DOWNSAMPLING FOR SPARSE SUBSPACE CLUSTERING

*Xianghui Mao, Xiaohan Wang, and Yuantao Gu**

State Key Laboratory on Microwave and Digital Communications
Tsinghua National Laboratory for Information Science and Technology
Department of Electronic Engineering, Tsinghua University, Beijing 100084, CHINA

ABSTRACT

Sparse subspace clustering (SSC) is a technique to partition unlabeled samples according to the subspaces they locate in. With the rapid increase of data amount, efficiently downsampling a big dataset, while at the same time keeping the structure of subspaces, becomes an important topic for SSC. In order to reduce the computational cost while preserving clustering accuracy, a new approach of SSC with downsampling (SSCD) is proposed in this paper. In SSCD, the numbers of samples located in respective subspaces are estimated utilizing the ℓ_1 norm of the sparse representation. Then a downsampling strategy is designed to decimate samples with the probabilities that are in reverse ratio to the amounts of samples in respective subspaces. As a consequence, the samples in different subspaces are expected to be balanced after the downsampling operation. Theoretical analysis proves the correctness of the proposed strategy. Numerical simulations also verify the efficiency of SSCD.

Index Terms— Downsampling, sparse subspace clustering, unbalanced dataset, ℓ_1 minimization, atomic norm.

1. INTRODUCTION

With the increase of human being's ability on data collection, the scale of datasets need to be processed is expanding dramatically. Efficiently extracting the latent structure of a set of data becomes extremely important. As a consequence, unsupervised learning attracts more and more attention in recent years. As an emerging clustering method, sparse subspace clustering (SSC) [1–3], demonstrates its power in motion segmentation [2], face clustering [3], handwritten digit detection [4]. Since the latent structure of a variety of data can be depicted as a union of subspaces, SSC possesses a wide range of applications. SSC also has great potential for being applied on other large-scale datasets, such as network data, gene series, medical images, etc. Due to high computational complexity, however, efficiently processing large-scale datasets becomes a crucial problem.

Many works have been done to address the above problem. To perform SSC on randomly compressed samples appears in [5] and [6]. Because compression reduces the dimension of ambient signal space, the computational cost on finding self-representation in SSC can be efficiently drawn down.

Nevertheless, high cardinality of the dataset is still a tough problem in almost every step of SSC. Direct downsampling is a ready approach to reduce the overall computations. However, uniform downsampling may lose information of some latent subspaces, especially when the amounts of raw samples are unbalanced among subspaces, which is pervasive in practical datasets. With regard to dealing with the downsampling problem on unbalanced datasets, some works have appeared these years. In order to preserve the geometric property of 3D shapes, [7] solved the problem of downsampling unbalanced dataset by updating the cluster representations and the objective function iteratively. [8] proposed density-dependent downsampling fitting for agglomerate clustering, which were applied in cell analysis. However, the above downsampling methods are usually based on decimating among the nearest neighbors in the sense of Euclidean distance, which do not work in the scenario of SSC.

As far as we know, downsampling for SSC is still an open problem and there is no state-of-the-art solution. In this paper, we tackle this problem for the first time. We expect that in the downsampled dataset, all clusters are preserved and the sample amounts in various clusters are approximately equal to each other. Inspired by the spirit of importance sampling [9] in Monte Carlo simulation, we set proper decimating probability for each sample and endow higher probabilities to the samples in smaller clusters. In such way, a strategy named sparse subspace clustering with downsampling (SSCD) is proposed. Both theoretical analysis and numerical simulations are provided to validate the correctness and efficiency of SSCD.

2. PRELIMINARY

2.1. Problem formulation

Let $\chi = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\}, \mathbf{x}_i \in \mathbb{R}^n, \|\mathbf{x}_i\|_2 = 1, \forall i \in \{1, \cdots, N\}$ denotes a normalized dataset. This dataset can be partitioned into L subsets, $\chi_l = \{\mathbf{x}_{l(1)}, \cdots, \mathbf{x}_{l(N_l)}\}$, where N_l denotes the number of samples in the l^{th} subset, i.e., $\sum_{l=1}^{L} N_l = N$. For each $l \in \{1, \cdots, L\}$, the samples in χ_l lie in a d-dimensional subspace U_l, where the samples are drawn independently and uniformly. Given the dataset, the problem studied in this paper is to recover all the subspaces. It is assumed that N_ls are large numbers, and all subspaces are well separated, that is, subspace detection property [10] holds. Furthermore, we consider clustering clean datasets, i.e., without noise or outliers.

