

# Tree-Based Structural Twin Support Tensor Clustering with Square Loss Function

Reshma Rastogi<sup>(✉)</sup>  and Sweta Sharma 

Department of Computer Science, Faculty of Mathematics and Computer Science,  
South Asian University, New Delhi, India  
`reshma.khemchandani@sau.ac.in`

**Abstract.** Most of the real-life applications involving images, videos etc. deals with matrix data (second order tensor space). Tensor based clustering models can be utilized for identifying patterns in matrix data as they take advantage of structural information in multi-dimensional framework and reduce computational overheads as well. Despite such numerous advantages, tensor clustering has still remained relatively unexplored research area. In this paper, we propose a novel clustering technique, termed as Treebased Structural Least Squares Twin Support Tensor Clustering (Tree-SLSTWSTC), that builds a cluster model as a binary tree, where each node comprises of proposed Structural Least Squares Twin Support Tensor Machine (S-LSTWSTM) classifier that considers the structural risk minimization of data alongside a symmetrical L2-norm loss function. The proposed approach results in time-efficient learning. Initialization framework based on tensor  $k$ -means has been proposed and implemented in order to overcome the instability disseminated by random initialization. To validate the efficacy of the proposed framework, computational experiments have been performed with relevant tensor based models on face recognition and optical digit recognition datasets.

**Keywords:** Twin Support Tensor Machine · Unsupervised learning · Tree-based clustering · Tensor space

## 1 Introduction

In machine learning applications, particularly image processing, computer vision and bioinformatics, data is often represented in matrix form (second order tensor space). For example- a gray scale image is a order-2 tensor and a video is a order-3 tensor. Here, one of the critical task is to identify the hidden patterns in training data [1]. Most commonly, such data are converted in vector form so as to facilitate the use of vector based clustering or classification models. This arrangements, however, suffers from the limitations of under-representation and high dimensionality (sometimes over-fitting) problem leading to high training time complexity [2, 3].

Clustering is a powerful technique that aims to group together similar elements in same cluster while maximizing the segregation between dissimilar elements. Recently, in view of limitations of point-based clustering methods in dealing the data which is not distributed around several cluster points, plane based clustering methods such as Maximum Margin clustering (MMC) and Twin support vector Clustering (TWSVC) [4] have attracted considerable research interest. Taking motivation from TWSVC, we propose Treebased clustering framework for clustering second order tensor data. The main contributions of paper includes the following: First, we propose a modified tensor based LS-TWSTM named as Structural LSTWSTM (S-LSTWSTM) [5] classifier that formulates convex optimization problems as system of linear equations that takes care of structural risk associated with the data. Then, S-LSTWSTM has been extended to binary decision structure based clustering framework, termed as Tree-SLSTWSTC, which leads to fast and efficient cluster assignment in tensor framework. Finally, to make our Tree-SLSTWSTC more robust and stable, initialization technique based on Tensor  $k$ -means is proposed.

Experiments have been carried out on popular image datasets that establish the out-performance of our proposed algorithm over other vector and tensor based clustering techniques significantly.

The rest of the paper is organized as follows. Section 2 gives the background for our proposed approach. Section 3 discusses our proposed work. Experimental results have been shown in Sect. 4. Finally, Sect. 5 concludes our work and state possible future direction of work.

## 2 Related Work

Let  $X = \{X_1, X_2, \dots, X_m\}$  be a training set of  $m$  data samples in second order tensor space i.e.  $X_i \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$ . Let  $I_1$  represent the set of indices with  $y_i = 1$ , and  $I_2$  represent the set of indices with label  $y_i = -1$ .

### 2.1 Least Squares Twin Support Tensor Machine

Working on the tensor generalization of Twin Support Vector Machine [7], Zhao et al. [5] proposed Least Squares Twin Support Tensor Machine (LS-TWSTM) which aims to find a pair of non-parallel hyperplanes given by  $f_1(X) = u_1^T X v_1 + b_1$  and  $f_2(x) = u_2^T X v_2 + b_2$  where  $u_1, u_2 \in \mathbb{R}^{n_1}$ ,  $v_1, v_2 \in \mathbb{R}^{n_2}$  and  $b_1, b_2 \in \mathbb{R}$ . Following two QPPs are solved to find the corresponding non-parallel hyperplanes:

$$\begin{aligned}
 \text{(LS-TWSTM 1)} \quad & \min_{u_1, v_1, b_1, \xi_2} \frac{1}{2} \sum_{i \in I_1} (u_1 X_i v_1 + b_1)^2 + c_1 \sum_{j \in I_2} \xi_{2j}^2 \\
 & \text{subject to} \quad - (u_1^T X_j v_1 + b_1) + \xi_{2j} = 1, \quad j \in I_2.
 \end{aligned}$$

$$\begin{aligned}
\text{(LS-TWSTM 2)} \quad & \min_{u_2, v_2, b_2, \xi_1} \frac{1}{2} \sum_{j \in I_2} (u_2 X_j v_2 + b_2)^2 + c_2 \sum_{i \in I_1} \xi_{1i} \\
& \text{subject to } (u_2^T X_i v_2 + b_2) + \xi_{1i} = 1, \quad i \in I_1.
\end{aligned}$$

Since the hyperplane parameters are interdependent, the problems are solved using alternate projection method [6]. A test point is assigned a label depending upon its proximity from two hyperplanes. Please refer to [5] for details.

