Support Vector Data Description Using Clustering Method

Hyung jin Chang[1] Pyo-Jae Kim[2] Jung Hwan Choi[3] and Jin Young Choi[4]
School of Electrical Engineering and Computer Sciences
Seoul National University
San 56-1, Sillim-Dong, Gwanak-Gu, Seoul 151-742, Korea
Tel: +82-2-872-7283, Fax: +82-2-885-4459
E-mail: hjchang@neuro.snu.ac.kr, pjkim@neuro.snu.ac.kr, jhchoi@neuro.snu.ac.kr, jychoi@neuro.snu.ac.kr

Abstract: Support Vector Data Description (SVDD) is a good one-class classifier, but it has a runtime complexity of $O(N^3)$ that SVDD has a limitation of dealing with a large data set. To solve this scale problem, we propose SVDD using clustering method. Our method is similar to divide-and-conquer strategy; trains each decomposed subproblems to get support vectors. We use two different clustering algorithms; K-means clustering and Mean Shift clustering. K-means clustering shows better performance in computational time reduction, and Mean Shift clustering preserves and describes well the characteristic of data distribution. Through experiments, we compare the results depending on which clustering algorithm is used, and show efficiency of our method.

1. Introduction

SVDD [1] is a well-known one class classification algorithm. This method finds a boundary of a target class in data space by assuming a hypersphere which has minimum volume enclosing almost all target class objects in feature space. SVDD uses support vectors to describe the boundary of target class as Support Vector Machine (SVM) does. Support vectors are found by solving convex quadratic programming (QP). However solving QP takes too much time that SVDD cannot apply to a large data set.

In this paper, we propose a new method for a scale problem of SVDD. Our method is not aimed to solve QP quickly, but based on the divide-and-conquer strategy. We decompose a large data set into a series of small sub-data groups using clustering algorithm. Each small sub-problem is solved by its own local expert with SVDD.

We use two different clustering algorithm; K-means clustering and Mean Shift clustering[2][3]. Each clustering algorithm has its own characteristics. K-means clustering is the fastest clustering method, but we need to know K as a prior knowledge, and the results are not same all the time. Mean Shift clustering takes more time than K-means clustering, but the clustered sub-groups are containing the characteristic of data distribution that we can use the sub-groups to extract some knowledge about the data later.

The proposed method is applied to data sets which are various in size and shape. Comparing with the original SVDD, we show our method has similar data description results with less computational load. Also we compare the results depending on which clustering method used.

2. C-SVDD: Support Vector Data Description using Clustering

In this section, we give detail explanation of the proposed method. We present the basic theory of SVDD and Mean Shift Clustering, and then introduce a model of C-SVDD.

2. 1 Basic Theory of SVDD

SVDD is similar to the hyperplane approach, which estimates decision plane to separate the target objects from the origin with maximal margin. However, instead of using a hyperplane, SVDD uses a minimum hypershpere. The sphere is characterized by center a and radius R>0, and demanded containing all training objects. SVDD minimizes the volume of the sphere by minimizing R^2

Analogous to the Support Vector Classifier (Vapnik, 1998), the Support Vector Data Description defines the error function to minimize:

$$F(R,a) = R^2 + C\sum_i \xi_i \,, \tag{1}$$

with the constraints:

$$\|x_i - a\|^2 \le R^2 + \xi_i, \qquad \xi_i \ge 0, \quad \forall i, \tag{2}$$

where ξ_i are slack variables $\xi_i \ge 0$, and the parameter C controls the trade-off between the volume and the errors [1]. Constraints can be incorporated into the cost function by using Lagrange multipliers:

$$L(R,a,\alpha_i,\gamma_i,\xi_i) = R^2 + C\sum_i \xi_i - \sum \alpha_i \left\{R^2 + \xi_i - \left(\|x_i\|^2 - 2a\cdot x_i + \|a\|^2\right)\right\} - \sum_i \gamma_i \xi_i \tag{3}$$

$$\alpha_i \ge 0, \gamma_i \ge 0$$

L should be minimized with respect to R, a, ξ_i and maximized with respect to α_i, γ_i. After setting partial derivatives of L to zero, solve each equation with a QP solver. Those objects x_i with $0 < \alpha_i < C$ are called *support vectors*, and used to describe a boundary description. Especially, objects x_i with $\alpha_i = C$ are called *outliers.*

To test an object z, the distance to the center of the sphere is used to make a decision.

$$\|z-a\|^2 = K(z,z) - 2\sum_i \alpha_i K(z,x_i) + \sum_{i,j} \alpha_i \alpha_j K(x_i,x_i) \le R^2, \tag{4}$$

where K is a kernel function to solve non-separable problems. When the distance is smaller or equal than the radius, then a test object z is accepted.

2. 2 Basic Theory of Mean Shift Clustering

Mean Shift Algorithms is a non-parametric density gradient estimation method proposed by Fukunaga and Hostetler [4]. In n-dimension space X and a Kernel set S, when the kernerl's center is $x \in X$, x's sample mean m(x) is

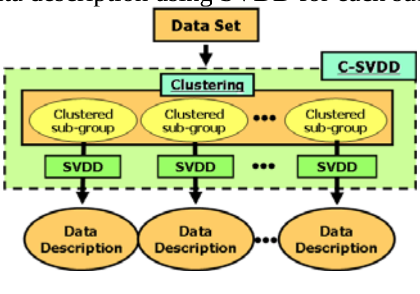
$$m(x) = \frac{\sum_{s \in S} K(s-x)s}{\sum_{s \in S} K(s-x)}, \qquad (5)$$

the difference between m(x) and kernel's center x is called *Mean Shift*, and the *Mean Shift Algorithm* is moving the kernel's center to m(x) iteratively. If the kernel's moving stops, then the location x is a mode of that distribution. Mean Shift Clustering is performed based on this mode.

