Journal of Biomedical Informatics 84 (2018) 164–170

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Classification of ADHD with bi-objective optimization

Lizhen Shao*, Yadong Xu, Dongmei Fu

School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China

ARTICLE INFO

Keywords:
ADHD
Bi-objective SVM
FMRI

ABSTRACT

Attention Deficit Hyperactive Disorder (ADHD) is one of the most common diseases in school aged children. In this paper, we consider using fMRI data with classification techniques to aid the diagnosis of ADHD and propose a bi-objective ADHD classification scheme based on L_1-norm support vector machine (SVM). In our classification model, two objectives, namely, the margin of separation and the empirical error are considered at the same time. Then the normal boundary intersection (NBI) method of Das and Dennis is used to solve the bi-objective optimization problem. A representative nondominated set which reflects the entire trade-off information between the two objectives is obtained. Each representative nondominated point in the set corresponds to an efficient classifier. Finally a decision maker can choose a final efficient classifier from the set according to the performance of each classifier. Our scheme avoids the trial and error process for regularization hyper-parameter selection. Experimental results show that our bi-objective optimization classification scheme for ADHD diagnosis performs considerably better than some traditional classification methods.

1. Introduction

Attention Deficit Hyperactivity Disorder (ADHD) is a very common mental disorder in childhood. ADHD symptoms include inattention, hyperactivity, and impulsively. It affects approximately 5–7% of all school-age children and more than one half of the ADHD children continue to manifest clinically significant symptoms after reaching adulthood [1]. As the pathogenesis is not clear, the main diagnostic method is based on the subjective experience of doctors, which results in many children not being able to receive good treatment in the early stage of ADHD. In addition to the traditional clinical diagnosis, there is a pressing need to find a set of more distinctive and objective features to characterize ADHD that can be used to facilitate ADHD diagnosis.

As a promising neuroimaging tool, functional MRI (fMRI) has been widely used to examine the brain of ADHD patients. Abnormal brain activations were found in task-related experiments on the dorsal anterior cingulate cortex (dACC), the ventrolateral prefrontal cortex (VLPFC) and the putamen [2]. Using resting-state fMRI, abnormalities were found in prefrontal cortex, inferior frontal cortex, sensorimotor cortex, anterior cingulated cortex, putamen, temporal cortex and cerebellum. In addition, Castellanos et al. (2008) [3] found ADHD-related decreases of functional connectivity between anterior cingulate and precuneus/posterior cingulate cortex regions, as well as between precuneus and other default-mode network components, including ventromedial prefrontal cortex and portions of posterior cingulate cortex. This suggests that functional connectivity information of functional MRI data can be used as a classification feature for ADHD diagnosis.

With the development of machine learning techniques, many efforts have also been made to predict ADHD disease of patients. Mueller et al. [4] have introduced a machine learning system that uses support vector machine (SVM) to differentiate ADHD adults from control groups on the base of the event related potentials that are generated from the EEG measurements. In [5], Peng et al. utilized structural MRI data and ELM for ADHD classification. Dai et al. [6] proposed using multimodal magnetic resonance imaging to classify ADHD children. Mourão-Miranda et al. [7] applied SVM algorithm to perform multivariate classification of brain states from whole fMRI volumes. They demonstrated that SVM outperforms Fisher Linear Discriminant (FLD) classifier in classification performance as well as in robustness of the spatial maps obtained by a comparative analysis.

Traditional machine learning classification methods mentioned above for ADHD are based on single optimization technique. One or more hyper-parameters need to be selected before training a classifier. This often leads to time-consuming training sessions and classification inefficiency with changes in sample size. For example, when using a SVM to classify mild cognitive impairment subtypes [8], Haller [9] explored the gamma parameter iteratively from 0.01 to 0.09. The main reason is that the optimization problem in training procedure normally involves more than one objective (for example, two typical objectives in SVMs are maximizing the margin of separation and minimizing

* Corresponding author.
E-mail addresses: lshao@ustb.edu.cn (L. Shao), 1019340972@qq.com (Y. Xu), fdm2003@163.com (D. Fu).

https://doi.org/10.1016/j.jbi.2018.07.011
Received 29 September 2017; Received in revised form 28 May 2018; Accepted 11 July 2018

empirical error). However, traditional machine learning techniques generally use a trade-off parameter to sum up all the objectives into one and thus turn a multi-objective optimization problem into a single objective one. Hence in order to obtain an efficient classifier normally trial and error process is needed for choosing suitable parameters.

In this paper, we propose a bi-objective optimization scheme for the training session of a classifier. The paper is organized as follows. In Section 2, fMRI data is preprocessed to get the functional connection matrix, then we use SPM toolbox (Statistical Parametric Mapping) to analyze the statistical difference between ADHD patients and healthy controls. Next, we utilize principal component analysis method to reduce the dimension of the feature vector after comparing different dimension reduction methods. Furthermore, we present our bi-objective optimization classification scheme which is based on L_1-norm SVM for ADHD classification. In this scheme, normal boundary intersection (NBI) method of Das and Dennis is adopted to solve the bi-objective optimization problem. In Section 3, experimental results and discussions are presented. Finally, we draw some conclusions in Section 4.

