Fusion over the Grassmann Manifold for Incomplete-Data Clustering

Anonymous authors
Paper under double-blind review

Abstract

This paper presents a new paradigm to cluster incomplete vectors using subspaces as proxies to exploit the geometry of the Grassmannian. We leverage this new perspective to develop an algorithm to cluster and complete data in a union of subspaces via a fusion penalty formulation. Our approach does not require prior knowledge of the number of subspaces, is naturally suited to handle noise, and only requires an upper bound on the subspaces' dimensions. In developing our model, we present local convergence guarantees. We describe clustering, completion, model selection, and sketching techniques that can be used in practice, and complement our analysis with synthetic and real-data experiments.

1 Introduction

Suppose we observe a subset of entries in a data matrix $\mathbf{X}$ whose columns lie near a union of subspaces, for example:

$$\begin{bmatrix} 1 & -4 & 6 & 9 & 16 & * & 8 & * & * \\ * & 5 & 4 & * & 16 & 5 & * & * & * \\ * & 7 & 6 & -12 & 2 & 18 & -1 & * & * \\ 2 & * & 5 & * & * & * & 1 & 0 & -1 \\ 5 & * & * & 6 & 19 & 9 & 5 & 2 & -3 \\ 8 & 1 & * & 7 & * & 14 & * & -13 & 8 \end{bmatrix},$$

where the unobserved entries are marked with $*$. Our goals are (i) to complete the unobserved entries, (ii) to cluster the columns according to the subspaces, and (iii) to learn the underlying subspaces. In the example above, we should (i) obtain the following (ground truth) completion:

$$\mathbf{X} = \begin{bmatrix} 1 & -4 & 6 & 9 & 16 & -1 & 8 & -7 & 3 \\ 1 & 5 & 4 & 14 & 16 & 5 & -7 & -18 & 10 \\ 8 & 7 & 6 & -12 & 2 & 18 & -1 & 28 & -18 \\ 2 & 1 & 5 & 4 & 11 & 4 & 1 & 0 & -1 \\ 5 & 1 & 9 & 6 & 19 & 9 & 5 & 2 & -3 \\ 8 & 1 & -1 & 7 & 3 & 14 & 9 & -13 & 8 \end{bmatrix},$$

we should also (ii) cluster the columns of $\mathbf{X}$ into two groups, $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_6, \mathbf{x}_7\}$ and $\{\mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_8, \mathbf{x}_9\}$, and (iii) obtain bases for two 2-dimensional subspaces (given by any subset of linearly independent columns from each group).

This problem is often known as *high-rank matrix completion* (HRMC) [24, 21] or as *subspace clustering with missing data*, and it has a wide range of applications, including tracking moving objects in computer vision [13, 14, 30, 31, 33, 35, 42], predicting target interactions for drug discovery [28, 44, 45, 49], and identifying groups in recommender systems [37, 66, 77]. While there exists theory detailing conditions under which the HRMC goals above are feasible (e.g., sufficient sampling and subspaces genericity) [51], existing algorithms present a variety of shortcomings (more details in Section 2 below).

The fundamental difficulty that all HRMC approaches face lies in assessing distances (e.g., euclidean, or in the form of inner products) between partially observed vectors, for the simple reason that this requires overlapping observations which become increasingly unlikely in low-sampling regimes [24]. **To circumvent this problem,** we introduce a new paradigm to cluster incomplete vectors, using subspaces as proxies, thus avoiding the need to calculate distances or inner products or other notions of similarity between incomplete vectors, as other methods require. To this end we assign each (incomplete-data) point its own (full-data) subspace, and simultaneously minimize over the Grassmann manifold: (a) the chordal distance between each point and its assigned subspace, to guarantee that the subspace stays *near* the observed entries, and (b) the geodesics between subspaces of all data, to encourage the subspaces from points that belong together to *fuse* (i.e, represent the same space). At the end of this minimization, clustering the proxy subspaces using standard procedures like k-means or spectral clustering [9, 23, 56, 61, 67, 71, 73] can be done as a proxy for clustering the incomplete-data (goal *ii*). The ability to cluster the subspaces rather than the incomplete-data is the key strength we gain by moving to the Grassmannian. After clustering, the missing entries can be filled (goal *i*) using low-rank matrix completion. Once the data is clustered and completed, the underlying subspaces can be trivially inferred (goal *iii*) with a singular value decomposition. Local convergence guarantees follow easily from known manifold optimization results. We complement our theoretical results with experiments on both synthetic and real data that show the potential of the foundational fusion-over-the-Grassmann formulation.

2 Related Work

Due to its broad applicability, HRMC has attracted considerable attention in recent years. Existing approaches can be divided in three main groups: generalizations from *low-rank matrix completion* (LRMC), generalizations from subspace clustering (SC), and methods specifically tailored for HRMC (see [38] for a recent survey).

	Number of Subspaces	
	1	**K**
Full-Data	PCA	SC
Missing Data	LRMC	**HRMC (This paper)**

Figure 1: HRMC is a generalization of principal component analysis (PCA), LRMC, and SC.

HRMC vs LRMC. LRMC seeks to exactly recover the missing entries of a data matrix $\mathbf{X}$ whose columns lie in a single low-dimensional subspace [11]. One can view HRMC as a generalization of LRMC, where the columns of $\mathbf{X}$ are known to lie in a *union of subspaces* (UoS), each of low dimension, but it is not known to which subspace each column belongs (see Figure 1). Research in LRMC over the last decades has resulted in theory and algorithms that guarantee perfect recovery under reasonable assumptions (e.g., random sampling and bounded-coherence of the data) [10, 11, 15, 16, 29, 53]. Hence, given a HRMC problem, if the number of underlying subspaces, say K, and the maximum of their dimension, say r, are low, one could be tempted to cast HRMC as a LRMC problem. In such case, the single subspace containing all the columns of $\mathbf{X}$ would have dimension no larger than $\mathrm{r}' := \mathrm{r} \cdot \mathrm{K}$. This would, however, completely ignore the union structure present in the data, and therefore require more observed entries in order to complete $\mathbf{X}$. We can see this by noting that each column must have more observed entries than the subspace containing it [51]. This means that even in the fortunate case where r' is low enough, using LRMC would require K times more observations than HRMC. This is especially prohibitive in applications such as Metagenomics or Drug Discovery, where data is extremely sparse and costly to acquire. In general, r' may be too large to even allow the use of LRMC.

HRMC vs SC. SC aims to cluster the columns of a full-data matrix $\mathbf{X}$ according to a UoS that is not known a priori [22]. One can thus view HRMC as the generalization of SC to the case where data is missing (see Figure 1). There exists a vast repertoire of theory and algorithms that guarantee perfect clustering under reasonable assumptions (e.g., sufficient sampling and subspace separation) [68, 43, 65, 2, 59, 19]. Hence, a natural approach to HRMC is thus to fill missing entries *naively* (with zeros, means, or LRMC) prior to clustering with a full-data method, like sparse subspace clustering [22, 40, 75]. Unfortunately, this approach may work if data is missing at a rate inversely proportional to the dimension of the subspaces [64], but fails with moderate volumes of missing data, as data filled naively no longer lies in a union of subspaces [21].

Tailored HRMC algorithms. Algorithms specifically designed to solve the HRMC problem can be further divided in the following subgroups: (1) *neighborhood* methods that cluster points according to their overlapping coordinates [24], (2) *alternating* methods, like EM [50], k-subspaces [7], group-lasso [52, 54], S^3LR [39], or MCOS [41] (3) *liftings*, which exploit the second-order algebraic structure of unions of subspaces [68, 70, 48, 26, 27], and (4) *integer programming* [57]. Neighborhood methods require either abundant observations or a super-polynomial number of samples (to produce enough overlaps). Liftings require squaring the dimension of an already high-dimensional problem, which severely limits their applicability. Integer programming approaches are similarly restricted to small data.

To summarize, while much research has been devoted to HRMC, current algorithms have shortcomings, and little is known regarding their theoretical guarantees.

Our work in context. Among the methods discussed above, the approach of this paper is perhaps closer in principle to [47], which uses a similar Grassmannian optimization model to study the single-subspace problem of LRMC. This paper generalizes these ideas to the much harder multiple-subspace problem of HRMC, while maintaining the local convergence guarantees of Proposition 5.1 in [47]. The main difference between [47] and our formulation is that the former only considers a predefined subset of geodesic distances (see equations (17)-(19) in [47]), which determine the Grassmannian points that must be matched. In [47], these subsets of geodesics can be chosen somewhat arbitrarily, because in LRMC all points belong to the same subspace. A so-called *gossip* protocol is therefore suitable in the easier problem of LRMC. In contrast, HRMC requires that only certain subsets of the Grassmannian be matched (the points corresponding to the unknown clustering to be learnt). Without knowing a priori the correct clusters, one cannot utilize the *gossip* method and must therefore use *all* pairwise geodesics so as to not introduce bias.

Note: Appendices A-E contain a review on the mathematical background involved in our formulation.

3 Model and Main Results

Let $\mathbf{x}_1, \dots, \mathbf{x}_\mathrm{n} \in \mathbb{R}^\mathrm{m}$ lie near a union of subspaces with dimension upper bounded by r. Let $\mathbf{x}_\mathrm{i}^\Omega \in \mathbb{R}^{|\Omega_\mathrm{i}|}$ denote the observed entries of $\mathbf{x}_\mathrm{i}$, indexed by $\Omega_\mathrm{i} \subset \{1, \dots, \mathrm{m}\}$. We propose assigning to each observed vector $\mathbf{x}_\mathrm{i}^\Omega$ a proxy subspace $\mathcal{U}_\mathrm{i} := \mathrm{span}(\mathbf{U}_\mathrm{i})$. Our goal is to estimate the true subspace $\mathcal{U}_\mathrm{i}^\star$ to which $\mathbf{x}_\mathrm{i}$ belongs by (a) enforcing that the proxy space $\mathcal{U}_\mathrm{i}$ contains a possible completion of $\mathbf{x}_\mathrm{i}^\Omega$ and (b) minimizing the distance between individual proxy spaces $\mathcal{U}_\mathrm{i}$ and $\mathcal{U}_\mathrm{j}$ to build consensus. This is done via the following optimization problem, where the first term achieves goal (a) and the second term achieves goal (b):

$$\min_{\mathbf{U}_1,\dots,\mathbf{U}_\mathrm{n} \in \mathbb{S}(\mathrm{m},\mathrm{r})} \quad \sum_{\mathrm{i}=1}^{\mathrm{n}} d_c^2(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i}) \; + \; \frac{\lambda}{2} \sum_{\mathrm{i},\mathrm{j}=1}^{\mathrm{n}} d_g^2(\mathbf{U}_\mathrm{i}, \mathbf{U}_\mathrm{j}), \tag{1}$$

where

$$d_c(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i}) := \sqrt{1 - \sigma_1^2(\mathbf{X}_\mathrm{i}^{0\mathsf{T}} \mathbf{U}_\mathrm{i})} \qquad \text{and} \qquad d_g(\mathbf{U}_\mathrm{i}, \mathbf{U}_\mathrm{j}) := \sqrt{\sum_{\ell=1}^{\mathrm{r}} \arccos^2 \sigma_\ell(\mathbf{U}_\mathrm{i}^\mathsf{T} \mathbf{U}_\mathrm{j})}.$$

