Multi-Task Learning and Its Applications to Biomedical Informatics

by

Jiayu Zhou

A Dissertation Presented in Partial Fulfillment
of the Requirement for the Degree
Doctor of Philosophy

Approved May 2014 by the
Graduate Supervisory Committee:

Jieping Ye, Chair
Hans Mittelmann
Baoxin Li
Yalin Wang

ARIZONA STATE UNIVERSITY

August 2014

ABSTRACT

In many fields one needs to build predictive models for a set of related machine learning tasks, such as information retrieval, computer vision and biomedical informatics. Traditionally these tasks are treated independently and the inference is done separately for each task, which ignores important connections among the tasks. Multi-task learning aims at simultaneously building models for all tasks in order to improve the generalization performance, leveraging inherent relatedness of these tasks. In this thesis, I firstly propose a clustered multi-task learning (CMTL) formulation, which simultaneously learns task models and performs task clustering. I provide theoretical analysis to establish the equivalence between the CMTL formulation and the alternating structure optimization, which learns a shared low-dimensional hypothesis space for different tasks. Then I present two real-world biomedical informatics applications which can benefit from multi-task learning. In the first application, I study the disease progression problem and present multi-task learning formulations for disease progression. In the formulations, the prediction at each point is a regression task and multiple tasks at different time points are learned simultaneously, leveraging the temporal smoothness among the tasks. The proposed formulations have been tested extensively on predicting the progression of the Alzheimer's disease, and experimental results demonstrate the effectiveness of the proposed models. In the second application, I present a novel data-driven framework for densifying the electronic medical records (EMR) to overcome the sparsity problem in predictive modeling using EMR. The densification of each patient is a learning task, and the proposed algorithm simultaneously densify all patients. As such, the densification of one patient leverages useful information from other patients.

DEDICATION

I lovingly dedicate this thesis to my wife, Chenxue Huang, for her love, care and understanding every step of the way. I also dedicate this thesis to my parents and my parents-in-law, for their never ending support.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor, Dr. Jieping Ye, for his guidance, encouragement, and support during my dissertation research. He is an outstanding mentor, an easygoing friend, and the most dedicated researcher I have ever known. The experiences with him are my lifelong assets. I would like to thank my dissertation committee members, Dr. Hans Mittelmann, Dr. Baoxin Li and Dr. Yalin Wang, for their valuable interactions and feedback.

Members of our Machine Learning Lab inspired me a lot through discussions, seminars, and project collaborations, and I would like to thank the following people for their valuable interactions: Dr. Rita Chattopadhyay, Dr. Jianhui Chen, Dr. Pinghua Gong, Dr. Shuiwang Ji, Ji Liu, Dr. Jun Liu, Yashu Liu, Dr. Binbin Lin, Zhi Nie, Dr. Liang Sun, Qian Sun, Dr. Jie Wang, Dr. Zheng Wang, Shuo Xiang, Sen Yang, Dr. Lei Yuan, Dr. Chao Zhang.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Chapter 1

BACKGROUND AND INTRODUCTION

1.1 Multi-Task Learning

1.1.1 Early Works of Multi-Task Learning

In many machine learning tasks, the quality of a model is limited by information contained in the training data of the learning task. Examples of machine learning tasks are regression, classification (Li *et al.*, 2012), clustering (Chang *et al.*, 2013b), estimation of means (Feldman *et al.*, 2012), metric learning (Chang *et al.*, 2013a; Li *et al.*, 2013) and etc. The fundamental hypothesis of the multi-task learning is to assume that if tasks are *related* and then learning of one task can benefit from learning of other tasks.

Dating back to 1962, Zellner studied the seemingly unrelated regression equations (SURE), where there are a set of regression models (the learning of each regression model is a task) and Zellner proposed a procedure to perform the regressions simultaneously by applying Aitken's generalized least-squares, and showed that for general scenarios the proposed procedure is asymptotically more efficient than learning regression models independently (Zellner, 1962). The SURE models have high impact in the econometrics (Srivastava and Dwivedi, 1979), and is similar to the mutli-task learning, which use information from other tasks to improve efficiency rather than generalization performance.

Is it necessary for all tasks related in order to perform multi-task learning? The answer is no. This is rather surprisingly answer, known as the *Stein's paradox*: Dating back to 1956, Stein has shown that estimating the mean of one T distribution can

benefit from samples drawn from different means. Here consider the estimation of the mean of one T distribution is a learning task, and there are tasks for distributions with different means. And the implication is that the learning of one task can benefit from seemingly unrelated tasks. This problem was revisited in a recent study by Romera *et. al.* (Romera-Paredes *et al.*, 2012), which learned a set of related tasks with another group of unrelated tasks, aiming at transfer beneficial information from the unrelated tasks.

The multi-task learning is also motivated from the human *life-long learning* process (Thrun, 1996b): human beings encounter multiple learning tasks in their lifetime, and thus improve their ability to learn. Thrun defined machine learning algorithms that are capable of *learning to learn*: Given 1) a set of tasks, 2) training experience for each of these tasks, and 3) a set of performance measures (e.g., one for each task), the learning to learn algorithm is expected to have improved performance with *both* experience *and* the number of tasks. Such algorithms must be able to transfer knowledge from tasks to task and improve the expected task-performance.

Thrun and O'Sullivan (1996) proposed the task-clustering (TC) algorithm, which learns tasks into clusters. When a new task arrives, the TC algorithm firstly select the most related task and leverage knowledge only within the cluster. In (Caruana, 1997) Caruana for the first time formally defined the term multi-task learning and showed how multi-task learning works in neural network setting, and demonstrated that multi-task learning is effective in several real domains. Baxter approached multi-task learning in a Bayesian model (Baxter, 1997). Because that the tasks to be learned are sampled from a distribution over an environment or context of related tasks, the author modeled the environment using a shared objective prior distribution which is learned from the tasks. Moreover, the author provided some theoretical guarantees related to the multi-task learning in this context. The idea of obtaining a share prior

is also explored in (Mallick and Walker, 1997).

The formal definition of multi-task learning is given as follows:

Definition Multi-Task Learning (Caruana, 1997). Multitask Learning is an approach to inductive transfer that improves generalization by using the domain information contained in the training signals of related tasks as an inductive bias. It does this by learning tasks in parallel while using a shared representation; what is learned for each task can help other tasks be learned better.

The definition points out several key aspects of multi-task learning:

- Multi-task learning is one type of domain adaptation (or transfer learning) (Thrun, 1996a; Daumé III, 2007; Qi *et al.*, 2011), and belongs to inductive transfer (Baxter, 2000; Pan and Yang, 2010).
- Multi-task learning simultaneously learns tasks in parallel.
- Multi-task learning emphasizes on generalization performance of all tasks involved.

The *shared representation* has many different forms, as will elaborated later. It can be a shared feature representation in neural network, the same set of features in sparse linear models or the same subspace with different coefficients in low rank modeling.

We note that in the transfer learning, typically a source domain and a target domain are defined and we transfer knowledge from the source domain to the target domain. In the transfer learning, we only care the generalization of the target domain. In multi-task learning, however, because that we care the generalization performance of all tasks, each task is both a source domain (transferes knowledge to other tasks) and a target domain (use knowledge from other tasks).

1.1.2 Multi-Task Learning Frameworks

In this section, we show three main approaches for multi-task learning: neural network approach, bayesian approach, and regularization-based approach. We note that the three approaches are not mutually exclusive and they are overlapped in some ways, as elaborated later.

Neural Network Approach

Multi-task learning can be naturally incorporated in the context of neural networks. When building neural network models for multiple tasks, one may train one network for each task. In (Caruana, 1997), inductive transfer among the tasks is achieved by using multi-task neural network and used the multi-task ANN: training one ANN with a set of output nodes (one for each task) and all outputs are fully connected to a hidden layer that they share. The shared hidden layers serve as the shared (low dimensional) representation that transferes knowledge between tasks. The figure is shown in 1.1.

Baxter (1997) proposed to use Bayesian model of multi-task learning and illustrate how the Bayesian inference can be done in the neural network for learning low dimensional representation. Heskes presented a practical implementation of Baxter's neural network framework for multi-task learning in (Heskes *et al.*, 2000). Bakker and Heskes (2003) offered a neural network model to perform task clustering and gating, in which parameters of the hidden low dimensional feature space are shared for all tasks (as in (Baxter, 1997)) and output model parameters are connected using a joint prior distribution learned from the data. Heskes *et al.* (1998) proposed to solve a huge number of similar tasks by combining the neural network and hierarchical Bayesian approach. In (Collobert and Weston, 2008), Collobert and Weston proposed a unified

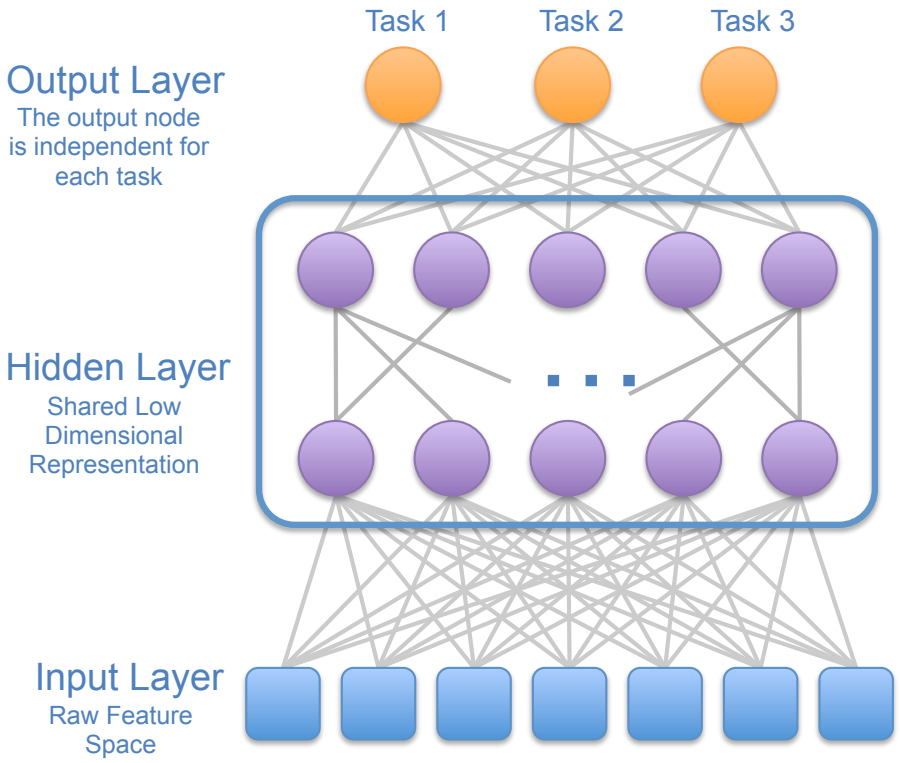


Figure 1.1: Illustration of Multi-Task Neural Network (MTNN) (Caruana, 1997). There are three layers in the MTNN. In the input layer, all tasks have the same feature space; in the hidden layer, all tasks share the same low-dimensional representation; in the output layer, each task has an output node that is independent from other nodes.

convolutional neural network (CNN, one type of deep NN) for different natural language processing (NLP) tasks. The proposed network jointly learned all tasks using weight-sharing strategy, and the multi-task model was shown to have significantly outperformed the stat-of-the-art performance.

Wilson *et al.* (2012) introduced the Gaussian Process Regression Networks (GPRN), which combines the structural properties of Bayesian neural networks and the non-parametric flexibility of Gaussian processes. The CPRN can be considered as a mixture of Gaussian processes.

Hierarchical Bayesian and Random Process Approach

One important approach of the multi-task learning the Hierarchical Bayesian (HB) approach, which places common priors on the hyperparameters of the task models to model task relatedness. The random process such as Gaussian process (GP) and Dirichlet process (DP) can be used to model the multi-task learning. Unlike the Bayesian approach, in which priors are placed in parameters, in random process models directly assume prior over functions (Rasmussen and Williams, 2006). The methods in this approach capture correlation between outputs/responses of the related tasks, and the correlation can be used to improve the performance of these tasks.

Hierarchical Bayesian

In the Bayesian approach, the model parameters θ is a multivariate random variable, and during the learning we estimate the posterior density $p(\theta|\mathcal{D})$ from data $\mathcal{D}$ and the prior $p(\theta)$:

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{p(\mathcal{D})}$$

In traditional Bayesian inference, the prior $p(\theta)$ reflects our (weak) confidence about the parameter θ, and thus is called *subject prior*. In the multi-task learning, it is reasonable to assume that all tasks are sampled from an environment, which describes how tasks are related (Baxter, 1997). Since we are learning a set of related tasks, and it is possible to learn an *objective prior* that reflects the environment $p(\theta|\pi^*)$. We are able to infer the posterior probability using the Bayes' rule:

$$p(\pi|\theta^n) = \frac{p(\theta^n|\pi)p(\pi)}{p(\theta^n)}.$$

which indicates that as $n \to \infty$, then the posterior asymptotically converges to the true posterior π^*. Therefore, in the context of HB, parameters for different tasks are

assumed to be drawn from a common hyper prior distribution (π), which enables the knowledge transfer among tasks and through which the tasks regularize each other.

Baxter (1997) formally introduced the objective prior in the multi-task learning and the true prior can be learned using Bayesian inference on neural network. The author provided bounds from the perspective of information theory, showing how much information is needed to learn a task when it is learned simultaneously with other tasks. The bounds also showed that sampling from multi-tasks can be highly beneficial when we have little information about the true prior while its dimensionality is small.

Gaussian Process

Minka and Picard (1997) firstly inspected the relationship between GP and neural network model in (Baxter, 1997) and linked the multi-task learning problem to the fitting of the covariance matrix in a GP. In the paper the authors also introduces how data from tasks can be automatically separate using a mixture model, and discussed the issue of performing task clustering. Learning multi-task covariance matrices are expensive, and in (Lawrence and Platt, 2004) the authors used sparse approximation of GP and provided a more general GP approach for multi-task learning with parametric covariance function. By assuming that the training sets of tasks are independent, the covariance is thus block diagonal, and the authors applied standard information vector machine (Lawrence *et al.*, 2003) algorithm to estimate the parameters using the maximum likelihood. In (Schwaighofer *et al.*, 2004) Scwaighofer *et. al.* introduced a GP that firstly considered the (non-parametric) covariance matrix of GP, followed by a second extrapolation step to learn kernel functions. In (Yu *et al.*, 2005), Yu *et. al.* exploited the equivalence between parametric linear and non-parametric CP, and introduced a hierarchical Bayesian model to learn multiple tasks with a Normal-Inverse-Wishart prior, and proposed an EM-algorithm to solve

the model. The model was later applied to solve stochastic relational models involving multiple related GPs (Yu *et al.*, 2006). Teh *et. al.* introduced a semi-parametric latent factor model (Seeger *et al.*, 2005), which assumed that the multiple related response variables came from a linearly mix of a set of Gaussian processes. The authors also presented an efficient algorithm that has linear complexity w.r.t. the number of training samples. In (Bonilla *et al.*, 2007), Bonilla *et. al.* proposed Gaussian process two models to perform multi-task learning when task-specific features present (the setting in (Bakker and Heskes, 2003)): one approach combining data from different tasks and one combining models. In order to identify outlier/irrelevant tasks, Yu *et. al.* introduced the t-processes (TP) which allows robust multi-task learning (Yu *et al.*, 2007), which is capable of identifying outlier tasks. Bonilla *et. al.* introduced a multi-task GP approach that directly induce correlations between task (Williams *et al.*, 2007). The approach assumes that the covariance matrix is consist of two components: one PSD matrix that models inter-task similarities, and a parametric covariance function. This approach was later applied to model robot inverse dynamics (Williams *et al.*, 2008). Zhang and Yeung (Zhang and Yeung, 2010b) proposed to extend the covariance matrix in (Williams *et al.*, 2007) by considering it to be a random matrix with an inverse-Wishart prior, leading to a multi-task generalized t-process. Lazaric and Ghavamzadeh (Lazaric and Ghavamzadeh, 2010) considered a bayesian approach for multi-task reinforcement learning, which assumes that the value functions of different tasks are all sampled from a common Gaussian process prior. To overcome the problem of computational complexity, Pillonetto *et. al.* offered a Bayesian online multi-task learning of Gaussian processes. The focused GP, proposed in (Leen *et al.*, 2012), introduced an "explaining away" model for each of the additional tasks to model their non-related variation, in order to focus the transfer to the task-of-interest. Swersky *et. al.* (Swersky *et al.*, 2013) applied the frame-

work of Bayesian optimization on the multi-task GP, which significantly reduced the computational costs of the optimization process.

Chai quantified the generalization error and learning curve for the multi-task Gaussian Process in the asymmetric two multi-task scenario (Chai, 2009, 2010). The learning curve with arbitrary number of tasks is studied in (Sollich and Ashton, 2012).

Dirichlet Process

In (Yu *et al.*, 2004), Yu *et. al.* introduced a nonparametric hierarchical Bayesian framework for information filtering. Learning preference models for each user can be considered as task, and the proposed method proposed a nonparametric common prior among the tasks, assuming a sample is generated from a Dirichlet process (DP). However, the approximate DP prior in (Yu *et al.*, 2004) cannot be used to improve the generalization performance of multiple tasks when learning them together, and in (Xue *et al.*, 2007b), Xue *et. al.* introduced a multi-task classification framework using Dirichlet process priors, which can learn similarity between tasks and thus can obtain task clusters. In (Xue *et al.*, 2007a), Xue *et. al.* proposed a new matrix stick-breaking process (MSBP) to perform multi-task learning. The MSBP improved the DP prior by allowing 'local clustering' over different feature components. An *et. al.* extended the MSBP to incorporate kernels and applied the proposed kernel stick-breaking process to perform image analysis (An *et al.*, 2008). While aforementioned methods model the multi-task learning when data from all tasks is available, Ni *et. al.* studied the multi-task learning model for sequential data (Ni *et al.*, 2007), in which the authors imposed a nested Dirichlet process (nDP) prior on the base distribution of the infinite hidden Markov model (iHMM).

In (Li *et al.*, 2011), Li *et. al.* proposed a nonparameteric bayesian multi-task learning method with (cluster-wise) feature selection. The model is achieved by employing a DP and beta-Bernoulli process (BBP), where the DP clusters the tasks into

groups, and for each BBP selects features that are relavant to the group. Passos *et. al.* (Passos *et al.*, 2012) offered a flexible nonparametric Bayesian model, which used DP and *Indian Buffet Process*/Beta Process so that the number of mixture components and the number of latent dimension do not need to be specified *a prior*. Gupta *et. al.* (Gupta *et al.*, 2013) offered the factorial multi-task learning method, which clusters the tasks by their relatedness in a subspace and enables different relatedness by sharing the subspace across the groups. This is done by a nonparametric prior that extends the beta process prior using a DP, which allows infinite child beta processes.

The multi-task learning with DP priors was applied to compressive sensing (Qi *et al.*, 2008), in which each task is defined to be a compressive sensing problem. Li *et. al.* offered a multi-task reinforcement learning, in which the modeling the agent's behavior in each environment is a task and is given by a parametric model (Li *et al.*, 2009). The authors imposed the nonparametric DP prior on the model parameters to transfer knowledge between tasks.

Other Hierarchical Bayesian Approaches

Besides the random process approaches, there are other hierarchical Bayesian approaches. Heskes *et al.* (1998) proposed a multi-task learning approach that combines neural network and hierarchical Bayesian approach. In (Arora *et al.*, 1998), Arora *et. al.* introduced a hierarchical Bayesian model for marketing, which considered both the primary demand and the second demand. Consider the modeling of each of these demands as task, the model is to transfer knowledge among the tasks using a hierarchical Bayesian model. In (Bakker and Heskes, 2003) Bakker and Heskes used a hierarchical Bayesian approach to learn multiple tasks, in which some of the model parameters are shared and others are related through a prior distribution. Müller *et. al.* (Müller *et al.*, 2004) proposed a nonparametric hierarchical model, combining in-

ference across related nonparametric Bayesian, with a special case of Dirichlet process mixtures. In (Zhang *et al.*, 2005), Zhang *et. al.* (Zhang *et al.*, 2008) assumed that the tasks parameters are generated from independent sources and the tasks are related through these latent sources. They thus proposed a probabilistic multi-task learning model based on Independent Component Analysis. The authors later extended the latent variable approach to handle different relatedness. In (Daumé III, 2009), Daume III offered a Bayesian latent hierarchical model for multi-task learning, with shared covariance structure across tasks. The model subsumed (Yu *et al.*, 2005) and (Xue *et al.*, 2007b) as special cases. Rai (Rai and Daume, 2010) proposed a nonparametric Bayesian multi-task learning model that assumed tasks parameters share a latent subspace, the same assumption as in (Ando and Zhang, 2005).

Ji *et. al.* (Ji *et al.*, 2009) proposed to use hierarchical Bayesian model to learn multiple compressive sensing tasks, using a common prior (Gamma prior) on the hyper-parameters. In (Hernández-Lobato *et al.*, 2010), Hernández-Lobato *et. al.* introduced a Bayesian model for multi-task feature selection, which utilizes a general *spike and slab sparse prior* to enforce the selection of a common set of features different across tasks. Titsias and Lázaro-gredilla (Lázaro-gredilla and Titsias, 2011) proposed a variational Bayesian inference for multi-task and multiple kernel learning, based on the spike and slab prior. Hernández-Lobato *et. al.* (Hernandez-Lobato and M. Hernandez-Lobato, 2013) proposed to use the *horseshoe prior* to learn dependencies in the process of identifying relevant features for prediction.

While most nonparametric Bayesian multi-task models influence posterior by imposing common priors on the model parameters or functions, Zhu *et. al.* (Zhu *et al.*, 2011) proposed to impose posterior regularization by combining the large-margin idea, learning predictive latent features.

Regularized Linear Approach

The linear models assume that the response is a function of the linear combination of the input. Linear models are simple and yet powerful in that flexible regularization can be designed according to desired structures and assumptions. Moreover, the point estimation of linear models often yields efficient optimization algorithms.

In the multi-task learning, there are a significant amount of research efforts belonging to this approach, in which the regularization terms are designed to bridge the tasks and transfer knowledge between the tasks. In nature, most of the regularization terms are coming from our prior knowledge about the models, and however the exact probabilistic interpretations for many regularization approaches are unknown.

$$\min_{W} \sum_{i=1}^{T} \ell(X_i, y_i, w_i) + \mathcal{R}(W) \tag{1.1}$$

where the regularization function $\mathcal{R}(.)$ encourages the shared representation among the tasks.

1.1.3 Task Relatedness

The key of the multi-task learning is to connect the tasks via a shared representation, which in turn benefits (via bias) the tasks to be learned. Each shared representation encode certain assumptions on the task relatedness. In this section, we review common assumptions and their associated representations. The realization of these share representation can done using the approaches as mentioned in the previous section.

Common Prior

One straight-forward assumption on the multiple related tasks is that the parameters of different tasks come from a common prior. This is the assumption in most hierarchical Bayesian approaches in the previous section. If the task parameters come from a Gaussian distribution, then they should be close to some mean values. We can decompose the parameters into two parts $w_t = w_0 + v_t$, i.e., the mean and how a task deviate from the mean. Evgeniou and Pontil (Evgeniou and Pontil, 2004; Evgeniou *et al.*, 2005) proposed a regularization-based approach to explicit model the two components and learn them from the training data.

Low-Dimensional Subspace

In many real-world applications, forcing the tasks from the same distribution may be too restrictive. Instead of assuming that all tasks share the same prior, we can assume the that there are some latent variables and the tasks are related via the latent variables.

In the context of linear model, Ando and Zhang (Ando and Zhang, 2005) assumed that the tasks share a latent low-dimensional subspace and proposed an Alternating Structure Optimization (ASO) approach explicitly learn this subspace in the learning formulation. The formulation of ASO is non-convex and in (Chen *et al.*, 2009), Chen *et. al.* proposed a convex relaxation of ASO. In (Xu and Lafferty, 2012), Xu ad Lafferty proposed a model that assumed that the group of regression models shared a common low-rank matrix dictionary and the model matrix is a sparse combination of the dictionaries. The shared representation was also explored in the multi-task clustering (Gu and Zhou, 2009), and nonparametric hierarchical Bayesian (Rai and Daume, 2010).

A closely related approach is multi-task feature learning (Argyriou *et al.*, 2008e; Evgeniou and Pontil, 2007; Argyriou *et al.*, 2008a), which learns a feature mapping from the original feature space and then enforces all tasks to select a shared subset of features after mapping. The formulation leads to a low-rank structure on the model matrix W. Formulations directly involving on the rank function are intractable, and the trace norm regularization is used as a convex alternative (Amit *et al.*, 2007; Ji and Ye, 2009; Pong *et al.*, 2010).

In some scenarios the matrix W may be close to but not low-rank, and thus assuming the matrix W is low-rank may be too restrictive. In (Chen *et al.*, 2010a, 2012a), Chen *et. al.* offers an approach to decompose W into a low rank matrix and a sparse matrix. Another extension is to assume the model matrix W is both low-rank and sparse (Mei *et al.*, 2012; Chen and Ye, 2013). In (Argyriou *et al.*, 2008b,c), Argyriou *et. al.* proposed to first cluster samples into groups and encourage a shared representation within the groups. Kang *et. al.* (Kang *et al.*, 2011) offered an approach that simultaneously learned shared feature representation and modeled task relatedness.

Shared Feature Subset

Another popular approach for modeling task relatedness is to assume that tasks have a shared subset of features, or joint feature learning. For linear models, the joint feature learning can be done via imposing a group lasso on the model matrix W (Jebara, 2004; Turlach *et al.*, 2005; Yuan and Lin, 2006; Obozinski *et al.*, 2006, 2010), where each column of W is treated as a group, and the corresponding feature is selected for all tasks if the group is non-zero after learning. The tasks can be either homogenous or heterogenous (involving regression and classification tasks) (Yang *et al.*, 2009), as

long as they have the same feature space.

$$\mathcal{R}(W) = \lambda \|W\|_{1,q} = \lambda \sum_{i=1}^{d} \|w^i\|_q$$

Given $q \geq 1$ and the loss function is convex, then the group Lasso problem with the above group Lasso regularization is also convex, leading to efficient algorithms to obtain optimal solutions (Liu *et al.*, 2009a,b; Quattoni *et al.*, 2009). And the regularization has equipped with a probabilistic interpretation (Zhang *et al.*, 2010). Otherwise the problem is non-convex and is discussed in (Rakotomamonjy *et al.*, 2011). From the perspective of theory, Lounici (Lounici *et al.*, 2009) showed that the joint feature learning formulation enjoys nice sparsity oracle inequalities and variable selection properties. Also, the union suport recovery of these joint feature learning formulations was studied in (Kolar *et al.*, 2011), offering analysis on properties of different regularizations. Being a group Lasso problem, the joint feature learning can be efficiently done in an online fashion (Yang *et al.*, 2010).

Models combining the feature learning and joint feature selection are offered in (Argyriou *et al.*, 2008e; Evgeniou and Pontil, 2007; Argyriou *et al.*, 2008a), which learn a shared feature mapping and enforce all the tasks to select the same set of features after mapping. In (Jalali *et al.*, 2010), Jalali *et. al.* proposed a dirty model for joint feature learning, assuming that the underlying model matrix W is corrupted. The model decomposes the model matrix into two components: a clean joint sparse model and a component with element-wise sparsity. In (Xu and Leng, 2012) Xu and Huan considered a joint feature learning assumption where the responses are corrupted by gross sparse error, and designed a learning formulation that is robust to the sparse error.

The joint feature learning model has many extensions and relevant multi-task learning models. Usually the regularization parameter for the group Lasso is set ei-

ther manually or determined using cross validation. In (Lee *et al.*, 2010), Lee *et. al.* proposed to adaptively learn the parameters and incorporate external knowledge. Swirszcz and Lozano (Swirszcz and Lozano, 2012) offered a multi-level lasso that has two levels of sparsity and one of which enables joint feature selection. In (Balasubramanian *et al.*, 2013), the authors presented a method to consider the joint feature selection from another perspective: perform joint selection based on a random effects model. Jebara (Jebara, 2011) proposed to perform multi-task joint feature selection under the framework of maximum entropy discrimination. Another multi-task learning approach that imposes an totally different assumption: encourage features to share different sets of features, called exclusive Lasso and was proposed in (Zhou *et al.*, 2010).

The joint feature learning can also be done in the hierarchical Bayesian (Hernández-Lobato *et al.*, 2010) and probabilistic framework (Xiong *et al.*, 2007; Zhang *et al.*, 2010). In (Hernández-Lobato *et al.*, 2010), Hernández-Lobato utilizes a *spike and slab* sparse prior to achieve common feature selection among tasks. Zhang *et. al.* (Zhang *et al.*, 2010) considered a family of $\ell_{1.q}$ norm instead of selecting a specific norm.

The joint feature learning has a wide range of applications because of its excellent interpretation. For example, Tomioka and Haufe applied group sparsity in the area of Brain Computer Interface (Tomioka and Haufe, 2008); Rao (Rao *et al.*, 2013) designed a sparse overlapping sets Lasso for fMRI analysis.

Learning task relationship

In most task relatedness methods, the tasks are assumed to be equally related, i.e., each task contributes equally to the shared representation, and the tasks are equally related. In many real-world applications such an assumption is too strong, which leads to a significant amount of efforts on the study of task relationship. While in some

applications, we may be able to come up with similarity among tasks using domain knowledge (e.g., when a task network is available (Kato *et al.*, 2007)). When such side information is not available, however, one is able to use a data driven approach to learn task relationships from data. The approaches can be grouped into the following categories:

Identifying outlier tasks. In this class of approaches the models assume that most of the tasks are related to each other, while there are a few *outlier* tasks that do not relate to other tasks. The approaches are called *robust multi-task learning*, aiming to be robust against outlier tasks. In the shared low-dimensional subspace setting (Chen *et al.*, 2011), we can assume that the model W can be decomposed into two parts, where in one part is the low-dimensional subspace for all tasks, and the other part captures the information that cannot fit into the shared subspace. A similar robust models for feature learning is offered by Gong *et. al.* (Gong *et al.*, 2012). The robust model can also be used in the Bayesian setting (Yu *et al.*, 2007).

