The Annals of Applied Statistics
2019, Vol. 13, No. 2, 1016–1042
https://doi.org/10.1214/18-AOAS1220

SPARSE PRINCIPAL COMPONENT ANALYSIS WITH MISSING OBSERVATIONS[1]

BY SEYOUNG PARK AND HONGYU ZHAO

Sungkyunkwan University and Yale University

Principal component analysis (PCA) is a commonly used statistical method in a wide range of applications. However, it does not work well when the number of features is larger than the sample size. We consider the estimation of the sparse principal subspace in the high dimensional setting with missing data motivated by the analysis of single-cell RNA sequence data. We propose a two step estimation procedure, and establish the rates of convergence for estimating the principal subspace. Simulated examples with various missing mechanisms show its competitive performance compared to existing sparse PCA methods. We apply the method to single-cell data and show that the proposed method can better distinguish cell types than other PCA methods.

1. Introduction. Principal component analysis (PCA) is one of the most commonly used methods to reduce data dimension. It is effective in capturing the main structure of the data in many application areas such as data compression, visualization, and clustering. PCA is especially useful in the analysis of high-dimensional data that arise in a wide range of fields such as genomics, signal processing, and risk management. For example, gene expression data generated from various platforms have 20,000 or more features whereas the sample size is much more limited, often in the dozens or hundreds range.

This paper is motivated by the analysis of single-cell RNA sequence (scRNA-seq) data, which enable high-throughput single-cell genomic measurements and allow the analysis of cell-to-cell heterogeneity. One major effort of single cell analysis is to identify distinct cell types, that may be characterized by a small number of genes among tens of thousands of genes. Sparse PCA (Zou, Hastie and Tibshirani (2006)) of scRNA-seq data may be an effective approach to inferring different cell types through identifying a few genes that describe most of the biological features contained in the data. One major challenge of scRNA-seq data, compared to bulk RNA-seq or microarray gene expression data, is that they have many missing values due to technical and sampling issues, where the missing mechanism in scRNA-seq data is rather complex.

Sparse PCA was proposed to overcome inconsistency of the standard PCA for high-dimensional data. In the classical setting where the dimension of the data

Received June 2017; revised September 2018.

[1] Supported by NIH Grant GM59507; Seyoung Park is supported by a National Research Foundation of Korea grant funded by the Korea government (MSIP) (No. NRF-2019R1C1C1003805).

Key words and phrases. PCA, missing data, high dimensional, single-cell data.

is small compared with the sample size, the principal components are obtained by the leading eigenvectors of the sample covariance matrix. But this estimate is ill-posed when the number of variables p is larger than the sample size n. See Baik and Silverstein (2006), Paul (2007), and Johnstone and Lu (2009) for details. Sparse PCA (Zou, Hastie and Tibshirani (2006)) generally assumes that only a few variables constitute principal components.

Various estimators of the sparse principal components have been proposed. Amini and Wainwright (2009), Journée et al. (2010), Yuan and Zhang (2013), Vu and Lei (2013), and Berthet and Rigollet (2013) focused on estimating the leading principal eigenvector v under various sparsity assumptions on v. Ma (2013) proposed a new method using iterative thresholding for estimating the principal subspace. Cai, Ma and Wu (2013) introduced an adaptive procedure for estimating the principal subspace and established that the estimator attains the optimal rates of convergence. Qi, Luo and Zhao (2013) proposed a sparse PCA method by introducing a new norm in traditional eigenvalue problem to obtain orthogonal loadings. Wang, Lu and Liu (2014) studied an efficient algorithm to estimate the principal subspace under more general model conditions, and established the computational and statistical convergences. More recently, Deshpande and Montanari (2016) proposed a covariance thresholding algorithm for the support recovery of principal vectors.

Although these works have established nice theoretical properties, our empirical evidence suggests that existing sparse PCA methods do not work well when data have many missing values, an area that has not been well studied. Standard PCA often replaces the missing data with the mean or an extreme value (Dodge (1985), Chen (2002)). However, such a strategy is no longer valid when a large portion of the data is missing. See Chen (2002) for an example of such data. Tsalmantza and Hogg (2012), Bailey (2012), and Delchambre (2014) present a weighted principal component analysis on noisy datasets with missing values, where missing data are limiting cases of weight 0. Compared to the standard PCA, these methods can be advantageous in that they naturally account for poorly measured and missing data.

For sparse PCA, Lounici (2013) considered estimating the first principal component v when each component of the data is observed independently of the other components with probability $\delta \in (0, 1]$. Kundu, Drineas and Magdon-Ismail (2015) considered recovering v by sampling the data. But in practice, we are not able to sample the data, but rather the samples are given. These works only estimate the first principal component v in the exact sparse case, that is, v has a bounded number of nonzero components.

In this paper, we consider sparse PCA with missing observations by utilizing the matrix completion problem for estimating the principal subspace. We impose the sparsity condition on the principal vectors $V = [v_1, \dots, v_r]$ such that each of the principal components $v_1, \dots, v_r$ has at most s_0 nonzero entries. For simplicity of notation, let s be the number of nonzero rows of the V, thus, $s_0 \le s \le rs_0$. This conditions is essentially the same as the total sparsity condition (Ma (2013), Kundu, Drineas and Magdon-Ismail (2015), Deshpande and Montanari (2016)). Specifically, we consider the more challenging case where the sparsity parameter s could be large, that is, $s = n^{\varsigma} = O(\sqrt{n})$ for some small $\varsigma > 0$, motivated by Deshpande and Montanari (2016). This sparsity condition is adopted because we believe that in our motivating scRNA-seq data only a few genes are associated with principal components that can be used to define cell types. The rest of the paper is organized as follows. In the remainder of this section, we summarize the notation used in this paper, present a connection between matrix completion and PCA analysis, and describe our PCA model. The proposed method and algorithm are described in Section 2. Theoretical properties of our method are presented in Section 3. A detailed description of the algorithm is given in Section 4. We show simulation results in Sections 5 and 6. Real data examples using scRNA-seq data are given in Section 7. All technical proofs and additional results are presented in the Supplementary Material (Park and Zhao (2019)).

1.1. *Some notations.* For any positive integer p, define $[p] := \{1, \dots, p\}$. For a matrix $X \in \mathbb{R}^{n \times p}$ and sets $C \subseteq [p]$ and $R \subseteq [n]$, $X_{\cdot,C}$ and $X_{R,\cdot}$ denote sub-matrices of X, which consist of the columns and rows of X, indexed by C and R, respectively. Similarly, $X_{R,C}$ is the sub-matrix of X consisting of the rows in R and columns in C. Specifically, $X_{i,\cdot}$ and $X_{\cdot,j}$ denote the ith row and jth column of the matrix X, $\sigma_j(X)$ represents the jth largest singular value, $\|X\|_2 = \sigma_1(X)$ is a spectral norm, $\|X\|_F = \sqrt{\sum_{i,j} X_{ij}^2}$ is a Frobenius norm, and $\|X\|_* = \sum_j \sigma_j(X)$ is the nuclear norm of X. For matrices X and Y of the same dimension, let $\langle X, Y\rangle = \operatorname{tr}(X^T Y)$. For a matrix $X \in \mathbb{R}^{n \times p}$ and a set $\Omega \subset [n] \times [p]$, define $X_\Omega \in \mathbb{R}^{n \times p}$ as a matrix with the elements in the set Ω preserved, and the other entries replaced with 0:

$$(X_\Omega)_{ij} = X_{ij} \qquad \text{for } (i, j) \in \Omega, \qquad (X_\Omega)_{ij} = 0 \qquad \text{for } (i, j) \notin \Omega.$$

For two vectors $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^p$, $u \otimes v$ denotes an n by p matrix satisfying $(u \otimes v)_{i,j} = u_i v_j$. Let 1_n be the n-dimensional vector of ones. For numbers $a_1, \dots, a_n$, $\operatorname{diag}(a_1, \dots, a_n)$ denotes an n by n diagonal matrix whose diagonal entries starting in the upper left corner are $a_1, \dots, a_n$. We use C for absolute constants that may change from line to line. We write $a = O(b)$ or $a \lesssim b$ if $a \le Cb$ for some positive absolute constants C. If $a \ge Cb$ for some positive constants C, then we write $a = \Omega(b)$ or $a \gtrsim b$. We use $a \asymp b$ when $a \lesssim b$ and $b \lesssim a$. For two numbers a and b, we use the notation $a \vee b = \max(a, b)$ and $a \wedge b = \min(a, b)$.

1.2. *Motivation of the proposed sparse PCA.* For the column-centered data $X \in \mathbb{R}^{n \times p}$, standard PCA (Hotelling (1933), Jolliffe (1986)) seeks the best rank r estimate:

$$\widehat{Z} = \underset{Z:\operatorname{rank}(Z) \le r}{\operatorname{argmin}} \|X - Z\|_F^2. \tag{1}$$

The solution is provided by the singular value decomposition (SVD) of $X = UDV^T$ (Eckart and Young (1936)): $\widehat{Z} = U_{\cdot,[r]}D_{[r],[r]}V_{\cdot,[r]}^T$, where $U_{\cdot,[r]} \in \mathbb{R}^{n\times r}$, $V_{\cdot,[r]} \in \mathbb{R}^{p\times r}$, and $D_{[r],[r]} \in \mathbb{R}^{r\times r}$. Here, the r columns of $V_{,[r]}$ are PCA loadings and $U_{\cdot,[r]}D_{[r],[r]}$ represents PCA scores. This PCA output can also be obtained by the eigen-decomposition of the matrix X^TX/n to find the first r leading sample principal eigenvectors. However, in the presence of missing observations, where $\Omega \subset [1,n]\times[1,p]$ is the set of observed elements, the method based on the eigen-decomposition of the Gram matrix $\Gamma = X_\Omega^T X_\Omega$ is not appropriate as Γ is not informative in this setting. For example, if the ith and jth columns of X have few observed entries on the same rows, the analysis based on Γ_{ij} can be ill-posed.

One can consider the following optimization, which is motivated by (1):

$$\widehat{Z} = \underset{Z:\mathrm{rank}(Z)\le r}{\operatorname{argmin}} \|X_\Omega - Z_\Omega\|_F^2,$$

but it does not have a closed form solution. As a matrix completion method, Mazumder, Hastie and Tibshirani (2010) considered the following convex relaxation: for $\mu \ge 0$,

$$\widehat{Z} = \underset{Z\in\mathbb{R}^{n\times p}}{\operatorname{argmin}} \frac{1}{2}\|X_\Omega - Z_\Omega\|_F^2 + \mu\|Z\|_*. \tag{2}$$

REMARK 1. As an alternative to (2), Srebro, Rennie and Jaakkola (2005) and Hastie et al. (2015) considered the following optimization objective:

$$(\widehat{A},\widehat{B}) = \underset{\widehat{A}\in\mathbb{R}^{n\times r},\widehat{B}\in\mathbb{R}^{p\times r}}{\operatorname{argmin}} \frac{1}{2}\|X_\Omega - (AB^T)_\Omega\|_F^2 + \frac{\mu}{2}(\|A\|_F^2 + \|B\|_F^2), \tag{3}$$

which is based on the fact (Srebro, Rennie and Jaakkola (2005)) that

$$\|Z\|_* = \min_{A,B:Z=AB^T} \frac{1}{2}(\|A\|_F^2 + \|B\|_F^2).$$

Note that the optimization problems (2) and (3) provide an equivalent solution, that is, $\widehat{Z} = \widehat{A}\widehat{B}^T$ when $r \ge \mathrm{rank}(\widehat{Z})$.

In the above settings, we assume that X is column-centered, but in practice we observe X_Ω with uncentered data X. We use the mean as an extra parameter (μ in (4)) for the optimization. We consider the following optimization by imposing the penalty term on the unobserved parts of the data: for $\gamma_d, \zeta_d \ge 0$,

$$\widehat{Z} = \underset{Z\in\mathbb{R}^{n\times p},\mu\in\mathbb{R}^p}{\operatorname{argmin}} \|Z_\Omega + (1_n \otimes \mu^T)_\Omega - X_\Omega\|_F^2 + \gamma_d\|Z\|_* + \rho_{\zeta_d}(Z_{\Omega^c}), \tag{4}$$

where $\rho_{\zeta_d} : \mathbb{R}^{n\times p} \to R$ is a penalty function, depending on a tuning parameter ζ_d, and $\mu = (\mu_1,\dots,\mu_p)^T$ corresponds to the mean vector. The last term in (4) enforces a certain type of structure on the solution $\widehat{Z}$. For example, if one believes that most of the unobserved entries in X are nearly zero or wants to identify unobserved entries having large values, one can use $\rho_{\zeta_d}(A) = \zeta_d \sum_{ij} |A_{ij}|$ for a matrix A.

