Clustering of Data with Missing Entries using Non-convex Fusion Penalties

Sunrita Poddar, *Student Member, IEEE,* and Mathews Jacob, *Senior Member, IEEE*

***Abstract*—The presence of missing entries in data often creates challenges for pattern recognition algorithms. Traditional algorithms for clustering data assume that all the feature values are known for every data point. We propose a method to cluster data in the presence of missing information. Unlike conventional clustering techniques where every feature is known for each point, our algorithm can handle cases where a few feature values are unknown for every point. For this more challenging problem, we provide theoretical guarantees for clustering using a l_0 fusion penalty based optimization problem. Furthermore, we propose an algorithm to solve a relaxation of this problem using saturating non-convex fusion penalties. It is observed that this algorithm produces solutions that degrade gradually with an increase in the fraction of missing feature values. We demonstrate the utility of the proposed method using a simulated dataset, the Wine dataset and the ASL dataset. It is shown that the proposed method is a promising clustering technique for datasets with large fractions of missing entries.**

I. Introduction

Clustering is an exploratory data analysis technique used to discover natural groupings in large datasets, with applications to analysis of gene expression data, image segmentation, identification of lexemes in handwritten text, search result grouping, and recommender systems [1]. A wide variety of clustering methods have been introduced over the years; see [2], [3], [4] for a review of classical methods. Common clustering techniques such as k-means [5], k-medians [6], and spectral clustering [7] are implemented using the Lloyd's algorithm. Recently, linear programming and semi-definite programming based convex relaxations of the above algorithms [8] were introduced to minimize the sensitivity to initialization. Hierarchical clustering methods [9], which produce easily interpretable and visualizable clustering results, have been recently introduced for applications where the number of clusters are unknown. The more recent convex clustering technique termed as sum-of-norms clustering [10] retains the advantages of hierarchical clustering, while being invariant to initialization, and producing a unique clustering path. Theoretical guarantees for successful clustering using the convex-clustering technique are also available [11].

Most of the above clustering algorithms cannot be directly applied to real-life datasets, when a large fraction of samples are missing. For example, gene expression data often contains missing entries due to image corruption, fabrication errors or contaminants [12], rendering gene cluster analysis difficult. Likewise, large databases used by recommender systems (e.g Netflix) usually have a huge amount of missing data, which makes pattern discovery challenging [13]. The presence of missing responses in surveys [14] and failing imaging sensors in astronomy [15], [16] are reported to make the analysis in these applications challenging. Several approaches were introduced to extend clustering to missing-data applications. For example, a partially observed dataset can be converted to a fully observed one using deletion of features that have missing entries or imputation of missing entries [17], followed by clustering. Similarly, an extension of the weighted sum-of-norms algorithm was done in [10], where the weights are estimated from the data points using imputation of missing entries [18]. Kernel-based methods for clustering have also been extended to deal with missing entries by replacing Euclidean distances with partial distances [19], [20]. A majorization-minimization algorithm was introduced to solve for the cluster-centers and cluster memberships in [21], which offers proven reduction in cost with iteration. In [22] and [23] the data points are assumed to lie on a mixture of K distributions, where K is known. The algorithms then alternate between the maximum likelihood estimation of the distribution parameters and the missing entries. While the above algorithms have been successfully demonstrated in a variety of applications, theoretical analysis of the clustering performance in the presence of missing entries is lacking. By contrast, missing data problems in the context of a variety of other data models has been well studied in the recent years. For instance, efficient algorithms along with theoretical guarantees have been proposed for low-rank matrix completion [24] and subspace clustering from data with missing entries [25], [26].

The main focus of this paper is to introduce an algorithm for the clustering of data with missing entries and to theoretically analyze the conditions for perfect clustering in the presence of missing data. The proposed algorithm is inspired by the sum-of-norms clustering technique [10]. Specifically, we formulate the recovery as an optimization problem, where each data point is assigned an auxiliary variable. The auxiliary variable is an estimate of the center of the cluster to which the specified point belongs. A fusion penalty is used to encourage the auxiliary variables to merge, whenever possible. We focus on the analysis of clustering using a ℓ_0 fusion penalty in the presence of missing entries, for an arbitrary number of clusters. The analysis reveals that perfect clustering is guaranteed with high probability, provided the number of measured entries (probability of sampling) is high enough; the required number of measured entries depends on parameters including intra-cluster variance and inter-cluster distance. Our analysis also shows that the performance is critically dependent on coherence, which is a measure of the concentration of inter cluster differences in the feature space. Specifically, if the separation between clusters is determined only by a very small subset of all the available features, then clustering becomes quite unstable if features in this subset are missing. Other factors

which influence the clustering technique are the number of features, number of clusters and total number of points.

We also introduce a relaxation of the above ℓ_0 penalty based clustering problem using non-convex saturating fusion penalties. The algorithm is demonstrated on a simulated dataset with different fractions of missing entries and cluster separations. We observe that the clustering performance degrades gradually with an increase in the number of missing entries. We also demonstrate the algorithm on clustering of the Wine dataset [27] and an Australian Sign Language (ASL) dataset [28].

II. Clustering using ℓ_0 fusion penalty

A. Background

We consider the clustering of points drawn from one of K distinct clusters $C_1, C_2, \ldots, C_K$. We denote the center of the clusters by $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_K \in \mathbb{R}^P$. For simplicity, we assume that there are M points in each of the clusters. The individual points in the k^{th} cluster are modelled as:

$$\mathbf{z}_k(m) = \mathbf{c}_k + \mathbf{n}_k(m); \quad m = 1, .., M, \;\; k = 1, \ldots, K \qquad (1)$$

Here, $\mathbf{n}_k(m)$ is the noise or the variation of $\mathbf{z}_k(m)$ from the cluster center $\mathbf{c}_k$. The set of input points $\{\mathbf{x}_i\}, i = 1, .., KM$ is obtained as a random permutation of the points $\{\mathbf{z}_k(m)\}$. The objective of a clustering algorithm is to estimate the cluster labels, denoted by $\mathcal{C}(\mathbf{x}_i)$ for $i = 1, .., KM$.

The sum-of-norms (SON) method is a recently proposed convex clustering algorithm [10]. Here, a surrogate variable $\mathbf{u}_i$ is introduced for each point $\mathbf{x}_i$, which is an estimate of the center of the cluster to which $\mathbf{x}_i$ belongs. As an example, let $K = 2$ and $M = 5$. Without loss of generality, let us assume that $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_5$ belong to $\mathcal{C}_1$ and $\mathbf{x}_6, \mathbf{x}_7, \ldots, \mathbf{x}_{10}$ belong to $\mathcal{C}_2$. Then, the desired solution is: $\mathbf{u}_1 = \mathbf{u}_2 = \ldots = \mathbf{u}_5 = \mathbf{c}_1$ and $\mathbf{u}_6 = \mathbf{u}_7 = \ldots = \mathbf{u}_{10} = \mathbf{c}_2$. In order to find the optimal $\{\mathbf{u}_i^*\}$, the following optimization problem is solved:

$$\{\mathbf{u}_i^*\} = \arg\min_{\{\mathbf{u}_i\}} \sum_{i=1}^{KM} \|\mathbf{x}_i - \mathbf{u}_i\|_2^2 + \lambda \sum_{i=1}^{KM} \sum_{j=1}^{KM} \|\mathbf{u}_i - \mathbf{u}_j\|_p \qquad (2)$$

The fusion penalty ($\|\mathbf{u}_i - \mathbf{u}_j\|_p$) can be enforced using different ℓ_p norms, out of which the ℓ_1, ℓ_2 and ℓ_∞ norms have been used in literature [10]. The use of sparsity promoting fusion penalties encourages sparse differences $\mathbf{u}_i - \mathbf{u}_j$, which facilitates the clustering of the points $\{\mathbf{u}_i\}$. For an appropriately chosen λ, the $\mathbf{u}_i$'s corresponding to $\mathbf{x}_i$'s from the same cluster converge to the same point. The above optimization problem is solved efficiently using the Alternating Direction Method of Multipliers (ADMM) algorithm and the Alternating Minimization Algorithm (AMA) [29]. Truncated ℓ_1 and ℓ_2 norms have also been used recently as the fusion penalty, which provide superior performance compared to convex penalties [30].

The sum-of-norms algorithm has also been used as a visualization and exploratory tool to discover patterns in datasets [18]. Clusterpath diagrams are a common way to visualize the data. This involves plotting the solution path as a function of the regularization parameter λ. For a very small value of λ, the solution is given by: $\mathbf{u}_i^* = \mathbf{x}_i$, i.e. each point forms its individual cluster. For a very large value of λ, the solution is given by: $\mathbf{u}_i^* = c$, i.e. every point belongs to the same cluster. For intermediate values of λ, more interesting behaviour is seen as various $\mathbf{u}_i^*$ merge and reveal the clusters in the data.

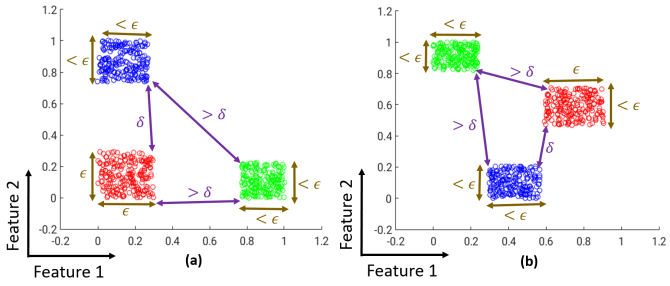


Fig. 1: Central Assumptions: (a) and (b) illustrate different instances where points belonging to $\mathbb{R}^2$ are to be separated into 3 different clusters (denoted by the colours red, green and blue). Assumptions A.1 and A.2 related to cluster separation and cluster size respectively, are illustrated in both (a) and (b). The importance of assumption A.3 related to feature concentration can also be appreciated by comparing (a) and (b). In (a), points in the red and blue clusters cannot be distinguished solely on the basis of feature 1, while the red and green clusters cannot be distinguished solely on the basis of feature 2. Thus, it is difficult to correctly cluster these points if either of the feature values is unknown. In (b), due to low coherence (as assumed in A.3), this problem does not arise.

In this paper, we extend the algorithm to account for missing entries in the data. We present theoretical guarantees for clustering using an ℓ_0 fusion penalty. Next, we approximate the ℓ_0 penalty by non-convex saturating penalties, and solve the resulting relaxed optimization problem using an iterative reweighted least squares (IRLS) strategy [31].

B. Central Assumptions

We make the following assumptions (illustrated in Fig 1), which are key to the successful clustering of the points:

A.1: **Cluster separation:** Points from different clusters are separated by $\delta > 0$ in the ℓ_2 sense, i.e:

$$\min_{\{m,n\}} \|\mathbf{z}_k(m) - \mathbf{z}_l(n)\|_2 \geq \delta; \;\; \forall\, k \neq l \qquad (3)$$

A.2: **Cluster size:** The maximum separation of points within any cluster in the ℓ_∞ sense is $\epsilon \geq 0$, i.e:

$$\max_{\{m,n\}} \|\mathbf{z}_k(m) - \mathbf{z}_k(n)\|_\infty = \epsilon; \;\; \forall k = 1, \ldots, K \qquad (4)$$

Thus, the k^{th} cluster is contained within a cube of size ϵ.

A.3: **Feature concentration:** The coherence of a vector $\mathbf{y} \in \mathbb{R}^P$ is defined as [24]:

$$\mu(\mathbf{y}) = \frac{P\|\mathbf{y}\|_\infty^2}{\|\mathbf{y}\|_2^2} \qquad (5)$$

By definition: $1 \leq \mu(\mathbf{y}) \leq P$. We bound the coherence of the difference between points from different clusters as:

$$\max_{\{m,n\}} \mu(\mathbf{z}_k(m) - \mathbf{z}_l(n)) \leq \mu_0; \;\; \forall\, k \neq l \qquad (6)$$

The coherence parameter μ plays a key role in the success of the algorithm. Intuitively, a vector with a high coherence has a few large values and several small ones. μ_0 defined in (6) is indicative of the difficulty of the clustering problem in the presence of missing data. If $\mu_0 = P$, then two clusters differ in only a single feature, suggesting that it is difficult to assign the correct cluster to a point if this feature is not sampled. The best case scenario is $\mu_0 = 1$, when all the features are equally important. In general, cluster recovery from missing data becomes challenging with increasing μ_0.