Here $\mathbb{S}(\mathrm{m}, \mathrm{r})$ denotes the Stiefel manifold of $\mathrm{m} \times \mathrm{r}$ orthonormal matrices, $\lambda \geq 0$ is a regularization parameter, $\sigma_\ell(\cdot)$ denotes the ℓ^{th} largest singular value, and $\mathbf{X}_\mathrm{i}^0$ is the orthonormal matrix spanning all the possible completions of a non-zero $\mathbf{x}_\mathrm{i}^\Omega$. The space of all possible completions of $\mathbf{x}_\mathrm{i}^\Omega$ is therefore $\mathcal{X}_\mathrm{i}^0 := \mathrm{span}(\mathbf{X}_\mathrm{i}^0)$, which clearly contains the true data $\mathbf{x}_\mathrm{i}$. The matrix $\mathbf{X}_\mathrm{i}^0$ can be easily constructed as follows. If $\mathbf{x}_\mathrm{i}^\Omega = \mathbf{0}$, then $\mathbf{X}_\mathrm{i}^0 = \mathbf{I}$, the identity matrix. Otherwise, $\mathbf{X}_\mathrm{i}^0$ is the $\mathrm{m} \times (\mathrm{m} - |\Omega_\mathrm{i}| + 1)$ matrix formed with $\mathbf{x}_\mathrm{i}^\Omega$ normalized and filled with zeros in the unobserved rows, concatenated with the $(\mathrm{m} - |\Omega_\mathrm{i}|)$ canonical vectors indicating

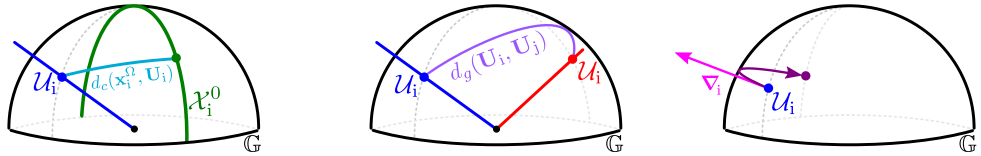

Figure 2: The semi-spheres represent the Grassmannian $\mathbb{G}(\mathrm{m}, \mathrm{r})$, where each point $\mathcal{U}_\mathrm{i}$ represents a subspace (in the particular case of $\mathbb{G}(3, 1)$, the line going from the origin to $\mathcal{U}_\mathrm{i}$). **Left:** Intuitively, the *chordal* distance $d_c(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i})$ is an informal measure of distance between the subspace $\mathcal{U}_\mathrm{i}$ and an incomplete point $\mathbf{x}_\mathrm{i}^\Omega$. The left image should only be taken as intuition since $\mathcal{X}_\mathrm{i}^0$ may not live on the same Grassmannian and the chordal distance should not be thought of as a geodesic distance. **Center:** The *geodesic* distance $d_g(\mathbf{U}_\mathrm{i}, \mathbf{U}_\mathrm{j})$ measures the distance over the Grassmannian between $\mathcal{U}_\mathrm{i}$ and $\mathcal{U}_\mathrm{j}$. **Right:** The Euclidean gradient vector $\boldsymbol{\nabla}_\mathrm{i}$ *falls* out of the Grassmann manifold; to account for the Grassmannian curvature, each geodesic step needs to be adjusted according to (4).

the unobserved rows of $\mathbf{x}_\mathrm{i}^\Omega$. For example, if $\mathbf{x}_\mathrm{i}^\Omega \neq \mathbf{0}$ is observed in the first $|\Omega_\mathrm{i}|$ rows, then

$$\mathbf{X}_\mathrm{i}^0 \;=\; \underbrace{\left[\begin{array}{c|c} \frac{\mathbf{x}_\mathrm{i}^\Omega}{\|\mathbf{x}_\mathrm{i}^\Omega\|} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I} \end{array}\right]}_{\mathrm{m}-|\Omega_\mathrm{i}|+1} \begin{array}{l} \}\, |\Omega_\mathrm{i}| \\ \Big\}\, \mathrm{m} - |\Omega_\mathrm{i}|. \end{array}$$

When $\mathbf{x}_\mathrm{i}^\Omega$ is fully observed, $\mathbf{X}_\mathrm{i}^0$ simplifies to $\mathbf{x}_\mathrm{i}$ normalized. Recall that Grassmannians $\mathbb{G}(\mathrm{m}, \mathrm{r})$ are quotient spaces of Stiefel manifolds $\mathbb{S}(\mathrm{m}, \mathrm{r})$ by action of the orthogonal group of r × r orthonormal matrices. Since both terms $d_c(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i})$ and $d_g(\mathbf{U}_\mathrm{i}, \mathbf{U}_\mathrm{j})$ are invariant under this quotient, the objective function in (1) does not depend on the choice of basis, and descends to a function on the Grassmannian.

Why should this work? The *chordal distance* $d_c(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i})$, as defined in [72], is not a formal distance on the Grassmannian, but rather measures how far $\mathcal{U}_\mathrm{i}$ is from containing a possible completion of $\mathbf{x}_\mathrm{i}^\Omega$. More precisely, $d_c(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i})$ is the cosine of the angle between the nearest completion of $\mathbf{x}_\mathrm{i}^\Omega$ and the r-plane $\mathcal{U}_\mathrm{i}$. If the top singular value $\sigma_1(\mathbf{X}_\mathrm{i}^{0\mathsf{T}}\mathbf{U}_\mathrm{i})$ is 1, then $\mathcal{X}_\mathrm{i}^0$ and $\mathcal{U}_\mathrm{i}$ intersect on at least a line, meaning that the proxy space $\mathcal{U}_\mathrm{i}$ contains a possible completion of $\mathbf{x}_\mathrm{i}^\Omega$. While merely forcing $\mathcal{U}_\mathrm{i}$ to contain a possible completion offers no way to distinguish one possible completion to another, consensus among data is built as different proxies $\mathcal{U}_\mathrm{i}$ and $\mathcal{U}_\mathrm{j}$ are forced towards one another by the geodesic term. In other words, $\mathcal{U}_\mathrm{i}$ and $\mathcal{U}_\mathrm{j}$ are allowed to be near each other, and hence form clusters, only if they both contain possible completions of both $\mathbf{x}_\mathrm{i}^\Omega$ and $\mathbf{x}_\mathrm{j}^\Omega$. The term *chordal distance* used in this way is adopted from [72] and should not be confused with the more common chordal distance between points on the Grassmannian [18]. See Figure 2 to build some intuition.

Solving (1) The gradients of (1) with respect to $\mathbf{U}_\mathrm{i}$ over the Grassmannian are given by:

$$\boldsymbol{\nabla} d_c^2(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i}) \;=\; -2\sigma_1(\mathbf{X}_\mathrm{i}^{0\mathsf{T}}\mathbf{U}_\mathrm{i})(\mathbf{I} - \mathbf{U}_\mathrm{i}\mathbf{U}_\mathrm{i}^\mathsf{T})\mathbf{v}_\mathrm{i}\mathbf{w}_\mathrm{i}^\mathsf{T}, \tag{2}$$

$$\boldsymbol{\nabla} d_g^2(\mathbf{U}_\mathrm{i}, \mathbf{U}_\mathrm{j}) \;=\; -\sum_{\ell=1}^{r} \frac{2\arccos\sigma_\ell(\mathbf{U}_i^\mathsf{T}\mathbf{U}_\mathrm{j})}{\sqrt{1-\sigma_\ell^2(\mathbf{U}_\mathrm{i}^\mathsf{T}\mathbf{U}_\mathrm{j})}} \cdot (\mathbf{I} - \mathbf{U}_\mathrm{i}\mathbf{U}_\mathrm{i}^\mathsf{T})\mathbf{v}_\mathrm{ij}^\ell\mathbf{w}_\mathrm{ij}^{\ell\mathsf{T}}, \tag{3}$$

where $\mathbf{v}_\mathrm{i}$ and $\mathbf{w}_\mathrm{i}$ are the leading left and right singular vectors of $\mathbf{X}_\mathrm{i}^0\mathbf{X}_\mathrm{i}^{0\mathsf{T}}\mathbf{U}_\mathrm{i}$, and $\mathbf{v}_\mathrm{ij}^\ell$, $\mathbf{w}_\mathrm{ij}^\ell$ are the ℓ^{th} left and right singular vectors of $\mathbf{U}_\mathrm{j}\mathbf{U}_\mathrm{j}^\mathsf{T}\mathbf{U}_\mathrm{i}$. The key behind these expressions is that tangent vectors on the Grassmannian can be computed as projections of gradient vectors in Euclidean space [1, 20]. In fact, that is exactly what the gradient expressions in (2) and (3) are: $-2\sigma_1(\mathbf{X}_\mathrm{i}^{0\mathsf{T}}\mathbf{U}_\mathrm{i})\mathbf{v}_\mathrm{i}\mathbf{w}_\mathrm{i}^\mathsf{T}$ in (2) is the gradient of $d_c^2(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i})$ with respect to the entries of $\mathbf{U}_\mathrm{i}$ in Euclidean space. The multiplication by $\mathbf{I} - \mathbf{U}_\mathrm{i}\mathbf{U}_\mathrm{i}^\mathsf{T}$ takes the horizontal direction of the tangent vector with respect to the quotient, thus mapping the gradient from Euclidean space to the Grassmannian. The same is true for the term $\mathbf{I} - \mathbf{U}_{\mathrm{i}}\mathbf{U}_{\mathrm{i}}^{\mathsf{T}}$ in (3). To summarize, (2) and (3) are gradient directions in the manifold of subspaces, rather than matrices: (2) is the steepest direction along which the subspace $\mathcal{U}_{\mathrm{i}}$ can be descended to a potential subspace that contains $\mathbf{x}_{\mathrm{i}}^{\Omega}$, and (3) is the steepest direction along which the subspace $\mathcal{U}_{\mathrm{i}}$ can be descended to the subspace $\mathcal{U}_{\mathrm{j}}$. The derivations of these gradients are in Appendix E. Putting together the chordal and geodesic gradients, our overall descent direction for $\mathbf{U}_{\mathrm{i}}$ is given by

$$\boldsymbol{\nabla}_{\mathrm{i}} \;:=\; \boldsymbol{\nabla} d_c^2(\mathbf{x}_{\mathrm{i}}^{\Omega}, \mathbf{U}_{\mathrm{i}}) + \frac{\lambda}{2}\sum_{\mathrm{j}=1}^{\mathrm{n}} \boldsymbol{\nabla} d_g^2(\mathbf{U}_{\mathrm{i}}, \mathbf{U}_{\mathrm{j}}).$$

