Neuroimage-based Clinical Prediction Using Machine Learning Tools

Ayşe Demirhan

Electrical and Electronics Engineering, Faculty of Technology, Gazi University, Ankara, Turkey

Received 1 December 2016; revised 17 January 2017; accepted 3 February 2017

ABSTRACT: Classification of structural brain magnetic resonance (MR) images is a crucial task for many neurological phenotypes that machine learning tools are increasingly developed and applied to solve this problem in recent years. In this study binary classification of T1-weighted structural brain MR images are performed using state-of-the-art machine learning algorithms when there is no information about the clinical context or specifics of neuroimaging. Image derived features and clinical labels that are provided by the International Conference on Medical Image Computing and Computer-Assisted Intervention 2014 machine learning challenge are used. These morphological summary features are obtained from four different datasets (each $N > 70$) with clinically relevant phenotypes and automatically extracted from the MR imaging scans using FreeSurfer, a freely distributed brain MR image processing software package. Widely used machine learning tools, namely; back-propagation neural network, self-organizing maps, support vector machines and k-nearest neighbors are used as classifiers. Clinical prediction accuracy is obtained via cross-validation on the training data ($N = 150$) and predictions are made on the test data ($N = 100$). Classification accuracy, the fraction of cases where prediction is accurate and area under the ROC curve are used as the performance metrics. Accuracy and area under curve metrics are used for tuning the training hyperparameters and the evaluation of the performance of the classifiers. Performed experiments revealed that support vector machines show a better success compared to the other methods on clinical predictions using summary morphological features in the absence of any information about the phenotype. Prediction accuracy would increase greatly if contextual information is integrated into the system. © 2017 Wiley Periodicals, Inc. Int J Imaging Syst Technol, 27, 89–97, 2017; Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/ima.22213

Key words: structural brain MR images; image classification; morphological features; clinical prediction; machine learning

I. INTRODUCTION

Structural MR imaging is a noninvasive and fast imaging technology that enables to examine organs and tissues in vivo. MR images (MRI) gives more clear and detailed images of the soft structures compared to the computer tomography (CT), X-ray, etc. It is an active research area that improving the quality of MRI (Güler et al., 2008; Meena Prakash and Shantha Selva Kumari, 2016) and developing new methods for early clinical diagnosis using MRI. Brain MRI provide high-quality images and rich information that are used to diagnose neurological diseases such as lesions, tumors, stroke and Alzheimer's disease (Baird and Warach, 1998; Frisoni et al., 2010; Demirhan and Güler, 2011; Yang and Fei, 2011; Demirhan et al., 2015b; Zhang et al., 2015).

Visual assessment of the MRI that plays a crucial role in the clinical diagnosis of the diseases related to the brain is used in clinical practice for qualitative analysis and MRI are marked manually for quantitative analysis. But the manual methods that use MRI for diagnosis or assessment of disease progression are subjective, not repetitive, time-consuming and tiring due to the huge amount of imaging data and the brightness and contrast of the displaying screen (Güler et al., 2009; Lladó et al., 2012; Zhang et al., 2015). Therefore, automated methods and tools are required for processing the MRI.

Automated classification of the brain MRI for the diagnostic purposes is a critical task to differentiate healthy and diseased images. Several automated methods were proposed for brain MRI classification in the last decade. Machine learning algorithms are the most extensively used methods for binary classification of the brain MRI. Chaplot et al. proposed an MRI classification method that use wavelets, self-organizing maps (SOM) and support vector machine (SVM) to discriminate between normal or abnormal images (Chaplot et al., 2006). Zhang et al. used wavelet transform for feature extraction and principal component analysis for feature reduction. Reduced features are sent to back-propagation (BP) neural network (NN) to classify a given brain MR image as normal or abnormal (Zhang et al., 2011). Katuwal et al. extracted brain morphometric features using FreeSurfer to classify structural brain MRI as autism spectrum disorder (ASD) or typically developing controls (TDC). They used Random Forest (RF), SVM and gradient boosting machine methods for classification. They used 373 ASD and 361 TDC male subjects from the ABIDE database and obtained 60% classification accuracy. The accuracy was improved to 67% when IQ and age information were added to morphometric features (Katuwal et al., 2015). Greenstein et al. used structural brain MRI and multivariate machine learning methods to classify schizophrenia and control groups. They obtained cortical and subcortical volumes using FreeSurfer and used

Correspondence to: Ayşe Demirhan; e-mail: ayseoguz@gazi.edu.tr

© 2017 Wiley Periodicals, Inc.

