2018

Subspace Clustering Using Ensembles of K-Subspaces

John Lipor
Portland State University, lipor@pdx.edu

David Hong
University of Michigan

Yan Shuo Tan
University of California, Berkeley

Laura Balzano
University of Michigan

Citation Details

Lipor, John; Hong, David; Tan, Yan Shuo; and Balzano, Laura, "Subspace Clustering Using Ensembles of K-Subspaces" (2018). *Electrical and Computer Engineering Faculty Publications and Presentations*. 499.
https://pdxscholar.library.pdx.edu/ece_fac/499

Subspace Clustering using Ensembles of K-Subspaces

John Lipor[1], David Hong[2]*, Yan Shuo Tan[3], and Laura Balzano[2]

[1]Department of Electrical & Computer Engineering, Portland State University
[2]Department of Electrical Engineering & Computer Science, University of Michigan
[3]Foundations of Data Analysis Institute, University of California, Berkeley
lipor@pdx.edu, dahong@umich.edu, yanshuo@berkeley.edu, girasole@umich.edu

Abstract

Subspace clustering is the unsupervised grouping of points lying near a union of low-dimensional linear subspaces. Algorithms based directly on geometric properties of such data tend to either provide poor empirical performance, lack theoretical guarantees, or depend heavily on their initialization. We present a novel geometric approach to the subspace clustering problem that leverages ensembles of the K-subspaces (KSS) algorithm via the evidence accumulation clustering framework. Our algorithm, referred to as ensemble K-subspaces (EKSS), forms a co-association matrix whose (i, j)th entry is the number of times points i and j are clustered together by several runs of KSS with random initializations. We prove general recovery guarantees for any algorithm that forms an affinity matrix with entries close to a monotonic transformation of pairwise absolute inner products. We then show that a specific instance of EKSS results in an affinity matrix with entries of this form, and hence our proposed algorithm can provably recover subspaces under similar conditions to state-of-the-art algorithms. The finding is, to the best of our knowledge, the first recovery guarantee for evidence accumulation clustering and for KSS variants. We show on synthetic data that our method performs well in the traditionally challenging settings of subspaces with large intersection, subspaces with small principal angles, and noisy data. Finally, we evaluate our algorithm on six common benchmark datasets and show that unlike existing methods, EKSS achieves excellent empirical performance when there are both a small and large number of points per subspace.

1 Introduction

In modern computer vision problems such as face recognition [1] and object tracking [2], researchers have found success applying the union of subspaces (UoS) model, in which data vectors lie near one of several low-rank subspaces. This model can be viewed as a generalization of principal component analysis (PCA) to the case of multiple subspaces, or alternatively a generalization of clustering models where the clusters have low-rank structure. The modeling goal is therefore to simultaneously identify these underlying subspaces and cluster the points according to their nearest subspace. Algorithms designed for this task are called *subspace clustering* algorithms. This topic has received a great deal of attention in recent years [3] due to various algorithms' efficacy on real-world problems such as face recognition [4], handwritten digit recognition [5], and motion segmentation [2].

Algorithms for subspace clustering can be divided into geometric methods [6–13], which perform clustering by directly utilizing the properties of data lying on a UoS, and self-expressive methods [14–18], which leverage the fact that points lying on a UoS can be efficiently represented by other points in the same subspace. For many geometric methods, the inner product between points is a fundamental tool used in algorithm design and theoretical analysis. In particular, the observation that the inner product between points on the same subspace is often greater than that between points on different subspaces plays a key role. This idea motivates the thresholded subspace clustering (TSC) algorithm [11], appears in the recovery guarantees of the conic subspace clustering algorithm [12], and has been shown to be an effective method of outlier rejection in both

[*] John Lipor and David Hong made equal contributions to this work.

robust PCA [19] and subspace clustering [13]. However, despite directly leveraging the UoS structure in the data, geometric methods tend to either exhibit poor empirical performance, lack recovery guarantees, or depend heavily on their initialization.

In this work, we aim to overcome these issues through a set of general recovery guarantees as well as a novel geometric algorithm that achieves state-of-the-art performance across a variety of benchmark datasets. We develop recovery guarantees that match the state-of-the-art and apply to *any* algorithm that builds an affinity matrix A with entries close to a monotonic transformation of pairwise absolute inner products, i.e., for which

$$|A_{i,j} - f\left(|\langle x_i, x_j \rangle|\right)| < \tau, \tag{1}$$

where f is a monotonic function, x_i, x_j are data points, and $\tau > 0$ is the maximum deviation. Such affinity matrices arise in settings where only approximate inner products are practically available (e.g., dimensionality-reduced data), as well as in settings where deviating from pairwise inner products produces better empirical performance (e.g., by incorporating higher-order structure). We propose the ensemble K-subspaces (EKSS) algorithm, which builds its affinity matrix by combining the outputs of many instances of the well-known K-subspaces (KSS) algorithm [6,8] via the *evidence accumulation* clustering framework [20]. We show that the affinity matrix obtained from the first iteration of KSS fits the observation model (1) and consequently enjoys strong theoretical guarantees. To the best of our knowledge, these results are the first theoretical guarantees characterizing an affinity matrix resulting from evidence accumulation, as well as the first recovery guarantees for any variant of the KSS algorithm. Finally, we demonstrate that EKSS achieves excellent empirical performance on several canonical benchmark datasets.

The remainder of this paper is organized as follows. In Section 2 we define the subspace clustering problem in detail and give an overview of the related work. In Section 3 we propose the Ensemble K-Subspaces algorithm. Section 4 contains the theoretical contributions of this paper. We demonstrate the strong empirical performance of EKSS on a variety of datasets in Section 5. Conclusions and future work are described in Section 6.

2 Problem Formulation & Related Work

Consider a collection of points $\mathcal{X} = \{x_1, \ldots, x_N\}$ in $\mathbb{R}^D$ lying near a union of K subspaces $\mathcal{S}_1, \ldots, \mathcal{S}_K$ having dimensions $d_1, \ldots, d_K$. Let $X \in \mathbb{R}^{D \times N}$ denote the matrix whose columns are the points in $\mathcal{X}$. The goal of subspace clustering is to label points in the unknown union of K subspaces according to their nearest subspace. Once the clusters have been obtained, the corresponding subspace bases can be recovered using principal components analysis (PCA).

2.1 Subspace Clustering

Most state-of-the-art approaches to subspace clustering rely on a *self-expressive* property of the data, which informally states that points in the UoS model can be most efficiently represented by other points within the same subspace. These methods typically use a self-expressive data cost function that is regularized to encourage efficient representation as follows:

$$\begin{aligned} \min_{Z} \quad & \|X - XZ\|_F^2 + \lambda \|Z\| \\ \text{subject to} \quad & \operatorname{diag}(Z) = 0, \end{aligned} \tag{2}$$

where λ balances the regression and penalization terms and $\|Z\|$ may be the 1-norm as in sparse subspace clustering (SSC) [16], nuclear norm as in low-rank representation (which omits the constraint on Z) [14], or a combination of these and other norms. An affinity/similarity matrix is then obtained as $|Z| + |Z|^T$, after which spectral clustering is performed. Other terms are considered in the optimization problem to provide robustness to noise and outliers, and numerous recent papers follow this framework [15, 21–23]. For large datasets, solving the above problem may be prohibitive, and algorithms such as [17, 18] employ orthogonal matching pursuit and elastic-net formulations to provide reduced computational complexity and improved connectivity. These algorithms are typically accompanied by theoretical results that guarantee no false connections (NFC), i.e., that points lying in different subspaces have zero affinity. These guarantees depend on a notion of distance between subspaces called the *subspace affinity* (9). Roughly stated, the closer any pair of underlying subspaces is, the more difficult the subspace clustering problem becomes. An excellent overview of these results is given in [24].

Aside from the self-expressive methods above, a number of geometric approaches have also been considered in the past. Broadly speaking, these methods all determine a set of q "nearest neighbors" for each point that are used to build an affinity matrix, with labels obtained via spectral clustering. An early example of this type of algorithm is the Spectral Local Best-Fit Flats (SLBF) algorithm [25], in which neighbors are selected in terms of Euclidean distance, with the optimal number of neighbors estimated via the introduced local best-fit heuristic. While this heuristic is theoretically motivated, no clustering guarantees accompany this approach, and its performance on benchmark datasets lags significantly behind that of self-expressive methods. The greedy subspace clustering (GSC) algorithm [10] greedily builds subspaces by adding points with largest projection in order to form an affinity matrix, with the number of neighbors fixed. This algorithm has strong theoretical guarantees, and while its performance is still competitive, it lags behind that of self-expressive methods. Thresholded subspace clustering (TSC) [11] chooses neighbors based on the largest absolute inner product, and the authors prove that this simple approach obtains correct clustering under assumptions similar to those considered in the analysis of SSC. However, empirical results show that TSC performs poorly on a number of benchmark datasets. Our proposed algorithm possesses the same theoretical guarantees of TSC while also achieving excellent empirical performance.

In contrast to the above methods, the K-subspaces (KSS) algorithm [6, 8] seeks to minimize the sum of residuals of points to their assigned subspace, i.e.,

$$\min_{\mathcal{C},\mathcal{U}} \sum_{k=1}^{K} \sum_{i:x_i \in c_k} \left\| x_i - U_k U_k^T x_i \right\|_2^2, \tag{3}$$

where $\mathcal{C} = \{c_1, \ldots, c_K\}$ denotes the set of estimated clusters and $\mathcal{U} = \{U_1, \ldots, U_K\}$ denotes the corresponding set of orthonormal subspace bases. We claim that this is a "natural" choice of objective function for the subspace clustering problem since its value is zero if a perfect UoS fit is obtained. Further, in the case of noiseless data, the optimal solution to (3) does not depend on how close any pair of subspaces is, indicating that a global solution to (3) may be more robust than other objectives to subspaces with high affinity. However, (3) was recently shown to be even more difficult to solve than the K-means problem in the sense that it is NP-hard to *approximate* within a constant factor [13] in the worst case. As a result, researchers have turned to the use of alternating algorithms to obtain an approximate solution. Beginning with an initialization of K candidate subspace bases, KSS alternates between (i) clustering points by nearest subspace and (ii) obtaining new subspace bases by performing PCA on the points in each cluster. The algorithm is computationally efficient and guaranteed to converge to a local minimum [6, 7], but as with K-means, the KSS output is highly dependent on initialization. It is typically applied by performing many restarts and choosing the result with minimum cost (3) as the output. This idea was extended to minimize the median residual (as opposed to mean) in [9], where a heuristic for intelligent initialization is also proposed. In [26], the authors use an alternating method based on KSS to perform online subspace clustering in the case of missing data. In [27], the authors propose a novel initialization method based on ideas from [25], and then perform the subspace update step using gradient steps along the Grassmann manifold. While this method is computationally efficient and improves upon the previous performance of KSS, it lacks theoretical guarantees. Most recently, the authors of [13] show that the subspace estimation step in KSS can be cast as a robust subspace recovery problem that can be efficiently solved using the Coherence Pursuit (CoP) algorithm [19]. The authors motivate the use of CoP by proving that it is capable of rejecting outliers from a UoS and demonstrate that replacing PCA with CoP results in strong empirical performance when there are many points per subspace. However, performance is limited when there are few points per subspace, and the algorithm performance is still highly dependent on the initialization. Moreover, CoP can be easily integrated into our proposed algorithm to provide improved performance.

Our method is based on the observation that the partially-correct clustering information from each random initialization of KSS can be leveraged using *consensus clustering* in such a way that the consensus is much more informative than even the best single run. Unlike the above-mentioned variations on KSS, our proposed approach has cluster recovery guarantees, and its empirical performance is significantly stronger.

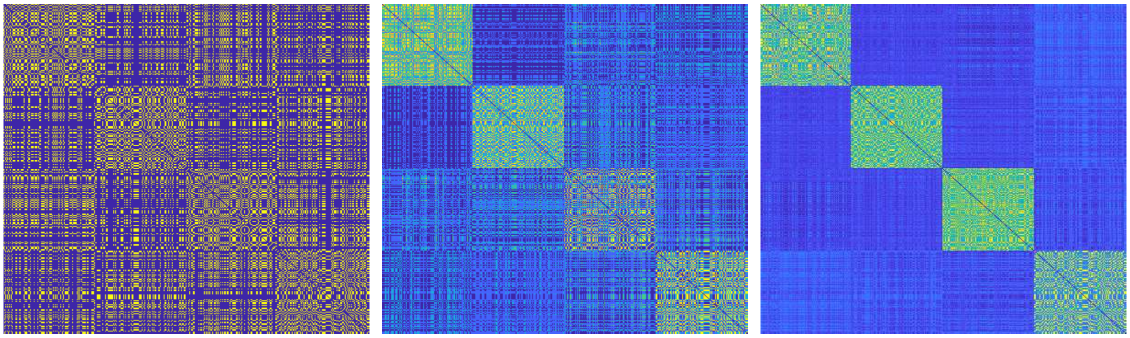

Figure 1: Co-association matrix of EKSS for $B = 1, 5, 50$ base clusterings. Data generation parameters are $D = 100$, $d = 3$, $K = 4$, $N = 400$, and the data is noise-free; the algorithm uses $\bar{K} = 4$ candidate subspaces of dimension $\bar{d} = 3$ and no thresholding. Resulting clustering errors are 61%, 25%, and 0%.

2.2 Consensus Clustering

Ensemble methods have been used in the context of general clustering for some time and fall within the topic of *consensus clustering*, with an overview of the benefits and techniques given in [28]. The central idea behind these methods is to obtain many clusterings from a simple base clusterer, such as K-means, and then combine the results intelligently. In order to obtain different base clusterings, diversity of some sort must be incorporated. This is typically done by obtaining bootstrap samples of the data [29,30], subsampling the data to reduce computational complexity [31], or performing random projections of the data [32]. Alternatively, the authors of [33, 34] use the randomness in different initializations of K-means to obtain diversity. We take this approach here for subspace clustering. After diversity is achieved, the base clustering results must be combined. The *evidence accumulation clustering* framework laid out in [20] combines results by voting, i.e., creating a co-association matrix A whose (i, j)th entry is equal to the number of times two points are clustered together[1]. A theoretical framework for this approach is laid out in [35], where the entries of the co-association matrix are modeled as Binomial random variables. This approach is studied further in [36, 37], in which the clustering problem is solved as a Bregman divergence minimization. These models result in improved clustering performance over previous work but are not accompanied by any theoretical guarantees with regard to the resulting co-association matrix. Further, in our experiments, we did not find the optimization-based approach to perform as well as simply running spectral clustering on the resulting co-association matrix.

