TWO MULTI-CLASS APPROACHES FOR REDUCED MASSIVE DATA SETS USING CORE SETS

LACHACHI Nour-Eddine, ADLA Abdelkader
Computer Science Department - Oran University - Algeria
E-Mail : {Lach_Nour,AekAdla}@Yahoo.fr

ABSTRACT

Current conventional approaches to biometric development for Text Independent Speaker Identification and Verification system present serious challenges in computational complexity and time variability. In this paper, we develop two approaches using SVMs which can be reduced to Minimal Enclosing Ball (MEB) problems in a feature space to produce simple data structures optimally matched to the input demands of different background of systems as UBM architectures in speaker recognition and identification systems. For this, we explore a technique to learn Support Vector Models (SVMs) when training data is partitioned among several data sources. Computation of such SVMs can be efficiently achieved by finding a corset for the image of the data in the feature space cited above.

Keywords: Quadratique Programming (QP), Support Vector Machines (SVMs), Minimum Enclosing Ball (MEB), core set, kernel methods.

1. INTRODUCTION

Support Vector Machines (SVMs) [1], are currently one of the most effective techniques to approach classification and other data analysis problems, improving more traditional techniques like decision trees and neural networks in several applications. But normal SVM algorithms are not suitable for classification of large data sets because of high training complexity. Traditional [1] and even modern methods [2][3] to obtain SVMs require that the set of examples be completely available in a common place to access them an arbitrary number of times in order to converge to an optimal decision function.

Here, we explore a technique to learn Support Vector Machines (SVMs) when training data is partitioned among several data sources. The basic idea is to consider SVMs which can be reduced to Minimal Enclosing Ball (MEB) problems in a feature space. Computation of such SVMs can be efficiently achieved by finding a core set for the image of the data in the feature space. However, in the standard technique for scaling up a two class SVM to handle large data sets and can only be used with certain kernel functions and kernel methods.

Kernel methods, such as the support vector machine (SVM), are often formulated as quadratic programming (QP) problems. However, scaling up these QPs is a major stumbling block in applying kernel methods on very large data sets, and placement of the naive method for finding the QP solutions is highly desirable. Many kernel methods can be equivalently formulated as minimum enclosing ball (MEB) problems in computational geometry. Then, by adopting an efficient approximate MEB algorithm, we can obtain provably approximately optimal solutions with the idea of core sets.

The goal of this paper is to develop an alternative method based on a recently proposed equivalence between SVMs and Minimal Enclosing Ball (MEB) problems from which important improvements on training efficiency has been reported [4][5] for large-scale datasets. We directly focus on multi-class problems exploring two methods to extend binary SVMs to the multi-category setting which preserve the equivalence between the model and MEBs. Algorithms to compute SVMs based on the MEB equivalence are based on the greedy computation of a core-set, a typically small subset of the data which provides the same MEB as the full dataset. Then, we formulate new multiclass SVM problem using core sets for reduce large data sets which can be considered optimally matched to the input demands of different background architectures of speaker systems. The core idea of these two approaches is to adopt multiclass SVMs formulation and Minimal Enclosing Ball to reduce data set without influence data noise.

This paper is presented as follows: section 2 introduces SVMs and Minimal Enclosing Balls. Section3 concerns Multiclass extension. Sections 4 present the equivalence between L2-SVMs and MEB Problems in Multiclass approach. Section 5 formulates our two algorithms. Section 6 provides the experimental methodology and finally section 7 summarizes the conclusions.

2. SVMs AND MINIMAL ENCLOSING BALLS (MEB)

Given a training data set $\mathcal{D} = \{(\mathbf{c_i}, y_i)\} \text{i=1 to n}$ where $\mathbf{c_i} \in \mathbb{R}^d$ and $y_i \in \{\pm1\}^n$. Support Vector Machines (SVMs) [1] address the problem of binary classification by building a hyperplane to represent the boundary between the two classes. This hyperplane $f(c) = (w^T \mathbf{c} + b)$ is built in a feature space $\tilde{\mathcal{Z}} = \mathcal{Z}(\mathcal{X})$ implicitly induced from $\mathcal{X}$ by means of a kernel function $\phi$(34,31),(774,980) which computes the dot products $\tilde{\mathcal{Z}} \phi(c_i) = \phi(c_i)^T \phi(c_{j}) \text{ in } \tilde{\mathcal{Z}} \text{ directly on } \mathcal{X}$. The so called L2-SVM chooses the separating hyperplane $f(\mathbf{c})$ by solving the following quadratic program:
\[
\min_{\alpha} \sum \alpha_i y_i b_i - \rho \\
\text{s.t. } \alpha_i \geq 0, \sum \alpha_i = 1
\]  