2. Materials and methods

2.1. Subjects

Functional magnetic resonance imaging (fMRI) measures brain activity by detecting changes associated with blood flow. This technique relies on the fact that cerebral blood flow and neuronal activation are coupled. When an area of the brain is in use, blood flow to that region also increases. The fMRI data can be represented as a series of three dimensional images. In this study we use the resting state fMRI data from the Neuron Bureau ADHD 200 competition (http://fcon_1000.projects.nitrc.org/). The ADHD-200 sample is dedicated to accelerating the scientific community's understanding of the neural basis of ADHD through the implementation of open data-sharing and discovery-based science. We use KKI, PU-1, PU-2, PU-3 datasets from Kennedy Krieger Institute (KKI) and Peking University (PU), respectively. Furthermore, we generate a dataset called PU-joint composed of PU-1, PU-2 and PU-3 and a big dataset KKI-PU-joint composed of KKI and PU-joint. In our experiment, the experimental validations of our proposed method are performed on four datasets KKI, PU-1, PU-joint and KKI-PU-joint. An overview of the data is shown in Table 1. It needs to be noted that KKI-PU-joint is only used for checking the performance of our method in handling large-scale classification problems since it is composed of data from different experiments.

2.2. Preprocessing

Data preprocessing is based on DPARSF toolbox (http://www.restfmri.net/forum/DPARSF/). Because of the instability of the initial signal and the subjects' adaptation to the situation, the first 10 images were discarded. Then we spatially normalized the realigned images to the standard echo-planar imaging template and resampled them to $3 \times 3 \times 3\,mm^3$. In the next, the functional images were spatially smoothed with a Gaussian kernel of $4 \times 4 \times 4\,mm^3$ FWHW to decrease spatial noise. Subsequently, the fMRI data was time-filtered from 0.01 to 0.08 Hz to eliminate the effects of low frequency drift and high frequency noise. After the preprocessing process above, according to the AAL template reported by Tzourio-Mazoyer et al. [10], the brain image is divided into 116 brain regions, where 90 regions in the cerebra and 26 regions in the cerebella. We obtained the mean time series of each of the 116 regions by averaging the fMRI time series over all voxels in the region using FC (functional connectivity) method. Pearson correlation coefficients are computed between each pair of them to obtain the resulting function connection matrix. The function connection matrix is a symmetric matrix. The lower triangular data of the matrix is taken as the initial features of a single sample. The dimension of the initial feature vector is 6670 ($116 \times 116/2-116/2$). Fig. 1 shows the functional connection matrix acquisition flowchart. The sizes of the initial feature matrices for the four data sets are 83×6670, 85×6670, 194×6670 and 277×6670, respectively.

Table 1
Description of the datasets.

Dataset	Total number of subjects	Number of ADHD subjects	Number of control subjects
KKI	83	22	61
PU-1	85	24	61
PU-joint	194	76	118
KKI-PU-joint	277	98	179

Moreover, statistical analysis is used to show the difference of the brain between ADHD patients and healthy controls. We collected ten patients and ten controls from the four datasets to perform a random-effect two-tailed two-sample t-test SPM5 (http://www.fil.ion.ucl.ac.uk/spm/) in Matlab 7.8. Using FC analysis of SPM5, the statistical difference results are obtained, see Fig. 2. The red highlighted areas which correspond to posterior cingulate cortex (PCC), dorsal posterior cingulate cortex (dPCC), dorsal anterior cingulate cortex (dACC) and so on are considered to have statistically significant difference between the two groups. This result further verifies that the viability of using feature matrix to classify ADHD data.

2.3. Data dimensionality reduction

For the four datasets, the number of original brain features for classification is 6670, while the sample sizes are 83 (KKI), 85 (PU-1), 194 (PU-joint) and 277 (KKI-PU-joint) respectively. The dimensionality of original brain features is much higher than the number of samples. Thus, dimensionality reduction is required to improve the performance of the classifier. For these small sample high dimension data, we have compared three traditional methods, namely, principal component analysis (PCA) [11], Isomap [12] and LLE [13] methods to reduce the dimension of the original features. After comparison, we chose PCA method as the dimensionality reduction method. Finally the sizes of the feature matrix for the four data sets are reduced to 83×49, 85×60, 194×132 and 277×180, respectively.

2.4. Bi-objective classification scheme

In this section, we are going to propose a multi-objective classification scheme. The main difference between our scheme and traditional classification methods based on regularization hyper-parameter selection can be seen in Figs. 3 and 4.

From the figures we can see that our scheme can provide a representative set of classifiers, which can make the decision maker (a person who is responsible for making decisions) select a most desirable classifier from it. Thus it avoids the trial and error process for regularization hyper-parameter selection.

2.4.1. Bi-objective SVMs

Support vector machines are supervised learning models with associated learning algorithms that analyze data used for classification in machine learning. In this section we consider formulating a bi-objective classification model based on SVMs.

