Neurocomputing 211 (2016) 129–142

Contents lists available at ScienceDirect

Neurocomputing

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

Building support vector machines in the context of regularized least squares

Jian-Xun Peng, Karen Rafferty *, Stuart Ferguson

The School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Ashby Building, Stranmillis Road, Belfast BT9 5AH, UK

ARTICLE INFO

Article history:
Received 19 August 2015
Received in revised form
8 March 2016
Accepted 21 March 2016
Available online 8 June 2016

Keywords:
Data classification
Support vector machines
Regularized least squares
Fast training algorithm
Cholesky decomposition

ABSTRACT

This paper formulates a linear kernel support vector machine (SVM) as a regularized least-squares (RLS) problem. By defining a set of indicator variables of the errors, the solution to the RLS problem is represented as an equation that relates the error vector to the indicator variables. Through partitioning the training set, the SVM weights and bias are expressed analytically using the support vectors. It is also shown how this approach naturally extends to sums with nonlinear kernels whilst avoiding the need to make use of Lagrange multipliers and duality theory. A fast iterative solution algorithm based on Cholesky decomposition with permutation of the support vectors is suggested as a solution method. The properties of our SVM formulation are analyzed and compared with standard SVMs using a simple example that can be illustrated graphically. The correctness and behavior of our solution (merely derived in the primal context of RLS) is demonstrated using a set of public benchmarking problems for both linear and nonlinear SVMs.

1. Introduction

Support vector machines (SVMs) are a set of empirical data modeling techniques that is firmly grounded in the framework of *VC theory* [1], a specific approach to computational learning theory. The SVM was originally developed for binary data classification problems. Conceptually, a machine maps the input space to a so-called *feature space* through some non-linear mapping chosen *a priori*. The feature space is of higher dimension than the input space. In this feature space a linear decision surface is constructed based on the structural risk minimization (SRM) principal: an upper bound on the expected risk (the expectation of the test error for a machine on an unseen point) is minimized by maximization of the margin [2]. The margin refers to the distance between the two parallel hyperplanes that bound the training points of the two classes, respectively. The hyperplane that lies midway between the two bounding hyperplanes is called the decision hyperplane, and the training points that determine the two parallel bounding hyperplanes are referred to as the *support vectors*.

It was shown [2] that if the training vectors are separated without errors by an optimal hyperplane the expectation value of the probability of committing an error on a test example is bounded above by the ratio between the expectation value of the number of support vectors and the number of training vectors. Particularly, this bound does not explicitly contain the dimensionality of the feature space. It follows from this bound, that if the optimal hyperplane can be constructed from a small number of support vectors, relative to the training set size, then the generalization ability will be high even in an infinite dimensional feature space.

This optimal margin algorithm is generalized by [3] to non-separable data sets by the introduction of non-negative slack variables as a measurement of the misclassification errors in the statement of the optimization problem, and by using a structural objective function with a penalty term on the training errors. For a sufficiently large penalty parameter C, the hyperplane is chosen so that it minimizes the number of errors on the training set, while the rest of the training points are separated with maximal margin; if the training data can be separated without errors, then the hyperplane obtained in the procedure coincides with the optimal margin hyperplane.

Compared with traditional methods employed by conventional neural networks, the SRM principle has been shown to be superior because it not only minimizes the error on the training data [4], but also minimizes the capability of the model [5]. This equips SVM with a greater ability to generalize, which is the goal in statistical learning. Experimental studies have demonstrated the competitive performance of SVMs in a range of application fields [6–9].

Typically, constructing a SVM involves a constrained quadratic (or convex) optimization problem. In the majority of textbooks and articles introducing SVMs, instead of directly solving the primal problem, a dual of the problem is formulated using Lagrange multipliers [3,10–12]. There are two reasons for doing this [10]:

* Corresponding author.
E-mail address: k.rafferty@ee.qub.ac.uk (K. Rafferty).

http://dx.doi.org/10.1016/j.neucom.2016.03.087
0925-2312/

(a) duality theory provides a convenient way to deal with the constraints and (b) with the Lagrange reformulation of the problem, the training data will only appear (in the actual training and test phases) in the form of dot products between input vectors. This is a crucial property which allows us to generalize SVMs to the nonlinear case. In addition, most popular algorithms and existing toolboxes (for example, interior point method [13] and the sequential minimal optimization (SMO) [14] algorithm) formulate their solution in the dual. This gives the strong impression that this is the only possible way to construct a SVM, particularly for SVMs with nonlinear kernels. There has also been quite a lot of interest in studying systems that have particular properties, for example Huang et al.'s [15] work on sparse learning and Li et al.'s [16] sparse least squares. However, there has been increasing interest in constructing SVMs directly in the primal. Fung and Mangasarian [17] formulated a primal least-square version of the SVM, which had been originally proposed in the dual [18]. Komarek [19] applied conjugate gradient schemes to logistic regression for data classification. Zhang et al. [20] proposed an algorithm for linear L1-SVMs that works by approximating the L1-loss function by a sequence of smooth modified logistic regression loss functions, this is followed by sequentially solving smooth primal modified logistic regression problems using nonlinear conjugate gradient methods. However, all the inequality constraints are replaced by equality constraints in least squares SVM. A particular drawback of that method is its inability to exploit the sparsity property of SVMs in which only the support vectors determine the final solution. To overcome this shortage of the least squares SVM, a pruning method was proposed based on the fact that support values reveal the relative importance of each of the training data points, where a small number of points, e.g., 5% in the training set [21], that have the smallest values in the sorted support vector values spectrum, are removed in each training loop, until some user-defined performance index degrades.