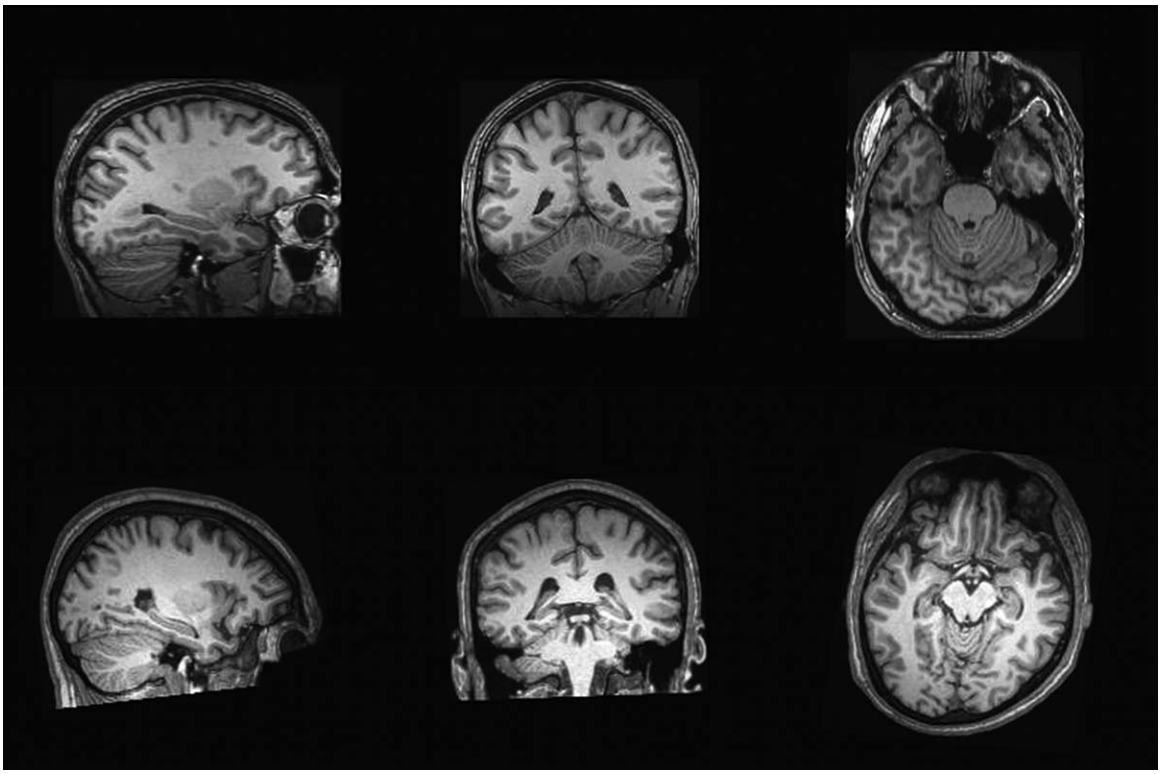


Figure 1. Sagittal, axial, and coronal MRI of a training subject (top panel) and a test subject (bottom panel).

RF for classification that reached 73% accuracy (Greenstein et al., 2012).

This study uses machine learning methods for binary classification of the structural brain MRI to investigate the effect of different algorithms on prediction accuracy. Prediction performance of the BP-NN, SOM, SVM, and k-nearest neighbors (kNN) are compared when the morphological summary features obtained from different datasets with clinically relevant phenotypes are used as input when there is no information about the clinical case.

II. MATERIALS AND METHODS

In this section details of the materials and methods used in this study are given. Structural brain MRI and morphological image features obtained from these MRI that are used as input are described. Properties and the implementation details of the machine learning methods that are used for binary classification are also stated.

A. Dataset. Image derived features and subject-specific clinical labels that are provided by the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2014 Machine Learning Challenge (MLC) are used in this study (MICCAI 2014 MLC; https://www.nmr.mgh.harvard.edu/lab/laboratory-computational-imaging-biomarkers/miccai-2014-machine-learning-challenge; Sabuncu and Konukoğlu, 2015). Summary features automatically extracted from the T1-weighted structural brain MRI that are obtained from large-scale neuroimaging studies are supplied in this challenge along with the original anonymized brain MRI in nifti format. FreeSurfer is used to process the structural brain MRI and compute morphological features that are provided by the challenge. FreeSurfer is used extensively by the researchers in neuroimaging studies since it is a greatly validated and an open source human brain MRI processing software (Fischl, 2012; FreeSurfer, freesurfer.net). A 184 features for 150 training and 100 test subjects are obtained from the original brain MRI. These features include volumes of cortical and subcortical structures such as the caudate and average thickness measurements within cortical regions, such as the precuneus. Because of head size variations, volume measurements are normalized with the intracranial volume. Details of the FreeSurfer image processing pipeline can be found at the web page of the challenge. Figure 1 gives example MRI of a training and a test subject.

B. Machine Learning Tools. State-of-the-art machine learning methods that are BP-NN, SOM, SVM, and kNN are used to evaluate the influence of different image processing tools on prediction accuracy. The implementation details of these methods are discussed in the following subsections. All the machine learning tools are applied using MATLAB.

B.1. Back-Propagation Neural Network. Back-propagation neural network (BP-NN) is a supervised and feed-forward network that has been used extensively in neuro-image based analysis like image registration, classification, and segmentation for the purpose of clinical diagnosis (Jiang et al., 2010; Demirhan, 2016). BP-NN has input, hidden, and output layers. Each layer has a number of neurons that has weight and bias values to map the input to the output. These weight and bias values can be adjusted dynamically by the supervised BP learning algorithm during the training of the network using gradient descent learning rule based on the error of the network output value compared to the target output value. Weights of the neurons are adjusted in the steepest descent direction that the performance function decreases most rapidly. BP of the error is used

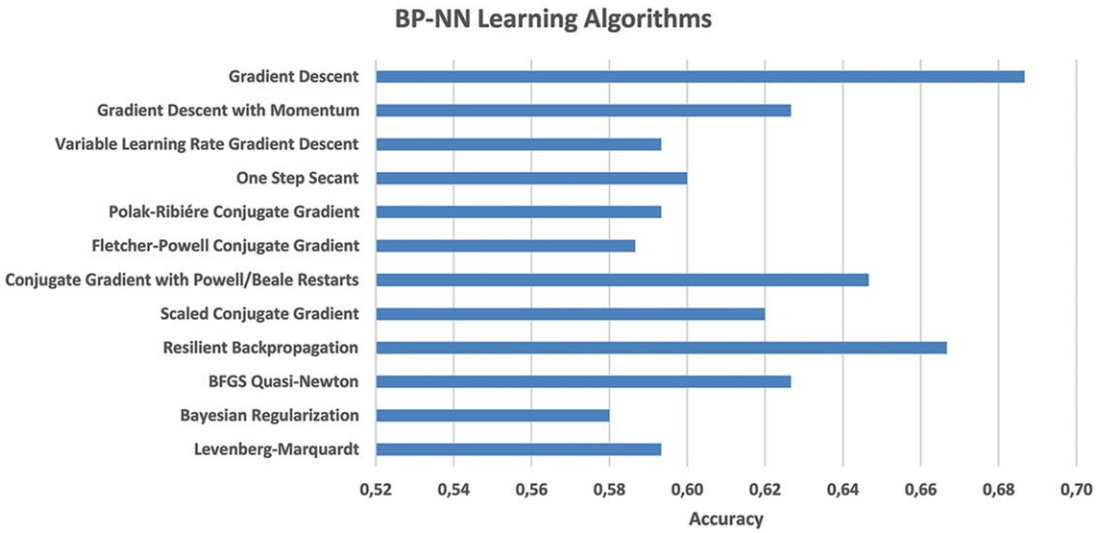


Figure 2. Performance comparison of BP-NN learning algorithms. [Color figure can be viewed at wileyonlinelibrary.com]

to compute the gradient of the error function relative to the hidden units after each training example is present to the network (Jiang et al., 2010; Demirhan, 2016).

