Neurocomputing 194 (2016) 218–226

Contents lists available at ScienceDirect

Neurocomputing

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

Quasi-newton method for L_P multiple kernel learning

Hu Qinghui [a,b], Wei Shiwei [b,*], Li Zhiyuan [b], Liu Xiaogang [a]

[a] Guangxi Colleges and Universities Key Laboratory Breeding Base of Robot and Welding Technology, Guilin University of Aerospace Technology, Guilin, China
[b] School of Computer Science and Engineering, Guilin University of Aerospace Technology, Guilin, China

ARTICLE INFO

Article history:
Received 3 November 2015
Received in revised form
14 January 2016
Accepted 29 January 2016
Available online 26 February 2016

Keywords:
Multiple kernel learning
Quasi-Newton method
Alternating optimization

ABSTRACT

Multiple kernel learning method has more advantages over the single one on the model's interpretability and generalization performance. The existing multiple kernel learning methods usually solve SVM in the dual which is equivalent to the primal optimization. Research shows solving in the primal achieves faster convergence rate than solving in the dual. This paper provides a novel L_P-norm($P>1$) constraint non-spare multiple kernel learning method which optimizes the objective function in the primal. Subgradient and Quasi-Newton approach are used to solve standard SVM which possesses superlinear convergence property and acquires inverse Hessian without computing a second derivative, leading to a preferable convergence speed. Alternating optimization method is used to solve SVM and to learn the base kernel weights. Experiments show that the proposed algorithm converges rapidly and that its efficiency compares favorably to other multiple kernel learning algorithms.

1. Introduction

Kernel method is an effective way to solve non-linear pattern recognition problems. For any kernel method, the data examples are first mapped to high dimensional Hilbert space H through a map $\phi: X \to H$, and then a liner decision boundary is found in that space. The map ϕ is computed implicitly through a kernel function $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, which is used to measure the similarity between data examples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. In the past several decades kernel methods have been widely used to solve machine learning problems such as classification [1,2], regression [3,4], density estimation [5] and subspace analysis [6]. For these tasks, the performance of the algorithm strongly depends on the data representation, which is implicitly chosen through the kernel function $k(.,.)$. Many kernel methods usually adopt a single predefined kernel function. However, in many real-world applications, it is usually not enough to use a single predefined kernel function because real data may come from multiple diverse sources or could be given in terms of different kinds of representations [7–13]. Multiple kernels based method has been extensively studied in the past few years [14–23]. A lot of applications have shown that using multiple kernels instead of a single one can effectively improve the interpretability and performances of the decision function and successfully resolve the challenges of speech recognition [24], anomaly detection [25] and protein–protein interaction extraction [26]. In such cases, we often consider that the kernel $k(.,.)$ is actually a convex combination of basis kernels

$$k(x_i, x_j) = \sum_{m=1}^{M} \theta_m k_m(x_i, x_j), \ \text{s.t.} \sum_{m=1}^{M} \theta_m = 1, \theta_m \geq 0$$

where M is the total number of basic kernels. θ_m is the combination weight corresponding to kernel $k_m(.,.)$. In such multiple kernel learning framework, data representation in the feature space is transformed into the selection of the basic kernels and weights.

Promoting training speed and predictive accuracy is the most active research directions in multiple kernel learning (MKL). Many effective methods have been investigated in recent years. The MKL problem has been introduced by Lanckriet et al. [14], which is exactly a constrained quadratic programming (QCQP) problem and becomes rapidly intractable as the number of data examples or basic kernels become large. Actually, the kernel learning problem is nonsmooth, which makes the direct application of gradient descent methods infeasible. Bach et al. [15] have considered a smoothed version of the problem so that gradient method such as SMO can be applied. Sonnenburg et al. [16] reformulate the MKL problem [15] as a semi-infinite linear program (SILP) and address that the problem can be solved by iteratively using existing single kernel classical SVM. Note that SILP may suffer from the instability of the solution of MKL. However, the approaches above employ mixed-norm regularization which results in slow convergence. Rakotomamonjy et al. [17] propose an algorithm, named SimpleMKL, reformulating the mixed-norm regularization in the MKL problem above as the weighted 2-norm regularization, which makes MKL more practical for large-scale learning. Then some improved methods have been proposed to solve this problem,

* Corresponding author. Tel.: +8613077692672.
E-mail address: swwei_2001@163.com (W. Shiwei).

http://dx.doi.org/10.1016/j.neucom.2016.01.079
0925-2312/

such as level-based optimization [18] and second-order Newton method [19].

In order to avoid overfitting, some regularization techniques are imposed on the weights. $L1$ norm MKL [17], for example, promotes the sparse solutions in terms of the kernel weights and thus it has preferable interpretability in kernel selection. Nevertheless, sparseness is not always beneficial in practice and sparse MKL is frequently observed to be outperformed by a regular SVM with an unweighted-sum kernel [20,21]. Recently, Kloft et al. [20,21] propose L_P-norm MKL method to extend the regular $L1$-norm MKL for arbitrary L_P-norm MKL with $P>1$. Compared with $L1$-norm MKL, L_P-norm can significantly improve the performance on diverse and relevant real-world datasets.

The existing methods mainly reformulate the MKL problem as a saddle point optimization problem which concentrates on solving in the dual. Primal optimization and the dual optimization are two equivalent ways to the same aim. Recent research shows solving in the primal achieves better convergence properties than solving in the dual [27–29]. Firstly, we can efficiently solve the primal problem without the need of the computations related to the variable switching. Secondly, when it comes to approximate solution, primal optimization is superior because it is more focused on minimizing what we are interested in, namely the primal objective.