2.2. Sparse Subspace Clustering

SSC was first proposed in [1]. It can be roughly divided into two steps. In the first step, the sparse representation of each sample $\mathbf{x}_i$ is estimated by solving an ℓ_1 minimization problem,

$$\hat{\mathbf{c}}_i = \arg\min \|\mathbf{c}_i\|_1, \qquad \text{s.t.} \quad \mathbf{x}_i = \mathbf{X}\mathbf{c}_i, \quad c_i(i) = 0, \qquad (1)$$

where $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_N] \in \mathbb{R}^{n \times N}$, $\mathbf{c}_i \in \mathbb{R}^N$.In the second step, spectral clustering [11] is adopted to partition the samples into clusters by processing a similarity graph $\mathcal{G}$, whose adjacency matrix is

This work was supported by National 973 Program of China (Grant No. 2013CB329203) and National Natural Science Foundation of China (NSFC 61371137). The corresponding author of this paper is Yuantao Gu (gyt@tsinghua.edu.cn).

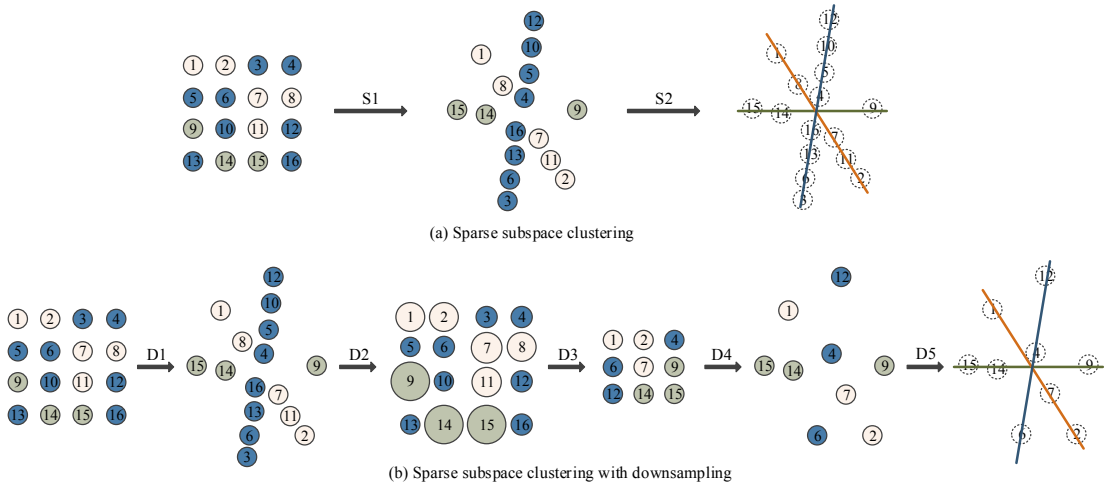

Fig. 1. An intuitive explanation of SSC and SSCD is provided by clustering 16 samples, denoted by circled numbers, into 3 subspaces, distinguished by different colors. (a) There are two steps included in SSC. S1: sparse self-representation is estimated to discover the relation among samples. S2: spectral clustering [11] is applied on the similarity graph to partition samples and then the structure of subspaces are extracted. (b) SSCD is proposed to improve the performance on large-scale unbalanced datasets. D1: sparse self-representation is estimated, where this step is exactly the same as S1 in SSC. D2: the probability of each sample to be downsampled is determined by its sparse representation, where the circle's area denotes the respective probability. D3: the original dataset is downsampled to produce a small and balanced subset according to the downsampling probability of respective samples. D4: new sparse self-representation is estimated among the balanced dataset. D5: Applying spectral clustering on the new similarity graph to find the structure of subspaces, where this step is similar to S2 in SSC but the former is executed on a small and balanced dataset.

given by $\mathbf{A} = |\hat{\mathbf{C}}| + |\hat{\mathbf{C}}|^{\mathrm{T}}$, where $\hat{\mathbf{C}} = [\hat{\mathbf{c}}_1, \cdots, \hat{\mathbf{c}}_N] \in \mathbb{R}^{N \times N}$ and $|\cdot|$ takes the elementwise absolute value. Entries of $\mathbf{A}$ denote similarity between corresponding pairs of samples. An intuitive description of SSC is demonstrated in Fig.1(a).