### 3 Proposed Work

In this work, we first propose a novel tensor classifier termed as Structural Least Squares Twin Support Tensor Machine (S-LSTWSTM), which we further use in an unsupervised framework to propose a binary treebased clustering approach termed as Tree-SLSTWSTC.

#### 3.1 Structural Least Squares Twin Support Tensor Machine

In the spirit of Least Squares Twin Support Tensor Machine (LS-TWSTM) [5], the proposed S-LSTWSTM seeks two non-parallel hyperplanes by considering the following optimization problems:

$$\begin{aligned}
\text{(S-LSTWSTM 1)} \quad & \min_{u_1, v_1, b_1, \xi_2} \frac{1}{2} \sum_{i \in I_1} (u_1 X_i v_1 + e_1 b_1)^2 + c_1 \sum_{j \in I_2} \xi_{2j}^2 + c_2 (u_1^T u_1 + v_1^T v_1 + b_1^2) \\
& \text{subject to } (u_1^T X_j v_1 + b_1 e_2) = e_2 - \xi_{2j}, \quad j \in I_2,
\end{aligned} \tag{1}$$

$$\begin{aligned}
\text{(S-LSTWSTM 2)} \quad & \min_{u_2, v_2, b_2, \xi_1} \frac{1}{2} \sum_{j \in I_2} (u_2 X_j v_2 + e_2 b_2)^2 + c_1 \sum_{i \in I_1} \xi_{1i}^2 + c_2 (u_2^T u_2 + v_2^T v_2 + b_2^2) \\
& \text{subject to } (u_2^T X_i v_2 + b_2 e_1) = e_1 - \xi_{1i}, \quad i \in I_1,
\end{aligned} \tag{2}$$

where  $\xi_1$  and  $\xi_2$  are error variables; and  $e_1$  and  $e_2$  are appropriate dimensional matrices of ones. The first term of Eqs. (1) and (2) calculates the empirical risk of the data. Thus, minimizing this term tends to keep the hyperplane close to the data matrices and the constraints require the hyperplane to be at unit distance from the other class. Further, S-LSTWSTM takes care of structural risk minimization (SRM) by introducing the term  $(u_i^T u_i + v_i^T v_i + b_i^2, i = 1, 2)$  in the objective function and thus improves the generalization ability. It also takes care of the possible ill-conditioning that might arise during matrix inversion.

Working on the lines on LS-TWSTM [5] for Eq. (1) and setting the gradient of objective function with respect to  $(u_1, v_1, b_1)$  to zero, indicates that  $u_1, v_1$  and  $b_1$  are inter-dependent and hence can not be solved independently. Therefore, we use alternating projection method [6].

For any given non-zero vector  $u_k \in \mathbb{R}^{n_1}$ , let  $x_i^T = u_k^T X_i$  and  $x_j^T = u_k^T X_j$ , we then solve for the following modified optimization problem (obtained after substituting the value of  $\xi_{2j}$  in the objective function):

$$\min_{v_k, b_k} \frac{1}{2} \sum_{i \in I_1} (x_i v_k + b_k)^2 + c_1 \sum_{j \in I_2} \|e_2 - (x_j v_k + b_k e_2)\|^2 + c_2 (v_k^T v_k + b_k^2). \quad (3)$$

Differentiating Lagrangian corresponding to (3) with respect to  $v_k$  and  $b_k$ , leads to the following system of linear equations:

$$\begin{bmatrix} v_k \\ b_k \end{bmatrix} = - \begin{bmatrix} \frac{1}{c_1} H_1^T H_1 + G_1^T G_1 + c_2 I \\ c_1 \end{bmatrix}^{-1} G_1^T e_2, \quad (4)$$

where  $H_1$  and  $G_1$  are matrices of points  $x_i$  and  $x_j$  augmented with a column of ones; and  $I$  is an identity matrix of appropriate dimensions.

