World Wide Web (2018) 21:1545–1562
https://doi.org/10.1007/s11280-017-0502-9

Supervised feature selection algorithm via discriminative ridge regression

Shichao Zhang[1] · Debo Cheng[1,2] · Rongyao Hu[1] · Zhenyun Deng[1]

Received: 30 June 2017 / Revised: 18 September 2017 / Accepted: 29 September 2017 /
Published online: 13 October 2017

Abstract This paper studies a new feature selection method for data classification that efficiently combines the discriminative capability of features with the ridge regression model. It first sets up the global structure of training data with the linear discriminant analysis that assists in identifying the discriminative features. And then, the ridge regression model is employed to assess the feature representation and the discrimination information, so as to obtain the representative coefficient matrix. The importance of features can be calculated with this representative coefficient matrix. Finally, the new subset of selected features is applied to a linear Support Vector Machine for data classification. To validate the efficiency, sets of experiments are conducted with twenty benchmark datasets. The experimental results show that the proposed approach performs much better than the state-of-the-art feature selection algorithms in terms of the evaluating indicator of classification. And the proposed feature selection algorithm possesses a competitive performance compared with existing feature selection algorithms with regard to the computational cost.

Keywords Ridge regression · Linear discriminant analysis · Representative coefficient matrix · Support vector machine

This article belongs to the Topical Collection: *Special Issue on Deep Mining Big Social Data*
Guest Editors: Xiaofeng Zhu, Gerard Sanroma, Jilian Zhang, and Brent C. Munsell

✉ Shichao Zhang
zhangsc@mailbox.gxnu.edu.cn

Debo Cheng
cheng7294@foxmail.com

[1] Guangxi Key Lab of Multi-source Information Mining & Security, Guangxi Normal University, Guilin, Guangxi, 541004, China

[2] Information Technology and Mathematical Sciences, University of South Australia, Adelaide, Australia

1 Introduction

Feature selection is one of important research topics in data mining [8, 14]. It is a process of selecting a subset of relevant features for use in data mining applications. A feature selection algorithm can be seen as a two-phase procedure, a search technique for proposing new feature subsets and an evaluation measure which scores the different feature subsets. Feature selection takes into account only those presentative features, and leads to the removal of those irrelevant and redundant features from original features, the dimension reduction, speeding up the training process and reinforcing the model generalization capability [9, 26]. Although feature selection based data mining methods are approximate solutions, they generate models significantly approximate to the real ones. Therefore, feature selection has successfully been applied to, such as dimensionality reduction in computer vision [33, 45, 46], large margin subspace learning in pattern recognition [18], alzheimers disease diagnosis in biological analysis [24, 42, 47].

This paper studies a new feature selection approach using the discriminative capability of features and the ridge regression, denoted as DRR-FS, aiming at improving the efficiency and generalization capability of feature selection algorithms. It first takes into account the global structures of the data distribution with the Linear Discriminant Analysis (LDA) [31], with respect to the class label information regarded as the global discrimination information embedded into the DRR-FS approach. And then, the ridge regression model is incorporated into the DRR-FS algorithm, so as to yield a coefficient matrix for measuring the importance of features. And the rank of these features is generated based on the coefficient matrix [27, 34]. Finally, the important feature subsets are selected as the new feature subset to instead of the original set of features, and applied to the linear Support Vector Machine(SVM) for data classification [38, 44, 48].

To evaluate the DRR-FS approach, sets of experiments were conducted with seven artificial datasets and twenty open datasets downloaded from the UCI machine learning repository and the feature selection website. The experimental results illustrated the efficiency of the designed algorithm for binary and multi-class classification tasks in terms of selecting the discriminative features, as well as the strengths of the DRR-FS approach as follows.

- The ridge regression model is as a biased estimate model for evaluating the error between feature variables and the class label information. The experiments showed the effectiveness of evaluating the represented coefficient error, as well as efficiently avoiding the over-fitting of the learning process.
- The global structure information of training examples is established with the LDA method taking into account the class labels. The experiments showed the efficiency of identifying the discriminative features after embedding the global structure information into the DRR-FS approach.

The remainder of this paper is organized as follows. Section 2 recalls the work related to this research. Section 3 elaborates the DRR-FS method. Section 4 describes experimental design and the analysis of the experimental results. Finally, conclusions this work and future work are presented in Section 5.

2 Related work

As well known, data mining is the necessity theoretical analysis for the core of big data [6, 23, 25] and multi-source data [40]. Various kinds of data mining algorithms based on

different kinds of data types and formats that can be more scientific exhibit the intrinsic value of the data. And they are precisely estimate data because these are the world's statisticians universally recognized statistical methods that is to go to deep in internal data and excavated recognized value [35, 36, 49]. In order to extract the real data of great value, data pre-processing is an important and essential part of data mining. Simultaneously, feature selection as an important step in data pre-processing which has become a very hot topic [5].

In many real applications, unlabeled data is ubiquitous due to label data is a time and labor consuming process. Therefore, Feature selection algorithms according to the method of the label information whether are used for model or not, which extensively worked in three categories: supervised feature selection algorithms, unsupervised feature selection algorithms, and semi-supervised feature selection algorithms [41, 43].

Supervised feature selection algorithms utilize the correlation between feature and class to distinguish the importance of feature. The typical supervised feature selection methods include: Laplacian Score(LapScore) for feature selection [13] and Fisher Score for feature selection (fsFisher) [7]. With sparse representation has been successful applied to feature selection, many regularization formulations have been proposed [19, 29], and Nie et al. proposed to joint $\ell_{2,1}-$norm minimization on both loss function and regularization for feature selection [20]. Recently, Zhu et al. proposed a novel supervised dimensionality reduction method that Self-taught on the high-dimensional small-sized data and obtained a better performance than compared algorithms [39]. Germain et al. used the non-negative matrix factorization(NMF) based algorithms to iteratively optimize the cost function, and proposed several heuristic stopping criteria to find well correlated with source separation performance [10].

