The Annals of Applied Statistics
2019, Vol. 13, No. 2, 1043–1067
https://doi.org/10.1214/18-AOAS1227

ADAPTIVE GPCA: A METHOD FOR STRUCTURED DIMENSIONALITY REDUCTION WITH APPLICATIONS TO MICROBIOME DATA

BY JULIA FUKUYAMA[1]

Indiana University

Exploratory analysis is an important first step for discovering latent structure and generating hypotheses in large biological data sets. However, when the number of variables is large compared to the number of samples, standard methods such as principal components analysis give results that are unstable and difficult to interpret.

Here, we present adaptive generalized principal components analysis (adaptive gPCA), a new method that solves these problems by incorporating information about the relationships among the variables. Adaptive gPCA gives a low-dimensional representation of the samples with axes that are interpretable in terms of groups of closely related variables. We show that adaptive gPCA does well at reconstructing true latent structure in simulated data and demonstrate its use on a study of the effect of antibiotics on the human gut microbiota.

1. Introduction. Biological data matrices often come with side information about the relationships among the variables. Two examples are microbiome datasets, which contain bacterial abundances plus information about the phylogenetic relationships among the bacteria, and transcriptomic datasets which often include gene expression levels plus information about gene interactions. In light of this, many methods have been developed to perform statistical analyses while taking into account variable structure. In the supervised setting we have the fused lasso for linearly-structured variables such as those in genomic datasets (Tibshirani and Wang (2008), Tibshirani et al. (2005), Rinaldo (2009)) and kernel-penalized regression (Randolph et al. (2018)) for tree-structured variables in microbiome data. The structure encoded by gene networks has also been used to aid in classification of microarray data (Rapaport et al. (2007)), regression analysis of genomic data (Li and Li (2008)) and understand the differences between experimental conditions or biological states as in gene set enrichment analysis (Subramanian et al. (2005)).

Fewer methods are available in the unsupervised setting, but some examples are double principal coordinates analysis (DPCoA), (Pavoine, Dufour and Chessel (2004)), fused-lasso penalized principal components analysis (PCA) (Witten,

Received April 2018; revised October 2018.

[1]Supported in part by a Stanford Interdisciplinary Graduate Fellowship and the Stanford Bio-X Fellowship.

Key words and phrases. PCA, microbiome, phylogeny, empirical Bayes.

Tibshirani and Hastie (2009)), weighted and unweighted Unifrac (Lozupone and Knight (2005), Lozupone et al. (2007)), a number of Unifrac variants (Chen et al. (2012), Chang, Luan and Sun (2011)) and edge PCA (Matsen and Evans (2013)). Aside from DPCoA, which can accommodate general variable structures, each of these methods is tailored to a certain type of structure, either linearly ordered (fused-lasso penalized PCA) or structured according to a phylogenetic tree (the Unifrac variants and edge PCA). Many of them are also distance based, limiting the interpretability of the results when they are used for dimension reduction.

Adaptive generalized principal components analysis (adaptive gPCA) allows the variable structure to be incorporated at either a fine or coarse scale, applies to general types of structure on the variables and is interpretable. It encourages similar variables to have similar loadings on the principal axes, but it adapts to the data instead of using a fixed level of similarity. It is motivated by a probabilistic model, making it flexible and extensible to other noise structures.

In Section 2 we introduce a motivating example in which the side information is particularly important. Section 3 provides a review of generalized PCA, and Section 4 introduces adaptive gPCA. Section 5 describes some properties and extensions of adaptive gPCA, and Sections 6 and 7 show the performance of adaptive gPCA on simulated and real data.

2. Motivating example.

2.1. *Overview.* The motivation for this method comes from our work with microbiome data. In a standard microbiome experiment the investigator sequences the variable segment of a gene that is present in all bacteria and uses clusters of similar sequences as a proxy for species. The taxa defined in this way are known as operational taxonomic units (OTUs). Because the sequences on their own are not informative, they are placed on a reference phylogenetic tree that describes how the sequences are related to each other and to known bacterial species. The result of a microbiome experiment is therefore a table containing OTU abundances in each of the samples along with a phylogenetic tree describing the OTUs, and our goal is to analyze the OTU abundances in light of the phylogenetic relationships.

2.2. *Bacterial species problem.* Defining OTUs as sequence clusters can seem arbitrary (how do you decide how big the clusters should be?), but it is related to a debate among microbiologists about whether bacterial species reflect real biological groupings or not. The prospecies side has theoretical justifications for why we would expect to see biologically meaningful groups of bacteria with smaller within- than between-group sequence divergence (Cohan (2002)), while the antispecies side cites the large amount of lateral gene transfer and homologous recombination as well as the amount of genetic dissimilarity within groups traditionally defined as species (Doolittle and Papke (2006)).

Although microbiologists may differ on the existence of bacterial species, they agree on the usefulness of the phylogenetic tree for describing the relationships between bacteria; see, for example, Brenner, Staley and Krieg (2005), Doolittle and Papke (2006), Cohan (2002). Therefore, the method we use to analyze these data should incorporate the phylogeny instead of (implicitly) assuming that the taxa are all equally distinct. Doing so allows us to worry less about using an arbitrary sequence similarity cutoff to define taxa and brings the analysis more in line with biologists' understanding of bacterial diversity.

2.3. *Antibiotic dataset*. We focus on a study of the effect of antibiotics on the composition of the human gut microbiome, described in Dethlefsen and Relman (2011). In this study 162 stool samples were collected from three individuals before, during and after administration of two courses of the antibiotic Ciprofloxacin. OTUs were defined by clustering together sequences with at least 95% sequence identity using the Uclust software (Edgar (2010)), leading to 2582 OTUs. The abundance of each OTU was defined as the number of sequences mapping to the cluster. After defining OTUs, the consensus sequence for each OTU was mapped to a reference phylogenetic tree from the Silva 100 reference database (Quast et al. (2013)) giving the phylogenetic relationships among the OTUs.

In addition to the theoretical reasons for using the phylogeny when analyzing this dataset, there are some study-specific reasons why we would expect incorporating the phylogeny to help our analysis. We have 2582 variables (the abundances of the OTUs) and only 162 samples, making the variable loadings from PCA difficult to interpret and unreliable (Johnstone and Lu (2009)). We do not expect sparsity in the principal axes because the main divisions in the samples (differences between individuals and differences due to the antibiotic) should be associated with changes in abundances of a large number of species. Therefore, we don't want to use a sparsity penalty to regularize PCA. On the other hand we do expect phylogenetically similar taxa to react in similar ways to the antibiotic, and so regularizing using the phylogeny should help us understand the effect of the antibiotic.

3. Background: Generalized PCA. Before we introduce adaptive gPCA we review generalized principal components analysis (gPCA). We follow the notation from the French multivariate tradition (Holmes (2008)) in considering gPCA on a triple $(\mathbf{X}, \mathbf{Q}, \mathbf{D})$, where $\mathbf{X} \in \mathbb{R}^{n \times p}$ is our data matrix of n samples measured on p variables, and $\mathbf{Q} \in \mathbb{R}^{p \times p}$ and $\mathbf{D} \in \mathbb{R}^{n \times n}$ are positive definite matrices. The sample scores on the top k gPCA axes of the triple $(\mathbf{X}, \mathbf{Q}, \mathbf{D})$ are the solutions to the optimization problem

$$
\begin{aligned}
\max_{\boldsymbol{u}_i \in \mathbb{R}^n} \quad & \boldsymbol{u}_i^T \mathbf{D}\mathbf{X}\mathbf{Q}\mathbf{X}^T\mathbf{D}\boldsymbol{u}_i, \quad i = 1, \ldots, k \\
\text{s.t.} \quad & \boldsymbol{u}_i^T \mathbf{D}\boldsymbol{u}_i = 1, \quad i = 1, \ldots, k, \\
& \boldsymbol{u}_i^T \mathbf{D}\boldsymbol{u}_j = 0, \quad 1 \le i < j \le k.
\end{aligned} \tag{1}
$$

Similarly, the principal axes for gPCA on the triple $(\mathbf{X}, \mathbf{Q}, \mathbf{D})$ are given by

$$\begin{aligned} &\max_{\boldsymbol{v}_i \in \mathbb{R}^p} && \boldsymbol{v}_i^T \mathbf{Q}\mathbf{X}^T\mathbf{D}\mathbf{X}\mathbf{Q}\boldsymbol{v}_i, \quad i = 1, \ldots, k \\ &\text{s.t.} && \boldsymbol{v}_i^T \mathbf{Q}\boldsymbol{v}_i = 1, \quad i = 1, \ldots, k, \\ &&& \boldsymbol{v}_i^T \mathbf{Q}\boldsymbol{v}_j = 0, \quad 1 \le i < j \le k. \end{aligned} \tag{2}$$

Generalized PCA can be interpreted as PCA in a nonstandard inner product space or as PCA on observations corrupted with nonspherical noise, and we give both interpretations below.

3.1. *Nonspherical noise*. Following Caussinus (1986), recall that PCA can be formulated as a maximum likelihood problem. Suppose that our observed data matrix is $\mathbf{X} \in \mathbb{R}^{n \times p}$, and our model is

$$\mathbf{X} \sim \mathcal{MN}_{n \times p}\big(\mathbf{U}\Lambda\mathbf{V}^T, \mathbf{D}^{-1}, \mathbf{Q}^{-1}\big), \tag{3}$$

where $\mathbf{U} \in \mathbb{R}^{n \times k}$ and $\mathbf{V} \in \mathbb{R}^{p \times k}$ are orthogonal, $\mathbf{\Lambda}$ is diagonal and $\mathbf{D} \in \mathbb{R}^{n \times n}$ and $\mathbf{Q} \in \mathbb{R}^{p \times p}$ are both positive definite. $\mathcal{MN}_{n \times p}$ denotes a matrix normal distribution. A random matrix $\mathbf{X}$ follows the matrix normal distribution $\mathcal{MN}_{n \times p}(\mathbf{M}, \mathbf{\Sigma}_1, \Sigma_2)$ if and only if

$$\operatorname{vec}(\mathbf{X}) \sim \mathcal{N}_{np}\big(\operatorname{vec}(\mathbf{M}), \mathbf{\Sigma}_1 \otimes \mathbf{\Sigma}_2\big),$$

where vec denotes the vectorization function and $\otimes$ represents the Kronecker product.