Under assumption A.2, the distance between two points in the same cluster is less than or equal to $\epsilon\sqrt{P}$. Also, by assumption A.1, the distance between two points in different clusters is greater than or equal to δ. Thus, the normalized ratio of the cluster size to cluster separation, specified by $\kappa = \frac{\epsilon\sqrt{P}}{\delta}$ is a measure of the difficulty of the clustering problem. Small values of κ suggest large inter-cluster separation compared to the cluster size; the recovery of such well-defined clusters is expected to be easier than the case with large κ values. Note that the ℓ_2 norm is used in the definition of δ, while the ℓ_∞ norm is used to define ϵ. If $\delta = \epsilon\sqrt{P}$, then $\kappa = 1$; this value of κ is of special importance since $\kappa < 1$ is a requirement for successful recovery in our main results. $\kappa < 1$ corresponds to the case where every intra-cluster distance is smaller than every inter-cluster distance.

We study the problem of clustering the points $\{\mathbf{x}_i\}$ in the presence of entries missing uniformly at random. We arrange the points $\{\mathbf{x}_i\}$ as columns of a matrix $\mathbf{X}$. The rows of the matrix are referred to as features. We assume that each entry of $\mathbf{X}$ is observed with probability p_0. The entries measured in the i^{th} column are denoted by:

$$\mathbf{y}_i = \mathbf{S}_i\,\mathbf{x}_i, \quad i = 1, .., KM \tag{7}$$

where $\mathbf{S}_i$ is the sampling matrix, formed by selecting rows of the identity matrix. We consider the following optimization problem to cluster data with missing entries:

$$\begin{aligned}\{\mathbf{u}_i^*\} = &\min_{\{\mathbf{u}_i\}} \sum_{i=1}^{KM}\sum_{j=1}^{KM} \|\mathbf{u}_i - \mathbf{u}_j\|_{2,0} \\ &\text{s.t } \|\mathbf{S}_i\,(\mathbf{x}_i - \mathbf{u}_i)\|_\infty \leq \frac{\epsilon}{2}, i \in \{1 \dots KM\}\end{aligned} \tag{8}$$

The $\ell_{2,0}$ norm is defined as:

$$\|\mathbf{x}\|_{2,0} = \begin{cases} 0 & \text{, if } \|\mathbf{x}\|_2 = 0 \\ 1 & \text{, otherwise} \end{cases} \tag{9}$$

Similar to the SON scheme (2), we expect that all $\mathbf{u}_i$'s that correspond to $\mathbf{x}_i$ in the same cluster are equal, while $\mathbf{u}_i$'s from different clusters are not equal. We consider the cluster recovery to be successful when there are no mis-classifications. We claim that the above algorithm can successfully recover the clusters with high probability when:

1) The clusters are well separated (i.e, low $\kappa = \frac{\epsilon\sqrt{P}}{\delta}$)).
2) The sampling probability p_0 is sufficiently high.
3) The coherence μ_0 is small.

C. Theoretical guarantees for correct clustering

We now move on to a formal statement and proof of this result. All the important symbols used in the paper have been summarized in Table I. We first define the following quantities, which will be used in our results below.

- Upper bound for probability that two points have less than half the expected number ($\frac{p_0^2 P}{2}$) of commonly observed locations:
$$\gamma_0 \coloneqq \left(\frac{e}{2}\right)^{-\frac{p_0^2 P}{2}} \tag{10}$$
- Upper bound for probability that two points from different clusters can yield the same $\mathbf{u}$ without violating the constraints in (8), when they have more than $\frac{p_0^2 P}{2}$ commonly observed locations:
$$\delta_0 \coloneqq e^{-\frac{p_0^2 P(1-\kappa^2)^2}{\mu_0^2}} \tag{11}$$
- Upper bound for probability that two points from different clusters can yield the same $\mathbf{u}$ without violating the constraints in (8), irrespective of the number of commonly observed locations:
$$\beta_0 \coloneqq 1 - (1-\delta_0)(1-\gamma_0) \tag{12}$$
- Upper bound for failure probability of (8):
$$\eta_0 \coloneqq \sum_{\{m_j\}\in\mathcal{S}} \left[\beta_0^{\frac{1}{2}(M^2 - \sum_j m_j^2)} \prod_j \binom{M}{m_j}\right] \tag{13}$$
where $\mathcal{S}$ is the set of all sets of positive integers $\{m_j\}$ such that: $2 \leq \mathcal{U}(\{m_j\}) \leq K$ and $\sum_j m_j = M$. Here, the function $\mathcal{U}$ counts the number of non-zero elements in a set. For example, if $K = 2$ then $\mathcal{S}$ contains all sets of 2 positive integers $\{m_1, m_2\}$, such that $m_1 + m_2 = M$. Thus, $\mathcal{S} = \{\{1, M-1\}, \{2, M-2\}, \{3, M-3\}, \dots, \{M-1, 1\}\}$ and (13) reduces to:
$$\eta_0 = \sum_{i=1}^{M-1} \left[\beta_0^{i(M-i)} \binom{M}{i}^2\right] \tag{14}$$
- Since the expression for η_0 is quite involved, we simplify it the special case where there are only two clusters. Under the assumption that $\log\beta_0 \leq \frac{1}{M-1} + \frac{2}{M-2}\log\frac{1}{M-1}$, it can be shown (derived in Appendix F) that η_0 is upper-bounded as:
$$\begin{aligned}\eta_0 &= \sum_{i=1}^{M-1} \left[\beta_0^{i(M-i)} \binom{M}{i}^2\right] \\ &\leq M^3 \beta_0^{M-1} \\ &\coloneqq \eta_{0,\text{approx}}\end{aligned} \tag{15}$$

We first consider the data consistency constraint in (8) and determine possible feasible solutions. All the points in any specified cluster can share a center without violating the data consistency constraint:

Lemma II.1. *Consider any two points $\mathbf{x}_1$ and $\mathbf{x}_2$ from the same cluster. Then, there exists a $\mathbf{u}$ that satisfies the data consistency conditions specified by:*

$$\|\mathbf{S}_i\,(\mathbf{x}_i - \mathbf{u})\|_\infty \quad \leq \quad \frac{\epsilon}{2}; \quad i = 1, 2 \tag{16}$$

TABLE I: Notations used

K	Number of clusters
M	Number of points in each cluster
P	Number of features for each point
$\mathcal{C}_i$	The i^{th} cluster
$\mathbf{c}_i$	center of $\mathcal{C}_i$
$\mathbf{z}_i(m)$	m^{th} point in $\mathcal{C}_i$
$\{\mathbf{x}_i\}$	Random permutation of KM points $\{\mathbf{z}_k(m)\}$ for $k \in \{1,2,\ldots,K\}, m \in \{1,2,\ldots,M\}$
$\mathbf{S}_i$	Sampling matrix for $\mathbf{x}_i$
$\mathbf{X}$	Matrix formed by arranging $\{\mathbf{x}_i\}$ as columns, such that the i^{th} column is $\mathbf{x}_i$
p_0	Probability of sampling each entry in $\mathbf{X}$
δ	Cluster separation defined in (3)
ϵ	Cluster size defined in (4)
κ	Defined as $\kappa = \frac{\epsilon\sqrt{P}}{\delta}$
μ_0	Parameter related to coherence defined in (6)
γ_0	Defined in (10)
δ_0	Defined in (11)
β_0	Defined in (12)
η_0	Defined in (13)
$\eta_{0,\text{approx}}$	Upper bound for η_0 for the case of 2 clusters, defined in (15)

This result follows from the assumption of the cluster size, and is proven in Appendix A. In contrast, points from different clusters cannot share a center with high probability:

Lemma II.2. *Consider any two points $\mathbf{x}_1$ and $\mathbf{x}_2$ from different clusters. If $\kappa < 1$, then with probability at least $1-\beta_0$ there exists no $\mathbf{u}$ that satisfies the data-consistency relations:*

$$\|\mathbf{S}_i(\mathbf{x}_i - \mathbf{u})\|_\infty \quad \leq \quad \frac{\epsilon}{2}; \quad i = 1,2 \tag{17}$$

The proof of this lemma is in Appendix C. To get some intuition regarding the above result, let us define $\mathbf{S}_1 = \mathbf{S}_{\mathcal{I}_1}$ and $\mathbf{S}_2 = \mathbf{S}_{\mathcal{I}_2}$, where $\mathcal{I}_1$ and $\mathcal{I}_2$ are the index sets of the features that are sampled (not missing) in $\mathbf{x}_1$ and $\mathbf{x}_2$ respectively. We observe that (17) can be satisfied, if the features of $\mathbf{x}_1$ and $\mathbf{x}_2$ are not very different on the index set $\mathcal{I}_1 \cap \mathcal{I}_2$, which is the set of commonly observed locations. If the probability of sampling p_0 is sufficiently high, then the cardinality of the set of common locations, specified by $|\mathcal{I}_1 \cap \mathcal{I}_2| = q$, will be high, with high probability. If the coherence μ_0 defined in assumption A3 is low, then with high probability the vector $\mathbf{x}_1 - \mathbf{x}_2$ does not have q small entries. Thus, for a small value of μ_0 and high p_0, (17) occurs with a low probability β_0.

The above result can be generalized to consider a large number of points from multiple clusters. If we choose M points such that not all of them belong to the same cluster, then it can be shown that with high probability, they cannot share the same $\mathbf{u}$ without violating the constraints in (8):

Lemma II.3. *Assume that $\{\mathbf{x}_i : i \in \mathcal{I}, |\mathcal{I}| = M\}$ is a set of points chosen randomly from multiple clusters (not all are from the same cluster). If $\kappa < 1$, a solution $\mathbf{u}$ does not exist for the following equations:*

$$\|\mathbf{S}_i(\mathbf{x}_i - \mathbf{u})\|_\infty \leq \frac{\epsilon}{2}; \quad \forall i \in \mathcal{I} \tag{18}$$

with probability exceeding $1 - \eta_0$.

The proof of this lemma is in Appendix D. The key message of the above result is that large clusters with mis-classified results are highly unlikely. We will show that all feasible solutions containing small mis-classified clusters are associated with higher cost than the correct solution. Thus, we can conclude that the algorithm recovers the ground truth solution with high probability, as summarized below.

Theorem II.4. *If $\kappa < 1$, the solution to the optimization problem* (8) *is identical to the ground-truth clustering with probability exceeding $1 - \eta_0$.*

The proof of the above theorem is in Appendix E. We note that for a low value of β_0 and a high value of M (number of points in each cluster), we will obtain a very low value of η_0. The only non-zero terms in the objective function of (8) are the differences between centers of distinct clusters. Intuitively, the value of the objective function could be made equal to 0 by assigning all the points to the same cluster. However, this is not allowed by the constraints of the optimization problem which are based on our known observations of the data points. Using similar arguments, under our theoretical assumptions, the objective function value cannot be made arbitrarily low by assigning all the points to a very few large clusters. This idea is captured by Lemma II.3. It turns out (as shown in the proof of Theorem II.4) that under our theoretical assumptions, the optimization problem has a unique minimizer given by the correct clustering with high probability. While the objective function value for this solution can be large if the data contains a large number of clusters, it will nevertheless be the minimum value that the objective function can attain under the given constraints.

For the special case where there are no missing entries, we have the following result which is proved in Appendix G.

Theorem II.5. *If $\kappa < 1$, the solution to the optimization problem* (8) *with $\mathbf{S}_i = \mathbf{I}, \forall i = 1,\ldots,KM$ is identical to the ground-truth clustering.*

III. Relaxation of the ℓ_0 penalty

The results in the previous section provide important insights on the difficulty of the clustering problem in the presence of missing data. However, the optimization problem in (8) is NP-hard. We hence consider a computationally feasible relaxation of the optimization problem (8) in this section.