Feed-forward BP-NN that has 1 hidden layer and 8 neurons at the hidden layer is used in this study. The network is trained using gradient descent BP learning algorithm that updates weight and bias values according to the gradient descent with 2,000 epochs, 0.01 learning rate, and 0.00001 minimum performance gradient. Parameters of the network are determined experimentally according to the correct classification accuracy that is calculated using five-fold cross validation (CV) of the training data.

B.2. Self-Organizing Maps. Self-organizing maps (SOM) uses the characteristic features of the input to map the input feature vector to an output space. SOM has two layers that are used to group the similar data together to reduce the dimension of the input vector. The first layer contains input nodes and the second layer contains output nodes. Output layer is in a two-dimensional grid format. Input nodes are connected to the output nodes and every output node is also connected to the other output nodes. All these connections are achieved with adjustable weights. SOM uses a competitive learning algorithm to cluster the data by having several output nodes compete for the current input. The network is trained by input feature vector and the weight vector of the closest unit to the current input becomes the winning node or best matching unit. Weight vectors of the winning node and the neighbors of the winning node are updated. Weight vector adjustments are calculated by using Eq. (1).

$$m_i(t+1)=m_i(t)+h_{ci}(t)+(x(t)-m_i(t)) \quad (1)$$

Here input of the SOM is $x(t)$ and weight vector for the unit i is $m_i(t)$. The $h_{ci}(t)$ is a symmetric and monotonically decreasing neighborhood function that specifies how much the winning unit signified by the subscript c and its neighbors is updated (Demirhan et al., 2015a). At the end of the training, output grid is ordered in a format that similar clusters get closer while dissimilar clusters get far from each other (Güler et al., 2009; Demirhan and Güler, 2010; Demirhan and Güler, 2011; Ortiz et al., 2013).

In this study, sequential learning algorithm that uses a hexagonal lattice and Gaussian neighborhood function is used to train the SOM network. Map grid size is determined as 15×10 that gives the highest prediction accuracy for the classification task. The grid is initialized using linear initialization along two greatest eigenvectors. The SOM network is trained in two phases; first rough training and then fine-tuning. SOM Toolbox that is a freely available software library for implementing the SOM algorithm is used for the training and testing of the SOM network (Alhoniemi et al., 2016; http://www.cis.hut.fi/projects/somtoolbox).

B.3. Support Vector Machines. Support vector machine (SVM) is a supervised binary classifier that is used frequently in neuroimaging studies for the purpose of diagnosis (Chaplot et al., 2006; Demirhan et al., 2015b; Zhang et al., 2015; Demirhan, 2016). SVM uses labeled training samples to learn class differences by mapping the input data to a very high-dimensional feature space in a nonlinear mode. Input data is divided into two classes by a decision surface

Table I. Performance comparison of the different number of hidden layers and neurons for the BP-NN

	1 Hidden Layer				2 Hidden Layers			
Number of Neurons	Accuracy	AUC	Sensitivity	Specificity	Accuracy	AUC	Sensitivity	Specificity
4	0.65	0.72	0.67	0.64	0.63	0.70	0.67	0.59
6	0.64	0.70	0.68	0.60	0.64	0.67	0.67	0.61
8	**0.69**	**0.75**	**0.72**	**0.65**	0.66	0.74	0.72	0.60
10	0.63	0.69	0.65	0.61	0.57	0.57	0.57	0.57
12	0.65	0.69	0.72	0.57	0.58	0.63	0.60	0.56
14	0.58	0.63	0.57	0.59	0.55	0.60	0.56	0.53

Table II. Number of epochs selection for the BP-NN

Number of Epochs	Accuracy	AUC	Sensitivity	Specificity
500	0.62	0.67	0.64	0.60
1,000	0.67	0.73	0.68	0.67
1,500	0.65	0.68	0.68	0.63
2,000	**0.69**	**0.75**	**0.72**	**0.65**
2,500	0.61	0.63	0.59	0.63
3,000	0.63	0.68	0.63	0.64

that is composed using a separating hyperplane. Separating hyperplane can be defined as $w'x+b=0$, where x is the input vector, b is the bias for the input vector and w is the weight vector. The decision surface is arranged by using the supportive and instructive samples of the training data. Minimizing w maximizes the margin between support vectors. Training constitutes two subspaces that corresponds to the two classes to be classified (Vert et al., 2004; Demirhan et al., 2015b).

There are polynomial, radial basis function (RBF) and sigmoid kernels that are used for nonlinear feature mapping of the SVM (Lukas et al., 2004; Vert et al., 2004; Demirhan et al., 2015b; Demirhan, 2016; Arbabshirani et al., accepted for publication in NeuroImage). The Gaussian RBF kernel is the most popular kernel in SVM due to its attractive characteristics like structure preserving. Definition of Gaussian RBF kernel is given in Eq. (2).

$$K(x,x')=\exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right) \quad (2)$$

Here σ is the width of the RBF function. RBF kernel has also a C parameter that is used for regularization of the soft-margin SVM. Small C values tend to produce a large margin, while $C=\infty$ leads to a hard margin. σ and C parameters determines the prediction accuracy of the SVM classifier with RBF kernel (Frisoni et al., 2010).

In this study, SVM is trained using sequential minimal optimization (SMO) learning function and the Gaussian RBF kernel. Because the parameter selection for σ and C is vital for performing the correct classification, the best parameter set is selected using a grid search with five-fold CV of the training data. Growing values of $C=[2^{-9},\ 2^{-8},\ldots,2^{15}]$ and $\sigma=[2^{-5},2^{-4},\ldots,2^{15}]$ are used to form a coarse grid. First a better region on the coarse grid is determined then a finer grid search is performed (Hsu et al., 2003). The best σ and C values obtained from grid search were used in training of the SVM.

B.4. k-Nearest Neighbors. kNN is a nonparametric supervised classifier that compares the data sample to be classified with the available training samples and finds the closest ones to it. Next step of the algorithm is to assign the class label in the majority to the data sample that will be classified among the labels of k-nearest training samples. The k value is the key parameter of the algorithm that is the number of nearest training examples in the feature space to be found for classification. Prediction performance of the kNN increases with the size of the training data. The main drawback of the kNN algorithm is that it has high computational requirements. kNN is preferred in neuroimage classification studies because it has a very simple algorithm and shows a stable performance (Cocosco et al., 2003; Zacharaki et al., 2011; Demirhan, 2016).

In this study, experiments are performed to find the best k value. k value is chosen as 13 that shows the best prediction accuracy.