Observe that $\mathbf{U}_{\mathrm{i}} - \eta\boldsymbol{\nabla}_{\mathrm{i}}$ *falls* out of the Grassmannian for every step size $\eta \neq 0$ (see Figure 2). To adjust for the curvature of the manifold, the update after taking a geodesic step of size η over the Grassmannian in the direction of $-\boldsymbol{\nabla}_{\mathrm{i}}$ is given by equation (2.65) in [20], which in our context reduces to:

$$\mathbf{U}_{\mathrm{i}} \;\leftarrow\; \begin{bmatrix}\mathbf{U}_{\mathrm{i}}\mathbf{E}_{\mathrm{i}} & \boldsymbol{\Gamma}_{\mathrm{i}}\end{bmatrix} \begin{bmatrix}\operatorname{diag}\cos(\eta\boldsymbol{\Upsilon}_{\mathrm{i}}) \\ \operatorname{diag}\sin(\eta\boldsymbol{\Upsilon}_{\mathrm{i}})\end{bmatrix} \mathbf{E}_{\mathrm{i}}^{\mathsf{T}}, \tag{4}$$

where $\boldsymbol{\Gamma}_{\mathrm{i}}\boldsymbol{\Upsilon}_{\mathrm{i}}\mathbf{E}_{\mathrm{i}}^{\mathsf{T}}$ is the compact singular value decomposition of $-\boldsymbol{\nabla}_{\mathrm{i}}$. In our implementation, we use Armijo step sizes, given by $\eta \;=\; \beta^{\nu}\eta_0$, where $\eta_0 > 0$, and $\beta, \gamma \in (0, 1)$ are the Armijo tuning parameters related to the initial step size and step search granularity [1], and ν is the smallest non-negative integer such that

$$\sum_{\mathrm{i}=1}^{\mathrm{n}} f_{\mathrm{i}}(\mathbf{U}_{\mathrm{i}}) - f_{\mathrm{i}}(R_{\mathbf{U}_{\mathrm{i}}}(\beta^{\nu}\eta_0\boldsymbol{\nabla}_{\mathrm{i}})) \;\geq\; -\gamma\sum_{\mathrm{i}=1}^{\mathrm{n}}\langle\boldsymbol{\nabla}_{\mathrm{i}}, \beta^{\nu}\eta_0(-\boldsymbol{\nabla}_{\mathrm{i}})\rangle,$$

where $f_{\mathrm{i}}(\mathbf{U}_{\mathrm{i}})$ is the component of the objective function (1) holding i fixed and ranging over j, and $R_{\mathbf{U}_{\mathrm{i}}}(\boldsymbol{\Delta})$ performs the geodesic step described by (4) in the direction of $\boldsymbol{\Delta}$.

Convergence guarantees. One advantage of our approach is that we can use standard techniques as in [47] to obtain local convergence guarantees like the following:

Proposition 1. Let $\{(\mathbf{U}_1, \mathbf{U}_2, \ldots, \mathbf{U}_{\mathrm{n}})\}$ be a sequence of iterates generated by the geodesic steps given by equation (4) with Armijo steps sizes η as defined above. Then the sequence will converge to a critical point of (1).

Proof. It suffices to show the (rather technical) fact that the gradient steps in (4) are an instance of Accelerated Line Search (ALS) given in [1] and outlined in Algorithm 1, where the product manifold $\mathbb{G}^n$ will serve as the Riemmannian manifold $\mathcal{M}$, the tangent space of which is the cartesian product of tangent spaces of each constituent $\mathbb{G}$. To see this, let $T\mathcal{M}$ denote the *tangent bundle* of (set of all tangent vectors to) $\mathcal{M}$, and let $T_{\mathcal{U}}\mathcal{M}$ denote the tangent space of $\mathcal{M}$ at $\mathcal{U} \in \mathcal{M}$. In our case, $\mathcal{U}$ is the tuple $(\mathbf{U}_1, \ldots, \mathbf{U}_{\mathrm{n}})$, and equation (4) serves as the *retraction*[1] $R_{\mathbf{U}_{\mathrm{i}}}$ on each component, so that $R_{\mathcal{U}} = (R_{\mathbf{U}_1}, \ldots, R_{\mathbf{U}_n})$. One can verify that this is indeed a retraction by recognizing (4) as the exponential map $\operatorname{Exp} : T_{\mathbf{U}}\mathbb{G} \to \mathbb{G}$ and noting that, on a Riemannian manifold, the exponential map is a retraction, and that the product of exponential maps is again an exponential map [1]. For our sequence of *gradient-related*[2] tangent vectors, we use the negative gradient, which is clearly always gradient-related. The

[1] A mapping R from $T\mathcal{M}$ to $\mathcal{M}$ such that its restriction to $T_{\mathcal{U}}\mathcal{M}$, denoted $R_{\mathcal{U}}$, satisfies a local rigidity condition which preserves gradients at $\mathcal{U}$; see the rightmost illustration in Figure 2 to build some intuition, or Chapters 3 and 4 of [1] for a more careful treatment of these definitions.

[2] Given a cost function f on a Riemannian manifold $\mathcal{M}$, a sequence of tangent vectors $\{\boldsymbol{\Delta}_{\mathrm{t}}\}$, $\boldsymbol{\Delta}_{\mathrm{t}} \in T_{\mathcal{U}_{\mathrm{t}}}\mathcal{M}$, is gradient-related if, for any sequence $\{\mathcal{U}_{\mathrm{t}}\}_{\mathrm{t}\in\mathcal{K}}$ that converges to a non-critical point of f, the corresponding subsequence $\{\boldsymbol{\Delta}_{\mathrm{t}}\}_{\mathrm{t}\in\mathcal{K}}$ is bounded and satisfies $\limsup_{\mathrm{t}\to\infty,\ \mathrm{t}\in\mathcal{K}}\langle\nabla f(\mathcal{U}_{\mathrm{t}}), \boldsymbol{\Delta}_{\mathrm{t}}\rangle < 0$.

Algorithm 1: Accelerated Line Search (ALS)

```
Require: Riemannian manifold $\mathcal{M}$; continuously differentiable scalar field $f$ on $\mathcal{M}$;
  retraction $R$ from $T\mathcal{M}$ to $\mathcal{M}$; scalars $\eta_0 > 0$, $c, \beta, \gamma \in (0, 1)$.
Input: Initial iterate $\mathcal{U} \in \mathcal{M}$
Output: Sequence of iterates $\{\mathcal{U}_t\}$.
for $t = 0, 1, 2, \ldots$ do
    Pick $\boldsymbol{\Delta}_t \in T_{\mathcal{U}_t}\mathcal{M}$ such that the sequence of tangent vectors $\{\boldsymbol{\Delta}_t\}$ is gradient related.
    Select $\mathcal{U}_{t+1}$ such that
        $f(\mathcal{U}_t) - f(\mathcal{U}_{t+1}) \geq c(f(\mathcal{U}_t) - f(R_{\mathcal{U}_t}(\eta_t \boldsymbol{\Delta}_t))),$   (5)
     where $\eta_t$ is the Armijo step size for the given $\eta_0, \beta, \gamma, \boldsymbol{\Delta}_t$.
end
```

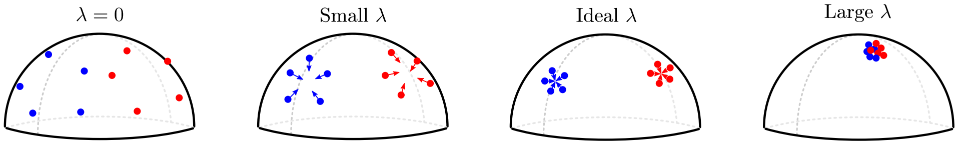

Figure 3: $\lambda \geq 0$ in (1) regulates how clusters fuse together. If $\lambda = 0$, each point is assigned to a subspace that exactly contains it (overfitting). The larger λ, the more we penalize subspaces being apart, which results in subspaces getting closer to form fewer clusters. The extreme case $\lambda = \infty$ is the special case of PCA and LRMC, where only one subspace is allowed to explain all data.

gradient on the product manifold is the cartesian product of the gradients on each constituent manifold, i.e., $\boldsymbol{\nabla}(f) = (\boldsymbol{\nabla}(f_1), \ldots, \boldsymbol{\nabla}(f_n))$. Moreover, the inner product on the tangent space is the sum of the inner products on the constituent tangent spaces. Therefore, if $\{\boldsymbol{\Delta}_{i,t}\}$, $\boldsymbol{\Delta}_{i,t} \in T_{\mathcal{U}_t}\mathcal{M}_i$ is gradient-related for each $\mathcal{M}_i$, then $\{(\boldsymbol{\Delta}_{1,t}, \ldots, \boldsymbol{\Delta}_{n,t})\}$ is gradient-related on the product manifold. Furthermore, setting $\mathcal{U}_{t+1} = R_{\mathcal{U}_t}(\eta_t \boldsymbol{\Delta}_t)$ satisfies the bound in (5) with $c = 1$. Thus Proposition 1 follows as consequence of Theorem 4.3.1 and Corollary 4.3.2 in [1]. □

4 Fusion in Practice

Clustering, completion, and subspace inference. Recall that the solution to (1) provides an estimator $\mathcal{U}_i$ of $\mathcal{U}_i^\star$, the true subspace from which $\mathbf{x}_i$ is drawn. After solving (1), one may form the matrix $\mathbf{D}$ whose $(i, j)^{th}$ entry is given by $d_g(\mathbf{U}_i, \mathbf{U}_j)$ and use it as input to any distance-based clustering method, such as k-means [9, 23, 56], spectral clustering [61, 67, 71, 73], or DBSCAN [25, 32, 55]. While prior knowledge of the number of subspaces K may be required for some clustering methods (e.g., k-means, or spectral clustering), it is not required at all to solve (1). Hence, by choosing a clustering method that doesn't require knowing K (e.g., DBSCAN), our approach can be applied to situations where K is unknown. After clustering, one can agglomerate all the data points corresponding to the kth cluster in the same matrix $\hat{\mathbf{X}}_k^\Omega$, and run any low-rank matrix completion (LRMC) algorithm (e.g., [6, 8, 10, 11, 12, 36, 53, 69]) to estimate its completion $\hat{\mathbf{X}}_k$. Finally, one can run principal component analysis (PCA) [46, 74] on $\hat{\mathbf{X}}_k$ to recover an estimate basis $\hat{\mathbf{U}}_k$ of the kth underlying subspace $\mathcal{U}_k^\star$.