Learning task clusters. In this approach we assume that the tasks form some clusters and the tasks are related to each other via the clusters. Within each cluster the tasks are related and share a common representation. In early researches, this was done in a two step fashion, where the tasks are first clustered and then learned within the tasks (Thrun and O'Sullivan, 1996, 1998; Bakker and Heskes, 2003). In recent researches the task cluster and model inference can be done simultaneously (Jacob *et al.*, 2008; Zhou *et al.*, 2011a; Zhong and Kwok, 2012). In (Zhou *et al.*, 2011a) the authors established an equivalent relationship between the alternating structure optimization for shared subspace learning and the clustered multi-task learning, which revealed the close underlying connection between the two seemingly unrelated multi-task learning formulations. In (Jacob *et al.*, 2008; Zhou *et al.*, 2011a), the authors offers clustered multi-task learning formulations which simultaneously performed model learning and

soft k-means clustering on the model. In (Kumar and Daume, 2012), Kumar *et. al.* offered a task grouping method by decomposing the model matrix and applying sparse inducing norm. Jawanpuria and Nath (Jawanpuria and Nath, 2012) offered a mixed norm regularization approach that searches the exponentially large space of all possible task groups and allows shared feature space within groups. In (Passos *et al.*, 2012), Passos *et. al.* proposed a model for learning latent tasks structures for multi-task learning, which seek a proper latent structure and subsumed many multi-task learning formulations including the clustered multi-task learning.

Modeling task similarity. A more general way is to directly learn a task covariance (or similarity) matrix, which evaluates the pairwise relationship between all tasks. In this approach, the outlier tasks and task clusters can be naturally given in the covariance matrix. When such a matrix is given, it can be incorporated in the learning formulations (Han *et al.*, 2010). It is more interesting to infer such a covariance matrix from data. When task-specific features are available, these features can be used to evaluate task similarity (Bonilla *et al.*, 2007; Yu *et al.*, 2006, 2009). While in some applications of multi-task learning, these task-specific features are not available, and therefore the task similarity is evaluated from the entire data directly (Williams *et al.*, 2007).

In (Williams *et al.*, 2007), Bonilla *et. al.* used the covariance matrix for task relationship for not only positive correlation between the tasks, but also negative ones. The authors proposed to use a low rank approximation of the task covariance to reduce computational complexity. In (Zhang and Yeung, 2010a) Zhang and Yeung proposed a convex multi-task relationship learning formulation to learn the covariance matrix. In (Zhang and Yeung, 2010c), the authors proposed to use task covariance matrix to perform transfer learning among metric learning tasks. Rai *et. al.* (Rai *et al.*, 2012) offered a model based on the conditional covariance structure, which

subsumed (Zhang and Yeung, 2010a) as a special case. In (Zhang and Schneider, 2010), Zhang and Schneider proposed to use sparse inverse covariance to couple the multiple tasks, which captures both task relatedness and feature representation. Fei and Huan (Fei and Huan, 2011) proposed to consider structured feature selection along with the learning of the task relatedness. In (Zhang and Yeung, 2013), Zhang and Yeung offered a learning formulation that considered high-order task relationships instead of the pairwise relationships. The task similarity is also extensively explored in the Bayesian settings (Williams *et al.*, 2007; Passos *et al.*, 2012; Yang and He, 2013; Gupta *et al.*, 2013; Yang *et al.*, 2013).

1.1.4 Other Task Relatedness

In addition to the aforementioned methods, there are many other approaches for modeling task relatedness: the models come from a manifold (Lin *et al.*, 2012; Agarwal *et al.*, 2010; Lin *et al.*, 2012); tasks have hierarchical structure (Daumé III, 2009; Görnitz *et al.*, 2011) or tree/graph structure (Kim and Xing, 2010; Chen *et al.*, 2010b; Kim and Xing, 2012; Chen *et al.*, 2012b; Widmer *et al.*, 2012).

1.2 Disease Progression via Multi-Task Learning

Alzheimer's disease (AD), the most common type of dementia, is characterized by the progressive impairment of neurons and their connections resulting in loss of cognitive function and ultimately death (Khachaturian, 1985). AD currently affects about 5.3 million individuals in United States and more than 30 million worldwide with a significant increase predicted in the near future (Association, 2010). Alzheimer's disease has been not only the substantial financial burden to the health care system but also the psychological and emotional burden to patients and their families. As the research on developing promising new treatments to slow or prevent AD progressing,

the need for markers that can track the progress of the disease and identify it early becomes increasingly urgent.

A definitive diagnosis of AD can only be made through an analysis of brain tissue during a brain biopsy or autopsy (Jeffrey *et al.*, 2003). Many clinical/cognitive measures have been designed to evaluate the cognitive status of the patients and used as important criteria for clinical diagnosis of probable AD, such as Mini Mental State Examination (MMSE) and Alzheimer's Disease Assessment Scale cognitive sub-scale (ADAS- Cog) (McKhann *et al.*, 1984). MMSE has been shown to be correlated with the underlying AD pathology and progressive deterioration of functional ability (Jeffrey *et al.*, 2003). ADAS-Cog is the gold standard in AD drug trial for cognitive function assessment (Rosen *et al.*, 1984). Since neurodegeneration of AD proceeds years before the onset of the disease and the therapeutic intervention is more effective in the early stage of the disease, there is thus an urgent need to address two major research questions: (1) how can we predict the progression of the disease measured by cognitive scores, e.g., MMSE and ADAS-Cog? (2) what is the smallest set of features (measurements) most predictive of the progression? The prime candidate markers for tracking disease progression include neuroimages such as magnetic resonance imaging (MRI), cerebrospinal fluid (CSF), and baseline clinical assessments (Dubois *et al.*, 2007).

The relationship between the cognitive scores and possible risk factors such as age, APOE gene, years of education and gender has been previously studied (Tombaugh, 2005; Ito *et al.*, 2010). Many existing works analyzed the relationship between cognitive scores and imaging markers based on MRI such as gray matter volumes, density and loss (Apostolova *et al.*, 2006; Chetelat and Baron, 2003; Frisoni *et al.*, 2002, 2010; Stonnington *et al.*, 2010), shape of ventricles (Ferrarini *et al.*, 2008; Thompson *et al.*, 2004) and hippocampal (Thompson *et al.*, 2004) by correlating these features with

baseline MMSE scores. In (Duchesne *et al.*, 2009), the intensity and volume of medial temporal lobe altogether with other risk factors and the gray matter were shown to be correlated with the 6-month MMSE score, which allowed us to predict near-future clinical scores of patients. Relations between 6-month atrophy patterns in medial temporal region and memory declination in terms of clinical scores had also been examined in (Murphy *et al.*, 2010). To predict the longitudinal response to Alzheimer's Disease progression, Ashford and Schmitt built a model with horologic function using "time-index" to measure the rate of dementia progression (Ashford and Schmitt, 2001). In (Davatzikos *et al.*, 2009), the so-called SPARE-AD index was proposed based on spatial patterns of brain atrophy and its linear effect against MMSE was reported. In a more recent study by Ito *et al.*, the progression rate of cognitive scores was modeled using power functions (Ito *et al.*, 2010).

Most existing work employed either the regression model (Duchesne *et al.*, 2009; Stonnington *et al.*, 2010) or the survival model (Vemuri *et al.*, 2009) for modeling the disease progression. The correlation between the ground truth and the prediction is used to evaluate the model (Duchesne *et al.*, 2009; Stonnington *et al.*, 2010). When the size of covariates is small, each covariate can be individually added to the model to examine its effectiveness for predicting the target (Ito *et al.*, 2010; Walhovd *et al.*, 2010), or univariate analysis is performed individually on all covariates and those who exceed a certain significance threshold are included in the model (Murphy *et al.*, 2010). When the number of covariates is large and significant correlations among covariates exist, these approaches are suboptimal. To deal with the curse of dimensionality, dimension reduction techniques are commonly employed. Duchesne *et al.* used principle components analysis (PCA) to build a low dimensional feature space from image data (Duchesne *et al.*, 2009). An obvious disadvantage of dimension reduction techniques such as PCA is that the model is no longer interpretable,

since all features are involved. Stonnington *et al.* used relevance vector regression (RVR), which integrated feature selection in the training stage (Stonnington *et al.*, 2010). These approaches only predict clinical scores at a single time point and their performances are far from satisfactory to be clinically useful for AD prognosis.

In this thesis, I propose a multi-task learning formulation for predicting the progression of the disease measured by the clinical scores at multiple time points and simultaneously selecting markers predictive of the progression. Specifically, I formulate the prediction of clinical scores at a sequence of time points as a multi-task regression problem, where each task concerns the prediction of a clinical score at one time point. Multi-task learning aims at improving the generalization performance by learning multiple related tasks simultaneously. The key of multi-task learning is to exploit the intrinsic relatedness among the tasks. For the disease progression considered in this thesis, it is reasonable to assume that a small subset of features is predictive of the progression, and the multiple regression models from different time points satisfy the smoothness property, that is, the difference of the cognitive scores between two successive time points is small. To this end, I develop a novel multi-task learning formulation based on a temporal group Lasso regularizer. The regularizer consists of two components including an $\ell_{2,1}$-norm penalty (Yuan and Lin, 2006) on the regression weight vectors, which ensures that a small subset of features will be selected for the regression models at all time points, and a temporal smoothness term, which ensures a small deviation between two regression models at successive time points.

I have performed extensive experimental studies to evaluate the effectiveness of the proposed algorithm. I use various types of data from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database including MRI scans, CSF, and clinical scores at the baseline to predict the MMSE and ADAS-Cog scores for the next three

years. Our experimental studies show that the proposed algorithm better captures the progression trend and the cross-sectional group differences of AD severity than existing methods. Results also show that most markers selected by the proposed algorithm are consistent with findings from existing cross-sectional studies.

1.3 Multi-Task Learning for Patient Record Densification

Patient *Electronic Medical Records* (EMR) are systematic collections of longitudinal patient health information generated from one or more encounters in any care delivery setting. Typical information contained in EMR includes patient demographics, encounter records, progress notes, problems, medications, vital signs, immunizations, laboratory data and radiology reports, and etc. Effective utilization of EMR is the key to many medical informatics research problems, such as disease early detection (Wu *et al.*, 2010), comparative effectiveness research (Markatou *et al.*, 2012) and risk stratification (Persell *et al.*, 2009).

Working directly with raw EMR is very challenging because it is usually sparse, noisy and irregular. Deriving better and more robust representation of the patients, or phenotyping, is very important in many medical informatics applications (Lasko *et al.*, 2013). One significant challenge for phenotyping with longitudinal EMR is *data sparsity*. To illustrate this, we show the EMR of a Congestive Heart Failure (CHF) patient in Fig.1.2, which is represented as a matrix. The horizontal axis is time with the granularity of days. The vertical axis is a set of medical events, which in this example is a set of diagnosis codes. Each dot in a matrix indicates that the corresponding diagnosis is observed for this patient at the corresponding day. From the figure we can see that there are only 37 nonzero entries within a 90-day window.

With those sparse matrices, many existing works just treat those zero values as actual zeros (Wu *et al.*, 2010; Wang *et al.*, 2012; Sun *et al.*, 2012), and construct

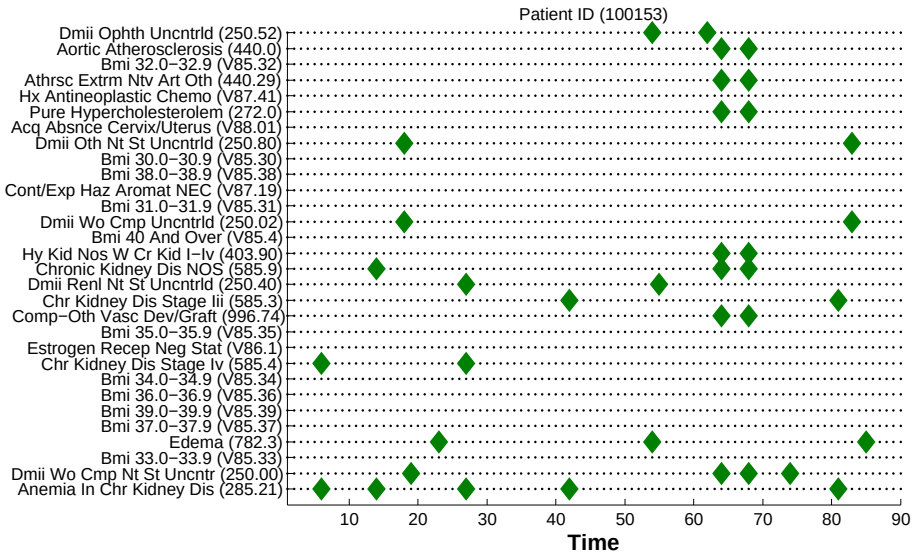


Figure 1.2: An example of the patient's EMR. The horizontal axis represents the number of days since the patient has records. The vertical axis corresponds to different diagnosis codes. A green diamond indicates the corresponding code is diagnosed for this patient at the corresponding day.

feature vectors from them with some summary statistics, then feed those feature vectors into computational models (e.g., classification, regression and clustering) for specific tasks. However, this may not be appropriate because many of those zero entries are not actual zeros but missing (the patient did not pay a visit and thus there is no corresponding record). Thus, the feature vectors constructed in this way are not accurate. As a consequence, the performance of the computational models will be compromised.

To handle the sparsity problem, I propose a general framework, Pacifier (PAtient reCord densIFIER), for phenotyping patients with their EMRs, which imputes the values of those missing entries by exploring the latent structures on both feature and time dimensions. Specifically, I assume those observed medical features in EMR (micro-phenotypes) can be mapped to some latent medical concept space with a much lower dimensionality, such that each medical concept can be viewed as a combination of several observed medical features (macro-phenotypes). In this way, we expect to

discover a much denser representation of the patient EMR in the latent space, and the values of those medical concepts evolve smoothly over time. I develop the following two specific formulations to achieve such goal:

- *Individual Basis Approach* (IBA), which approximates each individual EMR matrix as the product of two latent matrices. One is the mapping from those observed medical features to the latent medical concepts, the other describes how the values of those medical concepts evolve over time.
- *Shared Basis Approach* (SBA), which also approximates the EMR matrix for each patient as the product of two latent matrices, but the mapping matrix from those observed medical features to the latent medical concepts is shared over the entire patient population. Treating the densification of each patient as a task, the SBA approach is a multi-task learning problem.

When formulating PACIFIER, I enforce sparsity on the latent medical concept mapping matrix to encourage representative and interpretable medical concepts. I also enforce temporal smoothness on the concept value evolution matrix that captures the continuous nature of the patients. I develop an efficient *Block Coordinate Descent* (BCD) scheme for both formulations, that has the capability of processing large-scale datasets. I validate the effectiveness of our method in two real world case studies on predicting the onset risk of Congestive Heart Failure (CHF) patients and End State Renal Disease (ESRD) patients. Our results show that the average prediction AUC in both tasks can be improved significantly (from 0.689 to 0.816 on CHF prediction, and from 0.756 to 0.838 on ESRD respectively) with PACIFIER.

Chapter 2

CLUSTERED MULTI-TASK LEARNING

In the multi-task learning there is an important class of approaches, which assume that multiple predictors for different tasks share a common structure on the underlying predictor space. Alternating structure optimization (ASO) is one of the representative approach in this class, which is for linear predictors. ASO simultaneously performs inference on multiple tasks and discovers the shared low-dimensional predictive structure. In the high- dimensional setting, however, the computational cost of ASO is typically very high.

In this chapter I present a multi-task learning formulation called clustered multi-task learning (CMTL), which assumes models of the tasks form some types of clusters, and models within the same cluster is more similar to each other than those in different clusters. I establish the equivalence relationship between the CMTL and ASO, which means when the data is high-dimensional we can perform CMTL as an efficient alternative to ASO.

2.1 Alternating Structure Optimization and Clustered Multi-Task Learning

Assume we are given a multi-task learning problem with m tasks; each task $i \in \mathbb{N}_m$ is associated with a set of training data $\{(x_1^i, y_1^i), \ldots, (x_{n_i}^i, y_{n_i}^i)\} \subset \mathbb{R}^d \times \mathbb{R}$, and a linear predictive function f_i: $f_i(x_j^i) = w_i^T x_j^i$, where w_i is the weight vector of the i-th task, d is the data dimensionality, and n_i is the number of samples of the i-th task. We denote $W = [w_1, \ldots, w_m] \in \mathbb{R}^{d \times m}$ as the weight matrix to be estimated. Given a loss

function $\ell(\cdot,\cdot)$, the empirical risk is given by:

$$\mathcal{L}(W) = \sum_{i=1}^{m} \frac{1}{n_i} \left(\sum_{j=1}^{n_i} \ell(w_i^T x_j^i, y_j^i) \right).$$

We study the following multi-task learning formulation: $\min_W \mathcal{L}(W) + \Omega(W)$, where Ω encodes our prior knowledge about the m tasks. Next, we review ASO and CMTL and explore their inherent relationship.

Alternating structure optimization. In ASO Ando and Zhang (2005), all tasks are assumed to share a common feature space $\Theta \in \mathbb{R}^{h\times d}$, where $h \leq \min(m,d)$ is the dimensionality of the shared feature space and Θ has orthonormal columns, i.e., $\Theta\Theta^T = I_h$. The predictive function of ASO is: $f_i(x_j^i) = w_i^T x_j^i = u_i^T x_j^i + v_i^T \Theta x_j^i$, where the weight $w_i = u_i + \Theta^T v_i$ consists of two components including the weight u_i for the high-dimensional feature space and the weight v_i for the low-dimensional space based on Θ. ASO minimizes the following objective function: $\mathcal{L}(W) + \alpha \sum_{i=1}^m \|u_i\|_2^2$, subject to: $\Theta\Theta^T = I_h$, where α is the regularization parameter for task relatedness. We can further improve the formulation by including a penalty, $\beta \sum_{i=1}^m \|w_i\|_2^2$, to improve the generalization performance as in traditional supervised learning. Since $u_i = w_i - \Theta^T v_i$, we obtain the following ASO formulation:

$$\min_{W,\{v_i\},\Theta:\Theta\Theta^T=I_h} \mathcal{L}(W) + \sum_{i=1}^{m} \left(\alpha \|w_i - \Theta^T v_i\|_2^2 + \beta \|w_i\|_2^2 \right). \tag{2.1}$$

Clustered multi-task learning. In CMTL, we assume that the tasks are clustered into $k < m$ clusters, and the index set of the j-th cluster is defined as $\mathcal{I}_j = \{v | v \in \text{cluster } j\}$. We denote the mean of the jth cluster to be $\bar{w}_j = \frac{1}{n_j} \sum_{v\in\mathcal{I}_j} w_v$. For a given $W = [w_1, \cdots, w_m]$, the sum-of-square error (SSE) function in K-means clustering is given by Ding and He (2004); Zha *et al.* (2002):

$$\sum_{j=1}^{k} \sum_{v\in\mathcal{I}_j} \|w_v - \bar{w}_j\|_2^2 = \text{tr}\left(W^T W\right) - \text{tr}\left(F^T W^T W F\right), \tag{2.2}$$

where the matrix $F \in \mathbb{R}^{m \times k}$ is an orthogonal cluster indicator matrix with $F_{i,j} = \frac{1}{\sqrt{n_j}}$ if $i \in \mathcal{I}_j$ and $F_{i,j} = 0$ otherwise. If we ignore the special structure of F and keep the orthogonality requirement only, the relaxed SSE minimization problem is:

$$\min_{F:F^TF=I_k} \operatorname{tr}\left(W^TW\right) - \operatorname{tr}\left(F^TW^TWF\right), \tag{2.3}$$

resulting in the following penalty function for CMTL:

$$\Omega_{\text{CMTL}_0}(W,F) = \alpha\left(\operatorname{tr}\left(W^TW\right) - \operatorname{tr}\left(F^TW^TWF\right)\right) + \beta \operatorname{tr}\left(W^TW\right), \tag{2.4}$$

where the first term is derived from the K-means clustering objective and the second term is to improve the generalization performance. Combing Eq. (2.4) with the empirical error term $\mathcal{L}(W)$, we obtain the following CMTL formulation:

$$\min_{W,F:F^TF=I_k} \mathcal{L}(W) + \Omega_{\text{CMTL}_0}(W,F). \tag{2.5}$$

Equivalence of ASO and CMTL. In the ASO formulation in Eq. (2.1), it is clear that the optimal v_i is given by $v_i^* = \Theta w_i$. Thus, the penalty in ASO has the following equivalent form:

$$\begin{aligned}\Omega_{\text{ASO}}(W,\Theta) &= \sum_{i=1}^{m}\left(\alpha\|w_i - \Theta^T\Theta w_i\|_2^2 + \beta\|w_i\|_2^2\right) \\ &= \alpha\left(\operatorname{tr}\left(W^TW\right) - \operatorname{tr}\left(W^T\Theta^T\Theta W\right)\right) + \beta\operatorname{tr}\left(W^TW\right),\end{aligned} \tag{2.6}$$

resulting in the following equivalent ASO formulation:

$$\min_{W,\Theta:\Theta\Theta^T=I_h} \mathcal{L}(W) + \Omega_{\text{ASO}}(W,\Theta). \tag{2.7}$$

The penalty of the ASO formulation in Eq. (2.7) looks very similar to the penalty of the CMTL formulation in Eq. (2.5), however the operations involved are fundamentally different. In the CMTL formulation in Eq. (2.5), the matrix F is operated on the task dimension, as it is derived from the K-means clustering on the tasks;

while in the ASO formulation in Eq. (2.7), the matrix Θ is operated on the feature dimension, as it aims to identify a shared low-dimensional predictive structure for all tasks. Although different in the mathematical formulation, we show in the following theorem that the objectives of CMTL and ASO are equivalent.

Theorem 2.1.1. *The objectives of CMTL in Eq. (2.5) and ASO in Eq. (2.7) are equivalent if the cluster number, k, in K-means equals to the size, h, of the shared low-dimensional feature space.*

Proof. Denote $\mathcal{Q}(W) = \mathcal{L}(W) + (\alpha+\beta)\operatorname{tr}\left(W^TW\right)$, with $\alpha, \beta > 0$. Then, CMTL and ASO solve the following optimization problems:

$$\min_{W,F:F^TF=I_p} \mathcal{Q}(W) - \alpha \operatorname{tr}\left(WFF^TW^T\right), \quad \min_{W,\Theta:\Theta\Theta^T=I_p} \mathcal{Q}(W) - \alpha \operatorname{tr}\left(W^T\Theta^T\Theta W\right),$$

respectively. Note that in both CMTL and ASO, the first term $\mathcal{Q}$ is independent of F or Θ, for a given W. Thus, the optimal F and Θ for these two optimization problems are given by solving:

$$\text{[CMTL]} \quad \max_{F:F^TF=I_k} \operatorname{tr}\left(WFF^TW^T\right), \qquad \text{[ASO]} \quad \max_{\Theta:\Theta\Theta^T=I_k} \operatorname{tr}\left(W^T\Theta^T\Theta W\right).$$

Since WW^T and W^TW share the same set of nonzero eigenvalues, by the Ky-Fan Theorem Fan (1949), both problems above achieve exactly the same maximum objective value: $\|W^TW\|_{(k)} = \sum_{i=1}^{k} \lambda_i(W^TW)$, where $\lambda_i(W^TW)$ denotes the i-th largest eigenvalue of W^TW and $\|W^TW\|_{(k)}$ is known as the Ky Fan k-norm of matrix W^TW. Plugging the results back to the original objective, the optimization problem for both CMTL and ASO becomes $\min_W \mathcal{Q}(W) - \alpha\|W^TW\|_{(k)}$. This completes the proof of this theorem. □

2.2 Convex Relaxation of CMTL and Its Equivalence to cASO

Convex Relaxation of CMTL. The formulation in Eq. (2.5) is non-convex. A natural approach is to perform a convex relaxation on CMTL. We first reformulate

the penalty in Eq. (2.5) as follows:

$$\Omega_{\text{CMTL}_0}(W, F) = \alpha \operatorname{tr}\left(W((1+\eta)I - FF^T)W^T\right), \tag{2.8}$$

where η is defined as $\eta = \beta/\alpha > 0$. Since $F^T F = I_k$, the following holds:

$$(1+\eta)I - FF^T = \eta(1+\eta)(\eta I + FF^T)^{-1}.$$

Thus, we can reformulate Ω_{CMTL_0} in Eq. (2.8) as the following equivalent form:

$$\Omega_{\text{CMTL}_1}(W, F) = \alpha\eta(1+\eta) \operatorname{tr}\left(W(\eta I + FF^T)^{-1}W^T\right). \tag{2.9}$$

resulting in the following equivalent CMTL formulation:

$$\min_{W,F:F^TF=I_k} \mathcal{L}(W) + \Omega_{\text{CMTL}_1}(W, F). \tag{2.10}$$

Following Chen *et al.* (2009); Jacob *et al.* (2008), we obtain the following convex relaxation of Eq. (2.10), called cCMTL:

$$\min_{W,M} \mathcal{L}(W) + \Omega_{\text{cCMTL}}(W, M) \quad \text{s.t. } \operatorname{tr}(M) = k, M \preceq I, \ M \in \mathbb{S}_+^m. \tag{2.11}$$

where $\Omega_{\text{cCMTL}}(W, M)$ is defined as:

$$\Omega_{\text{cCMTL}}(W, M) = \alpha\eta(1+\eta) \operatorname{tr}\left(W(\eta I + M)^{-1}W^T\right). \tag{2.12}$$

The optimization problem in Eq. (2.11) is jointly convex with respect to W and M Argyriou *et al.* (2008d).

Convex Relaxation of ASO. A convex relaxation (cASO) of the ASO formulation in Eq. (2.7) has been proposed in Chen *et al.* (2009):

$$\min_{W,S} \mathcal{L}(W) + \Omega_{\text{cASO}}(W, S) \ \text{ s.t. } \operatorname{tr}(S) = h, S \preceq I, \ S \in \mathbb{S}_+^d, \tag{2.13}$$

where Ω_{cASO} is defined as:

$$\Omega_{\text{cASO}}(W, S) = \alpha\eta(1+\eta) \operatorname{tr}\left(W^T(\eta I + S)^{-1}W\right). \tag{2.14}$$

The cASO formulation in Eq. (2.13) and the cCMTL formulation in Eq. (2.11) are different in the regularization components: the respective Hessian of the regularization with respect to W are different.

Equivalence of cCMTL and cASO. Similar to Theorem 2.1.1, our analysis shows that cASO and cCMTL are equivalent.

Theorem 2.2.1. *The objectives of the cCMTL formulation in Eq. (2.11) and the cASO formulation in Eq. (2.13) are equivalent if the cluster number, k, in K-means equals to the size, h, of the shared low-dimensional feature space.*

Proof. Define the following two convex functions of W:

$$g_{\text{cCMTL}}(W) = \min_{M} \operatorname{tr}\left(W(\eta I + M)^{-1}W^T\right), \text{ s.t. } \operatorname{tr}(M) = k, M \preceq I, \ M \in \mathbb{S}_+^m, \tag{2.15}$$

and

$$g_{\text{cASO}}(W) = \min_{S} \operatorname{tr}\left(W^T(\eta I + S)^{-1}W\right), \text{ s.t. } \operatorname{tr}(S) = h, S \preceq I, \ S \in \mathbb{S}_+^d. \tag{2.16}$$

The cCMTL and cASO formulations can be expressed as unconstrained optimization w.r.t. W:

$$\text{[cCMTL]} \quad \min_{W} \mathcal{L}(W) + c \cdot g_{\text{CMTL}}(W), \qquad \text{[cASO]} \quad \min_{W} \mathcal{L}(W) + c \cdot g_{\text{ASO}}(W),$$

where $c = \alpha\eta(1+\eta)$. Let $h = k \leq \min(d, m)$. Next, we show that for a given W, $g_{\text{CMTL}}(W) = g_{\text{ASO}}(W)$ holds.

Let $W = Q_1 \Sigma Q_2$, $M = P_1 \Lambda_1 P_1^T$, and $S = P_2 \Lambda_2 P_2^T$, be the SVD of W, M, and S (M and S are symmetric positive semi-definite), respectively, where $\Sigma = \operatorname{diag}\{\sigma_1, \sigma_2, \ldots, \sigma_m\}$, $\Lambda_1 = \operatorname{diag}\{\lambda_1^{(1)}, \lambda_2^{(1)}, \ldots, \lambda_m^{(1)}\}$, and $\Lambda_2 = \{\lambda_1^{(2)}, \lambda_2^{(2)}, \ldots, \lambda_m^{(2)}\}$. Let $q < k$ be the rank of Σ. It follows from the basic properties of the trace that:

$$\operatorname{tr}\left(W(\eta I + M)^{-1}W^T\right) = \operatorname{tr}\left((\eta I + \Lambda_1)^{-1} P_1^T Q_2 \Sigma^2 Q_2^T P_1\right).$$

The problem in Eq. (2.15) is thus equivalent to:

$$\min_{P_1,\Lambda_1} \operatorname{tr}\left((\eta I + \Lambda_1)^{-1} P_1^T Q_2 \Sigma^2 Q_2^T P_1\right), \quad \text{s.t.} \quad P_1 P_1^T = I, P_1^T P_1 = I, \sum_{i=1}^{d} \lambda_i^{(1)} = k. \quad (2.17)$$

It can be shown that the optimal P_1^* is given by $P_1^* = Q_2$ and the optimal Λ_1^* is given by solving the following simple (convex) optimization problem Chen *et al.* (2009):

$$\Lambda_1^* = \operatorname*{argmin}_{\Lambda_1} \sum_{i=1}^{q} \frac{\sigma_i^2}{\eta + \lambda_i^{(1)}}, \quad \text{s.t.} \quad \sum_{i}^{q} \lambda_i^{(1)} = k, 0 \leq \lambda_i^{(1)} \leq 1. \quad (2.18)$$

It follows that $g_{\text{cCMTL}}(W) = \operatorname{tr}\left((\eta I + \Lambda_1^*)^{-1}\Sigma^2\right)$. Similarly, we can show that $g_{\text{cASO}}(W) = \operatorname{tr}\left((\eta I + \Lambda_2^*)^{-1}\Sigma^2\right)$, where

$$\Lambda_2^* = \operatorname*{argmin}_{\Lambda_2} \sum_{i=1}^{q} \frac{\sigma_i^2}{\eta + \lambda_i^{(2)}}, \quad \text{s.t.} \quad \sum_{i}^{q} \lambda_i^{(2)} = h, \ 0 \leq \lambda_i^{(2)} \leq 1.$$

It is clear that when $h = k$, $\Lambda_1^* = \Lambda_2^*$ holds. Therefore, we have $g_{\text{cCMTL}}(W) = g_{\text{cASO}}(W)$. This completes the proof. □

Remark 2.2.2. In the functional of cASO in Eq. (2.16) the variable to be optimized is $S \in \mathbb{S}_+^d$, while in the functional of cCMTL in Eq. (2.15) the optimization variable is $M \in \mathbb{S}_+^m$. In many practical MTL problems the data dimensionality d is much larger than the task number m, and in such cases cCMTL is significantly more efficient in terms of both time and space. Our equivalence relationship established in Theorem 2.2.1 provides an (equivalent) efficient implementation of cASO especially for high-dimensional problems.