Under the model in (3), if the row scores, principal axes and variances of gPCA on the triple $(\mathbf{X}, \mathbf{Q}, \mathbf{D})$ are given by $\hat{\mathbf{U}}$, $\hat{\mathbf{V}}$ and $\hat{\mathbf{\Lambda}}$, then the maximum likelihood estimate of $\mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$ is $\hat{\mathbf{U}}\hat{\mathbf{\Lambda}}\hat{\mathbf{V}}^T$. The matrix normal distribution allows us to account for more complicated error structures than the i.i.d. model; we can have correlation on the rows, on the columns or both. When $\mathbf{Q} = \mathbf{I}_p$ and $\mathbf{D} = \mathbf{I}_n$, the errors are i.i.d. $\mathcal{N}(0, 1)$, and we recover standard PCA.

3.2. *Nonstandard inner product*. The other interpretation of gPCA on the triple $(\mathbf{X}, \mathbf{Q}, \mathbf{D})$ is as PCA in a nonstandard inner product space. We can use $\mathbf{Q}$ and $\mathbf{D}$ to define inner products on $\mathbb{R}^p$ and $\mathbb{R}^n$ as follows:

$$\begin{aligned} \langle \boldsymbol{x}, \boldsymbol{y} \rangle_{\mathbf{Q}} &= \boldsymbol{x}^T\mathbf{Q}\boldsymbol{y}, \quad \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^p, \\ \langle \boldsymbol{x}, \boldsymbol{y} \rangle_{\mathbf{D}} &= \boldsymbol{x}^T\mathbf{D}\boldsymbol{y}, \quad \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n. \end{aligned}$$

From the form of the gPCA problem as shown in (1) and (2), we see that gPCA is simply standard PCA with the standard inner product replaced with the $\mathbf{Q}$- and $\mathbf{D}$-inner product for the rows and columns respectively. In particular gPCA of the triple $(\mathbf{X}, \mathbf{I}_p, \mathbf{I}_n)$ is equivalent to standard PCA. The Supplementary Material (Fukuyama (2019)) gives some intuition into how to interpret these inner product spaces.

3.3. *Fraction of variance explained.* In gPCA of $(\mathbf{X}, \mathbf{Q}, \mathbf{D})$, the fraction of the variance explained by the top generalized principal components is reported in relation to the $\mathbf{Q}$- or $\mathbf{D}$-inner products. The formula is analogous to the fraction of the variance explained by the top principal components in PCA. If $\mathbf{X}_k = \mathbf{U}_k \mathbf{\Lambda}_k \mathbf{V}_k^T$ gives the gPCA approximation to $\mathbf{X}$, then the fraction of the variance explained by the top k generalized principal components is

$$\operatorname{tr}(\mathbf{D}\mathbf{X}_k\mathbf{Q}\mathbf{X}_k^T)/\operatorname{tr}(\mathbf{D}\mathbf{X}\mathbf{Q}\mathbf{X}^T). \tag{4}$$

4. New method: Adaptive gPCA. The idea behind adaptive gPCA is to put a prior on the data encoding our intuition that similar variables have similar behaviors; that is, for microbiome data species close together in the phylogenetic tree respond in the same way to environmental perturbations. We then use gPCA to obtain a low-dimensional representation of the posterior estimates in this model. This leads to a structured version of PCA in which similar variables are encouraged to have similar loadings. The strength of the prior is a tunable parameter, with stronger priors corresponding to more globally structured solutions and weaker priors corresponding to locally structured solutions. The strength of the prior is chosen automatically by maximum marginal likelihood.

4.1. *Data model.* Suppose we have the following model for our data matrix $\mathbf{X}$, which we assume is centered:

$$\boldsymbol{\mu}_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}_p, \sigma_1^2\mathbf{Q}), \quad i = 1, \ldots, n, \tag{5}$$

$$\boldsymbol{x}_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\boldsymbol{\mu}_i, \sigma_2^2\mathbf{I}_p), \quad i = 1, \ldots, n, \tag{6}$$

where $\boldsymbol{\mu}_i \in \mathbb{R}^p$, $\boldsymbol{x}_i \in \mathbb{R}^p$ is the ith row of $\mathbf{X}$ written as a column vector, and $\mathbf{Q} \in \mathbb{R}^{p\times p}$ is a positive definite matrix. If we choose $\mathbf{Q}$ to be a kernel matrix describing the similarities between the variables, this model incorporates our prior knowledge about the structure of the variables through $\mathbf{Q}$, and elements of $\boldsymbol{\mu}_i$ corresponding to similar variables will be positively correlated. We assume that $\mathbf{Q}$ is full rank, but the same analysis can be performed using a rank degenerate $\mathbf{Q}$ by replacing $\mathbf{Q}^{-1}$ with $\mathbf{Q}^+$, the Moore–Penrose pseudoinverse of $\mathbf{Q}$ (Penrose (1955)).

We are interested in the "true" values $\boldsymbol{\mu}_i$, not the observed data $\boldsymbol{x}_i$; so we compute the posterior distribution of the $\boldsymbol{\mu}_i$'s. Bayes' rule gives

$$\boldsymbol{\mu}_i \mid \boldsymbol{x}_i = \boldsymbol{x} \sim \mathcal{N}_p(\sigma_2^{-2}\mathbf{S}_{\sigma_1,\sigma_2}\boldsymbol{x}, \mathbf{S}_{\sigma_1,\sigma_2}), \tag{7}$$

with

$$\mathbf{S}_{\sigma_1,\sigma_2} = (\sigma_1^{-2}\mathbf{Q}^{-1} + \sigma_2^{-2}\mathbf{I}_p)^{-1}. \tag{8}$$

We then want a low-dimensional representation of the posteriors $\boldsymbol{\mu}_i \mid \boldsymbol{x}_i$. To do this properly, we need to take into account the nonspherical posterior variance $\mathbf{S}_{\sigma_1,\sigma_2}$ and decide what values to use for σ_1 and σ_2.

4.2. *Generalized PCA on $\mu_i \mid x_i$.* The posterior distributions $\boldsymbol{\mu}_i \mid \boldsymbol{x}_i$ have nonspherical variance, and so we should use gPCA and not standard PCA to get a low-dimensional representation of the posteriors. In particular by performing gPCA on the triple $(\mathbf{X}\mathbf{S}_{\sigma_1,\sigma_2}, \mathbf{S}_{\sigma_1,\sigma_2}^{-1}, \mathbf{I}_n)$, we obtain a low-dimensional representation of the posterior means (the rows of $\mathbf{X}\mathbf{S}_{\sigma_1,\sigma_2}$), taking into account that they have variance $\mathbf{S}_{\sigma_1,\sigma_2}$.

The triple $(\mathbf{X}\mathbf{S}_{\sigma_1,\sigma_2}, \mathbf{S}_{\sigma_1,\sigma_2}^{-1}, \mathbf{I}_n)$ can be simplified to the triple $(\mathbf{X}, \mathbf{S}_{\sigma_1,\sigma_2}, \mathbf{I}_n)$ (Theorem 1 below). Since we do not think $\mathbf{X}$ has row covariance $\mathbf{S}_{\sigma_1,\sigma_2}^{-1}$, the structured error interpretation no longer applies, and we should think of this gPCA as PCA in a nonstandard inner product space.

THEOREM 1. *The row scores from gPCA on the posterior estimates $\boldsymbol{\mu}_i \mid \boldsymbol{x}_i$ from the model given by equations (5)–(6) are the same, up to a scaling factor, as the row scores from gPCA on $(\mathbf{X}, \mathbf{S}_{\sigma_1,\sigma_2}, \mathbf{I}_n)$. The principal axes from gPCA on the posterior estimates are the same, up to a scaling factor, as the principal axes from gPCA on $(\mathbf{X}, \mathbf{S}_{\sigma_1,\sigma_2}, \mathbf{I}_n)$ premultiplied by $\mathbf{S}_{\sigma_1,\sigma_2}$.*

PROOF. See the Supplementary Material (Fukuyama (2019)). □

Theorem 1 shows that when we perform gPCA on the posteriors obtained from the model (5)–(6), different scalings of the prior and the noise variances lead to gPCAs on the same data matrix with different row inner products.

4.3. *Selection of σ_1 and σ_2.* Our second task is to choose σ_1 and σ_2. They can be estimated using the model described by equations (5)–(6). The marginal distribution of $\boldsymbol{x}_i$ in that model is

$$\boldsymbol{x}_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}_p, \sigma_1^2\mathbf{Q} + \sigma_2^2\mathbf{I}_p). \tag{9}$$

Up to a constant factor, the overall log likelihood of the data is therefore

$$\ell(\mathbf{X}; \sigma_1, \sigma_2) = -\frac{n}{2}\log|\sigma_1^2\mathbf{Q} + \sigma_2^2\mathbf{I}_p| - \sum_{i=1}^{n}\frac{1}{2}\boldsymbol{x}_i^T\left(\sigma_1^2\mathbf{Q} + \sigma_2^2\mathbf{I}_p\right)^{-1}\boldsymbol{x}_i, \tag{10}$$

and we can choose σ_1 and σ_2 to maximize $\ell(\mathbf{X}; \sigma_1, \sigma_2)$.

This likelihood is not convex, and there is no closed-form solution for the maximum, but we can transform it into one-dimensional optimization on the unit interval. Let

$$r = \sigma_1^2/(\sigma_1^2 + \sigma_2^2), \qquad \sigma^2 = \sigma_1^2 + \sigma_2^2. \tag{11}$$

Let $\mathbf{Q} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T$ be the eigendecomposition of $\mathbf{Q}$ where $\mathbf{V}$ is an orthogonal matrix and $\boldsymbol{\Lambda}$ is diagonal containing the eigenvalues $\lambda_1, \ldots, \lambda_p$. Finally, let $\tilde{\boldsymbol{x}}_i = \mathbf{V}^T\boldsymbol{x}_i$

and $\tilde{x}_{ij}$ be the jth element of $\tilde{\boldsymbol{x}}_i$. The log likelihood in the new parameterization is

$$\begin{aligned}\ell(\mathbf{X}; r, \sigma) &= -\frac{np}{2}\sigma^2 \log|r\mathbf{Q} + (1-r)\mathbf{I}_p| \\ &\quad - \sigma^{-2}\sum_{i=1}^{n}\frac{1}{2}\boldsymbol{x}_i^T\big(r\mathbf{Q} + (1-r)\mathbf{I}_p\big)\boldsymbol{x}_i \end{aligned} \tag{12}$$

$$\begin{aligned}&= -\frac{np}{2}\sigma^2 \sum_{j=1}^{p}\log(r\lambda_j + 1 - r) \\ &\quad - \sigma^{-2}\sum_{i=1}^{n}\sum_{j=1}^{p}\frac{1}{2}\frac{\tilde{x}_{ij}^2}{r\lambda_j + 1 - r}.\end{aligned} \tag{13}$$

Based on the expression above, we can find a closed-form solution for the maximizing value of σ^2 for any fixed r:

$$\sigma^{2^*}(r) = \frac{1}{np}\sum_{i=1}^{n}\sum_{j=1}^{p}\tilde{x}_{ij}^2/(r\lambda_i + 1 - r). \tag{14}$$

We can then rewrite the likelihood as a function of r only. It is still nonconvex and lacks a closed-form solution, but, since we have a single parameter in the unit interval, the optimization can easily be performed numerically.

