Engineering Applications of Artificial Intelligence 53 (2016) 176–189

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

A sparse extreme learning machine framework by continuous optimization algorithms and its application in pattern recognition

Liming Yang*, Siyun Zhang

College of Science, China Agricultural University, Beijing 100083, China

ARTICLE INFO

Article history:
Received 17 July 2015
Received in revised form
14 January 2016
Accepted 11 April 2016
Available online 11 May 2016

Keywords:
Extreme learning machine
Zero-norm
DC programming
Exact penalty technique
Least absolute deviation
Hardness of licorice seeds

ABSTRACT

Extreme learning machine (ELM) has demonstrated great potential in machine learning owing to its simplicity, rapidity and good generalization performance. In this investigation, based on least-squares estimate (LSE) and least absolute deviation (LAD), we propose four sparse ELM formulations with zero-norm regularization to automatically choose the optimal hidden nodes. Furthermore, we develop two continuous optimization methods to solve the proposed problems respectively. The first is DC (difference of convex functions) approximation approach that approximates the zero-norm by a DC function, and the resulting optimizations are posed as DC programs. The second is an exact penalty technique for zero-norm, and the resulting problems are reformulated as DC programs, and the corresponding DCAs converge finitely. Moreover, the proposed framework is applied directly to recognize the hardness of licorice seeds using near-infrared spectral data. Experiments in different spectral regions illustrate that the proposed approaches can reduce the number of hidden nodes (or output features), while either improve or show no significant difference in generalization compared with the traditional ELM methods and support vector machine (SVM). Experiments on several benchmark data sets demonstrate that the proposed framework is competitive with the traditional approaches in generalization, but selects fewer output features.

1. Introduction

Extreme learning machine (ELM) (Huang et al., 2006, 2010) is a popular and important learning algorithm for single-hidden-layer feedforward neural networks (SLFNs) (Huang et al., 2006). With good generalization performance, ELM has been applied successfully in regression and classification applications. Compared with traditional neural networks, the main advantages of ELM are that it runs fast and is easy to implement. Its hidden nodes and input weights are randomly generated and the output weights are expressed analytically. Moreover ELM overcomes some drawbacks of traditional neural networks, such as local minima, imprecise learning rates and slow convergence rates. However, the traditional ELM does not explicitly combine output features (or hidden nodes) and generalization of the model, which makes it difficult to control automatically the balance between prediction accuracy and the number of selected features.

According to statistical learning theory (SLT) (Vapnik, 1998), to ensure better generalization performance on test set, an algorithm should not only achieve low training error on training set, but also have a lower Vapnik–Chervonenkis (VC) dimension. Recently, researches (Liu et al., 2012; Huang et al., 2015) indicate that the VC dimension of ELM has a specific value and depends strongly on the number of the hidden-layer nodes. In addition, according to the theories (Liu et al., 2012; Huang et al., 2015), ELM has universal approximation capability. It can achieve low approximation error on training set. Therefore, ELM is a potential learning method, and its hidden layer neurons are important for building ELM network with good generalization.

However, some hidden nodes might be closely correlated owing to the randomness of the input weights and hidden node biases in ELM. Thus it is very necessary for regularization to prevent over-fitting and enhance the generalization capability. In addition, ELM outputs its weight based on the least-squares estimate (LSE) (Xiang et al., 2012), and its outputs lack sparseness. Therefore looking for compact ELM networks and choosing the optimal hidden nodes remain important subjects to achieve good performance. Recently, several techniques have been developed to obtain the sparse ELM network, which are summarized as follows:

(1) Double-regularized ELM models such as TROP-ELM (Miche et al., 2011) and regularized ELM with missing data (Yu et al., 2013).

TROP-ELM is an improvement of the optimally pruned extreme learning machine (OP-ELM) (Miche et al., 2010). It uses a cascade

* Corresponding author.
E-mail addresses: cauyanglm@163.com (L. Yang), 944890706@qq.com (S. Zhang).

http://dx.doi.org/10.1016/j.engappai.2016.04.003
0952-1976/

of two regularization penalties, the l_1-norm and l_2-norm. TROP-ELM first constructs a SLFN like ELM, then ranks the best neurons by l_1 regularization, finally selects the optimal number of neurons by l_2 regularization. TROP-ELM introduces the l_2 regularization in the calculation of the pseudo-inverse by the singular value decomposition (SVD), and it uses the leave-one-out (LOO) error to select the optimal number of neurons. Thus this is complicated to implement. The regularized ELM with missing data (Yu et al., 2013) is a modification version of TROP-ELM. It uses a cascade of l_1-penalty and l_2-penalty in ELM to solve the missing data problem.

(2) Robust ELM models such as RELM (Horata et al., 2013) and robust ELM with outliers (Barreto and Barros, 2016).

The RELM (Horata et al., 2013) proposes an extended complete orthogonal decomposition (ECOD) method to compute the weights of the ELM. And the paper also proposes the other three algorithms—the iteratively reweighed least squares (IRWLS-ELM), ELM based on the multivariate least-trimmed squares (MLTS-ELM) and ELM based on the one-step reweighed MLTS (RMLTS-ELM)—to solve the outlier robustness problems. Robust ELM with outliers (Barreto and Barros, 2016) is designed to apply M-estimators (Bai and Wu, 1997) in the output weights instead of the standard ordinary least squares method.

(3) Sparse ELM models such as the $l_{1/2}$ regularized ELM (Khan et al., 2014; Han et al., 2015) and l_1-regularization approach (Balasundaram and Kapil, 2014; Luo and Zhang, 2014).