C. Evaluation Metrics. Classification performance of the methods is measured quantitatively using accuracy, area under curve (AUC), sensitivity and specificity metrics. Let TP be true positive, FN be false negative, FP be false positive, and TN be true negative. Accuracy is the proportion of correct predictions among the total number of predictions. It is calculated as $(TP+TN)/(TP+TN+FP+FN)$. Sensitivity is also known as true positive rate (TPR) and it measures the probability that a classification result will be positive when the actual label is positive. Sensitivity is calculated as $TP/(TP+FN)$. Specificity is also known as true negative rate (TNR) and it is the probability that a classification result will be negative when the actual label is negative. Specificity is calculated as $TN/(FN+TN)$. AUC is

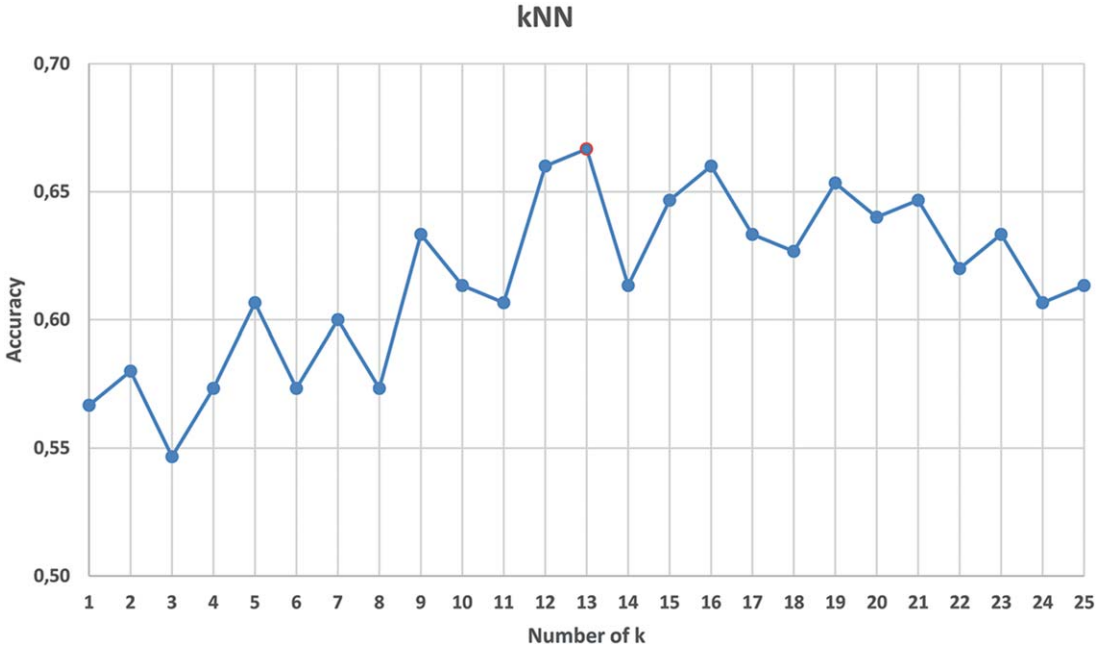


Figure 3. Accuracies for different k values of kNN algorithm. [Color figure can be viewed at wileyonlinelibrary.com]

Table III. Classification accuracies of different SOM network map sizes

Map Size	Accuracy	AUC	Sensitivity	Specificity
5 × 5	0.56	–	0.44	0.68
10 × 5	0.53	–	0.45	0.60
10 × 10	0.54	–	0.47	0.61
15 × 10	**0.61**	–	**0.56**	**0.67**
15 × 15	0.57	–	0.59	0.55
20 × 15	0.59	–	0.61	0.57
20 × 20	0.55	–	0.47	0.64

the average sensitivity over all possible specificities for evaluating classification performance. AUC is obtained by calculating the area under receiver operating characteristic (ROC) curve. The ROC curve is created by plotting the TPR against the false positive rate (FPR) at several threshold settings. Better performance of the classifier increases the TPR rate and the AUC.

III. RESULTS AND DISCUSSION

Binary classification of the T1-weighted structural brain MRI of the 150 training and 100 test subjects is performed using 184 summary features that include volumes and thickness of the anatomical structures of the brain. Both training and test data are divided into 55-folds that all folds include the same number of cases and controls. Parameter selection for all methods is performed using 5-fold CV of the training data to reduce the selection-related bias. Then the best parameters selected from the previous step is used for training and the model obtained from training is used to test the prediction performance of the methods. Before training and test operations input of the classifiers is normalized that each column of the input is centered to have zero mean and scaled to have unit standard deviation. The 5-fold CV is also used to evaluate the classification performance of the methods with accuracy, AUC, sensitivity and specificity metrics. Continuous predictions were made binary by thresholding with 0.5 to calculate accuracy. Values >0.5 are assigned as 1 that means predicted as class 1 and values <0.5 are assigned to class 0.

Parameters of the RBF kernel of the SVM, σ and C, are selected by performing a grid search that details of it is given in section B.3. Three different learning functions, namely SMO, iterative single data algorithm and $L1$ soft-margin minimization by quadratic programming are tested to find the best optimization routine but changing learning algorithm did not increase the accuracy of the classification. The decision function of the SVM produces a distance value of the input to the separating hyperplane. But analyzing posterior probability of classification with metrics like AUC is useful instead of relying on labels to reveal that a particular input falls into one class. The output of the SVM is converted to a posterior probability by using the sigmoid probability transformation function to calculate the AUC score.

Table IV. Prediction performance of the methods on the train dataset

	Accuracy		AUC		Sensitivity		Specificity	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
SVM	0.70	0.07	0.75	0.05	0.71	0.04	0.69	0.14
BP-NN	0.69	0.05	0.75	0.09	0.72	0.03	0.65	0.09
kNN	0.67	0.05	0.67	0.08	0.55	0.11	0.79	0.11
SOM	0.61	0.11	–	–	0.56	0.16	0.67	0.08

Parameters of the BP-NN are selected experimentally with 5-fold CV of the training data since there is not a standard way to determine them. Learning algorithm of the BP-NN is chosen as gradient descent according to the accuracy comparison given in Fig. 2. Number of hidden layers, number of neurons and number of epochs are other critical parameters that need to be determined for the BP-NN model. Table I gives a comparison of the performance obtained with different number of hidden layers and neurons. Number of hidden layers is selected as 1 and number of neurons in the hidden layer is selected as 8 since this parameter combination gives the best classification accuracies. BP-NN is also trained with different number of training epochs. Table II gives the mean performance measures that are obtained with 500–3,000 training epochs.

