM step can be solved efficiently. For the $\mathbf{\Theta}$-update, we can observe, from the first order optimality condition of (10a) and the implicit constraint $\mathbf{\Theta} \succ 0$, that the solution consists basically of a singular value decomposion.

The $\mathbf{Z}$-update (10b) can be computed in closed form, as follows

$$\mathbf{Z}^{l+1} = S_{\lambda_2/\rho}\Big(\mathbf{\Theta}^{l+1} + \mathbf{U}^l\Big), \tag{11}$$

where $S_{\lambda_2/\rho}(\cdot)$ is an element-wise soft-thresholding operator (Boyd et al., 2011). Finally, the updates for $\mathbf{U}$ in (10c) are already in closed form.

3.3.2 Log Linear Models

As described previously, our model can also be applied to classification. Let us assume that

$$p\left(y_k^i \middle| \mathbf{x}_k^i, \mathbf{w}_k\right) = \mathrm{Be}\left(y_k^i \middle| h\left(\mathbf{w}_k^\top \mathbf{x}_k^i\right)\right),$$

where $h(\cdot)$ is the sigmoid function, and $\mathrm{Be}(p)$ is a Bernoulli distribution. Therefore, following the same construction as in Section 3.3.1, parameters $\mathbf{W}$ and $\mathbf{\Omega}$ can be obtained by solving the following minimization problem:

$$\min_{\mathbf{W},\mathbf{\Omega}\succ 0} \sum_{k=1}^{m} \frac{1}{n_k} \sum_{i=1}^{n_k} \left(y_k^i \mathbf{w}_k^\top \mathbf{x}_k^i - \log(1 + e^{\mathbf{w}_k^\top \mathbf{x}_k^i})\right) + \lambda_0 \operatorname{tr}(\mathbf{W}\mathbf{\Omega}\mathbf{W}^\top) - d\log|\mathbf{\Omega}| + \lambda_1\|\mathbf{W}\|_1 + \lambda_2\|\mathbf{\Omega}\|_1. \tag{12}$$

The loss function is the logistic loss, where we have considered a 2-class classification setting. In general, we can consider any generalized linear model (GLM) (Nelder and Baker, 1972), with different link functions $h(\cdot)$, and therefore different probability densities, such as Poisson, Multinomial, and Gamma, for the conditional distribution. For any such model, our framework requires the optimization of an objective function of the form

$$\min_{\mathbf{W},\mathbf{\Omega}\succ 0} \sum_{k=1}^{m} \mathcal{L}(\mathbf{y}_k, \mathbf{X}_k\mathbf{w}_k) + \lambda_0 \operatorname{tr}(\mathbf{W}\mathbf{\Omega}\mathbf{W}^\top) - d\log|\mathbf{\Omega}| + \lambda_1\|\mathbf{W}\|_1 + \lambda_2\|\mathbf{\Omega}\|_1, \tag{13}$$

where $\mathcal{L}(\cdot)$ is a convex loss function obtained from a GLM.

Note that the objective function in (12) is similar to the one obtained for multi-task learning with linear regression in (3) in Section 3.3.1. Therefore, we use the same alternating minimization algorithm described in Section 3.3.1 to solve the problem in (12).

3.4 p-MSSL Interpretation as Using a Product of Distributions as Prior

From a probabilistic perspective, sparsity can be enforced using the so-called sparsity promoting priors, such as the Laplacian-like (double exponential) prior (Park and Casella, 2008). Accordingly, instead of exclusively assuming a multivariate Gaussian distribution as a prior for the rows of tasks parameter matrix $\mathbf{W}$, we can consider an improper prior which consists of the product of multivariate Gaussian and Laplacian distributions, of the form

$$p_{GL}\Big(\hat{\mathbf{w}}_j|\boldsymbol{\mu},\boldsymbol{\Omega},\lambda_0,\lambda_1\Big) \propto |\boldsymbol{\Omega}|^{1/2}\exp\left\{-\frac{\lambda_0}{2}(\hat{\mathbf{w}}_j-\boldsymbol{\mu})^\top\boldsymbol{\Omega}(\hat{\mathbf{w}}_j-\boldsymbol{\mu})\right\}\exp\left\{-\frac{\lambda_1}{2}\|\hat{\mathbf{w}}_j\|_1\right\}, \quad (14)$$

where we introduced the λ_0 parameter to control the strength of the Gaussian prior. By changing λ_0 and λ_1, we alter the relative effect of the two component priors in the product. Setting λ_0 to one and λ_1 to zero, we return to the exclusive Gaussian prior as in (2). Hence, p-MSSL formulation in (3) can be seen exactly (assuming sparse precision matrix in Gaussian prior) as a MAP inference of the conditional posterior distribution (with $\boldsymbol{\mu}=0$)

$$p\Big(\mathbf{W}|(\mathbf{X},\mathbf{Y}),\boldsymbol{\Omega}\Big) \propto \prod_{k=1}^{K}\prod_{i=1}^{N_k}\mathcal{N}\Big(y_k^i|\mathbf{w}_k^\top\mathbf{x}_k^i,\sigma^2\Big)\prod_{j=1}^{D}p_{GL}\Big(\hat{\mathbf{w}}_j|\boldsymbol{\Omega},\lambda_0,\lambda_1\Big). \quad (15)$$

Equivalently, the p-MSSL with GLM formulation as in (13) can be obtained by replacing the conditional Gaussian in (15) by another distribution in the exponential family.

3.5 Adding New Tasks

Suppose now that, after estimating all the tasks parameters and the precision matrix, a new task arrives and needs to be trained. This is known as the *asymmetric* MTL problem (Xue et al., 2007). Clearly, it will be computationally prohibitive in real applications to re-run the MSSL every time a new task arrives. Fortunately, MSSL can easily incorporate the new learning task into the framework using the information from the previous trained tasks.

After the arrival of the new task $\tilde{m}$, where $\tilde{m}=m+1$, the extended sample covariance matrix $\tilde{\mathbf{S}}$, computed from the parameter matrix $\mathbf{W}$, and the precision matrix $\tilde{\boldsymbol{\Omega}}$ are partitioned in the following form

$$\tilde{\boldsymbol{\Omega}}=\begin{pmatrix}\boldsymbol{\Omega}_{11} & \boldsymbol{\omega}_{12}\\ \boldsymbol{\omega}_{12}^\top & \omega_{22}\end{pmatrix} \qquad \tilde{\mathbf{S}}=\begin{pmatrix}\mathbf{S}_{11} & \mathbf{s}_{12}\\ \mathbf{s}_{12}^\top & s_{22}\end{pmatrix}$$

where $\mathbf{S}_{11}$ and $\boldsymbol{\Omega}_{11}$ are the sample covariance and precision matrix, respectively, corresponding to the previous tasks, which have already been trained and will be kept fixed during the estimation of the parameters associated with the new task.

Let $\mathbf{w}_{\tilde{m}}$ be the set of parameters associated with the new task $\tilde{m}$ and $\tilde{\mathbf{W}}=[\mathbf{W}_m\ \mathbf{w}_{\tilde{m}}]_{d\times\tilde{m}}$, where $\mathbf{W}_m$ is the matrix with the task parameters of all previous m tasks. For the learning of $\mathbf{w}_{\tilde{m}}$, we modify problem (4a) to include only those terms on which $\mathbf{w}_{\tilde{m}}$ depends

$$f_{\tilde{\boldsymbol{\Omega}}}(\mathbf{w}_{\tilde{m}};\mathbf{X}_{\tilde{m}},\mathbf{y}_{\tilde{m}},\lambda_0,\lambda_1)=\frac{1}{n_{\tilde{m}}}\sum_{i=1}^{n_{\tilde{m}}}(\mathbf{w}_{\tilde{m}}^\top\mathbf{x}_{\tilde{m}}^i-y_{\tilde{m}}^i)^2+\lambda_0\mathrm{tr}(\tilde{\mathbf{W}}\tilde{\boldsymbol{\Omega}}\tilde{\mathbf{W}}^\top)+\lambda_1\|\mathbf{w}_{\tilde{m}}\|_1, \quad (16)$$

and the same optimization methods for (4a) can be applied.

Recall that the task dependence learning problem (8) is equivalent to solving a graphical Lasso problem. Based on Banerjee et al. (2008), Friedman et al. (2008) proposed a block coordinate descent method which updates one column (and the corresponding row) of the matrix $\tilde{\mathbf{\Omega}}$ per iteration. They show that if $\tilde{\mathbf{\Omega}}$ is initialized with a positive semidefinite matrix, then the final (estimated) $\tilde{\mathbf{\Omega}}$ matrix will be positive semidefinite, even if $d > m$. Setting initial values of $\boldsymbol{\omega}_{12}$ as zero and ω_{22} as one (the new task is supposed to be conditionally independent on all other previous tasks), the extended precision matrix $\tilde{\mathbf{\Omega}}$ is assured to be positive semidefinite. From Friedman et al. (2008), $\boldsymbol{\omega}_{12}$ and ω_{22} are obtained as:

$$\boldsymbol{\omega}_{12} = -\hat{\boldsymbol{\beta}}\theta_{22} \tag{17a}$$

$$\omega_{22} = 1/(\theta_{22} - \boldsymbol{\theta}_{12}^{\top}\hat{\boldsymbol{\beta}}), \tag{17b}$$

where $\hat{\boldsymbol{\beta}}$ is computed from

$$\hat{\boldsymbol{\beta}} := \arg\min_{\boldsymbol{\alpha}} \ \Big\{\frac{1}{2}\|\tilde{\mathbf{\Omega}}_m^{1/2}\boldsymbol{\alpha} - \tilde{\mathbf{\Omega}}_m^{-1/2}\mathbf{s}_{12}\|_2^2 + \delta\|\boldsymbol{\alpha}\|_1\Big\}, \tag{18}$$

where $\eta > 0$ and $\delta > 0$ are sparsity regularization parameters; and $\boldsymbol{\theta}_{12}^{\top} = \tilde{\mathbf{\Omega}}_{11}^{-1}\hat{\boldsymbol{\beta}}$ and $\theta_{22} = s_{22} + \delta$. See Friedman et al. (2008) for further details. The problem (18) is a simple Lasso formulation for which efficient algorithms have been proposed (Beck and Teboulle, 2009; Boyd et al., 2011). Then to learn the coefficients for the new task $\tilde{m}$ and its relationship with the previous tasks, we iterate over solving (16) and (17) until convergence.

