HHS Public Access

Author manuscript

Inf Process Med Imaging. Author manuscript; available in PMC 2017 September 21.

Published in final edited form as:

Inf Process Med Imaging. 2017 June ; 10265: 184–197. doi:10.1007/978-3-319-59050-9_15.

Multi-Source Multi-Target Dictionary Learning for Prediction of Cognitive Decline

Jie Zhang[1,*], **Qingyang Li**[1,*], **Richard J. Caselli**[2], **Paul M. Thompson**[3], **Jieping Ye**[4], and **Yalin Wang**[1]

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

[2]Dept. of Neurology, Mayo Clinic Arizona, Scottsdale

[3]Imaging Genetics Center, Institute for Neuroimaging and Informatics, Univ.of Southern California, Marina del Rey, CA

[4]Dept. of Computational Medicine and Bioinformatics, Univ. of Michigan, Ann Arbor, MI

Abstract

Alzheimer's Disease (AD) is the most common type of dementia. Identifying correct biomarkers may determine pre-symptomatic AD subjects and enable early intervention. Recently, Multi-task sparse feature learning has been successfully applied to many computer vision and biomedical informatics researches. It aims to improve the generalization performance by exploiting the shared features among different tasks. However, most of the existing algorithms are formulated as a supervised learning scheme. Its drawback is with either insufficient feature numbers or missing label information. To address these challenges, we formulate an unsupervised framework for multi-task sparse feature learning based on a novel dictionary learning algorithm. To solve the unsupervised learning problem, we propose a two-stage Multi-Source Multi-Target Dictionary Learning (MMDL) algorithm. In stage 1, we propose a multi-source dictionary learning method to utilize the common and individual sparse features in different time slots. In stage 2, supported by a rigorous theoretical analysis, we develop a multi-task learning method to solve the missing label problem. Empirical studies on an $N = 3970$ longitudinal brain image data set, which involves 2 sources and 5 targets, demonstrate the improved prediction accuracy and speed efficiency of MMDL in comparison with other state-of-the-art algorithms.

Keywords

Multi-task; Alzheimer's Disease; Dictionary Learning

1 Introduction

Alzheimer's disease (AD) is known as the most common type of dementia. It is a slow progressive neurodegenerative disorder leading to a loss of memory and reduction of

*These two authors contributed equally to this work

cognitive function. Many clinical/cognitive measures such as Mini Mental State Examination (MMSE) and Alzheimer's Disease Assessment Scale cognitive subscale (ADAS-Cog) have been designed to evaluate a subject's cognitive decline. Subjects are commonly divided into three different groups: AD, Mild Cognitive Impairment (MCI) and Cognitively Unimpaired (CU), defined clinically based on behavioral and above assessments. It is crucial to predict AD related cognitive decline so an early intervention or prevention becomes possible. Prior research have shown that measures from brain magnetic resonance (MR) images correlate closely with cognitive changes and have great potentials to provide early diagnostic markers to predict cognitive decline presymptomatically in a sufficiently rapid and rigorous manner.

The main challenge in AD diagnosis or prognosis with neuroimaging arises from the fact that the data dimensionality is intrinsically high while only a small number of samples are available. In this regard, machine learning has been playing a pivotal role to overcome this so-called "large p, small n" problem. A dictionary that allows us to represent original features as superposition of a small number of its elements so that we can reduce high dimensional image to a small number of features. Dictionary learning [8] has been proposed to use a small number of basis vectors to represent local features effectively and concisely and help image content analysis. However, most existing works on dictionary learning focused on the prediction of target at a single time point [19] or some region-of-interest [18]. In general, a joint analysis of tasks from multiple sources is expected to improve the performance but remains a challenging problem.

Multi-Task Learning (MTL) has been successfully explored for regression with different time slots. The idea of multi-task learning is to utilize the intrinsic relationships among multiple related tasks in order to improve the prediction performance. One way of modeling multi-task relationship is to assume all tasks are related and the task models are connected to each other [6], or the tasks are clustered into groups [21]. Alternatively, one can assume that tasks share a common subspace [4], or a common set of features [1]. Recently, Maurer *et al.* [12] proposed a sparse coding model for MTL problems based on the generative methods. In this paper, we proposed a novel unsupervised multi-source dictionary learning method to learn the different tasks simultaneously which utilizes shared and individual dictionaries to encode both consistent and individual imaging features for longitudinal image data analysis.

Although a general unsupervised dictionary learning may overcome the missing label problem to obtain the sparse features, we still need to consider the prediction labels at different time points after we learn the sparse features. A forthright method is to perform linear regression at each time point and determine weighted matrix W separately. However, even when we have the common dictionary which models the relationship among different tasks, if prediction is purely based on linear regression which treats all tasks independently and ignores the useful information reserved in the change along the time continuum, there still exists strong bias to predict future multiple targets clinical scores.

