# Sparse Subspace Clustering with Missing Entries

**Congyuan Yang** YANGCY@JHU.EDU
**Daniel Robinson** DANIEL.P.ROBINSON@JHU.EDU
**René Vidal** RVIDAL@CIS.JHU.EDU
Johns Hopkins University, 3400 N Charles St., Baltimore, MD, USA

## Abstract

We consider the problem of clustering incomplete data drawn from a union of subspaces. Classical subspace clustering methods are not applicable to this problem because the data are incomplete, while classical low-rank matrix completion methods may not be applicable because data in multiple subspaces may not be low rank. This paper proposes and evaluates two new approaches for subspace clustering and completion. The first one generalizes the sparse subspace clustering algorithm so that it can obtain a sparse representation of the data using only the observed entries. The second one estimates a suitable kernel matrix by assuming a random model for the missing entries and obtains the sparse representation from this kernel. Experiments on synthetic and real data show the advantages and disadvantages of the proposed methods, which all outperform the natural approach (low-rank matrix completion followed by sparse subspace clustering) when the data matrix is high-rank or the percentage of missing entries is large.

## 1. Introduction

In many real world applications, we are faced with the problem of clustering incomplete data drawn from a union of low-dimensional subspaces. For example, in the motion segmentation problem in computer vision, feature point trajectories extracted from a video sequence with multiple moving objects lie in multiple low-dimensional subspaces, one per moving object. However, due to occlusions or objects entering or leaving the field of view, such trajectories are often incomplete. Therefore, we are faced with the problem of clustering these incomplete trajectories in order to segment the video into multiple motions.

**Prior work.** The problem of clustering complete data in a union of subspaces has received increasing attention over the past decade (see Vidal (2011) for a tutorial). Classical methods, such as K-subspaces (Bradley & Mangasarian, 2000; Tseng, 2000) and mixture of probabilistic PCAs (Tipping & Bishop, 1999) suffer from local minima, while algebraic methods such as Generalized Principal Component Analysis (Vidal et al., 2005) suffer from robustness to data corruptions. This has motivated the development of convex optimization methods based on sparse (Elhamifar & Vidal, 2009; 2010; 2013) and low-rank (Liu et al., 2010; 2013; Lu et al., 2012; Favaro et al., 2011; Vidal & Favaro, 2014) representation techniques. For example, the Sparse Subspace Clustering (SSC) algorithm of (Elhamifar & Vidal, 2009) is based on expressing each data point as a sparse linear combination of all other data points. When the subspaces are sufficiently separated, and the data points are sufficiently well distributed inside the subspaces, the nonzero coefficients of one point correspond to other points in the same subspace. Hence the sparse representation can be used to construct an affinity for clustering the data using spectral clustering. The theoretical conditions guarantee the correctness of clustering and are applicable to noiseless data (Elhamifar & Vidal, 2013), data corrupted by outliers (Soltanolkotabi & Candès, 2013; Soltanolkotabi et al., 2014) and data corrupted by noise (Wang & Xu, 2013).

In sharp contrast, the case where the data points are incomplete has received significantly less attention. Gruber & Weiss (2004) model the data with a mixture of probabilistic PCAs and use the EM algorithm to both segment the data and find the missing entries. However, this approach suffers from local minima and cannot guarantee exact matrix completion or exact clustering. Vidal & Hartley (2004) assume that, although the data lie in a union of subspaces, the data matrix is still low-rank, hence they use low-rank matrix completion techniques followed by subspace clustering techniques. In practice, the number of subspaces and their dimensions may not be small enough so that the data matrix is low rank. Eriksson et al. (2012) use a local neighborhood of each incomplete point to complete it, and refine the estimated subspaces to recover the full matrix. Under certain

*Proceedings of the $32^{nd}$ International Conference on Machine Learning*, Lille, France, 2015. JMLR: W&CP volume 37. Copyright 2015 by the author(s).

conditions, this method can complete the data with high accuracy. However, the conditions require having an arbitrarily large number of points, which makes it impractical. Finally, in an unpublished abstract, Candes et al. (2014) propose to extend SSC to the case of missing data by replacing a certain incomplete kernel matrix by its expected value and then computing the sparse representation of each point using a bias corrected Dantzig selector. While the first step is very clever and promising, the precise incomplete data model and the computation of the expectation are not given. Moreover, using a bias corrected Dantzig selector for finding the sparse representation is computationally more complex than classical SSC, which is based on an ADMM implementation of LASSO.

**Paper contributions.** We present two new approaches for subspace clustering and completion. The first one uses the knowledge of which entries are observed to formulate a convex optimization problem that estimates a sparse representation of the data. The second one is based on the observation that the LASSO version of SSC does not require one to know the full data matrix $X$, but rather the kernel matrix $X^\top X$. Under a certain probability model for the missing entries, we show that it is possible to obtain an unbiased estimator for $X^\top X$ from the observed entries of $X$ and the fraction of missing entries $\delta$. (This approach was outlined in (Candes et al., 2014) without proof.) The estimator is then used to define a convex optimization problem for computing a sparse representation of the data. The last step of both approaches is the same as that of SSC: apply spectral clustering to an affinity matrix built from the sparse representation. Experiments on synthetic and real data show that the proposed algorithms are effective with each having advantages and disadvantages, as well as complementary strengths when compared to the basic approach of matrix completion followed by SSC. In particular, when $X$ is low-rank and the fraction of missing entries $\delta$ is small, matrix completion works extremely well and our approaches are inferior. However, when either $X$ is high rank or $\delta$ is large, matrix completion fails, and our approaches are superior.

## 2. SSC with Complete Data: A Review

Consider $n$ linear or affine subspaces $\{S_l \subset \mathbb{R}^D\}_{l=1}^n$, each of dimension $d_l < D$ for $l = 1, \dots, n$. Assume we are given $N$ data points $\{\boldsymbol{x}_j \in \mathbb{R}^D\}_{j=1}^N$ lying in the union of the $n$ subspaces. Then the observed data matrix is

$$X = [\boldsymbol{x}_1 \dots \boldsymbol{x}_N] \in \mathbb{R}^{D\times N}. \quad (1)$$

The goal of subspace clustering is to identify the number of subspaces, their dimensions, a basis for each subspace, and the membership of each data point to its correct subspace.

The SSC algorithm (Elhamifar & Vidal, 2009) is based on the observation that data in a union of subspaces are *self-expressive*. That is, each data point can be represented as a linear combination of other points that lie in the same subspace. Therefore, we can write each data point as

$$\boldsymbol{x}_j = X\boldsymbol{c}_j,\ c_{jj} = 0,\ (\mathbf{1}^\top \boldsymbol{c}_j = 1), \quad (2)$$

where the vector of coefficients $\boldsymbol{c}_j$ has at most $d_l$ nonzero entries if $\boldsymbol{x}_j \in S_l$. The additional constraint $\mathbf{1}^\top \boldsymbol{c}_j = 1$, where $\mathbf{1}$ is the vector of all ones, is used in the case of affine subspaces, because the coefficients must add up to 1 to give an affine combination rather than a linear combination.

