title

Tinghui Zhou

1   Introduction

The proposed approach consists of two major components: 1) providing an efficient and effective data description scheme based on the idea of Minimum Enclosing Balls (MEBs); 2) developing a new SVM variant (referred as Ball SVM for now) that is suitable for learning from data described by the MEBs.

One application of this approach is a much faster version of the image matching framework proposed in [1]. As pointed out in the paper, a big limitation of their approach is the speed, since "it requires training an SVM (with hard-negative mining) at query time ... too slow for many practical applications...". However, noting that the negative set in their case remains the same for all query images, an interesting question would be: is it possible to describe the negative set using an alternative representation such that it's efficient to be trained on? To do that, we propose a data description scheme using a set of MEBs, which tries to characterize the data with an efficient representation that can be utilized by our proposed SVM variant at query time.

Several benefits of this approach:

- Less memory storage. The negative set can be discarded once its corresponding MEBs are extracted.

- Faster SVM training at query time. Originally the hard-negative mining needs to be performed over millions of image patches. With our approach, the size of negative set reduces to the number of MEBs, which would be several orders of magnitude smaller.

2   Data description with Minimum Enclosing Balls

In this section, we describe a method for data description using a set of Minimum Enclosing Balls (MEBs).

2.1 Problem Formulation

Suppose we are given a set of data points $X = \{x_1, \ldots, x_N\}$. The goal is to find a set of balls $B = \{(c_1, r_1), \ldots, (c_K, r_K)\}$, with c_k and r_k denote the center and radius of the k-th ball respectively, such that they enclose as many data points with as little volume as possible.

To fit a MEB to each cluster, we use the method proposed by Tax and Duin [2], where they solve this problem via a formulation similar to Support Vector Classifier:

$$\min_{c,r} \ r^2 + C \sum_i \xi_i$$
$$\text{s.t. } \|x_i - c\|^2 \leq r^2 + \xi_i, \ \xi_i \geq 0 \ \ \forall i \tag{1}$$

2.2 Solving SVDD with gradient descent

The SVDD objective given in Eq. 1 can also be formulated as follows:

$$\min_{c,r} \ r^2 + C \sum_i \max(0, \|x_i - c\|^2 - r^2). \tag{2}$$

Let F denote the corresponding objective function. We minimize F by iteratively updating r and c. Specifically, the gradient over c given fixed r is given by

$$\frac{\partial F}{\partial c} = C \sum_i g_c(x_i),$$

where

$$g_c(x_i) = \begin{cases} 0 & \text{if } \|x_i - c\|^2 - r^2 \leq 0 \\ 2(c - x_i) & \text{otherwise.} \end{cases}$$

Similarly, the gradient over r given fixed c is

$$\frac{\partial F}{\partial r} = 2r + C \sum_i g_r(x_i),$$

where

$$g_r(x_i) = \begin{cases} 0 & \text{if } \|x_i - c\|^2 - r^2 \leq 0 \\ -2r & \text{otherwise.} \end{cases}$$

Note that in Page 5 of [2], it's claimed that $C \geq 1$ indicates the hard-margin solution. This is because if there's ≥ 1 points that are not enclosed, the gradient over r as in Eq. 2.2 will be negative. So r will become larger and larger until it can enclose all the points. In other words, the valid range for C is $(0, 1)$.

2

3 Ball SVM

Now we introduce an SVM variant, referred as Ball SVM, which finds the separating hyperplane with maximum margin for data points described by MEBs. We shall see later that SVM turns out to be a special case of Ball SVM.

Let the training data be $D = \{\mathbf{c}_k, r_k, n_k, y_k\}_{k=1,...,K}$, where $\mathbf{c}_k$ is the center of the ball, r_k is its corresponding radius, n_k is the number of points enclosed by the ball, and s_k equals to 1 if the ball encloses positive points and 0 otherwise. Our goal is to find a hyperplane $\mathbf{w}$ that separates the positive and negative balls with maximum margin. First, we define the distance from a ball to the hyperplane as $d_k = \frac{y_k(\mathbf{w}^\top \mathbf{c}_k + b)}{\|\mathbf{w}\|} - r_k$, i.e. the distance from the center to the hyperplane minus the radius. Following the standard SVM derivation in [3], we would like to find a $\mathbf{w}$ such that $d_k > 0$ for all k, and the margin in this case is given by $\min_k d_k$, i.e. the distance from the closest ball to the hyperplane. Therefore, our goal of finding a maximum margin solution can be formulated as

$$\max_{\mathbf{w}} \left\{ \min_k \left[\frac{y_k(\mathbf{w}^\top \mathbf{c}_k + b)}{\|\mathbf{w}\|} - r_k \right] \right\} \iff \max_{\mathbf{w}} \left\{ \frac{1}{\|\mathbf{w}\|} \min_k \left[y_k(\mathbf{w}^\top \mathbf{c}_k + b) - \|\mathbf{w}\| r_k \right] \right\}. \quad (3)$$