2.3. Atomic Norm and Polytope's Volume

Atomic norm, which is equivalent to Minkowski functional [12], plays an essential role in connecting the sparse representation to the sample amount.

Definition 1 ([13]). *For a bounded and symmetric atom set $\mathcal{A} = \{\pm\mathbf{a}_1, \pm\mathbf{a}_2, \cdots, \pm\mathbf{a}_m, \mathbf{a}_i \in \mathbb{R}^n, \forall i \in \{1, 2, \cdots, m\}\}$, where m is an positive integer, the atomic norm of a vector $\mathbf{x} \in \mathbb{R}^n$ is defined as*

$$\|\mathbf{x}\|_{\mathcal{A}} = \inf_t \{\mathbf{x} \in t \cdot \mathrm{conv}(\mathcal{A})\}, \tag{2}$$

where $\mathrm{conv}(\mathcal{A})$ denotes the convex hull of atoms in $\mathcal{A}$.

Lemma 1 ([10]). *When strong duality holds for the linear programming problem*

$$\hat{\mathbf{x}} = \arg\min \|\mathbf{x}\|_1, \qquad \text{s.t.} \quad \mathbf{y} = \mathbf{\Phi}\mathbf{x}, \tag{3}$$

it satisfies that

$$\|\hat{\mathbf{x}}\|_1 = \|\mathbf{y}\|_{\mathcal{A}}, \tag{4}$$

where $\mathcal{A}$ is composed of the columns of $\mathbf{\Phi}$ and $-\mathbf{\Phi}$.

The volume of a high-dimensional random inscribed polytope is estimated in Lemma 2.

Lemma 2 ([14]). *Assuming that $B^{(d)}$ is the Euclidean ball in $\mathbb{R}^d$, and A_N is a random inscribed polytope with all of its N vertices independently uniformly chosen on the boundary of $B^{(d)}$, i.e., S^{d-1}, one has*

$$\mathbb{E}\{\mathrm{Vol}_d(A_N)\} = \mathrm{Vol}_d(B^{(d)}) - (c_{B^{(d)}} + o(1))N^{-\frac{2}{d-1}}, \tag{5}$$

where $\mathrm{Vol}_d(\cdot)$ denotes the d-dimensional volume and $c_{B^{(d)}}$ is a constant depending on d.

3. SPARSE SUBSPACE CLUSTERING WITH DOWNSAMPLING

In order to handle the large-scale unbalanced dataset for SSC, we propose a downsampling technique, which is visualized in Fig.1(b). Compared with SSC in Fig.1(a), the proposed SSCD performs subspace clustering operations (generating similarity graph, performing spectral clustering, and recovering subspace structure) on the downsampled dataset (D4, D5) rather than on the set of raw data. We illustrate in detail the steps of SSCD in Algorithm 1. Processing well downsampled dataset largely reduces the clustering time and preserves high subspace recovery accuracy as well. Therefore downsampling (D1, D2, D3) plays an important role in SSCD, which simultaneously downscales the dataset and balances the sample amounts among all subspaces.

We first demonstrate the basic idea of downsampling. The self-representation (D1) of SSCD is exactly the same with that of SSC (S1), which is mainly based on the property that samples lying in the same subspace could be mutually represented. When the subspace is of low dimensionality, the representation vectors are sparse. Though the first steps in SSC and SSCD are the same, their purposes are quite different. While SSC builds a similarity graph utilizing the representation vectors, SSCD determines the downsampling probabilities from the ℓ_1 norm of the vectors (D2). As a consequence, the downsampled dataset is accordingly chosen (D3).

Algorithm 1 The Procedure of SSCD

Input: A dataset matrix $\mathbf{X} \in \mathbb{R}^{n\times N}$, of which the ith column denotes a sample $\mathbf{x}_i, i \in \{1, \cdots, N\}$; dimension of subspace d; cardinality of the downsampled subset M.

Output: Orthonormal bases of each subspace.

