MELANCHOLIC DEPRESSION PREDICTION BY IDENTIFYING REPRESENTATIVE FEATURES IN METABOLIC AND MICROARRAY PROFILES WITH MISSING VALUES

ZHI NIE†, TAO YANG†, YASHU LIU†, BINBIN LIN†, QINGYANG LI†, VAIBHAV A NARAYAN‡, GAYLE WITTENBERG‡, JIEPING YE†

†*Department of Computer Science and Engineering, Center for Evolutionary Medicine and Informatics, The Biodesign Institute, Arizona State University, Tempe, AZ 85287, USA*

‡*Johnson & Johnson Pharmaceutical Research & Development, LLC, Titusville, NJ, USA*

E-mail: †{*Zhi.Nie, T.Yang, Yashu.Liu, Binbin.Lin, Qingyang.Li, Jieping.Ye*}*@asu.edu,*
‡{*VNaray16, GWittenb*}*@its.jnj.com*

Recent studies have revealed that melancholic depression, one major subtype of depression, is closely associated with the concentration of some metabolites and biological functions of certain genes and pathways. Meanwhile, recent advances in biotechnologies have allowed us to collect a large amount of genomic data, e.g., metabolites and microarray gene expression. With such a huge amount of information available, one approach that can give us new insights into the understanding of the fundamental biology underlying melancholic depression is to build disease status prediction models using classification or regression methods. However, the existence of strong empirical correlations, e.g., those exhibited by genes sharing the same biological pathway in microarray profiles, tremendously limits the performance of these methods. Furthermore, the occurrence of missing values which are ubiquitous in biomedical applications further complicates the problem. In this paper, we hypothesize that the problem of missing values might in some way benefit from the correlation between the variables and propose a method to learn a compressed set of representative features through an adapted version of sparse coding which is capable of identifying correlated variables and addressing the issue of missing values simultaneously. An efficient algorithm is also developed to solve the proposed formulation. We apply the proposed method on metabolic and microarray profiles collected from a group of subjects consisting of both patients with melancholic depression and healthy controls. Results show that the proposed method can not only produce meaningful clusters of variables but also generate a set of representative features that achieve superior classification performance over those generated by traditional clustering and data imputation techniques. In particular, on both datasets, we found that in comparison with the competing algorithms, the representative features learned by the proposed method give rise to significantly improved sensitivity scores, suggesting that the learned features allow prediction with high accuracy of disease status in those who are diagnosed with melancholic depression. To our best knowledge, this is the first work that applies sparse coding to deal with high feature correlations and missing values, which are common challenges in many biomedical applications. The proposed method can be readily adapted to other biomedical applications involving incomplete and high-dimensional data.

Keywords: melancholic depression, sparse coding, missing value, clustering, disease prediction, biomarker identification, feature learning.

1. Introduction

Understanding the fundamental biology underlying melancholic depression is a very challenging problem of great clinical importance for researchers from medical and psychiatric research communities. Unlike some other subtypes of depression, melancholic depression is described as "mainly biologically based rather than determined by personality or life circumstances",[1]

which motivates researchers to discover biological evidence of the disease. Research with regard to this aspect has made progress in recent years. For instance, it has been shown that an elevated level of concentration of certain metabolites in plasma is found among the depressive patients with melancholia.[2] More recently, Gabbay *et al.*[3] pointed out the significance of kynurenine pathway in adolescent depression with melancholic features through comparing adolescents with melancholic depression with non-melancholic depression and healthy adolescents. Also, recently through gene ontology and pathway analyses, certain biological functions of differentially expressed mRNAs were identified as related to fundamental metabolic processes and brain disorders.[4]

On the other hand, recent advances in biotechnologies have made it possible to detect a large number of metabolites from human tissue extract.[5] Meanwhile, the microarray technology has taken us from being able to analyze the biological functions of only a few related genes or proteins at one time to the place where global investigation of cellular activities is possible.[6] With data on such a large scale available, one promising approach that can potentially offer us a deeper understanding of collective impact of numerous factors involved in the pathogenesis of melancholic depression and its prospective treatments is to build predictive models based on all of the information available using machine learning approaches. However, the "curse of dimensionality" due to the fact that the number of variables of interest far exceeds the number of samples available renders most of traditional classification/regression algorithms less effective in this setting. Furthermore, strong empirical correlations between the variables, especially in the case of microarray data where there is high degree of linear dependence between expression measures of a group of genes sharing the same biological pathways,[7] tremendously limit the prediction performance of traditional machine learning algorithms. Another major issue with data collected on a large scale is the presence of missing values, which is ubiquitous in biomedical applications.

Most of the existing methods were designed to deal with either the problem of strong empirical correlations between the variables or the problem of missing values. For instance, Bühlmann *et al.*[8] recently proposed a bottom-up agglomerative clustering algorithm to deal with correlations between the variables, but their method cannot be readily used in the context of missing values. As for the issue of missing values, there are two basic approaches to dealing with missing data. We can either discard the samples with missing values or impute the missing data. The shortcoming of the first approach is obvious. It does not make full use of available information. The second approach, i.e., imputation of missing data, generally involves certain assumptions about the missing pattern of the data which may not be satisfied in applications. The most commonly used imputation technique, EM, for example, assumes that data is sampled from a Gaussian distribution and the missing-at-random (MAR) assumption is satisfied.

