Omega 59 (2016) 40–46

Contents lists available at ScienceDirect

Omega

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

Constrained subspace classifier for high dimensional datasets

Orestis P. Panagopoulos [a], Vijay Pappu [b], Petros Xanthopoulos [a,*], Panos M. Pardalos [b]

[a] Department of Industrial Engineering and Management Systems, University of Central Florida, 4000 Central Florida blvd., Orlando 32816, FL, United States
[b] Department of Industrial Engineering, University of Florida, 401 Weil Hall, Gainesville 32608, FL, United States

ARTICLE INFO

Article history:
Received 31 August 2014
Accepted 11 May 2015
Available online 10 June 2015

Keywords:
Constrained subspace classifier
High dimensional datasets
Principal angles
Local subspace classifier

ABSTRACT

Datasets with significantly larger number of features, compared to samples, pose a serious challenge in supervised learning. Such datasets arise in various areas including business analytics. In this paper, a new binary classification method called *constrained subspace classifier (CSC)* is proposed for such high dimensional datasets. CSC improves on an earlier proposed classification method called *local subspace classifier (LSC)* by accounting for the relative angle between subspaces while approximating the classes with individual subspaces. CSC is formulated as an optimization problem and can be solved by an efficient alternating optimization technique. Classification performance is tested in publicly available datasets. The improvement in classification accuracy over LSC shows the importance of considering the relative angle between the subspaces while approximating the classes. Additionally, CSC appears to be a robust classifier, compared to traditional two step methods that perform feature selection and classification in two distinct steps.

1. Introduction

High dimensional datasets are currently prevalent in many business applications. The methodical collection of every facet of the data has lead to a significant increase in its dimensionality. Examples include but are not limited to financial services [43], e-commerce [12] and marketing [32]. Other examples of datasets with a high number of features are shown in Table 1.

Classification tasks on high dimensional datasets pose significant challenges to the standard statistical methods and render many existing classification techniques impractical [22]. The generalization ability of many classification algorithms is compromised due to *curse of dimensionality* arising from high number of features of the input space [26]. Earlier studies have revealed the geometrical distortion that arises in high dimensional data spaces, where the ratio of distances between the farthest and nearest neighbors to a given target is almost equal to 1 for a wide variety of data distributions and distance functions [4]. Moreover, several statistical methods require knowing class covariances *a priori*. In the case that class covariances are unavailable, such estimates from sample data would be unreliable due to small sample sizes. One common approach to address the aforementioned challenges involves reducing the dimensionality of the dataset either by using feature extraction [29] and/or feature selection prior to classification [34,8].

Feature selection is usually performed in different ways through filter, wrapper, and embedded methods. Filter methods access features during a separate process prior to classification. Variables are given a score according to a filtering function and are ordered accordingly. Features with the lowest scores are discarded while the rest are used from the classifier. Hypothesis testing and statistic tests such as *t*-test have also been used as filtering procedures [17]. Wrapper methods on the other hand use the classifier structure itself to evaluate the importance of features based on the idea that the classifier can provide a better estimate of accuracy than a separate independent process [6]. The main drawback of wrapper methods is that increased computational power is often required since the classification process has to be repeated for each feature set considered. Metaheuristics used for feature selection can also be classified as wrapper methods [40,30,47]. Embedded methods perform feature selection in a way so that the classification algorithm is executed while variables are evaluated and selected. Examples include the weighting of features in support vector machines [18], where the authors developed the SVM method of recursive feature elimination for feature selection, and the use of random forests for feature evaluation [21]. In the later, feature elimination occurs for the attributes with the lowest raw importance score.

Feature extraction techniques transform the input data into a set of *meta*-features that extract the relevant information from the input data for classification. One popular technique called *principal component analysis (PCA)* finds a set of linearly uncorrelated variables called *principal components* from a set of observations of possibly correlated variables [23,36]. PCA removes redundancy by transforming the data from a higher dimensional space into an orthogonal lower dimensional space. This transformation is

[*] Corresponding author.
E-mail addresses: orepana@gmail.com (O.P. Panagopoulos), psnvijay.iitm@gmail.com (V. Pappu), petrosx@ucf.edu (P. Xanthopoulos), pardalos@ufl.edu (P.M. Pardalos).

http://dx.doi.org/10.1016/j.omega.2015.05.009
0305-0483/© 2015 Elsevier Ltd. All rights reserved.