Unsupervised feature selection algorithms usually take the geometrical structure of the data distribution into account for selecting the important features. In general, unsupervised feature selection algorithms are much hard to study due to the lack of labels. However, there are many typical algorithms have been proposed in these categories, such as maximum variance, unsupervised feature selection Principal Component Analysis(PCA) [12], and unsupervised feature analysis with class margin optimization [30] and so on. Maximum variance method in terms of the largest variances of the features to select the important features, such as unsupervised feature selection method based on PCA use the PCA method to select a subset of features that can best reconstruct other features [12]. Wang et al. proposed a new unsupervised feature analysis method that uses the Maximum Margin criterion and the sparsity-based model to construct a robust unsupervised feature selection framework [30]. Nonnegative Discriminative Feature Selection(NDFS) is proposed by Li et al. [17]. NDFS effectively combine the learning of the cluster labels and feature selection matrix which could enhance to select the important discriminative features.

Semi-supervised feature selection algorithms are the effective means for dealing with the large data which includes unlabeled data is large and labeled data is small. In last decades, many semi-supervised feature selection algorithms have been proposed. A novel semi-supervised feature selection method was proposed by Pierre et al. based on spectral analysis [22]. Bellal et al. proposed a new method which combines a bagged ensemble of standard semi-supervised approaches with a permutation-based out-of-bag feature importance measure that takes into account both labeled and unlabeled data [2]. Zeng et al. proposed a novel semi-supervised feature selection method which uses linear regression to model the correlations between the data samples with supervision information and their class labels, and then, uses $\ell_{2,1}-$norm to guarantee the sparsity of the feature selection matrix and exploit the sharing information between supervised and unsupervised data samples jointly [32]. Alalga et al. proposed a unified framework for semi-supervised multi-label feature selection, based on

Laplacian score [1]. The method transforms the labeled part of data into soft constraints and shows how to integrate them in a measure of feature relevance, according to the available labels.

3 Proposed approach

3.1 Notation

Throughout the paper, we denote matrices as boldface uppercase letters, vectors as boldface lowercase letters and scalars as normal italc letters, respectively. For a matrix $\mathbf{X} = [x_{ij}]$, its i-th row and j-th column are denoted as x^i and x_j, respectively. Also, we denote the Frobenius norm as $\|\mathbf{X}\|_F = \sqrt{\sum_i \|\mathbf{x}^i\|_2^2} = \sqrt{\sum_j \|\mathbf{x}_j\|_2^2}$. We further denote the transpose operator, the trace operator and the inverse of a matrix $\mathbf{X}$ as $\mathbf{X}^T$ and $\mathbf{X}^{-1}$, respectively.

3.2 Discriminative feature selection algorithm

In this section, we study a supervised learning algorithm via selecting the compactly discriminative features from the original features for improving the classified performance of the high-dimensional data. Given $\mathbf{X} \in \mathbb{R}^{n\times d}$ represents a data matrix, where n and d are the numbers of the sample variables and the numbers of the feature variables, respectively. And $\mathbf{Y} \in \mathbb{R}^{n\times c}$ denotes the indicator matrix of the class label, where c is the number of the classes. Suppose as follows:

$$\mathbf{Y} = \mathbf{X}w_{ij} + \varepsilon \tag{1}$$

where ε represents the representation error, and w_{ij} denotes the representation coefficient element. Therefore, we rewrite it through its matrix form.

$$\mathbf{Y} = \mathbf{XW} + \mathbf{E} \tag{2}$$

where $\mathbf{E}$ denotes representation error matrix and $\mathbf{W}$ is the representation coefficient matrix. For ensuring the representation error $\mathbf{E}$ as small as possible, we use $\mathbf{Y} - \mathbf{XW}$ to evaluate it and employ it as the loss term. For making the solution of the loss term always reversible, so we employ a penalty term in the loss function to avoid it. So the fundamental framework of this work becomes:

$$\min_{\mathbf{W}} \phi(\mathbf{Y} - \mathbf{XW}) + \lambda\varphi(\mathbf{W}) \tag{3}$$

In view of the successful application of ridge regression algorithm in data mining [21], and it also satisfies the (3). Hence, in this paper, we rewrite the ridge regression model for adapting to this work. So the object function of this work as follows:

$$\min_{\mathbf{W}} \|\mathbf{Y} - \mathbf{XW}\|_F^2 + \lambda \|\mathbf{W}\|_F^2 \tag{4}$$

where λ is a positive parameter for tuning the model, and $\mathbf{W} \in \mathbb{R}^{d\times c}$ is a coefficient matrix which is reflects the relations between the label and features. The optimal solution of (4) can be described as a closed solution $\mathbf{W} = (\mathbf{XX}^T + \delta\mathbf{I})^{-1}\mathbf{XY}$, where $\mathbf{I} \in R^{n\times n}$ is an identity matrix. And the time complexity of the (4) is $\mathbf{O}(n^2)$. Different form the traditional ridge regression algorithms, the (4) is a feature-level model for feature selection algorithm.

For taking the label information into account, *i.e.,* the global structures of the data for improving the performance of our algorithm. Therefore, we think that employs a Fisher's

LDA [16] for our object function, which considers the global data distribution based on between within class variance and between class variance to find the main relevant class. However, the Fisher's LDA penalize term $\frac{\mathbf{W}^T\sum_g\mathbf{W}}{\mathbf{W}^T\sum_h\mathbf{W}}$ is the non-convexity, where $\sum_g$ denotes the within-class variance and $\sum_h$ denotes the between-class variance. Fortunately, Ye [31] has proposed a multivariate linear regression model that defines the class label matrix $\mathbf{Y} = [y_{i,k}]$ to replace the Fisher's LDA penalized term.

$$y_{i,k} = \begin{cases} \sqrt{\frac{n}{n_k}} - \sqrt{\frac{n_k}{n}} & if\ l(x_i) = k \\ -\sqrt{\frac{n_k}{n}} & otherwise \end{cases} \tag{5}$$

where n_k denotes the sample size of the class k and $l(x_i)$ is a class label of x_i. So we can efficiently use the global structure of the data via the class indicator matrix $\mathbf{Y}$, and cannot transform the original features space into a low dimensional space. Hence, the matrix $\mathbf{Y}$ of the (4) comes from the (5) for considering the global structure of the data.

4 Experimental analysis

To demonstrate the validity of the DRR-FS algorithm, we perform it on twenty open datasets. The datasets are from the UCI machine learning repository[1] and the feature selection website.[2] These datasets include ten binary-class problems and ten multi-class problems. In order to compactly represent the name of the datasets, we named Hill-valley-withnoise, GLA-BRA-180, SMK-CAN-187, CLL-SUB-111 as Hill-valley, GLA-BRA, SMK-CAN, CLL-SUB, respectively. Then we list the details of the experimental data in Table 1.

