Decomposing the Apoptosis Pathway Into Biologically Interpretable Principal Components

Cancer Informatics
Volume 17: 1–13
© The Author(s) 2018
DOI: 10.1177/1176935118771082
SAGE

Min Wang[1], Steven M Kornblau[2] and Kevin R Coombes[3]

[1]Mathematical Biosciences Institute, The Ohio State University, Columbus, OH, USA.
[2]Department of Leukemia, The University of Texas MD Anderson Cancer Center, Houston, TX, USA.
[3]Department of Biomedical Informatics, The Ohio State University, Columbus, OH, USA.

ABSTRACT: Principal component analysis (PCA) is one of the most common techniques in the analysis of biological data sets, but applying PCA raises 2 challenges. First, one must determine the number of significant principal components (PCs). Second, because each PC is a linear combination of genes, it rarely has a biological interpretation. Existing methods to determine the number of PCs are either subjective or computationally extensive. We review several methods and describe a new R package, PCDimension, that implements additional methods, the most important being an algorithm that extends and automates a graphical Bayesian method. Using simulations, we compared the methods. Our newly automated procedure is competitive with the best methods when considering both accuracy and speed and is the most accurate when the number of objects is small compared with the number of attributes. We applied the method to a proteomics data set from patients with acute myeloid leukemia. Proteins in the apoptosis pathway could be explained using 6 PCs. By clustering the proteins in PC space, we were able to replace the PCs by 6 "biological components," 3 of which could be immediately interpreted from the current literature. We expect this approach combining PCA with clustering to be widely applicable.

KEYWORDS: Dimension reduction, Bayes rule, Auer-Gervini, broken stick, randomization-based procedure

RECEIVED: October 2, 2017. **ACCEPTED:** March 11, 2018.

TYPE: Original Research

FUNDING: The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This project is supported by NIH/NCI grants P30 CA016058 and R01 CA182905.

DECLARATION OF CONFLICTING INTERESTS: The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

CORRESPONDING AUTHOR: Kevin R Coombes, Department of Biomedical Informatics, The Ohio State University, Columbus, OH 43210, USA.
Email: coombes.3@osu.edu

Introduction

Since the earliest days of gene expression microarrays, 2-way clustered heatmaps have been a standard feature of papers studying genome-wide biological data sets.[1,2] Such heatmaps remain ubiquitous, despite numerous difficulties in interpretation, in reproducibility, and in assigning statistical significance. Good clustering of the genes in such data sets is critical for understanding the biology. Because many biologists are more interested in signaling pathways than in individual genes, they want to find a source of consistent, robust, and interpretable blocks of genes that drive distinct functional characteristics of the pathways. These blocks of genes form clusters that are relevant to comprehensive understanding of critical biological processes. For example, apoptosis is an important biological process, which is characterized by distinct morphological states and energy-dependent biochemical mechanisms.[3] Understanding how proteins cluster in the apoptotic pathway will help elucidate its underlying molecular mechanisms. Now, clustering can be thought of as a form of dimension reduction, and a natural question is the "true dimension" of the data. Various techniques have been developed to determine the dimensionality, the most common being principal component analysis (PCA). For our purposes, an important problem in PCA is to determine the number of statistically significant components.

Numerous methods have already been developed to estimate the number of significant components. There are 4 types of approaches: (1) ad hoc subjective and graphical rules; (2) methods based on distributional assumptions; (3) computationally extensive procedures relying on Monte Carlo, permutation, cross-validation, bootstrap, or jackknife[4,5]; and (4) Bayesian methods based on the probabilistic formulation of Tipping and Bishop[6] with marginalization estimated through Laplace approximation and its variants.[7–9] The scree plot method, which consists of plotting a curve of the eigenvalues of the sample covariance matrix versus their rank and looking for an "elbow" in the curve, is the most famous graphical approach.[10] However, this method relies on the user's subjective experience to find any possible "elbow." Even so, other methods are not always superior to the simple scree plot. Legendre and Legendre[11] used the "broken stick" distribution to compare the extra information in a model to one with fewer parameters. Ferre[12] conducted an empirical study of many methods to select the number of PCs, using data simulated from known parameters. He concluded that there is no "ideal" solution to the problem of dimensionality in PCA. He also concluded that Bartlett's tests[13] are an improvement because they are less subjective but may have a tendency to overestimate the true number of components. Peres-Neto et al[14] conducted an extensive simulation study to evaluate a wider variety of methods. They concluded that several methods, especially those based on randomization and permutation proposed by ter Braak,[15,16] outperform the others and should be applied to study general data sets. More recently, Josse and Husson[5] showed that the generalized cross-validation method performs well. Sobczyk et al[9] proposed a Bayesian approach called PEnalized SEmi-integrated Likelihood (PESEL); by comparing it with other state-of-the-art methods, they found

that PESEL-based criteria are more robust against deviations from the assumptions of a probabilistic model than the other methods.

In 2008, Auer and Gervini[17] addressed the problem of selecting principal components in the context of Bayesian model selection. Although their method has strong theoretical foundations and appears to work well in practice, it still depends on the subjective evaluation of a graphical display of how the maximum posterior estimate of the number of components depends on a parameter describing the choice of the prior distribution. Moreover, its performance has only been compared with the scree plot and broken stick methods and not to more sophisticated methods that have performed well in comparative studies.

In this article, we consider several algorithms to extend and automate the Auer-Gervini method by providing objective rules to select the number of significant principal components. Using an extensive set of simulations, we compare these algorithms with the broken stick model, Bartlett's test, ter Braak's randomization methods, generalized cross-validation, and Bayesian probabilistic PCA. The methods chosen for comparison were the "winners" from the previous comparative studies.[7,9,12,14] Our extensions to the Auer-Gervini method are implemented in the PCDimension R package. Because the most promising versions of the randomization algorithms appear not to be readily available, we have also implemented them in the PCDimension package. For convenience, the package also implements the broken stick method. For Bartlett's test, we rely on an existing implementation in the nFactors package.[18] For generalized cross-validation, we use the implementation in the FactoMineR package.[19] The implementation of Bayesian approximation methods including Minka's approach and PESEL from probabilistic PCA is available in the pesel package and code on Github.[9]

This article is organized as follows. In section "Methods," we review the theoretical framework of different types of methods. In section "Simulation Study" of the "Results" section, we perform simulation studies to test the performance of the proposed algorithms. In the "Decomposing the Apoptosis Pathway in AML" section, we apply the methods to a study of apoptosis in acute myeloid leukemia (AML) using reverse phase protein arrays (RPPAs). Finally, we conclude the article and make several remarks in section "Conclusions." A simple example to illustrate the implementation of different methods in the PCDimension package is also provided in the supplementary material.

Methods

Let X denote an $n \times m$ data matrix, where each row represents an object to be analyzed, and each column represents a measured attribute. In PCA, each principal component is a linear combination of the attributes. Much effort has been expended on investigating many objects using relatively few attributes, but the opposite scenario—few objects and many attributes—is usually ignored. One practical application would be clustering a few genes or proteins from a single biological pathway using their expression values for many patients. Therefore, we are primarily interested in biological applications where $n \ll m$. In this section, we briefly review the methods used to estimate the number of statistically significant PCs.

Bartlett's test

Bartlett[13] proposed a statistical method to conduct a hypothesis test on the significance of the principal components based on the eigenvalues of Σ, the correlation matrix of the objects. This test is designated to check whether the remaining eigenvalues of the correlation structure are equal after removing the well-determined (highly significant) components. Let the eigenvalues of Σ be $\lambda_1, \ldots, \lambda_n$ with $\lambda_1 \geq \cdots \geq \lambda_n \geq 0$. The procedure, for various values of k, starting at $n-2$ in decreasing order, is to test the null hypothesis H_0^k that the "$(n-k)$ smallest eigenvalues of the correlation matrix are equal" against the alternative hypothesis H_A^k that "at least 2 of the $(n-k)$ eigenvalues are different." The test statistic is as follows:

$$\chi^2 = -\left\{m - \frac{1}{6}(2n+5) - \frac{2}{3}k\right\}\log(R_{n-k}) \tag{1}$$

where

$$R_{n-k} = \frac{\det(\Sigma)}{\lambda_1 \cdots \lambda_k \left[\frac{\lambda_{k+1} + \cdots + \lambda_n}{n-k}\right]^{n-k}}$$

Under the null hypothesis, the test statistic follows a χ^2 distribution with $(1/2)(n-k)(n-k-1)$ degrees of freedom. The optimal number of principal components is the smallest k where H_0^k is accepted. Both Lawley[20] and Anderson[21] made some modifications to the multiplicative factor $\{m-(1/6)(2n+5)-(2/3)k\}$ for equation (1); these are viewed as improved variants of Bartlett's test.