We hypothesize that the problem of missing values might potentially benefit from the correlations between the variables; for example, a variable with missing values could borrow information from its correlated variables. However, simply imputing the missing values of a variable by exploiting information from its correlated variables still leaves the problem of empirical correlations between the variables unsolved. Therefore, instead of discarding the

incomplete samples or imputing the missing values, we attempt to generate a compressed set of representative features for all the samples from the data with a group of correlated variables represented by one or a few features. We demonstrate in this paper that sparse coding, which has been shown to be very effective in object recognition and image denoising applications,[9,10] is desirable for such a task. Specifically, we apply sparse coding in such a way that the learned dictionary corresponds to a set of representative features and each variable is represented as a sparse combination of these features. Furthermore, we develop an efficient algorithm to solve the proposed sparse coding formulation to deal with missing values.

We apply the proposed algorithm to datasets of metabolic and microarray profiles collected from a group of subjects consisting of both patients with melancholic depression and healthy controls. Results from our experiments revealed that features obtained from our method significantly outperform those generated from several baseline methods based on traditional clustering methods and standard data imputation techniques. In particular, in comparison with our baseline methods, the representative features learned by the proposed method achieve much better performance in predicting the disease status of the subjects with melancholic depression on both datasets. In addition, on the dataset of metabolic profiles, we found that most of the known metabolites within each cluster are biologically relevant. These results demonstrate the promise of the proposed method for learning from incomplete and high-dimensional biomedical data.

The rest of the paper is organized as follows. In section 2, we formulate the sparse coding problem in the presence of missing values. In section 3, we describe the datasets used in the analysis and present experimental results. Section 4 concludes the paper.

2. Learning Representative Features via Sparse Coding

In this section, we present our sparse coding formulation to learn a compressed representative set of features such that the observations from all the samples on each variable can be represented as a sparse linear combination of these learned features. The proposed formulation can naturally deal with missing values.

Suppose we are given a dataset of m samples and their observations on n variables with missing values which we denote as $\mathcal{X} = \{(x_1, \Omega_1), \cdots, (x_n, \Omega_n)\}$. Each x_i $(1 \leq i \leq n)$ is an m dimensional column vector representing measurements of all the samples on the i-th variable (e.g., concentrations of the i-th metabolite or measurements of the i-th gene expression), and Ω_i is an ordered set of integers ranging from 1 to m including the indices of samples whose measurements on the i-th variable are observed. If there is no missing value in x_i, then Ω_i includes all integers between 1 and m. Our goal is to use sparse coding to learn a set of k representative features such that each variable x_i can be well represented by a sparse combination of these k features. In the presence of missing values, the sparse coding problem can be formulated as the following optimization problem:

$$\begin{aligned} &\min_{\mathcal{D}, z_1, \cdots, z_n} \sum_{i=1}^{n} \frac{1}{2} \|\mathcal{P}_{\Omega_i}(\mathcal{D} z_i - x_i)\|_2^2 + \lambda \|z_i\|_1 \\ &s.t. \qquad \|\mathcal{D}_{\cdot j}\|_2 \leq 1; 1 \leq j \leq k, \end{aligned} \tag{1}$$

Algorithm 2.1 Stochastic Coordinate Coding with Missing Values

Initialization:

Samples $X = \{x_1, x_2, \ldots, x_n\}$, missing indices $\Omega = \{\Omega_1, \Omega_2, \ldots, \Omega_n\}$, $\lambda \in \mathbb{R}$, initial dictionary $\mathcal{D}^0 \in \mathbb{R}^{m \times k}$, initial combination coeficients $Z = \{z_1, z_2, \ldots, , z_n\}$, number of iterations T.

$\mathcal{H} \in \mathbb{R}^{k \times k} \leftarrow 0$
for $t = 1$ to T **do**
  for $i = 1$ to n **do**
    Update coefficients via a few iterations of coordinate descent according to (3)
    $z_i \leftarrow \arg\min_{z_i} f_{\mathcal{D}}(z_i) \equiv \frac{1}{2}\|\mathcal{P}_{\Omega_i}(\mathcal{D}z_i - x_i)\|_2^2 + \lambda\|z_i\|_1$.
    Update Hessian matrix
    $\mathcal{H} = \mathcal{H} + z_i z_i^T$,
    Update the dictionary $\mathcal{D}^{i-1}$ colum by column
    for $j \in \{t | 1 \leq t \leq k, t \in \mathbb{N}, z_i(t) \neq 0\}$ **do**
      $u_j = \mathcal{P}_{\Omega_i}(\mathcal{D}^{i-1}_{\cdot,j}) - \frac{1}{\mathcal{H}[j,j]} z_i(j) * \mathcal{P}_{\Omega_i}(\mathcal{D}^{i-1} * z_i - x_i)$.
      $\mathcal{P}_{\Omega_i}(\mathcal{D}^{i}_{\cdot,j}) \leftarrow u_j$.
      $\mathcal{D}_{\cdot j} \leftarrow \frac{1}{\max\{\|\mathcal{D}_{\cdot j}\|_2, 1\}} \mathcal{D}_{\cdot j}$.
    end for
  end for
  $\mathcal{D}^0 \leftarrow \mathcal{D}^n$
end for