(1)

If \( y_i \alpha_i > \rho > 0 \), \( z_i \) is correctly classified. Variable \( \rho \), called the margin, is hence a measure of classification confidence and the slacks \( \varepsilon_i \) a measure of the amount of confidence violation. Variable \( \rho \) is thus maximized in the objective function and slacks penalized using a hyperparameter \( C \). The term \( \|w\|^2 + b^2 \), on the other hand, encourages sparsity or simplicity of the solution.

After introducing Lagrange multipliers, it can be shown that the latter problem is equivalent to solve

\[
\min_{\alpha} \sum \alpha_i y_i b_i - \rho \\
\text{s.t. } \alpha_i \geq 0, \sum \alpha_i = 1
\]  

(2)

where \( H \) is the Kronecker delta function and \( \delta_{ij} \) implements the dot-product \( \langle x_i, x_j \rangle \).

Its optimal value is determined using model selection techniques and depends on the degree of noise and overlap among the classes [6]. From the Karush-Kuhn-Tucker (KKT) conditions, the hyperplane parameters is recovered as \( w = \sum \alpha_i y_i x_i \) and \( b = \sum \alpha_i z_i \). Note that the solution finally depends only on the examples for which \( \alpha_i > 0 \) which are called the support vectors.

Although the L2-SVM is slightly different from the original SVM formulation, both models obtain comparable performance in practice [5]. As shown in [5], the main appeal of the L2 implementation is that it supports a convenient reduction to a minimal enclosing ball (MEB) problem when the kernel used in the SVM is normalized, that is, \( \langle x_i, x_j \rangle = \delta \forall x \in \mathcal{X} \), where \( \delta \) is a constant. The advantage of this equivalence is that the Bádoiu and Clarkson algorithm [7] can efficiently approximate the solution of a MEB problem with any degree of accuracy.

To simplify the notation let us denote the pair \((x_i, y_i)\) as \( z_i \). Now the training data set can be denoted as \( Z = \{z_i\}_{i=1}^m \). Let \( Z \) a space equipped with a dot product \( \langle z_i, z_j \rangle \) that corresponding to norm \( \|z\|^2 = z^T z \). We define the ball \( \mathcal{B}(c, R) \) of center \( c \in Z \) and radius \( R \) in \( Z \) as the subset of points \( z \in Z \) for which \( \|z - c\|^2 \leq R^2 \). The minimal-enclosing ball [5] of a set of points \( Z = \{z_i \in Z\} \) in \( Z \) is in turn the ball \( \mathcal{B}(c^*, R^*) \) of smallest radius that contains \( Z \), that is, the solution to the following optimization problem.

\[
\min_{c,R} \frac{R^2}{2} \langle z_i - c, z_i - c \rangle \quad \forall z_i \in Z
\]  

(3)

After introducing Lagrange multipliers we obtain from the optimality conditions the following dual problem

\[
\min_{\alpha} \sum \alpha_i y_i b_i - \rho \\
\text{s.t. } \alpha_i \geq 0, \sum \alpha_i = 1 \quad \forall i \in I
\]  

(4)

in matrix form is as

\[
\min_{\alpha} \alpha^T K \alpha - \alpha^T \text{diag}(K) \\
\text{s.t. } 0 \leq \alpha, \; \alpha^T 1 = 1
\]  

(5)

where \( \alpha = [\alpha_1, \ldots, \alpha_m]^T \) is the vector of Lagrange multipliers, \( 0 = [0, \ldots, 0]^T \), \( 1 = [1, \ldots, 1]^T \) and \( K_{xy} = < z_x, z_y > \) is the corresponding kernel matrix. But if we consider that \( \sum \alpha_i y_i b_i - \rho = \alpha^T \text{diag}(K) \) a constant as is supposed in L2-SVM formulation above, we can drop it from the dual objective in (3), we obtain a simpler QP problem

\[
\min_{\alpha} \frac{1}{2} \alpha^T K \alpha - \alpha^T 1 \\
\text{s.t. } 0 \leq \alpha, \; \alpha^T 1 = 1
\]  

