

# Clustering Multi-way Data via Adaptive Subspace Iteration

Wei Peng  
Xerox Innovation Group  
Xerox Corporation  
NY 14580  
wei.peng@xerox.com

Tao Li  
School of Computer Science  
Florida International University  
FL 33199  
taoli@cs.fiu.edu

Bo Shao  
School of Computer Science  
Florida International University  
FL 33199  
bshao001@cs.fiu.edu

## ABSTRACT

Clustering multi-way data is a very important research topic due to the intrinsic rich structures in real-world datasets. In this paper, we propose the subspace clustering algorithm on multi-way data, called **ASI-T** (Adaptive Subspace Iteration on Tensor). ASI-T is a special version of High Order SVD (HOSVD), and it *simultaneously* performs subspace identification using 2DSVD and data clustering using K-Means. The experimental results on synthetic data and real-world data demonstrate the effectiveness of ASI-T.

**Categories and Subject Descriptors:** I.5.3 [Pattern Recognition]: Clustering- Algorithms

**General Terms:** Algorithms, Experimentation, Performance

**Keywords:** Multi-way data, Tensor, Clustering, Subspace

## 1. INTRODUCTION

Multi-way data or tensors can be represented as  $\underline{\mathbf{X}} \in \mathbb{R}^{d_1, d_2, \dots, d_m}$ . When  $m = 3$ ,  $\underline{\mathbf{X}}$  is a three-way data with three modes: data units, features, occasions. The three-way data can be *matricized* to form a flattened matrix. The sum-up matrix is  $(\mathbf{X} = \sum_i(\underline{\mathbf{X}}_i))$ , where  $\underline{\mathbf{X}}_i$  is the  $i$ -th frontal slice. Multi-way data naturally appear in many applications such as webpage personalization and high-order web link analysis [8]. One way to cluster three-way data is to convert the three-way data into two-way matrices, but this approach may result in information loss and fail to capture the underlying structures in three-way datasets [1]. On the other hand, tensor factorization methods can be used to cluster three-way data by discretizing their component matrices. There are generally two types of tensor decomposition models: **Rank-1 Decomposition** [10] and **Tucker Decomposition** including HOSVD and 2DSVD [12, 5]. Many multi-way models can be considered as the extensions or modifications of the above two types.

Despite significant progress made on subspace clustering for two-way data, few attempts have been made to develop subspace clustering algorithms on three-way data. Most tensor factorization models only deal with the subspace selection (data reduction) problems. In this paper, we propose a subspace clustering algorithm on multi-way data via adaptive subspace iteration, called **ASI-T**.

ASI-T model is a *special version* of HOSVD model. We show that the clustering algorithm is also equivalent to K-Means clustering and 2DSVD, and it is a subspace clustering extension on three-way data. More specifically, ASI-T consists of two simultaneous steps: select the subspaces using 2DSVD (identifying the subspace structure in mode 2 and mode 3 of the tensor from the current data clusters) and cluster the three-way data units (mode 1) using K-Means clustering. These two tasks are performed alternatively and iteratively to achieve the stable clustering partitions and subspaces.

## 2. FACTORIZATION MODELS

**Notations** Scalars are denoted by lowercase letters, e.g.  $x$ , and vectors are denoted by boldface lowercase letters, e.g.  $\mathbf{x}$ , where the  $i$ -th entry is  $x_i$ . Matrices are denoted by boldface capital letter, e.g.  $\mathbf{X}$ , where the  $i$ -th row of matrix  $\mathbf{X}$  is  $\mathbf{x}_i$ , the  $j$ -th column of matrix  $\mathbf{X}$  is  $\mathbf{x}_j$ , and the  $(i, j)$ -th entry is  $x_{ij}$ . Three-way arrays (tensors) are denoted by boldface underline letters  $\underline{\mathbf{X}}$ , and the  $(i, j, l)$ -th entry is  $x_{ijl}$ .  $\mathbf{X}_{n_1, n_2, n_3}$  is denoted as the flattened matrix by *matricizing*  $\underline{\mathbf{X}}$  in the first mode. The Kronecker product  $\otimes$  is the operation between two matrices such that Kronecker product of the matrix  $\mathbf{A} \in \mathbb{R}^{a \times b}$  and the matrix  $\mathbf{B} \in \mathbb{R}^{c \times d}$  is the matrix  $\mathbf{C} \in \mathbb{R}^{ac \times bd}$ , where each entry is the product of two entries from  $\mathbf{A}$  and  $\mathbf{B}$  respectively. We present a brief overview of various tensor factorization models.