Output: $\mathcal{D}^n$.

where z_i represents sparse combination coefficients (also called sparse code) for x_i, and $\mathcal{D} \in \mathbb{R}^{m \times k}$, the dictionary or codebook, represents the learned set of features with its j-th column denoted as $D_{\cdot j}$. $\mathcal{P}_{\Omega_i}(\cdot)$ projects a matrix into its submatrix consisting of rows indexed by Ω_i. The cardinality of Ω_i is denoted by m_i. Minimization of the first term in (1) leads to a feature set $\mathcal{D}$ such that the observed entries of each variable can be well represented by the features in $\mathcal{D}$. Note that variables with similar combination coefficients are likely to be correlated. Minimization of the second term in (1) induces sparsity on combination coefficients of each variable, enforcing each variable to be represented by only a small subset of features in $\mathcal{D}$. λ controls the sparsity of each z_i. The larger the λ is, the sparser each z_i will be. With a proper λ, minimization of these two terms combined will yield a feature set $\mathcal{D}$ such that the observed part of each variable can be well represented by a small subset of features from $\mathcal{D}$.

Although the problem in (1) is convex with respect to either z_i or $\mathcal{D}$, it is not jointly convex. Thus, it is difficult to obtain a globally optimal solution. Most algorithms solving the sparse coding problem alternate the step of optimizing over z_i with a fixed $\mathcal{D}$ and the step of optimizing over $\mathcal{D}$ with a fixed z_i.[11] In this paper, we extend the framework proposed by Lin *et al.*,[12] which applies to data without missing values, to solve sparse coding with missing values in the data matrix. The detailed description of the algorithm to solve the above problem is presented in Algorithm 2.1.

With a fixed $\mathcal{D}$, updating z_i amounts to solving a Lasso[13] problem which can be formulated

as follows:

$$\min_{z_i} f_{\mathcal{D}}(z_i) \equiv \frac{1}{2}\|\mathcal{P}_{\Omega_i}(\mathcal{D}z_i - x_i)\|_2^2 + \lambda\|z_i\|_1. \tag{2}$$

Suppose that we iteratively update the sparse code of each variable for T epochs. The total number of Lasso problems involved is Tn. Even with state-of-the-art solvers, the total cost of solving so many Lasso problems is prohibitive, particulary in the case of the microarray data where there are usually at least tens of thousands of genes involved. In our algorithm, we adopt the stragtegy of solving the Lasso problem incrementally by updating only the support of z_i for a few times via coordinate descent with a warm start. This strategy has proved to be computationally efficient in practice while still yielding competitive performance.

In the algorithm, each time we pick one element, say the j-th element z_{ij} $(1 \leq j \leq k)$, to update with all the other coordinates fixed. Under this circumstance, (2) can be converted to a problem with a closed form solution. Let $z_i = z_i^s$ before z_{ij} is updated. Let $z_i = z_i^{s+1}$ after z_{ij} is updated , and $\overline{z}_{ij} = [z_{i,1}, \cdots, z_{i,j-1}, z_{i,j+1}, \cdots, z_{i,k}]^T$, $\overline{\mathcal{D}}_{\cdot j} = [\mathcal{D}_{\cdot 1}, \cdots, \mathcal{D}_{\cdot j-1}, \mathcal{D}_{\cdot j+1}, \cdots, \mathcal{D}_{\cdot k}]$. Apparently, $\overline{z}_{ij}^{s+1} = \overline{z}_{ij}^s$, and z_{ij}^{s+1} is the only unknown variable. Plugging z_i^{s+1} into $f_{\mathcal{D}}$, we have

$$\begin{aligned}
f_{\mathcal{D}}(z_i^{s+1}) &= \frac{1}{2}\|\mathcal{P}_{\Omega_i}(x_i - \overline{\mathcal{D}}_{\cdot j}\overline{z}_{ij}^{s+1})\|_2^2 - \big(\mathcal{P}_{\Omega_i}(x_i - \overline{\mathcal{D}}_{\cdot j}\overline{z}_{ij}^{s+1})\big)^T\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})z_{ij}^{s+1} \\
&\quad + \frac{1}{2}\|\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})\|_2^2(z_{ij}^{s+1})^2 + \lambda|z_{ij}^{s+1}| + \lambda\|\overline{z}_{ij}^{s+1}\|_1 \\
&= \frac{1}{2}\|\mathcal{P}_{\Omega_i}(x_i - \overline{\mathcal{D}}_{\cdot j}\overline{z}_{ij}^{s})\|_2^2 - \big(\mathcal{P}_{\Omega_i}(x_i - \overline{\mathcal{D}}_{\cdot j}\overline{z}_{ij}^{s})\big)^T\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})z_{ij}^{s+1} \\
&\quad + \frac{1}{2}\|\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})\|_2^2(z_{ij}^{s+1})^2 + \lambda|z_{ij}^{s+1}| + \lambda\|\overline{z}_{ij}^{s}\|_1.
\end{aligned}$$