Given a set of training examples of two classes $S = \{(x^1, y^1), (x^2, y^2), ..., (x^m, y^m)\}$, where m is the number of training samples, $x^i \in X \subset R^n$ (n is the dimension of samples) is the i-th observation and it is sampled from an input space X, and $y^i \in \{-1, +1\}$ is the corresponding label for x^i. The aim of a SVM is to find a hyperplane that separates S into two half-spaces such that samples of the same class are in the same half-space. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class since in general the larger the margin the lower the

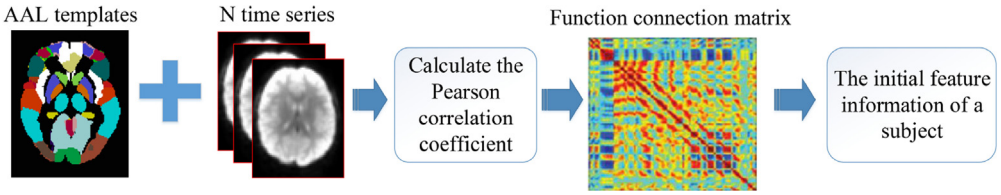

Fig. 1. Functional connection matrix acquisition flowchart.

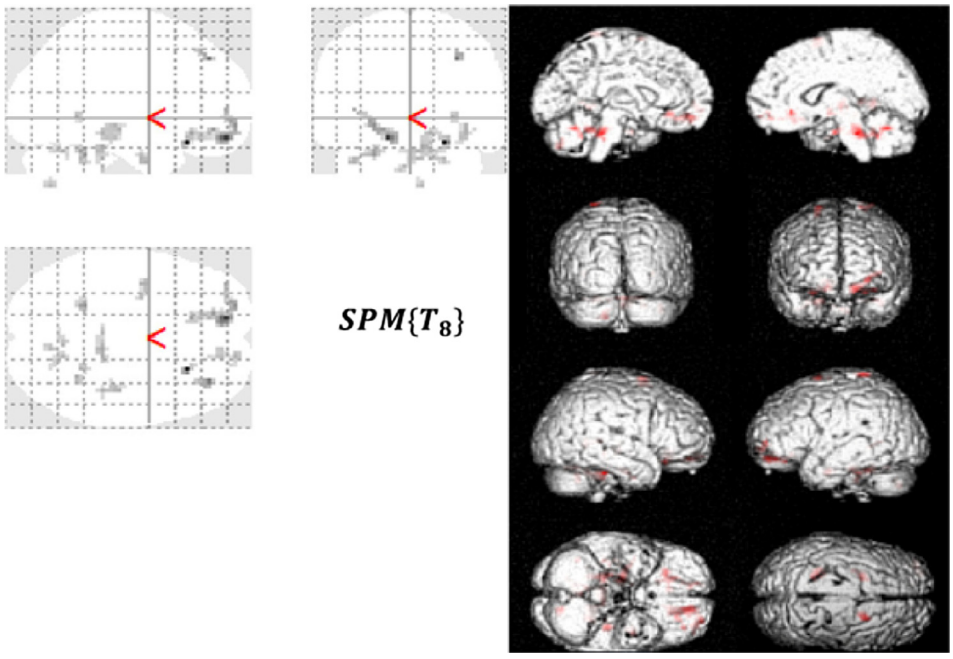

Fig. 2. Statistical differences between healthy controls and ADHD patients using FC method (left: active area detection results; right: 3D brain surface activation diagram).

generalization error of the classifier. However, practically it often happens that the sets to discriminate are not linearly separable. Thus in a soft-margin SVM, in addition to maximizing the margin of separation, minimizing the measure of empirical error is also considered as an objective. The two objectives are usually summed up by a weighting factor C (which is used to balance the importance between the two objectives on training samples) to form a weighted sum objective function.

For a single sample, the empirical error is typically measured by its distance to the margin. As for the margin of separation, in the case of L_1-norm is used, maximizing the margin of separation implies minimizing L_1-norm of the normal vector of the separating hyperplane. Therefore, a typical soft-margin SVM formulation based on L_1-norm can be described as follows.

$$\begin{aligned} \text{L}_1\text{CSVM} \quad \min \quad & \|\omega\|_1 + C\sum_{i=1}^{m}\xi_i \\ \text{s.t.} \quad & y_i(\omega\cdot x_i + b) \geqslant 1-\xi_i, \quad i = 1, 2, \ldots, m \\ & \xi_i \geqslant 0 \end{aligned}$$

where ξ_i denotes the empirical error associated with instance i and C is a weighting factor. If C is changed, then the importance placed on each objective changes as well. This may result in a different optimal solution and hence a different classifier. In practice, the chosen of C value is typically by experimentation. By changing the value of C, a set of efficient classifiers can be created and hence the most preferred one can be observed by a decision maker. However, the main disadvantage of experimenting with C is that it is hard to predict the changes of the two objectives with the changing of C. For example, suppose that one thinks that the two objectives are equally important, then a default C value of 1 is used. If the classifier obtained has a large empirical error, then one would like to try a larger value of C in expecting a lower empirical error. However, only after experimentation the suitable value of C can be identified since it is highly data dependent and cannot be known in advance whether a C value of 5, 20, or 200 would lead to an acceptable classifier. In an effort, Chang and Lin (2011) [14] suggest varying C exponentially to overcome this problem.

