Published in final edited form as:
Neuroimage. 2014 November 15; 102 Pt 1: 192–206. doi:10.1016/j.neuroimage.2013.08.015.

Bi-level Multi-Source Learning for Heterogeneous Block-wise Missing Data

Shuo Xiang[1,2], **Lei Yuan**[1,2], **Wei Fan**[3], **Yalin Wang**[1], **Paul M. Thompson**[4], **Jieping Ye**[1,2], and **for the Alzheimer's Disease Neuroimaging Initiative**[*]

[1]School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ, USA

[2]Center for Evolutionary Medicine and Informatics, The Biodesign Institute, Arizona State University, Tempe, AZ, USA

[3]Huawei Noah's Ark Lab, Hong Kong

[4]Imaging Genetics Center, Laboratory of Neuro Imaging, Department of Neurology & Psychiatry, UCLA School of Medicine, Los Angeles, CA, USA

Abstract

Bio-imaging technologies allow scientists to collect large amounts of high-dimensional data from multiple heterogeneous sources for many biomedical applications. In the study of Alzheimer's Disease (AD), neuroimaging data, gene/protein expression data, etc., are often analyzed together to improve predictive power. Joint learning from multiple complementary data sources is advantageous, but feature-pruning and data source selection are critical to learn interpretable models from high-dimensional data. Often, the data collected has block-wise missing entries. In the Alzheimer's Disease Neuroimaging Initiative (ADNI), most subjects have MRI and genetic information, but only half have cerebrospinal fluid (CSF) measures, a different half has FDG-PET; only some have proteomic data. Here we propose how to effectively integrate information from multiple heterogeneous data sources when data is block-wise missing. We present a unified "bi-level" learning model for complete multi-source data, and extend it to incomplete data. Our major contributions are: (1) our proposed models unify feature-level and source-level analysis, including several existing feature learning approaches as special cases; (2) the model for incomplete data avoids imputing missing data and offers superior performance; it generalizes to other applications with block-wise missing data sources; (3) we present efficient optimization algorithms for modeling complete and incomplete data. We comprehensively evaluate the proposed models

Please address correspondence to: Dr. Jieping Ye, Department of Computer Science and Engineering, Center for Evolutionary Medicine and Informatics, The Biodesign Institute, Arizona State University, 699 S. Mill Ave, Tempe, AZ 85287, jieping.ye@asu.edu.

[*]Data used in preparation of this article were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.ucla.edu). As such, the investigators within the ADNI contributed to the design and implementation of ADNI and/or provided data but most of them did not participate in analysis or writing of this report. A complete listing of ADNI investigators may be found at: http://adni.loni.ucla.edu/wp-content/uploads/how_to_apply/ADNI_Acknowledgement_List.pdf

including all ADNI subjects with at least one of four data types at baseline: MRI, FDG-PET, CSF and proteomics. Our proposed models compare favorably with existing approaches.

Keywords

Alzheimer's disease; multi-modal fusion; multi-source; block-wise missing data; optimization

1 Introduction

Alzheimer’s Disease (AD), the most common form of dementia, is a highly prevalent neurodegenerative disease, in which memory and other cognitive functions decline gradually and progressively over time. AD accounts for 50–80% of dementia cases and the number of people affected by AD is expected to increase substantially over the coming decades (Brookmeyer et al., 2007). Currently there is no known cure for AD, but the detection and diagnosis of the onset and progression of AD in its earliest stages is invaluable and is the target of intensive investigation world-wide.

Recent advances in data collection technologies make it possible to collect a large amount of data to study and monitor the progression of AD. Often, these data come from multiple sources, and many studies involve multi-modality imaging. For example, different types of measurements based on magnetic resonance imaging (MRI) of the brain, positron emission tomography (PET), cerebrospinal fluid (CSF), blood tests, gene/protein expression data, and genetic data have been collected. These data are not redundant, and each of them provides complementary information for the diagnosis of AD (Calhoun et al., 2009; Fjell et al., 2010; Landau et al., 2010; Walhovd et al., 2010a). Extraction of the most useful information from such multi-source (i.e., multi-modality) data is critical in AD research. Data mining and machine learning methods have been increasingly used to analyze multi-source data (Troyanskaya et al., 2003; Xu et al., 2007; Crammer et al., 2008; Fan et al., 2008; Ye et al., 2008; Calhoun et al., 2009; Vemuri et al., 2009; Walhovd et al., 2010b; Hinrichs et al., 2011; Zhang et al., 2011; Wang et al., 2012; Yuan et al., 2012; Zhang and Shen, 2012). It is clear that both diagnostic and predictive power can be significantly improved if information from different sources is properly integrated and leveraged. Multi-source learning has thus attracted great attention in biomedical research (Ye et al., 2008; Calhoun et al., 2009; Huopaniemi et al., 2010). Multi-source learning is closely related to an area known as “multi-view” learning, but the two approaches differ in several important respects. More specifically, multi-view learning mainly focuses on semi-supervised learning and using unlabeled data to maximize the agreement between different views (Ando and Zhang, 2007; Culp et al., 2009). In this paper, we focus on multi-source learning in the supervised setting and we do not assume there are abundant unlabeled data available. In addition, we do not attempt to reduce the disagreement between multiple sources but try to extract complementary information from them, as is often the case in biomedical applications such as the study of AD.

In many applications including the study of AD, some of the available data also have a very high dimensionality, e.g., neuroimages or gene/protein expression data. However, this high-dimensional data often contains redundant information, as well as noisy or corrupted entries, and thus poses a potential challenge. To build a stable and comprehensive learning model with good generalization, it is common to apply *feature selection* - which identifies a small set of the most informative features - as a pre-processing step for classification or regression. One simple approach is to pool data from multiple sources together to create a single data matrix and apply traditional feature selection methods directly to the pooled data matrix. However, such an approach treats all sources as equally important, and ignores within-source and between-source relationships.

Another popular approach is to adopt multiple kernel learning (MKL) to perform data fusion (Lanckriet et al., 2004; Xu et al., 2007; Ye et al., 2008). This provides a principled method to perform source-level analysis, i.e., a particular source is considered relevant to the learning task only if its corresponding kernel is selected in the MKL approach. However, MKL only performs source-level analysis, ignoring feature-level analysis. Such an approach is suboptimal when the individual data sources are high-dimensional, and an interpretable model is desired. To fully take advantage of multi-source data, it is desirable to build a model that performs both individual feature-level and source-level analysis. In this paper, we will use the term “bi-level analysis”, which was introduced in (Breheny and Huang, 2009), to refer to feature- and source-level analysis, performed simultaneously.

Besides the multi-modality aspects and the high dimensionality of the data, a further problem is very commonly encountered: the existence of (block-wise) missing data is another major challenge encountered in AD and other biomedical applications. Figure 1 provides an illustration of how block-wise missing data arises in AD research. In this example, we have 245 participants in total and 3 types of measurements (PET, MRI and CSF) represented in different colors. The blank region means that data from the corresponding source is missing. In this example, participants 1–139 have available data for PET and MRI but lack CSF information, while participants 149–245 have only MRI data. The block-wise missing data situation tends to emerge in several scenarios: low-quality data sources of certain samples may be discarded; some data-collecting mechanisms (like PET) may be too costly to apply to every participant; participants may not be willing to allow certain measurements, for various reasons (e.g., lack of consent, contraindications, participant attrition, non-compliance with a long scan). Note that the missing data often emerges in a block-wise fashion, i.e., for a patient, a certain data source is either present or missing completely.

Considerable efforts have been made to deal with missing data, both in the data mining and neuroimaging communities. Some well-known missing value estimation techniques like EM (Duda et al., 1997), iterative singular value decomposition (SVD) and matrix completion (Mazumder et al., 2010) have been extended to biomedical applications by performing *imputation* on the missing part of the data. Although these approaches are effective in handling random missing entries, they often deliver sub-optimal performance in AD research (Yuan et al., 2012) for the following reasons: (1) these imputation approaches fail to capture the pattern of the missing data, i.e., the missing elements are not randomly scattered across the data matrix but emerge block-wise. However, such prior knowledge is completely discarded in imputation methods; (2) due to the high dimensionality of the data, these methods often have to estimate a significant amount of missing values, which can lead to unstable performance.

To overcome the aforementioned drawbacks of standard imputation methods, we previously proposed an incomplete Multi-Source Feature learning method (iMSF) which avoids direct imputation (Yuan et al., 2012). The iMSF method first partitions the patients into disjoint groups, so that patients from the same group possess identical data source combinations. Feature learning is then carried out independently in each group and finally the results from all groups are appropriately combined to obtain a consistent feature learning result. Such a mechanism enables iMSF to perform feature selection without estimating the missing values. Even so, the resulting model is unable to provide source-level analysis, i.e., we cannot tell which data sources are most relevant to collect in clinical practice. Such a drawback may limit the performance of iMSF in applications where noisy or corrupted data sources are frequently encountered. In addition, the iMSF method does not provide a consistent prediction model for a specific data source across different groups, though the same set of features are selected in all groups. This makes it difficult to do “out-of-sample” prediction, i.e., when the testing data involves a different data source combination from the training data.

In this paper, we propose a novel bi-level learning model, which performs simultaneous feature-level and source-level analysis. Bi-level analysis has recently drawn increasing attention (Breheny and Huang, 2009; Xiang et al., 2013), but how to extend existing techniques to deal with block-wise missing data remains largely unexplored. In this paper, we fill in this gap by proposing bi-level feature learning models for both complete and block-wise missing data. We also provide an alternative two-stage method, in which multiple data sources are first transformed into a matrix consisting of model scores with missing entries; in the second stage, the missing entries in the score matrix are estimated, and the completed score matrix is used to learn a second level model. Our contributions are three-fold: (1) we propose a unified feature learning model for multi-source data, which includes several existing feature learning approaches as special cases; (2) we further extend this model to fit block-wise missing data. The resulting incomplete model avoids direct imputation of the missing data, and is capable of bi-level feature learning; (3) the proposed models for both complete and incomplete data require solving non-convex optimization problems. We present efficient optimization algorithms, to find the solution by solving a sequence of convex sub-problems. The proposed incomplete model learns a single model for each data source across different groups (each group corresponds to one data source combination), and learns the prediction model for each group by computing a weighted combination of the models (one model for each source) involved in the group, thus it provides “out-of-sample” prediction, overcoming the limitation of the iMSF method.

We also evaluate the effectiveness of the proposed models, compared to existing methods using data from the Alzheimer's Disease Neuroimaging Initiative (ADNI). A total of 780 subjects, who have at least one of the four major types of data (MRI, PET, CSF, and proteomics) were available at baseline, and were included in our study. Our experiments show the potential of the proposed models for analyzing multiple heterogeneous sources with block-wise missing data.

2 Subjects

We use data from the Alzheimer's disease Neuroimaging Initiative (ADNI) (www.adni-info.org). ADNI was launched in 2003 by the National Institute on Aging (NIA), the National Institute of Biomedical Imaging and Bioengineering (NIBIB), the Food and Drug Administration (FDA), private pharmaceutical companies and non-profit organizations, as a 5-year public private partnership. ADNI's primary goal has been to test whether serial magnetic resonance imaging (MRI), positron emission tomography (PET), other biological markers, and clinical and neuropsychological assessments can be combined to measure the progression of mild cognitive impairment (MCI) and early Alzheimer's disease (AD). Determination of sensitive and specific markers of very early AD progression is intended to aid researchers and clinicians to develop new treatments and monitor their effectiveness, as well as lessen the time and cost of clinical trials.

ADNI is the result of efforts of many co-investigators from a broad range of academic institutions and private corporations, and subjects have been recruited from over 50 sites across the U.S. and Canada. ADNI's initial goal was to recruit 800 subjects, but follow-on projects, known as ADNI-GO and ADNI-2, have recruited over 1500 adults, aged 55 to 90, to participate in the research, consisting of cognitively normal older individuals, people with early or late MCI, and people with early AD. The follow-up intervals for each diagnostic subgroup is specified in the protocols for ADNI-1, ADNI-2 and ADNI-GO. Subjects originally recruited for ADNI-1 and ADNI-GO had the option of being followed longitudinally in ADNI-2.

In this paper, we use four types of data sources, e.g., MRI, PET, CSF, and proteomics, including a total of 780 subjects (i.e., anyone who had at least one of these measures at baseline). The MRI image features in this study were based on the imaging data from the ADNI database processed by the UCSF team, who performed cortical reconstruction and volumetric segmentations with the FreeSurfer image analysis suite (http://surfer.nmr.mgh.harvard.edu/). We note that many other measures could be, and have been, derived from the MRIs, but this is a representative set, intended to illustrate our approach. The processed MRI features come from a total of 648 subjects (138 AD, 142 progressive MCI, 177 stable MCI and 191 Normal), and may be grouped into 5 categories: average cortical thickness, standard deviation in cortical thickness, the volumes of cortical parcellations, the volumes of specific white matter parcellations, and the total surface area of the cortex. There were 305 MRI features in total. We also downloaded baseline FDG-PET images from 327 subjects (76 AD, 70 progressive MCI, 100 stable MCI and 81 Normal) from the ADNI website. We processed these FDG-PET images using SPM8 (http://www.fil.ion.ucl.ac.uk/spm/). Specifically, we applied Automated Anatomical Labeling (AAL) (Tzourio-Mazoyer et al., 2002) to extract each of the 116 anatomical volumes of interest (AVOI) and derived average image values from each AVOI, for every subject. Baseline CSF samples were acquired from 409 subjects (100 AD, 84 progressive MCI, 111 stable MCI and 114 Normal) by the ADNI Biomarker Core laboratory at the University of Pennsylvania Medical Center (Tzourio-Mazoyer et al., 2002). The proteomics data set (112 AD, 163 progressive MCI, 233 stable MCI and 54 Normal) was produced by the Biomarkers Consortium Project "Use of Targeted Multiplex Proteomic Strategies to Identify Plasma-

Based Biomarkers in Alzheimer's Disease"[*]. We use 147 measures from the proteomic data downloaded from the ADNI web site. As a result, for a subject with all four types of data available, a total of 571 measures were analyzed in our study. The statistics of these data sources are shown in Table 1.