By setting $\partial f_{\mathcal{D}}(z_i^{s+1})/\partial z_{ij}^{s+1} = 0$, we have

$$-(\mathcal{P}_{\Omega_i}(x_i - \overline{\mathcal{D}}_{\cdot j}\overline{z}_{ij}^{s}))^T\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j}) + \|\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})\|_2^2(z_{ij}^{s+1}) + \lambda\,\mathrm{sign}(z_{ij}^{s+1}) = 0.$$

Adding and subtracting $\|\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})\|_2^2 z_{ij}^s$, we have

$$-(\mathcal{P}_{\Omega_i}(x_i - \mathcal{D}z_i^{s}))^T\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j}) + \|\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})\|_2^2(z_{ij}^{s+1}) - \|\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})\|_2^2(z_{ij}^{s}) + \lambda\,\mathrm{sign}(z_{ij}^{s+1}) = 0.$$

This equation has a closed form solution which is given by

$$z_{ij}^{s+1} = S_a\Big(\frac{(\mathcal{P}_{\Omega_i}(x_i - \mathcal{D}z_i^{s}))^T\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})}{\|\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})\|_2^2} + z_{ij}^s\Big). \tag{3}$$

where S is the shrinkage operator defined by $S_\alpha(x) = (|x| - \alpha)_+\,\mathrm{sign}(x), x, \alpha \in \mathbb{R}$ and $a = \lambda/\|\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})\|_2^2$. Note that although x_i may have missing values, z_i does not contain missing values. Only the rows of $\mathcal{D}$ corresponding to the rows of x_i where values are observed are used to update z_i. It is worth emphasizing that we only update all the coordinates of z_i in the first iteration due to the fact that the dictionary has changed since z_i was updated last time. For iterations afterwards, only the support of z_i is updated.

With a fixed z_i, we only use the newly updated z_i and the corresponding x_i to update $\mathcal{D}$ using gradient descent. The problem can be formulated as follows:

$$\min_{\mathcal{D}} g_{z_i}(\mathcal{D}) \equiv \frac{1}{2}\|\mathcal{P}_{\Omega_i}(x_i - \mathcal{D}z_i)\|_2^2 \quad s.t.\ \|\mathcal{D}_{\cdot j}\|_2 \leq 1,\ 1 \leq j \leq k. \tag{4}$$

The gradient of g_{z_i} with respect to $\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})$ is $\nabla_{\mathcal{P}_{\Omega_i}(\mathcal{D}_{\cdot j})} g_{z_i} = \mathcal{P}_{\Omega_i}(\mathcal{D}z_i - x_i)z_{ij}$. Note that only the columns of $\mathcal{D}$ corresponding to the support of z_i need to be updated. As to the learning rate, following the practice of Mairal *et al.*[11] and Lin *et al.*,[12] we set the learning rate to be $1/\mathcal{H}[j,j]$ where $\mathcal{H}$ is initialized to be zero and accumulates $z_i z_i^T$. Finally, $\mathcal{D}_{\cdot j}$ is normalized to be within the unit ball.

Note that most existing works apply sparse coding on signals, e.g., images to learn a representation of each signal. One novel aspect of the proposed framework is that we apply sparse coding to learn a sparse representation for each variable and use the dictionary $\mathcal{D}$ as features where each row represents a sample and each column represents a feature. In addition, most sparse coding formulations assume that the data is complete, while our proposed framework can naturally deal with missing values in the data. From the perspective of clustering, different from those traditional clustering algorithms such as Kmeans which assign each data point to one cluster, sparse coding can be considered as a soft version of clustering in that it allows a point to belong to different clusters at the same time, depending on the number of non-zero elements in its sparse representation vector. Such flexibility is desirable in many applications, since some data points may be close to multiple clusters. The sparse representation of a data point may be a zero vector, especially when regularization parameter λ in the algorithm is set to be a large value. In this case, the data point can be regarded as an outlier or a noisy point.

3. Data and Experiments

The datasets used in our analysis were collected from a study initiated by Brain Resource Company (BRC) and Johnson & Johnson Pharmaceutical Research & Development, L.L.C(J&J PRD). The overall objective of the study is to identify the best molecular profiles, cognitive and psychophysiological biomarkers in people with depression. In the study, about 100 depressive subjects evenly distributed in gender and age as well as an equal number of matched healthy controls are recruited nationwide by BRC in Australia. All the subjects that are included in the study have been screened to satisfy certain criteria on Hamilton Depression Rating Scale (HAM-D) score, CORE score,[14] toxicology tests, and so on. Part of the study is dedicated mainly to collecting the following information from all the participants : a) Personal medical history; b) Cognition; c) Electrical brain-body function (EBBF); d) Brain structure (e.g. structural MRI, functional MRI); e) Molecular profiles (which includes metabolite, microarray, protein and transcripts profiles). However, not all the subjects have all five blocks of information or all sub-categories of one type of information recorded due to a variety of reasons such as participant dropout, failure of quality control, long storage time, etc. In our analysis, we use metabolite and microarray data from the molecular profiles to demonstrate the effectiveness of the proposed algorithm in dealing with correlated variables as well as the significant discriminative power of the resulting compressed set of features. As for the target, we are interested in melancholic depression. The decision of whether or not a subject is diagnosed with melancholic depression is made on the basis of pyschomotor findings in the CORE scale which consists of 18 items measuring a subject's interactiveness, motor agitation, etc. The score of each item ranges from 0 (no symptom) to 3 (severe symptom). A subject will be labeled as melancholically depressive if he or she has a total score on CORE over 8.