To excavate the correlations among the cognitive scores, several multi-task models were put forward. Wang *et al.* [14] proposed a sparse multi-task regression and feature selection method to jointly analyze the neuroimaging and clinical data in prediction of the memory

performance. Zhang and Shen [17] exploited a $l_{2,1}$-norm based group sparse regression method to select features that could be used to jointly represent the different clinical status and two clinical scores (MMSE and ADAS-cog). Xiang *et al.* [16] proposed a sparse regression-based feature selection method for AD/MCI diagnosis to maximally utilize features from multiple sources by focusing on a missing modality problem. However, the clinical scores for many patients are missing at some time points, i.e., the target vector y_i may be incomplete and the above methods all failed to model this issue. A simple strategy is to remove all patients with missing target values. It, however, significantly reduces the number of samples. Zhou *et al* [21] considered multi-task with missing target values in the training process, but the algorithm did not incorporate multiple sources data.

In this paper, we propose a novel integrated unsupervised framework, termed Multi-Source Multi-Target Dictionary Learning (MMDL) algorithm, we utilize shared and individual dictionaries to encode both consistent and changing imaging features along longitudinal time points. Meanwhile, we also formulate different time point clinical score predictions as multi-task learning and overcome the missing target values in the training process. The pipeline of our method is illustrated in Fig. 1. We evaluate the proposed framework on the $N = 3970$ longitudinal images from Alzheimer's Disease Neuroimaging Initiative (ADNI) database and use longitudinal hippocampal surface features to predict future cognitive scores. Our experimental results outperform some other state-of-the-art methods and demonstrate the effectiveness of the proposed algorithm.

Our main contributions can be summarized into threefold. Firstly, we considered the variance of subjects from different time points (Multi-Source) and proposed an unsupervised dictionary learning method in stage 1 of the MMDL algorithm, in which not only does a patient share features between different time slots but different patients share some common features within the same time point. We also explore the relationship between the shared and individual dictionary in stage 1. Secondly, we use sparse features learned from dictionary learning as an input and multiple future clinical scores as corresponding labels (Multi-Target) to train the multi-task prediction model in stage 2 of the MMDL Algorithm. To the best of our knowledge, it is the first learning model which unifies both multiple source inputs and multiple target outputs with dictionary learning research for brain imaging analysis. Lastly, we also take into account the incomplete label problem. We deal with the missing label problem during the regression process and theoretically prove the correctness of the regression model. Our extensive experimental results on the ADNI dataset show the proposed MMDL achieves faster running speed and lower estimation errors, as well as reasonable prediction scores when comparing with other state-of-the-art algorithms.

2 Multi-Source Multi-Target Dictionary Learning

2.1 Stage 1: Multi-Source Dictionary Learning Stage

Given subjects from T time points: $\{X_1, X_2, \ldots, X_T\}$, our goal is to learn a set of sparse codes $\{Z_1, Z_2, \ldots, Z_T\}$ for each time point where $X_t \in \mathbb{R}^{p \times n_t}$, $Z_t \in \mathbb{R}^{l_t \times n_t}$ and $t \in \{1, \ldots, T\}$. p is the feature dimension of each subject, n_t is the number of subjects for X_t and l_t is the dimension of each sparse code in Z_t. When employing the online dictionary learning (ODL) method [11] to learn the sparse codes Z_t by X_t individually, we obtain a set of dictionary

$\{D_1, \ldots, D_T\}$ but there is no correlation between learnt dictionaries. Another solution is to construct the subjects $\{X_1, \ldots, X_T\}$ into one data matrix X to obtain the dictionary D. However, only one dictionary D is not sufficient to model the variations among subjects from different time points. To address this problem, we integrate the idea of multi-task learning into the ODL method. We propose a novel online dictionary learning algorithm, called Multi-Source Multi-Target Dictionary Learning (MMDL), to learn the subjects from different time points.

For the subject matrix X_t of a particular time point, MMDL learns a dictionary D_t and sparse codes Z_t. D_t is composed of two parts: $D_t = [\hat{D}_t, \bar{D}_t]$ where $\hat{D}_t \in \mathbb{R}^{p \times \hat{l}}$, $\bar{D}_t \in \mathbb{R}^{p \times \bar{l}_t}$ and $\hat{l} + \bar{l}_t = l_t$. $\hat{D}_t$ is the common dictionary among all the learnt dictionaries $\{D_1, \ldots, D_T\}$ while $\bar{D}_t$ is different from each other and only learnt from the corresponding matrix X_t. Therefore, objective function of MMDL can be reformulated as follows:

$$\min_{\substack{D_1,\cdots,D_T \in \Psi_t \\ Z_1,\cdots,Z_T}} \sum_{t=1}^{T} \frac{1}{2} \|X_t - [\hat{D}_t, \bar{D}_t] Z_t\|_F^2 + \lambda \sum_{t=1}^{T} \|Z_t\|_1, \text{subject to: } \hat{D}_1 = \cdots = \hat{D}_T \quad (1)$$

where $\Psi_t = \{D_t \in \mathbb{R}^{p \times l_t} : \forall j \in 1, \ldots, l_t, \|[D_t]_j\|_2 \leq 1\}$ $(t = 1, 2, \cdots, T)$ and $[D_t]_j$ is the jth column of D_t.

Fig. 2 illustrates the framework of MMDL with subjects of ADNI from three different time points which represents as X_1, X_2 and X_3, respectively. Through the multi-source dictionary learning stage of MMDL, we obtain the dictionary and sparse codes for subjects from each time point t: D_t and Z_t. In Stage 1, a dictionary D_t is composed by a shared part $\hat{D}_t$ and an individual part $\bar{D}_t$. In this example, $\hat{D}_1$, $\hat{D}_2$ and $\hat{D}_3$ are the same. For the individual part of dictionaries, MMDL learns different $\bar{D}_t$ only from the corresponding matrix X_t. We vary the number of columns $\bar{l}_t$ in $\bar{D}_t$ to introduce the variant in the learnt sparse codes Z_t. As a result, the feature dimensions of learnt sparse codes matrix Z_t are different from each other. Then we employ the max-pooling [2] method to extract the features and use extracted features to perform the regression across different time points.