When the subspaces are sufficiently separated and the data points are well distributed inside each subspace, the theoretical analysis in (Elhamifar & Vidal, 2013; Soltanolkotabi & Candès, 2013; Soltanolkotabi et al., 2014) shows that the solutions of the set of $\ell_1$ minimization problems

$$\min_{\boldsymbol{c}_j} ||\boldsymbol{c}_j||_1 \ \text{s.t.}\ \boldsymbol{x}_j = X\boldsymbol{c}_j,\ c_{jj} = 0,\ j = 1, \dots, N, \quad (3)$$

satisfy $c_{ij} \neq 0$ only if points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are in the same subspace. In matrix form, we can write this problem as

$$\min_C ||C||_1 \ \text{ s.t. }\ X = XC,\ \operatorname{diag}(C) = 0, \quad (4)$$

where $C = [\boldsymbol{c}_1 \cdots \boldsymbol{c}_N] \in \mathbb{R}^{N\times N}$ is the coefficient matrix.

When the data are contaminated by noise, the self-expressiveness constraint $X = XC$ is relaxed and the following LASSO problem is solved:

$$\min_C ||C||_1 + \frac{\lambda}{2}\|X - XC\|_F^2 \ \text{ s.t. }\ \operatorname{diag}(C) = 0, \quad (5)$$

where $\lambda > 0$ is a parameter. In this case, Wang & Xu (2013) show that when the amount of noise is small enough, the subspaces are sufficiently separated, and the data are well distributed, the matrix of coefficients gives the correct clustering with high probability.

Given the matrix of sparse coefficients $C$, SSC constructs a weighted graph with affinity matrix $\mathcal{A} = |C| + |C|^\top$ and uses spectral clustering algorithms to cluster the data.

## 3. SSC with Missing Entries

Let us now consider the case where some entries of the data matrix $X = [x_{ij}] \in \mathbb{R}^{D\times N}$ are missing. Specifically, let $W = [w_{ij}] \in \{0, 1\}^{D\times N}$ be a matrix such that $w_{ij} = 1$ if $x_{ij}$ is observed, and $w_{ij} = 0$ otherwise. The locations of the observed entries for the $j$th data point or for the entire data matrix can then be indexed, respectively, by the sets:

$$\Omega_j = \{i : w_{ij} = 1\} \quad \text{and} \quad \Omega = \{(i, j) : w_{ij} = 1\}. \quad (6)$$

Given the observed entries of $X$, $\{x_{ij}\}_{(i,j)\in\Omega}$, our goal is to determine which columns of $X$ belong to the same subspace. SSC does so by solving for the matrix of coefficients $C$ in (5). However, since some entries of $X$ are missing, we cannot directly solve the optimization problem in (5). In this section, we present various approaches for addressing this problem and discuss their strengths and weaknesses.

### 3.1. Matrix Completion + SSC (MC+SSC)

A first approach is to apply a matrix completion (MC) algorithm to recover the missing entries in $X$ and then solve (5). More specifically, let $\mathcal{P}_\Omega(X)$ denote the entries of $X$ in $\Omega$. The *MC+SSC* approach to subspace clustering with missing entries uses the MC approach in (Cai et al., 2008) followed by the SSC approach in (5), that is:

$$A = \arg\min_A \|A\|_* + \frac{\tau}{2}\|A\|_F^2 \ \text{ s. t. } \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(A),$$

$$C = \arg\min_C \|C\|_1 + \frac{\lambda}{2}\|A - AC\|_F^2 \ \text{ s. t. } \operatorname{diag}(C) = 0,$$

for appropriately chosen positive constants $\tau$ and $\lambda$. The MC+SSC approach is likely to be a good strategy when the rank of the data matrix and the percentage of missing entries are sufficiently small. In this case, existing results (Cai et al., 2008; Candès & Recht, 2009; Candès & Tao, 2010; Gross, 2011; Keshavan et al., 2010; Zhou et al., 2010; Recht, 2011) guarantee the correctness of completion, while existing results (Elhamifar & Vidal, 2013; Soltanolkotabi & Candès, 2013; Soltanolkotabi et al., 2014) guarantee the correctness for clustering. However, MC+SSC is likely to fail as soon as the data matrix is full rank, e.g., a data matrix from 20 subspaces of dimension 5 in $\mathbb{R}^{100}$ can be of full rank 100, or the percentage of missing entries is high, as in both cases the MC step may fail.

### 3.2. SSC by Entry-Wise Zero-Fill (SSC-EWZF)

A second approach is to fill the missing entries in $X$ with 0s, i.e., to replace $X$ by $X_{\text{miss}} = W \odot X$ where $\odot$ is the Hadamard product, and then solve (5). We call this approach *Zero-Fill+SSC (ZF+SSC)*. However, this heuristic may fail because it may not provide the correct completion of the data, hence SSC may not provide the correct clustering either. Moreover, the ZF approach does not minimize the correct error. Specifically, the self-expressiveness error

$$\|X - XC\|_F^2 = \sum_{i=1}^{D}\sum_{j=1}^{N}(x_{ij} - \sum_{k=1}^{N} x_{ik}c_{kj})^2 \tag{7}$$

becomes

$$\|X_{\text{miss}} - X_{\text{miss}}C\|_F^2 = \sum_{i=1}^{D}\sum_{j=1}^{N}(w_{ij}x_{ij} - \sum_{k=1}^{N} w_{ik}x_{ik}c_{kj})^2. \tag{8}$$

When $w_{ij} = 0$, this encourages the term $\sum_{k=1}^{N} w_{ik}x_{ik}c_{kj}$ to be close to zero, while this term should not be penalized because we do not observe $x_{ij}$ when $w_{ij} = 0$. We could address this by summing only over observed entries of $X$:

$$\|\mathcal{P}_\Omega(X - XC)\|_F^2 = \sum_{i=1}^{D}\sum_{j=1}^{N} w_{ij}\left(x_{ij} - \sum_{k=1}^{N} x_{ik}c_{kj}\right)^2. \tag{9}$$

However, $\mathcal{P}_\Omega(XC)$ cannot be computed when $X$ has missing entries. To address this issue, we propose to replace (9) by $\|\mathcal{P}_\Omega(X_{\text{miss}} - X_{\text{miss}}C)\|_F^2$, which is equal to:

$$\sum_{i=1}^{D}\sum_{j=1}^{N} w_{ij}(x_{ij} - \sum_{k=1}^{N} w_{ik}x_{ik}c_{kj})^2. \tag{10}$$

