Pattern Recognition 55 (2016) 231–246

Contents lists available at ScienceDirect

Pattern Recognition

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

# Multi-class Support Vector Machine classifiers using intrinsic and penalty graphs

Alexandros Iosifidis*, Moncef Gabbouj

*Department of Signal Processing, Tampere University of Technology, P.O. Box 553, FIN-33720 Tampere, Finland*

ARTICLE INFO

*Article history:*
Received 6 July 2015
Received in revised form
25 January 2016
Accepted 4 February 2016
Available online 12 February 2016

*Keywords:*
Multi-class classification
Maximum margin classification
Support Vector Machine
Graph Embedding

ABSTRACT

In this paper, a new multi-class classification framework incorporating geometric data relationships described in both intrinsic and penalty graphs in multi-class Support Vector Machine is proposed. Direct solutions are derived for the proposed optimization problem in both the input and arbitrary-dimensional Hilbert spaces for linear and non-linear multi-class classification, respectively. In addition, it is shown that the proposed approach constitutes a general framework for SVM-based multi-class classification exploiting geometric data relationships, which includes several SVM-based classification schemes as special cases. The power of the proposed approach is demonstrated in the problem of human action recognition in unconstrained environments, as well as in facial image and standard classification problems. Experiments indicate that by exploiting geometric data relationships described in both intrinsic and penalty graphs the SVM classification performance can be enhanced.



## 1. Introduction

Support Vector Machine (SVM) [43] is a standard classification technique that has been shown to provide state-of-the-art performance in many classification problems. In addition to their good generalization ability, the popularity of SVMs is a consequence of their ability to represent the classification problem at hand as a quadratic convex optimization problem, leading to a global optimal solution, while non-linear decision functions can be learned by exploiting the well-known kernel trick [31,39,25,36].

The standard SVM is a binary classifier that learns a hyperplane separating two classes with maximum margin. While the maximum margin property of the SVM classifier is very powerful, it has been shown that enhanced performance can be achieved by incorporating geometric data information in the SVM optimization process. This is due to the fact that, by exploiting such additional information, the classifier takes into account geometric properties of the classes in addition to the position of the support vectors. Specifically, it has been shown that the incorporation of the intra-class variance information (described by the corresponding within-class scatter matrix) in the SVM optimization problem leads to enhanced performance in frontal face verification [42], as well as in various other classification problems, e.g. gender determination, eye glass detection and neutral facial expression recognition and standard classification problems [50,45,51,19]. In addition, it has been shown that the exploitation of intrinsic graph structures defined under the Graph Embedding framework [49] further enhances the performance of the resulting classifier [1]. Graph Embedding, is a general framework which can be exploited in order to define Subspace Learning techniques, such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Marginal Discriminant Analysis (MDA) and Local Fisher Discriminant Analysis (LFDA). This is achieved by defining an intrinsic graph expressing properties of the data that are subject to minimization (e.g. the within-class variance in the case of LDA) and a penalty graph expressing properties of the data that are subject to maximization (i.e. the between-class variance in the case of LDA).

In all the above mentioned SVM-based classification approaches, the One-Versus-Rest (OVR) or One-Versus-One (OVO) binary classifier combination schemes are employed in order to perform multi-class classification [35]. That is, for a classification problem formed by data belonging to $K$ classes, multiple[1] binary classifiers are trained, each of which solves a sub-problem of the original multi-class classification problem. In the test phase, a test sample is introduced to all the binary classifiers and their responses are combined in order to provide the final classification result [30]. Such an approach inherently sets the assumption that the

* Corresponding author. Tel.: +358 408 267 830, +306 974 027 924.
*E-mail addresses:* alexandros.iosifidis@tut.fi (A. Iosifidis), moncef.gabbouj@tut.fi (M. Gabbouj).

[1] The number of binary classifiers is equal to $K$ for the OVR and $\frac{K(K-1)}{2}$ for the OVO combination schemes. For the standard SVM formulation, combined OVR and OVO classification schemes have also been used [20,27], where a model formed by $K \leq M \leq \frac{K(K-1)}{2}$ binary classifiers is created.

http://dx.doi.org/10.1016/j.patcog.2016.02.002
0031-3203/

classification problems solved by the various binary classifiers are independent. In order to overcome this assumption, multi-class SVM formulations and their counterparts incorporating the within-class variance of the training data in a multi-class SVM formulation, have been proposed in [47,48,21,22,3].

In [21,22], only the case where the within-class variance of the training data is exploited for SVM-based multi-class classification is proposed. In addition, the non-linear extension of the proposed classifiers is achieved by applying a two-step process. Specifically, in [21], the training data are projected to the range of the within-class scatter matrix first and standard multi-class SVM classification is applied on the projected data [47,48]. In [22], it is shown that the kernel formulation of the proposed multi-class classifier is equivalent to applying kernel PCA on the training data, followed by the application of the proposed linear classifier exploiting the within-class variance of the projected training data.