The initialization of dictionaries in MMDL is critical to the whole learning process. We propose a random patch method to initialize the dictionaries from different time points. The main idea of the random patch method is to randomly select l image patches from n subjects $\{x_1, x_2, \ldots, x_n\}$ to construct D where $D \in \mathbb{R}^{p \times l}$. It is a similar way to perform the random patch approach in MMDL. In MMDL, the way we initialize $\hat{D}_t$ is to randomly select $\hat{l}$ subjects from subjects across different time points $\{X_1, \cdots, X_T\}$ to construct it. For the individual part of each dictionary, we randomly select $\bar{l}$ subjects from the corresponding matrix X_t to construct $\bar{D}_t$. After initializing dictionary D_t for each time point, we set all the sparse codes Z_t to be zero at the beginning. For each sample X_t at t-th time point, $X_t \in \mathbb{R}^{p \times n_t}$.

2.2 Stage 2: Multi-Target Learning with Missing Label

In the longitudinal AD study, we measure the cognitive scores of selected patients at multiple time points. Instead of considering the prediction of cognitive scores at a single time point as a regression task, we formulate the prediction of clinical scores at multiple future time points as a multi-task regression problem. We employ multi-task regression formulations in place 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. However, the clinical scores for many patients are missing at some time points, especially for 36 and 48 months ADNI data. It is necessary to formulate a multitask regression problem with missing target values to predict clinical scores.

In this paper, we use a matrix $\Theta \in \mathbb{R}^{m_t \times n_t}$ to indicate missing target values, where $\Theta_{i,j} = 0$ if the target value of label $Y_t(i, j)$ is missing and $\Theta_{i,j} = 1$ otherwise. Give the sparse codes $\{Z_1, \ldots, Z_T\}$ and corresponding labels $\{Y_1, \ldots, Y_T\}$ from different times where $Y_t \in \mathbb{R}^{m_t \times n_t}$, we formulate the multi-target learning stage with missing target values as:

$$\min_{W_1,\cdots,W_T} \sum_{t=1}^{T} \|\Theta(Y_t - W_t Z_t)\|_F^2 + \xi \sum_{t=1}^{T} \|W_t\|_F^2 \quad (2)$$

Algorithm 1

Multi-Source Multi-Target Dictionary Learning (MMDL)

Input: Samples and corresponding labels from different time points: $\{X_1, X_2, \ldots..X_T\}$ and $\{Y_1, Y_2, \ldots..Y_T\}$

Output: The model for different time points: $\{W_1, \ldots, W_T\}$.

Stage 1: Multi-Source Dictionary Learning

for $k = 1$ to κ **do**

For each image patch $x_t(i)$ from sample X_t, $i \in \{1, \ldots, n_t\}$ and $t \in \{1, \ldots, T\}$.

Update $\hat{D}_t^k : \hat{D}_t^k = \Phi$.

Update $z_t^{k+1}(i)$ and index set $I_t^{k+1}(i)$ by a few steps of CCD:

$$[z_t^{k+1}(i), I_t^{k+1}(i)] = CCD(D_t^k, \overline{D}_t^k, x_t(i), I_t^k(i), z_t^k(i))$$

Update the $\hat{D}_t$ and $\bar{D}_t$ by one step SGD:

$$[\hat{D}_t^{k+1}, \overline{D}_t^{k+1}] = SGD(\hat{D}_t^k, \overline{D}_t^k, x_t(i), I_t^{k+1}(i), z_t^{k+1}(i))$$

Normalize $\hat{D}_t^{k+1}$ and $\overline{D}_t^{k+1}$ based on the index set $I_t^{k+1}(i)$.

Update the shared dictionary Φ: $\Phi = \hat{D}_t^{k+1}$.

end for

Obtain the learnt dictionaries and sparse codes: $\{D_1, \ldots, D_T\}$, $\{Z_1, \ldots, Z_T\}$.

Stage 2: Multi-Target Regression with incomplete label

14: **for** $t = 1$ to T **do**

15: Given the jth column $Y_t(j)$ in Y_t, for the jth model $w_t(j)$ in W_t

16: $$w_t(j)=(\tilde{Z}_t\tilde{Z}_t^T+\xi I)^{-1}\tilde{Z}_t\tilde{Y}_t(j)$$

17: **end for**

Although the Eqn. 2 is associated with missing values on the labels, we show that it has a close form and present the theoretical analysis of stage 2 as follows:

Theorem—*For the data matrix pair* (Z_t, Y_t)*, we denote the jth row's labels* $\tilde{Y}_t(j)$ *in* Y_t*. We use* $\widetilde{Z_t}$ *and* $\tilde{Y}_t(j)$ *to represent the remaining datasets after removing the missing value in* $Y_t(j)$*. The problem of (*Eqn. 2*) can decomposed as the following equation:*

$$\min_{w_t(j)}\|(\tilde{Y}_t(j)-w_t(j)\tilde{Z}_t)\|_2^2+\xi\|w_t(j)\|_2^2 \quad (3)$$