3.6 MSSL with Gaussian Copula Models

In the Gaussian graphical model associated with the problem (4b) the rows of the weight matrix $\mathbf{W}$ are assumed to be normally distributed. As such assumption may not hold in some cases, we need a more flexible model. A promising candidate is the copula model.

Copulas are class of flexible multivariate distributions that are expressed by its univariate marginals and a copula function that describes the dependence structure between the variables. Consequently, copulas decompose a multivariate distribution into its marginal distributions and the copula function connecting them. Copulas are founded on Sklar (1959) theorem which states that: *any m-variate distribution $f(V_1, ..., V_m)$ with continuous marginal functions $f_1, ..., f_m$ can be expressed as its copula function $C(\cdot)$ evaluated at its marginals, that is, $f(V_1, ..., V_m) = C(f_1(V_1), ..., f_m(V_m))$* and, conversely, *any copula function $C(\cdot)$ with marginal distributions $f_1, ..., f_m$ defines a multivariate distribution.* Several copulas have been described, which typically exhibit different dependence properties. Here, we focus on the Gaussian copula that adopts a balanced combination of flexibility and interpretability that has attracted a lot of attention (Xue and Zou, 2012).

3.6.1 Gaussian Copula Distributions

The Gaussian copula $C_{\mathbf{\Sigma}^0}$ is the copula of an m-variate Gaussian distribution $\mathcal{N}_m(0,\ \mathbf{\Sigma}^0)$ with $m \times m$ positive definite correlation matrix $\mathbf{\Sigma}^0$

$$C(V_1, ..., V_m; \mathbf{\Sigma}^0) = \Phi_{\mathbf{\Sigma}^0}\Big(\Phi^{-1}(V_1), ..., \Phi^{-1}(V_m)\Big), \tag{19}$$

where Φ^{-1} is the inverse of a standard normal distribution function and $\Phi_{\mathbf{\Sigma}^0}$ is the joint distribution function of a multivariate normal distribution with mean vector zero and covariance matrix equal to the correlation matrix $\mathbf{\Sigma}^0$. Note that without loss of generality, the covariance matrix $\mathbf{\Sigma}^0$ can be viewed as a correlation matrix, as observations can be replaced by their normal-scores. Therefore, Sklar's theorem allows to construct a multivariate distribution with non-Normal marginal distributions and the Gaussian copula.

A more general formulation of the Gaussian copula is the semiparametric Gaussian copulas (Liu et al., 2009; Xue and Zou, 2012), which allows the marginals to follow any non-parametric distribution.

Definition 1 (Semiparametric Gaussian copula models) *Let $f = \{f_1, ..., f_m\}$ be a set of continuous monotone and differentiable univariate functions. An m-dimensional random variable $V = (V_1, ..., V_m)$ has a semiparametric Gaussian Copula distribution if the joint distribution of the transformed variable $f(V)$ follows a multivariate Gaussian distribution with correlation matrix $\mathbf{\Sigma}^0$, that is, $f(V) = (f_1(V_1), ..., f_m(V_m))^\top \sim \mathcal{N}_m(0, \mathbf{\Sigma}^0)$.*

From the definition we notice that the copula does not have requirements on the marginal distributions as long the monotone continuous functions $f_1, ..., f_m$ exist.

The semiparametric Gaussian copula model is completely characterized by two unknown parameters: the correlation matrix $\mathbf{\Sigma}^0$ (or its inverse, the precision matrix $\mathbf{\Omega}^0 = (\mathbf{\Sigma}^0)^{-1}$) and the marginal transformation functions $f_1, ..., f_m$. The unknown marginal distributions can be estimated by existing nonparametric methods. However, as will be seen next, when estimating the dependence parameter is the ultimate aim, one can directly estimate $\mathbf{\Omega}^0$ without explicitly computing the functions.

Let $Z = (Z_1, ..., Z_m) = (f(V_1), ..., f(V_m))$ be a set of latent variables. By the assumption of joint normality of Z, we know that $\mathbf{\Omega}^0_{ij} = 0 \iff Z_i \perp\!\!\!\perp Z_j | Z_{\setminus\{i,j\}}$. Interestingly, Liu et al. (2009) showed that $Z_i \perp\!\!\!\perp Z_j | Z_{\setminus\{i,j\}} \iff V_i \perp\!\!\!\perp V_j | V_{\setminus\{i,j\}}$, that is, variables V and Z share exactly the same conditional dependence graph. As we focus on sparse precision matrix, to estimate the parameter $\mathbf{\Omega}^0$ we can resort to the ℓ_1-penalized maximum likelihood method, the graphical Lasso problem (8).

Let $r_{1i}, ..., r_{ni}$ be the rank of the samples from variable V_i and the sample mean $\bar{r}_j = \frac{1}{n}\sum_{i=1}^n r_{ij} = \frac{n+1}{2}$. We start by reviewing the Spearman's ρ and Kendal's τ statistics:

$$\text{(Spearman's } \rho) \quad \hat{\rho}_{ij} = \frac{\sum_{t=1}^n (r_{ti} - \bar{r}_i)(r_{tj} - \bar{r}_j)}{\sqrt{\sum_{t=1}^n (r_{ti} - \bar{r}_i)^2 \cdot \sum_{t=1}^n (r_{tj} - \bar{r}_j)^2}}, \tag{20a}$$

$$\text{(Kendall's } \tau) \quad \hat{\tau}_{ij} = \frac{2}{n(n-1)} \sum_{1 \le t \le t' \le n} \text{sign}\Big((v_{ti} - v_{t'i})(v_{tj} - v_{t'j})\Big). \tag{20b}$$

We observe that Spearman's rho is computed from the ranks of the samples and Kendall's correlation is based on the concept of concordance of pairs, which in turn is also computed from the ranks r_i. Therefore, both measures are invariant to monotone transformation of the original samples and rank-based correlations such as Spearman's ρ and Kendal's τ of the observed variables V and the latent variables Z are identical. In other words, if we are only interested in estimating the precision matrix $\mathbf{\Omega}^0$, we can treat the observed variable V as the unknown variable Z, thus avoiding estimating the transformation functions $f_1, ..., f_m$.

To connect Spearman's ρ and Kendal's τ rank-based correlation to the underlying Pearson correlation in the graphical Lasso formulation (8) of the inverse covariance selection problem, for Gaussian random variables a result due to Kendall (1948) is used:

$$\hat{\mathbf{S}}^{\rho}_{ij} = \begin{cases} 2\sin\left(\frac{\pi}{6}\hat{\rho}_{ij}\right), & i \neq j \\ 1 & , \ i = j \end{cases} \quad , \quad \hat{\mathbf{S}}^{\tau}_{ij} = \begin{cases} \sin\left(\frac{\pi}{2}\hat{\tau}_{ij}\right), & i \neq j \\ 1 & , \ i = j. \end{cases}$$

We then replace $\mathbf{S}$ in (8) by $\hat{\mathbf{S}}^{\rho}$ or $\hat{\mathbf{S}}^{\tau}$ and the same ADMM proposed in Section 3.3.1 is applied. The MSSL algorithms with Gaussian copula models are called p-MSSL$_{\text{cop}}$ and r-MSSL$_{\text{cop}}$, for the parameter and residual-based versions, respectively.

Liu et al. (2012) suggested that the SGC models can be used as a safe replacement of the popular Gaussian graphical models, even when the data are truly Gaussian. Compared with the Gaussian graphical model (8), the only additional cost of the SGC model is the computation of the $m(m-1)/2$ pairs of Spearman's ρ or Kendal's τ statistics, for which efficient algorithms have complexity $O(m \log m)$.

Other copula distributions also exist, such as the Archimedean class of copulas (McNeil and Nešlehová, 2009), which are useful to model tail dependence. Nevertheless, Gaussian copula is a compelling distribution for expressing the intricate dependency graph structure.