For scRNA-seq data analysis, we suggest using a regularizer $\rho_{\zeta_d}(A) = \zeta_d \sum_{ij} A_{ij}^2$, which also empirically works well for various missing mechanism cases as presented in Sections 5–6:

$$(5)\quad (\widehat{Z}, \widehat{\mu}) = \underset{Z \in \mathbb{R}^{n \times p}, \mu \in \mathbb{R}^p}{\operatorname{argmin}} \|Z_\Omega + (1_n \otimes \mu^T)_\Omega - X_\Omega\|_F^2 + \gamma_d \|Z\|_* + \zeta_d \|Z_{\Omega^c}\|_F^2.$$

The last term improves the conditioning of the problem and gives preference to $\widehat{Z}_{\Omega^c}$ with smaller norms. Note that Gaussian functions $\exp(-\lambda x^2)$ for some $\lambda > 0$ are known as possible dropout event generating functions for normalized scRNA-seq data X (Pierson and Yau (2015), Wang et al. (2017)), that is, X_{ij} are independently observed with $P(X_{ij}\text{is observed}) = 1 - \exp(-\lambda X_{ij}^2)$. In this case, we have

$$\log\{P(X_{ij} \text{ is unobserved for all } (i,j) \in \Omega^c \mid X)\} = -\lambda \|X_{\Omega^c}\|_F^2.$$

Hence, the last term in (5) plays a role for imputing the unobserved parts X_{Ω^c} of scRNA-seq data X by indirectly maximizing the conditional probability that the entries in Ω^c are indeed unobserved. When $\zeta_d = 0$ and $\mu = 0$, (5) reduces to (2) that only minimizes the distance of Z and X on the observed part Ω. In this paper, we propose a two-step procedure. In the first step, we select significant variables among p variables by a certain criterion. The second step uses the sub-matrix of X_Ω with the selected variables to solve optimizations motivated by (5). See Section 2 for the detailed algorithm.

1.3. *Model.* In this subsection, we present the sparse PCA model that will be used for theoretical analysis of the proposed method. Suppose that we have an $n \times p$ matrix X with observed entries indexed by the set $\Omega := \{(i,j) : X_{ij}\text{is observed}\}$ in the following model:

$$(6)\qquad X = 1_n \otimes (\mu^o)^T + X_0 + W = 1_n \otimes (\mu^o)^T + UDV^T + W.$$

Here $\mu^o = (\mu_1^o, \ldots, \mu_p^o)^T$ represents the mean vector, U is an $n \times r$ random matrix with independent and identically distributed (i.i.d.) $N(0,1)$ entries, $D = \operatorname{diag}(\lambda_1^{1/2}, \ldots, \lambda_r^{1/2})$ with $0 < \lambda_r \le \cdots \le \lambda_1$, $V \in \mathbb{R}^{p \times r}$ has orthonormal columns, and W has i.i.d. $N(0, \sigma^2)$ entries which are independent of U. Note that we impose the sparsity condition on the principal vectors $V = [v_1, \ldots, v_r]$ such that each of the principal components $v_1, \ldots, v_r$ has at most s_0 nonzero entries. We consider the high-dimensional setting where $p > n$. Note that the noise model (6) with $\mu^o = 0$ is used by Ma (2013) and Cai, Ma and Wu (2013) for sparse PCA analysis. Throughout the paper, the dimension p, the rank r, the sparsity parameter s, the spikes λ_j, and the σ^2 can be regarded as functions of the sample size n.

Regarding the mechanism of missingness, we assume that the X_{ij} are independently observed with probability p_{ij}, where the p_{ij} may depend on μ^o, D, and V in model (6). For $j \in [p]$, let

$$S_j := \{i \in [n] : (i, j) \in \Omega\} \tag{7}$$

be the set of indices i where $(i, j) \in \Omega$. We allow the probability p_{ij} to be different across i and j, and $X_{\cdot, j_1}$ and $X_{\cdot, j_2}$ may have few observed entries on the same rows (i.e., $|S_{j_1} \cap S_{j_2}| = o(n)$). Note that for this case, unbiased (or corrected) sample covariance matrices obtained by assuming that the p_{ij} are the same for all i or the size of $S_{j_1} \cap S_{j_2}$ is large enough for any j_1 and j_2, as used in Lounici (2013), Loh and Wainwright (2012), and Cai and Zhang (2016), are not available. The proposed method can estimate the principal subspace span(V) based on the partial observation X_Ω under some conditions. Note that the principal subspace span(V) is uniquely identified with the projection matrix VV^T (Cai, Ma and Wu (2013)). We use the following loss function: $L(V, \widehat{V}) = \|VV^T - \widehat{V}\widehat{V}^T\|_F^2$.

2. Proposed method.

We provide a detailed description of our two-step algorithm. In the first step, we select a significant variable set $\widehat{S} \subset [p]$. In the second step, we use the selected variables $\widehat{S}$ to estimate the principal subspace. Recall that for $j \in [p]$, $S_j := \{i \in [n] : (i, j) \in \Omega\} \subseteq [n]$ is the set of indices i where $(i, j) \in \Omega$. Let $n_j = |S_j|$ be the number of observed entries of X in the jth column. Let $\bar{n}_j := n - n_j$ be the number of unobserved entries of X in the jth column.

2.1. *Step* 1: *Using matrix factorization.*

In the first step, we select the significant variable set $\widehat{S} \subset [p]$. We normalize data X_Ω to $X_\Omega/\sqrt{n}$ and consider the following optimization problem involving group lasso penalty: for fixed $\widetilde{r}$,

$$\min_{\mu \in \mathbb{R}^p, A \in \mathbb{R}^{n \times \widetilde{r}}, A^T A = I_{\widetilde{r}}, B \in \mathbb{R}^{p \times \widetilde{r}}} F(A, B, \mu) \qquad \text{where} \tag{8}$$

$$F(A, B, \mu) = \frac{1}{2}\left\| \frac{X_\Omega}{\sqrt{n}} - (1_n \otimes \mu^T)_\Omega - (AB^T)_\Omega \right\|_F^2 + \sum_{i=1}^{p} \lambda^{(i)} \|B_{i,\cdot}\|_2. \tag{9}$$

See Lemma 1 for the theoretical lower bound of $\widetilde{r}$. Let $(\widehat{A}, \widehat{B}, \widehat{\mu})$ be the solution to (8). We will select the variables based on the row support set of $\widehat{B}$. Note that one can solve (5) without variable screening, but this step substantially reduces the computation time, especially when p is large.

REMARK 2. The matrix factorization in (8) makes us consider a low rank matrix AB^T that targets the low rank matrix $X_0/\sqrt{n} = UDV^T/\sqrt{n}$ as in (6), and this approximately reduces the row sparsity of V into B as $\|B_{i,\cdot}\|_2^2$ targets $V_{i,\cdot}D^2V_{i\cdot}^T/n$ due to $A^TA = I_{\widetilde{r}}$ and $U^TU/n \approx I_r$. Hence $\|B_{i,\cdot}\|_2$ may have a large value when $\|V_{i,\cdot}\|_2$ is large, that is, the ith variable is significant. When $\widetilde{r} = r$, A and B correspond to $U/\sqrt{n} \in \mathbb{R}^{n \times r}$ and $VD^T \in \mathbb{R}^{p \times r}$, respectively. Note that the regularization parameters in (8) must satisfy $\lambda^{(i)} \approx \sigma\sqrt{n_i/n}$ for $i \in [p]$. See Lemma 1 for detailed formula.

The low rank induced term $\|AB^T\|_*$ is not included in (8) as the rank of AB^T is already at most $\widetilde{r}$ due to matrix factorization. Although (8) is not jointly convex, it can be solved by using iterative algorithms. Consider A and μ fixed, and we solve Problem (8) for B. Note that this problem decouples into the following p separate regression problems: for $j \in [p]$,

$$\widehat{B}_{j,\cdot} = \underset{\beta \in \mathbb{R}^{\widetilde{r}}}{\operatorname{argmin}} \frac{1}{2} \left\| \frac{X_{S_j,j}}{\sqrt{n}} - \mu_j 1_{n_j} - A_{S_j,\cdot}\beta \right\|_F^2 + \lambda^{(j)} \|\beta\|_2,$$

which has a different regression matrix $A_{S_j,\cdot}$ across $j \in [p]$. By applying the general idea of the Majorization–Minimization (MM) algorithm (Lange, Hunter and Yang (2000)), we will update $B_{j,\cdot}$ using the same regression matrix for all $j \in [p]$. This reduces the computation of an update of B (Hastie et al. (2015)). Note that Hastie et al. (2015) considered fast alternating least squares in terms of matrix completion and low-rank SVD. They also utilized the majorizers to update each iterate, but their objective function is different from ours. We summarize the algorithm of the first step as follows:

```
ALGORITHM (Step 1): Obtain $\widehat{S}$ and $\widehat{\mu}$.
**Inputs**: Data matrix $X_\Omega$, initial estimates $A_0$, $B_0$, and $\mu_0$, and $k = 0$.
**Outputs**: $\widehat{A}$, $\widehat{B}$, $\widehat{\mu}$ as an estimate of minimizer of problem (8) and $\widehat{S}$.
**Repeat until** $(B_k, \mu_k)$ **converges**
1. k ← k+1.
2. $\widetilde{X} := X_\Omega/\sqrt{n} + (A_k B_k^T)_{\Omega^c} - (1_n \otimes \mu_k^T)_\Omega$, and $B_k^T \widetilde{X}^T = \widetilde{U}\widetilde{D}\widetilde{V}^T$ be the reduced SVD.
3. $A_{k+1} = \widetilde{V}\widetilde{U}^T$.
4. Update $\widetilde{X} = X_\Omega/\sqrt{n} + (A_{k+1} B_k^T)_{\Omega^c} - (1_n \otimes \mu_k^T)_\Omega$.
5. Update $B_{k+1}$ as follows: for $\ell = 1, \dots, p$, the $\ell$th row of $B_{k+1}$ satisfies
       if    $\|A_{k+1}^T \widetilde{X}_{\cdot\ell}\|_2 \le \lambda^{(\ell)}$,    then $(B_{k+1})_{\ell,\cdot} = 0$,
       else    $(B_{k+1})_{\ell,\cdot} = \frac{\|A_{k+1}^T \widetilde{X}_{\cdot,\ell}\|_2 - \lambda^{(\ell)}}{\|A_{k+1}^T \widetilde{X}_{\cdot,\ell}\|_2} A_{k+1}^T \widetilde{X}_{\cdot,\ell}$.
6. Update $\mu_{k+1} = \frac{1}{n}\sum_{i=1}^n \ddot{X}_{i,\cdot}^T$, where $\ddot{X} = (\frac{X}{\sqrt{n}} - A_{k+1} B_{k+1}^T)_\Omega + (1_n \otimes \mu_k^T)_{\Omega^c}$.
7. $\widehat{S}_{k+1} = \{\ell \in [p] \mid B_{k+1,\ell\cdot} \ne 0\}$ and update $\widehat{S} \leftarrow \widehat{S}_{k+1}$.
```

Details of the above algorithm with its stopping criterion and theoretical properties are deferred to Section 4. Note that the goal of Step 1 is to obtain the estimated mean vector $\widehat{\mu}$ and to select the significant variable set $\widehat{S}$ using the row support set of $\{B_k\}$.

REMARK 3. Johnstone and Lu (2009), Ma (2013), and Cai, Ma and Wu (2013) performed variable screening based on column norms of X. Our numerical examples demonstrate that Step 1 serves a similar role in principle and has similar theoretical properties but performs better than the existing methods.