Selection of a good value for the number of nearest training examples, k, is the main factor that determines the efficiency of the kNN classification. Therefore, different k values are used to select the one that is the best to discriminate classes and k value is determined as 13. Fig. 3 gives accuracy comparison for different k values.

One of the parameters that affect the performance of the SOM network is map size that should be determined before training the

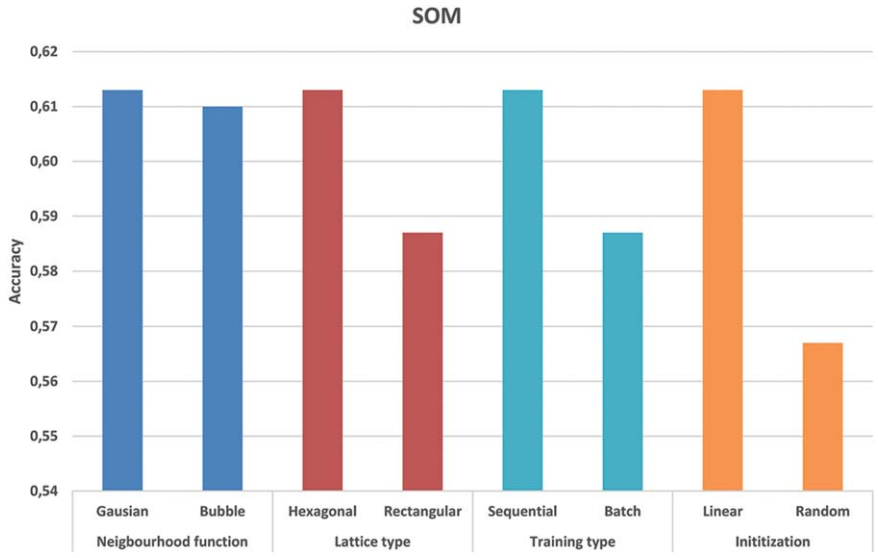


Figure 4. Accuracies for SOM network parameters. [Color figure can be viewed at wileyonlinelibrary.com]

Table V. Prediction performance of the methods on the test dataset

	Accuracy		AUC		Sensitivity		Specificity	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
SVM	0.62	0.05	0.68	0.10	0.62	0.13	0.62	0.21
BP-NN	0.57	0.09	0.61	0.13	0.60	0.19	0.54	0.24
kNN	0.54	0.08	0.56	0.10	0.64	0.09	0.44	0.22
SOM	0.61	0.07	–	–	0.54	0.11	0.68	0.13

network. Table III gives the effect of different map sizes on SOM network. As seen from Table III, 15 × 10 map size gives the best performance. Other SOM network parameters that affect the results of the network are neighborhood function, lattice type, training type and initialization. Gaussian or Bubble can be chosen as neighborhood function and hexagonal or rectangular can be chosen as lattice type. There are two training algorithms; sequential or batch and two initialization methods; linear or random that should be determined for constructing the SOM network model. Performance comparisons for all these parameters are given in Fig. 4. According to the results given in Fig. 4, selected parameters are Gaussian for the neighborhood function, hexagonal for the lattice type, sequential for the training algorithm and linear for the initialization method.

Table IV gives the mean and standard deviation of the performance measures that are obtained using 5-fold CV of the 150 training subjects. The best prediction accuracy is obtained using SVM for both accuracy and AUC scores. Specificity of the SVM is also the highest among all methods. These results are consistent with the prediction accuracies given in Table V that are obtained using 100 test subjects. Confidence intervals given in Table V are also achieved using 5-fold CV. AUC scores could not be calculated for SOM since SOM gives only the predicted classes not the prediction scores. Prediction performances obtained from both training and test data seems moderate but organizing team of the MLC challenge who supplied the data used in this study indicates that this classification problem is particularly challenging and a prediction accuracy over 0.6 is acceptable (MICCAI 2014 MLC, https://www.nmr.mgh.harvard.edu/lab/laboratory-computational-imaging-biomarkers/miccai-2014-machine-learning-challenge). Performance comparison of the methods on both training and test data are given in Fig. 5.

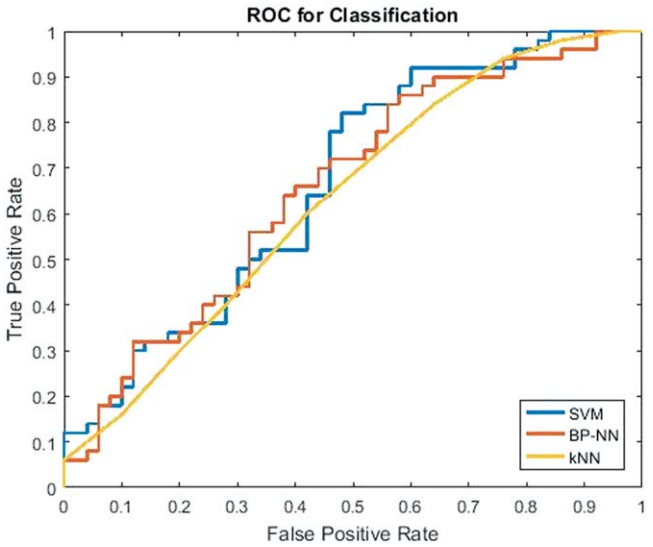


Figure 6. ROC Curves of SVM, BP-NN, and kNN. [Color figure can be viewed at wileyonlinelibrary.com]

ROC curves that are used to calculate AUC scores of the SVM, BP-NN, and kNN obtained from test data are given in Fig. 6. Clinical efficiency of the methods can be measured from these curves since the ROC curve measures the success of the methods over a range of test cut-off points instead of a single cut-off point as in accuracy. SVM and BP-NN are close in performance but SVM exhibits a slightly predictive advantage. From the Fig. 6, it can be seen that kNN shows the worst performance for this classification problem.