3.7 Residual Precision Structure

In the residual structure based MSSL, called r-MSSL, the relationship among tasks will be modeled in terms of partial correlations among the errors $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_m)^{\top}$, instead of considering explicit dependencies between the coefficients $\mathbf{w}_1, \ldots, \mathbf{w}_m$ for the different tasks. To illustrate this idea, let us consider the regression scenario where $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_m)$ is a vector of desired outputs for each task, and $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_m)^{\top}$ are the covariates for the m tasks. The assumed linear model can be denoted by

$$\mathbf{Y} = \mathbf{X}\mathbf{W} + \boldsymbol{\xi}, \tag{21}$$

where $\boldsymbol{\xi} = \mathbf{Y} - \mathbf{X}\mathbf{W} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^0)$. In this model, the errors are not assumed to be i.i.d., but vary jointly over the tasks following a Gaussian distribution with precision matrix $\boldsymbol{\Omega} = (\boldsymbol{\Sigma}^0)^{-1}$. Finding the dependence structure among the tasks now amounts to estimating the precision matrix $\boldsymbol{\Omega}$. Such models are commonly used in spatial statistics (Mardia and Marshall, 1984) in order to capture spatial autocorrelation between geographical locations. We adopt the framework in order to capture "loose coupling" between the tasks by means of a dependence in the error distribution. For example, in domains such as climate or remote sensing, there often exist noise autocorrelations over the spatial domain under consideration. Incorporating this dependence by means of the residual precision matrix is therefore more interpretable than the explicit dependence among the coefficients in $\mathbf{W}$.

Following the above definition, the multi-task learning framework can be modified to incorporate the relationship between the errors $\boldsymbol{\xi}$. We assume that the coefficient matrix $\mathbf{W}$ is fixed, but unknown. Since $\boldsymbol{\xi}$ follows a Gaussian distribution, maximizing the likelihood of the data, penalized with a sparse regularizer over $\boldsymbol{\Omega}$, reduces to the optimization problem

$$\min_{\mathbf{W}, \boldsymbol{\Omega} \succ 0} \left(\sum_{k=1}^{m} \frac{1}{n_k} \|\mathbf{y}_k - \mathbf{X}_k \mathbf{w}_k\|_2^2 \right) - d \log |\boldsymbol{\Omega}| + \lambda_0 \mathrm{tr}\Big((\mathbf{Y} - \mathbf{X}\mathbf{W}) \boldsymbol{\Omega} (\mathbf{Y} - \mathbf{X}\mathbf{W})^{\top} \Big) + \lambda_1 \|\mathbf{W}\|_1 + \lambda_2 \|\boldsymbol{\Omega}\|_1. \tag{22}$$

We use the alternating minimization scheme illustrated in previous sections to solve the problem in (22). Since the cost function is biconvex and convex in each of its arguments $\mathbf{W}$

and $\mathbf{\Omega}$, thus a *partial optimum* will be found (Gorski et al., 2007). Fixing $\mathbf{W}$, the problem of estimating $\mathbf{\Omega}$ is exactly the same as (8), but with the interpretation of capturing the conditional dependence among the residuals instead of the coefficients. The problem of estimating the tasks coefficients $\mathbf{W}$ will be slightly modified due to the change in the trace term, but the algorithms presented in Section 3.3.1 can still be used. Further, the model can be extended to losses other than the squared loss, used here due to the fact that $\boldsymbol{\xi}$ follows a Gaussian distribution.

Two instances of MSSL have been provided, p-MSSL and r-MSSL, along with their Gaussian copula versions, p-MSSL$_{\text{cop}}$ and r-MSSL$_{\text{cop}}$. In summary, p-MSSL and p-MSSL$_{\text{cop}}$ can be applied to both regression and classification problems. On the other hand, r-MSSL and r-MSSL$_{\text{cop}}$, can only be applied to regression problems, as the residual error of a classification problem is clearly non-Gaussian.

3.8 Complexity Analysis

The complexity of an iteration of the MSSL algorithms can be measured in terms of the complexity of its $\mathbf{W}$-step and $\mathbf{\Omega}$-step. Each iteration of the FISTA algorithm in the $\mathbf{W}$-step involves the element-wise operations, for both the $\mathbf{z}$-update and the proximal operator, which takes $\mathcal{O}(md)$ operations each. Gradient computation of the squared loss with trace penalization involves matrices multiplication which costs $\mathcal{O}(\max(mn^2d, dm^2))$ operations for dense matrix $\mathbf{W}$ and $\mathbf{\Omega}$, but can be reduced as both matrices are sparse. We are assuming that all tasks have the same number of samples n.

In an ADMM iteration, the dominating operation is clearly the SVD decomposition when solving the subproblem (10a). It costs $\mathcal{O}(m^3)$ operations. The other two steps amount to element-wise operations which costs $\mathcal{O}(m^2)$ operations. As mentioned previously, the copula-based MSSL algorithms have the additional cost of $\mathcal{O}(m\log(m))$ for computing Kendal's τ or Spearman's ρ statistics.

The memory requirements include $\mathcal{O}(md)$ for the $\mathbf{z}$ and previous weight matrix $\mathbf{W}^{(t-1)}$ in the $\mathbf{W}$-step and $\mathcal{O}(m^2)$ for the dual variable $\mathbf{U}$ and the auxiliary matrix $\mathbf{Z}$ in the ADMM for the $\mathbf{\Omega}$-step. We should mention that the complexity is evidently associated with the optimization algorithms used for solving problems 4a and 4b.

4. Experimental Results

In this section we provide experimental results to show the effectiveness of the proposed framework for both regression and classification problems.

4.1 Regression

We start with experiments on synthetic data and then move to the problem of predicting land air temperature in South and North America by the use of multi-model ensemble.

To select the penalty parameters λ_1 and λ_2 we use a stability selection procedure described in Meinshausen and Bühlmann (2010). It is a sub-sampling approach that provides a way to find stable structures and hence a principle to choose a proper amount of regularization for structure estimation. The parameter λ_0 was set to one in all experiments.

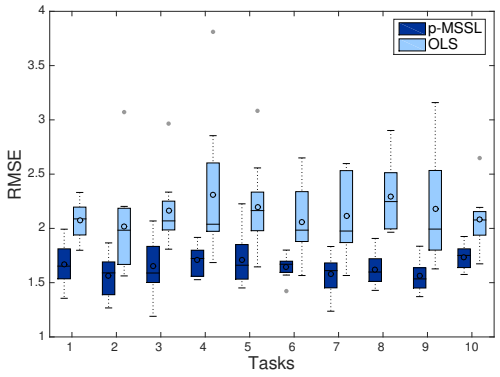


Figure 1: RMSE per task comparison between p-MSSL and Ordinary Least Square over 30 independent runs. p-MSSL gives better performance on related tasks (1-4 and 5-10).

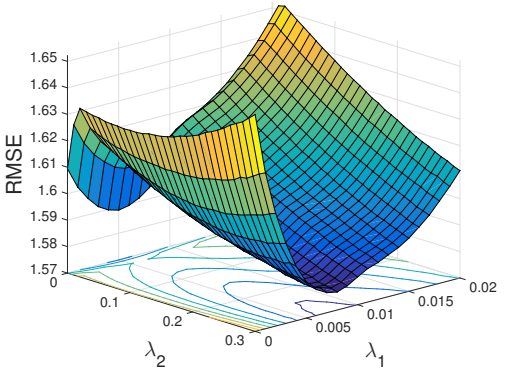


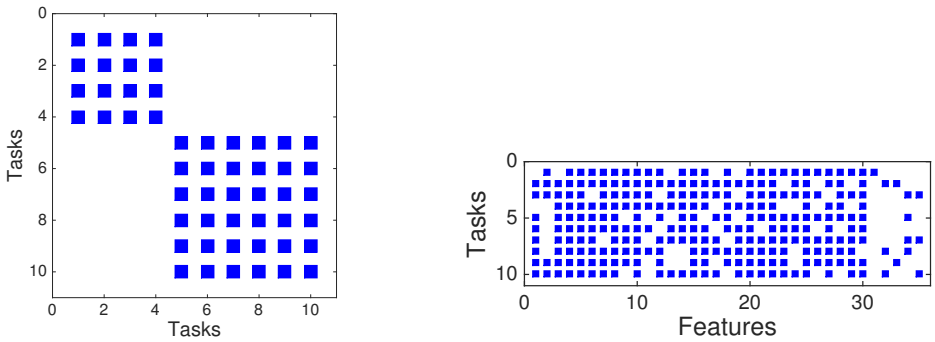
Figure 2: Average RMSE error on the test set of synthetic data for all tasks varying parameters λ_2 (controls sparsity on $\mathbf{\Omega}$) and λ_1 (controls sparsity on $\mathbf{W}$).

4.1.1 Synthetic Data set

We created a synthetic data set with 10 linear regression tasks of dimension $D = D_r + D_u$, where D_r and D_u are the number of relevant and non-relevant (unnecessary) variables, respectively. This is to evaluate the ability of the algorithm to discard non-relevant features. We used $D_r = 30$ and $D_u = 5$. For each task, the relevant input variables $\mathbf{X}'_k$ are generated i.i.d. from a multivariate normal distribution, $\mathbf{X}'_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{D_r})$. The corresponding output variable is generated as $\mathbf{y}_k = \mathbf{X}'_k \mathbf{w}_k + \boldsymbol{\xi}$ where $\xi_i \sim \mathcal{N}(0, 1), \forall i = 1, ..., n_k$. Unnecessary variables are generated as $\mathbf{X}''_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{D_u})$. Hence, the total synthetic input data of the k-th task is formed as the concatenation of both set of variables, $\mathbf{X}_k = [\mathbf{X}'_k \ \mathbf{X}''_k]$. Note that only the relevant variables are used to produce the output variable $\mathbf{y}_k$. The parameter vectors for all tasks are chosen so that tasks 1 to 4 and 5 to 10 form two groups. Parameters for tasks 1-4 were generated as: $\mathbf{w}_k = \mathbf{w}_a \odot \mathbf{b}_k + \boldsymbol{\xi}$, where $\odot$ is the element-wise Hadamard product; and for tasks 5-10: $\mathbf{w}_k = \mathbf{w}_b \odot \mathbf{b}_k + \boldsymbol{\xi}$, where $\boldsymbol{\xi} = \mathcal{N}(\mathbf{0}, 0.2\mathbf{I}_{D_r})$. Vectors $\mathbf{w}_a$ and $\mathbf{w}_b$ are generated from $\mathcal{N}(\mathbf{0}, \mathbf{I}_{D_r})$, while $\mathbf{b}_k \sim \mathcal{U}(0, 1)$ are uniformly distributed D_r-dimensional random vectors. In summary, we have two clusters of mutually related tasks. We train the p-MSSL model with 50 data instances and test it on 100 data instances.