2.2. *Step* 2: *Solve a nuclear-norm-regularized problem.* Recall that $\widehat{\mu}$ is the estimate of μ and $\widehat{S} \subset [p]$ is the selected set obtained from Step 1. Let $\widehat{s} := |\widehat{S}|$. In the second step, we only utilize the set of variables in $\widehat{S}$ to estimate the principal subspace. First, we estimate $X_{0,\Omega}$ by solving the following problem, which is a special version of (5): for $\gamma_d > 0$,

$$\widehat{X}_1 = \underset{Z \in \mathbb{R}^{n \times \widehat{s}}}{\operatorname{argmin}} \| Z_\Omega + (1_n \otimes \hat{\mu}_{\widehat{S}})_\Omega - (X_{\cdot,\widehat{S}})_\Omega / \sqrt{n} \|_F^2 + \gamma_d \|Z\|_*. \tag{10}$$

Then, we solve the following problem, which is also a special version of (5): for $\zeta_d > 0$,

$$\widehat{X}_2 = \underset{Z \in \mathbb{R}^{n \times \widehat{s}}, Z_\Omega = (\hat{X}_1)_\Omega}{\operatorname{argmin}} \|Z\|_* + \zeta_d \|Z_{\Omega^c}\|_F^2. \tag{11}$$

See Theorem 1 for the theoretical conditions on ζ_d and γ_d. In (10), we focus on estimating $X_{0,\Omega}$, and (11) utilizes this estimate to update the remaining part of X_0. We found that the two steps (10)–(11) are more efficient than estimating X_0 in an one step in that it provides a more tight bound on $\|\widehat{X}_2 - X_0\|$ as well as empirically produces an accurate estimator. Moreover, solving (10) and (11) require the same time complexity as presented in Section 1 of the Supplementary Material (Park and Zhao (2019)), and it does not require much time to solve even for large scale data as in Table 8.

Then we define $\widehat{X} \in \mathbb{R}^{n \times p}$ satisfying $\widehat{X}_{\cdot,\widehat{S}} = \widehat{X}_2$ and $\widehat{X}_{\cdot,\widehat{S}^c} = 0$. For the target dimension r_0 of a principal subspace, we construct $\widehat{V}_{r_0} \in \mathbb{R}^{p \times r_0}$ whose columns have the first r_0 right eigenvectors of $\widehat{X}$. We summarize the proposed two-step algorithm as follows:

ALGORITHM Two-step procedure.

Inputs: Data matrix X_Ω and r_0, which is the target dimension of the principal component space of interest.

Outputs: principal components $\widehat{V} \in \mathbb{R}^{p \times r_0}$.

Steps:

```
Step 1. Obtain $\widehat{A}$, $\widehat{B}$, $\widehat{\mu}$, and $\widehat{S}$ from Step 1.
Step 2(a). Use $\widehat{\mu}$ and $X_{\cdot,\widehat{S}}$ to solve (10) and obtain $\widehat{X}_1 \in \mathbb{R}^{n \times \widehat{s}}$.
Step 2(b). Use obtained $(\hat{X}_1)_\Omega$ to solve (11) and obtain $\widehat{X}_2 \in \mathbb{R}^{n \times \widehat{s}}$.
Step 2(c). Construct $\widehat{X} \in \mathbb{R}^{n \times p}$ such that $\widehat{X}_{\cdot,\widehat{S}} = \widehat{X}_2$ and $\widehat{X}_{\cdot,\widehat{S}^c} = 0$.
Step 2(d). $\widehat{V}_{r_0} \in \mathbb{R}^{p \times r_0}$ consists of the first $r_0$ right eigenvectors of $\widehat{X}$.
```

REMARK 4. In our implementation, we solve (10)–(11) with an additional constraint $\|Z\|_* \le \rho$ for some $\rho > 0$ by using the composite gradient descent algorithm. Since they are strongly convex, it enjoys a computational convergence as that in Agarwal, Negahban and Wainwright (2012). This additional side constraint is included to ensure good behavior of the algorithm in the first few iterations (Agarwal, Negahban and Wainwright (2012)). We use $\widehat{A}\widehat{B}^T_{\widehat{S},\cdot}$ as an initial input in the composite gradient descent algorithm, which empirically provides a good output rather than using an arbitrary matrix as an initial input. We terminate the composite gradient descent update after 500 iterations based on empirical observation that the iterates are stable after 100 runs. Analysis of convergence rates of the composite gradient descent algorithm (Agarwal, Negahban and Wainwright (2012)) in our setting is left open for future research.

2.3. *Choosing the penalty parameters.* In Step 1, based on Lemma 1, we choose $\tilde{r} = [n/\log p]$ and $(\lambda^{(j)})^2 = \frac{n_j}{n}(\hat{\sigma}^2 + \hat{\sigma}^2\sqrt{\frac{20\log p}{n_j}})$ for $j = 1, \dots, p$, where $\widehat{\sigma}^2 = \text{median}\{\|X_{S_j,j} - \widehat{\mu}_j 1_{n_j}\|_2^2/n_j : j \in [p]\}$ is the estimate of σ^2 (Johnstone and Lu (2009), Ma (2013)).

In Step 2, we solve (10) and (11) by using the composite gradient descent algorithm (Agarwal, Negahban and Wainwright (2012)) with regularization parameters ζ_d, γ_d, and ρ, where ρ is stated in Remark 4. Based on the theoretical rate of γ_d as in Theorem 1, we choose γ_d using simulation as follows: for a fixed $\zeta_d > 0$,

$$\gamma_d = \max_{1 \le k \le 100} \|W_\Omega^{(k)}\|_2/\sqrt{n}, \tag{12}$$

where $W^{(k)} \in \mathbb{R}^{n \times \hat{s}}$ for $k = 1, \dots, 100$ are independent of each other and each has i.i.d. $N(0, \hat{\sigma}^2)$ entries, and $\widehat{A}$ and $\widehat{B}$ are the output obtained from Step 1. We choose ρ as

$$\rho = 2\|\widehat{A}\widehat{B}^T_{\widehat{S},\cdot}\|_*. \tag{13}$$

Similarly, we choose $\zeta_d = \sqrt{n\hat{s}}/(2\,\text{rank}(\widehat{B}_{\widehat{S},\cdot})\|\widehat{A}\widehat{B}^T_{\widehat{S},\cdot}\|_2)$ based on the fact that $\text{rank}(\widehat{A}\widehat{B}^T_{\widehat{S},\cdot}) = \text{rank}(\widehat{B}_{\widehat{S},\cdot})$. Those choices of the regularization parameters do not give the best results for any given cases, but they lead to good empirical results in various settings and can help us understand how the proposed method performs with reasonable choices of these tuning parameters.

3. Theoretical properties.

In this section, we investigate the theoretical properties of the proposed principal subspace estimator. As stated in Section 1.3, the dimension p, the rank r, the sparsity parameter s, the spikes λ_j, and the σ^2 can be regarded as functions of the sample size n.

3.1. *Main results.* We define the following key quantities: $n_+ = \max_{j \in S} n_j$ and $p_{\min} = \min_{i \in [n], j \in S} p_{ij}$. The following Lemma 1 shows the exact row selection property of $\widehat{S}$ with high probability.

LEMMA 1. *Suppose Conditions* S1–S4 *in the Supplementary Material* (*Park and Zhao* (2019)). *Let* $(\widehat{A}, \widehat{B})$ *be the solution to* (8) *with* $\widetilde{r} \ge s$ *and* $(\lambda^{(i)})^2 = \frac{n_i}{n}(\sigma^2 + \sigma^2\sqrt{\frac{20\log p}{n_i}})$ *for* $i \in [p]$. *Let* $\widehat{S} := \{j \in [p] : \widehat{B}_{j,\cdot} \neq 0\}$. *Then, with probability* $1 - 3/p$, $\hat{S} = S$.

The following theorem shows the statistical convergence of the estimator under some conditions. We include the required assumptions in the Supplementary Material (Park and Zhao (2019)).

THEOREM 1. *For $r_0 \le r$, let $V_{r_0} := V_{\cdot,[r_0]} \in \mathbb{R}^{p\times r_0}$. Suppose all conditions in Lemma 1 hold. Let $\widehat{V}_{r_0} \in \mathbb{R}^{p\times r_0}$ be the proposed estimate of V_{r_0} with $\zeta_d \sim \sqrt{ns}/(2r\lambda_1)$ and $\gamma_d \sim \|W_\Omega\|_2/\sqrt{n}$. Then, with probability tending to one,*

$$\|\widehat{V}_{r_0}\widehat{V}_{r_0}^T - V_{r_0}V_{r_0}^T\|_F \lesssim \frac{\log n}{\sqrt{p_{\min}\lambda_1}}\big[n^{\frac{\varsigma-1}{2}}(1 \vee \sigma n^{\varsigma})\big] + \frac{\sqrt{r}}{\sqrt{n}}\sqrt{r \vee \log n}. \tag{14}$$

Note that the upper bound in (14) mainly consists of two terms. The first term is induced by controlling missing entries and the noise of the sparse PCA model, The upper bound in (14) is $o(1)$, provided that ς is some small constant and $p_{\min}$ is bounded below by some negligible value satisfying $o(1)$, as in Condition S1 of the Supplementary Material (Park and Zhao (2019)), showing that the proposed method for principal subspace could accurately estimate the underlying principal subspace.

4. Computational algorithm. In this section, we present details of the algorithm (Step 1) given in Section 2 with its theoretical properties. For fixed $\tilde{r}$, recall the function $F(A, B, \mu)$, which is the objective function of $A \in \mathbb{R}^{n\times\tilde{r}}$, $B \in \mathbb{R}^{p\times\tilde{r}}$, and $\mu \in \mathbb{R}^p$, as in (9). For fixed A, B, and μ, define the functions $Q_A(\widetilde{A}|A, B, \mu)$, $Q_B(\widetilde{B}|A, B, \mu)$, and $Q_\mu(\widetilde{\mu}|A, B, \mu)$ as

$$\begin{aligned}
Q_A(\widetilde{A}|A, B, \mu) &= \frac{1}{2}\left\|\left(\frac{X}{\sqrt{n}} - \widetilde{A}B^T\right)_\Omega + (AB^T - \widetilde{A}B^T)_{\Omega^c} - (1_n \otimes \mu^T)_\Omega\right\|_F^2 \\
&\quad + \sum_{i=1}^p \lambda^{(i)}\|B_{i,\cdot}\|_2, \\
Q_B(\widetilde{B}|A, B, \mu) &= \frac{1}{2}\left\|\left(\frac{X}{\sqrt{n}} - A\widetilde{B}^T\right)_\Omega + (AB^T - A\widetilde{B}^T)_{\Omega^c} - (1_n \otimes \mu^T)_\Omega\right\|_F^2 \\
&\quad + \sum_{i=1}^p \lambda^{(i)}\|\widetilde{B}_{i,\cdot}\|_2, \\
Q_\mu(\widetilde{\mu}|A, B, \mu) &= \frac{1}{2}\left\|\left(\frac{X}{\sqrt{n}} - AB^T\right)_\Omega - (1_n \otimes \widetilde{\mu}^T)_\Omega + (1_n \otimes (\mu - \widetilde{\mu})^T)_{\Omega^c}\right\|_F^2 \\
&\quad + \sum_{i=1}^p \lambda^{(i)}\|B_{i,\cdot}\|_2.
\end{aligned}$$

We can easily check that $Q_A(\widetilde{A}|A, B, \mu)$, $Q_B(\widetilde{B}|A, B, \mu)$, and $Q_\mu(\widetilde{\mu}|A, B, \mu)$ are majorizers of $F(\widetilde{A}, B, \mu)$, $F(A, \widetilde{B}, \mu)$, and $F(A, B, \widetilde{\mu})$, respectively. That is, it holds that

$$F(\widetilde{A}, B, \mu) \leq Q_A(\widetilde{A}|A, B, \mu), F(A, \widetilde{B}, \mu) \leq Q_B(\widetilde{B}|A, B, \mu), F(A, B, \widetilde{\mu}) \leq Q_\mu(\widetilde{\mu}|A, B, \mu),$$

$$F(A, B, \mu) = Q_A(A|A, B, \mu) = Q_B(B|A, B, \mu) = Q_\mu(\mu|A, B, \mu).$$

We update $\{(A_k, B_k, \mu_k)\}$ by the following iterative procedure:

$$A_{k+1} = \operatorname*{argmin}_{A \in \mathbb{R}^{n \times \tilde{r}} : A^T A = I_{\tilde{r}}} Q_A(A|A_k, B_k, \mu_k), B_{k+1} = \operatorname*{argmin}_{B \in \mathbb{R}^{p \times \tilde{r}}} Q_B(B|A_{k+1}, B_k, \mu_k),$$

$$\mu_{k+1} = \operatorname*{argmin}_{\mu \in \mathbb{R}^p} Q_\mu(\mu|A_{k+1}, B_{k+1}, \mu_k).$$

Note that one can update μ_k without using the majorizer Q_μ, but our empirical results show that we can obtain a more stable output with the majorizer and this also yields convergence of μ_k in terms of $\|\mu_{k+1} - \mu_k\|_2$ as in Lemma 5, which can not be obtained without the majorizer. The following lemmas show that we can easily solve each iteration in Step 1.