**Rank-1 Decomposition:** The objective function for Rank-1 decomposition can be written as

$$J_{rank-1} = \sum_{i,j,l} \left( x_{ijl} - \sum_{k=1}^K u_{ik} v_{jk} w_{lk} \right)^2 \quad (1)$$

where  $\mathbf{U} \in \mathbb{R}^{n_1 \times K}$ ,  $\mathbf{V} \in \mathbb{R}^{n_2 \times K}$ ,  $\mathbf{W} \in \mathbb{R}^{n_3 \times K}$ . Rank-1 decomposition is also called Parafac [10] with orthogonality constraints.

**2DSVD:** 2DSVD is an extension of SVD that it approximates each frontal slices of the three-way data  $\underline{\mathbf{X}}$  by minimizing

$$J_{2dsvd} = \sum_{l=1}^{n_3} \|\underline{\mathbf{X}}_l - \mathbf{U} \mathbf{S}_l \mathbf{V}^T\|_F^2 = \sum_{i,j,l} \left( x_{ijl} - \sum_{p,q} s_{pq} u_{ip} v_{jq} \right)^2$$

s.t.  $\mathbf{U}^T \mathbf{U} = \mathbf{I}, \mathbf{V}^T \mathbf{V} = \mathbf{I}.$  (2)

where  $\mathbf{U} \in \mathbb{R}^{n_1 \times k_1}$ ,  $\mathbf{V} \in \mathbb{R}^{n_2 \times k_2}$ , and  $\mathbf{S} \in \mathbb{R}^{k_1 \times k_2 \times n_3}$ .

**HOSVD:** HOSVD (High Order SVD) minimizes

$$J_{hosvd} = \sum_{i,j,l} \left( x_{ikl} - \sum_{p,q,r} s_{pqr} u_{ip} v_{jq} w_{lr} \right)^2$$

s.t.  $\mathbf{U}^T \mathbf{U} = \mathbf{I}, \mathbf{V}^T \mathbf{V} = \mathbf{I}, \mathbf{W}^T \mathbf{W} = \mathbf{I}.$  (3)

where  $\mathbf{U} \in \mathbb{R}^{n_1 \times k_1}$ ,  $\mathbf{V} \in \mathbb{R}^{n_2 \times k_2}$ ,  $\mathbf{W} \in \mathbb{R}^{n_3 \times k_3}$ , and  $\mathbf{S} \in \mathbb{R}^{k_1 \times k_2 \times k_3}$ .

**ASI-T Model:** ASI-T model minimizes

$$J_{asi-t} = \sum_{i,j,l} \left( x_{ijl} - \sum_{p,q,r} y_{pqr} d_{ip} f_{jq} g_{lr} \right)^2. \quad (4)$$

s.t.  $\mathbf{F}^T \mathbf{F} = \mathbf{I}, \mathbf{G}^T \mathbf{G} = \mathbf{I}, \mathbf{D}$  is binary and row stochastic.

$\mathbf{G} \in \mathbb{R}^{n_1 \times k_1}$  and  $\mathbf{F} \in \mathbb{R}^{n_2 \times k_2}$  are two column-wise orthonormal matrices, and  $\mathbf{Y} \in \mathbb{R}^{k_1 \times k_2 \times k_3}$  is a core tensor. Different from HOSVD,  $\mathbf{D}$  is a binary and row-stochastic matrix, i.e., there is only one entry with value 1 in each row, and other entries are all 0s.