A. Relaxed optimization problem

We consider a relaxed problem where ϕ is a function approximating the ℓ_0 norm:

$$\begin{aligned}\{\mathbf{u}_i^*\} = &\min_{\{\mathbf{u}_i\}} \sum_{i=1}^{KM}\sum_{j=1}^{KM} \phi\left(\|\mathbf{u}_i - \mathbf{u}_j\|_2\right) \\ &\text{s.t } \|\mathbf{S}_i(\mathbf{x}_i - \mathbf{u}_i)\|_\infty \leq \frac{\epsilon}{2}, i \in \{1\ldots KM\}\end{aligned} \tag{19}$$

Some examples of such functions are:

- ℓ_p norm: $\phi(x) = |x|^p$, for some $0 < p < 1$.
- H_1 penalty: $\phi(x) = 1 - e^{-\frac{x^2}{2\sigma^2}}$.

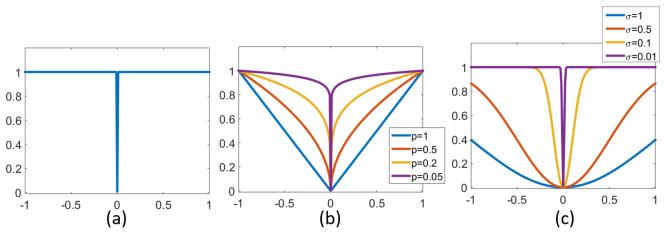


Fig. 2: Different penalty functions ϕ. (a) The ℓ_0 norm (b) The ℓ_p penalty function which is non-convex for $0 < p < 1$ and convex for $p = 1$ (c) The H_1 penalty function. The ℓ_p and H_1 penalties closely approximate the ℓ_0 norm for low values of p and σ respectively.

These functions approximate the ℓ_0 penalty more accurately for lower values of p and σ, as illustrated in Fig 2. We can solve (19) using a majorize-minimize strategy. Specifically, by majorizing the penalty ϕ using a quadratic surrogate functional, we obtain:

$$\phi(x) \leq w(x)x^2 + d \tag{20}$$

where $w(x) = \frac{\phi^{'}(x)}{2x}$, and d is a constant. For the two penalties considered here, we obtain the weights as:

- ℓ_p norm: $w(x) = (\frac{2}{p}x^{(2-p)} + \alpha)^{-1}$. The infinitesimally small α term is introduced to deal with situations where $x = 0$. For $x \neq 0, w(x) \approx \frac{p}{2}x^{p-2}$.
- H_1 penalty: $w(x) = \frac{1}{2\sigma^2}e^{-\frac{x^2}{2\sigma^2}}$.

With this majorization, (19) can be solved by alternating between the box-constrained quadratic optimization problem:

$$\begin{aligned}\{\mathbf{u}_i^*\} = &\min_{\{\mathbf{u}_i\}} \sum_{i=1}^{KM}\sum_{j=1}^{KM} w_{i,j}\|\mathbf{u}_i - \mathbf{u}_j\|_2 \\ &\text{s.t } \|\mathbf{S}_i(\mathbf{x}_i - \mathbf{u}_i)\|_\infty \leq \frac{\epsilon}{2}, i \in \{1 \dots KM\}\end{aligned} \tag{21}$$

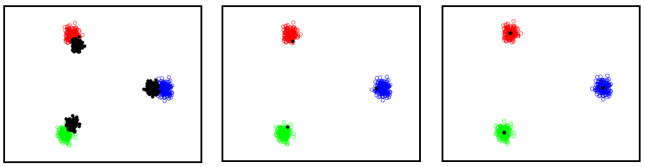
and the computation of the weights $w_{i,j}$. We refer to this iterative solution as the constrained solution.

While the above formulation is consistent with our theoretical formulation, it is computationally intensive to solve the constrained problem for large datasets. In addition, the exact value of ϵ may be unknown in practical applications. Therefore, we propose to solve the following unconstrained problem:

$$\{\mathbf{u}_i^*\} = \arg\min_{\{\mathbf{u}_i\}} \sum_{i=1}^{KM} \|\mathbf{S}_i(\mathbf{u}_i - \mathbf{x}_i)\|_2^2 + \lambda \sum_{i=1}^{KM}\sum_{j=1}^{KM} \phi(\|\mathbf{u}_i - \mathbf{u}_j\|_2) \tag{22}$$

We now state a majorize-minimize formulation for (22):

$$\begin{aligned}\{\mathbf{u}_i^*, w_{ij}^*\} = \arg &\min_{\{\mathbf{u}_i, w_{ij}\}} \sum_{i=1}^{KM} \|\mathbf{S}_i(\mathbf{u}_i - \mathbf{x}_i)\|_2^2 \\ &+\lambda \sum_{i=1}^{KM}\sum_{j=1}^{KM} w_{ij}\|\mathbf{u}_i - \mathbf{u}_j\|_2^2\end{aligned} \tag{23}$$



(a) l_1 norm (b) $l_{0.1}$ penalty (c) H_1 penalty

Fig. 3: Comparison of different penalties. We show here the 2 most significant principal components of the solutions obtained using the IRLS algorithm. (a) It is seen that the ℓ_1 penalty is unable to cluster the points even though the clusters are well-separated. (b) The $\ell_p; p = 0.1$ penalty is able to cluster the points correctly. However, the cluster-centers are not correctly estimated. (c) The H_1 penalty correctly clusters the points and also gives a good estimate of the centers.

In order to solve problem (23), we alternate between two sub-problems till convergence. At the n^{th} iteration, these sub-problems are:

$$w_{ij}^{(n)} = \frac{\phi^{'}\left(\|\mathbf{u}_i^{(n-1)} - \mathbf{u}_j^{(n-1)}\|_2\right)}{2\|\mathbf{u}_i^{(n-1)} - \mathbf{u}_j^{(n-1)}\|_2} \tag{24}$$

$$\begin{aligned}\{\mathbf{u}_i^{(n)}\} = \arg\min_{\{\mathbf{u}_i\}} &\sum_{i=1}^{KM} \|\mathbf{S}_i(\mathbf{u}_i - \mathbf{x}_i)\|_2^2 \\ &+\lambda \sum_{i=1}^{KM}\sum_{j=1}^{KM} w_{ij}^{(n)}\|\mathbf{u}_i - \mathbf{u}_j\|_2^2\end{aligned} \tag{25}$$

We observe experimentally that the formulations (22) and (19) offer qualitatively similar results. Our experiments also show that for the optimal value of λ, the formulation in (22) may offer more accurate solutions than the constrained formulation in (19). Note that the data consistency term in (19) is the maximum likelihood term when the cluster centers are corrupted by Gaussian noise; it is expected to provide more noise-robust estimates. For certain choices of the ϕ function, it is guaranteed that the IRLS iterations will converge to a critical point of the objective function (23). This result is stated below.

Theorem III.1. *Let $\phi : [0, \infty) \to [0, \infty)$ be a continuously twice differentiable function such that $\phi \circ \sqrt{.}$ is a strictly concave function, with $\phi(0) = 0$, $\phi^{'}(0) = c$ and $0 < \phi^{'}(x) \leq c$. Then the iterates defined as* (24) *and* (25) *converge to a stationary point of the objective function in* (23)*, or the accumulation points of $\{\mathbf{U}^{(n)}\}$ form a continuum in the set of stationary points of the objective function in* (23)*.*

The proof of the above result is in Appendix H and follows from [32]. The H_1 penalty satisfies the conditions required by the above theorem. Thus, the IRLS iterates (24) and (25) defined for the H_1 penalty will converge to a critical point of (22), unless the objective function in (22) contains flat regions, i.e. the critical points are not isolated. These flat regions will be present if a particular row of $\mathbf{X}$ is never sampled. This can be regarded as an artefact of the sampling scheme. In this case,

the iterates will form a continuum in the flat region. Moreover, regardless of the existence of flat regions, the iterates will converge to a minimum, unless it is initialized at a maximum.

B. Comparison of penalties

We compare the performance of different fusion penalties when used as a surrogate for the ℓ_0 norm. For this purpose, we use a simulated dataset with points in $\mathbb{R}^{50}$ belonging to three well-separated clusters, such that $\{\mathbf{x}_i\}_{i=1}^{200} \in \mathcal{C}_1$, $\{\mathbf{x}_i\}_{i=201}^{400} \in \mathcal{C}_2$ and $\{\mathbf{x}_i\}_{i=401}^{600} \in \mathcal{C}_3$. We do not consider the presence of missing entries for this experiment. We solve (22) to cluster the points using the ℓ_1, $\ell_{0.1}$ and H_1 (for $\sigma = 0.5$) penalties. The results are shown in Fig 3. Only for the purpose of visualization, we take a PCA of the data matrix $\mathbf{X} \in \mathbb{R}^{50\times 600}$ and retain the two most significant principal components to get a matrix of points $\in \mathbb{R}^{2\times 600}$. These points are plotted in the figure, with red, blue and green representing points from different clusters. We similarly plot the two most significant components of the estimated centers in black. In (b) and (c), we note that $\mathbf{u}_1^* = \mathbf{u}_2^* = \ldots = \mathbf{u}_{200}^*$, $\mathbf{u}_{201}^* = \mathbf{u}_{202}^* = \ldots = \mathbf{u}_{400}^*$ and $\mathbf{u}_{401}^* = \mathbf{u}_{402}^* = \ldots = \mathbf{u}_{600}^*$. Thus, the $\ell_{0.1}$ and H_1 penalties are able to correctly cluster the points. This behaviour is not seen in (a). We conclude that the convex ℓ_1 penalty is unable to cluster the points.

The cluster-centers estimated using the $\ell_{0.1}$ penalty are inaccurate. The H_1 penalty out-performs the other two penalties and accurately estimates the cluster-centers. We can explain this behaviour intuitively by Fig 2. The ℓ_1 norm penalizes differences between all pairs of points. The $\ell_{0.1}$ and H_1 functions penalize differences between points that are close. Due to the saturating nature of the penalties, they do not heavily penalize differences between points that are further away. However, we note that the H_1 penalty saturates to 1 very quickly, similar to the ℓ_0 norm. This behaviour is missing for the $\ell_{0.1}$ penalty, and for this reason, it shrinks the distance between the estimated centers of different clusters.

C. Initialization Strategies

Since the cost function is non-convex, the algorithm requires good initialization of the weights w_{ij} for convergence to the correct cluster center estimates. We consider two different strategies for initializing the weights:

- Partial Distances [25], [33]: Consider a pair of points $\mathbf{x}_1, \mathbf{x}_2$ observed by sampling matrices $\mathbf{S}_1 = \mathbf{S}_{\mathcal{I}_1}$ and $\mathbf{S}_2 = \mathbf{S}_{\mathcal{I}_2}$ respectively. Let the set of common indices be $\omega := \mathcal{I}_1 \cap \mathcal{I}_2$. We define the partial distance as $\|\mathbf{y}_\omega\| = \sqrt{\frac{P}{|\omega|}}\|\mathbf{x}_{1\omega} - \mathbf{x}_{2\omega}\|$, where $\mathbf{x}_{i\omega}$ represents the set of entries of $\mathbf{x}_i$ restricted to the index set ω. Instead of the actual distances which are not available, the partial distances $\|\mathbf{y}_\omega\|$ can be used for computing the weights.

- Imputation Methods: The weights can be computed from estimates $\{\mathbf{u}_i^{(0)}\}$, where:

$$\mathbf{u}_i^{(0)} = \mathbf{S}_i\mathbf{x}_i + (\mathbf{I} - \mathbf{S}_i)\mathbf{m} \tag{26}$$

  Here $\mathbf{m}$ is a constant vector, specific to the imputation technique. The zero-filling technique corresponds to $\mathbf{m} = \mathbf{0}$. Better estimation techniques can be derived where the j^{th} row of $\mathbf{m}$ can be set to the mean of all measured values in the j^{th} row of $\mathbf{X}$.

We will observe experimentally that for a good approximation of the initial weights $\mathbf{W}^{(0)}$, we get the correct clustering. Conversely, the clustering fails for a bad initial guess. Our experiments demonstrate the superiority of a partial distance based initialization strategy over a zero-filled initialization.