LEMMA 2. *Let* $\widehat{A} = \operatorname{argmin}_{\widetilde{A} \in \mathbb{R}^{n \times \tilde{r}} : \widetilde{A}^T \widetilde{A} = I_{\tilde{r}}} Q_A(\widetilde{A}|A, B, \mu)$. *Let* $\widetilde{X} = X_\Omega / \sqrt{n} + (AB^T)_{\Omega^c} - (1_n \otimes \mu^T)_\Omega$. *Consider the reduced SVD,* $B^T \widetilde{X}^T = \widetilde{U} \widetilde{D} \widetilde{V}^T$, *where* $\widetilde{U}$ *and* $\widetilde{V}$ *have orthonormal columns, respectively. Then* $\widehat{A} = \widetilde{V} \widetilde{U}^T$.

LEMMA 3. *Let* $\widehat{B} = \operatorname{argmin}_{\widetilde{B} \in \mathbb{R}^{p \times \tilde{r}}} Q_B(\widetilde{B}|A, B, \mu)$. *Then, for* $\ell = 1, \ldots, p$, *the* ℓ*th row of* $\widehat{B}$ *satisfies the following*:

$$\text{if } \|A^T \widetilde{X}_{\cdot,\ell}\|_2 \leq \lambda^{(\ell)}, \qquad \text{then } \widehat{B}_{\ell,\cdot} = 0,$$

$$\text{else } \widehat{B}_{\ell,\cdot} = \frac{\|A^T \widetilde{X}_{\cdot,\ell}\|_2 - \lambda^{(\ell)}}{\|A^T \widetilde{X}_{\cdot,\ell}\|_2} A^T \widetilde{X}_{\cdot,\ell}.$$

LEMMA 4. *Let* $\widehat{\mu} = \operatorname{argmin}_{\widetilde{\mu} \in \mathbb{R}^p} Q_\mu(\widetilde{\mu}|A, B, \mu)$. *Let* $\ddot{X} = (\frac{X}{\sqrt{n}} - AB^T)_\Omega + (1_n \otimes \mu^T)_{\Omega^c}$. *Then* $\widehat{\mu} = \frac{1}{n} \sum_{i=1}^n \ddot{X}_{i,\cdot}^T$ *is the sample mean vector of* $\ddot{X}$.

By Lemmas 2–4, we obtain the algorithm (Step 1) given in Section 2.1. The following Lemma 5 establishes a convergence of the algorithm (Step 1) and shows that the iterates $\{B_k\}_{k=1,2,\ldots,}$ and $\{\mu_k\}_{k=1,2,\ldots,}$ converge.

LEMMA 5. *Let* $(A_k, B_k, \mu_k), k \geq 0$ *be the sequence generated by the algorithm in Step* 1. *Then, the sequence of* $F(A_k, B_k, \mu_k)$ *is decreasing and converges to some positive constant* F_L, *and*

$$\eta_k := \frac{1}{2} \|B_{k+1} - B_k\|_F^2 + n \|\mu_{k+1} - \mu_k\|_2^2 \to 0 \qquad \text{as } k \to \infty.$$

Specifically, $\min_{k \in [T]} \eta_k \leq \frac{F(A_0, B_0, \mu_0) - F_L}{T+1}$.

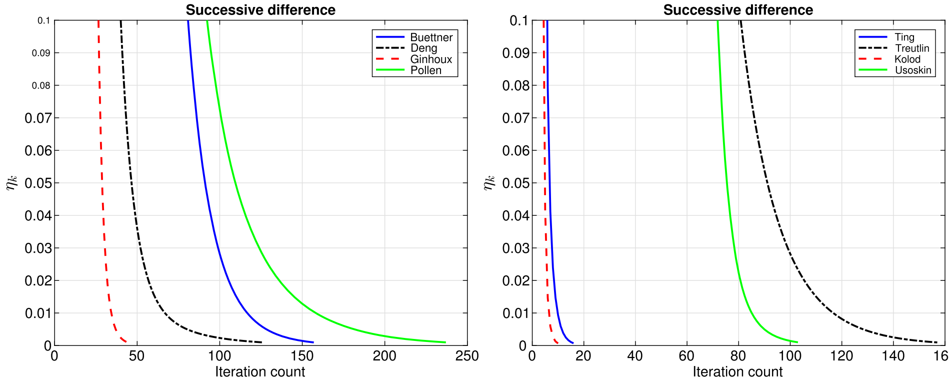

FIG. 1. *Plots of the successive difference η_k as iteration increases for the selected eight scRNA-seq data sets. The left and right plots consider the four scRNA-seq data sets, respectively.*

All technical proofs are presented in the Supplementary Material (Park and Zhao (2019)). Based on results of Lemma 5, one can use the stopping criterion $\eta_k \le \varepsilon$ for some small constant $\varepsilon > 0$. We use $\varepsilon = 10^{-4}$ in our implementation. Regarding the computational convergence rate, we see that the initial objective function value of F in (9) is $F(A_0, B_0, \mu_0) = \frac{1}{2}\|\frac{X_\Omega}{\sqrt{n}} - (1_n \otimes \mu_0^T)\|_F^2$, where $B_0 = 0$ and μ_0 is the sample mean vector of $X_\Omega/\sqrt{n}$. Thus, we have $F(A_0, B_0, \mu_0) \lesssim \|\frac{1}{\sqrt{n}} UDV^T\|_F^2 \lesssim \mathrm{tr}(VD^2V^T) = \mathrm{tr}(D^2) \lesssim r\lambda_1$. Hence the number of iterates of the algorithm is $O(r\lambda_1)$ because the algorithm converges at a rate of $O(F(A_0, B_0, \mu_0)/K)$, where K denotes the number of iterations of the algorithm.

Figure 1 shows the successive difference η_k as iterates increase. We observe that the stopping criterion is satisfied within 250 iterates in all the cases, and the successive difference converges to zero as the iterate count k increases that corresponds to the results of Lemma 5.

5. Simulation: First study. In this section, we compare our principal subspace estimate with other methods. For comparisons, we consider the following PCA estimators: the standard PCA (“PCA”); the diagonal thresholding sparse PCA (“DTSPCA”) by Johnstone and Lu (2009); the iterative thresholding sparse PCA with hard thresholding (“ITSPCA”) by Ma (2013); the correlation augmented sparse PCA (“CORSPCA”) by Nadler (2009); and the augmented sparse PCA (“AUGSPCA”) by Paul and Johnstone (2007). For these PCA estimators, we use the imputed data by replacing the missing values with zeros following Xu and Su (2015), Wang et al. (2017), and Shao and Höfer (2017), because these PCA methods are not designed to take into account missing values in data. We also consider the weighted PCA methods that are designed to take into account dropouts in data: “EMPCA” by Bailey (2012) and “WPCA” by Delchambre (2014). In the implementation, we use the following regularization parameters: $\alpha_n = 3\sqrt{\log p/n}$ for “DTSPCA”; $\gamma = 0.75$ for “ITSPCA”; $\kappa = 2.1$ for “AUGSPCA” and $\xi = 0$ for “WPCA.” For the other methods, we use the regularization parameters suggested in the original papers.

For the performance measurement, we record $\|\widehat{V}\widehat{V}^T - VV^T\|$ for various estimates $\widehat{V}$ of the principal coefficient V. Denote the row support set by $S = \{j \in [p] : V_{j,\cdot} \neq 0\}$ and $|S| = s$. We consider the two different PCA models based on constructing the matrix V. In the first model, the matrix V is obtained by orthonormalizing a $p \times r$ matrix M, where

$$M_{i,j} \sim N(0, i^3) \quad \text{for } i = 1, \dots, s \quad \text{and} \quad M_{i,\cdot} = 0 \qquad \text{for } i > s \tag{15}$$

so that the norms of the sorted rows in V have rapid decay. The λ_i take r equally-spaced values such that $\lambda_r = 10$ and $\lambda_1 = 30$. In the second model, V is obtained by orthonormalizing a $p \times r$ matrix M, where

$$M_{i,j} \sim \text{Uniform}(0, 1) \qquad \text{for } i = 1, \dots, s \quad \text{and} \quad M_{i,\cdot} = 0 \qquad \text{for } i > s \tag{16}$$

so that the norms of nonzero rows in V have similar values.

We introduce dropout events according to the eight different generating functions $g(x)$ shown in Figure 2, where $g(\cdot)$ is symmetric about the origin and monotonically decreasing over $[0, \infty)$. We observe $\widetilde{X}$ as

$$\widetilde{X}_{ij} = X_{ij}\delta_{ij}, \qquad \text{where } \delta_{ij} \sim \text{Bernoulli}\bigl(1 - g(X_{ij})\bigr),$$

where the δ_{ij} are independent Bernoulli random variables, that is, X_{ij} is independently observed with probability $1 - g(X_{ij})$. We consider the case when $n = 1000$, $p = 2000$, $r = 10$, $\sigma = 1$, and $s \in \{30, 60\}$.

Figure 2 displays the eight different monotone function $g(x)$’s: the first two functions have Gaussian function form $\exp(-\lambda x^2)$, which was used as a dropout event generating function for scRNA-seq data (Pierson and Yau (2015), Wang et al. (2017)); the third and fourth functions are constants, that is, missing occurs completely at random; the fifth and sixth functions have double exponential function form $\exp(-\lambda|x|)$; and the seventh and eighth consider the cases that X_{ij}’s are always unobserved when they are less than a certain threshold level and observed with probability 0.7 when they are greater than the threshold level.

In Figures 3–4, we compare the proposed estimator with the other PCA methods for the missing cases 1, 3, 5, and 7 when $s = 30$. For the other settings and missing cases, see Section 10 of the Supplementary Material (Park and Zhao (2019)). The performance measurements are averaged over 200 repetitions. Across all the cases, the proposed sparse PCA outperforms the other methods in terms of estimating the principal subspace, while it generally takes a few seconds and comparable to those of other estimators.

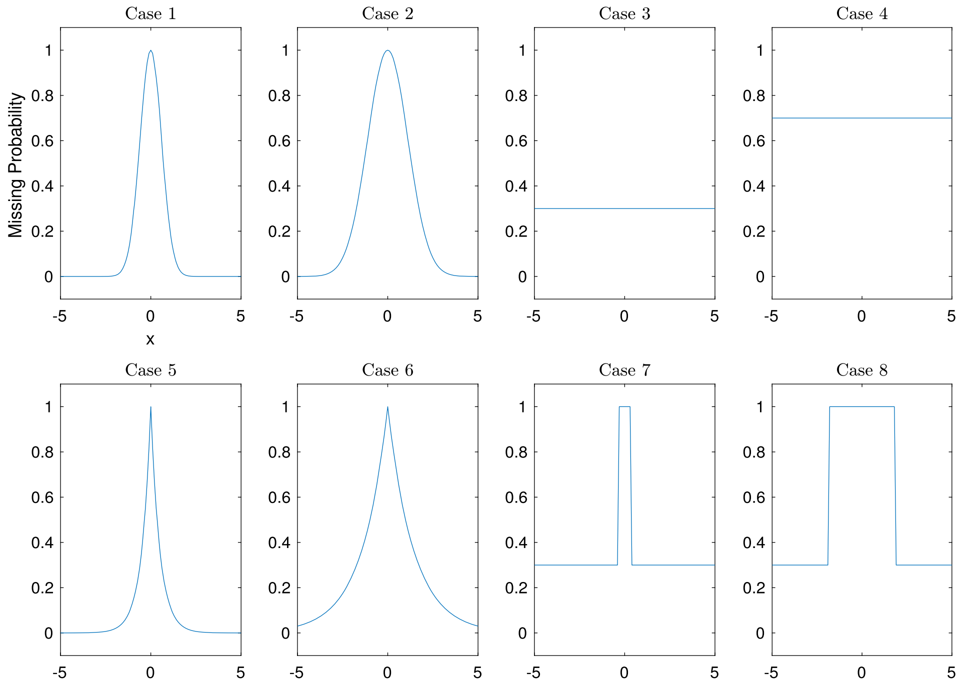

FIG. 2. *Eight different dropout event generating functions*: $g_1(x) = \exp(-1.5x^2)$, $g_2(x) = \exp(-0.5x^2)$, $g_3(x) = 0.3$, $g_4(x) = 0.7$, $g_5(x) = \exp(-2|x|)$, $g_6(x) = \exp(-0.7|x|)$, $g_7(x) = 1_{\{|x|<0.2\}} + 0.3 \times 1_{\{|x|\geq 0.2\}}$, *and* $g_8(x) = 1_{\{|x|<1\}} + 0.3 \times 1_{\{|x|\geq 1\}}$. *These eight missing cases generally yield the following missing proportions for our simulated datasets*: 0.85, 0.93, 0.2, 0.4, 0.7, 0.86, 0.33, *and* 0.39, *respectively.*