In contrast to the traditional SVMs, a bi-objective version of L_1-norm SVM formulation is as follows:

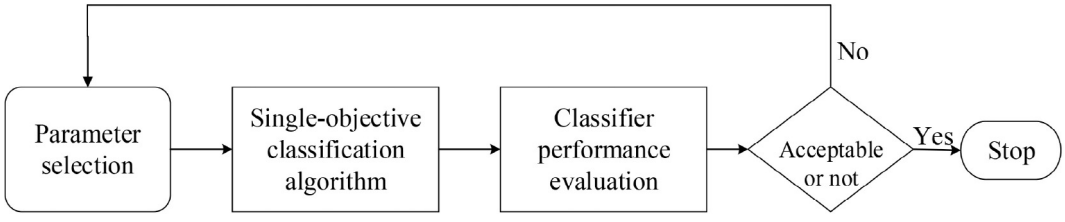

Fig. 3. Traditional classification methods based on regularization hyper-parameter selection.

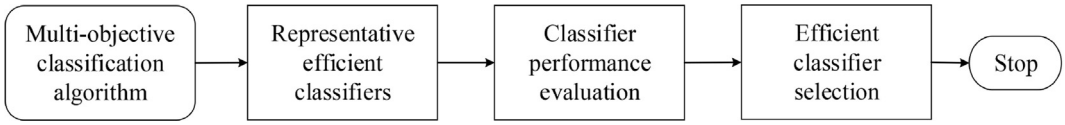

Fig. 4. Multi-objective classification scheme.

$$(\mathrm{L_1BioSVM}) \quad \min \ \|\omega\|_1, \quad \min \sum_{i=1}^{m} \xi_i \quad \text{s.t. } y_i(\omega \cdot x_i + b) \geqslant 1-\xi_i, \quad i = 1, 2, \ldots, m \quad \xi_i \geqslant 0$$

Here, we maximize the separation margin and minimize the empirical error at the same time.

For a bi-objective optimization problem, the two objectives are usually conflicting so that a feasible solution optimizing both objectives simultaneously does not exist. Therefore, the purpose of bi-objective optimization is to obtain nondominated points. A nondominated point in objective space is the image of an efficient solution in variable space. An efficient solution is defined as a feasible solution for which an improvement in one objective will always lead to a deterioration in the other objective.

According to the definition of efficient solution, Aytug and Sayin [15] defined efficient SVM classifier for problem (L_1BioSVM).

For a particular training data set, a classifier (ω^0, b^0) is said to be dominated by another classifier (ω, b), if $\|\omega\|_1 < \|\omega^0\|_1$ and $\sum_i \xi_i \leqq \sum_i \xi_i^0$ or $\|\omega\|_1 \leqq \|\omega^0\|_1$ and $\sum_i \xi_i < \sum_i \xi_i^0$, where ξ_i^0 and ξ_i denote the empirical errors under each classifier respectively. A classifier is called efficient if there does not exists other classifier dominating it.

All the efficient classifiers form the efficient frontier. For a decision maker, any efficient classifier in the frontier may possibly be of interest because it reflects a particular trade-off information between the margin of separation and empirical error. Obviously, any classifier which does not on the efficient frontier do not need be considered because an alternative one that dominates it is known to be sure exist.

To make problem (L_1BioSVM) solvable, we use the difference of two positive variables ω^+ and ω^- to represent ω, i.e., $\omega = (\omega^+ - \omega^-)$. Then the first objective $\|\omega\|_1$ can be written as $\min e^{\mathrm{T}}(\omega^+ + \omega^-)$. Therefore problem ($L_1$BioSVM) is transformed into problem (TL_1BioSVM) as follows.

$$(\mathrm{TL_1bioSVM}) \quad \min e^{\mathrm{T}}(\omega^+ + \omega^-), \ \sum_{i=1}^{m} \xi_i \quad \text{s.t. } y_i((\omega^+ - \omega^-) \cdot x_i + b) \geqslant 1-\xi_i, \quad i = 1, 2, \ldots, l \quad \xi_i \geqslant 0 \quad \omega^+, \omega^- \geqslant 0$$

where e is a column of ones and e^{T} is the transpose of e.

Accordingly, for problem (TL_1bioSVM), a classifier corresponding to $(\overline{\omega^+}, \overline{\omega^-}, \overline{b}, \overline{\xi})$ is called efficient if there does not exist any other feasible solution (w^+, w^-, b, ξ) such that $e^{\mathrm{T}}(w^+ + w^-) \leqq e^{\mathrm{T}}(\overline{\omega^+} + \overline{\omega^-})$ and $\sum_i \xi_i < \sum_i \overline{\xi_i}$ or $e^{\mathrm{T}}(w^+ + w^-) < e^{\mathrm{T}}(\overline{\omega^+} + \overline{\omega^-})$ and $\sum_i \xi_i \leqq \sum_i \overline{\xi_i}$.

2.4.2. Normal boundary intersection method for solving bi-objective L_1-norm SVM

Researchers have developed a lot of methods to solve multi-objective programming (MOP) problems. These algorithms basically fall into two categories, one is for approximating the nondominated set, the other is for finding a discrete representation of the nondominated set. While for classification problems, since a decision maker normally prefers a representative set of efficient classifiers for decision making, we focus on algorithms for finding a discrete representation of the nondominated set. For discrete representation, Sayin (2000) [16] defined three quality criteria, namely cardinality, uniformity and coverage. According to three quality criteria, a good representation should have small cardinality, small coverage error and a high uniformity level. Therefore we use the normal boundary intersection method proposed by [17] to solve the L_1-norm bi-objective SVM and thus enumerates a representative set of efficient classifiers.

