ROYAL STATISTICAL SOCIETY
DATA | EVIDENCE | DECISIONS

Journal of the Royal Statistical Society
Statistical Methodology
Series B

J. R. Statist. Soc. B (2020)
82, *Part* 2, *pp.* 329–359

Sparse principal component analysis via axis-aligned random projections

Milana Gataric, Tengyao Wang and Richard J. Samworth

University of Cambridge, UK

[Received July 2018. Revised December 2019]

Summary. We introduce a new method for sparse principal component analysis, based on the aggregation of eigenvector information from carefully selected axis-aligned random projections of the sample covariance matrix. Unlike most alternative approaches, our algorithm is non-iterative, so it is not vulnerable to a bad choice of initialization. We provide theoretical guarantees under which our principal subspace estimator can attain the minimax optimal rate of convergence in polynomial time. In addition, our theory provides a more refined understanding of the statistical and computational trade-off in the problem of sparse principal component estimation, revealing a subtle interplay between the effective sample size and the number of random projections that are required to achieve the minimax optimal rate. Numerical studies provide further insight into the procedure and confirm its highly competitive finite sample performance.

Keywords: Dimensionality reduction; Eigenspace estimation; Ensemble learning; Sketching; Statistical and computational trade-offs

1. Introduction

Principal component analysis (PCA) is one of the most widely used techniques for dimensionality reduction in statistics, image processing and many other fields. The aim is to project the data along directions that explain the greatest proportion of the variance in the population. In the simplest setting where we seek a single, univariate projection of our data, we may estimate this optimal direction by computing the leading eigenvector of the sample covariance matrix.

Despite its successes and enormous popularity, it has been well known for a decade or more that PCA breaks down as soon as the dimensionality p of the data is of the same order as the sample size n. More precisely, suppose that $X_1, \ldots, X_n \sim^{\mathrm{IID}} N_p(0, \Sigma)$, with $p \geqslant 2$, are observations from a Gaussian distribution with a spiked covariance matrix $\Sigma = I_p + v_1 v_1^{\mathrm{T}}$ whose leading eigenvector is $v_1 \in \mathcal{S}^{p-1} := \{v \in \mathbb{R}^p : \|v\|_2 = 1\}$, and let $\hat{v}_1$ denote the leading unit length eigenvector of the sample covariance matrix $\hat{\Sigma} := n^{-1} \Sigma_{i=1}^n X_i X_i^{\mathrm{T}}$. Then Johnstone and Lu (2009) and Paul (2007) showed that $\hat{v}_1$ is a consistent estimator of v_1, i.e. $|\hat{v}_1^{\mathrm{T}} v_1| \to^{\mathrm{p}} 1$, if and only if $p = p_n$ satisfies $p/n \to 0$ as $n \to \infty$. It is also worth noting that the principal component v_1 may be a linear combination of all elements of the canonical basis in $\mathbb{R}^p$, which can often make it difficult to interpret the estimated projected directions (Jolliffe *et al.*, 2003).

To remedy this situation, and to provide additional interpretability to the principal components in high dimensional settings, Jolliffe *et al.* (2003) and Zou *et al.* (2006) proposed sparse principal component analysis (SPCA). Here it is assumed that the leading population eigenvectors belong to the k-sparse unit ball

Address for correspondence: Richard J. Samworth, Department of Pure Mathematics and Mathematical Statistics, Centre for Mathematical Sciences, University of Cambridge, Wilberforce Road, Cambridge, CB3 0WB, UK.
E-mail: r.samworth@statslab.cam.ac.uk

$$\mathcal{B}_0^{p-1}(k) := \left\{ v = (v^{(1)}, \ldots, v^{(p)})^{\mathrm{T}} \in \mathcal{S}^{p-1} : \sum_{j=1}^{p} \mathbb{1}_{\{v^{(j)} \neq 0\}} \leqslant k \right\}$$

for some $k \in \{1, \ldots, p\}$. In addition to the easier interpretability, a large amount of research effort has shown that such an assumption facilitates improved estimation performance (e.g. Johnstone and Lu (2009), Paul and Johnstone (2012), Vu and Lei (2013), Cai *et al.* (2013), Ma (2013) and Wang *et al.* (2016)). To give a flavour of these results, let $\mathcal{V}_n$ denote the set of all estimators of v_1, i.e. the class of Borel measurable functions from $\mathbb{R}^{n \times p}$ to $\mathcal{S}^{p-1}$. Vu and Lei (2013) introduced a class $\mathcal{Q}$ of sub-Gaussian distributions whose first principal component v_1 belongs to $\mathcal{B}_0^{p-1}(k)$ and showed that

$$\inf_{\tilde{v}_1 \in \mathcal{V}_n} \sup_{Q \in \mathcal{Q}} \mathbb{E}_Q \{1 - (\tilde{v}_1^{\mathrm{T}} v_1)^2\} \asymp \frac{k \log(p)}{n}, \tag{1}$$

where $a_n \asymp b_n$ means that $0 < \liminf_{n\to\infty} |a_n/b_n| \leqslant \limsup_{n\to\infty} |a_n/b_n| < \infty$. Thus, consistent estimation is possible in this framework provided only that $k = k_n$ and $p = p_n$ satisfy $k \log(p)/n \to 0$. Vu and Lei (2013) showed further that this estimation rate is achieved by the natural estimator

$$\hat{v}_1 \in \operatorname*{arg\,max}_{v \in \mathcal{B}_0^{p-1}(k)} v^{\mathrm{T}} \hat{\Sigma} v. \tag{2}$$

However, results such as expression (1) do not complete the story of SPCA. Indeed, computing the estimator defined in expression (2) turns out to be an 'NP hard' problem (e.g. Tillmann and Pfetsch (2014)): the naive approach would require searching through all $\binom{p}{k}$ of the $k \times k$ symmetric submatrices of $\hat{\Sigma}$, which takes exponential time in k. Therefore, in parallel with the theoretical developments that were described above, numerous alternative algorithms for SPCA have been proposed in recent years. For instance, several references have introduced techniques based on solving the non-convex optimization problem (2) by invoking an l_1-penalty (e.g. Jolliffe *et al.* (2003), Zou *et al.* (2006), Shen and Huang (2008) and Witten *et al.* (2009)). Typically, these methods are fast but lack theoretical performance guarantees. In contrast d'Aspremont *et al.* (2007) proposed to compute problem (2) via semidefinite relaxation. This approach and its variants were analysed by Amini and Wainwright (2009), Vu *et al.* (2013) and Wang *et al.* (2014, 2016) and have been proved to achieve the minimax rate of convergence under certain assumptions on the underlying distribution and asymptotic regime, but the algorithm is slow compared with other approaches. In a separate, recent development, it is now understood that, conditionally on a planted clique hypothesis from theoretical computer science, there is an asymptotic regime in which no randomized polynomial time algorithm can attain the minimax optimal rate (Wang *et al.*, 2016). Various fast iterative algorithms were introduced by Johnstone and Lu (2009), Paul and Johnstone (2012) and Ma (2013); the last of these was shown to attain the minimax rate under a Gaussian spiked covariance model. We also mention the computationally efficient combinatorial approaches that were proposed by Moghaddam *et al.* (2006) and d'Aspremont *et al.* (2008) that aim to find solutions to the optimization problem (2) by using greedy methods.

A common feature to all of the computationally efficient algorithms mentioned above is that they are iterative, in the sense that, starting from an initial guess $\hat{v}^{[0]} \in \mathbb{R}^p$, they refine their guess by producing a finite sequence of iterates $\hat{v}^{[1]}, \ldots, \hat{v}^{[T]} \in \mathbb{R}^p$, with the estimator defined to be the final iterate. A major drawback of such iterative methods is that a bad initialization may yield a disastrous final estimate. To illustrate this point, we ran a simple simulation in which the underlying distribution is $N_{400}(0, \Sigma)$, with

$$\Sigma = \begin{pmatrix} 10J_{10} & \\ & 8.9J_{390} + I_{390} \end{pmatrix} + 0.01 I_{400}, \tag{3}$$

where $J_q := \mathbf{1}_q \mathbf{1}_q^{\mathrm{T}}/q \in \mathbb{R}^{q \times q}$ denotes the matrix with each entry equal to $1/q$. In this example, $v_1 = (\mathbf{1}_{10}^{\mathrm{T}}, \mathbf{0}_{390}^{\mathrm{T}})^{\mathrm{T}}/\sqrt{10}$, so $k = 10$. Fig. 1 shows, for several SPCA algorithms, different sample sizes and different initialization methods, the average values of the loss function

$$L(u, v) := \sin \measuredangle(u, v) = \{1 - (u^{\mathrm{T}} v)^2\}^{1/2}, \tag{4}$$

over 100 repetitions of the experiment. In Figs 1(a) and 1(b), the initialization methods that were used were the default recommendations of the respective authors, namely diagonal thresholding (d'Aspremont *et al.*, 2008; Ma, 2013) and classical PCA (Zou *et al.*, 2006; Shen and Huang, 2008; Witten *et al.*, 2009). We note that the consistency of diagonal thresholding relies on a spiked covariance structure, which is violated in this example. In Figs 1(c) and 1(d), we ran the same algorithms with 10 independent initializing vectors chosen uniformly at random on $\mathcal{S}^{p-1}$, and we selected the solution $\hat{v}$ from these 10 that maximizes $v \mapsto v^{\mathrm{T}} \hat{\Sigma} v$. The main observation is that each of the previously proposed algorithms that were mentioned above produces very poor estimates, with some almost orthogonal to the true principal component! The reason for this is that all the default initialization procedures are unsuccessful in finding a good starting point. For some methods, this problem may be fixed by increasing the number of random initializations, but it may take an enormous number of such random restarts (and consequently a very long time) to achieve this. We demonstrate this in Figs 1(e) and 1(f), where, for $n = 350$ (Fig. 1(e)) and $n = 2000$ (Fig. 1(f)), we plot the logarithm of the average loss as time increases through the number of random restarts. As an alternative method, in Figs 1(a)–1(d), we also present the corresponding results for the variants of Wang *et al.* (2016) of the semidefinite programming algorithm that was introduced by d'Aspremont *et al.* (2007). This method is guaranteed to converge from any initialization and so does not suffer the same poor performance as mentioned above. However, the semidefinite programming algorithm took even longer to reach algorithmic convergence than any of the alternative approaches, so, in the setting of Figs 1(e) and 1(f), it finally reached a logarithmic average loss of around -4 (Fig. 1(e)) and -5.9 (Fig. 1(f)) after an average time of $\exp(8) \approx 3000$ s (Fig. 1(e)) and $\exp(9.25) \approx 10000$ s (Fig. 1(f)); this slow running time means that it does not appear in Figs 1(e) and 1(f). We refer to Section 4.2 for further comparisons using different examples.

In Section 2 of this paper, we propose a novel algorithm for SPCA that aggregates estimates over carefully chosen axis-aligned random projections of the data into a lower dimensional space. In contrast with the other algorithms that were mentioned above, it is non-iterative and does not depend on a choice of initialization, so it has no difficulty with the simulation example above. Indeed, from the blue curve in Fig. 1, we see that it outperforms even the semidefinite programming algorithm, compared with which it was over 7000 times faster in the $n = 2000$ case.

Our algorithm, which we refer to as SPCAvRP, turns out to be attractive for both theoretical and computational reasons. From a theoretical point of view, our algorithm provides a new perspective on the statistical and computational trade-off that is involved in the SPCA problem. As we show in Section 3, when the effective sample size is large, the SPCAvRP procedure can attain the minimax optimal rate with a number of projections that grows only polynomially in the problem parameters. In contrast, if one were to use a number of random projections exponentially large in k, SPCAvRP could even achieve this minimax rate in a much smaller effective sample size regime. Although this exponentially large number of projections may seem discouraging, we emphasize that it is in fact not a drawback of the SPCAvRP algorithm but

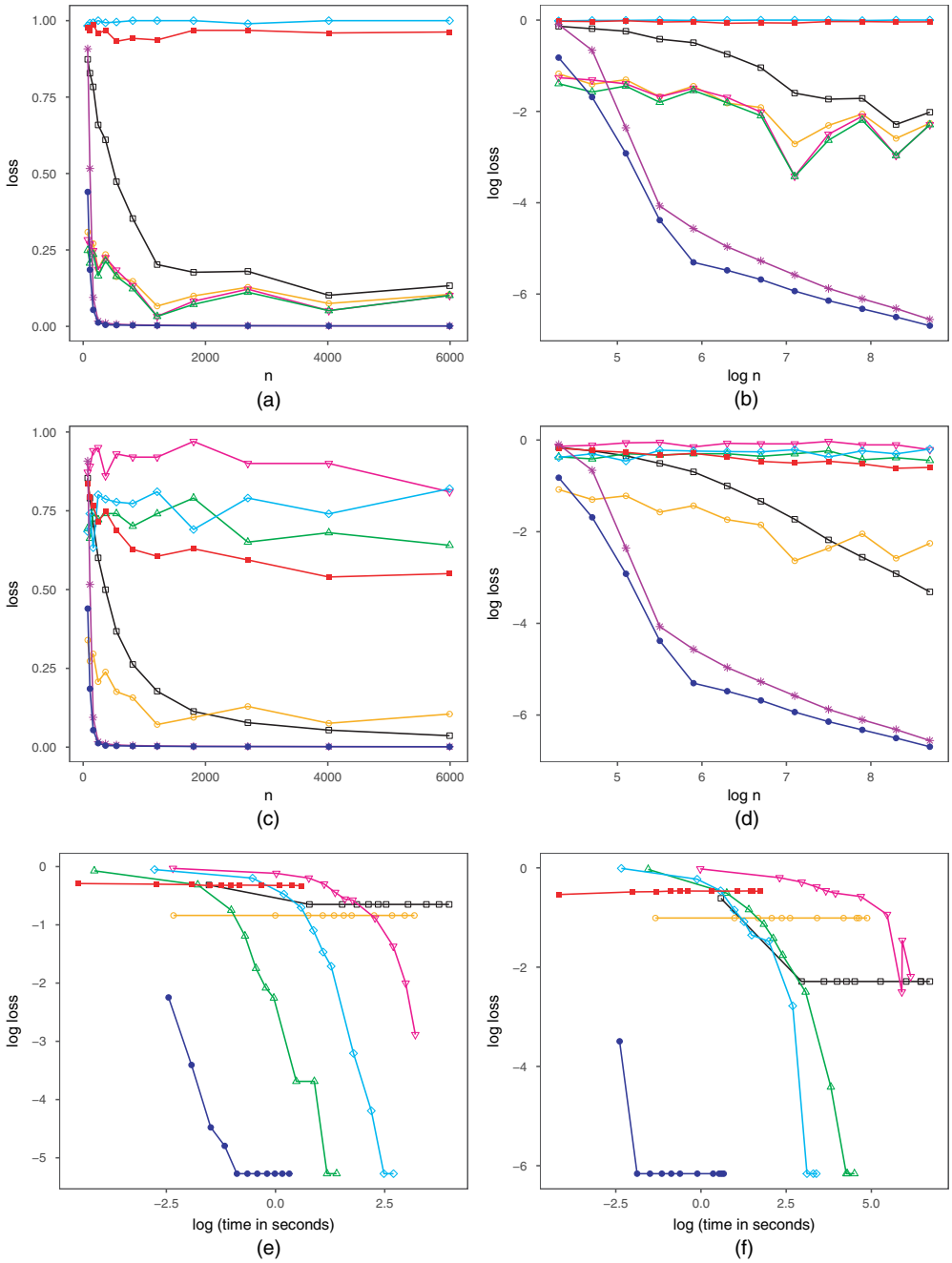


Fig. 1. Comparison of various approaches by using covariance model (3) (□, Zou *et al.* (2006); ▽, Shen and Huang (2008), l_1-thresholding; △, Shen and Huang (2008), l_0-thresholding; ◇, d'Aspremont *et al.* (2008); ○, Witten *et al.* (2009); ■, Ma (2013); ✱, semidefinite programming; ●, SPCAvRP): in (a), (b), (c), (d) average loss (4) for different sample sizes n; in (a), (c) the normal scale; in (b), (d) the log–log-scale; in (a), (b) default initialization; in (c), (d) best of 10 random initializations; in (e), (f) average loss (4) against time in seconds on the log–log-scale when $n = 350$ in (e) and $n = 2000$ in (f) (we vary the number of random projections ($A \in (50, 200)$ and $B = \lceil A/2 \rceil$) for SPCAvRP and the number of random initializations (from 1 to 250) for the other iterative competing methods)

simply a reflection of the fundamental difficulty of the problem in this effective sample size regime. Indeed, Wang *et al.* (2016) established a computational lower bound, which reveals that no randomized polynomial time algorithm can attain the minimax rate of convergence for these effective sample sizes. The elucidation of the transition from polynomial to exponentially large number of projections is an illustration of the fascinating fundamental statistical and computational trade-off in this problem. The computational attractions of the algorithm proposed include the fact that it is highly scalable because of easy parallelization and does not even require computation of $\hat{\Sigma} \in \mathbb{R}^{p \times p}$, since it suffices to extract principal submatrices of $\hat{\Sigma}$, which can be done by computing the sample covariance matrices of the projected data. This may result in a significant computational saving if p is very large. Several numerical aspects of the algorithm, including a finite sample simulation comparison with alternative methods on both simulated and real data, are considered in Section 4. These reveal that our SPCAvRP algorithm has very competitive performance and, furthermore, it enjoys robustness properties that iterative algorithms do not share. The proofs of all of our results are given in Appendix A.

Algorithms based on random projections have recently been shown to be highly effective for several different problems in high dimensional statistical inference. For instance, in the context of high dimensional classification, Cannings and Samworth (2017) showed that their random projection ensemble classifier that aggregates over projections that yield small estimates of the test error can result in excellent performance. Marzetta *et al.* (2011) employed an ensemble of random projections to construct an estimator of the population covariance matrix and its inverse in the setting where $n < p$. Fowler (2009) introduced a so-called compressive projection PCA that reconstructs the sample principal components from many low dimensional projections of the data. Finally, to decrease the computational burden of classical PCA, Qi and Hughes (2012) and Pourkamali-Anaraki and Hughes (2014) proposed estimating $v_1(\Sigma)$ by the leading eigenvector of $n^{-1}\Sigma_{i=1}^{n} P_i X_i X_i^{\mathrm{T}} P_i$, where $P_1, \ldots, P_n$ are random projections of a particular form.