In the remainder of this paper, we apply ideas from consensus clustering to the subspace clustering problem. We describe our ensemble KSS algorithm and its guarantees and demonstrate the algorithm's state-of-the-art performance on both synthetic and real datasets.

3 Ensemble K-Subspaces

In this section, we describe our method for subspace clustering using ensembles of the K-subspaces algorithm, which we refer to as Ensemble K-subspaces (EKSS). Our key insight leading to EKSS is the fact that the partially-correct clustering information from each random initialization of KSS can be combined to form a more accurate clustering of the data. We therefore run several random initializations of KSS and form a co-association matrix combining their results that becomes the affinity matrix used in spectral clustering to obtain cluster labels.

In more technical detail, our EKSS algorithm proceeds as follows. For each of $b = 1, \ldots, B$ base clusterings, we obtain an estimated clustering $\mathcal{C}^{(b)}$ from a single run of KSS with a random initialization of candidate bases. The (i, j)th entry of the co-association matrix is the number of base clusterings for which x_i and x_j are clustered together. We then threshold the co-association matrix as in [11] by taking the top

[1] In the context of consensus clustering, we use the terms *affinity matrix* and *co-association matrix* interchangeably.

Algorithm 1 Ensemble K-subspaces (EKSS)

```
Input: $\mathcal{X} = \{x_1, x_2, \dots, x_N\} \subset \mathbb{R}^D$: data, $\bar{K}$: number of candidate subspaces, $\bar{d}$: candidate dimension, $K$: number of output clusters, $q$: threshold parameter, $B$: number of base clusterings, $T$: number of KSS iterations
Output: $\mathcal{C} = \{c_1, \dots, c_K\}$: clusters of $\mathcal{X}$
for $b = 1, \dots, B$ (in parallel) do
    $U_1, \dots, U_{\bar{K}} \overset{iid}{\sim} \mathrm{Unif}(\mathrm{St}(D, \bar{d}))$                Draw $\bar{K}$ random subspace bases
    $c_k \leftarrow \left\{ x \in \mathcal{X} \ : \ \forall j \ \left\|U_k^T x\right\|_2 \geq \left\|U_j^T x\right\|_2 \right\}$ for $k = 1, \dots, \bar{K}$                Cluster by projection
    for $t = 1, \dots, T$ (in sequence) do
        $U_k \leftarrow \mathrm{PCA}\left(c_k, \bar{d}\right)$ for $k = 1, \dots, \bar{K}$                Estimate subspaces
        $c_k \leftarrow \left\{ x \in \mathcal{X} \ : \ \forall j \ \left\|U_k^T x\right\|_2 \geq \left\|U_j^T x\right\|_2 \right\}$ for $k = 1, \dots, \bar{K}$                Cluster by projection
    end for
    $\mathcal{C}^{(b)} \leftarrow \{c_1, \dots, c_{\bar{K}}\}$
end for
$A_{i,j} \leftarrow \frac{1}{B} \left| \left\{ b : x_i, x_j \text{ are co-clustered in } \mathcal{C}^{(b)} \right\} \right|$ for $i, j = 1, \dots, N$                Form co-association matrix
$\bar{A} \leftarrow \textsc{Thresh}(A, q)$                Keep top $q$ entries per row/column
$\mathcal{C} \leftarrow \textsc{SpectralClustering}(\bar{A}, K)$                Final Clustering
```

q values from each row/column. Once this thresholded co-association matrix is formed, cluster labels are obtained using spectral clustering. Pseudocode for EKSS is given in Alg. 1, where Thresh sets all but the top q entries in each row/column to zero as in [11] (pseudocode for this procedure is given in Appendix B.1) and SpectralClustering [38] clusters the data points based on the co-association matrix A. Note that the number of candidates $\bar{K}$ and candidate dimension $\bar{d}$ need not match the number K and dimension d of the true underlying subspaces. Fig. 1 shows the progression of the co-association matrix as $B = 1, 5, 50$ base clusterings are used for noiseless data from $K = 4$ subspaces of dimension $d = 3$ in an ambient space of dimension $D = 100$ using $\bar{K} = 4$ candidates of dimension $\bar{d} = 3$. We discuss the choice of parameters for EKSS in the following sections.

3.1 Computational Complexity

Recall the relevant parameters: K is the number of output clusters, $\bar{K}$ is the number of candidate subspaces in EKSS, $\bar{d}$ is the dimension of those candidates, N is the number of points, D is the ambient dimension, B is the number of KSS base clusterings to combine, and T is the number of iterations within KSS. To form the co-association matrix, the complexity of EKSS is $O(BT(\bar{K}D^2\bar{d} + \bar{K}D\bar{d}N))$. We run the KSS base clusterings in parallel and use very few iterations, making the functional complexity of EKSS $O(\bar{K}D^2\bar{d} + \bar{K}D\bar{d}N)$, which is competitive with existing methods. In comparison, TSC has complexity $O(DN^2)$ and SSC-ADMM has complexity $O(TN^3)$, where T is the number of ADMM iterations. Note that typically $N > D$ and sometimes much greater. We have not included the cost of spectral clustering, which is $O(KN^2)$. For most modern subspace clustering algorithms (except SSC-ADMM), this dominates the computational complexity as N grows.

3.2 Parameter Selection

EKSS requires a number of input parameters, whose selection we now discuss. As stated in Section 3.1, we use a small number of KSS iterations, setting $T = 3$ in all experiments. Typically, B should be chosen as large as computation time allows. In our experiments on real data, we choose $B = 1000$. The number of output clusters K is required for all subspace clustering algorithms, and methods such as those described in [39] can be used to estimate this value. Hence, the relevant parameters for selection are the candidate parameters $\bar{K}$ and $\bar{d}$ and the thresholding parameter q.

When possible, the candidate parameters should be chosen to match the true UoS parameters. In particular, it is advised to set $\bar{K} = K$ and $\bar{d} = d$ when they are known. In practice, a good approximating dimension for the underlying subspace is often known. For example, images of a Lambertian object under varying illumination are known to lie near a subspace with $d = 9$ [1] and moving objects in video are known to lie near an affine subspace with $d = 3$ [40]. However, as we will show in the following section, our theoretical guarantees hold even if there is model mismatch. Namely, the choice of $\bar{K} = 2$ and $\bar{d} = 1$ still provably yields correct clustering, though this results in a degradation of empirical performance.

The thresholding parameter q can be chosen according to data-driven techniques as in [39], or following the choice in [11]. In our experiments on real data, we select q (or the relevant thresholding parameter in the case of SSC) by sweeping over a large range of values and choosing the value corresponding to the lowest clustering error. Note that q is applied to the co-association matrix A, and hence the computational complexity of performing model selection is much lower than that of running the entire EKSS algorithm numerous times.

We briefly consider the parameters required by existing algorithms. SSC [16] and EnSC [18] both require two parameters to be selected when solving the sparse regression problem (2). SSC also performs thresholding on the affinity matrix, which in our experiments appears critical to performance on real data. See the author code of [16] for details. TSC requires the thresholding parameter q to be selected. To the best of our knowledge, no principled manner of selecting these parameters has been proposed in the literature, and we consider this an important issue for future study.

3.3 Base Clustering Accuracy

A natural heuristic to improve the clustering performance of EKSS is to add larger values to the co-association matrix for base clusterings believed to be more accurate, and smaller values for those believed to be less accurate. Here, we briefly describe one such approach. Note that Step 12 in EKSS is equivalent to adding a unit weight to each entry corresponding to co-clustered points, i.e., $A \leftarrow \frac{1}{B} \sum_{b=1}^{B} A^{(b)} w(b)$, where $A_{i,j}^{(b)} := \mathbb{1}\left\{x_i, x_j \text{ are clustered together in } \mathcal{C}^{(b)}\right\}$ and $w(b) = 1$. The key idea is that this weight $w(b)$ can instead be chosen to reflect some estimation of the quality of the bth clustering; we propose using the KSS cost function as a measure of clustering quality. Let $\mathcal{C}^{(b)} = \left\{c_1^{(b)}, \ldots, c_K^{(b)}\right\}$ denote the bth base clustering, and let $\mathcal{U}^{(b)} = \left\{U_1^{(b)}, \ldots, U_K^{(b)}\right\}$ denote the set of subspace bases estimated by performing PCA on the points in the corresponding clusters. The clustering quality can then be measured as

$$w(b) = 1 - \sum_{k=1}^{K} \sum_{i : x_i \in c_k^{(b)}} \left\| x_i - U_k^{(b)} {U_k^{(b)}}^T x_i \right\|_2^2 / \left\| X \right\|_F^2 , \tag{4}$$

a value between 0 and 1 that decreases as the KSS cost increases. We employ this value of $w(b)$ in all experiments on real data.

3.4 Alternative Ensemble Approaches

As KSS is known to perform poorly in many cases, one may wonder whether better performance can be obtained by applying the evidence accumulation framework to more recent algorithms such as SSC and GSC. We attempted such an approach by subsampling the data to obtain diversity in SSC-OMP [17] and EnSC [18]. However, the resulting clustering performance did not always surpass that of the base algorithm run on the full dataset. Similar behavior occurred for ensembles of the GSC algorithm [10] as well as the Fast Landmark Subspace Clustering algorithm [41]. We also experimented with MKF as a base clustering algorithm but found little or no benefit at a significant increase in computational complexity. Hence, it seems that the success of our proposed approach depends both on the evidence accumulation framework *and* the use of KSS as a base clustering algorithm. Toward this end, we found that EKSS did benefit from the recent CoP-KSS algorithm [13] as a base clusterer for larger benchmark datasets, as discussed in Section 5. The appropriate combination of ensembles of other algorithms is nontrivial and an exciting open topic for future research.

4 Recovery Guarantees

Recovery guarantees for KSS remain elusive despite nearly twenty years of use since its introduction. Intelligent initialization methods based on probabilistic farthest insertion are used in [9, 27], but these too lack theoretical guarantees. In this section, we provide a first step toward recovery guarantees for EKSS (Alg. 1). In particular, we show that (a) any "angle preserving" affinity matrix can be used to obtain clustering with guarantees matching those of state-of-the-art subspace clustering methods, and (b) EKSS has such an affinity matrix after the first KSS clustering step with high probability. Put together, these findings provide state-of-the-art guarantees for EKSS in the case where only the first KSS iteration is performed (i.e., $T = 0$ in Alg. 1). We refer to this parameter choice as *EKSS-0* and include explicit pseudocode for this specialization in Appendix B.1. To the best of our knowledge, our work is the first to provide any recovery guarantees for a KSS algorithm as well as the first characterization of a co-association matrix in the context of consensus clustering.

Section 4.1 presents the notion of an angle preserving affinity matrix and extends the guarantees of [11] to all algorithms that use such affinity matrices. Though developed here to analyze EKSS, these results apply broadly and provide a promising approach for analyzing other geometrically-based subspace clustering algorithms in the future. Section 4.2 shows that the affinity/co-association matrix of EKSS with $T = 0$ is angle preserving with high probability, and presents the resulting recovery guarantees: correct clustering for noiseless data and no false connections (NFC) for noisy data or data with missing entries.

We use N_{max} (N_{min}) throughout to refer to the maximum (minimum) number of points on any single subspace and d_{max} to refer to the maximum subspace dimension. The proofs of all results in this section are in Appendix A.

4.1 Recovery Guarantees for Angle Preserving Affinity Matrices

This section extends the NFC and connectedness guarantees of [11] to any algorithm that uses angle preserving affinity matrices. The key idea is that these affinity matrices sufficiently capture the information contained in pairwise angles and obtain good recovery when the angles differentiate the clusters well. Observe that using angles need not be a "goal" of such methods; deviating may in fact produce better performance in broader regimes, e.g., by incorporating higher order structure. Nevertheless, so long as the relative angles among points are sufficiently captured, the method immediately enjoys the guarantees of this section.

Definition 4.1 (Angle Preserving). *An affinity matrix A is* τ-angle preserving *for a set of points $\mathcal{X}$ with respect to a strictly increasing function $f : \mathbb{R}_+ \to \mathbb{R}_+$ if*

$$|A_{i,j} - f\left(|\langle x_i, x_j\rangle|\right)| \le \tau, \quad i, j \in [N], \tag{5}$$

where we note that $\cos^{-1}\left(|\langle x_i, x_j\rangle|\right)$ is the angle between the points x_i and x_j.

Note that f is an arbitrary monotonic transformation that takes small angles (large absolute inner products) to large affinities and takes large angles (small absolute inner products) to small affinities, and τ quantifies how close the affinity matrix is to such a transformation. Taking $f(\alpha) = \alpha$ and $\tau = 0$ recovers the absolute inner product.

To guarantee correct clustering (as opposed to NFC only), it is sufficient to show that the thresholded affinity matrix has both NFC and exactly K connected components [11, Appendix A]. We formalize this fact for clarity in the proposition below.

Proposition 4.2 (NFC and connectedness give correct clustering [11, Equation (15)]). *Assume that the thresholded affinity matrix formed by an algorithm satisfies NFC with probability at least $1 - \varepsilon_1$ and given NFC satisfies the connectedness condition with probability at least $1 - \varepsilon_2$. Then spectral clustering correctly identifies the components with probability at least $1 - \varepsilon_1 - \varepsilon_2$.*

Thus, we study conditions under which NFC and connectedness are guaranteed; conditions for correct clustering follow. In particular, we provide upper bounds on τ that guarantee NFC (Theorem 4.4) and connectedness (Theorem 4.5). The upper bound for NFC is given by a property of the data that we call the *q-angular separation*, defined as follows. We later bound this quantity in a variety of contexts.