3 A Unified Feature Learning Model for Multi-source Complete Data

We first present a unified learning model for multi-source data without missing values. We show how to extend the model to deal with block-wise missing data in Section 4.

Assume we are given a collection of *m* samples from *S* data sources:

$$\boldsymbol{X}=[\boldsymbol{X}_1,\boldsymbol{X}_2,\ldots,\boldsymbol{X}_S]\in\mathbb{R}^{m\times n},\boldsymbol{y}\in\mathbb{R}^{m},$$

where $\boldsymbol{X}_i \in \mathbb{R}^{m\times p_i}$ is the data matrix of the *i*th source with each sample being a p_i-dimensional vector, and $\boldsymbol{y}$ is the corresponding outcome for each sample. We consider the following linear model:

$$\boldsymbol{y}=\sum_{i=1}^{S}\boldsymbol{X_i}\boldsymbol{\beta}_i+\boldsymbol{\varepsilon}=\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{\varepsilon}, \quad (1)$$

where each column of $\boldsymbol{X}$ is normalized to be zero mean and to have a standard deviation of 1, and ε represents the noise term. β is the underlying true model, and is usually unknown in real-world applications. Based on ($\boldsymbol{X}, \boldsymbol{y}$), we want to learn an estimator of β, denoted as $\hat{\beta}$, whose non-zero elements $\mathscr{F} = \{j: \hat{\beta}_j \neq 0\}$ correspond to the relevant features. In other words, features corresponding to the zero elements of $\hat{\beta}$ are discarded. We consider the following regularization framework:

$$\min_{\boldsymbol{\beta}}\boldsymbol{L}(\boldsymbol{\beta})+\boldsymbol{\Omega}(\boldsymbol{\beta}),$$

where $\boldsymbol{L}(\cdot)$ represents the data-fitting term and Ω(·) is the regularization term which encodes our prior knowledge about β. Specifically, the choice of Ω(·) should also enable us to perform both feature-level and source-level analysis simultaneously. Towards this end, a natural approach is a two-stage model. First we learn different models for each data source and then combine these learned models appropriately. The regularization should be imposed independently on each stage, to provide a bi-level analysis. We formalize our intuition as follows:

$$\min_{\boldsymbol{\alpha},\boldsymbol{\gamma}}\frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\gamma_i\cdot\boldsymbol{X}_i\boldsymbol{\alpha}_i\right\|_2^2+\sum_{i=1}^{S}\frac{\lambda_i}{p}\|\boldsymbol{\alpha}_i\|_p^p+\sum_{i=1}^{S}\frac{\eta_i}{q}|\gamma_i|^q, \quad (2)$$

where the minimization is taken with respect to (**α, γ**) jointly. According to the intuition above, α_i denotes the model learned using the *i*th data source and γ is the weight that

[*] http://adni.loni.ucla.edu/wp-content/uploads/2010/11/BC_Plasma_Proteomics_Data_Primer.pdf

combines those learned models together. The regularization is taken independently over α and γ and therefore we have the flexibility to choose different values of *p* and *q* to induce sparsity on either the feature-level or the source-level, thus achieving the goal of feature selection and source selection. Notice that model (2) is not jointly convex, and direct optimization towards (2) would be difficult. We provide an equivalent, but simpler, formulation in the following theorem, and discuss its optimization in the next section.

Theorem 1 *The formulation (***2***) is equivalent to the following optimization problem:*

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\sum_{i=1}^{S}\nu_i\|\boldsymbol{\beta}_i\|_p^{\frac{pq}{p+q}}. \quad (3)$$

Proof. Without loss of generality, we assume that α ≠ **0** for all *i* = 1,2, ⋯, *S*. Since if $\alpha_i = 0$ for some *i*, the optimal γ_i must be 0 and therefore both α_i and γ_i may be removed from (2).

Let $\beta_i = \gamma_i \cdot \alpha_i$ and replace γ_i with $\frac{\|\boldsymbol{\beta}_i\|_p}{\|\boldsymbol{\alpha}_i\|_p}$, we can obtain an equivalent formulation:

$$\min_{\boldsymbol{\alpha},\boldsymbol{\beta}} \frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\sum_{i=1}^{S}\frac{\lambda_i}{p}\|\boldsymbol{\alpha}_i\|_p^p+\sum_{i=1}^{S}\frac{\eta_i}{q}\left(\frac{\|\boldsymbol{\beta}_i\|_p}{\|\boldsymbol{\alpha}_i\|_p}\right)^q. \quad (4)$$

Taking the partial derivative with respect to α_i, and setting it to zero, leads to:

$$\eta_i\|\boldsymbol{\beta}_i\|_p^q=\lambda_i\|\boldsymbol{\alpha}_i\|_p^{p+q}, i=1,2,\cdots,S. \quad (5)$$

Plugging (5) back into (4) with the change of variables, we get the formulation (3).

3.1 Relation to previous work

Formulation (2) [or its equivalent form (3)] is a very general model. Assigning different values to *p* and *q* leads to various kinds of regularization and feature learning models. Next, we show several widely-used convex models are actually special cases of our model.

Let *p* = 1 and *q* = ∞. In this case, the regularization term in (3) becomes the ℓ_1-regularization, and the resulting model becomes Lasso (Tibshirani, 1996):

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\lambda\|\boldsymbol{\beta}\|_1. \quad (6)$$

It is well-known that the ℓ_1-regularization leads to a sparse solution, which coincides with the goal of feature selection. However, it does not consider the source structure, as it treats all features from different sources equally.

On the other hand, if both *p* and *q* are equal to 2, then the ℓ_2-regularization is applied on each source. Letting $\nu_i=\lambda\sqrt{p_i}$ leads to the group lasso model (Yuan and Lin, 2006):

$$\min_{\beta}\frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\lambda\sum_{i=1}^{S}\sqrt{p_i}\|\boldsymbol{\beta}_i\|_2. \tag{7}$$

Similarly, if $p = \infty$ and $q = 1$, we obtain the $\ell_{1,\infty}$-regularization model (Turlach et al., 2005; Quattoni et al., 2009), which penalizes the largest elements of β_i for each source:

$$\min_{\beta}\frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\sum_{i=1}^{S}\nu_i\|\boldsymbol{\beta}_i\|_{\infty}. \tag{8}$$

Besides these common convex formulations, our general model also includes a family of non-convex formulations, which have not been fully explored in the literature. In particular, setting $p = 1$ and $q = 2$ leads to the following non-convex model:

$$\min_{\beta}\frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\sum_{i=1}^{S}\nu_i\|\boldsymbol{\beta}_i\|_1^{\frac{2}{3}}, \tag{9}$$

and if $p = 2$ and $p = 1$, model (3) reduces to:

$$\min_{\beta}\frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\sum_{i=1}^{S}\nu_i\|\boldsymbol{\beta}_i\|_2^{\frac{2}{3}}. \tag{10}$$

For the convex models such as lasso, both the optimization algorithms and the statistical properties have received intensive study (Efron et al., 2004; Zhao and Yu, 2006; Bickel et al., 2009; Bach, 2011). However, due to the non-convexity nature, it is more difficult to compute the optimal solution of models (9) and (10). We present a difference of convex functions (DC) framework for these formulations in the Appendix.

Remark 1. Although we only consider the least squares loss function here, the above derivations can be easily extended to other widely-used convex loss functions, such as the logistic function.

Remark 2. In the proposed multi-source learning formulation, both feature and source level analyses are performed in a unified formulation. As an alternative, we also present a two-stage approach, which performs feature-level and source-level analyses separately. The simple structure of the two-stage approach makes it easier to deal with the block-wise missing data. Details are presented in the Appendix.

4 Incomplete Source-Feature Selection (ISFS) Model

In this section, we consider the more challenging and more realistic situation of block-wise missing data, as shown in Figure 1. In such situations, many (or even the majority of) patients do not have complete data collected from every data source, and lack one or more data blocks. To apply existing feature learning approaches directly, we can either discard all samples that have missing entries, or we can estimate the missing values based on the observed entries. However, the former approach may significantly reduce the size of the data set while the latter approach heavily relies on our prior knowledge about the missing values. Moreover, both approaches neglect the block-wise missing patterns in the data and therefore could lead to sub-optimal performance.

As in the case of complete data, an ideal model performs both feature- and source-level analysis simultaneously. Next, we show how to extend the model on complete data presented in the previous section to a more general setting with missing data. Our intuition of designing such an incomplete Source-Feature Selection (iSFS) model is illustrated in Figure 2. We follow a similar strategy used in our complete model (2): the individual model is learned on each data source, and then all models are properly integrated via extra regularizations/constraints. As shown in Figure 2, we try to learn the models represented by β_1, β_2 and β_3, corresponding to measurements from PET, MRI and CSF, respectively. A subtle issue is how to learn the coefficients α, as model (2) is not applicable due to the presence of missing data blocks. To address this, we partition the whole data set into multiple groups according to the availability of data sources, as illustrated in the red boxes in Figure 2. For this particular case, we partition the data into 4 groups, where the first group includes all the samples that have PET and MRI, the second group of subjects possesses all three data sources, the third group of subjects has MRI and CSF measurements, while the last group of subjects only has MRI data. Note that within each of these groups, we do have complete data, and the analysis from the previous section can be applied.

The proposed model is closely related to the iMSF model proposed in (Yuan et al., 2012), but they differ in several significant respects: (1) the proposed method partitions the data into multiple groups according to the availability of data sources. The resulting groups are not disjoint, compared to that of the iMSF. Generally, our partition method results in more samples for each group; (2) in the proposed approach, the model learned for each data source is consistent across different data source combinations, while iMSF does not. This is beneficial, when we are encountered with samples whose data source combination does not appear in the training set, therefore providing the “out-of-sample” prediction. (3) In iSFS, we can represent the weight vector of the group with profile m as $[\boldsymbol{\alpha}_{\boldsymbol{m}}^{\mathbf{1}}\boldsymbol{\beta}^{\mathbf{1}}, \boldsymbol{\alpha}_{\boldsymbol{m}}^{\mathbf{2}}\boldsymbol{\beta}^{\mathbf{2}}, \cdots, \boldsymbol{\alpha}_{\boldsymbol{m}}^{\boldsymbol{s}}\boldsymbol{\beta}^{\boldsymbol{s}}]$, where $\boldsymbol{\alpha}_{\boldsymbol{m}}^{\boldsymbol{i}}$ is the weight assigned to the *i*th source in the group, with $\boldsymbol{\alpha}_{\boldsymbol{m}}^{\boldsymbol{i}}=0$ if the *i*th source is not involved in the profile m, and $\beta_{\boldsymbol{i}}$ is the (consistent) weight vector of the model parameters for the *i*th source. In the proposed formulation, we learn the weights and the weight vectors simultaneously. Note that the weights for different sources in a specific group may differ and they are learnt adaptively. In essence, the iSFS formulation constrains each data source to learn a consistent model across multiple source combinations, resulting in a much smaller number of model parameters than iMSF. iSFS can be considered as a constrained version of iMSF. Thus, iSFS is expected to achieve better generalization performance than iMSF especially when the number of samples in the training set is small or noisy data sources are present.

4.1 Formulation

Before presenting the formal description of our iSFS model, we first introduce some notations, which will simplify the discussion. Suppose we have data sources in total, and each participant has at least one data source available. Then there are $2^S - 1$ possible missing patterns: the number of all possible combinations of *S* data sources, except for the case that all data sources are missing. For each participant, based on whether a certain data source is present, we obtain a binary indicator vector ***I***[1 ⋯ *S*], where ***I***[*i*] = 1 indicates the th data source is available. Recall in Figure 1, participants 1 ~ 139 possess the same indicator vector [1,1,0] while the indicator vector of participants 149 ~ 245 is [0,1,0]. Using such indicator vectors simplifies our analysis. Moreover, we do not even need to store the complete vector for each participant but just need to record a single decimal integer if we convert this binary vector to a binary number, i.e., the information in the indicator vector can be completely described by a decimal integer, called "**profile**". All these profiles are stored in an *n*-dimensional vector ***pf*** [1..*n*] where *n* is the number of participants.