2.3 Experiment

In this section, we empirically evaluate the effectiveness and the efficiency of the proposed algorithms on synthetic and real-world data sets. The normalized mean square error (nMSE) and the averaged mean square error (aMSE) are used as the performance measure Argyriou *et al.* (2008a). Note that in this proposal we have

not developed new MTL formulations; instead our main focus is on the theoretical understanding of the inherent relationship between ASO and CMTL. Thus, an extensive comparative study of various MTL algorithms is out of the scope of this proposal. As an illustration, in the following experiments we only compare cCMTL with two baseline techniques: ridge regression STL (RidgeSTL) and regularized MTL (RegMTL) Evgeniou and Pontil (2004).

Simulation Study We apply the proposed cCMTL formulation in Eq. (2.11) on a synthetic data set (with a predefined cluster structure). We use 5-fold cross-validation to determine the regularization parameters for all methods. We construct the synthetic data set following a procedure similar to the one in Jacob *et al.* (2008): the constructed synthetic data set consists of 5 clusters, where each cluster includes 20 (regression) tasks and each task is represented by a weight vector of length $d = 300$. Details of the construction is provided in the supplemental material. We apply RidgeSTL, RegMTL, and cCMTL on the constructed synthetic data. The correlation coefficient matrices of the obtained weight vectors are presented in Figure 2.1. From the result we can observe (1) cCMTL is able to capture the cluster structure among tasks and achieves a small test error; (2) RegMTL is better than RidgeSTL in terms of test error. It however introduces unnecessary correlation among tasks possibly due to the assumption that all tasks are related; (3) In cCMTL we also notice some 'noisy' correlation, which may because of the spectral relaxation.

Effectiveness Comparison Next, we empirically evaluate the effectiveness of the cCMTL formulation in comparison with RidgeSTL and RegMTL using real world benchmark datasets including the School data[1] and the Sarcos data[2]. The regularization parameters for all algorithms are determined via 5-fold cross validation; the

[1] http://www.cs.ucl.ac.uk/staff/A.Argyriou/code/

[2] http://gaussianprocess.org/gpml/data/

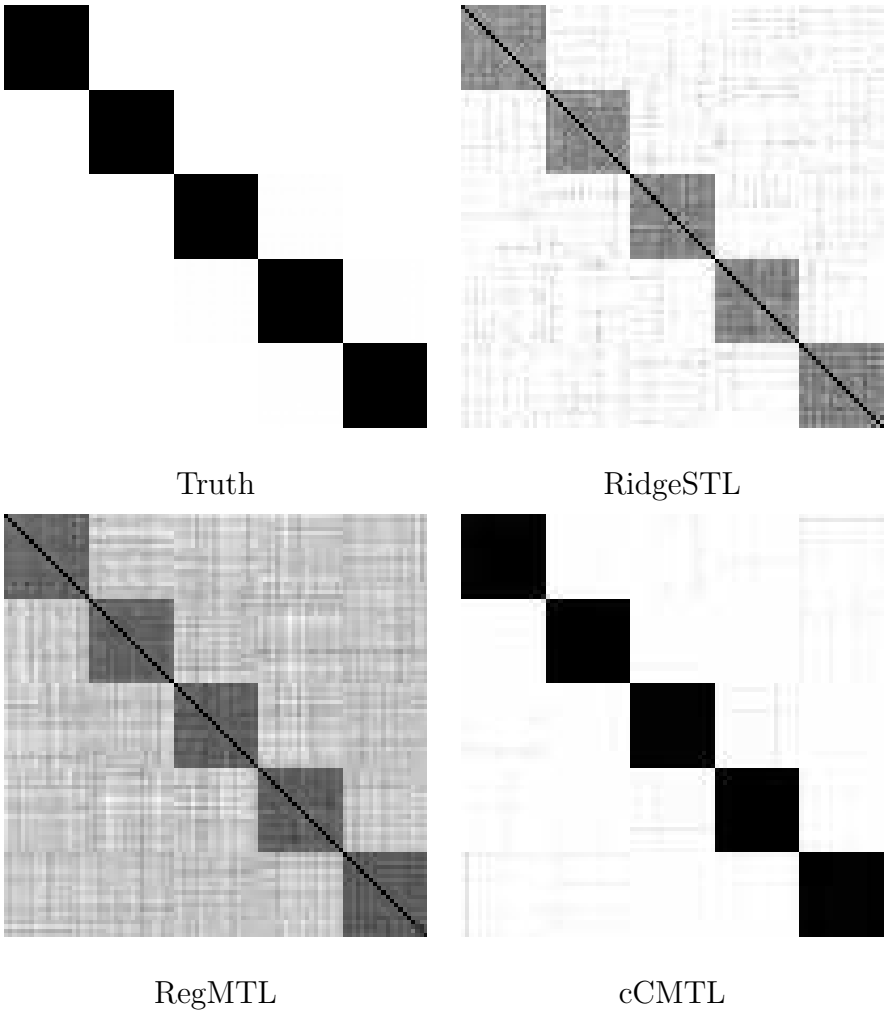


Figure 2.1: The correlation matrices of the ground truth model, and the models learned from RidgeSTL, RegMTL, and cCMTL. Darker color indicates higher correlation. In the ground truth there are 100 tasks clustered into 5 groups. Each task has 200 dimensions. 95 training samples and 5 testing samples are used in each task. The test errors (in terms of nMSE) for RidgeSTL, RegMTL, and cCMTL are 0.8077, 0.6830, 0.0354, respectively.

Table 2.1: Performance comparison on the School data in terms of nMSE and aMSE. Smaller nMSE and aMSE indicate better performance. All regularization parameters are tuned using 5-fold cross validation. The mean and standard deviation are calculated based on 10 random repetitions.

Measure	Ratio	RidgeSTL	RegMTL	cCMTL
nMSE	10%	1.3954 ± 0.0596	1.0988 ± 0.0178	1.0850 ± 0.0206
	15%	1.1370 ± 0.0146	1.0636 ± 0.0170	0.9708 ± 0.0145
	20%	1.0290 ± 0.0309	1.0349 ± 0.0091	0.8864 ± 0.0094
	25%	0.8649 ± 0.0123	1.0139 ± 0.0057	0.8243 ± 0.0031
	30%	0.8367 ± 0.0102	1.0042 ± 0.0066	0.8006 ± 0.0081
aMSE	10%	0.3664 ± 0.0160	0.2865 ± 0.0054	0.2831 ± 0.0050
	15%	0.2972 ± 0.0034	0.2771 ± 0.0045	0.2525 ± 0.0048
	20%	0.2717 ± 0.0083	0.2709 ± 0.0027	0.2322 ± 0.0022
	25%	0.2261 ± 0.0033	0.2650 ± 0.0027	0.2154 ± 0.0020
	30%	0.2196 ± 0.0035	0.2632 ± 0.0028	0.2101 ± 0.0016

reported experimental results are averaged over 10 random repetitions. The School data consists of the exam scores of 15362 students from 139 secondary schools, where each student is described by 27 attributes. We vary the training ratio in the set $5 \times \{1, 2, \cdots, 6\}\%$ and record the respective performance. The experimental results are presented in Table 2.1. We can observe that cCMTL performs the best among all settings. Experimental results on the Sarcos dataset is available in the supplemental material.

Efficiency Comparison We compare the efficiency of the three algorithms including altCMTL, apgCMTLand graCMTL for solving the cCMTL formulation in Eq. (2.11). For the following experiments, we set $\alpha = 1$, $\beta = 1$, and $k = 2$ in cCMTL. We observe a similar trend in other settings. Specifically, we study how the feature dimensionality,

the sample size, and the task number affect the required computation cost (in seconds) for convergence. The experimental setup is as follows: we terminate apgCMTL when the change of objective values in two successive steps is smaller than 10^{-5} and record the obtained objective value; we then use such a value as the stopping criterion in graCMTL and altCMTL, that is, we stop graCMTL or altCMTL when graCMTL or altCMTL attains an objective value equal to or smaller than the one attained by apgCMTL. We use Yahoo Arts data for the first two experiments. Because in Yahoo data the task number is very small, we construct a synthetic data for the third experiment.

In the first experiment, we vary the feature dimensionality in the set $[500 : 500 : 2500]$ with the sample size fixed at 4000 and the task numbers fixed at 17. The experimental result is presented in the left plot of Figure 2.2. In the second experiment, we vary the sample size in the set $[3000 : 1000 : 9000]$ with the dimensionality fixed at 500 and the task number fixed at 17. The experimental result is presented in the middle plot of Figure 2.2. From the first two experiments, we observe that larger feature dimensionality or larger sample size will lead to higher computation cost. In the third experiment, we vary the task number in the set $[10 : 10 : 190]$ with the feature dimensionality fixed at 600 and the sample size fixed at 2000. The employed synthetic data set is constructed as follows: for each task, we generate the entries of the data matrix X_i from $\mathcal{N}(0, 1)$, and generate the entries of the weight vector from $\mathcal{N}(0, 1)$, the response vector y_i is computed as $y_i = X_i w_i + \xi$, where $\xi \sim \mathcal{N}(0, 0.01)$ represents the noise vector. The experimental result is presented in the right plot of Figure 2.2. We can observe that altCMTL is more efficient than the other two algorithms.

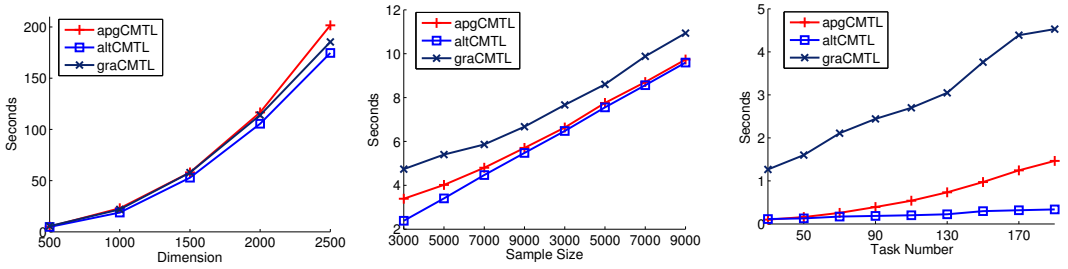


Figure 2.2: Sensitivity study of altCMTL, apgCMTL, graCMTL in terms of the computation cost (in seconds) with respect to feature dimensionality (left), sample size (middle), and task number (right).

Chapter 3

MODELING DISEASE PROGRESSION VIA MULTI-TASK LEARNING

In the longitudinal AD study, we measure the cognitive scores of selected patients repeatedly at multiple time points. By considering the prediction of cognitive scores at a single time point as a regression task, tasks at different time points are temporally related to each other. In this chapter, I formulate the prediction of clinical scores at multiple future time points as a multi-task regression problem. We employ multi-task regression formulations instead of solving a set of independent regression problems since the intrinsic temporal smoothness information among different tasks can be incorporated into the model as prior knowledge.

3.1 Modeling Disease Progression via Temporal Group Lasso

Consider a multi-task regression problem of t time points with n training samples of d features. Let $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$ be the input data at the baseline, and $\{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n\}$ be the targets, where each $\mathbf{x}_i \in \mathbb{R}^d$ represents a sample (patient), and $\mathbf{y}_i \in \mathbb{R}^t$ is the corresponding targets (clinical scores) at different time points. In this proposal we employ linear models for the prediction. Specifically, the prediction model for the ith time point is given by $f^i(\mathbf{x}) = \mathbf{x}^T\mathbf{w}^i$, where $\mathbf{w}^i$ is the weight vector of the model. Let $X = [\mathbf{x}_1, \cdots, \mathbf{x}_n]^T \in \mathbb{R}^{n\times d}$ be the data matrix, $Y = [\mathbf{y}_1, \cdots, \mathbf{y}_n]^T \in \mathbb{R}^{n\times t}$ be the target matrix, and $W = [\mathbf{w}^1, \mathbf{w}^2, ..., \mathbf{w}^t] \in \mathbb{R}^{d\times t}$ be the weight matrix. One simple approach is to estimate W by minimizing the following objective function:

$$\min_W \|XW - Y\|_F^2 + \theta_1 \|W\|_F^2,$$

where the first term measures the empirical error on the training data, $\theta_1 > 0$ is a regularization parameter, and $\|W\|_F$ is the Frobenius norm, defined as $\sqrt{\sum_{i=1}^d \sum_{j=1}^t W_{i,j}^2}$.

The formulation is illustrated in Figure 3.1. The regression method above is known as the *ridge regression* and it admits an analytical solution given by:

$$W = (X^T X + \theta_1 I)^{-1} X^T Y.$$

In building models with high dimensional features ($d \gg n$), feature selection methods are typically employed to identify a small set of relevant features. Lasso Tibshirani (1996), is a popular method for sparse linear regression, which simultaneously performs feature selection and regression. In the context of disease progression, the Lasso formulation solves the following optimization problem:

$$\min_W \|XW - Y\|_F^2 + \theta_1 \|W\|_1 ,$$

where $\|W\|_1$ is the ℓ_1 norm of W defined as $\sum_{i=1}^{d} \sum_{j=1}^{t} |W_{i,j}|$.

One major limitation of the regression models above is that the tasks at different time points are assumed to be independent with each other, which is not the case in the longitudinal AD study considered in this proposal.

3.1.1 Temporal Smoothness Prior

Applying single task learning methods such as ridge or Lasso regression on modeling disease progression often yields fluctuated prediction values at different time point for one patient, as shown in Figure 3.2. In the course of disease progression, it is reasonable to assume that the difference of the cognitive scores between two successive time points is relatively small. During the inference of our models, for a patient i with two consecutive predictions $\hat{y}_i^{(j)}$ and $\hat{y}_i^{(j+1)}$ at time point j and $j+1$ respectively, a large difference between the predictions $|\hat{y}_i^{(j)} - \hat{y}_i^{(j+1)}|$ is discouraged. Since we use linear models ($y_i^{(j)} \approx \hat{y}_i^{(j)} = \mathbf{x}_i^T \mathbf{w}^j$), the difference between the predictions can be

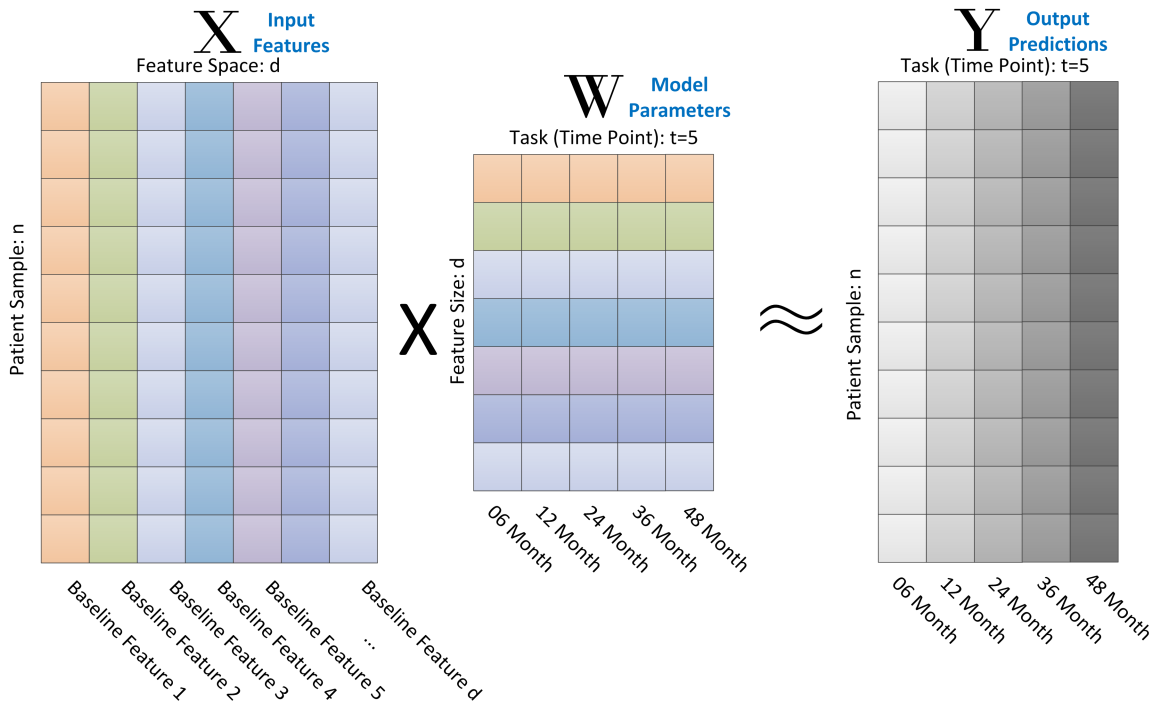


Figure 3.1: Illustration of disease prediction modeling. We denote $X = [\mathbf{x}_1, \cdots, \mathbf{x}_n]^T \in \mathbb{R}^{n \times d}$ as the data matrix, $Y = [\mathbf{y}_1, \cdots, \mathbf{y}_n]^T \in \mathbb{R}^{n \times t}$ as the target matrix, and $W = [\mathbf{w}^1, \mathbf{w}^2, ..., \mathbf{w}^t] \in \mathbb{R}^{d \times t}$ as the weight matrix. Specifically, for the input matrix X, each row represents a patient and each column represents a feature at baseline, and for the output matrix Y, each row corresponds to a patient, and each column corresponds to the score at a future time point. In the prediction model we assume a linear relationship between input X and output Y, i.e., for the i-th patient, we have $\mathbf{x}_i^T W \approx \mathbf{y}_i^T$.

related to the difference between models at those time points:

$$|\hat{y}_i^{(j)} - \hat{y}_i^{(j+1)}| = |\mathbf{x}_i^T \mathbf{w}^j - \mathbf{x}_i^T \mathbf{w}^{j+1}| = |\mathbf{x}_i^T (\mathbf{w}^j - \mathbf{w}^{j+1})|. \tag{3.1}$$

Inspired by Eq. (3.1), in order to capture the temporal smoothness of the cognitive scores at different time points, we introduce a regularization term in the regression model that penalizes large deviations between predictions at neighboring time points, resulting in the following formulation:

$$\min_W \|XW - Y\|_F^2 + \theta_1 \|W\|_F^2 + \theta_2 \sum_{i=1}^{t-1} \left\|\mathbf{w}^i - \mathbf{w}^{i+1}\right\|_2^2, \tag{3.2}$$

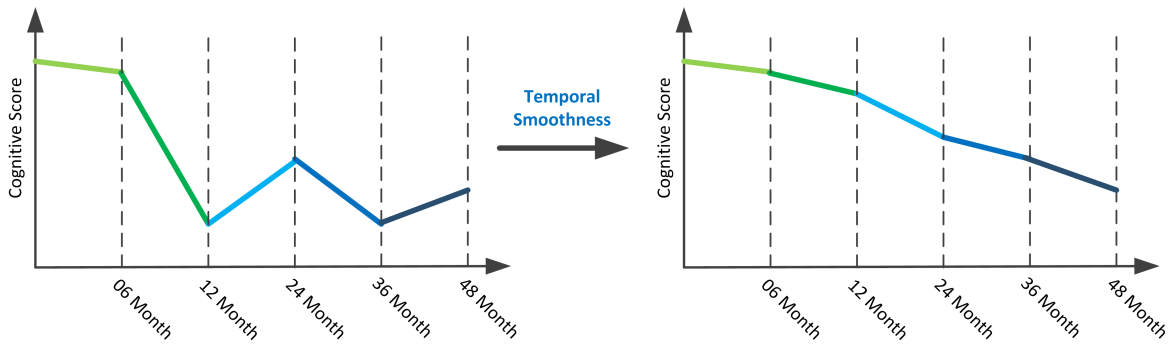


Figure 3.2: Illustration of temporal smoothness. We assume that the difference of the cognitive scores between two successive time points is relatively small (right figure). Since we use linear predictive models, the difference between the predicted cognitive scores can be related to the difference between models at those time points, and therefore the temporal smoothness can be enforced by penalizing the difference between models of consecutive time points. In single task learning formulations, such as Ridge and Lasso, the predicted scores of the same patient at different time points may fluctuate as shown in the left figure.

where $\theta_2 \geq 0$ is a regularization parameter controlling the temporal smoothness. This temporal smoothness term can be expressed as:

$$\sum_{i=1}^{t-1} \left\| \mathbf{w}^i - \mathbf{w}^{i+1} \right\|_F^2 = \|WH\|_F^2 ,$$

where $H \in \mathbb{R}^{t \times (t-1)}$ is defined as follows: $H_{ij} = 1$ if $i = j$, $H_{ij} = -1$ if $i = j + 1$, and $H_{ij} = 0$ otherwise. The formulation in Eq.(3.2) becomes:

$$\min_W \|XW - Y\|_F^2 + \theta_1 \|W\|_F^2 + \theta_2 \|WH\|_F^2 . \tag{3.3}$$

We now show that the optimization problem in Eq.(3.3) admits an analytical solution. Denote $\mathcal{P}_r(.)$ as the row selection operator parameterized by a selection vector r. The resulting matrix of $\mathcal{P}_r(A)$ includes only A_i such that $r_i \neq 0$, where A_i is the ith row of A. Let S^i be the ith column of S. We therefore denote $X_{(i)} = \mathcal{P}_{S^i}(X) \in \mathbb{R}^{n_i \times d}$ as the input data matrix of the ith task, and $y_{(i)} = \mathcal{P}_{S^i}(Y^i) \in \mathbb{R}^{n_i \times 1}$ as the corresponding target vector, where n_i is number of samples from the ith task.

First, we take the derivative of Eq. (3.3) with respect to W and set it to zero:

$$X^T XW - X^T Y + \theta_1 W + \theta_2 WHH^T = 0, \tag{3.4}$$

$$\left(X^T X + \theta_1 I_d\right) W + W\left(\theta_2 HH^T\right) = X^T Y, \tag{3.5}$$

where I_d is the identity matrix of size d by d. Since both matrices $(X^T X + \theta_1 I_d)$ and $\theta_2 HH^T$ are symmetric, we write the eigen-decomposition of these two matrices by $Q_1 \Lambda_1 Q_1^T$ and $Q_2 \Lambda_2 Q_2^T$, where $\Lambda_1 = diag(\lambda_1^{(1)}, \lambda_1^{(2)}, \ldots, \lambda_1^{(d)})$ and $\Lambda_2 = diag(\lambda_2^{(1)}, \lambda_2^{(2)}, \ldots, \lambda_2^{(d)})$, are their eigenvalues, and Q_1 and Q_2 are orthogonal. Plugging them into Eq. (3.5) we get:

$$Q_1 \Lambda_1 Q_1^T W + W Q_2 \Lambda_2 Q_2^T = X^T Y, \tag{3.6}$$

$$\Lambda_1 Q_1^T W Q_2 + Q_1^T W Q_2 \Lambda_2 = Q_1^T X^T Y Q_2. \tag{3.7}$$

Denote $\hat{W} = Q_1^T W Q_2$ and $D = Q_1^T X^T Y Q_2$. Eq. (3.7) becomes $\Lambda_1 \hat{W} + \hat{W} \Lambda_2 = D$. Thus $\hat{W}$ is given by:

$$\hat{W}_{i,j} = \frac{D_{i,j}}{\lambda_1^{(i)} + \lambda_2^{(j)}}. \tag{3.8}$$

The optimal weight matrix is then given by $W^* = Q_1 \hat{W} Q_2^T$.

We want to emphasize that the temporal smoothness is only employed during the inference of the model, and when it comes to the prediction phase only baseline features are needed to compute the predicted cognitive scores at the future time points. This is also the case for other models in the proposal.

3.1.2 Dealing with Incomplete Data

The clinical scores for many patients are missing at some time points, i.e., the target vector $y_i \in \mathbb{R}^t$ may not be complete. A simple strategy is to remove all patients with missing target values, which, however, significantly reduces the number

of samples. We consider to extend the formulation in Eq. (3.3) with missing target values in the training process. In this case, the analytical solution to Eq. (3.3) no longer exists. We show how the algorithm above can be adapted to deal with missing target values.

We use a matrix $S \in \mathbb{R}^{n \times t}$ to indicate missing target values, where $S_{i,j} = 0$ if the target value of sample i is missing at the jth time point, and $S_{i,j} = 1$ otherwise. We use the component-wise operator $\odot$ as follows: $Z = A \odot B$ denotes $z_{i,j} = a_{i,j} b_{i,j}$, for all i, j. The formulation in Eq. (3.3) can be extended to the case with missing target values as:

$$\min_{W} \|S \odot (XW - Y)\|_F^2 + \theta_1 \|W\|_F^2 + \theta_2 \|WH\|_F^2 . \tag{3.9}$$

The optimization problem in Eq. (3.9) can be solved efficiently as shown below.

Similar to the case without missing target values considered in Section 3.1.1, we take the derivative of Eq. (3.9) with respect to w^i $(2 \leq i \leq t-1)$ and set it to zero:

$$Aw^{i-1} + M_i w^i + Aw^{i+1} = T_i, \tag{3.10}$$

where A, M_i, and T_i are defined as follows:

$$\begin{aligned} A &= -\theta_2 I_d, \\ M_i &= X_{(i)}^T X_{(i)} + \theta_1 I_d + 2\theta_2 I_d, \\ T_i &= X_{(i)}^T y_{(i)}. \end{aligned}$$

For the special case $i = 1$, the term $\|w^{i-1} - w^i\|_2^2$ does not exist, nor is the term $\|w^i - w^{i+1}\|_2^2$ for $i = t$. We combine the equations for all tasks $(1 \leq i \leq t)$, which can

be represented as a block tridiagonal linear system:

$$\begin{pmatrix} M_1 & A & & & 0 \\ A & M_2 & A & & \\ & & \ddots & & \\ & & A & M_{t-1} & A \\ 0 & & & A & M_t \end{pmatrix} \begin{pmatrix} w^1 \\ w^2 \\ \vdots \\ w^{t-1} \\ w^t \end{pmatrix} = \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_{t-1} \\ T_t \end{pmatrix} \tag{3.11}$$

For a general linear system of size td, it can be solved using Gaussian elimination with a time complexity of $O((td)^3)$. For our block tridiagonal system, the complexity is reduced to $O(d^3t)$ using block Gaussian elimination. For large-scale linear systems, the LSQR algorithm Paige and Saunders (1982), a popular iterative method for the solution of large linear systems of equations, can be employed with a time complexity of $O(Ntd^2)$, where N, the number of iterations, is typically small.

3.1.3 Temporal Group Lasso Regularization

Because of the limited availability of subjects in the longitudinal AD study and a relatively large number of features (e.g., MRI features) at ADNI, the prediction model suffers from the so called "curse of dimensionality". In addition, many patients drop out from the longitudinal study after a certain period of time, which reduces the effective number of samples. One effective approach is to reduce the dimensionality of the data. However, traditional dimension reduction techniques such as PCA are not desirable since the resulting model is not interpretable, and traditional feature selection algorithms are not suitable for multi-task regression with missing target values. In the proposed formulation, we employ the group Lasso regularization based on the $\ell_{2,1}$-norm penalty for feature selection Yuan and Lin (2006), which assumes that a small set of features are predictive of the progression. The group Lasso regularization ensures that all regression models at different time points share a common set

of features. Together with the temporal smoothness penalty, we obtain the following Temporal Group Lasso (TGL) formulation:

$$\min_W \|S \odot (XW - Y)\|_F^2 + \theta_1 \|W\|_F^2 + \theta_2 \|WH\|_F^2 + \delta \|W\|_{2,1} \tag{3.12}$$

where $\|W\|_{2,1} = \sum_{i=1}^{d} \sqrt{\sum_{j=1}^{t} W_{i,j}^2}$, and δ is a regularization parameter. When there is only one task, i.e., $t = 1$, the above formulation reduces to Lasso Tibshirani (1996). When $t > 1$, the weights of one feature over all tasks are grouped using the ℓ_2-norm, and all features are further grouped using the ℓ_1-norm. Thus, the $\ell_{2,1}$-norm penalty tends to select features based on the strength of the feature over all t tasks.

The objective in Eq. (3.12) can be considered as a combination of a smooth term and a non-smooth term. The gradient descent or accelerated gradient method (AGM) Nesterov (2004); Nemirovski (2005) can be applied to solve the optimization. One of the key steps in AGM is the computation of the proximal operator associated with the $\ell_{2,1}$-norm regularization. We employ the algorithm in the SLEP package Liu *et al.* (2009c), which computes the proximal operator associated with the general ℓ_1/ℓ_q-norm efficiently.