IV. Results

We demonstrate the impact of the different parameters on the theoretical bounds in Theorem II.4. We also test the proposed algorithm on simulated and real datasets. The simulations are used to study the performance of the algorithm with change in parameters such as fraction of missing entries, number of points to be clustered etc. We also study the effect of different initialization techniques on the algorithm performance. We demonstrate the algorithm on a Wine dataset [27], and an Australian sign language (ASL) dataset [28].

A. Variation of theoretical prediction with parameters

We plot the quantities $\gamma_0, \delta_0, \beta_0$ and η_0 (defined in section II-C) as a function of parameters p_0, P, κ and M in Fig 4. γ_0 is an upper bound for the probability that a pair of points have $< \frac{p_0^2 P}{2}$ entries observed at common locations. In Fig 4 (a), the change in γ_0 is shown as a function of p_0 for different values of P. In subsequent plots, we fix $P = 50$ and $\mu_0 = 1.5$. δ_0 is an upper bound for the probability that a pair of points from different clusters can share a common center, given that $\geq \frac{p_0^2 P}{2}$ entries are observed at common locations. In Fig 4 (b), the change in δ_0 is shown as a function of p_0 for different values of κ. In Fig 4 (c), the behaviour of $\beta_0 = 1 - (1-\gamma_0)(1-\delta_0)$ is shown, which is the probability mentioned in Lemma II.2.

We consider the two cluster setting, (i.e. $K = 2$) for subsequent plots. η_0 is the probability of failure of the clustering algorithm (8). In (d), plots are shown for $(1-\eta_0)$ as a function of p_0 for different values of κ and M. As expected, the probability of success of the clustering algorithm increases with increase in p_0 and M and decrease in κ.

B. Clustering of Simulated Data

We simulated datasets with $K = 2$ disjoint clusters in $\mathbb{R}^{50}$ with a varying number of points per cluster. The points in each cluster follow a uniform random distribution, around the mean. We study the probability of success of the H_1 penalty based clustering algorithm with partial-distance based initialization as a function of κ, M and p_0. For a particular set of parameters the experiment was conducted twenty times to compute the probability of success of the algorithm. Between these trials, the cluster-centers remain the same, while the points sampled from these clusters are different and the locations of the missing entries are different. Fig 5 (a) shows the result for datasets with $\kappa = 0.39$ and $\mu_0 = 2.3$. The theoretical guarantees for successfully clustering the dataset are shown in (b). Note that the theoretical guarantees do not assume that the points are taken from a uniform random distribution. Also, the

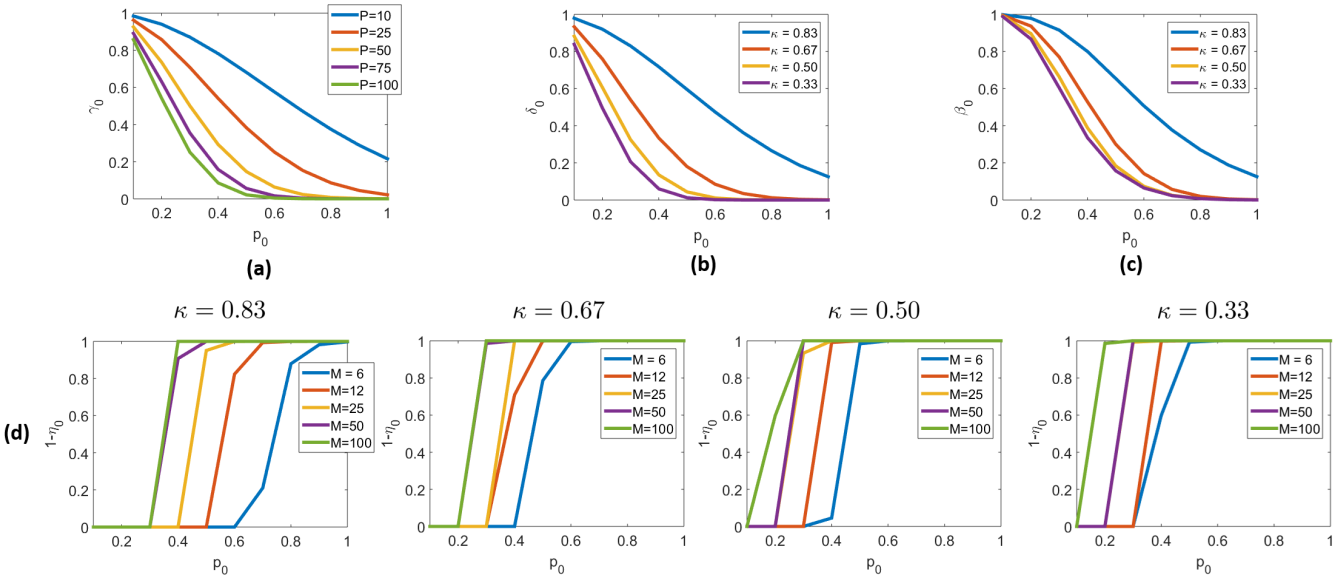


Fig. 4: Variation of the recovery probabilities with parameters: The quantities γ_0, δ_0 and β_0 defined in Section II-C are plotted in (a), (b) and (c), respectively. In (b) and (c), $P = 50$ and $\mu_0 = 1.5$ are assumed. β_0 gives the probability that 2 points from different clusters can share a center. As expected, this value decreases with increase in p_0 and decrease in κ. Considering $K = 2$ clusters, a lower bound for the probability of successful clustering $(1 - \eta_0)$ using the proposed algorithm is shown in (d) for different values of κ.

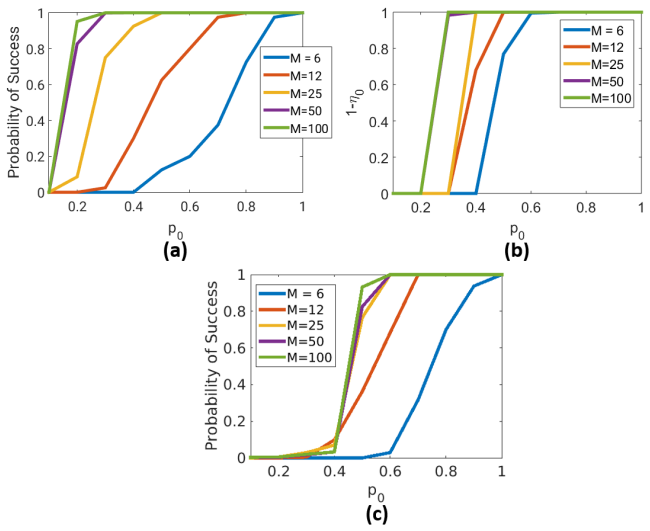


Fig. 5: Experimental results for probability of success. Guarantees are shown for a simulated dataset with $K = 2$ clusters. The clustering was performed using (23) with an H_1 penalty and partial distance based initialization. For (a) and (b) it is assumed that $\kappa = 0.39$ and $\mu_0 = 2.3$. (a) shows the experimentally obtained probability of success of clustering for clusters with points from a uniform random distribution. (b) shows the theoretical lower bound for the probability of success. (c) shows the experimentally obtained probability of success for a more challenging dataset with $\kappa = 1.15$ and $\mu_0 = 13.2$. Note that we do not have theoretical guarantees for this case, since our analysis assumes that $\kappa < 1$.

bounds assume that we are solving the original problem using a ℓ_0 norm, whereas the experimental results were generated for the H_1 penalty. Our guarantees hold for $\kappa < 1$. However, we demonstrate in (c) that even for the more challenging case where $\kappa = 1.15$ and $\mu_0 = 13.2$, the clustering is successful.

We simulated Dataset-1 with $K = 3$ disjoint clusters in $\mathbb{R}^{50}$ and $M = 200$ points in each cluster. In order to generate this dataset, three cluster centers in $\mathbb{R}^{50}$ were chosen from a uniform random distribution. The distances between the three pairs of cluster-centers are 3.5, 2.8 and 3.3 units respectively. For each of these three cluster centers, 200 noisy instances were generated by adding zero-mean white Gaussian noise of variance 0.1. The dataset was sub-sampled with varying fractions of missing entries ($p_0 = 1, 0.9, 0.8, \ldots, 0.3, 0.2$). The locations of the missing entries were chosen uniformly at random from the full data matrix. We also generate Dataset-2 by halving the distance between the cluster centers, while keeping the intra-cluster variance fixed. Both the proposed initialization techniques for the IRLS algorithm (i.e. zero-filling and partial-distance) are also tested here. The results are shown in Fig 6. The centres are estimated for different values of λ. Since the estimated centres lie in $\mathbb{R}^{50}$, we take a PCA of the estimated centers (similar to Fig 3) and plot the two most significant components. The lines in the figure trace the path of the individual cluster centers as the regularization parameter λ increases. For lower λ values, the cluster centres are the same as the points themselves. For very high λ values, all clusters merge and all the estimated cluster centres are equal in value. The three colours distinguish the estimated cluster centres according to their ground-truth cluster memberships. These colours have been shown only for easier visualization.

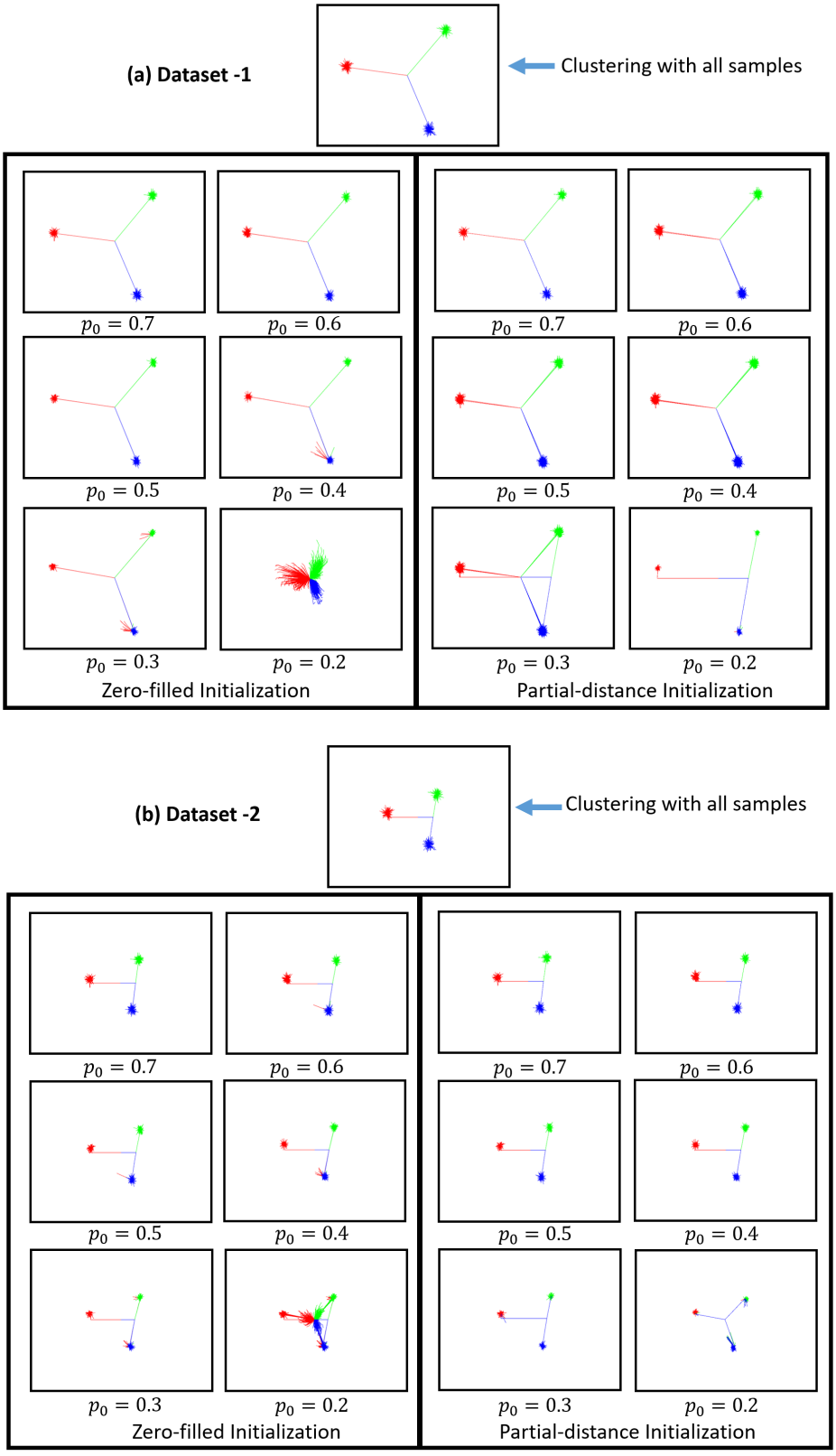


Fig. 6: Clustering results in simulated datasets. The H_1 penalty is used to cluster two datasets with varying fractions of missing entries. Results are presented with different initialization techniques (zero-filled and partial-distance based). We show here the two most significant principal components of the solutions. The clusterpaths are shown as a function of the regularization parameter λ. Inter-cluster distances in Dataset 2 are half of those in Dataset 1, while intra-cluster distances remain the same. Consequently, Dataset 1 performs better at a higher fraction of missing entries. For the partial-distance based initialization, the cluster center estimates are relatively stable with varying fractions of missing entries.