Downsampling:

D1: Estimate the sparse representation for $\mathbf{x}_i$ by $\hat{\mathbf{c}}_i = \arg\min \|\mathbf{c}_i\|_1, \quad \text{s.t.} \quad \mathbf{x}_i = \mathbf{X}\mathbf{c}_i, c_i(i) = 0;$

D2: For all $i \in \{1, \cdots, N\}$, calculate the *weight* of $\mathbf{x}_i$ $w_i = \left(1 - 1/\|\hat{\mathbf{c}}_i\|_1^d\right)^{\frac{d-1}{2}}$, and normalize it to the downsampling probability $p_i = w_i / \sum_{j=1}^N w_j$;

D3: Randomly choose M different samples from $\{\mathbf{x}_i\}_{i=1}^N$ according to the probability $\{p_i\}_{i=1}^N$ and construct a new dataset $\{\mathbf{x}'_i\}_{i=1}^M$;

Sparse Subspace Clustering:

D4: Estimate the sparse representation for all $\mathbf{x}'_i$;

D5: Construct an M-node similarity graph $\mathcal{G}$. Then apply spectral clustering to obtain the partition and perform Principal Component Analysis (PCA) on each cluster, the first d components constitute an approximated orthonormal bases of the corresponding subspace.

The core problem in the downsampling strategy is to set the downsampling probabilities (D2). Inspired by the spirit of rare event simulation [15], we assign the samples in smaller clusters with larger weights. According to Borel's law of large numbers [16], a good choice is to balance the sum of downsampling probabilities among different clusters. Notice that the subspace detection property [10] can be preserved by properly setting the cardinality of the downsampled dataset M, and we assume that $M \ll N$. By setting the probability distribution as

$$p_i = \frac{1}{LN_{l_i}}, \quad l_i \in \{1, \cdots, L\}, i \in \{1, \cdots, N\}, \tag{6}$$

where l_i indexes the subspace $\mathbf{x}_i$ lies in, one can expect that in the downsampled dataset the amounts of samples in different subspaces are balanced. Therefore, the problem of setting the downsampling probabilities turns to estimating the amount of samples lying in respective subspaces. Based on Proposition 1, by setting the weights as in D2 of Algorithm 1, we draw the downsampling probability approximating (6). In such way, the downsampled dataset is approximately subspace-balanced.

Proposition 1. *In the scenario of SSC described in section 2.2, the amount of samples lying in the same subspace with $\mathbf{x}_i$ is approximately in inverse proportion to $w_i \triangleq \left(1 - 1/\|\hat{\mathbf{c}}_i\|_1^d\right)^{\frac{d-1}{2}}$, where $\hat{\mathbf{c}}_i$ is the optimal solution as denoted in (1).*

4. PROOF OF PROPOSITION 1

According to the self-representation step in SSC, one may define $\mathcal{A}_i^l = \{\pm\mathbf{x}_{i_l(1)}, \cdots, \pm\mathbf{x}_{i_l(N_l)}\}\backslash\{\pm\mathbf{x}_i\}, i \in \{1, \cdots, N\}, l \in \{1, \cdots, L\}, \mathbf{x}_i \in U_l$, which is a bounded and symmetric atomic set generated by the samples in the l^{th} subspace except $\mathbf{x}_i$. Utilizing Lemma 1, it can be readily accepted that

$$\|\hat{\mathbf{c}}_i\|_1 = \|\mathbf{x}_i\|_{\mathcal{A}_i^l}, \tag{7}$$

where $\hat{\mathbf{c}}_i$ is given in (1). The assumption that the subspaces are well separated is adopted to derive (7). Since the samples are normalized, i.e. $\|\mathbf{x}_i\|_2 = 1$, one may utilize $1/\|\mathbf{x}_i\|_{\mathcal{A}_i^l}$ to denote the Euclidean length of a section of $\mathbf{x}_i$ that is contained in the polytope $\text{conv}(\mathcal{A}_i^l)$. Based on geometric intuition, we come up with Lemma 3.

Lemma 3. *For a deterministic polytope $\text{conv}(\mathcal{A})$ with all vertices fixed on the boundary of S^{d-1} and a random variable $\mathbf{x} \in \mathbb{R}^d$ satisfies a uniform distribution on S^{d-1}, one has*

$$\mathbb{E}\left\{\frac{1}{\|\mathbf{x}\|_{\mathcal{A}}^d}\right\} = \frac{\text{Vol}_d(\text{conv}(\mathcal{A}))}{\text{Vol}_d(B^{(d)})}, \tag{8}$$

where $B^{(d)} = \{\mathbf{x} \in \mathbb{R}^d | \|\mathbf{x}\|_2 \le 1\}$.