Once a non-zero vector  $v_k \in \mathbb{R}^{n_2}$  is obtained, let  $\hat{x}_i^T = X_i v_k$  and  $\hat{x}_j^T = X_j v_k$ , we solve for the following modified optimization problem:

$$\min_{u_k, b_k} \frac{1}{2} \sum (\hat{x}_i u_k + b_k)^2 + c_1 \|(\hat{x}_j u_k + b_k e_2) - e_2\|^2 + c_2 (u_k^T u_k + b_k^2). \quad (5)$$

Working on the lines as above, we obtain  $(u_k, b_k)$  as follows

$$\begin{bmatrix} u_k \\ b_k \end{bmatrix} = - \begin{bmatrix} \frac{1}{c_1} H_2^T H_2 + G_2^T G_2 + c_2 I \\ c_1 \end{bmatrix}^{-1} G_2^T e_2, \quad (6)$$

where  $H_2$  and  $G_2$  are matrices of points  $\hat{x}_i$  and  $\hat{x}_j$  augmented with a column of ones. The Eqs. (4) and (6) are solved alternatively until  $u_k$ ,  $v_k$  and  $b_k$  converges.

On the similar lines as above, the solution of (2) is obtained. A new test point is assigned a class label similar to LS-TWSTM [5] based on proximity criteria.

### 3.2 Tree-based Structural Least Squares Twin Support Vector Clustering

Tree-SLSTWSTC algorithm creates a binary tree of clusters which partitions the data at multiple levels of the tree until desired number of clusters are obtained. Unlike TWSVC [4], Tree SLSTWSTC uses symmetric squared loss function at each internal node that handles the issue of premature convergence of cluster framework. The proposed algorithm Tree-SLSTWSVC starts with initial labels  $(+1, -1)$ . By using the initial labels, the data  $X$  with  $m$  data matrices is divided into two clusters,  $A$  and  $B$ , of size  $(n_1 \times n_2 \times m_1)$  and  $(n_1 \times n_2 \times m_2)$  respectively (where  $m = m_1 + m_2$ ). Each group is then individually partitioned further by considering inter-cluster relationship and is able to generate more stable results in lesser time. Tree-SLSTWSTC is summarized in Algorithm 1.

### 3.3 Initialization

In conventional plane-based clustering scenarios, the initial cluster labels for data are obtained by randomization which is highly unstable and inefficient

**Algorithm 1.** Tree-LSTWSTC

**Input:**  $X$ : Unlabeled data in order-2 tensor space i.e.  $\mathbb{R}^{n_1 \times n_2}$ ;  $K$ : number of cluster;  $\epsilon$ : tolerance level;  $imax$ : maximum number of iterations.

**Output:** Label  $Y_i$  corresponding to each datapoint in  $X_i$ ,  $i = 1, \dots, m$ .

1. Determine the initial labels  $Y_k$ , for each tensor data using approach discussed in Sect. 3.3 for  $K$  clusters.
2. Repeat
  - (a) Create two clusters using tensor based  $k$ -means as  $A_{new}$  and  $B_{new}$ .
  - (b) Select initial  $[u_1^j, v_1^j, b_1^j]$  and  $[u_2^j, v_2^j, b_2^j]$ , where  $j = 0$ .
  - (c) Update the hyperplane parameters to find  $[u_1^{j+1}, v_1^{j+1}, b_1^{j+1}]$  and  $[u_2^{j+1}, v_2^{j+1}, b_2^{j+1}]$ .
  - (d) Stop if  $||[u_i^{j+1}; v_i^{j+1}; b_i^{j+1}] - [u_i^j; v_i^j; b_i^j]|| < \epsilon$ , and then  $u_i^{final} = u_i^{j+1}$ ,  $v_i^{final} = v_i^{j+1}$  and  $b_i^{final} = b_i^{j+1}$ . Otherwise, go to step (c).
3. Use  $Y_k$  to determine if  $A_{new}$  and  $B_{new}$  can be further partitioned i.e. if there are labels from more than one clusters. If required, recursively partition  $A_{new}$  and  $B_{new}$  and goto Step 1.
4. End.

technique. Here, we propose a novel tensor-based initialization algorithms which uses frobenius norm to find the distance between two order-2 tensors (matrices). For example, the distance between two data points  $x^\alpha = x(n_1, n_2, 1)$  and  $x^\beta = x(n_1, n_2, 2)$  is calculated as

$$d(x^\alpha, x^\beta) = \sqrt{\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} (x_{ij}^\alpha - x_{ij}^\beta)^2}. \quad (7)$$

We have implemented Tensor  $k$ -means (Tk-means), which uses tensor data as input and return corresponding cluster labels in the spirit similar to vector based  $k$ -means algorithm. Similar to traditional  $k$ -means, iterative relocation algorithm is followed which minimize the mean squared error locally. Henceforth, the centroid of cluster is updated and the process is repeated until labels converges i.e. no more change in label is detected.