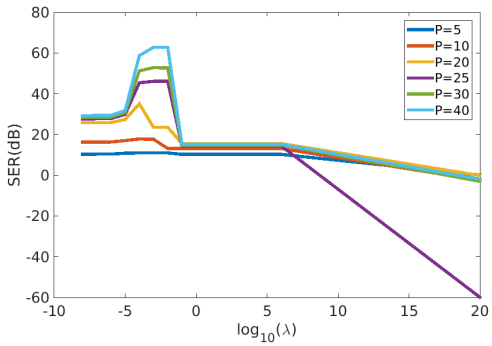


Fig. 7: Effect of changing dimension in simulated datasets. The proposed scheme is used to cluster datasets with varying ambient dimension P. The SER of the estimated cluster centres is plotted as a function of the regularization parameter λ. It is observed that the achievable SER increases with increase in the ambient dimension.

The colours are not inferred by the algorithm, but known to us as ground-truth. As expected, we observe that the clustering algorithms are more stable with fewer missing entries. We also observe that the partial distance based initialization technique out-performs the zero-filled initialization. Thus, we use this scheme for subsequent experiments on real datasets.

We study the performance of the proposed scheme with change in the ambient dimension P in Fig. 7 . For all these experiments, we consider simulated data with $p_0 = 0.4$. For the study on effect of ambient dimension, datasets with $K = 3$ clusters are considered. It is observed that the performance improves with an increase in ambient dimension, as is predicted by theory by the exponential dependence of γ_0 and δ_0 on P.

We study the performance as a function of the number of clusters K in Fig. 8. For the study on varying number of clusters, we consider two cases. For both these cases, an ambient dimension of $P = 50$ is assumed. In the first case, the number of points in each cluster is fixed as $M = 200$, and thus the total number of points is $200K$. In the second case, the total number of points is fixed to 600. The number of points per cluster is then $\frac{600}{K}$. The results are illustrated in Fig 8. We observe that in case-1, the performance of the algorithm is invariant to the number of clusters. In case-2, the performance of the algorithm degrades with an increase in the number of clusters. In this case, more observed entries per data point are required as K grows.

C. Comparison with other methods

We have compared the proposed scheme with several other schemes on the task of clustering data with missing entries. We first present results for both the constrained (19) and unconstrained (22) versions of the proposed scheme. The competing methods include several different versions of the sum-of-norms formulation, as well as methods for subspace clustering in the presence of missing entries. The SON formulations are of the type:

$$\{\mathbf{u}_i^*\} = \arg\min_{\{\mathbf{u}_i\}} \sum_{i=1}^{KM} \|\mathbf{S}_i(\mathbf{x}_i - \mathbf{u}_i)\|_2^2 + \lambda \sum_{i=1}^{KM}\sum_{j=1}^{KM} w_{ij}\|\mathbf{u}_i - \mathbf{u}_j\| \tag{27}$$

The different versions we use here are:

1) l1-ZF-W-m: In this formulation, the weights $\{w_{ij}\}$ are computed using partial distances.
2) l1-ZF-W: Here, the weights are obtained as above, and $\mathbf{S}_i = \mathbf{I}, \forall i$. The missing entries are imputed to zero.
3) l1-ZF: Here, we set $w_{ij} = 1, \forall i, j$, and $\mathbf{S}_i = \mathbf{I}, \forall i$. The missing entries are imputed to zero.
4) l1-ZF-m: In this formulation, $w_{ij} = 1, \forall i, j$ and the masks $\mathbf{S}_i$ are retained to account for missing entries, as in l1-ZF-W-m.

In addition we compare to Group subspace clustering (GSSC) and Mixture subspace clustering (MSC) [34], Matrix completion with noise (LRMC) [35], Sparse subspace clustering with missing entries (SSC-CEC) [33] and High rank matrix completion (HRMC) [26] methods for data with missing entries.

The algorithms are evaluated on the simulated dataset containing 3 clusters (termed 'Dataset-2 in the previous subsection) with $p_0 = 0.4$. We have used the signal to error ratio (SER) of the recovered cluster centres as an indicator of the goodness of the recovered cluster centers. We note that different algorithms are differentially sensitive to the regularization parameter λ. Hence, plotting the SER as a function of λ shows the best case performance of each algorithm. The SER of the estimated cluster centres are shown in Fig 9 as a function of λ. We observe from the SER plots that the proposed algorithm using the non-convex penalty estimates the cluster centres most accurately. The best SER value over all values of λ are shown in Table II for all the methods, along with classification accuracy. We also observe from the clusterpaths that the non-convex penalty of the proposed scheme results in perfect clustering, while the other SON-based techniques fail to cluster the points for any value of λ. Thus, the classification accuracy of the other SON based techniques is not reported in the table. We observe the estimated cluster centres from the constrained version of the proposed scheme is not identical to the best centre estimates obtained from the unconstrained version. However, the results are comparable. We note that in the constrained case, the cluster centres do not exactly converge to the ground-truth centres as in the unconstrained case. The unconstrained setting allows the user to view the clusterpath as a function of λ, instead of choosing the right value, which might vary between datasets. In contrast, the ϵ parameter in the constrained case might be challenging to set. We note that cluster centres are inaccurately estimated for both GSSC and MSC. However, GSSC has a high percentage of correctly clustered points, as compared to MSC. It is to be noted, that both GSSC and MSC require the number of clusters to be specified apriori, which is not required for the proposed scheme. The LRMC technique fails to cluster the points. It assigns each point to its individual cluster for low λ values and assigns all the points to a single cluster for larger λ values. When spectral clustering is applied on the co-efficient

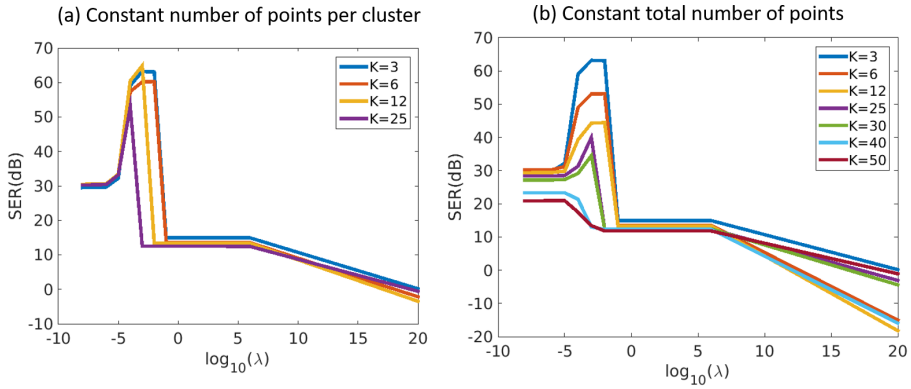


Fig. 8: Effect of changing number of clusters in simulated datasets. The proposed scheme is used to cluster datasets with varying number of clusters K. Two cases are considered: (a) Number of points per cluster is constant. In this case, the clustering performance is invariant to the number of clusters. (b) Total number of points is constant. In this case, the clustering performance degrades with increase in the number of clusters.

matrices obtained from both SSC-CEC and HRMC, we get good clustering performance and centre estimates. However, the number of clusters needs to be specified apriori.

TABLE II: Performance comparison of different methods

Method	SER (dB)	% correctly classified	Number of clusters
Prop-uncons	62.12	100	Not required
Prop-cons	46.48	100	Not required
l1-ZF-w-m	35.72	-	Not required
l1-ZF-W	37.68	-	Not required
l1-ZF	24.84	-	Not required
l1-ZF-m	16.94	-	Not required
MSC	-6.63	67.83	Required as input
GSSC	-0.93	97.50	Required as input
LRMC	30.33	-	Not required
SSC-CEC	45.2	99.17	Required as input
HRMC	-	-	Required as input

D. Clustering of Wine Dataset

We apply the clustering algorithm to the Wine dataset [27]. The data consists of the results of a chemical analysis of wines from three different cultivars. Each data point has $P = 13$ features. The three clusters have 59, 71 and 48 points respectively, resulting in a total of 178 data points. We created a dataset without outliers by retaining only $M = 40$ points per cluster, resulting in a total of 120 data points. We under-sampled these datasets using uniform random sampling with different fractions of missing entries. The results are displayed in Fig 10 using the PCA technique as explained in the previous sub-section. It is seen that the clustering is quite stable and degrades gradually with increasing fractions of missing entries. Our code and dataset for this experiment is available at: https://github.com/sunrita-poddar/Clustering-with-missing-entries.

E. Clustering of ASL Dataset

We apply the clustering algorithm to subsets of words from the Australian Sign Language high quality dataset [28]. The original dataset contained 2565 signs, each repeated 27 times by a single user over a period of 9 weeks. 22 features are measured for each sign, with an average length of 57 time frames for each feature. These features correspond to the relative positions and orientations of the fingers, measured using gloves and magnetic position trackers. We picked the most important frame for each frame, resulting in feature vector of length 22 for each word. We next formed two datasets containing subsets of words. The first dataset contained all instances of the four words "alive", "answer", "boy" and "cold". The second dataset contained all instances of the four words "alive", "boy", "change" and "love". For each dataset, the feature vectors were arranged as columns of the matrix $\mathbf{X}$. Both the datasets were of size 22×108. The datasets were undersampled uniformly at random using different fractions of missing entries. The results are displayed in Fig 11 for both datasets. It is observed that clustering the first dataset in the presence of missing entries is relatively easier, since the words are more well-separated, as is predicted by theory.

V. Discussion

We have proposed a novel algorithm to cluster data, when some of the features of the points are missing at random. We theoretically studied the performance of an algorithm that minimizes an ℓ_0 fusion penalty subject to certain constraints relating to consistency with the known features. We concluded that under favorable clustering conditions, such as well-separated clusters with low intra-cluster variance, the proposed method performs the correct clustering even in the presence of missing entries. However, since the problem is NP-hard, we propose to use relaxations of the ℓ_0 norm. We observe experimentally that the H_1 penalty is a good surrogate for the ℓ_0 norm. This non-convex saturating penalty is shown to perform better in the clustering task than previously used convex norms and penalties. We describe an IRLS based strategy to solve the relaxed problem.

