Knowledge-Based Systems 65 (2014) 21–30

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Bagging-like metric learning for support vector regression

Peng-Cheng Zou *, Jiandong Wang, Songcan Chen *, Haiyan Chen

College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

ARTICLE INFO

Article history:
Received 26 December 2013
Received in revised form 28 February 2014
Accepted 2 April 2014
Available online 19 April 2014

Keywords:
Distance metric learning
Support vector regression
Ensemble learning
Bagging
Distance-based kernel

ABSTRACT

Metric plays an important role in machine learning and pattern recognition. Though many available off-the-shelf metrics can be selected to achieve some learning tasks at hand such as for k-nearest neighbor classification and k-means clustering, such a selection is not necessarily always appropriate due to its independence on data itself. It has been proved that a task-dependent metric learned from the given data can yield more beneficial learning performance. Inspired by such success, we focus on learning an embedded metric specially for support vector regression and present a corresponding learning algorithm termed as SVRML, which both minimizes the error on the validation dataset and simultaneously enforces the sparsity on the learned metric matrix. Further taking the learned metric (positive semi-definite matrix) as a base learner, we develop a bagging-like effective ensemble metric learning framework in which the resampling mechanism of original bagging is specially modified for SVRML. Experiments on various datasets demonstrate that our method outperforms the single and bagging-based ensemble metric learnings for support vector regression.

1. Introduction

Metric learning plays an important role in many learning tasks including k-nearest neighbor classification, k-means clustering and kernel-based algorithms such as support vector machines [1–5]. In recent years, many studies have demonstrated empirically and theoretically that it is often beneficial for a learning task to learn a metric from the given data, instead of using an off-the-shelf one such as Euclidean distance metric.

Depending on the availability of the given data, these methods roughly fall into two main categories: unsupervised metric learning and supervised metric learning. Each unsupervised metric learning method is essentially to learn a distance metric without supervised information [6,7]. While in supervised metric learning, more information about data such as label information is used to learn the metric and it is better to capture the idiosyncrasies of the data of interest [8,9]. We pay particular attention to the supervised methods in this paper.

Supervised distance metric learning can be further divided into task-independent and task-dependent metric learnings. The task-independent methods usually include two separated learning steps: in the first step, a metric is learned by solving an optimization problem with the supervised information. Then the second step uses the learned metric to solve a subsequent task. The classical Linear Discriminant Analysis (LDA) though as a dimensionality reduction method can also be viewed as a pseudo-metric learning method [10]. The metric learned by LDA can be used in many subsequent tasks such as k-nearest neighbor classification. In addition, MMC by Xing et al. learns a metric by minimizing the distances in equivalence constraints and maximizing the distances in inequivalence constraints. Then the metric learned by MMC is used in different clustering tasks [1].

Though the task-independent methods have used the supervised information when learning the metrics, such a two step method cannot guarantee the learned metric is optimal for the subsequent task. Therefore, a more desirable method is to learn the metric directly via incorporating the specific subsequent task, just as the task-dependent distance metric learning. It is similar to the feature selection problem that embedding methods can usually achieve better performance than filter methods [11]. The task-independent metric learning is corresponding to the filter method and the task-dependent metric learning is corresponding to the embedding method. One of the most representative works is Large Margin Nearest Neighbor (LMNN) [2], in which the learned metric is tailored specially for k-nearest neighbor classification and leads to significant improvement compared to k-nn with task-independent metrics. Several related researches have also been proposed, such as Neighborhood components analysis (NCA) [4], multi-task LMNN [12] and Non-linear LMNN [13], etc.

It should be noted that most of the existing task-dependent metric learning methods are designed for classification tasks especially k-nn. Similar to classification, regression is another important task in machine learning and its performance is also

* Corresponding authors. Tel.: +86 15850685790.
E-mail addresses: zou_pc@163.com (P.-C. Zou), s.chen@nuaa.edu.cn (S. Chen).

http://dx.doi.org/10.1016/j.knosys.2014.04.002
0950-7051/

highly dependent on the chosen metric. However, these methods designed for classification tasks cannot be used directly for regression tasks. Only few of the metric learning methods have been proposed specially for regression tasks so far. A typical one is MLKR [14] which learns a metric specially for kernel regression. Unfortunately, the improvement of regression performance achieved by MLKR is limited that it is still difficult to achieve a comparable performance with some sophisticated methods such as support vector regression [15] on many datasets.

To explore further the metric learning method for regression tasks, we consider learning a metric via incorporation of support vector regression (SVR) which is one of the most popular regression algorithms. Metric is also important for SVR especially with kernels. Typical kernels for SVR have no prior knowledge about the meaning of the features and are assumed to be isotropic. Therefore, we focus on learning an embedded metric in SVR to improve the regression performance. We propose a corresponding learning algorithm termed as SVRML, which minimizes the error on the validation set and enforces the sparsity on the learned metric matrix simultaneously. The learning process combines the Mahalanobis [16] metric learning with the training of SVR. More importantly, to make the metric learned by SVRML more effective, we propose a bagging-like ensemble metric learning framework. It extends the original bagging algorithm [17] in which a positive semi-definite matrix is taken as a base-learner rather than either classifier or regressor.

The proposed SVRML algorithm has the following desirable properties: (1) SVRML learns a sparse Mahalanobis metric which is capable of removing potential redundancy or noise in data. (2) SVRML can parallelly learn multiple base metrics by using a bagging-like ensemble metric learning framework and obtain an aggregated metric to achieve better generalization performance for SVR. (3) It is easy to implement and can be treated as an alternative feature selection method to provide a convenient way to pre-process the data automatically. The primary contributions of this work are therefore as follows: (1) We propose a task-dependent metric learning algorithm for SVR. (2) We develop an effective bagging-like ensemble metric learning framework in which the resampling mechanism of original bagging is specially modified for SVRML.