Figure 1 shows the RMSE error for p-MSSL and for the case where Ordinary Least Squares (OLS) was applied individually for each task. As expected, sharing information among related tasks improves prediction accuracy. p-MSSL does well on related tasks 1 to 4 and 5 to 10. Figures 3a and 3b depict the sparsity pattern of the task parameters $\mathbf{W}$ and the precision matrix $\mathbf{\Omega}$ estimated by the p-MSSL algorithm. As can be seen, our model is able to recover the true dependence structure among tasks. The two clusters of tasks were clearly revealed, indicated by the filled squares, meaning non-zero entries in the precision matrix, and then, relationship among tasks. Additionally, p-MSSL was able to discard most of the irrelevant features (last five) intentionally added into the synthetic data set.

Sensitivity analysis of p-MSSL sparsity parameters λ_1 (controls sparsity on $\mathbf{W}$) and λ_2 (controls sparsity on $\mathbf{\Omega}$) on the synthetic data is presented in Figure 2. We observe that the smallest RMSE was found with a value of $\lambda_1 > 0$, which implies that a reduced set of variables is more representative than the full set, as it is indeed the case for the synthetic



(a) (b)

Figure 3: Sparsity pattern of the p-MSSL estimated parameters on the synthetic data set: (a) precision matrix $\mathbf{\Omega}$; (b) weight matrix $\mathbf{W}$. The algorithm identified the true task relationship in (a) and removed most of the non-relevant features (last five columns) in (b).

data set. The best solution is with a sparse precision matrix, as we can see in Figure 2 (smallest RMSE with $\lambda_2 > 0$). We should mention that as we increase λ_1 we encourage sparsity on $\mathbf{W}$ and, as a consequence, it becomes harder for p-MSSL to capture the true relationship among the column vectors (tasks parameters), since it learns $\mathbf{\Omega}$ from $\mathbf{W}$. This drawback is overcome in the r-MSSL algorithm, in which the precision matrix is estimated from the residuals instead of being estimated from the task parameters directly.

4.1.2 Combining Earth System Models

An Earth System Model (ESM) is a complex mathematical representation of the major climate system components (atmosphere, land surface, ocean, and sea ice), and their interactions. They are run as computer simulations, to predict climate variables such as temperature, pressure, and precipitation over multiple centuries. Several ESMs have been proposed by climate science institutes from different countries around the world.

The forecasts of future climate variables as predicted by these models have high variability, which in turn introduces uncertainty in analysis based on these predictions. One of the reasons for such uncertainty in the response of the ESMs comes from the model variability due to the fact that ESMs implements certain climatic process in different ways. Then, suitably combining outputs from multiple ESMs can greatly reduce the variability. Another equally important source of uncertainty is due to initial conditions. As ESMs are non-linear dynamic systems, changes in initial conditions can lead to different realizations of climate. In this work we focus only on the model variability. Modeling uncertainty from initial conditions is an ongoing work.

We consider the problem of combining ESM outputs for land surface temperature prediction in both South and North America, which are the world's fourth and third-largest continents, respectively, and jointly cover approximately one third of the Earth's land area. The climate is very diversified in those areas. In South America, the Amazon River basin in the north has the typical hot wet climate suitable for the growth of rain forests. The Andes Mountains, on the other hand, remain cold throughout the year. The desert regions of Chile are the driest part of South America. As for North America, the subartic climate in North

ESM	Origin	Refs.
BCC_CSM1.1	Beijing Climate Center, China	Zhang et al. (2012)
CCSM4	National Center for Atmospheric Research, USA	Washington et al. (2008)
CESM1	National Science Foundation, NCAR, USA	Subin et al. (2012)
CSIRO	Commonwealth Scient. and Ind. Res. Org., Australia	Gordon et al. (2002)
HadGEM2	Met Office Hadley Centre, UK	Collins et al. (2011)
IPSL	Institut Pierre-Simon Laplace, France	Dufresne et al. (2012)
MIROC5	Atmosphere and Ocean Research Institute, Japan	Watanabe et al. (2010)
MPI-ESM	Max Planck Inst. for Meteorology, Germany	Brovkin et al. (2013)
MRI-CGCM3	Meteorological Research Institute, Japan	Yukimoto et al. (2012)
NorESM	Norwegian Climate Centre, Norway	Bentsen et al. (2012)

Table 1: Description of the Earth System Models used in the experiments.

Canada contrasts with the semi-arid climate in western United States and Mexico's central area. The Rocky Mountains have a large impact in land's climate, and temperature significantly varies due to topographic effects (elevation and slope) (Kinel et al., 2002). Southeast of the United States is characterized by its subtropical humid climate with relatively high temperatures and an evenly distributed precipitation throughout the year.

For the experiments we use 10 ESMs from the CMIP5 data set (Taylor et al., 2012). Details about the ESMs data sets are listed in Table 1. The global observation data for surface temperature is obtained from the Climate Research Unit (CRU) at the University of East Anglia. Both, ESM outputs and observed data are the raw temperatures (not anomalies) measured in degree Celsius. We align the data from the ESMs and CRU observations to have the same spatial and temporal resolution, using publicly available climate data operators (CDO). For all the experiments, we used a $2.5^o \times 2.5^o$ grid over latitudes and longitudes in South and North America, and monthly mean temperature data for 100 years, 1901-2000, with records starting from January 16, 1901. In other words, we have two data sets: (1) South America with 250 spatial locations; and (2) North America with 490 spatial locations over land. Data sets and code are available at: `bitbucket.org/andreric/mssl-code`. For the MTL framework, each geographical location represents a task (regression problem).

From an MTL perspective, the two data sets have different levels of difficulty. North America data set has almost twice the number of tasks as compared to South America, so that we discuss the performance of MSSL in problems with high number of tasks. It brings new challenges to MTL methods. On the other hand, South America has a more diverse climate, which makes task dependence structure more complex. Preliminary results on South America were published in Gonçalves et al. (2015) employing a high-level description format.

Baselines and Evaluation: We consider the following eight baselines for comparison and evaluation of MSSL performance for the ESM combination problem. The first two baselines (MMA and Best-ESM) are commonly used in climate sciences due to their stability and simple interpretation. We will refer to these baselines and MSSL as the "models" in the sequel and the constituent ESMs as "submodels". Four well known MTL methods were also added in the comparison. The eight baselines are:

1. **Multi-model Average (MMA)**: is the current technique used by Intergovernmental Panel on Climate Change (IPCC). It gives equal weight to all ESMs at every location.

2. **Best-ESM**: uses the predicted outputs of the best ESM in the training phase (lowest RMSE). This baseline is not a combination of submodels, but a single ESM instead.

3. **Ordinary Least Squares (OLS)**: performs an ordinary least squares regression for each geographic location, independently of the others.

4. **Spatial Smoothing Multi Model Regression ($\mathbf{S^2M^2R}$)**: proposed by Subbian and Banerjee (2013) to deal with ESM outputs combination, can be seen as a special case of MSSL with the pre-defined dependence matrix $\mathbf{\Omega}$ equal to the Laplacian matrix.

5. **MTL-FEAT** (Argyriou et al., 2007): all the tasks are assumed to be related and share a low-dimensional feature subspace. The following two methods, 6 and 7, can be seen as relaxations of this assumption. We used the code provided in MALSAR package (Zhou et al., 2011a).

6. **Group-MTL** (Kang et al., 2011): groups of related tasks are assumed and tasks belonging to the same group share a common feature representation. The code was taken from the author's homepage: `http://www-scf.usc.edu/~zkang/GoupMTLCode.zip`.

7. **GO-MTL** (Kumar and Daume III, 2012): founded on a relaxation of the group idea in Kang et al. (2011) by allowing subspaces shared by each group to overlap between them. We obtained the code directly from the authors.

8. **MTRL** (Zhang and Yeung, 2010): the covariance matrix among tasks coefficients is captured by imposing a matrix-variate normal prior over the coefficient matrix $\mathbf{W}$. The non-convex MAP problem is relaxed and an alternating minimization procedure is proposed to solve the convex problem. The code was taken from author's homepage: `http://www.comp.hkbu.edu.hk/~yuzhang/codes/MTRL.zip`.