Proof: Eqn (3) is known the Ridge regression [7]. To optimize the problem, we calculate the gradient and set the gradient to be zero. Then we can get the optimal $w_t(j)$ by the following steps:

$$2\tilde{Z}_t(\tilde{Z}_t^T w_t(j)-\tilde{Y}_t(j))+2\xi w_t(j)=0,\ \tilde{Z}_t\tilde{Z}_t^T w_t(j)-\tilde{Z}_t\tilde{Y}_t(j)+\xi w_t(j)=0,$$
$$(\tilde{Z}_t\tilde{Z}_t^T+\xi I)w_t(j)=\tilde{Z}_t\tilde{Y}_t(j),\ w_t(j)=(\tilde{Z}_t\tilde{Z}_t^T+\xi I)^{-1}\tilde{Z}_t\tilde{Y}_t(j)$$

After solving $w_t(j)$ for every time point where $j \in \{1, \ldots, m_t\}$, we can obtain the learnt model $\{W_1, \ldots, W_T\}$ to predict the clinical scores.

Our MMDL algorithm can be summarized into Algorithm 1. k denotes the epoch number where $k \in \{1, \ldots, \kappa\}$. Φ represents the shared part of each dictionary D_t which is initialized by the random patch method. For each image patch $x_t(i)$ extracted from X_t, we learn the i-th sparse code $z_t^{k+1}(i)$ from Z_t by several steps of Cyclic Coordinate Descent (CCD) [3]. Then we use learnt sparse codes $z_t^{k+1}(i)$ to update the dictionary $\hat{D}_t^{k+1}$ and $\overline{D}_t^{k+1}$ by one step Stochastic Gradient Descent (SGD)[20]. Since $z_t^{k+1}(i)$ is very sparse, we use the index set $I_t^{k+1}(i)$ to record the location of non-zero entries in $z_t^{k+1}(i)$ to accelerate the update of sparse codes and dictionaries. Φ is updated by the end of the k-th iteration to ensure $\hat{D}_t^{k+1}$ is the same part among all the dictionaries.

2.3 Updating the sparse codes

After we pick an image patch $x_t(i)$ from the sample X_t at the time point t, we fix the dictionary and update the sparse codes by following the ODL method. Then the optimization problem we need to solve becomes the following equation:

$$\min_{z_t(i)} F(z_t(i)) = \frac{1}{2}\|x_t(i) - [\hat{D}_t, \overline{D}_t] z_t(i)\|_2^2 + \lambda \|z_t(i)\|_1. \quad (4)$$

It is known as the Lasso problem [13]. Coordinate descent [3] is known as one of the state-of-the-art methods for solving this problem. In this study, we perform the CCD to optimize Eqn (4). Empirically, the iteration may take thousands of steps to converge, which is time-consuming in the optimization process of dictionary learning. However, we observed that after a few steps, the support of the coordinates, i.e., the locations of the non-zero entries in $z_t(i)$, becomes very accurate, usually after less than ten steps. In this study, we perform P steps CCD to generate the non-zero index set I_t^{k+1}, recording the non-zero entry of $z_t^{k+1}(i)$. Then we perform S steps CCD to update the sparse codes only on the non-zero entries of $z_t^{k+1}(i)$, accelerating the learning process significantly. SCC [9, 10] employs a similar strategy to update the sparse codes in a single task. For the multi-task learning, we summarize the updating rules as follows:

a. Perform P steps CCD to update the locations of the non-zero entries $I_t^{k+1}(i)$ and the model $z_t(i)^{k+1}$.

b. Perform S steps CCD to update the $z_t(i)^{k+1}$ in the index of $I_t^{k+1}(i)$.

In (a), for each step CCD, we will pick up j-th coordinate to update the model $z_t(i)_j$ and non-zero entries, where $j \in \{1, \ldots, l_t\}$. We perform the update from the 1st coordinate to the l_t-th coordinate. For each coordinate, we calculate the gradient g based on the objective function (4) then update the model $z_t^{k+1}(i)_j$ based on g. The calculation of g and $z_t^{k+1}(i)_j$ follows the equations:

$$g = [\hat{D}_t^k, \overline{D}_t^k]_j^T (\Omega([\hat{D}_t^k, \overline{D}_t^k], z_t^k(i), I_t^k(i)) - x_t(i)), \quad (5)$$

$$z_t^{k+1}(i)_j = \Gamma_\lambda(z_t^k(i)_j - g), \quad (6)$$

where Ω is a sparse matrix multiplication function that has three input parameters. Take $\Omega(A, b, I)$ as an example, A denotes a matrix, b is a vector and I is an index set that records the locations of non-zero entries in b. The returning value of function Ω is defined as: $\Omega(A, b, I) = Ab$. When multiplying A and b, we only manipulate the non-zero entries of b and corresponding columns of A based on the index set I, speeding up the calculation by utilizing the sparsity of b. Γ is the soft thresholding shrinkage function [5] and the definition of Γ is given by:

$$\Gamma_\varphi(x) = sign(x)(|x| - \varphi). \quad (7)$$

In the end of (a), we count the non-zero entries in $z_t^{k+1}(i)$ and store the nonzero index in $I_t^{k+1}(i)$. In (b), we perform S steps CCD by only considering the non-zero entries in $z_t^{k+1}(i)$. As a result, for each index μ in $I_t^{k+1}(i)$, we calculate the gradient g and update the $z_t^{k+1}(i)_\mu$ by:

$$g=[\hat{D}_t^k, \overline{D}_t^k]_\mu^T(\Omega([\hat{D}_t^k, \overline{D}_t^k], z_t^{k+1}(i), I_t^{k+1}(i))-x_t(i)), \quad (8)$$

$$z_t^{k+1}(i)_\mu=\Gamma_\lambda((z_t^{k+1}(i)_\mu-g). \quad (9)$$

Since we only focus on the non-zero entries of the model and P is less than 10 iteration and S is a much larger number, we accelerate the learning process of sparse codes significantly.

2.4 Updating the dictionaries

We update the dictionaries by fixing the sparse codes and updating the current dictionaries. Then, the optimization problem becomes as follow:

$$\min_{\hat{D}_t, \overline{D}_t} F(\hat{D}_t, \overline{D}_t)=\frac{1}{2}\|x_t(i)-[\hat{D}_t, \overline{D}_t]z_t(i)\|_2^2 \quad (10)$$

After we update the sparse codes, we have already known the non-zero entries of $z_t^{k+1}(i)$. Another key insight of MMDL is that we just need to focus on updating the non-zero entries of the dictionaries but not all columns of the dictionaries, and it accelerates the optimization dramatically. For example, when we update the i-th column and j-th row's entry of the dictionary D, the gradient of $D_{j,i}$ is set to be $\nabla D_{j,i}=z_i(D_j^T z-x_j)$. If the i-th entry of z is equal to zero, the gradient would be zero. As a result, we do not need to update the i-th column of the dictionary D. The learning rate is set to be an approximation of the inverse of the Hessian matrix H_t^{k+1}, which is updated by the sparse codes $z_t^{k+1}(i)$ in k-th iteration. In the beginning, we update the Hessian matrix by:

$$H_t^{k+1}=H_t^k+z_t^{k+1}(i)z_t^{k+1}(i)^T. \quad (11)$$

We perform one step SGD to update the dictionaries: $\hat{D}_t^{k+1}$ and $\overline{D}_t^{k+1}$. To speed up the computation, we use a vector to store the information $Dz - x$:

$$R=\Omega([\hat{D}_t^k, \overline{D}_t^k], z_t^{k+1}(i), I_t^{k+1}(i))-x_t(i). \quad (12)$$

For entry of dictionary in the μ-th column and j-th row, the procedure of learning dictionaries take the form of 1

$$[\hat{D}_t^{k+1}, \overline{D}_t^{k+1}]_{j,\mu} = [\hat{D}_t^k, \overline{D}_t^k]_{j,\mu} - \frac{1}{H_t^{k+1}(\mu,\mu)} z_t^{k+1}(i)_\mu R_j, \quad (13)$$

where μ is the non-zero entry stored in $I_t^{k+1}(i)$. For the μ-th column of dictionary, we set the learning rate as the inverse of the diagonal element of the Hessian matrix, which is $1/H_t^{k+1}(\mu,\mu)$ Due to $D_t \in \mathcal{V}_t$ in equation (1), it is necessary to normalize the dictionaries $\hat{D}_t^{k+1}$ and $\overline{D}_t^{k+1}$ after updating them. We can perform the normalization on the corresponding columns of non-zero entries from $z_t^{k+1}(i)$ because the dictionaries updating only occurs on these columns. Utilizing the non-zero information from $I_t^{k+1}(i)$ can accelerate the whole learning process significantly.

3 Experiments

3.1 Experimental Setting

We studied multiple time points structural MR Imaging from ADNI baseline (837) and 6-month (733) datasets. The responses are the MMSE and ADAS-cog coming from 5 different time points: M12, M18, M24, M36 and M48. Thus, we learned a total of 3970 images which combines 2 sources and 5 targets. The sample sizes corresponding to 5 targets are 728, 326, 641, 454 and 251. For the experiments, we used hippocampal surface multivariate statistics [15] as learning features, which is a 4 × 1 vector on each vertex of 15000 vertices on every hippocampal surface.

We built a prediction model for the above datasets using MMDL algorithm. To train the prediction models, 1102 patches of size 10 × 10 are extracted from surface mesh structures and each patch dimension is 400. The model was trained on an Intel(R) Core(TM) i7-6700 K CPU with 4.0GHz processors, 64 GB of globally addressable memory and a single Nvidia GeForce GTX TITAN X GPU. In the experimental setting of Stage 1 in MMDL, the sparsity λ = 0.1. Also, we selected 10 epochs with a batch size of 1 and 3 iterations of CCD (P is set to be 1 and S is 3). When the dictionaries and sparse codes were learned, Max-Pooling was used to generate features for annotation and get a 1 × 1000 vector feature for each images. In the Stage 2, 5-fold cross validation is used to select model parameters ξ in the training data (between 10^{-3} and 10^3).