3.2 Proposed Method II: Fused Sparse Group Lasso

The TGL formulation constrains the models from all time points to share a common set of features. In order to better capture the temporal patterns of the biomarkers in disease progression Jack Jr *et al.* (2010); Caroli *et al.* (2010), we further propose a convex fused sparse group Lasso (cFSGL) formulation which allows simultaneous joint feature selection for multiple tasks and task-specific feature selection, and in the meantime incorporates the temporal smoothness. Mathematically, the cFSGL formulation solves the following convex optimization problem:

$$\min_W L(W) + \lambda_1 \|W\|_1 + \lambda_2 \left\|RW^T\right\|_1 + \lambda_3 \|W\|_{2,1}, \tag{3.13}$$

where $\|W\|_1$ is the Lasso penalty, $\|W\|_{2,1} = \sum_{i=1}^{d} \sqrt{\sum_{j=1}^{t} W_{ij}^2}$ is the group Lasso penalty, $\left\|RW^T\right\|_1$ is the fused Lasso penalty, $R = H^T$ is a $(t-1) \times t$ sparse matrix, and λ_1, λ_2 and λ_3 are regularization parameters. The combination of Lasso and group Lasso penalties is also known as the sparse group Lasso penalty, which allows simultaneous joint feature selection for all tasks and selection of a specific set of features for each task. The fused Lasso penalty is employed to incorporate the temporal smoothness. The cFSGL formulation involves three non-smooth terms, and is thus challenging to solve. We propose to solve the optimization problem by the accelerated gradient method (AGM) Nesterov (2004); Nemirovski (2005). One of the key steps in using AGM is the computation of the proximal operator associated with the composite of non- smooth penalties defined as follows:

$$\pi(V) = \arg\min_{W} \frac{1}{2}\|W - V\|_2^2 + \lambda_1\|W\|_1 + \lambda_2\|RW^T\|_1 + \lambda_3\|W\|_{2,1}. \tag{3.14}$$

It is clear that each row of W is decoupled in Eq. (3.14). Thus, for obtaining the ith row $\mathbf{w}_i$, we only need to solve the following optimization problem:

$$\pi(\mathbf{v}_i) = \arg\min_{\mathbf{w}_i} \frac{1}{2}\|\mathbf{w}_i - \mathbf{v}_i\|_2^2 + \lambda_1\|\mathbf{w}_i\|_1 + \lambda_2\|R\mathbf{w}_i\|_1 + \lambda_3\|\mathbf{w}_i\|_2, \tag{3.15}$$

where $\mathbf{v}_i$ is the ith row of V. The proximal operator in Eq. (3.15) is challenging to solve due to the presence of three non-smooth terms. We show that the proximal operator exhibits a certain decomposition property, based on which we can efficiently compute the proximal operator in two stages, as summarized below.

Theorem 3.2.1. *Define*

$$\pi_{\mathrm{FL}}(\mathbf{v}) = \arg\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w} - \mathbf{v}\|_2^2 + \lambda_1\|\mathbf{w}\|_1 + \lambda_2\|R\mathbf{w}\|_1 \tag{3.16}$$

$$\pi_{\mathrm{GL}}(\mathbf{v}) = \arg\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w} - \mathbf{v}\|_2^2 + \lambda_3\|\mathbf{w}\|_2. \tag{3.17}$$

Then the following holds:

$$\pi(\mathbf{v}) = \pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v})). \tag{3.18}$$

Proof: The necessary and sufficient optimality conditions for (3.15), (3.16), and (3.17) can be written as:

$$\mathbf{0} \in \pi(\mathbf{v}) - \mathbf{v} + \lambda_1 \mathrm{SGN}(\pi(\mathbf{v})) + \lambda_2 R^T \mathrm{SGN}(R\pi(\mathbf{v})) + \lambda_3 \partial g(\pi(\mathbf{v})), \tag{3.19}$$

$$\mathbf{0} \in \pi_{\mathrm{FL}}(\mathbf{v}) - \mathbf{v} + \lambda_1 \mathrm{SGN}(\pi_{\mathrm{FL}}(\mathbf{v})) + \lambda_2 R^T \mathrm{SGN}(R\pi_{\mathrm{FL}}(\mathbf{v})), \tag{3.20}$$

$$\mathbf{0} \in \pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v})) - \pi_{\mathrm{FL}}(\mathbf{v}) + \lambda_3 \partial g(\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))), \tag{3.21}$$

where $\mathrm{SGN}(\mathbf{x})$ is a set defined in a component-wise manner as:

$$(\mathrm{SGN}(\mathbf{x}))_i = \begin{cases} [-1,1] & x_i = 0 \\ \{1\} & x_i > 0 \\ \{-1\} & x_i < 0, \end{cases} \tag{3.22}$$

and

$$\partial g(\mathbf{x}) = \begin{cases} \frac{\mathbf{x}}{\|\mathbf{x}\|_2} & \mathbf{x} \neq \mathbf{0} \\ \{\mathbf{y} : \|\mathbf{y}\|_2 \leq 1\} & \mathbf{x} = \mathbf{0}. \end{cases} \tag{3.23}$$

It follows from (3.21) and (3.23) that: 1) if $\|\pi_{\mathrm{FL}}(\mathbf{v})\|_2 \leq \lambda_3$, then $\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v})) = \mathbf{0}$; and 2) if $\|\pi_{\mathrm{FL}}(\mathbf{v})\|_2 > \lambda_3$, then

$$\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v})) = \frac{\|\pi_{\mathrm{FL}}(\mathbf{v})\|_2 - \lambda_3}{\|\pi_{\mathrm{FL}}(\mathbf{v})\|_2} \pi_{\mathrm{FL}}(\mathbf{v}).$$

It is easy to observe that, 1) if the i-th entry of $\pi_{\mathrm{FL}}(\mathbf{v})$ is zero, so is the i-th entry of $\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))$; 2) if the i-th entry of $\pi_{\mathrm{FL}}(\mathbf{v})$ is positive (or negative), so is he i-th entry of $\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))$. Therefore, we have

$$\mathrm{SGN}(\pi_{\mathrm{FL}}(\mathbf{v})) \subseteq \mathrm{SGN}(\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))). \tag{3.24}$$

Meanwhile, 1) if the i-th and the $i+1$-th entries of $\pi_{\mathrm{FL}}(\mathbf{v})$ are identical, so are those of $\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))$; 2) if the i-th entry is larger (or smaller) than the $i+1$-th entry in $\pi_{\mathrm{FL}}(\mathbf{v})$, so is in $\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))$. Therefore, we have

$$\mathrm{SGN}(R\pi_{\mathrm{FL}}(\mathbf{v})) \subseteq \mathrm{SGN}(R\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))). \tag{3.25}$$

It follows from (3.20), (3.21), (3.24), and (3.25) that

$$\begin{aligned}\mathbf{0} \in & \pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v})) - \mathbf{v} + \lambda_1 \mathrm{SGN}(\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))) \\ & + \lambda_2 R^T \mathrm{SGN}(R\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))) + \lambda_3 \partial g(\pi_{\mathrm{GL}}(\pi_{\mathrm{FL}}(\mathbf{v}))).\end{aligned} \tag{3.26}$$

Since (3.15) has a unique solution, we can get (3.18) from (3.19) and (3.26). □

Note that the fused Lasso signal approximator Friedman *et al.* (2007) in Eq. (3.17) can be effectively solved using Liu *et al.* (2010). The complete algorithm for solving the proximal operator associated with cFSGL is given in Algorithm 1.

Algorithm 1 Proximal operator associated with the Convex Fused Sparse Group Lasso (cFSGL)

Input: V, R, λ_1, λ_2, λ_3

Output: W

1: **for** i = 1 : t **do**

2: $\quad \mathbf{u}_i = \arg\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w} - \mathbf{v}_i\|_2^2 + \lambda_1\|\mathbf{w}\|_1 + \lambda_2\|R\mathbf{w}\|_1$

3: $\quad \mathbf{w}_i = \arg\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w} - \mathbf{u}_i\|_2^2 + \lambda_3\|\mathbf{w}\|_2$

4: **end for**

We illustrate the models built by different approaches in Figure 3.3. In the left figure we show the model built by Lasso regression. The sparsity introduced by applying Lasso has no specific patterns across tasks, as the models for different tasks are built independently. The middle figure shows the model built by TGL. Because of the use of $\ell_{2,1}$-norm regularization to capture temporal relation, the features selected for all time points are the same. The model built by cFSGL, as shown in the right figure, has two levels of sparsity: 1) a small set of features shared across all tasks, 2) task-specific features for each time point. In addition, one key advantage of fused Lasso in cFSGL is that under the fused Lasso penalty the selected features across different time points are similar to each other, satisfying the temporal smoothness property,

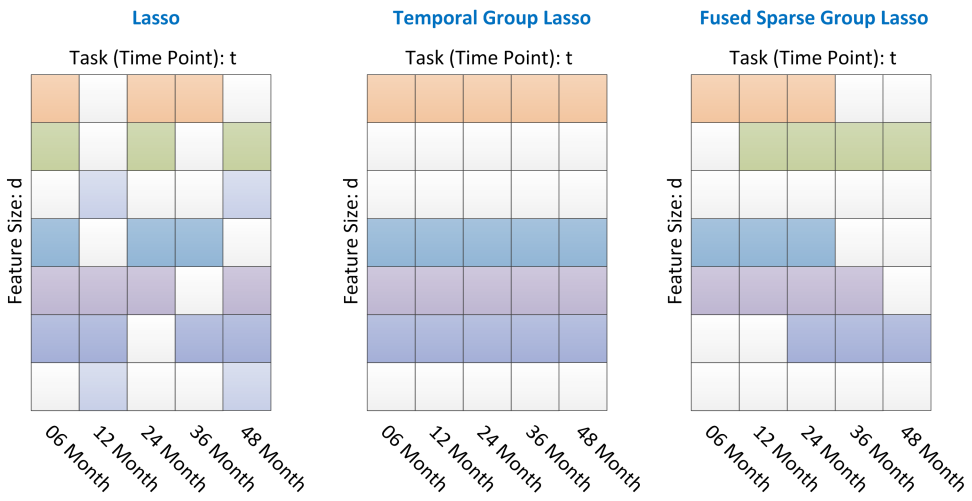


Figure 3.3: A comparison of models built by different approaches. In Lasso, the models for different tasks are built independently, thus no specific sparsity patterns are observed across different tasks (left figure). The TGL formulation restricts all models from different time points to select a common set of features (middle figure). In cFSGL, the selected features across different time points are smooth due to the use of the fused Lasso penalty (right figure), that is, the selected features at nearby time points are similar to each other. For the example shown in the right figure, the models at M06 and M12 differ in one feature (the second feature); the models at M12 and M24 differ in one feature (the sixth feature); the models at M24 and M36 differ in two features (the first and fourth features); the models at M36 and M48 differ in one feature (the fifth feature).

while the Laplacian-based penalty focuses on the smoothing of the prediction models across different time points.

3.3 Longitudinal Stability Selection for Identifying Temporal Patterns of Biomarkers

Stability selection Meinshausen and Bühlmann (2010), based on subsampling/bootstrapping, provides a general method to perform model selection using information from a set of regularization parameters. The stability ranking score gives a probability which makes it naturally interpretable. Stability selection has been successfully applied to bioinformatics applications especially in genome-related biomarker selection problems where sample size is much smaller than feature dimension ($n \ll d$) Eleftherohorinou

et al. (2011); Ryali *et al.* (2012); Stekhoven *et al.* (2012); Vounou *et al.* (2012).

We propose to extend the idea of stability selection to longitudinal study. The framework, called *longitudinal stability selection*, is to quantify the importance of the features selected by the proposed formulations for disease progression. Specifically, we apply stability selection to multi-task learning models for longitudinal study. The stability score (between 0 and 1) of each feature is indicative of the importance of the specific feature for disease progression. In this proposal, we propose to use longitudinal stability selection with TGL and cFSGL to analyze the temporal patterns of biomarkers. The temporal pattern of stability scores of the features selected at different time point can potentially reveal how disease progresses temporally and spatially.

The longitudinal stability selection algorithm with TGL and cFSGL is given as follows. Let F be the index set of features, and let $f \in F$ denote the index of a particular feature. Let Δ be the regularization parameter space and let the stability iteration number be denoted as γ. For cFSGL an element $\delta \in \Delta$ is a triple $\langle \lambda_1, \lambda_2, \lambda_3 \rangle$. Let $B_{(i)} = \{X_{(i)}, Y_{(i)}\}$ be a random subsample from input data $\{X, Y\}$ of size $\lfloor n/2 \rfloor$ without replacement. For a given $\delta \in \Delta$, let $\hat{W}^{(i)}$ be the optimal solution of TGL or cFSGL on $B_{(i)}$. The set of features selected by the model $\hat{W}^{(i)}$ of the task at time point p is denoted by

$$U_p^{\delta}(B_{(i)}) = \{f : \hat{W}_{f,p}^{(i)} \neq 0\}.$$

We repeat this process for γ times and obtain the *selection probability* $\hat{\Pi}_{f,p}^{\delta}$ of each feature f at time point p:

$$\hat{\Pi}_{f,p}^{\delta} = \sum_{i=1}^{\gamma} I(f \in U_p^{\delta}(B_{(i)}))/\gamma,$$

where $I(.)$ is the indicator function defined as: $I(c) = 1$ if c is true and $I(c) = 0$ otherwise. The computation of selection probability is illustrated in Figure 3.4.

Repeat the above procedure for all $\delta \in \Delta$, we obtain the *stability score* for each feature f at time point p:

$$\mathcal{S}_p(f) = \max_{\delta \in \Delta}(\hat{\Pi}^{\delta}_{f,p}).$$

The computation of stability score at one time point is illustrated in Figure 3.5. The *stability vector* of a feature f at all t time points is given by $\mathcal{S}(f) = [\mathcal{S}_1(f) \dots \mathcal{S}_t(f)]$, which reveals the change of the importance of feature f at different time points. We define the *stable features* at time point p as:

$$\hat{U}_p = \{f : S_p(f) \text{ ranks among top } \eta \text{ in } F\} \tag{3.27}$$

and choose $\eta = 20$ in our experiments. We are interested in the stable features at all time points, i.e., $f \in \hat{U} = \cup_{p=1}^{t} \hat{U}_p$. Note that $\mathcal{S}(f)$ is dependent on the progression model used.

Note that if we use TGL in longitudinal stability selection, we obtain a common list of features for all time points. If we use cFSGL in longitudinal stability selection, the features selected for different time points may differ. However, the selected features at nearby time points are similar to each other. Thus, the distribution of stability scores is expected to exhibit the temporal smoothness property, that is, for each feature the stability score is smooth across different time points.

3.4 Experiments

In this section we perform experimental studies to evaluate the proposed progression models and analyze the biomarkers identified using longitudinal stability selection.

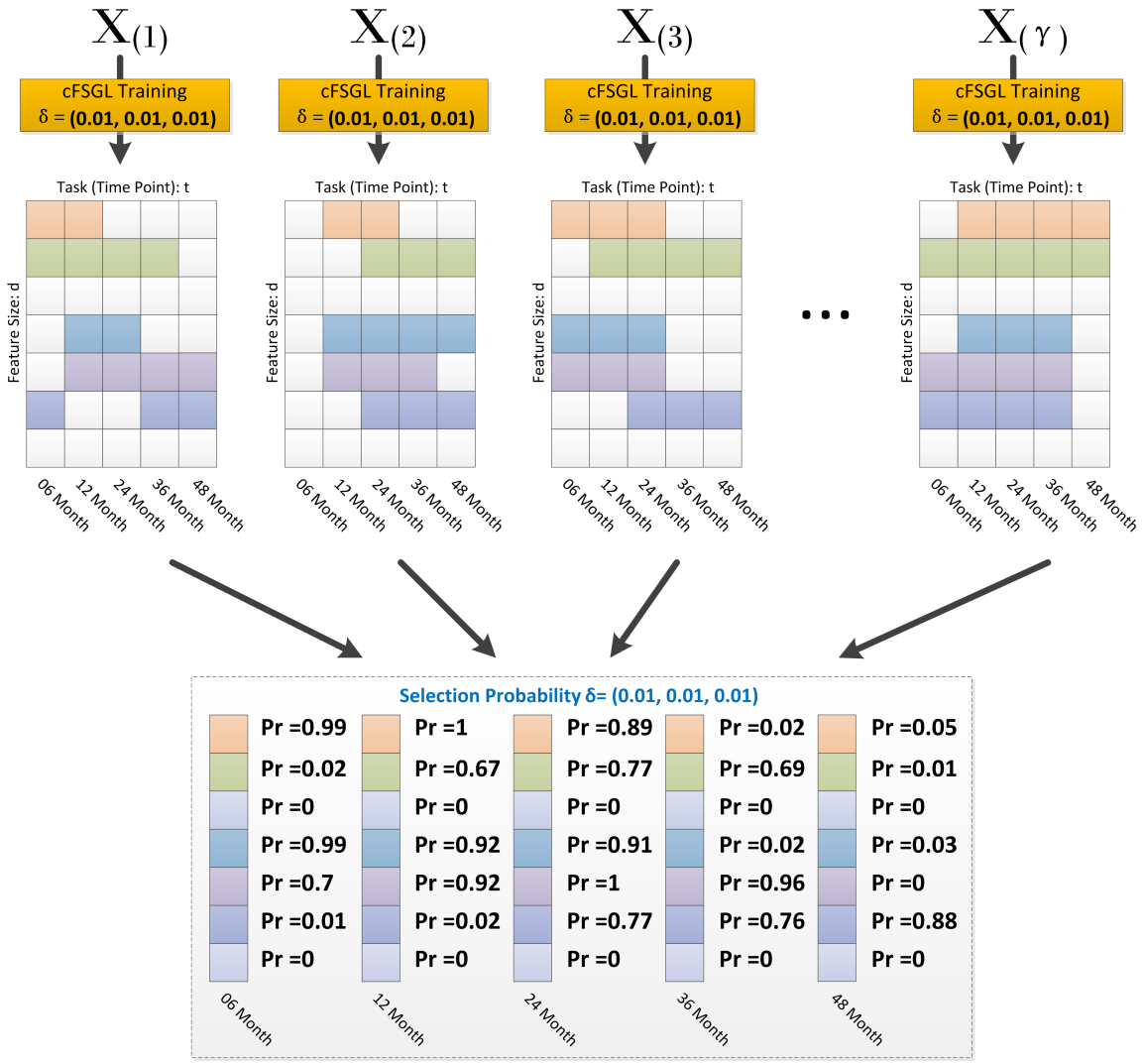


Figure 3.4: Illustration of the computation of selection probabilities for all features at all time points in longitudinal stability selection. Given a fixed parameter tuple δ, the selection probabilities are estimated based on a set of γ progression models using γ bootstrapping samples. For each feature, the selection probability at a particular time point is estimated by computing the fraction of γ models at this time point that includes a nonzero coefficient for this feature. The selection probability indicates how likely a feature is selected at one particular time point by the model parameterized by δ.

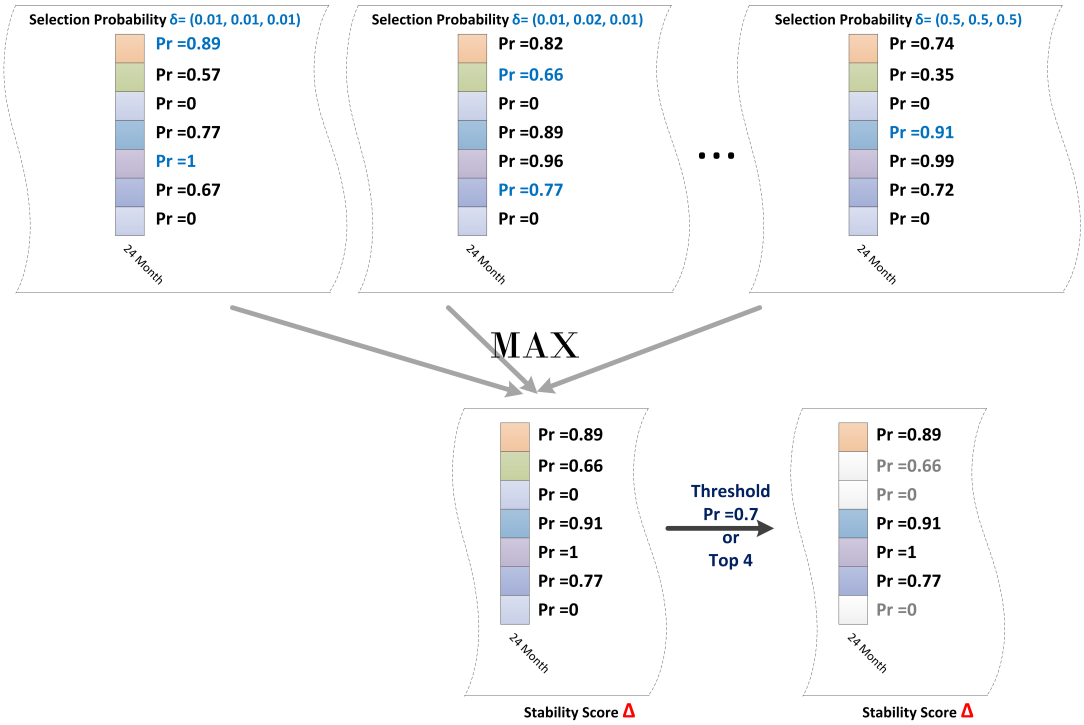


Figure 3.5: Illustration of the computation of the stability score in longitudinal stability selection at a particular time point. At each time point, the stability score of a feature is the maximum selection probability it obtains at this time point over all $\delta \in \Delta$. For the example shown in the figure, the maximum selection probability for the first feature is 0.89. After the stability score is computed, we can select features at each time point by either providing a threshold on the selection probabilities or the number of features with top selection probabilities.

3.4.1 Prediction Performance using baseline MRI features

In the first experiment, for each target we build a prediction model using baseline MRI features and baseline MMSE. We compare the proposed methods including Temporal Group Lasso (TGL) and Convex Fused Sparse Group Lasso (cFSGL) with single-task learning methods including ridge regression (Ridge) and Lasso regression (Lasso) on the prediction of MMSE and ADAS-Cog. Note that Lasso is a special case of cFSGL when both λ_2 and λ_3 are set to 0. We randomly split the data into training and testing sets using a ratio 9 : 1, i.e., we build models on 90% of the data and evaluate these models on the remaining 10% of the data. Since there are

model parameters to be selected during the training, we use 5-fold cross validation on the training data to select these parameters. For the overall regression performance measures, we use normalized mean square error (nMSE) as used in the multi-task learning literature Zhang and Yeung (2010b); Argyriou *et al.* (2008a) and weighted correlation coefficient (wR) as employed in the medical literature addressing AD progression problems Duchesne *et al.* (2009); Stonnington *et al.* (2010); Ito *et al.* (2010). For the task-specific regression performance measures, we use root mean square error (rMSE). The MSE, nMSE and weighted R-value are defined as follows:

$$\text{rMSE}(y, \hat{y}) = \sqrt{\frac{\|y - \hat{y}\|_2^2}{n}}, \tag{3.28}$$

$$\text{nMSE}(Y, \hat{Y}) = \frac{\sum_{i=1}^{t} \frac{\|Y_i - \hat{Y}_i\|_2^2}{\sigma(Y_i) n_i} n_i}{\sum_{i=1}^{t} n_i}, \tag{3.29}$$

$$\text{wR}(Y, \hat{Y}) = \frac{\sum_{i=1}^{t} \text{Corr}(Y_i, \hat{Y}_i) n_i}{\sum_{i=1}^{t} n_i}, \tag{3.30}$$

where for rMSE, y is the ground truth of target at a single time point and $\hat{y}$ is the corresponding prediction by a prediction model, for nMSE and wR, Y_i is the ground truth of target at time point i, $i = [1 : t]$ and $\hat{Y}_i$ is the corresponding predicted value, and Corr is the correlation coefficient between two vectors. We report the mean and standard deviation based on 20 iterations of experiments on different splits of data. The experimental results using 90% training data are presented in Table 3.1.

Overall our proposed approaches outperform Ridge and Lasso, in terms of both nMSE and correlation coefficient. We have the following observations: 1) The proposed multi-task learning models (TGL and cFSGL) outperform single task learning models, which verifies the use of temporal smoothness assumption in our multi-task learning formulations. 2) cFSGL performs better than TGL. This may be due to the restrictive assumption imposed in TGL. 3) The proposed cFSGL formulation witnesses significant improvement for later time points. This may be due to the

Table 3.1: Comparison of our proposed approaches (TGL and cFSGL) and single-task learning approaches (Ridge, Lasso) on longitudinal MMSE and ADAS-Cog prediction using MRI features (M) in terms of normalized mean square error (nMSE), weighted correlation coefficient (wR) and root mean square error (rMSE) at each time point. 90 percent of data is used as training data.

	Ridge	Lasso	TGL	cFSGL
Target: MMSE				
nMSE	0.548 ± 0.057	0.459 ± 0.042	0.449 ± 0.045	$\mathbf{0.395 \pm 0.052}$
wR	0.689 ± 0.030	0.746 ± 0.031	0.755 ± 0.029	$\mathbf{0.796 \pm 0.031}$
M06 rMSE	2.269 ± 0.207	2.071 ± 0.261	$\mathbf{2.038 \pm 0.262}$	2.071 ± 0.213
M12 rMSE	3.266 ± 0.556	2.973 ± 0.654	2.923 ± 0.643	$\mathbf{2.762 \pm 0.669}$
M24 rMSE	3.494 ± 0.599	3.371 ± 0.747	3.363 ± 0.733	$\mathbf{3.000 \pm 0.642}$
M36 rMSE	4.003 ± 0.853	3.786 ± 0.926	3.768 ± 0.962	$\mathbf{3.265 \pm 0.803}$
M48 rMSE	4.328 ± 1.310	3.653 ± 1.268	3.631 ± 1.226	$\mathbf{2.871 \pm 0.884}$
Target: ADAS-Cog				
nMSE	0.532 ± 0.095	0.520 ± 0.084	0.464 ± 0.067	$\mathbf{0.391 \pm 0.059}$
wR	0.705 ± 0.043	0.716 ± 0.036	0.747 ± 0.033	$\mathbf{0.803 \pm 0.024}$
M06 rMSE	5.213 ± 0.522	4.976 ± 0.518	4.820 ± 0.489	$\mathbf{4.451 \pm 0.340}$
M12 rMSE	6.079 ± 0.775	6.193 ± 0.766	5.813 ± 0.697	$\mathbf{5.230 \pm 0.589}$
M24 rMSE	7.409 ± 1.154	7.275 ± 1.099	6.835 ± 1.052	$\mathbf{6.249 \pm 0.996}$
M36 rMSE	7.143 ± 1.351	7.139 ± 1.444	6.938 ± 1.363	$\mathbf{5.928 \pm 1.064}$
M48 rMSE	6.644 ± 2.750	6.879 ± 2.465	6.000 ± 2.738	$\mathbf{5.980 \pm 1.979}$

data sparseness in later time points, as the proposed sparsity- inducing models are expected to achieve better prediction performance in this case.

We also explore the prediction models by including baseline demographic information: age, years of education and ApoE genotyping information, and baseline

ADAS-Cog scores of the patients. We follow the same experimental procedure as above. The prediction performance results are shown in Table 3.2. We see that the performance of predicting the two scores is improved significantly. For example, the weighted correlation coefficient between predicted value and true value on testing data has increased from 0.796 to 0.824 ($p < 10e-5$) for MMSE prediction and 0.803 to 0.854 ($p < 10e-5$) for ADAS-Cog prediction. We also witness the improvement in prediction performance at all time points. We show the scatter plots for the predicted values versus actual values for MMSE and ADAS-Cog on the testing data in Figure 3.6 and Figure 3.7, respectively. Since there are few samples available at the last time point (M48), we only show the scatter plots for the first four time points. In the scatter plots, we see that the predicted values and actual clinical scores have high correlation. The scatter plots show that the prediction performance for ADAS-Cog is better than that of MMSE.

In the study of ADNI, cognitive normal individuals and stable MCI patients are less likely to have significant changes on the cognitive scores and therefore many existing studies focus on subgroups of patients only (e.g., Duchesne *et al.* (2009)). To this end, we apply our models on the subgroup that consists of MCI converters and AD patients only. At the last time point M48, there are only very few samples available and we therefore exclude the last time point from our study. We follow the same experimental setting as in the previous experiment, and the results are shown in Table (3.3). We observe that cFSGL achieves the best performance among all methods, with an average performance of $R = 0.671(p < 10e-5)$ in predicting longitudinal MMSE scores and an average of $R = 0.751(p < 10e-5)$ in predicting ADAS-Cog.

Table 3.2: Comparison of our proposed approaches (TGL and cFSGL) and single-task learning approaches (Ridge, Lasso) on longitudinal MMSE and ADAS-Cog prediction using MRI, demographic, and ApoE genotyping features in terms of normalized mean square error (nMSE), weighted correlation coefficient (wR) and root mean square error (rMSE) at each time point. 90 percent of data is used as training data.