Table 1
Examples of high dimensional datasets.

Dataset	Reference
Customer relationship management data	[39]
Covariation information of stocks	[7]
Text datasets for classification	[20]
Data collected from surveys	[2]
Netflix dataset	[3]
MRI data	[24]
Mass spectroscopy data	[14]

performed in a way that the first principal component captures as much variation in the data as possible, and each succeeding component accounts for a decreasing amount of variance [42]. The number of retained principal components is usually less than or equal to the number of original variables and are determined using several criteria like the eigenvalue-one criterion, scree test and proportion of variance accounted for.

The aforementioned dimensionality reduction techniques decrease the complexity of the classification model and attempt to improve the classification performance [34]. The choice of the dimensionality reduction technique depends on the nature (e.g. level of correlation, presence of outliers) of the data that is used for classification.

Local subspace classifier (LSC) utilizes PCA to perform classification. During the training phase, a lower dimensional subspace is found for each class that approximates the data [27]. In the testing phase, a new data point is classified by calculating the distance of the point to each subspace and choosing the class with minimal distance. Although LSC is simple and relatively easy to implement, it has its own limitations. LSC finds the subspaces for each class *separately* without the *knowledge* of the presence of the other class. While each subspace approximates the data well, however these projections may not be *ideal* from a classification perspective. In this paper, we construct another classifier called *constrained subspace classifier* (CSC) which expands LSC by including the relative orientation of the subspaces of two classes in an integrated optimization model. LSC formulation is modified to include the relative angle between the subspaces and is solved efficiently using alternating optimization techniques. The performance of CSC on publicly available datasets is evaluated and compared with LSC and other classifiers.

The remainder of the paper is organized as follows. Section 2 gives an introduction to LSC and Section 3 introduces the CSC. In Section 4 we demonstrate a first comparison on a toy dataset whereas in Section 5 we present the computational experiment on six real datasets along with their discussion as well as we provide the comparative computational results for CSC against support vector machine (SVM), PCA/SVM and Naive Bayes classifier. Lastly, in Section 6 we discus potential future extensions of this algorithm.

2. Local subspace classifier

Consider a binary classification problem. Let the matrices $\mathcal{X}_1 \in \mathbb{R}^{p\times m}$ and $\mathcal{X}_2 \in \mathbb{R}^{p\times l}$ be given, whose columns represent the training examples of two classes $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively. The number of samples in $\mathcal{C}_1$ and $\mathcal{C}_2$ are given by m and n respectively. The number of features is given by p. Local subspace classifier attempts to find two subspaces separately, one for each class that *best* approximates the data. Let $\boldsymbol{U}_1 = [\boldsymbol{u}_1^{(1)}, \boldsymbol{u}_2^{(1)}, \ldots, \boldsymbol{u}_k^{(1)}]_{p\times k}$ and $\boldsymbol{U}_2 = [\boldsymbol{u}_1^{(2)}, \boldsymbol{u}_2^{(2)}, \ldots, \boldsymbol{u}_k^{(2)}]_{p\times k}$ represent orthonormal bases of two k-dimensional linear subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$ that approximate classes $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively. We assume the dimensionality of subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$ to be same and equal to k without loss of generality. $\mathcal{S}_1$ and $\mathcal{S}_2$ attempt to capture *maximal* variance in classes $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively by optimizing the following optimization problems:

$$\begin{aligned} &\underset{\boldsymbol{U}_1 \in \mathbb{R}^{p\times k}}{\text{maximize}} \quad \mathrm{tr}(\boldsymbol{U}_1^T \mathcal{X}_1 \mathcal{X}_1^T \boldsymbol{U}_1) \\ &\text{subject to} \quad \boldsymbol{U}_1^T \boldsymbol{U}_1 = \boldsymbol{I}_k \end{aligned} \tag{1}$$

where $\boldsymbol{I}_k$ is the identity matrix of size k.