Methodology

We assume here that sub-models skills are stationary, that is, the coefficient associated with each sub-model does not change over time. To have an overall measure of the capability of the method, we considered scenarios with different amount of data available for training. For each scenario, the same number of training data (columns of Table 2) are used for all tasks, and the remaining data is used for test. Starting from one year of temperature measures (12 samples), we increase till ten years of data for training. The remained data was used as test set. For each scenario 30 independent runs are performed. Therefore, the results are reported as the average and standard deviation of RMSE for all scenarios.

Results

Table 2 report the average and standard deviation RMSE for all locations in South and North America. In South America, except for the smallest training sample (12 months) the average model (MMA) has the highest RMSE for all training sample size. Best-ESM presented a better temperature future projection compared to MMA. Generally speaking, the MTL methods performed significantly better than non-MTL ones, particularly when a small number of samples are available for training. As the spatial smoothness assumption is true for temperature, S^2M^2R obtained results comparable with those yielded by MTL methods. However, this assumption does not hold for other climate variables, such as

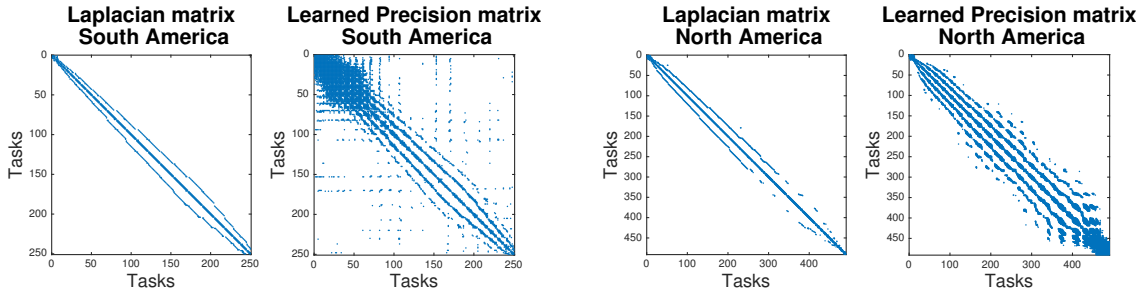


Figure 4: Laplacian matrix (on grid graph) assumed by S^2M^2R and the precision matrix learned by r-MSSL$_{\text{cop}}$ on both South and North America. r-MSSL$_{\text{cop}}$ can capture spatial relations beyond immediate neighbors. While South America is densely connected in the Amazon forest area (corresponding to top left corner) along with many spurious connections, North America is more spatially smooth.

precipitation and S^2M^2R may not succeed in those problems. On the other hand, MTL methods are general enough and in principle can be used for any climate variable. In the realm of MTL methods, all the four MSSL instantiations outperform the four other MTL contenders. It is worth observing that the two MSSL methods based on Gaussian Copula models provided smaller RMSE than the two with Gaussian models, particularly for problems with small training sample size. As Gaussian Copula models are more flexible, it is able to capture a wider range of task dependences than ordinary Gaussian models.

Similar behavior is observed in North America data set, except for the fact that MMA does much better than Best-ESM for all training sample settings. Again, all the four MSSL instantiations provided better future temperature projection. We also note that the residual-based structure dependence modeling with Gaussian Copula, r-MSSL$_{\text{cop}}$, produced slightly better results than the other three MSSL instantiations. As will be left clear in Figures 4 and 6, residual-based MSSL coherently captures related geographical locations, indicating that it can be used as an alternative to parameter-based task dependence modeling.

Figure 4 shows the precision matrix estimated by the r-MSSL$_{\text{cop}}$ algorithm and the Laplacian matrix assumed by S^2M^2R in both South and North America. Blue dots means negative entries in the matrix, while red, positive. Interpreting the entries of the matrix in terms of partial correlation, $\mathbf{\Omega}_{ij} < 0$ means positive partial correlation between $\mathbf{w}_i$ and $\mathbf{w}_j$, while $\mathbf{\Omega}_{ij} > 0$ means negative partial correlation. Not only is the precision matrix for r-MSSL$_{\text{cop}}$ able to capture the relationship among a geographical locations' immediate neighbors (as in a grid graph) but it also recovers relationships between locations that are not immediate neighbors. The plots also provide an information of the range of neighborhood influence, which can be useful in spatial statistics analysis.

The RMSE per geographical location for r-MSSL$_{\text{cop}}$ and three common approaches used in climate sciences, MMA, Best-ESM, and OLS, are shown in Figures 5. As previously mentioned, South and North America have a diverse climate and not all of the ESMs are designed to take into account and capture this scenario. Hence, averaging the model outputs, as done by MMA, reduces prediction accuracy. On the other hand r-MSSL$_{\text{cop}}$

Algorithms	Months									
	12	24	36	48	60	72	84	96	108	120
South America										
Best-ESM	1.61 (0.02)	1.56 (0.01)	1.54 (0.01)	1.53 (0.01)	1.53 (0.01)	1.53 (0.01)	1.52 (0.01)	1.52 (0.01)	1.52 (0.01)	1.52 (0.00)
MMA	1.68 (0.00)	1.68 (0.00)	1.68 (0.00)	1.68 (0.00)	1.68 (0.00)	1.68 (0.00)	1.68 (0.00)	1.68 (0.00)	1.68 (0.00)	1.68 (0.00)
OLS	3.53 (0.45)	1.16 (0.04)	1.03 (0.02)	0.97 (0.01)	0.94 (0.01)	0.92 (0.01)	0.91 (0.01)	0.90 (0.01)	0.89 (0.01)	0.88 (0.00)
S^2M^2R	1.06 (0.03)	0.98 (0.03)	0.94 (0.01)	0.92 (0.01)	0.91 (0.01)	0.90 (0.01)	0.89 (0.01)	0.88 (0.01)	0.88 (0.01)	0.88 (0.00)
Group-MTL	1.09 (0.04)	1.01 (0.04)	0.96 (0.01)	0.93 (0.01)	0.92 (0.01)	0.91 (0.01)	0.90 (0.01)	0.89 (0.01)	0.89 (0.01)	0.88 (0.00)
GO-MTL	1.11 (0.04)	0.98 (0.03)	0.94 (0.01)	0.92 (0.01)	0.92 (0.01)	0.91 (0.01)	0.90 (0.01)	0.90 (0.01)	0.89 (0.01)	0.89 (0.00)
MTL-FEAT	1.05 (0.04)	0.99 (0.04)	0.94 (0.01)	0.92 (0.01)	0.91 (0.01)	0.90 (0.01)	0.89 (0.01)	0.88 (0.01)	0.88 (0.01)	0.88 (0.00)
MTRL	1.01 (0.04)	0.97 (0.03)	0.95 (0.02)	0.95 (0.02)	0.94 (0.01)	0.94 (0.01)	0.94 (0.01)	0.94 (0.01)	0.94 (0.01)	0.93 (0.01)
p-MSSL	1.02 (0.03)	0.94* (0.03)	0.90* (0.01)	0.89* (0.01)	0.88* (0.01)	0.88* (0.01)	**0.87*** (**0.01**)	**0.87*** (**0.01**)	**0.87*** (**0.01**)	**0.86*** (**0.00**)
p-MSSL$_{\text{cop}}$	**0.98*** (**0.03**)	0.93* (0.03)	**0.90*** (**0.01**)	0.89* (0.01)	**0.88*** (**0.01**)	**0.88*** (**0.01**)	**0.87*** (**0.01**)	**0.87*** (**0.01**)	**0.87*** (**0.01**)	0.87* (0.00)
r-MSSL	1.02 (0.03)	0.94* (0.03)	0.91* (0.01)	0.89* (0.01)	0.89* (0.01)	0.88* (0.01)	**0.87*** (**0.01**)	**0.87*** (**0.01**)	**0.87*** (**0.01**)	**0.86*** (**0.00**)
r-MSSL$_{\text{cop}}$	1.00 (0.03)	**0.93*** (**0.03**)	**0.90*** (**0.01**)	**0.89*** (**0.01**)	**0.88*** (**0.01**)	0.88* (0.01)	**0.87*** (**0.01**)	**0.87*** (**0.01**)	**0.87*** (**0.01**)	0.87 (0.00)
North America										
Best-ESM	3.85 (0.07)	3.75 (0.05)	3.70 (0.04)	3.68 (0.04)	3.64 (0.03)	3.64 (0.03)	3.61 (0.02)	3.60 (0.02)	3.60 (0.02)	3.58 (0.02)
MMA	2.94 (0.00)	2.94 (0.00)	2.94 (0.01)	2.94 (0.01)	2.94 (0.01)	2.94 (0.01)	2.94 (0.01)	2.94 (0.01)	2.94 (0.01)	2.94 (0.01)
OLS	10.06 (1.16)	3.37 (0.09)	2.96 (0.07)	2.79 (0.05)	2.69 (0.03)	2.64 (0.04)	2.59 (0.02)	2.56 (0.02)	2.54 (0.02)	2.53 (0.03)
S^2M^2R	3.14 (0.17)	2.79 (0.05)	2.70 (0.05)	2.64 (0.03)	2.59 (0.03)	2.56 (0.03)	2.54 (0.02)	2.52 (0.02)	2.51 (0.02)	2.50 (0.02)
Group-MTL	2.83 (0.13)	2.69 (0.04)	2.64 (0.04)	2.60 (0.03)	2.57 (0.02)	2.54 (0.03)	2.52 (0.02)	2.51 (0.01)	2.50 (0.02)	2.50 (0.02)
GO-MTL	3.02 (0.15)	2.73 (0.05)	2.63 (0.05)	2.58 (0.04)	2.53 (0.03)	2.51 (0.03)	2.49 (0.02)	2.49 (0.02)	2.48 (0.02)	2.48 (0.02)
MTL-FEAT	2.76 (0.12)	2.62 (0.04)	2.59 (0.04)	2.57 (0.03)	2.53 (0.02)	2.52 (0.02)	2.50 (0.02)	2.49 (0.01)	2.49 (0.01)	2.48 (0.02)
MTRL	2.93 (0.17)	2.83 (0.10)	2.78 (0.09)	2.81 (0.09)	2.75 (0.04)	2.77 (0.05)	2.75 (0.04)	2.76 (0.04)	2.75 (0.05)	2.77 (0.04)
p-MSSL	**2.71*** (**0.11**)	2.58* (0.05)	2.53* (0.04)	2.53* (0.04)	2.49* (0.02)	2.50* (0.02)	2.49 (0.02)	2.49 (0.01)	2.48 (0.02)	2.49 (0.01)
p-MSSL$_{\text{cop}}$	**2.71*** (**0.11**)	**2.57*** (**0.05**)	**2.52*** (**0.04**)	2.52* (0.04)	2.49* (0.02)	2.49* (0.02)	2.48* (0.02)	2.48* (0.01)	2.47 (0.02)	2.48 (0.01)
r-MSSL	2.71* (0.11)	2.58* (0.05)	2.53* (0.04)	2.53* (0.04)	2.49* (0.02)	2.49* (0.02)	2.49 (0.02)	2.48 (0.01)	2.48 (0.02)	2.49 (0.01)
r-MSSL$_{\text{cop}}$	2.71* (0.11)	**2.57*** (**0.05**)	**2.52*** (**0.04**)	**2.52*** (**0.04**)	**2.48*** (**0.02**)	**2.49*** (**0.02**)	**2.48*** (**0.02**)	**2.48*** (**0.01**)	**2.47*** (**0.02**)	**2.48** (**0.01**)