	Ridge	Lasso	TGL	cFSGL
Target: MMSE				
nMSE	0.477 ± 0.055	0.368 ± 0.048	0.364 ± 0.046	$\mathbf{0.341 \pm 0.039}$
wR	0.743 ± 0.022	0.809 ± 0.026	0.811 ± 0.027	$\mathbf{0.824 \pm 0.021}$
M06 rMSE	2.211 ± 0.241	1.938 ± 0.214	$\mathbf{1.900 \pm 0.211}$	1.980 ± 0.219
M12 rMSE	2.968 ± 0.685	2.679 ± 0.769	2.654 ± 0.767	$\mathbf{2.546 \pm 0.748}$
M24 rMSE	3.454 ± 0.550	3.107 ± 0.570	3.133 ± 0.579	$\mathbf{2.943 \pm 0.582}$
M36 rMSE	3.736 ± 0.792	3.311 ± 0.756	3.313 ± 0.798	$\mathbf{3.046 \pm 0.701}$
M48 rMSE	3.469 ± 1.030	2.645 ± 0.845	2.761 ± 0.883	$\mathbf{2.364 \pm 0.792}$
Target: ADAS-Cog				
nMSE	0.396 ± 0.075	0.335 ± 0.048	0.317 ± 0.044	$\mathbf{0.296 \pm 0.048}$
wR	0.791 ± 0.031	0.830 ± 0.020	0.837 ± 0.017	$\mathbf{0.854 \pm 0.021}$
M06 rMSE	4.384 ± 0.522	3.936 ± 0.430	3.858 ± 0.441	$\mathbf{3.863 \pm 0.516}$
M12 rMSE	4.906 ± 0.708	4.578 ± 0.756	4.455 ± 0.661	$\mathbf{4.209 \pm 0.564}$
M24 rMSE	6.587 ± 1.038	6.153 ± 1.145	5.945 ± 1.120	$\mathbf{5.657 \pm 1.017}$
M36 rMSE	6.312 ± 1.068	5.849 ± 1.028	5.613 ± 0.936	$\mathbf{5.066 \pm 0.854}$
M48 rMSE	5.679 ± 2.200	5.087 ± 2.082	5.181 ± 2.383	$\mathbf{5.182 \pm 1.606}$

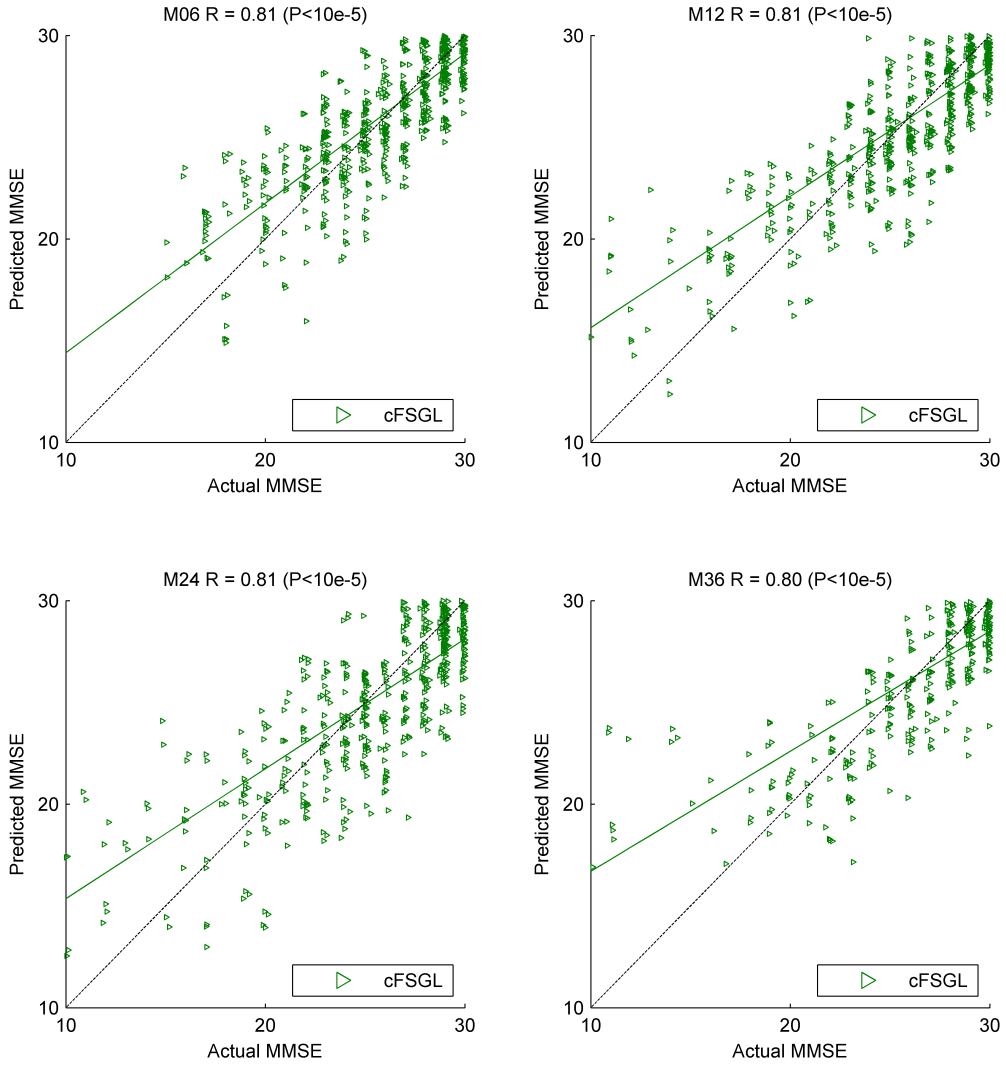


Figure 3.6: Scatter plots of actual MMSE versus predicted values on testing data using cFSGL based on baseline MRI features, demographic, and ApoE genotyping features. The black dashed line in each figure is a reference of perfect correlation (predicted value exactly equals to actual value). We perform least squares regression on the points shown in the scatter plots and the green solid line is the regression line, which serves as a visual indicator of overall performance. The closer between the regression line and the reference line, the better are the prediction results. We see that the patients with low actual MMSE scores are less predictable, compared to the ones with high actual MMSE scores.

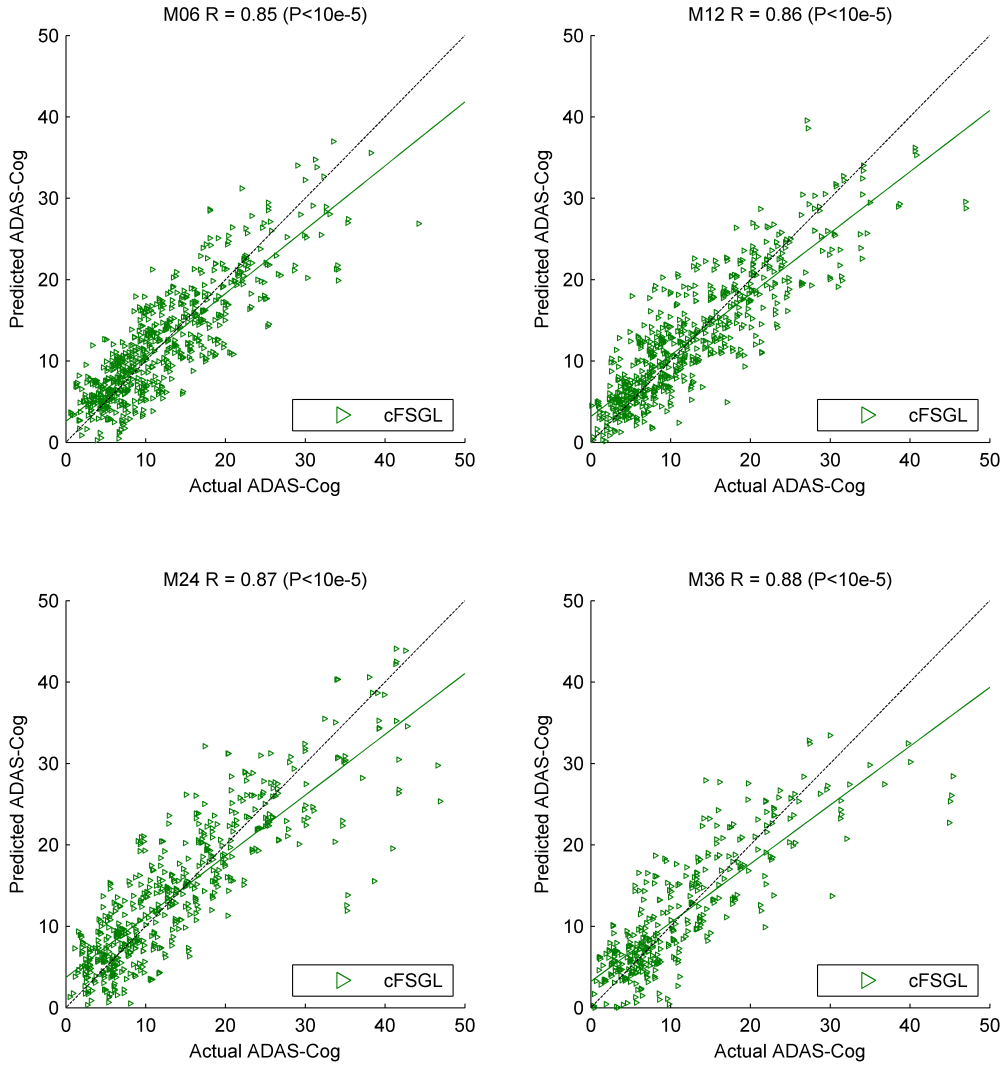


Figure 3.7: Scatter plots of actual ADAS-Cog versus predicted values on testing data using cFSGL based on baseline MRI features, demographic, and ApoE genotyping features. The black dashed line in each figure is a reference of perfect correlation (predicted value exactly equals to actual value). We perform least squares regression on the points shown in the scatter plots and the green solid line is the regression line, which serves as a visual indicator of overall performance. The closer between the regression line and the reference line, the better are the prediction results. We see high correlation between the two values. The visual prediction performance for ADAS-Cog is better than that of MMSE as shown in Figure 3.6.

Table 3.3: Comparison of our proposed approaches (TGL and cFSGL) and single-task learning approaches (Ridge, Lasso) on longitudinal MMSE and ADAS-Cog prediction for MCI converters and AD patients using MRI, demographic, and ApoE genotyping features in terms of normalized mean square error (nMSE), weighted correlation coefficient (wR) and root mean square error (rMSE) at each time point. 80 percent of data is used as training data.

	Ridge	Lasso	TGL	cFSGL
Target: MMSE				
nMSE	1.161 ± 0.269	0.860 ± 0.137	0.761 ± 0.143	$\mathbf{0.725 \pm 0.128}$
wR	0.526 ± 0.080	0.633 ± 0.068	0.660 ± 0.059	$\mathbf{0.671 \pm 0.054}$
M06 rMSE	3.420 ± 0.381	3.031 ± 0.280	2.881 ± 0.245	$\mathbf{2.862 \pm 0.231}$
M12 rMSE	4.025 ± 0.482	3.680 ± 0.531	3.391 ± 0.489	$\mathbf{3.315 \pm 0.506}$
M24 rMSE	5.531 ± 0.756	4.988 ± 0.924	4.636 ± 0.883	$\mathbf{4.551 \pm 0.870}$
M36 rMSE	5.971 ± 1.214	5.011 ± 1.231	4.686 ± 1.077	$\mathbf{4.422 \pm 1.046}$
Target: ADAS-Cog				
nMSE	1.031 ± 0.200	0.748 ± 0.078	0.675 ± 0.079	$\mathbf{0.533 \pm 0.101}$
wR	0.569 ± 0.059	0.695 ± 0.045	0.704 ± 0.042	$\mathbf{0.751 \pm 0.046}$
M06 rMSE	6.256 ± 0.813	5.692 ± 0.591	5.381 ± 0.583	$\mathbf{5.140 \pm 0.800}$
M12 rMSE	7.320 ± 0.988	6.334 ± 1.022	5.934 ± 0.884	$\mathbf{5.196 \pm 0.829}$
M24 rMSE	10.423 ± 1.224	9.353 ± 1.301	8.964 ± 1.331	$\mathbf{7.486 \pm 1.249}$
M36 rMSE	10.968 ± 1.833	9.319 ± 2.082	8.782 ± 1.801	$\mathbf{6.958 \pm 1.499}$

Chapter 4

MULTI-TASK LEARNING FOR PATIENT RECORD DENSIFICATION

In this chapter, I present a framework and two concrete algorithms for patient record densification, which densify the sparse Electronic Medical Records (EMR) data, imputing the values of those missing entries by exploring the latent structures on both feature and time dimensions. From the densified EMR data, we expect to build more effective predictive models than the ones built on raw EMR data. The first densification algorithm utilizes information within each patient, i.e., the densification process for one patient is independent from those for other patients. Considering the densification of each patient as a task, the second densification algorithm performs multi-task densification. The multi-task densification model utilizes the information from other patients when densifying one patient, exploring useful information from patients with similar medical conditions.

The rest of this chapter is organized as follows: Section 4.1 presents the general representation of EMR and the problem of patient risk prediction which is one important problem that patient phenotyping will be applied to. In Section 4.2 we introduce the details of Pacifier. The experimental results are presented in Section 4.3. In Section 4.4 we discuss the connection of the proposed approaches to related work and insights for future works.

4.1 Patient Risk Prediction with Electronic Medical Records

Risk prediction is among the most important applications in clinical decision support systems and care management systems, where it often requires building predictive models for a specific disease condition. As Electronic Medical Records (EMR)

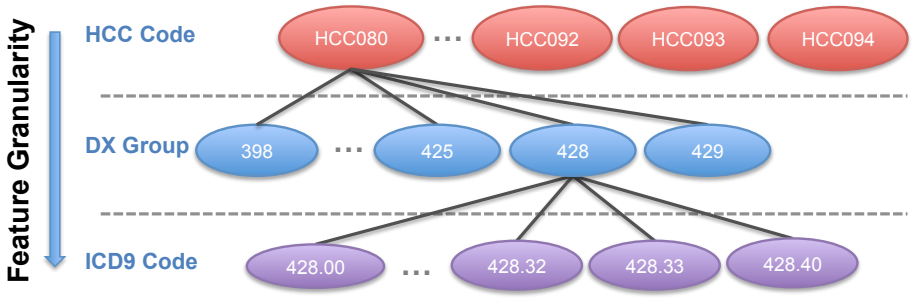


Figure 4.1: Granularity of medical features. For diagnosis events, features can be constructed at different levels of granularity: ICD9 code, diagnosis code (DxGroup) and HCC code.

data becomes widely available, informative features for risk prediction can be constructed from EMR. Based on the EMR data, for example, care providers usually want to assess the risk scores of a patient developing different disease conditions, such as congestive heart failure Wu *et al.* (2010); Fonarow *et al.* (2005), diabetes Stern *et al.* (2008), and end stage renal disease Blacher *et al.* (2001). Once the risk of a patient is predicted, proper intervention and care plan can be designed accordingly.

The detailed EMR data documents the patient events in time, which typically includes diagnosis, medication, and clinical notes. The diagnosis events are among the most structured, feasible and informative events, and are prime candidates for constructing features for risk prediction Philbin and DiSalvo (1999); Van Staa *et al.* (2002). The diagnosis events, often in the form of International Classification of Diseases 9 (ICD9) codes, also come with well-defined feature groups at various levels of granularity such as diagnosis group (DxGroup) and higher-level hierarchical condition categories (HCC). For example, the code 401.1 *Benign Hypertension* belongs to DxGroup 401 *Essential Hypertension*, which is a subcategory in HCC 091 *Hypertension*.

One of the key steps of risk prediction from EMR is to construct features vectors from EMR events, which are used as inputs for classifiers. The goal of feature construction is to capture sufficient clinical nuances that are informative to a specific risk prediction task. Traditionally the feature vectors are directly derived from the

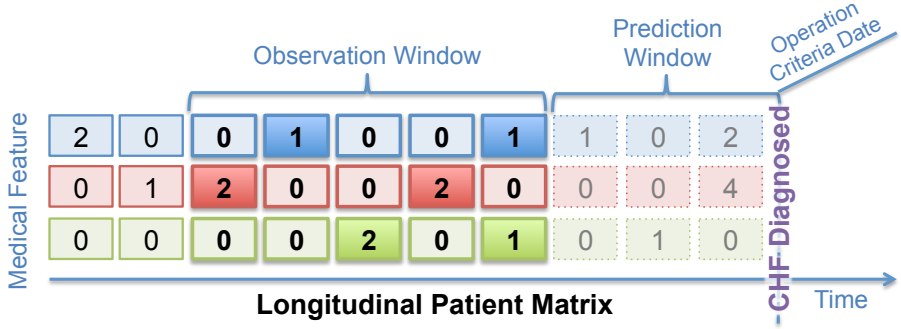


Figure 4.2: Construction of the longitudinal patient matrix Wang *et al.* (2012) from Electronic Medical Records (EMR). The goal is to predict disease status of a patient at the operation criteria date (OCD), given the past medical information before the prediction window. For each patient, we construct a longitudinal patient matrix, using medical features at a specific granularity. For each patient, the feature vector for classification/regression is finally generated by extracting summary statistics from the longitudinal matrix within the observation window.

raw EMR records Wu *et al.* (2010); Wang *et al.* (2012); Sun *et al.* (2012). In this paper for each patient we first construct a *longitudinal patient matrix*, with a feature dimension and a time dimension Wang *et al.* (2012). Maintaining the time dimension enables us to leverage the temporal information of the patients during feature construction. We present the procedure of constructing feature vectors via longitudinal patient matrices as follows.

In a cohort for a disease study, each patient is also associated with a disease status date called *operation criteria date* (OCD), on which the disease is diagnosed. A typical risk prediction task is to predict the disease status of the patients at a certain time point in the future (e.g., half a year). We call this period as the *prediction window*. To build useful predictive models, a prediction window before the OCD is usually specified, and the records before the prediction window are used to train the models, i.e., all records within the prediction window before the OCD are considered to be invisible. Figure 4.2 illustrates the raw EMR data, OCD, and prediction window.

The next step is to construct a longitudinal patient matrix for each patient from

the available EMR events, which consists of two dimensions: the feature dimension and the time dimension. One straightforward way to construct such matrices is to use the finest granularity in both dimensions: use the types of medical events as the feature space for the feature dimension and use *day* as the basic unit for time dimension. Unfortunately the patient matrices constructed in this way are too sparse to be useful. As a remedy, we use *weekly* aggregated time, and the value of each medical feature at one time point is given by the counts of the corresponding medical events within that week. Recall that the medical features can be retrieved at different levels of granularity, which also moderately reduces some sparsity in the data. The choice of feature granularity should not be too coarse, otherwise predictive information within features at a finer level may be lost during the retrieval, as we will show in the experiments. Note that after these preprocessing steps, the constructed patient matrices are still very sparse.

Finally we need to extract summary statistics from the longitudinal patient matrices as the feature vectors for classifiers. Since patients have different lengths of records, typically an *observation window* of interest is defined and the summary statistics (e.g., mean, standard deviation) are extracted within the observation window for all patients. The overall process is given in Figure 4.2.

4.2 Temporal Densification via Pacifier

During the aforementioned feature construction process, there are many zeros in the longitudinal patient matrices due to the extreme sparsity in the raw EMR data. However, many of these zeros are not real zeros and instead, they indicate missing information (i.e, no visit). Treated as informative values in the feature extraction process, these values are likely to bias the training of classifiers and yield suboptimal performance. In this paper we propose to treat the zeros in the longitudinal patient

matrices as missing values, and we densify the sparse matrices before extracting features to reduce the bias introduced by the sparsity, in hopes of that, the densified matrices provide better phenotyping of patients. We propose novel frameworks of densifying the partially observed longitudinal patient matrices, leveraging their observed medical histories. The proposed framework explores the latent structures on both feature and time dimensions and encourages the temporal smoothness of each patient.

Let there be n patients with EMR records available in the cohort, and there be in total p medical features. After the feature construction process we obtain n longitudinal patient matrices with missing entries, one for each patient. For the ith patient, its time dimension is denoted by t_i, i.e., there are medical event records covering a time span of t_i before the prediction window. We denote the ground truth matrix of the ith patient as $X_{(i)} \in \mathbb{R}^{p \times t_i}$, and in our medical records we only have a partial observation of the matrix at some locations, whose indices are given by a set $\Omega_{(i)}$. Accrording to the marco phenotype assumption, we assume the medical features can be mapped to some latent medical concepts space with a much lower *latent dimension* of size k, such that each medical concept can be viewed as a combination of several observed medical features.

Specifically, we assume that the full longitudinal patient matrix can be approximated by a low rank matrix $X_{(i)} \approx U_{(i)}V_{(i)}$, which can be factorized into a sparse matrix $U_{(i)} \in \mathbb{R}^{p \times k}$ whose columns provide mappings from medical features to medical concepts, and a dense matrix $V_{(i)} \in \mathbb{R}^{k \times t_i}$ whose rows indicates the temporal evolution of these medical concepts acting on the patient over time. We call $U_{(i)}$ the *latent medical concept mapping matrix* (abbr. *latent mapping matrix*) and $V_{(i)}$ the *concept value evolution matrix* (abbr. *evolution matrix*). For each patient we assume that the values of those medical concepts evolve smoothly over time. Given

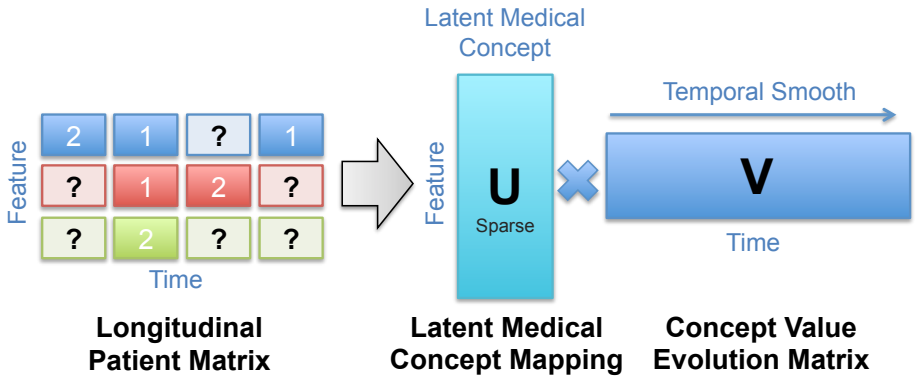


Figure 4.3: Illustration of the PACIFIER framework. We treat a longitudinal patient matrix as a partially observed matrix from a *complete patient matrix*. We assume the medical features can be mapped to some latent medical concepts with a much lower dimensionality such that each medical concept can be viewed as a combination of several observed medical features. For each patient, the values of those medical concepts evolve smoothly over time. Thus the complete patient matrix for each patient can be factorized into a latent medical concept mapping matrix and a concept value evolution matrix.

the values and locations of observed elements in the longitudinal patient matrices, our proposed densification method learns their latent mapping matrices and evolution matrices. We call this densification framework PACIFIER, which stands for PAtient reCord densIFIER. The idea of PACIFIER is illustrated in Figure 4.3.

Based on different natures of the medical cohorts, homogeneous or heterogeneous, we propose two densification formulations: an individual basis approach for heterogeneous patients and a shared basis approach for homogeneous patients, and then we provide an efficient optimization algorithm for PACIFIER that can be used to solve large-scale problems. Here and later we abuse the word *basis* to denote the columns of a concept mapping matrix, while we don't require them to be orthonormal. Note that the *real basis* of the space spanned by the columns of the latent mapping matrix can always be obtained by performing QR factorization on this basis matrix U_i.

4.2.1 Individual Basis Approach for Heterogeneous Cohort

In the heterogeneous cohort where patients are very different from each other in nature, the medical concepts for each patient may also be different from one patient to another. In the individual basis approach (PACIFIER-IBA), we allow patients to have different latent medical concepts.

Let $\Omega_{(i)}^{\mathsf{c}}$ denote the complement of $\Omega_{(i)}$. We adopt the projection operator $\mathcal{P}_{\Omega_{(i)}}(X_{(i)})$ used in matrix completion Cai *et al.* (2010):

$$\mathcal{P}_{\Omega_{(i)}}(X_{(i)}) = \begin{cases} X_{(i)}(j,k) & \text{if } (j,k) \in \Omega_{(i)} \\ 0 & \text{if } (j,k) \in \Omega_{(i)}^{\mathsf{c}} \end{cases}$$

An intuitive approach for formulating PACIFIER-IBA is to solve the following problem for each patient:

$$\min_{U_{(i)} \geq 0, V_{(i)}} \frac{1}{2t_i} \|\mathcal{P}_{\Omega_{(i)}}(U_{(i)}V_{(i)} - X_{(i)})\|_F^2 + \mathcal{R}(U_{(i)}, V_{(i)}) \tag{4.1}$$

where $\mathcal{R}(U_{(i)}, V_{(i)})$ denotes the regularization terms that encode our assumptions and prevent overfitting. We also impose a non-negative constraint on the medical concept $U_{(i)}$ because most medical events and measurements in EMR are non-negative, and meaningful medical concepts consist of these medical events should also be non-negative. We now discuss how to design proper terms in $\mathcal{R}(U_{(i)}, V_{(i)})$ that lead to some desired properties:

1) Sparsity. We want only a few significant medical features to be involved in each medical concept so that the concepts can be interpretable. Therefore, we introduce sparsity in the latent mapping matrix $U_{(i)}$ via sparse inducing ℓ_1-norm on $U_{(i)}$. Indeed the non-negativity constraint may have already brought a certain amount of sparsity, and it has been shown that for non-negative matrix factorization, the sparsity regularization can further improve the decomposition Hoyer (2004).

2) Overfitting. To overcome overfitting we introduce an ℓ_2 regularization on the concept value evolution matrix $V_{(i)}$. It can be shown that this term also improves the numerical condition of computing a matrix inversion in our algorithm.

3) Temporal smoothness. A patient matrix describes the continuous evolution of medical features for a patient over time. Thus, along the time dimension it makes intuitive sense to impose the temporal smoothness, such that the value of one column of a longitudinal patient matrix is close to those of its previous and next columns. To this end, we introduce the temporal smoothness regularization on the columns of the concept value evolution, which describes the smooth evolution on the medical concepts. One commonly used strategy to enforce temporal smoothness is via penalizing pairwise difference Zhou *et al.* (2011b):

$$\|V_{(i)}R_{(i)}\|_F^2 = \sum\nolimits_{j=1}^{t_i-1}(V_{(i)}(:,j) - V_{(i)}(:,j+1))^2$$

where $R_{(i)} \in \mathbb{R}^{t_i \times t_i+1}$ is the temporal smoothness coupling matrix defined as follows: $R_{(i)}(j,k) = 1$ if $i = j$, $R_{(i)}(j,k) = -1$ if $i = j+1$, and $R_{(i)}(j,k) = 0$ otherwise.

In the loss function of Eq. (4.1) we want the values of the low-rank matrix to be close to $X_{(i)}$ at the observed locations, directly solving which may lead to complex algorithms. An alternative way is to introduce an intermediate matrix S_i such that $\mathcal{P}_{\Omega_i}(S_i) = \mathcal{P}_{\Omega_i}(X_i)$, and we want $U_{(i)}V_{(i)}$ to be close to $S_{(i)}$. An immediate advantage of propagating the observed information from $X_{(i)}$ to $U_{(i)}V_{(i)}$ indirectly is that we can derive very efficient algorithms and data structures, which give the capability of solving large-scale problems, as we will show later. To this end, we propose the

following PACIFIER-IBA learning model for each patient:

$$\begin{aligned}\min_{S_{(i)},U_{(i)},V_{(i)}} &\tfrac{1}{2t_i}\|S_{(i)} - U_{(i)}V_{(i)}\|_F^2 + \lambda_1\|U_{(i)}\|_1 \\ &+\lambda_2\tfrac{1}{2t_i}\|V_{(i)}\|_F^2 + \lambda_3\tfrac{1}{2t_i}\|V_{(i)}R_{(i)}\|_F^2, \\ \text{subject to:}\quad &\mathcal{P}_{\Omega_{(i)}}(S_{(i)}) = \mathcal{P}_{\Omega_{(i)}}(X_{(i)}), U_{(i)} \geq 0\end{aligned} \tag{4.2}$$

which can be collectively written as:

$$\begin{aligned}\min_{\{S_{(i)},U_{(i)},V_{(i)}\}} &\sum_{i=1}^{n}\tfrac{1}{2t_i}\|S_{(i)} - U_{(i)}V_{(i)}\|_F^2 + \lambda_1\sum_{i=1}^{n}\|U_{(i)}\|_1 \\ &+\lambda_2\sum_{i=1}^{n}\tfrac{1}{2t_i}\|V_{(i)}\|_F^2 + \lambda_3\sum_{i=1}^{n}\tfrac{1}{2t_i}\|V_{(i)}R_{(i)}\|_F^2. \\ \text{subject to:}\quad &\mathcal{P}_{\Omega_{(i)}}(S_{(i)}) = \mathcal{P}_{\Omega_{(i)}}(X_{(i)}), U_{(i)} \geq 0, \forall i\end{aligned} \tag{4.3}$$

4.2.2 Shared Basis Approach for Homogeneous Cohort

In homogeneous cohorts where the medical concepts of patients are very similar to each other, we can assume that all patients share the same medical concept mapping $U \in \mathbb{R}^{p\times k}$. We propose the following PACIFIER-SBA formulation:

$$\begin{aligned}\min_{\{S_{(i)}\},U,\{V_{(i)}\}} &\sum\nolimits_{i=1}^{n}\tfrac{1}{2t_i}\|S_{(i)} - UV_{(i)}\|_F^2 + \lambda_1\|U\|_1 \\ &+\lambda_2\sum\nolimits_{i=1}^{n}\tfrac{1}{2t_i}\|V_{(i)}\|_F^2 + \lambda_3\sum\nolimits_{i=1}^{n}\tfrac{1}{2t_i}\|V_{(i)}R_{(i)}\|_F^2 \\ \text{subject to:}\quad &\mathcal{P}_{\Omega_{(i)}}(S_{(i)}) = \mathcal{P}_{\Omega_{(i)}}(X_{(i)}), U \geq 0\end{aligned} \tag{4.4}$$

Since the densification of all patients are now coupled via the shared concept mapping, an immediate benefit of the PACIFIER-SBA formulation is that, we can transfer some knowledge among the patients, which is attractive especially when the available information for each patient is very limited and the patients are homogeneous in nature. We demonstrate in the experiments that the PACIFIER-SBA performs better than IBA when patients are homogeneous.

4.2.3 Optimization Algorithm

The formulations of PACIFIER are non-convex and we present a block coordinate descent (BCD) optimization algorithm to obtain a local solution. Note that for each patient the subproblem of PACIFIER-IBA in Eq. (4.2) is a special case of the problem of PACIFIER-SBA in Eq. (4.4) given $n = 1$. Therefore in this section we present the algorithm for Eq. (4.4).

1) Solve U^+ given $V^-_{(i)}$ and $S^-_{(i)}$:

$$U^+ = \underset{U \geq 0}{\operatorname{argmin}} \sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i} \|S^-_{(i)} - U V^-_{(i)}\|_F^2 + \lambda_1 \|U\|_1. \tag{4.5}$$

This is a standard non-negative ℓ_1-norm regularized problem and can be solved efficiently using scalable first order methods such as spectral projected gradient Wright *et al.* (2009) and proximal Quasi-Newton method Lee *et al.* (2012).