1.1. Notation

We conclude this introduction with some notation that is used throughout the paper. For $r \in \mathbb{N}$, let $[r] := \{1, \ldots, r\}$. For a vector $u \in \mathbb{R}^p$, we write $u^{(j)}$ for its jth component and $\|u\|_2 := \{\Sigma_{j=1}^{p}(u^{(j)})^2\}^{1/2}$ for its Euclidean norm. For a real symmetric matrix $U \in \mathbb{R}^{p \times p}$, let $\lambda_1(U) \geqslant \lambda_2(U) \geqslant \ldots \geqslant \lambda_p(U)$ denote its eigenvalues, arranged in decreasing order, and let $v_1(U), \ldots, v_p(U)$ denote the corresponding eigenvectors. In addition, for $m \in [p]$, we write $V_m(U) := (v_1(U), \ldots, v_m(U))$ for the $p \times m$ matrix whose columns are the m leading eigenvectors of U. In the special case where $U = \Sigma$, we drop the argument and write $\lambda_r = \lambda_r(\Sigma)$, $v_r = v_r(\Sigma)$ and $V_m = V_m(\Sigma)$. For a general $U \in \mathbb{R}^{p \times m}$, we define $U^{(j,j')}$ to be the (j, j')th entry of U, and $U^{(j,\cdot)}$ the jth row of U, regarded as a column vector. Given $S \subseteq [p]$ and $S' \subseteq [m]$, we write $U^{(S,S')}$ for the $|S| \times |S'|$ matrix that is obtained by extracting the rows of U indexed by S and columns indexed by S'; we also write $U^{(S,\cdot)} := U^{(S,[m])}$. We write $\|U\|_{\mathrm{op}} := \sup_{x \in \mathcal{S}^{m-1}} \|Ux\|_2$ and $\|U\|_{\mathrm{F}} := (\Sigma_{j=1}^{p}\Sigma_{j'=1}^{m}|U^{(j,j')}|^2)^{1/2}$ for the operator and Frobenius norms of U respectively. We denote the set of real orthogonal $p \times p$ matrices by $\mathbb{O}_p$ and the set of real $p \times m$ matrices with orthonormal columns by $\mathbb{O}_{p,m}$. For matrices $U, V \in \mathbb{O}_{p,m}$, we define the loss function

$$L(U, V) := \|\sin\{\Theta(U, V)\}\|_{\mathrm{F}},$$

where the sine function acts elementwise, and where $\Theta(U, V)$ is the $m \times m$ diagonal matrix whose jth diagonal entry is the jth principal angle between U and V, i.e. $\cos^{-1}(\sigma_j)$, where σ_j is the jth singular value of $U^{\mathrm{T}}V$. Observe that this loss function reduces to expression (4) when $m = 1$.

For any index set $J \subseteq [p]$ we write P_J to denote the projection onto the span of $\{e_j : j \in J\}$,

where $e_1, \ldots, e_p$ are the standard Euclidean basis vectors in $\mathbb{R}^p$, so that P_J is a $p \times p$ diagonal matrix whose jth diagonal entry is $\mathbb{1}_{\{j \in J\}}$. Finally, for $a, b \in \mathbb{R}$, we write $a \lesssim b$ to mean that there is a universal constant $C > 0$ such that $a \leqslant Cb$.

2. Sparse principal component analysis via random projections

2.1. Single principal component estimation

In this section, we describe our algorithm for estimating a single principal component v_1 in detail; more general estimation of multiple principal components $v_1, \ldots, v_m$ is treated in Section 2.2. Let $x_1, \ldots, x_n$ be data points in $\mathbb{R}^p$ and let $\hat{\Sigma} := n^{-1} \Sigma_{i=1}^n x_i x_i^{\mathrm{T}}$. We think of $x_1, \ldots, x_n$ as independent realizations of a zero-mean random vector X, so a practitioner may choose to centre each variable so that $\Sigma_{i=1}^n x_i^{(j)} = 0$ for each $j \in [p]$. For $d \in [p]$, let $\mathcal{P}_d := \{P_S : S \subseteq [p], \, |S| = d\}$ denote the set of d-dimensional, axis-aligned projections. For fixed $A, B \in \mathbb{N}$, consider projections $\{P_{a,b} : a \in [A], b \in [B]\}$ independently and uniformly distributed on $\mathcal{P}_d$. We think of these projections as consisting of A groups, each of cardinality B. For each $a \in [A]$, let

$$b^*(a) := \operatorname*{sarg\,max}_{b \in [B]} \lambda_1(P_{a,b} \hat{\Sigma} P_{a,b})$$

denote the index of the selected projection within the ath group, where sarg max denotes the smallest element of the arg max in the lexicographic ordering. The idea is that the non-zero entries of $P_{a,b^*(a)} \hat{\Sigma} P_{a,b^*(a)}$ form a principal submatrix of $\hat{\Sigma}$ that should have a large leading eigenvalue, so the non-zero entries of the corresponding leading eigenvector $\hat{v}_{a,b^*(a);1}$ of $P_{a,b^*(a)} \hat{\Sigma} P_{a,b^*(a)}$ should have some overlap with those of v_1. Observe that, if $d = k$ and $\{P_{a,b} : b \in [B]\}$ were to contain all $\binom{p}{k}$ projections, then the leading eigenvector of $P_{a,b^*(a)} \hat{\Sigma} P_{a,b^*(a)}$ would yield the minimax optimal estimator in problem (2). Of course, it would typically be too computationally expensive to compute all such projections, so instead we consider only B randomly chosen projections.

The remaining challenge is to aggregate over the selected projections. For this, for each co-ordinate $j \in [p]$, we compute an importance score $\hat{w}^{(j)}$, defined as an average over $a \in [A]$ of the squared jth components of the selected eigenvectors $\hat{v}_{a,b^*(a);1}$,

$$\hat{w}^{(j)} := \frac{1}{A} \sum_{a=1}^{A} (\hat{\lambda}_{a,b^*(a);1} - \hat{\lambda}_{a,b^*(a);2}) (\hat{v}^{(j)}_{a,b^*(a);1})^2, \tag{5}$$

weighted by the eigengap $\lambda_1(P_{a,b^*(a)} \hat{\Sigma} P_{a,b^*(a)}) - \lambda_2(P_{a,b^*(a)} \hat{\Sigma} P_{a,b^*(a)})$. This means that we take account, not just of the frequency with which each co-ordinate is chosen, but also their corresponding magnitudes in the selected eigenvector, as well as an estimate of the signal strength. Finally, we select the l indices $\hat{S}$ corresponding to the largest values of $\hat{w}^{(1)}, \ldots, \hat{w}^{(p)}$ and output our estimate $\hat{v}_1$ as the leading eigenvector of $P_{\hat{S}} \hat{\Sigma} P_{\hat{S}}$. Pseudocode for our SPCAvRP algorithm is given in algorithm 1 in Table 1.

Besides the intuitive selection of the most important co-ordinates, the use of axis-aligned projections facilitates faster computation as opposed to the use of general orthogonal projections. Indeed, the multiplication of $\hat{\Sigma} \in \mathbb{R}^{p \times p}$ by an axis-aligned projection $P \in \mathcal{P}_d$ from the left (or right) can be recast as the selection of d rows (or columns) of $\hat{\Sigma}$ corresponding to the indices of the non-zero diagonal entries of P. Thus, instead of the typical $\mathcal{O}(p^2 d)$ matrix multiplication complexity, only $\mathcal{O}(pd)$ operations are required. We also remark that, instead of storing P, it suffices to store its non-zero indices.

More generally, the computational complexity of algorithm 1 can be analysed as follows. Generating AB initial random projections takes $\mathcal{O}(ABd)$ operations. Next, we need to compute

Table 1. Algorithm 1: pseudocode for the SPCAvRP algorithm for a single principal component

Input: $x_1, \ldots, x_n \in \mathbb{R}^p$, $A, B \in \mathbb{N}$, $d, l \in [p]$
Generate $\{P_{a,b} : a \in [A], b \in [B]\}$ independently and uniformly from $\mathcal{P}_d$
Compute $\{P_{a,b}\hat{\Sigma}P_{a,b} : a \in [A], b \in [B]\}$, where $\hat{\Sigma} := n^{-1}\Sigma_{i=1}^n x_i x_i^{\mathrm{T}}$
for $a = 1, \ldots, A$ *do*
  for $b = 1, \ldots, B$ *do*
    Compute $\hat{\lambda}_{a,b;1} := \lambda_1(P_{a,b}\hat{\Sigma}P_{a,b})$, $\hat{\lambda}_{a,b;2} := \lambda_2(P_{a,b}\hat{\Sigma}P_{a,b})$ and $\hat{v}_{a,b;1} \in v_1(P_{a,b}\hat{\Sigma}P_{a,b})$
  end
  Compute

$$b^*(a) := \operatorname*{sarg\,max}_{b \in [B]} \hat{\lambda}_{a,b;1}$$

end
Compute $\hat{w} = (\hat{w}^{(1)}, \ldots, \hat{w}^{(p)})^{\mathrm{T}}$, where

$$\hat{w}^{(j)} := \frac{1}{A}\sum_{a=1}^{A} (\hat{\lambda}_{a,b^*(a);1} - \hat{\lambda}_{a,b^*(a);2})(\hat{v}^{(j)}_{a,b^*(a);1})^2,$$

and let $\hat{S} \subseteq [p]$ be the index set of the l largest components of $\hat{w}$
Output: $\hat{v}_1 := \operatorname{sarg\,max}_{v \in \mathcal{S}^{p-1}} v^{\mathrm{T}} P_{\hat{S}}\hat{\Sigma}P_{\hat{S}}v$

$P_{a,b}\hat{\Sigma}P_{a,b}$ for all a and b, which can be done in two ways. One option is to compute $\hat{\Sigma}$, and then for each projection $P_{a,b}$ to select the corresponding $d \times d$ principal submatrix of $\hat{\Sigma}$, which requires $\mathcal{O}(np^2 + ABd^2)$ operations. Alternatively, we can avoid computing $\hat{\Sigma}$ by computing the sample covariance matrix of the projected data $\{P_{a,b}x_1, \ldots, P_{a,b}x_n : a \in [A], b \in [B]\}$, which has $\mathcal{O}(ABnd^2)$ complexity. If $p^2 \gg ABd^2$, then the second option is preferable.

The rest of algorithm 1 entails computing an eigendecomposition of each $d \times d$ matrix, and computing $\{b^*(a) : a \in [A]\}$, $\hat{w}$, $\hat{S}$ and $\hat{v}_1$, which altogether amounts to $\mathcal{O}(ABd^3 + Ap + l^3)$ operations. Thus, assuming that $n \geqslant d$, the overall computational complexity of the SPCAvRP algorithm is

$$\mathcal{O}(\min\{np^2 + ABd^3 + Ap + l^3, ABnd^2 + Ap + l^3\}).$$

We also note that, because of the use of random projections, the algorithm is highly parallelizable. In particular, both 'for' loops of algorithm 1 can be parallelized, and the selection of good projections can easily be carried out by using different (up to A) machines.

Finally, we note that the numbers A and B of projections, the dimension d of those projections and the sparsity l of the final estimator need to be provided as inputs to algorithm 1. The effect of these parameter choices on the theoretical guarantees of our SPCAvRP algorithm is elucidated in our theory in Section 3, whereas their practical selection is discussed in Section 4.1.

2.2. Multiple principal component estimation

The estimation of higher order principal components is typically achieved via a deflation scheme. Having computed estimates $\hat{v}_1, \ldots, \hat{v}_{r-1}$ of the top $r-1$ principal components, the aim of such a procedure is to estimate the rth principal component based on modified observations, which have had their correlation with these previously estimated components removed (e.g. Mackey (2009)). For any matrix $V \in \mathbb{R}^{p \times r}$ of full column rank, we define the projection onto the orthogonal complement of the column space of V by $\mathrm{Proj}^{\perp}(V) := I_p - V(V^{\mathrm{T}}V)^{-1}V^{\mathrm{T}}$ if $V \neq 0$ and I_p otherwise. Then, writing $\hat{V}_{r-1} := (\hat{v}_1, \ldots, \hat{v}_{r-1})$, one possibility to implement a deflation scheme is to set $\tilde{x}_i := \mathrm{Proj}^{\perp}(\hat{V}_{r-1})x_i$ for $i \in [n]$. Note that in sparse PCA, by contrast with classical PCA, the estimated principal components from such a deflation scheme are typically

Table 2. Algorithm 2: pseudocode of the modified deflation scheme

Input: $x_1, \ldots, x_n \in \mathbb{R}^p$, $A, B \in \mathbb{N}$, $m, d, l_1, \ldots, l_m \in [p]$
Let $\hat{v}_1$ be the output of algorithm 1 with inputs $x_1, \ldots, x_n$, A, B, d and l_1
for $r = 2, \ldots, m$ *do*
  let $H_r := \mathrm{Proj}^{\perp}(\hat{V}_{r-1})$, where $\hat{V}_{r-1} := (\hat{v}_1, \ldots, \hat{v}_{r-1})$
  let $\tilde{v}_r$ be the output of algorithm 1 with inputs $H_r x_1, \ldots, H_r x_n$, A, B, d and l_r
  let $\tilde{S}_r := \{j \in [p] : \tilde{v}_r^{(j)} \neq 0\}$ and $H_{\tilde{S}_r} := \mathrm{Proj}^{\perp}(P_{\tilde{S}_r} \hat{V}_{r-1})$
  Compute

$$\hat{v}_r := v_1(H_{\tilde{S}_r} P_{\tilde{S}_r} \hat{\Sigma} P_{\tilde{S}_r} H_{\tilde{S}_r})$$

end
Output: $\hat{v}_1, \ldots, \hat{v}_m$

Table 3. Algorithm 3: pseudocode of the SPCAvRP algorithm for eigenspace estimation

Input: $x_1, \ldots, x_n \in \mathbb{R}^p$, $A, B \in \mathbb{N}$, $d, l \in [p]$, $m \in [d]$
Generate $\{P_{a,b} : a \in [A], b \in [B]\}$ independently and uniformly from $\mathcal{P}_d$
Compute $\{P_{a,b} \hat{\Sigma} P_{a,b} : a \in [A], b \in [B]\}$, where $\hat{\Sigma} := n^{-1} \Sigma_{i=1}^n x_i x_i^{\mathrm{T}}$
for $a = 1, \ldots, A$ *do*
  for $b = 1, \ldots, B$ *do*
    for $r \in [m+1]$, compute $\hat{\lambda}_{a,b;r} := \lambda_r(P_{a,b} \hat{\Sigma} P_{a,b})$ and the corresponding eigenvector $\hat{v}_{a,b;r}$, with the convention that $\hat{\lambda}_{a,b;d+1} := 0$
  end
  Compute $b^*(a) := \mathrm{sarg\,max}_{b \in [B]} \Sigma_{r=1}^m \hat{\lambda}_{a,b;r}$
end
Compute $\hat{w} = (\hat{w}^{(1)}, \ldots, \hat{w}^{(p)})^{\mathrm{T}}$ with

$$\hat{w}^{(j)} := \frac{1}{A} \sum_{a=1}^{A} \sum_{r=1}^{m} (\hat{\lambda}_{a,b^*(a);r} - \hat{\lambda}_{a,b^*(a);m+1})(\hat{v}_{a,b^*(a);r}^{(j)})^2$$

Let $\hat{S} \subseteq [p]$ be the index set of the l largest components of $\hat{w}$
Output: $\hat{V}_m = (\hat{v}_1, \ldots, \hat{v}_m)$, where $\hat{v}_1, \ldots, \hat{v}_m$ are the principal eigenvectors of $P_{\hat{S}} \hat{\Sigma} P_{\hat{S}}$

not orthogonal. In algorithm 2 in Table 2, we therefore propose a modified deflation scheme, which in combination with algorithm 1 can be used to compute arbitrary $m \in [p]$ principal components that are orthogonal (as well as sparse), as verified in lemma 1 below.

Lemma 1. For any $m \in [p]$, the outputs $\hat{v}_1, \ldots, \hat{v}_m$ of algorithm 2 are mutually orthogonal.

We remark that, in fact, our proposed deflation method can be used in conjunction with any SPCA algorithm.

Although algorithm 2 can conveniently be used to compute sparse principal components up to order m, it requires algorithm 1 to be executed m times. Instead, we can modify algorithm 1 to estimate directly the leading eigenspace of dimension m—the subspace that is spanned by the columns of matrix $V_m = (v_1, \ldots, v_m)$—at a computational cost that is not much higher than that of executing algorithm 1 only once. For this, we propose a generalization of the SPCAvRP algorithm for eigenspace estimation in algorithm 3 in Table 3. In this generalization, A projections are selected from a total of $A \times B$ random projections, by computing

$$b^*(a) := \mathrm{sarg\,max}_{b \in [B]} \sum_{r=1}^{m} \lambda_r(P_{a,b} \hat{\Sigma} P_{a,b})$$

for each $a \in [A]$. We can regard $\Sigma_{r=1}^m (\hat{\lambda}_{a,b^*(a);r} - \hat{\lambda}_{a,b^*(a);m+1})(\hat{v}_{a,b^*(a);r}^{(j)})^2$ as the contribution of

the ath selected projection to the importance score of the jth co-ordinate, and, analogously to the single-component-estimation case, we average these contributions over $a \in [A]$ to obtain a vector of final importance scores. Again, similarly to the case $m = 1$, we then threshold the top l importance scores to obtain a final projection and our m estimated principal components. A notable difference, then, between algorithm 3 and the deflation scheme (algorithm 2) is that now we estimate the union of the supports of the leading m eigenvectors of Σ simultaneously rather than one at a time. A consequence is that algorithm 3 is particularly well suited to a sparsity setting known in the literature as 'row sparsity' (Vu and Lei, 2013), where leading eigenvectors of interest may share common support, because it borrows strength regarding the estimation of this support from the simultaneous nature of the multiple-component estimation. However, algorithm 2 may have a slight advantage in cases where the leading eigenvectors have disjoint supports; see Section 4.2.2 for further discussion.

Observe that, for $m = 1$, both algorithm 2 and algorithm 3 reduce to algorithm 1. Furthermore, for any m, up to the step where $\hat{w}$ is computed, algorithm 3 has the same complexity as algorithm 1, with the total complexity of algorithm 3 amounting to $\mathcal{O}(\min\{np^2 + ABd^3 + Amp + l^3, ABnd^2 + Amp + l^3\})$, provided that $n \geqslant d$.

3. Theoretical guarantees

In this section, we focus on the general algorithm 3. We assume that $X_1, \ldots, X_n$ are independently sampled from a distribution Q satisfying a restricted covariance concentration (RCC) condition that was introduced in Wang *et al.* (2016). Recall that, for $K > 0$, we say that a zero-mean distribution Q on $\mathbb{R}^p$ satisfies an RCC condition with parameter K, and write $Q \in \mathrm{RCC}_p(K)$, if, for all $\delta > 0$, $n \in \mathbb{N}$ and $r \in [p]$, we have

$$\mathbb{P}\left(\sup_{u \in \mathcal{B}_0^{p-1}(r)} |u^{\mathrm{T}}(\hat{\Sigma} - \Sigma)u| \geqslant K \max\left[\sqrt{\left\{ \frac{r \log(p/\delta)}{n} \right\}}, \frac{r \log(p/\delta)}{n} \right] \right) \leqslant \delta. \tag{6}$$

In particular, if $Q = N_p(0, \Sigma)$, then $Q \in \mathrm{RCC}_p[8\lambda_1\{1 + 9/\log(p)\}]$; and if Q is sub-Gaussian with parameter σ^2, in the sense that $\int_{\mathbb{R}^p} \exp(u^{\mathrm{T}}x)\mathrm{d}Q(x) \leqslant \exp(\sigma^2\|u\|_2^2/2)$ for all $u \in \mathbb{R}^p$, then $Q \in \mathrm{RCC}_p[16\sigma^2\{1 + 9/\log(p)\}]$ (Wang *et al.* (2016), proposition 1).