NBI method is proposed for solving the generalized formulation of MOP problem

$$(\mathrm{MOP}) \ \min \ f(x) = (f_1(x), f_2(x), \ldots, f_p(x))^T \quad \text{s.t.} \ \ x \in R^n: g(x) = (g_1(x), g_2(x), \ldots, g_m(x))^T \leqslant 0.$$

$X = \{x \in R^n: g(x) = (g_1(x), g_2(x), \ldots, g_m(x))^T \leqslant 0\}$ is the feasible set in decision space R^n, we assume it is nonempty. The feasible set Y in objective space R^p is defined by

$$Y = \{f(x): x \in X\}.$$

For problem (MOP), a feasible solution $\hat{x} \in X$ is an efficient solution of problem (MOP) if there exists no $x \in X$ such that $f(x) \leqslant f(\hat{x})$. The set of all efficient solutions of problem (MOP) will be denoted by X_E and called the efficient set in decision space. Correspondingly, $\hat{y} = f(\hat{x})$ is called a nondominated point and $Y_N = \{f(x): x \in X_E\}$ is the nondominated set in objective space.

NBI method is dedicated to find a finite representative subset R of the nondominated set Y_N. Fig. 5 shows how the NBI method works for a bi-objective optimization problem example. This method calculates a reference plane, and places uniformly distributed reference points on the reference plane, then projects the reference points to the boundary of Y along the normal direction. Finally, a finite subset R of Y_N (and their corresponding efficient solutions) that represents Y_N can be obtained. The NBI algorithm for solving bi-objective optimization problem is summarized in Algorithm 1. For a more detailed description, the reader is referred to [17].

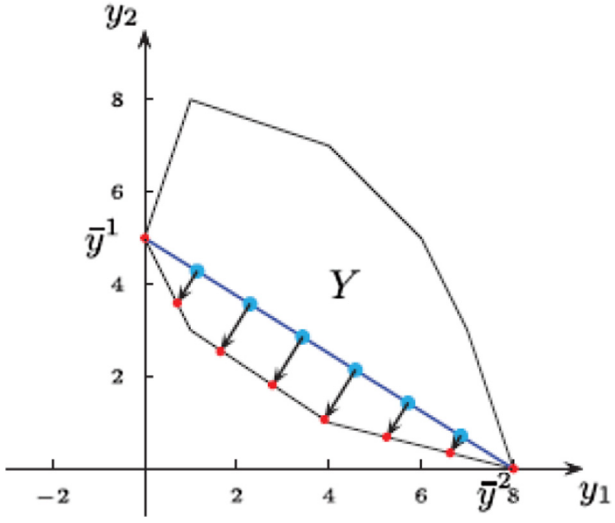

Fig. 5. The nondominated points obtained in the NBI method.

Algorithm 1. The normal boundary intersection method for bi-objective problem (MOP).

INPUT: Training set and the number of reference points r

Step 1. y_k^I: =min$\{f_k(x)$: $x \in X\}$, $k = 1, 2$

Solve min$\{y_2$: $y \in Y$, $y_1 = y_1^I\}$ and min$\{y_1$: $y \in Y$, $y_2 = y_2^I\}$, let $\bar{y}^1$ and $\bar{y}^2$ be the optimal solutions, respectively. Line segment $(\bar{y}^1, \bar{y}^2)$ is the reference plane.

Step 2. Compute r equally spaced reference points q_i on the reference plane.

Step 3. For $i = 1$ to r, solve

$$\begin{aligned} \max \quad & t \\ \text{s.t.} \quad & q_i + t\hat{n} = f(x) \\ & t \geqslant 0 \\ & x \in X \end{aligned}$$

and store the nondominated points $q_i + t\hat{n}$ to R.

OUTPUT: The representative nondominated subset R

For bi-objective optimization problem, Algorithm 1 can obtain a representative non-dominated set. With a suitable number of reference points, NBI method can guarantee uniformity and coverage. In the case of bi-objective optimization problem is (TL$_1$bioSVM), the nondominated points are corresponding to efficient classifiers, thus a set of representative efficient classifiers can be obtained.

3. Results

We implement NBI algorithm in Matlab for solving bi-objective classification problem (TL$_1$bioSVM). Each of the four datasets is divided into three sets, i.e., training set, validation set and testing set by stratified random sampling. More precisely, we randomly sample 60% of both positive data and negative data as a training dataset, 20% for validation dataset and 20% for testing dataset, respectively. We then run NBI method to solve problem (TL$_1$bioSVM) using training data to obtain a representative nondominated set (each nondominated point in the set corresponds an efficient classifier) to explore the trade-off between the two objectives, namely, the margin of separation and empirical error. Next we test the representative efficient classifiers on the validation and testing data.