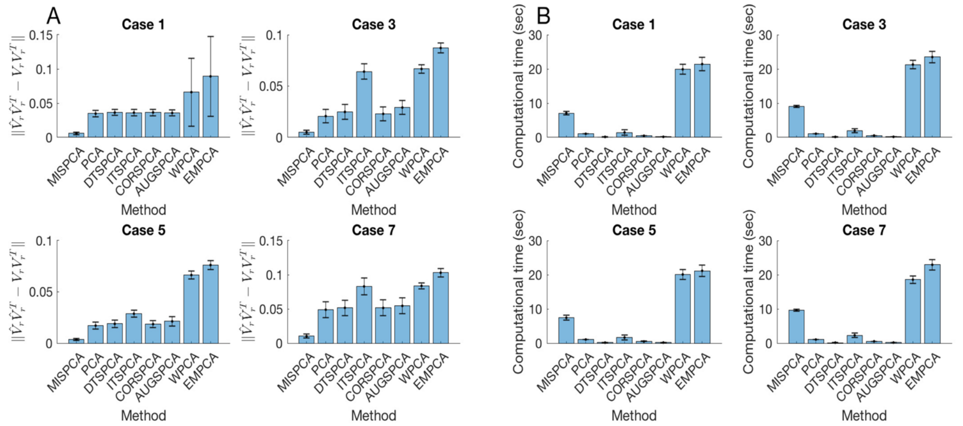

FIG. 3. (A) *The bar graph with average of the performance measurements* $\|\widehat{V}_r \widehat{V}_r^T - V_r V_r^T\|_2$ *and its one standard deviation over* 200 *repetitions.* (B) *The bar graph with average of computational time* (*seconds*) *and its one standard deviation. We consider the missing cases* 1, 3, 5, *and* 7 *when* $n = 1000$, $p = 2000$, $r = 10$, $\sigma = 1$, $s = 30$, *and* V *follows* (15).

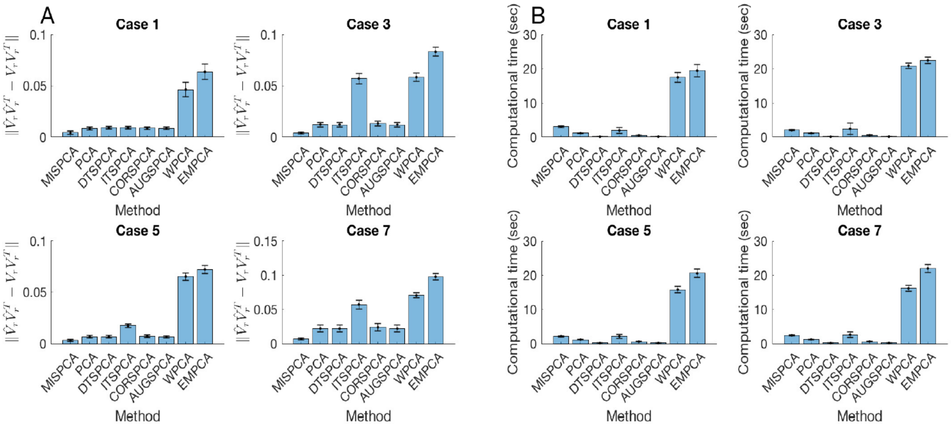

FIG. 4. (A) *The bar graph with average of the performance measurements* $\|\widehat{V}_r \widehat{V}_r^T - V_r V_r^T\|_2$ *and its one standard deviation over* 200 *repetitions*. (B) *The bar graph with average of computational time* (*seconds*) *and its one standard deviation. We consider the missing cases* 1,3,5, *and* 7 *when* $n = 1000$, $p = 2000$, $r = 10$, $\sigma = 1$, $s = 30$, *and V follows* (16).

6. Simulation: Second study. In this section, we assess the performances of the proposed method in terms of clustering, as will be considered in Section 7, by simulation. In the experiments, we use two types of simulation data. We generate the first simulation model using the following four steps as in Park and Zhao (2018), shown in (A1)–(A4) below. In the first step, C points are generated on a circle in the two-dimensional latent space, where each point is considered to be the center of one cluster. The n points are generated by adding independent noise to the center of the corresponding cluster. In the second step, we project the previously generated two-dimensional data to a p-dimensional space, which represents gene expression data. In the third step, we simulate a noisy gene expression data by adding independent Gaussian noise. In the last step, we introduce a dropout event such that each entry is independently observed with a certain probability. See Section 7 of the Supplementary Material (Park and Zhao (2019)) for details of the simulation model. In the simulation, we fix $n = 500$, $p = 1000$, $q = 50$, $C = 5$, $d = 10$, $\sigma^2 = 1$, $\gamma = 0.01$, and $\sigma_l^2 = 1$.

In the second simulation model, we generate the data using Gaussian mixture model. To distinguish different cell types, it is likely that only some genes are informative, and noninformative and highly noisy genes can increase the difficulty of identifying cell types. Under this context, we use a few attributes to distinguish the clustering labels. We generate the simulation model based on the Gaussian mixture model. See Section 7 of the Supplementary Material (Park and Zhao (2019)) for details of the simulation model. We consider a high-dimensional setting $n = 1000$, $p = 2000$, $q = 50$, $C = 5$, $\sigma_{\text{signal}} \in \{12, 16, 20\}$, and $\gamma = 0.01$.

We also consider the other state-of-the-art dimension reduction methods used to analyze scRNA-seq data: "ZIFA" by Pierson and Yau (2015) and "CIDR" by Lin,

TABLE 1
Simulation model 1: *Average* (*one standard deviation*) *of NMI and ARI values over* 200 *repetitions for various* σ_{signal}. *The average missing rate is* 65%

	NMI			**ARI**		
Method	$\sigma_{\text{signal}} = 12$	$\sigma_{\text{signal}} = 16$	$\sigma_{\text{signal}} = 20$	$\sigma_{\text{signal}} = 12$	$\sigma_{\text{signal}} = 16$	$\sigma_{\text{signal}} = 20$
MIS-PCA	0.85 (0.05)	0.95 (0.04)	0.99 (0.01)	0.87 (0.05)	0.99 (0.03)	0.99 (0.01)
PCA	0.47 (0.07)	0.83 (0.03)	0.96 (0.02)	0.47 (0.06)	0.86 (0.04)	0.97 (0.02)
DTSPCA	0.59 (0.07)	0.87 (0.03)	0.97 (0.02)	0.60(0.07)	0.85 (0.05)	0.94 (0.02)
ITSPCA-Hard	0.62 (0.05)	0.84 (0.04)	0.94 (0.03)	0.69 (0.06)	0.91 (0.03)	0.97 (0.02)
ITSPCA-Soft	0.70 (0.05)	0.90 (0.04)	0.97 (0.02)	0.72 (0.06)	0.94 (0.04)	0.96 (0.02)
CORSPCA	0.69 (0.06)	0.90 (0.03)	0.97 (0.03)	0.70 (0.06)	0.91 (0.04)	0.97 (0.01)
AUGSPCA	0.58 (0.05)	0.83 (0.03)	0.97 (0.02)	0.47 (0.07)	0.87 (0.03)	0.96 (0.02)
EMPCA	0.78 (0.07)	0.90 (0.04)	0.95 (0.02)	0.80 (0.07)	0.93 (0.04)	0.96 (0.03)
WPCA	0.80 (0.07)	0.91 (0.04)	0.96 (0.03)	0.82 (0.07)	0.93 (0.04)	0.97 (0.03)
CIDR	0.79 (0.07)	0.93 (0.03)	0.97 (0.02)	0.82 (0.06)	0.92 (0.04)	0.96 (0.01)
ZIFA	0.73 (0.09)	0.92 (0.05)	0.96 (0.02)	0.73 (0.09)	0.91(0.03)	0.96 (0.03)

Troup and Ho (2017). Note that "ZIFA" implements a probabilistic PCA method by incorporating a zero inflated model to account for dropout characteristics. "CIDR" is a novel clustering method through imputation and dimensionality reduction that uses a simple implicit imputation approach to alleviate the impact of dropouts in scRNA-seq data. For the proposed method "MIS-PCA," we use $\widehat{X}\widehat{P} \in \mathbb{R}^{n\times r_0}$ as a principal component score, where $\widehat{X} \in \mathbb{R}^{n\times p}$ and $\widehat{P} \in \mathbb{R}^{p\times r_0}$ are the output matrix and principal components obtained from Step 2, respectively. For the number of iterations needed for convergence of "MIS-PCA," see Figure 1. We use the true cluster number for the target dimension r_0 as well as the target cluster number.

We use three performance measures to evaluate the consistency between the obtained clustering and the true labels: Normalized Mutual Information (NMI) (Strehl and Ghosh (2003)), Adjusted Rand Index (ARI) (Wagner and Wagner (2007)), and Purity. NMI and Purity take on values between zero and one, but ARI can have negative values. These metrics measure the concordance of two clustering labels such that higher value refers to higher concordance. For details of these metrics, see Section 9 of the Supplementary Material (Park and Zhao (2019)).

Table 1 and Table S1 in the Supplementary Material (Park and Zhao (2019)) show the summary of NMI, ARI, and Purity measures for different σ_{signal} (i.e., signal to ratio) levels when the first simulation model is considered. We observe that the performances become better as the σ_{signal} level is increased. For all the cases, the proposed method "MIS-PCA" outperforms the other competitors in terms of clustering samples. The results for the second simulation model (Table 2 and Table S2 in the Supplementary Material (Park and Zhao (2019))) are also consistent with those of the first simulation model.

TABLE 2

Simulation model 2: *Average* (*one standard deviation*) *of NMI and ARI values over* 200 *repetitions for various* σ_{signal}. *The average missing rate is* 70%

	NMI			**ARI**		
Method	$\sigma_{\text{signal}} = 12$	$\sigma_{\text{signal}} = 16$	$\sigma_{\text{signal}} = 20$	$\sigma_{\text{signal}} = 12$	$\sigma_{\text{signal}} = 16$	$\sigma_{\text{signal}} = 20$
MIS-PCA	0.83 (0.05)	0.95 (0.01)	0.99 (0.00)	0.84 (0.06)	0.98 (0.01)	0.99 (0.00)
PCA	0.43 (0.06)	0.80 (0.03)	0.94 (0.02)	0.45 (0.08)	0.83 (0.03)	0.95 (0.01)
DTSPCA	0.55 (0.07)	0.83 (0.03)	0.94 (0.02)	0.57(0.08)	0.85 (0.04)	0.96 (0.02)
ITSPCA-Hard	0.63 (0.03)	0.85 (0.02)	0.95 (0.02)	0.66 (0.04)	0.88 (0.02)	0.96 (0.01)
ITSPCA-Soft	0.68 (0.03)	0.88 (0.02)	0.96 (0.01)	0.71 (0.04)	0.91 (0.02)	0.97 (0.01)
CORSPCA	0.64 (0.06)	0.87 (0.03)	0.95 (0.02)	0.67 (0.07)	0.89 (0.03)	0.96 (0.01)
AUGSPCA	0.43 (0.06)	0.80 (0.03)	0.93 (0.02)	0.45 (0.07)	0.83 (0.03)	0.95 (0.02)
EMPCA	0.75 (0.03)	0.88 (0.03)	0.91 (0.04)	0.78 (0.04)	0.91 (0.03)	0.93 (0.03)
WPCA	0.78 (0.04)	0.88 (0.03)	0.93 (0.04)	0.80 (0.05)	0.93 (0.03)	0.95 (0.02)
CIDR	0.74 (0.07)	0.90 (0.02)	0.94 (0.03)	0.78 (0.06)	0.90 (0.02)	0.95 (0.02)
ZIFA	0.68 (0.08)	0.89 (0.03)	0.92 (0.04)	0.71 (0.08)	0.89 (0.03)	0.94 (0.03)

Furthermore, we evaluate the sensitivity of the clustering results for these methods with respect to departures from assumptions in some interesting missing data patterns. We first consider the setting, where p_{ij} are generated from the Uniform distribution, that is, $p_{ij} \sim \text{unif}(p_{\min}, p_{\max})$, where $p_{\max} = (p_{\min} + 0.3) \wedge 1$. That is, the missing mechanism in (A4) of the aforementioned two simulation models is replaced with the Uniform distribution. Note that our theoretical results imply that it becomes more challenging as $p_{\min}$ approaches to zero given that the other parameters in the model are fixed. We consider $p_{\min} \in \{0.01, 0.1, 0.3, 0.5, 0.7\}$ to evaluate the sensitivity to the minimum observed probability. Tables 3–4 show the sensitivity of the clustering results (NMI) with respect to $p_{\min}$ when Simulation models 1 and 2 are considered, respectively. It is seen that the accuracy of clustering results is getting worse as $p_{\min}$ approaches zero, as we expected, but "MIS-PCA" is less sensitive to the change of $p_{\min}$ than other methods. We found that "MIS-PCA" and "CIDR" generally acheive more accurate clustering results compared to other methods, while the variation of clustering results of "MIS-PCA" is less than those of other methods for each simulation setting, suggesting that "MIS-PCA" provides more stable results.