Furthermore, we also construct seven artificial data sets which include clean data and noisy data by the literature [4] for regression experiment and we list the details of artificial datasets in Table 2. And the method for generating the artificial data sets as follows. s We constructed these artificial data by the model of $\mathbf{Y} = \mathbf{XBA}^T + \mathbf{E}$. To be specific, for the $n \times p$ matrix $\mathbf{X}$ which was constructed by multivariate normal distribution N(0, Σ_x) and Σ_x was composed of diagonal elements 1 and off-diagonal elements ρ_x. And for the $p \times r$ matrix $\mathbf{B}$ that its first p_0 rows were constructed by N(0, 1) and the remain $p - p_0$ rows were set to be zero. Then the $q \times r$ matrix $\mathbf{A}$ was generated from N(0, 1). Ultimately, for the $n \times q$ random noise matrix $\mathbf{E}$ is constructed by N(0, $\sigma^2\Sigma_e$) and Σ_e has off-diagonal elements ρ_e and diagonal elements 1. However, the magnitude of the noise σ^2 has some provisions, i.e., $\frac{trace(C^T\Sigma_x C)}{trace(E)} = 1$. Furthermore, we will set different parameters by the size of n and p, i.e., $n > p$ and $n \leq p$.

4.1 Experimental Setting

In our experiments, we employ fsFisher, fsTest, LapScore, L21R21, SD, MI algorithms as our compared algorithms. We introduce them as follows:

fsFisher [11]: This method used Fisher's score to evaluate for each feature individually, and then sorted the features in an ascending order according to the score.

[1] http://archive.ics.uci.edu/ml/

[2] http://featureselection.asu.edu/datasets.php

Table 1 The detail of open datasets

Datasets	Samples(n)	Features(d)	Class
Breastcancer	569	30	2
Covertype	571	54	2
GLI-85	85	22283	2
Hill-valley	1212	100	2
Ionosphere	351	34	2
Spambase	4601	57	2
Sonar	208	60	2
Splice	1000	60	2
Specft	80	44	2
SMK-CAN	187	19993	2
CLL-SUB	111	11340	3
GLA-BRA	180	49151	4
TOX-171	171	5748	4
Landsat	4435	36	6
Satimage	620	36	6
orlraws10P	100	10304	10
pixraw10P	100	10000	10
warpAR10P	130	2400	10
warpPIE10P	210	2420	10
Isolet	7797	617	26

fsTest [28]: This method used Ttest's score for each feature to evaluation individually. Moreover, it sorted the features according to the score in an ascending order.

LapScore [15]: This conducted supervised feature selection with an assumption that data of the same class tend to distribute to each other in the feature space while those of different classes are apart from each other. The importance of a feature is evaluated by its power of a Laplacian score.

L21R21 [37]: This method utilized the sample self-representation structure to select an representation response matrix, then get the proposed structure embed into the sparse learning model for feature selection. The importance of a feature is evaluated by its coefficient in the response matrix.

Table 2 The detail of artificial datasets

Datasets	Samples(n)	Features(d)	Class
1a-data	100	30	10
1b-data	100	30	10
1c-data	100	30	10
1d-data	100	30	10
2a-data	100	100	10
2b-data	100	300	30
3-data	100	100	10

SD [3]: This method got a mapping function to process the original data, then the high scores and low scores correspond to the importance of the strength and the property. Ultimately, extracted features by the top-ranking scores.

MI [3]: This method used the logarithmic compression to process the original data from the beginning, then there was a mapping function utilized mapping process for the original data. Lastly, it can get the importance of features by the top-ranking scores.

In these experiments, we employ 10-fold partitioning and cross-validation of the data. In other words, each dataset is split randomly into ten subsets, and one of those sets is reserved as a test set, the rest data is regarded as training set, the whole process of partitioning is done in a completely random and arbitrary manner. The classification accuracy on each dataset was obtained via 10-fold cross validation. Running with the various algorithms were carried out on the same training sets and evaluated on the same test samples. We repeated the whole process ten times to avoid the possible bias in our experiments. Our experiments carried out in MATLAB2012a on windows 7 system running on a PC.

We used classification accuracy as the evaluation for the classification task. The classification accuracy is defined as follows:

$$Accuracy = \frac{N_{correct}}{N} \tag{6}$$

where N is the number of all samples, $N_{correct}$ is the number of correct classification samples. The higher accuracy the algorithm is, the better performance of classification it is.

In additional, we used other three additional measures (*e.g.,* sensitivity (SEN), specificity (SPE) and Area Under Curve (AUC)) to evaluate binary classification. In binary classification, the outcomes were denoted as either Positive (P) or Negative (N). We parted the results into four groups, i.e., True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). Therefore, we defined SEN as:

$$SEN = \frac{N_{TP}}{N_P} \tag{7}$$

where N_{TP} means the number of TP, and N_P means the number of actual positive labels.

SPE was defined as:

$$SPE = \frac{N_{TN}}{N_N} \tag{8}$$

where N_{TN} means the number of TN, and N_N means the number of actual negative labels.

AUC was defined as:

$$AUC = \frac{S_0 - \frac{N_P(N_P+1)}{2}}{N_P N_N} \tag{9}$$

where $S_0 = \sum_{i=1}^{N_P} r_i$ and r_i denotes the i-th positive sample sort position.

Furthermore, the average correlation coefficient (aCC) and average root mean squared error (aRMSE) were employed to evaluate the performance for regression analysis. The aCC evaluation index indicates the correlation between prediction and observation, and its formula is defined as follows:

$$aCC = \frac{1}{d}\sum_{i=1}^{d} CC = \frac{1}{d}\sum_{i=1}^{d} \frac{\sum_{l=1}^{N_{test}} (y_i^{(l)} - \bar{y}_i)(\hat{y}_i^{(l)} - \bar{\hat{y}}_i)}{\sqrt{\sum_{l=1}^{N_{test}} (y_i^{(l)} - \bar{y}_i)^2 \sum_{l=1}^{N_{test}} (\hat{y}_i^{(l)} - \bar{\hat{y}}_i)^2}} \tag{10}$$

Table 3 The performance of the binary classification in terms of Accuracy (mean±STD)(/%)