The solution to the optimization problem (1) is given by eigenvectors corresponding to the k largest eigenvalues of matrix $\mathcal{X}_1\mathcal{X}_1^T$ [15]. Similarly, the following optimization problem is solved to obtain the orthonormal basis $\boldsymbol{U}_2$ representing $\mathcal{S}_2$:

$$\begin{aligned} &\underset{\boldsymbol{U}_2 \in \mathbb{R}^{p\times k}}{\text{maximize}} \quad \mathrm{tr}(\boldsymbol{U}_2^T \mathcal{X}_2 \mathcal{X}_2^T \boldsymbol{U}_2) \\ &\text{subject to} \quad \boldsymbol{U}_2^T \boldsymbol{U}_2 = \boldsymbol{I}_k \end{aligned} \tag{2}$$

The orthonormal basis $\boldsymbol{U}_2$ is obtained by choosing eigenvectors corresponding to the k largest eigenvalues of matrix $\mathcal{X}_2\mathcal{X}_2^T$. A new point $\boldsymbol{x}$ is classified by computing its distance from subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$:

$$\mathrm{dist}(\boldsymbol{x}, \mathcal{S}_i) = \mathrm{tr}(\boldsymbol{U}_i^T \boldsymbol{x}\boldsymbol{x}^T \boldsymbol{U}_i) \tag{3}$$

and the class of $\boldsymbol{x}$ is determined as

$$\mathrm{class}(\boldsymbol{x}) = \arg \min_{i \in \{1,2\}} \{\mathrm{dist}(\boldsymbol{x}, \mathcal{S}_i)\} \tag{4}$$

Though the subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$ approximate the classes well, these projections may not be *ideal* for classification tasks as each of them are obtained *without* the knowledge of another class/subspace. Hence, from a classification performance perspective, these subspaces may not be the *best* projections for the classes. In order to account for the presence of another subspace, we consider the relative orientation of the subspaces.

3. Constrained subspace classifier

Constrained subspace classifier finds two subspaces *simultaneously*, one for each class, such that each subspace accounts for maximal variance in the data in the *presence* of the other class/subspace. Thus, CSC allows for a *tradeoff* between approximating the classes well and the relative orientation among the subspaces. The relative orientation between subspaces is generally defined as principal angles [19]. We briefly review principal angles between subspaces below, which are further utilized to modify the formulation of LSC to include the relative orientation among the subspaces.

Definition 1. Let $\boldsymbol{U}_1 \in \mathbb{R}^{p\times k}$ and $\boldsymbol{U}_2 \in \mathbb{R}^{p\times k}$ be two orthonormal matrices spanning subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$. The principal angles $0 \le \theta_1 \le \theta_2 \le \theta_3 \le \cdots \le \theta_k \le \pi/2$ between subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$, are defined recursively by

$$\begin{aligned} &\cos\theta_i = \max_{\boldsymbol{x}_m \in \mathcal{S}_1} \max_{\boldsymbol{y}_n \in \mathcal{S}_2} \quad \boldsymbol{x}_m^T \boldsymbol{y}_n \\ &\text{subject to} \quad \boldsymbol{x}_m^\top \boldsymbol{x}_n = 1, \quad \boldsymbol{y}_m^\top \boldsymbol{y}_n = 1 \quad \text{for } m = n \\ &\qquad \boldsymbol{x}_m^\top \boldsymbol{x}_n = 0, \quad \boldsymbol{y}_m^\top \boldsymbol{y}_n = 0 \quad \text{for } m \ne n \\ &\qquad \forall m, n = 1, 2, \ldots, k. \end{aligned} \tag{5}$$

where $\boldsymbol{x}_m$ and $\boldsymbol{y}_n$ are the column vectors of $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ respectively. Intuitively, the first principal angle θ_1 is the smallest angle between all pairs of unit vectors in the first and second subspaces. The rest of the principal angles are similarly defined.

Theorem 1. *Let $U_1 \in \mathbb{R}^{p\times k}$ and $U_2 \in \mathbb{R}^{p\times k}$ be rectangular matrices whose column vectors span the subspaces $S_1 \in \mathbb{R}^k$ and $S_2 \in \mathbb{R}^k$ respectively. Let $M = U_1^\top U_2 \in \mathbb{R}^{k\times k}$, using singular value decomposition we can express M by*

$$M = YCZ^\top \tag{6}$$

where $Y^\top Y = I_k$, $Z^\top Z = I_k$ and $C = diag(\sigma_1, \sigma_2, \ldots, \sigma_k)$.