As mentioned in Section 2.2, our theoretical justification of algorithm 3 does not require that the leading eigenvectors enjoy disjoint supports. Instead, we ask for V_m to have not too many non-zero rows, and for these non-zero rows to have comparable Euclidean norms (i.e. to satisfy an incoherence condition). More precisely, writing $\mathrm{nnzr}(V)$ for the number of non-zero rows of a matrix V, for $\mu \geqslant 1$, we consider the setting where V_m belongs to the set

$$\mathbb{O}_{p,m,k}(\mu) := \left\{ V \in \mathbb{O}_{p,m},\ \mathrm{nnzr}(V) \leqslant k,\ \frac{\max_{j:\|V^{(j,\cdot)}\|_2 \neq 0} \|V^{(j,\cdot)}\|_2}{\min_{j:\|V^{(j,\cdot)}\|_2 \neq 0} \|V^{(j,\cdot)}\|_2} \leqslant \mu \right\}. \tag{7}$$

Writing $S_0 := \{j \in [p] : V_m^{(j,\cdot)} \neq 0\}$ for the set of indices of the non-zero rows of V_m, since $\sum_{j \in S_0} \|V_m^{(j,\cdot)}\|_2^2 = \|V_m\|_{\mathrm{F}}^2 = m$, a consequence of our incoherence parameter definition is that, for $V_m \in \mathbb{O}_{p,m,k}(\mu)$, we have

$$\frac{m^{1/2}}{k^{1/2}\mu} \leqslant \|V_m^{(j,\cdot)}\|_2 \leqslant \frac{m^{1/2}\mu}{k^{1/2}}, \qquad \forall\ j \in S_0. \tag{8}$$

The following theorem is our main result on the performance of our SPCAvRP algorithm.

Theorem 1. Suppose that $Q \in \mathrm{RCC}_p(K)$ has an associated covariance matrix $\Sigma = I_p + V_m \Theta V_m^{\mathrm{T}}$, where $V_m \in \mathbb{O}_{p,m,k}(\mu)$ and $\Theta = \mathrm{diag}(\theta_1, \ldots, \theta_m)$, with $\theta_1 \geqslant \ldots \geqslant \theta_m > 0$. Let $X_1, \ldots,$

$X_n \sim^{\mathrm{IID}} Q$ and let $\hat{V}_m$ be the output of algorithm 3 with input $X_1, \ldots, X_n$, A, B, m, d and l. Suppose that $d \geqslant \max\{m+1, k\}$, $l \geqslant k$, and

$$32K\sqrt{\left\{\frac{d\log(p)}{n}\right\}} \leqslant \frac{\theta_m}{k\mu^2}. \tag{9}$$

Then, with probability at least $1 - 2p^{-3} - p\exp\{-A\theta_m^2/(50p^2\mu^8\theta_1^2)\}$, we have

$$L(\hat{V}_m, V_m) \leqslant 4K\sqrt{\left\{\frac{ml\log(p)}{n\theta_m^2}\right\}}.$$

An immediate consequence of theorem 1 is that, provided that $A \gtrsim p^2\mu^8\theta_1^2\theta_m^{-2}\log(p)$ and

$$p^{-3} \leqslant K\sqrt{\left\{\frac{ml\log(p)}{n\theta_m^2}\right\}},$$

our SPCAvRP algorithm achieves the bound

$$\mathbb{E}\{L(\hat{V}_m, V_m)\} \lesssim K\sqrt{\left\{\frac{ml\log(p)}{n\theta_m^2}\right\}} \tag{10}$$

under the conditions of theorem 1. The salient observation here is that this choice of A, together with the algorithmic complexity analysis given in Section 2.2, ensures that algorithm 3 achieves the rate in bound (10) in polynomial time (provided that we consider μ, θ_1 and θ_m as constants). The minimax lower bound that is given in proposition 1 below complements theorem 1 by showing that this rate is minimax optimal, up to logarithmic factors, over *all* possible estimation procedures, provided that $l \lesssim k$, that $m \lesssim \log(p/k) \asymp \log(p)$ and that we regard K and μ as constants (as well as other regularity conditions). It is important to note that this does not contradict the fundamental statistical and computational trade-off for this problem that was established in Wang *et al.* (2016), because condition (9) ensures that we are in the high effective sample size regime defined in that work. Assuming the planted clique hypothesis from theoretical computer science, this is the only setting in which any (randomized) polynomial time algorithm can be consistent.

The following proposition establishes a minimax lower bound for principal subspace estimation. It is similar to existing minimax lower bounds in the literature for SPCA under row sparsity, e.g. Vu and Lei (2013), theorem 3.1. The main difference is that we show that imposing an incoherence condition on the eigenspace does not make the problem any easier from this minimax perspective. For any $V \in \mathbb{O}_{p,m}$ and $\theta > 0$, we write $P_{V,\theta} := N_p(0, I_p + \theta VV^{\mathrm{T}})$, and recall the definition of $\mathbb{O}_{p,m,k}(\mu)$ from expression (7).

Proposition 1. Assume that $p \geqslant 5k$, $k \geqslant 4m$, $k\log\{(p-m)/k\} \geqslant 17$ and $nm^2\theta^2 \geqslant k^2\max\{m, \log(p/k)\}$. Then

$$\inf_{\tilde{V}} \sup_{V \in \mathbb{O}_{p,m,k}(3)} \mathbb{E}_{P_{V,\theta}}\{L(\tilde{V}, V)\} \gtrsim \sqrt{\left[\frac{k\{m + \log(p/k)\}}{n\theta^2}\right]}$$

where the infimum is taken over all estimators $\tilde{V} = \tilde{V}(X_1, \ldots, X_n)$ and the expectation is with respect to $X_1, \ldots, X_n \sim^{\mathrm{IID}} P_{V,\theta}$.

An interesting aspect of theorem 1 is that the same conclusion holds for every $B \in \mathbb{N}$. It is attractive that we do not need to make any restrictions here; however, we would also expect the statistical performance of the algorithm to improve as B increases. Indeed, this is what we observe empirically; see Fig. 2 in Section 4.1.1. It turns out that we can demonstrate the effect of

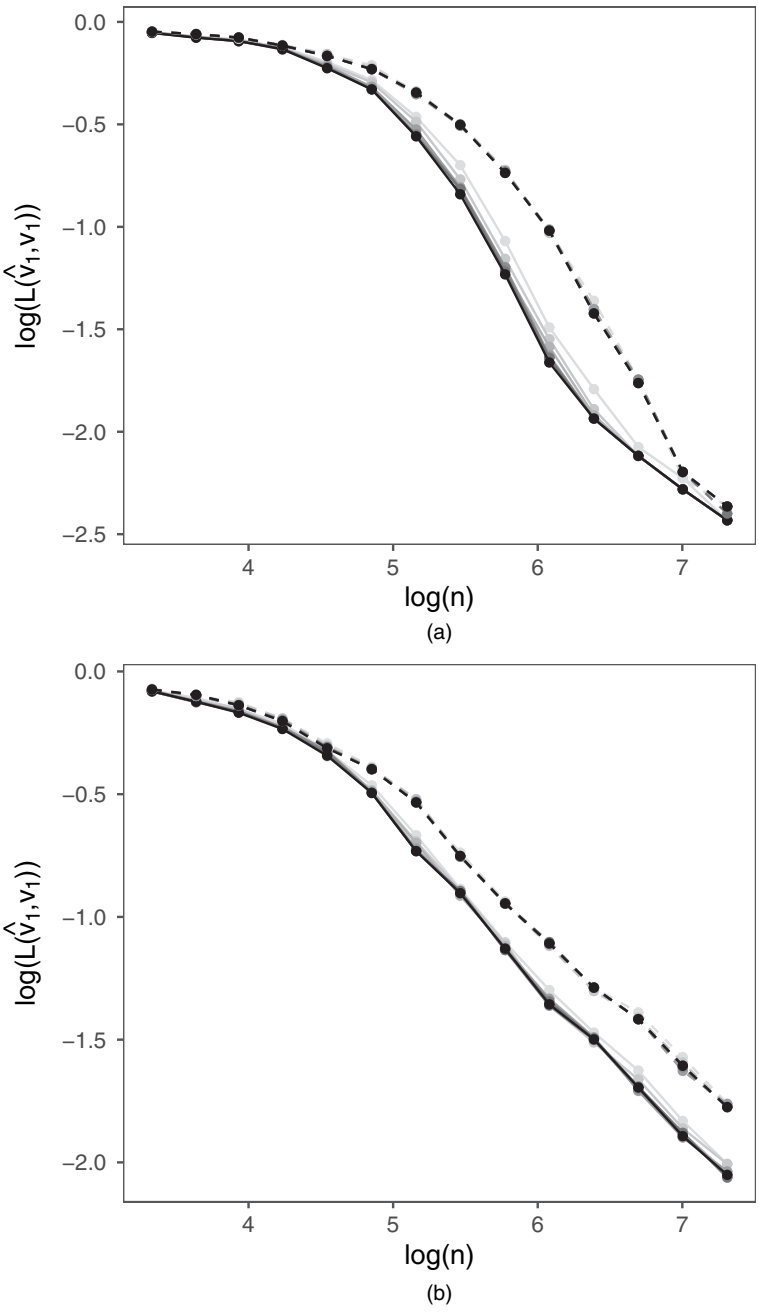


Fig. 2. Average loss $L(\hat{v}_1, v_1)$ against the sample size n, on the log–log-scale, when $B = 1$ (– – –) and $B > 1$ (———) (in each case, n observations are generated from $N_p(0, I_p + v_1 v_1^\top)$, with $p = 50$ and $k = 7$, and the loss $L(\hat{v}_1, v_1)$ is computed for $\hat{v}_1$ as in algorithm 1, with $d = l = k$ and A and B selected as described next, which is then averaged over 100 repetitions; light to dark grey curves, $A \in \{50, 100, 200, 300, 400, 500, 600\}$ and $B = A/2$; light to dark grey broken curves, $A \in \{50 \times 25, 100 \times 50, 200 \times 100, 300 \times 150, 400 \times 200, 500 \times 250, 600 \times 300\}$ and $B = 1$): (a) $v_1 = k^{-1/2}(\mathbf{1}_k^\top, \mathbf{0}_{p-k}^\top)^\top$; (b) $v_1 \propto (k, k-1, \ldots, 1, \mathbf{0}_{p-k}^\top)^\top$

increasing B theoretically in the special setting where all signal co-ordinates have homogeneous signal strength, i.e. $V_m \in \mathbb{O}_{p,m,k}(1)$. As illustrated by the following corollary (to theorem 1) and its proof, as B increases, signal co-ordinates are selected with increasing probability by the best projection within each group of B projections, and this significantly reduces the number of groups A that are required for rate optimal estimation.

Recall that the hypergeometric distribution HyperGeom(d, k, p) models the number of white balls that are obtained when drawing d balls uniformly and without replacement from an urn containing p balls, k of which are white. We write $F_{\mathrm{HG}}(\cdot; d, k, p)$ for its distribution function.

Corollary 1. In addition to the conditions of theorem 1, assume that $\mu = 1$, $\theta_1 = \ldots = \theta_m$ and that $B = \lceil 2^{-1}\{1 - F_{\mathrm{HG}}(t-1; d, k, p)\}^{-1} \rceil$ for some $t \in [k]$. Then

$$\mathbb{P}\left[L(\hat{V}_m, V_m) > 4K\sqrt{\left\{\frac{ml\log(p)}{n\theta_m^2}\right\}}\right] \leqslant 2p^{-3} + p\exp\left(-\frac{At^2}{800k^2}\right).$$

Since, in this corollary, we use lemma 4 in Appendix A.5 instead of expression (16) in Appendix A.2 to control the inclusion probability of signal co-ordinates, the condition $d \geqslant k$ from theorem 1 is in fact no longer needed. We note that, for any fixed t, the function $F_{\mathrm{HG}}(t-1; d, k, p)$ is decreasing with respect to $d \in [p]$. Thus, corollary 1 also illustrates a computational trade-off between the choice of d and B. This trade-off is also demonstrated numerically in Fig. 6 in Section 4.1.2.

Finally, we remark that our algorithm enables us to understand the statistical and computational trade-off in SPCA in a more refined way. Recall that, in the limiting case when $B = \infty$, the estimator that is produced by algorithm 3 (with $d = l = k$ and, for the simplicity of discussion, $m = 1$) is equal to the estimator $\hat{v}_1$ given in problem (2), i.e. the leading k-sparse eigenvector of $\hat{\Sigma}$. In fact, this is already true with high probability for $B \gtrsim \binom{p}{k}$. Hence, for B exponentially large, the SPCAvRP estimator is minimax rate optimal as long as $n \gtrsim mk\theta_m^{-2}\log(p)$, which corresponds to the intermediate effective sample size regime that was defined in Wang *et al.* (2016). For such a choice of B, however, algorithm 3 will not run in polynomial time, which is in agreement with the conclusion of Wang *et al.* (2016) that there is no randomized polynomial time algorithm that can attain the minimax rate of convergence in this intermediate effective sample size regime. In contrast, as mentioned above, SPCAvRP is minimax rate optimal, using only a polynomial number of projections, in the high effective sample size regime as discussed after theorem 1. Therefore, the flexibility in varying the number of projections in our algorithm enables us to analyse its performance in a continuum of scenarios ranging from where consistent estimation is barely possible, through to high effective sample size regimes where the estimation problem is much easier.

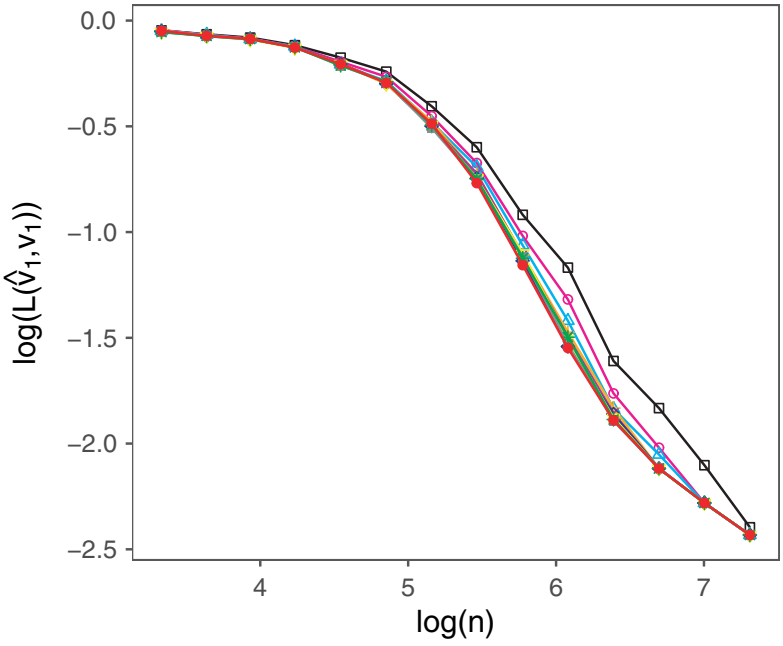
4. Numerical experiments

In this section we demonstrate the performance of our proposed method in different examples and discuss the practical choice of its input parameters. We also compare our method with several existing sparse principal component estimation algorithms on both simulated and experimental data. All experiments were carried out using the R package `SPCAvRP` (Gataric *et al.*, 2018).

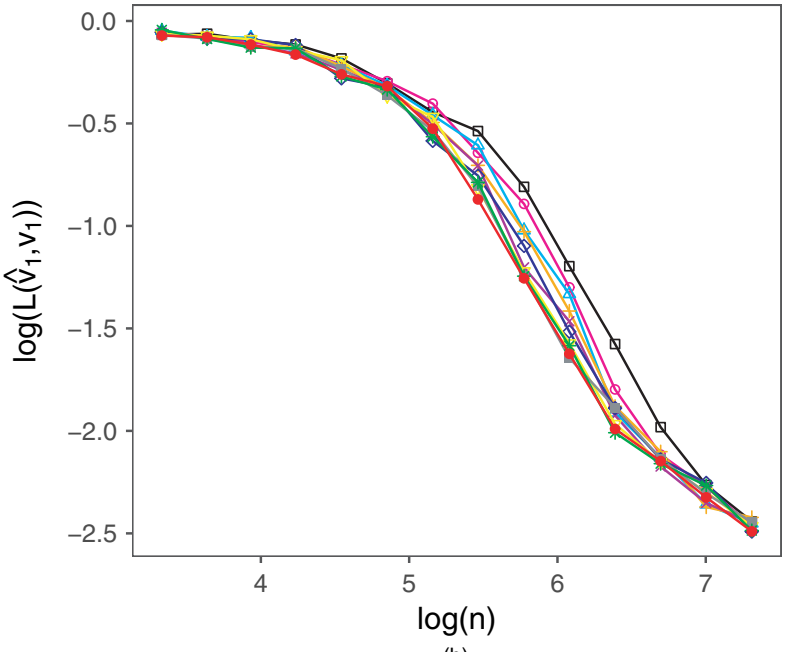
4.1. Choice of input parameters

4.1.1. Choice of A and B

In Fig. 2, we show that choosing $B > 1$, which ensures that we make a non-trivial selection within



(a)



(b)

Fig. 3. Average loss $L(\hat{v}_1, v_1)$ as the sample size n increases for various choices of A or B (the distribution is $N_p(0, I_p + v_1 v_1^{\mathsf{T}})$ with $v_1 = k^{-1/2}(\mathbf{1}_k^{\mathsf{T}}, \mathbf{0}_{p-k}^{\mathsf{T}})^{\mathsf{T}}$, $p = 50$ and $k = 7$, and the other algorithmic parameters are $d = l = 7$): (a) $B = 100$ and A is varied (□, $A = 5$; ○, $A = 10$; △, $A = 15$; +, $A = 20$; ×, $A = 25$; ◇, $A = 30$; ▽, $A = 35$; ■, $A = 40$; ✱, $A = 50$; ●, $A = 100$); (b) $A = 200$ and B is varied (□, $B = 5$; ○, $B = 10$; △, $B = 15$; +, $B = 25$; ×, $B = 40$; ◇, $B = 75$; ▽, $B = 100$; ■, $B = 150$; ✱, $B = 200$; ●, $B = 300$)

each group of projections, considerably improves the statistical performance of the SPCAvRP algorithm. Specifically, we see that, using the same total number of random projections, our two-stage procedure has superior performance over the naive aggregation over all projections, which corresponds to setting $B = 1$ in the SPCAvRP algorithm. Interestingly, Fig. 2 shows that simply increasing the number of projections, without performing a selection step, does not noticeably improve the performance of the basic aggregation. We note that, even for the relatively small choices $A = 50$ and $B = 25$, the SPCAvRP algorithm does significantly better than the naive aggregation over 180000 projections.