4.4. *Summary.* Putting everything together, adaptive gPCA is the following procedure:

1. Let $\mathbf{X} \in \mathbb{R}^{n\times p}$ be the centered data matrix, and let $\mathbf{Q} \in \mathbb{R}^{p\times p}$ be a kernel matrix containing the similarities between the variables.
2. Find $\hat{\sigma}_1$ and $\hat{\sigma}_2$ that maximize the likelihood function in equation (10).
3. Let $\mathbf{S} = (\hat{\sigma}_1^{-2}\mathbf{Q}^{-1} + \hat{\sigma}_2^{-2}\mathbf{I}_p)^{-1}$, and perform gPCA on the triple $(\mathbf{X}, \mathbf{S}, \mathbf{I})$.

The sample scores for adaptive gPCA are given by the row scores of this gPCA, and the variable scores for adaptive gPCA are given by the column scores of this gPCA premultiplied by $\mathbf{S}$.

This gives us a structured version of PCA in which similar variables have similar loadings on the principal axes. In the next section we explain what this structure looks like and why it occurs.

5. Properties and extensions of adaptive gPCA. In this section we describe some properties of adaptive gPCA, show how it can be extended to accommodate other noise structures or uncertainty in the structure of the variables and describe its relationship with existing methods.

5.1. *Properties*.

5.1.1. *Global vs. local structure*. To describe the adaptive gPCA solutions, we introduce the concepts of global and local structure. If a solution reflects the global structure of the variables, the distances between dissimilar variables are preserved, that is, sets of variables with very dissimilar loadings on the principal axes will also be very dissimilar according to our prior definition of similarity on the variables. If a solution reflects the local structure of the variables, the distances between similar variables are preserved.

In adaptive gPCA we use an inner product from the family $\mathbf{S}_{\sigma_1,\sigma_2}$. The family of inner product matrices $\mathbf{S}_{\sigma_1,\sigma_2}$ nominally has two parameters, but modulo a scaling factor only the relative sizes of σ_1 and σ_2 matter, and we can think of it as one dimensional. The endpoints of this family are obtained as $\sigma_1/\sigma_2 \to 0$ or as $\sigma_1/\sigma_2 \to \infty$. As $\sigma_1/\sigma_2 \to 0$, $\sigma_1^{-2}\mathbf{S}_{\sigma_1,\sigma_2} \to \mathbf{Q}$, and as $\sigma_1/\sigma_2 \to \infty$, $\sigma_2^{-2}\mathbf{S}_{\sigma_1,\sigma_2} \to \mathbf{I}_p$. Using $\mathbf{Q}$ as a row inner product gives the most globally structured solutions, using $\mathbf{I}_p$ gives the most locally structured solutions and using $\mathbf{S}_{\sigma_1,\sigma_2}$ in between those two extremes gives solutions with an intermediate type of structure.

The variable loadings from gPCA on the antibiotic dataset with some row inner products from the $\mathbf{S}_{\sigma_1,\sigma_2}$ family are illustrated in Figure 1(A). In these plots each point represents a variable (for this dataset, a bacterial taxon), and the points are colored by phylum. We see that in gPCA on $(\mathbf{X}, \mathbf{Q}, \mathbf{I}_n)$, the taxa from different phyla load in disjoint regions of the principal plane. This corresponds to the taxa loadings respecting the global structure: taxa from different phyla are very dissimilar from each other, and so they are required to have very dissimilar loadings on the principal axes.

As we decrease σ_2 and/or increase σ_1, the local structure is preserved, but the global structure is lost. In particular, as σ_1 increases compared to σ_2, closely related taxa continue to have similar loadings on the principal axes, but taxa from different phyla are no longer on opposite halves of the principal plane. The smaller amount of emphasis on the global structure means that the variable loadings continue to be structured phylogenetically, but at finer scales.

The most locally structured endpoint is obtained as $\sigma_1/\sigma_2 \to \infty$, that is, as the row inner product approaches $\mathbf{I}_p$. As we see in the facet labeled $\mathbf{I}_p$ in Figure 1(A), when we use this inner product we see no relationship between the phylogeny and the taxa loadings. This is "locally" structured in the sense that only the zero distances between each variable and itself are preserved.

5.1.2. *Relationship with double principal coordinates analysis*. The family of row inner product matrices $\mathbf{S}_{\sigma_1,\sigma_2}$ bridges the gap between standard PCA and double principal coordinates analysis (DPCoA), (Pavoine, Dufour and Chessel (2004)), a method for incorporating information on the structure of the variables. DPCoA creates a low-dimensional representation of ecological count data (the abundances of species at different locations) taking into account information about

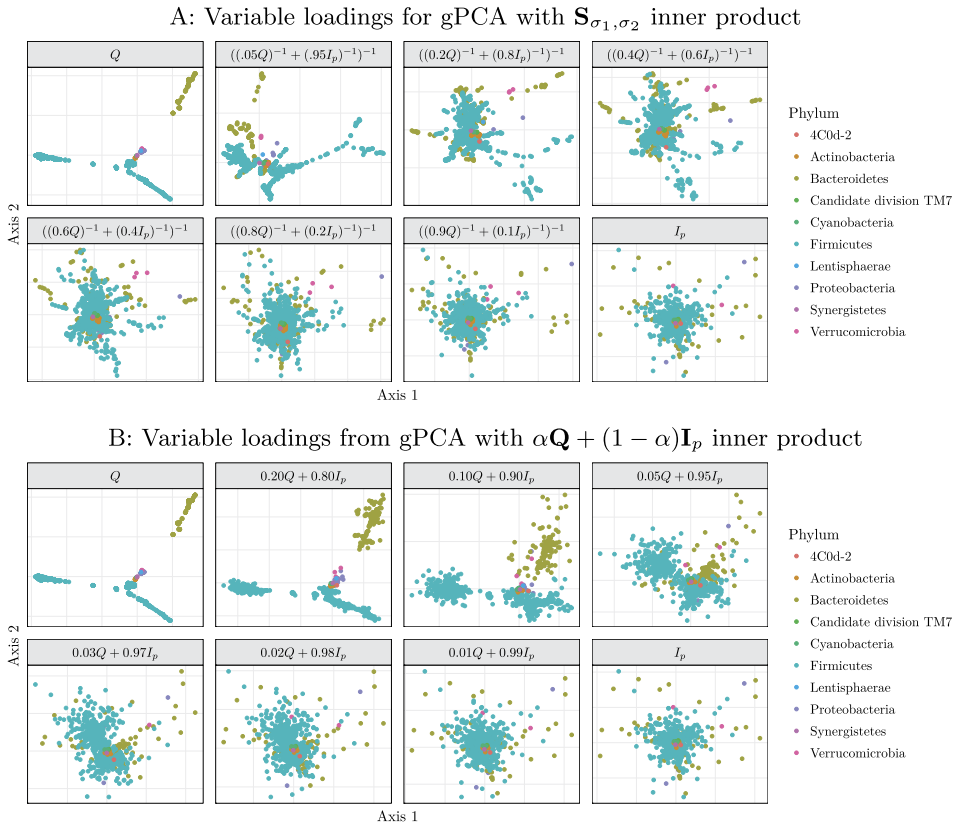

FIG. 1. *Plots of the variable loadings from gPCA on* $(\mathbf{X}, \cdot, \mathbf{I}_n)$, *where* $\mathbf{X}$ *is a data matrix taken from the antibiotic study. Top panel* (A) *uses row inner products from the* $\mathbf{S}_{\sigma_1,\sigma_2}$ *family, bottom panel* (B) *uses row inner products from the* $\alpha\mathbf{Q}+(1-\alpha)\mathbf{I}_p$ *family. The inner product matrix used is given in the facet label.*

the similarities between species. DPCoA takes as input a matrix of Euclidean distances between the species and a matrix giving the abundance of each species at each sampling site. It consists of the following steps:

1. Perform a full multidimensional scaling on the similarities between species.
2. Place each sampling site at the center of mass of the species vector corresponding to that site.
3. Perform PCA on the matrix of sampling site coordinates and project both the sampling site points and the species points onto the PCA axes.