Proof. The ratio of volume element can be calculated as

$$\frac{\text{dVol}_d(\text{conv}(\mathcal{A}))}{\text{dVol}_d(B^{(d)})}\text{d}\mu = \frac{1}{\|\mathbf{x}\|_{\mathcal{A}}^d}\text{d}\mu. \tag{9}$$

where μ denotes a Haar measure defined on S^{d-1} [17]. In consequence, one has

$$\mathbb{E}\left\{\frac{1}{\|\mathbf{x}\|_{\mathcal{A}}^d}\right\} = \int_{S^{d-1}} \frac{1}{\|\mathbf{x}\|_{\mathcal{A}}^d}\text{d}\mu = \frac{\text{Vol}_d(\text{conv}(\mathcal{A}))}{\text{Vol}_d(B^{(d)})}.$$

□

By applying Lemma 3 in the scenario of SSC, it comes directly that

$$\mathbb{E}\left\{\frac{1}{\|\mathbf{x}_i\|_{\mathcal{A}_i^l}^d}\middle|\mathcal{A}_i^l\right\} = \frac{\text{Vol}_d(\text{conv}(\mathcal{A}_i^l))}{\text{Vol}_d(B^{(d)})}. \tag{10}$$

Considering the randomness of samples on U_l, one may take the expectation of both sides of (10) with respect to $\mathcal{A}_i^l$ and get

$$\mathbb{E}\left\{\frac{1}{\|\mathbf{x}_i\|_{\mathcal{A}_i^l}^d}\right\} = \frac{\mathbb{E}\left\{\text{Vol}_d(\text{conv}(\mathcal{A}_i^l))\right\}}{\text{Vol}_d(B^{(d)})}. \tag{11}$$

Applying Lemma 2 on the volume of high-dimensional random polytope in (11), one may readily arrive at Proposition 2.

Proposition 2. *The number of samples in the l^{th} subspace, N_l, can be approximated by*

$$\hat{N}_l = C_d\left(1 - \mathbb{E}\left\{\frac{1}{\|\hat{\mathbf{c}}_i\|_1^d}\middle|\mathbf{x}_i \in U_l\right\}\right)^{-\frac{d-1}{2}}, \tag{12}$$

where $C_d = \frac{1}{2}\left(\frac{\text{Vol}_d(B^{(d)})}{c_{B^{(d)}}+o(1)}\right)^{-\frac{d-1}{2}}$ is a constant only depending on d and $c_{B^{(d)}}$ is the same as in Lemma 2.

Proof. In the scenario of SSC, $2N_l - 2$ samples in U_l construct an atomic set of $\mathcal{A}_i^l$ to represent $\mathbf{x}_i$, where half atoms in the atomic set follow uniform distribution and the rest ones are determined accordingly to satisfy symmetry. By relaxing the dependence of the two halves of atoms, one may approximate $\mathcal{A}_i^l$ by $\tilde{\mathcal{A}}_i^l$, whose $2N_l - 2$ atoms are drawn independently in S^{d-1} by uniform distribution. Adopting Lemma 3, as a consequence, one has

$$\begin{aligned}&\mathbb{E}\left\{\text{Vol}_d(\text{conv}(\mathcal{A}_i^l))\right\} \approx \mathbb{E}\left\{\text{Vol}_d(\text{conv}(\tilde{\mathcal{A}}_i^l))\right\}\\ =&\text{Vol}_d(B^{(d)}) - (c_{B^{(d)}} + o(1))(2N_l - 2)^{-\frac{2}{d-1}}.\end{aligned} \tag{13}$$

Utilizing (7) and (13) in (11), one may readily arrive at

$$\mathbb{E}\left\{\frac{1}{\|\hat{\mathbf{c}}_i\|_1^d}\middle|\mathbf{x}_i\in U_l\right\}\approx 1-\frac{(c_{B^{(d)}}+o(1))(2N_l-2)^{-\frac{2}{d-1}}}{\mathrm{Vol}_d(B^{(d)})}, \tag{14}$$

which is finally rewritten to (12) to provide an estimate of N_l by utilizing ℓ_1 norm of sparse representation coefficients. □

Because it is assumed that all subspaces share the same dimension, the constant C_d in (12) can be eliminated when estimating the ratio of sample amounts in various subspaces. Furthermore, the random volume of hyper-dimensional inscribed polytope is concentrated around its expectation with high probability, which is a main result in [18]. In consequence, we approximate (12) by a feasible form

$$\tilde{N}_l = C_d\left(1-\frac{1}{\|\hat{\mathbf{c}}_i\|_1^d}\right)^{-\frac{d-1}{2}}, \tag{15}$$

and finally prove Proposition 1.