2) Solve $V^+_{(i)}$ given U^+ and $S^-_{(i)}$:

$$\begin{aligned} \{V^+_{(i)}\} = \underset{\{V_{(i)}\}}{\operatorname{argmin}} & \sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i} \|S^-_{(i)} - U^+ V_{(i)}\|_F^2 \\ & + \lambda_2 \sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i} \|V_{(i)}\|_F^2 + \lambda_3 \sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i} \|V_{(i)} R_{(i)}\|_F^2 \end{aligned} \tag{4.6}$$

Note that the terms are decoupled for each patient, resulting in a set of minimization problems:

$$V^+_{(i)} = \underset{V_{(i)}}{\operatorname{argmin}} \tfrac{1}{2} \|S^-_{(i)} - U^- V_{(i)}\|_F^2 + \tfrac{\lambda_2}{2} \|V_{(i)}\|_F^2 + \tfrac{\lambda_3}{2} \|V_{(i)} R_{(i)}\|_F^2, \tag{4.7}$$

The problem in (4.7) can be solved using existing optimization solvers. Moreover, since the problem is smooth, it admits a simple analytical solution Zhou *et al.* (2011b).

Lemma 4.2.1. *Let $Q_1 \Lambda_1 Q_1^T = U^T U + \lambda_2 I$ and $Q_2 \Lambda_2 Q_2^T = \lambda_3 R_{(i)} R_{(i)}^T$ be eigen-decompositions, and let $D = Q_1^T U^T S_{(i)} Q_2$, the problem (4.7) admits an analytical solution:*

$$V^*_{(i)} = Q_1 \hat{V} Q_2, \quad \textit{where } \hat{V}_{j,k} = \tfrac{D_{j,k}}{\Lambda_1(j,j) + \Lambda_2(k,k)}. \tag{4.8}$$

Proof. We first set the gradient of (4.7) to zero:

$$U^T U V_{(i)}^* - U^T S_{(i)} + \lambda_2 V_{(i)}^* + \lambda_3 V_{(i)}^* R_{(i)} R_{(i)}^T = 0$$
$$\Rightarrow (U^T U + \lambda_2 I) V_{(i)}^* + \lambda_3 V_{(i)}^* R_{(i)} R_{(i)}^T = U^T S_{(i)}$$

To obtain $V_{(i)}^*$ we need to solve the above linear system. We perform eigen decomposition $Q_1 \Lambda_1 Q_1^T = U^T U + \lambda_2 I$ and $Q_2 \Lambda_2 Q_2^T = \lambda_3 R_{(i)} R_{(i)}^T$ and it follows that

$$Q_1 \Lambda_1 Q_1^T V_{(i)}^* + V_{(i)}^* Q_2 \Lambda_2 Q_2^T = U^T S_{(i)}$$
$$\Rightarrow \Lambda_1 Q_1^T V_{(i)}^* Q_2 + Q_1^T V_{(i)}^* Q_2 \Lambda_2 = Q_1^T U^T S_{(i)} Q_2$$
$$\Rightarrow \Lambda_1 \hat{V} + \hat{V} \Lambda_2 = Q_1^T U^T S_{(i)} Q_2 := D$$

therefore we can solve $\hat{V}_{j,k} = \frac{D_{j,k}}{\Lambda_1(j,j) + \Lambda_2(k,k)}$ and the optimal solution is thus given by $V_{(i)}^* = Q_1 \hat{V} Q_2$, which completes the proof. □

Note that the parameter λ_2 improves the stability of the 'inversion' in $V_{j,k}$ so that the denominator is guaranteed to be a positive number. Excluding the time of the two QR factorizations, the cost of computing the analytical form solution for each sample is given by $O(k^2 pt)$. The computation can be greatly accelerated as shown in the next section. Including the time of QR factorizations, obtaining the results from the analytical form is typically 100 times faster than that of solving (4.6) using optimization solvers.

3) Solve $S_{(i)}^+$ given U^+ and $V_{(i)}^+$:

$$\{S_{(i)}^+\} = \underset{\{S_{(i)}\}}{\operatorname{argmin}} \sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i} \|S_{(i)} - U^+ V_{(i)}^+\|_F^2. \tag{4.9}$$
$$\text{subject to:} \quad \mathcal{P}_{\Omega_{(i)}}(S_{(i)}) = \mathcal{P}_{\Omega_{(i)}}(X_{(i)})$$

The problem is a constrained Euclidean projection, and is decoupled for each $S_{(i)}^+$. The subproblem for each one admits a closed-form solution: $S_{(i)}^+ = \mathcal{P}_{\Omega_{(i)}^c}(U^+ V_{(i)}^+) + \mathcal{P}_{\Omega_{(i)}}(X_{(i)})$.

Algorithm 2 The BCD algorithm for solving the PACIFIER-SBA in formulation (4.4). Given $n = 1$, the algorithm also solves the PACIFIER-IBA for each patient in the formulation (4.2).

Input: Observed locations $\{\Omega_{(i)}\}_1^n$, values of the observed entries for each patient $\{\mathcal{P}_{\Omega_{(i)}}(X_{(i)})\}_1^n$, initial solutions $\{V_{(i)}^0\}_1^n$, sparse parameter λ_1, parameter λ_2, smooth parameter λ_3, latent factor k.

Output: U^+, $\{V_{(i)}^+\}_1^n$, $\{S_{(i)}^+\}_1^n$.

Set $V_{(i)}^- = V_{(i)}^0$, $S_{(i)}^- = \mathcal{P}_{\Omega_{(i)}}(X_{(i)})$ for all i.

while true **do**

Update U^+ by solving (4.5) via ℓ_1 solvers (e.g. Lee *et al.* (2012); Wright *et al.* (2009)).

Update $V_{(i)}^+$ by computing (4.8).

Update $S_{(i)}^+ = \mathcal{P}_{\Omega_{(i)}^c}(U^+ V_{(i)}^+) + \mathcal{P}_{\Omega}(X_{(i)})$

if U^+ and $\{V_{(i)}^+\}_1^n$ converge **then**

return U^+ and $\{V_{(i)}^+\}_1^n$

end if

Set $V_{(i)}^- = V_{(i)}^+$ and $S_{(i)}^- = S_{(i)}^+$ for all i.

end while

We summarize the BCD algorithm of PACIFIER-SBA in Algorithm 2. In our implementation, we randomly generate the initial concept evolution matrix $V_{(i)}^0$, and set $U_{(i)}^0 = (0)$. Therefore the initial value of $S_{(i)}^-$ is given by $S_{(i)}^- = \mathcal{P}_{\Omega_{(i)}}(X_{(i)}) + \mathcal{P}_{\Omega_{(i)}^c}(\mathbf{0}V_{(i)}^0) = \mathcal{P}_{\Omega_{(i)}}(X_{(i)})$. Since the problem of PACIFIER is non-convex, and thus it is easy to fall into a local minimum. One way to escape from local minimum is to 'restart' the algorithm by slightly perturbing V_i after the algorithm converges, and compute a new solution. Among the many solutions, we use the one with the lowest function value. In the following section we discuss how to accelerate the algorithm to

solve large-scale problems.

4.2.4 Efficient Computation for Large Scale Problems

For large scale problems, the storage of the matrix S_i and $O(d^2)$-level computations are prohibitive. However, we notice that in each iteration, we have that $S^+_{(i)} = \mathcal{P}_{\Omega_i^c}(U^+V^+_{(i)}) + \mathcal{P}_{\Omega_i}(X_{(i)}) = U^+V^+_{(i)} + \mathcal{P}_{\Omega_i}(X_{(i)} - U^+V^+_{(i)})$. The "low rank + sparse" structure of $S^+_{(i)}$ indicates that there is no need to store the full matrices. Instead we only need to store two smaller matrices depending on k and a sparse residual matrix $\mathcal{P}_{\Omega_i}(X_{(i)} - U^+V^+_{(i)})$. This structure can be used to greatly accelerate the computation of Eqs. (4.5) and (4.6). In the following discussion we denote $S_{(i)} = U_{S_{(i)}}V_{S_{(i)}} + S_{S_{(i)}}$.

1) Solve U. The major computational cost of Eq. (4.5) lies on the evaluation of the loss function and the gradient of the smooth part. Taking advantage of the structure of S_i. We show that all prohibitive $O(d^2)$ level operations can be avoided given the special structures of $S^+_{(i)}$.

Gradient Evaluation:

$$
\begin{aligned}
&\nabla_U \left(\sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i} \|S_{(i)} - UV_{(i)}\|_F^2 \right) \\
=&\nabla_U \left(\sum\nolimits_{i=1}^{n} \tfrac{1}{2t_i} \|(U_{S_{(i)}}V_{S_{(i)}} + S_{S_{(i)}}) - UV_{(i)}\|_F^2 \right) \\
=&\sum\nolimits_{i=1}^{n} \tfrac{1}{t_i} \operatorname{tr}\left(UV_{(i)}V_{(i)}^T - S_{(i)}V_{(i)}^T \right) \\
=&\sum\nolimits_{i=1}^{n} \tfrac{1}{t_i} \left(U(V_{(i)}V_{(i)}^T) - U_{S_{(i)}}(V_{S_{(i)}}V_{(i)}^T) - S_{S_{(i)}}V_{(i)}^T \right)
\end{aligned}
$$

Objective Evaluation:

$$
\begin{aligned}
&\sum_{i=1}^{n} \tfrac{1}{2t_i} \|S_{(i)} - UV_{(i)}\|_F^2 \\
=&\sum_{i=1}^{n} \tfrac{1}{2t_i} \operatorname{tr}\left(S_{(i)}^T S_{(i)} - 2S_{(i)}^T UV_{(i)} + V_{(i)}^T U^T UV_{(i)}\right) \\
=&\sum_{i=1}^{n} \frac{1}{2t_i} \operatorname{tr}\left((U_{S_{(i)}} V_{S_{(i)}} + S_{S_{(i)}})^T (U_{S_{(i)}} V_{S_{(i)}} + S_{S_{(i)}})\right) + \\
&\sum_{i=1}^{n} \frac{1}{2t_i} \operatorname{tr}\left(-2(U_{S_{(i)}} V_{S_{(i)}} + S_{S_{(i)}})^T UV_{(i)} + V_{(i)}^T U^T UV_{(i)}\right) \\
=&\sum_{i=1}^{n} \tfrac{1}{2t_i} \Bigg(\operatorname{tr}\left(V_{S_{(i)}}^T (U_{S_{(i)}}^T U_{S_{(i)}} V_{S_{(i)}})\right) + \operatorname{tr}\left(S_{S_{(i)}}^T S_{S_{(i)}}\right) \\
&+ 2\operatorname{tr}\left((S_{S_{(i)}}^T U_{S_{(i)}}) V_{S_{(i)}}\right) + \operatorname{tr}\left(V_{(i)}^T (U^T UV_{(i)})\right) \\
&- 2\operatorname{tr}\left(V_{S_{(i)}}^T (U_{S_{(i)}}^T UV_{(i)})\right) - 2\operatorname{tr}\left((S_{S_{(i)}}^T U) V_{(i)}\right) \Bigg)
\end{aligned}
$$

For the evaluation of the loss function, it can be shown that the complexity is $O(k^2npt)$ if all patients have t time slices. Similarly the complexity of computing the gradient is also given by $O(k^2npt)$. Therefore in the optimization, the computational cost in each iteration is linear with respect to n, p and t. Thus the algorithm is scalable to large data.

2) Solve V. The term $U^T S_{(i)}$ can again be computed efficiently using the similar strategy as above. Recall that in solving $V_{(i)}^+$ we need to perform eigen-decomposition on two matrices: a $\mathbb{R}^{k\times k}$ matrix $U^T U$ and a $\mathbb{R}^{t\times t}$ tridiagonal matrix $R_{(i)} R_{(i)}^T$. The two matrices are equipped with special structures: the matrix $U^T U$ is a low-rank matrix, and the matrix $R_{(i)} R_{(i)}^T$ is a tridiagonal matrix (a very sparse matrix), whose eigen-decomposition can be solved efficiently.

Note that the complexity of time dimension is less critical, because that in most EMR cohorts, the time dimension of the patients are often less than 1000. Recall that the finest time unit of the EMR records is day. Using weekly granularity, 1000 time dimension covers up to 20 years of records. In our implementation we use the

built-in eigen-decomposition of Matlab, which typically takes less than 1 sec for a matrix with a time dimension of 1000 on regular desktop computers.

4.2.5 Latent Dimension Estimation

In the formulations in Eq. (4.2) and Eq. (4.4), we need to estimate the latent dimension of the patient matrices. Indeed, we can choose the latent dimension via validation methods, as done for other regularization parameters. As an alternative, we can use the rank estimation heuristic to adaptively set the latent dimension of the matrices by inspecting the information in the QR decomposition of the latent concept mapping matrix U, assuming that the latent dimension information of all patients is collectively accumulated in U after a few iterations of updates. The idea was originally proposed in Wen *et al.* (2012); Shen *et al.* (2012) to estimate the rank during the matrix completion of a single matrix.

In order to be self-contained we briefly summarize the algorithm as follows. After a specified iterations of updates, we perform the economic QR factorization on $UE = Q_U R_U$, where E is a permutation matrix such that $|\operatorname{diag}(R_U)| := [r_1 \dots r_k]$ is non-increasing after the permutation. Denote $\mathcal{Q}_p = r_p / r_{p+1}$, and $\mathcal{Q}_{\max} = \max(\mathcal{Q}_p)$, and the location is given by $p_{\max}$. We compute the following ratio:

$$\tau = \frac{(K-1)\mathcal{Q}_{\max}}{\sum_{\{p \neq p_{\max}\}} \mathcal{Q}_i}.$$

A large τ indicates a large drop in the magnitude of $\mathcal{Q}_i$ after $p_{\max}$ elements, and we thus reduce the latent factor k to $p_{\max}$, retaining only the first $p_{\max}$ columns of U and the corresponding rows of the evolution matrices $\{V_{(i)}\}$. In our implementation we only perform the estimation once. Empirically as shown in Section 4.3.3, the latent dimension estimation works well when the Pacifier-Sba works, i.e., patients are homogeneous, sharing a few latent concepts.

In the IBA approach the completion of patients are independent. If we apply latent dimension estimation on each patient, then each patient matrix may have a latent dimension different from others. This imposes difficulties when it comes to analyze the patients, and thus the estimation is not used in IBA.

4.3 Empirical Study

In this section we present the experimental results to demonstrate the performance of the proposed PACIFIER methods IBA and SBA. We first provide a toy example showing the parameter sensitivity of the PACIFIER formulations and discuss when to choose one approach over another. We then study the scalability of the proposed algorithm with varying feature dimensions, time dimensions, sample sizes, latent dimensions, and ratios of the observed entries. We then apply the proposed PACIFIER framework on two real clinical cohorts to demonstrate the improvement on predictive performance achieved by our approaches.

4.3.1 A. Toy Example

In this experiment we manually construct two small datasets and explore the properties of the two proposed methods. We generate a dataset called IND of 3 samples where the samples have different basis, and a dataset SHA of 3 samples where the samples share the same basis. First we generate a sparse matrix $B \in \mathbb{R}^{12\times 6}$ such that the ith column only has two non-negative numbers at $2(i-1)+1$ and $2(i-1)+1$. Therefore the locations of non-sparse elements do not overlap among different columns. We constructed the datasets as follows:

Dataset IND. For each patient we choose 2 different columns from matrix B to form $U_{(i)} \in \mathbb{R}^{12\times 2}$, and we generate $V_{(i)} \in \mathbb{R}^{2\times 13}$ such that it simulates temporal smoothness: every row is given by a scaled and randomly shifted sine function $(\sin(5(t+x))+1)/2$,

where $t = 0 : 0.1 : 1.2$ and x is a random number uniformly drawn from $[0, 1]$ for each row to introduce randomness. The full matrix for each sample is then given by $X_{(i)} = U_{(i)}V_{(i)}$. In this way the three samples are spanned by basis orthogonal to each other with different temporal information. The IND dataset simulates heterogeneous cohorts.

Dataset SHA. For all patients we choose the same two columns from B to form $U_{(i)}$ and use the same strategy to form $V_{(i)}$ and compute $X_{(i)}$. Thus all samples can be seen as being generated from the same set of basis vectors with different temporal information. The SHA dataset simulates homogenous cohorts.

For each patient in the above datasets, we randomly select 20% elements as observed (denoting their locations as $\Omega_{(i)}$), leaving the rest as testing elements. We use PACIFIER-SBA and PACIFIER-IBA formulations to recover the 'unseen' elements, and evaluate the performance by the recovery error defined as follows:

$$\sum_{i=1}^{n} \|\mathcal{P}_{\Omega^c_{(i)}}(\hat{X}_{(i)}) - \mathcal{P}_{\Omega^c_{(i)}}(X_{(i)})\|_F^2/n,$$

where $n = 3$ in the toy data. We repeat the above procedure for 20 times on 20 randomly generated IND and SHA datasets. The average recovery error are reported in Figure 4.4.

We see that PACIFIER-IBA performs better than PACIFIER-SBA on IND datasets, while the latter performs better on SHA datasets. These results coincide with the design of the two PACIFIER formulations: the PACIFIER-IBA performs better on the heterogenous dataset IND, whereas the PACIFIER-SBA performs better on the homogeneous dataset SHA. We see that the shared basis assumption improves the performance by better exploiting the sharing information among samples, while in the case that patients share little information, imposing the assumption may degrade the performance.

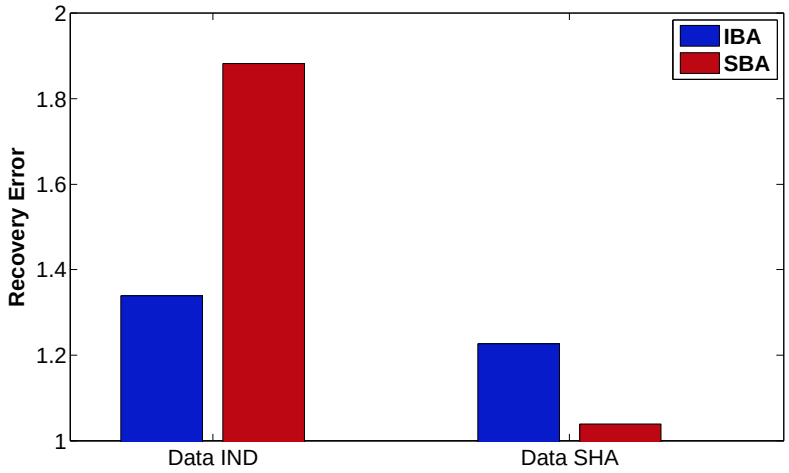
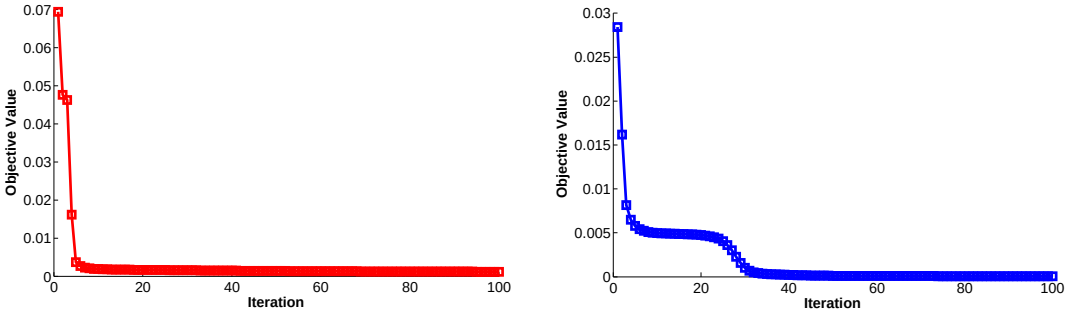


Figure 4.4: The performance of PACIFIER-IBA and PACIFIER-SBA in terms of recovery error on the two toy datasets. We see that the PACIFIER-IBA performs better on the heterogenous dataset IND, whereas the PACIFIER-SBA performs better on the homogeneous dataset SHA.

In Figure 4.5 we present the convergence plots for the two algorithms on one dataset. We see that for both algorithms the objective decreases very fast in the first few iterations, and after that, extra iterations only give a little decrease on the objective value. This is the case for most of our datasets. When the regularizers are not dominant (i.e., the value of the regularizers are comparable to the loss function), then in the later iterations the decrease objective value mainly reflects how closer between the low rank approximation UV and the data X on the observed elements. In this sense if the noise level of the data is relatively high, which is the case for most clinical cohorts, then only a few iterations are needed to obtain a completed matrix that gives very good predictive performance in practice. In the experiment in Section 4.3.3, we observe that the completed dataset using only 20 iterations gives almost the same performance as the one fully converged (relative change of objective values in consecutive iterations is less than $1e-5$).

In the formulations of PACIFIER there are three parameters: sparsity parameter λ_1, ridge-type parameter λ_2 and smoothness parameter on the time dimension λ_3.



(a) IBA-Pacifier (b) SBA-Pacifier

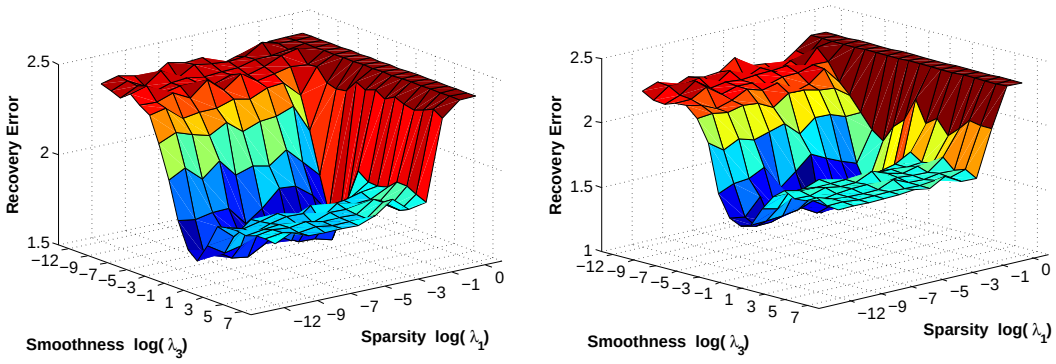
Figure 4.5: Empirical convergence of Pacifier. Note that the objectives of the two methods are different and thus are not comparable.

Note that the parameter λ_2 serves the sole purpose of improving the numerical condition and we fix it to be a small number $1e-8$ in all experiments. In Figure 4.6 we present the average recovery error given different combinations of the sparsity parameter λ_1 and the smoothness parameter λ_2 for Iba and Sba, respectively. We find that 1) a proper amount of smoothness regularization can greatly improve the recovery performance. 2) sparsity regularization can slightly improve the performance, and parameter is generally not sensitive in terms of the recovery performance. However, a moderate sparse regularization gives a more sparse solution of the factor U (not shown in the figure), which may provide better interpretability in practice.

4.3.2 Scalability

In this section we study the scalability of the proposed algorithm using synthetic datasets. In each of the following studies, we generate random datasets with a specified sample n, feature dimension p, average time dimension t, latent dimension k, and observation density $\|\Omega_i\|$. For simplicity we let all samples have the same time dimension. We report the average time cost over 50 iterations. For the two algorithms we set all parameters to be $1e-8$ in all studies.

Sample Size. We fix $p = 100$, $t = 100$, $r = 10$, $\|\Omega_i\| = 0.01$, and vary the sample



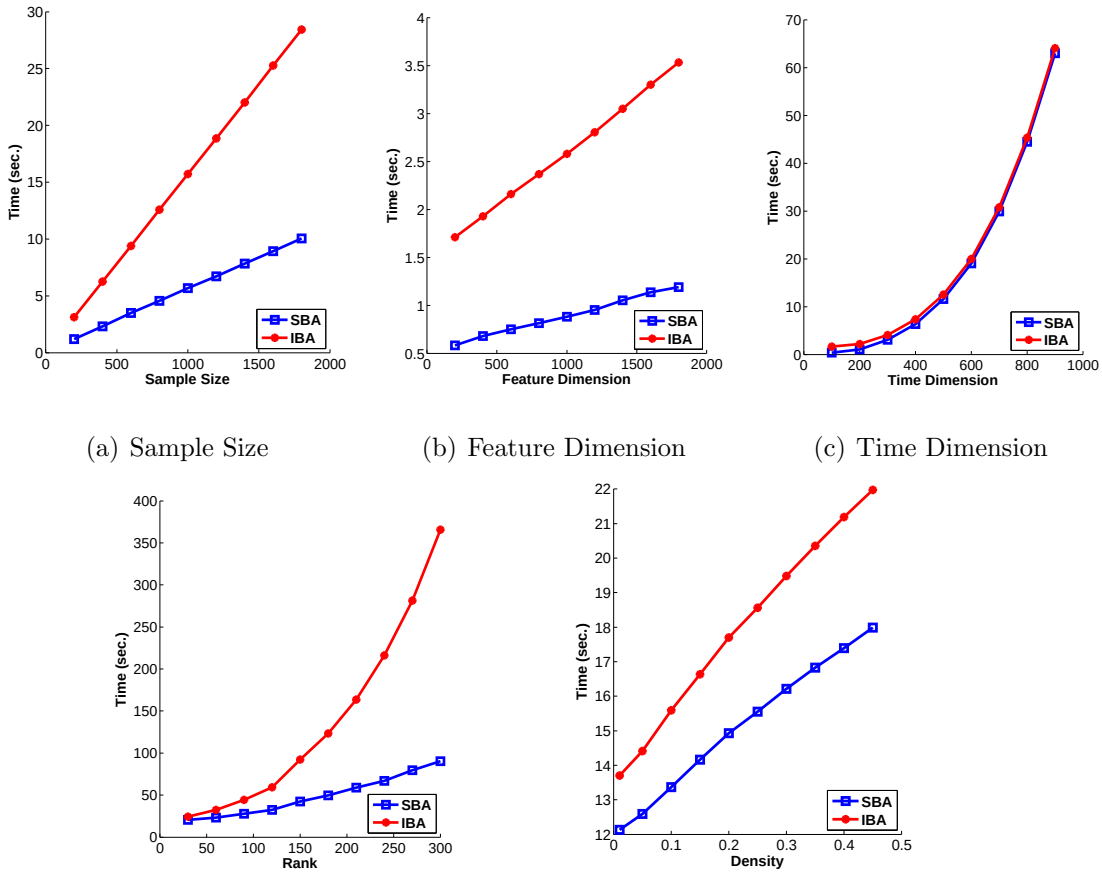
(a) PACIFIER-IBA (b) PACIFIER-SBA

Figure 4.6: Sensitivity study of the sparsity and smoothness parameters of PACIFIER. The smoothness regularization greatly improves the performance in terms of recovery error. The sparsity regularization increases the sparsity of the factor component, while is not sensitive in terms of recovery performance.

size $n = 200 : 200 : 1800$. The results are given in Figure 4.7(a). We observe that for both methods the time costs increase linearly with respect to the sample size. The cost of IBA grows faster than the SBA version, which is expected because in IBA the computation costs of the loss and the gradients are more than those of SBA.

Feature Dimension. We fix $n = 100$, $t = 100$, $r = 10$, use $\|\Omega_i\| = 0.01$, and vary the feature dimension $p = 200 : 200 : 1800$. The results are given in Figure 4.7(b). We see that the time costs for both methods increase linearly with respect to feature dimension, which is consistent with our complexity analysis. The linear complexity of feature dimension is desired in clinical applications, since one might want to use as much information available as possible, resulting in a large feature space.

Time Dimension. We fix $n = 100$, $p = 100$, $r = 10$, $\|\Omega_i\| = 0.01$, and vary the time dimension $t = 100 : 100 : 900$. The results are given in Figure 4.7(c). We find superlinear complexity on the time dimension for both methods, which mainly comes from the eigen decomposition. The complexity on time dimension is less critical in the sense that for most medical records and longitudinal study, the time dimension is very limited. For example, if the time granularity is weekly, then we have 52 time



(a) Sample Size (b) Feature Dimension (c) Time Dimension

(d) Latent Dimension (e) Observed Entries

Figure 4.7: Studies of scalability of PACIFIER-IBA and PACIFIER-SBA. In each study we vary one of the scale factors while fix other factors, and record the time costs. Both methods have the same complexity: linear with respect to samples size and feature dimension; superlinear with respect to time dimension and latent dimension; sublinear with respect to the number of observed entries.

dimensions each year. If 20-year records are available for one patient, then it yields only 1040 time dimensions. Besides, the eigen decomposition can be implemented in the way that utilizes the extreme sparsity of the temporal smoothness coupling matrix.

Latent Dimension. We fix $n = 100$, $p = 500$, $t = 500$, $\|\Omega_i\| = 0.01$, and vary the latent dimension input of the algorithms $r = 20 : 20 : 160$. The results are given in Figure 4.7(d). We find that the time costs increase superlinearly with respect to

latent dimension for both methods, and the complexity of SBA is close to be linear.

Observed Entries. We fix $n = 100$, $p = 1000$, $t = 500$, $r = 10$, and vary the percentage of the observed entries $\|\Omega_i\| = 0.05 : 0.05 : 0.45$. The results are given in Figure 4.7(d). We see that the time costs increase only sub-linearly with respect to the set of observed entries.

We note that the complexity of PACIFIER-IBA is of the same order as that of SBA. The difference between the two methods comes from the computation of the objective value and gradient in the U step. It is obvious that the IBA methods can be parallelized because the computation of all samples are decoupled. Similarly, the major computational complexity of SBA comes from the computation of U in the optimization and eigen-decomposition of $V_{(i)}$, which can also be parallelized by segmenting the computation of each patient.

4.3.3 Predictive Performance on Real Clinical Cohorts

To gauge the performance of the proposed PACIFIER framework we apply the two formulations on two real EMR cohorts from one of our clinical partners. In one cohort we study the predictive modeling of congestive heart failure (CHF), and in the other cohort we study end stage renal disease (ESRD). In both EMR cohorts we are given a set of patients associated with their outpatient diagnosis events in ICD9 codes and the corresponding timestamps. In our experiments we use the prediction windows lengths suggested by physicians (180 days for CHF and 90 days for ESRD), and we remove all events within the prediction window before the operation criteria date.