(6)

As well-known, this is a QP problem. In [5], it show that the primal variables \( \alpha \) and \( R \) can be recovered from the optimal

\[
\alpha = \sum \alpha_i y_i z_i, \; R = \sum \alpha_i z_i z_i^T z_i = \sqrt{\alpha^T \text{diag}(K)}
\]

The algorithm of Bádoiu and Clarkson [7] approximate the solution to this problem exploits the ideas of core-set and \( \varepsilon \)-approximation to the minimal enclosing ball of a set of points. A set \( Z_\varepsilon \subseteq Z \) will be called a core-set of \( Z \) if the minimal enclosing ball computed over \( Z_\varepsilon \) is equivalent to the minimal enclosing ball considering all the points in \( Z \). A ball \( \mathcal{B}(c, R) \) is said a \( \varepsilon \)-approximation to the minimal enclosing ball \( \mathcal{B}(c^*, R^*) \) of \( Z \) if \( R \leq R^* \) and it contains \( Z_\varepsilon \) up to precision \( \varepsilon \), that is \( Z \subseteq \mathcal{B}(c, (1 + \varepsilon)R) \). Consequently, a set \( Z_\varepsilon \) is called a \( \varepsilon \)-core-set if the minimal enclosing ball of \( Z_\varepsilon \) is a \( \varepsilon \)-approximation to \( \mathcal{B}(c^*, R^*) \).

![Figure 1](image1.png)

**Figure 1.** The inner circle is the MEB of the set of squares and its \((1 + \varepsilon)\) expansion (the outer circle) covers all the points. The set of squares is thus a core-set.

Here we present the most usual version of the algorithm [7]

**Algorithm 1 Bandoiu-Clarkson Algorithm**

1. Initialize the core-set \( Z_\varepsilon \)
2. Compute the minimal-enclosing-ball \( \mathcal{B}(c, R) \) of the core-set \( Z_\varepsilon \)
3. while A point \( z \notin Z_\varepsilon \) out of the ball \( \mathcal{B}(c, R) \) exist do
4. Include \( z \) in \( Z_\varepsilon \)
5. Compute the minimal-enclosing-ball
\[ \mathcal{B}(c, r) \] of the core-set \( \mathcal{C} \).

6: end while

In [7] is proved that the algorithm of Badoiu and Clarkson is a greedy approach to find a \( \varepsilon \)-core-set of \( \mathcal{S} \), which converges in no more than \( O(\varepsilon^{-2}) \) iterations. Since each iteration adds only one point to the core-set, the final size of the core-set is also \( O(\varepsilon^{-2}) \). Hence, the accuracy/complexity tradeoff of the obtained solution monotonically depends on \( \varepsilon \).

3. MULTI-CLASS EXTENSIONS

In a multi-class problem, examples \( \{x_i\} \) belong to a set of \( L \) categories \( c \in \{c_1, c_2, \ldots, c_L\} \) with \( L \geq 2 \) and hence the two “codes” \( +1 \) and \( -1 \) used to denote the two sides of a separating hyperplane are no longer enough to implement a decision function.

There are two types of extensions to build multiclass SVMs. One corresponds to use several binary classifiers, separately trained and joined into a multiclass decision function. For example in one-versus-the-rest (OVR, [6]), where a different binary SVM is used to separate each class from the rest; one-versus-one (OVO, [8]) where one binary SVM is used to separate each possible pair of classes; and DDAG [1] where one-versus-one classifiers are trained and then organized in a directed acyclic graph decision structure. Previous experiments in the context of SVMs show however that OVO frequently obtains a better performance both in terms of accuracy and training time [2]. Another type of extension consists in reformulating the quadratic program underlying SVMs to directly address the multiple classes in a single optimization problem. For the standard formulation of the SVM (L1-SVMs) examples of this approach are in [3], [9] and [10].