Broken stick model

Under the assumption that the total variance of the multivariate data is divided at random among all possible components, the expected distribution of the eigenvalues in the covariance or correlation matrix follows a broken stick distribution.[22] This model says that if we have a stick of unit length, broken at random into n segments, then the expected length of the kth longest piece is as follows:

$$b_k = \frac{1}{n}\sum_{i=k}^{n}\frac{1}{i} \tag{2}$$

As the expected values under the broken stick model are obtained in decreasing order, it is necessary to rank the relative proportions of the variance that are accounted for by the PCs in the same way. The estimated number of PCs is the maximum index where the observed relative proportion of variance is greater than or equal to the expected value from the broken stick distribution.

Randomization-based procedure

This procedure involves a randomization approach to generate a large number of data sets by scrambling the observed data in a manner of sampling without replacement.[15,16] These randomized data sets are then used to compute empirical *P* values for the statistics of interest characterize the internal structure of the eigenvalues in the correlation matrix. The test procedure is as follows: (1) randomize the values with all the attribute columns of the data matrix, (2) perform PCA on the scrambled data matrix, and (3) compute the test statistics. All 3 steps are repeated a total of $(B-1)$ times, where B is a large enough integer to guarantee the accuracy of estimating the *P* value; in practice, B is usually set to equal 1000. In each randomization, 2 test statistics are computed: (1) the eigenvalue λ_k for the kth principal component and (2) a pseudo *F* ratio computed as $\lambda_k / \sum_{i=k+1}^{n} \lambda_i$. Finally, the *P* value for each k and each statistic of interest is estimated to be the proportion of the test statistics in all data sets that are greater than or equal to the one in the observed data matrix.

Bayesian approximation methods

The probabilistic PCA model $\mathcal{M}_d, d \in \{1,\ldots,n\}$ assumes that each observation $\boldsymbol{x}_i \in \mathbb{R}^n$ $(i = 1,\ldots,m)$ arises from the following model:

$$\boldsymbol{x}_i = \boldsymbol{\mu} + H\boldsymbol{w}_i + \boldsymbol{\epsilon}_i \tag{3}$$

where μ is the mean vector, $\boldsymbol{w}_i \sim N(0, I_d)$ is a d-dimensional Gaussian latent vector, H is an $n \times d$ parameter coefficient matrix, and $\boldsymbol{\epsilon}_i \sim N(0, \sigma^2 I_n)$ is Gaussian noise. Tipping and Bishop[6] showed that the principal components of X can be retrieved using the maximum likelihood estimator of H, which is given as follows:

$$H_{ML} = U(L - \nu I_d)^{1/2} W,\ U^T U = I_d,\ WW^T = I_d \tag{4}$$

where U is the $n \times d$ matrix of ordered principal eigenvectors of $X^T X$, ν is an estimator of the true noise level σ^2, L is the $d \times d$ diagonal matrix with the estimates of the eigenvalues of the covariance matrix, and W is an arbitrary orthogonal matrix.

Finding the number of principal components is then equivalent to a model selection problem from $\mathcal{M}_d$. The optimal number of components can be obtained by applying the following criteria:

$$\hat{d} = \arg \max_{d \in \{1,\ldots,n\}} p(X \mid d) \tag{5}$$

where the number of components d is implicit in the matrix H_{ML}. With a noninformative prior, the mean vector $\boldsymbol{\mu}$ can be integrated out and the probability of X given H_{ML} and ν is obtained as follows:

$$\begin{aligned} &\boldsymbol{p}(X \mid H_{ML}, \nu) = \\ &m^{-d/2}(2\pi)^{-(m-1)d/2} \mid H_{ML} H_{ML}^T + \nu I_n \mid^{-(m-1)/2} \times \\ &\exp\left(-\frac{m}{2} tr(H_{ML} H_{ML}^T + \nu I_n)^{-1} S \right) \end{aligned} \tag{6}$$

where S is the sample covariance matrix.[7] With a prior $p(U, L, W, \nu)$, the probability of observing X given the signal dimensionality d is as follows:

$$p(X \mid d) = \int p(X \mid H_{ML}, \nu)\, p(U, L, W, \nu)\, dU dL dW d\nu \tag{7}$$

Because of the computational burden in calculating equation (7), Minka[7] proposed a conjugate prior for (U, L, W, ν) and derived a Laplace approximation of the marginal likelihood in equation (7) which is then used in equation (5) to compute the optimal number of components. This approximation approach has been empirically shown to be efficient in small-sample scenarios. In 2008, Hoyle[8] considered the high-dimensional scenarios when m and n grow to infinity and proposed another approximation where more terms in the asymptotic expansion of the integrand of equation (7) are used to guarantee more accurate evaluation. However, similar to Minka's Laplace approximation, it depends on the selection of the prior on (U, L, W, ν). This approach is not considered here because there is no publicly available implementation.

Sobczyk et al[9] explored 2 other high-dimensional scenarios: (1) n is fixed and $m \to \infty$ and (2) m is fixed and $n \to \infty$, where the second one is out of the scope of this article. For the 2 different scenarios, they modeled either the rows or the columns of the matrix via fixed effects models (with slightly different forms depending on the relative size of m and n). The 2 models are similar to equation (3) where either the matrix of PCA scores or the matrix of PCA loadings are used. They imposed a prior on these matrices and presented a methodology for approximating the posterior probability of d. Using 2 different prior distributions on terms similar to $\boldsymbol{w}$ in equation (3), they proposed 2 corresponding forms of PESEL where either all the singular values in PCA are homogeneous (abbreviation *homo*) or heterogeneous (abbreviation *hete*). Therefore, there are 4 criteria to maximize the posterior probability of d given the PESEL: PESEL_m^{hete} and PESEL_m^{homo} for Scenario 1 and PESEL_n^{hete} and PESEL_n^{homo} for Scenario 2. In the rest of this article, we study the same 3 criteria PESEL_m^{hete}, PESEL_n^{hete}, and PESEL_n^{homo} used in the work by Sobczyk et al.

Auer-Gervini model

We briefly review the Auer-Gervini method.[17] Suppose that $\boldsymbol{x}_i \in \mathbb{R}^n\ (i = 1,\ldots,m)$ are all columns of data matrix X, and $\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_m\}$ is a random sample with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ. Note that in the work by Auer and Gervini, they assume that the rows are i.i.d. Gaussian. Here, we make a transposition and assume the columns of X are i.i.d. Gaussian to guarantee the consistency of notations with previous sections. Write $\Sigma = \Gamma\Lambda\Gamma^T$ where $\Gamma = (\gamma_1,\ldots,\gamma_n)$ is orthonormal and $\Lambda = \mathrm{diag}(\lambda_1,\ldots,\lambda_n)$ with $\lambda_1 \geq \cdots \geq \lambda_n$. Let $\mathcal{M}_d$ be the model with d significant components or eigenvalues, that is, $\lambda_1 \geq \cdots \geq \lambda_d, \lambda_d > \lambda_{d+1}$, and $\lambda_{d+1} = \cdots = \lambda_n$, for $d \leq n-1$. Under $\mathcal{M}_d$, a random sample is as follows:

$$\boldsymbol{x} = \boldsymbol{\mu} + \sum_{k=1}^{d} z_k(\lambda_k - \lambda_{d+1})^{\frac{1}{2}}\boldsymbol{\gamma}_k + \lambda_{d+1}^{\frac{1}{2}}\boldsymbol{\epsilon}$$

where $z_1,\ldots,z_d$ are uncorrelated random variables with mean 0 and variance 1, and $\boldsymbol{\epsilon}$ is a random error vector with mean 0 and covariance matrix $\boldsymbol{I}_n$. Therefore, the problem of selecting the number of PCs is transformed into the problem of choosing the correct model $\mathcal{M}_d$.