Our theoretical analysis reveals the various factors that determine whether the points will be clustered correctly in the presence of missing entries. As expected, the performance

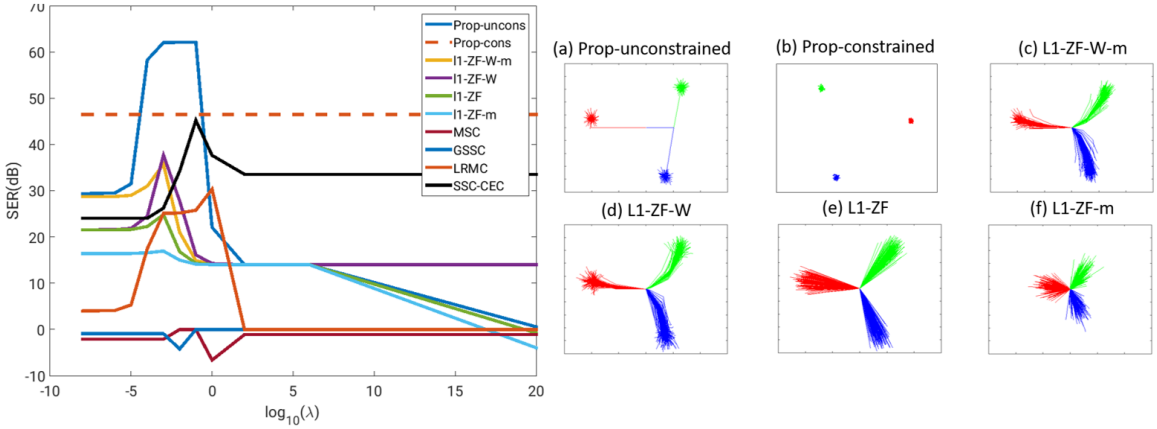


Fig. 9: Comparison with other methods. The proposed scheme is compared to several other versions of sum-of-norms clustering using weighted and unweighted convex penalties. It is observed from the clusterpaths that the convex penalties are unable to cluster the points for any value of λ. The corresponding SER values of the estimated cluster centres is also low as compared to the proposed scheme. The subspace clustering based methods (GSSC and MSC) [34] produce poor estimates of the cluster centres. However, as seen from Table II, GSSC provides good classification accuracy. LRMC [35] is unable to cluster the points. In contrast, SSC-CEC [33] and HRMC [26] provide good clustering performance and centre estimates. It is to be noted that the methods GSSC, MSC, SSC-CEC and HRMC require the number of clusters to be provided as an input.

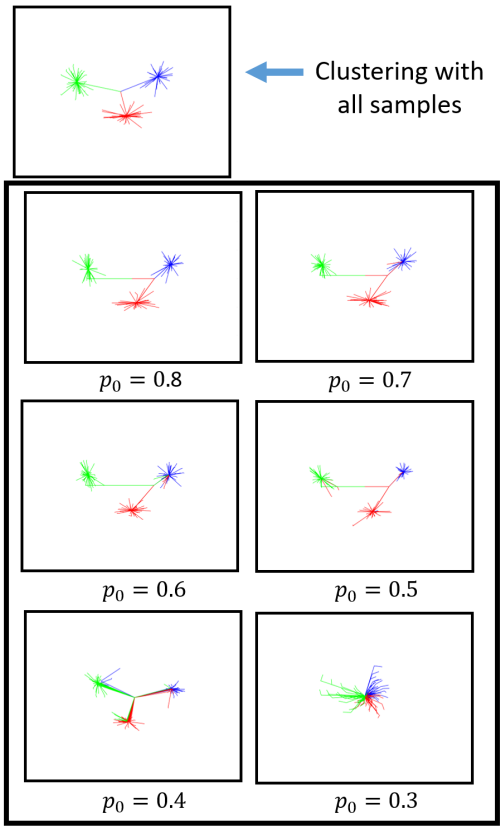


Fig. 10: Clustering on Wine dataset. The H_1 penalty is used to cluster the Wine datasets with varying fractions of missing entries. The clustering performance is accurate for around 30% missing entries.

degrades with the decrease in the fraction of sampled entries (p_0). Moreover, our results show that the difference between points from different clusters should have low coherence (μ_0) to obtain good clustering performance. This means that the expected clustering should not be dependent on only a few features of the points. Intuitively, if the points in different clusters can be distinguished by only one or two features, then a point missing these particular feature values cannot be clustered correctly. Moreover, we note that a high number of points per cluster (M), high number of features (P) and a low number of clusters (K) make the data less sensitive to missing entries.

We finally note that well-separated clusters with low intra-cluster variance (resulting in low values of κ) is desirable. While $\kappa < 1$ is assumed for our theoretical results, this is a restrictive assumption which may not be satisfied for many real datasets. Though the proposed algorithm is shown to work well on the Wine and ASL datasets, the assumption $\kappa < 1$ is violated for all these datasets. However, we observe that κ is still a good measure for the difficulty of the problem. Specifically, we observe the values $\kappa = 8.13$ for the wine dataset, and $\kappa = 3.78$ and 9.77 for the first and second ASL datasets respectively. However, as expected the first ASL dataset having a lower κ value is easier to cluster than the second one. The relaxation of this constraint to match the theory and the practical observations is a focus of our future work. We also plan to simplify the theoretical guarantees such that the relationship between p_0 and the probability of failure of the algorithm η_0 is more intuitive.

Our experimental results show great promise for the proposed technique. In particular, for the simulated data, we note that the cluster-center estimates degrade gradually with increasing fraction of missing entries. Depending on the characteristics of the data such as number of points and

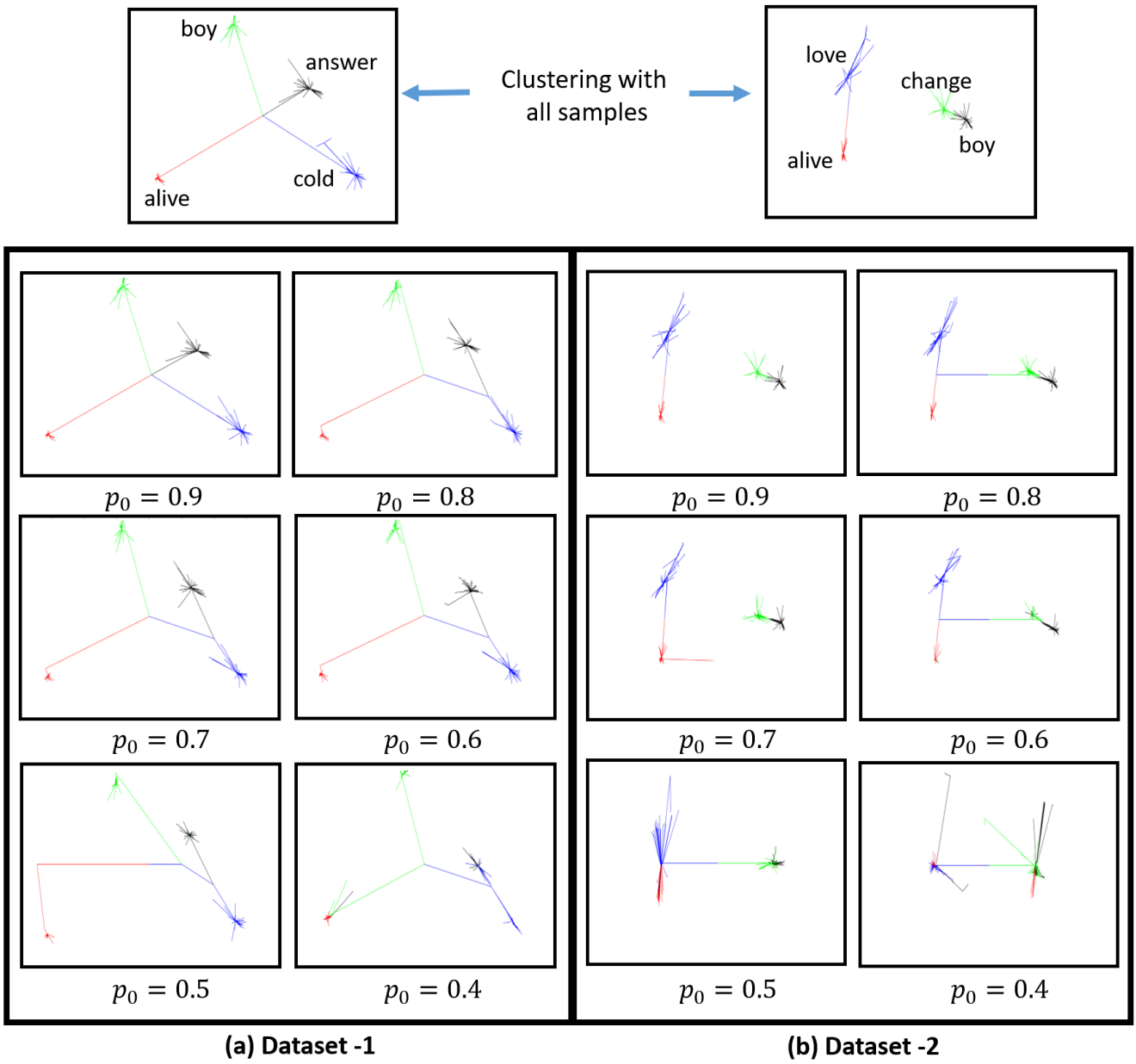


Fig. 11: Clustering on subsets of words taken from the ASL dataset. 2 datasets have been shown here, with instances of 4 words in each case. Dataset-2 is more challenging to cluster in the presence of missing entries due to greater similarity between the 4 words, as indicated by a smaller separation distance. Dataset-1 is accurately clustered even for 40% missing entries, while Dataset-2 is accurately clustered for around 20% missing entries.

cluster separation distance, the clustering algorithm fails at some particular fraction of missing entries. We also show the importance of a good initialization for the IRLS algorithm, and our proposed initialization technique using partial distances is shown to work very well. Our theory assumes well-separated clusters and without the presence of any outliers. Theoretical and experimental analysis for the clustering performance in the presence of outliers will be investigated in future work.

VI. Conclusion

We propose a clustering technique for data in the presence of missing entries. We prove theoretically that a constrained ℓ_0 norm minimization problem recovers the clustering correctly with high probability. An efficient algorithm that solves a relaxation of the above problem is presented next. It is demonstrated that the cluster center estimates obtained using the proposed algorithm degrade gradually with an increase in the number of missing entries. The algorithm is also used to cluster a Wine and an ASL dataset. The presented theory and results demonstrate the utility of the proposed scheme in clustering data in presence of missing entries.

REFERENCES

[1] A. Saxena, M. Prasad, A. Gupta, N. Bharill, O. P. Patel, A. Tiwari, M. J. Er, W. Ding, and C.-T. Lin, "A review of clustering techniques and developments," *Neurocomputing*, 2017.

[2] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[3] G. Gan, C. Ma, and J. Wu, *Data clustering: theory, algorithms, and applications*. Siam, 2007, vol. 20.

[4] A. Fahad, N. Alshatri, Z. Tari, A. Alamri, I. Khalil, A. Y. Zomaya, S. Foufou, and A. Bouras, "A survey of clustering algorithms for big data: Taxonomy and empirical analysis," *IEEE transactions on emerging topics in computing*, vol. 2, no. 3, pp. 267–279, 2014.

[5] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA., 1967, pp. 281–297.

[6] P. S. Bradley, O. L. Mangasarian, and W. N. Street, "Clustering via concave minimization," in *Advances in neural information processing systems*, 1997, pp. 368–374.

[7] A. Y. Ng, M. I. Jordan, Y. Weiss *et al.*, "On spectral clustering: Analysis and an algorithm," in *NIPS*, vol. 14, no. 2, 2001, pp. 849–856.

[8] P. Awasthi, A. S. Bandeira, M. Charikar, R. Krishnaswamy, S. Villar, and R. Ward, "Relax, no need to round: Integrality of clustering formulations," in *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*. ACM, 2015, pp. 191–200.

[9] J. H. Ward Jr, "Hierarchical grouping to optimize an objective function," *Journal of the American statistical association*, vol. 58, no. 301, pp. 236–244, 1963.

[10] T. D. Hocking, A. Joulin, F. Bach, and J.-P. Vert, "Clusterpath an algorithm for clustering using convex fusion penalties," in *28th international conference on machine learning*, 2011, p. 1.

[11] C. Zhu, H. Xu, C. Leng, and S. Yan, "Convex optimization procedure for clustering: Theoretical revisit," in *Advances in Neural Information Processing Systems*, 2014, pp. 1619–1627.

[12] M. C. De Souto, P. A. Jaskowiak, and I. G. Costa, "Impact of missing data imputation methods on gene expression clustering and classification," *BMC bioinformatics*, vol. 16, no. 1, p. 64, 2015.