To construct the longitudinal patient matrices to be imputed, we use EMR data at the weekly granularity as discussed in Section 4.1. We select the patients with more than 100 events. Note that we are working on a large feature dimension, and thus for a patient with 100 EMR events the longitudinal patient matrix is still extremely

sparse. Note that in our cohorts the number of case patients is much smaller than control patients, which is very common in most clinical studies. To avoid the effects of biased samples, we perform random under-sampling on the control patients so that we have the equal number of case and control patients in our datasets. To this end, we have constructed two datasets: 1) CHF dataset with 249 patients in each class; 2) ESRD dataset with 187 patients in each class.

The raw feature space in the low-level ICD9 codes is 14313. Because the matrix constructed using the low-level ICD9 codes is too sparse, we retrieve the medical features at coarser granularities. In order to study the effects of features at different granularities, we compare the medical features at ICD9 diagnosis group level (DxGroup) and HCC level. At DxGroup level there are 1368 features and at HCC level there are 252 features. In the two studies we consider the following commonly-used baselines methods:

- Zero Imputation (RAW). An intuitive way to impute missing values, which is equivalent to mean value imputation when the data set is first normalized (zero mean and unit standard deviation). This method is standard in the current medical literature for clinical studies Sun *et al.* (2012); Wang *et al.* (2012); Wu *et al.* (2010).

- Row Average (AVG). In this baseline approach we fill the missing value using the average value of the observed values of the feature over time.

- Interpolation (INT) Engels and Diehr (2003). We use the next observation and previous observation along the timeline to interpolate the missing elements.

- Next Observation Carry Backward (NOCB) Engels and Diehr (2003). Missing values are filled using the next observation of this medical feature alone the timeline.

- Last Observation Carry Forward (LOCF) Engels and Diehr (2003). Missing values are filled using the previous observation of this medical feature along the timeline.

We compare the baseline methods with the following competing methods:

- Individual Basis Pacifier (Iba). Each patient is densified using Algorithm 2.
- Iba without temporal smoothness (Iba-NT). This variant of Pacifier-IBA sets the temporal regularization λ_3 to 0.
- Shared Basis Pacifier (Sba) using Algorithm 2.
- Sba without temporal smoothness (SbaNT). This variant of Pacifier-Sba sets the temporal regularization λ_3 to 0.
- Sba with Latent Dimension Estimation (Sba-E). The latent dimension estimation is described in Section 4.2.5, and only used once during the algorithm.
- Sba without Temporal Smoothness and with Latent Dimension Estimation (SbaNT-E). This variant of Pacifier-Sba sets the temporal regularization λ_3 to 0 and uses latent dimension estimation once.

Note that for the extremely sparse matrix as the clinical data in our studies, classical imputation methods such as those based on k-nearest neighbor Hastie *et al.* (1999) and expectation maximization Schneider (2001) do not work. The methods IbaNT and SbaNT are included in the study to explore the effectiveness of the proposed temporal smoothness. For the parameter estimation we have separated an independent set of samples for validation, and we select the parameters that give the lowest recovery error on the validation set. In Iba, Sba, and SbaNT, the latent dimension k is also determined via the validation set.

We finally test the predictive performance on the completed datasets using sparse logistic regression classifier (we use the SLEP implementation Liu *et al.* (2009d)). From the completed datasets, we derive features by averaging the features along the time dimension within a given observation window (52 weeks). To this end, each patient is represented as a vector of the same dimension as the feature dimension. We then randomly split the samples into 90% training and 10% testing, and train the classifier on the training data. The classifier parameter is tuned using standard 10 fold cross validation. We repeat the random splitting for 20 iterations, and report the average performance over all iterations. In order to be comparable, the splitting is the same for all methods in each iteration.

CHF Cohort. The predictive performance of competing methods is presented in Table 4.1. We find that in the CHF cohort: 1) most of the proposed Pacifier approaches and their variants significantly improve the predictive performance as compared to the baseline RAW approach. The best AUC obtained by Pacifier-IBA dataset is 0.816 while the baseline is only 0.689 (a gain of 0.127); 2) the individual basis approaches outperform shared based ones; 3) temporal regularization significantly improves the predictive performance for all methods; 4) the methods with latent dimension estimation perform worse than those that do not use latent dimension estimation on this cohorts; 5) the features at DxGroup level outperform HCC level, which might be due to that in this predictive task, a fine granularity is likely to maintain more predictive information, than a coarse one.

ESRD Cohort. The predictive performance on ESRD cohort is given in Table 4.2. For the DxGroup features we observe similar patterns that is, Iba outperforms all other methods, which achieves an AUC of 0.828, compared to the baseline RAW method that achieves 0.756 (a gain of 0.072). The variants with temporal smoothness perform much better than the ones without temporal smoothness. For the HCC

features we see that: 1) the shared basis approaches perform as well as the independent basis, where SBA-E achieves an AUC of 0.827. 2) again the temporal smoothness significantly improves the performance. 3) latent dimension estimation works well and outperforms the ones without latent dimension estimation.

As a summary, the experimental results have demonstrated the effectiveness of the proposed methods on real clinical data, and the temporal smoothness regularization brings significant improvements on predictive performance. In real clinical data, the samples tend to be heterogenous and therefore the independent basis approaches perform better. However, using the HCC features of the two datasets, shared basis approaches perform better than using the DxGroup features. One potential explanation is that, using HCC features where the features space is smaller and features themselves are coarser (in terms of clinical concepts), the patients tend to be more homogeneous. We also notice that the latent dimension estimation only works well when shared basis works well. Recall that the idea of latent dimension estimation is to detect the jumps in the diagonal elements from the R_U factor of QR factorization. This is expected because if the patients are homogeneous and share only a few basis, then obviously there are such natural jumps.

4.3.4 Marco Phenotypes Learnt from Data

In this section we show some meaningful medical concepts learnt by the proposed PACIFIER-SBA method. In the latent medical concept mapping matrix U, we are able to obtain feature groups from data, because of the sparsity on the matrix. We first normalize weights of the columns such that the sum of each column is equal to 1. The normalized weights indicate the percentages of medical features contributing to the medical concept. We rank the medical features according to their contributions and find that in most of the medical concepts the top medical features are typically

related and are comorbidities of a certain disease. In Figure 4.3, we show a list of medical concepts obtained from our CHF cohort. For example, in the first medical concept, the highly ranked diagnosis groups are all related to *Cardiovascular Disease*, e.g., Heart failure (428), Hypertension (401) and Dysrhythmias (427), and the second medical concepts include features that are typical related to *Diabetes* and its related comorbidities such as Hypertension (401), Chronic renal failure (585). In the CHF cohort, we have also found very similar medical concepts.

Table 4.3: Medical concepts discovered by the PACIFIER-SBA in our CHF cohort. In each medical concept, we firstly normalize the weights of the medical features in the medical concepts learnt and rank the features. For each medical concept we list top 10 medical features and their diagnosis group codes (DxGrp). We observe that the medical features in one medical concept are usually related to a certain type of disease.

Weight	**DxGrp**	**Description**
Medical Concept: Cardiovascular Diseases		
0.164	428	Heart failure
0.121	401	Essential hypertension
0.113	427	Cardiac dysrhythmias
0.108	780	General sympt.
0.141	414	Other form of chronic ischemic heart disease
0.053	785	Symp. inv. cardiovascular sys.
Continued on next page		

Table 4.3 – continued from previous page

Weight	DxGrp	Description
0.052	786	Symp. inv. respir. sys. and other chest sympt.
0.046	402	Hypertensive heart disease
0.042	272	Diso. of lipoid metabolism
Medical Concept: Diabetes		
0.211	250	Diabetes mellitus
0.129	272	Diso. of lipoid metabolism
0.115	278	Obesity and other hyperalign.
0.095	593	Other diso. of kidney and ureter
0.093	585	Chronic renal failure
0.068	599	Other diso. of urethra and urinary tract
0.065	790	Nonspe. find on exam of blood
0.058	401	Essential hypertension
0.023	366	Cataract
0.019	285	Other and unspecified anemias
Medical Concept: Lung Diseases		
0.117	518	Other diseases of lung
0.112	496	Chronic airways obstruction
0.110	786	Sympt. inv. respir. sys. and other chest symp.
0.098	V72	Special investigations and exam
0.089	493	Asthma
0.087	599	Other diso. of urethra and urinary tract
0.086	466	Acute bronchitis and bronch.

Continued on next page

Table 4.3 – continued from previous page

Weight	DxGrp	Description
0.078	780	General symp.
0.067	787	Symp. inv. digestive sys.
0.057	793	Nonspec. ab. find on radio. and other exam of body structure
Medical Concept: Osteoarthrosis		
0.185	729	Other diso. of soft tissues
0.123	715	Osteoarthrosis and allied diso.
0.120	726	Peripheral enthesopathies and allied syndr.
0.118	401	Essential hypertension
0.082	733	Other diso. of bone and cartilage
0.081	366	Cataract
0.069	719	Other and unspec. diso. of joint
0.066	272	Diso. of lipoid metabolism
0.065	780	General symp.
0.008	244	Acquired hypothyroidism
Medical Concept: Disorder of joints and softtissues		
0.103	719	Other and unspec. diso. of joint
0.096	729	Other diso. of soft tissues
0.081	789	Other symp. involving abdomen and pelvis
0.078	722	Intervertebral disc diso.
0.058	724	Other and unspec. diso. of back
0.056	780	General symp.

Continued on next page

Table 4.3 – continued from previous page

Weight	DxGrp	Description
0.055	721	Spondylosis and allied diso.
0.053	728	Diso. of muscle, ligament, and fascia
0.048	733	Other diso. of bone and cartilage
0.048	723	Other diso. of cervical region

4.4 Related Works and Discussion

In this paper we treat the zeros in the longitudinal patient matrices as missing values, and proposed a novel framework PACIFIER to perform temporal matrix completion via low-rank factorization. To the best of our knowledge, there are no prior work that applies matrix completion techniques to solve the data sparsity in EMR data. The proposed PACIFIER framework aims at densifying the extremely sparse EMR data by performing factorization based matrix completion. The differences between the proposed completion method and existing works are that: instead of treating each patient as vectors and forming a single matrix, we treat each patient as a matrix with missing entries and consider a set of related matrix completion problems. We further propose to incorporate the temporal smoothness in the matrix completion to utilize the hidden temporal information of each patient.

The problem of imputation via matrix completion problem is one of the hottest topics in data mining and machine learning. In many areas such as information retrieval and social network, the data matrix is so sparse that classical imputation methods does not work well. The basic problem setting of the matrix completion is to recover the unknown data from only a few observed entries, imposing certain types

of assumptions on the matrix to be recovered. The most popular assumption is to assume that the matrix has a low rank structure Cai *et al.* (2010); Mazumder *et al.* (2010); Wen *et al.* (2012); Xiong *et al.* (2010). There are two types of matrix completion in terms of the assumption on the observed entries: The first type assumes that the observation has no noise, and the goal is to find a low rank matrix whose values at the observed locations are exactly the same as the given ones Cai *et al.* (2010); Candès and Recht (2009); Meka *et al.* (2010). In real world applications, however, noise is ubiquitous and thus the rigid constraint on the observed locations may result in overfitting. In contrast, the noisy matrix completion methods only require the values at the observed locations to be close to the given data Mazumder *et al.* (2010); Wen *et al.* (2012). Directly dealing with the rank function in objectives are shown to be NP-Hard. Therefore many approaches seek to use the trace norm which is the convex envelope of the rank function Cai *et al.* (2010); Meka *et al.* (2010); Mazumder *et al.* (2010). Most of these approaches, however, require singular value decomposition (SVD) on large matrices, the complexity of which is prohibitive for large scale problems. Recent years have witnessed surging interests on the local search methods, which seek a local solution with extremely efficient algorithms Salakhutdinov and Mnih (2008); Wen *et al.* (2012). The PACIFIER framework is among these efficient local approaches, which does not require SVD and can be applied to solving large scale problems.

The completed data for each patient has the factorization form of $X_{(i)} = U_{(i)}V_{(i)}$, and for SBA all patients have the same $U_{(i)}$. Clearly, one advantage of SBA is that we have simultaneously learned a shared low-dimensional feature space for all patients, and their coordinates that can be used as new (and reduced) features. To see this, let $U = Q_U R_U$ be the QR factorization of U, then for each patient we have that $X_{(i)} = UV_{(i)} = Q_U(R_U V_{(i)})$, indicating that rows of $(RV_{(i)})$ can be considered as

coordinates on the low dimensional space whose bases are given by columns of Q_U. One issue brought by the shared mapping is that the latent dimension is limited by the lowest time dimension of the patient, i.e., $\min_i t_i > k$. One solution is that we can extend the time dimension of the patients with non-informative time dimensions of all zeros.

We have shown in the experiments that a shared concept mapping works better on homogeneous samples while individual mappings work better on heterogenous samples. In reality the samples may form some groups such that within the groups the patients are homogeneous and patients from different groups may be heterogenous. The degree of homogeneous/heterogenous is also affected by feature granularity as shown in our real clinical experiments, where in finer feature level the patients appear to be more heterogeneous. It is thus interesting to explore how to simultaneously identify feature groups and patient groups to further improve the quality of matrix completion:

$$\min_{\mathcal{G},\{S_i,U_j,V_i\}} \frac{1}{g} \sum_{j=1}^{g} \frac{1}{|\mathcal{G}_j|} \sum_{i \in \mathcal{G}_j} \|S_i - U_j V_i\|_F^2 + \mathcal{R}(\{U\},\{V\})$$

where $\mathcal{G}$ is the patient group assignment matrix, and patients within each group $\mathcal{G}_j$ share the same basis U_j. To do so, we can incorporate group learning into the objective as done in Zhou *et al.* (2011a). We leave this interesting study to our future works. One final note – the proposed PACIFIER framework proposed in this paper is not limited to healthcare domain, they can also be applied to temporal collaborative filtering Koren (2009); Lu *et al.* (2009); Xiong *et al.* (2010), where each user has a rating preference that changes overtime.

Table 4.1: Predictive performance on the CHF cohort using DxGroup and HCC features.

DxGroup Features			
Method	AUC	Sensitivity	Specificity
RAW	0.689 ± 0.058	0.747 ± 0.046	0.528 ± 0.115
AVG	0.671 ± 0.051	0.744 ± 0.064	0.482 ± 0.083
INT	0.644 ± 0.066	0.803 ± 0.062	0.468 ± 0.110
NOCB	0.658 ± 0.048	0.845 ± 0.073	0.443 ± 0.096
LOCF	0.689 ± 0.055	0.866 ± 0.082	0.456 ± 0.087
IBA	$\mathbf{0.816 \pm 0.040}$	$\mathbf{0.843 \pm 0.054}$	$\mathbf{0.657 \pm 0.078}$
IBANT	0.754 ± 0.056	0.762 ± 0.089	0.597 ± 0.097
SBA	0.750 ± 0.062	0.776 ± 0.067	0.640 ± 0.106
SBANT	0.706 ± 0.054	0.672 ± 0.079	0.631 ± 0.066
SBA-E	0.730 ± 0.064	0.695 ± 0.074	0.653 ± 0.095
SBANT-E	0.661 ± 0.073	0.678 ± 0.090	0.588 ± 0.095
HCC Features			
Method	AUC	Sensitivity	Specificity
RAW	0.645 ± 0.089	0.672 ± 0.086	0.529 ± 0.072
AVG	0.660 ± 0.053	0.683 ± 0.063	0.526 ± 0.089
INT	0.596 ± 0.072	0.768 ± 0.093	0.489 ± 0.082
NOCB	0.602 ± 0.081	0.694 ± 0.088	0.511 ± 0.093
LOCF	0.625 ± 0.067	0.852 ± 0.079	0.480 ± 0.083
IBA	$\mathbf{0.755 \pm 0.071}$	0.747 ± 0.085	$\mathbf{0.641 \pm 0.084}$
IBANT	0.727 ± 0.060	0.740 ± 0.087	0.614 ± 0.070
SBA	0.736 ± 0.066	$\mathbf{0.753 \pm 0.089}$	0.629 ± 0.074
SBANT	0.645 ± 0.070	0.686 ± 0.087	0.550 ± 0.095
SBA-E	0.702 ± 0.079	0.688 ± 0.106	0.616 ± 0.067
SBANT-E	0.669 ± 0.062	0.702 ± 0.082	0.538 ± 0.079

Table 4.2: Predictive performance on the ESRD cohort with DxGroup and HCC features.

DxGroup Features			
Method	AUC	Sensitivity	Specificity
RAW	0.756 ± 0.086	0.831 ± 0.113	0.581 ± 0.077
AVG	0.775 ± 0.079	0.821 ± 0.093	0.592 ± 0.084
INT	0.747 ± 0.083	0.919 ± 0.104	0.568 ± 0.110
NOCB	0.766 ± 0.092	0.914 ± 0.099	0.556 ± 0.103
LOCF	0.787 ± 0.085	0.958 ± 0.107	0.577 ± 0.079
IBA	$\mathbf{0.838 \pm 0.072}$	$\mathbf{0.842 \pm 0.099}$	0.658 ± 0.106
IBANT	0.796 ± 0.066	0.806 ± 0.101	0.600 ± 0.095
SBA	0.811 ± 0.065	0.769 ± 0.091	$\mathbf{0.722 \pm 0.097}$
SBANT	0.763 ± 0.068	0.719 ± 0.109	0.697 ± 0.075
SBA-E	0.803 ± 0.056	0.753 ± 0.098	0.681 ± 0.090
SBANT-E	0.770 ± 0.082	0.689 ± 0.099	0.700 ± 0.110
HCC Features			
Method	AUC	Sensitivity	Specificity
RAW	0.758 ± 0.058	0.747 ± 0.085	0.656 ± 0.093
AVG	0.778 ± 0.055	0.789 ± 0.088	0.660 ± 0.088
INT	0.729 ± 0.067	0.752 ± 0.091	0.652 ± 0.094
NOCB	0.752 ± 0.079	0.775 ± 0.089	0.658 ± 0.095
LOCF	0.771 ± 0.068	0.808 ± 0.082	0.665 ± 0.081
IBA	0.826 ± 0.051	0.800 ± 0.085	0.708 ± 0.080
IBANT	0.802 ± 0.064	0.775 ± 0.094	0.714 ± 0.089
SBA	0.820 ± 0.064	0.789 ± 0.091	$\mathbf{0.722 \pm 0.092}$
SBANT	0.771 ± 0.082	0.733 ± 0.084	0.681 ± 0.102
SBA-E	$\mathbf{0.827 \pm 0.067}$	$\mathbf{0.814 \pm 0.077}$	0.706 ± 0.096
SBANT-E	0.785 ± 0.060	0.736 ± 0.065	0.717 ± 0.092

Chapter 5

CONCLUSION AND OUTLOOK

In this chapter, I summarize the major contributions made in this thesis and discuss possible future directions.

5.1 Summary of Contributions

The major theme of this thesis is to demonstrate that how multi-task learning can help scientific discoveries in the biomedical field. I present a formulation for the clustered multi-task learning and show that the clustered multi-task learning is equivalent to another type of multi-task learning that seeks a shared subspace among the tasks. The finding has provided significant insights into the nature of the two types of multi-task learning approaches and important implication on computational efficiency of the multi-task learning. In the area of Alzheimer's disease research, I have designed effective multi-task learning approaches to model the disease progression, which lead to improved prognosis and diagnosis of the Alzheimer's disease. In the area of biomedical informatics, I have designed novel multi-task matrix completion methods to learn marco-phenotypes from the patients' partially observed electronic medical records (EMR). These methods complete the EMR records during the learning of the marco-phenotypes, and based on which we can build predictive models with significantly improved the predictive performance.

In many traditional multi-task learning formulations, it is often assumes that all tasks are related. However, this assumption is too strong and may not work well in real applications. In the thesis, I consider the clustered multi-task learning problem (CMTL) where the tasks form clusters and within each cluster the tasks are closely

related to each other. I formulate the CMTL and establish the equivalent relationship between clustered multi-task learning to the well known alternating structure optimization (ASO), which is a multi-task learning approach that learns a low- dimensional common subspace shared among tasks. I further provide a convex relaxation to the CMTL formulation and establish the equivalent relationship between the relaxed CMTL and the convex relaxation of ASO. Recall that the CMTL performs clustering on tasks, and on the other hand the ASO seeks dimension reduction, or, feature clustering. Establishing the equivalence between CMTL and ASO has significant practical implications: feature clustering in multi-task learning can be considered to be same as the task clustering. Thus when there are more features than tasks, ASO can be used to solve the problem efficiently and otherwise we can choose CMTL instead. In addition, I provide three different optimization schemes for the convex CMTL problem and show efficiencies of the schemes.

In the area of prognosis and diagnosis of the Alzheimer's disease, I propose a novel multi-task learning framework for disease progression modeling. Traditionally, in order to predict the disease status of a patient, one builds regression models to predict certain measurements/scores in the future. In the framework I propose to learn a set of regression models together, in which I build one regression model at each time point. Since the regression models at different time points are collectively learned, I design algorithms to model the temporal relations among the tasks. In the first model I propose to use a $\ell_{2,1}$-norm regularization to joint select features among the tasks, and a ℓ_1 fuse term is added to enforce similar weights for tasks that are temporally next to each other. In my experimental results, the model can perform much better than the single task learning approaches where the temporal relationships are not modeled explicitly. In the second model I propose to use the fused Lasso term to model temporal smoothness, in addition to the joint feature selection using the

$\ell_{2,1}$-norm regularization. The advantage of using the fused Lasso is that we allow the models different time points to have different set of features, however, we ask the feature selected to be similar when tasks are close to each other temporally. I perform extensive experiments on the Alzheimer's Disease Neuroimaging Initiative (ADNI) datasets and demonstrate the effectiveness of the proposed models in predicting future cognitive scores of the Alzheimer's disease patients.

In the predictive modeling using electronic medical records (EMR), the quality of the the predictive models are heavily depend on the quality of the raw EMR data, which is often sparse, noisy and with tons of missing information. As in each office visit/hospitalization, the patient only carry out some certain panels or imaging studies that targets to certain diseases. As such, directly building predictive models from raw EMR may not be optimal. Therefore, I propose a framework that densifies the EMR records based on the observed information, and predictive models are built based on the densified data. In order to densify the EMR records, I design algorithms to leverage the temporal information within the EMR records, and consider a matrix representation for each patient, which describes how medical measurements/features evolve over time. To this end, for each patient I formulate the densification as a matrix completion task. Furthermore, the completion tasks can be done simultaneously so that by completing the matrix of one patient, we can transfer knowledge from other patients with similar medical conditions. During the densification, the formulation can also learn marco phenotypes from the patients, which are high level medical concepts consist of groups of fine-grained medical features. For the two propose densification algorithms, I designed efficient block coordinate descent algorithm that can handle large scale input data. I perform extensive experiments on real-world data from hospitals, and the proposed method can significantly improve the performance of the predictive modeling using EMR data.

5.2 Future Directions

Non-convex multi-task learning models. Currently most of multi-task learning algorithms are formulated by convex optimization problems. Indeed, the convex has many merits such as global optimal and in many situations lead to simple optimization problems. Especially, cardinality optimization problems are computationally intractable, and convex relaxations such as ℓ_1-norm and trace norm provide feasible algorithms with theoretical guarantees. However, from the perspective of performance of original formulations, the convex relaxation solutions may not be good enough. One future direction of multi-task learning is to explore how non-convex formulation can benefit. For example, in the low-rank modeling formulation, we can consider a function that is closer to the original rank function than the trace norm.

Distributed algorithms for multi-task learning. In many real-world applications the training data of the learning tasks is so large that it cannot be stored in single computer. In the case that data is stored in multiple computers, we need scale our learning algorithms into the distributed environment. However, currently many algorithms in multi-task learning involves complex loss functions and/or regularization terms that are necessary to model the task relatedness, and thus cannot be easily extended to distributed environment. For example, the low-rank assumption requires an iterative algorithm involving singular projection, and thus in every iteration we need transfer the model in a central node to perform singular thresholding, which may cause huge network transfer overhead. Thus, in order to leverage the benefit of multi-task learning in big data era, one future direction is to design MTL algorithm that can be efficiently computed in the distributed environment.

Multi-task learning with asymmetric task relationship. In most current multi-task learning frameworks, tasks are treated equally and their relationship is symmetric.

For example, learning a common subspace or shared set of features. However, in real world scenarios, the task relationship should asymmetric, i.e., the learning one task may benefit more from a certain task while less from other tasks. As such, each task can control how much to learn from other tasks. One future direction of the multi-task learning is to design formulations that allow task relationship to be asymmetric.

REFERENCES

Agarwal, A., H. D. Iii and S. Gerber, "Learning multiple tasks using manifold regularization", in "Advances in neural information processing systems", pp. 46–54 (2010).

Amit, Y., M. Fink, N. Srebro and S. Ullman, "Uncovering shared structures in multiclass classification", in "Proceedings of the 24th international conference on Machine learning", pp. 17–24 (ACM, 2007).

An, Q., C. Wang, I. Shterev, E. Wang, L. Carin and D. B. Dunson, "Hierarchical kernel stick-breaking process for multi-task image analysis", in "Proceedings of the 25th international conference on Machine learning", pp. 17–24 (ACM, 2008).

Ando, R. and T. Zhang, "A framework for learning predictive structures from multiple tasks and unlabeled data", The Journal of Machine Learning Research **6**, 1817–1853 (2005).

Apostolova, L. *et al.*, "3D mapping of mini-mental state examination performance in clinical and preclinical Alzheimer disease", Alzheimer Disease & Associated Disorders **20**, 4, 224 (2006).

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

Argyriou, A., A. Maurer and M. Pontil, "An algorithm for transfer learning in a heterogeneous environment", in "Proceedings of the 2008 European Conference on Machine Learning and Knowledge Discovery in Databases-Part I", pp. 71–85 (Springer-Verlag, 2008b).

Argyriou, A., A. Maurer and M. Pontil, "An algorithm for transfer learning in a heterogeneous environment", in "Machine Learning and Knowledge Discovery in Databases", pp. 71–85 (Springer, 2008c).

Argyriou, A., C. Micchelli, M. Pontil and Y. Ying, "A spectral regularization framework for multi-task structure learning", Advances in Neural Information Processing Systems **20**, 25–32 (2008d).

Argyriou, A., C. Micchelli, M. Pontil and Y. Ying, "A spectral regularization framework for multi-task structure learning", NIPS pp. 25–32 (2008e).

Arora, N., G. M. Allenby and J. L. Ginter, "A hierarchical bayes model of primary and secondary demand", Marketing Science **17**, 1, 29–44 (1998).

Ashford, J. and F. Schmitt, "Modeling the time-course of Alzheimer dementia", Current Psychiatry Reports **3**, 1, 20–28 (2001).

Association, A., "2010 alzheimer's disease facts and figures", Alzheimer's & Dementia **6**, 158–194 (2010).

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

Balasubramanian, K., K. Yu and T. Zhang, "High-dimensional joint sparsity random effects model for multi-task learning", arXiv preprint arXiv:1309.6814 (2013).

Baxter, J., "A bayesian/information theoretic model of learning to learn via multiple task sampling", Machine Learning **28**, 1, 7–39 (1997).

Baxter, J., "A model of inductive bias learning", J. Artif. Intell. Res. **12**, 149–198 (2000).

Blacher, J., A. P. Guerin, B. Pannier, S. J. Marchais and G. M. London, "Arterial calcifications, arterial stiffness, and cardiovascular risk in end-stage renal disease", Hypertension **38**, 4, 938–942 (2001).

Bonilla, E. V., F. V. Agakov and C. Williams, "Kernel multi-task learning using task-specific features", in "International Conference on Artificial Intelligence and Statistics", pp. 43–50 (2007).

Cai, J.-F., E. J. Candès and Z. Shen, "A singular value thresholding algorithm for matrix completion", SIAM J. on Opt. **20**, 4, 1956–1982 (2010).

Candès, E. and B. Recht, "Exact matrix completion via convex optimization", Foundations of Computational Mathematics **9**, 6, 717–772 (2009).

Caroli, A., G. Frisoni *et al.*, "The dynamics of alzheimer's disease biomarkers in the alzheimer's disease neuroimaging initiative cohort", Neurobiology of aging **31**, 8, 1263–1274 (2010).

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

Chai, K. M., "Generalization errors and learning curves for regression with multi-task gaussian processes", in "Advances in neural information processing systems", pp. 279–287 (2009).

Chai, K. M., "Multi-task learning with gaussian processes", Dissertation (2010).

Chang, S., G.-J. Qi, J. Tang, Q. Tian, Y. Rui and T. S. Huang, "Multimedia lego: Learning structured model by probabilistic logic ontology tree", in "Data Mining (ICDM), 2013 IEEE 13th International Conference on", pp. 979–984 (IEEE, 2013a).

Chang, S., M.-H. Tsai and T. S. Huang, "Clustering multimedia data", Data Clustering: Algorithms and Applications p. 339 (2013b).

Chen, J., J. Liu and J. Ye, "Learning incoherent sparse and low-rank patterns from multiple tasks", in "Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining", pp. 1179–1188 (ACM, 2010a).

Chen, J., J. Liu and J. Ye, "Learning incoherent sparse and low-rank patterns from multiple tasks", ACM Transactions on Knowledge Discovery from Data (TKDD) **5**, 4, 22 (2012a).

Chen, J., L. Tang, J. Liu and J. Ye, "A convex formulation for learning shared structures from multiple tasks", in "Proceedings of the 26th Annual International Conference on Machine Learning", pp. 137–144 (ACM, 2009).

Chen, J. and J. Ye, "Sparse trace norm regularization", Computational Statistics (2013).

Chen, J., J. Zhou and J. Ye, "Integrating low-rank and group-sparse structures for robust multi-task learning", in "Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining", pp. 42–50 (ACM, 2011).

Chen, X., S. Kim, Q. Lin, J. G. Carbonell and E. P. Xing, "Graph-structured multi-task regression and an efficient optimization method for general fused lasso", arXiv preprint arXiv:1005.3579 (2010b).

Chen, X., X. Shi, X. Xu, Z. Wang, R. Mills, C. Lee and J. Xu, "A two-graph guided multi-task lasso approach for eqtl mapping", in "International Conference on Artificial Intelligence and Statistics", pp. 208–217 (2012b).

Chetelat, G. and J. Baron, "Early diagnosis of Alzheimer's disease: contribution of structural neuroimaging", Neuroimage **18**, 2, 525–541 (2003).