Up to our knowledge the only proposal of this nature directly addressing the multi-class extension of L2-SVMs is in [11]. This extension preserves the reduction to a minimal-enveloping-ball problem characteristic of the binary L2-SVM, which is the key requirement of our algorithms. The formulation associates each class \( c_k \in L \) of the problem looks for a projector \( P \) operating on the feature space \( \mathcal{F} = \phi(\mathcal{X}) \) which should allow to recover the correct code \( c_k \) for a given input \( \mathcal{X} \). Denote by \( t_k \) the vector associated to a class \( k \in L \). An easy and convenient way to account for the discrepancy between both vectors is by the dot product. If the codes are normalized to have the same norm, the greater dot product will yield the larger element of the label vector \( \mathcal{Y} \) corresponding to \( z_i \). One of the convenient ways is to choose it as

\[ y_{iz} = \begin{cases} \frac{|z_i - b|}{2b} & \text{if } z_i \text{ belongs to category } k, \\ 0 & \text{otherwise} \end{cases} \]

The inner product between the vectors will then be

\[ \langle y_i, y_j \rangle = \begin{cases} 1 & \text{if } z_i \text{ and } z_j \text{ are of same class}, \\ \frac{b - |y_i - y_j|}{b} & \text{otherwise}. \end{cases} \]

4. EQUIVALENCE BETWEEN L2-SVMs AND MEB PROBLEMS IN MULTI-CLASS APPROACH

Now, suppose we are computing the minimal-enveloping-ball in feature space \( \mathcal{F} = \phi(\mathcal{X}) \) which has been induced from \( \mathcal{X} \) by a mapping function \( \phi : \mathcal{X} \rightarrow \mathcal{F} \) and suppose we can compute dot products in \( \mathcal{F} \) directly from \( \mathcal{X} \) using a kernel function \( K(x_i, x_j) = \phi(x_i)^r \phi(x_j)^r = \mathcal{F}(x_i, x_j) \). Additionally suppose that the kernel is normalized, i.e., \( \forall x \in \mathcal{X}, K(x, x) = 1 \) with \( K \in \mathbb{R} \) a constant. Problem (3) is hence equivalent to solve the following quadratic program

\[ \min_{\alpha_i} \frac{1}{2} \langle \mathcal{H} \mathcal{C} \mathcal{C}^T \rangle + \langle \mathcal{B} \mathcal{C} \rangle + \mathcal{C} \quad \text{s.t.} \quad \alpha_i \geq 0, \quad \sum_{i=1}^L \alpha_i = 1, \quad \forall i \in \mathcal{I} \]
where \( f_{ij} = h(x_i, y_j) \). This problem coincides with the binary L2-SVM problem (2) and its multi-class implementation (8) if we set 
\[ h(x_i, y_j) = \gamma_i y_j h(x_i, x_j) \] in the binary case, and 
\[ h(x_i, y_j) = \gamma_i y_j h(x_i, x_j) \] in the multi-category case. The key requirement of the latter equivalence is the normalization constraint on \( f \). Note however that the binary and multi-category case, \( f \) is constant when the kernel used by the SVMs \( \kappa \) is, i.e., \( \kappa(x, x) = \kappa \) a constant. This is a property satisfied by any kernel of the form 
\[ h(x_i, y_j) = \exp \left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right), \] for example the gaussian or RBF kernel 
\[ h(x_i, y_j) = \exp \left(-\gamma\|x_i - x_j\|^2\right), \] which is the most commonly used in practice.

Thus, we can train L2-SVMs by solving a MEB problem in which the kernel \( f \) implementing its geometry depends on the kernel, the hyper-parameter \( \gamma \) and the codes used to represent the classes by the SVM.

5. REDUCED DATA APPROACHES

5.1 FORMULATION

The key idea of our method is to cast an SVM as a MEB problem in a feature space \( \mathcal{F} = \phi(\mathcal{X}) \) where the training examples are embedded via a mapping \( \phi \). Hence, we first formulate an algorithm to compute the MEB of the images \( \mathcal{S} \) of \( \mathcal{X} \) when \( \mathcal{X} \) is decomposed in a collection of sub-sets \( \mathcal{S}_j \). Then we will instantiate the solution for classifiers supporting the reduction to MEB problems.

