Neurocomputing 204 (2016) 183–188

Contents lists available at ScienceDirect

Neurocomputing

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

Hessian regularization by patch alignment framework

Weifeng Liu [a], Hongli Liu [a], Dapeng Tao [b,*]

[a] College of Information and Control Engineering, China University of Petroleum, Qingdao 266580, China
[b] School of Information Science and Engineering, Yunnan University, Kunming 650091, China

ARTICLE INFO

Article history:
Received 9 April 2015
Received in revised form
20 June 2015
Accepted 9 July 2015
Available online 9 April 2016

Keywords:
Semi-supervised learning
Hessian
Patch alignment
Least squares

ABSTRACT

In recent years, semi-supervised learning has played a key part in large-scale image management, where usually only a few images are labeled. To address this problem, many representative works have been reported, including transductive SVM, universum SVM, co-training and graph-based methods. The prominent method is the patch alignment framework, which unifies the traditional spectral analysis methods. In this paper, we propose Hessian regression based on the patch alignment framework. In particular, we construct a Hessian using the patch alignment framework and apply it to regression problems. To the best of our knowledge, there is no report on Hessian construction from the patch alignment viewpoint. Compared with the traditional Laplacian regularization, Hessian can better match the data and then leverage the performance. To validate the effectiveness of the proposed method, we conduct human face recognition experiments on a celebrity face dataset. The experimental results demonstrate the superiority of the proposed solution in human face classification.

1. Introduction

The ease of accessing large-scale images or videos accelerates the development of semi-supervised learning (SSL) [1–3]. SSL aims to solve classification problems with a small number of labeled samples and a large number of unlabeled samples. Hence, SSL has become one of the most attractive research topics and has been applied to many computer vision applications [4–6], including image annotation [7], image retrieval [8,9], image re-ranking [10,11], gait recognition [12] and social media retrieval [13].

Many prominent SSL algorithms have been reported. The representative works can be briefly divided into four groups: (1) transductive support vector machines (TSVM), (2) universum SVM, (3) co-training and (4) graph-based methods.

Joachims [14] first proposed the transductive support vector machines (TSVM) method, which could speed up computation by exploiting a label-switching-retraining procedure [15]. Subsequently, Li et al. [16] proposed the two-view TSVM method to solve Internet classification problems, significantly improving recognition rate and stability by making the most of multiple perspectives on both toy and real-life datasets. Recently, Kumar and Poornima [17] proposed an efficient period prediction system based on TSVM to more accurately read the ancient Tamil epigraphical scripts, which consist of image acquisition, binarization, preprocessing, feature extraction, segmentation and classification and prediction.

Zhang et al. [18] proposed the universum SVM method and improved the classification rate by simultaneously utilizing the labeled data, unlabeled data and universum data. Cherkassky and Dai [19] applied the random-averaging universum SVM method to real-life and synthetic conditions by incorporating a priori knowledge into the learning process, enabling it to avoid ill-posed and overfitting modeling problems in sparse high-dimensional data. Jiao et al. [20] improved the universum SVM classifier to solve the tongue image classification by selecting the in-between universum samples as the helpful irrelevant data and finding suitable parameters.

Co-training [21,22] methods train and iterate two classifiers based on two sufficient and independent features (views). Nigam and Ghani [23] explained in more detail why co-training algorithms are both discriminative in nature and robust to the assumptions of their embedded classifier with a large number of experiments on a real-world dataset. Then, the co-training method was proposed to settle the cross-lingual sentiment classification problem [24], in which English features and Chinese features were regarded as two independent views.

Graph-based methods [25,26] could spread the label sample to its neighbors until the whole dataset is stable. Mao et al. [27] proposed an $\ell 1$-graph-based semi-supervised learning to handle the object-tracking problem and obtained robust and accurate performance. Aittokallio and Schwikowski [28] outlined the recent research of graph-based methods for analyzing complex cellular networks in cell biology. Recently, Zhang

* Corresponding author.
E-mail address: dapeng.tao@gmail.com (D. Tao).

http://dx.doi.org/10.1016/j.neucom.2015.07.152
0925-2312/

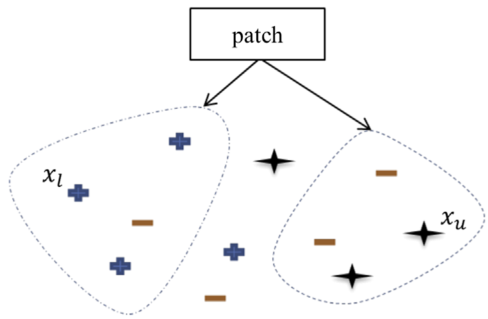

Fig. 1. Part optimization of a Hessian.

et al. [33] proposed a unified framework called patch alignment that provides a powerful analysis and development for graph regularization algorithms. Hong et al. [29] constructed the multi-view hypergraph Laplacian by using the patch alignment framework and demonstrated attractive results. Yu et al. [39] proposed a sparse patch alignment framework to settle image clustering, in which sparse representation is adopted to select a few neighbors of each data point in the patch optimization stage and the manifold is built during the whole alignment stage. Conversely, Hessian regularization has demonstrated promising performance for semi-supervised classifiers [7,30]. Therefore, it is essential to disclose the construction of a Hessian by using the patch alignment framework.