Penalty parameter and model selection. Intuitively, the chordal term in (1) forces each subspace to be close to its assigned data point, and the geodesic term forces subspaces from different data points to be close to one another. The tradeoff between these two quantities is determined by the penalty parameter $\lambda \geq 0$. If $\lambda = 0$, then the geodesic term is ignored and there is a trivial solution where each subspace exactly contains its assigned data point (thus attaining the minimum, zero, for the chordal distance). If $\lambda > 0$, the geodesic term forces subspaces from different data points to get closer, even if they no longer contain exactly their assigned data points. As λ grows, subspaces get closer and closer (see Figure 3). The extreme case ($\lambda = \infty$) forces all subspaces to fuse into one (to attain zero in the second term), allowing only one subspace to explain all data, which is the equivalent of PCA in the complete-data case, and LRMC if data is missing. In other words, PCA and LRMC are the special cases of our formulation with $\lambda = \infty$.

Ultimately, the effect of λ will be reflected in the distance matrix $\mathbf{D}$, which in turn determines the number of clusters. The smaller λ, the more clusters, up to the extreme where each data point is in its own cluster. Conversely, the larger λ, the fewer clusters, up to the extreme point where all points are clustered together. The more subspaces, the more accuracy, but the more degrees of freedom (overfitting). To determine the best λ, one can compute a goodness of fit test, like the minimum effective dimension [33, 68], that quantifies the tradeoff between accuracy and degrees of freedom. Similarly, we can iteratively increase r in (1) to find all the data points that lie in 1-dimensional subspaces, then all the data points that lie in 2-dimensional subspaces, and so on (pruning the data at each iteration). This will result in an estimate of the number of subspaces K, and their dimensions.

Initialization. In our implementation we initialize (1) with a solution to the problem when $\lambda = 0$, i.e., when each subspace perfectly contains the observed entries of its assigned data point. To this end, for each i we first construct an $\mathrm{m} \times \mathrm{r}$ matrix whose first column is equal to $\mathbf{x}_\mathrm{i}^\Omega$ in its observed entries, and whose remaining entries are filled with standard normal entries, known to produce incoherent and uniformly distributed subspaces [24]. This matrix is then orthonormalized to produce the initial estimate $\mathbf{U}_\mathrm{i}$, which, by construction, contains $\mathbf{x}_\mathrm{i}^\Omega$, thus producing $d_c(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i}) = 0$.

Computational complexity. We point out that the main caveat of our approach is its quadratic complexity in the number of samples. Fortunately, subspace clustering allows a simple approach to sketching both samples and features [62]. That is, one may solve (1) with a subset of $\mathrm{n}' \leq \mathrm{n}$ columns, and a subset of $\mathrm{m}' \leq \mathrm{m}$ rows (e.g., those with most observations), resulting in an improved complexity, quadratic in n' as opposed to n. With the solution of (1), one can use a clustering method, a LRMC algorithm, and PCA, as described above, to produce subspace estimates $\hat{\mathbf{U}}_1, \dots, \hat{\mathbf{U}}_{\mathrm{K}'}$, with $\mathrm{K}' \leq \mathrm{K}$. Each of the remaining $\mathrm{n} - \mathrm{n}'$ incomplete data points $\mathbf{x}_\mathrm{i}^\Omega$ that were not used to solve (1) and that have more than r observations (a fundamental requirement of subspace clustering [51]) can be trivially assigned to the subspace estimate producing the largest projection coefficient $\boldsymbol{\theta}_\mathrm{i}^\mathrm{k} = (\hat{\mathbf{U}}_\mathrm{k}^{\Omega\mathsf{T}} \hat{\mathbf{U}}_\mathrm{k}^\Omega)^{-1} \mathbf{x}_\mathrm{i}^\Omega$, where $\hat{\mathbf{U}}_\mathrm{k}^\Omega \in \mathbb{R}^{|\Omega_\mathrm{i}| \times \mathrm{r}}$ denotes the restriction of $\hat{\mathbf{U}}_\mathrm{k}$ to the observed rows of $\mathbf{x}_\mathrm{i}^\Omega$ (notice that $\hat{\mathbf{U}}_\mathrm{k}^{\Omega\mathsf{T}} \hat{\mathbf{U}}_\mathrm{k}^\Omega$ is invertible for almost every rank-r $\hat{\mathbf{U}}_\mathrm{k}$ whenever $|\Omega_\mathrm{i}| > \mathrm{r}$ [51]). If $\mathbf{x}_\mathrm{i}^\Omega$ is assigned to $\hat{\mathbf{U}}_\mathrm{k}$, its completion can be trivially estimated as $\hat{\mathbf{x}}_\mathrm{i} = \hat{\mathbf{U}}_\mathrm{k} \boldsymbol{\theta}_\mathrm{i}^\mathrm{k}$. All the data points $\mathbf{x}_\mathrm{i}^\Omega$ that are too far from all of the subspace estimates (equivalently, the data points whose coefficients are smaller than a pre-determined parameter) can be used to solve (1) again for a refined clustering.

5 Experiments

In this section we present a series of experiments on real and synthetic data, in particular the Hopkins155 dataset [63], and the Smartphone dataset for Human Activity Recognition in Ambient Assisted Living (AAL) [4]. Rather than establishing a new state-of-the-art, these experiments have the intention to serve as proof of concept, showing the potential of our approach, which in this first introduction and basic formulation performs comparable to prominent methods [75]. In our experiments we initialize (1) as described in Section 4, with r fixed, as is known a priori in both, the simulations, and the real datasets. We do not specify K, and we make no special adjustments to handle noise, as is not required by our approach. The attained solution to (1) is used as input to spectral clustering [61, 67, 71, 73] (though, as described in Section 4, other clustering algorithms, such as k-means [9, 23, 56] or DBSCAN [25, 32, 55] could be used). We measure accuracy in terms of clustering error, given by $\min_M \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{\{M(\hat{\mathbf{y}}) \neq \mathbf{y}\}}$, where $\mathbb{1}$ denotes the indicator function, and M is a function that maps the estimated cluster labels $\hat{\mathbf{y}} \in \{1, \dots, \hat{\mathrm{K}}\}^\mathrm{n}$ assigned to $\mathbf{x}_1^\Omega, \dots, \mathbf{x}_\mathrm{n}^\Omega$, to the true labels $\mathbf{y} \in \{1, \dots, \mathrm{K}\}^\mathrm{n}$.

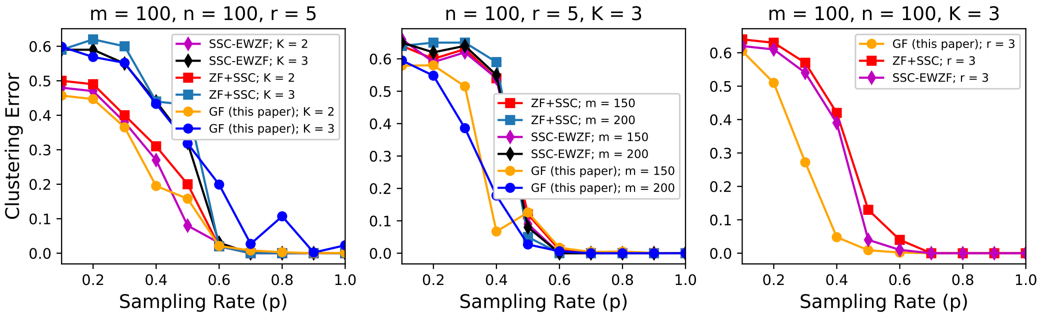

Figure 4: Clustering error (average over 10 trials) as a function of sampling rate for different synthetic settings.

Baseline comparisons and full-data sub-optimality. A recent survey [38] shows that most state-of-the-art algorithms for HRMC (including MC+SSC [75], EM [50], GSSC [52], k-subspaces [7], and more) have similar performance, with varying winners on specific scenarios depending on subspace vs ambient dimension gap, fraction of missing data, and number of subspaces. Based on this recent survey [38], and others [52], we chose ZF+SSC and SSC-EWZF as baselines, which has been seen in [38] to have nearly identical performance as MC+SSC, EM, GSSC, k-subspaces and k-GROUSE [7] in the scenarios discussed in our paper.

Synthetic data. In all our simulations we first generate K matrices $\mathbf{U}_{\mathrm{k}}^{\star} \in \mathbb{R}^{\mathrm{m}\times\mathrm{r}}$ with i.i.d. $N(0,1)$ entries, to use as bases of the *true* subspaces. For each k we generate a matrix $\mathbf{\Theta}_{\mathrm{k}}^{\star} \in \mathbb{R}^{\mathrm{r}\times\mathrm{n_k}}$, also with i.i.d. $N(0,1)$ entries, to use as coefficients of the columns in the k^{th} subspace. We then form $\mathbf{X}$ as the concatenation $[\mathbf{U}_1^{\star}\mathbf{\Theta}_1^{\star},\ \mathbf{U}_2^{\star}\mathbf{\Theta}_2^{\star},\ \dots,\ \mathbf{U}_{\mathrm{K}}^{\star}\mathbf{\Theta}_{\mathrm{K}}^{\star}]$. To induce missing data we sample each entry independently with probability p. Figure 4 shows the clustering results as a function of the sampling rate for a variety of settings, tuning the parameter λ manually. Notice that, even with this first formulation, we perform comparable to existing methods, and even better in some cases, especially in low-sampling regimes.

Object tracking in Hopkins 155. This dataset contains 155 videos of $\mathrm{K} = 2$ or $\mathrm{K} = 3$ moving objects, such as checkerboards, vehicles, and pedestrians. In each video, a collection of n mark points are tracked through all frames. The locations over time of the i^{th} point are stacked to produce $\mathbf{x}_{\mathrm{i}} \in \mathbb{R}^{\mathrm{m}}$, so that the points corresponding to the same object lie near a low-dimensional subspace [60, 35] (r varies from video to video, from 1 to 3). In all cases we fixed the penalty parameter λ to 1. To induce missing data (e.g., produced by occlusions) we sample each entry independently with probability p. Figure 5 shows the clustering results.

Human activity recognition in Smartphone AAL Dataset. This dataset contains $\mathrm{n} = 5744$ instances, each with $\mathrm{m} = 561$ features related to pre-processed accelerometer and gyroscope time series and summary statistics [3], related to $\mathrm{K} = 2$ activities: walking, and other movements, each approximated by a subspace of dimension $\mathrm{r} = 4$. Recall that the complexity of (1) is quadratic in n, so if solved directly, this dataset that would produce an unmanageable computational complexity. However, using the sketching techniques described in Section 4, it can be solved quite efficiently. In particular, we only used $\mathrm{m}' = 158$ features (related to the accelerometer's and gyroscope's minimum, maximum, standard deviation, and mean parameters over time), and $\mathrm{n}' = 100$ samples selected at random, evenly distributed among classes. In all cases we fixed the penalty parameter λ to 10^{-5}. The results are summarized in Figure 5. Notice that our approach outperforms existing methods in the low-sampling regime.