3.1. *Analysis on Metabolic Profiles*

3.1.1. *Data Preprocessing*

During the stage of medical screening, a sample of 20ml of plasma was obtained from each of the participants by BRC and was later on sent to J&J PRD where the molecular profiling analysis was carried out. Based on Gas chromatography-mass spectrometry (GS-MS) and Liquid chromatography-mass spectrometry (LC-MS/MS), 272 peaks were acquired with 160 of them being known metabolites and the rest unknown. Considering the fact that concentrations of metabolites change with the increase of storage time, we removed all the samples stored for over 200 days and performed a linear regression of concentration with storage time at the temperature of -20 degrees centigrade on the remaining samples to control for the confounding effects caused by storage time. Also, over 40 metabolites whose concentrations were detected to be highly correlated with storage time were excluded from our analysis. After the preprocessing, we are left with 118 samples and 228 metabolites in total. Among all the 118 samples, 21 were diagnosed with melancholic depression and 97 were healthy controls. About 1.27% of all the entries in the data matrix are missing.

The method of sparse coding proposed in this paper can deal with missing entries in the data matrix. To demonstrate the capability of our method to generate a compressed discriminative set of features even under the presence of missing values, we also include several baseline methods for comparison, which impute the missing entries using some standard missing value imputation techniques including: 1) HalfMin: Impute the missing entries on each variable by filling in half of the minimum of the observed values on that variable; 2) KNN: Find the k nearest neighbors of the variable with missing values based on observed part and assign the missing values to be a weighted combination of its nearest neighbors with the weight determined by the inverse of the Euclidean distance between the variable concerned and the neighbor; 3) Expectation Maximization (EM): Assuming that the underlying distribution of the samples follows a mixture of Gaussian distribution, it iterates between updating the posterior probability of each of the data points and updating the mean, covariance matrix and mixing coefficient of each Gaussian component and filling in the missing entries with conditional expectation given the observed part; 4) Singular value decomposition (SVD): Assuming that there is an inherent low-rank structure in the data, it fills in the missing entries with the values obtained from the low-rank approximation of the data.

Before further data analysis, each variable was normalized to have zero mean and unit standard deviation. In the case of variables with missing values, we simply omitted the missing values when computing the mean and standard deviation.

3.1.2. *Classification*

With the ratio of the number melancholic depressive subjects to the number of healthy controls being almost 1 to 5, the dataset is extremely imbalanced. Direct application of traditional classification methods like Support Vector Machine (SVM) in this situation would severely biased the classifier toward majority class. Drawing on the experience from Dubey *et al.*(2014),[15] we implemented a scheme which combines the techniques of data undersampling and model ensemble methods to deal with the issue of data imbalance.

Table 1. Classification performance on metabolic profiles

Method	RF Vote				RF Weighted Vote			
	Acc	Sen	Spe	AUC	Acc	Sen	Spe	AUC
HalfMin	0.7624	0.6333	0.7922	0.7128	0.7624	0.6333	0.7922	0.7128
EM	0.7367	0.5833	0.7711	0.6772	0.7290	0.5833	0.7611	0.6722
KNN	0.7624	0.6833	0.7822	0.7328	0.7624	0.6333	0.7922	0.7128
SVD	0.7450	0.5833	0.7811	0.6822	0.7547	0.6333	0.7822	0.7078
HC	0.7540	0.8000	0.7489	0.7744	0.7214	0.6500	0.7411	0.6956
Kmeans	0.7778	0.7167	0.7922	0.7544	0.7861	0.7167	0.8022	0.7594
SC	**0.8315**	**0.8333**	**0.8344**	**0.8339**	**0.8315**	**0.8333**	**0.8344**	**0.8339**

In this scheme, samples from each of the two classes were randomly partitioned into 10 folds of (approximately) equal size. One fold from both classes were set aside for testing and the rest were used as training set. During the training stage, we used all the samples from the minority class and randomly subsampled with replacement the same amount of samples from the majority class to build a classifier. The process of subsampling was repeated p times (in our experiments, we choose $p = 30$) so that p different classifiers will be built on the same training set. Each of the p classifiers will give a prediction of the label of each testing sample. In the ensemble stage, predictions from different classifiers were combined in different ways. In our experiments we adopted two strategies to combine the predictions from different classifiers. The first strategy counts the number of times that a given testing sample is predicted positive and the number of times the sample is predicted negative. The final label of the sample is given by the majority of the votes. If there is a tie, then we randomly assign the testing sample to one of the two classes. The second strategy weights the prediction of each of the classifier with its confidence accompanying the prediction. The final predicted label is determined by the sign of the confidence weighted sum of prediction from each of the p classifiers. Each of the 10 folds from both classes is used as the testing fold once so that each of the samples is used as testing sample exactly once. We regard it as a convention throughout the paper that the positive class consists of subjects with melancholic depression and the negative class consists of healthy controls. The basic classifiers we used in the paper include SVM with linear kernel and Random Forest (RF). We used the following four measures to evaluate the performance of ensemble of classifiers: accuracy, sensitivity, specificity and area under curve (AUC).