Fig. 3 demonstrates the effect of increasing either A or B while keeping the other fixed. We can see from Fig. 3(a) that increasing A steadily improves the quality of estimation, especially in the medium effective sample size regime and when A is relatively small. This agrees with the result in theorem 1, where the bound on the probability of attaining the minimax optimal rate improves as A increases. Thus, in practice, we should choose A to be as large as possible subject to our computational budget. The choice of B, however, is a little more delicate. In some settings, such as the single-spiked homogeneous model in Fig. 3(b), the performance appears to improve as B increases, though the effect is only really noticeable in the intermediate effective sample size regime. In contrast, we can also construct examples where, as B increases, some signal co-ordinates will have increasingly high probability of inclusion compared with other signal co-ordinates, making the latter less easily distinguishable from the noise co-ordinates. Hence the performance does not necessarily improve as B increases; Fig. 4.

In general, we find that A and B should increase with p. On the basis of our numerical experiments, we suggest using $B = \lceil A/3 \rceil$ with $A = 300$ when $p \approx 100$, and $A = 800$ when $p \approx 1000$.

4.1.2. Choice of d

So far in our simulations we have assumed that the true sparsity level k is known and we took the dimension d of the random projections to be equal to k, but in practice k may not be known in advance. In Fig. 5, however, we see that, for a wide range of values of d, the loss curves are relatively close to each other, indicating the robustness of the SPCAvRP algorithm to the choice of d. For the homogeneous signal case, the loss curves for different choices of d merge in the high effective sample size regime, whereas, in the intermediate effective sample size regime, we may in fact see improved performance when d exceeds k. In the inhomogeneous case, the loss curves improve as d increases up to k and then exhibit little dependence on d when $d \geqslant k$.

Although decreasing d reduces computational time, for a smaller choice of d it is then less likely that each signal co-ordinate will be selected in a given random projection. This means that a smaller d will require a larger number of projections A and B to achieve the desired accuracy, thereby increasing computational time. To illustrate this computational trade-off, in Fig. 6, for a single-spiked homogeneous model, we plot the trajectories of the average loss as a function of time (characterized by the choices of A and B), for various choices of d. Broadly speaking, the figures reveal that choosing $d < k$ needs to be compensated by a very large choice of A and B to achieve similar statistical performance to that which can be obtained with d equal to, or even somewhat larger than, k.

In practice, we suggest using $d = k$ where k is known but, when k is not given in advance, we would advocate erring on the side of projecting into a subspace of dimension slightly larger than the level of sparsity of the true eigenvectors, as this enables a significantly smaller choice of A and B, which results in an overall time saving.

4.1.3. Choice of l

The parameter l corresponds to the sparsity of the computed estimator; large values of l increase

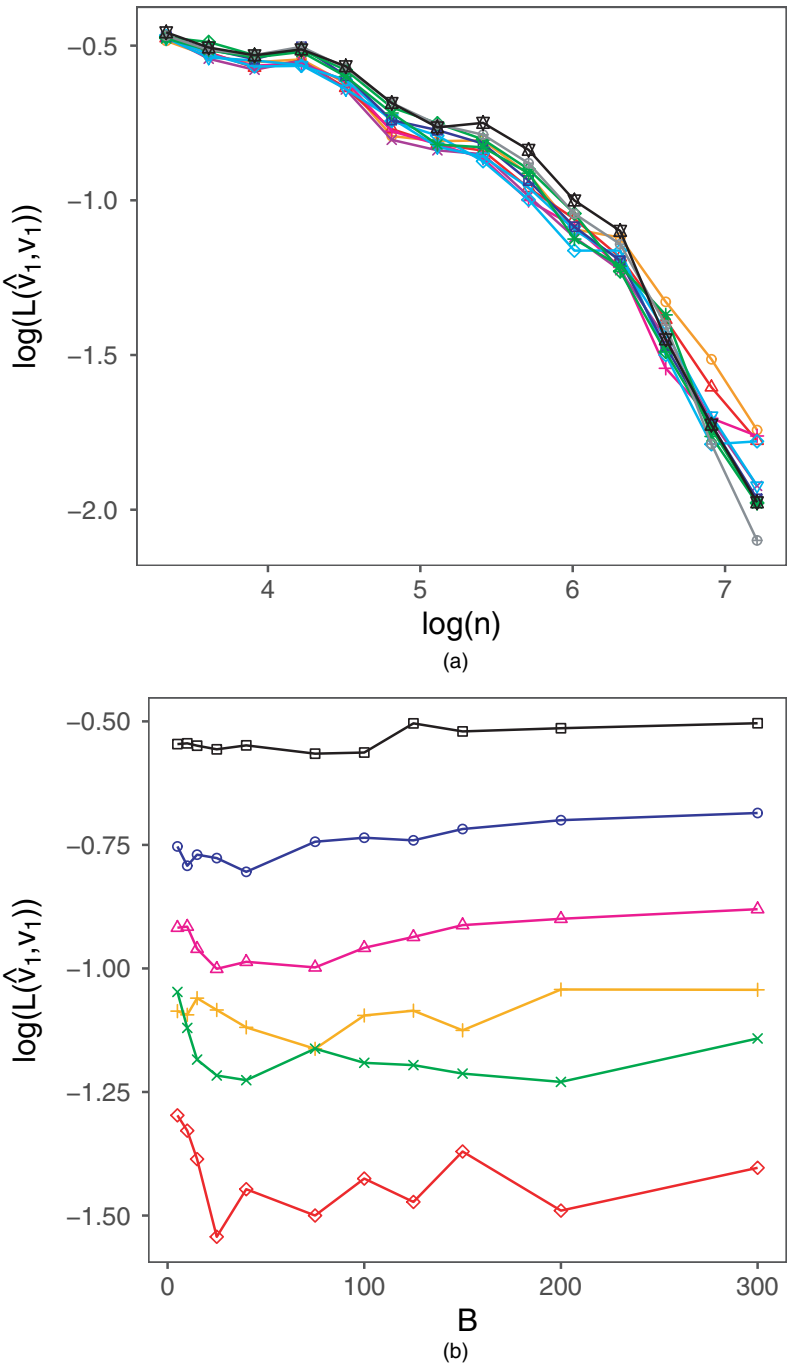


Fig. 4. Trade-off in the choice of B (the distribution is $N_p(0, I_p + 10v_1v_1^\top + 9v_2v_2^\top)$ with $v_1 = k^{-1/2}(\mathbf{1}_k^\top, \mathbf{0}_{p-k}^\top)^\top$, $v_2 = k^{-1/2}(\mathbf{0}_3^\top, -1, 1, -1, 1, -1, 1, 1, \mathbf{0}_{p-k-3}^\top)^\top$, $p = 50$ and $k = 7$, and algorithmic parameters $A = 200$ and $d = l = 7$): (a) average loss as a function of n, on the log–log-scale, where B is varied (○, $B = 10$; △, $B = 15$; +, $B = 25$; ×, $B = 50$; ◇, $B = 75$; ▽, $B = 100$; ⊠, $B = 125$; ✳, $B = 150$; ◈, $B = 200$; ⊕, $B = 250$; ✡, $B = 300$); (b) logarithm of average loss as a function of B, where n is varied (□, $n = 68$; ○, $n = 123$; △, $n = 302$; +, $n = 408$; ×, $n = 551$; ◇, $n = 743$)

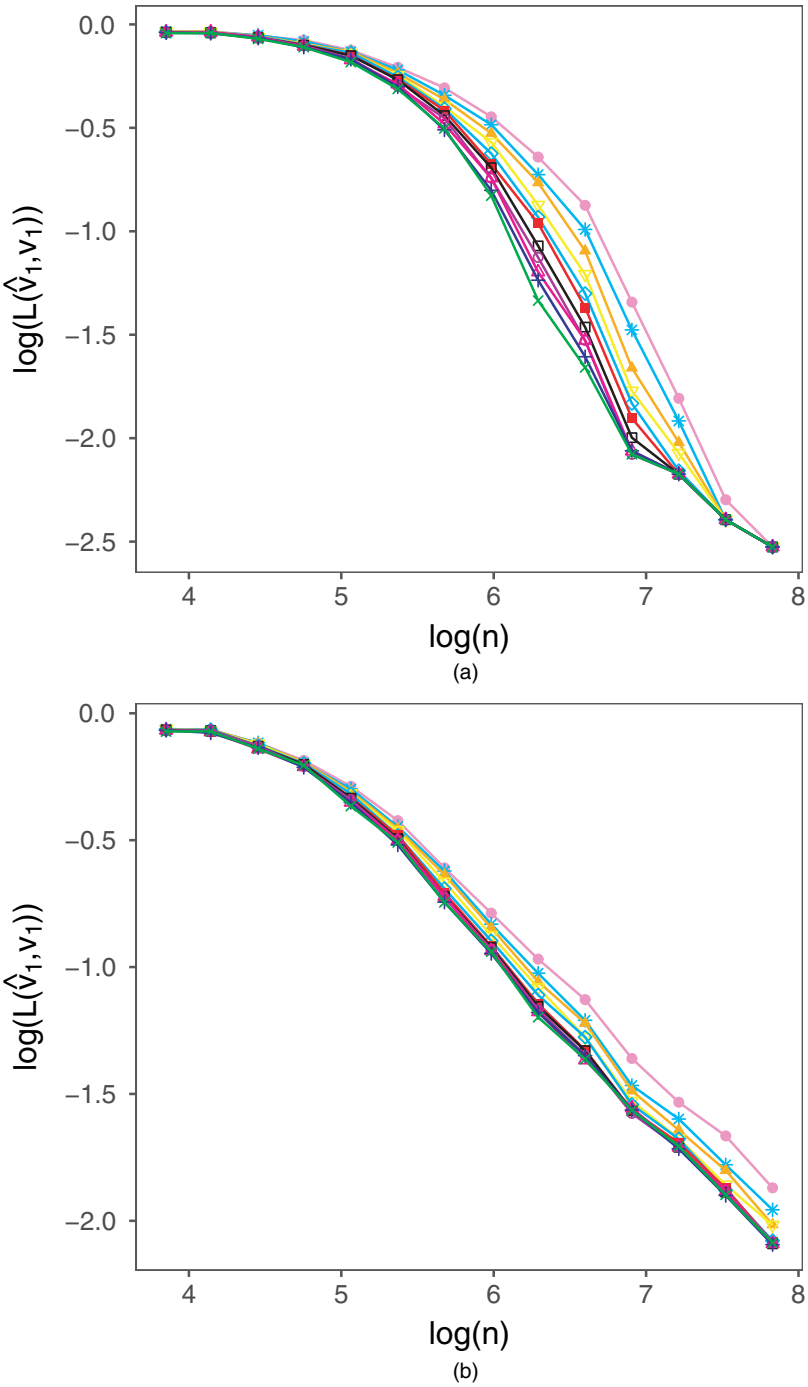


Fig. 5. Average loss $L(\hat{v}_1, v_1)$ as n increases for various choices of d (the distribution is $N_p(0, I_p + v_1 v_1^T)$ with $p = 100$ and $k = 10$; the other algorithmic parameters are $A = 150$, $B = 50$ and $l = k$) (●, $d = k - 5$; ✱, $d = k - 4$; ▲, $d = k - 3$; ▽, $d = k - 2$; ◇, $d = k - 1$; ■, $d = k$; □, $d = k + 1$; ○, $d = k + 2$; △, $d = k + 3$; +, $d = k + 4$; ×, $d = k + 5$): (a) $v_1 = k^{-1/2}(\mathbf{1}_k^T, \mathbf{0}_{p-k}^T)^T$; (b) $v_1 \propto (k, k-1, \ldots, 1, \mathbf{0}_{p-k}^T)^T$

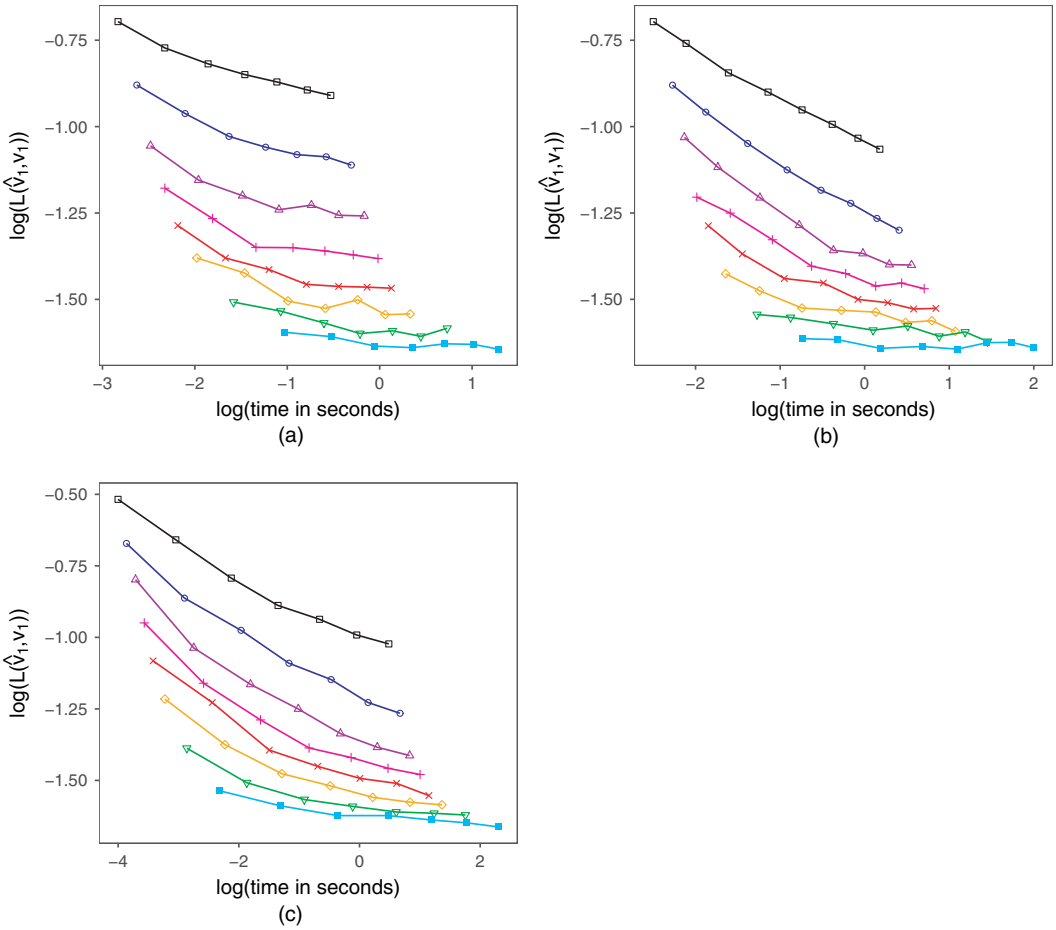


Fig. 6. Computational trade-off in the choice of d and A and B (we generated $n = 600$ observations from distribution $N_p(0, I_p + v_1 v_1^T)$, where $p = 100$, $k = 10$ and $v_1 = k^{-1/2}(\mathbf{1}_k^T, \mathbf{0}_{p-k}^T)^T$; for a fixed $d \in \{4, \ldots, 30\}$ we plot the trajectory realized) (□, $d = 4$; ○, $d = 6$; △, $d = 8$; +, $d = 10$; ×, $d = 12$; ◇, $d = 15$; ▽, $d = 20$; ■, $d = 30$): (a) $A \in \{30, \ldots, 300\}$, $B = 50$; (b) $A = 100$, $B \in \{20, \ldots, 300\}$; (c) $A \in \{30, \ldots, 300\}$, $B = A/2$

the chance that signal co-ordinates are discovered but also increase the probability of including noise co-ordinates. This statistical trade-off is typical for any algorithm that aims to estimate the support of a sparse eigenvector. It is worth noting that many of the SPCA algorithms that are proposed in the literature have a tuning parameter corresponding to the level of sparsity, and thus cross-validation techniques have been proposed in earlier works (e.g. Witten *et al.* (2009)).

A particularly popular approach in the SPCA literature (e.g. Shen and Huang (2008)) is to choose l by inspecting the total variance. More precisely, for each l on a grid of plausible values, we can compute an estimate $\hat{v}_{1,l} \in \mathcal{B}_0^{p-1}(l)$ and its explained variance $\mathrm{var}_l := \hat{v}_{1,l}^T \hat{\Sigma} \hat{v}_{1,l}$, and then plot var_l against l. As can be seen from Fig. 7, var_l increases with l, but plateaus off for $l \geqslant k$. An attractive feature of our algorithm is that this procedure does not significantly increase the total computational time, since there is no need to rerun the entire algorithm for each value of l. Recall that $\hat{w}$ in expression (5) ranks the co-ordinates by their importance. Therefore, we need to compute $\hat{w}$ only once and then to calculate var_l by selecting the top l co-ordinates in $\hat{w}$ for each value of l.