Lastly, we note the disparity in performance between our model and existing algorithms when the missing data rate is low. The main motivation for our approach is incomplete data. While our formulation can certainly be used with full data, we acknowledge that it would be an over-kill, and consequently suboptimal in that scenario, which has been extensively studied. Hence, it is not surprising that methods tailored for full-data outperform ours in such setting. However, no full-data method outperforms ours when data is missing.

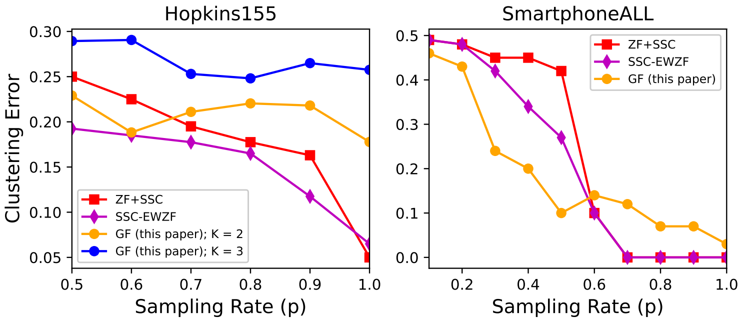

Figure 5: Clustering error as a function of sampling rate for real datasets. **Left:** average over 120 videos with K = 2 objects, and 35 videos with K = 3 objects. **Right:** average over 20 trials.

6 Future Directions and Challenges

The main formulation presented in this paper is a non-convex optimization that relies on the simultaneous interactions of many terms. A proper analysis of the model is therefore challenging. One difficulty confronted is the complex geometry of the zero-set of the chordal distance term, or more precisely, the intersections of many of these zero-sets, one for each column of data. While intuition suggests that the model is encouraged by the geodesic term to find regions of "dense" intersection, and therefore build consensus, a more precise formulation of this intuition has evaded us. This is further highlighted in Figures 4 and 5 by the fact that the performance of our model decreases as K grows, indicating that it is not currently understood how the combination of the chordal and geodesic terms encourage consensus amongst so many cross-cluster terms. It is our belief that a well-designed weighted version of (1), such as

$$\min_{\mathbf{U}_1,\ldots,\mathbf{U}_\mathrm{n}\in\mathbb{S}^{\mathrm{m}\times\mathrm{r}}} \quad \sum_{\mathrm{i}=1}^{\mathrm{n}} d_c^2(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i}) \ + \ \frac{\lambda}{2}\sum_{\mathrm{i,j}=1}^{\mathrm{n}} w_{\mathrm{ij}} d_g^2(\mathbf{U}_\mathrm{i}, \mathbf{U}_\mathrm{j}), \tag{6}$$

where the weights $w_{\mathrm{ij}} \geq 0$ quantify how much attention is given to each penalty, is key to unlocking better performance and understanding of the model. Our immediate future work will focus on investigating options for these weights, such as inverse distance functions, or k-nearest neighbors, known to dramatically improve the performance, computational complexity, and tolerance to K of fusion formulations in Euclidean space [5, 17, 58, 76].

7 Conclusions

This paper presents a new paradigm for clustering incomplete datapoints using subspaces as proxies to leverage the geometry of the Grassmannian. This new perspective enables clustering and completion of data in a union of subspaces. This work should be understood as the first introduction to the idea of fusion penalties in the Grassmann manifold, for the problem of *high-rank matrix completion*. Rather than establishing our approach as the state-of-the-art, our experiments have the intention to serve as proof of concept, showing that there is potential in our approach, in the hopes to ignite future work on several directions, such as the study of weighted versions described in (6), the choice of penalty parameters, and variants robust to outliers.

References

[1] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[2] Pankaj K Agarwal and Nabil H Mustafa. "K-means projective clustering". In: *Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. 2004, pp. 155–165.

[3] Davide Anguita et al. "A public domain dataset for human activity recognition using smartphones." In: *Esann*. Vol. 3. 2013, p. 3.

[4] Davide Anguita et al. *UCI Machine Learning Repository*. 2013. URL: `http://archive.ics.uci.edu/ml`.

[5] Anestis Antoniadis and Jianqing Fan. "Regularization of wavelet approximations". In: *Journal of the American Statistical Association* 96.455 (2001), pp. 939–967.

[6] Laura Balzano, Robert Nowak, and Benjamin Recht. "Online identification and tracking of subspaces from highly incomplete information". In: *2010 48th Annual allerton conference on communication, control, and computing (Allerton)*. IEEE. 2010, pp. 704–711.

[7] Laura Balzano et al. "K-subspaces with missing data". In: *2012 IEEE Statistical Signal Processing Workshop (SSP)*. IEEE. 2012, pp. 612–615.

[8] Jonathan Bauch, Boaz Nadler, and Pini Zilber. "Rank 2r iterative least squares: efficient recovery of ill-conditioned low rank matrices from few entries". In: *SIAM Journal on Mathematics of Data Science* 3.1 (2021), pp. 439–465.

[9] Leon Bottou and Yoshua Bengio. "Convergence properties of the k-means algorithms". In: *Advances in neural information processing systems*. 1995, pp. 585–592.

[10] Emmanuel J. Candes and Terence Tao. "The Power of Convex Relaxation: Near-Optimal Matrix Completion". In: *IEEE Transactions on Information Theory* 56.5 (2010), pp. 2053–2080. DOI: `10.1109/TIT.2010.2044061`.

[11] Emmanuel J Candès and Benjamin Recht. "Exact matrix completion via convex optimization". In: *Foundations of Computational mathematics* 9.6 (2009), pp. 717–772.

[12] Emmanuel J Candès et al. "Robust principal component analysis?" In: *Journal of the ACM (JACM)* 58.3 (2011), pp. 1–37.

[13] Jiuwen Cao et al. "Extreme learning machine and adaptive sparse representation for image classification". In: *Neural networks* 81 (2016), pp. 91–102.

[14] Guangliang Chen and Gilad Lerman. "Spectral curvature clustering (SCC)". In: *International Journal of Computer Vision* 81.3 (2009), pp. 317–330.

[15] Y. Chen. "Incoherence-optimal matrix completion". In: *IEEE Transactions on Information Theory* 61.5 (2013), pp. 2909–2923.

[16] Y. Chen et al. "Coherent matrix completion". In: *International Conference on Machine Learning* 61.5 (2014), pp. 674–682.

[17] Eric C Chi and Kenneth Lange. "Splitting methods for convex clustering". In: *Journal of Computational and Graphical Statistics* 24.4 (2015), pp. 994–1013.

[18] John H. Conway, Ronald H. Hardin, and Neil J. A. Sloane. "Packing lines, planes, etc., packing in Grassmannian spaces". In: *Exper. Math.* 5.2 (1996), pp. 139–159.

[19] Harm Derksen et al. "Segmentation of multivariate mixed data via lossy coding and compression". In: *Visual Communications and Image Processing 2007*. Vol. 6508. SPIE. 2007, pp. 170–181.

[20] Alan Edelman, Tomás A Arias, and Steven T Smith. "The geometry of algorithms with orthogonality constraints". In: *SIAM journal on Matrix Analysis and Applications* 20.2 (1998), pp. 303–353.

[21] Ehsan Elhamifar. "High-rank matrix completion and clustering under self-expressive models". In: *Advances in Neural Information Processing Systems* 29 (2016), pp. 73–81.

[22] Ehsan Elhamifar and René Vidal. "Sparse subspace clustering: Algorithm, theory, and applications". In: *IEEE transactions on pattern analysis and machine intelligence* 35.11 (2013), pp. 2765–2781.

[23] Charles Elkan. “Using the triangle inequality to accelerate k-means”. In: *Proceedings of the 20th international conference on Machine Learning (ICML-03)*. 2003, pp. 147–153.

[24] Brian Eriksson, Laura Balzano, and Robert Nowak. “High-rank matrix completion”. In: *Artificial Intelligence and Statistics*. PMLR. 2012, pp. 373–381.

[25] Martin Ester et al. “A density-based algorithm for discovering clusters in large spatial databases with noise.” In: *Kdd*. Vol. 96. 34. 1996, pp. 226–231.

[26] Jicong Fan and Tommy WS Chow. “Non-linear matrix completion”. In: *Pattern Recognition* 77 (2018), pp. 378–394.

[27] Jicong Fan and Madeleine Udell. “Online high rank matrix completion”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 8690–8698.

[28] Sonny Gan. “The Application of Spectral Clustering in Drug Discovery”. PhD thesis. University of Sheffield, 2013.

[29] D. Gross. “Recovering low-rank matrices from few coefficients in any basis”. In: *IEEE Transactions on Information Theory* 57.3 (2011), pp. 1548–1566.

[30] Jeffrey Ho et al. “Clustering appearances of objects under varying illumination conditions”. In: *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.* Vol. 1. IEEE. 2003, pp. I–I.

[31] Wei Hong et al. “Multiscale hybrid linear models for lossy image representation”. In: *IEEE Transactions on Image Processing* 15.12 (2006), pp. 3655–3671.

[32] Jian Hou, Huijun Gao, and Xuelong Li. “DSets-DBSCAN: A Parameter-Free Clustering Algorithm”. In: *IEEE Transactions on Image Processing* 25.7 (2016), pp. 3182–3193. DOI: 10.1109/TIP.2016.2559803.

[33] Kun Huang, Yi Ma, and René Vidal. “Minimum effective dimension for mixtures of subspaces: A robust GPCA algorithm and its applications”. In: *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* Vol. 2. IEEE. 2004, pp. II–II.

[34] Lee M. John. *Introduction to Riemannian Manifolds, 2nd.* Springer International Publishing, 2018.

[35] Ken-ichi Kanatani. “Motion segmentation by subspace separation and model selection”. In: *Proceedings Eighth IEEE International Conference on computer Vision. ICCV 2001.* Vol. 2. IEEE. 2001, pp. 586–591.

[36] Raghunandan H. Keshavan, Andrea Montanari, and Sewoong Oh. “Matrix Completion From a Few Entries”. In: *IEEE Transactions on Information Theory* 56.6 (2010), pp. 2980–2998. DOI: 10.1109/TIT.2010.2046205.

[37] Hamidreza Koohi and Kourosh Kiani. “A new method to find neighbor users that improves the performance of collaborative filtering”. In: *Expert Systems with Applications* 83 (2017), pp. 30–39.

[38] Connor Lane et al. “Classifying and comparing approaches to subspace clustering with missing data”. In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE. 2019.

[39] Chun-Guang Li and René Vidal. “A structured sparse plus structured low-rank framework for subspace clustering and completion”. In: *IEEE Transactions on Signal Processing* 64.24 (2016), pp. 6557–6570.