Datasets	fsFisher	fsTtest	LapScore	L21R21	SD	MI	DRR-FS
Breastcancer	75.82±13.5	75.99±13.7	75.63±13.2	75.44±12.97	75.11±12.6	75.81±13.45	**77.78±4.83**
Covertype	75.21±3.05	69.18±6.04	73.51±4.92	72.62±6.69	75.90±4.00	75.40±2.63	**79.43±4.33**
GLI-85	92.08±8.74	89.44±12.94	89.54±9.34	89.58±7.86	90.42±7.39	90.83±8.41	**94.03±5.99**
Hill-valley	97.11±1.86	**97.53±1.22**	79.46±2.7	88.77±5.30	97.35±1.85	**97.53±1.32**	94.15±1.73
Ionosphere	92.32±4.41	91.45±4.24	84.31±3.59	91.78±4.71	93.15±3.68	93.14±3.95	**94.40±3.46**
Spambase	85.42±1.45	82.63±1.88	82.79±1.87	84.44±1.80	81.98±3.53	82.66±3.39	**89.56±0.34**
Sonar	83.09±6.68	84.61±5.89	79.31±4.86	81.73±6.45	81.71±7.59	83.71±5.65	**86.09±3.23**
Splice	82.50±4.46	77.69±1.58	80.50±2.85	76.99±4.60	82.09±3.77	81.71±3.28	**84.30±2.75**
Specft	90.0±12.25	87.50±9.68	78.75±8.00	90.0±10.9	91.25±8.0	90.0±9.35	**93.75±6.25**
SMK-CAN	80.23±6.67	79.15±7.26	79.15±6.50	73.83±9.30	82.34±5.40	78.63±9.58	**85.00±6.04**

The significance of bold denotes the best performance among seven algorithms in terms of each evaluation index

Table 4 The performance of the binary classification in terms of SEN, SPE, AUC(mean±STD)(/%)

Datasets	EI	fsFisher	fsTtest	laplacian	L21R21	SD	MI	DRR-FS
Breast cancer	SEN	98.71±1.25	98.54±1.28	98.44±1.43	99.41±1.17	98.45±1.35	98.89±1.19	**99.43±1.14**
	SPE	35.78±36.31	**36.27±37.0**	35.71±36.22	34.92±35.19	34.76±35.0	36.09±36.75	34.29±41.12
	AUC	86.28±28.02	86.26±28.08	86.15±28.41	86.63±26.98	86.87±26.24	86.22±28.19	**97.12±1.63**
Cover type	SEN	76.25±9.47	53.35±13.81	68.11±9.77	72.72±9.79	74.33±6.89	75.44±6.14	**82.77±6.18**
	SPE	75.64±8.47	**88.22±10.6**	79.52±6.53	72.03±10.10	76.51±6.39	75.74±7.40	75.44±7.78
	AUC	74.04±18.2	72.10±16.00	74.79±15.84	71.39±16.08	71.72±17.63	73.42±17.93	**82.61±3.72**
GLI-85	SEN	85.0±18.93	80.0±33.99	81.67±24.09	80.0±25.6	**88.33±18.3**	81.67±24.09	86.83±15.05
	SPE	95.0±10.67	93.0±8.62	93.0±8.62	93.33±11.06	91.67±11.18	95.0±7.64	**96.47±3.67**
	AUC	93.72±10.13	93.78±9.11	90.0±14.01	94.17±7.60	90.56±12.62	90.28±10.63	**96.11±5.58**
Hill-valley	SEN	96.67±2.41	**96.88±1.70**	77.43±4.22	88.83±5.07	96.41±2.02	96.40±3.08	81.92±9.91
	SPE	97.51±2.34	98.21±1.33	81.63±4.75	88.78±6.11	98.35±1.43	**98.68±1.42**	81.36±14.04
	AUC	10.73±29.10	10.44±28.93	22.57±21.73	15.34±27.69	10.53±29.38	10.55±29.73	**97.28±0.49**
Iono sphere	SEN	96.38±4.44	95.75±4.25	96.66±5.66	96.43±1.83	97.41±2.70	97.92±2.65	**99.25±0.40**
	SPE	85.15±10.31	82.72±9.27	61.46±5.53	84.49±10.25	**86.78±9.78**	85.59±7.36	66.02±3.23
	AUC	83.27±26.82	84.99±25.77	74.49±18.56	87.75±24.59	83.16±27.87	85.02±27.23	**90.78±1.99**
Spam base	SEN	80.46±2.82	75.54±3.21	75.25±2.67	78.59±2.02	90.26±8.07	92.21±6.26	**94.93±0.36**
	SPE	93.05±1.38	93.35±1.78	**94.44±1.41**	93.51±2.42	69.68±13.93	67.86±11.64	81.30±1.10
	AUC	5.72±0.85	7.91±1.18	7.25±1.10	6.72±1.35	8.18±1.46	7.83±0.63	**94.50±0.33**
Sonar	SEN	86.87±10.87	**91.96±6.36**	88.04±7.09	85.55±9.45	86.11±8.20	85.19±8.56	89.51±6.32
	SPE	79.05±13.02	76.08±12.60	68.22±15.76	77.60±15.33	76.04±13.55	**83.23±9.76**	79.22±8.26
	AUC	14.47±8.34	16.37±9.99	21.98±6.95	16.36±9.10	18.20±9.24	12.04±6.23	**82.79±2.28**
Splice	SEN	79.88±5.75	71.16±4.83	77.72±4.42	73.63±5.92	79.75±4.42	80.15±5.25	**80.46±6.25**
	SPE	85.07±5.11	84.83±5.17	83.69±3.42	80.59±6.46	84.66±5.10	83.70±5.55	**86.34±5.23**
	AUC	88.75±4.50	81.64±2.77	86.60±3.97	82.01±5.44	88.03±2.93	87.86±3.42	**89.09±2.69**

Table 4 (continued)

Datasets	EI	fsFisher	fsTtest	laplacian	L21R21	SD	MI	DRR-FS
Specft	SEN	90.0±16.58	90.0±12.25	87.5±16.77	90.0±16.58	**95.0±10.00**	**95.0±10.00**	91.25±6.37
	SPE	90.0±16.58	85.0±16.58	70.0±17.71	90.0±16.58	85.0±12.5	87.5±15.58	**97.50±7.50**
	AUC	11.88±16.41	8.13±8.41	21.88±9.78	75.0±8.75	5.63±9.04	8.75±9.35	**79.41±7.12**
SMK-CAN	SEN	75.56±12.96	74.44±10.00	75.56±11.97	68.89±13.88	**80.00±12.9**	74.44±15.75	53.67±7.03
	SPE	84.78±12.29	83.44±13.68	82.33±9.28	78.44±12.29	84.33±7.58	82.56±9.19	**93.98±1.84**
	AUC	82.06±7.83	84.85±7.38	80.01±6.69	76.38±13.01	**83.11±9.60**	81.15±10.13	78.29±3.44