[13] R. M. Bell, Y. Koren, and C. Volinsky, "The bellkor 2008 solution to the netflix prize," *Statistics Research Department at AT&T Research*, 2008.

[14] J. M. Brick and G. Kalton, "Handling missing data in survey research," *Statistical methods in medical research*, vol. 5, no. 3, pp. 215–238, 1996.

[15] K. L. Wagstaff and V. G. Laidler, "Making the most of missing values: Object clustering with partial data in astronomy," in *Astronomical Data Analysis Software and Systems XIV*, vol. 347, 2005, p. 172.

[16] K. Wagstaff, "Clustering with missing values: No imputation required," in *Classification, Clustering, and Data Mining Applications*. Springer, 2004, pp. 649–658.

[17] J. K. Dixon, "Pattern recognition with partly missing data," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 10, pp. 617–621, 1979.

[18] G. K. Chen, E. C. Chi, J. M. O. Ranola, and K. Lange, "Convex clustering: An attractive alternative to hierarchical clustering," *PLoS Comput Biol*, vol. 11, no. 5, p. e1004228, 2015.

[19] R. J. Hathaway and J. C. Bezdek, "Fuzzy c-means clustering of incomplete data," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 31, no. 5, pp. 735–744, 2001.

[20] M. Sarkar and T.-Y. Leong, "Fuzzy k-means clustering with missing values." in *Proceedings of the AMIA Symposium*. American Medical Informatics Association, 2001, p. 588.

[21] J. T. Chi, E. C. Chi, and R. G. Baraniuk, "k-pod: A method for k-means clustering of missing data," *The American Statistician*, vol. 70, no. 1, pp. 91–99, 2016.

[22] L. Hunt and M. Jorgensen, "Mixture model clustering for mixed data with missing information," *Computational Statistics & Data Analysis*, vol. 41, no. 3, pp. 429–440, 2003.

[23] T. I. Lin, J. C. Lee, and H. J. Ho, "On fast supervised learning for normal mixture models with missing information," *Pattern Recognition*, vol. 39, no. 6, pp. 1177–1187, 2006.

[24] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, p. 717, 2009.

[25] B. Eriksson, L. Balzano, and R. D. Nowak, "High-rank matrix completion and subspace clustering with missing data," *CoRR*, vol. abs/1112.5629, 2011. [Online]. Available: http://arxiv.org/abs/1112.5629

[26] E. Elhamifar, "High-rank matrix completion and clustering under self-expressive models," in *Advances in Neural Information Processing Systems*, 2016, pp. 73–81.

[27] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[28] M. W. Kadous *et al.*, *Temporal classification: Extending the classification paradigm to multivariate time series*. University of New South Wales, 2002.

[29] E. C. Chi and K. Lange, "Splitting methods for convex clustering," *Journal of Computational and Graphical Statistics*, vol. 24, no. 4, pp. 994–1013, 2015.

[30] W. Pan, X. Shen, and B. Liu, "Cluster analysis: unsupervised learning via supervised learning with a non-convex penalty." *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 1865–1889, 2013.

[31] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *Acoustics, speech and signal processing, 2008. ICASSP 2008. IEEE international conference on*. IEEE, 2008, pp. 3869–3872.

[32] A. H. Delaney and Y. Bresler, "Globally convergent edge-preserving regularized reconstruction: an application to limited-angle tomography," *IEEE Transactions on Image Processing*, vol. 7, no. 2, pp. 204–221, 1998.

[33] C. Yang, D. Robinson, and R. Vidal, "Sparse subspace clustering with missing entries," in *International Conference on Machine Learning*, 2015, pp. 2463–2472.

[34] D. Pimentel-Alarcón, L. Balzano, R. Marcia, R. Nowak, and R. Willett, "Group-sparse subspace clustering with missing data," in *2016 IEEE Statistical Signal Processing Workshop (SSP)*. IEEE, 2016, pp. 1–5.

[35] E. J. Candes and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 925–936, 2010.

[36] D. W. Matula, *The largest clique size in a random graph*. Department of Computer Science, Southern Methodist University, 1976.

APPENDIX A
PROOF OF LEMMA II.1

Proof. Since $\mathbf{x}_1$ and $\mathbf{x}_2$ are in the same cluster, $\|\mathbf{x}_1 - \mathbf{x}_2\|_\infty \leq \epsilon$. For all the points in this particular cluster, let the p^{th} feature be bounded as: $f^p_{min} \leq \mathbf{x}(p) \leq f^p_{max}$. Then we can construct a vector $\mathbf{u}$, such that $\mathbf{u}(p) = \frac{1}{2}(f^p_{min} + f^p_{max})$. Now, since $f^p_{max} - f^p_{min} \leq \epsilon$, the following condition will be satisfied for this particular choice of $\mathbf{u}$:

$$\|\mathbf{x}_i - \mathbf{u}\|_\infty \quad \leq \quad \frac{\epsilon}{2}; \quad i = 1, 2 \tag{28}$$

From this, it follows trivially that the following will also hold:

$$\|\mathbf{S}_i\left(\mathbf{x}_i - \mathbf{u}\right)\|_\infty \quad \leq \quad \frac{\epsilon}{2}; \quad i = 1, 2 \tag{29}$$

□

APPENDIX B
LEMMA B.1

Lemma B.1. *Consider any pair of points $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^P$ observed by sampling matrices $\mathbf{S}_1 = \mathbf{S}_{\mathcal{I}_1}$ and $\mathbf{S}_2 = \mathbf{S}_{\mathcal{I}_2}$, respectively. We assume the set of common indices ($\omega := \mathcal{I}_1 \cap \mathcal{I}_2$) to be of size $q = |\mathcal{I}_1 \cap \mathcal{I}_2|$. Then, for some $0 < t < \frac{q}{P}$, the following result holds true regarding the partial distance $\|\mathbf{y}_\omega\|_2 = \|\mathbf{S}_{\mathcal{I}_1 \cap \mathcal{I}_2}\left(\mathbf{x}_1 - \mathbf{x}_2\right)\|_2$:*

$$\mathbb{P}\left(\|\mathbf{y}_\omega\|_2^2 \leq \left(\frac{q}{P} - t\right)\|\mathbf{y}\|_2^2\right) \leq e^{-\frac{2t^2P^2}{q\mu_0^2}} \tag{30}$$

where $\mathbf{y} = \mathbf{x}_1 - \mathbf{x}_2$.

Proof. We use some ideas for bounding partial distances from Lemma 3 of [25]. Let $\mathbf{y} = \mathbf{x}_1 - \mathbf{x}_2$. We rewrite the partial distance $\|\mathbf{y}_\omega\|_2^2$ as the sum of q variables drawn uniformly at random from $\{y_1^2, y_2^2, \ldots, y_P^2\}$. By replacing a particular

variable in the summation by another one, the value of the sum changes by at most $\|\mathbf{y}\|_\infty^2$. Applying McDiarmid's Inequality, we get:

$$\mathbb{P}\left(E(\|\mathbf{y}_\omega\|_2^2) - \|\mathbf{y}_\omega\|_2^2 \geq c\right) \leq e^{-\frac{2c^2}{\sum_{i=1}^q \|\mathbf{y}\|_\infty^4}} = e^{-\frac{2c^2}{q\|\mathbf{y}\|_\infty^4}} \tag{31}$$

From our assumptions, we have $E(\|\mathbf{y}_\omega\|_2^2) = \frac{q}{P}\|\mathbf{y}\|_2^2$. We also have $\frac{\|\mathbf{y}\|_2^2}{\|\mathbf{y}\|_\infty^2} \geq \frac{P}{\mu_0}$ by (6). We now substitute $c = t\|\mathbf{y}\|_2^2$, where $0 < t < \frac{q}{P}$. Using the results above, we simplify (31):

$$\mathbb{P}\left(\|\mathbf{y}_\omega\|_2^2 \leq \left(\frac{q}{P} - t\right)\|\mathbf{y}\|_2^2\right) \leq e^{-\frac{2t^2\|\mathbf{y}\|_2^4}{q\|\mathbf{y}\|_\infty^4}} \leq e^{-\frac{2t^2P^2}{q\mu_0^2}} \tag{32}$$

□

Appendix C
Proof of Lemma II.2

Proof. We will use proof by contradiction. Specifically, we consider two points $\mathbf{x}_1$ and $\mathbf{x}_2$ belonging to different clusters and assume that there exists a point $\mathbf{u}$ that satisfies:

$$\|\mathbf{S}_i(\mathbf{x}_i - \mathbf{u})\|_\infty \quad \leq \quad \frac{\epsilon}{2}; i = 1, 2 \tag{33}$$

We now show that the above assumption is violated with high probability. Following Lemma B.1, we denote $\mathbf{y} = \mathbf{x}_1 - \mathbf{x}_2$ and the partial distance is:

$$\|\mathbf{y}_\omega\|_2 = \|\mathbf{S}_{\mathcal{I}_1 \cap \mathcal{I}_2}\ (\mathbf{x}_1 - \mathbf{x}_2)\|_2 \tag{34}$$

Using (33) and applying triangle inequality, we obtain $\|\mathbf{y}_\omega\|_\infty \leq \epsilon$, such that $\|\mathbf{y}_\omega\|_2 \leq \epsilon\sqrt{q}$, where $q = |\mathcal{I}_1 \cap \mathcal{I}_2|$ is the number of commonly observed locations. We will show that with high probability, the partial distances satisfy:

$$\|\mathbf{y}_\omega\|_2^2 > \epsilon^2 q \tag{35}$$

which contradicts (33). We first find a lower bound for q. Using the Chernoff bound and setting $\mathbb{E}(q) = p_0^2 P$, we have:

$$\mathbb{P}\left(q \geq \frac{p_0^2 P}{2}\right) > 1 - \gamma_0 \tag{36}$$

where $\gamma_0 = (\frac{e}{2})^{-\frac{p_0^2 P}{2}}$. Using Lemma B.1, we have the following result for the partial distances:

$$\mathbb{P}\left(\|\mathbf{y}_\omega\|_2^2 \leq \left(\frac{q}{P} - t\right)\|\mathbf{y}\|_2^2\right) \leq e^{-\frac{2t^2P^2}{q\mu_0^2}} \tag{37}$$

Since $\mathbf{x}_1$ and $\mathbf{x}_2$ are in different clusters, we have $\|\mathbf{y}\|_2 \geq \delta$. We will now determine the value of t for which the above upper bound will equal the RHS of (35):

$$\left(\frac{q}{P} - t\right)\|\mathbf{y}\|_2^2 = \epsilon^2 q \tag{38}$$

or equivalently:

$$t = \frac{q}{P} - \frac{\epsilon^2 q}{\|\mathbf{y}\|_2^2} \geq \frac{q}{P} - \frac{\epsilon^2 q}{\delta^2} = \frac{q}{P}(1 - \kappa^2) \tag{39}$$

Since $t > 0$, we require $\kappa < 1$, where $\kappa = \frac{\epsilon\sqrt{P}}{\delta}$. Using the above, we get the following bound if we assume that $q \geq \frac{p_0^2 P}{2}$:

$$\frac{t^2}{q} \geq \frac{q}{P^2}(1 - \kappa^2)^2 \geq \frac{p_0^2}{2P}(1 - \kappa^2)^2 \tag{40}$$

We obtain the following bound for any $q \geq \frac{p_0^2 P}{2}$:

$$\begin{aligned}\mathbb{P}\left(\|\mathbf{y}_\omega\|^2 > \epsilon^2 q\right) &\geq 1 - e^{-\frac{2t^2P^2}{q\mu_0^2}} \\ &\geq 1 - e^{-\frac{p_0^2 P(1-\kappa^2)^2}{\mu_0^2}} \\ &= 1 - \delta_0\end{aligned} \tag{41}$$

Combining (36) and (41), the probability for (33) to hold is $\leq 1 - (1 - \gamma_0)(1 - \delta_0) = \beta_0$.