The algorithm is based on the idea of computing core-sets \( \mathcal{C}_j \) for each set \( \mathcal{S}_j = \phi(\mathcal{S}_j) \) and taking its union \( \mathcal{C} = \bigcup_j \mathcal{C}_j \) as an approximation to a core-set for \( \mathcal{S} = \bigcup_j \mathcal{S}_j \). The generic procedure is depicted as algorithm (2). In a first step the algorithm extracts a core-set for each sub-set \( \mathcal{S}_j \). In the second step the MEB of the union of the core-sets is computed.

**Algorithm 2** Computation of the MEB of \( \mathcal{S} = \phi(\mathcal{S}) \)

Require: A partition of the set \( \mathcal{S} \) based Nearest Neighbor (algorithm 3) in a collection of subsets \( \mathcal{S}_j \)

1: for Each subset \( \mathcal{S}_j \), \( j = 1,...,N \) do
2: Compute a \( \epsilon \)-core-set \( \mathcal{C}_j \) for \( \mathcal{S} = \phi(\mathcal{S}) \)
3: end for
4: Join the core-sets \( \mathcal{C} = \mathcal{C}_1 \cup ... \cup \mathcal{C}_N \)
5: Compute the minimal enclosing ball of \( \mathcal{C} \). This is the minimal enclosing ball of \( \mathcal{S} \) that define the reduced data sets.

For the computation of the core-sets we use the Bădoiu and Clarkson algorithm [7] described in the previous section.

5.2. NEAREST NEIGHBOR ALGORITHM

A nearest neighbor (NN) or Voronoi vector quantizer [13] is a special class of vector quantizer (centroïde) in which the partition is completely determined by the codebook and a distortion measure. The vector quantizer is defined as the average value of all the inputs which fall within the cell. This definition requires that the centroid lies within the cell boundaries.

**Algorithm 3** Nearest Neighbor

1: initialization: \( \hat{d} = d_0 \), \( j = 1 \), ..., \( j = 1 \)
2: compute \( D_j = d(x, y_j) \),
3: if \( D_j < \hat{d} \), set \( D_j = \hat{d} \) and set \( j = j \)
4: if \( j < N \) then set \( j = j + 1 \) and goto 2
5: if \( j = N \), stop

where \( d_0 \) must be larger than any expected distortion (typically it is set to the processor’s largest positive value) and \( N \) define the number of sub-set \( \mathcal{S}_j \). In the nearest neighbor encoding algorithm shown above, an exhaustive search of the codebook is performed: that is, all of the vectors quantizers are compared to the input vector and the best match is chosen. Note that in our study, we used this algorithm just for partition the data set.

5.3. INSTANTIATION FOR THE OVO APPROACH
From the previous section we have that training a binary L2-SVM on a dataset \( \mathcal{Z} \) is equivalent to build a minimal enclosing-ball of \( \mathcal{Z} \) if \( \phi(x)^T \phi(x) \) is implemented using the kernel
\[
\hat{k}(x_1,x_2) = \gamma y_1 y_2 k(x_1,x_2) + \frac{1}{2} \alpha_i
\]
The OVO procedure to obtain a multi-category SVM works by combining one binary SVM for each pair of classes. An instantiation of algorithm (2) would hence consist in computing core-sets for the subset of examples belonging to each pair of classes, and then joining them and finally recovering the binary model for this pair. However, since each class participates in \( L \) models, core-sets for each pair of classes can be highly redundant overloading the network unnecessarily. Thus, we proceed as in algorithm (4), joining the core-sets at each node before sending the results to the coordinator node.

**Algorithm 4** Computation of the MEB of \( \mathcal{Z} = \phi(\mathcal{S}) \) using One-Versus-One L2-SVMs

1: for Each subset \( \mathcal{S}_n \), \( n = 1, \ldots, p \) do 
2: for Each Class \( k = 1, \ldots, l - 1 \) do 
3: for Each Class \( m = k + 1, \ldots, l \) do 
4: Let \( \mathcal{S}_{km} \) the subset of \( \mathcal{S}_k \) not corresponding to class \( k \) and \( m \). 
5: Label \( \mathcal{S}_{km} \) using the standard binary codes +1 and -1 for class \( k \) and \( m \) respectively. 
6: Compute a core-set \( C_{km}^{\mathcal{S}} \) of \( \mathcal{S}_{km} \) 
Using the kernel
\[
\hat{k}(x_1,x_2) = \gamma y_1 y_2 k(x_1,x_2) + \gamma y_2 + \frac{1}{2} \alpha_i
\]
7: end for 
8: end for 
9: Take the union of the core-set inferred for each pair of classes \( C_{km} = C_{km}^{\mathcal{S}} \cup \ldots \cup C_{km}^{\mathcal{S}} \) 
10: end for 
11: Join core-set \( C_{z} = C_{z} \cup \ldots \cup C_{z} \) 
12: Compute the minimal enclosing ball of \( C_{z} \) using the same kernel \( \hat{k} \)