[40] Chun-Guang Li, Chong You, and René Vidal. “Structured sparse subspace clustering: A joint affinity learning and subspace clustering framework”. In: *IEEE Transactions on Image Processing* 26.6 (2017), pp. 2988–3001.

[41] Ziheng Li et al. “Matrix completion with column outliers and sparse noise”. In: *Information Sciences* 573 (2021), pp. 125–140.

[42] Le Lu and René Vidal. “Combined central and subspace clustering for computer vision applications”. In: (2006), pp. 593–600.

[43] Yi Ma et al. “Estimation of subspace arrangements with applications in modeling and segmenting mixed data”. In: *SIAM review* 50.3 (2008), pp. 413–458.

[44] Mohamed G Malhat, Hamdy M Mousa, and Ashraf B El-Sisi. “Clustering of chemical data sets for drug discovery”. In: *2014 9th international conference on informatics and systems*. IEEE. 2014, DEKM–11.

[45] Korosh Mashayekh et al. "Clustering and Sampling of the c-Met Conformational Space: A Computational Drug Discovery Study". In: *Combinatorial chemistry & high throughput screening* 22.9 (2019), pp. 635–648.

[46] Thomas Minka. "Automatic choice of dimensionality for PCA". In: *Advances in neural information processing systems* 13 (2000), pp. 598–604.

[47] B. Mishra et al. "A Riemannian gossip approach to subspace learning on Grassmann manifold". In: *Machine Learning* 108.10 (2019), pp. 1783–1803.

[48] Greg Ongie et al. "Algebraic variety models for high-rank matrix completion". In: *International Conference on Machine Learning*. PMLR. 2017, pp. 2691–2700.

[49] Nolen Joy Perualila-Tan et al. "Weighted similarity-based clustering of chemical structures and bioactivity data in early drug discovery". In: *Journal of bioinformatics and computational biology* 14.04 (2016), p. 1650018.

[50] Daniel Pimentel, R Nowak, and Laura Balzano. "On the sample complexity of subspace clustering with missing data". In: *2014 IEEE Workshop on Statistical Signal Processing (SSP)*. IEEE. 2014, pp. 280–283.

[51] Daniel Pimentel-Alarcon and Robert Nowak. "The information-theoretic requirements of subspace clustering with missing data". In: *International Conference on Machine Learning*. PMLR. 2016, pp. 802–810.

[52] Daniel Pimentel-Alarcón et al. "Group-sparse subspace clustering with missing data". In: *2016 IEEE Statistical Signal Processing Workshop (SSP)*. IEEE. 2016, pp. 1–5.

[53] Benjamin Recht. "A Simpler Approach to Matrix Completion." In: *Journal of Machine Learning Research* 12.12 (2011).

[54] Budhaditya Saha et al. "Sparse subspace clustering via group sparse coding". In: *Proceedings of the 2013 SIAM International Conference on Data Mining*. SIAM. 2013, pp. 130–138.

[55] Erich Schubert et al. "DBSCAN revisited, revisited: why and how you should (still) use DBSCAN". In: *ACM Transactions on Database Systems (TODS)* 42.3 (2017), pp. 1–21.

[56] David Sculley. "Web-scale k-means clustering". In: *Proceedings of the 19th international conference on World wide web*. 2010, pp. 1177–1178.

[57] Akhilesh Soni et al. "Integer Programming Approaches To Subspace Clustering With Missing Data". In: ().

[58] Kean Ming Tan and Daniela Witten. "Statistical properties of convex clustering". In: *Electronic journal of statistics* 9.2 (2015), p. 2324.

[59] Michael E Tipping and Christopher M Bishop. "Probabilistic principal component analysis". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 61.3 (1999), pp. 611–622.

[60] Carlo Tomasi and Takeo Kanade. "Shape and motion from image streams under orthography: a factorization method". In: *International journal of computer vision* 9.2 (1992), pp. 137–154.

[61] Tao Tong et al. "One-step spectral clustering based on self-paced learning". In: *Pattern Recognition Letters* 135 (2020), pp. 8–14.

[62] Panagiotis A Traganitis and Georgios B Giannakis. "Sketched subspace clustering". In: *IEEE Transactions on Signal Processing* 66.7 (2017), pp. 1663–1675.

[63] Roberto Tron and René Vidal. "A benchmark for the comparison of 3-d motion segmentation algorithms". In: *2007 IEEE conference on computer vision and pattern recognition*. IEEE. 2007, pp. 1–8.

[64] Manolis Tsakiris and René Vidal. "Theoretical analysis of sparse subspace clustering with missing entries". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 4975–4984.

[65] Paul Tseng. "Nearest q-flat to m points". In: *Journal of Optimization Theory and Applications* 105.1 (2000), pp. 249–252.

[66] Farman Ullah, Ghulam Sarwar, and Sungchang Lee. "N-screen aware multicriteria hybrid recommender system using weight based subspace clustering". In: *The Scientific World Journal* 2014 (2014).

[67] Patrick Veenstra, Colin Cooper, and Steve Phelps. “Spectral clustering using the kNN-MST similarity graph”. In: *2016 8th Computer Science and Electronic Engineering (CEEC)*. IEEE. 2016, pp. 222–227.

[68] Rene Vidal, Yi Ma, and Shankar Sastry. “Generalized principal component analysis (GPCA)”. In: *IEEE transactions on pattern analysis and machine intelligence* 27.12 (2005), pp. 1945–1959.

[69] René Vidal and Paolo Favaro. “Low rank subspace clustering (LRSC)”. In: *Pattern Recognition Letters* 43 (2014), pp. 47–61.

[70] René Vidal, Roberto Tron, and Richard Hartley. “Multiframe motion segmentation with missing data using PowerFactorization and GPCA”. In: *International Journal of Computer Vision* 79.1 (2008), pp. 85–105.

[71] Ulrike Von Luxburg. “A tutorial on spectral clustering”. In: *Statistics and computing* 17.4 (2007), pp. 395–416.

[72] Ely Kerman Wei Dai and Olgica Milenkovic. “A Geometric Approach to Low-Rank Matrix Completion”. In: *IEEE Transaction on Information Theory* 58.1 (2012), pp. 237–247.

[73] Guoqiu Wen. “Robust self-tuning spectral clustering”. In: *Neurocomputing* 391 (2020), pp. 243–248.

[74] Svante Wold, Kim Esbensen, and Paul Geladi. “Principal component analysis”. In: *Chemometrics and intelligent laboratory systems* 2.1-3 (1987), pp. 37–52.

[75] Congyuan Yang, Daniel Robinson, and Rene Vidal. “Sparse subspace clustering with missing entries”. In: *International Conference on Machine Learning*. PMLR. 2015, pp. 2463–2472.

[76] Ming Yuan and Yi Lin. “Model selection and estimation in regression with grouped variables”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68.1 (2006), pp. 49–67.

[77] Wen Zhang et al. “RP-LGMC: rating prediction based on local and global information with matrix clustering”. In: *Computers & Operations Research* 129 (2021), p. 105228.

A Stiefel and Grassmann Manifolds

The primary mathematical object involved in this work is the Grassmannian $\mathbb{G}(m, r)$. This is a smooth compact manifold of dimension $r(m - r)$. A full expository on the Stiefel and Grassmann manifolds is given [20]. Here, we record the most basic necessary ideas needed in order to have a working understanding of the tools used in the above. To describe this, it is necessary to define precursor objects; the orthogonal group $O(m)$ and the Stiefel manifold $\mathbb{S}(m, r)$. The objects of interest are thus:

1. The orthogonal group $O(m)$ consisting of $m \times m$ orthogonal matrices;
2. The Stiefel manifold $\mathbb{S}(m, r)$ consisting of $m \times r$ orthonormal matrices;
3. The Grassmann manifold $\mathbb{G}(m, r)$ obtained by identifying those matrices in $\mathbb{S}(m, r)$ whose columns span the same subspace (a quotient manifold).

In this setting, the Stiefel manifold is defined as a quotient space of the orthogonal group. Here, two orthogonal matrices $\mathbf{U}$ and $\mathbf{V}$ are identified if their first r columns are identical or, equivalently, if $\mathbf{U} = \left(\begin{smallmatrix} \mathbf{I} & 0 \\ 0 & \mathbf{Q} \end{smallmatrix}\right)\mathbf{V}$, where $\mathbf{Q}$ is an orthogonal $(m - r) \times (m - r)$ block. Therefore $\mathbb{S}(m, r) = O(m)/O(m - r)$. Going further, the Grassmannian is defined a quotient space of the Stiefel manifold where two Stiefel elements are identified if their columns span the same r-dimensional subspace. Therefore $\mathbb{G}(m, r) = \mathbb{S}(m, r)/O(r)$.

Given the above, it is clear that we may describe elements of the Stiefel and Grassmann manifolds using concrete representatives that can be stored on a computer. A point on the Stiefel manifold may be stored as an $m \times r$ orthonormal matrix. A point on the Grassmann manifold, however, being a linear subspace, does not have a unique representative and can be stored as an arbitrary $m \times r$ orthonormal matrix so long as it spans the correct subspace.

B Principal Angles and Singular Values

Recall the notion of principal angles between subspaces: let $\mathbf{U} \in \mathbb{S}(m,p)$ and $\mathbf{V} \in \mathbb{S}(m,q)$ be orthonormal bases for two arbitrary subspaces of $\mathbb{R}^m$. Assume, without loss of generality, that $1 \leq p \leq q \leq m$. The principal angles between $\text{span}(\mathbf{U})$ and $\text{span}(\mathbf{V})$ are defined via the following construction. Let $\mathbf{u}_1 \in \text{span}(\mathbf{U})$ and $\mathbf{v}_1 \in \text{span}(\mathbf{V})$ be unit vectors such that $|\mathbf{u}_1^T \mathbf{v}_1|$ is maximal. Inductively, let $\mathbf{u}_k \in \text{span}(\mathbf{U})$ and $\mathbf{v}_k \in \text{span}(\mathbf{V})$ be unit vectors such that $\mathbf{u}_k^T \mathbf{u}_{\mathrm{j}} = 0$ and $\mathbf{v}_k^T \mathbf{v}_{\mathrm{j}} = 0$ for all $1 \leq j < k$ and $|\mathbf{u}_k^T \mathbf{v}_k|$ is maximal. The principal angles are defined as $\alpha_k = \arccos \mathbf{u}_k^T \mathbf{v}_k$ for all $k = 1, 2, \ldots, p$.