Second, we consider the missing case, $p_{ij} = 1 - \exp(-\gamma X_{ij}^2)$, where $\gamma > 0$ represents the exponential decay parameter. Note that this missing pattern is known to be a possible dropout event generating function for normalized scRNA-seq data X (Pierson and Yau (2015), Wang et al. (2017)). We consider the sensitivity of the clustering results to changes in the decay parameter γ. Specifically, we consider $\gamma \in [10^{-5}, \dots, 1]$ in (A4) of the two simulation models. Tables S3–S4 of the Supplementary Material (Park and Zhao (2019)) show the sensitivity of the clustering results (NMI) with respect to the decay parameter when Simulation models

TABLE 3

Sensitivity analysis when Simulation model 1 *with* $p_{ij} \sim \text{unif}(p_{\min}, p_{\max})$, *where* $p_{\max} = p_{\min} + 0.3$ *and* $\sigma_{\text{signal}} = 12$ *is considered*: *Average* (*one standard deviation*) *of NMI values over* 200 *repetitions for various* γ

Method	$p_{\min} = 0.01$	$p_{\min} = 0.1$	$p_{\min} = 0.3$	$p_{\min} = 0.5$	$p_{\min} = 0.7$
MIS-PCA	0.39 (0.12)	0.55 (0.10)	0.69 (0.07)	0.80 (0.05)	0.92 (0.03)
PCA	0.37 (0.13)	0.50 (0.12)	0.61 (0.09)	0.68 (0.07)	0.81 (0.05)
DTSPCA	0.37 (0.15)	0.51 (0.14)	0.64 (0.10)	0.73 (0.08)	0.83 (0.05)
ITSPCA-Hard	0.35 (0.19)	0.50 (0.13)	0.65 (0.11)	0.75 (0.08)	0.83 (0.07)
ITSPCA-Soft	0.36 (0.17)	0.50 (0.15)	0.64 (0.10)	0.73 (0.07)	0.86 (0.05)
CORSPCA	0.36 (0.16)	0.48 (0.13)	0.65 (0.10)	0.76 (0.08)	0.85 (0.08)
AUGSPCA	0.37 (0.20)	0.49 (0.17)	0.66 (0.10)	0.75 (0.07)	0.85 (0.06)
EMPCA	0.38 (0.19)	0.50 (0.15)	0.64 (0.14)	0.75 (0.09)	0.86 (0.06)
WPCA	0.37 (0.18)	0.51 (0.14)	0.62 (0.12)	0.73 (0.09)	0.84 (0.07)
CIDR	0.37 (0.16)	0.53 (0.14)	0.67 (0.10)	0.78 (0.07)	0.92 (0.05)
ZIFA	0.37 (0.20)	0.52 (0.16)	0.66 (0.14)	0.77 (0.12)	0.88 (0.08)

1 and 2 are considered, respectively. We observe that “MIS-PCA” consistently produced more accurate clustering results with less variations compared to other methods. This suggests that “MIS-PCA” may be favorable when the underlying missing probability satisfies $p_{ij} = 1 - \exp(-\gamma X_{ij}^2)$. This may be due to the effects of imputing X_{Ω^c} using the proposed optimization, which actually updates X_{Ω^c} by considering the conditional probability that the entries in Ω^c are indeed unobserved when the underlying missing probability follows a decaying squared exponential, as presented in Section 1.2. This finding suggests that when the miss-

TABLE 4

Sensitivity analysis when Simulation model 2 *with* $p_{ij} \sim \text{unif}(p_{\min}, p_{\max})$, *where* $p_{\max} = p_{\min} + 0.3$ *and* $\sigma_{\text{signal}} = 12$ *is considered*: *Average* (*one standard deviation*) *of NMI values over* 200 *repetitions for various* γ

Method	$p_{\min} = 0.01$	$p_{\min} = 0.1$	$p_{\min} = 0.3$	$p_{\min} = 0.5$	$p_{\min} = 0.7$
MIS-PCA	0.40 (0.14)	0.56 (0.11)	0.69 (0.08)	0.82 (0.05)	0.92 (0.04)
PCA	0.38 (0.16)	0.50 (0.14)	0.59 (0.11)	0.70 (0.08)	0.80 (0.07)
DTSPCA	0.39 (0.18)	0.52 (0.16)	0.66 (0.13)	0.76 (0.10)	0.82 (0.06)
ITSPCA-Hard	0.36 (0.19)	0.52 (0.15)	0.65 (0.12)	0.73 (0.10)	0.80 (0.08)
ITSPCA-Soft	0.36 (0.18)	0.53 (0.16)	0.65 (0.11)	0.75 (0.09)	0.83 (0.06)
CORSPCA	0.38 (0.17)	0.51 (0.14)	0.64 (0.12)	0.74 (0.07)	0.83 (0.05)
AUGSPCA	0.38 (0.20)	0.51 (0.16)	0.65 (0.12)	0.73 (0.06)	0.83 (0.06)
EMPCA	0.38 (0.18)	0.50 (0.14)	0.66 (0.12)	0.76 (0.07)	0.83 (0.03)
WPCA	0.39 (0.22)	0.50 (0.18)	0.64 (0.13)	0.77 (0.11)	0.86 (0.09)
CIDR	0.41 (0.19)	0.53 (0.16)	0.66 (0.12)	0.79 (0.09)	0.89 (0.06)
ZIFA	0.40 (0.21)	0.52 (0.18)	0.67 (0.15)	0.78 (0.10)	0.86 (0.05)

TABLE 5
Summary of the characteristics of the 11 *real single-cell data sets*

Data set	**# cells (n)**	**# genes (p)**	**# populations**	**Missing proportion**
Buettner	182	8989	3	42%
Usoskin	622	15,332	4	40%
Pollen	249	14,805	11	55%
Kolodziejczyk	704	11,235	3	88%
Deng	317	1001	11	38%
Ginhoux	251	11,834	3	73%
Ting	114	14,405	5	57%
Treutlein	80	9352	5	53%
Tasic	1727	5832	49	33%
Zeisel	3005	4412	47	47%
Macosko	6418	12,822	39	85%

ing patterns in scRNA-seq data follows the decaying squared exponential, that is, $\exp(-\gamma X_{ij}^2)$, then one can expect that "MIS-PCA" provides more stable and accurate clustering results than other methods.

7. Applications to single-cell RNA sequence data.

In this section, we apply the proposed sparse PCA to single cell datasets. The scRNA-seq data $X \in \mathbb{R}^{n\times p}$ is high-dimensional, that is, it usually has tens of thousands genes (p) and a few hundred cells (n). One major challenge of scRNA-seq data, compared to bulk RNA-seq or gene expression microarrays, is that they have many missing values due to technical and sampling issues. Furthermore, the missingness in scRNA-seq data is not at random, that is, the probability of missing a data may be related to its value. It is also known that only a limited number of genes out of thousands of genes are significantly differentially expressed in distinct cell types. These facts motivate us to apply the proposed sparse PCA to these datasets. We consider the 11 different single cell datasets as summarized in Table 5. Each of the 11 scRNA-seq data sets represents several types of dynamic processes such as cell differentiation, cell cycle, and response upon external stimulus. See Section 8 of the Supplementary Material (Park and Zhao (2019)) for details and references of data. Each scRNA-seq data set contains cells for which the labels were known a priori or validated in the respective studies.

Figures 5–7 visualize the cells into two dimensions based on the PCA scores for the three datasets, Usoskin (Usoskin et al. (2014)), Pollen (Pollen et al. (2014)), and Ting (Ting et al. (2014)). Each cell is colored based on the true label information to visualize differences between cell populations. We observe that "MIS-PCA" can better distinguish cell types than other PCA methods. See Figures S7–S11 of the Supplementary Material (Park and Zhao (2019)) for the PCA scatterplots using the five selected datasets with sample sizes less than 1000.

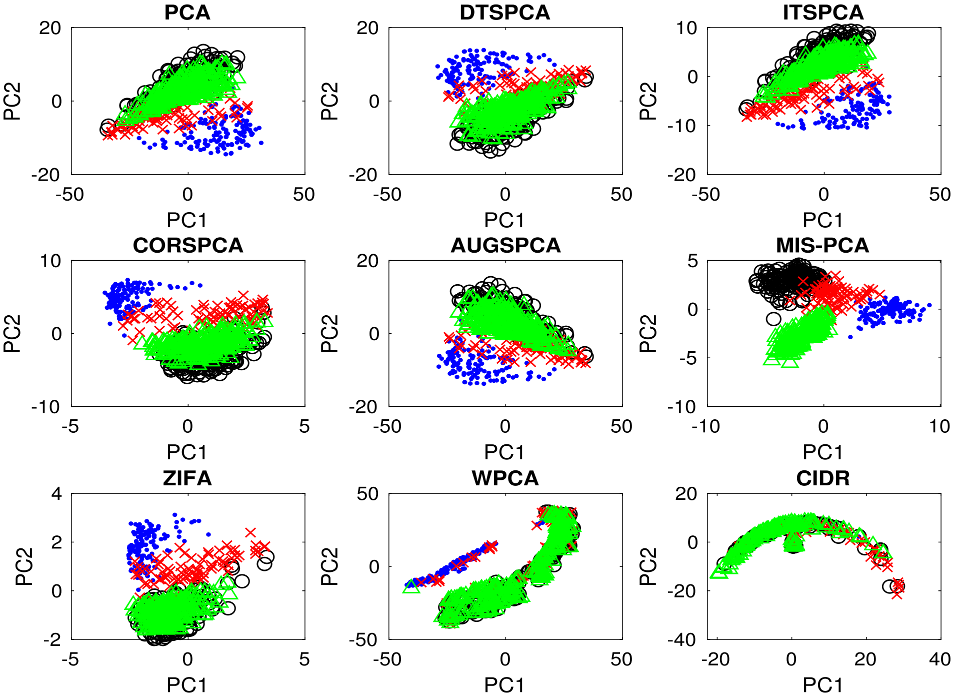

FIG. 5. *Visualization of Usoskin data (Usoskin et al. (2014)) based on obtained scores: Each point represents a cell and is colored and marked based on the true label information. The axes are in arbitrary units.*

As observed in Figures 5–7, for the other PCA methods, two PCA scores can be similar in terms of the relative position of points in lower-dimensional space even when two PCA loadings $V_1, V_2 \in \mathbb{R}^{p\times r}$ are very different in terms of the norm $\|V_1V_1^T - V_2V_2^T\|$. This often occurs when data X is sparse (i.e., has many missing values) and two PCA loadings V_1 and V_2 are sparse or only significantly different in rows corresponding to the columns of the data X with many missing values (i.e., zeros). For example, when it comes to the ith and jth samples, the distances between two scores are $d_1 := \|(X_{i,\cdot} - X_{j,\cdot})V_1\|$ and $d_2 := \|(X_{i,\cdot} - X_{j,\cdot})V_2\|$ corresponding to V_1 and V_2 loadings, respectively. Then the difference of these distances is

$$|d_1 - d_2| \le \|(X_{i,\cdot} - X_{j,\cdot})(V_1 - V_2)\| = \|(X_{i,S_i\cup S_j} - X_{j,S_i\cup S_j})(V_1 - V_2)_{S_i\cup S_j,\cdot}\|,$$

which has a small value when V_1 and V_2 are sparse in the rows in $S_i \cup S_j$ or $|S_i \cup S_j|$ is small.