Purdom (2011) showed that DPCoA is equivalent to gPCA using a certain nonstandard inner product for the special case of tree-structured variables, and the result can be extended to any Euclidean distance structure on the variables. For general Euclidean distances on the variables, we have the following:

THEOREM 2. *Suppose we have a count matrix* $\mathbf{X} \in \mathbb{R}^{n\times p}$ *and a set of Euclidean distances between the* p *variables. We construct a matrix* $\boldsymbol{\delta} \in \mathbb{R}^{p\times p}$ *containing the squares of the distances between the variables. Let* $\boldsymbol{w}_L = \mathbf{X}\mathbf{1}/\mathbf{1}^T\mathbf{X}\mathbf{1}$, $\boldsymbol{w}_S = \mathbf{X}^T\mathbf{1}/\mathbf{1}^T\mathbf{X}\mathbf{1}$, *and for any weight vector* $\boldsymbol{w}$ *let* $\mathbf{P}_{\boldsymbol{w}} = \mathbf{I} - \mathbf{1}\boldsymbol{w}^T$ *and* $\mathbf{D}_w$ *denote the diagonal matrix with* $\boldsymbol{w}$ *on the diagonal. Then*:

1. *The row scores from DPCoA on* $\mathbf{X}$ *using the distances implied by* $\boldsymbol{\delta}$ *are the same* (*up to a sign change*) *as the row scores obtained from gPCA on* $(\mathbf{D}_{\boldsymbol{w}_L}^{-1}\mathbf{X}\mathbf{P}_{\boldsymbol{w}_S}, \mathbf{P}_{\boldsymbol{w}_S}(-\boldsymbol{\delta}/2)\mathbf{P}_{\boldsymbol{w}_S}, \mathbf{D}_{\boldsymbol{w}_L})$.
2. *If the column scores from gPCA on* $(\mathbf{D}_{\boldsymbol{w}_L}^{-1}\mathbf{X}\mathbf{P}_{\boldsymbol{w}_S}, \mathbf{P}_{\boldsymbol{w}_S}(-\boldsymbol{\delta}/2)\mathbf{P}_{\boldsymbol{w}_S}, \mathbf{D}_{w_L})$ *are given by* $\mathbf{Z}$, *then the column scores from DPCoA on* $\mathbf{X}$ *using the distances implied by* $\boldsymbol{\delta}$ *are the same* (*up to a sign change*) *as* $\boldsymbol{P}_{w_S}(-\boldsymbol{\delta}/2)\mathbf{P}_{w_S}\mathbf{Z}$.

PROOF. See the Supplementary Material (Fukuyama (2019)). □

DPCoA was designed for count data, and it implicitly transforms the counts to relative abundances and retains the row and column sums as weights providing information on how accurately the samples were measured (this procedure is standard in French multivariate analysis of count data; see, e.g., the section on correspondence analysis in Holmes (2008)). Therefore, the gPCA formulation of DPCoA uses weighted centering matrices ($\mathbf{P}_{\boldsymbol{w}_S}$) and a column inner product that weights the rows by their counts ($\mathbf{D}_{\boldsymbol{w}_L}$). In the more general setup we do not have measures of the precision for the variables or the samples, and so the natural generalization of DPCoA to noncount data would be to weight all the variables equally. With this modification the DPCoA triple becomes $(\mathbf{X}\mathbf{P}, \mathbf{P}(-\boldsymbol{\delta}/2)\mathbf{P}, \mathbf{I}_n)$, where $\mathbf{P} = \mathbf{I}_p - \mathbf{1}_p\mathbf{1}_p^T/p$ is a centering matrix. The row inner product matrix here is equal to $\lim_{\sigma_1/\sigma_2\to 0}\sigma_1^{-2}\mathbf{S}_{\sigma_1,\sigma_2}$, and the data matrix is simply a standard centered data matrix. Thus, we see that a slight generalization of DPCoA is the same as gPCA using the globally structured of the endpoint in the $\mathbf{S}_{\sigma_1,\sigma_2}$ family of row inner products.

5.1.3. *Comparison with another family of inner product matrices.* We just showed that the family of inner products defined by the model in Section 4.1 bridges the gap between DPCoA and PCA. This might lead us to ask about other families of inner products with $\mathbf{Q}$ and $\mathbf{I}$ as endpoints. In particular another way to interpolate between an inner product matrix $\mathbf{Q}$ and the standard inner product $\mathbf{I}_p$ is to use the family $\alpha\mathbf{Q} + (1-\alpha)\mathbf{I}_p$ with $0 \le \alpha \le 1$. As with $\mathbf{S}_{\sigma_1,\sigma_2}$, the endpoints are $\mathbf{I}_p$ and $\mathbf{Q}$, but path in between is different. The variable loadings on the principal axes from gPCA using members of this family are plotted in Figure 1(B). Unlike in the $\mathbf{S}_{\sigma_1,\sigma_2}$ family, as we move from $\mathbf{Q}$ to $\mathbf{I}_p$ the local structure is lost while the global structure is preserved. The taxa loadings using the $\alpha\mathbf{Q} + (1-\alpha)\mathbf{I}_p$ inner product look like a noisy version of the taxa loadings using the $\mathbf{Q}$ inner product. This is undesirable because if the axes given by the gPCA on $(\mathbf{X}, \mathbf{Q}, \mathbf{I}_n)$

were not useful, a noisy version of these axes is unlikely to be much better. We gain very little at the cost of losing the local structure and interpretability of the variables. This behavior is discussed further in Section 3 of the Supplementary Material (Fukuyama (2019)).

5.1.4. *Relationship with factor analysis.* The model in (5)–(6) used to define adaptive gPCA gives the same marginal likelihood for the data as the following confirmatory factor analysis (CFA) model:

$$\mathbf{u}_i \sim \mathcal{N}(\mathbf{0}_p, \sigma_1^2 \mathbf{I}_p), \quad i = 1, \ldots, n, \tag{15}$$

$$\mathbf{x}_i \mid \mathbf{u}_i \sim \mathcal{N}(\mathbf{F}\mathbf{u}_i, \sigma_2^2 \mathbf{I}_p), \quad i = 1, \ldots, n, \tag{16}$$

where the columns of $\mathbf{F} \in \mathbb{R}^{p \times p}$ are the prespecified latent factors, defined as $\mathbf{F} = \mathbf{V}\mathbf{D}^{1/2}$, where as before $\mathbf{V} \in \mathbb{R}^{p \times p}$ has as columns the eigenvectors of $\mathbf{Q}$, $\mathbf{D} \in \mathbb{R}^{p \times p}$ is a diagonal matrix with diagonal elements corresponding to the eigenvalues of $\mathbf{Q}$, $\mathbf{x}_i \in \mathbb{R}^p$ is the ith row of $\mathbf{X}$, written as a column vector, and $\mathbf{u}_i \in \mathbb{R}^p$ gives the scores of the ith sample on each of the p latent factors.

In model (15)–(16) we can obtain posterior estimates of the $\mathbf{u}_i$'s, the sample scores along the fixed latent factors, and it is natural to ask about the relationship between the posterior estimates in the factor analysis model and the sample scores in adaptive gPCA. The posterior means of the sample scores in the CFA model turn out to be a linear transformation of the posterior means found in equation (7), specifically, $\mathbf{E}[\mathbf{u}_i \mid \mathbf{x}_i] = \mathbf{D}^{1/2}\mathbf{V}^T\mathbf{E}[\boldsymbol{\mu}_i|\mathbf{x}_i]$ (Theorem S1 in the Supplementary Material, Fukuyama (2019)).

Despite the equivalence between the factor analysis model and the model used to motivate adaptive gPCA, the factor analysis interpretation does not help us in our initial task of obtaining a low-dimensional representation of the samples. In the model defined in equations (15)–(16), we have p latent factors, and so the sample scores along the latent factors have the same dimensionality as the raw data.

If we wanted a low-dimensional representation of the samples based on a CFA model, we could imagine two strategies. We could take only the sample scores along the top k latent factors, or we could modify the model so that it only incorporates k latent factors, that is, use the confirmatory factor analysis model

$$\mathbf{u}_i \sim \mathcal{N}(\mathbf{0}_k, \sigma_1^2 \mathbf{I}_k), \quad i = 1, \ldots, n, \tag{17}$$

$$\mathbf{x}_i \mid \mathbf{u}_i \sim \mathcal{N}(\mathbf{F}_{(k)}\mathbf{u}_i, \sigma_2^2 \mathbf{I}_p), \quad i = 1, \ldots, n, \tag{18}$$

where $\mathbf{F}_{(k)} = \mathbf{V}_{(k)}\mathbf{D}_{(k)}^{1/2}$, $\mathbf{V}_{(k)} \in \mathbb{R}^{p \times k}$ has as columns the top k eigenvectors of $\mathbf{Q}$, $\mathbf{D}_{(k)} \in \mathbb{R}^{k \times k}$ is a diagonal matrix with diagonal elements corresponding to the top k eigenvalues of $\mathbf{Q}$, $\mathbf{u}_i \in \mathbb{R}^k$ gives the scores of the ith sample on each of the k latent factors, and $\mathbf{x}_i$ is as before.

These two dimension-reduction strategies give the same k-dimensional representations of the samples for any value of k. The posterior means of the sample

scores estimated in model (17)–(18) are the same as the posterior means of the sample scores estimated in model (15)–(16), restricted to the first k factors, and that representation is simply a projection onto the top k eigenvectors of $\mathbf{Q}$ followed by a rescaling (Theorem S1 in the Supplementary Material, Fukuyama (2019)). The scores obtained via these two strategies are not the same as the adaptive gPCA sample scores, and the space the samples are projected onto does not depend on the data matrix $\mathbf{X}$. The two CFA-based strategies give to low-dimensional representations similar to those given by DPCoA, as shown in Figure S3 and Section 4 of the Supplementary Material (Fukuyama (2019)).

There is, however, a way to obtain the adaptive gPCA sample scores and principal axes from the p-factor CFA model defined in equations (15)–(16). If we take the sample scores from the full-dimensional CFA model and perform a weighted PCA, where the weight for factor j is taken to be $\sigma_1^{-2} + \sigma_2^{-2}\mathbf{D}_{jj}^{-1}$, then the resulting sample scores on the top k principal axes are the same as the sample scores in adaptive gPCA. The principal axes from this weighted PCA are linear combinations of the latent factors, and when rewritten in terms of the original variables are the same as the principal axes in adaptive gPCA (Theorem S2 in the Supplementary Material, Fukuyama (2019)). We see that here, different values of σ_1 and σ_2 lead to different weights, and in particular, when $\sigma_2 \gg \sigma_1$, the weights on the latent factors are approximately equal, whereas when $\sigma_1 \gg \sigma_2$, the latent factors corresponding to the top eigenvectors of $\mathbf{Q}$ are downweighted.

Therefore, we see that the added value of adaptive gPCA is that it uses the data to find the subspace on which to project the samples, whereas the corresponding confirmatory factor analysis model does not. In CFA, no matter what values of σ_1 and σ_2 we use, and no matter how many of the latent factors we include when estimating the covariance, the sample scores along the latent axes will always be the same (up to a scaling factor that depends on σ_1 and σ_2). In contrast in adaptive gPCA, different estimates of σ_1 and σ_2 lead to different choices principal axes and therefore different choices of the subspace on which to project the samples.

5.2. *Extensions*.