This constructive definition is too cumbersome to use in practice. We opt for the following alternative computation via the singular value decomposition. Let $\mathbf{u}_1, \ldots, \mathbf{u}_p$ and $\mathbf{v}_1, \ldots, \mathbf{v}_q$ be the columns of $\mathbf{U}$ and $\mathbf{V}$ respectively. Compute the singular value decomposition $\mathbf{U}^T\mathbf{V} = \bar{\mathbf{U}}\text{diag}[\ldots, \sigma_{\mathrm{i}}, \ldots]\bar{\mathbf{V}}^T$. Set $\mathbf{U}' = \mathbf{U}\bar{\mathbf{U}}$ and $\mathbf{V}' = \mathbf{V}\bar{\mathbf{V}}$ and denote their columns by $\mathbf{u}'_{\mathrm{i}}$ and $\mathbf{v}'_{\mathrm{j}}$ respectively. Observe that $\text{span}(\mathbf{U}) = \text{span}(\bar{\mathbf{U}})$ and $\text{span}(\mathbf{V}) = \text{span}(\bar{\mathbf{V}})$ and furthermore that $\mathbf{U}'^T\mathbf{V}' = \text{diag}[\ldots, \sigma_{\mathrm{i}}, \ldots]$, that is

$$\mathbf{u}'^T_{\mathrm{i}} \mathbf{v}'_{\mathrm{j}} = \begin{cases} \sigma_{\mathrm{i}} & i = j \\ 0 & i \neq j. \end{cases}$$

The vectors $\mathbf{u}'_{\mathrm{i}}$ and $\mathbf{v}'_{\mathrm{j}}$ correspond to those in the constructive definition. We therefore have that the i-th principal angle α_{i} relates to the i-th singular value via $\sigma_{\mathrm{i}} = \cos \alpha_{\mathrm{i}}$.

C Chordal Distance

The chordal distance between points on the Grassmannian $\mathbb{G}(m,r)$, introduced and studied in [18], is defined via

$$\sqrt{\sum_{i=1}^{r} \sin^2 \alpha_{\mathrm{i}}},$$

where the α_{i} are the principal angles between the points described above. The authors of [72] introduce a notion of distance between a partially observed vector $\mathbf{x}^\Omega \in \mathbb{R}^m$ and a subspace $\mathbf{U} \in \mathbb{G}(m,r)$ via a formulation closely related to the chordal distance, which they give the same name.

Let $\mathbf{X}^0$ denoted the orthonormal matrix spanning all possible completions of $\mathbf{x}^\Omega$: If $\mathbf{x}^\Omega = \mathbf{0}$, then $\mathbf{X}^0 = \mathbf{I}$, the identity matrix. Otherwise, $\mathbf{X}^0$ is the $\mathrm{m} \times (\mathrm{m} - |\Omega| + 1)$ matrix formed with $\mathbf{x}^\Omega$ normalized and filled with zeros in the unobserved rows, concatenated with the $(\mathrm{m} - |\Omega|)$ canonical vectors indicating the unobserved rows of $\mathbf{x}^\Omega$. Let $\sigma_1(\mathbf{X}^{0T}\mathbf{U})$ denote the largest singular value of $\mathbf{X}^{0T}\mathbf{U}$. Then

$$d_c(\mathbf{x}^\Omega, \mathbf{U}) = \sin \alpha_1 = \sqrt{1 - \sigma_1^2(\mathbf{X}^{0T}\mathbf{U})}.$$

This metric is studied in [72]. Of particular importance is the following fact stated as Theorem 2 in [72]: the preimage of 0 under $d_c^2(\mathbf{x}^\Omega, \cdot)$ is the closure of the preimage of 0 under $f_F(\mathbf{x}^\Omega, \cdot)$, where

$$f_F(\mathbf{x}^\Omega, \mathbf{U}) = \min_{w \in \mathbb{R}^r} \|x^\Omega - \mathcal{P}_\Omega(\mathbf{U}w)\|_F^2,$$

and $\mathcal{P}_\Omega$ denotes projection onto the entries indexed by Ω. That is, f_F is the Frobenious norm, which is often used to search for subspaces $\mathbf{U}$ consistent with data. The Frobenius norm may not be continuous, whereas the chordal distance is continuous and differentiable.

D Geodesic Distance on the Grassmannian

The geodesic distance $d_g(\mathbf{U}_{\mathrm{i}}, \mathbf{U}_{\mathrm{j}})$ is derived from the intrinsic geometry of the Grassmann manifold and depends on the metric which defines the manifold structure. Let $\gamma : [a,b] \to \mathcal{M}$

be a curve on a general Riemannian manifold $(\mathcal{M}, g)$ with metric g. Then the length of γ is defined as [34]

$$L(\gamma) = \int_a^b \sqrt{g(\dot{\gamma}(t), \dot{\gamma}(t))}dt.$$

The canonical metric for the Grassmann manifold coincides with the Euclidean metric inherited from $O(m)$: $g_c(\dot{\mathbf{U}}_\mathrm{i}, \dot{\mathbf{U}}_\mathrm{j}) = g_e(\dot{\mathbf{U}}_\mathrm{i}, \dot{\mathbf{U}}_\mathrm{j}) = \mathsf{tr}(\dot{\mathbf{U}}_\mathrm{i}{}^\mathsf{T}\dot{\mathbf{U}}_\mathrm{j})$ [20]. To compute the geodesic distance, we therefore require knowledge of the geodesic segment connecting $\mathbf{U}_\mathrm{i}$ and $\mathbf{U}_\mathrm{j}$ with respect to the metric g_c. This is described in Lemma 1 of [72]: Let $\mathbf{V}_\mathrm{i}\mathbf{\Sigma}\mathbf{V}_\mathrm{j}^\mathsf{T}$ be the singular value decomposition of the matrix $\mathbf{U}_\mathrm{i}^\mathsf{T}\mathbf{U}_\mathrm{j}$, and denote the ℓ-th singular value by $\sigma_\ell = \cos\alpha_\ell$. Set $\bar{\mathbf{U}}_i = \mathbf{U}_\mathrm{i}\mathbf{V}_i$ and $\bar{\mathbf{U}}_\mathrm{j} = \mathbf{U}_\mathrm{j}\mathbf{V}_\mathrm{j}$ and note that $\bar{\mathbf{U}}_\mathrm{i}^\mathsf{T}\bar{\mathbf{U}}_\mathrm{j} = \mathbf{\Sigma}$. Then the geodesic with respect to g_c from $\mathbf{U}_\mathrm{i}$ to $\mathbf{U}_\mathrm{j}$ is given by $\mathbf{U}(t), 0 \leq t \leq 1$, where the path $\mathbf{U}(t)$ is given by

$$[\bar{\mathbf{U}}_\mathrm{i}, \mathbf{G}] \left[\begin{array}{c} \mathrm{diag}\left([\ldots, \cos\alpha_\ell t, \ldots]\right) \\ \mathrm{diag}\left([\ldots, \sin\alpha_\ell t, \ldots]\right) \end{array} \right] \mathbf{V}_\mathrm{i}^\mathsf{T},$$

where the columns of $\mathbf{G} = [\ldots, \boldsymbol{g}_\ell, \ldots] \in \mathbb{S}(r, m)$ are defined as

$$\boldsymbol{g}_\ell = \begin{cases} \frac{\bar{\mathbf{U}}_{2,:\ell} - \sigma_i \bar{\mathbf{U}}_{1,:\ell}}{\|\bar{\mathbf{U}}_{2,:\ell} - \sigma_\ell \bar{\mathbf{U}}_{1,:\ell}\|} & \text{if } \lambda_\ell \neq 1 \\ \mathbf{0} & \text{if } \lambda_\ell = 1. \end{cases}$$

Here, the subscript $:\ell$ denotes the ℓ-th column of the corresponding matrix.

We therefore have

$$\dot{\mathbf{U}}(t) = [\bar{\mathbf{U}}_\mathrm{i}, \mathbf{G}] \left[\begin{array}{c} \mathrm{diag}\left([\ldots, -\alpha_\ell \sin\alpha_\ell t, \ldots]\right) \\ \mathrm{diag}\left([\ldots, \quad \alpha_\ell \cos\alpha_\ell t, \ldots]\right) \end{array} \right] \mathbf{V}_\mathrm{i}^\mathsf{T}.$$

Denote $\mathbf{S} = \mathrm{diag}\left([\ldots, -\alpha_\ell \sin\alpha_\ell t, \ldots]\right)$ and $\mathbf{C} = \mathrm{diag}\left([\ldots, \alpha_\ell \cos\alpha_\ell t, \ldots]\right)$. Then

$$\begin{aligned} \dot{\mathbf{U}}^\mathsf{T}\dot{\mathbf{U}} &= \mathbf{V}_\mathrm{i}[\mathbf{S}, \mathbf{C}] \left[\begin{array}{c} \bar{\mathbf{U}}_\mathrm{i}^\mathsf{T} \\ \mathbf{G}^\mathsf{T} \end{array} \right] [\bar{\mathbf{U}}_\mathrm{i}, \mathbf{G}] \left[\begin{array}{c} \mathbf{S} \\ \mathbf{C} \end{array} \right] \mathbf{V}_\mathrm{i}^\mathsf{T} \\ &= \mathbf{V}_\mathrm{i}[\mathbf{S}, \mathbf{C}] \left[\begin{array}{cc} \bar{\mathbf{U}}_\mathrm{i}^\mathsf{T}\bar{\mathbf{U}}_\mathrm{i} & \bar{\mathbf{U}}_\mathrm{i}^\mathsf{T}\mathbf{G} \\ \mathbf{G}^\mathsf{T}\bar{\mathbf{U}}_\mathrm{i} & \mathbf{G}^\mathsf{T}\mathbf{G} \end{array} \right] \left[\begin{array}{c} \mathbf{S} \\ \mathbf{C} \end{array} \right] \mathbf{V}_\mathrm{i}^\mathsf{T} \\ &= \mathbf{V}_\mathrm{i}(\mathbf{S}^2 + \mathbf{C}^2)\mathbf{V}_\mathrm{i}^\mathsf{T} \\ &= \mathbf{V}_\mathrm{i}\mathrm{diag}([\ldots, \alpha_\ell^2, \ldots])\mathbf{V}_\mathrm{i}^\mathsf{T}, \end{aligned}$$

where we use the fact that $\bar{\mathbf{U}}_\mathrm{i}, \mathbf{G} \in \mathbb{S}(m, r)$, and that $\bar{\mathbf{U}}_\mathrm{i}^\mathsf{T}\mathbf{G} = \mathbf{0}$ [72]. Recall that $\mathsf{tr}(AB) = \mathsf{tr}(BA)$, hence $\mathsf{tr}(\dot{\mathbf{U}}^\mathsf{T}\dot{\mathbf{U}}) = \mathsf{tr}(\mathrm{diag}([\ldots, \alpha_\ell^2, \ldots])) = \sum_\ell \alpha_\ell^2$. We therefore have $L(\mathbf{U}(t)) = \int_0^1 \sqrt{\sum_i \alpha_\ell^2}dt = \sqrt{\sum_\ell \alpha_\ell^2}$. Recalling that $\sigma_\ell = \cos\alpha_\ell$, we finally have

$$d_g(\mathbf{U}_\mathrm{i}, \mathbf{U}_\mathrm{j}) \ = \ \sqrt{\sum_{\ell=1}^{\mathrm{r}} \arccos^2 \sigma_\ell(\mathbf{U}_\mathrm{i}^\mathsf{T}\mathbf{U}_\mathrm{j})}.$$

E Gradients on the Grassmannian

In this section, we derive the expressions in (2) and (3) that govern the fusion steps of our formulation. For a function $F(\mathbf{U})$ defined on the Grassmannian, the graduate of F at $\mathbf{U}$ is given by equation (2.70) in [20], which we record here:

$$\nabla F = F_\mathbf{U} - \mathbf{U}\mathbf{U}^\mathsf{T}F_\mathbf{U},$$

where $F_\mathbf{U}$ is the matrix whose entries are given by $[F_\mathbf{U}]_\mathrm{ij} = \frac{\partial F}{\partial \mathbf{U}_\mathrm{ij}}$.