This modified self-expressiveness error was proposed in (Balzano et al., 2010) for a different but related problem (column subset selection with missing entries). In principle, this modified error is not correct either since it replaces the linear combination $\sum_{k=1}^{N} x_{ik}c_{kj}$ by $\sum_{k=1}^{N} w_{ik}x_{ik}c_{kj}$. However, notice that the two sums coincide if for all $k$ such that $w_{ik} = 0$ it happens to be the case that $c_{kj} = 0$. Since the $\ell_1$ term will bias $c_{kj}$ to be zero by penalizing $|c_{kj}|$, the *SSC by Entry-Wise Zero-Fill (SSC-EWZF)* approach,

$$\min_C \|C\|_1 + \frac{\lambda}{2}\|\mathcal{P}_\Omega(X_{\text{miss}} - X_{\text{miss}}C)\|_F^2 \ \text{ s. t. } \operatorname{diag}(C) = 0, \tag{11}$$

will effectively try to express the $j$th column of $X$, $\boldsymbol{x}_j$, as a linear combination of other columns of $X$ that are in the same subspace as $\boldsymbol{x}_j$ and whose entries in $\Omega_j$ are the most complete. Ideally, this approach will work perfectly if, for each point $\boldsymbol{x}_j \in S_l$, there are at least $d_l$ other data points $\boldsymbol{x}_{j_1}, \ldots, \boldsymbol{x}_{j_{d_l}}$, whose $i$th entries are known for all $i \in \Omega_j$. However, the use of the projection $\mathcal{P}_\Omega$ could affect the correctness of clustering, e.g., if two different subspaces become indistinguishable after projection. As it is customary in classical matrix completion results, we need to assume that each subspace is incoherent with the pattern of missing entries. We conjecture that the SSC-EWZF is guaranteed to give the correct clustering and completion provided that (1) the subspaces are sufficiently separated, (2) the data points are well distributed inside the subspaces, and (3) the subspaces are incoherent with the pattern of missing entries.

### 3.3. SSC by Expectation-based Completion (SSC-EC)

This approach exploits the fact that the self-expressiveness error depends on $X^\top X$ rather than $X$ because

$$\|X - XC\|_F^2 = \text{trace}((I - C)^\top X^\top X(I - C)). \tag{12}$$

This observation was the basis for the Kernel SSC method of Patel & Vidal (2014). Here, we use this idea to complete the kernel matrix $X^\top X$ in lieu of the data matrix $X$. To this end, we assume a random model in which each entry of $X$ is missing independently and with equal probability $\delta \in [0, 1]$. Under this model, the following lemma, which was stated in (Candes et al., 2014) without proof, shows how to obtain an unbiased estimator for $X^\top X$ from $X_{\text{miss}}$ and $\delta$.

**Lemma 1.** *Let $Z = [z_{ij}] \in \{0, 1\}^{D\times N}$ be a random matrix whose entries are i.i.d. Bernoulli with parameter $\delta$, i.e.,*

$$P[z_{ij} = k] = \delta^{1-k}(1 - \delta)^k, \ \ k = 0, 1, \tag{13}$$

*for all $i = 1, \ldots, D$ and all $j = 1, \ldots, N$. Then the matrix*

$$\Gamma \triangleq Y^\top Y - \delta \operatorname{diag}(Y^\top Y), \tag{14}$$

*where $Y \triangleq \frac{1}{1-\delta} Z \odot X$, is an unbiased estimator for $X^\top X$.*

*Proof.* If $j \neq i$, then

$$\gamma_{ij} = (Y^\top Y)_{ij} = \boldsymbol{y}_i^\top \boldsymbol{y}_j = \frac{1}{(1-\delta)^2} \sum_{k=1}^{D} x_{ki} x_{kj} z_{ki} z_{kj},$$

where $\boldsymbol{y}_i$ is the $i$th column of $Y$, $i = 1, \dots N$. Then,

$$\begin{aligned} E[\gamma_{ij}] &= \frac{1}{(1-\delta)^2} \sum_{k=1}^{D} x_{ki} x_{kj} E[z_{ki} z_{kj}] \\ &= \sum_{k=1}^{D} x_{ki} x_{kj} = \boldsymbol{x}_i^\top \boldsymbol{x}_j = (X^\top X)_{ij}, \end{aligned}$$

where we have $E[z_{ki} z_{kj}] = E[z_{ki}] E[z_{kj}]$ because of independence, and $E[z_{ki}] = 1 - \delta$ by (13).

If $j = i$, then

$$\begin{aligned} \gamma_{ii} &= (Y^\top Y)_{ii} - \delta (Y^\top Y)_{ii} = (1-\delta)(Y^\top Y)_{ii} \\ &= (1-\delta) \sum_{k=1}^{D} y_{ki} y_{ki} = \frac{1}{1-\delta} \sum_{k=1}^{D} x_{ki}^2 z_{ki}^2. \end{aligned}$$

So the expectation of $\gamma_{ii}$ for all $i = 1, \dots, N$ is

$$E[\gamma_{ii}] = \frac{1}{1-\delta} \sum_{k=1}^{D} x_{ki}^2 E[z_{ki}^2] = (X^\top X)_{ii},$$

where we use the fact that $E[z_{ki}^2] = 1 - \delta$ by (13).

Combining the above two cases, we have

$E[\gamma_{ij}] = (X^\top X)_{ij}$ for all $i = 1, \dots, D$ and $j = 1, \dots, N$,

hence $\Gamma$ is an unbiased estimator for $X^\top X$ as claimed. □

Lemma 1 suggests a simple approach for solving the incomplete SSC problem where we solve the problem in (5) after replacing the kernel matrix $X^\top X$ by

$$\widehat{\Gamma} = \frac{1}{(1-\delta)^2} (X_{\text{miss}}^\top X_{\text{miss}} - \delta \operatorname{diag}(X_{\text{miss}}^\top X_{\text{miss}})). \quad (15)$$

Notice, however that, while the matrix $X^\top X$ is positive semidefinite, the matrix $\widehat{\Gamma}$ in (15) might not be, hence the optimization problem in (5) may no longer be convex. To address this issue, we regularize $\widehat{\Gamma}$ as follows:

$$\widetilde{\Gamma} = \widehat{\Gamma} + \frac{\delta}{(1-\delta)^2} \max\{\operatorname{diag}(X_{\text{miss}}^\top X_{\text{miss}})\} I, \quad (16)$$

where $\max\{\operatorname{diag}(X_{\text{miss}}^\top X_{\text{miss}})\}$ is the maximum value of the diagonal entries of $X_{\text{miss}}^\top X_{\text{miss}}$. One can check that $\widetilde{\Gamma}$ is positive semi-definite. This leads to the following *SSC by Expectation-based Completion (SSC-EC)* algorithm:

$$\begin{aligned} \min_{C} \quad & \|C\|_1 + \frac{\lambda}{2} \operatorname{trace}((I - C)^\top \widetilde{\Gamma} (I - C)) \\ \text{s.t.} \quad & \operatorname{diag}(C) = 0. \end{aligned} \quad (17)$$

We use the Alternating Direction Method of Multipliers (ADMM) (Boyd et al., 2010) to solve (17). The steps of the ADMM algorithm for solving the original problem (5) are given in (Elhamifar & Vidal, 2013). Since the ADMM algorithm for SSC depends only on $X^\top X$, to solve (17) we simply replace $X^\top X$ by $\widetilde{\Gamma}$ in the ADMM algorithm of Elhamifar & Vidal (2013).