The rest of this paper is organized as follows: we provide an overview of the related work in Section 2. Section 3 explains how to learn an embedded metric for SVR. The bagging-like ensemble metric learning framework is discussed detailedly in Section 4. Experimental studies are shown in Section 5. Finally, we draw the conclusions and list our future works in Section 6.

2. Related works

Over the last decade, several task-dependent metric learning algorithms have been proposed [2,4,18,14]. However, only few of them are designed specially for regression tasks. Support vector regression which is very popular for regression tasks also depend heavily on the metric. As far as we know, our work is the first to combine metric learning with support vector regression. Our proposed method SVRML is also in the family of task-dependent distance metric learning.

Weinberger and Tesauro constructed a metric learning algorithm for kernel regression termed as MLKR [14] which learns a task-specific (pseudo-)metric over the input space where small distances between two vectors imply similar target values. This metric in MLKR is learned by directly minimizing the leave-one-out regression error. Similarly, Xu et al. [19] proposed a metric learning algorithm for support vector classification by minimizing the 0–1 classification error. Inspired by these work, we consider learning a metric for SVR by minimizing the regression error on a validation set. But one drawback of them is that they incline to overfit the validation data [8].

As a remedy, ensemble learning is an alternate method we can use to combine with the metric learning process, as ensemble learning is able to improve the generalization performance of learning systems [20]. Some ensemble learning methods such as boosting [21] have already been introduced into metric learning. For example, Shen et al. [22] proposed a boosting-based technique BoostMetric to learn a metric using trace-one rank-one matrices as weak learners. Chang [23] developed a metric base-learner specific to the boosting framework by improving a loss function iteratively. Mu et al. [24] proposed a local discriminative metrics ensemble learning algorithm. But none of them focus on regression tasks. To fill the gap, we propose a bagging-like ensemble framework designed specially for SVRML to improve the regression performance. Different from the existing methods such as BoostMetric which iteratively learns the base metrics, our framework retains the parallelism like bagging. In our framework, the resampling mechanism of original bagging is specially modified for SVRML to achieve better performance.

In addition to the above, our work is also inspired by the kernel-parameter selection methods for SVR. For example, Chang and Lin [25] derived various leave-one-out bounds for SVR parameter selection to improve the generalization performance. The kernel-parameter selection for SVR can be analyzed on the metric learning perspective that the adjusting of the inner product leads to different distance metrics. Different from choosing a single or a few kernel-parameters, our method optimizes the entire metric matrix and learns a nonlinear metric.

3. Metric learning for support vector regression (SVRML)

3.1. Support vector regression

Our method is based on L2-SVR [15], one of the most commonly used varieties of SVR. Given a set of training examples $\{x_i, y_i\}_{i=1}^{\ell}$ of size ℓ, where the input vector $x_i \in \mathbb{R}^d$, and the target value $y_i \in \mathbb{R}$, L2-SVR solves the primal problem:

$$\begin{aligned} &\min_{\mathrm{w},b,\xi,\xi^*} \quad \tfrac{1}{2}\mathrm{w}^T\mathrm{w} + \tfrac{C}{2}\sum_{i=1}^{\ell}\xi_i^2 + \tfrac{C}{2}\sum_{i=1}^{\ell}(\xi_i^*)^2 \\ &\text{s.t.} \quad -\varepsilon - \xi_i^* \leqslant \mathrm{w}^T\phi(\mathrm{x}_i) + b - y_i \leqslant \varepsilon + \xi_i, i = 1,\ldots,\ell. \end{aligned} \tag{1}$$

In order to solve the above problem effectively, practically we solve the dual problem of (1) instead:

$$\begin{aligned} &\min_{\alpha,\alpha*} \quad \tfrac{1}{2}(\alpha^* - \alpha)^T\tilde{K}(\alpha^* - \alpha) + \varepsilon\sum_{i=1}^{l}(\alpha_i^* + \alpha_i) - \sum_{i=1}^{l}y_i(\alpha_i^* - \alpha_i) \\ &\text{s.t.} \quad \textstyle\sum_{i=1}^{l}(\alpha_i^* - \alpha_i) = 0, i = 1,\ldots,\ell, \\ &\qquad \alpha_i, \alpha_i^* \geqslant 0, i = 1,\ldots,\ell, \end{aligned} \tag{2}$$

where $k(x_i, x_j) = \phi(x_i)^T\phi(x_j)$ is the kernel function. $\tilde{K} = K + I/C$ and $I^{d\times d}$ is an identity matrix. The final prediction function is

$$g(x) = \mathrm{w}^T\phi(x) + b = \sum_{i=1}^{\ell}(\alpha_i^* - \alpha_i)k(x_i, x) + b. \tag{3}$$

As the convenience of narrative, we do not distinguish L2-SVR from SVR in the following sections any longer. Many kernel functions are used for SVR. In fact, any function $k(\cdot,\cdot)$ can be used as a well-defined kernel if only it is positive semi-definite. In this paper, we use the popular kernel function RBF kernel uniformly due to its popularity and particularity that it depends on the distance function directly. The RBF kernel is defined as follows:

$$k(x_i, x_j) = \exp\left\{-d^2(x_i, x_j)\right\}, \tag{4}$$

where $d(\cdot,\cdot)$ is the distance metric of data. In the RBF kernel, it is commonly the squared Euclidean distance with a kernel width parameter $\sigma(\sigma > 0)$. When training the SVR, the prediction performance can be improved by choosing an effective parameter σ.