2. 3 C-SVDD Algorithm

Step1. Sub-grouping using Clustering method

Step2. Data description using SVDD for each sub-group



3. Experimental Results

We performed simulations using two classes data set, which we made for comparison in training time and accuraty. Tax's data description toolbox[5] and quadratic program solver(quadprog) provided by MATLAB are used in all experiments so the training time difference caused by the QP solver doesn't occur. Simulations are run on a PC with a 3.0 GHz Intel Pentium IV processor and 1G RAM.

3. 1 Computational Time Reduction

Figure 1. shows the computational time reduction effect of our algorithm. Comparing with original SVDD, clustering combined SVDD takes much less time.

Data #	Learning Time (sec)		
	K-means + SVDD	Mean Shift + SVDD	SVDD
300	1.47	2.28	52.99
400	1.82	2.99	133.49
500	3.04	5.11	276
600	3.38	6.52	568.13
700	4.54	8.41	975.3
800	5.20	10.39	1650

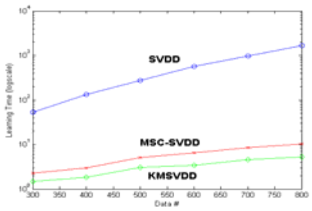


Figure 1. Computational time compare (in log scale)

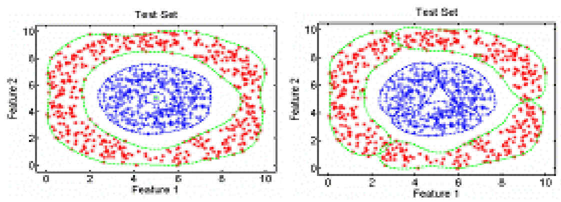


Figure 2. Data description result of doughnut shape

(L) Original SVDD (R) K-means clustering SVDD

The computational time reduction performance, especially K-means clustering combined SVDD, is outstanding, comparing with the original SVDD. Also the description results are similar to the original one.

3. 2 Clustering Preserving the Characteristic of Data Distribution

Mean Shift Clustering is clustering data based on the mode of data distribution, that each clustered set has only one mode and presercving the data distribution shape. Figure 3. shows the original data distribution described by Parzen window method. Figure 4. is K-means clustering SSVDD and Mean Shift Clustering-SVDD trained results.

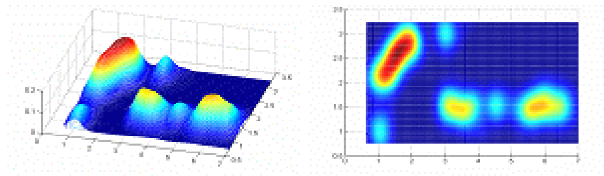


Figure 3. Original Data Distribution (using Parzen window)

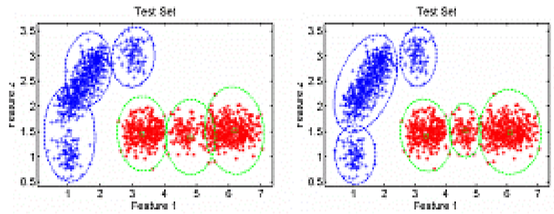


Figure 4. Clustering SVDD

(L) K-means clustering SVDD (R) Mean Shift clustering SVDD

Comparing Figure 4. with Figure 3., Mean Shift Clustering-SVDD preserves the data distribution and describes better than K-means clustering SVDD.

4. Discussion and Conclusion

To deal with a large data set with SVDD, we proposed Clustering combined SVDD. We used two different clustering methods; K-mean clustering and Mean Shift clustering. Both clustering methods showed outstanding computatinal time reduction effect when thery are combined with SVDD, especially the K-means clustering, and the description results are not that different from the original SVDD's. Mean Shift clustering SVDD also showed an obvious computational time reduction preserving the characteristic of data distribution. We can use the description results gotten by Mean Shift clustering SVDD in analysing the data distribution and density estimation.

References

[1] David M.J. Tax, *"Support Vector Data Description,"* Machine Learning, vol, 54, pp. 45-66, 2004.

[2] Y. Cheng, *"Mean shift, mode seeking, and clustering,"* IEEE Trans. Pattern Anal. Machine Intell., vol. 17, 790-799, 1995.

[3] D. Comaniciu, and P. Meer, *"Mean shift: A robust approach toward feature space analysis,"* IEEE Trans. Pattern Anal. Machine Intell., pages 24(5):603-619, 2002.

[4] K. Fukunaga and L.D. Hostetler, *"The estimation of the gradient of a density function, with applications in pattern recognition,"* IEEE Trans. Information Theory, vol. 21, pp. 32-40, 1975.

[5] Tax, D.M.J.: Data Description Toolbox for Matlab. http://www.ict.ewi.tudelft.nl/~ddavidt/dd_tools.html(2006)