Since rescaling of $\mathbf{w}$ does not change the distance, we set $y_k(\mathbf{w}^\top \mathbf{c}_k + b) - \|\mathbf{w}\| r_k = 1$ for the ball closest to the hyperplane. Then all other balls would satisfy $y_k(\mathbf{w}^\top \mathbf{c}_k + b) - \|\mathbf{w}\| r_k \geq 1$. Finally, adding the slack variable ξ_k to take into account the non-separable case, we have the following optimization problem:

$$\min_{\mathbf{w}, \xi_k} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{k=1}^{K} n_k \xi_k$$

$$\text{subject to} \quad y_k(\mathbf{w}^\top \mathbf{c}_k + b) - r_k \|\mathbf{w}\| \geq 1 - \xi_k, \quad k = 1, \ldots, K. \quad (4)$$

Note that each ξ_k is further weighted by n_k so that the learned hyperplane places more focus on balls enclosing more points. Now we can see that standard SVM is a special case of Ball SVM with data points having zero radius.

3.1 Gradient Descent for Ball SVM

We rewrite the objective we want to minimize in an unconstrained form:

$$L_D(\mathbf{w}, b) = \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{k=1}^{K} n_k \max[0, 1 - y_k(\mathbf{w}^\top \mathbf{c}_k + b) + r_k \|\mathbf{w}\|]. \quad (5)$$

The gradient over $\mathbf{w}$ is then given by

$$\nabla_{\mathbf{w}} L_D = \mathbf{w} + C \sum_{k=1}^{K} n_k h(\mathbf{w}, b, \mathbf{c}_k, r_k, y_k),$$

3

where

$$h(\mathbf{w}, b, \mathbf{c}_k, r_k, y_k) = \begin{cases} 0 & \text{if } y_k(\mathbf{w}^\top \mathbf{c}_k + b) - r_k \|\mathbf{w}\| \geq 1 \\ -y_k \mathbf{c}_k^\top + r_k \frac{\mathbf{w}}{\|\mathbf{w}\|} & \text{otherwise.} \end{cases}$$

Similarly, the gradient over b is given by

$$\nabla_b L_D = C \sum_{k=1}^{K} n_k f(\mathbf{w}, b, \mathbf{c}_k, r_k, y_k),$$

where

$$f(\mathbf{w}, b, \mathbf{c}_k, r_k, y_k) = \begin{cases} 0 & \text{if } y_k(\mathbf{w}^\top \mathbf{c}_k + b) - r_k \|\mathbf{w}\| \geq 1 \\ -y_k & \text{otherwise.} \end{cases}$$

3.2   K-means SVM (Baseline)

As a baseline, we use only the K-means cluster centers to represent data. Consequently, it is equivalent to Ball SVM with radius of the balls being zero. The objective function then becomes

$$\min_{\mathbf{w}, \xi_k} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{k=1}^{K} n_k \xi_k$$

$$\text{subject to} \quad y_k(\mathbf{w}^\top \mathbf{c}_k + b) \geq 1 - \xi_k, \quad k = 1, \dots, K. \tag{6}$$

Rewrite the objective in unconstrained form:

$$L_D(\mathbf{w}, b) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{k=1}^{K} n_k \max[0, 1 - y_k(\mathbf{w}^\top \mathbf{c}_k + b)]. \tag{7}$$

The gradient over $\mathbf{w}$ is then given by

$$\nabla_{\mathbf{w}} L_D = \mathbf{w} + C \sum_{k=1}^{K} n_k h(\mathbf{w}, b, \mathbf{c}_k, y_k),$$

where

$$h(\mathbf{w}, b, \mathbf{c}_k, r_k, y_k) = \begin{cases} 0 & \text{if } y_k(\mathbf{w}^\top \mathbf{c}_k + b) \geq 1 \\ -y_k \mathbf{c}_k^\top & \text{otherwise.} \end{cases}$$