### 3. ASI-T MODEL

ASI-T is equivalent to simultaneous K-Means clustering and 2DSVD. The objective function in Eq.(4) of ASI-T can be transformed into

$$J_{asi-t} = \|\mathbf{X}_{n_1, n_2, n_3} - \mathbf{D} \mathbf{Y}_{k_1, k_2, k_3} (\mathbf{G} \otimes \mathbf{F})^T\|_F^2. \quad (5)$$

Let  $\mathbf{Z}_{k_1, n_2, n_3} = \mathbf{Y}_{k_1, k_2, k_3} (\mathbf{G} \otimes \mathbf{F})^T$ , Eq.(5) can be written as

$$J_{asi-t} = \|\mathbf{X}_{n_1, n_2, n_3} - \mathbf{D} \mathbf{Z}_{k_1, n_2, n_3}\|_F^2. \quad (6)$$

Note that Eq.(6) is the same as the K-Means objective function where  $\mathbf{D}$  is the cluster indication matrix (binary and row-stochastic) and  $\mathbf{Z}_{k_1, n_2, n_3}$  is the cluster centroid matrix. Moreover, note  $z_{ijl} = \sum_{p,q} y_{ipq} f_{jq} g_{lr}$ . This leads to a formulation of 2DSVD approximation as shown in Eq.(2). Thus ASI-T clusters data units using K-Means and compresses data using 2DSVD simultaneously. ASI-T can also be viewed as an extension of two-way subspace clustering on three-way data [4, 9]. The derivation is omitted.

We use the alternating least square algorithm to optimize  $\mathbf{D}$ ,  $\mathbf{F}$  and  $\mathbf{G}$  by updating one while fixing the others iteratively until convergence. Let  $\mathbf{M} = \mathbf{D}(\mathbf{D}^T \mathbf{D})^{-1} \mathbf{D}^T$ . Then the objective function can be written as  $\|\mathbf{X}_{n_1, n_2, n_3}\|_F^2 - \|\mathbf{M} \mathbf{X}_{n_1, n_2, n_3} (\mathbf{G} \mathbf{G}^T \otimes \mathbf{F} \mathbf{F}^T)\|_F^2$ . Note that we need to maximize the second part.

**Update F:** We maximize

$$\text{Trace} \left( \mathbf{F}^T \mathbf{X}_{n_2, n_3, n_1} ((\mathbf{M} - \mathbf{I}) \otimes \mathbf{G} \mathbf{G}^T) \mathbf{X}_{n_2, n_3, n_1}^T \mathbf{F} \right).$$

The optimal  $\mathbf{F}$  can be obtained by taking the first  $k_2$  eigenvectors of  $\mathbf{X}_{n_2, n_3, n_1} ((\mathbf{M} - \mathbf{I}) \otimes \mathbf{G} \mathbf{G}^T) \mathbf{X}_{n_2, n_3, n_1}^T$  [6].

**Update G:**  $\mathbf{G}$  can be optimized by maximizing

$$\text{Trace} \left( \mathbf{G}^T \mathbf{X}_{n_3, n_1, n_2} (\mathbf{F} \mathbf{F}^T \otimes (\mathbf{M} - \mathbf{I})) \mathbf{X}_{n_3, n_1, n_2}^T \mathbf{G} \right).$$

**Update D:**  $\mathbf{D}$  is updated by assigning each data unit to a cluster.

### 4. EXPERIMENTS

Five Synthetic data are generated by using the algorithm proposed by Milligan [7] with different configurations. The real datasets are extracted from the DBLP computer science bibliography that can be downloaded at <http://www.informatik.uni-trier.de/~ley/db/>. The data are three-way arrays with the author, term, and year modes. We conduct experiments on two such three-way datasets, one of which is DBLP1000 (1000 authors  $\times$  1000 terms  $\times$  20 years), the other of which is DBLP100 (100 authors  $\times$  200 terms  $\times$  20 years). Normalized Mutual Information (NMI), Adjusted Rand Index (ARI), and Accuracy (ACC) are used as our performance measures. Generally, the larger the values of these measures, the better the clustering performance. The clustering performance of ASI-T is compared with a wide range of clustering algorithms: 2 Tensor factorization methods: (1) **Rank-1** approximation method and (2) **HOSVD**; and 7 Two-way data clustering methods: (3) **KMeans(sum)**: K-Means on the sum-up matrix (authors  $\times$  terms); (4) **KMeans(ext)**: K-Means on the unfolded matrix in the first mode of the three-way

array; (5) **KMeans(pca)**: Perform PCA first on the unfolded matrix in the first mode and then use K-Means algorithm; (6) **InfoCo**: Run information theoretic co-clustering algorithm [3] on the sum-up matrix; (7) **EuclCo**: Run Euclidean co-clustering algorithm [2] on the sum-up matrix; (8) **MinSqCo**: Performs minimum squared residue co-clustering algorithm [2] on the sum-up matrix. (9) **ClusterAgg**: Run K-Means clustering on each frontal slice of the three-way array, and combine them using clustering aggregation [11]. The clustering results are computed by averaging ten runs.