We are ready to give a concise description of our model. Following the aforementioned intuitions, we learn a consistent model (variable β) across different source combinations, while within each combination, the weights (variable α) for different sources are learned adaptively. Mathematically, the proposed model solves the following formulation:

$$\begin{array}{ll}\min\limits_{\boldsymbol{\alpha},\boldsymbol{\beta}} & \frac{1}{|\boldsymbol{pf}|}\sum_{m\in\boldsymbol{pf}} f(\boldsymbol{X}_m,\boldsymbol{\beta},\boldsymbol{\alpha}_m,\boldsymbol{y}_m)+\lambda\boldsymbol{R}_{\beta}(\boldsymbol{\beta}) \\ s.t. & \boldsymbol{R}_{\alpha}(\boldsymbol{\alpha}_m)\le 1,\forall m\in\boldsymbol{pf},\end{array} \qquad (11)$$

where

$$f(\boldsymbol{X},\boldsymbol{\beta},\boldsymbol{\alpha},\boldsymbol{y})=\frac{1}{n}\boldsymbol{L}\left(\sum_{i=1}^{S}\alpha^i\boldsymbol{X}^i\beta^i,\boldsymbol{y}\right) \qquad (12)$$

and $\boldsymbol{R}_{\alpha}$, $\boldsymbol{R}_{\beta}$ are regularizations on α**,** β respectively. The *m* subscript in (11) means that the matrix/vector is restricted to the samples that contain *m* in their profiles. $\boldsymbol{X}^i$ and β^i in (12) represent the data matrix and the model of the *i*th source, respectively. ***L*** can be any convex loss function such as the least squares loss function, or the logistic loss function, and *n* is the number of rows of ***X***.

It is worthwhile to note that for the case with complete data, the two formulations in Eq. (2) and Eq. (11) differ in how they employ the penalty on the source-level. Specifically, in Eq. (2) the source-level penalty appears as a regularizer, while in Eq. (11), such a penalty is incorporated into the model via an explicit constraint.

4.2 Optimization

One of the advantages of iMSF is its efficient optimization algorithm. In fact, iMSF may be solved by standard convex multi-task learning algorithms (Argyriou et al., 2008; Liu et al., 2009). The proposed iSFS model involves a more complicated optimization problem. In fact, (11) is not jointly-convex w.r.t. α and β, posing a major challenge. We adapt the alternating minimization method to solve (11). More specifically, we first initialize β and compute the optimal α. Then β is updated based on the computed α. We keep this iterative procedure until convergence. For simplicity, we focus on the least squares loss function in the following discussion. The techniques can be easily extended to other loss functions, e.g., the logistic loss function.

4.2.1 Computing α when β is fixed—As shown in Figure 2, we learn the weight α for each source combination independently. Therefore, when β is fixed, the objective function of (11) is actually decoupled w.r.t. α_m and the optimal α_m is given by the optimal solution of the following problem:

$$\begin{aligned} \min_{\boldsymbol{\alpha},\boldsymbol{\beta}} \quad & \left\| \textstyle\sum_{i=1}^{S} \alpha^i \boldsymbol{X}^i \boldsymbol{\beta}^i - \boldsymbol{y} \right\|_2^2 \\ s.t. \quad & \boldsymbol{R}_{\boldsymbol{\alpha}}(\boldsymbol{\alpha}) \le 1. \end{aligned} \quad (13)$$

For many choices of the regularization term $\boldsymbol{R}_\alpha$, such as the ridge penalty, the ℓ_1-norm penalty, and other sparsity-induced penalties (Bach, 2011; Ye and Liu, 2012) the optimal solution of (13) may be efficiently computed via the accelerated gradient algorithm (Beck and Teboulle, 2009).

4.2.2 Computing β when α is fixed—When we keep α fixed and seek the optimal β, (11) becomes an unconstrained regularization problem:

$$\min_{\boldsymbol{\beta}} g(\boldsymbol{\beta}) + \lambda \boldsymbol{R}_\beta(\boldsymbol{\beta}) \quad (14)$$

where

$$g(\beta) = \frac{1}{|\boldsymbol{pf}|} \sum_{m \in \boldsymbol{pf}} \frac{1}{2n_m} \left\| \sum_{i=1}^{S} (\boldsymbol{\alpha}_m^i \boldsymbol{X}_m^i) \boldsymbol{\beta}_m^i - \boldsymbol{y}_m \right\|_2^2 .$$

and n_m is number of rows of $\boldsymbol{X}_m$. We can observe that g(β) is a quadratic function of β and thus the overall formulation is to minimize the summation of a quadratic term and a regularization term: a typical formulation that can be solved efficiently via the accelerated gradient method, provided that the following proximal operator:

$$\min_{\boldsymbol{\beta}} \frac{1}{2} \|\boldsymbol{\beta} - \boldsymbol{\nu}\|_2^2 + \lambda \boldsymbol{R}_\beta(\boldsymbol{\beta})$$

can be computed efficiently. Indeed, this is the case for many widely used regularization terms. In addition, in order to apply standard first-order lasso solvers, we only need to provide the gradient of β at any given point without knowing the explicit quadratic form. For each data source i, we can compute the gradient of the g(β) w.r.t. β^i as follows:

$$\nabla g(\boldsymbol{\beta}^i) = \frac{1}{|\boldsymbol{pf}|} \sum_{m \in \boldsymbol{pf}} \frac{1}{n_m} \boldsymbol{I}(m \& 2^{S-i} \neq 0) (\boldsymbol{\alpha}_m^i \boldsymbol{X}_m^i)^T \left(\sum_{i=1}^{S} (\boldsymbol{\alpha}_m^i \boldsymbol{X}_m^i \boldsymbol{\beta}_m^i - \boldsymbol{y}_m \right), \quad (15)$$

where $\boldsymbol{I}(\cdot)$ is the indicator function which has value 1 when the condition is satisfied and 0 otherwise. The expression m & $2^{S-i} \neq 0$ ensures that the ith source exists in the combination m, where & denotes the bit-wise "AND" operation. Then we can obtain ∇g(β) by stacking all of ∇$g(\beta^i)$, for $i = 1,2, \cdots, S$ and finally obtain a global solution of (14) via applying the accelerated gradient method. Algorithm 1 summarizes our alternating minimization scheme.

Algorithm 1

The proposed alternating algorithm for solving (11)

```
Input: $\boldsymbol{X}$, $\boldsymbol{y}$, $\lambda$
Output: solution $\alpha$, $\beta$ to (11)
Initialize $(\beta^i)^0$ by fitting each source individually on the available data.
for $k = 1,2, \cdots$ do
  Compute $(\alpha)^k$ via solving a constrained optimization problem (13).
  Update $(\beta)^k$ via solving a regularized optimization problem (14).
  If the objective stops decreasing then
    return $\beta = (\beta)^k$, $\alpha = (\alpha)^k$
  end if
end for
```

Remark 1. Our model can be easily extended to the logistic loss function, which is widely used in classification problems. Computing α in (13) amounts to solving a constrained logistic regression problem while computing β in (14) requires solving a regularized logistic regression problem. In fact, any convex loss function can be applied to our model as long as the gradient information can be efficiently obtained.

Remark 2. We may apply different forms of $\boldsymbol{R}_\alpha$ and $\boldsymbol{R}_\beta$ in order to capture more complex structures, as long as the associated proximal operator can be efficiently computed. In particular, we can employ the ℓ_1-norm penalty to achieve the simultaneous feature- and source-level selection.

Remark 3. A special case of the proposed iSFS model can be obtained by setting α_m to $1/n_m$ for every m, where n_m is the number of samples that have profile m. As a result, the optimization (11) only involves β and becomes a convex programming problem. In fact, this can be considered as an extension of the classical lasso method to the block-wise missing data. To the best of our knowledge, such an extension is not known in the existing literature.

Remark 6. Note that source selection can also be achieved via a two-stage approach, in which we first train a model for each individual data source and make predictions, and then we build a regression model on these predictions in the second stage. The zero coefficients in the regression model in the second stage correspond to irrelevant data sources. One disadvantage of this approach, as well as other similar ones that perform the feature-level and source-level learning in two stages, is that the optimization is not carried out jointly w.r.t β (feature-level) and α (source-level) and the solution may be suboptimal. On the contrary, the proposed model learns β and α together by updating them iteratively.

5 Results

To examine the efficacy of the proposed bi-level feature learning models, we report the performance of the proposed models for complete and block-wise missing data, based on both synthetic data and ADNI data. Specifically, the following aspects are evaluated: (i) models (9) and (10) for complete data; (ii) model (11) for block-wise missing data; (iii) the capability of source-level analysis; (iv) the benefit of utilizing incomplete data; (v) model ensemble.

5.1 Comparison on complete data

We first evaluate the effectiveness of the complete models (9) and (10) on synthetic data generated by the linear model (1). The parameter settings follow the similar strategy described in (Friedman et al., 2010; Yang et al., 2010). Specifically, we have $S = 20$ sources in total and the underlying true model $\beta = [\beta_{\mathbf{1}}^{\boldsymbol{T}}, \beta_{\mathbf{2}}^{\boldsymbol{T}}, \cdots, \beta_{\boldsymbol{s}}^{\boldsymbol{T}}]^{\boldsymbol{T}}$ only takes non-zero values in the first six sources, whose values are 10, 8, 6, 4, 2 and 1 respectively. The data matrix $\boldsymbol{X} = [\boldsymbol{X}_{\mathbf{1}}, \boldsymbol{X}_{\mathbf{2}}, \cdots, \boldsymbol{X}_{\boldsymbol{s}}]$ and the noise term ε all follow the Gaussian distribution with zero mean and standard deviation of 0.5. To evaluate the performance of bi-level feature learning, we consider the following two situations: (1) all features within the six sources are useful, i.e., the elements of $\beta_{\boldsymbol{i}}$, $i = 1,2, \cdots 6$ are all non-zero; (2) not all features within the six sources are useful, i.e., $\beta_{\boldsymbol{i}}$ is sparse for $i = 1,2, \cdots, 6$ Specifically, only the first 3 features within each $\beta_{\boldsymbol{i}}$ are nonzero. Figure 3 illustrates these two settings.

For each scenario, we partition the data set into a disjoint training set and test set, and we compare models (9), (10) with lasso, group lasso and sparse group lasso. 5-fold cross-validation is employed to tune the parameters for each model. Specifically, the set of tuning parameters for lasso, group lasso, model (9) and model (10) are chosen from the interval $M = [10^{-8}, 10^{2}]$. For the sparse group lasso, its parameters are chosen from the product space of $M \times M$. We report the number of features and groups selected by each model and the mean squared error (MSE) on the testing set. In addition, as we know the underlying true model β, we also include the parameter estimation error: $\|\hat{\beta} - \boldsymbol{\beta}\|_2^2$, where $\hat{\beta}$ is the estimated model. All the results are averaged over 10 replications, and are listed in Table 2. For simplicity, we use *FRAC*(1,2) to denote model (9) ($p = 1$, $q = 2$) and *FRAC*(2,1) to denote model (10). The experimental results show that, in the situation of sparse features, model (9) achieves the least MSE and parameter estimation error, while for the non-sparse feature scenario, model (10) outperforms the others. In addition, in both cases, models (9) and (10) demonstrate significant improvement over the lasso, group lasso and sparse group lasso.

5.2 Comparison on block-wise missing data

Next, we consider the more realistic setting, where data is block-wise missing. We evaluate our models for the diagnostic classification of individuals in ADNI, based on their collected data. As noted earlier, we use the Alzheimer's Disease Neuroimaging Initiative (ADNI) data set (Mueller et al., 2005; Jack et al., 2008) and choose 4 data sources for each patient: proteomics, PET, MRI and CSF. We investigate the classification of subjects as AD patients, normal control (NC) subjects, stable MCI subjects (non-converters) and progressive MCI subjects (converters). Imputation methods such as Mean-value imputation, EM, KNN, iterative SVD and matrix completion, the two-stage approaches (called ScoreComp; please refer to the Appendix for details) using KNN and EM, as well as the iMSF feature learning model, are included for comparison. The evaluations for imputation and feature learning methods are achieved in two steps. First, we either apply the feature learning methods to select informative features or the imputation methods to fill in the missing entries in the data. Then in the second step, the Random Forest classifier[†] is applied to perform the classification. We consider using 10% and 50% of the ADNI data for the training stage respectively and report the accuracy, sensitivity, specificity, the area under the ROC curve (AUC value) as well as the error bars on the remaining test data. 5-fold cross-validation is used to select suitable parameters for iSFS, iMSF, KNN and SVD. In particular, for iSFS, iMSF and matrix completion, we choose five values from $[10^{-5}, 10^{1}]$ on the log scale, as candidates. For KNN, the size of the neighborhood is selected from [1, 5, 10, 15, 20, 25]. The rank parameter in the SVD is chosen from [5, 10, 15, 20, 25, 30]. In addition, we employ the ℓ_1-norm penalty for both $\boldsymbol{R}_\alpha$ and $\boldsymbol{R}_\beta$. The results are presented in Table 3 to Table 8. All results are averaged over 10 repetitions. From the evaluation results, we can observe that: (1) among all imputation methods, the mean-value imputation and EM demonstrate better performance in terms of accuracy. However, their results are not stable, as revealed by the low sensitivity/specificity value in some tasks; (2) the ScoreComp methods deliver superior performances in the classification of AD patients and normal controls; (3) the feature learning models, such as iSFS and iMSF, outperform the imputation methods and often achieve uniform improvement across all the measurements. This coincides with our intuition that estimating the missing blocks directly is usually difficult and unstable and approaches avoiding imputation are preferred. In particular, iSFS clearly delivers the best performance among all approaches. We can also observe from the results that when 10% of the data is used for training, iSFS consistently outperforms iMSF. However, iSFS and iMSF achieve comparable performance when 50% of the data is used for training. This is consistent with our analysis in Section 4, in which we show that the iSFS formulation can be considered as a constrained version of iMSF and it involves a much smaller number of model parameters than iMSF. Thus, iSFS is expected to outperform iMSF especially when the number of samples in the training set is small.