**5.4. INSTANTIATION FOR DIRECT METHOD APPROACH**

In contrast to the OVO decomposition heuristic, a direct implementation is defined by a single optimization which coincides with a MEB problem just by using the kernel
\[
\hat{k}(x_1,x_2) = \gamma y_1 y_2 k(x_1,x_2) + \gamma y_2 + \frac{1}{2} \alpha_i
\]The use of algorithm (2) is hence straightforward and consists in computing any dot product \( \phi(x_1)^T \phi(x_2) = \hat{k}(x_1,x_2) \) using this kernel. The instantiation is depicted as algorithm (5).

**Algorithm 5** Computation of the MEB of \( \mathcal{Z} = \phi(\mathcal{S}) \) using Direct Multiclass L2-SVM

1: for Each subset \( \mathcal{S}_n \), \( n = 1, \ldots, p \) do 
2: Label each example \( x_i \in \mathcal{S}_n \) with the code \( y_i \) assigned to the class of \( x_i \) and let \( y_i \) such label 
3: Compute a core-set \( C_{n} \) of \( \mathcal{S}_n \) using the kernel \( \hat{k}(x_i,x_j) = \gamma y_i y_j k(x_i,x_j) + \gamma y_j + \frac{1}{2} \alpha_i \)
4: end for 
5: Join the core sets \( C_z = C_{z} \cup \ldots \cup C_{z} \) 
6: Compute the minimal enclosing ball of \( C_{z} \) using the same kernel \( \hat{k} \)

**6. EXPERIMENTS**

This section presents the performance of text-independent speaker verification task based on the **Gaussian Mixture Model – Universal Background Model (GMM-UBM)** described in [15]. We compare the performance of speaker verification system with three UBMs, the first one was created directly from the Speaker Recognition corpus [14] (formerly known as Speaker Verification), consists of telephone speech from 91 and the two last later is the reduced first one from the application of our two algorithms developed in section 5. In particular, we train a 1024-mixture gender-independent from each UBM with diagonal covariance matrices. Speaker GMMs are trained by adapting only the mean vectors from the UBM using a relevance factor \( r \) of 16.

Figure 3 shows the detection error tradeoff (DET) curves for the speaker verification system using three UBMs. The system based reduced GMM-UBM1 from Direct Multiclass L2-SVM slightly outperforms the GMM-UBM with an equal-error-rate (EER) of 8.49 \%, compared to 8.78 \% of the GMM-UBM2. The system based reduced GMM-UBM from One-Versus-One L2-SVMs exhibits the best performance with an EER of 8.10 \%.

**7. CONCLUSION**

In this paper, we proposed two algorithms that compute an approximation to the minimum enclosing ball of a given finite set of vectors. Both algorithms are especially well-suited for large-scale instances of the minimum enclosing ball problem and can compute a small core set whose size depends only on the approximation parameter.

We have explored two methods based on the computation of core-sets to train multi-category SVM models when the set of examples is fragmented. The main contribution has been to demonstrate through our experiments, that the methods proposed can reproduce with high accuracy of a
solution where the noisy sample in huge data set are eliminated, without complex and costly computation. SVMs based on core-sets have shown however important advantages in large-scale applications, which can hence be extended to distributed data-mining problems. A real contribution of this work has been a new direct implementation of multi-category SVMs supporting a reduction to a minimal-enclosing-ball (MEB) problem. Although the core-sets method exhibits always better prediction accuracy used with the OVO scheme, the direct implementation shows a lower complexity and it is better than the previous direct implementation proposed for MEB based SVMs.

We have developed two such algorithms for the minimum enclosing ball problem in this paper. We intend to continue our work on developing specialized algorithms for other classes of large-scale structured optimization problems in the near future

REFERENCES