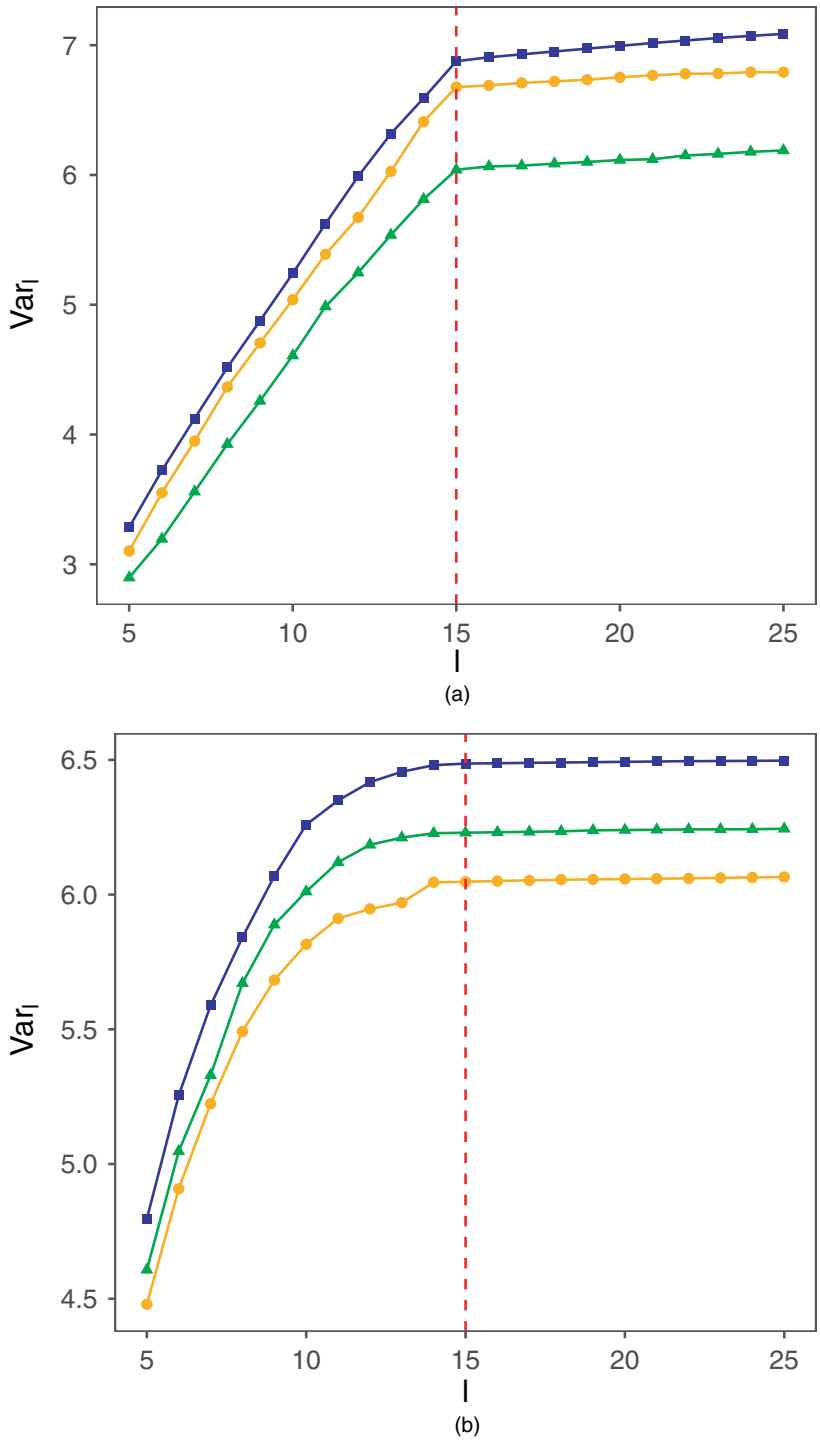


Fig. 7. Selecting l by inspecting the total variance var_l (observations are generated from $N_p(0, I_p + 5v_1v_1^\top)$ with $k = 10$; SPCAvRP is used with parameters $d = 10$, $A = 300$ and $B = 100$) (■, $p = 100$, $n = 3800$; ▲, $p = 500$, $n = 4700$; ●, $p = 1000$, $n = 5000$): (a) $v_1 = k^{-1/2}(\mathbf{1}_k^\top, \mathbf{0}_{p-k}^\top)^\top$; (b) $v_1 \propto (k, k-1, \ldots, 1, \mathbf{0}_{p-k}^\top)^\top$

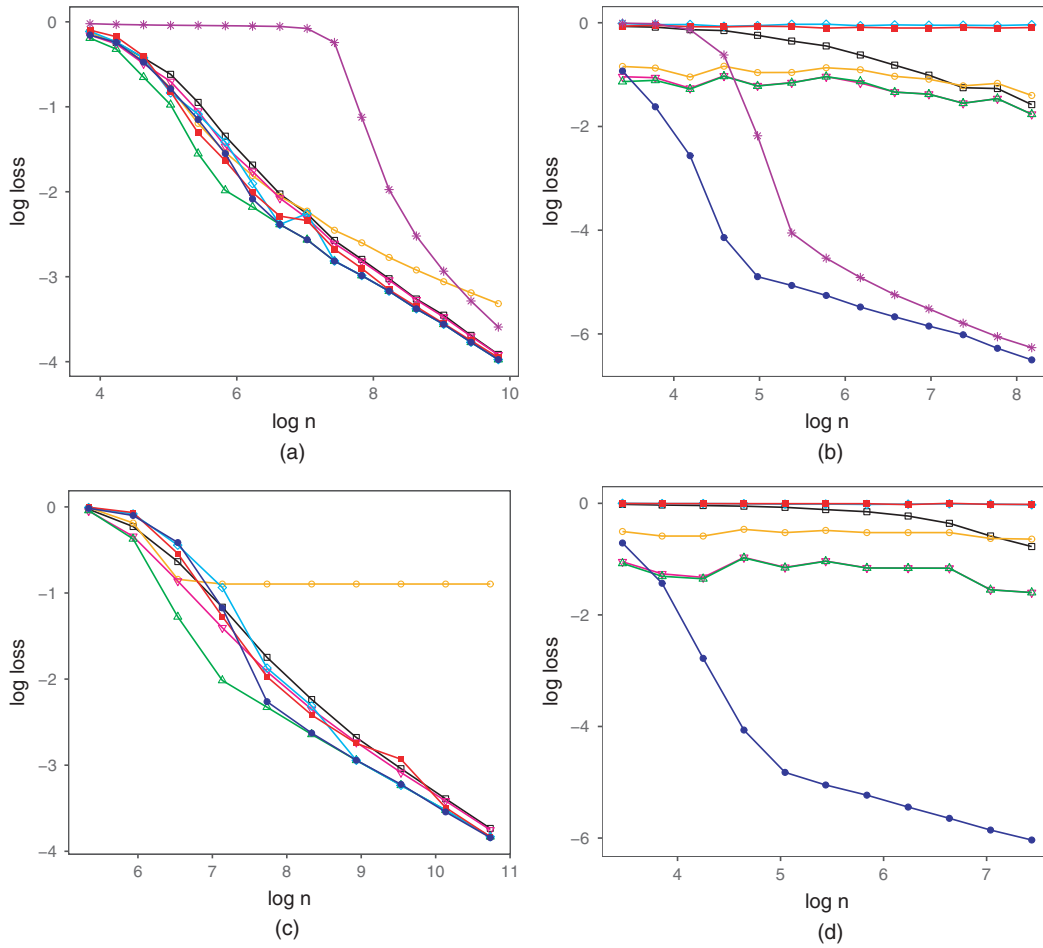


Fig. 8. Comparison of various principal component estimators (average loss against sample size n, on the log–log-scale, using two different covariance structures from expression (11) (□, Zou *et al.* (2006) with given k; ▽, Shen and Huang (2008), l_1-thresholding; △, Shen and Huang (2008), l_0-thresholding; ◇, d'Aspremont *et al.* (2008); ○, Witten *et al.* (2009) with parameters chosen by their default cross-validation; ■, Ma (2013) with the default parameters; ✱, semidefinite programming; ●, SPCAvRP with (a), (b) $A = 300$ and $B = 150$ or (c), (d) $A = 800$ and $B = 300$): (a) $\Sigma_{(1)}$ with $p = 100$ and $k = 10$; (b) $\Sigma_{(2)}$ with $p = 200$ and $k = 10$; (c) $\Sigma_{(1)}$ with $p = 1000$ and $k = 30$ (d) $\Sigma_{(2)}$ with $p = 2000$, $k = 30$

In cases where higher order principal components need to be computed, namely when $m > 1$, we can choose $l = \text{nnzr}(V_m)$ in algorithm 3, and $l_r = \|v_r\|_0$, $r \in [m]$, in algorithm 2, when these quantities are known. If this is not so, we can choose l in algorithm 3 in a similar fashion to that described above, by replacing $\hat{v}_{1,l}$ with $\hat{V}_{m,l}$ where $\text{nnzr}(\hat{V}_{m,l}) \leqslant l$, or we can choose l_r by inspecting the total variance at each iteration r of algorithm 2.

4.2. Comparison with existing methods

In this subsection, we compare our method with several existing approaches for SPCA. We first present examples where only the first principal component is computed, followed by examples of higher order principal component estimation and an illustration on some genetic data.

Table 4. Comparison of various subspace estimators when $m = 2$†

Estimator	$L(\hat{V}_2, V_2)$	$L(\hat{v}_1, v_1)$	$L(\hat{v}_2, v_2)$	$\lvert\hat{v}_1^{\mathrm{T}}\hat{v}_2\rvert$
$S_1 \cap S_2 \neq \emptyset$				
Algorithm 2	8.51×10^{-2}	9.18×10^{-2}	9.58×10^{-2}	$< 10^{-15}$
Algorithm 3	6.72×10^{-2}	1.59×10^{-1}	1.68×10^{-1}	$< 10^{-15}$
Ma (2013)	7.89×10^{-2}	1.51×10^{-1}	1.61×10^{-1}	$< 10^{-15}$
Witten *et al.* (2009)	9.26×10^{-2}	1.50×10^{-1}	1.52×10^{-1}	5.04×10^{-4}
Zou *et al.* (2006)	1.80×10^{-1}	2.06×10^{-1}	2.23×10^{-1}	2.59×10^{-4}
$S_1 \cap S_2 = \emptyset$				
Algorithm 2	5.42×10^{-2}	4.18×10^{-2}	5.32×10^{-2}	$< 10^{-15}$
Algorithm 3	8.03×10^{-2}	1.64×10^{-1}	1.75×10^{-1}	$< 10^{-15}$
Ma (2013)	8.91×10^{-2}	1.43×10^{-1}	1.53×10^{-1}	$< 10^{-15}$
Witten *et al.* (2009)	8.97×10^{-2}	1.11×10^{-1}	1.09×10^{-1}	1.36×10^{-3}
Zou *et al.* (2006)	9.97×10^{-2}	7.13×10^{-2}	9.62×10^{-2}	$< 10^{-15}$

†Observations are generated from $N_p(0, \Sigma)$, $\Sigma = I_p + \sum_{r=1}^{2} \theta_r v_r v_r^{\mathrm{T}}$, $\theta_1 = 50$, $\theta_2 = 30$, $p = 200$ and $n = 150$, where v_1 and v_2 have homogeneous signal strengths with $S_1 = \{1, \ldots, 14\}$, and $S_2 = \{7, \ldots, 20\}$ (top) and $S_2 = \{15, \ldots, 28\}$ (bottom). The SPCAvRP estimators computed by algorithms 2 and 3, with $A = 300$, $B = 150$, $m = 2$, $d = l_1 = l_2 = k$ and $l = |S_1 \cup S_2|$, are compared with estimators computed by algorithms proposed by Zou *et al.* (2006), Witten *et al.* (2009) and Ma (2013), which are used with their default parameters.

4.2.1. First principal component

In addition to the example that was presented in Fig. 1 in Section 1, we consider four further examples with data generated from an $N_p(0, \Sigma)$ distribution, where Σ takes one of the two following forms:

$$\Sigma_{(1)} = \begin{pmatrix} 2J_k & & \\ & J_k & \\ & & \mathbf{0} \end{pmatrix} + I_p, \qquad \Sigma_{(2)} = \begin{pmatrix} kJ_k & & \\ & 0.99kJ_{3k} & \\ & & I_{(p-4k)} \end{pmatrix} + 0.01 I_p, \tag{11}$$

with various choices of $p \in \{100, 200, 1000, 2000\}$ and $k \in \{10, 30\}$. Observe that $v_1 = k^{-1/2}(\mathbf{1}_k^{\mathrm{T}}, \mathbf{0}_{p-k}^{\mathrm{T}})^{\mathrm{T}}$ in all of these examples. The covariance matrix $\Sigma_{(1)}$ is double spiked with $\theta_1 = 2$, $\theta_2 = 1$ and $v_2 = k^{-1/2}(\mathbf{0}_k^{\mathrm{T}}, \mathbf{1}_k^{\mathrm{T}}, \mathbf{0}_{p-2k}^{\mathrm{T}})^{\mathrm{T}}$. We compare the empirical performance of our algorithm with methods proposed by Zou *et al.* (2006), Shen and Huang (2008), d'Aspremont *et al.* (2008), Witten *et al.* (2009) and Ma (2013), as well as the semidefinite programming method that was mentioned in Section 1, by computing the average loss for each algorithm over 100 repetitions on the same set of data. We note that these are all iterative methods, whose success, with the exception of the semidefinite programming method, depends on good initialization, so we recall their default choices. The methods by Zou *et al.* (2006), Shen and Huang (2008) and Witten *et al.* (2009) use eigendecomposition of the sample covariance matrix, i.e. classical PCA, to compute their initial point, whereas d'Aspremont *et al.* (2008) and Ma (2013) selected their initialization according to the largest diagonal entries of $\hat{\Sigma}$.

In Fig. 8, we see that although the average losses of all algorithms decay appropriately with the sample size n in the double-spiked $\Sigma_{(1)}$-setting, most of them perform very poorly in the setting of $\Sigma_{(2)}$, where the spiked structure is absent. Indeed, only the SPCAvRP and SDP algorithms

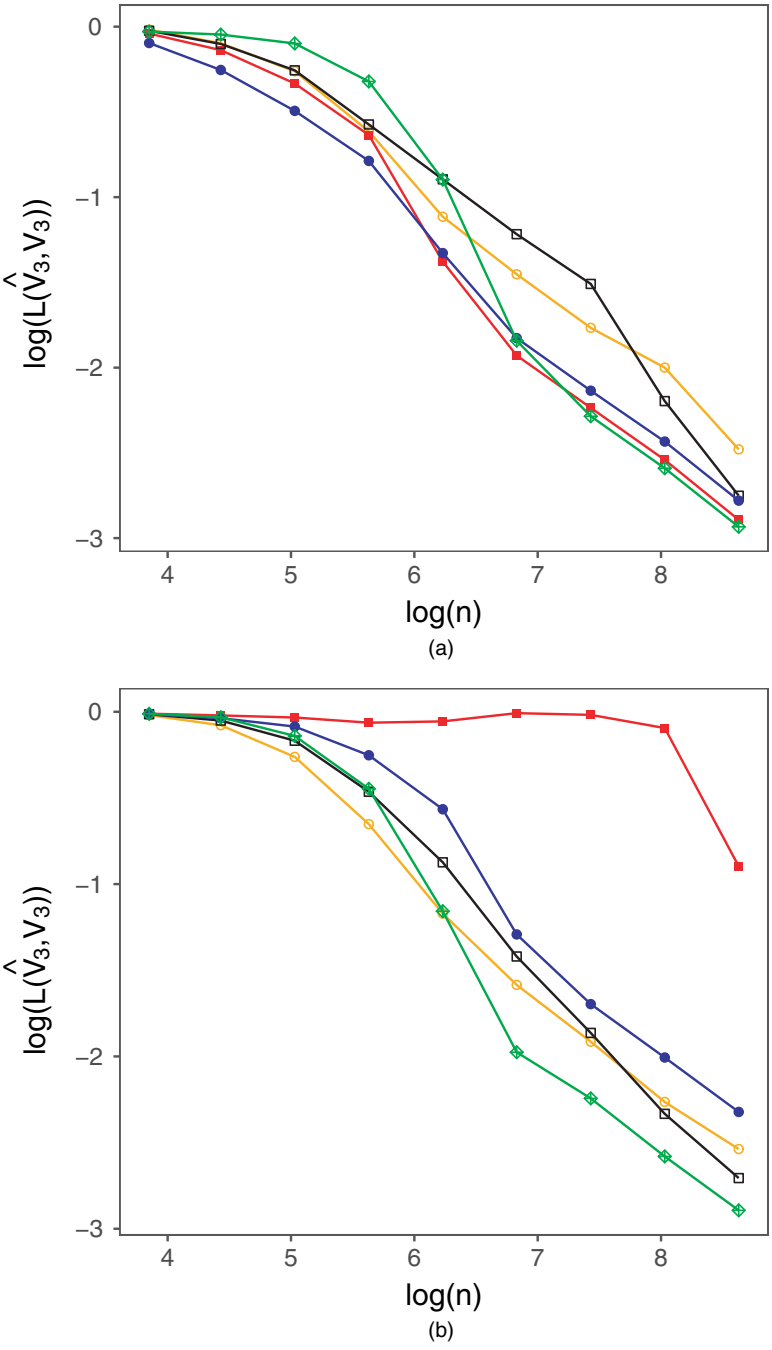


Fig. 9. Comparison of various subspace estimators when $m = 3$ (□, Zou *et al.* (2006); ○, Witten *et al.* (2009); ■, Ma (2013); ◈, SPCAvRP, algorithm 2 ($A = 400$, $B = 200$ and $m = 3$, and $d = I_1 = I_2 = I_3 = 10$); ●, SPCAvRP, algorithm 3 ($A = 400$, $B = 200$ and $m = 3$, and $d = I = |S_1 \cup S_2 \cup S_3|$): average loss $L(\hat{V}_3, V_3)$ is plotted against sample size n, on the log–log-scale; observations are generated from $N_p(0, \Sigma)$, $\Sigma = I_p + \sum_{r=1}^{3} \theta_r v_r v_r^{\top}$, $\theta_1 = 3$, $\theta_2 = 2$, $\theta_3 = 1$ and $p = 100$, where v_1, v_2 and v_3 have homogeneous signals strengths with (a) $S_1 = \{1, \ldots, 10\}$, $S_2 = \{3, \ldots, 12\}$ and $S_3 = \{5, \ldots, 14\}$ or (b) $S_1 = \{1, \ldots, 10\}$, $S_2 = \{11, \ldots, 20\}$ and $S_3 = \{21, \ldots, 30\}$

produce consistent estimators in both settings, but the empirical performance of the SPCAvRP algorithm is much better in both Fig. 8(a) and Fig. 8(b); moreover, since semidefinite programming takes such a long time when $p \in \{1000, 2000\}$, we do not present it in Figs 8(c) and 8(d).