While the main goal of PCA is dimension reduction, the dimension reduction methods are also commonly used for clustering data as seen in Section 6. In the single-cell data analysis, Yan et al. (2013), Deng et al. (2014), Wang et al. (2017), and Park and Zhao (2018) clustered and visualized the cells by projecting the cells

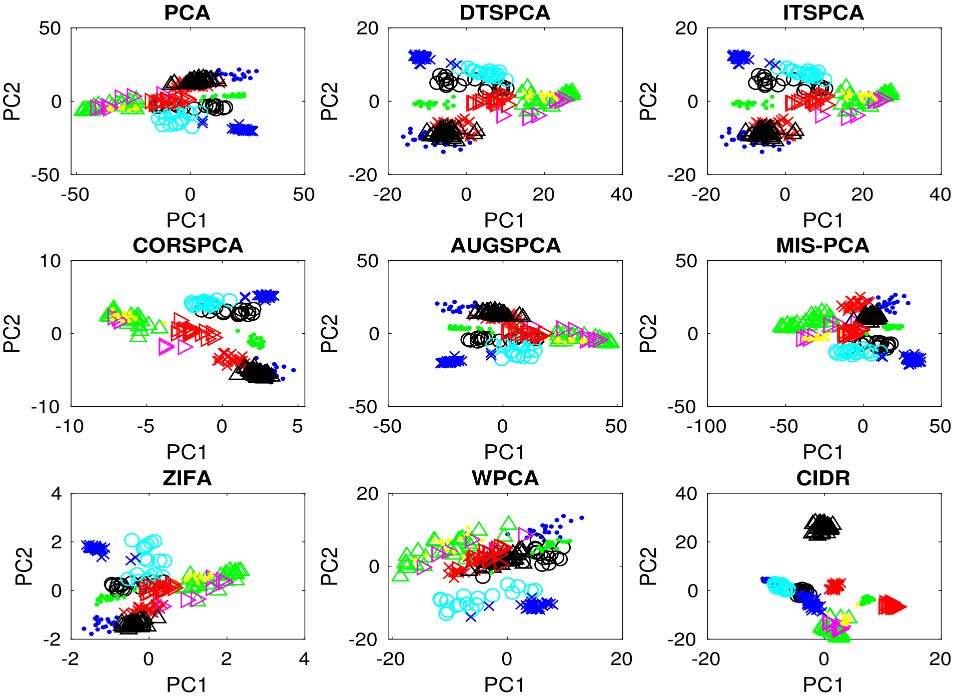

FIG. 6. *Visualization of Pollen data (Pollen et al. (2014)) based on obtained scores.*

onto the lower dimensional space. One can use the first several principal components derived from a PCA to cluster the cells. Note that for "MIS-PCA," we apply k-means clustering (Forgy (1965), MacQueen (1967)) based on the principal component scores $\widehat{X}\widehat{P} \in \mathbb{R}^{n\times r_0}$ to assign labels to each cell. Similarly, we obtain the clusters of cells for each method by using the true cluster number for the target dimension r_0 as well as the target cluster number.

Tables 6–7 and Table S5 in the Supplementary Material (Park and Zhao (2019)) summarize the NMI, ARI, and Purity values, respectively. In many cases, the proposed method "MIS-PCA" generally has higher values of these measurements, which shows that it generally performs better than other competitors. This demonstrates that the proposed method can better uncover cell-to-cell similarity and dissimilarity structures well than other dimension reduction methods. Table 8 summarizes the computational time of the methods. The "MIS-PCA" generally takes less than a minute even when the large scale data are considered. Note that "ZIFA" uses an iterative expectation-maximization algorithm for inference, which makes it computationally intensive.

In addition to comparisons based on clustering performance measure, we include some analysis for some selected data. Among the 11 data sets, we mainly analyze the three data sets based on the obtained clustering results. The first data set, called Usoskin data set (Usoskin et al. (2014)), contains 622 sensory neuron

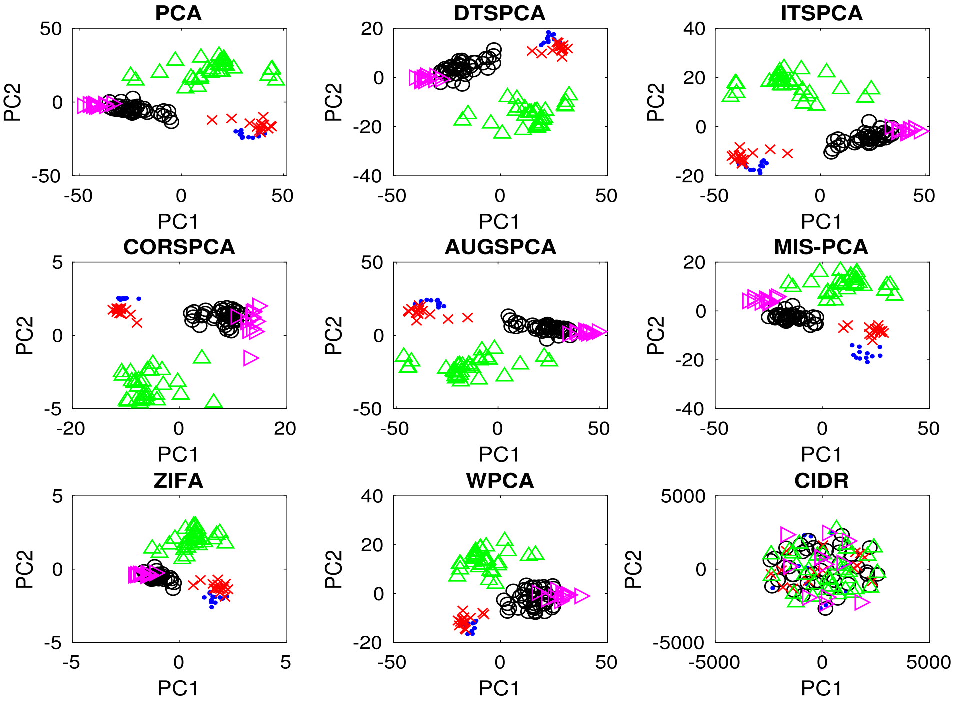

FIG. 7. *Visualization of Ting data (Ting et al. (2014)) based on obtained scores.*

cells from the mouse dorsal root ganglion, with an average of 1.14 million reads per cell. Usoskin et al. (2014) divided the cells into four neuronal types: "peptidergic nociceptors," "nonpeptidergic nociceptors," "neurofilament containing,"

TABLE 6
NMI values for the 11 *single-cell data sets. Higher values indicate better performance*

Data set	**PCA**	**MIS-PCA**	**DTS**	**ITS**	**COR**	**AUG**	**WPCA**	**EMPCA**	**ZIFA**	**CIDR**
Buettner	0.45	0.61	0.36	0.36	0.44	0.42	0.46	0.47	0.65	0.48
Usoskin	0.65	0.87	0.67	0.68	0.82	0.70	0.63	0.61	0.67	0.58
Pollen	0.94	0.96	0.94	0.91	0.94	0.94	0.88	0.85	0.93	0.91
Kolod	0.55	0.91	0.79	0.79	0.85	0.88	0.65	0.62	0.76	0.86
Deng	0.72	0.74	0.72	0.72	0.66	0.75	0.74	0.73	0.70	0.56
Ginhoux	0.51	0.61	0.58	0.52	0.53	0.52	0.54	0.43	0.39	0.35
Ting	0.89	0.94	0.91	0.91	0.93	0.89	0.92	0.91	0.91	0.78
Treutlein	0.73	0.87	0.71	0.71	0.85	0.73	0.60	0.56	0.89	0.82
Tasic	0.46	0.60	0.48	0.48	0.52	0.49	0.52	0.50	0.55	0.57
Zeisel	0.55	0.66	0.53	0.54	0.55	0.52	0.57	0.56	0.60	0.64
Macosko	0.52	0.77	0.54	0.55	0.57	0.59	0.59	0.58	0.66	0.77

TABLE 7
ARI values for the 11 *single-cell data sets. Higher values indicate better performance*

Data set	**PCA**	**MIS-PCA**	**DTS**	**ITS**	**COR**	**AUG**	**WPCA**	**EMPCA**	**ZIFA**	**CIDR**
Buettner	0.39	0.61	0.34	0.34	0.40	0.39	0.42	0.44	0.66	0.39
Usoskin	0.53	0.75	0.52	0.53	0.70	0.50	0.54	0.52	0.56	0.48
Pollen	0.93	0.95	0.93	0.87	0.93	0.93	0.86	0.78	0.92	0.84
Kolod	0.55	0.95	0.83	0.83	0.90	0.94	0.51	0.50	0.71	0.89
Deng	0.48	0.51	0.48	0.48	0.37	0.49	0.48	0.46	0.46	0.39
Ginhoux	0.48	0.63	0.45	0.45	0.40	0.50	0.56	0.40	0.38	0.32
Ting	0.87	0.92	0.89	0.89	0.94	0.87	0.91	0.91	0.91	0.84
Treutlein	0.55	0.78	0.54	0.54	0.64	0.55	0.38	0.44	0.80	0.61
Tasic	0.46	0.63	0.52	0.52	0.54	0.56	0.57	0.58	0.60	0.62
Zeisel	0.55	0.67	0.54	0.56	0.56	0.55	0.57	0.57	0.62	0.63
Macosko	0.52	0.74	0.54	0.54	0.55	0.56	0.57	0.55	0.63	0.67

and "tyrosine hydroxylase containing." Note that the sensory neuron cell types are characterized by distinct cell sizes, and the results in Usoskin et al. (2014) suggest that the four principal neuronal types are consistent with the known developmental origin of sensory neuron type. We see that "MIS-PCA" can distinguish these four neuron cell types as in Figure 5, while the other methods fail to separate the two subtypes, "nonpeptidergic nociceptors" and "tyrosine hydroxylase containing," marked black & big circle and green & triangle, respectively. This may be due to the fact that the two subtypes share the channel "TRPA1," which is activated by pungent chemical such as mustard oil, ginger and clove, and itch receptors (Usoskin et al. (2014)).

TABLE 8
Computational time (*seconds*) *for the* 11 *single-cell data sets*

Data set	**PCA**	**MIS-PCA**	**DTS**	**ITS**	**COR**	**AUG**	**WPCA**	**EMPCA**	**ZIFA**	**CIDR**
Buettner	1.5	4.2	1.6	2.5	14.5	3.8	7.7	85.4	1791.5	1.8
Usoskin	15.8	38.8	69.4	72.9	87.6	128.5	22.9	124.3	8495.1	26.5
Pollen	3.5	12.7	5.3	7.8	49.1	10.7	61.3	296.6	3012.5	4.3
Kolod	4.2	8.0	4.8	6.8	24.8	4.6	9.6	280.2	9041.5	5.2
Deng	3.2	7.9	3.1	4.3	29.3	4.1	14.1	102.5	3551.6	1.4
Ginhoux	4.4	4.3	7.3	8.8	26.5	8.0	12.3	92.8	1801.6	2.8
Ting	4.2	3.9	7.1	9.2	44.0	10.4	8.5	102.9	3051.6	1.3
Treutlein	0.3	0.7	0.3	0.3	0.2	0.2	0.6	1.0	77.2	0.2
Tasic	5.0	7.3	15.5	22.1	25.6	8.9	13.6	31.4	8125.1	4.1
Zeisel	5.3	8.4	19.2	25.0	27.1	8.6	18.1	40.5	6935.8	3.3
Macosko	4.2	6.0	13.2	20.8	23.2	5.8	11.4	25.2	6005.3	3.0

The second data set, the Ginhoux data set (Schlitzer et al. (2015)), consists of the expression values of 11,834 genes for 251 dendritic cell progenitors with one of three cellular states: Common Dendritic cell Progenitors (CDPs), Pre-Dendritic Cells (PreDCs), and Monocyte and Dendritic cell Progenitors (MDPs), where DC progenitors, originated from hematopoietic stem cells in the bone marrow, transit through a plethora of cellular states (Schlitzer et al. (2015)). Note that dendritic cells play an critical role in the activation of the adaptive immune systems in vertebrates, but some of the mechanisms involved in this process are controversial (Winter and Amit (2015), Cannoodt (2016)). Figure S10 in the Supplementary Material (Park and Zhao (2019)) visualizes the cells in 2-D space for the dimension reduction methods. We found that same type of cells generally group together well when using "MIS-PCA" than when other methods are used, but some of the two types of cells, CDPs and PreDCs (marked circle and cross, respectively), are mixed and difficult to distinguish, which is even worse when the other methods are used. This suggests that the proposed method can better differentiate different cell types even when the underlying process associated with the cells are known to be difficult to distinguish.