Definition 4.3 (Angular Separation). *The* q-angular separation ϕ_q *of the points* $\mathcal{X} = \mathcal{X}_1 \cup \cdots \cup \mathcal{X}_K$ *with respect to a strictly increasing function* $f : \mathbb{R}_+ \to \mathbb{R}_+$ *is*

$$\phi_q = \min_{l \in [K], i \in [N_l]} \frac{f\left(\left|\left\langle x_i^{(l)}, x_{\neq i}^{(l)} \right\rangle\right|_{[q]}\right) - f\left(\max_{k \neq l, j \in [N_k]} \left|\left\langle x_i^{(l)}, x_j^{(k)} \right\rangle\right|\right)}{2}, \tag{6}$$

where $x_i^{(l)}$ *denotes the* i*th point of* $\mathcal{X}_l$*, and* $\left|\left\langle x_i^{(l)}, x_{\neq i}^{(l)} \right\rangle\right|_{[q]}$ *denotes the* q^{th} *largest absolute inner product between the point* $x_i^{(l)}$ *and other points in subspace* l*.*

In words, the q-angular separation quantifies how far apart the clusters are, as measured by the transformed absolute inner products. When this quantity is positive and large, pairwise angles differentiate clusters well. The following theorem connects this data property to angle preserving affinity matrices.

Theorem 4.4 (No false connections (NFC)). *Suppose* $\mathcal{X} = \mathcal{X}_1 \cup \cdots \cup \mathcal{X}_K$ *have* q*-angular separation* ϕ_q *with respect to a strictly increasing function* f*. Then the* q*-nearest neighbor graph for any* ϕ_q*-angle preserving affinity matrix (with respect to* f*) has no false connections.*

Theorem 4.4 states that sufficiently small deviation τ guarantees NFC as long as the data has positive q-angular separation. The next theorem provides an upper bound on τ that guarantees connectedness within a cluster with high probability given NFC. Under NFC, the q-nearest neighbors of any point (with respect to the affinity matrix) are in the same subspace, and so the theorem is stated with respect to only points within a single subspace. In particular, we restrict to the d-dimensional subspace and so consider the q-nearest neighbor graph $\tilde{G}$ for points $a_1, \ldots, a_n$ uniformly distributed on the sphere $\mathbb{S}^{d-1}$.

Theorem 4.5 (Connectedness). *Let* $a_1, \ldots, a_n \in \mathbb{R}^d$ *be i.i.d. uniform on* $\mathbb{S}^{d-1}$ *and let* $\tilde{G}$ *be their corresponding* q*-nearest neighbor graph formed from a* τ*-angle preserving affinity matrix. Let* $\gamma \in (1, n/\log n)$ *be arbitrary, and let* θ *be the spherical radius of a spherical cap covering* $\gamma \log n/n$ *fraction of the area of* $\mathbb{S}^{d-1}$*. Suppose* $q \in [4(24\pi)^{d-1}\gamma \log n, n]$ *and* $\theta < \pi/48$*. Then if* $\tau < C_3$*,*

$$\mathbb{P}\{\tilde{G} \text{ is connected } \} \geq 1 - \frac{2}{n^{\gamma-1}\gamma \log n}, \tag{7}$$

where C_3 *depends only on* d*,* n*,* f*, and* γ *and is defined in the proof.*

We now provide explicit high-probability lower bounds on the q-angular separation ϕ_q from (6) in some important settings relevant to subspace clustering. These results can be used to guarantee NFC by bounding the deviation level τ. Consider first the case where there is no intersection between any pair of subspaces but there are potentially unobserved entries, i.e., missing data. Lemma 4.6 bounds ϕ_q from below in such a setting; the bound depends on a variant of the minimum principal angle between subspaces that accounts for missing data.

Lemma 4.6 (Angular separation for missing data). *Let* $\mathcal{S}_k$*,* $k = 1, \ldots, K$ *be subspaces of dimension* $d_1, \ldots, d_K$ *in* $\mathbb{R}^D$*. Let the* N_k *points in* $\mathcal{X}_k$ *be drawn as* $x_j^{(k)} = U^{(k)} a_j^{(k)}$*, where* $a_j^{(k)}$ *are i.i.d. uniform on* $\mathbb{S}^{d_k - 1}$ *and* $U^{(k)} \in \mathbb{R}^{D \times d_k}$ *has (not necessarily orthonormal) columns that form a basis for* $\mathcal{S}_k$*. In each* $x_j \in \mathcal{X}$*, up to* s *entries are then unobserved, i.e., set to zero. Let* $\rho \in [0, 1)$ *be arbitrary and suppose that* $N_{min} > N_0$*, where* N_0 *is a constant that depends only on* d_{max} *and* ρ*. Suppose that* $q < N_{min}^{\rho}$ *and*

$$r_s = \frac{\max_{k,l:k \neq l, \mathcal{D}:|\mathcal{D}| \leq 2s} \left\| {U_{\mathcal{D}}^{(k)}}^T U^{(l)} \right\|_2}{\min_{l,\mathcal{D}:|\mathcal{D}| \leq 2s, \|a\|=1} \left\| {U_{\mathcal{D}}^{(l)}}^T U^{(l)} a \right\|_2} < 1, \tag{8}$$

where $U_{\mathcal{D}}^{(l)}$ *is the matrix obtained from* $U^{(l)}$ *by setting the rows indexed by* $\mathcal{D} \subset \{1, \ldots, D\}$ *to zero. Then* $\phi_q > C_1$ *with probability at least* $1 - \sum_{k=1}^{K} N_k e^{-c_1(N_k - 1)}$*, where* $c_1 > 0$ *is a numerical constant that depends on* N_{min}^{ρ}*, and* $C_1 > 0$ *depends only on* r_s *and* f*. Both* c_1 *and* C_1 *are defined in the proof.*

To gain insight to the above lemma, note that for full data $s = 0$, and r_s simplifies to $\max_{k,l:k \neq l} ||{U^{(k)}}^T U^{(l)}||_2$, which is less than one if and only if there is no intersection between subspaces. In this case, Lemma 4.6 states that ϕ_q is positive (i.e., NFC is achievable) as long as there is no intersection between any pair of subspaces. We next turn to the case where the subspaces are allowed to intersect and points may be corrupted by additive noise. Lemma 4.7 bounds ϕ_q from below in such a setting; it requires the subspaces to be sufficiently far apart with respect to their affinity, which is defined as [11, 42]

$$\operatorname{aff}(\mathcal{S}_k, \mathcal{S}_l) = \frac{1}{\sqrt{d_k \wedge d_l}} \left\| U_k^T U_l \right\|_F , \tag{9}$$

where U_k and U_l form orthonormal bases for the d_k- and d_l-dimensional subspaces $\mathcal{S}_k$ and $\mathcal{S}_l$. Note that $\operatorname{aff}(\mathcal{S}_k, \mathcal{S}_l)$ is a measure of how close two subspaces are in terms of their principal angles and takes the value 1 if two subspaces are equivalent and 0 if they are orthogonal.

Lemma 4.7 (Angular separation for noisy data)**.** *Let the points in $\mathcal{X}_k$ be the set of N_k points $x_i^{(k)} = y_i^{(k)} + e_i^{(k)}$, where the $y_i^{(k)}$ are drawn i.i.d. from the set $\{y \in \mathcal{S}_k : \|y\| = 1\}$, independently across k, and the $e_i^{(k)}$ are i.i.d. $\mathcal{N}(0, \frac{\sigma^2}{D} I_D)$. Let $\mathcal{X} = \mathcal{X}_1 \cup \cdots \cup \mathcal{X}_K$ and $q < N_{min}/6$. Suppose that*

$$\max_{k,l:k \neq l} \operatorname{aff}(\mathcal{S}_k, \mathcal{S}_l) + \frac{\sigma(1+\sigma)}{\sqrt{\log N}} \frac{\sqrt{d_{max}}}{\sqrt{D}} \leq \frac{1}{15 \log N}, \tag{10}$$

with $D > 6 \log N$. Then $\phi_q > C_2$ with probability at least $1 - \frac{10}{N} - \sum_{k=1}^K N_k e^{-c_2(N_k - 1)}$, where $c_2 > 0$ is a numerical constant, and $C_2 > 0$ depends only on σ, D, d_{max}, N, $\max_{k,l:k \neq l} \operatorname{aff}(\mathcal{S}_k, \mathcal{S}_l)$, and f. Both c_2 and C_2 are defined in the proof.

Lemmas 4.6 and 4.7 state that under certain conditions on the arrangement of subspaces and points, the separation ϕ_q defined in (6) is positive with high probability and with given lower bounds. In the following section, we show that taking sufficiently many base clusterings B in EKSS-0 guarantees the affinity matrix is sufficiently angle preserving with high probability.

4.2 EKSS-0 Recovery Guarantees

In this section, we show that the co-association/affinity matrix formed by EKSS-0 is angle preserving, leading to a series of recovery guarantees for the problem of subspace clustering. We say that two points are *co-clustered* if they are assigned to the same candidate subspace in line 5 of Algorithm 1 (note that lines 6-9 are not computed for EKSS-0). The key to our guarantees lies in the fact that for points lying on the unit sphere, the probability of co-clustering is a monotonically increasing function of the absolute value of their inner product, as shown in Lemma 4.9 below. For EKSS-0, the entries of the affinity matrix A are empirical estimates of these probabilities, and hence the deviation level τ is appropriately bounded with high probability by taking sufficiently many base clusterings B. These results allow us to apply Theorems 4.4 and 4.5 from the previous section. We remind the reader that the parameters $\bar{K}$ and $\bar{d}$ are the number and dimension of the *candidate* subspaces in EKSS, and need not be related to the data being clustered.

Theorem 4.8 (EKSS-0 is angle preserving)**.** *Let $A \in \mathbb{R}^{N \times N}$ be the affinity matrix formed by EKSS-0 (line 12, Alg. 1) with parameters $\bar{K}, \bar{d}$ and B. Let $\tau > 0$. Then with probability at least $1 - N(N-1)e^{-c_3 \tau^2 B}$, the matrix A is τ-angle preserving, where the increasing function $f_{\bar{K},\bar{d}}$ is defined in the proof of Lemma 4.9, $c_3 = 2\sqrt{\log 2}$, and the probability is taken with respect to the random subspaces drawn in EKSS-0 (line 4, Alg. 1).*

In the context of the previous section, Theorem 4.8 states that the affinity matrix formed by EKSS-0 is τ-angle preserving and hence satisfies the main condition required for Theorems 4.4 and 4.5. We refer to the transformation function as $f_{\bar{K},\bar{d}}$ to denote the dependence on the EKSS-0 parameters, noting that $f_{\bar{K},\bar{d}}$ is increasing for *any* natural numbers $\bar{K}$ and $\bar{d}$. A consequence of Theorem 4.8 is that by increasing the number of base clusterings B, we can reduce the deviation level τ to be arbitrarily small while maintaining a fixed probability that the model holds. This fact allows us to apply the results of the previous section to provide recovery guarantees for EKSS-0. The major nontrivial aspect of proving Theorem 4.8 lies in establishing the following lemma.

Lemma 4.9. *The (i, j)th entry of the affinity matrix A formed by EKSS-0 (line 12, Alg. 1) has expected value*

$$\mathbb{E}A_{i,j} = f_{\bar{K},\bar{d}}(|\langle x_i, x_j\rangle|) \tag{11}$$

where $f_{\bar{K},\bar{d}} : \mathbb{R}_+ \to \mathbb{R}_+$ is a strictly increasing function (defined in the proof), and the expectation is taken with respect to the random subspaces drawn in EKSS-0 (line 4, Alg. 1). The subscripts $\bar{K}$ and $\bar{d}$ indicate the dependence of $f_{\bar{K},\bar{d}}$ on those EKSS-0 parameters.

Proof. We provide a sketch of the proof here; the full proof can be found in Appendix A. For notational compactness, we instead prove that the probability of two points being co-clustered is a *decreasing* function of the angle θ between them. Denote this probability by $p_{\bar{K},\bar{d}}(\theta)$. Let $U_1, U_2, \ldots, U_{\bar{K}} \in \mathbb{R}^{D\times\bar{d}}$ be the K candidate bases. Let $\tilde{p}(\theta)$ be the probability that any two points with corresponding angle θ are assigned to the candidate U_1. Then by symmetry we have $p_{\bar{K},\bar{d}}(\theta) = K\tilde{p}(\theta)$, and it suffices to prove that $\tilde{p}$ is strictly decreasing. Without loss of generality, let $x_i = e_1$ and $x_j = \cos(\theta)e_1 + \sin(\theta)e_2$, where $e_m \in \mathbb{R}^D$ is the mth standard basis vector. We then have that

$$\tilde{p}(\theta) = \mathbb{P}\left\{Qx_i, Qx_j \text{ both assigned to } U_1\right\},$$

where Q is an arbitrary orthogonal transformation of $\mathbb{R}^D$. Let E denote the event of interest and L denote the span of e_1 and e_2. The event E can then be written as

$$z^T Q P_L (P_1 - P_k) P_L Q z > 0, \quad \text{for} \quad 1 < k \le K \quad \text{and} \quad z = x_i, x_j, \tag{12}$$

where P_L denotes the orthogonal projection onto the subspace L and P_k denotes the orthogonal projection onto the subspace spanned by U_k. By restricting to L, (12) can be reduced to a two-dimensional quadratic form, and we can compute in closed form $\mathbb{P}\{E \mid U_1, \ldots, U_{\bar{K}}\}$. Differentiating shows that this term is decreasing and hence (by the law of total probability) so is $\tilde{p}(\theta)$. □

It is interesting to note that the result of Lemma 4.9 does not depend on the underlying data distribution, i.e., the number or arrangement of subspaces, but instead says that clustering with EKSS-0 is (in expectation) a function of the absolute inner product between points, regardless of the parameters. Thus, the results of this section all hold even with the simple parameter choice of $\bar{K} = 2$ and $\bar{d} = 1$ in Algorithm 1. Our empirical results suggest that decreasing $\bar{K}$ and increasing $\bar{d}$ increases the probability of co-clustering. However, when running several iterations of KSS (EKSS with $T > 0$), we find that it is advantageous to choose $\bar{K}$ and $\bar{d}$ to match the true parameters of the data as closely as possible, allowing KSS to more accurately model the underlying subspaces.

Combined with the results of Section 4.1, Theorem 4.8 enables us to quickly obtain recovery guarantees for EKSS-0, which we now present. We first consider the case where the data are noiseless, i.e., lie perfectly on a union of K subspaces. Theorems 4.10 and 4.11 provide sufficient conditions on the arrangement of subspaces such that EKSS-0 achieves *correct clustering* with high probability.