5.3 Capability of source selection

Motivated by the strategies used in (Lanckriet et al., 2004), we add two random (noisy) data sources to the ADNI data set, to verify the performance of source-level learning. We compare our iSFS model with iMSF and report their performance in Figure 5. Besides the previous tasks, we perform two additional evaluations: AD patients vs. MCI and MCI vs. Normal Controls. We can see that our method outperforms the iMSF model in most of the cases. Such a result again justifies the importance of source-level analysis, especially when noisy/corrupted data sources are present.

5.4 Benefit of utilizing incomplete data

The proposed approach makes full use of all available data: every sample with at least one available data source could contribute to the overall system. Here we provide a concrete study to show how this could be beneficial and potentially improve the performance. As in the previous evaluations, we utilize the data sources of Proteomics, PET, MRI and CSF, and extract all the samples that have all four data sources. The classification given by iSFS on both complete and incomplete data and other feature learning approaches, including lasso

[†] http://www.stat.berkeley.edu/~breiman/RandomForests/

and group lasso (on the smaller complete data) are reported in Figure 6, where iSFSC denotes the result given by iSFS on only complete data. We can observe that, by incorporating the information provided by related but incomplete samples, the classification performance on the complete data can be improved substantially.

5.5 Ensemble learning methods

In this experiment, we employed various ensemble learning approaches to further boost the performance for classification of the ADNI data. Ensemble learning is a commonly used scheme in machine learning and data mining, which properly integrates the models/results learned by different algorithms. In our evaluation, we consider the following two simple ensemble strategies: (1) majority vote; (2) learning the combination coefficients via linear regression. In the first approach, the prediction of a given sample is based on majority voting by all of the algorithms. In other words, all of the participating algorithms are treated equally. By contrast, we learn the combination weights for each algorithm, in the second approach. Therefore the final prediction is based on a weighted-combination of the results obtained from each individual algorithm. Specifically, we include two imputation models: mean-value imputation and KNN. In addition, for each of iMSF and iSFS, we select two parameters (0.001, 0.01), which results in 6 models in total. Figure 7 illustrates the ensemble learning results with varying ratios of training data - we can observe that model ensemble often improves the overall performance of the learning system.

6 Discussion

In this paper, we investigate a bi-level feature learning approach, motivated by the multi-modal data analysis in AD research. We propose systematic approaches for data model learning, for both complete and block-wise missing data. Specifically, we introduce a unified feature learning model for complete data, which contains several classical convex models as special cases. We further show that the model for complete data can be easily extended to handle the more challenging block-wise missing data, which is often a major challenge encountered in AD and other biomedical applications.

Numerical results on algorithm efficiency

The proposed bi-level learning approach involves solving a non-convex optimization problem, which is often more difficult than its convex counterpart. Because of the complicated heterogeneity nature of the missing data problem, it is much advantageous to develop an efficient numerical scheme. Our experience shows that the proposed alternating minimization method can achieve a reasonable efficiency performance. Figure 8 illustrates the efficiency of Algorithm 1 where the objective value of (11) is plotted as the iteration increases. We can see that the proposed algorithm converges quickly after the first few iterations. We also report the running time of the proposed optimization procedure with increasing number of samples and number of sources in Figure 9. The results demonstrate the efficiency of the proposed algorithm.

7 Conclusion and Future Work

We present a bi-level multi-source feature learning framework for both complete and block-wise missing data. Our proposed model is general and may lead to various kinds of feature learning models. The proposed source-level analysis is particularly useful when noisy/corrupted data sources are present. We also propose efficient numerical schemes to solve the introduced non-convex optimization problems. Our experiments on both synthetic and ADNI data sets demonstrate the efficacy of our proposed framework. Ongoing work is to extend the current model to other missing data problems. For example, although block-wise missing data is common in biomedical applications, random missing entries may also appear during the data collection process. How to generalize our model to deal with random missing values would be an interesting topic for us to explore in the future.

Acknowledgments

Data collection and sharing for this project was funded by the Alzheimer's Disease Neuroimaging Initiative (ADNI) (National Institutes of Health Grant U01 AG024904). ADNI is funded by the National Institute on Aging, the National Institute of Biomedical Imaging and Bioengineering, and through generous contributions from the following: Abbott; Alzheimer’s Association; Alzheimer’s Drug Discovery Foundation; Amorfix Life Sciences Ltd.; AstraZeneca; Bayer HealthCare; BioClinica, Inc.; Biogen Idec Inc.; Bristol-Myers Squibb Company; Eisai Inc.; Elan Pharmaceuticals Inc.; Eli Lilly and Company; F. Hoffmann-La Roche Ltd and its affiliated company Genentech, Inc.; GE Healthcare; Innogenetics, N.V.; Janssen Alzheimer Immunotherapy Research & Development, LLC.; Johnson & Johnson Pharmaceutical Research & Development LLC.; Medpace, Inc.; Merck & Co., Inc.; Meso Scale Diagnostics, LLC.; Novartis Pharmaceuticals Corporation; Pfizer Inc.; Servier; Synarc Inc.; and Takeda Pharmaceutical Company. The Canadian Institutes of Health Research is providing funds to support ADNI clinical sites in Canada. Private sector contributions are facilitated by the Foundation for the National Institutes of Health (www.fnih.org). The grantee organization is the Northern California Institute for Research and Education, and the study is coordinated by the Alzheimer's Disease Cooperative Study at the University of California, San Diego. ADNI data are disseminated by the Laboratory for Neuro Imaging at the University of California, Los Angeles. This research was also supported by NIH grants P30 AG010129, K01 AG030514, and the Dana Foundation.

APPENDIX

Optimization for Complete Models

We first focus on formulation (10), which is clearly a non-convex optimization problem. Gasso et al. has shown in (Gasso et al., 2009) that the ℓ_q-regularized least squares problem with $q < 1$ can be efficiently solved using the difference of convex functions (DC) algorithm (Tao and An, 1997). The DC decomposition presented in (Gasso et al., 2009) requires the regularization term to be a concave function. However, this is not the case for our formulation, according to the following proposition:

Proposition 1 *Let* $\boldsymbol{f}(\boldsymbol{\beta})=\|\boldsymbol{\beta}\|_{\mathbf{2}}^{\frac{\mathbf{2}}{\mathbf{3}}}$. *Then* $\boldsymbol{f}$ *is neither convex nor concave w.r.t.* |β| *unless* β *is a scalar, where* |·| *denotes the absolute value.*

Proof. The proof is carried out by computing the Hessian of *f*. Without loss of generality, we assume that β ≠ 0. It can be shown that:

$$\frac{\partial f}{\partial|\beta_i|}=\frac{2}{3}\|\boldsymbol{\beta}\|_2^{-\frac{4}{3}}|\beta_i|,\quad \frac{\partial^2 f}{\partial|\beta_i|\partial|\beta_j|}=-\frac{8}{9}\|\boldsymbol{\beta}\|_2^{-\frac{10}{3}}|\beta_i\beta_j|+\mathbf{1}_{\{i=j\}}\cdot\frac{2}{3}\|\boldsymbol{\beta}\|_2^{-\frac{4}{3}},$$

where $\mathbf{1}_{\{i=j\}}$ is the indicator function. It is clear that, unless β is a scalar, in which case it is obvious that *f* is a concave function, $\frac{\partial^2 f}{\partial |\beta_i|^2}$ can be either positive or negative. In other words, the sign of the diagonal elements of the Hessian of *f* can be either positive or negative, which means that *f* is neither convex nor concave.

To employ the DC algorithm, we need to avoid the non-concavity of the regularization item. We introduce new variables t_i, $i = 1,2, \cdots, S$ and transform (9) into the following formulation:

$$\begin{array}{ll}\min\limits_{\boldsymbol{\beta},\boldsymbol{t}} & \frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\sum_{i=1}^{S}\nu_i t_i^{\frac{2}{3}}\\ s.t. & \|\boldsymbol{\beta}_i\|_2 \le t_i, i=1,2,\cdots,S.\end{array} \qquad (16)$$

It is clear that (16) is equivalent to the original formulation (9), but the regularization term in (16) is concave with respect to t_i, as shown in Proposition 1. We apply the DC algorithm, i.e., for each $t_i^{\frac{2}{3}}$, we rewrite it as the difference of two convex functions as follows:

$$t_i^{\frac{2}{3}}=t_i-\left(t_i-t_i^{\frac{2}{3}}\right).$$

Therefore, (16) becomes:

$$\begin{array}{ll}\min\limits_{\boldsymbol{\beta},\boldsymbol{t}} & \frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\sum_{i=1}^{S}\nu_i t_i-\sum_{i=1}^{S}\nu_i\left(t_i-t_i^{\frac{2}{3}}\right)\\ s.t. & \|\boldsymbol{\beta}_i\|_2 \le t_i, i=1,2,\cdots,S.\end{array} \qquad (17)$$

Next we replace the second convex item $t_i-t_i^{\frac{2}{3}}$ by its affine minorant at the previous iteration. Specifically, suppose at the previous iteration the value of t_i is $\hat{t}_\iota$; now we approximate $t_i-t_i^{\frac{2}{3}}$ by its first-order Taylor expansion at $\hat{t}_\iota$ as follows:

$$\hat{t}_\iota-\hat{t}_\iota^{\frac{2}{3}}+(1-\frac{2}{3}\hat{t}_\iota^{-\frac{1}{3}})(t_i-\hat{t}_\iota).$$

Plugging the above expression back to (17) and dropping the constant, we get:

$$\begin{array}{ll}\min\limits_{\boldsymbol{\beta},\boldsymbol{t}} & \frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\sum_{i=1}^{S}\frac{2}{3}\hat{t}_\iota^{-\frac{1}{3}}\nu_i t_i\\ s.t. & \|\boldsymbol{\beta}_i\|_2 \le t_i, i=1,2,\cdots,S.\end{array} \qquad (18)$$

Since ν_i and $\hat{t}_\iota$ are nonnegative, all constraints in (18) must be active at the optimal points. Thus, (18) is equivalent to the following group lasso problem:

$$\min_{\boldsymbol{\beta}}\frac{1}{2}\left\|\boldsymbol{y}-\sum_{i=1}^{S}\boldsymbol{X}_i\boldsymbol{\beta}_i\right\|_2^2+\sum_{i=1}^{S}\frac{2}{3}\hat{t}_\iota^{-\frac{1}{3}}\nu_i\|\boldsymbol{\beta}_i\|_2.$$

After β is obtained, we update $\hat{t}_\iota$ with $\|\beta_i\|_2$ and continue the iteration until convergence. Notice that $\hat{t}_\iota^{-\frac{1}{3}}$ can be very large if $\|\beta_i\|_2$ is small. For numerical stability, we add a smoothing term θ to each $\hat{t}_\iota$ as suggested by (Gasso et al., 2009). The overall procedure is summarized in Algorithm 1.

Algorithm 1

The proposed DC algorithm for solving (10)

```
Input: $\boldsymbol{X}, \boldsymbol{y}, \nu$
Output: solution $\beta$ to (10)
Initialize $\theta, \mu_i^{(0)}, i = 1,2, \cdots, S$
for $k = 1,2, \cdots$ do
    Update $\beta$ and $\mu_i$ by:
        $\hat{\beta}^k = \min_{\boldsymbol{\beta}} \frac{1}{2} \left\| \boldsymbol{y} - \sum_{i=1}^{S} \boldsymbol{X}_i \boldsymbol{\beta}_i \right\|_2^2 + \sum_{i=1}^{S} \mu_i^{k-1} \|\boldsymbol{\beta}_i\|_2$
        $\mu_i^k = \frac{2}{3} \nu_i \left( \left\| \hat{\boldsymbol{\beta}}_\iota^{\,k} \right\|_2 + \theta \right)^{-\frac{1}{3}}, i = 1, 2, \cdots, S.$
    If the stopping criterion is satisfied then
        return $\beta = \beta^{\hat{k}}$
    end if
end for
```

Remark 7. Model (9) can be solved in exactly the same way as above. The only difference is that in each iteration we need to solve a weighted lasso problem to get $\beta^{(\hat{k})}$.

A Two-Stage Approach

Here, we propose a two-stage approach, which performs feature-level and source-level analyses separately. A nice feature of the two-stage approach is that it can naturally be extended to deal with the block-wise missing data using existing missing data estimation techniques.

1 Complete Data

The proposed two-stage approach on the complete multi-source data can be viewed as a simplified version of our unified learning framework discussed in Section 3. Instead of performing bi-level analysis in one optimization problem, we aim to divide the learning problem into two stages. Given a multi-source data set, we first train a base model on each individual data source, and the base model is applied to produce prediction scores for the corresponding samples for this data source; thus each data source is represented as a single column of scores. All data sources together are then represented as a matrix of prediction scores, which are treated as newly derived features to train our final classifier.