### 3.4. SSC by Column-wise Expectation-based Completion (SSC-CEC)

A concern with the SSC-EC method is that it is based on the self-expressiveness error $\|X - XC\|_F^2$, rather than its incomplete version $\|P_\Omega(X - XC)\|_F^2$. As discussed in §3.2 for the ZF+SSC method, this introduces an undesired bias in the estimation of $C$. Moreover, observe that when $\delta$ is small enough, the expression for $\widehat{\Gamma}$ in (15) becomes $\widehat{\Gamma} \approx \frac{1}{(1-\delta)^2} X_{\text{miss}}^\top X_{\text{miss}}$. Thus, the SSC-EC method effectively reduces to the ZF+SSC method with $\lambda$ replaced by $\frac{\lambda}{(1-\delta)^2}$.

To address this issue, we modify the self-expressiveness error to account only for observed entries, similar to our SSC-EWZF method. As it turns out, rather than solving for the entire matrix of coefficients as per (5), it will be more convenient to equivalently solve $N$ optimization problems, one for each column $\boldsymbol{c}_j$, $j = 1, \dots, N$, of $C$:

$$\min_{\boldsymbol{c}_j} \|\boldsymbol{c}_j\|_1 + \frac{\lambda}{2} \|\boldsymbol{x}_{\Omega_j, j} - X_{\Omega_j} \boldsymbol{c}_j\|_2^2 \;\; \text{s.t.} \;\; c_{jj} = 0. \quad (18)$$

Here $\boldsymbol{x}_{\Omega_j, j}$ is a vector with the entries of $\boldsymbol{x}_j$ indexed by $\Omega_j$ (i.e., the observed entries of $\boldsymbol{x}_j$), and $X_{\Omega_j}$ is a matrix with the rows of $X$ indexed by $\Omega_j$. After expanding the second term in the objective function above, we obtain

$$\|\boldsymbol{x}_{\Omega_j, j}\|_2^2 - 2 \boldsymbol{c}_j^\top X_{\Omega_j}^\top \boldsymbol{x}_{\Omega_j, j} + \boldsymbol{c}_j^\top X_{\Omega_j}^\top X_{\Omega_j} \boldsymbol{c}_j. \quad (19)$$

Therefore, to solve the above optimization problem, we only need to estimate $X_{\Omega_i}^\top x_{\Omega_j, j}$ and $X_{\Omega_j}^\top X_{\Omega_j}$, which are specifically tuned to each data point and its missing entries. The following lemma gives unbiased estimators for them.

**Lemma 2.** *For an arbitrary matrix $A$, let $A_\omega$ denote the submatrix of $A$ whose rows are indexed by $\omega \subseteq \{1, \dots, D\}$. Then, under the random model in Lemma 1, the kernel matrix $\Gamma(\omega) \in \mathbb{R}^{N \times N}$ defined as*

$$\Gamma(\omega) \triangleq Y_\omega^\top Y_\omega - \delta \operatorname{diag}(Y_\omega^\top Y_\omega) \quad (20)$$

*is an unbiased estimator for $X_\omega^\top X_\omega$.*

*Proof.* For this proof, we let $\gamma_{ij}$ denote the $(i, j)$th entry of $\Gamma(\omega)$. Similar to the proof of Lemma 1, we have

$$\gamma_{ij} = (Y_\omega^\top Y_\omega)_{ij} = \frac{1}{(1-\delta)^2} \sum_{k \in \omega} x_{ki} x_{kj} z_{ki} z_{kj} \quad (j \neq i),$$

$$\gamma_{ii} = (1-\delta)(Y_\omega^\top Y_\omega)_{ii} = \frac{1}{1-\delta} \sum_{k \in \omega} x_{ki}^2 z_{ki}^2.$$

Then, since $E[z_{ki}z_{kj}] = E[z_{ki}]E[z_{kj}] = (1-\delta)^2$ if $j \neq i$, and $E[z_{ki}^2] = 1-\delta$, we have

$$E[\gamma_{ij}] = (X_\omega^\top X_\omega)_{ij} \text{ for all } i = 1, \dots, D \text{ and } j = 1, \dots, N.$$

Thus $\Gamma(\omega)$ is an unbiased estimator for $X_\omega^\top X_\omega$. □

It follows from Lemma 2 that the matrix

$$\widehat{\Gamma}^{(j)} = \frac{(X_{\text{miss}})_{\Omega_j}^\top (X_{\text{miss}})_{\Omega_j} - \delta \operatorname{diag}\big((X_{\text{miss}})_{\Omega_j}^\top (X_{\text{miss}})_{\Omega_j}\big)}{(1-\delta)^2}$$

and its $j$th column $\widehat{\gamma}^{(j)}$, respectively, estimate the matrix $X_{\Omega_j}^\top X_{\Omega_j}$ and vector $X_{\Omega_j}^\top \boldsymbol{x}_{\Omega_j,j}$. However, since $\widehat{\Gamma}^{(j)}$ may not be positive semidefinite, like before we regularize it as:

$$\widetilde{\Gamma}^{(j)} = \widehat{\Gamma}^{(j)} + \frac{\delta}{(1-\delta)^2} \max\{\operatorname{diag}\big((X_{\text{miss}})_{\Omega_j}^\top (X_{\text{miss}})_{\Omega_j}\big)\} I.$$

Similarly, we define $\widetilde{\gamma}^{(j)}$ as the $j$th column of $\widetilde{\Gamma}^{(j)}$. This leads to the convex optimization problem:

$$\min_{\boldsymbol{c}_j} \|\boldsymbol{c}_j\|_1 + \frac{\lambda}{2}\big(\boldsymbol{c}_j^\top \widetilde{\Gamma}^{(j)} \boldsymbol{c}_j - 2\boldsymbol{c}_j^\top \widetilde{\gamma}^{(j)}\big) \ \text{s.t.}\ c_{jj} = 0, \quad (21)$$

which we solve using the ADMM algorithm, as before.