The results are shown in Figs. 6–9. In each figure, the representative nondominated points (efficient classifiers) generated by NBI algorithm are the square points on the piecewise linear curve. Each square point also represents a pair of parameters ($\|\omega\|_1$, $\sum \xi_i$) for each classifier, and other points with the same x-axis value represent the performance on the dataset corresponding to the classifier. The nondominated point curve depicts reasonable trade-offs between the margin of separation and the empirical error. We can see that the $\sum \xi_i$ curve approximates the training error curve and it is used as a criterion for error counts. Comparing the validation and testing errors with the training errors, we can see that the validation and testing error curves are not only not smooth but also not consistent with the training error curve. Usually a large training dataset is necessary to make sure classification accuracy since most classification algorithms are probabilistic so that an ideal classification testing error cannot be always get. Moreover, with the increase of the training data set, the three error curves are supposed to converge. We also notice that the classification errors for KKI-PU-joint are quite high. This may be due to that the data are from different experiments.

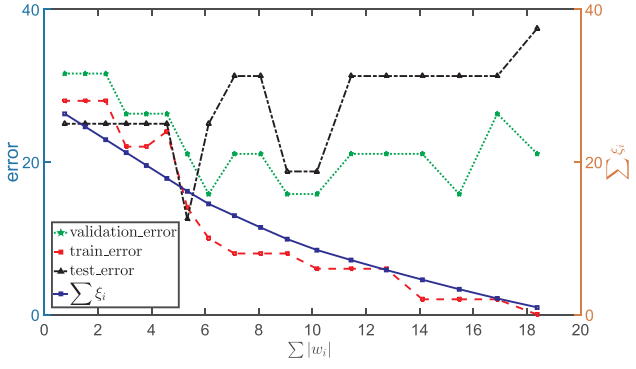

Fig. 6. Norm versus empirical error: training, validation and testing sets for KKI dataset.

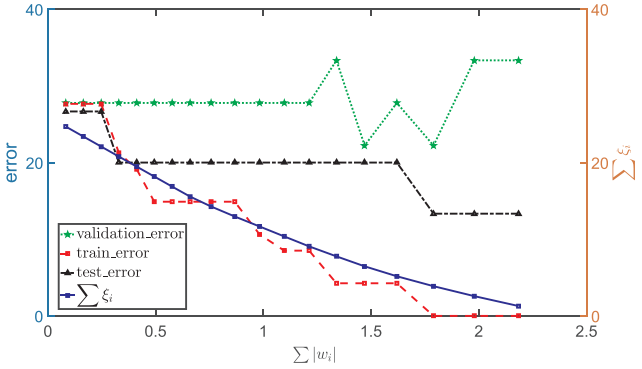

Fig. 7. Norm versus empirical error: training, validation and testing sets for PU-1 dataset.

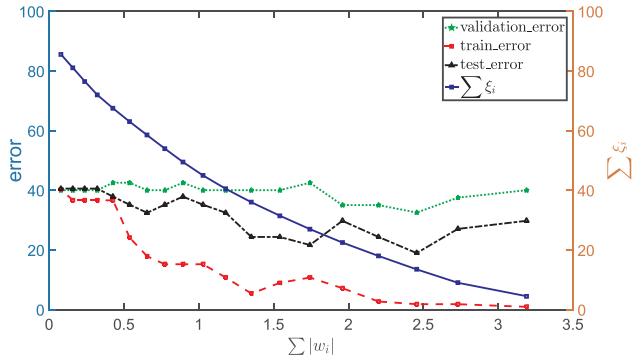

Fig. 8. Norm versus empirical error: training, validation and testing sets for PU-joint dataset.

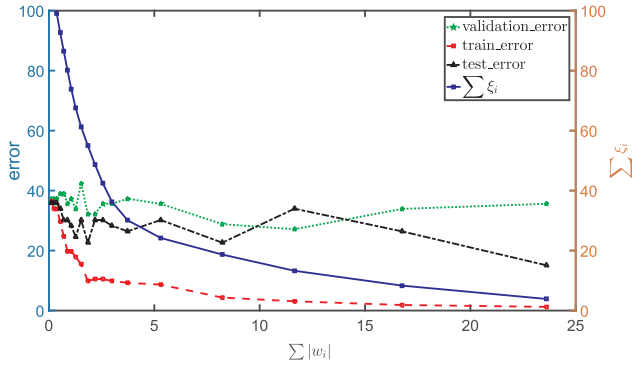

Fig. 9. Norm versus empirical error: training, validation and testing sets for KKI-PU-joint dataset.

3.1. Discussion

Considering the classification error of each specific classifier, we

Table 2
The performance on training/testing dataset of KKI.

Classifier	(9.46, 9.29)	(8.50, 10.72)	(7.57, 12.18)	(6.65, 13.64)
Sensitivity	1.0000/0.7500	0.9722/0.6667	0.9722/0.6667	0.9722/0.6667
Specificity	0.7857/1.0000	0.7857/1.0000	0.7857/0.7500	0.7857/0.7500
Accuracy	0.9400/0.8125	0.9200/0.7500	0.9200/0.6875	0.9200/0.6875

Table 3
The performance on training/testing dataset of PU-1.

Classifier	(1.21, 9.06)	(1.46, 6.46)	(1.61, 5.16)	(1.78, 3.87)
Sensitivity	1.0000/1.0000	1.0000/1.0000	1.0000/1.0000	1.0000/1.0000
Specificity	1.0000/0.4731	0.8227/0.2306	0.8227/0.2306	0.6419/0.2306
Accuracy	1.0000/0.8000	0.9559/0.8000	0.9559/0.8000	0.9175/0.8000

Table 4
The performance on training/testing dataset of PU-joint.