The significance of bold denotes the best performance among seven algorithms in terms of each evaluation index

In the following, aRMSE is defined as the squared root of predicted value and the ground-truth. Its formula is defined as follows:

$$aRMSE = \frac{1}{d}\sum_{i=1}^{d} RMSE = \frac{1}{d}\sum_{i=1}^{d}\sqrt{\frac{\sum_{l}^{N_{test}}(y_i^{(l)}-\hat{y}_i^{(l)})^2}{N_{test}}} \tag{11}$$

where N_{test} means the size of test data set, then $y_i^{(l)}$ and $\hat{y}_i^{(l)}$ denote the vectors of the actual and predicted targets for $x^{(l)}$, respectively. Besides, $\bar{y}$ and $\bar{\hat{y}}$ be the vectors of averages of the actual and predicted targets, respectively. A larger aCC shows better correlation coefficient results, while a smaller aRMSE means better robust.

4.2 Experimental results

In this section, we evaluated the performance of the DRR-FS approach by comparing two comparison algorithms on real and artificial datasets, in terms of two data mining tasks, i.e., classification and regression.

4.2.1 Binary classification results

We summarized the experimental results of the binary classification Accuracy in Table 3, and the results of the binary classification in terms of SEN, SPE, AUC in Table 4. Besides, we also summarized the time consuming of binary data sets in Table 5.

From Table 3, we easily recognized that the proposed DRR-FS approach achieved the best classification accuracy in the most of methods for binary classification tasks except Hill-valley dataset. On the Hill-valley dataset, both fsTtest method and MI method have the same performance compared with our method that only increased 3.38%. Especially, on the binary datasets of GLI-85, Hill-valley, Ionosphere, Specft that our method gets the better results more than 90%. In addition, MI and SD compared with fsFisher and fsTtest which can have better effects, that is to say the general mapping method is superior to the conducted score method on most of the binary classification tasks. Moreover, our method also got the best performance by SEN, SPE and AUC evaluation indices on the most binary

Table 5 The performance of the binary classification in terms of time consuming (/seconds)

Datasets	fsFisher	fsTtest	LapScore	L21R21	SD	MI	DRR-FS
Breastcancer	0.0065	5.8e-03	0.0283	0.0389	0.0042	3.3e-03	**1.1e-03**
Covertype	0.0063	0.0011	0.0189	0.0353	0.0097	0.0102	**3.7e-04**
GLI-85	4.3200	0.1987	0.2103	3.4530	2.9570	3.0423	**0.0352**
Hill-valley	0.0147	0.0035	0.1228	0.2186	0.0216	0.0225	**5.8e-03**
Ionosphere	0.0035	4.2e-04	0.0217	0.0208	0.0032	0.0042	**2.4e-04**
Spambase	0.0123	**0.0034**	0.9210	0.3118	0.0271	0.0276	0.0142
Sonar	0.0133	0.0011	0.0287	0.0969	0.0100	0.0101	**3.6e-04**
Splice	0.0117	0.0028	0.0815	0.0947	0.0155	0.0156	**1.6e-04**
Specft	0.0075	7.1e-04	0.0157	0.0561	0.0070	0.0112	**1.6e-04**
SMK-CAN	2.2573	0.2143	0.2596	12.1924	1.5738	2.2955	**0.0496**

The significance of bold denotes the best performance among seven algorithms in terms of each evaluation index

Table 6 The performance of multi-class classification in terms of accuracy (mean±STD) (/%)

Datasets	fsFisher	fsTtest	LapScore	L21R21	SD	MI	DRR-FS
CLL-SUB	78.38±7.16	65.32±17.40	72.98±6.80	73.06±13.68	78.30±8.27	80.94±7.95	**85.76±4.24**
GLA-BRA	73.22±8.58	72.86±11.15	74.94±9.00	66.62±6.72	71.03±6.12	67.98±10.74	**75.79±5.83**
TOX-171	98.30±2.60	98.30±2.60	97.71±3.75	98.34±2.34	98.58±1.91	98.26±3.64	**98.77±1.75**
Landsat	80.38±1.90	80.49±1.49	80.25±2.12	67.98±1.26	79.21±2.06	79.51±2.39	80.97±1.42
Satimage	86.43±3.89	82.57±6.25	86.15±4.10	84.51±4.38	84.86±2.99	84.70±3.45	**87.12±3.55**
orlraws10P	88.68±3.58	87.64±4.08	86.34±3.88	83.25±4.24	87.05±4.21	87.53±4.83	**92.90±2.55**
pixraw10P	93.17±3.15	**95.52±2.31**	93.73±3.51	86.35±3.15	88.21±5.32	87.63±4.88	95.37±1.47
warpAR10P	90.05±3.74	89.61±4.13	90.81±2.88	88.71±3.57	91.21±3.54	91.75±3.10	**97.83±1.12**
warpPIE10P	**96.63±2.51**	94.31±2.11	93.71±3.01	91.23±4.21	92.51±3.31	92.13±3.70	96.54±2.42
Isolet	95.58±0.64	92.47±0.9	93.05±1.08	89.75±1.31	95.06±0.84	95.33±0.52	**96.07±0.19**

The significance of bold denotes the best performance among seven algorithms in terms of each evaluation index

datasets in Table 4. Ultimately, we compared the time consuming for each algorithm modeling of each desired in Table 5, on the Spambase dataset that our method could not get the short time, but compared with other datasets that our DRR-FS approach could get the shortest time consuming.

4.2.2 Multi-class classification results

We summarized the experimental results of the multi-class classification accuracy in Table 6, and the results of the multi-class classification in terms of time consuming in Table 7.

As for the performance of multi-class classification tasks and time consuming, respectively, showed the best results in Tables 6 and 7. Such as, except the pixraw10P and warpPIE10P dataset, compared with our method only increased by 0.15% and 0.09%, respectively, appeared in the fsTtest and fsFisher algorithm. At the same time, our method got the lower results than 90% in the CLL-SUB, GLA-BRA, Landsat, Satimage dataset, but the proposed method still had the best effects compared other methods. Furthermore, compared with the binary tasks, in contrast, the general conducted score method is superior to the mapping method, i.e., the fsFisher and fsTtest compared with MI and SD which have better effects due to the multi-class classification tasks. From Table 7, it is showed the modeling time of one cross-validation of all methods that our method can get the shortest time consuming for the most of multi-class datasets. However, in the Lamdsat, warpAR10P and Isolet dataset, our method only get the second least time consuming.