### 3.5. Discussion and Bias-corrected Dantzig Selector

Notice that solving the problem in (17) is more efficient than solving the $N$ problems in (21) because (17) involves inverting a matrix that is common for all columns of $C$, while (21) involves inverting a different matrix for each column of $C$. Notice also that both methods are different from the bias-corrected Dantzig selector (BCDS)

$$\min_{\boldsymbol{c}_j} \|\boldsymbol{c}_j\|_1 \ \text{s.t.}\ \|\widehat{\gamma}^{(j)} - \widehat{\Gamma}^{(j)}\boldsymbol{c}_j\|_\infty \leq \lambda \text{ and } c_{jj} = 0 \quad (22)$$

proposed by (Candes et al., 2014) because, instead of trying to penalize the standard reconstruction error $\|\boldsymbol{x}_j - X\boldsymbol{c}_j\|_2^2$, this approach tries to penalize $\|X^\top \boldsymbol{x}_j - X^\top X \boldsymbol{c}_j\|_\infty$. In our experience, this approach is computationally more expensive than SSC-CEC and sensitive to the choice of $\lambda$: a value that is too small leads to infeasible problems, while too large of a value leads to poor clustering performance.

## 4. Experiments

In this section, we evaluate the performance of MC+SSC, ZF+SSC, SSC-EWZF, SSC-EC, SSC-CEC, and BCDS on both synthetic data and the Hopkins 155 motion segmentation dataset (Tron & Vidal, 2007). All algorithms involve a penalty parameter $\lambda$ that should be carefully chosen so as to balance reconstruction error and sparsity: a small $\lambda$ may lead to sparse solutions, but a large reconstruction error, while a large $\lambda$ may give very good reconstruction, but non sparse solutions. In (Elhamifar & Vidal, 2013), an adaptive choice for $\lambda$ in (5) is given for a complete data matrix $X$ as:

$$\lambda = \alpha / \min_j \max_{i \neq j} |X^\top X|_{ij}, \quad (23)$$

where $\alpha \geq 1$ is a new tuning parameter. The justification for this choice for $\lambda$ is that, if $\alpha \geq 1$, every column of $C$ is guaranteed to be nonzero. Considering the specific form of ZF+SSC, SSC-EWZF, SSC-EC, and SSC-CEC, the denominator in (23) changes to $\min_j \max_{i\neq j} |X_{\text{miss}}^\top X_{\text{miss}}|_{ij}$, $\max_{i\neq j} |(X_{\text{miss}})_{\Omega_j}^\top (X_{\text{miss}})_{\Omega_j}|_{ij}$, $\min_j \max_{i\neq j} |\widetilde{\gamma}_{ij}|$, and $\max_{i\neq j} |\widetilde{\gamma}_{ij}^{(j)}|$, respectively.

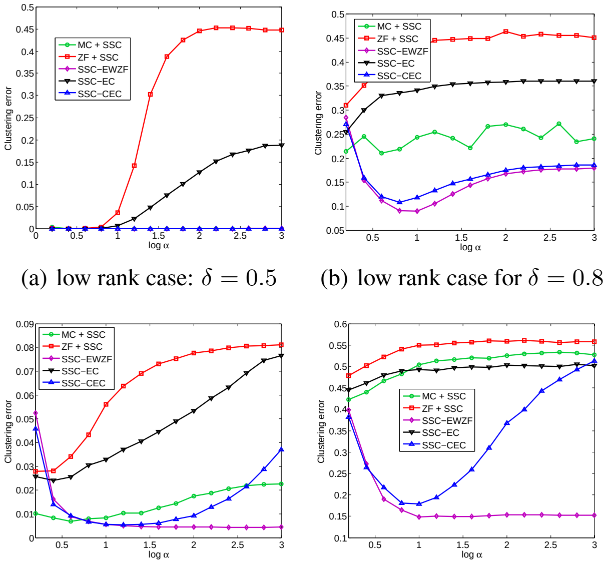
The clustering accuracy or alternatively the clustering error is used as the metric for comparing the performance of different methods. Specifically, the clustering accuracy is the ratio of correctly classified data points over all the data points and the clustering error $= 1-$ clustering accuracy.

### 4.1. Synthetic Data

The synthetic data is generated by drawing $N_0 \geq d$ points per subspace from a union of $n$ subspaces of equal dimension $d \ll D$ in $\mathbb{R}^D$. The data is generated as follows. Let $A_l \in \mathbb{R}^{D\times d}$, $B_l \in \mathbb{R}^{d \times N_0}$ be independent random Gaussian matrices, for $l = 1, \dots, n$. Since $\operatorname{rank}(A_l B_l) = d$ with high probability, the columns of $A_l B_l$ lie in a $d$-dimensional linear subspace $S_l = \operatorname{span}(A_l B_l)$. Also the randomness guarantees that the subspaces $\{S_l\}_{l=1}^n$ are independent with high probability. Therefore the complete data matrix $X$ can be constructed as $X = [A_1 B_1, \dots, A_n B_n]$, where $X \in \mathbb{R}^{D \times N}$ if $N_0 = N/n$. We also normalize the columns of $X$ to have unit $\ell_2$ norm. To generate the incomplete data, we draw a random matrix $W$ whose entries are i.i.d. Bernoulli with missing rate $\delta$.

In order to study the influence of the rank of the data matrix on the clustering performance, we evaluate all methods both in the low-rank and high-rank regimes. Note that the rank of $X$ will be approximately equal to the sum of the dimensions of the subspaces, i.e., $\operatorname{rank}(X) \approx nd$, since the subspaces are independent with high probability. In our experiments, we use $D = 100, N_0 = 50, d = 5, n = 2$ to simulate the low-rank case ($dn \ll D$), and $D = 25, N_0 = 50, d = 5, n = 5$ to simulate the high-rank case ($dn \approx D$).

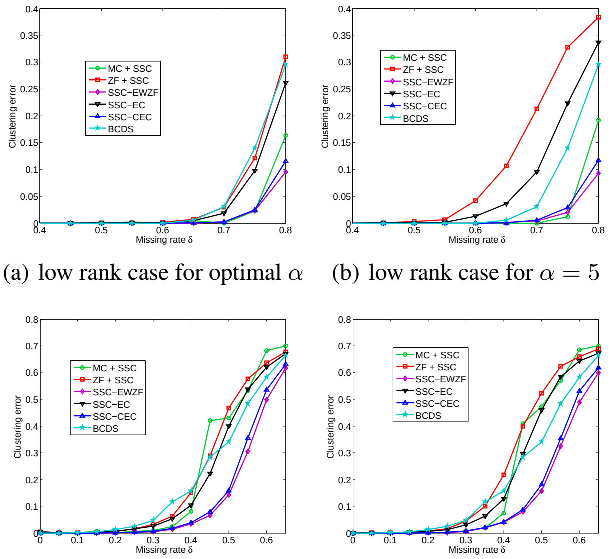
**Effect of the penalty parameter.** Figures 1(a)-1(b) show the clustering error versus $\alpha$ in the low-rank case for a missing rate of $\delta = 0.5$ and 0.8. When $\delta = 0.5$, we can see that methods that use the complete self-expressiveness error (ZF+SSC and SSC-EC) are very sensitive to the choice of $\alpha$. In contrast, methods based on the incomplete error (SSC-EWZF and SSC-CEC) work perfectly for almost all values of $\alpha$. Also, SSC-EC is better than ZF+SSC, as expected. The performance of MC+SSC is not very sensitive to $\alpha$, but as $\delta$ increases from 0.5 to 0.8 the clustering error



(a) low rank case: $\delta = 0.5$ (b) low rank case for $\delta = 0.8$

(c) high rank case for $\delta = 0.3$ (d) high rank case for $\delta = 0.5$

*Figure 1.* Clustering error versus $\alpha$ (log scale). (a)-(b) show the low-rank case with $D = 100$, $N_0 = 50$, $d = 5$, $n = 2$. (c)-(d) show the high-rank case with $D = 25$, $N_0 = 50$, $d = 5$, $n = 5$.