5.2.1. *Choice of kernel matrix.* $\mathbf{Q}$ can be any positive definite similarity matrix on the variables. This sort of similarity matrix is often a natural way to encode relationships between variables; for example, if the variables are the nodes in a graph, there are many graph kernels available to describe the similarities between the nodes, mostly based on the graph Laplacian (Kondor and Lafferty (2002)).

We might also start out with distances between variables instead of similarities. If these distances are Euclidean (a distance matrix on a set of p objects is called Euclidean if there exists an embedding of p points in Euclidean space such that the distances between the points match the distances provided), a natural way to create a positive definite similarity matrix is as follows: Suppose $\boldsymbol{\delta} \in \mathbb{R}^{p\times p}$ is a matrix with the squared distances between the variables, and let $\mathbf{P} = \mathbf{I}_p - \mathbf{1}_p\mathbf{1}_p^T/p$ be the

centering matrix. Then $-\mathbf{P}\boldsymbol{\delta}\mathbf{P}$ is a positive definite similarity matrix. This matrix contains the inner products between points if they are embedded in $\mathbb{R}^p$ such that the distances between them match the distances implied by $\boldsymbol{\delta}$, and they are centered around the origin.

5.2.2. *Uncertainty in the kernel matrix.* Another issue in the choice of $\mathbf{Q}$ is what to do if we have uncertainty in the structure of the variables, as we might if we use a phylogenetic tree that was itself estimated from data. Based on the model we introduced in Section 4.1, introducing uncertainty into $\mathbf{Q}$ is just a question of rewriting the model with another level:

$$\mathbf{Q} \sim \mathcal{D}, \tag{19}$$

$$\boldsymbol{\mu}_i \sim \mathcal{N}_p(\mathbf{0}_p, \sigma_1^2\mathbf{Q}), \quad i = 1, \ldots, n, \tag{20}$$

$$\boldsymbol{x}_i \sim \mathcal{N}_p(\boldsymbol{\mu}_i, \sigma_2^2\mathbf{I}_p), \quad i = 1, \ldots, n, \tag{21}$$

where $\mathcal{D}$ denotes a distribution over positive definite matrices in $\mathbb{R}^{p\times p}$ describing the uncertainty in the variable structure. Adaptive gPCA can easily be modified to work with this model. In principle we can still write the marginal likelihood of $\mathbf{X}$, and we can estimate σ_2, σ_1, and $\mathbf{Q}$ by maximum marginal likelihood. Then, instead of thinking of $\mathbf{Q}$ as fixed and performing gPCA on $(\mathbf{X}, (\hat{\sigma}_1^{-2}\mathbf{Q}^{-1} + \hat{\sigma}_2^{-2}\mathbf{I}_p)^{-1})$, we would perform gPCA on $(\mathbf{X}, (\hat{\sigma}_1^{-2}\hat{\mathbf{Q}}^{-1} + \hat{\sigma}_2^{-2}\mathbf{I}_p)^{-1}, \mathbf{I}_n)$. Whether this is computationally feasible would depend on the particular form of the distribution $\mathcal{D}$ describing $\mathbf{Q}$, but, after estimating $\mathbf{Q}$, all of the derivations would procede in the same way as for fixed $\mathbf{Q}$.

5.2.3. *Other noise structures.* Adaptive gPCA can also be modified to accommodate other noise models but with an additional computational cost. If we retain the assumption of normal noise but replace the noise covariance with a scalar multiple of $\mathbf{W}$, that is, change the model given in equations (5)–(6) to

$$\boldsymbol{\mu}_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}_p(\mathbf{0}_p, \sigma_1^2\mathbf{Q}), \quad i = 1, \ldots, n, \tag{22}$$

$$\boldsymbol{x}_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}_p(\boldsymbol{\mu}_i, \sigma_2^2\mathbf{W}), \quad i = 1, \ldots, n, \tag{23}$$

following the same argument leads to gPCA on the triple $(\mathbf{X}, (\hat{\sigma}_1^{-2}\mathbf{Q}^{-1} + \hat{\sigma}_2^{-2}\mathbf{W})^{-1}, \mathbf{I}_n)$, where $\hat{\sigma}_1$ and $\hat{\sigma}_2$ are again estimated by maximum marginal likelihood. The main difference is that when using $\mathbf{W}$ instead of $\mathbf{I}_p$ as the noise, the likelihood computations become more computationally expensive. Recall that the estimation of σ_1 and σ_2 required a numerical optimization step, searching for the best value of a parameter r. When using $\mathbf{W} \neq \mathbf{I}_p$, checking the likelihood at each value of r requires the eigendecomposition of a $p \times p$ matrix compared with just one for any number of likelihood evaluations when we use $\mathbf{W} = \mathbf{I}_p$.

6. Simulations. We evaluated the performance of adaptive gPCA on simulated datasets with varying amounts of structure on the principal axes. To match our motivating example of microbiome abundance data with information about the phylogenetic relationships between the bacteria, the variables are related to each other by a phylogenetic tree. We used a random tree (using the function `rtree` in the `ape` package (Paradis, Claude and Strimmer (2004)) in R (R Core Team (2017)), and the similarity matrix $\mathbf{Q} \in \mathbb{R}^{p \times p}$ encoding the tree structure was

$$\mathbf{Q} = \mathbf{1}\boldsymbol{s}^T + \boldsymbol{s}\mathbf{1}^T - \boldsymbol{\delta}, \tag{24}$$

where $\boldsymbol{s} \in \mathbb{R}^p$ gives the distance between each leaf node and the root and $\boldsymbol{\delta} \in \mathbb{R}^{p \times p}$ gives the distance on the tree between the leaf nodes. This definition gives $\mathbf{Q}$ with $\mathbf{Q}_{ij}$ proportional to the amount of shared ancestry between nodes i and j, and it is also equal to the covariance matrix of a Brownian motion on the phylogenetic tree.

In our simulations we compared four procedures:

1. PCA, that is, generalized PCA of the triple $(\mathbf{X}, \mathbf{I}_p, \mathbf{I}_n)$.
2. Generalized PCA of the triple $(\mathbf{X}, \mathbf{Q}, \mathbf{I}_n)$.
3. Generalized PCA of the triple $(\mathbf{X}, 0.1\mathbf{Q} + 0.9\mathbf{I}_p, \mathbf{I}_n)$.
4. Adaptive gPCA.

Generalized PCA of $(\mathbf{X}, \mathbf{Q}, \mathbf{I}_n)$ is the extension of DPCoA to real-valued data described in Section 5.1.2 and corresponds to the limit of our model when the prior dominates.

We included generalized PCA of $(\mathbf{X}, 0.1\mathbf{Q} + 0.9\mathbf{I}_p, \mathbf{I}_n)$ to compare a member of the "ridged" family described in Section 5.1.1 to adaptive gPCA. There is no simple way to choose a member of the ridged family automatically, and so we chose $0.1\mathbf{Q} + 0.9\mathbf{I}_p$ because it gave results that were qualitatively intermediate between using $\mathbf{Q}$ and using $\mathbf{I}_p$.

6.1. *Simulation A.* In the first simulation we generated our data matrix $\mathbf{X}$ as rank-one plus noise:

$$\mathbf{X} = \boldsymbol{u}\boldsymbol{v}^T + \mathbf{E}, \tag{25}$$

$$\mathbf{E}_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2), \quad i = 1, \ldots, n, \, j = 1, \ldots, p, \tag{26}$$

$$\boldsymbol{u} \sim \mathcal{N}_n(\mathbf{0}_n, \mathbf{I}_n), \tag{27}$$

$$\boldsymbol{v} \sim \mathcal{N}_p\big(\mathbf{0}_p, \mathbf{V}_{(m)}\mathbf{V}_{(m)}^T\big). \tag{28}$$

$\mathbf{V}_{(m)} \in \mathbb{R}^{p \times m}$ denotes the matrix whose columns are the top m eigenvectors of $\mathbf{Q}$. The value of m governs how smooth $\boldsymbol{v}$ is: if m is small, $\boldsymbol{v}$ has coefficients that are very smooth on the tree, and as m increases the coefficients get more and more rough. At the extreme case of $m = p$, $\mathbf{V}_{(m)}\mathbf{V}_{(m)}^T = \mathbf{I}_p$, and there is no relationship at all between the coefficients of $\boldsymbol{v}$ and the tree structure.

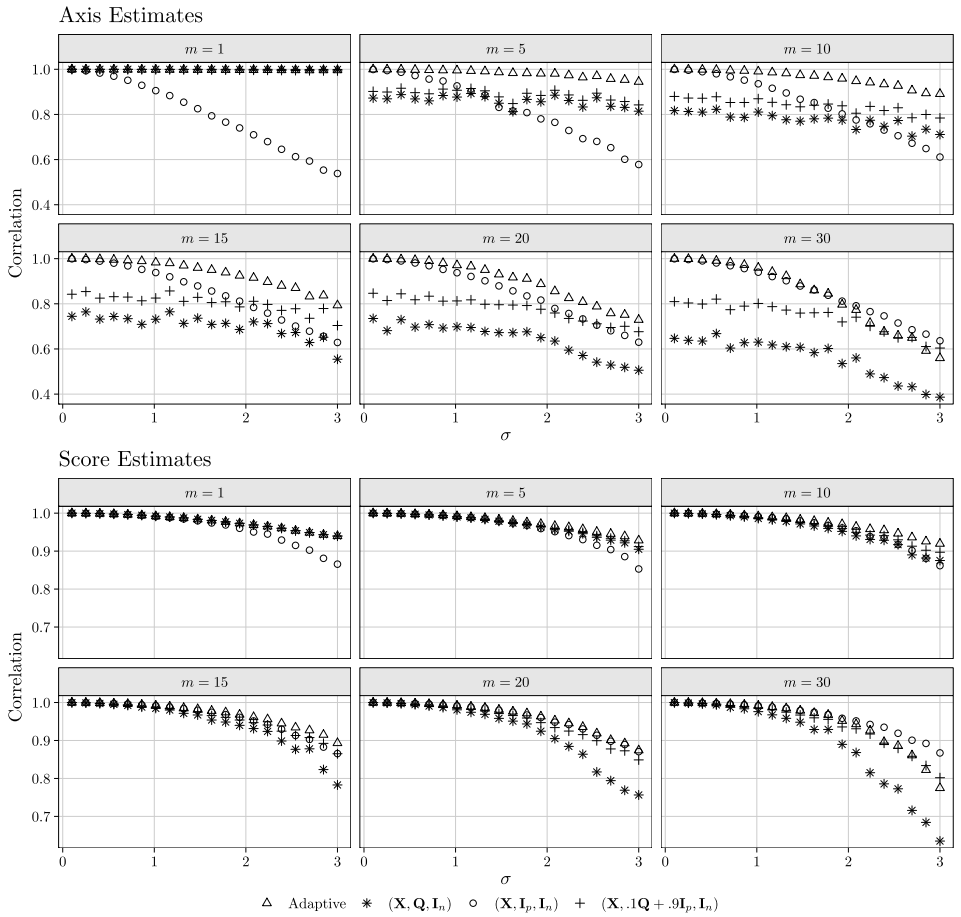

FIG. 2. *Results from simulation A. Correlations between the true and estimated principal axis* (*top*) *and true and estimated scores* (*bottom*) *for different values of m* (*columns*, *see text for explanation of m*).