Theorem 4.10 (EKSS-0 provides correct clustering for disjoint subspaces)**.** *Let $\mathcal{S}_k$, $k = 1, \ldots, K$ be subspaces of dimension $d_1, \ldots, d_K$ in $\mathbb{R}^D$. Let the N_k points in $\mathcal{X}_k$ be drawn as $x_j^{(k)} = U^{(k)}a_j^{(k)}$, where $a_j^{(k)}$ are i.i.d. uniform on $\mathbb{S}^{d_k-1}$ and $U^{(k)} \in \mathbb{R}^{D\times d_k}$ has orthonormal columns that form a basis for $\mathcal{S}_k$. Let $\rho \in [0, 1)$ be arbitrary and suppose that $N_{min} > N_0$, where N_0 is a constant that depends only on d_{max} and ρ. Suppose that $q \in [c_4 \log N_{max}, N_{min}^{\rho}]$ and*

$$r_0 = \max_{k,l:k\neq l} \left\| {U^{(k)}}^T U^{(l)} \right\|_2 < 1, \tag{13}$$

where $c_4 = 12(24\pi)^{d_{max}-1}$. Then $\bar{A}$ obtained by EKSS-0 results in correct clustering of the data with probability at least $1 - \sum_{k=1}^{K}\left(N_k e^{-c_1(N_k-1)} + 2N_k^{-2}\right) - N(N-1)e^{-c_3 B \min\{C_1, C_3\}^2}$, where $c_1, c_3 > 0$ are numerical constants, $C_1 > 0$ depends on r_0 and the function $f_{\bar{K},\bar{d}}$ defined in Theorem 4.8, and $C_3 > 0$ depends on d_{max}, N_{min}, and $f_{\bar{K},\bar{d}}$.

Theorem 4.11 (EKSS-0 provides correct clustering for subspaces with bounded affinity)**.** *Let $\mathcal{S}_k$, $k = 1, \ldots, K$ be subspaces of dimension $d_1, \ldots, d_K$ in $\mathbb{R}^D$. Let the points in $\mathcal{X}_k$ be a set of N_k points drawn* *uniformly from the unit sphere in subspace k, i.e., from the set* $\{x \in \mathcal{S}_k : \|x\| = 1\}$*. Let* $\mathcal{X} = \mathcal{X}_1 \cup \cdots \cup \mathcal{X}_K$ *and* $N = \sum_k N_k$*. Let* $q \in [c_4 \log N_{max}, N_{min}/6)$*, where* $c_4 = 12(24\pi)^{d_{max}-1}$*. If*

$$\max_{k,l:k\neq l} \operatorname{aff}(\mathcal{S}_k, \mathcal{S}_l) \leq \frac{1}{15 \log N},$$

then $\bar{A}$ *obtained by EKSS-0 results in correct clustering of the data with probability at least* $1 - \frac{10}{N} - \sum_{k=1}^{K} \left(N_k e^{-c_2(N_k-1)} - 2N_k^{-2}\right) - N(N-1)e^{-c_3 B \min\{C_2, C_3\}^2}$*, where* $c_2, c_3 > 0$ *are numerical constants,* $C_2 > 0$ *depends only on* $\max_{k,l:k\neq l} \operatorname{aff}(\mathcal{S}_k, \mathcal{S}_l)$*,* D*,* d_{max}*,* N*, and the function* $f_{\bar{K},\bar{d}}$ *defined in Theorem 4.8, and* $C_3 > 0$ *depends on* d_{max}*,* N_{min}*, and* $f_{\bar{K},\bar{d}}$*.*

We next consider two forms of data corruption. Theorem 4.12 shows that the affinity matrix built by EKSS-0 has NFC in the presence of data corrupted by additive Gaussian noise. Theorem 4.13 shows that EKSS-0 maintains NFC even in the presence of a limited number of missing (unobserved) entries.

Theorem 4.12 (EKSS-0 has NFC with noisy data)**.** *Let* $\mathcal{S}_k$*,* $k = 1, \ldots, K$ *be subspaces of dimension* $d_1, \ldots, d_K$ *in* $\mathbb{R}^D$*. Let the points in* $\mathcal{X}_k$ *be the set of* N_k *points* $x_i^{(k)} = y_i^{(k)} + e_i^{(k)}$*, where the* $y_i^{(k)}$ *are drawn i.i.d. from the set* $\{y \in \mathcal{S}_k : \|y\| = 1\}$*, independently across* k*, and the* $e_i^{(k)}$ *are i.i.d.* $\mathcal{N}(0, \frac{\sigma^2}{D} I_D)$*. Let* $\mathcal{X} = \mathcal{X}_1 \cup \cdots \cup \mathcal{X}_K$ *and* $q < N_{min}/6$*. If*

$$\max_{k,l:k\neq l} \operatorname{aff}(\mathcal{S}_k, \mathcal{S}_l) + \frac{\sigma(1+\sigma)}{\sqrt{\log N}} \frac{\sqrt{d_{max}}}{\sqrt{D}} \leq \frac{1}{15 \log N},$$

with $D > 6 \log N$*, then* $\bar{A}$ *obtained from running EKSS-0 has no false connections with probability at least* $1 - \frac{10}{N} - \sum_{k=1}^{K} N_k e^{-c_2(N_k-1)} - N(N-1)e^{-c_3 C_2^2 B}$*, where* $c_2, c_3 > 0$ *are numerical constants, and* $C_2 > 0$ *depends only on* $\max_{k\neq l}$ *aff*$(\mathcal{S}_k, \mathcal{S}_l)$*,* σ*,* D*,* d_{max}*,* N *and the function* $f_{\bar{K},\bar{d}}$ *defined in Theorem 4.8.*

Theorem 4.13 (EKSS-0 has NFC with missing data)**.** *Let the* n *points in* $\mathcal{X}_k$ *be drawn as* $x_j^{(k)} = U^{(k)} a_j^{(k)}$*, where* $a_j^{(k)}$ *are i.i.d. uniform on* $\mathbb{S}^{d-1}$ *and the entries of* $U^{(k)} \in \mathbb{R}^{D\times d}$ *are i.i.d.* $\mathcal{N}(0, \frac{1}{D})$*. Let* $\rho \in [0, 1)$ *be arbitrary and suppose that* $n > N_0$*, where* N_0 *is a constant that depends only on* d *and* ρ*. Suppose that* $q < n^\rho$*, and assume that in each* $x_j \in \mathcal{X}$ *up to* s *arbitrary entries are unobserved, i.e., set to 0. Let* $\mathcal{X} = \mathcal{X}_1 \cup \cdots \cup \mathcal{X}_K$*. If*

$$D - 3c_5 d - c_5 \log K \geq s \left(c_5 \log\left(\frac{De}{2s}\right) + c_6\right), \tag{14}$$

then $\bar{A}$ *obtained by EKSS-0 has no false connections with probability at least* $1 - Ne^{-c_1(n-1)} - N(N-1)e^{-c_3 C_1^2 B} - 4e^{-c_7 D}$*, where* $c_1, c_3, c_5, c_6, c_7 > 0$*, are numerical constants and* $C_1 > 0$ *depends only on the ratio* r_s *defined in* (8) *and the function* $f_{\bar{K},\bar{d}}$ *defined in Theorem 4.8.*

4.3 Discussion of Results

The data model considered in Theorems 4.10-4.13 is known as the "semi-random" model [21], due to the fixed arrangement of subspaces with randomly-drawn points, and has been analyzed widely throughout the subspace clustering literature [11, 21, 24, 43, 44]. Our guarantees under this model are identical (up to constants and log factors) to those for TSC and SSC (see [11, Section VII] for further discussion of their guarantees). The key difference between our results and those of TSC is that we pay a $N(N-1)/2e^{-c_3\tau^2 B}$ penalty in recovery probability due to the approximate observations of the transformed inner products. Although our experiments indicate that EKSS-0 appears to have no benefits over TSC, we do find that by running a small number of KSS iterations, significant performance improvements are achieved. While the above analysis holds only for the case of $T = 0$, letting $T > 0$ is guaranteed to not increase the KSS cost function [6]. In our experiments, we found that setting $T > 0$ uniformly improved clustering performance, and our empirical results indicate that EKSS is in fact more robust (than EKSS-0 and TSC) to subspaces with small principal angles.

While the explicit choice of B is tied to the unknown function $f_{\bar{K},\bar{d}}$, our results provide intuition for setting this value; namely, the closer the underlying subspaces (in terms of principal angles), the more base

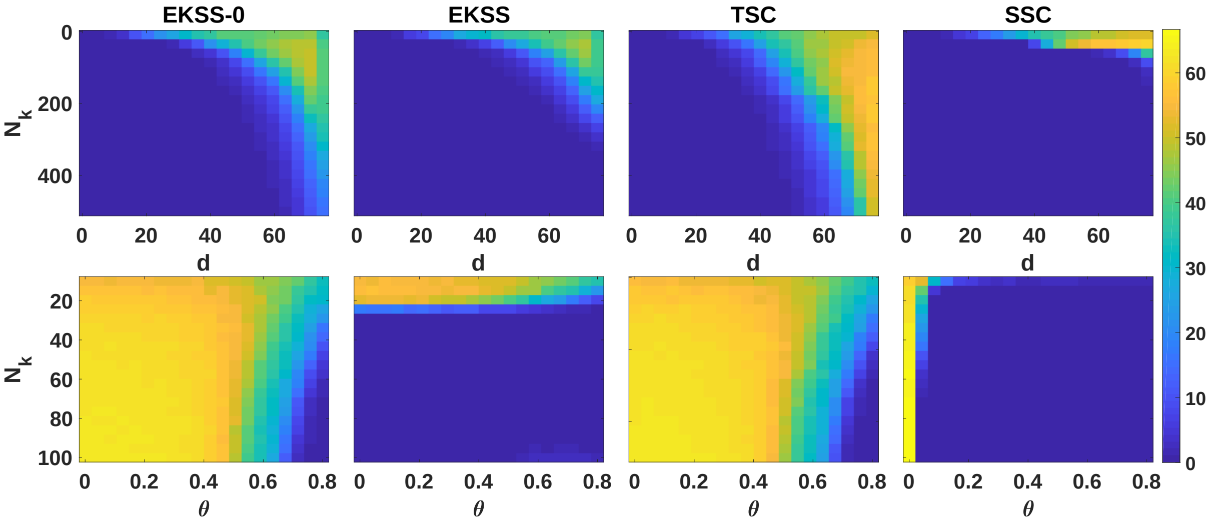

Figure 2: Clustering error (%) for proposed and state-of-the-art subspace clustering algorithms as a function of problem parameters N_k, number of points per subspace, and true subspace dimension d or angle between subspaces θ. Fixed problem parameters are $D = 100$, $K = 3$.

clusterings required. The inverse dependence on $\log N$ in Theorems 4.11 and 4.12 indicates a tension as the problem size grows. On one hand, points from the same subspace are more likely to be close when N is large, improving the angular separation. On the other hand, points are also more likely to fall near the intersection of subspaces, potentially degrading the angular separation. In all experimental results, we see that both EKSS and TSC perform better with larger N. Finally, we note that the leading $O\left(N^2\right)$ coefficient in the above probabilities results from applying a union bound and is likely conservative.

5 Experimental Results

In this section, we demonstrate the performance in terms of clustering error (defined in Appendix B.2) of EKSS on both synthetic and real datasets. We first show the performance of our algorithm as a function of the relevant problem parameters and verify that EKSS-0 exhibits the same empirical performance as TSC, as expected based on our theoretical guarantees. We also show that EKSS can recover subspaces that either have large intersection or are extremely close. We then demonstrate on benchmark datasets that EKSS not only improves over previous geometric methods, but that it achieves state-of-the-art results competitive with those obtained by self-expressive methods.

5.1 Synthetic Data

For all experiments in this section, we take $q = \max(3, \lceil N_k/20 \rceil)$ for EKSS-0 and TSC and $q = \max(3, \lceil N_k/6 \rceil)$ for EKSS, where $\lceil c \rceil$ denotes the largest integer greater than or equal to c. We set $B = 10,000$ for EKSS-0 and EKSS. When the angles between subspaces are not explicitly specified, it is assumed that the subspaces are drawn uniformly at random from the set of all d-dimensional subspaces of $\mathbb{R}^D$. For all experiments, we draw points uniformly at random from the unit sphere in the corresponding subspace and show the mean error over 100 random problem instances. We use the code provided by the authors for TSC and SSC. We employ the ADMM implementation of SSC and choose the parameters that result in the best performance in each scenario.

We explore the influence of some relevant problem parameters on the EKSS algorithm in Fig. 2. We take the ambient dimension to be $D = 100$, the number of subspaces to be $K = 3$, and generate noiseless data. We first consider the dependence on subspace dimension and the number of points per subspace. The top row of Fig. 2 shows the misclassification rate as the number of points per subspace ranges from $10 - 500$

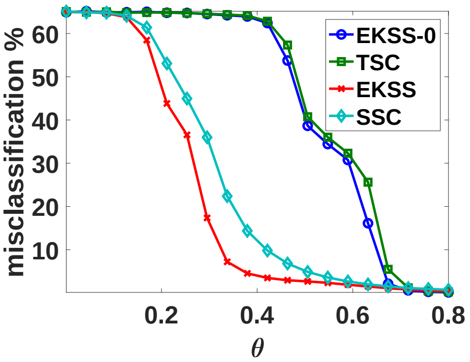

Figure 3: Clustering error (%) as a function of subspace angles with noisy data. Problem parameters are $D = 100$, $d = 10$, $K = 3$, $N_k = 500$, $\sigma^2 = 0.05$.

and the subspace dimension ranges from $1 - 75$. When $2d > D$ (i.e., $d \geq 51$), pairs of subspaces necessarily have intersection, and the intersection dimension grows with d. First, the figures demonstrate that EKSS-0 achieves roughly the same performance as TSC, resulting in correct clustering even in the case of subspaces with large intersection. Second, we see that EKSS can correctly cluster for subspace dimensions larger than that of TSC as long as there are sufficiently many points per subspace. For large subspace dimensions with a moderate number of points per subspace, SSC achieves the best performance.