Chordal gradient. To obtain the gradient of the chordal distance $d_c^2(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i})$ presented in (2), consider the partial derivative with respect to the $(\mathrm{a}, \mathrm{b})^\mathrm{th}$ element of $\mathbf{U}_\mathrm{i}$:

$$\left[\frac{\partial d_c^2(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i})}{\partial \mathbf{U}_\mathrm{i}}\right]_\mathrm{ab} \ = \ \frac{\partial}{\partial [\mathbf{U}_\mathrm{i}]_\mathrm{ab}} d_c^2(\mathbf{x}_\mathrm{i}^\Omega, \mathbf{U}_\mathrm{i}) \ = \ -2\sigma_1(\mathbf{X}_\mathrm{i}^{0\mathsf{T}}\mathbf{U}_\mathrm{i}) \frac{\partial \sigma_1(\mathbf{X}_\mathrm{i}^{0\mathsf{T}}\mathbf{U}_\mathrm{i})}{\partial [\mathbf{U}_\mathrm{i}]_\mathrm{ab}}.$$

To obtain the partial derivative of the leading singular value σ_1, observe that $\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}}$ and $\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}}$ share singular values: if $\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}} = \mathbf{V}\boldsymbol{\Sigma}\mathbf{W}^{\mathsf{T}}$, then $\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}} = (\mathbf{X}_{\mathrm{i}}^0\mathbf{V})\boldsymbol{\Sigma}\mathbf{W}^{\mathsf{T}}$ with $\mathbf{X}_{\mathrm{i}}^0\mathbf{V} \in \mathbb{S}(m,r)$, so the result is a compact singular value decomposition. Recall that $\mathbf{v}_{\mathrm{i}}$ and $\mathbf{w}_{\mathrm{i}}$ denote the leading left and right singular vectors of $\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}}$. Since $\mathbf{v}_{\mathrm{i}}^{\mathsf{T}}\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}}\mathbf{w}_{\mathrm{i}} = \sigma_1$, we have that

$$\frac{\partial\sigma_1}{\partial[\mathbf{U}_{\mathrm{i}}]_{\mathrm{ab}}} \;=\; \frac{\partial\mathbf{v}_{\mathrm{i}}^{\mathsf{T}}}{\partial[\mathbf{U}_{\mathrm{i}}]_{\mathrm{ab}}}\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}}\mathbf{w}_{\mathrm{i}} \;+\; \mathbf{v}_{\mathrm{i}}^{\mathsf{T}}\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\frac{\partial\mathbf{U}_{\mathrm{i}}}{\partial[\mathbf{U}_{\mathrm{i}}]_{\mathrm{ab}}}\mathbf{w}_{\mathrm{i}} \;+\; v_{\mathrm{i}}^{\mathsf{T}}\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}}\frac{\partial\mathbf{w}_{\mathrm{i}}}{\partial[\mathbf{U}_{\mathrm{i}}]_{\mathrm{ab}}}.$$

The first and third terms are zero, because $\mathbf{w}_{\mathrm{i}}$ is the leading right singular vector of $\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}}$, so $(\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}})\mathbf{w}_{\mathrm{i}} = \sigma_1\mathbf{v}_{\mathrm{i}}$, which implies

$$\frac{\partial\mathbf{v}_{\mathrm{i}}^{\mathsf{T}}}{\partial[\mathbf{U}_{\mathrm{i}}]_{\mathrm{ab}}}(\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{U}_{\mathrm{i}})\mathbf{w}_{\mathrm{i}} \;=\; \sigma_1\frac{\partial\mathbf{v}_{\mathrm{i}}^{\mathsf{T}}}{\partial[\mathbf{U}_{\mathrm{i}}]_{\mathrm{ab}}}\mathbf{v}_{\mathrm{i}},$$

and because $\frac{\partial\mathbf{v}_{\mathrm{i}}^{\mathsf{T}}}{\partial[\mathbf{U}_{\mathrm{i}}]_{ab}}\mathbf{v}_{\mathrm{i}} = 0$, as seen by differentiating both sides of $\mathbf{v}_{\mathrm{i}}^{\mathsf{T}}\mathbf{v}_{\mathrm{i}} = 1$ (and similarly for the third term). To compute the second term, note that $\mathbf{v}_{\mathrm{i}} \in \mathrm{span}(\mathbf{X}_{\mathrm{i}}^0)$ since it is a column of $\mathbf{X}_{\mathrm{i}}^0\mathbf{V}$, and the space spanned by $\mathbf{X}_{\mathrm{i}}^0$ is invariant under multiplication by $\mathbf{V}$. Now, $(\mathbf{v}_{\mathrm{i}}^{\mathsf{T}}\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}})^{\mathsf{T}} = \mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}\mathbf{v}_{\mathrm{i}} = \mathbf{v}_{\mathrm{i}}$, since $\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}}$ acts on vectors as the projection onto $\mathrm{span}(\mathbf{X}_{\mathrm{i}}^0)$. Hence $\mathbf{v}_{\mathrm{i}}^{\mathsf{T}}\mathbf{X}_{\mathrm{i}}^0\mathbf{X}_{\mathrm{i}}^{0\mathsf{T}} = \mathbf{v}_{\mathrm{i}}^{\mathsf{T}}$. The second term then becomes $\mathbf{v}_{\mathrm{i}}^{\mathsf{T}}\frac{\partial\mathbf{U}_{\mathrm{i}}}{\partial[\mathbf{U}_{\mathrm{i}}]_{\mathrm{ab}}}\mathbf{w}_{\mathrm{i}} = [\mathbf{v}_{\mathrm{i}}]_{\mathrm{a}}[\mathbf{w}_{\mathrm{i}}]_{\mathrm{b}}$. It follows that

$$\frac{\partial\sigma_1}{\partial[\mathbf{U}_{\mathrm{i}}]_{\mathrm{ab}}} = [\mathbf{v}_{\mathrm{i}}]_{\mathrm{a}}[\mathbf{w}_{\mathrm{i}}]_{\mathrm{b}}. \tag{7}$$

From this, we have $\boldsymbol{\nabla} d_c^2(\mathbf{x}_{\mathrm{i}}^{\Omega}, \mathbf{U}_{\mathrm{i}}) = -2(\mathbf{I} - \mathbf{U}_{\mathrm{i}}\mathbf{U}_{\mathrm{i}}^{\mathsf{T}})\sigma_1\mathbf{v}_{\mathrm{i}}\mathbf{w}_{\mathrm{i}}^{\mathsf{T}}$, where multiplication by $\mathbf{I} - \mathbf{U}_{\mathrm{i}}\mathbf{U}_{\mathrm{i}}^{\mathsf{T}}$ projects onto the tangent space of the Grassmannian at $\mathbf{U}_{\mathrm{i}}$, as described before [1, 20].

Geodesic gradient. For the gradient of the geodesic distance $d_g^2(\mathbf{U}_{\mathrm{i}}, \mathbf{U}_{\mathrm{j}})$ in (3) let us use σ_ℓ as shorthand for $\sigma_\ell(\mathbf{U}_{\mathrm{i}}^{\mathsf{T}}\mathbf{U}_{\mathrm{j}})$, and recall that $\mathbf{v}_{\mathrm{ij}}^{\ell}$ and $\mathbf{w}_{\mathrm{ij}}^{\ell}$ denote the ℓ^{th} left and right singular vectors of $\mathbf{U}_{\mathrm{j}}\mathbf{U}_{\mathrm{j}}^{\mathsf{T}}\mathbf{U}_{\mathrm{i}}$. Then the partial derivative with respect to the $(a,b)^{th}$ element of $\mathbf{U}_i$ is

$$\left[\frac{\partial d_g^2(\mathbf{U}_{\mathrm{i}}, \mathbf{U}_{\mathrm{j}})}{\partial\mathbf{U}_{\mathrm{i}}}\right]_{ab} \;=\; \sum_{\ell=1}^{r}\frac{-2\arccos\sigma_\ell}{\sqrt{1-\sigma_\ell^2}}\frac{\partial\sigma_\ell}{\partial[\mathbf{U}_{\mathrm{i}}]_{ab}} \;=\; \sum_{\ell=1}^{r}\frac{-2\arccos\sigma_\ell}{\sqrt{1-\sigma_\ell^2}}\mathbf{v}_{\mathrm{ij}}^{\ell}\mathbf{w}_{\mathrm{ij}}^{\ell\mathsf{T}},$$

where the first equality follows because $\sigma_\ell(\mathbf{U}_{\mathrm{i}}^{\mathsf{T}}\mathbf{U}_{\mathrm{j}}) = \sigma_\ell(\mathbf{U}_{\mathrm{j}}^{\mathsf{T}}\mathbf{U}_{\mathrm{i}}) = \sigma_\ell(\mathbf{U}_{\mathrm{j}}\mathbf{U}_{\mathrm{j}}^{\mathsf{T}}\mathbf{U}_{\mathrm{i}})$, and the second equality follows by parallel arguments as the derivation of (7) for the leading singular value. The last equation is the Euclidean gradient. Projecting onto the tangent space at $\mathbf{U}_{\mathrm{i}}$, as described before [1, 20], we obtain the following gradient on the Grassmannian

$$\boldsymbol{\nabla} d_g^2(\mathbf{U}_{\mathrm{i}}, \mathbf{U}_{\mathrm{j}}) \;=\; \sum_{\ell=1}^{r}\frac{-2\arccos\sigma_\ell}{\sqrt{1-\sigma_\ell^2}}(\mathbf{I} - \mathbf{U}_{\mathrm{i}}\mathbf{U}_{\mathrm{i}}^{\mathsf{T}})\mathbf{v}_{\mathrm{ij}}^{\ell}\mathbf{w}_{\mathrm{ij}}^{\ell\mathsf{T}}.$$