In order to evaluate the model, we randomly split the data into training and testing sets using a 9:1 ratio and used 10-fold cross validation to avoid data bias. Lastly, we evaluated the overall regression performance using weighted correlation coefficient (wR) and root mean square error (rMSE) for task-specific regression performance measures. The two measures are defined as $wR(Y,\hat{Y}) = \sum_{i=1}^{t} Corr(Y_i, \hat{Y}_i) n_i / \sum_{i=1}^{t} n_i$, $rMSE(y,\hat{y}) = \sqrt{\|y-\hat{y}\|_2^2/n}$. For wR, Y_i is the ground truth of target of task i and $\hat{Y}_i$ is the corresponding predicted value,

Corr is the correlation coefficient between two vectors and n_i is the number of subjects of task i. For each task of rMSE, y and n is the ground truth of target and the number of subjects and $\hat{y}$ is the corresponding prediction. The smaller rMSE, the bigger wR mean the better results. We report the mean and standard deviation based on 50 iterations of experiments on different splits of data.

We compared MMDL with multiple state-of-the-art methods, ODL-L: the single-task online dictionary learning [11] followed by Lasso, L21: the multi-task method called $L_{2,1}$ norm regularization with least square loss [1]. TGL: the disease multi-task progression model called Temporal group Lasso [21], as well as Ridge and Lasso. For the parameters selection, we used the same method with the experimental setting in our stage 2.

3.2 Experimental Results

The Size of Common Dictionaries in MMDL—In Stage 1 of MMDL, the common dictionary is assumed to be shared by different tasks. It is necessary to evaluate what is an appropriate size of such common dictionary. Therefore, we set the dictionary size to be 1000 and partitioned the dictionary by different proportions: 125:875, 250:750,500:500, 750:250 and 875:125, where the left number is the size of common dictionary while the right one is the size of individual dictionary for each task. Fig. 3 shows the results of rMSE of MMSE and ADAS-cog prediction. As it shows in Fig. 3, the rMSE of MMSE and ADAS-Cog are lowest when we split the dictionary by half and a half. It means the both of common and individual dictionaries are of equal importance during the multi-task learning. In all experiments, we use the split of 500:500 as the size of common and individual dictionaries, the dimension of each sparse code in MMDL is 1000.

Time Efficiency Comparison—We compare the efficiency of our proposed MMDL with the state-of-the-art online dictionary learning (ODL). In this experiment, we focus on the single batch size setting, that is, we process one image patch in each iteration. We vary the dictionary size as: 500, 1000 and 2000. For MMDL, the ratio between the common dictionary and the individual parts is 1:1. We report the results in Table 1. We observe that the proposed MMDL use less time than ODL. When the size of dictionary are increasing, MMDL is more efficient and has a higher speedup compared to ODL.

Performance Comparison—We report the results of MMDL and other methods on the prediction model of MMSE with ADNI group in Table 2. The proposed approach MMDL outperformed ODL-L, Lasso and Ridge, in terms of both rMSE and correlation coefficient wR on four different time points. The results of Lasso and Ridge are very close while sparse coding methods are superior to them. For sparse coding models, we observe that MMDL obtained a lower rMSE and higher correlation result than traditional sparse coding method ODL-L since we consider the correlation between different time slots for different tasks and the relationship with different time points on the same patient among all tasks. We also notice that the proposed MMDL's significant accuracy improvement for later time points. This may be due to the data sparseness in later time points, as the proposed sparsity-inducing models are expected to achieve better prediction performance in this case.

We follow the same experimental procedure in the MMSE study and explore the prediction model by ADAS-cog scores. The prediction performance results are shown in Table 3. We can observe that the best performance of predicting scores of ADAS-Cog is achieved by MMDL for four time points.

Comparing with L21, after MMDL dealing with missing label, the results more linear, reasonable and accurate. Due to the dimension of M36 and M48 is too small, it is hard to learn a complete model. TGL also considered the issue of missing labels, however, MMDL still achieved the better results because MMDL incorporates multiple sources data and uses common and individual dictionaries. Although the result of MMDL had bias, MMDL still achieved the best result compared with the other five methods on predicting both MMSE and ADAS-cog, which shows our method is more efficient about dealing with missing data.

We show the scatter plots for the predicted values versus the actual values for MMSE and ADAS-Cog on the M12 and M48 in Fig. 4. In the scatter plots, we see the predicted values and actual clinical scores have a high correlation. The scatter plots show that the prediction performance for ADAS-Cog is better than that of MMSE.

4 Conclusion and Future Work

In this paper, we propose a novel Multi-Source Multi-Target Dictionary Learning for modeling cognitive decline, which allows simultaneous selections of a common set of biomarkers for multiple time points and specific sets of biomarkers for different time points using dictionary learning. We consider predicting future clinical scores as multi-task and deal with the missing labels problem. The effectiveness of the proposed progression model is supported by extensive experimental studies. The experimental results demonstrate that the proposed progression model is more effective than other state-of-the-art methods. In future, we will extend our algorithm to multi-modality data and propose more completely multiple sources with multiple targets algorithms.

Acknowledgments

The research was supported in part by NIH (R21AG049216, RF1AG051710, U54EB020403) and NSF (DMS-1413417, IIS-1421165).