Estimation of generalization accuracy is an important part of the machine learning that shows how well a particular method will generalize using training data. Generalization accuracy computed via

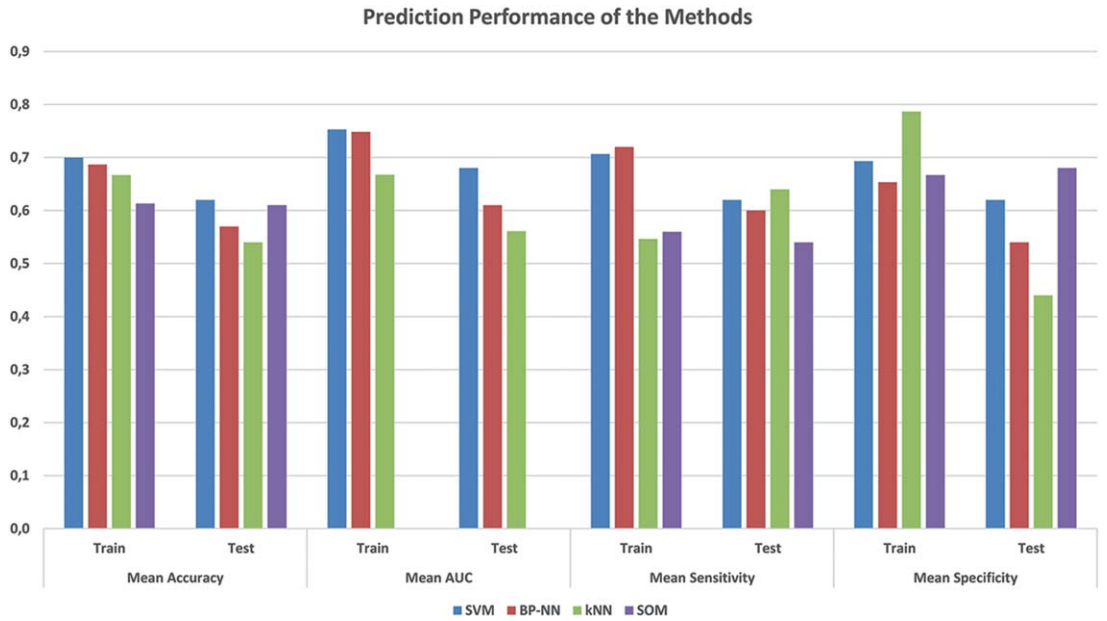


Figure 5. Prediction performances of the methods. [Color figure can be viewed at wileyonlinelibrary.com]

Table VI. Bootstrap confidence intervals and estimated performance measures

Method	Mean Accuracy	Std Accuracy	Accuracy Estimated 5-fold CV	Mean AUC	Std AUC	AUC Estimated 5-fold CV
SVM	0.60	0.05	0.70	0.67	0.06	0.75
BP-NN	0.57	0.05	0.69	0.66	0.05	0.75
kNN	0.56	0.05	0.67	0.63	0.06	0.67
SOM	0.51	0.05	0.61	–	–	–