4.2.2. Higher order components

In Table 4 and Fig. 9 we compare algorithms 2 and 3 with existing SPCA algorithms for subspace estimation, namely those proposed by Zou *et al.* (2006), Witten *et al.* (2009) and Ma (2013). For this we simulate observations from a normal distribution with a covariance matrix which is two and three spiked respectively. From Table 4 and Fig. 9, we observe that the SPCAvRP estimators computed by algorithms 2 and 3 perform well when compared with the alternative approaches. When the supports of leading eigenvectors are disjoint, namely $S_r \cap S_q = \emptyset$, $r \neq q$, $r, q \in [m]$, where $S_r := \{j \in [p] : v_r^{(j)} \neq 0\}$, we observe that the deflation scheme that is proposed in algorithm 2 may perform better than algorithm 3, since it estimates each support S_r individually. In contrast, if their supports are overlapping, algorithm 3 may perform better than algorithm

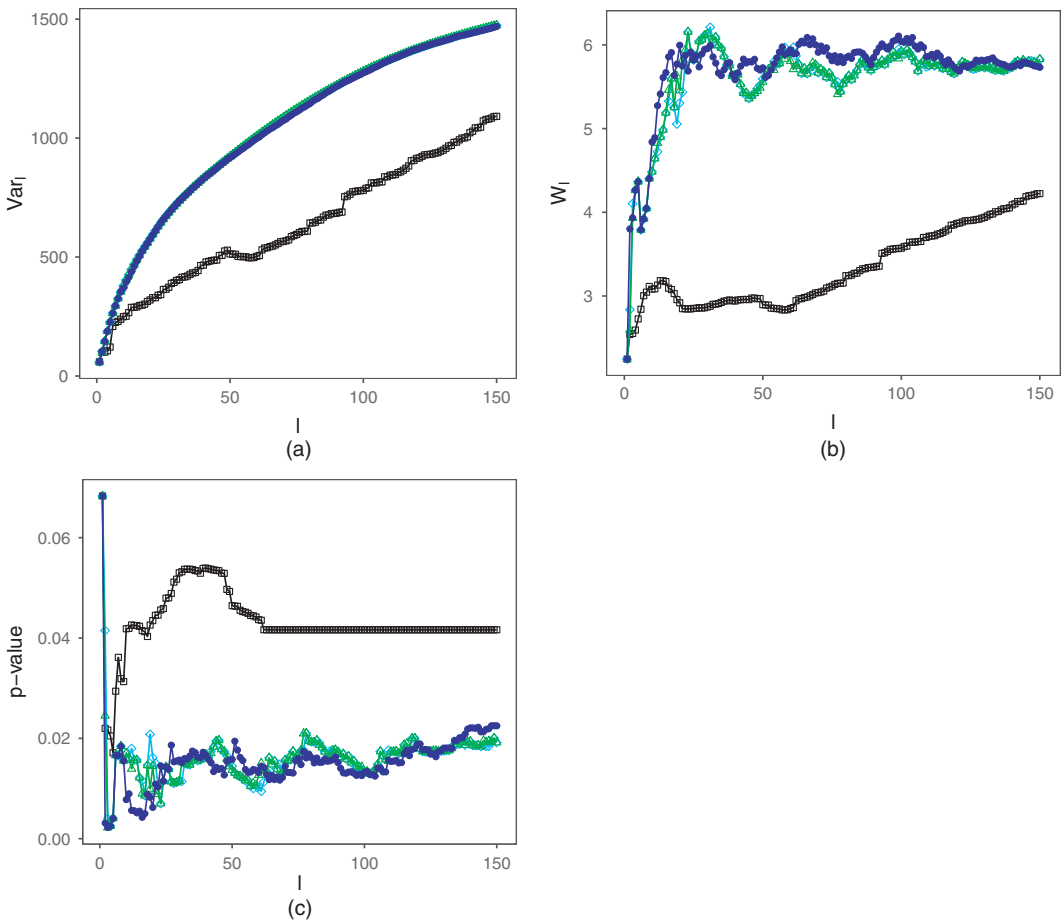


Fig. 10. (a) var_l, (b) Wasserstein distance W_l between the empirical distributions of the two classes projected along $\hat{v}_{1,l}$ and (c) p-value of Welch's t-test for the two classes projected along $\hat{v}_{1,l}$, where $\hat{v}_{1,l}$ is the estimator of v_1 for varied sparsity level l (□, Zou *et al.* (2006); △, Shen and Huang, l_0-thresholding; ◇, d'Aspremont *et al.* (2008); ●, SPCAvRP ($d = 30$, $A = 1200$ and $B = 200$))

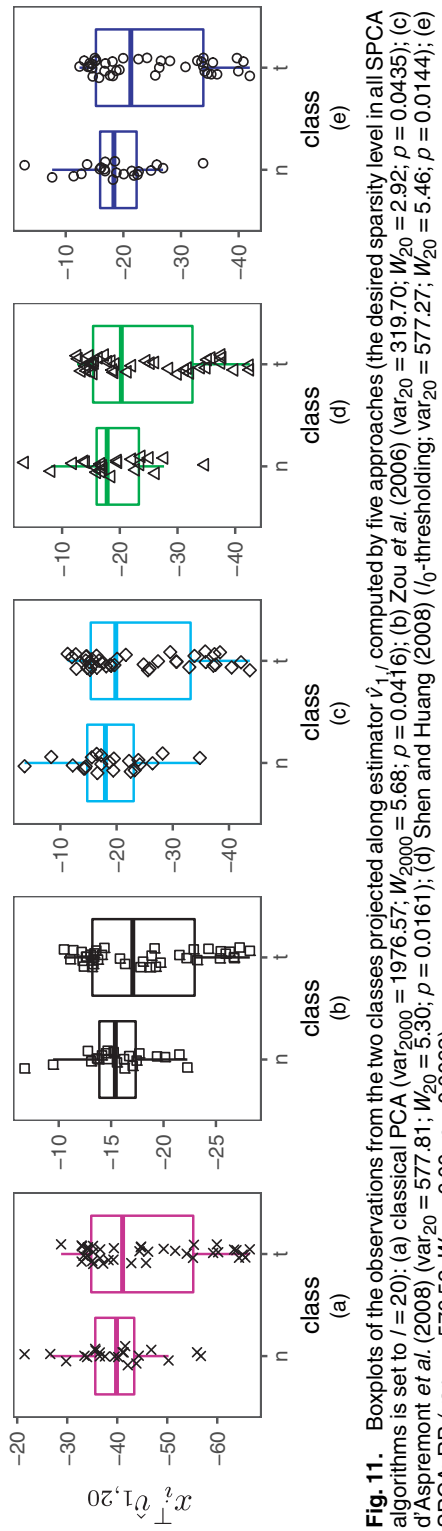


Fig. 11. Boxplots of the observations from the two classes projected along estimator $\hat{v}_{1,l}$ computed by five approaches (the desired sparsity level in all SPCA algorithms is set to $l = 20$): (a) classical PCA ($\text{var}_{2000} = 1976.57$; $W_{2000} = 5.68$; $p = 0.0416$); (b) Zou *et al.* (2006) ($\text{var}_{20} = 319.70$; $W_{20} = 2.92$; $p = 0.0435$); (c) d'Aspremont *et al.* (2008) ($\text{var}_{20} = 577.81$; $W_{20} = 5.30$; $p = 0.0161$); (d) Shen and Huang (2008) (l_0-thresholding; $\text{var}_{20} = 577.27$; $W_{20} = 5.46$; $p = 0.0144$); (e) SPCAvRP ($\text{var}_{20} = 576.52$; $W_{20} = 6.00$; $p = 0.0062$)

2, since it directly estimates $\cup_{r=1}^{m} S_r$. From Table 4, we also see that only SPCAvRP algorithms and the algorithm that was proposed by Ma (2013) compute components that are orthogonal in both cases $S_1 \cap S_2 = \emptyset$ and $S_1 \cap S_2 \neq \emptyset$.

4.2.3. *Microarray data*

We test our SPCAvRP algorithm on the Alon *et al.* (1999) gene expression data set, which contains 40 colon tumour and 22 normal observations. A preprocessed data set can be downloaded from the R package `datamicroarray` (Ramey, 2016), with a total of $p = 2000$ features and $n = 62$ observations. For comparison with alternative SPCA approaches, we use algorithms that accept the output sparsity l as an input parameter, namely those proposed by Zou *et al.* (2006), d'Aspremont *et al.* (2008) and Shen and Huang (2008). For each l considered, we computed the estimator $\hat{v}_{1,l}$ of the first principal component, and in Fig. 10 we plot the explained variance $\mathrm{var}_l := \hat{v}_{1,l}^{\mathrm{T}} \hat{\Sigma} \hat{v}_{1,l}$ as well as two different metrics for the separability of the two classes of observations projected along the first principal component $\hat{v}_{1,l}$, namely the Wasserstein distance W_l of order 1 and the p-value of Welch's t-test (Welch, 1947). Furthermore, in Fig. 11, we display their corresponding values for $l = 20$ together with the boxplots of the observations projected along $\hat{v}_{1,20}$. From Figs 10 and 11, we observe that the SPCAvRP algorithm performs similarly to those proposed by d'Aspremont *et al.* (2008) and Shen and Huang (2008), all of which are superior in this instance to the SPCA algorithm of Zou *et al.* (2006). In particular, for small values of l, we observe a steep slope of the blue Wasserstein and p-value curves corresponding to the SPCAvRP algorithm in Fig. 10, indicating that the two classes are well separated by projecting the observations along the estimated principal component which contains expression levels of only a few different genes.

Acknowledgements

The research of the first and third authors was supported by an Engineering and Physical Sciences Research Council grant EP/N014588/1 for the Centre for Mathematical and Statistical Analysis of Multimodal Clinical Imaging. The second and third authors were supported by Engineering and Physical Sciences Research Council Fellowships EP/J017213/1 and EP/P031447/1, and grant RG81761 from the Leverhulme Trust. The authors also thank the Isaac Newton Institute for Mathematical Sciences for its hospitality during the programme 'Statistical scalability', which was supported by Engineering and Physical Sciences Research Council grants LNAG/036 and RG91310. We thank the reviewers for their helpful and constructive comments.

Appendix A: Proofs of theoretical results

A.1. *Proof of lemma 1*

To verify that $\hat{v}_r$ is orthogonal to $\hat{v}_1, \ldots, \hat{v}_{r-1}$, observe that, since the support of $\hat{v}_r$ is contained in $\tilde{S}_r$, we have

$$\hat{v}_r^{\mathrm{T}} \hat{V}_{r-1} = \hat{v}_r^{\mathrm{T}} P_{\tilde{S}_r} \hat{V}_{r-1} + \hat{v}_r^{\mathrm{T}} P_{\tilde{S}_r^c} \hat{V}_{r-1} \frac{\hat{v}_r^{\mathrm{T}} H_{\tilde{S}_r} P_{\tilde{S}_r} \hat{\Sigma} P_{\tilde{S}_r} H_{\tilde{S}_r} P_{\tilde{S}_r} \hat{V}_{r-1}}{\lambda_1(H_{\tilde{S}_r} P_{\tilde{S}_r} \hat{\Sigma} P_{\tilde{S}_r} H_{\tilde{S}_r})} = 0,$$

where the final equality follows from the fact that $H_{\tilde{S}_r}$ is a projection onto the orthogonal complement of the column space of $P_{\tilde{S}_r} \hat{V}_{r-1}$, so $H_{\tilde{S}_r} P_{\tilde{S}_r} \hat{V}_{r-1} = 0$.

A.2. *Proof of theorem 1*

For notational simplicity, we drop the subscript m from $\hat{V}$ and V in this proof, write $X := (X_1, \ldots, X_n)$ and

define $\binom{[p]}{d} := \{S \subseteq [p] : |S| = d\}$. For any $S \in \binom{[p]}{d}$, we note that $\Sigma^{(S,S)} = I_d + V^{(S,\cdot)}\Theta(V^{(S,\cdot)})^{\mathrm{T}}$ is a rank (at most) m perturbation of the identity. Hence,

$$\sum_{r=1}^{m} \lambda_r(\Sigma^{(S,S)}) = \mathrm{tr}(\Sigma^{(S,S)}) - (d-m) = m + \mathrm{tr}\{V^{(S,\cdot)}\Theta(V^{(S,\cdot)})^{\mathrm{T}}\} = m + \sum_{r=1}^{m}\sum_{j \in S \cap S_0} \theta_r (V^{(j,r)})^2. \tag{12}$$

By the definition of $\mathrm{RCC}_p(K)$ in expression (6), there is an event Ω_{RCC} with probability at least $1 - 2p^{-3}$ such that, on Ω_{RCC}, we have

$$\sup_{u \in \mathcal{B}_0^{p-1}(d)} u^{\mathrm{T}}(\hat{\Sigma} - \Sigma)u \leqslant 2K\sqrt{\left\{\frac{d\log(p)}{n}\right\}}$$

and

$$\sup_{u \in \mathcal{B}_0^{p-1}(l)} u^{\mathrm{T}}(\hat{\Sigma} - \Sigma)u \leqslant 2K\sqrt{\left\{\frac{l\log(p)}{n}\right\}}.$$

On Ω_{RCC}, by equation (12), Weyl's inequality (Weyl (1912) and Stewart and Sun (1990), corollary IV.4.9) and expression (9), we have for any $S \in \binom{[p]}{d}$ that

$$\begin{aligned}\left|\sum_{r=1}^{m} \lambda_r(\hat{\Sigma}^{(S,S)}) - m - \sum_{r=1}^{m}\sum_{j \in S \cap S_0} \theta_r (V^{(j,r)})^2\right| &= \left|\sum_{r=1}^{m}\{\lambda_r(\hat{\Sigma}^{(S,S)}) - \lambda_r(\Sigma^{(S,S)})\}\right| \\ &\leqslant 2Km\sqrt{\left\{\frac{d\log(p)}{n}\right\}} \leqslant \frac{m\theta_m}{16k\mu^2}. \end{aligned} \tag{13}$$

By expression (8), we have $\sum_{r=1}^{m} \theta_r (V^{(j,r)})^2 \geqslant \theta_m \|V^{(j,\cdot)}\|_2^2 \geqslant m\theta_m k^{-1}\mu^{-2}$ for every $j \in S_0$, which is more than twice the right-hand side of inequality (13). Thus, an important consequence of inequality (13) is that on Ω_{RCC}, for any $S, S' \in \binom{[p]}{d}$,

$$\text{if } S \cap S_0 \subsetneq S' \cap S_0, \qquad \text{then } \sum_{r=1}^{m} \lambda_r(\hat{\Sigma}^{(S,S)}) < \sum_{r=1}^{m} \lambda_r(\hat{\Sigma}^{(S',S')}). \tag{14}$$

Fix $a \in [A]$, and for any $\tilde{j} \in [p]$ define $q_{\tilde{j}} := \mathbb{P}(\tilde{j} \in S_{a,b^*(a)} | X)$. Now, fix some $j \in S_0$ and $j' \in [p] \setminus S_0$. We claim that

$$q_j \geqslant q_{j'} \qquad \text{on } \Omega_{\mathrm{RCC}}. \tag{15}$$

Before proving the claim, we first observe that, if result (15) holds, then, since the same inequality would hold if we replace j' by any other index in S_0^{c}, we would have on Ω_{RCC} that

$$q_j \geqslant \frac{\sum_{\tilde{j} \in ([p] \setminus S_0) \cup \{j\}} q_{\tilde{j}}}{p-k+1} = \frac{d - \sum_{\tilde{j} \in S_0 \setminus \{j\}} q_{\tilde{j}}}{p-k+1} \geqslant \frac{d-k+1}{p-k+1} \geqslant \frac{1}{p}. \tag{16}$$

To verify the claim, define for $\tilde{j} \in \{j, j'\}$ and $b \in [B]$ the following sets:

$$\mathcal{S}_{b,\tilde{j}} := \{(S_{a,1}, \ldots, S_{a,B}) : b^*(a) = b,\ \tilde{j} \in S_{a,b}\}$$

and

$$\mathcal{S}_b := \{(S_{a,1}, \ldots, S_{a,B}) : b^*(a) = b\}.$$

Let $\psi : \binom{[p]}{d} \to \binom{[p]}{d}$ be defined such that $\psi(S) := (S \setminus \{j'\}) \cup \{j\}$ if $j' \in S$ and $j \notin S$ and $\psi(S) := S$ otherwise. Since, for every $S \in \binom{[p]}{d}$, we have either $\psi(S) = S$ or $S \cap S_0 \subsetneq \psi(S) \cap S_0$, by result (14) we have on Ω_{RCC} that $\sum_{r=1}^{m} \lambda_r(\hat{\Sigma}^{(S,S)}) \leqslant \sum_{r=1}^{m} \lambda_r(\hat{\Sigma}^{(\psi(S),\psi(S))})$. Thus, for any $b \in [B]$ and any fixed $\hat{\Sigma}$ satisfying Ω_{RCC}, the map ψ induces an injection $\Psi : \mathcal{S}_{b,j'} \to \mathcal{S}_{b,j}$, given by $\Psi(S_{a,1}, \ldots, S_{a,B}) := (S_{a,1}, \ldots, S_{a,b-1}, \psi(S_{a,b}), S_{a,b+1}, \ldots, S_{a,B})$, which in particular means that $|\mathcal{S}_{b,j'}| \leqslant |\mathcal{S}_{b,j}|$. Therefore, on Ω_{RCC}, we have for all $b \in [B]$ that

$$\begin{aligned}\mathbb{P}\{j \in S_{a,b^*(a)} | X, b^*(a) = b\} &= \frac{\mathbb{P}\{j \in S_{a,b^*(a)}, b^*(a) = b | X\}}{\mathbb{P}\{b^*(a) = b | X\}} = \frac{|\mathcal{S}_{b,j}|}{|\mathcal{S}_b|} \\ &\geqslant \frac{|\mathcal{S}_{b,j'}|}{|\mathcal{S}_b|} = \frac{\mathbb{P}\{j' \in S_{a,b^*(a)}, b^*(a) = b | X\}}{\mathbb{P}\{b^*(a) = b | X\}} = \mathbb{P}\{j' \in S_{a,b^*(a)} | X, b^*(a) = b\},\end{aligned}$$

and consequently $q_j \geqslant q_{j'}$ as claimed in expression (15).