In this paper, we propose a new optimization problem for SVM-based multi-class classification exploiting geometric information of the training data. To do this, we incorporate geometric data information described in both intrinsic and penalty graphs as designed in the context of the Graph Embedding framework. Compared to the solution proposed in [1] exploiting only intrinsic graphs for binary classification problems, the proposed classifier exploits general graph structures expressing both intrinsic (expressing data relationships to be minimized) and penalty (expressing data relationships to be maximized) criteria, under a multi-class SVM formulation. Compared to the solutions proposed in [21,22] exploiting only the within-class variance of the training data for multi-class classification, the proposed approach is able to exploit more generic intrinsic graph structures, as well as penalty ones. In addition, we propose a direct solution for the optimization problem solved for non-linear data classification. Finally we show that the proposed approach constitutes a general framework for SVM-based multi-class classification exploiting geometric information of the training data and that the methods in [42,50,1,35,21,22,45] are special cases of the proposed approach.

We apply the proposed method in facial image and standard classification problems and to the problem of human action recognition in unconstrained environments, usually also referred to as 'action recognition in the wild'. A lot of research has been conducted in this area during the last decade. The interested reader may refer to [26]. Perhaps the most well studied and successful approach for action representation is based on the Bag of Words (BoWs) model [11]. According to this model, each action video is represented by a vector obtained by applying quantization on the features describing it and using a set of feature prototypes forming the so-called codebook. This codebook is usually determined by clustering the features describing training action videos, while discriminative codebook construction methods have also been recently proposed [12]. This approach has been tested in most of the existing benchmark datasets and its efficacy has been proven, since it provides state-of-the-art performance in most cases. We follow the state-of-the-art approach [46] describing videos depicting actions by using five descriptor types, i.e. Histogram of Oriented Gradient (HOG), Histogram of Optical Flow (HOF), Motion Boundary Histogram along the direction x (MBHx), Motion Boundary Histogram along the direction y (MBHy) and (normalized) Trajectory, evaluated on the trajectories of densely sampled interest points. Such an action description has been evaluated in most of the existing benchmark datasets, where it has been shown that it provides satisfactory performance (state-of-the-art in most cases).

In summary, the contributions of the paper are as follows:

- A new optimization problem for SVM-based multi-class classification is proposed that exploits geometric data relationships described in both intrinsic and penalty graphs.
- A new direct solution is proposed for the optimization problem used to determine non-linear decision functions for multi-class classification.
- The proposed approach is shown to constitute a general framework for SVM-based classification exploiting geometric data information that includes several SVM-based classifiers as special cases.

The reminder of the paper is organized as follows. We provide an overview of related previous work in Section 2. The proposed method is described in detail in Section 3. Experiments conducted in order to evaluate its performance are described in Section 4. Conclusions are drawn in Section 5.

## 2. Previous work

Let us denote by $\{\mathbf{x}_i, l_i\}$, $i=1,\ldots,N$ a set of $D$-dimensional vectors $\mathbf{x}_i$ and the corresponding class labels $l_i \in \{1,\ldots,K\}$. We would like to train a multi-class classification scheme that is able to classify a test vector $\mathbf{x}_t \in \mathbb{R}^D$ to one of the $K$ classes.

### 2.1. Binary SVM classifier

As previously described, multi-class classification can be achieved by training multiple binary classifiers [30]. Let us define the binary labels $y_i \in \{-1,1\}$ determining whether the vectors $\mathbf{x}_i$ belong to the positive or negative class of the binary classification problem at hand. In SVM, the optimal separating hyperplane is the one that separates the two classes with maximum margin. The SVM optimization problem is defined as

$$\min_{\mathbf{w},b} \frac{1}{2}\mathbf{w}^T\mathbf{w} + c\sum_{i=1}^{N}\xi_i, \tag{1}$$

subject to the constraints:

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i=1,\ldots,N, \tag{2}$$

where $\mathbf{w} \in \mathbb{R}^D$ is the vector defining the separating hyperplane, $b$ represents the offset of the hyperplane from the origin, $\xi_i$, $i=1,\ldots,N$ are the so-called slack variables and $c>0$ is a regularization parameter denoting the importance of the training error in the optimization problem. The solution of the above-described optimization problem is a quadratic convex optimization problem of the form:

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j, \tag{3}$$

subject to the constraint $0 \leq \alpha_i \leq c$, $i=1,\ldots,N$. $\boldsymbol{\alpha} \in \mathbb{R}^N$ is a vector containing the Lagrange multipliers $\alpha_i$, $i=1,\ldots,N$.

In order to derive non-linear decision functions, the so-called kernel trick is exploited. That is, it is assumed that the training vectors $\mathbf{x}_i$ are non-linearly mapped to an arbitrary-dimensional feature space $\mathcal{F}$ (usually having the properties of Hilbert spaces [31,39]) by employing a function $\phi(\cdot): \mathbf{x}_i \in \mathbb{R}^D \rightarrow \phi(\mathbf{x}_i) \in \mathcal{F}$. In $\mathcal{F}$, dot products between training vectors are defined by a kernel function $\kappa(\cdot,\cdot)$ and are stored in the so-called kernel matrix $\mathbf{K} \in \mathbb{R}^{N\times N}$. Thus, (3) can be given in the form:

$$\max_{\boldsymbol{\alpha}} \mathbf{1}^T\boldsymbol{\alpha} - \frac{1}{2}(\boldsymbol{\alpha}\circ\mathbf{y})^T\mathbf{K}(\boldsymbol{\alpha}\circ\mathbf{y}), \tag{4}$$

where $\mathbf{y} \in \mathbb{R}^N$ is a vector containing the binary labels $y_i$, $i=1,\ldots,N$ and $\circ$ denotes the Hadamard (element-wise) product operator.