Table VII. Comparison of challenge results

Order	User	Mean Accuracy	Std Accuracy	Accuracy - Reported 5-fold CV	Mean AUC	Std AUC	AUC—Reported 5-fold CV
1	roberto,bellotti	0.680 (1)	0.047 (51)	0.660 (30)	0.700 (2)	0.053 (27)	0.729 (22)
2	matthieu,le	0.670 (2)	0.047 (50)	0.807 (3)	0.689 (5)	0.053 (26)	0.877 (3)
3	onyme,anne	0.660 (3)	0.047 (49)	0.730 (12)	0.660 (14)	0.055 (21)	0.810 (7)
4	schreibensieruhig	0.650 (4)	0.048 (45)	0.710 (17)	0.650 (17)	0.044 (48)	0.760 (9)
5	ender,konukoglu	0.650 (5)	0.048 (48)	0.650 (34)	0.678 (7)	0.053 (25)	0.690 (35)
6	547901203	0.641 (6)	0.048 (47)	0.720 (15)	0.641 (19)	0.047 (46)	0.717 (28)
7	maglietta	0.640 (7)	0.048 (44)	0.680 (25)	0.706 (1)	0.052 (30)	0.725 (24)
8	sluo1	0.639 (8)	0.048 (43)	0.923 (1)	0.640 (21)	0.039 (49)	0.977 (1)
9	nicolas,cordier	0.631 (9)	0.048 (41)	0.690 (23)	0.650 (18)	0.056 (14)	0.710 (30)
10	jonathan,young	0.620 (10)	0.049 (34)	0.653 (32)	0.692 (4)	0.050 (34)	0.653 (45)
11	453783508	0.620 (11)	0.049 (33)	0.720 (15)	0.620 (25)	0.049 (40)	0.717 (28)
12	robben,david	0.620 (12)	0.049 (36)	0.633 (36)	0.674 (9)	0.054 (23)	0.686 (37)
13	m,goetz	0.620 (13)	0.049 (35)	0.693 (21)	0.640 (22)	0.056 (13)	0.748 (15)
14	sliu4512	0.620 (14)	0.048 (46)	0.807 (4)	0.673 (10)	0.052 (29)	0.817 (6)
15	203091484	0.619 (15)	0.049 (38)	0.767 (6)	0.619 (26)	0.049 (41)	0.767 (8)
16	sadoii	0.610 (16)	0.049 (40)	0.720 (15)	0.610 (29)	0.048 (44)	0.722 (25)
17	**This Study-SVM**	**0.601 (17)**	**0.049 (21)**	**0.700 (19)**	**0.669 (12)**	**0.055 (15)**	**0.753 (12)**
18	virgile,fritsch+mlc2014_3	0.601 (18)	0.049 (32)	0.747 (10)	0.658 (15)	0.055 (20)	0.747 (16)
19	robbendavid	0.600 (19)	0.049 (37)	0.633 (36)	0.699 (3)	0.052 (31)	0.706 (34)
20	weiyi1991	0.600 (20)	0.049 (30)	0.767 (6)	0.600 (30)	0.049 (37)	0.767 (8)
21	1058049527	0.600 (21)	0.049 (31)	0.720 (15)	0.600 (31)	0.049 (39)	0.718 (27)
22	hungway,lee	0.599 (22)	0.049 (39)	0.713 (16)	0.599 (32)	0.049 (42)	0.713 (29)
23	hyshen	0.599 (23)	0.048 (42)	0.690 (23)	0.672 (11)	0.054 (22)	0.660 (43)
24	virgile,fritsch+mlc2014_2	0.590 (24)	0.049 (25)	0.727 (14)	0.680 (6)	0.054 (24)	0.756 (10)
25	amtorteya	0.581 (25)	0.049 (29)	0.760 (7)	0.581 (35)	0.049 (38)	0.760 (9)
26	a_m_torteya	0.580 (26)	0.050 (19)	0.727 (13)	0.580 (36)	0.049 (35)	0.727 (23)
27	jose,cpadilla	0.570 (27)	0.050 (13)	0.767 (6)	0.570 (37)	0.050 (33)	0.767 (8)
28	congjinyu1231	0.570 (28)	0.050 (16)	0.747 (9)	0.570 (38)	0.048 (45)	0.749 (13)
29	wakamo	0.570 (29)	0.049 (24)	0.790 (5)	0.611 (28)	0.056 (11)	0.850 (5)
30	alexandermendelson	0.561 (30)	0.050 (17)	0.667 (29)	0.485 (49)	0.060 (1)	0.680 (39)
31	andrea,tateo	0.560 (31)	0.049 (28)	0.673 (28)	0.589 (33)	0.057 (9)	0.710 (31)
32	joseh85102	0.560 (32)	0.049 (27)	0.687 (24)	0.560 (39)	0.049 (36)	0.687 (36)
33	virgile,fritsch	0.550 (33)	0.050 (20)	0.680 (25)	0.497 (48)	0.059 (5)	0.709 (33)
34	paolo,ingls	0.540 (34)	0.049 (22)	0.680 (25)	0.640 (20)	0.055 (18)	0.733 (20)
35	sonia,tangaro2	0.540 (35)	0.050 (9)	0.693 (22)	0.634 (23)	0.056 (12)	0.745 (17)
36	leonardo,iheme2	0.531 (36)	0.050 (18)	0.755 (8)	0.530 (44)	0.031 (51)	0.867 (4)
37	matthew,blaschko	0.530 (37)	0.050 (8)	0.633 (37)	0.621 (24)	0.058 (6)	0.684 (38)
38	rosangela,errico	0.520 (38)	0.050 (12)	0.660 (30)	0.656 (16)	0.055 (16)	0.739 (19)
39	nicola,amoroso	0.520 (39)	0.050 (3)	0.687 (24)	0.674 (8)	0.055 (17)	0.755 (11)
40	luohuilan	0.520 (40)	0.050 (15)	0.679 (26)	0.616 (27)	0.052 (28)	0.744 (18)
41	a00789505	0.520 (41)	0.050 (5)	0.628 (38)	0.541 (43)	0.059 (3)	0.668 (41)
42	a00811434	0.511 (42)	0.049 (23)	0.644 (35)	0.545 (42)	0.058 (7)	0.679 (40)
43	alfonso,monaco	0.500 (43)	0.050 (2)	0.651 (33)	0.668 (13)	0.055 (19)	0.709 (32)
44	sd,ve91	0.500 (44)	0.050 (14)	0.659 (31)	0.500 (45)	0.037 (50)	0.655 (44)
45	msabuncu	0.500 (45)	0.050 (11)	0.700 (20)	0.587 (34)	0.057 (8)	0.600 (47)
46	simon,duchesne	0.499 (46)	0.050 (10)	0.853 (2)	0.499 (46)	0.045 (47)	0.943 (2)
47	david,robben	0.499 (47)	0.050 (7)	0.673 (27)	0.548 (41)	0.059 (4)	0.719 (26)
48	407223471	0.499 (48)	0.050 (1)	0.733 (11)	0.499 (47)	0.048 (43)	0.731 (21)
49	leonardo,iheme	0.480 (49)	0.050 (4)	0.607 (39)	0.480 (50)	0.051 (32)	0.667 (42)
50	melanie,ganz	0.470 (50)	0.050 (6)	0.707 (18)	0.557 (40)	0.060 (2)	0.748 (14)
51	phenriquezcastellano	0.460 (51)	0.049 (26)	0.604 (40)	0.414 (51)	0.057 (10)	0.604 (46)
52	alexander,mendelson,11	−100.000 (52)	−100.000 (52)	−100.000 (41)	−100.000 (52)	−100.000 (52)	−100.000 (48)
53	sonia,tangaro	−100.000 (52)	−100.000 (52)	0.693 (22)	−100.000 (52)	−100.000 (52)	0.745 (17)
54	alexander,meson	−100.000 (52)	−100.000 (52)	−100.000 (41)	−100.000 (52)	−100.000 (52)	−100.000 (48)
55	joseh8510	−100.000 (52)	−100.000 (52)	−100.000 (41)	−100.000 (52)	−100.000 (52)	−100.000 (48)

CV on the train dataset quantifies generalization ability of the classifiers. There is also a need to define classification confidence for the model specific approaches to estimate sampling distribution and calculate confidence intervals like mean and standard deviation of the classification accuracies. Numerical iterative sampling like bootstrapping can be used for this purposes. Bootstrapping is a nonparametric statistical method that does not need assumptions regarding the populations of the input variables and sampling with replacement is done many times to estimate the variability of the mean and variance of model outputs. Confidence intervals for each evaluation measure with 10,000 bootstrap (with replacement) samples are calculated for the classification results of the test dataset and given in Table VI along with the accuracy and AUC measures estimated with 5-fold CV using the training dataset. For the evaluation computations, scikit-learn package in python is used.

SVM gives the best bootstrap accuracies as expected according to the results obtained using both training and test data given in Tables IV and V. SVM bootstrap results are used to compare the results submitted to the MICCAI MLC Codalab website (CodaLab Competition, 2016; https://competitions.codalab.org/competitions/1471#results). Performance order of the studies is re-organized to include this study (Table VII). It can be concluded that prediction accuracy of the SVM without any modifications outperforms its several competitors and shows good performance for the classification task of the T1-weighted structural brain MRI when only summary morphometric features obtained from different datasets with clinically relevant phenotypes are supplied as input without any information related to the clinical case. Performances of the BP-NN, kNN and SOM can be interpreted as moderate. Classification results can be increased by incorporating additional features related to the clinical phenotype and specifics of imaging.

ACKNOWLEDGMENT

The author thanks the MICCAI 2014 MLC Workshop organizers for supplying the data used in this study. The author also thanks Ender Konukoğlu for supplying the test labels which are not present at MLC web page.

REFERENCES

E. Alhoniemi, J. Himberg, J. Parhankangas, and J. Vesanta, SOM Toolbox for MATLAB, Available at: http://www.cis.hut.fi/projects/somtoolbox (last accessed November 2, 2016).