□

Appendix D
Proof of Lemma II.3

Proof. We construct a graph where each point $\mathbf{x}_i$ is represented by a node. Lemma II.1 implies that a pair of points belonging to the same cluster can yield the same $\mathbf{u}$ in a feasible solution with probability 1. Hence, we will assume that there exists an edge between two nodes from the same cluster with probability 1. Lemma II.2 indicates that a pair of points belonging to different clusters can yield the same $\mathbf{u}$ in a feasible solution with a low probability of β_0. We will assume that there exists an edge between two nodes from different clusters with probability β_0. We will now evaluate the probability that there exists a fully-connected sub-graph of size M, where all the nodes have not been taken from the same cluster. We will follow a methodology similar to [36], which gives an expression for the probability distribution of the maximal clique (i.e. largest fully connected sub-graph) size in a random graph. Unlike the proof in [36], in our graph every edge is not present with equal probability.

We define the following random variables:

- $t \coloneqq$ Size of the largest fully connected sub-graph containing nodes from more than 1 cluster
- $n \coloneqq$ Number of M membered complete sub-graphs containing nodes from more than 1 cluster

Our graph can have an M membered clique iff n is non-zero. Thus, we have:

$$\mathbb{P}(t \geq M) = \mathbb{P}(n \neq 0) \tag{42}$$

Since the distribution of n is restricted only to the non-negative integers, it can be seen that:

$$\mathbb{P}(n \neq 0) \leq E(n) \tag{43}$$

Combining the above 2 results, we get:

$$\mathbb{P}(t \geq M) \leq E(n) \tag{44}$$

Let us consider the formation of a particular clique of size M using $m_1, m_2, \ldots, m_K$ nodes from clusters $C_1, C_2, \ldots, C_K$ respectively such that $\sum_{j=1}^K m_j = M$, and at least 2 of the variables $\{m_j\}$ are non-zero. The number of ways to choose such a collection of nodes is: $\prod_j \binom{M}{m_j}$. In order to form a solution $\{m_j\}$, we need $\frac{1}{2}(M^2 - \sum_j m_j^2)$ inter-cluster edges to be present. We recall that each of these edges is present with probability β_0. Thus, the probability that

such a collection of nodes forms a clique is $\beta_0^{\frac{1}{2}(M^2-\sum_j m_j^2)}$. This gives the following result:

$$E(N) = \sum_{\{m_j\}\in\mathcal{S}} \beta_0^{\frac{1}{2}(M^2-\sum_j m_j^2)} \prod_j \binom{M}{m_j} = \eta_0 \tag{45}$$

where $\mathcal{S}$ is the set of all sets of positive integers $\{m_j\}$ such that: $2 \leq \mathcal{U}(\{m_j\}) \leq K$ and $\sum_j m_j = M$. Here, the function $\mathcal{U}$ counts the number of non-zero elements in a set. Thus:

$$\mathbb{P}\left(t \geq M\right) \leq \eta_0 \tag{46}$$

This proves that with probability $\geq 1-\eta_0$, a set of points of cardinality $\geq M$ not all belonging to the same cluster cannot all have equal cluster-center estimates.

□

Appendix E
Proof of Theorem II.4

Proof. Lemma II.1 indicates that fully connected original clusters with size M can always be formed, while Lemma II.3 shows that the size of misclassified large clusters cannot exceed $M-1$ with very high probability. These results enable us to re-express the optimization problem (8) as a simpler maximization problem. We will then show that with high probability, any feasible solution other than the ground-truth solution results in a cost higher than the ground-truth solution.

Let a candidate solution have k groups of sizes $M_1, M_2, \ldots, M_k$ respectively. The center estimates for all points within a group are equal. These are different from the center estimates of other groups. Without loss of generality, we will assume that at most K of these groups each have points belonging to only a single ground-truth cluster, i.e. they are "pure". The rest of the clusters in the candidate solution are "mixed" clusters. If we have a candidate solution with greater than K pure clusters, they can always be merged to form K pure clusters; the merged solution will result in a lower cost.

The objective function in (8) can thus be rewritten as:

$$\begin{aligned}\sum_{i=1}^{KM}\sum_{j=1}^{KM} \|\mathbf{u}_i - \mathbf{u}_j\|_{2,0} &= \sum_{i=1}^{k} M_i(KM - M_i) \\ &= K^2M^2 - \sum_{i=1}^{k} M_i^2\end{aligned} \tag{47}$$

Since we assume that the first K clusters are pure, therefore they have a size $0 \leq M_i \leq M$, $i = 1, \ldots, K$. The remaining clusters are mixed and have size $\leq M-1$ with probability $\geq 1-\eta_0$. Hence, we have the constraints $0 \leq M_i \leq (M-1)$, $i = K+1, \ldots, k$. We also have a constraint on the total number of points, i.e. $\sum_{i=1}^{k} M_i = KM$. Thus, the problem (8) can be rewritten as the constrained optimization problem:

$$\begin{aligned}\{M_i^*, k^*\} = \max_{\{M_i\},k} & \sum_{i=1}^{k} M_i^2 \\ \text{s.t. } & 0 \leq M_i \leq M, i = 1, \ldots, K \\ & 0 \leq M_i \leq M-1, i = K+1, \ldots, k \\ & \sum_{i=1}^{k} M_i = KM\end{aligned} \tag{48}$$

Note that we cannot have $k < K$, with probability $\geq 1-\eta_0$, since that involves a solution with cluster size $> M$. We can evaluate the best solution $\{M_i^*\}$ for each possible value of k in the range $K \leq k \leq MK$. Then we can compare these solutions to get the solution with the highest cost. We note that the feasible region is a polyhedron and the objective function is convex. Thus, for each value of k, we only need to check the cost at the vertices of the polyhedron formed by the constraints, since the cost at all other points in the feasible region will be lower. The vertex points are formed by picking $k-1$ out of the k box constraints and setting M_i to be equal to one of the 2 possible extremal values. We note that all the vertex points have either K or $K+1$ non-zero values. As a simple example, if we choose $M = 10$ and $K = 4$, then the vertex points of the polyhedron (corresponding to different solutions $\{M_i\}$) are given by all permutations of:

- $(10, 10, 10, 10, 0, 0 \ldots 0)$: 4 clusters
- $(10, 10, 10, 0, 1, 9, 0 \ldots 0)$: 5 clusters
- $(10, 10, 0, 0, 2, 9, 9, 0 \ldots 0)$: 5 clusters
- $(10, 0, 0, 0, 3, 9, 9, 9, 0 \ldots 0)$: 5 clusters
- $(0, 0, 0, 0, 4, 9, 9, 9, 9, 0 \ldots 0)$: 5 clusters

In general, the vertices are given by permutations of:

- $(M, M, \ldots, M, 0, 0 \ldots 0)$: K clusters
- $(M, M, \ldots, 0, 0, 1, M-1, 0 \ldots 0)$: $K+1$ clusters
- $(M, M, \ldots, 0, 0, 2, M-1, M-1 \ldots 0)$: $K+1$ clusters
- $\ldots$
- $(0, 0, \ldots 0, K, M-1, M-1 \ldots M-1, 0)$: $K+1$ clusters

Now, it is easily checked that the 1^{st} candidate solution in the list (which is also the ground-truth solution) has the maximum cost. Mixed clusters with size $> M-1$ cannot be formed with probability $> 1-\eta_0$. Thus, with the same probability, the solution to the optimization problem (8) is identical to the ground-truth clustering. □

Appendix F
Upper Bound for η_0 in the 2-cluster case

Proof. We introduce the following notation:

1) $F(i) = i(M-i)\log\beta_0$, for $i \in [1, M-1]$.
2) $G(i) = 2[\log\Gamma(M+1) - \log\Gamma(i+1) - \log\Gamma(M-i+1)]$, for $i \in [1, M-1]$ where Γ is the Gamma function.

We note that both the functions F and G are symmetric about $i = \frac{M}{2}$, and have unique minimum and maximum respectively for $i = \frac{M}{2}$. We will show that the maximum for the function $F+G$ is achieved at the points $i = 1, M-1$. We note that:

$$G'(i) = -2[\Psi(i+1) - \Psi(M-i+1)] \tag{49}$$

where Ψ is the digamma function, defined as the log derivative of the Γ function. We now use the expansion:

$$\Psi(i+1) = \log i + \frac{1}{2i} \tag{50}$$

Substituting, we get:

$$G'(i) = -2\left[\log\frac{i}{M-i} + \frac{M-2i}{2i(M-i)}\right] \tag{51}$$

Thus, we have:

$$F'(i)+G'(i)=(M-2i)(\log\beta_0-\frac{1}{i(M-i)}) \\ -2\log\frac{i}{(M-i)}) \tag{52}$$

Now, in order to ensure that $F'(i)+G'(i)\leq 0$, we have to arrive at conditions such that:

$$\log\beta_0\leq\frac{1}{i(M-i)}+\frac{2}{M-2i}\log\frac{i}{M-i} \tag{53}$$

Since the RHS is monotonically increasing in the interval $i\in[1,\frac{M}{2}-1]$ the above condition reduces to:

$$\log\beta_0\leq\frac{1}{M-1}+\frac{2}{M-2}\log\frac{1}{M-1} \tag{54}$$

Under the above condition, for all $i\in[1,\frac{M}{2}]$:

$$F'(i)+G'(i)\leq 0 \tag{55}$$

Thus, the function $F+G$ reaches its maxima at the extremal points given by $i=1,M-1$. For $i\in\{1,2,\ldots,M-1\}$:

$$F(i)+G(i)=\log[\beta_0^{i(M-i)}\binom{M}{i}^2] \tag{56}$$

Thus, the function $\beta_0^{i(M-i)}\binom{M}{i}^2$ also reaches its maxima at $i=1,M-1$. This maximum value is given by: $\beta_0^{M-1}M^2$. This gives the following upper bound for η_0:

$$\eta_0\leq\sum_{i=1}^{M-1}[\beta_0^{M-1}M^2]\leq M^3\beta_0^{M-1}=\eta_{0,\text{approx}} \tag{57}$$

□

Appendix G
Proof of Theorem II.5

Proof. We consider any two points $\mathbf{x}_1$ and $\mathbf{x}_2$ that are in different clusters. Let us assume that there exists some $\mathbf{u}$ satisfying the data consistency constraint:

$$\|\mathbf{x}_i-\mathbf{u}\|_\infty\leq\epsilon/2,\quad i=1,2. \tag{58}$$

Using the triangle inequality, we have $\|\mathbf{x}_1-\mathbf{x}_2\|_\infty\leq\epsilon$ and consequently, $\|\mathbf{x}_1-\mathbf{x}_2\|_2\leq\epsilon\sqrt{P}$. However, if we have a large inter-cluster separation $\delta>\epsilon\sqrt{P}$, then this is not possible.

Thus, if $\delta>\epsilon\sqrt{P}$, then points in different clusters cannot be misclassified to a single cluster. Among all feasible solutions, clearly the solution with the minimum cost is the one where all points in the same cluster merge to the same $\mathbf{u}$. Thus, $\kappa<1$ ensures that we will have the correct clustering. □

Appendix H
Proof of theorem III.1

Proof. The proof follows from theorem 1 in [32]. The following optimization problem is proposed:

$$\min_{\mathbf{U}}Q(\mathbf{U})+\lambda\sum_{m=0}^{N-1}\psi[V_m(\mathbf{U})] \tag{59}$$

where $Q,V_m:\mathbb{R}^P\to[0,\infty)$, for $m=0,1,\ldots,N-1$ are continuously differentiable convex functionals, and $\psi:[0,\infty)\to[0,\infty)$ is a continuously twice differentiable concave function, with $\psi(0)=0,\psi^{'}(0)=1$ and $0<\psi^{'}(t)\leq 1$. We now consider the following:

1) $Q(\mathbf{U})=\sum_i\|\mathbf{S}_i(\mathbf{u}_i-\mathbf{x}_i)\|^2$
2) $V_m(\mathbf{U})=\|\mathbf{u}_i-\mathbf{u}_j\|^2$, for $m=(i-1)KM+j-1$ and $N=(KM)^2-1$

If we choose the functions ψ such that the required conditions are satisfied, then according to the theorem,$\{\mathbf{U}^{(n)}\}$ defined in (24) and (25) converges to a stationary point of (22), or the accumulation points of $\{\mathbf{U}^{(n)}\}$ form a continuum in the set of stationary points of (22).

□