Some promising primal algorithms have also been studied for standard linear SVMs [22–24], and implemented in toolboxes for linear SVMs, for example, in LIBLINEAR [25]. All of these algorithms are based on the fact that, for linear SVM, the feature space is the same as the input space, the normal vector to the separating hyperplane is thus explicitly presented in the linear SVM. However, for SVMs with nonlinear kernels, where some nonlinear map from the input space to the feature space exists, the map itself and many of its properties are unknown [26]. What is known is, a given kernel function involving a dot product in the feature space, a concept introduced by [27], thus the normal is not explicitly present in the final discriminative function of nonlinear SVMs again. This makes it difficult to apply primal solution algorithms in nonlinear kernel cases.

Chapelle [28] showed that when the goal is to find an approximate solution, primal optimization is superior because it is focused on minimizing what we are directly interested in: the primal objective function. Motivated by this, a Newton method is applied to the primal problem for both linear and nonlinear cases. For the nonlinear case, the optimal solution to the SVM is expressed by a linear combination of the kernel functions evaluated in all the training points based on the *representer theorem* of [29]. Given this linear combination solution, and using the representing property of the kernel, the problem is thus converted into one of optimizing the linear coefficients in the combination. This requires the full kernel matrix to be invertible (positive definite), given that the full kernel matrix is a symmetric matrix formed by pair-wise point inner products or kernel evaluations on the full training set. An iterative technique, IRWLS [30,31], based on re-weighed least squares produced the fastest algorithm of its time. The IRWLS approach was subsequently proved to converge to the SVM solution [32]. Since then there has been continued interest in primal and iterative least squares approaches to finding the best SVM solution [33–35].

Recently, particularly in the machine learning arena, recursive and weighted least squares has attracted interest in the context of twin support vectors, [36] provide an overview or nonparallel hyperplane algorithms and [37,38] illustrate recent work on twin support vector machines.

The goal of this paper is to show how a primal SVM algorithm can be constructed that removes some of the caveats on other formulations of primal solutions. Most notably: we use kernel matrices that need only be positive semi-definite and suggest a procedure that overcomes the lack-of-sparseness shortage of least squares SVMs. Those points without violations are not presented in the solution. Our SVM in this paper is different from the least squares SVM [18] in that our SVM is derived merely in the context of RLS, while the least squares SVM were originally derived in the dual. Secondly, the least squares SVM replaces the inequality constraints with equalities while the solution proposed in this paper minimizes the violations without that replacement.

Our formulation begins as a regularized least squares (RLS) problem as was done by [28]. LSSVM only needs to solve a linear equation set rather than dealing with a quadratic programming problem, by using equality constraints instead of inequality ones and a least squares loss function, which greatly reduces the computational complexity [39]. The training set is partitioned into two parts: the one includes those points that are bounded by the two class-bounding-hyperplanes and another one includes those points that are unbounded. The later is referred to as the *support vector set* hereafter. Accordingly, the error vector is partitioned into two parts. The main contribution of this paper is the derivation of the optimal solution with the use of only some matrix operations for the partitioned error vector and merely in the context of the RLS. Instead of giving the linear combination form of the optimal solution in advance as was done by [28], our optimal solution is derived and can be expressed as a linear combination of the inner products of the support vectors with an input point. This approach not only overcomes the drawback of the invertability requirement of the kernel matrix, but also makes it natural to generalize to cases of SVMs with nonlinear kernels.

In Section 2, an SVM with linear kernel is formulated as an unconstrained minimization problem with the L2-loss.

The main details of our approach are presented in Sections 3 and 4. Firstly Section 3 expresses the solution to the problem as an equation with regard to the error vector and a set of indicator variables. Then in Section 4, it is shown how the error vector may be partitioned into two parts and how the solution to the linear SVM is expressed as a linear combination of inner products of the support vectors with an input point. How the solution may be generalized to cases of nonlinear kernels is discussed in comparison with standard SVM. In Section 5 an iterative algorithm to solve our SVM formulation is described, this is based on Cholesky decomposition (an approach also favored by [40]) and offers the potential to contribute when it comes to develop a wider population of problems with nonlinear kernels. The accuracy of the method is examined in Section 6 by comparing the algorithm's output with that from some existing SVM software packages. Section 7 draws a few conclusions about our algorithm.

2. Linear support vector machines

This section briefly reviews the SVM and introduces the notation to be used in the paper.

Given a data set of N point-label pairs $\{(\mathbf{x}_k, y_k), k = 1, \ldots, N\}$, referred to as the training set, each point is represented in a row vector $\mathbf{x}_k \in \Re^{1\times n}$, to which a label either $+1$ or -1, i.e.,