In this paper we will show how to solve L_P-norm MKL in the primal with improved Quasi-Newton method [30]. Since Quasi-Newton method possesses superlinear convergence property and acquires inverse Hessian without computing a second derivative, the proposed algorithm obtains a faster convergence. Similar to other MKL methods, the alternating optimization algorithm is adopted to optimize classical SVM and the kernel weights respectively. Finally, we conduct a series of experiments to verify the efficiency and classification performance of our method.

The paper is organized as follows. Introduction of MKL problem is provided in Section 2. Section 3 describes the proposed MKL method in detail. Experiments are presented in Section 4 and some concluding remarks are given in the last section.

2. Multiple kernel learning problem

Support vector machine (SVM) is one of the most successful applications in kernel-based methods. Considering binary classification problem, the training data $D=\{(x_i,y_i)|\, i=1...n, x_i\in R^d\}$ where $y_i=\pm 1$ is the label of x_i. The data examples are first mapped to high dimensional Hilbert space H through a map ϕ, and then a linear decision boundary $\varphi(x)$ is found in that space, maximizing the margin between the two classes. Generally, the decision boundary is constructed by minimizing the following generic objective function:

$$Q(f,b)=\frac{\lambda}{2}\|f\|_H^2+C\sum_{i=1}^{n}\ell(y_i,f(x_i)+b) \tag{1}$$

where $\lambda(\lambda\geq 0)$ is a tuning parameter which is used to balance the effect of the two items on the right. The second one is the empirical risk of hypothesis f, and ℓ is a hinge loss function which is commonly used for binary classification with the following form:

$$\ell(y_i,f(x_i))=\max(0,1-y_i f(x_i)) \tag{2}$$

The decision boundary is defined as

$$\varphi(x)=f(x)+b \tag{3}$$

To simplify computation, real scalar b is omitted as it is only related to the position of boundary. Finally we can determine the class of the example, in which x_i belongs to $+1$ class if $f(x_i)>0$, otherwise it belongs to -1 class.

Considering a given feature map $\phi: X\to H$, where H corresponds to kernel function k such that $k(x_i,x_j)=\phi(x_i)^T\phi(x_j)$. K is a kernel matrix, $K_{(ij)}=k(x_i,x_j)$, and $K_{(i)}$ is the ith column of K. Based on the representation theorem [31], the decision boundary is as follows:

$$f(x)=\sum_{i=1}^{n}\alpha_i k(x_i,x) \tag{4}$$

Here we denote the α_i's as expansion coefficients instead of Lagrange multipliers α_i in standard SVM. Then we can transform the objective function (1) into the following form:

$$Q(\alpha)=\frac{\lambda}{2}\alpha^T K\alpha+C\sum_{i=1}^{n}\max(0,1-y_i K_i^T\alpha) \tag{5}$$

Considering $\phi_m: X\to H_m$, $m=1...M$ are M different feature maps corresponding to kernel function k_m. The aim of MKL is to learn the linear convex combination of basic kernels

$$k(x_i,x_j)=\sum_{m=1}^{M}\theta_m k_m(x_i,x_j) \tag{6}$$

The boundary $f(x)$ is defined as

$$f(x)=\sum_{m=1}^{M}f_m(x) \tag{7}$$

According to Eqs. (5) and (6), the final optimization objection function can be formulated as

$$\begin{aligned}Q(\alpha,\theta)=&\frac{\lambda}{2}\alpha^T\left(\sum_{m=1}^{M}\theta_m K_m\right)\alpha\\&+C\sum_{i=1}^{n}\max\left(0,1-y_i\left(\sum_{m=1}^{M}\theta_m K_{m(i)}\right)\alpha\right)\end{aligned} \tag{8}$$

$$\text{s.t.}\left\{\sum_{m=1}^{M}(\theta_m^P)\right\}^{1/P}\leq 1,\theta_m\geq 0,P>1$$

Here arbitrary L_P-norm constraint ($P>1$) is imposed on weights θ to achieve non-spare solutions.

3. Optimizing the MKL problem in primal

One general approach for solving problem $Q(\alpha,\theta)$ is to use alternating optimization algorithm applied in [17–21]. In the first step, $Q(\alpha,\theta)$ is optimized with respect to $\alpha$with θ being fixed. Then in the second step, the weights are updated to decrease the objective function $Q(\alpha,\theta)$ with α being fixed. The two steps are alternated until a predefined criterion is satisfied.

Fixed θ, Eq. (8) is exactly a nonsmooth function with respect to α. We adopt an improved quasi-Newton method, named subLBFGS [30] to solve this nonsmooth optimization problem. We first present some details of this optimization technique.

Quasi-Newton method is an effective method for solving nonlinear optimization problem with superlinear convergence property. An approximation is engaged to the inverse Hessian, which is built up on the basis of information gathered during the descent process, in place of true inverse required in Newton's method. There are several kinds of quasi-Newton methods according to different approximations to Hessian matrix, such as BFGS and LBGFS.

Definition 1. (Subgradient): Considering a convex function Q: $\Re^d\mapsto\Re$, Vector $g\in\Re^d$ is a subgradient of Q at point w if and only if $\forall w'\in\Re^d$, there is $Q(w')\geq Q(w)+(w'-w)^T g$.

Definition 2. (Subdifferential): The set of all subgradients of the convex function Q at point w is subdifferential, and is denoted by $\partial Q(w)$.

Based on the above definitions we can conclude that function Q is subdifferentiable at point w if the set of subgradients is not