A prior probability is assigned to $\mathcal{M}_d$ of the following form:

$$p(d) = C\exp\left(-\frac{m}{2}\theta d\right), \quad d = 0,\ldots,n-1 \tag{9}$$

where C is a normalizing constant that satisfies $\sum_{d=0}^{n-1} p(d) = 1$ for $\theta > 0$. Then, under certain approximations, one can use Bayes rule to derive a formula for the maximum posterior estimate of d as a function of the prior parameter θ:

$$\begin{aligned}\hat{d}(\theta) &= \delta(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_m)\\ &= \mathrm{argmax}_{0\leq d\leq n-1}\, p(\mathcal{M}_d \mid \boldsymbol{x}_1,\ldots,\boldsymbol{x}_m)\\ &\approx \mathrm{argmax}_{0\leq d\leq n-1}\, \hat{p}(\mathcal{M}_d \mid \boldsymbol{x}_1,\ldots,\boldsymbol{x}_m)\\ &= \mathrm{argmax}_{0\leq d\leq n-1}\, \frac{\hat{p}(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_m \mid \mathcal{M}_d)\,p(d)}{\hat{p}(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_m \mid \mathcal{M}_{n-1})\,p(n-1)}\\ &= \mathrm{argmax}_{0\leq d\leq n-1}\left\{\left(\frac{\hat{G}_d}{\hat{A}_d}\right)^{n-d} e^{\theta(n-1-d)}\right\}\end{aligned} \tag{10}$$

where $\hat{p}(\mathcal{M}_d \mid \boldsymbol{x}_1,\ldots,\boldsymbol{x}_m)$ is an estimator of $p(\mathcal{M}_d \mid \boldsymbol{x}_1,\ldots,\boldsymbol{x}_m)$, $\{\hat{\lambda}_k\}$ are the eigenvalues of the sample covariance matrix and also the maximum likelihood estimators of $\{\lambda_k\}$ under model $\mathcal{M}_d$, and $\hat{G}_d$ and $\hat{A}_d$ are the geometric and the arithmetic means of $\hat{\lambda}_{d+1},\ldots,\hat{\lambda}_n$, respectively. The formula describes $\hat{d}(\theta)$ as a nonincreasing step function with respect to θ, with $\hat{d}(0) = n-1$. The step function is then plotted and the "highest dimension for which the step length is significantly large" is selected to be the optimal number of components.

In other words, an exponential prior is placed on the number of significant components. The prior depends on a hyperparameter $\theta \geq 0$ that governs how fast the distribution decays. As θ goes to ∞, the prior drops off rapidly and the posterior estimate of the number of PCs will go to 0. Auer and Gervini proposed graphing the posterior estimate as a step function of θ, which can visually help select the highest "nontrivial step length." A large step length means that the estimated number of PCs is optimal under a wide range of prior model probabilities. However, the notion of "nontrivial step length" remains subjective, which is similar to the situation where one needs to select a recognizable "elbow" in the scree plot. Automating the definition of nontrivial step length is further complicated by the fact that the final step for $d = 0$ is theoretically infinite. We will operationalize the final subjective step by putting an upper bound on the largest "reasonable" estimate of θ and will develop criteria to automatically choose the significantly large step length.

Automating the Auer-Gervini method

As originally described, the Auer-Gervini model is a visual Bayesian approach, and the critical final step is to decide what constitutes a significantly large length of a step. This problem can be thought of as one of classification, in which the set of step lengths must be partitioned into 2 groups (short and long). We propose to test multiple algorithms to solve the problem as follows.

TwiceMean. Use twice the mean of the set of step lengths as a cutoff to separate the long and short steps. Intuitively, the idea is to view the distribution of the step lengths as exponential when the data arise from random noise. As the mean equals the standard deviation for an exponential distribution, twice the mean is the same as the mean plus one standard deviation and provides a plausible cutoff to select "long" step lengths. This simple idea is inspired in part by Chaterjee,[23] who considered recovery of low-rank matrices by thresholding singular values. He proposed that one could have a single universal choice of the threshold parameter which is slightly greater than 2 and gives near-optimal mean square error for singular value hard thresholding.

K-means. Because the goal is to partition the step lengths into 2 groups, a natural solution is to cluster them using the traditional *K*-means algorithm with $K = 2$. We seed the algorithm with starting centers using the minimum and maximum step lengths. As we will discuss in more detail below, the final step (when $d = 0$) is theoretically infinite. We will bound this last step but it will ensure that at least one step is "long."

Kmeans3. Our initial experience (data not shown) using *K*-means with large n found it to be overly conservative when assigning steps to the "long" group. Given its dependence on

Euclidean distances, that is exactly how one should expect it to perform when the true mixture distribution is skewed right. To address this problem, we modified the algorithm as follows. If the number of objects is large $(n \geq 55)$, we use the *K*-means algorithm, with $K = 3$ and seeds of the minimum, median, and maximum values, to separate the step lengths into 3 groups: Low, Intermediate, and High. We then treat both Intermediate and High groups as long.

Spectral. Use spectral clustering to divide the step lengths into 2 groups. Spectral clustering is one of the most popular modern clustering algorithms. It sometimes outperforms the traditional clustering algorithms such as *K*-means.[24,25]

CPT. Instead of simply clustering the step lengths into 2 groups, we can instead sort them into increasing order and view the problem as one of change point detection. One existing solution to this problem is provided by the At Most One Change (AMOC) method implemented by the cpt.mean function from the changepoint R package.[26] The detection of the first change point is posed as a hypothesis test and a generalized likelihood ratio–based approach is extended to changes in variance within normally distributed observations (the reader can consult Hinkley[27] and Gupta and Tang[28] for more details).

CpmFun. The cpm R package defines several other "Change Point Models."[29] These are implemented by the detectChangePointBatch function, which processes the step lengths in one batch and returns information regarding whether the sequence contains a change point. The default is to use the "Exponential" method, which computes a generalized likelihood ratio statistic for the exponential distribution.

Ttest. We also implemented a novel change point detection algorithm based on the *t* test. We begin by sorting the steps lengths in increasing order. Then, we compute the gaps between successive step lengths. At each (sorted) step, we use the *t* distribution to determine the likelihood that the next gap is larger than expected from the previously observed gaps. The first time that the next gap is significantly larger than expected, we assert that this next step length is the smallest one that constitutes a "long" step length.

Ttest2. Where the *K*-means algorithm was found to be conservative, the Ttest algorithm just described was sometimes found to be anticonservative. This can happen when the first few step lengths are all about the same size, which yields a small standard deviation. In this case, a relative short next step will be falsely discovered based on the Ttest criterion. To avoid this problem, it may be appropriate to include the next (test) step length and gap when estimating the mean and standard deviation of the gap distributon. We modified the Ttest algorithm in this way to make it more conservative.

Bounding the last step

All of the proposed methods for separating short from long steps require us to bound the permissible length of the final step when $\hat{d} = 0$, which would otherwise be infinite. This step is important because it allows the algorithm to conclude in some cases that the only long step is the final one, and the true number of principal components should equal 0. We use the largest of the following 3 quantities:

1. $\theta_0 = 3\%$ further than the final change point (to $d = 0$) in the step function.
2. $\theta_0 = -2\log(0.01)/m$, where m is the number of attributes. This procedure selects the value of θ for which 99% of the prior probability is assigned to $d = 0$.
3. $\theta_0 = (18.8402 + 1.9523 * n + 0.0005 * n^2)/m$ if $m \geq n$ and $\theta_0 = (18.8402 + 1.9523 * n + 0.0005 * n^2) * m/n^2$, otherwise, where n is the number of objects. This formula was derived empirically from a Monte Carlo study on data sets with random noise. We estimate θ_0 as the maximum of the empirical largest change point in the step function for various values of m and n in 2 scenarios: $m \geq n$ and $m < n$. It can be seen as the value where the maximum point for $d = 0$ could be achieved when various *d's* are sharing the prior information on θ under the uninformative noise structure.

Note that all simulations and computations in this article were performed using version 3.2.2 of the R statistical software environment with version 1.1.3 of the PCDimension package, which we have developed, version 2.3.3 of the nFactors package, version 1.39 of the FactoMineR package, and version 0.7.1 of the pesel package. The details on how to select the number of PCs for a simple example using the PCDimension package are provided in the supplementary material.

Results

Simulation study

We follow a Monte Carlo procedure to study the robustness of different types of methods described above for estimating the number of PCs. For real data sets, we will never know the "correct" answer. So, we simulate a collection of data sets with different correlation structures to compare the numerical Auer-Gervini model we have implemented with the other types. We start by supplying details about the correlation structures and data sets used in the simulations.