Table 2: Mean and standard deviation over 30 independent runs for several amounts of monthly data used for training. The symbol "*" indicates statistically significant (t-test, P < 0.05) improvement when compared to the best contender. MSSL with Gaussian copula provides better prediction accuracy.

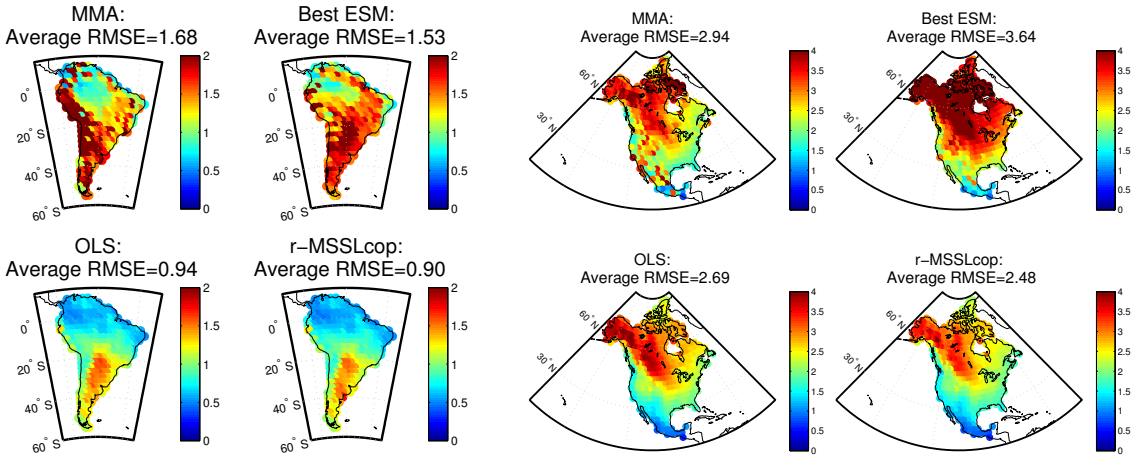


Figure 5: [Best viewed in color] RMSE per location for r-MSSL$_{\text{cop}}$ and three common methods in climate sciences, computed using 60 monthly temperature measures for training. It shows that r-MSSL$_{\text{cop}}$ substantially reduces RMSE, particularly in Northern South America and Northwestern North America.

performs better because it learns a more appropriate weight combination on the model outputs and incorporates spatial smoothing by learning the task relationship.

Figure 6 presents the dependence structure estimated by r-MSSL$_{\text{cop}}$ for South and North America data sets. Blue connections indicate dependent regions.

We immediately observe that locations in the northwest part of South America are densely connected. This area has a typical tropical climate and comprises the Amazon rainforest which is known for having hot and humid climate throughout the year with low temperature variation (Ramos, 2014). The cold climates which occur in the southernmost parts of Argentina and Chile are clearly highlighted. Such areas have low temperatures throughout the year, but there are large daily variations (Ramos, 2014). An important observation can be made about South America west cost, ranging from central Chile to Venezuela passing through Peru which has one of the driest deserts in the world. These areas are located to the left side of Andes Mountains and are known for arid climate. The average model is not performing well on this region compared to r-MSSL$_{\text{cop}}$. We can see the long lines connecting these coastal regions, which probably explains the improvement in terms of RMSE reduction achieved by r-MSSL$_{\text{cop}}$. The algorithm uses information from related locations to enhance its performance on these areas.

In the structure learned for North America, a densely connected area is also observed in the northeast part of North America, particularly the regions of Nunavut and North Quebec, which are characterized by their polar climate, with extremely severe winters and cold summers. Long connections between Alaska and regions from Northwestern Canada, which share similar climate patterns, can also be seen. Again, the r-MSSL$_{\text{cop}}$ algorithm had no access to the latitude and longitude of the locations during the training phase. r-MSSL$_{\text{cop}}$ also identified related regions, in terms of model skills, in the Gulf of Mexico. We hypothesize that no more connections were seen due to the high variability in air and sea

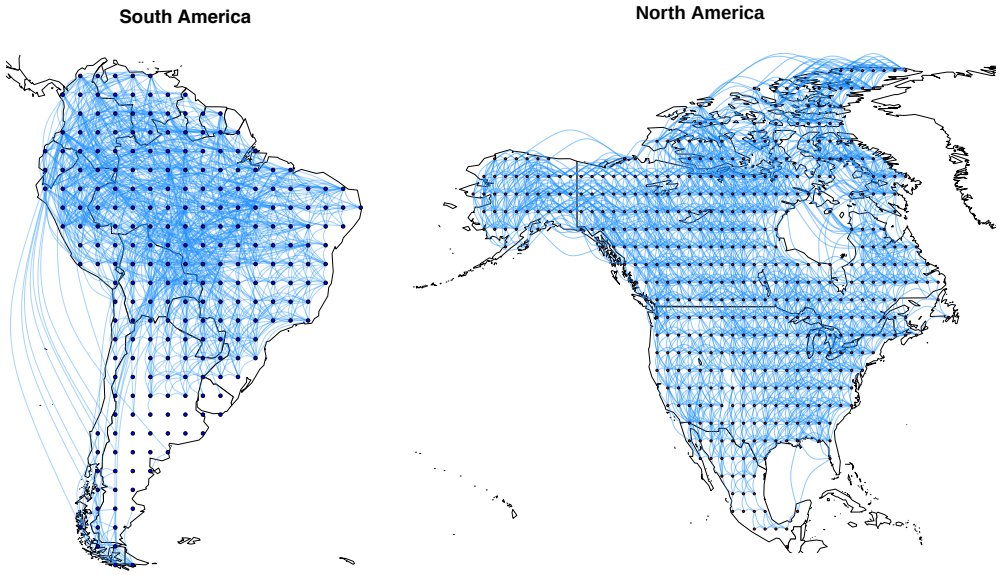


Figure 6: Relationships between geographical locations estimated by the r-MSSL$_{\text{cop}}$ algorithm using 120 months of data for training. The blue lines indicate that connected locations are conditionally dependent on each other. As expected, temperature is very spatially smooth, as we can see by the high neighborhood connectivity, although some long range connections are also observed.

surface temperature in these area (Twilley, 2001), that in turn has a strong impact on Gulf of Mexico costal regions.

4.1.3 MSSL sensitivity to initial values of $\mathbf{W}$

As discussed earlier, the MSSL algorithms may be susceptible to the choice of initial values of the parameters $\mathbf{\Omega}$ and $\mathbf{W}$, as the optimization function (3) is not jointly convex on $\mathbf{\Omega}$ and $\mathbf{W}$. In this section we analyze the impact of different parameter initializations on the RMSE and the number of non-zero entries in the estimated $\mathbf{\Omega}$ and $\mathbf{W}$ parameters.

Table 3 shows the mean and standard deviation over 10 independent runs with random initialization of $\mathbf{W}$ in the interval [-0.5,0.5] for the South America data set. For the $\mathbf{\Omega}$ matrix we started with an identity matrix, as it is reasonable to assume tasks independence beforehand. The results showed that the solutions are not sensitive to initial values of $\mathbf{W}$. The largest variation was found in the number of non-zero entries in the $\mathbf{\Omega}$ matrix for North America data set. However, it corresponds to 0.07% of the average number of non-zero entries and was not enough to significantly alter the RMSE of the solutions. Figure 7 shows the convergence of p-MSSL for several random initializations of $\mathbf{W}$. We note that in all runs the cost function decreases smoothly and similarly to each other, showing the stability of the method.