M.R. Arbabshirani, S. Plis, J. Sui, V.D. Calhoun, Single subject prediction of brain disorders in neuroimaging: Promises and pitfalls, (accepted for publication in NeuroImage) doi: Available at: https://doi.org/10.1016/j.neuroimage.2016.02.079.

A.E. Baird and S. Warach, Magnetic resonance imaging of acute stroke, J Cereb Blood Flow Metab 18 (1998), 583–609.

S. Chaplot, L.M. Patnaik, and N.R. Jagannathan, Classification of magnetic resonance brain images using wavelets as input to support vector machine and neural network, Biomed Signal Proces 1 (2006), 86–92.

C.A. Cocosco, A.P. Zijdenbos, and A.C. Evans, A fully automatic and robust brain MRI tissue classification method, Med Image Anal 7 (2003), 513–527.

CodaLab Competition. Available at: https://competitions.codalab.org/competitions/1471#results (last accessed November 2, 2016).

A. Demirhan, Classification of structural MRI for detecting Alzheimer's disease, ıjısae 4 (2016), 195–198.

A. Demirhan and İ. Güler, Image segmentation using self-organizing maps and gray level co-occurrence matrices, J Fac Eng Arch Gazi Univ 25 (2010), 285–291.

A. Demirhan and İ. Güler, Combining stationary wavelet transform and self-organizing maps for brain MR ımage segmentation, Eng Appl Artif Intel 24 (2011), 358–367.

A. Demirhan, T.M. Nir, A. Zavaliangos-Petropulu, C.R. Jack, M. Weiner, M. Bernstein, P. Thompson, and N. Jahanshad, Feature selection ımproves the accuracy of classifying Alzheimer disease using diffusion tensor ımages, Proc Int Symp Biomed Imaging (ISBI) 2015-July (2015b), 126–130, Article number: 7163832.

A. Demirhan, M. Törü, and İ. Güler, Segmentation of tumor and edema along with healthy tissues of brain using wavelets and neural networks, IEEE J Biomed Health Inform 19 (2015a), 1451–1458.

B. Fischl, FreeSurfer, Neuroimage 62 (2012), 774–781.

FreeSurfer. Available at: freesurfer.net (last accessed November 2, 2016).

G.B. Frisoni, N.C. Fox, C.R. Jack, Jr., P. Scheltens, and P.M. Thompson, The clinical use of structural MRI in Alzheimer disease, Nat Rev Neurol 6 (2010), 67–77.

D. Greenstein, J.D. Malley, B. Weisinger, L. Clasen, and N. Gogtay, Using multivariate machine learning methods and structural MRI to classify childhood onset schizophrenia and healthy controls, Front Psychiatry 3 (2012), 53.

İ. Güler, A. Demirhan, and R. Karakış, Interpretation of MR images using self-organizing maps and knowledge-based expert systems, Digit Signal Process 19 (2009), 668–677.

İ. Güler, A. Toprak, A. Demirhan, and R. Karakış, MR images restoration with the use of fuzzy filter having adaptive membership parameters, J Med Syst 32 (2008), 229–234.

C.-W. Hsu, C.-C. Chang, and C.-J. Lin, A practical guide to support vector classification, Technical Report, Department of Computer Science, National Taiwan University, Taipei, Taiwan, 2003.

J. Jiang, P. Trundle, and J. Ren, Medical image analysis with artificial neural networks, Comput Med Imaging Graph 34 (2010), 617–631.

G.J. Katuwal, N.D. Cahill, S.A. Baum, and A.M. Michael, The predictive power of structural MRI in Autism diagnosis, 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2015, pp. 4270–4273.

X. Lladó, A. Oliver, M. Cabezas, J. Freixenet, J.C. Vilanova, A. Quiles, L. Valls, L. Ramió-Torrentà, and À. Rovira, Segmentation of multiple sclerosis lesions in brain MRI: A review of automated approaches, Inform Sci 186 (2012), 164–185.

L. Lukas, A. Devos, J.A.K. Suykens, L. Vanhamme, F.A. Howe, C. Majós, A. Moreno-Torres, M. Van Der Graaf, A.R. Tate, C. Arús, and S. Van Huffel, Brain tumor classification based on long echo proton MRS signals, Artif Intell Med 31 (2004), 73–89.

R. Meena Prakash and R. Shantha Selva Kumari, Fuzzy C means integrated with spatial information and contrast enhancement for segmentation of MR brain images, Int J Imaging Syst Technol 26 (2016), 116–123.

MICCAI 2014 Machine Learning Challenge. Available at: https://www.nmr.mgh.harvard.edu/lab/laboratory-computational-imaging-biomarkers/miccai-2014-machine-learning-challenge (last accessed November 2, 2016).

A. Ortiz, J.M. Górriz, J. Ramírez, D. Salas-Gonzalez, and J.M. Llamas-Elvira, Two fully-unsupervised methods for MR brain image segmentation using SOM-based strategies, Appl Soft Comput 13 (2013), 2668–2682.

M.R. Sabuncu E. Konukoglu, and The Alzheimer's Disease Neuroimaging Initiative, Clinical prediction from structural brain MRI scans: A large-scale empirical study, Neuroinform 13 (2015), 31–46.

J.P. Vert, L. Tsuda, and B. Schölkopf, A primer on kernel methods," In Kernel methods in computational biology, B. Schölkopf, L. Tsuda, and J.P. Vert (Editors), MIT Press, Cambridge, Massachusetts, 2004, pp. 35–70.

X. Yang and B. Fei, A multiscale and multiblock fuzzy C-means classification method for brain MR images, Med Phys 38 (2011), 2879–2891.

E.I. Zacharaki, V.G. Kanas, and C. Davatzikos, Investigating machine learning techniques for MRI-based classification of brain neoplasms, Int J Comput Assist Radiol Surg 6 (2011), 821–828.

Y. Zhang, Z. Dong, S. Wang, G. Ji, and J. Yang, Preclinical diagnosis of magnetic resonance (MR) brain images via discrete wavelet packet transform with Tsallis entropy and generalized eigenvalue proximal support vector machine (GEPSVM), Entropy 17 (2015), 1795–1813.

Y. Zhang, Z. Dong, L. Wu, and S. Wang, A hybrid method for MRI brain image classification, Expert Syst Appl 38 (2011), 10049–10053.