We next explore the clustering performance as a function of the distance between subspaces, as shown in the second row of Fig. 2. We set the subspace dimension to $d = 10$ and generate $K = 3$ subspaces such that the principal angles between subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$, as well as those between $\mathcal{S}_1$ and $\mathcal{S}_3$ are θ, for 20 values in the range $[0.001, 0.8]$. Most strikingly, EKSS is able to resolve subspaces with even the smallest separation. This stands in contrast to TSC; it fails in this regime because when the subspaces are extremely close, the inner products between points on different subspaces can be nearly as large as those within the same subspace. Similarly, in the case of SSC, points on different subspaces can be used to regress any given point with little added cost, and so it fails at very small subspace angles. However, as long as there is still some separation between subspaces, EKSS is able to correctly cluster all points. The theory presented here does not capture this phenomenon, and recovery guarantees that take into account multiple iterations of KSS are an important topic for future work.

As a final comparison, we show the clustering performance with noisy data. Fig. 3 shows the clustering error as a function of the angle between subspaces for the case of $K = 3$ subspaces of dimension $d = 10$, with $N_k = 500$ points corrupted by zero-mean Gaussian noise with covariance $0.05 I_D$. We again consider 20 values of the angle θ between 0.001 and 0.08. EKSS-0 and TSC obtain similar performance, and more importantly EKSS is more robust to small subspace angles than SSC, even in the case of noisy data.

5.2 Benchmark Data

In this section, we show that EKSS achieves competitive subspace clustering performance on a variety of datasets commonly used as benchmarks in the subspace clustering literature. We consider the Hopkins-155 dataset [2], the cropped Extended Yale Face Database B [4, 45], COIL-20 [46] and COIL-100 [47] object databases, the USPS dataset provided by [48], and 10,000 digits of the MNIST handwritten digit database [5], where we obtain features using a scattering network [49] as in [18]. Descriptions of these datasets and the relevant problem parameters are included in Appendix B.3. We compare the performance of EKSS to several benchmark algorithms: KSS [6], CoP-KSS [13], Median K-Flats (MKF) [9], TSC [11], the ADMM implementation of SSC [16], SSC-OMP [17], and Elastic Net Subspace Clustering (EnSC) [18]. For all algorithms, we selected the parameters that yielded the lowest clustering error, performing extensive model selection where possible. We point out that this method of parameter selection requires knowledge of the ground truth labels, which are typically unavailable in practice. For the larger USPS and MNIST datasets, we obtained a small benefit by replacing PCA (line 7, Alg. 1) with the more robust Coherence Pursuit, i.e.,

Algorithm	Hopkins	Yale B	COIL-20	COIL-100	USPS	MNIST-10k
EKSS	**0.26**	14.31	13.47	**28.57**	**15.84**	**2.39**
KSS	0.35	54.28	33.12	66.04	18.31	2.60
CoP-KSS	0.69	52.59	29.10	51.38	**7.73**	**2.57**
MKF	**0.24**	41.32	35.69	59.50	28.49	28.17
TSC	2.07	22.20	15.28	29.82	31.57	15.98
SSC-ADMM	1.07	**9.83**	**13.19**	44.06	56.61	19.17
SSC-OMP	25.25	**13.28**	27.29	34.79	77.94	19.19
EnSC	9.75	18.87	**8.26**	**28.75**	33.66	17.97

Table 1: Clustering error (%) of subspace clustering algorithms for a variety of benchmark datasets. The lowest two clustering errors are given in bold. Note that EKSS is among the best three for all datasets, but no other algorithm is in the top five across the board.

we use CoP-KSS as a base clustering algorithm instead of KSS. Further implementation details, including parameter selection and data preprocessing, can be found in Appendix B.3.

The clustering error for all datasets and algorithms is shown in Table 1, with the lowest two errors given in bold. First, note that EKSS outperforms its base clustering algorithm (KSS or CoP-KSS) in all cases except the USPS dataset, and sometimes by a very large margin. This result emphasizes the importance of leveraging all clustering information from the B base clusterings, as opposed to simply choosing the best single clustering. While CoP-KSS achieves lower clustering error than EKSS on the USPS dataset, a deeper investigation of the performance of CoP-KSS revealed that only 17 of the 1000 individual clusterings achieved an error lower than the 15.84% obtained by EKSS. A more sophisticated weighting scheme than that described in Section 3.3 could be employed to add more significant weights for the small number of base clusterings corresponding to low error. Alternative measures of clustering quality based on subspace margin [50] or novel internal clustering validation metrics [51] may provide improved performance. Next, the results show that EKSS is among the top performers in all datasets considered, achieving nearly perfect clustering of the Hopkins-155 dataset, which is known to be well approximated by the UoS model. Scalable algorithms such as SSC-OMP and EnSC perform poorly on this dataset, likely due to the small number of points. For the larger COIL-100, USPS, and MNIST datasets, EKSS also achieves strong performance, demonstrating its flexibility to perform well in both the small and large sample regimes. The self-expressive methods outperform EKSS on the Yale and COIL-20 datasets, likely due to the fact that they do not explicitly rely on the UoS model in building the affinity matrix. However, EKSS still obtains competitive performance on both datasets, making it a strong choice for a general-purpose algorithm for subspace clustering.

6 Conclusion

In this work, we presented the first known theoretical guarantees for both evidence accumulation clustering and the KSS algorithm. We showed that with a given choice of parameters, the EKSS algorithm can provably cluster data from a union of subspaces under the same conditions as existing algorithms. The theoretical guarantees presented here match existing guarantees in the literature, and our experiments on synthetic data indicate that the iterative approach of KSS provides a major improvement in robustness to small angles between subspaces. Further, our results generalize those in the existing literature, yielding the potential to inform future algorithm design and analysis. We demonstrated the efficacy of our approach on both synthetic and real data, and showed that our method achieves excellent performance on several real datasets.

A number of important open problems remain. First, extending our analysis to the general case of Alg. 1 (i.e., $T > 0$) is an important next step that is difficult because of the alternating nature of KSS. In selecting tuning parameters, we chose the combination that resulted in the lowest clustering error, which is not known in practice. Methods for unsupervised model selection are an important practical consideration for EKSS and subspace clustering in general. Finally, while we did not have success in implementing ensembles of state-of-the-art algorithms such as SSC, a deeper study of this topic could yield improved empirical performance.

A Proofs of Theoretical Results

The results of this section make use of the following notation. We define the absolute inner product between points $x_i \in \mathcal{S}_l$ and $x_j \in \mathcal{S}_k$ as

$$z_{i,j}^{(l,k)} = \left| \left\langle x_i^{(l)}, x_j^{(k)} \right\rangle \right|,$$

where k may be equal to l. We denote the qth largest absolute inner product between $x_i^{(l)}$ and other points in the subspaces $\mathcal{S}_l$ as $z_{(i,q)}^{(l)}$, i.e., we have

$$z_{(i,q)}^{(l)} = \left| \left\langle x_i^{(l)}, x_{\neq i}^{(l)} \right\rangle \right|_{[q]}$$

in the context of Definition 4.3.

A.1 Proof of Theorem 4.4

We first prove the statement for a fixed $x_i \in \mathcal{S}_l$. The statement of the theorem can be written as

$$\hat{f}_{(i,q)}^{(l)} > \max_{k \neq l, j} \hat{f}_{i,j}^{(l,k)}, \tag{15}$$

where $\hat{f}_{(i,q)}^{(l)}$ denotes the qth largest value in the set $\left\{ \hat{f}_{i,j}^{(l,l)} \right\}$. We first bound $\hat{f}$ in terms of f. Let $x_\iota \in \mathcal{S}_{k^*}$ be such that $\max_{k \neq l, j} \hat{f}_{i,j}^{(l,k)} = \hat{f}_{i,\iota}^{(l,k^*)}$ and note that $z_{i,\iota}^{(l,k^*)} \leq \max_{k \neq l, j} z_{i,j}^{(l,k)}$. Then we have

$$\begin{aligned} \max_{k \neq l, j} \hat{f}_{i,j}^{(l,k)} = \hat{f}_{i,\iota}^{(l,k^*)} & \leq f\left(z_{i,\iota}^{(l,k^*)} \right) + \tau \\ & \leq f\left(\max_{k \neq l, j} z_{i,j}^{(l,k)} \right) + \tau, \end{aligned}$$

where the second line follows by monotonicity of f. To lower bound $\hat{f}_{(i,q)}^{(l)}$, let x_κ be such that $\hat{f}_{(i,q)}^{(l)} = \hat{f}_{i,\kappa}^{(l,l)}$. If $z_{i,\kappa}^{(l,l)} \geq z_{(i,q)}^{(l)}$, then $f\left(z_{i,\kappa}^{(l,l)} \right) \geq f\left(z_{(i,q)}^{(l)} \right)$ by monotonicity of f. For the case where $z_{i,\kappa}^{(l,l)} < z_{(i,q)}^{(l)}$, define $x_\lambda \in \mathcal{S}_l$ such that $z_{(i,q)}^{(l)} = z_{i,\lambda}^{(l,l)}$ and note that

$$\hat{f}_{i,\kappa}^{(l,l)} > \hat{f}_{i,\lambda}^{(l,l)} \geq f\left(z_{i,\lambda}^{(l,l)} \right) - \tau = f\left(z_{(i,q)}^{(l)} \right) - \tau.$$

Therefore

$$\hat{f}_{(i,q)}^{(l)} \geq f\left(z_{(i,q)}^{(l)} \right) - \tau,$$

and (15) holds as long as

$$f\left(z_{(i,q)}^{(l)} \right) - \tau > f\left(\max_{k \neq l, j} z_{i,j}^{(l,k)} \right) + \tau,$$

or equivalently if

$$\tau < \frac{f\left(z_{(i,q)}^{(l)} \right) - f\left(\max_{k \neq l, j} z_{i,j}^{(l,k)} \right)}{2}. \tag{16}$$

Taking the minimum right-hand side of (16) among all $x \in \mathcal{X}$ completes the proof.

A.2 Proof of Theorem 4.5

To prove Theorem 4.5 from the paper, we first prove a slightly more general result that we will then apply.

Lemma A.1. *Let $a_1, \ldots, a_n \in \mathbb{R}^d$ be i.i.d. uniform on $\mathbb{S}^{d-1}$ and let $\tilde{G}$ be the corresponding q-nearest neighbor graph with respect to the (transformed and noisy) inner products*

$$\hat{f}_{ij} = f(|\langle a_i, a_j \rangle|) + \tau_{ij}, \quad i, j \in 1, \ldots, n \tag{17}$$

where $f : \mathbb{R}_+ \to \mathbb{R}_+$ is a strictly increasing function and $\tau_{ij} \in [-\tau, \tau]$ are bounded measurement errors. Let $\delta \geq 0$ and $\gamma \in (1, n/\log n)$ be arbitrary, and let θ be the spherical radius of a spherical cap covering $\gamma \log n/n$ fraction of the area of $\mathbb{S}^{d-1}$. Then if $q \in [3(24\pi)^{d-1}\gamma \log n + 3\frac{\mathcal{L}(\mathbb{S}^{d-2})}{\mathcal{L}(\mathbb{S}^{d-1})}\frac{n}{d-1}(2\delta)^{d-1}, n]$, $\theta \leq (\pi/2 - \delta)/24$ and $\tau \leq \{f(\cos(16\theta)) - f(\cos(16\theta + \delta))\}/2$, we have

$$\mathbb{P}\{\tilde{G} \text{ is connected }\} \geq 1 - \frac{2}{n^{\gamma-1}\gamma \log n}, \tag{18}$$

where $\mathcal{L}$ denotes the Lebesgue measure of its argument.

Proof of Lemma A.1. Following the approach taken in [11, Appendix A.B], we partition the unit sphere $\mathbb{S}^{d-1}$ into $M := n/(\gamma \log n)$ non-overlapping regions $R_1, \ldots, R_M$ of equal area with spherical diameters upper bounded as

$$\sup_{x,y \in R_m} \arccos(\langle x, y \rangle) \leq 8\theta =: \theta^*$$

for all m; the existence of such a partition was shown in [52, Lemma 6.2]. Consider the events

$$\begin{aligned} A_m &:= R_m \text{ contains at least one of } a_1, \ldots, a_n \\ B_m &:= \text{Fewer than } q/2 \text{ samples are within } 3\theta^* + \delta \text{ of } c_m \text{ in spherical distance} \end{aligned}$$

where $c_1, \ldots, c_M$ are arbitrarily chosen points in $R_1, \ldots, R_M$, respectively, and the spherical distance between two points x and y is $\arccos(\langle x, y \rangle)$. The proof proceeds as in [11, Appendix A.B] by first showing that $\tilde{G}$ is connected if A_m and B_m hold for all $m = 1, \ldots, M$. It then follows that

$$\mathbb{P}\{\tilde{G} \text{ is connected}\} \geq \mathbb{P}\{\forall m \; A_m \wedge B_m\} \geq 1 - \sum_{m=1}^{M} \mathbb{P}\{\neg A_m\} - \sum_{m=1}^{M} \mathbb{P}\{\neg B_m\} \tag{19}$$

where $\wedge$ is conjunction, $\neg$ is negation, and the second inequality follows from a union bound. The proof concludes by upper bounding $\mathbb{P}\{\neg A_m\}$ and $\mathbb{P}\{\neg B_m\}$; substituting the bounds into (19) yields the final result (18).

Implication. We show that $\tilde{G}$ is connected if A_m and B_m hold for all $m = 1, \ldots, M$, by showing that all samples in neighboring regions are connected when B_m holds for all m. Since each region contains at least one sample when A_m holds for all m, it then follows that any pair of samples is connected via a chain of connections through neighboring regions and so $\tilde{G}$ is connected.

Let a_i and a_ℓ be arbitrary samples in neighboring regions R_m and R_n. Then a_ℓ is within $2\theta^*$ of a_i in spherical distance and thus $\hat{f}_{i\ell} \geq \tilde{f}(2\theta^*) - \tau$, where we define $\tilde{f}(\alpha) = f(\cos(\alpha))$ for convenience and note that it is decreasing on $[0, \pi/2]$. Any sample a_j for which $\hat{f}_{ij} \geq \tilde{f}(2\theta^*) - \tau$ must satisfy

$$\tilde{f}(\arccos|\langle a_i, a_j \rangle|) = \hat{f}_{ij} - \tau_{ij} \geq \hat{f}_{ij} - \tau \geq \tilde{f}(2\theta^*) - 2\tau = \tilde{f}(16\theta) - 2\tau \geq \tilde{f}(16\theta + \delta) = \tilde{f}(2\theta^* + \delta) \tag{20}$$

and so must also satisfy $\arccos|\langle a_i, a_j \rangle| \leq 2\theta^* + \delta$ because $\tilde{f}$ is decreasing. Namely, any such sample must be within $2\theta^* + \delta$ of either a_i or $-a_i$, and must hence be within $3\theta^* + \delta$ of either c_m or $c_{m'}$ where $R_{m'}$ is the region containing $-a_i$. Under B_m and $B_{m'}$, there are fewer than q such samples and so all must be connected to a_i. In particular, a_ℓ must be connected to a_i, and all samples in neighboring regions are connected when B_m holds for all m.