5. NUMERICAL RESULTS

Two propositions are tested in the first experiment. In order to verify Proposition 1, three 5-dimensional subspaces are generated in 100-dimensional ambient space. $500, 100$, and 30 samples are independently uniformly chosen in respective subspaces and then normalized. The *weight* w_i associated with each sample is plotted in the left subplot of Fig.2. To testify Proposition 2, a d-dimensional subspace is generated in 50-dimensional ambient space and N_l samples are independently uniformly chosen from this subspace. The number d and N_l vary from 5 to 10, and from 15 to 215, respectively. The average of $\tilde{N}_l/C_d$ (15) in 150 random trials are plotted in the right subplot of Fig. 2. One may easily read from Fig.2 that the sample amounts in a subspace are in good linearity with its estimate, i.e., the right hand side of (12), which verifies Proposition 2. Meanwhile, the scatterplot generated from one trial demonstrates that estimating the sample amount in the same subspace by $1/w_i$ in Proposition 1 is rather good, especially when the sample amount is large.

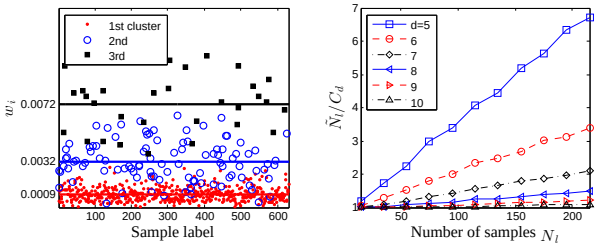

Fig. 2. The left subplot demonstrates w_i in one trial, where the solid lines denote the means of w_i in respective clusters. The right subplot depicts that the average of $\tilde{N}_l/C_d$ is in good linearity with N_l.

In the second experiment, the proposed SSCD is compared with SSC with uniformly downsampling (SSCU). The experiment setting is the same with that in the first experiment of verifying Proposition 1. The performance of SSCD is tested by varying the downsampling rate M/N from 1% to 30%. 100 trials are taken and the results are demonstrated in Fig.3, where subspace recovery accuracy is measured with the average of subspace affinities [10] between the estimated subspaces and the original ones. One may read from Fig.3 that in this scenario decimating only 10% samples of the original dataset are enough for SSCD to recover the structures of all three subspaces, while for SSCU at least 30% samples are required. The gap between the dashed lines in the right subplot of Fig.3 demonstrates the advantage of SSCD over SSCU in processing time when at least 99% subspace recovery accuracy is achieved.

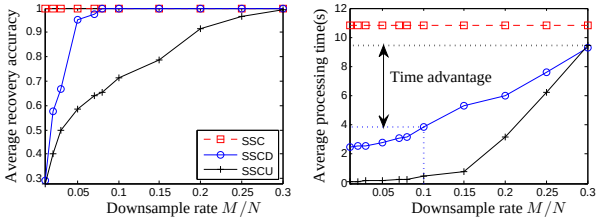

Fig. 3. The left subplot compares the recovery accuracy of SSC, SSCD, and SSCU. The right subplot displays the processing time of these three methods. In our experiment, ℓ_1 homotopy method [19] is applied in (S1, D1, D4) steps, random walk spectral clustering with eigengap heuristic [11] is performed in (S2) and (D5) steps.

In order to illustrate the efficiency of the downsampling step in SSCD, the similarity graphs in one trial are displayed in Fig.4. The color of pixels represents the similarity between samples, i.e., the entries of $\mathbf{A}$, where the blue color represents zero. The yellow squares in each subplot show the pattern of three clusters. From Fig.4, one may read that the similarity matrix generated in SSC is large and sparse. In consequence, directly performing spectral clustering and subspace recovery on it consumes much time. Furthermore, the sparsely connected components are usually difficult to cluster [20]. Uniformly downsampling can solve this problem, however, it may lose small clusters. As stated in the text, the proposed SSCD is free of the above problems by properly downsampling and producing dense and balanced clusters.