References

1. Argyriou A, Evgeniou T, Pontil M. Convex multi-task feature learning. Machine Learning. 2008; 73(3):243–272.
2. Boureau YL, Ponce J, LeCun Y. A theoretical analysis of feature pooling in visual recognition. Proceedings of the 27th Annual ICML. 2010:111–118.
3. Canutescu AA, Dunbrack RL. Cyclic coordinate descent: A robotics algorithm for protein loop closure. Protein science. 2003; 12(5):963–972. [PubMed: 12717019]
4. Chen, J., et al. A convex formulation for learning shared structures from multiple tasks. Proceedings of the 26th Annual ICML; ACM; 2009. p. 137-144.
5. Combettes PL, Wajs VR. Signal recovery by proximal forward-backward splitting. Multiscale Modeling & Simulation. 2005; 4(4):1168–1200.
6. Evgeniou T, Micchelli CA, Pontil M. Learning multiple tasks with kernel methods. Journal of Machine Learning Research. 2005:615–637.

7. Hoerl AE, Kennard RW. Ridge regression: Biased estimation for nonorthogonal problems. Technometrics. 1970; 12(1):55–67.
8. Lee, H., Battle, A., Raina, R., Ng, AY. Advances in neural information processing systems. 2006. Efficient sparse coding algorithms; p. 801-808.
9. Lin B, et al. Stochastic coordinate coding and its application for drosophila gene expression pattern annotation. 2014 arXiv preprint arXiv:1407.8147.
10. Lv, J., et al. MICCAI. Springer; 2015. Modeling task fmri data via supervised stochastic coordinate coding; p. 239-246.
11. Mairal, J., Bach, F., Ponce, J., Sapiro, G. Online dictionary learning for sparse coding. Proceedings of the 26th Annual ICML; ACM; 2009. p. 689-696.
12. Maurer, A., Pontil, M., Romera-Paredes, B. Sparse coding for multitask and transfer learning. Proceedings of the 26th Annual ICML 2013; Atlanta, GA, USA. 16–21 June 2013; 2013. p. 343-351.
13. Tibshirani R. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society. Series B (Methodological). 1996:267–288.
14. Wang, H., et al. ICCV. IEEE; 2011. Sparse multi-task regression and feature selection to identify brain imaging predictors for memory performance; p. 557-562.
15. Wang Y, et al. Surface-based TBM boosts power to detect disease effects on the brain: an N=804 ADNI study. Neuroimage. Jun; 2011 56(4):1993–2010. [PubMed: 21440071]
16. Xiang S, et al. Bi-level multi-source learning for heterogeneous block-wise missing data. NeuroImage. 2014; 102:192–206. [PubMed: 23988272]
17. Zhang D, Shen D, Initiative ADN, et al. Multi-modal multi-task learning for joint prediction of multiple regression and classification variables in alzheimer's disease. Neuroimage. 2012; 59(2): 895–907. [PubMed: 21992749]
18. Zhang, J., et al. MICCAI. Springer; 2016. Hyperbolic space sparse coding with its application on prediction of Alzheimer's disease in mild cognitive impairment; p. 326-334.
19. Zhang, J., et al. Applying sparse coding to surface multivariate tensor-based morphometry to predict future cognitive decline. Biomedical Imaging (ISBI), 2016 IEEE 13th International Symposium on; IEEE; 2016. p. 646-650.
20. Zhang, T. Solving large scale linear prediction problems using stochastic gradient descent algorithms. Proceedings of the 25th Annual ICML; ACM; 2004. p. 116
21. Zhou, J., Liu, J., Narayan, VA., Ye, J. Modeling disease progression via fused sparse group lasso. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining; ACM; 2012. p. 1095-1103.

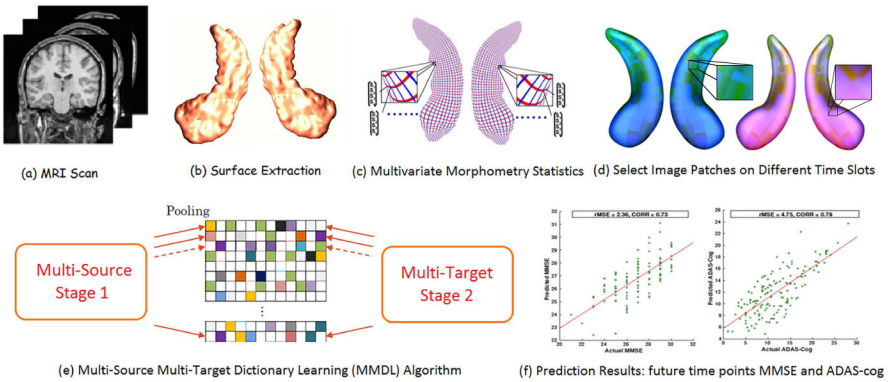

Fig. 1.

The pipeline of our method. We extracted hippocampi from MRI scans (a), then we registered hippocampal surfaces (b) and computed surface multivariate morphometry statistics (c). Image patches were extracted from the surface maps to initialize the dictionary (d) for Multi-Source Multi-Target Dictionary Learning (e). We used features from two time points to predict five future time points MMSE and ADAS-cog (f).