Upper bound on $\mathbb{P}\{\neg A_m\}$. As in [11, Eqs. (27)–(28)], we use the fact that each sample falls outside of R_m with probability $1 - 1/M$ since the samples are drawn uniformly from $\mathbb{S}^{d-1}$ and the M regions have equal area. The samples are furthermore drawn independently, and so

$$\mathbb{P}\{\neg A_m\} = \left(1 - \frac{1}{M}\right)^n \leq e^{-n/M} = \frac{1}{M}\frac{1}{n^{\gamma-1}\gamma \log n}. \tag{21}$$

Upper bound on $\mathbb{P}\{\neg B_m\}$. For convenience let $\mathcal{C}_m := \{x : \arccos(\langle x, c_m\rangle) \leq 3\theta^* + \delta\}$ denote the spherical cap of spherical radius $3\theta^* + \delta$ around c_m, and let N_m denote the number of samples in $\mathcal{C}_m$. In this notation, B_m is the event that $N_m \leq q/2$. As in [11, Appendix A.B], we note that N_m is a binomially distributed random variable with n trials and probability $p := \mathcal{L}(\mathcal{C}_m)/\mathcal{L}(\mathbb{S}^{d-1})$, where $\mathcal{L}$ is the area (Lebesgue measure) of a set.

We begin by bounding $q/2$ below by $3np$; this will make applying a binomial tail bound more convenient. By assumption, $3\theta^* + \delta = 24\theta + \delta \leq \pi/2$ and so we can apply [52, Equation (5.2)] as in [11] to bound p as

$$p := \frac{\mathcal{L}(\mathcal{C}_m)}{\mathcal{L}(\mathbb{S}^{d-1})} \leq \frac{\mathcal{L}(\mathbb{S}^{d-2})}{\mathcal{L}(\mathbb{S}^{d-1})} \frac{(3\theta^* + \delta)^{d-1}}{d-1} \leq \frac{1}{2}\left(\frac{\mathcal{L}(\mathbb{S}^{d-2})}{\mathcal{L}(\mathbb{S}^{d-1})} \frac{(6\theta^*)^{d-1}}{d-1} + \frac{\mathcal{L}(\mathbb{S}^{d-2})}{\mathcal{L}(\mathbb{S}^{d-1})} \frac{(2\delta)^{d-1}}{d-1}\right) \tag{22}$$

where the second inequality follows from the convexity of x^{d-1} (when $x > 0$) applied to the convex combination $x = 3\theta^* + \delta = 1/2(6\theta^*) + 1/2(2\delta)$. The first term can be further bounded since

$$\theta^* \leq 4\pi \left((d-1)\frac{\mathcal{L}(\mathbb{S}^{d-1})}{\mathcal{L}(\mathbb{S}^{d-2})} \frac{\gamma \log n}{n}\right)^{1/(d-1)} \tag{23}$$

as in [11, Equation (31)]; the proof is the same with $3(24\pi)^{d-1}$ in place of $6(12\pi)^{d-1}$. Substituting into (22) yields

$$p \leq \frac{1}{2}\left((24\pi)^{d-1}\frac{\gamma \log n}{n} + \frac{\mathcal{L}(\mathbb{S}^{d-2})}{\mathcal{L}(\mathbb{S}^{d-1})} \frac{(2\delta)^{d-1}}{d-1}\right) \tag{24}$$

and thus

$$3np \leq \frac{1}{2}\left(3(24\pi)^{d-1}\gamma \log n + 3\frac{\mathcal{L}(\mathbb{S}^{d-2})}{\mathcal{L}(\mathbb{S}^{d-1})} \frac{n}{d-1}(2\delta)^{d-1}\right) \leq \frac{q}{2}. \tag{25}$$

Applying the binomial tail bound [53, Theorem 1] as done in [11, Equation (29)] now yields

$$\mathbb{P}\{\neg B_m\} = \mathbb{P}\{N_m > q/2\} \leq \mathbb{P}\{N_m > 3np\} \leq e^{-np} \leq e^{-n/M} = \frac{1}{M}\frac{1}{n^{\gamma-1}\gamma \log n}. \tag{26}$$

The last inequality holds since $R_m \subset \mathcal{C}_m$ and so $p = \mathcal{L}(\mathcal{C}_m)/\mathcal{L}(\mathbb{S}^{d-1}) \geq \mathcal{L}(R_m)/\mathcal{L}(\mathbb{S}^{d-1}) = 1/M$. □

Remark A.2. *An alternative bound on $(\alpha + \beta)^{d-1}$ could have been used in the proof of Lemma A.1 to shift the constants more heavily on the δ term. For example,*

$$(\alpha + \beta)^{d-1} \leq \lambda \left(\frac{\alpha}{\lambda}\right)^{d-1} + (1-\lambda)\left(\frac{\beta}{1-\lambda}\right)^{d-1} \tag{27}$$

for any $\lambda \in (0, 1)$ and taking $\lambda \approx 1$ shifts the constants heavily onto the second term. The proof of Lemma A.1 uses $\lambda = 1/2$.

We are now prepared to prove Theorem 4.5 by applying Lemma A.1 with a particular choice of δ.

Proof of Theorem 4.5. Take

$$C_3 = \frac{f(\cos(16\theta)) - f(\cos(16\theta + \delta))}{2} > 0, \tag{28}$$

where we note that θ is implicitly a function of n, d and γ, and we define

$$\delta = \min\left\{12\pi\left(\frac{d-1}{3}\frac{\mathcal{L}(\mathbb{S}^{d-1})}{\mathcal{L}(\mathbb{S}^{d-2})}\frac{\gamma \log n}{n}\right)^{1/(d-1)}, \frac{\pi}{2} - 24\theta\right\} > 0, \tag{29}$$

which is also implicitly a function of n, d and γ. Now we need only to verify that the conditions of Theorem 4.5 satisfy Lemma A.1. Note first that by construction $\delta \leq \pi/2 - 24\theta$ and so $\theta \leq (\pi/2 - \delta)/24$. Furthermore

$$3\frac{\mathcal{L}(\mathbb{S}^{d-2})}{\mathcal{L}(\mathbb{S}^{d-1})}\frac{n}{d-1}(2\delta)^{d-1} \leq (24\pi)^{d-1}\gamma \log n \tag{30}$$

and so

$$q \geq 4(24\pi)^{d-1}\gamma \log n = 3(24\pi)^{d-1}\gamma \log n + (24\pi)^{d-1}\gamma \log n \tag{31}$$

$$\geq 3(24\pi)^{d-1}\gamma \log n + 3\frac{\mathcal{L}(\mathbb{S}^{d-2})}{\mathcal{L}(\mathbb{S}^{d-1})}\frac{n}{d-1}(2\delta)^{d-1}. \tag{32}$$

Hence all conditions of Lemma A.1 are satisfied and the conclusion follows. □

A.3 Proof of Lemma 4.6

We again prove the statement for a fixed $x_i \in \mathcal{S}_l$, taking a union bound to show the condition holds for all points. First define

$$\alpha = \min_{l,\mathcal{D}:|\mathcal{D}|\leq 2s, \|a\|=1} \left\| {U_{\mathcal{D}}^{(l)}}^T U^{(l)} a \right\|_2,$$

and note that by the assumption of the lemma, there exists an $\eta > 0$ such that

$$\max_{k,l:k\neq l,\mathcal{D}:|\mathcal{D}|\leq 2s} \left\| {U_{\mathcal{D}}^{(k)}}^T U^{(l)} \right\|_2 = \alpha - \eta. \tag{33}$$

Equation (33) implies that

$$\max_{k\neq l, j} z_{i,j}^{(l,k)} \leq \alpha - \eta \tag{34}$$

deterministically. Next, we show that

$$z_{(i,q)}^{(l)} \geq \alpha - \frac{\eta}{2} \tag{35}$$

with high probability. The proof is nearly identical to [11, Lemma 1]. First, we have that

$$\begin{aligned} z_{i,j}^{(l,l)} &\sim \left\| {U_{\mathcal{D}}^{(l)}}^T U_{\mathcal{E}}^{(l)} a_i^{(l)} \right\|_2 \left| \left\langle a_i^{(l)}, a_j^{(l)} \right\rangle \right| \\ &\geq \min_{l,\mathcal{D}:|\mathcal{D}|\leq 2s, \|a\|=1} \left\| {U_{\mathcal{D}}^{(l)}}^T U^{(l)} a \right\|_2 \left| \left\langle a_i^{(l)}, a_j^{(l)} \right\rangle \right|, \end{aligned}$$

where the sets $\mathcal{D}, \mathcal{E} \subset [D]$ are the indices of the unobserved entries of $x_j^{(l)}$ and $x_i^{(l)}$, respectively. Letting $\tilde{z}_{i,j}^{(l,l)} = \left| \left\langle a_i^{(l)}, a_j^{(l)} \right\rangle \right|$, we see that

$$\begin{aligned} \mathbb{P}\left\{ z_{i,j}^{(l,l)} \leq z \right\} &\leq \mathbb{P}\left\{ \min_{l,\mathcal{D}:|\mathcal{D}|\leq 2s, \|a\|=1} \left\| {U_{\mathcal{D}}^{(l)}}^T U^{(l)} a \right\|_2 \tilde{z}_{i,j}^{(l,l)} \leq z \right\} \\ &= \mathbb{P}\left\{ \tilde{z}_{i,j}^{(l,l)} \leq \frac{z}{\alpha} \right\}. \end{aligned}$$

We can bound the probability that (35) does not hold as

$$\begin{aligned} \mathbb{P}\left\{ z_{(i,q)}^{(l)} \leq \alpha - \frac{\eta}{2} \right\} &\leq \mathbb{P}\left\{ \tilde{z}_{(i,q)}^{(l)} \leq 1 - \frac{\eta}{2\alpha} \right\} \\ &\leq \left(e\frac{N_l - 1}{q-1} \right)^{q-1} p^{N_l - q}, \end{aligned}$$

where $p = \mathbb{P}\left\{ \tilde{z}_{i,j}^{(l,l)} \leq 1 - \frac{\eta}{2\alpha} \right\}$. Setting $\xi = \frac{N_l - 1}{N_l^{\rho} - 1}$, we obtain

$$\begin{aligned} \mathbb{P}\left\{ z_j^{(l)} \leq 1 - \frac{\eta}{2\alpha} \right\} &\leq (e\xi)^{\frac{N_l-1}{\xi}} p^{(N_l-1)\left(1-\frac{1}{\xi}\right)} \\ &= \left((e\xi)^{\frac{1}{\xi}} p^{1-\frac{1}{\xi}} \right)^{N_l - 1} \\ &\leq e^{-(N_l-1)c_1}, \end{aligned}$$

where the last inequality holds for a constant $c_1 > 0$ as long as

$$(e\xi)^{\frac{1}{\xi}} p^{1-\frac{1}{\xi}} < 1 \Leftrightarrow (e\xi)^{-\frac{1}{\xi-1}} > p.$$

This inequality can be satisfied for every $p < 1$ by taking N_0, and consequently ξ, sufficiently large. By inspection, we have $p < 1$ as long as $\eta > 0$, which is true by assumption of the lemma.

By monotonicity of f, (34) implies that

$$f\left(\max_{k\neq l,j} z_{i,j}^{(l,k)}\right) \leq f(\alpha - \eta)$$

and (35) implies that

$$f\left(z_{(i,q)}^{(l)}\right) \geq f\left(\alpha - \frac{\eta}{2}\right).$$

Finally, we have that

$$\begin{aligned} C_{i,l} &:= f\left(z_{(i,q)}^{(l)}\right) - f\left(\max_{k\neq l,j} z_{i,j}^{(l,k)}\right) \\ &\geq f\left(\alpha - \frac{\eta}{2}\right) - f(\alpha - \eta) > 0, \end{aligned}$$

where the second line follows by monotonicity of f, noting that $\alpha - \eta/2 > \alpha - \eta$. Taking $C_1 = \min_{l\in[K],i\in[N_l]} C_{i,l}/2$ and a union bound completes the proof.

A.4 Proof of Lemma 4.7

We again prove the statement for a fixed $x_i \in \mathcal{S}_l$, with a union bound completing the proof. Let $\nu = 2/3$, $N_l \geq 6q$, and $c_2 > 1/20$. From [11, Appendix C], we have that

$$z_{(i,q)}^{(l)} \geq \frac{\nu}{\sqrt{d_l}} - \varepsilon \tag{36}$$

and

$$\max_{k\neq l,j} z_j^{(k)} \leq \alpha + \varepsilon \tag{37}$$

with probability at least $1 - e^{-c_2(N_l-1)} - 10Ne^{-\beta^2/2}$, where

$$\alpha = \frac{\beta(1+\beta)}{\sqrt{d_l}} \max_{k\neq l} \frac{1}{\sqrt{d_k}} \left\| U^{(k)^T} U^{(l)} \right\|_F,$$

$$\varepsilon = \frac{2\sigma(1+\sigma)}{\sqrt{D}}\beta$$

and $\frac{1}{\sqrt{2\pi}} \leq \beta \leq \sqrt{D}$. Let $\beta = \sqrt{6\log N}$ and note that $D \geq 6\log N$ implies $\beta \leq \sqrt{D}$. Noting that $q < N_{min}/6$ implies $N > 6$, we have $(1+\beta) < 4\sqrt{\log N}$. These are sufficient to guarantee that $\alpha + \varepsilon < \frac{\nu}{\sqrt{d_l}} - \varepsilon$. By monotonicity of f, (37) implies that

$$f\left(\max_{k\neq l,j} z_{i,j}^{(l,k)}\right) \leq f(\alpha + \varepsilon)$$

and (36) implies that

$$f\left(z_{(i,q)}^{(l)}\right) \geq f\left(\frac{\nu}{\sqrt{d_l}} - \varepsilon\right).$$

Finally, we have that

$$\begin{aligned} C_{i,l} &:= f\left(z_{(i,q)}^{(l)}\right) - f\left(\max_{k\neq l,j} z_{i,j}^{(l,k)}\right) \\ &\geq f\left(\frac{\nu}{\sqrt{d_l}} - \varepsilon\right) - f(\alpha + \varepsilon) > 0, \end{aligned}$$

where the second line follows by monotonicity of f. Taking $C_2 = \min_{l\in[K],i\in[N_l]} C_{i,l}/2$ and a union bound completes the proof.