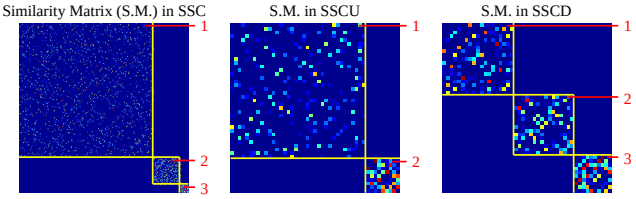

Fig. 4. The similarity matrices generated in SSC (left), SSCU (middle), and SSCD (right). Both downsampled dataset are of cardinality 45. For ease of understanding, we use yellow squares to label the edges of diagonal blocks and enlarge the area of each nonzero entries in the left subplot by a factor of 9.

6. CONCLUSION AND FUTURE WORK

We propose a downsampling strategy for sparse subspace clustering for the first time, and testify its efficiency with numerical experiments based on synthetic data. This proposed downsampling strategy, which is built on high-dimensional geometry and importance sampling, is verified to be capable of balancing the sizes of clusters, reducing computational cost, and recovering the structure of subspaces with high accuracy. Possible future works include studying the downsampling strategy for noisy dataset and designing general downsampling strategy for various scenarios.

7. REFERENCES

[1] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 2790–2797.

[2] M. Soltanolkotabi, E. Elhamifar, E. J. Candes *et al.*, "Robust subspace clustering," *The Annals of Statistics*, vol. 42, no. 2, pp. 669–699, 2014.

[3] E. L. Dyer, A. C. Sankaranarayanan, and R. G. Baraniuk, "Greedy feature selection for subspace clustering," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2487–2517, 2013.

[4] R. Heckel, E. Agustsson, and H. Bölcskei, "Neighborhood selection for thresholding-based subspace clustering," in *IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP)*, 2014, pp. 6761–6765.

[5] R. Heckel, M. Tschannen, and H. Bölcskei, "Subspace clustering of dimensionality-reduced data," in *IEEE International Symposium on Information Theory*, 2014, pp. 2997–3001.

[6] X. Mao and Y. Gu, "Compressed subspace clusteirng: A case study," in *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2014, pp. 616–620.

[7] H. Song and H.-Y. Feng, "A global clustering approach to point cloud simplification with a specified data reduction ratio," *Computer-Aided Design*, vol. 40, no. 3, pp. 281–292, 2008.

[8] P. Qiu, E. F. Simonds, S. C. Bendall, K. D. Gibbs Jr, R. V. Bruggner, M. D. Linderman, K. Sachs, G. P. Nolan, and S. K. Plevritis, "Extracting a cellular hierarchy from high-dimensional cytometry data with spade," *Nature biotechnology*, vol. 29, no. 10, pp. 886–891, 2011.

[9] J. Bucklew, *Introduction to rare event simulation*. Springer, 2004.

[10] M. Soltanolkotabi, E. J. Candes *et al.*, "A geometric analysis of subspace clustering with outliers," *The Annals of Statistics*, vol. 40, no. 4, pp. 2195–2238, 2012.

[11] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[12] H. Minkowski, "Volumen und oberfläche," in *Ausgewählte Arbeiten zur Zahlentheorie und zur Geometrie*. Springer, 1989, pp. 146–192.

[13] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A. S. Willsky, "The convex geometry of linear inverse problems," *Foundations of Computational Mathematics*, vol. 12, no. 6, pp. 805–849, 2012.

[14] E. W. Carsten Schütt, *Polytopes with Vertices Chosen Randomly from the Boundary of a Convex Body*. Geometric Aspects of Functional Analysis, 2003.

[15] P. L'Ecuyer, M. Mandjes, B. Tuffin *et al.*, "Importance sampling and rare event simulation," *Rare event simulation using Monte Carlo methods*, pp. 17–38, 2009.

[16] L. Wen, "An analytic technique to prove borel's strong law of large numbers," *American Mathematical Monthly*, pp. 146–148, 1991.

[17] I. Navernioukk, I. Sutskever, and H. Hatami, "Metric embeddings lecture 5: Dimension reduction," in *http://www.cs.toronto.edu/∼avner/teaching/S6-2414/LN5.pdf*.

[18] L. Wu, "Random inscribed polytopes," Ph.D. dissertation, UC San Diego.

[19] M. Salman Asif and J. Romberg, "Dynamic updating for ℓ_1 minimization," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, pp. 421–434, 2010.

[20] E. L. Dyer, C. Studer, and R. G. Baraniuk, "Subspace clustering with dense representations," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 3258–3262.