Simulated datatypes. We use a protocol similar to those discussed in recent papers.[5,14,17,30–32] In the simulations, the number of measured attributes is taken to be either $m = 100$ or $m = 400$. The range of 100 to 400 is chosen to represent small to moderately large data sets. We also consider data sets with either $n = 24$ or $n = 96$ objects. We view 24 objects as a small data set and 96 objects as a moderately large one.[33] The number

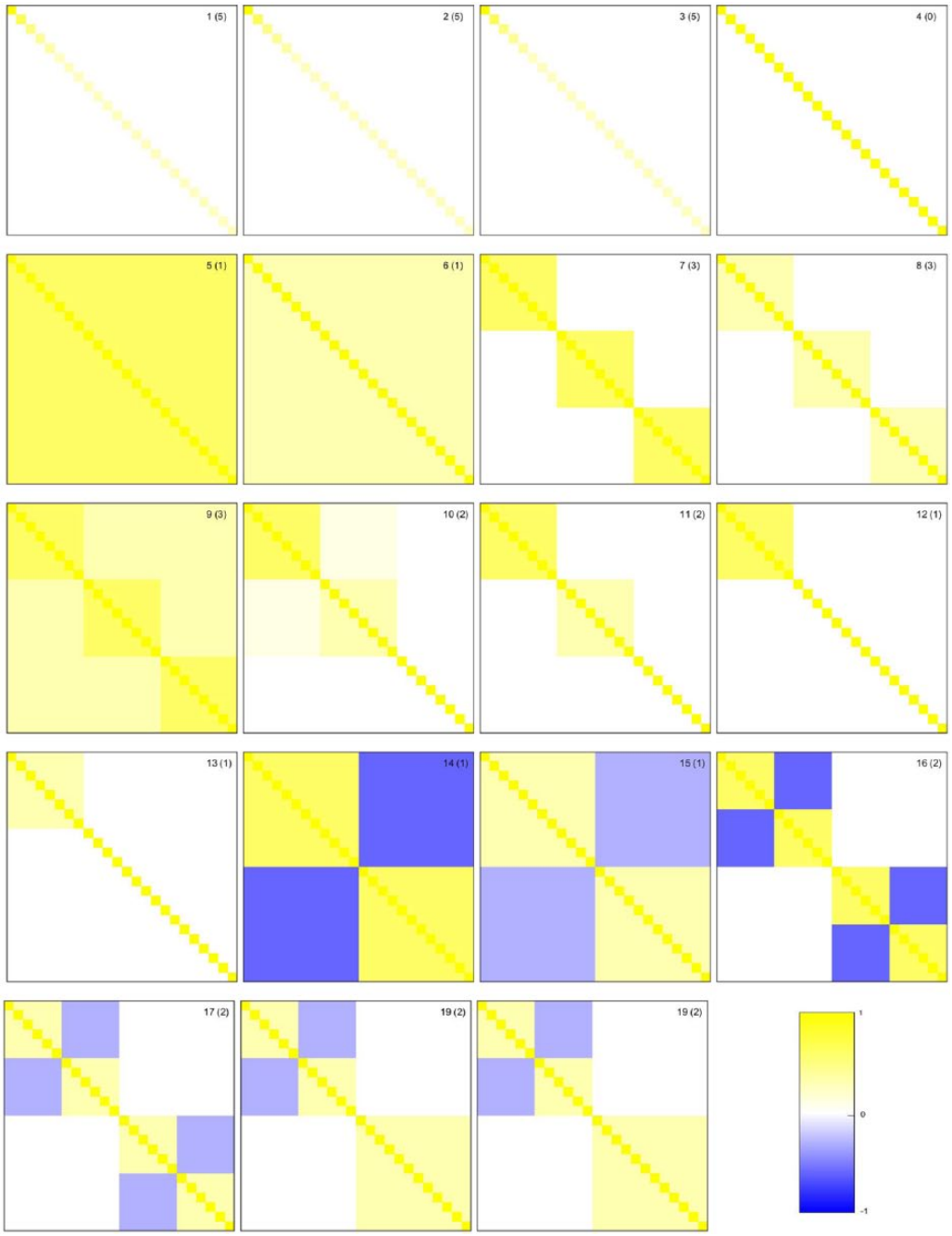

Figure 1. The 19 correlation matrices considered in the simulation study for 24 objects. Values of correlations are provided by the colorbar. Numbers in parentheses correspond to the known dimension.

of significant diagonal blocks is either the number shown in Figure 1 or twice that number (with finer correlation structures of double group blocks). By varying both the number of objects and the number of correlated blocks, we can explore the effects of the number of nontrivial components and the number of objects per component. To also explore the effects of different combinations of additional factors, including eigenvalue distributions, signed or unsigned signals, uncorrelated variation, and unskewed normal or skewed distributions, we use the 19 different covariance structures and correlation matrices illustrated in Figure 1. Matrices 1–3 are covariance matrices Σ with different marginal distributions: normal, marginal gamma, and marginal t distribution, where $\Sigma = \Gamma\Lambda\Gamma^T$, and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$ with $\lambda_i = 1/i$ for $1 \leq i \leq 5$ and $\lambda_i = 1/10$ for $i \geq 6$. Matrices 4–19 are correlation matrices $\text{corr}(X)$ and we set the corresponding covariance matrices to be $\sigma^2 * \text{corr}(X)$ where $\sigma^2 = 1$. Note that the correlation is the standardized version of the covariance, that is, $\text{corr}(X) = D^{-1}\Sigma D^{T-1}$ where $D = (\text{diag}(\Sigma))^{-1/2}$. We use this form for most of the covariance matrices because we want to control the signal to noise ratio (basically, σ^2 / σ_e^2) to test the robustness of methods to the noise level σ_e^2. Matrix 4 contains only noise; it is a purely uncorrelated structure.

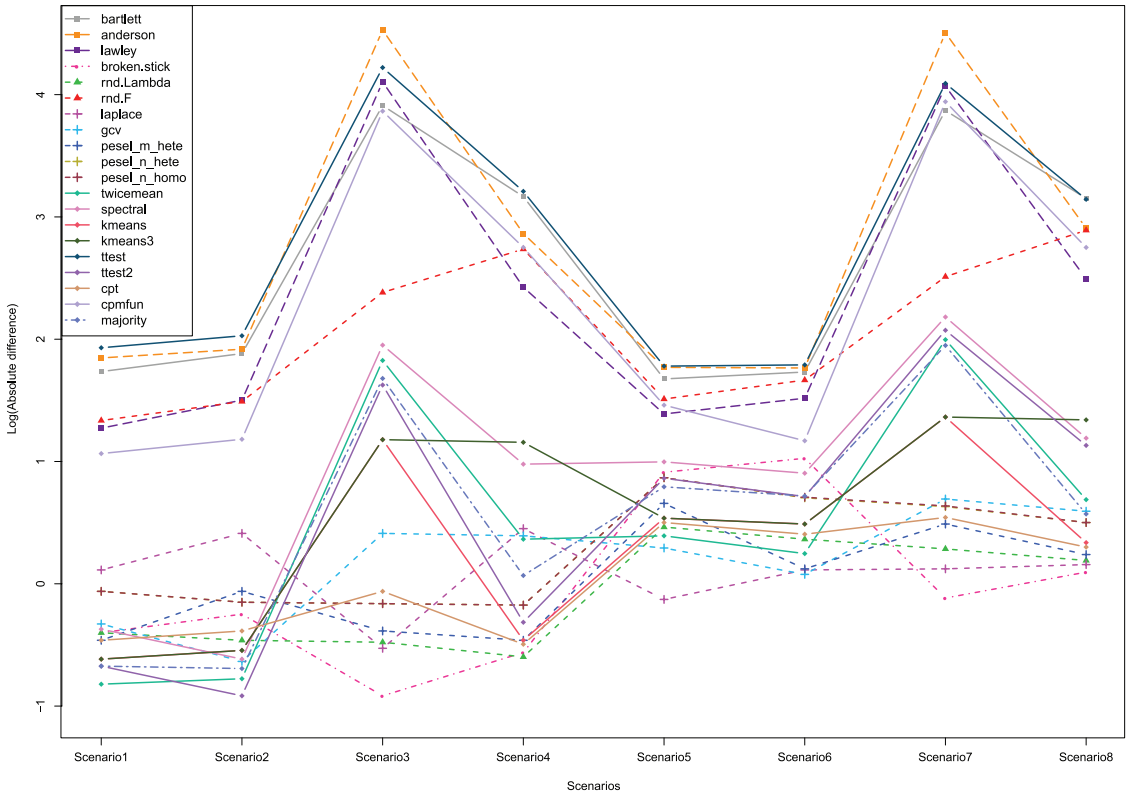

Figure 2. Log-transformed mean values, across the correlation matrices, of the absolute difference between the known dimension and the sample estimates from 15 different methods in 8 simulation scenarios (Scenario 1: 24×100, 1X correlated blocks; Scenario 2: 24×400, 1X; Scenario 3: 96×100, 1X; Scenario 4: 96×400, 1X; Scenario 5: 24×100, 2X; Scenario 6: 24×400, 2X; Scenario 7: 96×100, 2X; and Scenario 8: 96×400, 2X).