Classifier	(2.73, 8.96)	(2.45, 13.45)	(2.20, 17.95)	(1.95, 22.45)
Sensitivity	0.9850/0.7727	0.9850/0.9090	0.9850/0.9090	0.9701/0.9090
Specificity	0.9777/0.6667	0.9777/0.6667	0.9555/0.5333	0.8667/0.4000
Accuracy	0.9821/0.7297	0.9821/0.8108	0.9732/0.7567	0.9285/0.7027

Table 5
The performance on training/testing dataset of KKI-PU-joint.

Classifier	(11.40, 14.20)	(7.97, 19.93)	(4.88, 25.76)	(3.36, 32.03)
Sensitivity	0.9902/0.8529	1.0000/0.8235	1.0000/0.9117	0.9902/0.9117
Specificity	0.8983/0.7368	0.8135/0.5789	0.7627/0.5263	0.7627/0.5789
Accuracy	0.9567/0.8113	0.9320/0.7358	0.9135/0.7733	0.9074/0.7924

select four representative classifiers for each problem. They are corresponding to nondominated points (9.46, 9.29), (8.50, 10.72), (7.57, 12.18) and (6.65, 13.64) for KKI data; (1.21, 9.06), (1.46, 6.46), (1.61, 5.16) and (1.78, 3.87) for PU-1 data; (2.73, 8.96), (2.45, 13.45), (2.20, 17.95) and (1.95, 22.45) for PU-joint data; (11.40, 14.20), (7.97, 19.93), (4.88, 25.76) and (3.36, 32.03) for KKI-PU-joint data, which are shown in Figs. 6–9, respectively. For each classifier, we calculate sensitivity, specificity and accuracy for training and testing datasets. They are summarized in Tables 2–5.

According to the classification errors and the above three indicators, a decision maker can weigh the relationship between the two objectives and obtain a range of efficient classifiers with different properties. It needs to be noted that the specificity is generally low for the three data sets, this may be due to the imbalanced property of the data which makes the prediction of negative samples less accurate.

3.2. Comparison of different classification methods

To further show the performance our bi-objective classification scheme, we compare our method with L_1CSVM, L_2CSVM (norm-2 SVM) [18], k-nearest neighbors (KNN) [19,20], AdaBoost [21] Random Forest (RF) [22], Multi-layer Perceptron (MLP) [23] and extreme learning machine (ELM) [5] methods on KKI, PU-1 and PU-joint datasets.

Table 7
Average MCC value for different methods.

Dataset	L_1BioSVM	L_1CSVM	L_2CSVM	KNN	AdaBoost	RF	MLP	ELM
KKI	**0.66**	0.19	0.36	0.15	**0.66**	0.36	0.29	0.65
PU-1	**0.65**	0.50	0.56	0.21	**0.65**	**0.65**	0.00	0.45
PU-joint	**0.60**	0.36	0.55	0.37	0.42	0.23	0.32	0.52

We use the same stratified random sampling procedure as the one for (TL_1bioSVM) and tune the hyper-parameters on the validation datasets for above mentioned methods. For all the methods, we train the parameters of the classifiers based on the training datasets and test the performance of the classifiers on the testing datasets. We run the experiment ten times, finally we come out an average performance for each method. For our bi-objective method, each time for a different training dataset we choose a most desirable efficient classifier from all classifiers according to the performance on the validation set to avoid overfitting.

For all the other methods, the hyper-parameters used are listed as below. In consistence with our (TL_1bioSVM), no kernel are used for both L_1CSVM and L_2CSVM; the C value for both L_1CSVM and L_2CSVM is 0.8; the number of neighbors for KNN is 5; the base estimator for AdaBoost is decision tree, the number of estimators is 30 and learning rate is 0.2; for RF, the number of estimators is 10; for MLP, the number of hidden layers is 3 and the number of neurons for each layer is 20, 30 and 20; for ELM, the number of neurons in the hidden layer is 180 and the activation function is sigmoid. Parameters of all classifiers mentioned above are tuned according to the performance of the validation dataset and are evaluated on the performance of the testing dataset.

The average accuracy and Matthews' correlation coefficient (MCC) [24] of testing datasets are shown in Tables 6 and 7, respectively. From the tables, we can see that our method behaves well in general. The MCC parameters obtained by our method are the highest among all the methods for all the three datasets. We also notice that in Table 6 the accuracy of classification is not too high. This may be due to not enough dataset samples and data imbalanced..

4. Conclusion and future work

In this paper we have addressed the problem of classification of ADHD patients and healthy controls based upon fMRI data. We use the function connectivity information as the feature information for classification. Instead of the traditional weighted sum formulation, we adopt a bi-objective SVM formulation based on L_1-norm in which the two objectives are to minimize empirical error and maximize the margin of separation. The normal boundary intersection algorithm is used to generate a representative nondominated set. Each nondominated point in the set corresponds to an efficient classifier for decision makers to choose.

Experimental results show that our method can be used for obtaining a good representative set of classifiers in comparison to the traditional SVM formulation with a trade-off parameter C. With bi-objective optimization scheme, an efficient classifier can be selected from the representative set of classifier frontier by decision makers. Comparative analysis shows that our method gives a better performance than L_1CSVM, L_2CSVM, KNN, AdaBoost, RF, MLP and ELM

Table 6
Average classification accuracy for different methods.