In our experiments, we compared classification performance on features yielded from our method (SC, in abbreviation) with those generated by different data imputation and clustering methods. We tried different initializations, different values of K (number of keywords in the dictionary or number of clusters) and different values of λ (regularization parameter) on our method, and tried different initializations and different values of K on Kmeans and hierarchical clustering.

The classification performance is reported in Table 1. Due to space limit and the fact that SVM generally performed worse than RF on this dataset, we only report the classification performance by RF. In using KNN for data imputation, we tried a range of values for k and report the results from $k = 3$ since it gives the best performance. Also for Kmeans

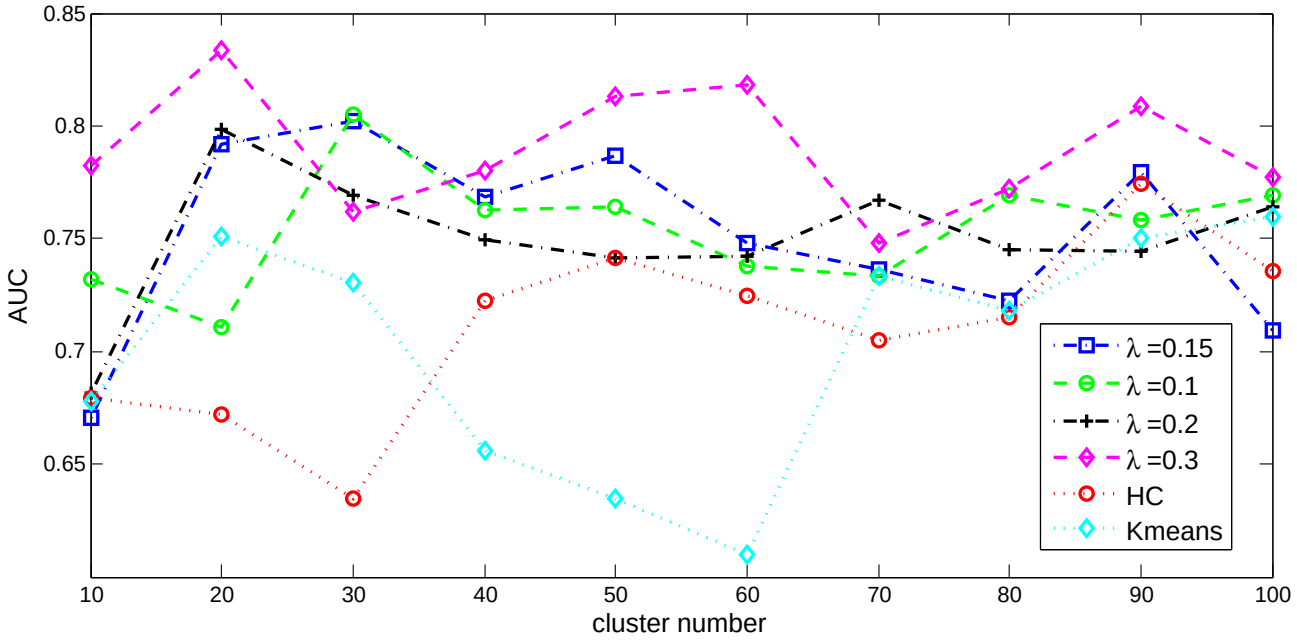

Fig. 1. Changes in AUC score with varying number of clusters for sparse coding with different regularization parameters, Kmeans and hierarchical clustering.

and hierarchical clustering, we imputed the raw design matrix using KNN before we applied these clustering techniques as KNN gave the best classification performance among all data imputation techniques. All the three clustering methods share one parameter, K, which we varied between 10 and 100 with a step size of 10. For sparse coding, there is an extra parameter λ which we set to be 0.1, 0.15, 0.2 and 0.3 in our experiments.

Fig. 1 shows how the AUC score changes when we ran classification on features generated by different clustering algorithms for different values of K. We can see that although the classification performance fluctuates with a growing number of clusters for all the clustering algorithms, sparse coding generally yields feature sets with stronger discriminative power when λ is set to be 0.3. This implies that there are indeed several clusters of metabolites in our dataset since a larger λ tends to drive the combination coefficient for each metabolite to be sparse and several metabolites are potentially outliers.