In this paper, we propose the derivation of a Hessian by using the patch alignment framework and implement Hessian regularization to solve semi-supervised regression problems. In particular, the process of Hessian construction by using the patch alignment framework consists of two steps: patch optimization and whole alignment. In the patch optimization stage, we construct the Hessian matrix of each patch by identifying neighbors, obtaining tangent coordinates and developing a Hessian estimator. In the whole alignment stage, the whole Hessian matrix could be obtained by iteration. Compared with traditional graph regularizations, the second-order Hessian energy can accurately describe the internal local geometric features of the data [7,30] and hence results in better performance.

The rest of this paper is organized as follows. Section 2 reviews the related work on Hessian LLE and the patch alignment framework. Section 3 describes the process of constructing a Hessian matrix in detail. Section 4 briefly introduces Hessian-regularized least squares for regression. Section 5 demonstrates experimental results on the celebrity face dataset for human face recognition. Section 6 concludes the paper.

2. Related work

In this section, we mainly review the related work on Hessian-based locally linear embedding (Hessian LLE) and the patch alignment framework.

2.1. Hessian LLE

Given scattered samples lying on a manifold M embedded in high-dimensional space, Hessian LLE [31,32] attempts the recovery of the underlying parameterization of the samples in an open, connected subset of low-dimensional space that is locally isometric to the original space.

Hessian LLE employs orthogonal coordinates on the tangent planes of M to define the Hessian of a function $f: M \to \mathcal{R}$, where $M \subset \mathcal{R}^D$ and D is the dimension of high-dimensional space. Suppose a point x has a neighborhood N_x, and $x' \in N_x$; then, define the function $g: G \to \mathcal{R}$ under the rule $g(\theta) = f(x')$, where G is a neighborhood of x in $\mathcal{R}^d$, $\theta = [\theta_1, \theta_2, \cdots, \theta_d]^T$ is the local coordinates, and d is the dimension of low-dimensional space, with $d \ll D$. Because the mapping $x' \to \theta$ is smooth, the Hessian of f at x in tangent coordinates can be defined as the ordinary Hessian of g and has the following expression

$$\left(H_f^{tan}(x)\right)_{ij} = \frac{\partial}{\partial\theta_i}\frac{\partial}{\partial\theta_j} g(\theta).$$

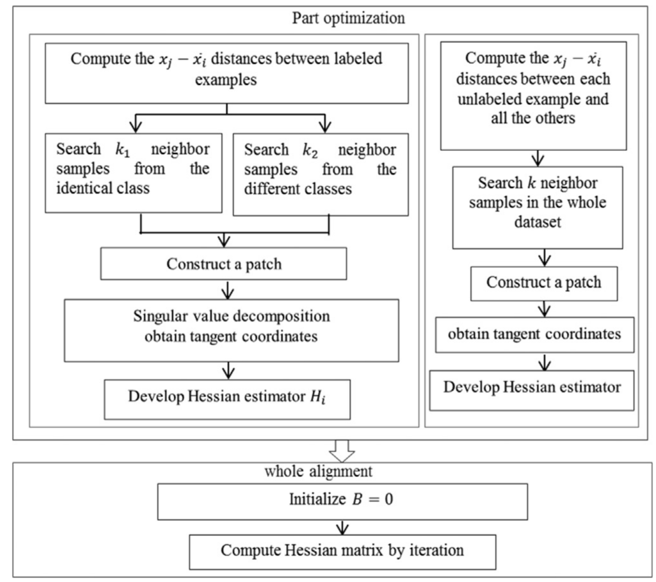

Fig. 2. Flowchart of Hessian regression.

In summary, at each point x, the Hessian LLE algorithm constructs a Hessian matrix using the tangent coordinates and differentiated f.

Finally, Hessian LLE obtains the locally isometric embedding of M by defining a quadratic functional $H(f) = \int_M \left\| H_f^{tan}(x) \right\|_F^2 dx$, where dx is a probability measure and $H(f)$ stands for the average "curviness" of f over the manifold M.

2.2. Patch alignment framework

Patch alignment [33] is a systematic framework used to understand the common properties and intrinsic differences in algorithms for dimensionality reduction. In particular, patch alignment is composed of two stages: part optimization and whole alignment. In the following, we briefly review the procedure of patch alignment.

(1) Patch optimization
Supposing that a data set $X = [x_1, x_2, \ldots, x_N] \in \mathcal{R}^{D \times N}$ contains N measurements x_i, $X_i = [x_i, x_{i_1}, \ldots, x_{i_k}] \in \mathcal{R}^{D \times (k+1)}$ stands for a patch, where x_i is selected at random and $x_{i_1}, \ldots, x_{i_k}$ is its k nearest neighbors. There exists a part mapping $f_i = X_i \to Y_i$ where $Y_i = \left[y_i, y_{i_1}, \ldots, y_{i_k}\right] \in \mathcal{R}^{d \times (k+1)}$. The part optimization is defined as $\arg\min_{Y_i} \mathrm{tr}\left(Y_i B_i Y_i^T\right)$, in which $B_i \in \mathcal{R}^{(k+1) \times (k+1)}$ encodes the objective function for the i^{th} patch.
(2) Whole alignment

Consider $Y_i = Y S_i$, where $Y = [y_1, y_2, \ldots, y_N]$, $S_i \in \mathcal{R}^{N \times (k+1)}$ is the selection matrix and $(S_i)_{pq} = \begin{cases} 1, p = F_i\{q\} \\ 0, else \end{cases}$, in which $F_i = \{i, i_1, \cdots, i_k\}$ is the set of indices.