A.5 Proof of Theorem 4.8

By Lemma 4.9, the expected entries of the co-association matrix obtained by EKSS-0 are an increasing function of the inner product between points. It remains to show how tightly these values concentrate around their mean. This concentration allows us to bound the noise level τ via the following lemma.

Lemma A.3. *Let A be the affinity matrix formed by EKSS-0 (line 12, Alg. 1). For two points $x_i, x_j \in \mathcal{X}$, let*

$$f_{\bar{K},\bar{d}}\left(|\langle x_i, x_j\rangle|\right) = \mathbb{E}A_{i,j} = \mathbb{P}\left\{x_i, x_j \textit{ co-clustered}\right\}$$

and

$$\hat{f}_{i,j} = A_{i,j} = \frac{1}{B}\sum_{b=1}^{B} \mathbb{1}\left\{x_i, x_j \textit{ co-clustered in } \mathcal{C}^{(b)}\right\}.$$

Then for all $\tau > 0$

$$\mathbb{P}\left\{\left|\hat{f}_{i,j} - f_{\bar{K},\bar{d}}\left(|\langle x_i, x_j\rangle|\right)\right| > \tau\right\} < 2e^{-c_3\tau^2 B}, \tag{38}$$

where $c_3 = 2\sqrt{\log 2}$ and the randomness is with respect to the subspaces drawn in EKSS-0 (line 4, Alg. 1).

Proof. The proof relies on sub-Gaussian concentration. The measurements $\hat{f}$ are bounded and hence sub-Gaussian with parameter $\frac{1}{\sqrt{\log 2}}$. Note that $\hat{f}_{i,j}$ is the empirical estimate of $f_{\bar{K},\bar{d}}\left(|\langle x_i, x_j\rangle|\right)$, and thus $\mathbb{E}\hat{f}_{i,j} = f_{\bar{K},\bar{d}}\left(|\langle x_i, x_j\rangle|\right)$. Therefore, by the General form of Hoeffding's inequality [54, Theorem 2.6.2]

$$\mathbb{P}\left\{\left|\hat{f}_{i,j} - \mathbb{E}\hat{f}_{i,j}\right| > \tau\right\} \leq 2e^{-c_3\tau^2 B},$$

where $c_3 = 2\sqrt{\log 2}$. □

Combining the results of Theorem 4.9 and Lemma A.3 shows that the (i,j)th entry of the affinity matrix is τ-angle preserving with high probability for a single point. A union bound over all $N(N-1)/2$ unique pairs completes the proof.

A.6 Proof of Lemma 4.9

For notational compactness, we instead prove that the probability is a *decreasing* function of the angle θ between points and note that $z = \cos(\theta)$. Let $U_1, U_2, \ldots, U_K \in \mathbb{R}^{D\times d}$ be the K candidate bases. Let $\tilde{p}(\theta)$ be the probability that two points that are at angle θ apart are assigned to the candidate U_1. Then we clearly have $p_{K,D}(\theta) = K\tilde{p}(\theta)$, and it suffices to prove that $\tilde{p}$ is strictly decreasing.

Let $e_1, \ldots, e_D$ be the standard basis vectors in $\mathbb{R}^D$. For a given θ, set $x_i := e_1$, and $x_j = x_j(\theta) := \cos(\theta)e_1 + \sin(\theta)e_2$. By definition, for any orthogonal transformation Q of $\mathbb{R}^D$,

$$\tilde{p}(\theta) = \mathbb{P}\left\{Qx_i, Qx_j \text{ both assigned to } U_1\right\}.$$

We may average out this equation over a choice subgroup of orthogonal matrices. Indeed, let L denote the span of e_1 and e_2, and let Q be a random matrix uniformly distributed over the set of orthogonal matrices that decompose into a rotation on L and the identity on $L^\perp$. We take expectations with respect to Q and exchange the order of integration to get

$$\begin{aligned}\tilde{p}(\theta) &= \mathbb{E}_Q\mathbb{P}_{U_1,\ldots,U_K}\left\{Qx_i, Qx_j \text{ both assigned to } U_1\right\}\\ &= \mathbb{E}_{U_1,\ldots,U_K}\mathbb{P}\left\{Qx_i, Qx_j \text{ both assigned to } U_1 \mid U_1, \ldots, U_K\right\}.\end{aligned}$$

Now fix $U_1, \ldots, U_K$. Let $A = A(\theta)$ be the event that Qx_i and $Qx_j(\theta)$ are both assigned to U_1. We claim that $\mathbb{P}\left\{A(\theta) \mid U_1, \ldots, U_K\right\}$ is non-increasing in θ. To see this, let us examine the event more closely. By the definition of candidate assignment, A occurs when U_1 is the *closest* candidate to both x_i and x_j. More mathematically, this is when

$$\left\|P_{U_1}Qz\right\|_2^2 > \left\|P_{U_k}Qz\right\|_2^2, \quad \text{for} \quad 1 < k \leq K, \quad \text{and} \quad z = x_i, x_j. \tag{39}$$

Here, we use P_F to denote the orthogonal projection onto a subspace F.

We shall attempt to rewrite (39) in a more useful form. First, observe that

$$\begin{aligned}\|P_{U_1}Qz\|_2^2 - \|P_{U_k}Qz\|_2^2 &= z^TQ^TP_{U_1}^TP_{U_1}z - z^TP_{U_k}^TP_{U_k}Qz \\ &= z^TQ^TP_L\left(P_{U_1}^TP_{U_1} - P_{U_k}^TP_{U_k}\right)P_L^TQz. \qquad (40)\end{aligned}$$

Let us also introduce some new notation. We use $\tilde{x}_i$ and $\tilde{x}_j$ to denote the two-dimensional coordinate vectors of x_i and x_j with respect to e_1 and e_2, we let $\tilde{Q}$ denote the restriction of Q to L, and similarly let $\tilde{P}_L$ be the projection P_L treated as a map from $\mathbb{R}^D$ to $\mathbb{R}^2$. We therefore have

$$z^TQ^TP_L\left(P_{U_1}^TP_{U_1} - P_{U_k}^TP_{U_k}\right)P_L^TQz = \tilde{z}^T\tilde{Q}^TM_k\tilde{Q}\tilde{z},$$

where $M_k := \tilde{P_L}\left(P_{U_1}^TP_{U_1} - P_{U_k}^TP_{U_k}\right)\tilde{P_L}^T$. Following these calculations, we see that (39) is equivalent to

$$\tilde{z}^T\tilde{Q}^TM_k\tilde{Q}\tilde{z} > 0, \quad \text{for} \quad 1 < k \le K, \quad \text{and} \quad \tilde{z} = \tilde{x}_i, \tilde{x}_j. \qquad (41)$$

When $\tilde{Q}$ is fixed, denote by $A_{\tilde{Q}}$ the event over which (41) holds.

Observe that M_k is a 2 by 2 real symmetric matrix. As such, the set S_k of points $\tilde{z}$ in $\mathbb{R}^2$ for which $\tilde{z}^TM_k\tilde{z} > 0$ comprises the union of two (possibly degenerate) antipodal *sectors*. The same is true for the intersection $S := \cap_{k>1}S_k$. Let $\phi = \phi(U_1, \dots, U_K)$ denote the angle spanned by one of the two sectors comprising S, and note that $0 \le \phi \le \pi$. Furthermore, let T be the union of the sector spanned by $\tilde{x}_i$ and $\tilde{x}_j$ with its antipodal reflection. Then $A_{\tilde{Q}}$ holds if and only if $\tilde{Q}T \subset S$ or $S^c \subset \tilde{Q}T$. It is a simple exercise to compute

$$\begin{aligned}\mathbb{P}\left\{\tilde{Q}T \subset S \mid U_1, \dots, U_K\right\} &= \frac{(\phi - \theta)_+}{\pi}, \\ \mathbb{P}\left\{S^c \subset \tilde{Q}T \mid U_1, \dots, U_K\right\} &= \frac{(\theta - \pi + \phi)_+}{\pi}.\end{aligned}$$

Since A is the disjoint union of these events, we have

$$\mathbb{P}\left\{A(\theta) \mid U_1, \dots, U_K\right\} = \frac{(\phi - \theta)_+}{\pi} + \frac{(\theta - \pi + \phi)_+}{\pi}. \qquad (42)$$

Differentiating at any point other than the obvious discontinuities, we have

$$\begin{aligned}\frac{d}{d\theta}\mathbb{P}\left\{A(\theta) \mid U_1, \dots, U_K\right\} &= \frac{d}{d\theta}\frac{(\phi - \theta)_+}{\pi} + \frac{(\theta - \pi + \phi)_+}{\pi} \\ &= -\frac{1}{\pi}1_{(0,\phi)}(\theta) + \frac{1}{\pi}1_{(\pi-\phi,\pi/2)}(\theta) \\ &= -\frac{1}{\pi} + \frac{1}{\pi}1_{(\phi,\pi/2)}(\theta) + \frac{1}{\pi}1_{(\pi-\phi,\pi/2)}(\theta) \\ &\le 0.\end{aligned}$$

Here, the last inequality follows from the fact that either $\phi \ge \pi/2$ or $\pi - \phi > \pi/2$, thereby completing the proof of the claim. Recalling that $\tilde{p}(\theta) = \mathbb{E}_{U_1,\dots,U_K}\mathbb{P}\left\{A(\theta) \mid U_1, \dots, U_K\right\}$, we have thus proved that $\tilde{p}$ is non-increasing. To see that it is strictly decreasing, simply note that $\frac{d}{d\theta}\mathbb{P}\left\{A(\theta) \mid U_1, \dots, U_K\right\} < 0$ whenever $\phi(U_1, \dots, U_K) < \pi/2$. This occurs on a set of positive measure.

A.7 Proof of Theorem 4.10

By Theorem 4.8, the co-association matrix $\bar{A}$ is τ-angle preserving with high probability. Applying Lemma 4.6 with $s = 0$, we obtain $C_1 > 0$ that lower bounds the separation ϕ_q defined in (6) with high probability. Applying Lemma 4.5 with $\gamma = 3$, we obtain $C_3 > 0$ such that the components corresponding to each subspace are connected with high probability. Setting $\tau = \min\{C_1, C_3\}$ in Theorem 4.8 completes the proof.

A.8 Proof of Theorem 4.11

By Theorem 4.8, the co-association matrix $\bar{A}$ is τ-angle preserving with high probability. Applying Lemma 4.7 with $\sigma = 0$, we obtain $C_2 > 0$ that lower bounds the separation ϕ_q defined in (6) with high probability. Applying Lemma 4.5 with $\gamma = 3$, we obtain $C_3 > 0$ such that the components corresponding to each subspace are connected with high probability. Setting $\tau = \min\{C_1, C_3\}$ in Theorem 4.8 completes the proof.

A.9 Proof of Theorem 4.12

By Theorem 4.8, the co-association matrix $\bar{A}$ is τ-angle preserving with high probability. Applying Lemma 4.7, we obtain $C_2 > 0$ that lower bounds the separation ϕ_q defined in (6) with high probability. Setting $\tau = \min\{C_1, C_3\}$ in Theorem 4.8 completes the proof.

A.10 Proof of Theorem 4.13

By Theorem 4.8, the co-association matrix $\bar{A}$ is τ-angle preserving with high probability. By [11, Lemma 4], the condition (8) holds with probability at least $1 - 4e^{-c_7 D}$ as long as (14) is satisfied. Thus, applying Lemma 4.6 with the parameters $N_k = n$, $d_k = d$ for all k, the result holds with the specified probability.

B Algorithmic and Simulation Details

In this section, we include implementation details beyond those included in the main body. We first provide pseudocode for the Thresh and EKSS-0 algorithms. We then describe all preprocessing steps and parameters used for our experiments on real data.

B.1 Pseudocode

In Algorithm 2 is the pseudocode for the Thresh routine used in the EKSS algorithm, which results in the same connectivity as thresholding in TSC [11]. Algorithm 3 gives the pseudocode for the EKSS-0 algorithm, which is analyzed in Section 4.

Algorithm 2 Affinity Threshold (Thresh)

```
Input: $A \in [0,1]^{N \times N}$: affinity matrix, $q$: threshold parameter
Output: $\bar{A} \in [0,1]^{N \times N}$: thresholded affinity matrix
for $i = 1, \ldots, N$ do
    $Z^{\text{row}}_{i,:} \leftarrow A_{i,:}$ with the smallest $N - q$ entries set to zero.        Threshold rows
    $Z^{\text{col}}_{:,i} \leftarrow A_{:,i}$ with the smallest $N - q$ entries set to zero.        Threshold columns
end for
$\bar{A} \leftarrow \frac{1}{2}\left(Z^{\text{row}} + Z^{\text{col}}\right)$        Average
```

B.2 Clustering Error

The clustering error, which is the metric used for all experimental results, is computed by matching the true labels and the labels output by a given clustering algorithm,

$$\text{err} = \frac{100}{N}\left(1 - \max_{\pi} \sum_{i,j} Q^{\text{out}}_{\pi(i)j} Q^{\text{true}}_{ij}\right),$$

where π is a permutation of the cluster labels, and Q^{out} and Q^{true} are the output and ground-truth labelings of the data, respectively, where the (i,j)th entry is one if point j belongs to cluster i and is zero otherwise.