## 4 Experimental Results

To evaluate the performance of the proposed method, experiments were carried out on image dataset of face recognition and optical digit recognition systems. In order to prove competence of our proposed work, we used the Metric accuracy [4] and Learning time as the performance criteria.

For comparison of our proposed approach against other algorithms, we implemented conventional  $k$ -means and  $k$ -Nearest neighbour graph algorithm in tensor framework. Further, to minimize the effect of randomization (in  $k$ -means) and value of  $k$  (in NNG), the experiments were performed multiple times, and the best results are reported.

**Table 1.** Clustering results on face recognition and optical digit recognition application

Dataset	Size	Tensor $k$ -means	Tensor NNG	Tree- SLSTWSTC <sup>+</sup>	Other methods*
	$(n_1 \times n_2 \times m)$	Metric accuracy	Learning time (in sec)		Accuracy
ORL <sup>a</sup>	$(92 \times 112 \times 400)$	91.09 (20.04)	87.93 (38.22)	<b>92.09</b> (14.01)	78.81 [9]
Yale <sup>b</sup>	$(32 \times 32 \times 165)$	88.23 (5.46)	85.02 (0.36)	<b>90.58</b> (2.52)	67.35 [9]
Optical digits <sup>c</sup>	$(32 \times 32 \times 946)$	95.19 (33.34)	92.62 (34.86)	<b>96.19</b> (26.35)	69.12 [8]
MNIST <sup>d</sup>	$(28 \times 28 \times 500)$	88.69 (9.46)	68.07 (6.25)	<b>89.99</b> (5.32)	82.04 [9]

\*State-of-art vector based clustering results.

<sup>a</sup><http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>

<sup>b</sup><http://archive.ics.uci.edu/ml>

<sup>c</sup><http://www.cad.zju.edu.cn/home/dengcai/Data/FaceData.html>

<sup>d</sup><http://www.cs.nyu.edu/~roweis/data.html>

Table 1 summarizes the results of experiments on the above-mentioned datasets. It is clearly evident here that  $k$ -means initialization based Tree-SLSTWSTC outperforms other methods in terms of clustering performance as well as learning time. We have also discussed clustering results obtained from other approaches in Table 1. It can be observed that the prediction accuracy of Tree-SLSTWSTC is significantly better than these methods. Also, it should be noticed that these methods use vector-based representation for clustering.

## 5 Conclusions

Based on the recently proposed LS-TWSTM, in this paper, we have proposed a novel treebased tensor based clustering algorithm namely Treebased Structural Least Squares Twin Support Tensor Clustering (Tree-SLSTWSTC) which has the capability to directly deal with the real world matrix data (second order tensor space) resulting into improved generalization and reduced Computational complexity. Moreover, it also handles the premature convergence problem as it considers structural risk associated with data. For initializing cluster labels, we have proposed Tensor  $k$ -means algorithm which helps to overcome the instability incurred by random initialization. Experimental comparisons of proposed approach against other related approaches on face recognition and handwritten image dataset, establish the suitability of the proposed algorithms to deal with the tensor based data directly (as direct image input).

In future, the application of proposed approach in more challenging real-world applications with higher order tensor space like image segmentation and computer vision can be explored.

## References

1. Khemchandani, R., Pal, A., Chandra, S.: Fuzzy least squares twin support vector clustering. *Neural Comput. Appl.* **1**, 1–11 (2016)
2. Cai, D., He, X., Wen, J.R., Han, J., Ma, W.Y.: Support tensor machines for text categorization (2006)
3. Zhang, X., Gao, X., Wang, Y.: Twin support tensor machines for MCs detection. *J. Electron. (China)* **26**(3), 318–325 (2009)

4. Wang, Z., Shao, Y.H., Bai, L., Deng, N.Y.: Twin support vector machine for clustering. *IEEE Trans. Neural Netw. Learn. Syst.* **26**(10), 2583–2588 (2015)
5. Zhao, X., Shi, H., Lv, M., Jing, L.: Least squares twin support tensor machine for classification. *J. Inf. Comput. Sci.* **11**(12), 4175–4189 (2014)
6. Tao, D., Li, X., Wu, X., Hu, W., Maybank, S.J.: Supervised tensor learning. *Knowl. Inf. Syst.* **13**(1), 142 (2007)
7. Jayadeva, Khemchandani, R., Chandra, S.: Twin support vector machines for pattern classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(5), 905–910 (2007)
8. Wang, X., Yang, C., Zhou, J.: Clustering aggregation by probability accumulation. *Pattern Recogn.* **42**(5), 668–675 (2009)
9. Peng, Y., Zheng, W.L., Lu, B.L.: An unsupervised discriminative extreme learning machine and its applications to data clustering. *Neurocomputing* **174**, 250–264 (2016)