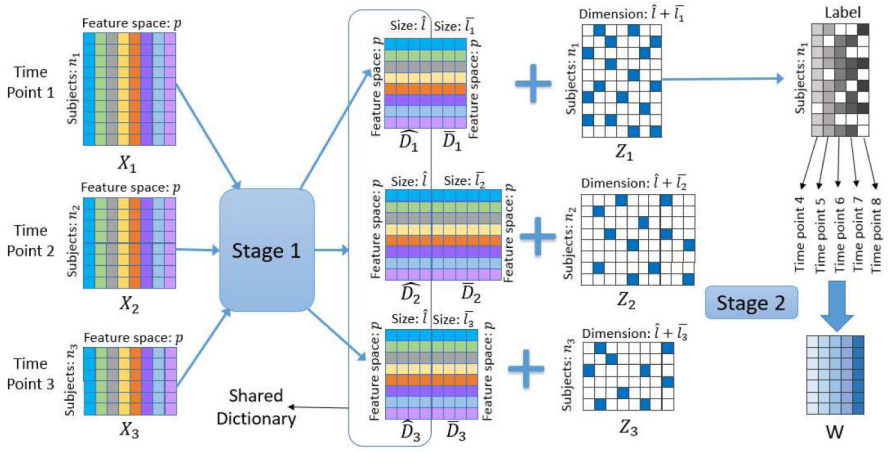

Fig. 2.

Illustration of the learning process of MMDL on ADNI datasets from multiple different time points to predict multiple future time points clinical scores.

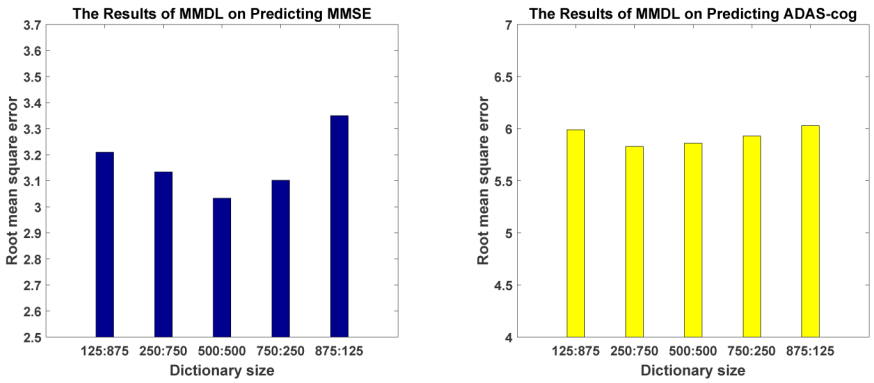

Fig. 3.

Comparison of rMSE performance by varying the size of common dictionary.

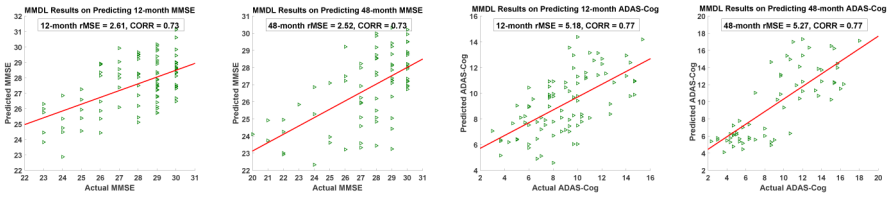

Fig. 4.

Scatter plots of actual MMSE and ADAS-Cog versus predicted values on M12 and M48 by using MMDL.

Table 1

Time comparisons of MMDL and ODL by varying dictionary size.

Dictionary Size	MMDL	ODL
500	1.74 hour	8.84 hour
1000	3.34 hour	21.95 hour
2000	6.93 hour	49.90 hour

Table 2

The Prediction Results of MMSE on Whole Dataset.

Methods	wR	M12	M18	M24	M36	M48
Lasso	0.40±0.09	4.04±0.77	3.46±0.97	5.53±0.86	4.39±0.74	4.73±1.49
Ridge	0.41±0.07	4.26±0.56	3.56±0.93	5.05±0.54	4.21±0.47	3.62±0.91
L21	0.57±0.01	3.32±0.63	4.75±0.75	4.64±0.88	4.08±1.01	3.11±1.05
ODL-L	0.63±0.08	2.99±0.63	**2.88±0.68**	4.29±0.84	3.62±1.45	2.93±1.07
TGL	0.70±0.05	2.73±0.72	4.00±1.31	4.00±0.64	3.19±1.38	2.60±1.42
MMDL	**0.73±0.02**	**2.61±0.55**	3.37±1.01	**3.66±0.78**	**2.73±1.09**	**2.52±1.20**

Table 3

The Prediction Results of ADAS-cog on Whole Dataset.

Methods	wR	M12	M18	M24	M36	M48
Lasso	0.49±0.05	6.81±1.03	6.87±0.74	7.62±0.87	8.08±1.39	6.55±1.34
Ridge	0.46±0.07	7.68±0.96	6.89±1.69	7.84±1.54	8.59±0.62	6.64±1.58
L21	0.53±0.07	6.40±0.51	6.95±0.88	8.07±0.67	8.00±1.04	5.92±0.60
ODL-L	0.53±0.05	5.65±0.73	4.97±0.67	7.30±0.77	7.25±0.69	5.56±1.22
TGL	0.72±0.04	5.52±1.15	5.70±0.53	6.85±1.06	**6.36±1.22**	5.73±0.61
MMDL	**0.77±0.02**	**5.18±0.88**	**4.64±1.12**	**6.76±1.35**	6.78±1.54	**5.27±1.76**