Algorithm 3 EKSS-0

```
Input: $\mathcal{X} = \{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^D$: data, $\bar{K}$: number of candidate subspaces, $\bar{d}$: candidate dimension,
    $K$: number of output clusters, $q$: threshold parameter, $B$: number of base clusterings,
Output: $\mathcal{C} = \{c_1, \ldots, c_K\}$: clusters of $\mathcal{X}$
for $b = 1, \ldots, B$ (in parallel) do
    $U_1, \ldots, U_{\bar{K}} \overset{iid}{\sim} \mathrm{Unif}(\mathrm{St}(D, \bar{d}))$        Draw $\bar{K}$ random subspace bases
    $c_k \leftarrow \left\{ x \in \mathcal{X} \ : \ \forall j \ \left\| U_k^T x \right\|_2 \geq \left\| U_j^T x \right\|_2 \right\}$ for $k = 1, \ldots, \bar{K}$        Cluster by projection
    $\mathcal{C}^{(b)} \leftarrow \{c_1, \ldots, c_{\bar{K}}\}$
end for
$A_{i,j} \leftarrow \frac{1}{B} \left| \left\{ b : x_i, x_j \text{ are co-clustered in } \mathcal{C}^{(b)} \right\} \right|$ for $i, j = 1, \ldots, N$        Form affinity matrix
$\bar{A} \leftarrow \textsc{Thresh}(A, q)$        Keep top $q$ entries per row/column
$\mathcal{C} \leftarrow \textsc{SpectralClustering}(\bar{A}, K)$        Final Clustering
```

Dataset	N	K	D
Hopkins-155	39-556	2-3	30-200
Yale	2432	38	2016
COIL-20	1440	20	1024
COIL-100	7200	100	1024
USPS	9298	10	256
MNIST-10k	10000	10	500

Table 2: Datasets used for experiments with relevant parameters; N: total number of samples, K: number of clusters, D: ambient dimension.

B.3 Experiments on Benchmark Data

In this section, we describe the benchmark datasets used in our experiments, as well as any preprocessing steps and the parameters selected for all algorithms. All datasets are normalized so that each column lies on the unit sphere in the corresponding ambient dimension, as is common in the literature [11,21,27]. Table 2 gives a summary of all datasets considered.

The Hopkins-155 dataset [2] consists of 155 motion sequences with $K = 2$ in 120 of sequences and $K = 3$ in the remaining 35. In each sequence, objects moving along different trajectories each lie near their own affine subspace of dimension at most 3. We perform no preprocessing steps on this dataset.

The Extended Yale Face Database B [4,45] consists of 64 images of each of 38 different subjects under a variety of lighting conditions. Each image is of nominal size 192 × 168 and is known to lie near a 9-dimensional subspace [1]. We downsample so that each image is of size 48 × 42, as in [16]. For EKSS, KSS, CoP-KSS, MKF, and TSC, we perform an initial whitening as in [11,25] by removing the first two singular components of the dataset and then project the data onto its first 500 principal components to reduce the computational complexity of these methods. Whitening resulted in worse performance for all other algorithms, so we omitted this step.

Algorithm	Hopkins	Yale	COIL-20	COIL-100	USPS	MNIST-10k
EKSS	$d = 3, q = 2$	$d = 2, q = 6$	$d = 2, q = 6$	$d = 8, q = 7$	$d = 13, q = 3$	$d = 13, q = 72$
KSS	$d = 3$	$d = 3$	$d = 1$	$d = 5$	$d = 9$	$d = 13$
CoP-KSS	$d = 4$	$d = 6$	$d = 9$	$d = 1$	$d = 7$	$d = 18$
MKF	$d = 3$	$d = 17$	$d = 19$	$d = 18$	$d = 20$	$d = 20$
TSC	$q = 3$	$q = 3$	$q = 4$	$q = 4$	$q = 3$	$q = 3$
SSC-ADMM	$\rho = 0.1, \alpha = 226.67$	$\rho = 0.1, \alpha = 670$	$\rho = 0.8, \alpha = 5$	$\rho = 1, \alpha = 20$	$\rho = 1, \alpha = 20$	$\rho = 1, \alpha = 20$
SSC-OMP	$\varepsilon = 2^{-52}, k_{max} = 2$	$\varepsilon = 2^{-52}, k_{max} = 2$	$\varepsilon = 2^{-52}, k_{max} = 2$	$\varepsilon = 2^{-52}, k_{max} = 2$	$\varepsilon = 2^{-52}, k_{max} = 29$	$\varepsilon = 2^{-52}, k_{max} = 17$
EnSC	$\lambda = 0.01, \alpha = 98$	$\lambda = 0.88, \alpha = 3$	$\lambda = 0.99, \alpha = 3$	$\lambda = 0.95, \alpha = 3$	$\lambda = 0.95, \alpha = 50$	$\lambda = 0.95, \alpha = 3$

Table 3: Parameters used in experiments on real datasets for all algorithms considered.

The COIL-20 [46] and COIL-100 [47] datasets consist of 72 images of 20 and 100 distinct objects (respectively) under a variety of rotations. All images are of size 32 × 32. On both datasets, we whiten by removing the first singular component when it improves algorithm performance.

The USPS dataset provided by [48] contains 9,298 total handwritten digits of size 16 × 16 with roughly even label distribution. No preprocessing is performed on this dataset.

The MNIST dataset [5] contains a total of 70,000 handwritten digits, of which we consider only the 10,000 "test" images. The images have nominal size 29 × 29, and we use the output of the scattering convolutional network [49] of size 3,472 and then project onto the first 500 principal components as in [18].

For all algorithms, we set K to be the correct number of clusters. For EKSS, we set $B = 1000$ and $T = 3$ for all datasets except MNIST, for which we set $T = 30$. Due to the benefits demonstrated in [13], we employed CoP-KSS instead of KSS as a base clustering algorithm for the USPS and MNIST datasets. For a fair comparison to KSS, CoP-KSS, and MKF, we ran 1000 trials of each and use the clustering result that achieves the lowest clustering error. The parameters used for all experiments are shown in Table 3, with the most common parameters given among the 155 datasets for the Hopkins database. For the Hopkins, Yale, and COIL-20 datasets, we performed extensive model sweeps over a wide range of values for each parameter for each algorithm. For the larger COIL-100, USPS, and MNIST-10k datasets, this was infeasible for SSC-ADMM and EnSC, so the values were instead chosen from an intelligently-selected subset of parameters.

References

[1] R. Basri and D. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 218–233, February 2003.

[2] R. Tron and R. Vidal, "A benchmark for the comparison of 3-D motion segmentation algorithms," in *IEEE Int. Conf. on Comp. Vision and Pattern Recog.*, 2011.

[3] R. Vidal, S. S. Sastry, and Y. Ma, *Generalized Principal Component Analysis.* Springer-Verlag, 2016.

[4] A. Georghiades, P. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.

[5] Y. LeCun, C. Cortes, and C. J. C. Burges, "The MNIST database of handwritten digits," 2016. [Online]. Available: yann.lecun.com/exdb/mnist

[6] P. S. Bradley and O. L. Mangasarian, "k-Plane clustering," *Journal of Global Optimization*, vol. 16, pp. 23–32, 2000.

[7] P. Tseng, "Nearest q-flat to m points," *Journal of Optimization Theory and Applications*, vol. 105, no. 1, pp. 249–252, 2000.

[8] P. K. Agarwal and N. H. Mustafa, "K-means projective clustering," in *Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems.* ACM, 2004, pp. 155–165.

[9] T. Zhang, A. Szlam, and G. Lerman, "Median k-flats for hybrid linear modeling with many outliers," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on.* IEEE, 2009, pp. 234–241.

[10] D. Park, C. Caramanis, and S. Sanghavi, "Greedy subspace clustering," in *Advances in Neural Information Processing Systems*, 2014, pp. 2753–2761.

[11] R. Heckel and H. Bölcskei, "Robust subspace clustering via thresholding," *IEEE Trans. Inf. Theory*, vol. 24, no. 11, pp. 6320–6342, 2015.

[12] A. Jalali and R. Willett, "Subspace clustering via tangent cones," in *Advances in Neural Information Processing Systems*, 2017, pp. 6744–6753.

[13] A. Gitlin, B. Tao, L. Balzano, and J. Lipor, "Improving k-subspaces via coherence pursuit," *IEEE Journal of Selected Topics in Signal Processing*, 2018, accepted for publication.

[14] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 663–670.

[15] C.-Y. Lu, H. Min, Z.-Q. Zhao, L. Zhu, D.-S. Huang, and S. Yan, "Robust and efficient subspace segmentation via least squares regression," *Computer Vision–ECCV 2012*, pp. 347–360, 2012.

[16] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 2765–2781, Nov. 2013.

[17] C. You, D. P. Robinson, and R. Vidal, "Scalable sparse subspace clustering by orthogonal matching pursuit," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2015.

[18] C. You, C.-G. Li, D. P. Robinson, and R. Vidal, "Oracle based active set algorithm for scalable elastic net subspace clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3928–3937.

[19] M. Rahmani and G. K. Atia, "Coherence pursuit: Fast, simple, and robust principal component analysis," *IEEE Transactions on Signal Processing*, vol. 65, no. 23, pp. 6260–6275, 2017.

[20] A. L. Fred and A. K. Jain, "Combining multiple clusterings using evidence accumulation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 6, pp. 835–850, 2005.

[21] M. Soltanolkotabi and E. J. Candes, "A Geometric Analysis of Subspace Clustering with Outliers," *The Annals of Statistics*, vol. 40, no. 4, pp. 2195–2238, 2012.

[22] R. Vidal and P. Favaro, "Low rank subspace clustering (lrsc)," *Pattern Recognition Letters*, vol. 43, pp. 47–61, 2014.

[23] J. Shen, P. Li, and H. Xu, "Online low-rank subspace clustering by basis dictionary pursuit," in *Proc. International Conference on Machine Learning*, 2016.

[24] Y. Wang, Y.-X. Wang, and A. Singh, "Graph connectivity in noisy sparse subspace clustering," *CoRR abs/1504.01046*, 2016.

[25] T. Zhang, A. Szlam, Y. Wang, and G. Lerman, "Hybrid linear modeling via local best-fit flats," *International Journal of Computer Vision*, vol. 100, pp. 217–240, 2012.

[26] L. Balzano, A. Szlam, B. Recht, and R. Nowak, "K-Subspaces with missing data," in *Statistical Signal Processing Workshop (SSP), 2012 IEEE.* IEEE, 2012, pp. 612–615.

[27] J. He, Y. Zhang, J. Wang, N. Zeng, and H. Hao, "Robust k-subspaces recovery with combinatorial initialization," in *Big Data (Big Data), 2016 IEEE International Conference on.* IEEE, 2016, pp. 3573–3582.

[28] J. Ghosh and A. Acharya, "Cluster ensembles," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 4, pp. 305–315, 2011.

[29] F. Leisch, "Bagged clustering," WU Vienna University of Economics and Business, Tech. Rep. 51, 1999.

[30] B. Minaei-Bidgoli, A. Topchy, and W. F. Punch, "Ensembles of partitions via data resampling," in *Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004. International Conference on*, vol. 2. IEEE, 2004, pp. 188–192.

[31] K. Tumer and A. K. Agogino, "Ensemble clustering with voting active clusters," *Pattern Recognition Letters*, vol. 29, no. 14, pp. 1947–1953, 2008.

[32] A. Topchy, A. K. Jain, and W. Punch, "Clustering ensembles: Models of consensus and weak partitions," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, no. 12, pp. 1866–1881, 2005.

[33] A. Fred and A. K. Jain, "Evidence accumulation clustering based on the k-means algorithm," in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR).* Springer, 2002, pp. 442–451.

[34] A. L. Fred and A. K. Jain, "Data clustering using evidence accumulation," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 4. IEEE, 2002, pp. 276–280.

[35] S. R. Bulò, A. Lourenço, A. Fred, and M. Pelillo, "Pairwise probabilistic clustering using evidence accumulation," in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR).* Springer, 2010, pp. 395–404.

[36] A. Lourenço, S. R. Bulò, N. Rebagliati, A. L. Fred, M. A. Figueiredo, and M. Pelillo, "Probabilistic evidence accumulation for clustering ensembles." in *ICPRAM*, 2013, pp. 58–67.

[37] ——, "Probabilistic consensus clustering using evidence accumulation," *Machine Learning*, vol. 98, no. 1-2, pp. 331–357, 2015.

[38] A. Ng, Y. Weiss, and M. Jordan, "On spectral clustering: Analysis and an algorithm," in *Proc. Neural Information Processing Systems*, 2001.

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

[40] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography," *Int'l J. Computer Vision*, vol. 9, no. 2, pp. 137–154, 1992.

[41] X. Wang and G. Lerman, "Fast landmark subspace clustering," *arXiv preprint arXiv:1510.08406*, 2015.

[42] D. Zhang and L. Balzano, "Global convergence of a grassmannian gradient descent algorithm for subspace estimation," in *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, vol. 51. Cadiz, Spain: PMLR, 09–11 May 2016, pp. 1460–1468.

[43] M. Soltanolkotabi and E. J. Candes, "Robust Subspace Clustering," *The Annals of Statistics*, vol. 42, no. 2, pp. 669–699, 2014.

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

[45] K. Lee, J. Ho, and D. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Anal. Mach. Intelligence*, vol. 27, no. 5, pp. 684–698, 2005.

[46] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (COIL-20)," Columbia University, Tech. Rep. CUCS-005-96, Feb. 1996.

[47] ——, "Columbia object image library (COIL-100)," Columbia University, Tech. Rep. CUCS-006-96, Feb. 1996.

[48] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized non-negative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1548–1560, 2011.

[49] J. Bruna and S. Mallat, "Invariant scattering convolution networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1872–1886, 2013.

[50] J. Lipor and L. Balzano, "Leveraging union of subspace structure to improve constrained clustering," in *International Conference on Machine Learning*, 2017, pp. 2130–2139.

[51] ——, "Clustering quality metrics for subspace clustering," in *Asilomar Conference on Signals, Systems, and Computers*, 2018, accepted for publication. [Online]. Available: http://web.cecs.pdx.edu/~lipor/Papers/lipor2018clustering.pdf

[52] P. Leopardi, “Diameter bounds for equal area partitions of the unit sphere,” *Electronic Transactions on Numerical Analysis*, pp. 1–16, 2009.

[53] S. Janson, “On concentration of probability,” *Contemporary Combinatorics*, pp. 289–301, 2002.

[54] R. Vershynin, *High-dimensional probability: An introduction with applications in data science.* Cambridge University Press, 2018, vol. 47.