We computed the correlations between the true parameters and the parameters estimated by the four methods for a range of values of m and σ. Figure 2 shows the results. Both standard PCA and adaptive gPCA recover the principal axis and the scores perfectly when there is no noise, but gPCA on $(\mathbf{X}, \mathbf{Q}, \mathbf{I}_n)$ does poorly at recovering the principal axis unless there is very strong long-range dependence in the coefficients of the principal axis ($m = 1$). gPCA on $(\mathbf{X}, 0.1\mathbf{Q} + 0.9\mathbf{I}_p, \mathbf{I}_n)$ generally does a bit better than gPCA on $(\mathbf{X}, \mathbf{Q}, \mathbf{I}_n)$, but it never does better than adaptive gPCA and usually does substantially worse. The performance of all the methods degrades with increasing noise, but adaptive gPCA does the best when the axes are at least moderately smooth.

We extended this simulation to create rank-three and rank-five plus noise matrices to see how well the methods performed when the low-rank structure was not rank one. The performances of the four methods are qualitatively similar, but in the rank-three and rank-five cases including information about the structure gives a bigger improvement over standard PCA than it does in the rank-one case. The results are shown in Figure 4.

6.2. *Simulation B.* In the second simulation we used a different noise model and a different method for choosing principal axes, with the goal of matching the microbiome example as closely as possible and to demonstrate that normally-distributed errors are not required for adaptive gPCA to perform well. We again have a tree $\mathcal{T}$ describing the relationships between the variables. Define the function

$$\text{desc}(b, \mathcal{T}) = \{j : \text{species } j \text{ descends from branch } b \text{ of } \mathcal{T}\}. \tag{29}$$

The principal axes are indicator vectors of clades. For any $b \in \mathcal{T}$, let $\boldsymbol{v} \in \mathbb{R}^p$ be such that

$$\boldsymbol{v}_j = \begin{cases} 1/\left|\text{desc}(b, \mathcal{T})\right|^{1/2} & j \in \text{desc}(b, \mathcal{T}) \\ 0 & \text{otherwise.} \end{cases} \tag{30}$$

For a given principal axis $\boldsymbol{v}$, we generate a count matrix $\mathbf{C} \in \mathbb{R}^{n \times p}$ as:

$$\mathbf{C}_{ij} \sim \text{Pois}(\boldsymbol{u}_i \boldsymbol{v}_j + e_{ij}), \tag{31}$$

$$\boldsymbol{u}_i \sim \text{Uniform}(0, 1), \quad i = 1, \ldots, n, \tag{32}$$

$$e_{ij} \sim \text{Gamma}(k, 1), \quad i = 1, \ldots, n, \ j = 1, \ldots, p. \tag{33}$$

$\mathbf{C}$ is then arcsinh transformed and centered to obtain a matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$. $\mathbf{X}$ is used as input to the methods being compared. We chose this simulation strategy to match our microbiome example. We start off with a sparse, heteroskedastic count matrix, apply an approximate variance-stabilizing transformation and then perform dimensionality reduction on the variance stabilized data matrix.

For this simulation, we again computed the correlations between the true parameters and the parameters estimated by the four methods. We did this for several values of k (which controls the error variance) and every branch b in the tree such that $50 < |\text{desc}(b, \mathcal{T})| < 200$. Figure 3 shows the results.

Adaptive gPCA has the best performance of all the methods, and it recovers principal axes corresponding to any of the branches. gPCA on $(\mathbf{X}, \mathbf{Q}, \mathbf{I}_n)$ and $(\mathbf{X}, 0.1\mathbf{Q} + 0.9\mathbf{I}_p, \mathbf{I}_n)$ are almost identical. Interestingly, the performance of these two methods depends strongly on the particular branch used (unlike adaptive gPCA and standard PCA) and is not purely a function of the number of descendants of the branch. We suspect that this is due to these methods being strongly biased toward one specific principal axis (the top eigenvector of $\mathbf{Q}$; see the Supplementary Material (Fukuyama (2019)) for why this would be). The cases where they perform well are ones where the true principal axis was similar to the favored axis while the cases they perform poorly are the opposite.

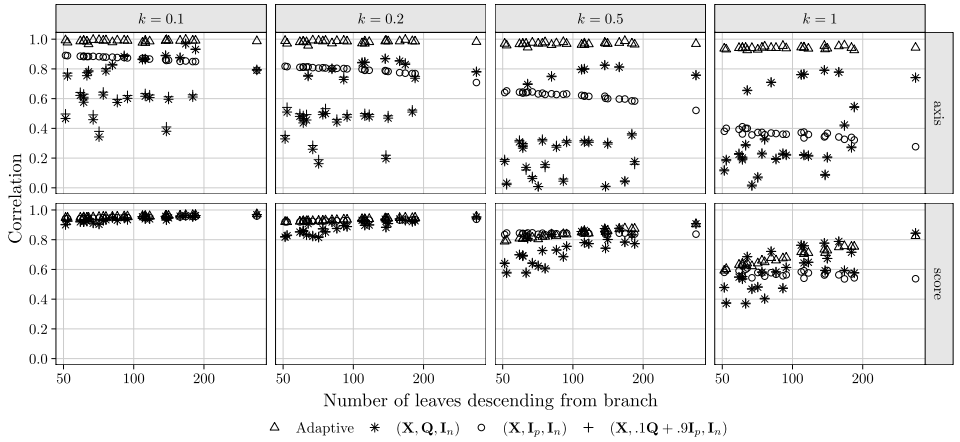

FIG. 3. *Results from simulation B. Correlations between the true and estimated principal axis (top) and the true and estimated scores along the principal axis (bottom) for different levels of noise (columns, labeled by shape parameter k in (33) in the text). For each simulation, the principal axis is non-zero on all the leaves descending from a certain branch in the tree, and the x-axis gives the number of non-zero elements.*

6.3. *Simulation C.* To compare how the methods perform when the latent structure is of higher rank, that is, when $\mathbf{X}$ is rank-k plus noise, for $k > 1$, we simulated $\mathbf{X}$ as follows:

$$\mathbf{X} = \mathbf{U}\mathbf{W}^T + \mathbf{E}, \tag{34}$$

$$\mathbf{E}_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2), \quad i = 1, \ldots, n,\ j = 1, \ldots, p, \tag{35}$$

$$\mathbf{U}_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1), \quad i = 1, \ldots, n,\ j = 1, \ldots, k, \tag{36}$$

$$\mathbf{W}_{\cdot j} \sim \mathcal{N}\bigl(\mathbf{0}_p, \mathbf{V}_{(m)}\mathbf{V}_{(m)}^T\bigr), \quad j = 1, \ldots, k. \tag{37}$$

$\mathbf{U} \in \mathbb{R}^{n\times k}$, $\mathbf{W} \in \mathbb{R}^{p\times k}$ and $\mathbf{W}_{\cdot j}$ denotes the jth column of $\mathbf{W}$. As described in Section 6.1, $\mathbf{V}_{(m)} \in \mathbb{R}^{p\times m}$ denotes the matrix whose columns are the top m eigenvectors of $\mathbf{Q}$.

We simulated $\mathbf{X}$ from this model for $k = 3$, $k = 5$ and a range of values of m and σ. We applied the four methods and computed the RV coefficients (Escoufier (1973)) between the true and estimated scores on the top k axes and the true and estimated k-dimensional principal subspace. The RV coefficient is a generalization of correlation to matrices, and the RV coefficient between two matrices $\mathbf{X} \in \mathbb{R}^{n\times p}$, $\mathbf{Y} \in \mathbb{R}^{q\times q}$ is defined as

$$\mathrm{RV}(\mathbf{X}, \mathbf{Y}) = \frac{\operatorname{tr}(\mathbf{X}\mathbf{X}^T\mathbf{Y}\mathbf{Y}^T)}{\sqrt{\operatorname{tr}(\mathbf{X}^T\mathbf{X})\operatorname{tr}(\mathbf{Y}^T\mathbf{Y})}}.$$

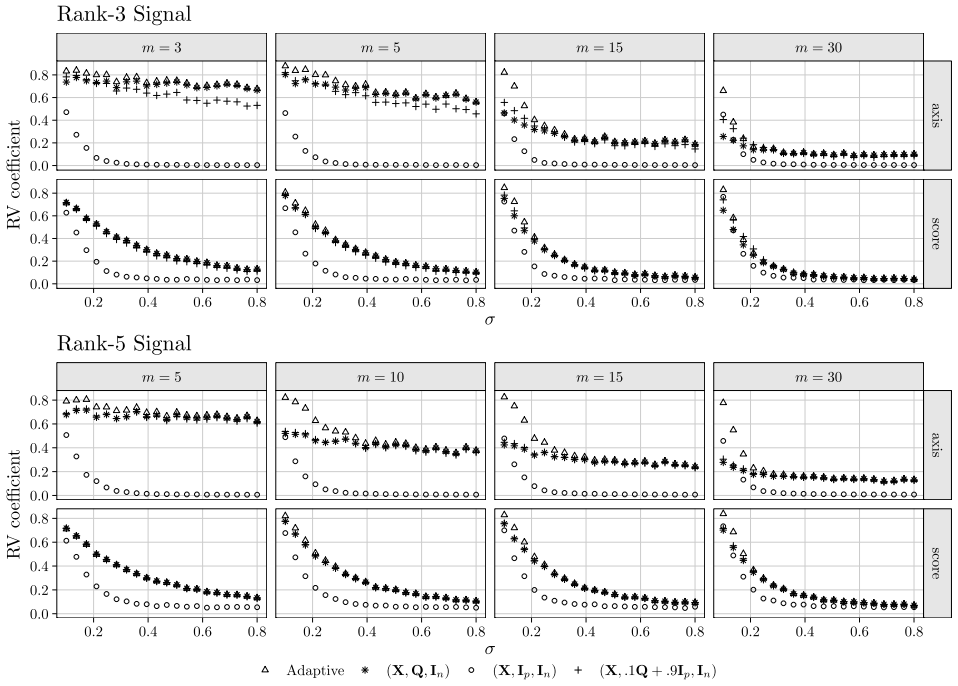

FIG. 4. *RV coefficients between true and estimated scores and principal axes for rank-*3 (*top two rows*) *and rank-*5 (*bottom two rows*) *models*.