(a) low rank case for optimal $\alpha$ (b) low rank case for $\alpha = 5$

(c) high rank case for optimal $\alpha$ (d) high rank case for $\alpha = 5$

*Figure 2.* Clustering error versus missing rate $\delta$. (a)-(b) show the low-rank case with $D = 100, N_0 = 50, d = 5, n = 2$. (c)-(d) show the high-rank case with $D = 25, N_0 = 50, d = 5, n = 5$.

increases drastically. This may be explained because MC fails when the missing rate is too high. The clustering errors of SSC-EWZF and SSC-CEC also increase with $\delta$, but less dramatically. Overall, SSC-EWZF and SSC-CEC are the most accurate and robust methods against changes in $\alpha$.

Figures 1(c)-1(d) show the corresponding results in the high-rank case for a missing rate of $\delta = 0.3$ and 0.5. By comparing Figures 1(d) and 1(a), we see that the clustering errors of all methods increase, arguably due to the increase in the number of subspaces. Notice also that, as before, ZF+SSC and SSC-EC give the highest errors, but this time MC+SSC also gives high errors because the MC step fails as the data matrix is not low rank. SSC-CEC performs better than the previous methods, but its sensitivity with respect to $\alpha$ increases. Overall, SSC-EWZF performs best as it gives lower errors for a broader range of values of $\alpha$.

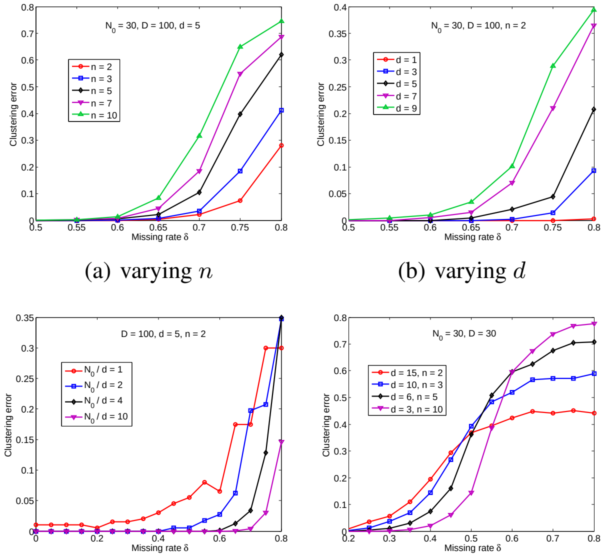
**Effect of the missing rate.** To evaluate performance as a function of the missing rate, we first need to decide how to do a fair comparison. This is because performance depends on the choice of the parameter $\alpha$ and the optimal range for $\alpha$ may not be the same for different methods. One possibility is to choose the optimal $\alpha$ for each method and then compare their performance. In practice, however, we may not know the best $\alpha$, hence fixing $\alpha$ for all methods is another possibility. We evaluate performance in both ways.

Figures 2(a)-2(b) show the clustering error versus $\delta$ in the low-rank case. Observe that all methods give nearly perfect clustering for $\delta < 0.6$. As before, the performance of ZF+SSC and SSC-EC deteriorates very quickly as $\delta$ increases, while the performance of MC+SSC, SSC-EWZF, and SSC-CEC remains very good. For this experiment only we evaluate the BCDS method. While BCDS uses the same completion strategy as SSC-CEC, the sparse representation is found by solving (22) instead of (21). As we can see, this change results in a higher clustering error. As discussed before, we believe this is because the performance of (22) is highly dependent on the choice of $\lambda$ in (22). We choose $\lambda = \sqrt{\frac{2\delta \log N}{n(1-\delta)}}$, as suggested in (Candes et al., 2014).

Figures 2(c)-2(d) show the results for the high-rank case. Observe that all methods give nearly perfect clustering for $\delta < 0.2$, but their performance deteriorates quickly as $\delta$ increases. In particular, MC+SSC fails since the data matrix is no longer low-rank, giving a clustering error that is similar to that of ZF+SSC, SSC-EC and BCDS. SSC-EWZF and SSC-CEC give clustering errors that are similar and clearly lower than those of ZF+SSC, SSC-EC and BCDS.

**Effect of the number of subspaces, the dimension of the subspaces, and the number of samples per subspace.** Figure 3 evaluates the performance of SSC-CEC as a function of the number of subspaces $n$, the dimension of the subspaces $d$, and the number of samples $N_0$ per subspace.

In Figure 3(a), we fix $N_0 = 30$, $D = 100$, $d = 5$, and vary $n$ as $2, 3, 5, 7, 10$. We can see that the break point for perfect clustering ($\delta \approx 0.55$) does not change too much as $n$ varies. This means that SSC-CEC is robust to variations of the number of clusters when the missing rate is low. For a high missing rate, however, the clustering error increases



(a) varying $n$

(b) varying $d$

(c) varying $N_0$

(d) varying $d$ and $n$

*Figure 3.* Clustering error versus number of subspaces $n$, dimension of the subspaces $d$, and number of samples $N_0$ per subspace for SSC-CEC. (a) Varying $n$ when $N_0 = 30$, $D = 100$ and $d = 5$; (b) Varying $d$ when $N_0 = 30$, $D = 100$, $n = 2$; (c) Varying $N_0$ when $D = 100$, $d = 5$, $n = 2$; (d) Varying $d$ and $n$ while fixing their product $nd = D$ when $N_0 = 30$ and $D = 30$.

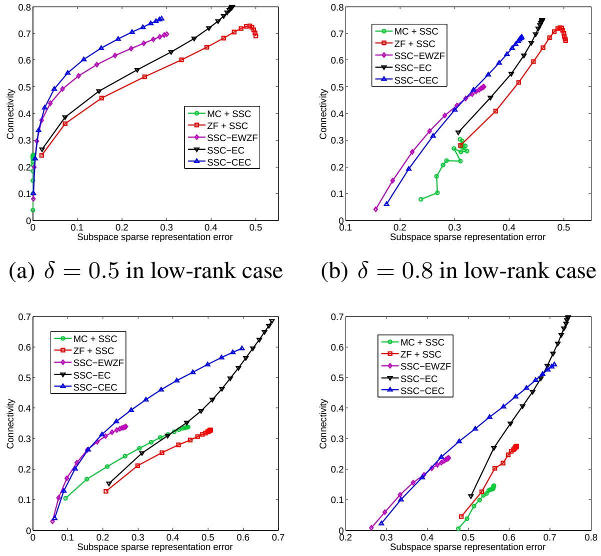
very quickly as the number of subspaces increases.