The use of l_1-norm regularization results in sparse solutions, thereby helping with feature selection. However, the l_1-norm regularization scheme is consistent in feature selection under some conditions with restrictive assumptions, while there exist certain cases where l_1-penalty technology is inconsistent in feature selection (Zou, 2006; Lin et al., 2009; Le Thi et al., 2015). Note that the l_1-norm regularization criterion generates many components that are close to zero but not exactly equal to zero. The $l_{1/2}-$norm regularization is easier to solve than the l_0-norm regularization, and more sparse than the l_1-norm regularization. But the performance of sparse representation using the l_0-norm regularization is stronger than that of the $l_{1/2}-$norm regularization.

(4) Regularized ELM such as pre-fitting and back-fitting approach (Li et al., 2014) and regularized ELM (Iosifidis et al., 2015).

The pre-fitting and back-fitting approach (Li et al., 2014) is with l_2-norm regularization. This two-stage approach is a greedy algorithm and time-consuming. RELM (Iosifidis et al., 2015) is based on Frobenius norm of matrix. This approach can choose appropriately hidden layer weights and leads to ELM space dimensions having varying values for different training samples. Usually, these methods cannot automatically produce sparse representation.

Feature selection for classification and regression problems is an important topic with many applications, the objectives of which are two-fold: selecting a small feature subset while maintaining high accuracy. Specifically, feature selection for a linear decision function $f(x) = \mathrm{sgn}(\beta^T x)$ can be posed as searching for a sparse weight vector β such that most elements of β are zero. This implies that when the ith component of β is zero, the ith component of an observation vector x is irrelevant to the class of x. The zero-norm of the vector β, $\| \beta \|_0 = card\{i|\beta_i \neq 0\}$, is defined to be the number of nonzero elements in β, meaning that zero-norm regularization criterion allows us to reduce the number of representative features in the decision function $f(x)$. Thus feature selection for the decision function $f(x)$ usually is posed as minimizing the $\| \beta \|_0$ under appropriate constraint conditions. Nevertheless, the sparse ELM model based on the zero-norm is relatively few discussed in the literature, the main reason for which is the discontinuity and nonconvexity of the zero-norm. Therefore, most work in dealing with feature selection has focused on effective approximation of the zero-norm. The l_1-norm is only a convex approximation of the zero-norm (see Fig. 1). Therefore, the main questions for feature selection include how to approximate the zero-norm effectively and which computational method to use for solving the resulting optimization problem.

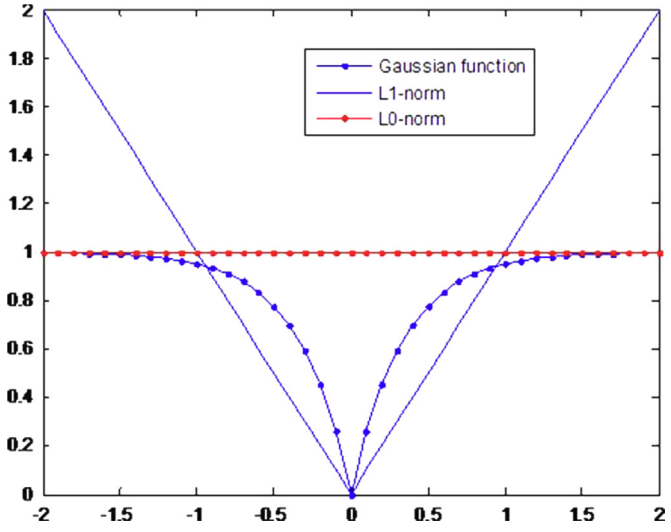

Fig. 1. Approximations to zero-norm for the Gaussian function $\eta(z)$.

In this paper, based on least-squares estimate (LSE) and least absolute deviation (LAD) (Cao and Liu, 2009; Yang et al., 2011), we present two sparse ELM frameworks with zero-norm regularization to select automatically output features. Moreover we present four continuous methods to solve the proposed problems. The first is a DC (Tao and An, 1997; Le Thi et al., 2014, 2008, 2015) approximation approach that approximates the zero-norm by a DC function. The second applies a new exact penalty technique (Le Thi et al., 2014) to reformulate equivalently the original problem as DC programs. The resulting problems all are posed as DC programs. The corresponding DC optimization algorithms converge linearly or finitely and only requires solving one quadratic program at each iteration.

Throughout the paper we adopt the following notations. The scalar product of two vectors x and y in the n-dimensional real space is denoted by $x^T y$ or $\langle x \cdot y \rangle$. For a n-dimension vector x, $\| x \|_1$ denotes the l_1-norm of x, $\| x \|_1 = \sum_{i=1}^{i=n} |x_i|$, where $|\cdot|$ denotes absolute value operator, and $\| x \|_2$ denotes the l_2-norm of x, $\| x \|_2 = \sqrt{x^T x}$. The base of the natural logarithm is denoted by ϵ. A vector of zeros in a real space of arbitrary dimension is denoted by 0. An arbitrary dimension vector of ones is denoted by e.

The rest of the paper is organized as follows. Section 2 briefly summarizes DC programming and ELM. In Section 3, we propose a sparse ELM framework with the zero-norm regularization, and develop four nonconvex optimization algorithms to solve the problems. The experimental results are analyzed in Section 4 and Section 5 give the concluding remarks, summarizes the main contributions of this work, and presents future directions of investigation.

2. Background

2.1. DC programming

DC programming and DCA, introduced by Pham Dinh (constitute the backbone of nonconvex continuous programming (Tao and An, 1997; Le Thi et al., 2014, 2008, 2015). Generally speaking, a DC program takes the form