For $b \in [B]$ and $r \in [d]$, define $v_{a,b;r} := v_r(P_{a,b}\Sigma P_{a,b})$ and $\lambda_{a,b;r} := \lambda_r(P_{a,b}\Sigma P_{a,b})$. Note that $\lambda_{a,b;m+1} = \ldots = \lambda_{a,b;d} = 1$. Write $V_{a,b} := (v_{a,b;1}, \ldots, v_{a,b;m})$, $\hat{V}_{a,b} := (\hat{v}_{a,b;1}, \ldots, \hat{v}_{a,b;m})$, $\Theta_{a,b} := \mathrm{diag}(\lambda_{a,b;1} - \lambda_{a,b;m+1}, \ldots, \lambda_{a,b;m} - \lambda_{a,b;m+1})$ and $\hat{\Theta}_{a,b} := \mathrm{diag}(\hat{\lambda}_{a,b;1} - \hat{\lambda}_{a,b;m+1}, \ldots, \hat{\lambda}_{a,b;m} - \hat{\lambda}_{a,b;m+1})$. By lemma 2 in Appendix A.5, on Ω_{RCC}, we have for all $\tilde{j} \in \{j, j'\}$ that

$$|(\hat{V}_{a,b^*(a)}\hat{\Theta}_{a,b^*(a)}\hat{V}_{a,b^*(a)}^{\mathrm{T}})^{(\tilde{j},\tilde{j})} - (V_{a,b^*(a)}\Theta_{a,b^*(a)}V_{a,b^*(a)}^{\mathrm{T}})^{(\tilde{j},\tilde{j})}| \leqslant 4m\|P_{a,b^*(a)}(\hat{\Sigma}-\Sigma)P_{a,b^*(a)}\|_{\mathrm{op}} \leqslant 8Km\sqrt{\left\{\frac{d\log(p)}{n}\right\}} \leqslant \frac{m\theta_m}{4k\mu^2}, \tag{17}$$

where we used expression (9) in the last inequality. Observe that

$$V_{a,b^*(a)}\Theta_{a,b^*(a)}V_{a,b^*(a)}^{\mathrm{T}} = \sum_{r=1}^{d}(\lambda_{a,b^*(a);r} - 1)v_{a,b^*(a);r}v_{a,b^*(a);r}^{\mathrm{T}} = P_{a,b^*(a)}(\Sigma - I_p)P_{a,b^*(a)}. \tag{18}$$

Also, we have

$$(\hat{V}_{a,b^*(a)}\hat{\Theta}_{a,b^*(a)}\hat{V}_{a,b^*(a)}^{\mathrm{T}})^{(\tilde{j},\tilde{j})} = \sum_{r=1}^{m}(\hat{\lambda}_{a,b^*(a);r} - \hat{\lambda}_{a,b^*(a);m+1})(\hat{v}_{a,b^*(a);r}^{(\tilde{j})})^2 =: \hat{w}_a^{(\tilde{j})}. \tag{19}$$

By expressions (8), (17), (18) and (19), we have on $\Omega_{\mathrm{RCC}} \cap \{j \in S_{a,b^*(a)}\}$ that

$$\begin{aligned}\frac{3m\theta_m}{4k\mu^2} \leqslant \theta_m\|V^{(j,\cdot)}\|_2^2 - \frac{m\theta_m}{4k\mu^2} \leqslant \Sigma^{(j,j)} - 1 - \frac{m\theta_m}{4k\mu^2} \leqslant \hat{w}_a^{(j)} \\ \leqslant \Sigma^{(j,j)} - 1 + \frac{m\theta_m}{4k\mu^2} \leqslant \theta_1\|V^{(j,\cdot)}\|_2^2 + \frac{m\theta_m}{4k\mu^2} \leqslant \frac{5m\theta_1\mu^2}{4k}.\end{aligned} \tag{20}$$

Moreover, on $\Omega_{\mathrm{RCC}} \cap \{j' \in S_{a,b^*(a)}\}$, we have

$$-\frac{m\theta_m}{4k\mu^2} \leqslant \hat{w}_a^{(j')} \leqslant \frac{m\theta_m}{4k\mu^2}. \tag{21}$$

Recall that for all $j \in [p]$, if $j \notin S_{a,b^*(a)}$, then $\hat{w}_a^{(j)} = 0$. Combining the lower bound on $\hat{w}_a^{(j)}$ in inequality (20) and the upper bound on $\hat{w}_a^{(j')}$ in inequality (21), we have by inequalities (15) and (16) that, on Ω_{RCC},

$$\mathbb{E}(\hat{w}_a^{(j)} - \hat{w}_a^{(j')}|X) = \mathbb{E}(\hat{w}_a^{(j)}\mathbb{1}_{\{j\in S_{a,b^*(a)}\}} - \hat{w}_a^{(j')}\mathbb{1}_{\{j'\in S_{a,b^*(a)}\}}|X) \geqslant \frac{q_j m\theta_m}{2k\mu^2} \geqslant \frac{m\theta_m}{2pk\mu^2}. \tag{22}$$

Now, let a, j and j' be freely varying again, and define $\Omega := \{\min_{j\in S_0}\hat{w}^{(j)} > \max_{j\notin S_0}\hat{w}^{(j)}\}$. Since inequality (22) holds for arbitrary $j \in S_0$ and $j' \notin S_0$, and, since $\hat{w}^{(j)} = A^{-1}\sum_{a=1}^{A}\hat{w}_a^{(j)}$, we have

$$\Omega^{\mathrm{c}} \subseteq \bigcup_{j\in S_0}\left\{\hat{w}^{(j)} - \mathbb{E}(\hat{w}^{(j)}|X) \leqslant -\frac{m\theta_m}{4pk\mu^2}\right\} \cup \bigcup_{j'\notin S_0}\left\{\hat{w}^{(j')} - \mathbb{E}(\hat{w}^{(j')}|X) \geqslant \frac{m\theta_m}{4pk\mu^2}\right\}.$$

Observe that $(\hat{w}_a^{(j)} : a \in [A])$ are independent and identically distributed conditionally on X. By inequalities (20) and (21), $\hat{w}_a^{(j)}$ is bounded on Ω_{RCC} for all $j \in [p]$. Thus, we can use a union bound and apply Hoeffding's inequality conditionally on X to obtain that, on Ω_{RCC},

$$\mathbb{P}(\Omega^{\mathrm{c}}|X) \leqslant p\exp\left\{-\frac{A}{2}\left(\frac{m\theta_m}{4pk\mu^2}\right)^2 \Big/ \left(\frac{5m\theta_1\mu^2}{4k}\right)^2\right\} \leqslant p\exp\left(-\frac{A\theta_m^2}{50p^2\mu^8\theta_1^2}\right).$$

Since $l \geqslant k$, on Ω, we have $\hat{S} \supseteq S_0$. Therefore, by Yu *et al.* (2015), theorem 2, on $\Omega_{\mathrm{RCC}} \cap \Omega$,

$$L(\hat{V}, V) \leqslant \frac{2m^{1/2}\|P_{\hat{S}}(\hat{\Sigma}-\Sigma)P_{\hat{S}}\|_{\mathrm{op}}}{\theta_m} \leqslant 4K\sqrt{\left\{\frac{ml\log(p)}{n\theta_m^2}\right\}}.$$

The desired result follows from the fact that

$$\mathbb{P}(\Omega_{\mathrm{RCC}} \cap \Omega) \geqslant 1 - \mathbb{P}(\Omega_{\mathrm{RCC}}^{\mathrm{c}}) - \mathbb{E}\{\mathbb{P}(\Omega^{\mathrm{c}}|X)\mathbb{1}_{\Omega_{\mathrm{RCC}}}\} \geqslant 1 - 2p^{-3} - p\exp\left(-\frac{A\theta_m^2}{50p^2\mu^8\theta_1^2}\right).$$

A.3. Proof of proposition 1

Let $\mathbb{O}_{p,m,k} := \{V \in \mathbb{O}_{p,m} : \mathrm{nnzr}(V) \leqslant k\}$. Writing $k = qm + h$ for $q \in \mathbb{N}$ and $h \in \{0, \ldots, m-1\}$, for $r \in [m]$, we define

$$u_r := \begin{cases} (q+1)^{-1/2}(\mathbf{0}^{\mathrm{T}}_{(r-1)(q+1)}, \mathbf{1}^{\mathrm{T}}_{q+1}, \mathbf{0}^{\mathrm{T}}_{p-r(q+1)})^{\mathrm{T}} & \text{if } 1 \leqslant r \leqslant h, \\ q^{-1/2}(\mathbf{0}^{\mathrm{T}}_{h(q+1)+(r-h-1)q}, \mathbf{1}^{\mathrm{T}}_{q}, \mathbf{0}^{\mathrm{T}}_{p-h(q+1)-(r-h)q})^{\mathrm{T}} & \text{if } h+1 \leqslant r \leqslant m, \end{cases}$$

and write $U := (u_1, \ldots, u_m) \in \mathbb{R}^{p \times m}$. By construction, $U^{\mathrm{T}}U = I_m$, so there exists $\tilde{U} \in \mathbb{O}_p$ whose first m columns are U. Moreover, for $j \in [k]$, we have

$$\frac{4m}{5k} \leqslant \frac{m}{k+m} \leqslant \frac{1}{q+1} \leqslant \|U^{(j,\cdot)}\|_2^2 \leqslant \frac{1}{q} \leqslant \frac{m}{k-m} \leqslant \frac{4m}{3k}. \tag{23}$$

Now, fix some $\epsilon \in (0, \sqrt{\{m/(16k)\}}]$ to be specified later. For any $J \in \mathbb{O}_{p-m,m,k-m}$, define

$$V_J := \tilde{U} \begin{pmatrix} \sqrt{(1-\epsilon^2)} I_m \\ \epsilon J \end{pmatrix} = U + \tilde{U} \begin{pmatrix} \{\sqrt{(1-\epsilon^2)} - 1\} I_m \\ \epsilon J \end{pmatrix} =: U + \tilde{U}\Delta_J.$$

For any matrix $M \in \mathbb{R}^{p \times m}$, we define its *two-to-infinity norm* as

$$\|M\|_{2\to\infty} := \sup_{v \in \mathcal{S}^{m-1}} \|Mv\|_\infty = \max_{j \in [p]} \|M^{(j,\cdot)}\|_2.$$

Then, for $J \in \mathbb{O}_{p-m,m,k-m}$, we have

$$\|V_J - U\|_{2\to\infty} \leqslant \|\tilde{U}\|_{2\to\infty} \|\Delta_J\|_{\mathrm{op}} = \|\Delta_J^{\mathrm{T}}\Delta_J\|_{\mathrm{op}}^{1/2} \leqslant \sqrt{2}\epsilon. \tag{24}$$

Combining inequalities (23) and (24), and, since $\epsilon \leqslant \sqrt{\{m/(16k)\}}$, we have $\|V_J^{(j,\cdot)}\|_2 \in [0.54(m/k)^{1/2}, 1.51(m/k)^{1/2}]$ for all $j \in [k]$, which implies that $V_J \in \mathbb{O}_{p,m,k}(3)$.

Using the definition of V_J and the triangle inequality, we have that, for any $J, J' \in \mathbb{O}_{p-m,m,k-m}$,

$$\|V_J^{\mathrm{T}} V_{J'}\|_{\mathrm{F}} = \|(1-\epsilon^2) I_m + \epsilon^2 J^{\mathrm{T}} J'\|_{\mathrm{F}} \geqslant (1-\epsilon^2)\|I_m\|_{\mathrm{F}} - \epsilon^2\sqrt{m}\|J^{\mathrm{T}}J'\|_{\mathrm{op}} = (1-2\epsilon^2)\sqrt{m}. \tag{25}$$

Writing $D_{\mathrm{KL}}(P\|Q)$ for the Kullback–Leibler divergence from a distribution P to a distribution Q and $\Sigma_J := I_p + \theta V_J V_J^{\mathrm{T}}$, we have for any $J, J' \in \mathbb{O}_{p-m,m,k-m}$ that

$$\begin{aligned} D_{\mathrm{KL}}\{N_p(0,\Sigma_J)\|N_p(0,\Sigma_{J'})\} &= \frac{1}{2}\operatorname{tr}(\Sigma_{J'}^{-1}\Sigma_J - I_p) = \frac{\theta}{2}\operatorname{tr}\{(I_p + \theta V_{J'}V_{J'}^{\mathrm{T}})^{-1}(V_J V_J^{\mathrm{T}} - V_{J'}V_{J'}^{\mathrm{T}})\} \\ &= \frac{\theta}{2}\operatorname{tr}\left\{\left(I_p - \frac{\theta}{1+\theta}V_{J'}V_{J'}^{\mathrm{T}}\right)(V_J V_J^{\mathrm{T}} - V_{J'}V_{J'}^{\mathrm{T}})\right\} = \frac{\theta^2}{2(1+\theta)}\{m - \operatorname{tr}(V_{J'}V_{J'}^{\mathrm{T}}V_J V_J^{\mathrm{T}})\} \\ &= \frac{\theta^2}{2(1+\theta)}(m - \|V_J^{\mathrm{T}}V_{J'}\|_{\mathrm{F}}^2) \leqslant \frac{2m\epsilon^2\theta^2}{1+\theta}, \end{aligned} \tag{26}$$

where we used inequality (25) in the final inequality. In contrast, we also have

$$L(V_J, V_{J'}) = \frac{1}{\sqrt{2}}\|V_J V_J^{\mathrm{T}} - V_{J'}V_{J'}^{\mathrm{T}}\|_{\mathrm{F}} = \{\epsilon^4 L^2(J,J') + \epsilon^2(1-\epsilon^2)\|J - J'\|_{\mathrm{F}}^2\}^{1/2} \geqslant \epsilon L(J,J'), \tag{27}$$

where we used Vu and Lei (2013), proposition 2.2, in the last inequality. Thus, if we can find some finite subset $\mathcal{J} \subseteq \mathbb{O}_{p-m,m,k-m}$ such that $3 \leqslant |\mathcal{J}| \leqslant \exp(nm^2\theta^2/k)$ and $\min_{J,J' \in \mathcal{J}: J \neq J'} L(J,J') \geqslant cm^{1/2}$ for some universal constant $c > 0$, then by expressions (26) and (27) and Fano's lemma (see, for example, Yu (1997), lemma 3), we have

$$\begin{aligned} \inf_{\tilde{V}} \sup_{V \in \mathbb{O}_{p,m,k}(3)} \mathbb{E}_{P_{V,\theta}}\{L(\tilde{V}, V)\} &\geqslant \inf_{\tilde{V}} \max_{J \in \mathcal{J}} \mathbb{E}_{P_{V_J,\theta}}\{L(\tilde{V}, V_J)\} \\ &\geqslant \frac{cm^{1/2}\epsilon}{2}\left\{1 - \frac{2nm\epsilon^2\theta^2/(1+\theta) + \log(2)}{\log|\mathcal{J}|}\right\} \geqslant \frac{cm^{1/2}\epsilon}{2}\left(\frac{1}{3} - \frac{2nm\epsilon^2\theta^2}{\log|\mathcal{J}|}\right), \end{aligned}$$

where we used the fact that $|\mathcal{J}| \geqslant 3$ in the final inequality. Choosing

$$\epsilon = \sqrt{\left(\frac{\log|\mathcal{J}|}{16nm\theta^2}\right)}$$

(noting that the condition $\log|\mathcal{J}| \leqslant nm^2\theta^2/k$ ensures that $\epsilon \leqslant \sqrt{\{m/(16k)\}}$), we obtain

$$\inf_{\tilde{V}} \sup_{V \in \mathbb{O}_{p,m,k}(3)} \mathbb{E}_{P_{V,\theta}}\{L(\tilde{V}, V)\} \geqslant \frac{cm^{1/2}\epsilon}{10} \gtrsim \sqrt{\left(\frac{\log|\mathcal{J}|}{n\theta^2}\right)}. \tag{28}$$

It remains to construct a suitable $\mathcal{J}$. By Szarek (1982) (see also Pajor (1998), proposition 8), there is a finite subset $\tilde{\mathcal{J}} \subseteq \mathbb{O}_{k-m,m}$ such that $|\tilde{\mathcal{J}}| = \lfloor \exp\{m(k-2m)\} \rfloor$ and $L(\tilde{J}, \tilde{J}') \geqslant cm^{1/2}$ for all distinct $\tilde{J}, \tilde{J}' \in \tilde{\mathcal{J}}$. Define $\mathcal{J} := \{(\tilde{J}^{\mathrm{T}}, \mathbf{0}^{\mathrm{T}}_{(p-k)\times m})^{\mathrm{T}} : \tilde{J} \in \tilde{\mathcal{J}}\}$. We have $\min_{J,J' \in \mathcal{J}: J \neq J'} L(J, J') = \min_{\tilde{J},\tilde{J}' \in \tilde{\mathcal{J}}: \tilde{J} \neq \tilde{J}'} L(\tilde{J}, \tilde{J}') \geqslant cm^{1/2}$ and $|\mathcal{J}| = |\tilde{\mathcal{J}}|$. Since $k \geqslant 4m$ and $nm\theta^2 \geqslant k^2$, we have $3 \leqslant |\mathcal{J}| \leqslant \exp(nm^2\theta^2/k)$ as desired. Hence, by expression (28),

$$\inf_{\tilde{V}} \sup_{V \in \mathbb{O}_{p,m,k}(3)} \mathbb{E}_{P_{V,\theta}}\{L(\tilde{V}, V)\} \gtrsim \sqrt{\left(\frac{mk}{n\theta^2}\right)}. \tag{29}$$

Alternatively, we can also construct $\mathcal{J}$ as follows. Recall the definition of $\binom{[p-m]}{k}$ from the proof of theorem 1. For any $S \in \binom{[p-m]}{k}$, define $J_S \in \mathbb{R}^{(p-m)\times m}$ such that $J_S^{(S,\cdot)} = U^{([k],\cdot)}$ and $J_S^{(S^c,\cdot)} = \mathbf{0}$. By the Gilbert–Varshamov lemma (see, for example, Massart (2007), lemma 4.10) and, since $p \geqslant 5k$, there exists $\mathcal{S} \subseteq \binom{[p-m]}{k}$ such that

$$|\mathcal{S}| = \left\lfloor \exp\left\{\frac{1}{15}k\log\left(\frac{p-m}{k}\right)\right\}\right\rfloor$$

and, for any distinct $S, S' \in \mathcal{S}$, $|S \cap S'| \leqslant k/2$. Let $\mathcal{J} := \{J_S : S \in \mathcal{S}\}$. Then $|\mathcal{J}| = |\mathcal{S}|$ and

$$\min_{J,J' \in \mathcal{J}: J \neq J'} L(J, J') = \min_{J,J' \in \mathcal{J}: J \neq J'} (m - \|J^{\mathrm{T}}J'\|_{\mathrm{F}}^2)^{1/2} \geqslant \left(m - \frac{k}{2q}\right)^{1/2} \geqslant \sqrt{\left(\frac{m}{3}\right)},$$

where the final inequality uses inequality (23). Since $k\log\{(p-m)/k\} \geqslant 17$ and $nm^2\theta^2 \geqslant k^2\log(p/k)$, we have $3 \leqslant |\mathcal{J}| \leqslant \exp(nm^2\theta^2/k)$ as desired. Hence, by expression (28),

$$\inf_{\tilde{V}} \sup_{V \in \mathbb{O}_{p,m,k}(3)} \mathbb{E}_{P_{V,\theta}}\{L(\tilde{V}, V)\} \gtrsim \sqrt{\left\{\frac{k\log(p/k)}{n\theta^2}\right\}}. \tag{30}$$

We complete the proof by combining expressions (29) and (30).

A.4. Proof of corollary 1

The proof of theorem 1 remains valid for the setting of corollary 1. Fix a specific $a \in [A]$. Since $V_m \in \mathbb{O}_{p,m,k}(1)$ and $\theta_1 = \ldots = \theta_m$, we have by expression (13) that on Ω_{RCC}, for any $S, S' \in \binom{[p]}{d}$, if $|S \cap S_0| < |S' \cap S_0|$, then $\sum_{r=1}^m \lambda_r(\hat{\Sigma}^{(S,S)}) < \sum_{r=1}^m \lambda_r(\hat{\Sigma}^{(S',S')})$. Thus, in particular, $|S_{a,b^*(a)} \cap S_0| = \max_{b \in [B]} |S_{a,b} \cap S_0|$ on Ω_{RCC}.