It is also of great interest to explore the groups of metabolites that are clustered together by looking into the coefficient matrix Z. The metabolites clustered into the ith group, which is represented by the ith column of $\mathcal{D}$, are those metabolites corresponding to the nonzero entries of the ith row of matrix Z. We looked into the most discriminative features (measured in terms of their p values) in the feature set $\mathcal{D}$ on which the best classification performance is achieved and their corresponding rows in the coefficient matrix Z. The most discriminative feature, which has a p-value of 2.93×10^{-16}, corresponds to six metabolites with four of them being unknown, one of them belonging to the general category of "Complex lipids, fatty acids and related" and one of them belonging to the general category of "Amino acids and related". The second most discriminative feature which has a p-value of 6.64×10^{-16} corresponds to twenty-four metabolites, with sixteen of them falling in the category "Amino acids and related", two of them belonging to the category "Nucleobases and related", five of them being unknown

and one of them belonging to the category "Hormones, signal substances and related". The third most discriminative feature which has a p-value of 7.92×10^{-16} corresponds to a group of nineteen metabolites, with nine of them falling into the category "Complex lipids, fatty acids and related", six of them unknown, two of them being "unknown lipid" and one of them being "Vitamins, cofactors and related". From these, we can see that sparse coding does produce meaningful clusters in that most of the known metabolites assigned into the same cluster belong to related categories.

3.2. *Analysis on Microarray Profiles*

Microarray is another modality of data collected on the subjects. This dataset included information on 54675 transcripts from 123 subjects with 28 of them being labeled as positive (with melancholic depression) and 95 of them being labeled as negative (normal control). The same framework introduced in 3.1.2 to deal with extreme imbalance between two classes was also used on this dataset. Among the 54675 transcripts, a list of 2261 transcripts was identified as related to depression and a list of 3297 transcripts was identified as related to immune system. We ran classification on all three sets of data. There is no missing value in this dataset.

We also compared the classification performance on features generated by sparse coding, Kmeans and hierarchical clustering. For all the clustering methods, we set the number of clusters to be 100, 200, 500, 1000. For sparse coding, we used the same set of values for λ that were used on the metabolic profile. We report the best classification performance on features generated by sparse coding on each λ over all the K values and best classification performance on features generated by Kmeans and hierarchical clustering over all the Ks. we only report the classification performance by SVM since SVM generally outperforms Random Forest on this dataset.

It is evident from Table 2, Table 3, Table 4 that the feature sets yielded by sparse coding have superior discriminative power than those generated by traditional clustering methods and raw data across all these three sets of data. In particular, feature sets obtained through sparse coding give rise to significantly improved sensitivity in classification performance, implying that it allows prediction with high accuracy of disease status in those who are diagnosed with melancholic depression. Overall, the feature sets given by sparse coding produce the best performance when $K = 500$. However, as shown in the results, a proper choice of λ is important as well.

4. Conclusion and Future Work

In this paper, we propose a method to learn a compressed set of representative features through an adapted version of sparse coding which is capable of simultaneously clustering variables with strong empirical correlation and dealing with the missing values in the design matrix. We apply the proposed method on datasets of metabolic and microarray profiles collected from a group of subjects consisting of patients with melancholic depression and healthy controls. Results show that our method can not only produce meaningful clusters of variables, but also generate a set of representative features which demonstrate superior discriminative power than those generated by traditional clustering and data imputation techniques. In particular,

Table 2. Classification performance on all genes

Method	SVM Vote				SVM Weighted Vote			
	Acc	Sen	Spe	AUC	Acc	Sen	Spe	AUC
Raw data	0.6411	0.6167	**0.6400**	0.6283	0.6090	0.6167	0.5978	0.6072
KM	0.6333	0.6833	0.6200	0.6517	0.6172	0.7167	0.5878	0.6522
HC	0.6007	0.6333	0.5878	0.6106	0.6167	0.6333	0.6089	0.6211
SC(λ=0.1)	0.6172	**0.7833**	0.5656	0.6744	0.6019	**0.7833**	0.5456	0.6644
SC(λ=0.15)	0.6578	0.7000	0.6389	0.6694	0.6578	0.7500	0.6278	0.6889
SC(λ=0.2)	**0.6668**	0.7667	**0.6400**	**0.7033**	0.6411	0.7667	0.6067	0.6867
SC(λ=0.3)	0.6578	0.7500	0.6289	0.6894	**0.6744**	0.7500	**0.6511**	**0.7006**

Table 3. Classification performance on genes related to depression

Method	SVM Vote				SVM Weighted Vote			
	Acc	Sen	Spe	AUC	Acc	Sen	Spe	AUC
Raw data	0.6822	0.5833	0.7056	0.6444	0.6822	0.5833	0.7056	0.6444
KM	0.6597	0.6333	0.6633	0.6483	0.6284	0.6667	0.6122	0.6394
HC	0.6681	0.6167	0.6756	0.6461	0.6394	0.6283	0.6386	0.6334
SC(λ=0.1)	0.7245	**0.7833**	0.7044	**0.7439**	0.7091	**0.7833**	0.6844	**0.7339**
SC(λ=0.15)	**0.7573**	0.7000	**0.7689**	0.7344	**0.7559**	0.7000	**0.7678**	**0.7339**
SC(λ=0.2)	0.7245	0.7500	0.7167	0.7333	0.7176	0.7000	0.7178	0.7089
SC(λ=0.3)	0.6912	0.7500	0.6733	0.7117	0.6906	0.7500	0.6722	0.7111