The results are shown in Figure 4. They are qualitatively similar to the results from the rank-one simulations, although here the structured methods have more of an advantage over standard PCA than they do in the rank-one simulations.

6.4. *Summary of simulation results*. In all of these simulations, the principal axes are structured according to the tree, but the data is not generated according to the data model described in Section 4.1. This suggests that adaptive gPCA is not overly dependent on the data coming from the model we used to motivate it and can perform well in many situations. In particular it is not dependent on the data coming from a multivariate normal distribution or having multivariate normal error structure.

We do see that adaptive gPCA performs best when the true latent axes are fairly smooth on the tree and can have similar performance to DPCoA when the amount of noise is large or the latent structure is not very smooth on the tree. However, there is a simple diagnostic for this situation, the relative sizes of σ_1 and σ_2. If σ_2 is very large compared with σ_1, adaptive gPCA will give similar results to DPCoA. σ_1 being much smaller than σ_2 also suggests that the tree-structured prior is not a very good fit to the data and some caution is warranted in relying on the probabilistic interpretation.

7. Real data example. To illustrate adaptive gPCA on real data, we return to the study described in Section 2. To review, the goal was to understand the effect of antibiotics on the gut microbiome, and the data set comprises fecal samples taken from three subjects before, during and after two courses of Ciprofloxacin. The samples are labeled either "abx" or "no abx." "abx" corresponds to samples taken while the subjects were taking the antibiotic or in the first week after the antibiotic was discontinued, and "no abx" refers to all the other samples. Bacterial abundances were measured using the procedure described in Section 2. For each of the samples we have the abundances of 1651 bacterial taxa and a tree describing the phylogenetic relationships between them.

Since microbiome data come in as heteroskedastic counts, we transformed the data before applying any of the methods. In general the correct transformation depends on the data-generating process, but for microbiome datasets some common choices are a started log transformation, a variance-stabilizing transformation from the package DESeq2 (McMurdie and Holmes (2014), Callahan et al. (2016), Love, Huber and Anders (2014)) or a centered log-ratio transformation if we are thinking of the data as compositional (Fernandes et al. (2014), Filzmoser, Hron and Reimann (2009)). For the data analyzed in this paper, we used both a started log and centered log-ratio transform and found comparable results. The figures show the results using the started log transformation.

We applied adaptive gPCA, DPCoA and standard PCA to this data set. In adaptive gPCA the similarity matrix $\mathbf{Q}$ used to incorporate the phylogeny is created in the same way as in the simulations (equation (24)); $\mathbf{Q}_{ij}$ gives the amount of shared ancestry between species i and j. For adaptive gPCA the value of r (defined in equation (11)) was estimated as 0.46, indicating that the prior and the data were given approximately equal weights and that the tree prior is reasonable for the data.

Figure 5 shows the output from the three methods. The top pair of plots shows DPCoA, the middle adaptive gPCA and the bottom standard PCA. In each pair the plot on the left shows the sample scores on the first and second principal axes, and the plot on the right shows the variable loadings on the first and second principal axes. All the pairs of plots can be interpreted as biplots, so if a sample has a large score on, for example. the first principal axis, we expect it to have larger values for variables that have large loadings on the first principal axis.

The three methods give very different results. Starting with the sample plots, DPCoA shows a small antibiotic effect, but hardly any subject effect. PCA and adaptive gPCA completely separate the samples from the different subjects and give some separation between the abx and no abx samples. The second adaptive gPCA axis describes the antibiotic perturbation very well. Plotting the sample scores along the second axis over time shows that these scores are stable when the subjects are not taking the antibiotic, drop upon administration of the antibiotic and return to baseline when the treatment is stopped (Figure 6).

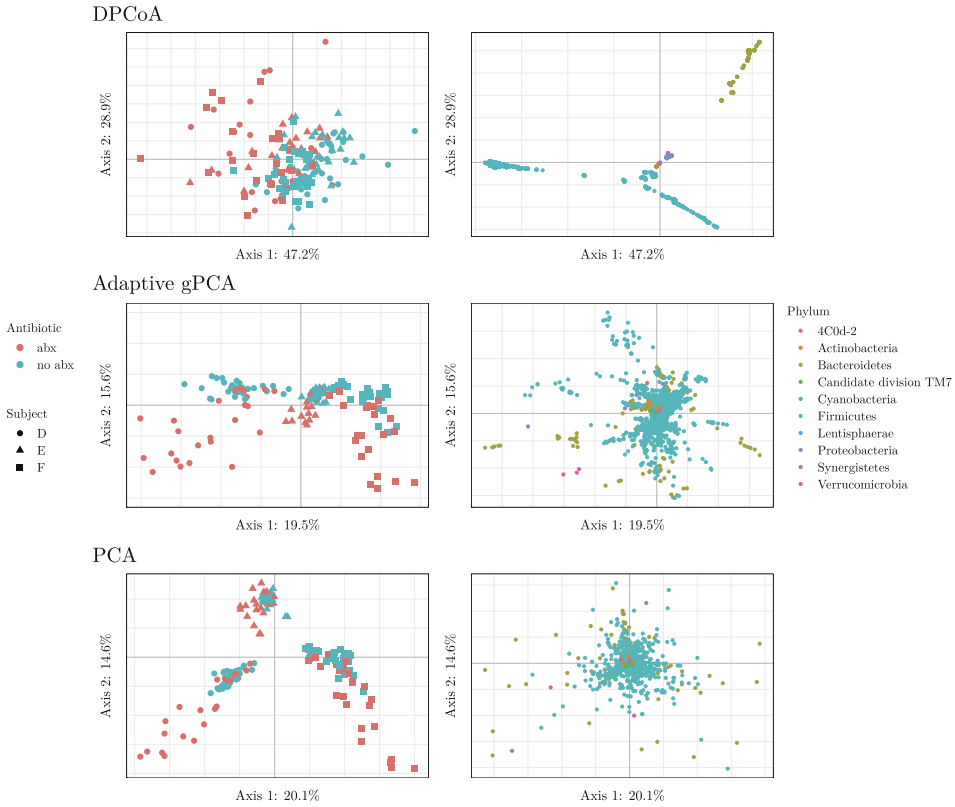

FIG. 5. *Sample* (*left*) *and taxon* (*right*) *plots for DPCoA* (*top*), *adaptive gPCA* (*middle*), *and standard PCA* (*bottom*). *Colors in the sample plots represent a binning of the sample points into abx* (*either when the subject was on antibiotics or the week immediately after*) *or no abx* (*all other times*). *The colors in the taxon plots represent phyla.*

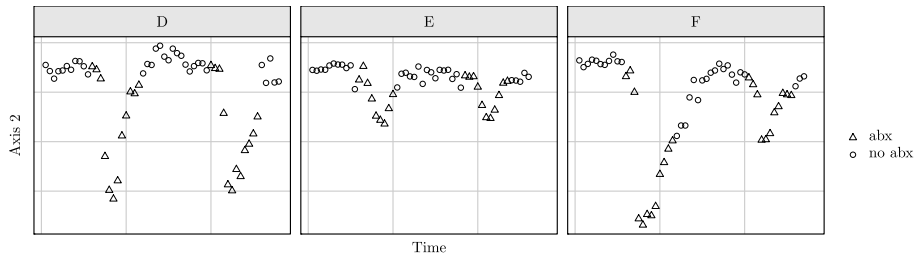

FIG. 6. *A plot of the scores along the second axis from adaptive gPCA by time, plotted for each of the three individuals. We see very clearly that this axis is capturing taxa that change during the administration of the antibiotic but which are stable otherwise. The corresponding plots for PCA and DPCoA are much less compelling.*

The three methods also give very different variable (taxa) loadings. The taxa loadings in PCA are not associated with the phylogeny. Similar taxa are no more likely to have similar loadings on the principal axes than dissimilar taxa. On the other end of the spectrum, the taxa loadings from DPCoA are very constrained by the tree, and in particular the loadings are such that variables corresponding to taxa in different phyla are present in disjoint regions in the principal plane. This corresponds to the global structure imposed by DPCoA that we described in Section 5.1.1. Adaptive gPCA gives results somewhere in the middle. We see that phylogenetically similar taxa are more likely to have similar loadings on the principal axes, but the phenomenon is more local—very closely related taxa have similar loadings, but distantly related taxa are not necessarily far apart.

A note about the variance explained: as described in Section 3.3, the fraction of variance explained is reported with respect to the gPCA inner product. Figure 5 indicates that the first DPCoA axes explain a very large fraction of the variance in terms of the $\mathbf{Q}$ inner product, but this is primarily due to $\mathbf{Q}$ being approximately low rank. If we compute the fraction of the variance explained by each of the axes in terms of the standard inner product instead, we get 20.1% and 14.6% for the fraction of variance explained by the top two PCA axes, 7.6% and 5.4% for the first two adaptive gPCA axes and 0.7% and 1% for the top two DPCoA axes. From this perspective it looks like adaptive gPCA is doing a better job of trading off between structure in the variables and fraction of variance explained than DPCoA.

Since the purpose of the study was to understand the effect of antibiotics on the gut microbiome and since the second adaptive gPCA axis seems to describe the disturbance due to the antibiotic, we looked in more detail at the behavior of the taxa with large positive or negative loadings on the second adaptive gPCA axis. The 27 taxa with the largest positive scores along the second adaptive gPCA axis are all of the genus *Faecalibacterium*. Although different members of the genus are present or absent in different subjects, when present they all decline in relative abundance during antibiotic treatment and rebound when the treatment is discontinued (see the top row of Figure 7). Consistent with what we see in Figure 6, Subject E shows much less of a disturbance compared to subjects D and F, and in subject F the second course of antibiotics yields a much smaller disturbance than the first.