We formally describe our two-stage method as follows. Denote $\boldsymbol{X}_s^i$ as the i^{th} sample from the s^{th} data source. The goal is to derive a prediction score matrix $\boldsymbol{A} \in \mathbb{R}^{m \times S}$ from the original data set. Details are given below:

Base Model Training Step—We first choose a learning algorithm $\mathscr{L}$, based on which a prediction model is constructed for each data source:

$$\mathscr{M}_s = \mathscr{L}(\boldsymbol{X_s}, y), s=1, \ldots, S.$$

These base models are then used to construct a prediction score matrix $\boldsymbol{A}$ given by:

$$\boldsymbol{A}_{i,s} = \mathscr{M}_s(\boldsymbol{X_s^i}),$$

where $\mathscr{M}_s\left(\boldsymbol{X_s^i}\right)$ is the prediction score of model $\mathscr{M}_S$ on feature vector X_s^i.

Final Model Training Step—In this step, we treat **A** as the newly derived feature matrix of the original multi-source data set. The final model $\mathscr{M}$ is learned using ($\boldsymbol{A}$, y) so that the sources are integrated.

Prediction of Unlabeled Samples—Given a set of unlabeled data $U = [U_1, \ldots U_S] \in \mathbb{R}^{t \times n}$, we first derive a feature matrix $\boldsymbol{B} = [\mathscr{M}_1(U_1), \ldots, \mathscr{M}_S(U_S)]$. We then apply the final model $\mathscr{M}$ to the feature matrix $\boldsymbol{B}$ to obtain the prediction of the unlabeled set.

2 Incomplete Data

Next, we show that the simple design of this two-stage approach facilitates the extension to the case with block-wise missing data. In the two-stage scheme, we first train a base model on each individual data source using all available samples, and the base model is applied to produce prediction scores for this data source; thus each data source is represented by a single column of (incomplete) scores. A missing value estimation method is applied to obtain a complete set of model scores, which are treated as newly derived features to train our final classifier. The overview of this method is demonstrated in Figure 10.

As shown in Figure 10, instead of estimating the complete blocks of missing values, we only need to impute missing prediction scores, which is a less challenging problem. We thus denote this two-stage scheme as the model score completion method (ScoreComp).

For notation simplicity, denote the set $sc(s) \subset \{1,2, \ldots, m\}$ as the available samples for the s^{th} data source. If $\boldsymbol{X_s^i}$ exists, we clearly have $i \in sc(s)$. A crucial step of the ScoreComp method is to obtain a completed prediction score matrix $\tilde{\boldsymbol{A}} \in \mathbb{R}^{m \times s}$ from the incomplete data set. The details of ScoreComp are as follows.

Base Model Training Step—We first choose a learning algorithm $\mathscr{L}$, based on which a prediction model is constructed for each data source:

$$\mathscr{M}_s=\mathscr{L}\{(X_s^i, y_i)|i \in sc(s)\}, s=1,\ldots,S.$$

These base models are then used to construct an incomplete prediction score matrix $\hat{\boldsymbol{A}}$ given by:

$$\hat{\boldsymbol{A}}_{i,s}=\begin{cases} \mathscr{M}_s(\boldsymbol{X_s^i}) & i \in sc(s) \\ \text{NaN} & \text{o/w} \end{cases},$$

where $\mathscr{M}_s\left(\boldsymbol{X_s^i}\right)$ is the prediction score of model $\mathscr{M}_S$ on feature vector X_s^i.

Score Imputation Step—After converting the original data matrix into *S* incomplete vectors of prediction scores, we have transformed the block-wise missing pattern into a random missing pattern. Traditional imputation methods can now be readily applied. We choose a missing value estimation algorithm ε such that $\tilde{\boldsymbol{A}}$ = ε ($\hat{\boldsymbol{A}}$), where $\tilde{A}$ is the completed prediction score matrix.

Final Model Training Step—In this step, we treat $\tilde{\boldsymbol{A}}$ as the newly derived feature matrix of the original multi-source data set. The final model is $\mathscr{M}$ learned using ($\tilde{\boldsymbol{A}}$, *y*) so that the sources are integrated.

Prediction of Unlabeled Samples—Given a set of unlabeled data $U=\{U_s^j|j \in sc_U(s)\}$, where $sc_U(s)$ denotes the available samples for the s^{th} data source in the unlabeled set. We first derive an incomplete feature matrix $\hat{B}$ by:

$$\hat{B}_{j,s}=\begin{cases} \mathscr{M}_s(U_s^j) & j \in sc(s) \\ \text{NaN} & \text{o/w} \end{cases}.$$

We then combine $\hat{B}$ with the previously obtained imputed matrix $\tilde{A}$ such that missing data imputation is performed:

$$C=\varepsilon\left(\begin{bmatrix} \tilde{A} \\ \hat{B} \end{bmatrix}\right).$$

Finally, by extracting the lower part of matrix *C*, we can obtain the derived feature matrix $\tilde{B}$ for the unlabeled data set. We then apply the final model $\mathscr{M}$ to $\tilde{B}$ to obtain the prediction of the unlabeled set.

REFERENCES

Ando, RK.; Zhang, T. Two-view feature generation model for semi-supervised learning; Proceedings of the 24th International Conference on Machine Learning (ICML); 2007. p. 25-32.

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

Bach F. Optimization with Sparsity-Inducing Penalties. Foundations and Trends text registered in Machine Learning. 2011; 4:1–106.

Beck A, Teboulle M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences. 2009; 2:183–202.

Bickel PJ, Ritov Y, Tsybakov AB. Simultaneous analysis of lasso and dantzig selector. The Annals of Statistics. 2009; 37:1705–1732.

Breheny P, Huang J. Penalized methods for bi-level variable selection. Statistics and its interface. 2009; 2:369–369. [PubMed: 20640242]

Brookmeyer R, Johnson E, Ziegler-Graham K, Arrighi HM. Forecasting the global burden of Alzheimer’s disease. Alzheimer's & dementia. 2007; 3:186–191.

Calhoun VD, Liu J, Adalı T. A review of group ICA for fMRI data and ICA for joint inference of imaging, genetic, and ERP data. Neuroimage. 2009; 45:S163. [PubMed: 19059344]

Crammer K, Kearns M, Wortman J. Learning from multiple sources. The Journal of Machine Learning Research. 2008; 9:1757–1774.

Culp M, Michailidis G, Johnson K. On Multi-view learning with additive models. The Annals of Applied Statistics. 2009; 3:292–318.

Duda RO, Hart PE, Stork DG. Pattern Classification. 1997

Efron B, Hastie T, Johnstone I, Tibshirani R. Least angle regression. The Annals of statistics. 2004; 32:407–499.

Fan Y, Resnick SM, Wu X, Davatzikos C. Structural and functional biomarkers of prodromal Alzheimer's disease: a high-dimensional pattern classification study. Neuroimage. 2008; 41:277–285. [PubMed: 18400519]

Fjell AM, Walhovd KB, Fennema-Notestine C, McEvoy LK, Hagler DJ, Holland D, Brewer JB, Dale AM. CSF biomarkers in prediction of cerebral and clinical change in mild cognitive impairment and Alzheimer's disease. J Neurosci. 2010; 30:2088–2101. [PubMed: 20147537]

Friedman J, Hastie T, Tibshirani R. A note on the group Lasso and a sparse group lasso. Arxiv preprint arXiv:1001.0736. 2010

Gasso G, Rakotomamonjy A, Canu S. Recovering sparse signals with a certain family of nonconvex penalties and DC programming. Signal Processing, IEEE Transactions on. 2009; 57:4686–4698.

Hinrichs C, Singh V, Xu G, Johnson SC. Predictive markers for AD in a multi-modality framework: an analysis of MCI progression in the ADNI population. Neuroimage. 2011; 55:574–589. [PubMed: 21146621]

Huopaniemi I, Suvitaival T, Nikkilä J, Orešič M, Kaski S. Multivariate multi-way analysis of multi-source data. Bioinformatics. 2010; 26:i391–i398. [PubMed: 20529933]

Jack CR, Bernstein MA, Fox NC, Thompson P, Alexander G, Harvey D, Borowski B, Britson PJ, L Whitwell J, Ward C, et al. The Alzheimer's disease neuroimaging initiative (ADNI): MRI methods. Journal of Magnetic Resonance Imaging. 2008; 27:685–691. [PubMed: 18302232]

Lanckriet GRG, De Bie T, Cristianini N, Jordan MI, Noble WS. A statistical framework for genomic data fusion. Bioinformatics. 2004; 20:2626–2635. [PubMed: 15130933]

Landau SM, Harvey D, Madison CM, Reiman EM, Foster NL, Aisen PS, Petersen RC, Shaw LM, Trojanowski JQ, Jack CR Jr, Weiner MW, Jagust WJ. Comparing predictors of conversion and decline in mild cognitive impairment. Neurology. 2010; 75:230–238. [PubMed: 20592257]

Liu, J.; Ji, S.; Ye, J. Multi-task feature learning via efficient $l_{2,1}$-norm minimization; Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence; 2009. p. 339-348.

Mazumder R, Hastie T, Tibshirani R. Spectral regularization algorithms for learning large incomplete matrices. The Journal of Machine Learning Research. 2010; 11:2287–2322.

Mueller SG, Weiner MW, Thal LJ, Petersen RC, Jack C, Jagust W, Trojanowski JQ, Toga AW, Beckett L. The Alzheimer's disease neuroimaging initiative. Neuroimaging Clinics of North America. 2005; 15:869–869. [PubMed: 16443497]

Quattoni, A.; Carreras, X.; Collins, M.; Darrell, T. An efficient projection for $l_{1,\infty}$ regularization; Proceedings of the 26th Annual International Conference on Machine Learning (ICML); 2009. p. 857-864.

Tao PD, An LTH. Convex analysis approach to dc programming: Theory, algorithms and applications. Acta Math. Vietnam. 1997; 22:289–355.

Tibshirani R. Regression shrinkage and selection via the Llsso. Journal of the Royal Statistical Society. Series B (Methodological). 1996:267–288.

Troyanskaya OG, Dolinski K, Owen AB, Altman RB, Botstein D. A Bayesian framework for combining heterogeneous data sources for gene function prediction (in Saccharomyces cerevisiae). Proceedings of the National Academy of Sciences. 2003; 100:8348–8348.

Turlach BA, Venables WN, Wright SJ. Simultaneous variable selection. Technometrics. 2005; 47:349–363.

Tzourio-Mazoyer N, Landeau B, Papathanassiou D, Crivello F, Etard O, Delcroix N, Mazoyer B, Joliot M. Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. Neuroimage. 2002; 15:273–289. [PubMed: 11771995]

Vemuri P, Wiste HJ, Weigand SD, Shaw LM, Trojanowski JQ, Weiner MW, Knopman DS, Petersen RC, Jack CR Jr. MRI and CSF biomarkers in normal, MCI, and AD subjects: predicting future clinical change. Neurology. 2009; 73:294–301. [PubMed: 19636049]

Walhovd KB, Fjell AM, Brewer J, McEvoy LK, Fennema-Notestine C, Hagler DJ Jr, Jennings RG, Karow D, Dale AM. Combining MR imaging, positron-emission tomography, and CSF biomarkers in the diagnosis and prognosis of Alzheimer disease. AJNR Am J Neuroradiol. 2010a; 31:347–354. [PubMed: 20075088]

Walhovd KB, Fjell AM, Dale AM, McEvoy LK, Brewer J, Karow DS, Salmon DP, Fennema-Notestine C. Multi-modal imaging predicts memory performance in normal aging and cognitive decline. Neurobiol Aging. 2010b; 31:1107–1121. [PubMed: 18838195]

Wang H, Nie F, Huang H, Risacher SL, Saykin AJ, Shen L. Identifying disease sensitive and quantitative trait-relevant biomarkers from multidimensional heterogeneous imaging genetics data via sparse multimodal multitask learning. Bioinformatics. 2012; 28:i127–i136. [PubMed: 22689752]

Xiang, S.; Shen, X.; Ye, J. Efficient Sparse Group Feature Selection via Nonconvex Optimization; The 30th International Conference on Machine Learning (ICML); 2013.

Xu, Z.; King, I.; Lyu, MR. Web page classification with heterogeneous data fusion; Proceedings of the 16th international conference on World Wide Web; 2007. p. 1171-1172.

Yang, H.; Xu, Z.; King, I.; Lyu, M. Online learning for group lasso; Proceedings of the 27th International Conference on Machine Learning (ICML); 2010.

Ye, J.; Chen, K.; Wu, T.; Li, J.; Zhao, Z.; Patel, R.; Bae, M.; Janardan, R.; Liu, H.; Alexander, G., et al. Heterogeneous data fusion for alzheimer's disease study; Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining; 2008. p. 1025-1033.

Ye J, Liu J. Sparse methods for biomedical data. ACM SIGKDD Explorations Newsletter. 2012; 14:4–15.