Matrices 5 and 6 represent correlation structures with various homogeneous cross-correlation strengths (unsigned signals) 0.3 and 0.8. Matrices 7–13 are correlation matrices where between-group (0.3, 0.1, or 0) and within-group (0.8 or 0.3) correlations of objects are fixed.[14,31] Matrices 14–19 are correlation structures where negative cross-correlations (−0.8 or −0.3, signed signals) are considered within groups and mixture of signed and unsigned signals are also included.

Empirical simulation results. For each of the 19 scenarios, we simulate 1000 data sets. Then, the numbers of components are estimated based on all (19 variants of the) types of methods. That is, for each variant within each type of model, we compute the estimated dimension. We also investigate a "majority rule" procedure for the Auer-Gervini model; that is, the dimension that more than 3 criteria out of 8 selected is the one that is estimated by the majority rule. Then, we calculate the absolute difference between the known dimensions and the estimated ones for each simulated sample and correlation structure. The mean of the absolute differences over both 1000 simulated data sets and 19 correlation matrices is plotted in Figure 2. The corresponding numeric values are also provided in Supplementary Table 1. The values in this table can help assess the quality of each method. The closer to 0 the values are, the better the corresponding variant within the type of method. However, the values do not describe whether a method overestimates or underestimates the number of nontrivial PCs.

In Figure 2 and Supplementary Table 1, one can see that, as anticipated, the results from most algorithms are better with fewer correlated blocks (Scenarios 1-4), probably because there are more objects representing each block. Also, accuracy in almost all methods is much better with fewer objects (Scenarios 1-2 and 5-6). The situation is more complicated when the number of attributes changes. In general, the worst performance occurs when the data matrix is nearly square (Scenarios 3 and 7).

Overall, the most accurate methods when averaging across all 8 scenarios are (1) the rnd-Lambda algorithm (mean deviation, MD = 1.005), (2) the PESEL method with criterion pesel_m^{hete} (MD = 1.105), (3) the Auer-Gervini model with criterion "CPT" (MD = 1.135), and (4) Minka's Laplace approximation (MD = 1.136). Interestingly, the rnd-Lambda and Minka-Laplace algorithms each produce the best average performance in only 1 out of 8 scenarios. Moreover, the Auer-Gervini CPT model and the pesel_m^{hete} model are not the best performers in any of the 8 individual scenarios. By contrast, 3 other methods (the "TwiceMean" Auer-Gervini model, the "ttest2" Auer-Gervini model, and generalized cross-validation) have the best performance in 1 out of 8 scenarios. The simple broken stick method is the best performer in 3 out of 8 scenarios.

This variabililty in the winners suggests that we look more closely at how the simulation parameters affect performance. When there are only 24 objects, the 4 best methods are (1) "TwiceMean" Auer-Gervini (MD = 0.915), (2)

Table 1. Average running time of all types of methods across correlation matrices (unit: seconds).

RULES	ORIGINAL BLOCKS (1X)				TWICE BLOCKS (2X)			
	24 OBJECTS		96 OBJECTS		24 OBJECTS		96 OBJECTS	
	$M = 100$	$M = 400$	$M = 100$	$M = 400$	$M = 100$	$M = 400$	$M = 100$	$M = 400$
Broken stick	0.002	0.003	0.008	0.018	0.002	0.003	0.008	0.018
Bartlett's test	0.004	0.004	0.013	0.015	0.004	0.004	0.013	0.016
GCV	0.004	0.006	0.012	0.032	0.003	0.006	0.012	0.032
Minka's Laplace	0.005	0.007	0.024	0.047	0.005	0.007	0.024	0.048
PESEL	0.010	0.102	0.026	0.191	0.008	0.104	0.026	0.183
Auer-Gervini	0.035	0.034	0.234	0.269	0.033	0.035	0.236	0.275
Rand. based	2.121	2.990	9.253	20.152	1.832	3.131	9.212	20.061

generalized cross-validation (MD = 0.918), (3) rnd-Lambda (MD = 1.005), and (4) "kmeans" Auer-Gervini (MD = 1.115). When there are 96 objects, the 4 best performers are (1) broken stick (MD = 0.7400), (2) rnd-Lambda (MD = 0.928), (3) pesel_m^{bete} (MD = 1.053), and (4) Minka-Laplace (MD = 1.115). A slight shuffling of the order of the same set of high-performing methods occurs when we vary the number of blocks (broken stick is best with 1X [MD = 0.605], Minka-Laplace with 2X [MD = 1.075]) or the number of attributes (Minka-Laplace is best with 100 [MD = 0.93], and rnd-Lambda with 400 [MD = 0.9575]).

The variants of Bartlett's test, as implemented in R, have the worst performance of all the stopping rules we have considered (Figure 2). Furthermore, the rnd-F algorithm is not as good as expected because a previous study found it to be one of the most successful rules tested.[14] Even though the CPT criterion is one of the best overall, and the TwiceMean criterion is the best when there are 24 objects, some of the criteria that we use to automate the Auer-Gervini model are not good candidates for computing the dimension. For example, the "Ttest" and "CPM" criteria often result in large deviations from the true dimension. The PESEL criteria pesel_n^{bete} and pesel_n^{homo} have middle-of-the-road performance, which may by due to considering $n \ll m$ in this article.

Running time. In addition to the absolute differences between the true dimension and the estimates, we computed the average running time of all types of methods over all correlation matrices per data set (Table 1). All timings were conducted on a computer with "Intel Xeon CPU E5-2690 v3 @ 2.60-GHz 2.59-GHz" processors running Windows Server 2008 R2 Enterprise. Note that the time shown in the table is the total time of all the variants or criteria within each type of model. It is obvious that the computation time becomes longer as the number of objects or attributes increases, and there is almost no change in time usage when the number of blocks is doubled. From the table, we can see that Bartlett's test and the broken stick method use the least time in computing the number of components. However, the accuracy using the broken stick approach is much better than in Bartlett's test. The most accurate overall method, rnd-Lambda, is by far the slowest, taking several orders of magnitude more time than the other methods. Of the remaining 3 methods, which achieve nearly the same level of accuracy, Minka's Laplace approximation is the fastest and the "CPT" Auer-Gervini model is the slowest.

High-accuracy methods: rnd-Lambda and Auer-Gervini with CPT. More detailed results on the performance of the rnd-Lambda algorithm are presented in Figure 3 and Supplementary Tables 2 and 3 for the case of the original block structure shown in Figure 1. Similar results for the "CPT" criterion in the Auer-Gervini model are presented in Figure 4 and Supplementary Tables 4 and 5. For each covariance or correlation matrix, we computed the percentage of deviations between the estimates and the known dimension. The results are similar regardless of the number of attributes (100 or 400) or objects (24 or 96). Both methods tend to underestimate the dimension for covariance matrices of unskewed or skewed distributions (matrices 1-3). They are quite accurate for correlation matrices of normal distribution (matrices 4-19). When they make errors with the normal distribution, the rnd-Lambda algorithm is more likely to slightly overestimate the dimension, whereas the Auer-Gervini CPT method is more likely to underestimate.

Special-case methods: TwiceMean and broken stick. We present details on the performance of the "TwiceMean" criterion in the Auer-Gervini model and the broken stick method, which were best when restricting to either the 24-object or 96-object

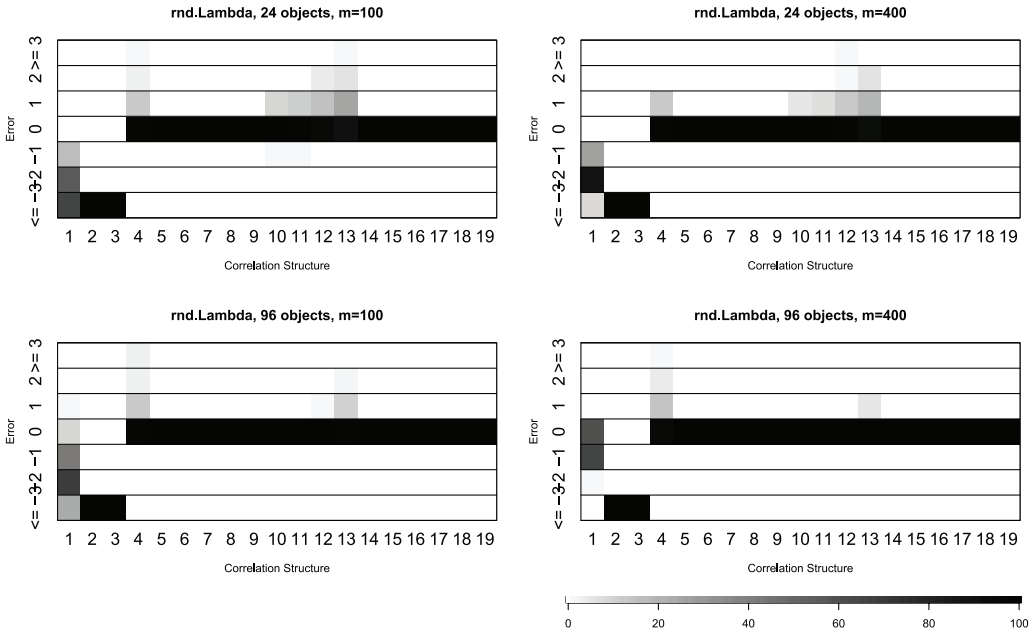

Figure 3. Percentage of deviations between the estimate from randomization-based procedure (rnd-Lambda) and known dimension.