4.2.3 Multi-output regression results

We summarized the artificial datasets results of the multi-output regression with aCC and aRMSE evaluation indices, respectively, correspond to the clean and noisy data in Tables 8 and 9. And the evaluation of different methods is based on two widely used regression metrics, i.e., aCC (average correlation coefficient) and aRMSE (average root mean squared error).

From Table 8, it showed that the performance of multi-output regression with clean data by aCC and aRMSE evaluation indices. In our method that it got lower results on 2a-data,

Table 7 The performance of the multi-class classification in terms of time consuming (/s)

Datasets	fsFisher	fsTtest	laplacian	L21R21	SD	MI	DRR-FS
CLL-SUB	2.7987	0.1067	0.1755	2.1900	1.6475	1.668	**0.026**
TOX	2.1753	0.0614	0.1306	3.9796	0.8769	0.8693	**0.0248**
GLA-BRA	13.4785	5.2967	0.3116	20.1474	4.6235	6.0365	**0.2319**
Landsat	0.0151	**0.0015**	0.8347	0.1779	0.0183	0.0226	0.0198
Satimage	0.0219	0.0011	0.0501	0.0407	0.0098	0.0106	**0.0011**
orlraws10P	8.3125	2.1723	0.5231	2.8121	0.8102	0.6580	**0.0140**
pixraw10P	13.8171	4.6521	0.7620	3.8462	1.3512	0.9412	**0.0350**
warpAR10P	5.2581	0.7842	**0.0257**	0.2812	2.2710	2.0764	0.0287
warpPIE10P	3.5414	2.0681	0.5610	0.8741	0.7544	0.8741	**0.1202**
Isolet	0.7949	**0.0352**	2.8202	9.0898	0.8374	0.9245	0.0652

The significance of bold denotes the best performance among seven algorithms in terms of each evaluation index

Table 8 The performance of the multi-output regression with clean data in terms of aCC and aRMSE(mean±STD) (/%)

Datasets	EI	fsFisher	fsTtest	laplacian	L21R21	SD	MI	DRR-FS
1a-data	aCC	97.±2.57	96.88±1.33	94.25±3.58	84.37±14.78	96.12±2.12	97.35±3.87	**99.99±0.003**
	aRMSE	2.28±1.21	3.21±2.04	2.87±1.48	1.18±0.81	2.34±0.98	1.73±1.27	**0.03±0.008**
1b-data	aCC	98.12±0.04	98.24±0.15	97.88±0.77	85.25±9.79	95.77±1.38	96.42±2.01	**99.99±0.002**
	aRMSE	1.01±0.006	0.91±0.002	0.47±0.004	1.36±0.60	3.48±0.77	2.74±1.32	**0.03±0.005**
1c-data	aCC	97.87±0.001	98.54±0.004	98.10±0.01	90.68±7.54	97.51±0.71	96.35±1.74	**99.45±0.003**
	aRMSE	1.07±0.03	0.61±0.07	0.43±0.77	0.98±0.68	0.51±0.06	0.71±0.35	**0.27±0.06**
1d-data	aCC	96.47±0.002	96.13±0.004	97.12±0.01	89.10±11.79	95.23±4.01	96.74±2.81	**99.99±0.003**
	aRMSE	0.43±0.007	0.67±0.01	0.51±0.02	0.82±0.55	0.82±0.09	0.94±0.15	**0.03±0.006**
2a-data	aCC	90.75±3.01	91.21±4.15	90.71±3.81	79.29±10.48	88.64±3.51	87.51±2.59	**94.50±2.46**
	aRMSE	1.24±0.67	2.04±0.78	1.89±1.07	1.76±0.44	0.79±0.35	1.01±0.71	**0.97±0.12**
2b-data	aCC	64.58±6.57	63.57±9.54	64.27±10.48	59.02±14.27	61.05±7.98	62.54±9.31	**65.26±11.33**
	aRMSE	3.79±1.78	3.44±0.79	4.24±1.67	2.57±0.20	5.21±2.10	4.87±1.99	**2.41±0.21**
3-data	aCC	74.48±4.67	73.94±5.47	75.02±6.42	71.50±8.28	72.87±3.22	73.15±4.11	**77.42±3.19**
	aRMSE	4.54±0.78	3.02±1.47	2.98±0.45	2.02±0.20	3.84±1.00	3.21±0.46	**1.79±0.13**

The significance of bold denotes the best performance among seven algorithms in terms of each evaluation index

Table 9 The performance of the multi-output regression with noisy data in terms of aCC and aRMSE(mean±STD) (/%)

Datasets	EI	fsFisher	fsTtest	laplacian	L21R21	SD	MI	DRR-FS
1a-data	aCC	20.88±11.76	20.72±10.83	19.47±10.79	18.53±10.25	16.84±11.8	16.03±11.8	**23.76±11.45**
	aRMSE	3.11±0.28	3.14±0.26	3.19±0.30	3.05±0.15	3.08±0.26	3.10±0.23	**3.01±0.27**
1b-data	aCC	**19.59±4.98**	17.86±5.55	17.48±5.86	16.50±5.69	11.29±5.90	13.44±5.91	13.70±5.83
	aRMSE	3.35±0.29	3.33±0.30	3.32±0.28	3.33±0.22	3.34±0.27	3.32±0.26	**3.31±0.23**
1c-data	aCC	**23.29±7.88**	21.11±8.22	21.57±7.95	21.60±8.97	16.29±8.09	13.99±8.11	20.81±8.75
	aRMSE	2.90±0.20	2.88±0.22	2.83±0.23	2.80±0.21	2.83±0.19	2.86±0.18	**2.80±0.18**
1d-data	aCC	33.47±7.56	33.67±6.84	34.02±7.14	29.98±7.07	26.27±9.28	27.47±9.31	**37.46±3.08**
	aRMSE	3.30±0.39	3.29±0.40	3.27±0.33	3.28±0.36	3.27±0.35	3.30±0.37	**3.25±0.47**
2a-data	aCC	7.65±6.23	7.24±7.70	7.43±6.89	5.14±5.15	6.64±9.23	6.32±9.11	**8.01±7.68**
	aRMSE	2.21±0.18	2.18±0.16	2.11±0.17	2.09±0.18	2.08±0.19	2.11±0.20	**2.08±0.15**
2b-data	aCC	6.88±5.40	6.97±5.74	7.21±6.23	5.64±3.99	5.55±4.71	5.43±4.70	**8.64±7.41**
	aRMSE	2.75±0.13	2.82±0.15	2.77±0.18	2.74±0.12	2.77±0.11	2.80±0.12	**2.73±0.09**
3-data	aCC	15.53±8.12	15.11±7.67	14.87±7.33	14.13±7.55	12.08±9.03	12.53±9.00	**16.27±6.88**
	aRMSE	2.88±0.23	2.84±0.22	2.87±0.21	2.71±0.27	2.78±0.24	2.72±0.22	**2.69±0.20**