	Synt.	South America	North America
RMSE	1.14 (2e-6)	0.86 (0)	2.46 (1.6e-4)
# nz $\mathbf{W}$	345 (0)	2341 (0.32)	4758 (2.87)
# nz $\mathbf{\Omega}$	55(0)	4954 (0.63)	73520 (504.4)

Table 3: p-MSSL sensitivity to initial values of $\mathbf{W}$ in terms of RMSE, mean and standard deviation, and number of non-zero entries in $\mathbf{W}$ and $\mathbf{\Omega}$.

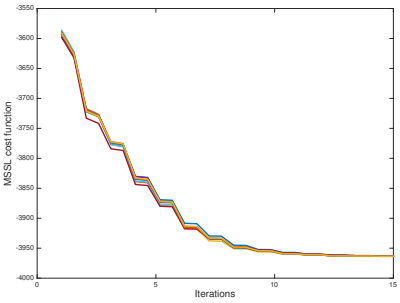


Figure 7: Convergence behavior of p-MSSL for distinct initializations of the weight matrix $\mathbf{W}$.

4.2 Classification

We test the performance of the p-MSSL on five data sets (six problems) described below. Recall that r-MSSL can not be applied for classification problems, once it relies on a Gaussian assumption of the residuals. This is currently the subject of an ongoing work. All data sets were standardized, then all features have zero mean and standard deviation one.

- **Landmine Detection:** Data from 19 landmine fields were collected, which have distinct characteristics. Each object in a given data set is represented by a 9-dimensional feature vector and the corresponding binary label (1 for landmine and 0 for clutter) (Xue et al., 2007). The feature vectors are extracted from radar images, concatenating four moment-based features, three correlation-based features, one energy ratio feature and one spatial variance feature. The goal is to classify between mine or clutter.

- **Spam Detection:** E-mail spam data set from ECML 2006 discovery challenge. It consists of two problems: Problem A with e-mails from 3 different users (2500 e-mails per user); and Problem B with e-mails from 15 distinct users (400 e-mails per user). We performed feature selection to get the 500 most informative variables using the Laplacian score feature selection algorithm (He et al., 2006). The goal is to classify between spam vs. ham. For both problems, we create different tasks for different users. This data set can be found at `http://www.ecmlpkdd2006.org/challenge.html`.

- **MNIST** data set consists of 28×28-size images of hand-written digits from 0 through 9. We transform this multiclass classification problem by applying the all-versus-all decomposition, leading to 45 binary classification problems (tasks). When a new test sample arrives, a voting scheme is performed among the classifiers. The number of samples for each classification problem is about 15000. This data set can be found at `http://yann.lecun.com/exdb/mnist/`.

- **Letter:** The handwritten letter data set consists of eight tasks, with each one being a binary classification of two letters: a/g, a/o, c/e, f/t, g/y, h/n, m/n and i/j. The input for each data point consists of 128 features representing the pixel values of the handwritten letter. The number of data points for each task varies from 3057 to 7931. This data set can be found at `http://ai.stanford.edu/~btaskar/ocr/`.

- **Yale-faces**: The face recognition data set contains 165 grayscale images with dimension 32x32 pixels of 15 individuals. Similar to MNIST, the problem is also transformed by all-versus-all decomposition, totalling 105 binary classification problems (tasks). For each task only 22 samples are available. This data set can be found at `http://vision.ucsd.edu/content/yale-face-database`.

Baseline algorithms: Four baseline algorithms were considered in the experiments and the regularization parameters for all algorithms were selected using cross-validation from the set {0.01, 0.1, 1, 10, 100}. The algorithms are:

1. **Logistic Regression (LR)**: learns separate logistic regression models for each task.
2. **MTL-FEAT** (Argyriou et al., 2007): employs an $\ell_{2,1}$-norm regularization term to capture the task relationship from multiple related tasks constraining all models to share a common set of features.
3. **CMTL** (Zhou et al., 2011b): incorporates a regularization term to induce clustering between tasks and then share information only to tasks belonging to the same cluster.
4. **Low rank MTL** (Abernethy et al., 2006): assumes that related tasks share a low dimensional subspace and applies a trace regularization norm to capture that relation.

Results: Table 4 shows the results of each algorithm for all data sets. It was obtained over 10 independent runs using a holdout cross-validation (2/3 for training and 1/3 for test). The performance of each run is measured by the average of the performance of all tasks.

Algorithms	Landmine	Spam 3-users	Spam 15-users	MNIST	Letter	Yale faces
LR	6.01 (0.37)	6.62 (0.99)	16.46 (0.67)	9.80 (0.19)	5.56 (0.19)	26.04 (1.26)
CMTL	5.98 (0.32)	3.93 (0.45)	8.01 (0.75)	2.06 (0.14)	8.22 (0.25)	9.43 (0.78)
MTL-FEAT	6.16 (0.31)	3.33 (0.43)	7.03 (0.67)	2.61 (0.08)	11.66 (0.29)	7.15 (1.60)
Trace	5.75 (0.28)	2.65 (0.32)	5.40 (0.54)	2.27 (0.09)	5.90 (0.21)	7.49 (1.72)
p-MSSL	**5.68** (**0.37**)	1.90* (0.27)	6.55 (0.68)	1.96* (0.08)	5.34* (0.19)	9.58 (0.91)
p-MSSL$_{\text{cop}}$	**5.68 (0.35)**	**1.77* (0.29)**	**5.32 (0.45)**	**1.95* (0.08)**	**5.29* (0.19)**	**5.28* (0.45)**

Table 4: Average classification error rates and standard deviation over 10 independent runs for all methods and data sets considered. Bold values indicate the best value and the symbol "*" means significant statistical improvement of the MSSL algorithm in relation to the contenders determined by t-test with $\mathrm{P} < 0.05$.

For all data sets p-MSSL$_{\text{cop}}$ presented statistically significant better results than the contenders for the most of the data sets. The three MTL methods presume the structure of the matrix $\mathbf{W}$, which may not be a proper choice for some problems. Such disagreement in the structure of the problem might explains the poor results in some data sets.

Focusing the analysis on p-MSSL and the copula version, p-MSSL$_{\text{cop}}$, their results are relatively similar for most of the data set, except for *Yale-faces*, where the difference is quite large. The two algorithms differ only in the way the inverse covariance matrix $\mathbf{\Omega}$ is computed. One hyphotesis for p-MSSL$_{\text{cop}}$ superiority on *Yale-faces* data set is that it may have captured hidden important dependencies among tasks, as the Copula Gaussian model can capture a wider class of dependencies than traditional Gaussian graphical models.

For the *Yale-faces* data set, which contains the smallest number of data available for training, all the MTL algorithms obtained considerable improvement compared to the traditional single task learning approach (LR), corroborating with the assertion that MTL approaches are particularly suitable for problems with few data samples.

5. Conclusion

We proposed a framework for multi-task structure learning. Our framework simultaneously learns the tasks and their relationship, with the task dependences defined as edges in an undirected graphical model. The formulation allows the direct extension of the graphical model to the recently developed semiparametric Gaussian copula models. As such model does not rely on the Gaussian assumption of task parameters, it gives more flexibility to capture hidden task conditional independence, thus helping to improve prediction accuracy. The problem formulation leads to a bi-convex optimization problem which can be efficiently solved using alternating minimization. We show that the proposed framework is general enough to be specialized to Gaussian models and generalized linear models. Extensive experiments on benchmark and climate data sets for regression tasks and real-world data sets for classification tasks illustrate that structure learning not only improves multi-task prediction performance, but also captures relevant qualitative behaviors among tasks.

Acknowledgments

We thank the AE and the three anonymous reviewers for their valuable comments and suggestions. The research was supported by NSF grants IIS-1029711, IIS-0916750, IIS-0953274, CNS-1314560, IIS-1422557, CCF-1451986, IIS-1447566, and by NASA grant NNX12AQ39A. AB acknowledges support from IBM and Yahoo. FJVZ thanks to CNPq for the financial support. ARG was supported by Science without Borders grant from CNPq, Brazil. Computing facilities were provided by University of Minnesota Supercomputing Institute (MSI).

References

J. Abernethy, F. Bach, T. Evgeniou, and J.P. Vert. Low-rank matrix factorization with attributes. Technical Report N-24/06/MM, Ecole des mines de Paris, France, 2006.

R. Ando, T. Zhang, and P. Bartlett. A framework for learning predictive structures from multiple tasks and unlabeled data. *J. of Machine Learning Research*, 6:1817–1853, 2005.

A. Argyriou, T. Evgeniou, and M. Pontil. Multi-task feature learning. In *Advances in Neural Information Processing Systems*, pages 41–50, 2007.

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

O. Banerjee, L. El Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *J. of Machine Learning Research*, 9:485–516, 2008.

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.

M. Bentsen et al. The Norwegian Earth System Model, NorESM1-M-Part 1: Description and basic evaluation. *Geoscience Model Development Discussion*, 5:2843–2931, 2012.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundation and Trends in Machine Learning*, pages 1–122, 2011.

V. Brovkin, L. Boysen, T. Raddatz, V. Gayler, A. Loew, and M. Claussen. Evaluation of vegetation cover and land-surface albedo in MPI-ESM CMIP5 simulations. *Journal of Advances in Modeling Earth Systems*, pages 48–57, 2013.