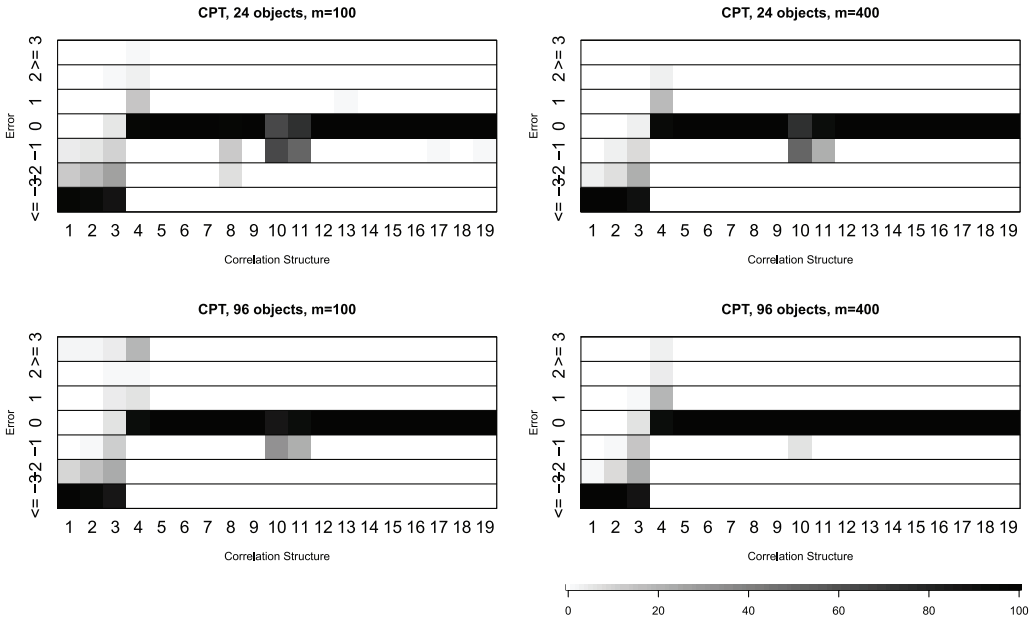

Figure 4. Percentage of deviations between the estimate from the "CPT" criterion in the Auer-Gervini model and known dimension.

simulations, respectively, as shown in Figure 5 and Supplementary Tables 6 and 7. For data sets with 24 objects, the Auer-Gervini method with criterion "TwiceMean" is very accurate for uniform matrices (matrices 5 and 6), correlated matrices (matrices 7-13) and unsigned data with or without signed signals (matrices 14-19). When the sample covariance matrix is either skewed or unskewed (matrices 1-3), the dimension is usually underestimated. Also, the results do not vary too much with different numbers of attibutes. For 96 objects, there is not much difference between the results of the broken stick method in Supplementary Table 7 and that of the criterion "TwiceMean" in Supplementary Table 6. And the results of the broken stick method for 100 attributes are slightly better than those for 400 attributes.

Robustness to random noise. We investigated the influence of random noise on the ability of different methods to correctly detect the underlying structure. We conducted additional simulation studies by adding different levels of (i.i.d. normal) noise corresponding to 3 different values of the variance: $\sigma_e^2 = 0.01$, $\sigma_e^2 = 0.1$, and $\sigma_e^2 = 1$. Summaries of the absolute difference between the known dimension and the estimates made using various methods under different levels of noise are presented in Supplementary Tables 8 to 10. Although the error rate tends to increase with increasing noise, we found that the relative performance of the methods is consistent regardless of the size of the noise. That is, most algorithms still perform better with fewer objects, and accuracy is almost always worse when the

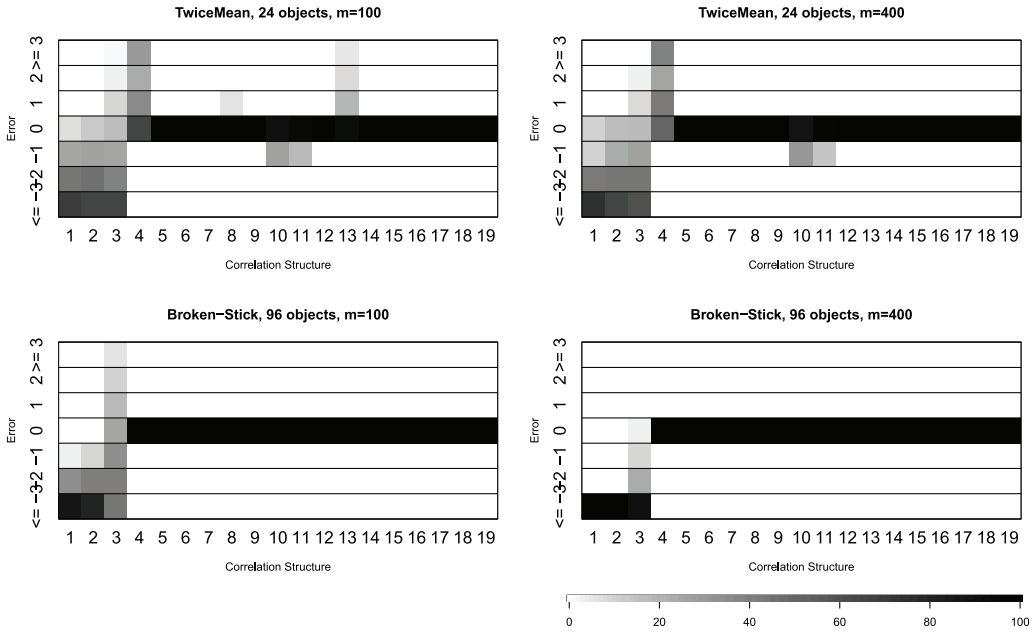

Figure 5. Percentage of deviations between the estimate from either the "TwiceMean" criterion in the Auer-Gervin model or the broken stick model and known dimension.

Table 2. Number of principal components (PCs) from different algorithms on reverse phase protein array data.

RULES	AUER-GERVINI		BROKEN STICK	RAND. BASED	MINKA	GCV	PESEL
	TWICEMEAN	CPT	BROKEN-STICK	RND-LAMBDA	LAPLACE	GCV	PESEL.M.HETE
PCs	6	1	1	8	20	12	15

number of blocks doubles. Most importantly, the best method for each scenario does not change in most cases as the noise moves from 0 to 1. The "TwiceMean" criterion in the Auer-Gervini model is better when the number of objects is small relative to the number of objects, whereas the broken stick method is better when the number of objects is close to the number of attributes. Finally, regardless of the noise level, the rnd-Lambda algorithm, the Auer-Gervini model with "CPT" criterion, the PESEL method with criterion pesel_m^{bete}, and Minka's Laplace approximation outperform the others on average across all scenarios.

Decomposing the apoptosis pathway in AML

Since the introduction of gene expression microarrays in the 1990s, most statistical analyses of omics data have treated pathways as second-class objects, in the following sense: primary analyses are performed at the gene level. That is, the data are first analyzed gene by gene to find differences between known groups of patients such as responders and nonresponders. Then, a significance cutoff is chosen and a second statistical test conditional on the gene-by-gene results (such as gene set enrichment analysis[34]) is performed to infer which pathways differ between the 2 groups. One reason analysts give precedence to individual genes is that univariate analyses are easier than the multivariate ones needed for pathways. However, many biologists are more interested in pathways than in individual genes because they give a higher-level functional picture of biological behavior.

Informally, biologists talk about pathways as though they are 1-dimensional entities. At a cell level they are "on" or "off"; at a tissue level, they have a simple "degree of activation." But we hypothesize that most pathways, including the apoptosis signaling pathway, are intrinsically multidimensional. To test this hypothesis, we used a subset of RPPA data on samples collected from 511 patients with AML.[35,36] The subset consists of 33 proteins that are involved in the apoptosis signaling pathway. Apoptosis is known to be an essential component of several processes including normal cell turnover, proper development and functioning of the immune system, and chemical-induced cell death.[3] It is generally characterized by distinct morphological states and energy-dependent biochemical mechanisms. Even though many important apoptotic proteins have been identified, the molecular mechanisms of these proteins still remain to be elucidated.