Dataset	L_1BioSVM	L_1CSVM	L_2CSVM	KNN	AdaBoost	RF	MLP	ELM
KKI	81.25%	69.70%	68.75%	62.50%	**87.50%**	68.75%	62.50%	81.25%
PU-1	**86.67%**	74.19%	73.33%	73.33%	**86.67%**	**86.67%**	73.33%	80.00%
PU-joint	**81.08%**	66.67%	78.38%	70.27%	72.97%	64.86%	64.86%	75.60%

methods. Further work can be done on the processing of imbalanced data in order to improve the accuracy of classification.

Conflict of interest statement

We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

Acknowledgments

This work has been partially supported by Beijing Municipal Natural Science Foundation [4152034].

References

[1] A. Mcguire, Diagnosing the diagnostic and statistical manual of mental disorders, Disab. Soc. 25 (2) (2014) 1–4.
[2] A.D.S. Siqueira, C.E.B. Junior, W.E. Comfort, L.A. Rohde, J.R. Sato, Abnormal functional resting-state networks in ADHD: graph theory and pattern recognition analysis of fMRI data, Biomed Res. Int. 2014 (4) (2014) 10.
[3] F.X. Castellanos, D.S. Margulies, C. Kelly, L.Q. Uddin, M. Ghaffari, A. Kirsch, D. Shaw, Z. Shehzad, A. Di Martino, B. Biswal, et al., Cingulate-precuneus interactions: a new locus of dysfunction in adult attention-deficit/hyperactivity disorder, Biol. Psychiatry 63 (3) (2008) 332–337.
[4] A. Mueller, G. Candrian, J.D. Kropotov, V.A. Ponomarev, G.M. Baschera, Classification of ADHD patients on the basis of independent ERP components using a machine learning system, Nonlinear Biomed. Phys. 4 (1) (2010) 1–12.
[5] X. Peng, P. Lin, T. Zhang, J. Wang, Extreme learning machine-based classification of ADHD using brain structural MRI data, Plos One 8 (11) (2013) 456–461.
[6] D. Dai, J. Wang, J. Hua, H. He, Classification of ADHD children through multimodal magnetic resonance imaging, Front. Syst. Neurosci. 6 (2012) 63.
[7] J. Mourão-Miranda, A.L. Bokde, C. Born, H. Hampel, M. Stetter, Classifying brain states and determining the discriminating activation patterns: support vector machine on functional MRI data, Neuroimage 28 (4) (2005) 980–995.
[8] J. Cao, Z. Li, B. Wang, F. Li, J. Yang, A fast gene selection method for multi-cancer classification using multiple support vector data description, J. Biomed. Inform. 53 (2015) 381–389.
[9] S. Haller, S. Badoud, D. Nguyen, V. Garibotto, K.O. Lovblad, P.R. Burkhard, Individual detection of patients with Parkinson disease using support vector machine analysis of diffusion tensor imaging data: initial results. Am. J. Neuroradiol. 33 (11) (2012) 2123–2128.
[10] N. Tzourio-Mazoyer, B. Landeau, D. Papathanassiou, F. Crivello, O. Etard, N. Delcroix, B. Mazoyer, M. Joliot, Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain, Neuroimage 15 (1) (2002) 273–289.
[11] L.I. Smith, A tutorial on principal components analysis, Inform. Fusion 51 (3) (2002) 52.
[12] J.B. Tenenbaum, V. De Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.
[13] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.
[14] C.C. Chang, C.J. Lin, Libsvm: a library for support vector machines, Acm Trans. Intell. Syst. Technol. 2 (3) (2012) 1–27.
[15] H. Aytug, S. Sayin, Exploring the trade-off between generalization and empirical errors in a one-norm SVM, Eur. J. Oper. Res. 218 (3) (2012) 667–675.
[16] S. Sayın, Measuring the quality of discrete representations of efficient sets in multiple objective mathematical programming, Math. Program. 87 (3) (2000) 543–560.
[17] I. Das, J.E. Dennis, Normal-boundary intersection: a new method for generating the pareto surface in nonlinear multicriteria optimization problems, SIAM J. Optim. 8 (3) (1998) 631–657.
[18] B.Y. Chu, C.H. Ho, C.H. Tsai, C.Y. Lin, C.J. Lin, Warm start for parameter selection of linear classifiers, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015, pp. 149–158.
[19] N.S. Altman, An introduction to kernel and nearest-neighbor nonparametric regression, Am. Stat. 46 (3) (1992) 175–185.
[20] M. Kumar, N.K. Rath, S.K. Rath, Analysis of microarray leukemia data using an efficient MapReduce-based K-nearest-neighbor classifier, J. Biomed. Inform. 60 (2016) 395.
[21] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, European Conference on Computational Learning Theory, Springer, 1995, pp. 23–37.
[22] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[23] Rumelhart, E. David, Hinton, E. Geoffrey, Williams, J. Ronald, Learning representations by back-propagating errors, Nature 323 (6088) (1986) 533–536.
[24] B.W. Matthews, Comparison of the predicted and observed secondary structure of t4 phage lysozyme, Biochim. Biophys. Acta 405 (2) (1975) 442.