Yuan L, Wang Y, Thompson PM, Narayan VA, Ye J. Multi-source feature learning for joint analysis of incomplete multiple heterogeneous neuroimaging data. NeuroImage. 2012; 61:622–632. [PubMed: 22498655]

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

Zhang D, Shen D. Multi-modal multi-task learning for joint prediction of multiple regression and classification variables in Alzheimer's disease. NeuroImage. 2012; 59:895–907. [PubMed: 21992749]

Zhang D, Wang Y, Zhou L, Yuan H, Shen D. Multimodal classification of Alzheimer's disease and mild cognitive impairment. Neuroimage. 2011; 55:856–867. [PubMed: 21236349]

Zhao P, Yu B. On model selection consistency of Lasso. The Journal of Machine Learning Research. 2006; 7:2541–2563.

Research Highlights

- Ability to fuse large multi-modal datasets for classification when large segments of the data are missing
- A unified framework to perform both feature-level and source-level analysis simultaneously
- Efficient optimization algorithms for both models with complete and incomplete data
- Detailed comparison between different methods on clinical group classification problems

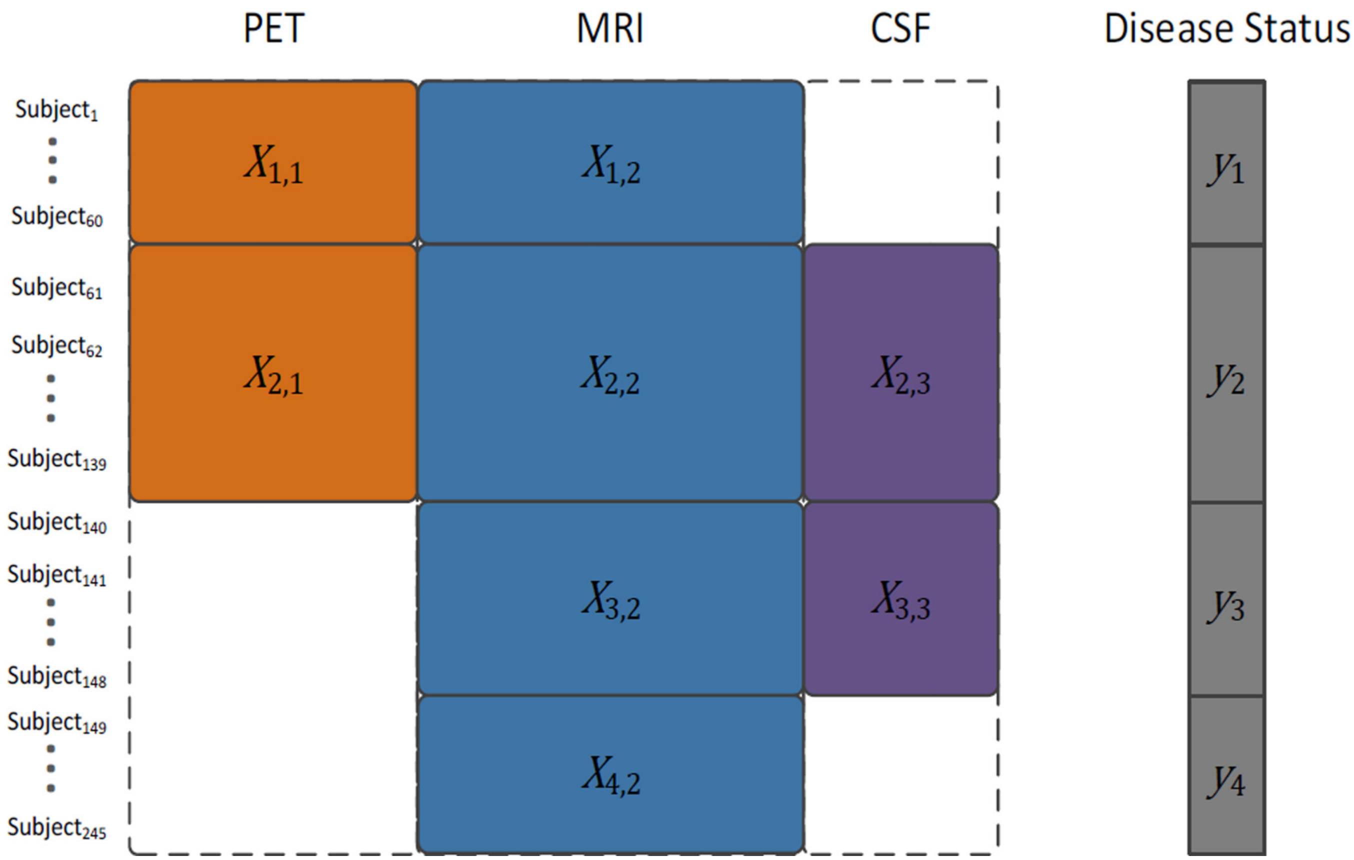

Figure 1.

An illustration of an incomplete multi-source data with three sources. In this example, there are 245 participants in total and 3 types of measurements (PET, MRI and CSF) represented in different colors. The blank region indicates that data is missing from the corresponding source. In the example shown above, some participants 1–139 have PET and MRI but lack CSF information, while other participants 149–245 have MRI data only.

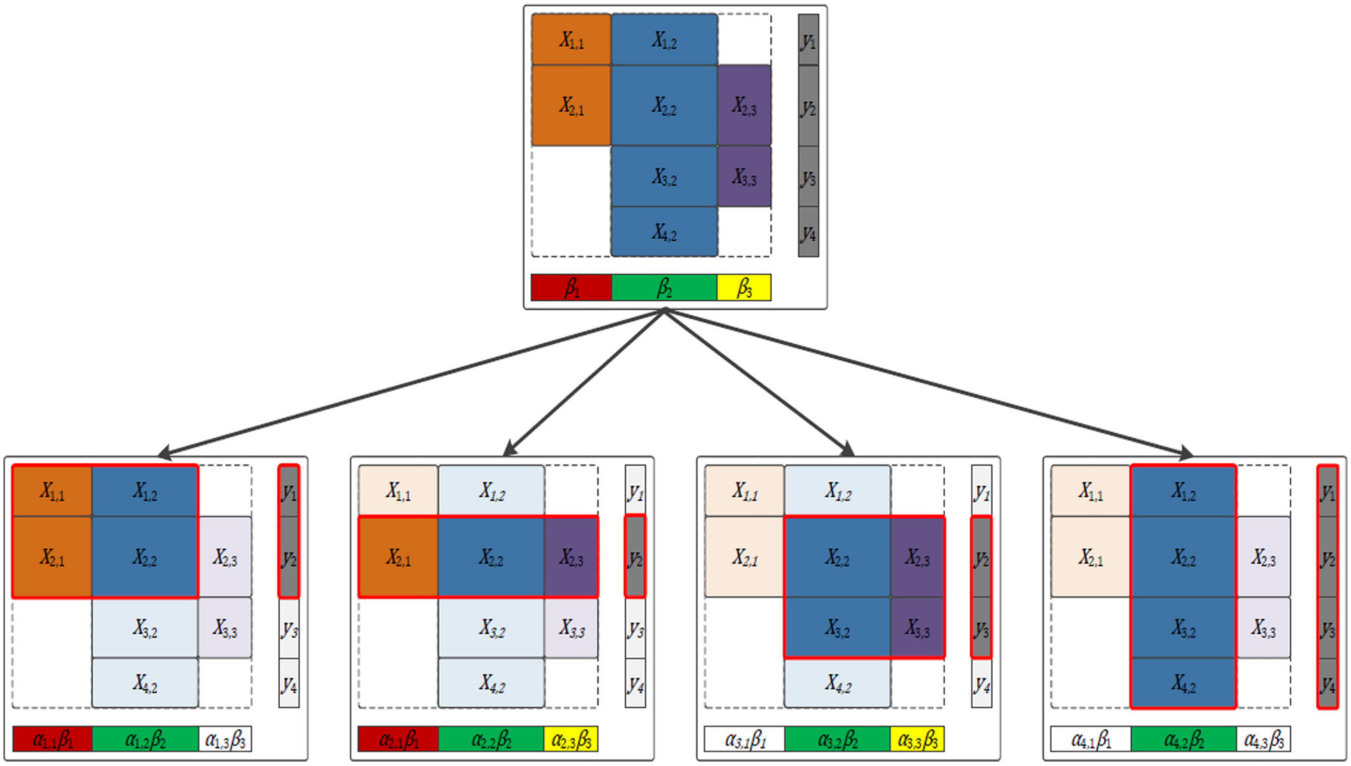

Figure 2.

Illustration of the proposed learning model. The data set is partitioned into four groups according to the availability of data sources, as highlighted by the red boxes. The goal is to learn three models β_1, β_2 and β_3 for each data source as well as the coefficient α that combines them. Notice that, for the *i*th data source, β_i remains identical while α may vary across different groups.

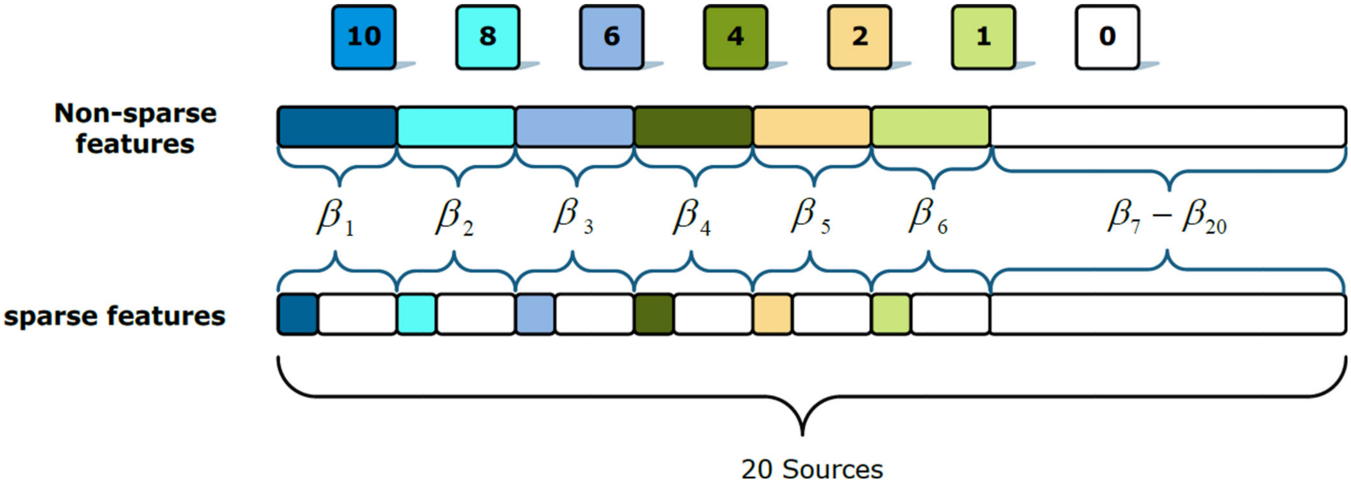

Figure 3.

Two scenarios of the underlying true model β: the top one corresponds to the situation of non-sparse features and the bottom one represents the situation of sparse features. The white block represents zero elements, while the non-zero values are represented by different colors, indicated in the first row.

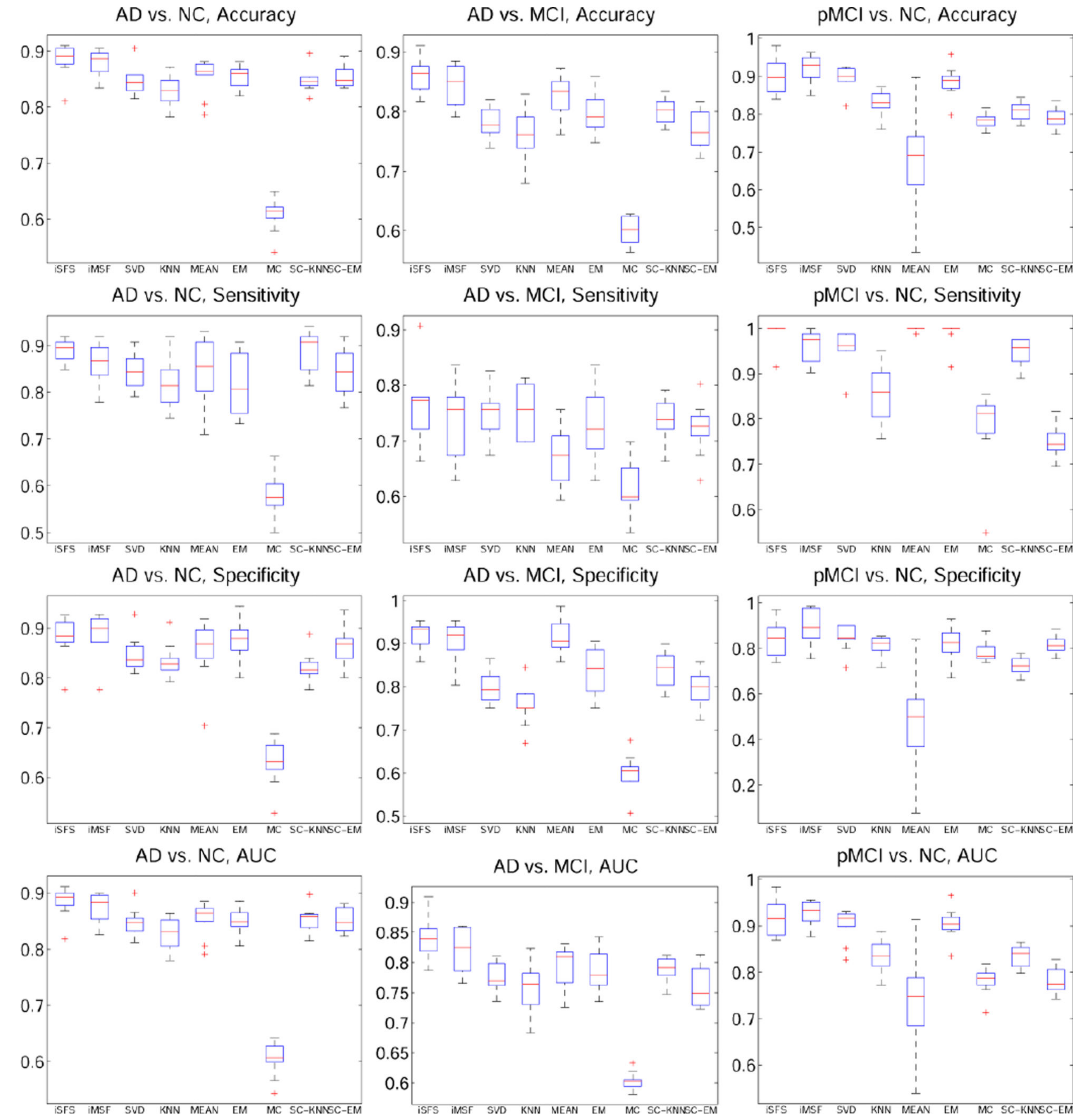

Figure 4.