Collobert, R. and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning", in "Proceedings of the 25th international conference on Machine learning", pp. 160–167 (ACM, 2008).

Daumé III, H., "Frustratingly easy domain adaptation", in "ACL", vol. 1785, p. 1787 (2007).

Daumé III, H., "Bayesian multitask learning with latent hierarchies", in "Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence", pp. 135–142 (AUAI Press, 2009).

Davatzikos, C., F. Xu, Y. An, Y. Fan and S. Resnick, "Longitudinal progression of Alzheimer's-like patterns of atrophy in normal older adults: the SPARE-AD index", Brain **132**, 8, 2026 (2009).

Ding, C. and X. He, "K-means clustering via principal component analysis", in "Proceedings of the twenty-first International Conference on Machine learning", p. 29 (ACM, 2004).

Dubois, B. *et al.*, "Research criteria for the diagnosis of Alzheimer's disease: revising the NINCDS-ADRDA criteria", The Lancet Neurology **6**, 8, 734–746 (2007).

Duchesne, S., A. Caroli, C. Geroldi, D. Collins and G. Frisoni, "Relating one-year cognitive change in mild cognitive impairment to baseline MRI features", NeuroImage **47**, 4, 1363–1370 (2009).

Eleftherohorinou, H., C. Hoggart, V. Wright, M. Levin and L. Coin, "Pathway-driven gene stability selection of two rheumatoid arthritis gwas identifies and validates new susceptibility genes in receptor mediated signalling pathways", Human molecular genetics **20**, 17, 3494–3506 (2011).

Engels, J. M. and P. Diehr, "Imputation of missing longitudinal data: a comparison of methods", Journal of clinical epidemiology **56**, 10, 968–976 (2003).

Evgeniou, A. and M. Pontil, "Multi-task feature learning", in "Advances in neural information processing systems: Proceedings of the 2006 conference", vol. 19, p. 41 (The MIT Press, 2007).

Evgeniou, T., C. A. Micchelli and M. Pontil, "Learning multiple tasks with kernel methods", in "Journal of Machine Learning Research", pp. 615–637 (2005).

Evgeniou, T. and M. Pontil, "Regularized multi–task learning", in "Proceedings of the tenth ACM SIGKDD International Conference on Knowledge discovery and data mining", pp. 109–117 (ACM, 2004).

Fan, K., "On a theorem of Weyl concerning eigenvalues of linear transformations I", Proceedings of the National Academy of Sciences of the United States of America **35**, 11, 652 (1949).

Fei, H. and J. Huan, "Structured feature selection and task relationship inference for multi-task learning", in "Data Mining (ICDM), 2011 IEEE 11th International Conference on", pp. 171–180 (IEEE, 2011).

Feldman, S., M. Gupta and B. Frigyik, "Multi-task averaging", in "Advances in Neural Information Processing Systems 25", pp. 1178–1186 (2012).

Ferrarini, L. *et al.*, "MMSE scores correlate with local ventricular enlargement in the spectrum from cognitively normal to Alzheimer disease", Neuroimage **39**, 4, 1832–1838 (2008).

Fonarow, G. C., K. F. Adams Jr, W. T. Abraham, C. W. Yancy, W. J. Boscardin *et al.*, "Risk stratification for in-hospital mortality in acutely decompensated heart failure", JAMA **293**, 5, 572–580 (2005).

Friedman, J., T. Hastie, H. Höfling, R. Tibshirani *et al.*, "Pathwise coordinate optimization", The Annals of Applied Statistics **1**, 2, 302–332 (2007).

Frisoni, G., N. Fox, C. Jack, P. Scheltens and P. Thompson, "The clinical use of structural MRI in Alzheimer disease", Nature Reviews Neurology **6**, 2, 67–77 (2010).

Frisoni, G. *et al.*, "Detection of grey matter loss in mild Alzheimer's disease with voxel based morphometry", Journal of Neurology, Neurosurgery & Psychiatry **73**, 6, 657 (2002).

Gong, P., J. Ye and C. Zhang, "Robust multi-task feature learning", in "Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining", pp. 895–903 (ACM, 2012).

Görnitz, N., C. K. Widmer, G. Zeller, A. Kahles, G. Rätsch and S. Sonnenburg, "Hierarchical multitask structured output learning for large-scale sequence segmentation", in "Advances in Neural Information Processing Systems", pp. 2690–2698 (2011).

Gu, Q. and J. Zhou, "Learning the shared subspace for multi-task clustering and transductive transfer classification", in "Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on", pp. 159–168 (IEEE, 2009).

Gupta, S., D. Phung and S. Venkatesh, "Factorial multi-task learning: A bayesian nonparametric approach", in "Proceedings of the 30th International Conference on Machine Learning (ICML-13)", (2013).

Han, Y., F. Wu, J. Jia, Y. Zhuang and B. Yu, "Multi-task sparse discriminant analysis (mtsda) with overlapping categories.", in "AAAI", (2010).

Hastie, T., R. Tibshirani, G. Sherlock, M. Eisen, P. Brown and D. Botstein, "Imputing missing data for gene expression arrays", Tech. Report (1999).

Hernández-Lobato, D., J. M. Hernández-Lobato, T. Helleputte and P. Dupont, "Expectation propagation for bayesian multi-task feature selection", in "Machine Learning and Knowledge Discovery in Databases", pp. 522–537 (Springer, 2010).

Hernandez-Lobato, D. and J. M. Hernandez-Lobato, "Learning feature selection dependencies in multi-task learning", in "Advances in Neural Information Processing Systems", (2013).

Heskes, T. *et al.*, "Solving a huge number of similar tasks: A combination of multi-task learning and a hierarchical bayesian approach.", in "ICML", vol. 15, pp. 233–241 (Citeseer, 1998).

Heskes, T. *et al.*, "Empirical bayes for learning to learn", in "ICML", pp. 367–374 (2000).

Hoyer, P. O., "Non-negative matrix factorization with sparseness constraints", J. of Mach. Learn. Res. **5**, 1457–1469 (2004).

Ito, K. *et al.*, "Disease progression model for cognitive deterioration from Alzheimer's Disease Neuroimaging Initiative database", Alzheimer's and Dementia **6**, 1, 39–53 (2010).

Jack Jr, C., D. Knopman, W. Jagust, L. Shaw, P. Aisen, M. Weiner, R. Petersen and J. Trojanowski, "Hypothetical model of dynamic biomarkers of the alzheimer's pathological cascade", The Lancet Neurology **9**, 1, 119–128 (2010).

Jacob, L., F. Bach and J. Vert, "Clustered multi-task learning: A convex formulation", Arxiv preprint arXiv:0809.2085 (2008).

Jalali, A., S. Sanghavi, C. Ruan and P. K. Ravikumar, "A dirty model for multi-task learning", in "Advances in Neural Information Processing Systems", pp. 964–972 (2010).

Jawanpuria, P. and J. Nath, "A convex feature learning formulation for latent task structure discovery", in "Proceedings of the 29th International Conference on Machine Learning (ICML-12)", pp. 137–144 (2012).

Jebara, T., "Multi-task feature and kernel selection for svms", in "Proceedings of the twenty-first international conference on Machine learning", p. 55 (ACM, 2004).

Jebara, T., "Multitask sparsity via maximum entropy discrimination", The Journal of Machine Learning Research **12**, 75–110 (2011).

Jeffrey, R. P., R. E. Coleman and P. M. Doraiswamy, "Neuroimaging and early diagnosis of alzheimer disease : A look to the future", Radiology **226**, 315–336 (2003).

Ji, S., D. Dunson and L. Carin, "Multitask compressive sensing", Signal Processing, IEEE Transactions on **57**, 1, 92–106 (2009).

Ji, S. and J. Ye, "An accelerated gradient method for trace norm minimization", in "Proceedings of the 26th Annual International Conference on Machine Learning", pp. 457–464 (ACM, 2009).

Kang, Z., K. Grauman and F. Sha, "Learning with whom to share in multi-task feature learning", in "Proceedings of the 28th International Conference on Machine Learning (ICML-11)", pp. 521–528 (2011).

Kato, T., H. Kashima, M. Sugiyama and K. Asai, "Multi-task learning via conic programming", in "Advances in Neural Information Processing Systems", pp. 737–744 (2007).

Khachaturian, Z., "Diagnosis of Alzheimer's disease", Archives of Neurology **42**, 11, 1097 (1985).

Kim, S. and E. P. Xing, "Tree-guided group lasso for multi-task regression with structured sparsity", in "Proceedings of the 27th International Conference on Machine Learning (ICML-10)", pp. 543–550 (2010).

Kim, S. and E. P. Xing, "Tree-guided group lasso for multi-response regression with structured sparsity, with an application to eqtl mapping", The Annals of Applied Statistics **6**, 3, 1095–1117 (2012).

Kolar, M., J. D. Lafferty and L. A. Wasserman, "Union support recovery in multi-task learning.", Journal of Machine Learning Research **12**, 2415-2435, 3 (2011).

Koren, Y., "Collaborative filtering with temporal dynamics", in "KDD", pp. 447–456 (2009).

Kumar, A. and H. Daume, "Learning task grouping and overlap in multi-task learning", in "Proceedings of the 29th International Conference on Machine Learning (ICML-12)", pp. 1383–1390 (2012).

Lasko, T. A., J. C. Denny and M. A. Levy, "Computational phenotype discovery using unsupervised feature learning over noisy, sparse, and irregular clinical data", PloS one **8**, 6, e66341 (2013).

Lawrence, N., M. Seeger, R. Herbrich *et al.*, "Fast sparse gaussian process methods: The informative vector machine", Advances in neural information processing systems pp. 625–632 (2003).

Lawrence, N. D. and J. C. Platt, "Learning to learn with the informative vector machine", in "Proceedings of the twenty-first international conference on Machine learning", p. 65 (ACM, 2004).

Lazaric, A. and M. Ghavamzadeh, "Bayesian multi-task reinforcement learning", in "Proceedings of the 27th International Conference on Machine Learning (ICML-10)", pp. 599–606 (Omnipress, 2010).

Lázaro-gredilla, M. and M. K. Titsias, "Spike and slab variational inference for multi-task and multiple kernel learning", in "Advances in Neural Information Processing Systems", pp. 2339–2347 (2011).

Lee, J., Y. Sun and M. Saunders, "Proximal newton-type methods for convex optimization", in "NIPS", vol. 25, pp. 836–844 (2012).

Lee, S., J. Zhu and E. P. Xing, "Adaptive multi-task lasso: with application to eqtl detection", in "Advances in neural information processing systems", pp. 1306–1314 (2010).

Leen, G., J. Peltonen and S. Kaski, "Focused multi-task learning in a gaussian process framework", Machine learning **89**, 1-2, 157–182 (2012).

Li, H., X. Liao and L. Carin, "Multi-task reinforcement learning in partially observable stochastic environments", The Journal of Machine Learning Research **10**, 1131–1186 (2009).

Li, H., X. Liao and L. Carin, "Nonparametric bayesian feature selection for multi-task learning", in "Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on", pp. 2236–2239 (IEEE, 2011).

Li, Z., L. Cao, S. Chang, J. R. Smith and T. S. Huang, "Beyond mahalanobis distance: Learning second-order discriminant function for people verification", in "Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on", pp. 45–50 (IEEE, 2012).

Li, Z., S. Chang, F. Liang, T. S. Huang, L. Cao and J. R. Smith, "Learning locally-adaptive decision functions for person verification", in "Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on", pp. 3610–3617 (IEEE, 2013).

Lin, B., S. Yang, C. Zhang, J. Ye and X. He, "Multi-task vector field learning", in "Advances in Neural Information Processing Systems", pp. 296–304 (2012).

Liu, H., M. Palatucci and J. Zhang, "Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery", in "Proceedings of the 26th Annual International Conference on Machine Learning", pp. 649–656 (ACM, 2009a).

Liu, J., S. Ji and J. Ye, "Multi-task feature learning via efficient $\ell_{2,1}$-norm minimization", in "Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence", pp. 339–348 (AUAI Press, 2009b).

Liu, J., S. Ji and J. Ye, *SLEP: Sparse Learning with Efficient Projections*, Arizona State University, URL `http://www.public.asu.edu/ jye02/Software/SLEP` (2009c).

Liu, J., S. Ji and J. Ye, *SLEP: Sparse Learning with Efficient Projections*, Arizona State University, URL `http://www.public.asu.edu/ jye02/Software/SLEP` (2009d).

Liu, J., L. Yuan and J. Ye, "An efficient algorithm for a class of fused lasso problems", in "Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining", KDD '10, pp. 323–332 (ACM, 2010).

Lounici, K., M. Pontil, A. B. Tsybakov and S. van de Geer, "Taking advantage of sparsity in multi-task learning", arXiv preprint arXiv:0903.1468 (2009).

Lu, Z., D. Agarwal and I. S. Dhillon, "A spatio-temporal approach to collaborative filtering", in "Proc. of the 3rd ACM Conf. on Rec. Sys.", pp. 13–20 (2009).

Mallick, B. K. and S. G. Walker, "Combining information from several experiments with nonparametric priors", Biometrika **84**, 3, 697–706 (1997).

Markatou, M., P. K. Don, J. Hu, F. Wang, J. Sun, R. Sorrentino and S. Ebadollahi, "Case-based reasoning in comparative effectiveness research", IBM Journal of Research and Development **56**, 5, 4 (2012).

Mazumder, R., T. Hastie and R. Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices", J. of Mach. Learn. Res. **11**, 2287–2322 (2010).

McKhann, G., D. Drachman, M. Folstein, R. Katzman, D. Price and E. Stadlan, "Clinical diagnosis of Alzheimer's disease: report of the NINCDS-ADRDA Work Group under the auspices of Department of Health and Human Services Task Force on Alzheimer's Disease", Neurology **34**, 939–44 (1984).

Mei, S., B. Cao and J. Sun, "Encoding low-rank and sparse structures simultaneously in multi-task learning", Technical Report (2012).

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

Meka, R., P. Jain and I. Dhillon, "Guaranteed rank minimization via singular value projection", NIPS pp. 937–945 (2010).

Minka, T. P. and R. W. Picard, "Learning how to learn is learning with point sets", Unpublished manuscript. Available at http://wwwwhite. media. mit. edu/˜ tpminka/papers/learning. html (1997).

Müller, P., F. Quintana and G. Rosner, "A method for combining inference across related nonparametric bayesian models", Journal of the Royal Statistical Society: Series B (Statistical Methodology) **66**, 3, 735–749 (2004).

Murphy, E. *et al.*, "Six-month atrophy in MTL structures is associated with subsequent memory decline in elderly controls", NeuroImage (2010).

Nemirovski, A., "Efficient methods in convex programming", Lecture Notes (2005).

Nesterov, Y., *Introductory lectures on convex optimization: A basic course* (Springer Netherlands, 2004).

Ni, K., L. Carin and D. Dunson, "Multi-task learning for sequential data via ihmms and the nested dirichlet process", in "Proceedings of the 24th international conference on Machine learning", pp. 689–696 (ACM, 2007).

Obozinski, G., B. Taskar and M. I. Jordan, "Multi-task feature selection", Statistics Department, UC Berkeley, Tech. Rep (2006).

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

Paige, C. and M. Saunders, "LSQR: An algorithm for sparse linear equations and sparse least squares", ACM Transactions on Mathematical Software (TOMS) **8**, 1, 43–71 (1982).

Pan, S. J. and Q. Yang, "A survey on transfer learning", Knowledge and Data Engineering, IEEE Transactions on **22**, 10, 1345–1359 (2010).

Passos, A., P. Rai, J. Wainer and H. D. Iii, "Flexible modeling of latent task structures in multitask learning", in "Proceedings of the 29th International Conference on Machine Learning (ICML-12)", pp. 1103–1110 (2012).

Persell, S. D., A. P. Dunne, D. M. Lloyd-Jones and D. W. Baker, "Electronic health record-based cardiac risk assessment and identification of unmet preventive needs", Medical care **47**, 4, 418–424 (2009).

Philbin, E. F. and T. G. DiSalvo, "Prediction of hospital readmission for heart failure: development of a simple risk score based on administrative data", J. of the Am. Co. of Card. **33**, 6, 1560–1566 (1999).

Pong, T. K., P. Tseng, S. Ji and J. Ye, "Trace norm regularization: reformulations, algorithms, and multi-task learning", SIAM Journal on Optimization **20**, 6, 3465–3489 (2010).

Qi, G.-J., C. Aggarwal, Y. Rui, Q. Tian, S. Chang and T. Huang, "Towards cross-category knowledge propagation for learning visual concepts", in "Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on", pp. 897–904 (IEEE, 2011).

Qi, Y., D. Liu, D. Dunson and L. Carin, "Multi-task compressive sensing with dirichlet process priors", in "Proceedings of the 25th international conference on Machine learning", pp. 768–775 (ACM, 2008).

Quattoni, A., X. Carreras, M. Collins and T. Darrell, "An efficient projection for $\ell_{1,\text{inf}}$ regularization", in "Proceedings of the 26th Annual International Conference on Machine Learning", pp. 857–864 (ACM, 2009).

Rai, P. and H. Daume, "Infinite predictor subspace models for multitask learning", in "International Conference on Artificial Intelligence and Statistics", pp. 613–620 (2010).

Rai, P., A. Kumar and H. D. Iii, "Simultaneously leveraging output and task structures for multiple-output regression", in "Advances in Neural Information Processing Systems", pp. 3194–3202 (2012).

Rakotomamonjy, A., R. Flamary, G. Gasso, S. Canu *et al.*, "\ ell_p-\ ell_q penalty for sparse linear and sparse multiple kernel multi-task learning", IEEE Trans. on Neural Networks **22**, 8, 1307–1320 (2011).

Rao, N., C. Cox, R. Nowak and T. T. Rogers, "Sparse overlapping sets lasso for multi task learning and its applications to fmri analysis", in "Advances in Neural Information Processing Systems", (2013).

Rasmussen, C. and C. Williams, "Gaussian processes for machine learning", Gaussian Processes for Machine Learning (2006).

Romera-Paredes, B., A. Argyriou, N. Berthouze and M. Pontil, "Exploiting unrelated tasks in multi-task learning", in "International Conference on Artificial Intelligence and Statistics", pp. 951–959 (2012).

Rosen, W., R. Mohs and K. Davis, "A new rating scale for Alzheimer's disease", American Journal of Psychiatry **141**, 11, 1356 (1984).

Ryali, Z., T. Chen, K. Supekar and V. Menon, "Estimation of functional connectivity in fmri data using stability selection-based sparse partial correlation with elastic net penalty", Neuroimage **59**, 1, 3852–3861 (2012).

Salakhutdinov, R. and A. Mnih, "Probabilistic matrix factorization", NIPS **20**, 1257–1264 (2008).

Schneider, T., "Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing values", J. of Climate **14**, 5, 853–871 (2001).

Schwaighofer, A., V. Tresp and K. Yu, "Learning gaussian process kernels via hierarchical bayes", in "Advances in Neural Information Processing Systems", pp. 1209–1216 (2004).

Seeger, M., Y.-W. Teh, M. I. Jordan and S. Hall, "Semiparametric latent factor models", in "Proc. of the 10th International Workshop on Artificial Intelligence and Statistics", pp. 333–340 (2005).

Shen, Y., Z. Wen and Y. Zhang, "Augmented lagrangian alternating direction method for matrix separation based on low-rank factorization", Opti. Met. and Soft. **0**, 0, 1–25 (2012).

Sollich, P. and S. Ashton, "Learning curves for multi-task gaussian process regression", in "Advances in Neural Information Processing Systems", pp. 1790–1798 (2012).

Srivastava, V. and T. Dwivedi, "Estimation of seemingly unrelated regression equations: A brief survey", Journal of Econometrics **10**, 1, 15–32 (1979).

Stekhoven, D. J., I. Moraes, G. Sveinbjörnsson, L. Hennig, M. H. Maathuis and P. Bühlmann, "Causal stability ranking", Bioinformatics **28**, 21, 2819–2823 (2012).

Stern, M., K. Williams, D. Eddy and R. Kahn, "Validation of prediction of diabetes by the archimedes model and comparison with other predicting models", Diab. Care **31**, 8, 1670–1671 (2008).

Stonnington, C., C. Chu, S. Klöppel, C. Jack Jr, J. Ashburner and R. Frackowiak, "Predicting clinical scores from magnetic resonance scans in Alzheimer's disease", NeuroImage **51**, 4, 1405–1413 (2010).

Sun, J., F. Wang, J. Hu and S. Edabollahi, "Supervised patient similarity measure of heterogeneous patient records", SIGKDD Explorations **14**, 1, 16–24 (2012).

Swersky, K., J. Snoek and R. P. Adams, "Multi-task bayesian optimization", (2013).

Swirszcz, G. and A. C. Lozano, "Multi-level lasso for sparse multi-task regression", in "Proceedings of the 29th International Conference on Machine Learning (ICML-12)", pp. 361–368 (2012).

Thompson, P. *et al.*, "Mapping hippocampal and ventricular change in Alzheimer disease", Neuroimage **22**, 4, 1754–1766 (2004).

Thrun, S., "Is learning the n-th thing any easier than learning the first?", in "Advances in Neural Information Processing Systems (NIPS) 8", edited by D. Touretzky and M. Mozer, pp. 640–646 (MIT Press, Cambridge, MA, 1996a).

Thrun, S., "Learning to learn: Introduction", in "In Learning To Learn", (Citeseer, 1996b).

Thrun, S. and J. O'Sullivan, "Discovering structure in multiple learning tasks: The tc algorithm", in "ICML", vol. 96, pp. 489–497 (1996).

Thrun, S. and J. O'Sullivan, "Clustering learning tasks and the selective cross-task transfer of knowledge", Learning to learn pp. 181–209 (1998).

Tibshirani, R., "Regression shrinkage and selection via the lasso", Journal of the Royal Statistical Society. Series B (Methodological) pp. 267–288 (1996).

Tombaugh, T., "Test-retest reliable coefficients and 5-year change scores for the MMSE and 3MS", Archives of clinical neuropsychology **20**, 4, 485–503 (2005).

Tomioka, R. and S. Haufe, "Combined classification and channel/basis selection with l1-l2 regularization with application to p300 speller system", in "In Proceedings of the 4th International Brain-Computer Interface Workshop and Training Course 2008. Verlag der Technischen Universitt Graz", (2008).

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

Van Staa, T., H. Leufkens and C. Cooper, "Utility of medical and drug history in fracture risk prediction among men and women", Bone **31**, 4, 508–514 (2002).

Vemuri, P. *et al.*, "MRI and CSF biomarkers in normal, MCI, and AD subjects: predicting future clinical change", Neurology **73**, 4, 294 (2009).

Vounou, M., E. Janousova, R. Wolz, J. Stein, P. Thompson, D. Rueckert and G. Montana, "Sparse reduced-rank regression detects genetic associations with voxel-wise longitudinal phenotypes in alzheimer's disease", NeuroImage **60**, 1, 700–716 (2012).

Walhovd, K. *et al.*, "Combining MR imaging, positron-emission tomography, and CSF biomarkers in the diagnosis and prognosis of Alzheimer disease", American Journal of Neuroradiology **31**, 2, 347 (2010).

Wang, F., N. Lee, J. Hu, J. Sun and S. Ebadollahi, "Towards heterogeneous temporal clinical event pattern discovery: a convolutional approach", in "Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining", pp. 453–461 (2012).

Wen, Z., W. Yin and Y. Zhang, "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm", Math. Prog. Comp. **4**, 4, 333–361 (2012).

Widmer, C., M. Kloft, N. Görnitz and G. Rätsch, "Efficient training of graph-regularized multitask svms", in "Machine Learning and Knowledge Discovery in Databases", pp. 633–647 (Springer, 2012).

Williams, C., E. V. Bonilla and K. M. Chai, "Multi-task gaussian process prediction", in "Advances in Neural Information Processing Systems", pp. 153–160 (2007).

Williams, C., S. Klanke, S. Vijayakumar and K. M. Chai, "Multi-task gaussian process learning of robot inverse dynamics", in "Advances in Neural Information Processing Systems", pp. 265–272 (2008).

Wilson, A., Z. Ghahramani and D. A. Knowles, "Gaussian process regression networks", in "Proceedings of the 29th International Conference on Machine Learning (ICML-12)", pp. 599–606 (2012).

Wright, S. J., R. D. Nowak and M. A. Figueiredo, "Sparse reconstruction by separable approximation", IEEE Trans. on Sig. Proc. **57**, 7, 2479–2493 (2009).

Wu, J., J. Roy and W. F. Stewart, "Prediction modeling using ehr data: challenges, strategies, and a comparison of machine learning approaches", Medical care **48**, 6, S106 (2010).

Xiong, L., X. Chen, T.-K. Huang, J. Schneider and J. G. Carbonell, "Temporal collaborative filtering with bayesian probabilistic tensor factorization", in "SDM", (2010).

Xiong, T., J. Bi, R. B. Rao and V. Cherkassky, "Probabilistic joint feature selection for multi-task learning.", in "SDM", (2007).

Xu, H. and C. Leng, "Robust multi-task regression with grossly corrupted observations", in "International Conference on Artificial Intelligence and Statistics", pp. 1341–1349 (2012).

Xu, M. and J. D. Lafferty, "Conditional sparse coding and grouped multivariate regression", in "Proceedings of the 29th International Conference on Machine Learning (ICML-12)", pp. 1479–1486 (2012).

Xue, Y., D. Dunson and L. Carin, "The matrix stick-breaking process for flexible multi-task learning", in "Proceedings of the 24th international conference on Machine learning", pp. 1063–1070 (ACM, 2007a).

Xue, Y., X. Liao, L. Carin and B. Krishnapuram, "Multi-task learning for classification with dirichlet process priors", The Journal of Machine Learning Research **8**, 35–63 (2007b).

Yang, H. and J. He, "Notam2: Nonparametric bayes multi-task multi-view learning", SDM 2013 submitted (2013).

Yang, H., Z. Xu, I. King and M. R. Lyu, "Online learning for group lasso", in "Proceedings of the 27th International Conference on Machine Learning (ICML-10)", pp. 1191–1198 (2010).

Yang, M., Y. Li and Z. M. Zhang, "Multi-task learning with gaussian matrix generalized inverse gaussian model", in "Proceedings of the 30th International Conference on Machine Learning (ICML-13)", pp. 423–431 (2013).

Yang, X., S. Kim and E. P. Xing, "Heterogeneous multitask learning with joint sparsity constraints", in "Advances in neural information processing systems", pp. 2151–2159 (2009).

Yu, K., W. Chu, S. Yu, V. Tresp and Z. Xu, "Stochastic relational models for discriminative link prediction", in "Advances in neural information processing systems", pp. 1553–1560 (2006).

Yu, K., J. Lafferty, S. Zhu and Y. Gong, "Large-scale collaborative prediction using a nonparametric random effects model", in "Proceedings of the 26th Annual International Conference on Machine Learning", pp. 1185–1192 (ACM, 2009).

Yu, K., V. Tresp and A. Schwaighofer, "Learning gaussian processes from multiple tasks", in "Proceedings of the 22nd international conference on Machine learning", pp. 1012–1019 (ACM, 2005).

Yu, K., V. Tresp and S. Yu, "A nonparametric hierarchical bayesian framework for information filtering", in "Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval", pp. 353–360 (ACM, 2004).

Yu, S., V. Tresp and K. Yu, "Robust multi-task learning with t-processes", in "Proceedings of the 24th international conference on Machine learning", pp. 1103–1110 (ACM, 2007).

Yuan, M. and Y. Lin, "Model selection and estimation in regression with grouped variables", Journal of the Royal Statistical Society: Series B (Statistical Methodology) **68**, 1, 49–67 (2006).

Zellner, A., "An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias", Journal of the American statistical Association **57**, 298, 348–368 (1962).

Zha, H., X. He, C. Ding, M. Gu and H. Simon, "Spectral relaxation for k-means clustering", Advances in Neural Information Processing Systems **2**, 1057–1064 (2002).

Zhang, J., Z. Ghahramani and Y. Yang, "Learning multiple related tasks using latent independent component analysis", in "Advances in neural information processing systems", pp. 1585–1592 (2005).

Zhang, J., Z. Ghahramani and Y. Yang, "Flexible latent variable models for multi-task learning", Machine Learning **73**, 3, 221–242 (2008).

Zhang, Y. and J. G. Schneider, "Learning multiple tasks with a sparse matrix-normal penalty", in "Advances in Neural Information Processing Systems", pp. 2550–2558 (2010).

Zhang, Y. and D. Yeung, "A convex formulation for learning task relationships in multi-task learning", in "UAI", pp. 733–742 (2010a).

Zhang, Y. and D.-Y. Yeung, "Multi-task learning using generalized t process", in "Proceedings of the 13th International Conference on Artificial Intelligence and Statistics", pp. 964–971 (2010b).

Zhang, Y. and D.-Y. Yeung, "Transfer metric learning by learning task relationships", in "Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining", pp. 1199–1208 (ACM, 2010c).

Zhang, Y. and D.-Y. Yeung, "Learning high-order task relationships in multi-task learning", in "Proceedings of the Twenty-Third international joint conference on Artificial Intelligence", pp. 1917–1923 (AAAI Press, 2013).

Zhang, Y., D.-Y. Yeung and Q. Xu, "Probabilistic multi-task feature selection", in "Advances in neural information processing systems", pp. 2559–2567 (2010).

Zhong, W. and J. T. Kwok, "Convex multitask learning with flexible task clusters", in "Proceedings of the 29th International Conference on Machine Learning (ICML-12)", pp. 49–56 (2012).

Zhou, J., J. Chen and J. Ye, "Clustered multi-task learning via alternating structure optimization", in "NIPS", pp. 702–710 (2011a).

Zhou, J., L. Yuan, J. Liu and J. Ye, "A multi-task learning formulation for predicting disease progression", in "KDD", pp. 814–822 (ACM, 2011b).

Zhou, Y., R. Jin and S. Hoi, "Exclusive lasso for multi-task feature selection", in "International Conference on Artificial Intelligence and Statistics", pp. 988–995 (2010).

Zhu, J., N. Chen and E. P. Xing, "Infinite latent svm for classification and multi-task learning", in "Advances in Neural Information Processing Systems", pp. 1620–1628 (2011).