In Figure 3(b), we fix $N_0 = 30$, $D = 100$, $n = 2$, and vary $d$ as $1, 3, 5, 7, 9$. We can see that as the dimension of each subspace increases, the clustering error increases. This is expected because the data is not as well distributed inside the subspaces when $N_0/d$ decreases, hence SSC may fail.

In Figure 3(c), we fix $D = 100$, $d = 5$, $n = 2$, and vary $N_0$ as $5, 10, 20, 50, 100$. Observed that as more samples are added, the performance of SSC-CEC becomes more and more stable. This is because the self-expressiveness property may fail when the number of samples per subspace is too small. Also, increasing the number of samples per subspace improves the break point for perfect clustering.

In Figure 3(d), we fix $N_0 = 30$, $D = 30$, and vary both $d$ and $n$ while keeping their product constant as $nd = D$. By construction, the subspaces are independent with high probability. Hence, this choice for $n$ and $d$ will make the data matrix almost full rank. It is interesting to see that when the missing rate is low, SSC-CEC performs best for low-dimensional subspaces, but when the missing rate is high, it performs best for high-dimensional subspaces.

**SSR error and connectivity.** While clustering error is the ultimate performance metric for subspace clustering, such a metric depends on the spectral clustering step. To evaluate the direct output of the SSC algorithms, it is also customary to evaluate the quality of the sparse representation matrix $C$ by measuring the subspace sparse representation (SSR) error and the connectivity of each cluster. For SRR, we compute the proportion of the sum of the absolute values of the entries from other subspaces in each column of $C$ and then average over all the columns. Since separability of different subspaces is desired, a lower SSR error is preferable. For connectivity, we calculate the second smallest eigenvalue $\lambda_2$ of the normalized Laplacian matrix for each cluster, which measures the connectivity of the corresponding subgraph. We take the smallest $\lambda_2$ (across all groups) as a measure of the connectivity for all groups. Note that higher connectivity is preferable, since it prevents over-clustering.



(a) $\delta = 0.5$ in low-rank case

(b) $\delta = 0.8$ in low-rank case

(c) $\delta = 0.3$ in high-rank case

(d) $\delta = 0.5$ in high-rank case

*Figure 4.* SSR error versus connectivity by varying the parameter $\alpha$. (a) Missing rate $\delta = 0.5$ in low-rank case; (b) Missing rate $\delta = 0.8$ in low-rank case; (c) Missing rate $\delta = 0.3$ in high-rank case; and (d) Missing rate $\delta = 0.5$ in high-rank case.

We report both the SSR error and connectivity by varying the penalty parameter $\alpha$. We observe that connectivity usually goes up when the SSR error increases. Thus we plot them together in an ROC-like curve, as shown in Figure 4.

Figures 4(a) and 4(b) plot SSR error versus connectivity for a missing rate $\delta$ of 0.5 and 0.8, respectively, in the low-rank case. Observe that MC+SSC is slightly better than SSC-EWZF and SSC-CEC when $\delta = 0.5$ since its curve is above and to the left of that for SSC-EWZF and SSC-CEC. Observe also that MC+SSC produces very low SSR errors but also very low connectivity for all values of $\alpha$. When $\delta = 0.8$, the performance of MC+SSC deteriorates, which is consistent with the fact that MC fails for very high missing rate. On the other hand, SSC-EWZF and SSC-CEC continue to be the best methods, with SSC-EWZF being slightly better for small SSR errors, and SSC-CEC being slightly better for high SSR errors.

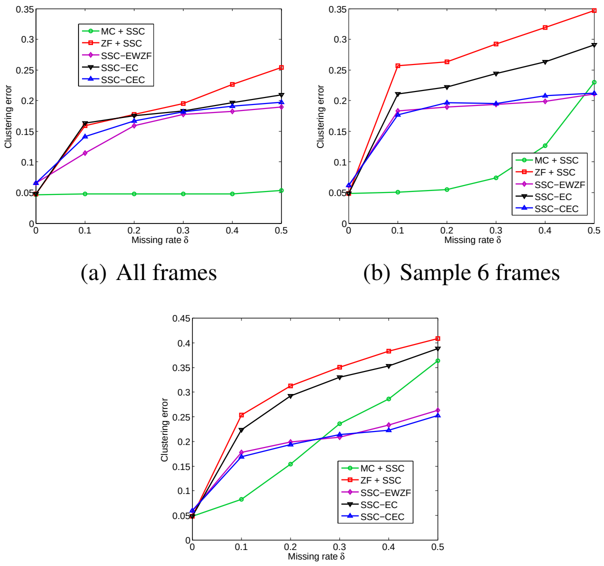
Figures 4(c) and 4(d) plot the SSR error versus connectivity for a missing rate $\delta$ of $0.3$ and $0.5$, respectively, in the high-rank case. As before, SSC-EWZF and SSC-CEC continue to be the best methods, showing that they are clearly advantageous in the high-rank case when MC does not work.

### 4.2. Motion Segmentation

In this experiment, we evaluate the performance of different subspace clustering and completion methods on the motion segmentation problem in computer vision. Given a video with multiple moving objects, feature extraction and tracking methods are used to extract $N$ feature points and track them across $F$ frames of the video. Under the affine projection model, the trajectories associated with one moving object lie in an affine subspace of $\mathbb{R}^{2F}$ of dimension $d = 1, 2, 3$. The task is to cluster these trajectories according to their corresponding motion subspaces.

We evaluate different methods on the Hopkins 155 data set, which contains 155 video sequences with 2 or 3 moving objects. Since in this case the dimension of each motion subspace is $d = 1, 2, 3$, the number of motions is $n = 2, 3$, and the dimension of the ambient space is $D = 2F \geq 30$, the data matrix is always low rank, and so we expect MC+SSC to work very well. Thus, to simulate the high-rank case, we also do experiments on subsampled trajectories with 3 or 6 frames so that the dimension of the ambient space is $D = 6$ or $12$. The sampled frames are chosen to be as equally spaced and spread out as possible. For example, if the original data has 20 frames, and the number of sampled frames is 3, then we sample the 1st, 10th and 19th frame. Thus, the full data, 6-frame data and 3-frame data cases represent the low-rank, mid-rank and high-rank cases, respectively. To make the data incomplete, we generate a mask matrix $W \in \{0,1\}^{D \times N}$ whose entries are i.i.d. Bernoulli with missing rate $\delta$ and let $X_{\text{miss}} = W \odot X$. Unlike the synthetic data experiments, in the Hopkins 155 data set the subspaces are affine. We handle this by adding the constraint $\mathbf{1}^\top C = \mathbf{1}^\top$ to each convex optimization problem, which enforces the sparse coefficients to add up to 1.