Observe that $|S_{a,b} \cap S_0| \sim^{\mathrm{IID}} \mathrm{HyperGeom}(d, k, p)$. Let $M := \max_{b \in [B]} |S_{a,b} \cap S_0|$ and $R := |\{b \in [B] : |S_{a,b} \cap S_0| = M\}|$. Conditionally on $R = 1$ and X such that Ω_{RCC} holds, each signal co-ordinate $j \in S_0$ has the same probability of being included in $S_{a,b^*(a)}$, which is the unique subset of maximal intersection with S_0. Thus, we have on Ω_{RCC} that

$$\mathbb{P}(\{j \in S_{a,b^*(a)}\} \cap \{R = 1\} | X) = \mathbb{P}(\{j' \in S_{a,b^*(a)}\} \cap \{R = 1\} | X) \tag{31}$$

for $j, j' \in S_0$. Recall the definition of q_j from the proof of theorem 1. By equation (31), for any $j \in S_0$, we have on Ω_{RCC} that

$$\begin{aligned} q_j &\geqslant \mathbb{P}(\{j \in S_{a,b^*(a)}\} \cap \{R = 1\} | X) = \frac{1}{k} \sum_{\tilde{j} \in S_0} \mathbb{E}(\mathbb{1}_{\{\tilde{j} \in S_{a,b^*(a)}\}} \mathbb{1}_{\{R=1\}} | X) \\ &= \frac{1}{k} \mathbb{E}(|S_{a,b^*(a)} \cap S_0| \mathbb{1}_{\{R=1\}} | X) \geqslant \frac{t}{k} \mathbb{P}(M \geqslant t, R = 1) \geqslant \frac{t}{4k}, \end{aligned} \tag{32}$$

where the penultimate inequality uses Markov's inequality and the fact that the pair (M, R) is independent of X, and the final bound follows from lemma 4 below. Now, using expression (32) in place of expression (16), we find that $\mathbb{E}(\hat{w}_a^{(j)} - \hat{w}_a^{(j')} | X) \geqslant tm\theta_m/(8k^2)$ instead of inequality (22). Thus, $\mathbb{P}(\Omega^{\mathrm{c}} | X) \leqslant p\exp\{-At^2/(800k^2)\}$. The desired result is then concluded in a similar fashion to that in theorem 1.

A.5. Additional lemmas

Lemma 2. Suppose that Σ and $\hat{\Sigma}$ are symmetric $d \times d$ matrices. For $r \in [d]$, let $\lambda_r := \lambda_r(\Sigma)$ and $v_r := v_r(\Sigma)$ be the eigenvalues and corresponding eigenvectors of Σ, and let $\hat{\lambda}_r := \lambda_r(\hat{\Sigma})$ and $\hat{v}_r := v_r(\hat{\Sigma})$ be the eigenvalues and corresponding eigenvectors of $\hat{\Sigma}$. Also, for $r \in [d]$, define $V_r := (v_1, \ldots, v_r)$, $\hat{V}_r := (\hat{v}_1, \ldots, \hat{v}_r)$, $\Theta_r := \operatorname{diag}(\lambda_1 - \lambda_{r+1}, \ldots, \lambda_r - \lambda_{r+1})$ and $\hat{\Theta}_r := \operatorname{diag}(\hat{\lambda}_1 - \hat{\lambda}_{r+1}, \ldots, \hat{\lambda}_r - \hat{\lambda}_{r+1})$ (with the convention that $\lambda_{d+1} = \hat{\lambda}_{d+1} := 0$). Then, for any $m \in [d]$,

$$\|\hat{V}_m \hat{\Theta}_m \hat{V}_m^{\mathrm{T}} - V_m \Theta_m V_m^{\mathrm{T}}\|_{\mathrm{op}} \leqslant 4m \|\hat{\Sigma} - \Sigma\|_{\mathrm{op}}.$$

Proof. By the Davis–Kahan theorem (see, for example Stewart and Sun (1990), theorem V.3.6) and Weyl's inequality, we have for any $r \in [d]$ that

$$(\lambda_r - \lambda_{r+1} - \|\hat{\Sigma} - \Sigma\|_{\mathrm{op}}) \|\sin\{\Theta(\hat{V}_r, V_r)\}\|_{\mathrm{op}} \leqslant \|\hat{\Sigma} - \Sigma\|_{\mathrm{op}}.$$

After rearranging, while noting that $\|\sin\{\Theta(\hat{V}_r, V_r)\}\|_{\mathrm{op}} \leqslant 1$, we obtain that

$$(\lambda_r - \lambda_{r+1}) \|\sin\{\Theta(\hat{V}_r, V_r)\}\|_{\mathrm{op}} \leqslant 2\|\hat{\Sigma} - \Sigma\|_{\mathrm{op}}. \tag{33}$$

Now, we can rewrite

$$V_m \Theta_m V_m^{\mathrm{T}} = \sum_{r=1}^{m} (\lambda_r - \lambda_{m+1}) v_r v_r^{\mathrm{T}} = \sum_{r=1}^{m} (\lambda_r - \lambda_{r+1}) V_r V_r^{\mathrm{T}},$$

and, similarly, $\hat{V}_m \hat{\Theta}_m \hat{V}_m^{\mathrm{T}} = \Sigma_{r=1}^{m} (\hat{\lambda}_r - \hat{\lambda}_{r+1}) \hat{V}_r \hat{V}_r^{\mathrm{T}}$. Thus,

$$\begin{aligned}
\|\hat{V}_m \hat{\Theta}_m \hat{V}_m^{\mathrm{T}} - V_m \Theta_m V_m^{\mathrm{T}}\|_{\mathrm{op}} &\leqslant \sum_{r=1}^{m} \|(\hat{\lambda}_r - \hat{\lambda}_{r+1}) \hat{V}_r \hat{V}_r^{\mathrm{T}} - (\lambda_r - \lambda_{r+1}) V_r V_r^{\mathrm{T}}\|_{\mathrm{op}} \\
&\leqslant \sum_{r=1}^{m} \{|\hat{\lambda}_r - \lambda_r - (\hat{\lambda}_{r+1} - \lambda_{r+1})| \|\hat{V}_r \hat{V}_r^{\mathrm{T}}\|_{\mathrm{op}} + (\lambda_r - \lambda_{r+1}) \|\hat{V}_r \hat{V}_r^{\mathrm{T}} - V_r V_r^{\mathrm{T}}\|_{\mathrm{op}}\} \\
&\leqslant \sum_{r=1}^{m} [|\hat{\lambda}_r - \lambda_r| + |\hat{\lambda}_{r+1} - \lambda_{r+1}| + (\lambda_r - \lambda_{r+1}) \|\sin\{\Theta(\hat{V}_r, V_r)\}\|_{\mathrm{op}}] \leqslant 4m \|\hat{\Sigma} - \Sigma\|_{\mathrm{op}},
\end{aligned}$$

where we used lemma 3 below in the penultimate inequality, and Weyl's inequality and inequality (33) in the final inequality.

Lemma 3. For $U, V \in \mathbb{O}_{d,r}$ with $r \leqslant d$, let $\lambda_1, \ldots, \lambda_s$ (where $s \leqslant r$) denote the non-zero eigenvalues of $\sin\{\Theta(U, V)\}$. Then the non-zero eigenvalues of $UU^{\mathrm{T}} - VV^{\mathrm{T}}$ are given by $\lambda_1, \ldots, \lambda_s, -\lambda_1, \ldots, -\lambda_s$. In particular, $\|UU^{\mathrm{T}} - VV^{\mathrm{T}}\|_{\mathrm{op}} = \|\sin\{\Theta(U, V)\}\|_{\mathrm{op}}$ and $\|UU^{\mathrm{T}} - VV^{\mathrm{T}}\|_{\mathrm{F}}^2 = 2\|\sin\{\Theta(U, V)\}\|_{\mathrm{F}}^2$.

Proof. We need to prove only the first statement. First assume that $2r \leqslant d$. By the first part of Stewart and Sun (1990), theorem I.5.2, there are $Q \in \mathbb{O}_d$ and $G, H \in \mathbb{O}_r$ such that

$$U = Q \begin{pmatrix} I_r \\ \mathbf{0}_{r \times r} \\ \mathbf{0}_{(d-2r) \times r} \end{pmatrix} G,$$

$$V = Q \begin{pmatrix} \Gamma \\ \Sigma \\ \mathbf{0}_{(d-2r) \times r} \end{pmatrix} H,$$

where $\Gamma = \operatorname{diag}(\gamma_1, \ldots, \gamma_r)$, $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_r)$, $0 \leqslant \gamma_1 \ldots \leqslant \gamma_r$, $\sigma_1 \geqslant \ldots \geqslant \sigma_r \geqslant 0$ and $\Gamma^2 + \Sigma^2 = I_r$. Hence, $U^{\mathrm{T}} V = G^{\mathrm{T}} \Gamma H$ has singular values $\gamma_1, \ldots, \gamma_r$ and $\sin\{\Theta(U, V)\} = \operatorname{diag}\{\sqrt{(1 - \gamma_1^2)}, \ldots, \sqrt{(1 - \gamma_r^2)}\}$ has eigenvalues $\sigma_1, \ldots, \sigma_r$. However, we compute that

$$Q^{\mathrm{T}}(UU^{\mathrm{T}} - VV^{\mathrm{T}})Q = \begin{pmatrix} \Sigma^2 & -\Gamma\Sigma & \mathbf{0} \\ -\Sigma\Gamma & -\Sigma^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix},$$

which after permuting rows and columns is a block diagonal matrix with diagonal blocks

$$\begin{pmatrix} \sigma_j^2 & -\sigma_j \gamma_j \\ -\sigma_j \gamma_j & -\sigma_j^2 \end{pmatrix}$$

for $j \in [r]$. Each of these diagonal blocks has eigenvalues $\pm\sigma_j$. Thus, the eigenvalues of $UU^{\mathrm{T}} - VV^{\mathrm{T}}$ are $\pm\sigma_1, \ldots, \pm\sigma_r, 0, \ldots, 0$.

Now, assume that $2r > d$ instead. Then by the second part of Stewart and Sun (1990), theorem I.5.2, there are $Q \in \mathbb{O}_d$ and $G, H \in \mathbb{O}_r$ such that

$$U = Q \begin{pmatrix} I_{d-r} & \mathbf{0}_{(d-r)\times(2r-d)} \\ \mathbf{0}_{(d-r)\times(d-r)} & \mathbf{0}_{(d-r)\times(2r-d)} \\ \mathbf{0}_{(2r-d)\times(d-r)} & I_{2r-d} \end{pmatrix} G,$$

$$V = Q \begin{pmatrix} \Gamma & \mathbf{0}_{(d-r)\times(2r-d)} \\ \Sigma & \mathbf{0}_{(d-r)\times(2r-d)} \\ \mathbf{0}_{(2r-d)\times(d-r)} & I_{2r-d} \end{pmatrix} H,$$

where $\Gamma = \mathrm{diag}(\gamma_1, \ldots, \gamma_{d-r})$, $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_{d-r})$ and $\Gamma^2 + \Sigma^2 = I_{d-r}$. We may assume that $\sigma_1 \geqslant \ldots \geqslant \sigma_{d-r}$. Hence,

$$U^{\mathrm{T}}V = G^{\mathrm{T}} \begin{pmatrix} \Gamma & \mathbf{0} \\ \mathbf{0} & I_{2r-d} \end{pmatrix} H$$

has singular values $\gamma_1, \ldots, \gamma_{d-r}, 1, \ldots, 1$ and $\sin\{\Theta(U, V)\}$ has eigenvalues $\sigma_1, \ldots, \sigma_{d-r}, 0, \ldots, 0$. However, we again have

$$Q^{\mathrm{T}}(UU^{\mathrm{T}} - VV^{\mathrm{T}})Q = \begin{pmatrix} \Sigma^2 & -\Gamma\Sigma & \mathbf{0} \\ -\Sigma\Gamma & -\Sigma^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix}.$$

Thus, $UU^{\mathrm{T}} - VV^{\mathrm{T}}$ has eigenvalues $\pm\sigma_1, \ldots, \pm\sigma_{d-r}, 0, \ldots, 0$ as desired.

Lemma 4. Let $Y_1, \ldots, Y_B$ be independent and identically distributed on $\mathbb{N} \cup \{0\}$ with distribution function F. Define $M := \max_{b\in[B]} Y_b$ and $R := |\{b : Y_b = M\}|$. Then for $B = \lceil 2^{-1}\{1 - F(t-1)\}^{-1} \rceil$, we have $\mathbb{P}(M \geqslant t, R = 1) \geqslant \frac{1}{4}$.

Proof. For $m \in \mathbb{N} \cup \{0\}$, define $p_m := \mathbb{P}(Y_1 = m)$ and $q_m := \mathbb{P}(Y_1 \geqslant m)$. By the definition of B, we have $(B-1)q_t \leqslant \frac{1}{2} \leqslant Bq_t$. Also, observe that

$$\mathbb{P}(M = m, R = 1) = B\,\mathbb{P}(X_1 = m) \prod_{b=2}^{B} \mathbb{P}(X_b < m) = Bp_m(1 - q_m)^{B-1}.$$

Therefore,

$$\mathbb{P}(M \geqslant t,\ R = 1) = \sum_{m=t}^{\infty} \mathbb{P}(M = m, R = 1) = \sum_{m=t}^{\infty} Bp_m(1 - q_m)^{B-1} \geqslant Bq_t\{1 - (B-1)q_t\} \geqslant \tfrac{1}{4}$$

as desired.

References

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D. and Levine, A. J. (1999) Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natn. Acad. Sci. USA*, **96**, 6745–6750.

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

d'Aspremont, A., Bach, F. and El Ghaoui, L. (2008) Optimal solutions for sparse principal component analysis. *J. Mach. Learn. Res.*, **9**, 1269–1294.

d'Aspremont, A., El Ghaoui, L., Jordan, M. I. and Lanckriet, G. R. G. (2007) A direct formulation for sparse PCA using semidefinite programming. *Adv. Neurl Inform. Process. Syst.*, **16**, 41–48.

Cai, T. T., Ma, Z. and Wu, Y. (2013) Sparse PCA: optimal rates and adaptive estimation. *Ann. Statist.*, **41**, 3074–3110.

Cannings, T. I. and Samworth, R. J. (2017) Random-projection ensemble classification (with discussion). *J. R. Statist. Soc.* B, **79**, 959–1035.

Fowler, J. E. (2009) Compressive-projection principal component analysis. *IEEE Trans. Im. Process.*, **18**, 2230–2242.

Gataric, M., Wang, T. and Samworth, R. J. (2018) SPCAvRP: sparse principal component analysis via random projections. *R Package Version 0.4.* Statistical Laboratory, University of Cambridge, Cambridge. (Available from `https://cran.r-project.org/web/packages/SPCAvRP/index.html`.)
Johnstone, I. M. and Lu, A. Y. (2009) On consistency and sparsity for principal components analysis in high dimensions. *J. Am. Statist. Ass.*, **104**, 682–693.
Jolliffe, I. T., Trendafilov, N. T. and Uddin, M. (2003) A modified principal component technique based on the LASSO. *J. Computnl Graph. Statist.*, **12**, 531–547.
Ma, Z. (2013) Sparse principal component analysis and iterative thresholding. *Ann. Statist.*, **41**, 772–801.
Mackey, L. W. (2009) Deflation methods for sparse PCA. *Adv. Neurl Inform. Process. Syst.*, **21**, 1017–1024.
Marzetta, T. L., Tucci, G. H. and Simon, S. H. (2011) A random matrix-theoretic approach to handling singular covariance estimates. *IEEE Trans. Inform. Theory*, **57**, 6256–6271.
Massart, P. (2007) *Concentration Inequalities and Model Selection*. Berlin: Springer.
Moghaddam, B., Weiss, Y. and Avidan, S. (2006) Spectral bounds for sparse PCA: exact and greedy algorithms. *Adv. Neurl Inform. Process. Syst.*, **18**, 915–922.
Pajor, A. (1998) Metric entropy of the Grassmanian manifold. In *Convex Geometric Analysis*. Bishkek: Mountain Societies Research Institute Publications.
Paul, D. (2007) Asymptotics of sample eigenstructure for a large dimensional spiked covariance model. *Statist. Sin.*, **17**, 1617–1642.
Paul, D. and Johnstone, I. M. (2012) Augmented sparse principal component analysis for high dimensional data. *Preprint arXiv:1202.1242v1*. University of California at Davis, Davis.
Pourkamali-Anaraki, F. and Hughes, S. (2014) Memory and computation efficient PCA via very sparse random projections. In *Proc. Int. Conf. Machine Learning 31*, pp. 1341–1349.
Qi, H. and Hughes, S. (2012) Invariance of principal components under low-dimensional random projection of the data. In *Proc. Int. Conf. Image Processing 19*, pp. 937–940.
Ramey, J. A. (2016) Collection of data sets for classification. *R Package*. Novi Laboratories, Austin. (Available from `https://github.com/ramhiser/datamicroarray`.)
Shen, H. and Huang, J. Z. (2008) Sparse principal component analysis via regularized low rank matrix approximation. *J. Multiv. Anal.*, **99**, 1015–1034.
Stewart, G. W. and Sun, J.-G. (1990) *Matrix Perturbation Theory*. San Diego: Academic Press.
Szarek, S. (1982) Nets of Grassmann manifold and orthogonal groups. In *Proc. Research Wrkshp Banach Space Theory, Iowa City, 1981* (ed. B.-L. Lin), pp. 169–185. Iowa City: University of Iowa.
Tillman, A. N. and Pfetsch, M. E. (2014) The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing. *IEEE Trans. Inform. Theory*, **60**, 1248–1259.
Vu, V. Q., Cho, J., Lei, J. and Rohe, K. (2013) Fantope projection and selection: a near-optimal convex relaxation of sparse PCA. *Adv. Neurl Inform. Process. Syst.*, **26**, 2670–2678.
Vu, V. Q. and Lei, J. (2013) Minimax sparse principal subspace estimation in high dimensions. *Ann. Statist.*, **41**, 2905–2947.
Wang, T., Berthet, Q. and Samworth, R. J. (2016) Statistical and computational trade-offs in estimation of sparse principal components. *Ann. Statist.*, **44**, 1896–1930.
Wang, Z., Lu, H. and Liu, H. (2014) Tighten after relax: minimax-optimal sparse PCA in polynomial time. *Adv. Neurl Inform. Process. Syst.*, **27**, 3383–3391.
Welch, B. L. (1947) The generalization of "Student's" problem when several different population variances are involved. *Biometrika*, **34**, 28–35.
Weyl, H. (1912) Das asymptotische Verteilungsgesetz der Eigenwerte linearer partieller Differentialgleichungen (mit einer Anwendung auf der Theorie der Hohlraumstrahlung). *Math. Ann.*, **71**, 441–479.
Witten, D. M., Tibshirani, R. and Hastie, T. (2009) A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, **10**, 515–534.
Yu, B. (1997) Assouad, Fano and Le Cam. In *Festschrift for Lucien Le Cam: Research Papers in Probability and Statistics* (eds D. Pollard, E. Torgersen and G. L. Yang), pp. 423–435. New York: Springer.
Yu, Y., Wang, T. and Samworth, R. J. (2015) A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, **102**, 315–323.
Zou, H., Hastie, T. and Tibshirani, R. (2006) Sparse principal components analysis. *J. Computnl Graph. Statist.*, **15**, 265–286.