Similarly, the gradient over b is given by

$$\nabla_b L_D = C \sum_{k=1}^{K} n_k f(\mathbf{w}, b, \mathbf{c}_k, y_k),$$

where

$$f(\mathbf{w}, b, \mathbf{c}_k, y_k) = \begin{cases} 0 & \text{if } y_k(\mathbf{w}^\top \mathbf{c}_k + b) \geq 1 \\ -y_k & \text{otherwise.} \end{cases}$$
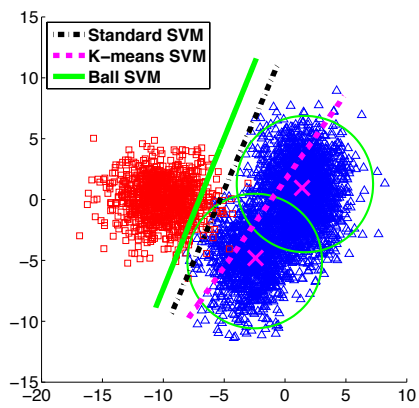
4 Results on a Toy Dataset

To visually compare the resulting hyperplane obtained by Ball SVM, baseline k-means SVM, and standard SVM, respectively, we perform binary classification on a toy 2D dataset, with results shown in Figure 1. The goal is to separate the positive points (shown in red squares) from the negative ones (shown in blue rectangles). First, we run k-means clustering on the negative set, and use the SVDD algorithm described in Section 2.2 to fit a MEB for each cluster. Then the Ball SVM is trained using the positive points (which are also treated as balls with zero radius) and the negative balls, the k-means SVM is trained using the positive points and the negative cluster centers, and the standard SVM is trained using the original data.

One can see that when k is small, Ball SVM provides much better approximation to standard SVM than k-means SVM. This is because Ball SVM is able to incorporate the information on the geometric span of each negative cluster when training using the MEBs, while such information is lost in k-means SVM. In addition, both Ball SVM and k-means SVM provide better and better approximation to standard SVM as the number of clusters k increases (but in all cases Ball SVM still appears to outperform k-means SVM).
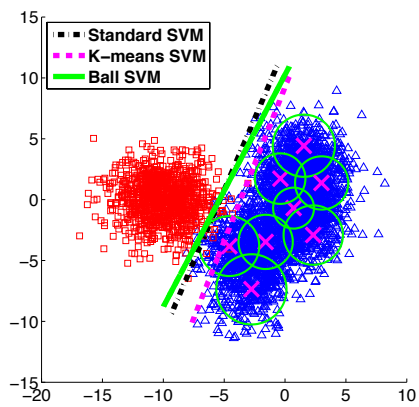
References

[1] A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros, "Data-driven visual similarity for cross-domain image matching," in *ACM SIGGRAPH ASIA*, 2011.

[2] D. M. Tax and R. P. Duin, "Support vector data description," *Machine Learning*, 2004.

[3] C. M. Bishop, *Pattern Recognition and Machine Learning.* Springer, 2006.
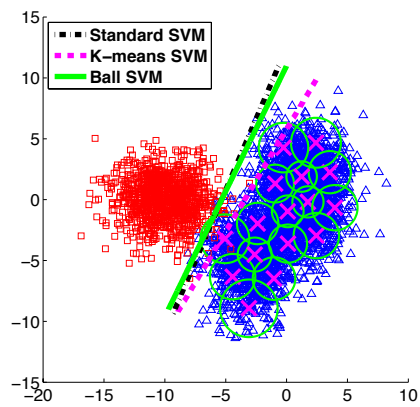
(a) $k = 2$
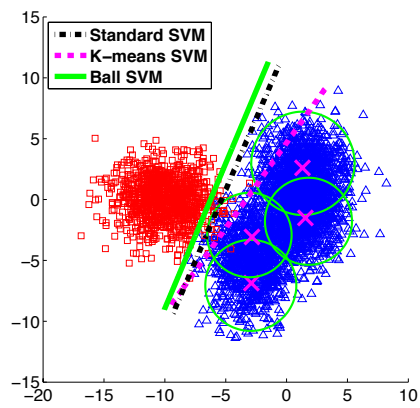
(b) $k = 4$

(c) $k = 8$

(d) $k = 16$

Figure 1: Results on a toy 2D dataset.