The analysis of the third data, Deng data set (Deng et al. (2014)), can be found in Section 10 of the Supplementary Material (Park and Zhao (2019)).

8. Discussion. Our proposed method for sparse PCA with missing observations can be considered a reformulation of the matrix completion problem from noisy entries. When the underlying missing probability follows a decaying squared exponential, which is a known as possible dropout event generating functions for scRNA-seq data, the proposed optimization essentially impute the unobserved parts of data by maximizing the conditional probability that the entries of the unobserved parts are indeed unobserved. The method allows the probability p_{ij}, the probability of the event that X_{ij}'s are observed, to be different across i and j, where unbiased sample covariance matrices are not available. Our theoretical results and assumptions do not assume the uniform sampling, and can also include the cases where the missing mechanism depends on unknown parameters. As the simulation with various missing mechanisms and application examples illustrate, when the data include missing values, the proposed sparse PCA method generally outperforms the other sparse PCA methods that rely on the unbiased sample covariance matrix.

Acknowledgments. We thank the Editor, the Associate Editor, and two anonymous reviewers for their constructive comments in reviewing our work, which greatly helped improve the manuscript.

SUPPLEMENTARY MATERIAL

Supplement to "Sparse principal component analysis with missing observations" (DOI: 10.1214/18-AOAS1220SUPP; .pdf). We provide proofs of the theoretical results presented in the main paper, characteristics of the used scRNA-seq data sets, performance metrics, and additional tables and figures for simulation and single cell data analysis.

REFERENCES

AGARWAL, A., NEGAHBAN, S. and WAINWRIGHT, M. J. (2012). Fast global convergence of gradient methods for high-dimensional statistical recovery. *Ann. Statist.* **40** 2452–2482. MR3097609

AMINI, A. A. and WAINWRIGHT, M. J. (2009). High-dimensional analysis of semidefinite relaxations for sparse principal components. *Ann. Statist.* **37** 2877–2921. MR2541450

BAIK, J. and SILVERSTEIN, J. W. (2006). Eigenvalues of large sample covariance matrices of spiked population models. *J. Multivariate Anal.* **97** 1382–1408. MR2279680

BAILEY, S. (2012). Principal component analysis with noisy and/or missing data. *Publications of the Astronomical Society of the Pacific* **124** 1015–1023.

BERTHET, Q. and RIGOLLET, P. (2013). Optimal detection of sparse principal components in high dimension. *Ann. Statist.* **41** 1780–1815. MR3127849

CAI, T. T., MA, Z. and WU, Y. (2013). Sparse PCA: Optimal rates and adaptive estimation. *Ann. Statist.* **41** 3074–3110. MR3161458

CAI, T. T. and ZHANG, A. (2016). Minimax rate-optimal estimation of high-dimensional covariance matrices with incomplete data. *J. Multivariate Anal.* **150** 55–74. MR3534902

CANNOODT, R. (2016). Scorpius improves trajectory inference and identifies novel modules in dendritic cell development. Preprint.

CHEN, H. (2002). Principal component analysis with missing data and outliers. Robust Image Understanding Laboratory. Tech. report.

DELCHAMBRE, L. (2014). Weighted principal component analysis: A weighted covariance eigendecomposition approach. *Mon. Not. R. Astron. Soc.* **446** 3545–3555.

DENG, Q., RAMSKÖLD, D., REINIUS, B. and SANDBERG, R. (2014). Single-cell rna-seq reveals dynamic, random monoallelic gene expression in mammalian cells. *Science* **343** 193–196.

DESHPANDE, Y. and MONTANARI, A. (2016). Sparse PCA via covariance thresholding. *J. Mach. Learn. Res.* **17** 1–41. MR3555032

DODGE, Y. (1985). *Analysis of Experiments with Missing Data. Wiley Series in Probability and Mathematical Statistics*: *Applied Probability and Statistics*. Wiley, Chichester. MR0811638

ECKART, C. and YOUNG, G. (1936). The approximation of one matrix by another of low rank. *Psychometrika* **1** 211.

FORGY, E. (1965). Cluster analysis of multivariate data: Efficiency versus interpretability of classifications. *Biometrics* **21** 768–769.

HASTIE, T., MAZUMDER, R., LEE, J. D. and ZADEH, R. (2015). Matrix completion and low-rank SVD via fast alternating least squares. *J. Mach. Learn. Res.* **16** 3367–3402. MR3450542

HOTELLING, H. (1933). Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.* **24** 417–441.

JOHNSTONE, I. M. and LU, A. Y. (2009). On consistency and sparsity for principal components analysis in high dimensions. *J. Amer. Statist. Assoc.* **104** 682–693. MR2751448

JOLLIFFE, I. T. (1986). *Principal Component Analysis*. *Springer Series in Statistics*. Springer, New York. MR0841268

JOURNÉE, M., NESTEROV, Y., RICHTÁRIK, P. and SEPULCHRE, R. (2010). Generalized power method for sparse principal component analysis. *J. Mach. Learn. Res.* **11** 517–553. MR2600619

KUNDU, A., DRINEAS, P. and MAGDON-ISMAIL, M. (2015). Approximating sparse pca from incomplete data. *Adv. Neural Inf. Process. Syst.* **28** 388–396.

LANGE, K., HUNTER, D. R. and YANG, I. (2000). Optimization transfer using surrogate objective functions. *J. Comput. Graph. Statist.* **9** 1–59. MR1819865

LIN, P., TROUP, M. and HO, J. W. K. (2017). Cidr: Ultrafast and accurate clustering through imputation for single-cell rna-seq data. *Genome Biol*. **18**.

LOH, P.-L. and WAINWRIGHT, M. J. (2012). High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity. *Ann. Statist*. **40** 1637–1664. MR3015038

LOUNICI, K. (2013). Sparse principal component analysis with missing observations. In *High Dimensional Probability VI. Progress in Probability* **66** 327–356. Birkhäuser/Springer, Basel. MR3443508

MA, Z. (2013). Sparse principal component analysis and iterative thresholding. *Ann. Statist*. **41** 772–801. MR3099121

MACQUEEN, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proc. Fifth Berkeley Sympos. Math. Statist. and Probability* (*Berkeley, Calif*., 1965/66) 281–297. Univ. California Press, Berkeley, CA. MR0214227

MAZUMDER, R., HASTIE, T. and TIBSHIRANI, R. (2010). Spectral regularization algorithms for learning large incomplete matrices. *J. Mach. Learn. Res*. **11** 2287–2322. MR2719857

NADLER, B. (2009). Discussion of "On consistency and sparsity for principal components analysis in high dimensions," by I. M. Johnstone and A. Y. Lu" [MR2751448]. *J. Amer. Statist. Assoc*. **104** 694–697. MR2751449

PARK, S. and ZHAO, H. (2018). Spectral clustering based on learning similarity matrix. *Bioinformatics* **34** 2069–2076.

PARK, S. and ZHAO, H. (2019). Supplement to "Sparse Principal Component Analysis with Missing Observations." DOI:10.1214/18-AOAS1220SUPP.

PAUL, D. (2007). Asymptotics of sample eigenstructure for a large dimensional spiked covariance model. *Statist. Sinica* **17** 1617–1642. MR2399865

PAUL, D. and JOHNSTONE, I. M. (2007). Augmented sparse principal component analysis for high dimensional data. Available at arXiv:1202.1242v1.

PIERSON, E. and YAU, C. (2015). Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biol*. **16** 1–10.

POLLEN, A., NOWAKOWSKI, T., SHUGA, J., WANG, X., LEYRAT, A., LUI, J., LI, N., SZPANKOWSKI, L., FOWLER, B., CHEN, P., RAMALINGAM, N., SUN, G., THU, M., NORRIS, M., LEBOFSKY, R., TOPPANI, D., KEMP, D., WONG, M., CLERKSON, B., JONES, B., WU, S., KNUTSSON, L., ALVARADO, B., WANG, J., WEAVER, L., MAY, A., JONES, R., UNGER, M., KRIEGSTEIN, A. and WEST, J. (2014). Low-coverage single-cell mrna sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex. *Nat. Biotechnol*. **32** 1053–1058.

QI, X., LUO, R. and ZHAO, H. (2013). Sparse principal component analysis by choice of norm. *J. Multivariate Anal*. **114** 127–160. MR2993878

SCHLITZER, A., SIVAKAMASUNDARI, V., CHEN, J., SUMATOH, H. R. B., SCHREUDER, J., LUM, J., MALLERET, B., ZHANG, S., LARBI, A., ZOLEZZI, F. et al. (2015). Identification of cdc1- and cdc2-committed dc progenitors reveals early lineage priming at the common dc progenitor stage in the bone marrow. *Nature Immunology* **16** 718–728.

SHAO, C. and HÖFER, T. (2017). Robust classification of single-cell transcriptome data by nonnegative matrix factorization. *Bioinformatics* **33** 235–242.

SREBRO, N., RENNIE, J. and JAAKKOLA, T. (2005). Maximum margin matrix factorization. *Adv. Neural Inf. Process. Syst*. **17**.

STREHL, A. and GHOSH, J. (2003). Cluster ensembles—A knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res*. **3** 583–617. Computational learning theory. MR1991087

TING, D. T., WITTNER, B. S., LIGORIO, M., JORDAN, N. V., SHAH, A. M., MIYAMOTO, D. T., ACETO, N., BERSANI, F., BRANNIGAN, B. W. et al. (2014). Single-cell rna sequencing identifies extracellular matrix gene expression by pancreatic circulating tumor cells. *Cell Reports* **8** 1905–1918.

TSALMANTZA, P. and HOGG, D. W. (2012). A data-driven model for spectra: Finding double redshifts in the sloan digital sky survey. *Astrophys. J.* **753** 1–16.

USOSKIN, D., FURLAN, A., ISLAM, S., ABDO, H., LÖNNERBERG, P., LOU, D., HJERLING-LEFFLER, J., HAEGGSTRÖM, J., KHARCHENKO, O., KHARCHENKO, P. V., LINNARSSON, S. and ERNFORS, P. (2014). Unbiased classification of sensory neuron types by large-scale single-cell rna sequencing. *Nat. Neurosci.* **18** 145–153.

VU, V. Q. and LEI, J. (2013). Minimax sparse principal subspace estimation in high dimensions. *Ann. Statist.* **41** 2905–2947. MR3161452

WAGNER, S. and WAGNER, D. (2007). Comparing clusterings: An overview. Universität Karlsruhe, Fakultät Für Informatik Karlsruhe.

WANG, Z., LU, H. and LIU, H. (2014). Tighten after relax: Minimax-optimal sparse pca in polynomial time. *Adv. Neural Inf. Process. Syst.* 3383–3391.

WANG, B., ZHU, J., PIERSON, E., RAMAZZOTTI, D. and BATZOGLOU, S. (2017). Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nat. Methods* **14** 414–416.

WINTER, D. R. and AMIT, I. (2015). Dcs are ready to commit. *Nature Immunology* **16** 683–685.

XU, C. and SU, Z. (2015). Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics* **31** 1974–1980.

YAN, L., YANG, M., GUO, H., YANG, L., WU, J., LI, R., LIU, P., LIAN, Y., ZHENG, X. et al. (2013). Single-cell rna-seq profiling of human preimplantation embryos and embryonic stem cells. *Nature Structure and Molecular Biology* **20** 1131–1142.

YUAN, X.-T. and ZHANG, T. (2013). Truncated power method for sparse eigenvalue problems. *J. Mach. Learn. Res.* **14** 899–925. MR3063614

ZOU, H., HASTIE, T. and TIBSHIRANI, R. (2006). Sparse principal component analysis. *J. Comput. Graph. Statist.* **15** 265–286. MR2252527

DEPARTMENT OF STATISTICS
SUNGKYUNKWAN UNIVERSITY
SUNGKYUNKWAN-RO, JONGNO-GU
SEOUL
SOUTH KOREA
E-MAIL: ishspsy@skku.edu

SCHOOL OF PUBLIC HEALTH
YALE UNIVERSITY
333 CEDAR STREET
NEW HAVEN, CONNECTICUT 06510
USA
E-MAIL: hongyu.zhao@yale.edu