Table 4. Classification performance on genes related to immune system

Method	SVM Vote				SVM Weighted Vote			
	Acc	Sen	Spe	AUC	Acc	Sen	Spe	AUC
Raw data	0.6981	0.6167	**0.7156**	0.6661	0.6828	0.6167	0.6956	0.6561
KM	0.7079	0.7167	0.7067	0.7117	0.7079	0.7167	0.7067	0.7117
HC	0.6975	0.7500	0.6822	0.7161	0.6975	0.7500	0.6822	0.7161
SC(λ=0.1)	0.6911	0.7833	0.6622	0.7228	0.6911	0.7833	0.6622	0.7228
SC(λ=0.15)	0.7149	**0.8333**	0.6822	0.7578	0.7309	**0.8667**	0.6933	**0.7800**
SC(λ=0.2)	0.7065	0.7333	0.6933	0.7133	0.7225	0.7833	0.7033	0.7433
SC(λ=0.3)	**0.7399**	**0.8333**	0.7144	**0.7739**	**0.7476**	0.8333	**0.7244**	0.7789

on both datasets, we found that in comparison with those traditional clustering algorithms, feature sets yielded by sparse coding give rise to significantly improved sensitivity scores, suggesting that learned features allow prediction with high accuracy of disease status in those who are diagnosed with melancholic depression.

One interesting future direction is to extend the current method to deal with data with multiple modalities and block-wise missing patterns (i.e., one sample may lack observations on one ore more modalities). Simply concatenating different types of data is not appropriate in this situation since there is a high risk that pseudo-correlation may be detected between variables belonging to different data types which are not really related possibly due to a limited number of observations available on these variables. One direction is to use sparse

coding to simultaneously learn a group of features shared by all data types and individual features specific to each data type.

5. Acknowledgement

This work is supported in part by grants from NIH (R01 LM010730) and NSF (IIS-0953662, IIS-1421057, and IIS-1421100).

References

1. P. J. McGrath, A. Y. Khan, M. H. Trivedi, J. W. Stewart, D. W. Morris, S. R. Wisniewski, S. Miyahara, A. A. Nierenberg, M. Fava and A. J. Rush, *Journal of Clinical Psychiatry* **69**, 1847 (2008).
2. C. M. Mazure, M. B. B. Jr., F. H. Jr., K. B. Miller and J. Nelson, *Biological Psychiatry* **22**, 1469 (1987).
3. V. Gabbay, R. G. Klein, Y. Katz, S. Mendoza, L. E. Guttman, C. M. Alonso, J. S. Babb, G. S. Hirsch, and L. Liebes, *J Child Psychol Psychiatry* **51**, 935 (2010).
4. Z. Liu, X. Li, N. Sun, Y. Xu, Y. Meng, C. Yang, Y. Wang and K. Zhang, *PLOS One* **9**, p. e93388 (2014).
5. D. S. Wishart, T. Jewison, A. C. C. Guo, M. Wilson, C. Knox, Y. Liu, Y. Djoumbou, R. Mandal, F. Aziat, E. Dong, S. Bouatra, I. Sinelnikov, D. Arndt, J. Xia, P. Liu, F. Yallou, T. Bjorndahl, R. Perez-Pineiro, R. Eisner, F. Allen, V. Neveu, R. Greiner and A. Scalbert, *Nucleic acids research* **41**, D801 (January 2013).
6. J. D. Hoheisel, *Nat Rev Genet* **7**, 200 (March 2006).
7. M. R. Segal, K. D. Dahlquist and B. R. Conklin, *Journal of Computational Biology* **10**, 961 (2003).
8. P. Bühlmann, P. Rütimann, S. van de Geer and C.-H. Zhang, *Journal of Statistical Planning and Inference* **143**, 1835 (2013).
9. J. Yang, K. Yu, Y. Gong and T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, 2009.
10. X. Lu, H. Y. Yan, P. Yan, L. Li and X. Li, Image denoising via improved sparse coding, in *Proceedings of the British Machine Vision Conference*, (BMVA Press, 2011). http://dx.doi.org/10.5244/C.25.74.
11. J. Mairal, F. Bach, J. Ponce and G. Sapiro, *J. Mach. Learn. Res.* **11**, 19 (March 2010).
12. B. Lin, Q. Li, Q. Sun, M.-J. Lai, I. Davidson, W. Fan and J. Ye, *CoRR* **abs/1407.8147** (2014).
13. R. Tibshirani, *Journal of the Royal Statistical Society, Series B* **58**, 267 (1994).
14. G. Parker and D. Hadzi-Pavlovic, *Melancholia: A Disorder of Movement and Mood* (Cambridge University Press, New York, 1996).
15. R. Dubey, J. Zhou, Y. Wang, P. M. Thompson and J. Ye, *NeuroImage* **87**, 220 (2014).