The significance of bold denotes the best performance among seven algorithms in terms of each evaluation index

2b-data and 3-data dataset compared with other datasets which average on 99.985%, respectively, decreased by 5.485%, 34.725% and 22.565%. However, the proposed method still achieved best consequences than any other contrast algorithms. Moreover, our DRR-FS approach can get the best results with aCC and the less results with aRMSE. Furthermore, in Table 9, we can come into the conclusion that the proposed DRR-FS approach outperforms other methods on most of the multi-output regression with seven noisy artificial datasets in terms of aCC and aRMSE evaluation indices. Due to the aCC evaluation that the DRR-FS approach gets the lower results on the 1b-data and 1c-data set in accordance with fsFisher method. Since all artificial datasets are constructed with too many noisy features and outlier samples to make useful features hardly extract. However, with aRMSE evaluation index, our method has the best performance compared with all methods among multi-output datasets.

In a word, our proposed method achieved the best performance, compared to the comparison methods on different kinds of classes classification and regression tasks. The reason is that the DRR-FS approach takes full account of the depth structure of the coefficient matrix between the number of samples and the number of features are unequal, so that we can get two corresponding different solutions.

5 Conclusion & future work

In this work, we focused on the problem of discriminative feature selection and ridge regression that samples and features have unequal numbers. The proposed method utilizes LDA and ridge regression method to deeply analysis of the number of samples and the number of features that have a deeply analysis of the representative coefficient matrix. In the future work, we will consider this correlation application for semi-supervised and unsupervised structure.

Acknowledgments This work was supported in part by the China Key Research Program (Grant No: 2016YFB1000905), the Nation Natural Science Foundation of China (Grants No: 61573270, 61672177, and 61363009), National Association of public funds, the Guangxi Natural Science Foundation (Grant No: 2015GXNSFCB139011), the Guangxi High Institutions Program of Introducing 100 High-Level Overseas Talents, the Guangxi Collaborative Innovation Center of Multi-Source Information Integration and Intelligent Processing, the Research Fund of Guangxi Key Lab of MIMS (16-A-01-01 and 16-A-01-02), and the Guangxi Bagui Teams for Innovation and Research.

References

1. Alalga, A., Benabdeslem, K., Taleb, N.: Soft-constrained laplacian score for semi-supervised multi-label feature selection. Knowl. Inf. Syst. **47**(1), 75–98 (2016)
2. Bellal, F., Elghazel, H., Aussem, A.: A semi-supervised feature ranking method with ensemble learning. Pattern Recogn. Lett. **33**(10), 1426–1433 (2012)
3. Borchani, H., Varando, G., Bielza, C., Larrañaga, P.: A survey on multi-output regression. Wiley Interdiscip. Rev. Data Min. Knowl. Discov. **5**(5), 216–233 (2015)
4. Chen, L., Huang, J.Z.: Sparse reduced-rank regression for simultaneous dimension reduction and variable selection. J. Amer. Stat. Assoc. **107**(500), 1533–1545 (2012)
5. Cheng, D., Zhang, S., Liu, X., Sun, K., Zong, M.: Feature selection by combining subspace learning with sparse representation. Multimed. Syst. **23**(3), 1–7 (2017)
6. Deng, Z., Zhu, X., Cheng, D., Zong, M., Zhang, S.: Efficient k nn classification algorithm for big data. Neurocomputing **195**(C), 143–148 (2016)
7. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern classification. Wiley, New Jersey (2012)