Error bars of the classification results with 50% data for training.

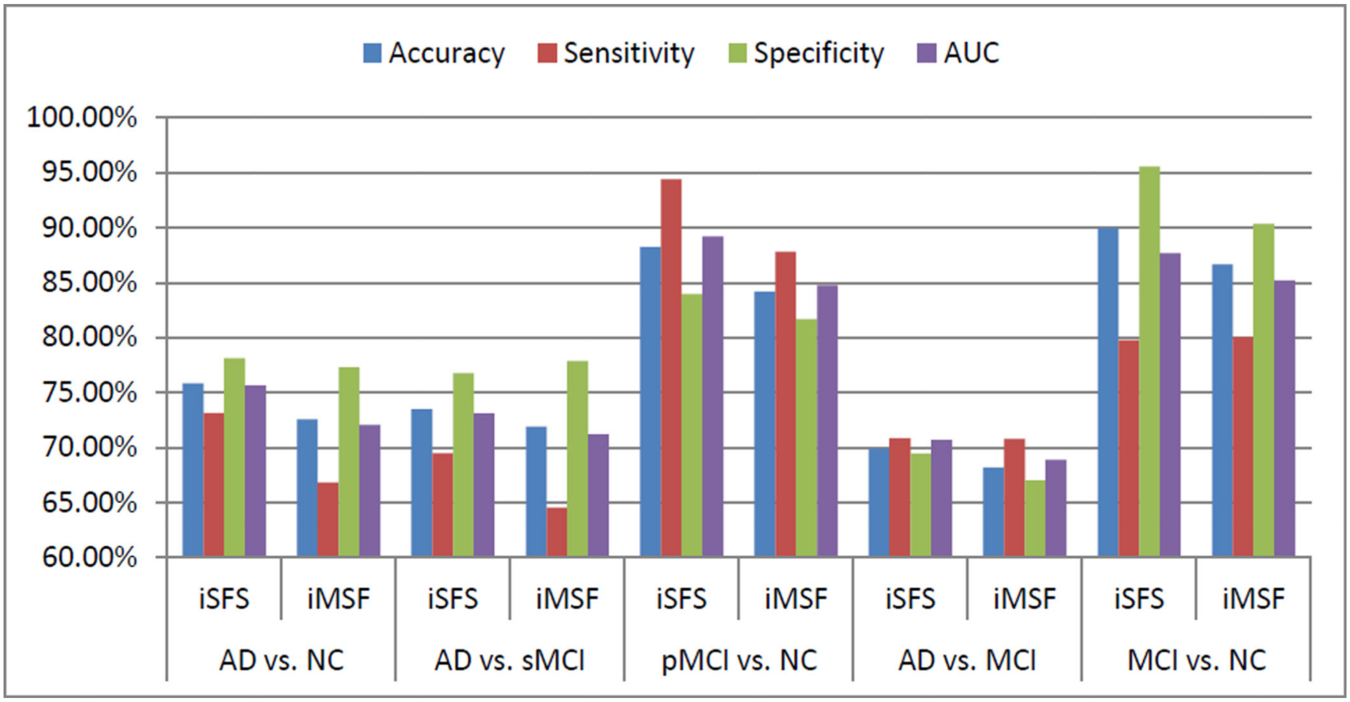

Figure 5.

Classification results are shown, for iSFS and iMSF on the ADNI data set, with additional noisy data sources.

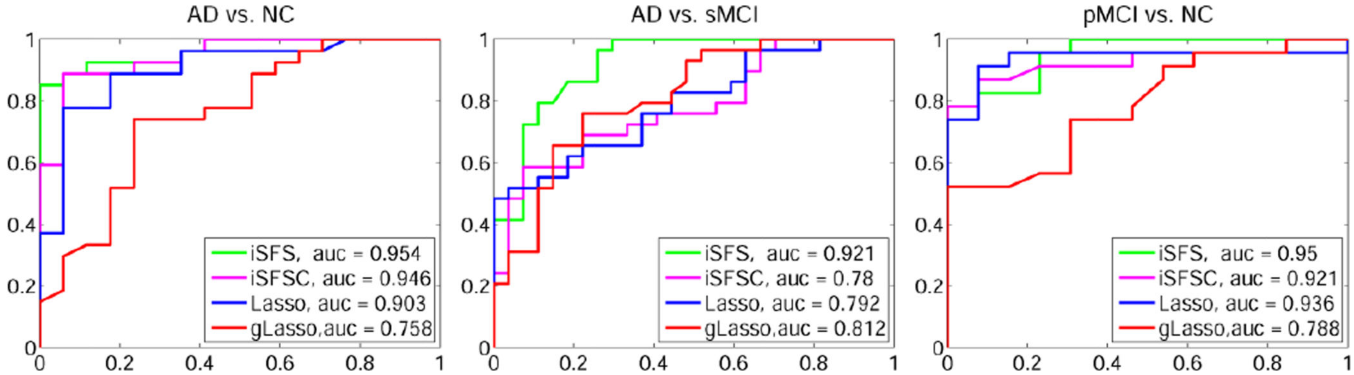

Figure 6.

ROC curves given by iSFS (on both complete and incomplete data), lasso and group lasso. Except for iSFS, the classification is carried out on a subset of the ADNI data set, where all the samples have four data sources available. For iSFS, it is evaluated on the whole incomplete data set.

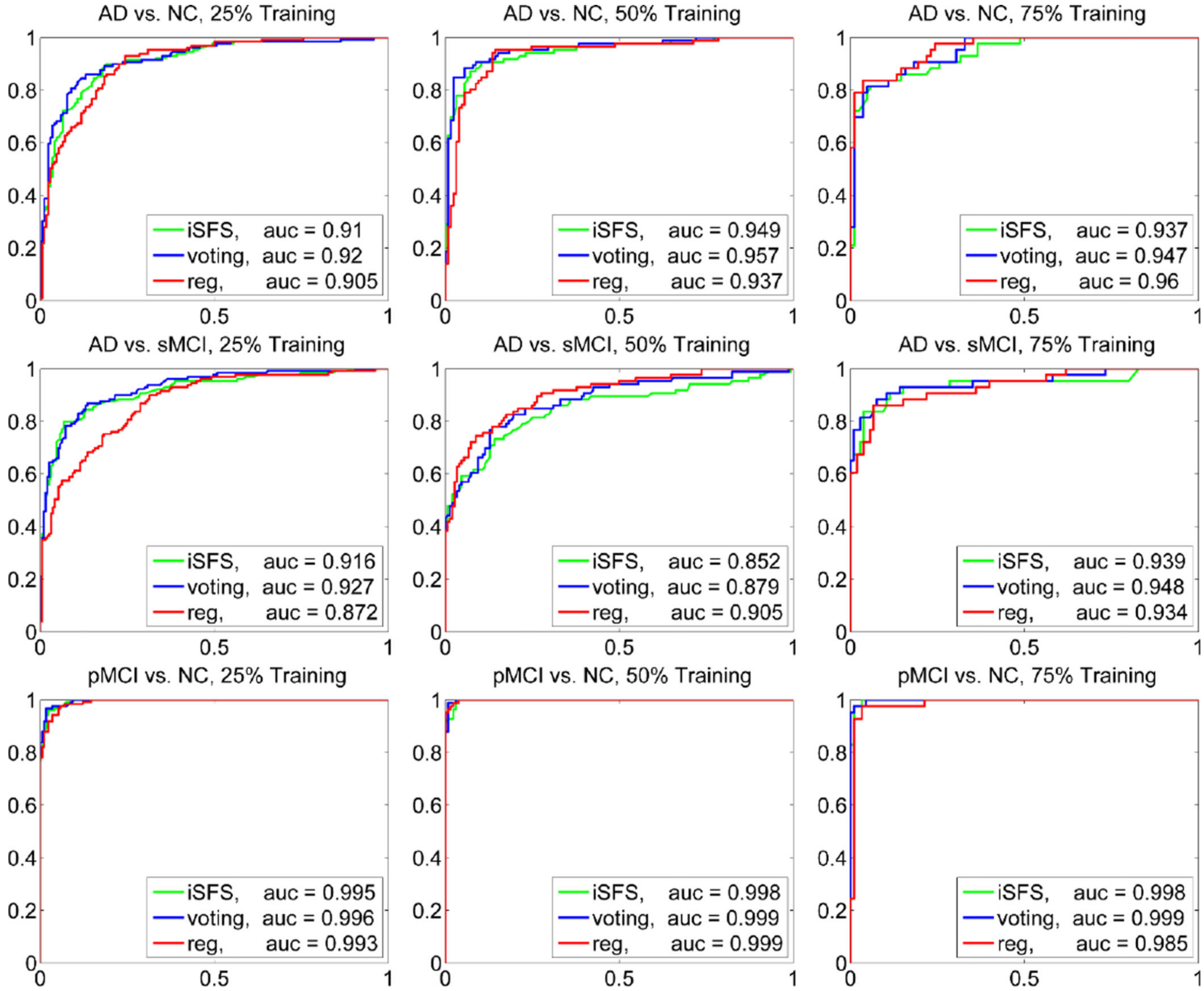

Figure 7.

ROC curves of the ensemble methods. The ratio of the training set varies from 25% to 75% and the performance on three tasks: AD vs. normal controls, AD vs. stable MCI and progressive MCI vs. normal controls, are reported. The *blue curve* denotes the majority voting approach, and the linear regression ensemble method is represented by the *red curve*.

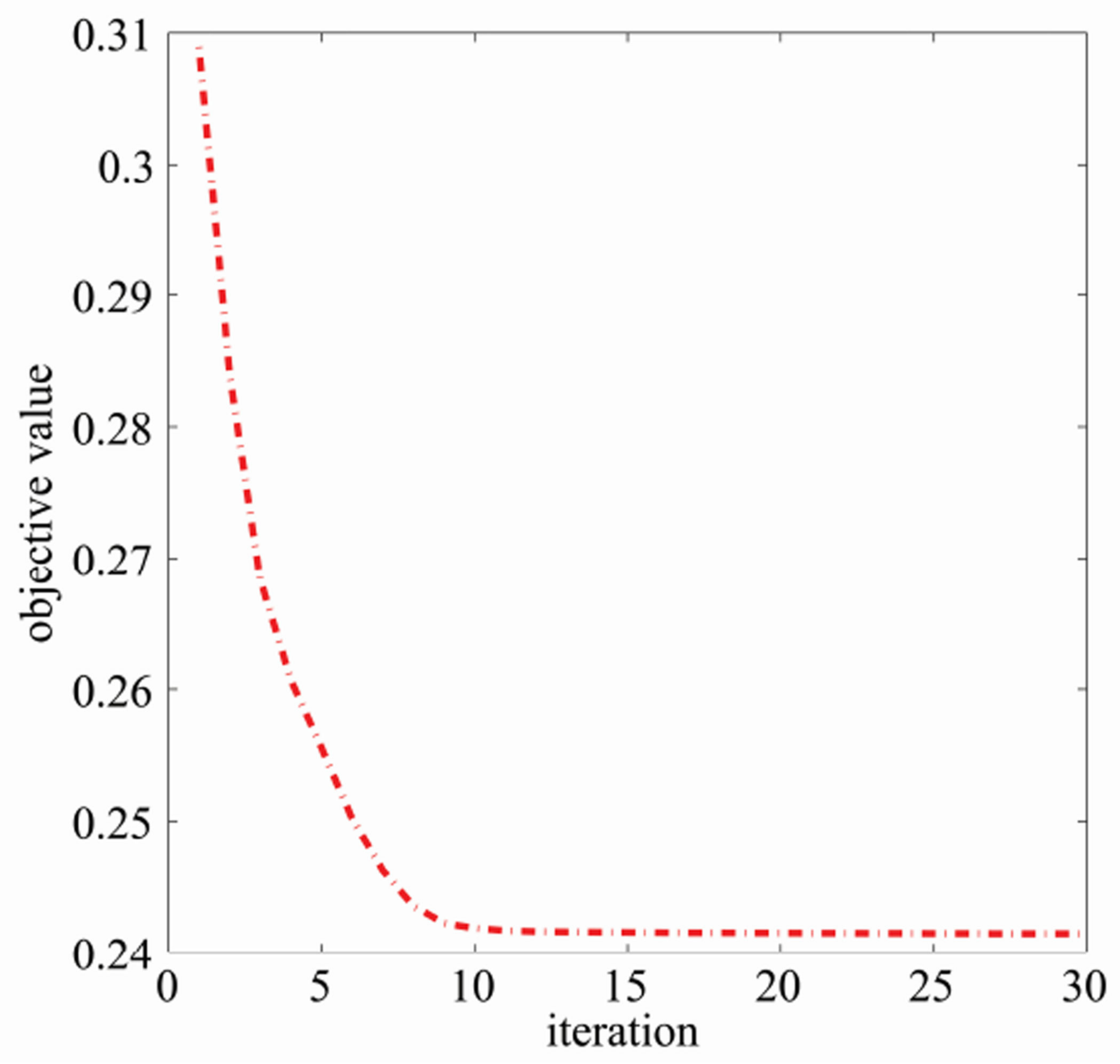

Figure 8.