**Table 1: The clustering performance comparison on 2 DBLP datasets among 10 clustering methods.**

Methods	DBLP100			DBLP1000		
	ARI	ACC	NMI	ARI	ACC	NMI
KMeans(sum)	0.523	0.675	0.479	0.157	0.388	0.319
KMeans(ext)	0.106	0.400	0.228	0.004	0.250	0.014
KMeans(pca)	0.312	0.550	0.330	0.199	0.424	0.342
Rank-1	<b>0.664</b>	0.800	0.612	0.154	0.386	0.189
HOSVD	0.416	0.700	0.535	0.187	0.408	0.204
ClusterAgg	0.654	0.815	0.653	0.099	0.305	0.187
InfoCo	0.510	0.775	0.510	0.253	0.415	0.245
EuclCo	0.506	0.675	0.551	0.129	0.361	0.207
MinSqCo	0.351	0.600	0.410	0.218	0.406	0.319
<b>ASI-T</b>	0.657	<b>0.825</b>	<b>0.672</b>	<b>0.415</b>	<b>0.480</b>	<b>0.464</b>

The clustering performance on five synthetic datasets are omitted due to space limitation. We observe that the best performance values are achieved by ASI-T among all these clustering methods with all measures on all synthetic datasets. The experimental results on DBLP datasets are presented in Table 1. We observe that clustering performance of ASI-T on DBLP100 and DBLP1000 is the best. In higher dimensions the advantages of ASI-T become more evident because the three-way subspace clustering of ASI-T can overcome the *curse of dimensionality*. For the same reason, KMeans(pca) and three co-clustering algorithms perform relatively better on DBLP1000 compared to their clustering performance on DBLP100. KMeans(ext) is the worst clustering methods for the curse of dimensionality. Rank-1 approximation and ClusterAgg assume that frontal slices should share the similar patterns or similar clustering results, thus achieve relatively better clustering results on DBLP100 than they are on DBLP1000.

**Acknowledgments:** The work of T. Li is partially supported by NSF under IIS-0546280, HRD-0317692, and IIP-0450552.

### 5. REFERENCES

- [1] E. Acar and B. Yener. Unsupervised multiway data analysis: A literature survey. Technical report, Rensselaer Polytechnic Institute, 2007.
- [2] H. Cho, I. Dhillon, Y. Guan, and S. Sra. Minimum sum squared residue co-clustering of gene expression data. In *Proc. of SIAM Data Mining*, 2004.
- [3] I. S. Dhillon, S. Mallela, and D. S. Modha. Information-theoretical co-clustering. In *SIGKDD*, 2003.
- [4] C. Ding and T. Li. Adaptive dimension reduction using discriminant analysis and k-means clustering. In *ICML*, 2007.
- [5] C. Ding and J. Ye. Two-dimensional singular value decomposition (2DSVD) for 2d maps and images. In *Proc. of SIAM Data Mining*, 2005.
- [6] G. H. Golub and C. F. V. Loan. *Matrix computations (3rd ed.)*. Johns Hopkins University Press, 1996.
- [7] G.W.Milligan. An algorithm for generating artificial test clusters. *Psychometrika*, 50:123–127, 1985.
- [8] T. G. Kolda and B. W. Bader. The tophits model for higher-order web link analysis. In *Workshop on Link Analysis, Counterterrorism and Security*, 2006.
- [9] T. Li, S. Ma, and M. Ogihara. Document clustering via adaptive subspace iteration. In *SIGIR*, 2004.
- [10] R.A.Harshman. Foundations of the parafac procedure: models and conditions for an ‘explanatory’ multi-modal factor analysis. *UCLA working papers in phonetics 16*, pages 1–84, 1970.
- [11] A. Strehl and J. Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *JMLR*, 3:583–617, 2002.
- [12] M. A. O. Vasilescu and D. Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *ECCV*, 2002.