We applied different methods to determine the number d of significant components in this RPPA data set; the results are displayed in Table 2 and Supplementary Table 11. The number of components that they find is highly variable, ranging from $d = 1$ to 31. Even the best methods from our simulations give

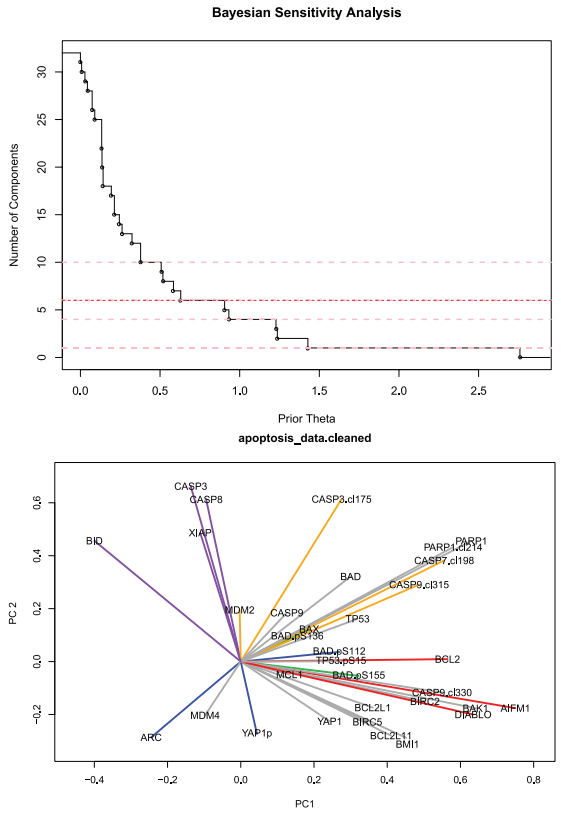

Figure 6. Analysis of AML RPPA data. Auer-Gervini step function relating the prior hyperparameter θ to the maximum posterior estimate of the number $\hat{d}$ of significant principal components (top). Projection of proteins on the space of the first 2 components; colors denote different clusters (bottom).

different values. However, our simulation studies suggest that the TwiceMean Auer-Gervini method works particularly well when the number of objects (in this case, 33 proteins) is small compared with the number of attributes (in this case, 511 samples). In the top panel of Figure 6, we have plotted the maximum posterior estimate of the number of components as a function of the prior parameter θ; this plot gives better support for $d = 6$ than for $d = 8$.

To understand the biology driving these mathematical principal components, we then projected the proteins into the 6-dimensional PC space and used their directions to cluster them using a von-Mises Fisher mixture model.[37] Using the Bayesian information criterion for model selection, we found an optimal clustering into 6 groups of proteins, which are displayed in different colors in a 2-dimensional projection in the bottom panel of Figure 6. The 6 protein clusters are as follows:

1. AIFM1, BCL2, and DIABLO, which we interpret as a block corresponding to the mitochondrial release of apoptosis-inducing factor[38];
2. BID, CASP3, CASP8, and XIAP, which are part of a caspase 3 feedback loop[39];
3. CASP3.cl175, CASP7.cl198, CASP9.cl315, and MDM2, a cleaved caspase block[40];
4. ARC, BAD.pS112, and YAP1p;
5. BAD.pS136, BAD.pS155, and BAX; and
6. The core group of 16 apoptosis-related proteins: BAD, BAK1, BCL2L1, BCL2L11, BIRC2, BIRC5, BMI1, CASP9, CASP9.cl330, MCL1, MDM4, PARP1, PARP1.cl214, TP53, TP53.pS15, and YAP1.

The fact that 3 of these 6 clusters can be immediately identified from the literature as coherent biological subcomponents of the apoptosis pathway provides strong support for our approach.

Conclusions

Principal component analysis is one of the most popular and important techniques in the multivariate analysis of general data sets. However, because principal components are linear combinations of correlated variables, these components usually lack interpretability when analyzing biological data sets, especially transcriptomic or proteomic data sets from patients with cancer. Our study of the apoptosis pathway using proteomic data from patients with AML shows that we can use mixture model clustering in principal component space to replace the uninterpretable mathematical components with natural collections of related genes that enhance the biological interpetability of the decomposition. We expect that applying these methods to the genes or proteins in other signaling pathways will divide these pathways into 1-dimensional "building blocks" that are interpretable, robust, and can yield new biological insights. It would be of particular interest to apply these ideas to overlapping pathways to better understand the way similar components are reused in different contexts.

Our ability to find biologically interpretable components, however, depends in a fundamental way on being able to determine the dimension of the principal component space. To accomplish this task, we introduced the PCDimension R package, which implements 3 types of models—the broken stick method, the randomization-based procedure of ter Braak,[15,16] and our enhancments to the model developed by Auer and Gervini[17]—to compute the number of significant principal components. Through extensive simulations, we have shown that the enhanced Auer-Gervini methods are competitive with the methods that performed best in previous comparative studies.

It has been claimed that simulation of multivariate data sets can always be criticized as unrepresentative because they can never explore more than a tiny fraction of the wide range of possible covariance and correlation structures.[4] As with previous simulation studies, our work may have the same limitation. However, Ferre has also pointed out that simulations are the only way to test and compare these methods.[12] It is still valuable to compare methods empirically when the dimension of the data set is known, and factors of interest can be manipulated under simulation. We have endeavored to explore a wide variety of different correlation structures to identify settings where each method is likely to fail.

In our simulations, the variants of Bartlett method clearly had the worst performance. This finding may be somewhat surprising: Peres-Neto et al[14] found these methods to be only a little worse than the best performers in their simulations and concluded that they were actually the best for distinguishing $d = 0$ from $d \geq 1$. There are 2 factors that distinguish their simulations from ours. First, we considered a wider variety of correlation structures. The matrices considered by Peres-Neto are represented by our matrices 4 to 13. Second, the matrices we considered were larger. Motivated by problems from ecology, they looked at matrices that were $9 \times (30 \text{ or } 50)$ or $18 \times (60 \text{ or } 100)$. Motivated by the larger gene or protein expression data sets currently being produced in biology, we looked at matrices that were $(24 \text{ or } 96) \times (100 \text{ or } 400)$. We think that both factors contribute to the different results. Supplementary Figure 3 shows that the errors arise primarily from matrices 1 to 4 and 10 to 12. This subset is comprised precisely of those matrices that include a substantial amount of unstructured noise. We suspect that the underlying difficulty arises because the stepwise hypothesis tests give rise to a classical problem of multiple comparisons. Thus, the likelihood of incorrectly rejecting a null hypothesis (and inflating the dimension) increases. This may also explain why Ferre[12] cautioned that Bartlett's test can overestimate the number of components.

The rnd-F method introduced by ter Brack also performed significantly worse in our simulations than in those of Perres-Neto. The plots in Supplementary Figure 3 reveal 2 things. First, for virtually every correlation matrix we considered, rnd-F is more variable and less accurate than rnd-Lambda. Second, the most serious large errors arise from matrices 10 to 13. These matrices contain a mix of both highly structured data and completely unstructured noise. Similar matrices were considered in the previous study, so we conclude that the rnd-F method simply works poorly with larger matrices.

We investigated the graphical Bayesian method of Auer and Gervini[17] in some detail. Specifically, we introduced and tested 8 algorithms to enhance the method by automatically selecting the number of components. Two of these—the novel T test–based changepoint algorithm and the exponential model from the CPM package—were abysmal. In virtually every simulation, the T test seriously overestimates the number of components. The exponential CPM model overestimates the number at least half the time. The remaining 6 methods have acceptable performance most of the time.

The overall winner in terms of accuracy from our simulation study is the rnd-Lambda randomization-based procedures. This additional accuracy, however, was obtained at a sustantial cost in computation time. The randomization methods take at least 2 orders of magnitude longer than any other methods that we studied. Three other methods were competitive with rnd-Lamda in terms of overall accuracy, but considerably faster: the Auer-Gervini model with criterion "CPT," the PESEL method with criterion pesel_m^{bete}, and Minka's Laplace approximation. It is interesting to note, however, that the 4 best overall methods rarely give the absolute best results for any fixed size of data matrix. Their ultimate strength is their consistency: their estimates are always competitive, and the average error in the estimated dimension is always less than 2.