Figure 5 shows the clustering error of different methods as a function of the missing rate $\delta$ with fixed $\alpha = 5$. As we can see, SSC-CEC outperforms ZF+SSC and SSC-EC in all cases. For the low-rank case shown in Figure 5(a), MC+SSC does a very good job as expected. However, as the dimension of the ambient space reduces, MC starts to fail. As a result, the clustering error for MC+SSC increases rapidly in Figure 5(b), and even more so in Figure 5(c). Meanwhile, SSC-EWZF and SSC-CEC are less sensitive to the dimension of the ambient space or the rank of the data matrix, and outperform MC+SSC in the high-rank case (Figure 5(c)) when the missing rate is higher than $0.3$. Therefore, SSC-EWZF and SSC-CEC are potentially better methods for the high-rank and high-missing rate scenario.



(a) All frames

(b) Sample 6 frames

(c) Sample 3 frames

*Figure 5.* Variation of performance with respect to missing rate $\delta$ on Hopkins 155 data set. Penalty parameter $\alpha$ is fixed as 5. Experiments are conducted on (a) full data, (b) 6 frames sampled data and (c) 3 frames sampled data respectively.

## 5. Conclusions

We have proposed two algorithms for subspace clustering with missing entries called sparse subspace clustering by entry wise zero fill (SSC-EWZF) and sparse subspace clustering by column wise expectation completion (SSC-CEC). SSC-EWZF is a natural generalization of SSC in which the self-expressiveness error is restricted only to the observed entries, while SSC-CEC is a natural generalization of SSC where the kernel matrix $X^\top X$ restricted to the observed entries is replaced by an unbiased estimator under a random model. Both algorithms were compared against the natural approaches of first filling in the missing entries either with zeros (ZF+SSC) or using matrix completion (MC+SSC). The results show that MC+SSC is competitive only when the data matrix is low rank and the percentage of missing entries is low. Otherwise, SSC-EWZF and SSC-CEC perform significantly better. Moreover, we conjectured that SSC-EWZF should give perfect clustering provided that the subspaces are sufficiently separated, the data are well distributed inside the subspaces, and the subspaces are incoherent with respect to the missing entries. Such conditions for correctness will be investigated in future work.

## Acknowledgements

This work was supported by NSF grant 1447822. The authors thank Li Chunguang for his help with the design of the experiment reported in Figure 3(d). René Vidal also thanks Li Chunguang, Chong You, Mahdi Soltanolkotabi, Lester Mackey, and Emmanuel Candès for extremely insightful discussions on the problem of subspace clustering with missing data.

## References

Balzano, L., Nowak, R., and Bajwa, W. Column subset selection with missing data. In *NIPS Workshop on Low-Rank Methods for Large-Scale Machine Learning*, 2010.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2010.

Bradley, P. S. and Mangasarian, O. L. k-plane clustering. *Journal of Global Optimization*, 16(1):23–32, 2000.

Cai, J-F, Candés, E. J., and Shen, Z. A singular value thresholding algorithm for matrix completion. *SIAM Journal of Optimization*, 20(4):1956–1982, 2008.

Candès, E. and Recht, B. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9:717–772, 2009.

Candès, E. and Tao, T. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.

Candes, E. J., Mackey, L., and Soltanolkotabi, M. From robust subspace clustering to full-rank matrix completion. Unpublished abstract, 2014.

Elhamifar, E. and Vidal, R. Sparse subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

Elhamifar, E. and Vidal, R. Clustering disjoint subspaces via sparse representation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2010.

Elhamifar, E. and Vidal, R. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11): 2765–2781, 2013.

Eriksson, Brian, Balzano, Laura, and Nowak, Robert. High-rank matrix completion. *Journal of Machine Learning Research*, Proceedings Track 22:373–381, 2012.

Favaro, P., Vidal, R., and Ravichandran, A. A closed form solution to robust subspace estimation and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

Gross, D. Recovering low-rank matrices from few coefficients in any basis. *IEEE Trans on Information Theory*, 57(3):1548–1566, 2011.

Gruber, A. and Weiss, Y. Multibody factorization with uncertainty and missing data using the EM algorithm. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume I, pp. 707–714, 2004.

Keshavan, R., Montanari, A., and Oh, S. Matrix completion from a few entries. *IEEE Transactions on Information Theory*, 2010.

Liu, G., Lin, Z., and Yu, Y. Robust subspace segmentation by low-rank representation. In *International Conference on Machine Learning*, 2010.

Liu, Guangcan, Lin, Zhouchen, Yan, Shuicheng, Sun, Ju, and Ma, Yi. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 35(1):171–184, Jan 2013.

Lu, Can-Yi, Min, Hai, Zhao, Zhong-Qiu, Zhu, Lin, Huang, De-Shuang, and Yan, Shuicheng. Robust and efficient subspace segmentation via least squares regression. In *Proceedings of European Conference on Computer Vision*, 2012.

Patel, V. M. and Vidal, R. Kernel sparse subspace clustering. In *IEEE International Conference on Image Processing*, 2014.

Recht, Benjamin. A simpler approach to matrix completion. *Journal of Machine Learning Research*, 12:3413–3430, 2011.

Soltanolkotabi, M. and Candès, E. J. A geometric analysis of subspace clustering with outliers. *Annals of Statistics*, 2013.

Soltanolkotabi, Mahdi, Elhamifar, Ehsan, and Candès, Emmanuel J. Robust subspace clustering. *Annals of Statistics*, 42(2):669–699, 2014.

Tipping, M. and Bishop, C. Mixtures of probabilistic principal component analyzers. *Neural Computation*, 11(2): 443–482, 1999.

Tron, R. and Vidal, R. A benchmark for the comparison of 3-D motion segmentation algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

Tseng, P. Nearest $q$-flat to $m$ points. *Journal of Optimization Theory and Applications*, 105(1):249–252, 2000.

Vidal, R. Subspace clustering. *IEEE Signal Processing Magazine*, 28(3):52–68, March 2011.

Vidal, R. and Favaro, P. Low rank subspace clustering (LRSC). *Pattern Recognition Letters*, 43:47–61, 2014.

Vidal, R. and Hartley, R. Motion segmentation with missing data by PowerFactorization and Generalized PCA. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume II, pp. 310–316, 2004.

Vidal, R., Ma, Y., and Sastry, S. Generalized Principal Component Analysis (GPCA). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1–15, 2005.

Wang, Yu-Xiang and Xu, Huan. Noisy sparse subspace clustering. In *Proceedings of International Conference on Machine Learning*, 2013.

Zhou, M., Wang, C., Chen, M., Paisley, J., Dunson, D., and Carin, L. Nonparametric bayesian matrix completion. In *Sensor Array and Multichannel Signal Processing Workshop*, 2010.