8. Gao, L., Song, J., Liu, X., Shao, J., Liu, J., Shao, J.: Learning in high-dimensional multimedia data: the state of the art. Multimed. Syst. **23**(3), 303–313 (2017)
9. Gao, L., Wang, Y., Li, D., Shao, J., Song, J.: Real-time social media retrieval with spatial, temporal and social constraints. Neurocomputing **253**, 77–88 (2017)
10. Germain, F.G., Mysore, G.J.: Stopping criteria for non-negative matrix factorization based supervised and semi-supervised source separation. IEEE Signal Process. Lett. **21**(10), 1284–1288 (2014)
11. Gu, Q., Li, Z., Han, J.: Generalized fisher score for feature selection. arXiv:1202.3725 (2012)
12. Guo, Q., Wu, W., Massart, D., Boucon, C., De Jong, S.: Feature selection in principal component analysis of analytical data. Chemometr. Intell. Lab. Syst. **61**(1), 123–132 (2002)
13. He, X., Cai, D., Niyogi, P.: Laplacian score for feature selection. In: Advances in Neural Information Processing Systems, pp. 507–514 (2005)
14. Hu, R., Zhu, X., Cheng, D., He, W., Yan, Y., Song, J., Zhang, S.: Graph self-representation method for unsupervised feature selection. Neurocomputing **220**, 130–137 (2017)
15. Huang, H., Feng, H., Peng, C.: Complete local fisher discriminant analysis with laplacian score ranking for face recognition. Neurocomputing **89**, 64–77 (2012)
16. Izenman, A.J.: Linear discriminant analysis. In: Modern Multivariate Statistical Techniques, pp. 237–280. Springer, Berlin (2013)
17. Li, Z., Yang, Y., Liu, J., Zhou, X., Lu, H., et al.: Unsupervised feature selection using nonnegative spectral analysis. AAAI **2**, 1026–1032 (2012)
18. Liu, B., Fang, B., Liu, X., Chen, J., Huang, Z., He, X.: Large margin subspace learning for feature selection. Pattern Recogn. **46**(10), 2798–2806 (2013)
19. Ng, A.Y.: Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In: Proceedings of the Twenty-first International Conference on Machine Learning, p. 78. ACM (2004)
20. Nie, F., Huang, H., Cai, X., Ding, C.H.: Efficient and robust feature selection via joint $l_{2,1}$-norms minimization. In: Advances in Neural Information Processing Systems, pp. 1813–1821 (2010)
21. Peng, X., Yu, Z., Yi, Z., Tang, H.: Constructing the l2-graph for robust subspace learning and subspace clustering. IEEE Trans. Cybern. **47**(4), 1053–1066 (2017)
22. Pierre, C.: Semi-supervised feature selection via spectral analysis (2007)
23. Qin, Y., Zhang, S., Zhu, X., Zhang, J., Zhang, C.: Semi-parametric optimization for missing data imputation. Appl. Intell. **27**(1), 79–88 (2007)
24. Song, J., Gao, L., Nie, F., Shen, H.T., Yan, Y., Sebe, N.: Optimized graph learning using partial tags and multiple features for image and video annotation. IEEE Trans. Image Process. **25**(11), 4999–5011 (2016)
25. Song, J., Gao, L., Zou, F., Yan, Y., Sebe, N.: Deep and fast: Deep learning hashing with semi-supervised graph construction. Image Vis. Comput. **55**, 101–108 (2016)
26. Song, J., Shen, H.T., Wang, J., Huang, Z., Sebe, N., Wang, J.: A distance-computation-free search scheme for binary code databases. IEEE Trans. Multimed. **18**(3), 484–495 (2016)
27. Song, J., Yang, Y., Huang, Z., Shen, H.T., Luo, J.: Effective multiple feature hashing for large-scale near-duplicate video retrieval. IEEE Trans. Multimed. **15**(8), 1997–2008 (2013)
28. Trivedi, S., Pardos, Z.A., Heffernan, N.T.: Clustering students to generate an ensemble to improve standard test score predictions. In: International Conference on Artificial Intelligence in Education, pp. 377–384. Springer (2011)
29. Wang, L., Zhu, J., Zou, H.: Hybrid huberized support vector machines for microarray classification. In: Proceedings of the 24th International Conference on Machine Learning, pp. 983–990. ACM, New York (2007)
30. Wang, S., Nie, F., Chang, X., Yao, L., Li, X., Sheng, Q.Z.: Unsupervised feature analysis with class margin optimization. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 383–398. Springer, Berlin (2015)
31. Ye, J.: Least squares linear discriminant analysis. In: Proceedings of the 24th international conference on Machine learning, pp. 1087–1093. ACM, New York (2007)
32. Zeng, Z., Wang, X., Zhang, J., Wu, Q.: Semi-supervised feature selection based on local discriminative information. Neurocomputing **173**(P1), 102–109 (2016)
33. Zhang, S.: Shell-neighbor method and its application in missing data imputation. Appl. Intell. **35**(1), 123–133 (2011)
34. Zhang, S., Jin, Z., Zhu, X.: Missing data imputation by utilizing information within incomplete instances. J. Syst. Softw. **84**(3), 452–459 (2011)
35. Zhang, S., Li, X., Zong, M., Zhu, X., Cheng, D.: Learning k for knn classification. ACM Trans. Intell. Syst. Technol. **8**(3), 43 (2017)
36. Zhang, S., Li, X., Zong, M., Zhu, X., Wang, R.: Efficient knn classification with different numbers of nearest neighbors. IEEE Transactions on Neural Networks and Learning Systems (2017). https://doi.org/10.1109/TNNLS.2017.2673241

37. Zhu, P., Zuo, W., Zhang, L., Hu, Q., Shiu, S.C.: Unsupervised feature selection by regularized self-representation. Pattern Recogn. **48**(2), 438–446 (2015)
38. Zhu, X., Huang, Z., Shen, H.T., Zhao, X.: Linear cross-modal hashing for efficient multimedia search. In: ACM MM, pp. 143–152 (2013)
39. Zhu, X., Huang, Z., Yang, Y., Shen, H.T., Xu, C., Luo, J.: Self-taught dimensionality reduction on the high-dimensional small-sized data. Pattern Recogn. **46**(1), 215–229 (2013)
40. Zhu, X., Li, X., Zhang, S.: Block-row sparse multiview multilabel learning for image classification. IEEE Trans. Cybern. **46**(2), 450–461 (2016)
41. Zhu, X., Li, X., Zhang, S., Ju, C., Wu, X.: Robust joint graph sparse coding for unsupervised spectral feature selection. IEEE Trans. Neural Netw. Learn. Syst. **28**(6), 1263–1275 (2017)
42. Zhu, X., Li, X., Zhang, S., Xu, Z., Yu, L., Wang, C.: Graph pca hashing for similarity search. IEEE Transactions on Multimedia (2017). https://doi.org/10.1109/TMM.2017.2703636
43. Zhu, X., Suk, H., Wang, L., Lee, S., Shen, D.: A novel relational regularization feature selection method for joint regression and classification in AD diagnosis. Med. Image Anal. **38**, 205–214 (2017)
44. Zhu, X., Suk, H.-I., Huang, H., Shen, D.: Low-rank graph-regularized structured sparse regression for identifying genetic biomarkers. IEEE Transactions on Big Data (2017). https://doi.org/10.1109/TBDATA.2017.2735991
45. Zhu, X., Suk, H.-I., Lee, S.-W., Shen, D.: Subspace regularized sparse multitask learning for multiclass neurodegenerative disease identification. IEEE Trans. Biomed. Eng. **63**(3), 607–618 (2016)
46. Zhu, X., Suk, H.-I., Shen, D.: A novel matrix-similarity based loss function for joint regression and classification in ad diagnosis. NeuroImage **100**, 91–105 (2014)
47. Zhu, X., Zhang, L., Huang, Z.: A sparse embedding and least variance encoding approach to hashing. IEEE Trans. Image Process. **23**(9), 3737–3750 (2014)
48. Zhu, Y., Lucey, S.: Convolutional sparse coding for trajectory reconstruction. IEEE Trans. Pattern Anal. Mach. Intell. **37**(3), 529–540 (2015)
49. Zhu, Y., Zhu, X., Kim, M., Shen, D., Wu, G.: Early diagnosis of alzheimers disease by joint feature selection and classification on temporally structured support vector machine. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 264–272 (2016)