Illustration of the convergence of Algorithm 1. The x-axis denotes the number of iterations and the y-axis denotes the objective value of (11). We can observe from the figure above that the proposed algorithm converges quickly after the first few iterations.

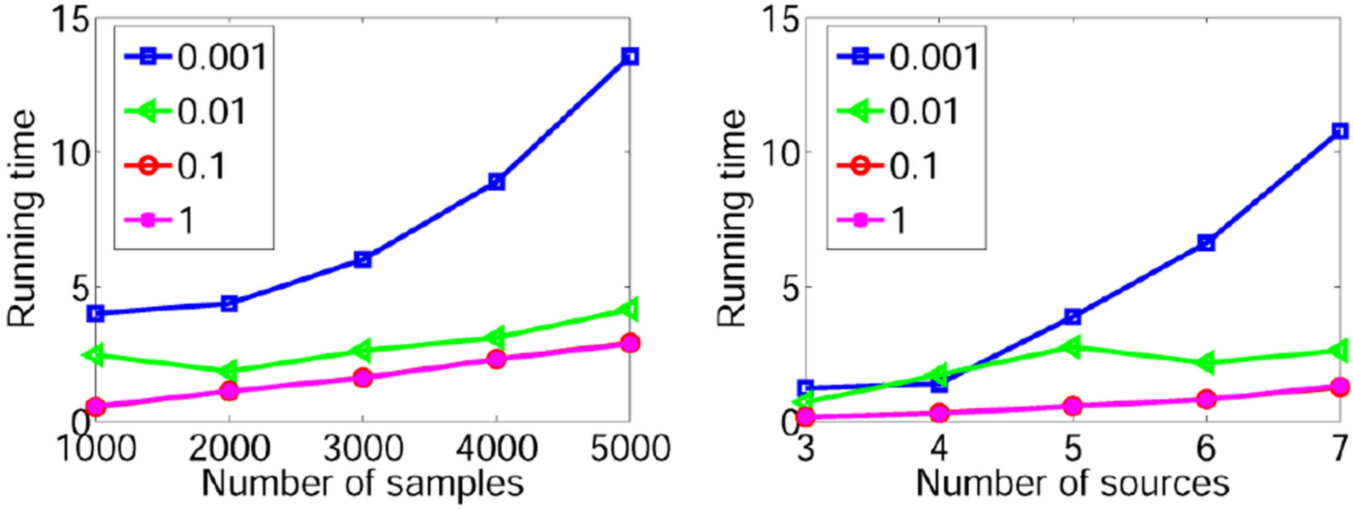

Figure 9.

Running time (in seconds) of the proposed algorithm with increasing number of samples and number of sources on synthetic data.

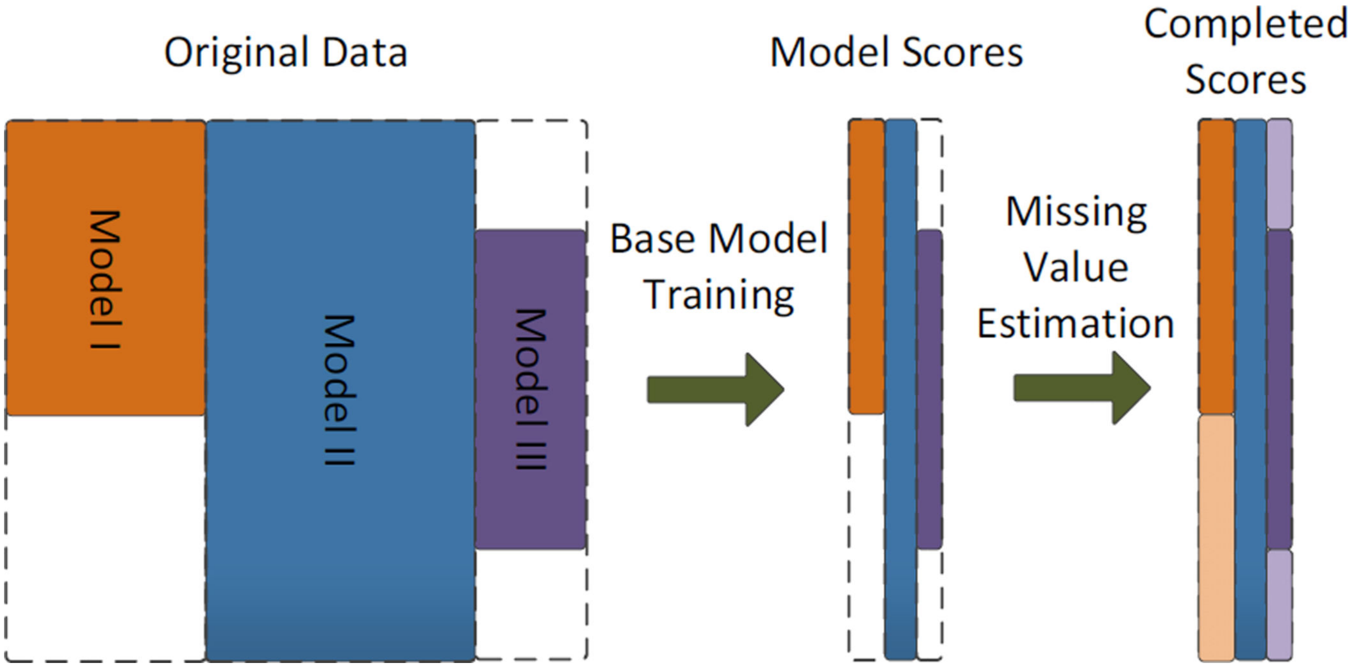

Figure 10.

Illustration of the two-stage scheme for block-wise incomplete data. We first train a base model on each individual data source using the available samples, and the base model is applied to produce prediction scores for this data source; thus each data source is represented by a single column of (incomplete) scores. A missing value estimation method is applied to obtain a complete set of model scores, which are treated as newly derived features to train our final classifier.

Table 1

Statistics of the ADNI data set and the data sources used in our evaluations, where AD, pMCI, sMCI and NC stand for Alzheimer’s disease patients, progressive mild cognitive impairment patients, stable mild cognitive impairment patients, and normal controls respectively.

	AD	pMCI	sMCI	NC	# of Samples	# of Measures
Proteomics	112	163	233	58	566	147
PET	76	70	100	81	327	116
MRI	138	142	177	191	648	305
CSF	100	84	111	114	409	3

Table 2

Performance on complete, synthetic data. The MSE denotes the mean squared error of prediction on the test set, and Esti stands for the parameter estimation error. For the scenario of sparse features, the underlying true model has 6 groups and 18 features, while for the situation of non-sparse features, the true model takes 6 groups and 60 features. All results are averaged over 10 replications.

Methods	Sparse features				Non-sparse features			
	MSE	Esti	#Group	#Feature	MSE	Esti	#Group	#Feature
lasso	256.84	257.47	17.10	71.70	2007.61	1617.81	19.30	141.30
glasso	165.55	162.35	13.50	135.00	669.80	493.23	12.40	124.00
sglasso	71.69	80.93	13.10	77.90	729.22	552.79	13.80	137.90
frac(1,2)	**17.04**	**15.36**	**6.10**	**30.60**	1618.47	1245.87	12.60	94.00
frac(2,1)	146.15	131.04	6.40	64.00	**242.27**	**221.01**	**5.10**	**51.00**

Table 3

Classification results of AD patients versus normal controls with 10% data for training. All results are averaged over 10 replications.

	iSFS	**iMSF**	**SVD**	**KNN**	**Mean**	**EM**	**MC**	**SC-KNN**	**SC-EM**
Accuracy	**0.8103**	0.7857	0.7756	0.7668	0.7789	0.8089	0.5957	0.7774	0.8092
Sensitivity	0.8077	0.7671	0.7770	0.7161	0.7845	0.7963	0.5710	**0.8252**	0.7890
Specificity	0.8124	0.8005	0.7746	0.8072	0.7744	0.8189	0.6155	0.7392	**0.8253**
AUC	**0.8101**	0.7838	0.7758	0.7617	0.7795	0.8076	0.5932	0.7822	0.8071

Table 4

Classification results of AD patients versus stable MCI patients with 10% data for training. All results are averaged over 10 replications.

	iSFS	iMSF	SVD	KNN	Mean	EM	MC	SC-KNN	SC-EM
Accuracy	**0.7489**	0.7172	0.6942	0.6774	0.7338	0.7174	0.6234	0.6634	0.6715
Sensitivity	**0.7032**	0.6910	0.6510	0.6819	0.6163	0.6323	0.6135	0.6271	0.6974
Specificity	0.7816	0.7359	0.7250	0.6742	**0.8177**	0.7782	0.6304	0.6894	0.6530
AUC	**0.7424**	0.7135	0.6880	0.6781	0.7170	0.7052	0.6220	0.6582	0.6752

Table 5

Classification results of progressive MCI patients versus normal controls with 10% data for training. All results are averaged over 10 replications.

	iSFS	iMSF	SVD	KNN	Mean	EM	MC	SC-KNN	SC-EM
Accuracy	**0.8754**	0.8611	0.7280	0.7272	0.7889	0.8027	0.7740	0.6930	0.7304
Sensitivity	0.9361	0.9190	0.7222	0.6381	**0.9531**	0.8281	0.7728	0.8490	0.7177
Specificity	**0.8297**	0.8174	0.7323	0.7944	0.6651	0.7836	0.7749	0.5754	0.7400
AUC	**0.8829**	0.8682	0.7273	0.7162	0.8091	0.8059	0.7738	0.7222	0.7288

Table 6

Classification results of AD patients versus normal controls with 50% data for training. All results are averaged over 10 replications.

	iSFS	iMSF	SVD	KNN	Mean	EM	MC	SC-KNN	SC-EM
Accuracy	**0.8848**	0.8782	0.8469	0.8374	0.8540	0.8536	0.6085	0.8469	0.8536
Sensitivity	**0.8895**	0.8733	0.8465	0.8407	0.8465	0.8163	0.5779	0.8849	0.8395
Specificity	**0.8816**	**0.8816**	0.8472	0.8352	0.8592	0.8792	0.6296	0.8208	0.8632
AUC	**0.8856**	0.8774	0.8469	0.8379	0.8529	0.8477	0.6038	0.8528	0.8514

Table 7

Classification results of AD patients versus stable MCI patients with 50% data for training. All results are averaged over 10 replications.

	iSFS	iMSF	SVD	KNN	Mean	EM	MC	SC-KNN	SC-EM
Accuracy	**0.8603**	0.8543	0.7808	0.7598	0.8269	0.7974	0.6004	0.8026	0.7697
Sensitivity	**0.7588**	0.7512	0.7500	0.7570	0.6733	0.7256	0.6116	0.7384	0.7221
Specificity	**0.9209**	0.9142	0.7986	0.7615	0.9162	0.8392	0.5939	0.8399	0.7973
AUC	**0.8384**	0.8327	0.7743	0.7592	0.7947	0.7824	0.6028	0.7891	0.7597

Table 8

Classification results of progressive MCI patients versus normal controls with 50% data for training. All results are averaged over 10 replications.

	iSFS	iMSF	SVD	KNN	Mean	EM	MC	SC-KNN	SC-EM
Accuracy	0.8986	**0.9189**	0.8896	0.8288	0.6882	0.8849	0.7821	0.8057	0.7906
Sensitivity	**0.9915**	0.9622	0.9585	0.8561	0.9976	0.9902	0.7829	0.9402	0.7488
Specificity	0.8400	**0.8915**	0.8462	0.8115	0.4931	0.8185	0.7815	0.7208	0.8169
AUC	0.9157	**0.9265**	0.9023	0.8338	0.7453	0.9044	0.7822	0.8305	0.7829