Two other methods perform well in complementary settings. The broken stick model is most accurate when there are 96 objects, and the Auer-Gervini method using the TwiceMean criterion is the most accurate when there are 24 objects. The TwiceMean criterion appears to overestimate the number of components when the size of the data matrix increases. By contrast, the broken stick model appears to benefit from having extra data available. MacArthur[22] and De Vita[41] showed that the broken stick model worked well when fitting the relative abundance data of species in ecological populations. It is possible that the distribution of the expected lengths in equation (2) will be better approximated with larger data matrices. This may explain why its performance improves.

Our simulation studies uncovered at least 2 (possibly related) contexts where it is particularly difficult to estimate the number of components correctly. Every reasonable method severely underestimates the dimension for correlation matrices 1 to 3. In addition, most methods overestimate the dimension when there are 96 objects and 100 attributes, especially when we doubled the number of correlated blocks. In both cases, there is very little redundancy in the signals we are trying to detect. (The biological contexts where we expect to apply these methods are expected to contain considerable redundancy in the form of highly correlated genes or proteins.) To handle more general data sets, however, new methods will need to be developed to improve performance in these examples without sacrificing it in other examples. Further work will also be needed to clarify what kinds of structural changes occur as the number of objects increases from 24 to 96 and beyond.

We do not expect our study to be the final word on how to determine the number of significant principal components; similar to Ferre,[12] we must conclude that there is no ideal solution to the problem. If forced to choose one method for all data sets, we would pick the Auer-Gervini model using the CPT criterion, as it is both reasonably accurate and reasonably fast. This would especially be the case if we were trying to analyze many data sets at once; for example, when performing an analysis similar to the one in our AML data set for a long list of different biological pathways or gene sets. When focused on only one data set, we would compute the estimates from multiple methods (including rnd-Lambda, PESEL with criterion pesel_m^{bete}, Minka's Laplace approximation, broken stick, and Auer-Gervini with TwiceMean), and review them in light of both the traditional scree plot and the Auer-Gervini plot of the maximum posterior dimension as a function of the hyperparameter. When we used this method with the RPPA apoptosis data set, we successfully selected a dimension that produced clusters that made biological sense. We believe that this combination of analytical and graphical methods, as provided in the

PCDimension package, will guide researchers to the most reliable results.

Author Contributions

MZ performed all analyses and drafted the manuscript. SMK performed the AML experiments and helped with the design of the project. KRC designed the project and oversaw all parts of the analysis.

REFERENCES

1. Weinstein JN, Myers TG, O'Connor PM, et al. An information-intensive approach to the molecular pharmacology of cancer. *Science*. 1997;275:343–349.
2. Eisen MB, Spellman PT, Brown PO, et al. Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci USA*. 1998;95:14863–14868.
3. Elmore S. Apoptosis: a review of programmed cell death. *Toxicol Pathol*. 2007;35:419–516.
4. Jolliffe LT. *Principal Component Analysis*. 2nd ed. New York, NY: Springer; 2002.
5. Josse J, Husson F. Selecting the number of components in principal component analysis using cross-validation approximations. *Comput Statist Data Anal*. 2012;56:1869–1879.
6. Tipping ME, Bishop CM. Probabilistic principal component analysis. *J Roy Stat Soc B*. 1999;61:611–622.
7. Minka TP. *Automatic Choice of Dimensionality for PCA in NIPS*. Vol 13. Cambridge, MA: MIT Press; 2000.
8. Hoyle DC. Automatic PCA dimension selection for high dimensional data and small sample sizes. *J Mach Learn Res*. 2008;9:2733–2759.
9. Sobczyk P, Bogdan M, Josse J. Bayesian dimensionality reduction with PCA using penalized semi-integrated likelihood. *J Comput Grap Statist*. 2017;26:826–839.
10. Cattell RB. The scree test for the number of factors. *J Multivar Behavior Res*. 1966;1:245–276.
11. Legendre L, Legendre P. *Ecologie Numerique*. Amsterdam, The Netherlands: Masson; 1983.
12. Ferre L. Selection of components in principal component analysis: a comparison of methods. *Comput Statist Data Anal*. 1995;19:669–682.
13. Bartlett MS. Tests of significance in factor analysis. *Br J Mathemat Statist Psychol*. 1950;3:77–85.
14. Peres-Neto PR, Jackson DA, Somers KM. How many principal components? stopping rules for determining the number of non-trivial axes revisited. *Comput Statist Data Anal*. 2005;49:974–997.
15. ter Braak CFJ. CANOCO: a Fortran program for canonical community ordination by [partial] [detrended] [canonical] correspondence analysis, principal components analysis and redundancy analysis (version 2.1). Agricultural Mathematics Group, Report LWA-88-02, Wageningen, The Netherlands; 1988.
16. ter Braak CFJ. *Update notes: CANOCO (Version 3.1)*. Wageningen, The Netherlands: Agricultural Mathematics Group; 1990.
17. Auer P, Gervini D. Choosing principal components: a new graphical method based on Bayesian model selection. *Commun Statist Simulat Comput*. 2008;37:962–977.
18. Raiche G, Magis D. Package "nFactors," 2015, https://cran.r-project.org/web/packages/nFactors/nFactors.pdf.
19. Husson F, Josse J, Le S, et al. Factominer: multivariate exploratory data analysis and data mining with R, 2014, https://cran.r-project.org/web/packages/FactoMineR/index.html.
20. Lawley DN. Tests of significance for the latent roots of covariance and correlation matrices. *Biometrika*. 1956;43:128–136.
21. Anderson TW. Asymptotic theory for principal component analysis. *Ann Mathemat Statist*. 1963;34:122–148.
22. MacArthur R. On the relative abundance of species. *Am Naturalist*. 1960;94:25–36.
23. Chatterjee S. Matrix estimation by universal singular value thresholding. *Ann Statist*. 2015;43:177–214.
24. Azar Y, Fiat A, Karlin AR, et al. Spectral analysis of data. Paper presented at: Proceedings of the Thirty-third Annual ACM Symposium on Theory of Computing; Heraklion, Greece; July 6-8, 2001:619–626.
25. Bai Z, Silverstein JW. *Spectral Analysis of Large Dimensional Random Matrices*. 2nd ed. New York, NY: Springer; 2010.
26. Killick R, Eckley IA. Changepoint: an R package for changepoint analysis. *J Statist Software*. 2014;58:1–19.
27. Hinkley DV. Inference about the change-point in a sequence of random variables. *Biometrika*. 1970;57:1–17.
28. Gupta AK, Tang J. On testing homogeneity of variances for Gaussian models. *J Statist Comput Simulat*. 1987;27:155–173.
29. Ross GJ. Parametric and nonparametric sequential change detection in R: the CPM package. *J Statist Software*. 2015;66:1–20.
30. Jackson D. Stopping rules in principal component analysis: a comparison of Heuristical and statistical approaches. *Ecology*. 1993;74:2204–2214.
31. Peres-Neto PR, Jackson DA, Somers KM. Giving meaningful interpretation to ordination axes: assessing loading significance in principal component analysis. *Ecology*. 2003;84:2347–2363.
32. Dray S. On the number of principal components: a test of dimensionality based on measurements of similarity between matrices. *Computat Statist Data Anal*. 2008;52:2228–2237.
33. Zwick WR, Velicer WF. Factors influencing four rules for determining the number of components to retain. *Multivar Behavior Res*. 1982;17:253–269.
34. Subramanian A, Tamayo P, Mootha VK, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc Nat Acad Sci*. 2005;102:15545–15550.
35. Kornblau S, Singh N, Qiu Y, et al. Highly phosphorylated FOXO3A is an adverse prognostic factor in acute myeloid leukemia. *Clin Cancer Res*. 2010; 16:1865–1874.
36. Hu CW, Kornblau SM, Slater JH, et al. Progeny clustering: a method to identify biological phenotypes. *Sci Rep*. 2015;5:12894.
37. Banerjee A, Dhillon I, Ghosh J, et al. Clustering on the unit hypersphere using von Mises-Fisher distributions. *J Mach Learn Res*. 2005;6:1345–1382.
38. Arnoult D, Gaume B, Karbowski M, et al. Mitochondrial release of AIF and EndoG requires caspase activation downstream of Bax/Bak-mediated permeabilization. *EMBO J*. 2003;22:4385–4399.
39. Ferreira K, Kreutz C, Macnelly S, et al. Caspase-3 feeds back on Caspase-8, Bid and XIAP in type I Fas signaling in primary mouse hepatocytes. *Apoptosis*. 2012;17:503–515.
40. Pochampally R, Fodera B, Chen L, et al. A 60 kd MDM2 isoform is produced by caspase cleavage in non-apoptotic tumor cells. *Oncogene*. 1998;17:2629–2636.
41. De Vita J. Niche separation and the Broken-Stick model. *Am Naturalist*. 1979; 114:171–178.