T. Cai, W. Liu, and X. Luo. A constrained ℓ_1 minimization approach to sparse precision matrix estimation. *J. of the American Statistical Association*, 106(494):594–607, 2011.

J. Chen, L. Tang, J. Liu, and J. Ye. A convex formulation for learning shared structures from multiple tasks. In *International Conference on Machine Learning*, pages 137–144, 2009.

J. Chen, J. Liu, and J. Ye. Learning incoherent sparse and low-rank patterns from multiple tasks. In *ACM Conference on Knowledge Discovery and Data Mining*, pages 1179–1188, 2010.

W.J. Collins et al. Development and evaluation of an Earth-system model–HadGEM2. *Geoscience Model Development Discussion*, 4:997–1062, 2011.

A.P. Dempster. Covariance selection. *Biometrics*, pages 157–175, 1972.

J.L. Dufresne et al. Climate change projections using the IPSL-CM5 Earth System Model: from CMIP3 to CMIP5. *Climate Dynamics*, pages 2123–2165, 2012.

T. Evgeniou and M. Pontil. Regularized multi–task learning. In *ACM Conference on Knowledge Discovery and Data Mining*, pages 109–117, 2004.

T. Evgeniou, C. A. Micchelli, and M. Pontil. Learning multiple tasks with kernel methods. *J. of Machine Learning Research*, 6:615–637, 2005.

J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9 3:432–441, 2008.

A.R. Gonçalves, P. Das, S. Chatterjee, V. Sivakumar, F.J. Von Zuben, and A. Banerjee. Multi-task Sparse Structure Learning. In *ACM International Conference on Information and Knowledge Management*, pages 451–460, 2014.

A.R. Gonçalves, F.J. Von Zuben, and A. Banerjee. A Multi-Task Learning View on Earth System Model Ensemble. *Computing in Science & Engineering*, pages 35–42, Dec. 2015.

H.B. Gordon et al. *The CSIRO Mk3 climate system model*, volume 130. CSIRO Atmospheric Research, 2002.

J. Gorski, F. Pfeuffer, and K. Klamroth. Biconvex sets and optimization with biconvex functions: a survey and extensions. *Mathematical Methods of Operations Research*, 66 (3):373–407, 2007.

X. He, D. Cai, and P. Niyogi. Laplacian score for feature selection. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems*, pages 507–514, 2006.

L. Jacob, F. Bach, and J.P. Vert. Clustered Multi-Task Learning: A Convex Formulation. In *Advances in Neural Information Processing Systems*, pages 745–752, 2008.

A. Jalali, P. Ravikumar, S. Sanghavi, and C. Ruan. A Dirty Model for Multi-task Learning. *Advances in Neural Information Processing Systems*, pages 964–972, 2010.

S. Ji and J. Ye. An Accelerated Gradient Method for Trace Norm Minimization. In *International Conference on Machine Learning*, pages 457–464, 2009.

Z. Kang, K. Grauman, and F. Sha. Learning with whom to share in multi-task feature learning. In *International Conference on Machine Learning*, pages 521–528, 2011.

M.G. Kendall. *Rank correlation methods*. Charles Griffin & Company, 1948.

S. Kim and E.P. Xing. Tree-Guided Group Lasso for MultiTask Regression with Structured Sparsity. In *International Conference on Machine Learning*, pages 543–550, 2010.

T.G.F. Kinel, P.E. Thornton, J. A. Royle, and T.N. Chase. Climates of the Rocky Mountains: historical and future patterns. *Rocky Mountain futures: an ecological perspective*, page 59, 2002.

A. Kumar and H. Daume III. Learning task grouping and overlap in multi-task learning. In *International Conference on Machine Learning*, pages 1383–1390, 2012.

S. L. Lauritzen. *Graphical Models*. Oxford University Press, Oxford, 1996.

H. Liu, J. Lafferty, and L. Wasserman. The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. *J. of Machine Learning Research*, 10:2295–2328, 2009.

H. Liu, F. Han, M. Yuan, J. Lafferty, and L. Wasserman. High Dimensional Semiparametric Gaussian Copula Graphical Models. *The Annals of Statistics*, 40(40):2293–2326, 2012.

K.V. Mardia and R.J. Marshall. Maximum likelihood estimation of models for residual covariance in spatial regression. *Biometrika*, 71(1):135–146, 1984.

A. J. McNeil and J. Nešlehová. Multivariate Archimedean copulas, $d-$monotone functions and ℓ_1-norm symmetric distributions. *The Annals of Statistics*, pages 3059–3097, 2009.

N. Meinshausen and P. Bühlmann. Stability selection. *Journal of the Royal Statistical Society B*, 72(4):417–473, 2010.

J.A. Nelder and R.J. Baker. *Generalized linear models*. Wiley Online Library, 1972.

G. Obozinski, B. Taskar, and M. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, pages 231–252, 2010.

T. Park and G. Casella. The Bayesian lasso. *J. of the American Statistical Association*, 103 (482):681–686, 2008.

P. Rai, A. Kumar, and H. Daume III. Simultaneously leveraging output and task structures for multiple-output regression. In *Advances in Neural Information Processing Systems*, pages 3185–3193, 2012.

V.A. Ramos. South America. In *Encyclopaedia Britannica Online Academic Edition*. Encyclopaedia Britannica, Inc., 2014. URL `http://www.britannica.com/EBchecked/topic/555844/South-America`.

A.J. Rothman, E. Levina, and J. Zhu. Sparse multivariate regression with covariance estimation. *J. of Computational and Graphical Statistics*, 19(4):947–962, 2010.

A. Sklar. *Fonctions de répartition à n dimensions et leurs marges*. Publ. Inst. Statis. Univ. Paris, 1959.

K. Subbian and A. Banerjee. Climate Multi-model Regression using Spatial Smoothing. In *SIAM International Conference on Data Mining*, pages 324–332, 2013.

Z.M. Subin, L.N. Murphy, F. Li, C. Bonfils, and W.J. Riley. Boreal lakes moderate seasonal and diurnal temperature variation and perturb atmospheric circulation: analyses in the Community Earth System Model 1 (CESM1). *Tellus A*, 64, 2012.

K.E. Taylor, R.J. Stouffer, and G.A. Meehl. An overview of CMIP5 and the experiment design. *Bulletin of the American Meteorological Society*, 93(4):485, 2012.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society B*, 58:267–288, 1996.

R.R. Twilley. Confronting climate change in the Gulf Coast region: Prospects for sustaining our ecological heritage, 2001.

H. Wang, A. Banerjee, C.J. Hsieh, P. Ravikumar, and I. Dhillon. Large scale distributed sparse precision estimation. In *Advances in Neural Information Processing Systems*, pages 584–592, 2013.

X. Wang, Zhang. C., and Z. Zhang. Boosted multi-task learning for face verification with applications to web image and video search. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 142–149, 2009.

W.M. Washington et al. The use of the Climate-science Computational End Station (CCES) development and grand challenge team for the next IPCC assessment: an operational plan. *Journal of Physics*, 125(1), 2008.

M. Watanabe et al. Improved climate simulation by MIROC5: Mean states, variability, and climate sensitivity. *Journal of Climate*, 23(23):6312–6335, 2010.

C. Widmer and G. Rätsch. Multitask learning in computational biology. *International Conference on Machine Learning - Work. on Unsupervised and Transfer Learning*, 27: 207–216, 2012.

L. Xue and H. Zou. Regularized rank-based estimation of high-dimensional nonparanormal graphical models. *The Annals of Statistics*, pages 2541–2571, 2012.

Y. Xue, X. Liao, L. Carin, and B. Krishnapuram. Multi-task learning for classification with Dirichlet process priors. *J. of Machine Learning Research*, 8:35–63, 2007.

M. Yang, Y. Li, and Z.M. Zhang. Multi-task learning with gaussian matrix generalized inverse gaussian model. In *International Conference on Machine Learning*, pages 423–431, 2013.

S. Yukimoto, Y. Adachi, and M. Hosaka. A new global climate model of the meteorological research institute: MRI-CGCM3: model description and basic performance. *Journal of the Meteorological Society of Japan*, 90:23–64, 2012.

L. Zhang, T. Wu, X. Xin, M. Dong, and Z. Wang. Projections of annual mean air temperature and precipitation over the globe and in China during the 21st century by the BCC Climate System Model BCC_CSM1. 0. *Acta Met. Sinica*, 26(3):362–375, 2012.

Y. Zhang and J.G. Schneider. Learning multiple tasks with sparse matrix-normal penalty. In *Advances in Neural Information Processing Systems*, pages 2550–2558, 2010.

Y. Zhang and D.-Y. Yeung. A convex formulation for learning task relationships in multi-task learning. In *Conference on Uncertainty in Artificial Intelligence*, pages 733–742, 2010.

J. Zhou, J. Chen, and J. Ye. *MALSAR: Multi-tAsk Learning via StructurAl Regularization.* Arizona State University, 2011a. URL `www.malsar.org`.

J. Zhou, J. Chen, and J. Ye. Clustered multi-task learning via alternating structure optimization. In *Advances in Neural Information Processing Systems*, pages 702–710, 2011b.

T. Zhou and D. Tao. Multi-task copula by sparse graph regression. In *ACM Conference on Knowledge Discovery and Data Mining*, pages 771–780, 2014.