A similar result holds for the 21 members of the Firmicutes phylum with the largest negative scores on the second adaptive gPCA axis. The members of this group are not present in every subject, but when they are present their abundance tends to increase with the antibiotic treatment.

This analysis shows us the interpretive advantage of adaptive gPCA over standard PCA and DPCoA. In standard PCA the axes are difficult to interpret because of the lack of relationship between the phylogenetic structure and the loadings of the variables on the principal axes. On the other hand DPCoA misses some of the latent structure we know is present in the data, consistent with the simulations showing that DPCoA only performs well when the latent axes are very smooth on

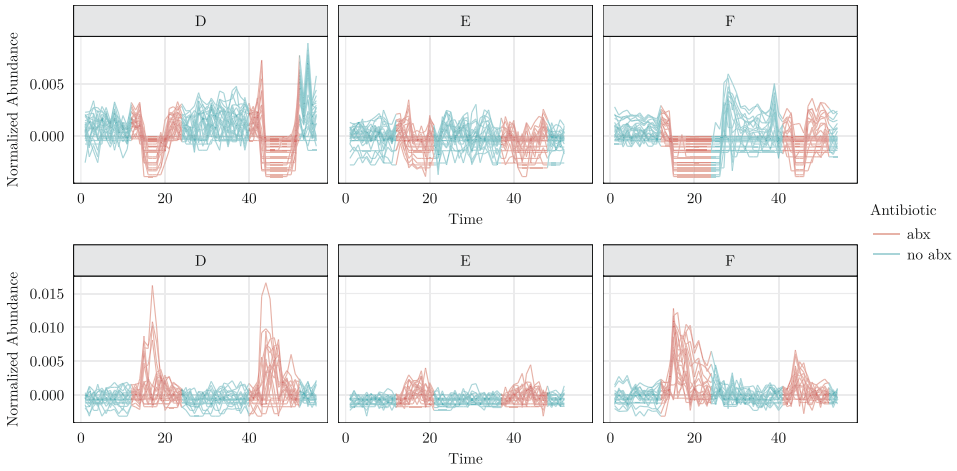

FIG. 7. *Normalized abundances for two groups of taxa. Each line represents a taxon, each facet represents a subject. The top row shows the normalized abundances of each of* 27 *OTUs with the largest positive loadings on the second adaptive gPCA axis, and the bottom row shows the normalized abundances of the* 21 *Firmicutes with the largest negative loadings on the second adaptive gPCA axis.*

the tree. Adaptive gPCA recovers the latent structure well and also has axes that are interpretable in terms of small groups of related taxa. This sort of structure helps us understand the underlying biology and can help provide suggestions about what hypotheses to consider next.

8. Conclusion. In this paper we presented a method for creating a low-dimensional representation of a data matrix while taking into account side information about the relationships between the variables. This is done by imposing a prior encoding the relationships between the variables and performing PCA on the resulting posteriors, taking into account the fact that the posteriors have nonspherical variance. We show that performing PCA on the posterior estimates obtained with this prior corresponds to a generalized PCA, with a one-dimensional family of gPCAs arising from varying the prior strength. A member of this family can then be picked by estimating the scalings of the prior and the noise by maximum marginal likelihood. We call the gPCA obtained in this manner adaptive gPCA.

One major advantage of adaptive gPCA is that it is motivated by a probabilistic model. This means that the representation of the samples has a simple interpretation as a representation of posterior estimates. Using this model also makes it conceptually simple to adapt the method to other noise and variable structures.

Other attractive features of adaptive gPCA are that we can obtain the global solution (i.e., the algorithm does not settle into a local minimum), that we can choose the amount of regularization to perform without having to resort to potentially time-consuming cross-validation, and that we can use any sort of structure on the variables.

Using adaptive gPCA on a real data set shows us some of the practical advantages of the method. We were able to identify the latent structure in the data (the differences between the individuals and the antibiotic treatment), and we were able to use the loadings of the variables on the principal axes to understand the biology behind this latent structure. For instance, the second adaptive gPCA axis was related to the administration of the antibiotic, and taxa loading strongly on the second axis consisted of related groups sharing a response to the antibiotic.

The current implementation of adaptive gPCA has two primary limitations. It assumes spherical noise, and it assumes that the variable structure is known without error. In Section 5.2 we sketched out how to relax either of those assumptions. However, both of the suggestions require significantly more development to work in practice. Using nonspherical noise leads to a much higher computational burden, and relaxing the assumption of known variable structure can be difficult to estimate in practice.

Adaptive gPCA can be extended in several directions. Information about the precision with which different variables or samples are measured can be incorporated as either sample or variable weights. The family of inner products described in this paper can be imported into other methods that work in non*a*standard inner product spaces, such as between- or within-class analysis (Dray, Pavoine and Aguirre de Cárcer (2015)), to encourage structured solutions. It can be used in conjunction with sparse gPCA (Allen, Grosenick and Taylor (2014)) for sparse and structured dimensionality reduction, and combining these with between-class analysis would give a sparse, structured method for explaining group differences.

An R implementation of adaptive gPCA is available from CRAN and can be installed in R using the command

```
install.packages("adaptiveGPCA")
```

In addition to implementing adaptive gPCA, the package includes a shiny app (Chang et al. (2016)) that allows users to interactively visualize the effects of different prior strengths. It also includes the antibiotic data used in this paper and a vignette that reproduces the analysis.

Acknowledgments. The author thanks Susan Holmes, Kris Sankaran, Lan Huong Nguyen, Julie Josse, the three anonymous reviewers and the two Editors whose suggestions greatly improved the manuscript.

SUPPLEMENTARY MATERIAL

Proofs and additional discussion (DOI: 10.1214/18-AOAS1227SUPP; .pdf). The supplemental material provides proofs of the theorems and additional discussion on interpretation of generalized PCA in terms of the eigendecomposition of the matrix $\mathbf{Q}$.

REFERENCES

ALLEN, G. I., GROSENICK, L. and TAYLOR, J. (2014). A generalized least-square matrix decomposition. *J. Amer. Statist. Assoc.* **109** 145–159. MR3180553

BRENNER, D. J., STALEY, J. T. and KRIEG, N. R. (2005). Classification of procaryotic organisms and the concept of bacterial speciation. In *Bergey's Manual of Systematic Bacteriology* 27–32. Springer, Berlin.

CALLAHAN, B. J., SANKARAN, K., FUKUYAMA, J. A., MCMURDIE, P. J. and HOLMES, S. P. (2016). Bioconductor workflow for microbiome data analysis: From raw reads to community analyses. *F*1000*Res* **5** 1492.

CAUSSINUS, H. (1986). Models and uses of principal component analysis. *Multidimensional Data Analysis* **86** 149–170.

CHANG, Q., LUAN, Y. and SUN, F. (2011). Variance adjusted weighted unifrac: A powerful beta diversity measure for comparing communities based on phylogeny. *BMC Bioinform.* **12** 1.

CHANG, W., CHENG, J., ALLAIRE, J., XIE, Y. and MCPHERSON, J. (2016). shiny: Web Application Framework for R. R package version 0.13.2.

CHEN, J., BITTINGER, K., CHARLSON, E. S., HOFFMANN, C., LEWIS, J., WU, G. D., COLLMAN, R. G., BUSHMAN, F. D. and LI, H. (2012). Associating microbiome composition with environmental covariates using generalized unifrac distances. *Bioinformatics* **28** 2106–2113.

COHAN, F. M. (2002). What are bacterial species? *Annual Reviews in Microbiology* **56** 457–487.

DETHLEFSEN, L. and RELMAN, D. A. (2011). Incomplete recovery and individualized responses of the human distal gut microbiota to repeated antibiotic perturbation. *Proc. Natl. Acad. Sci. USA* **108** 4554–4561.

DOOLITTLE, W. F. and PAPKE, R. T. (2006). Genomics and the bacterial species problem. *Genome Biol.* **7** 1.

DRAY, S., PAVOINE, S. and AGUIRRE DE CÁRCER, D. (2015). Considering external information to improve the phylogenetic comparison of microbial communities: A new approach based on constrained double principal coordinates analysis (cdpcoa). *Molecular Ecology Resources* **15** 242–249.

EDGAR, R. C. (2010). Search and clustering orders of magnitude faster than blast. *Bioinformatics* **26** 2460–2461.

ESCOUFIER, Y. (1973). Le traitement des variables vectorielles. *Biometrics* **29** 751–760. MR0334416

FERNANDES, A. D., REID, J. N., MACKLAIM, J. M., MCMURROUGH, T. A., EDGELL, D. R. and GLOOR, G. B. (2014). Unifying the analysis of high-throughput sequencing datasets: Characterizing rna-seq, 16s rrna gene sequencing and selective growth experiments by compositional data analysis. *Microbiome* **2** 15.

FILZMOSER, P., HRON, K. and REIMANN, C. (2009). Principal component analysis for compositional data with outliers. *Environmetrics* **20** 621–632. MR2838477

FUKUYAMA, J. (2019). Supplement to "Adaptive gPCA: A method for structured dimensionality reduction with applications to microbiome data." DOI:10.1214/18-AOAS1227SUPP.

HOLMES, S. (2008). Multivariate data analysis: The French way. In *Probability and Statistics*: *Essays in Honor of David A. Freedman. Inst. Math. Stat.* (*IMS*) *Collect.* **2** 219–233. IMS, Beachwood, OH. MR2459953

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

KONDOR, R. I. and LAFFERTY, J. (2002). Diffusion kernels on graphs and other discrete structures. In *Proceedings of the* 19*th International Conference on Machine Learning* 315–322.

LI, C. and LI, H. (2008). Network-constrained regularization and variable selection for analysis of genomic data. *Bioinformatics* **24** 1175–1182.

LOVE, M. I., HUBER, W. and ANDERS, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol*. **15** 550.

LOZUPONE, C. and KNIGHT, R. (2005). Unifrac: A new phylogenetic method for comparing microbial communities. *Applied and Environmental Microbiology* **71** 8228–8235.

LOZUPONE, C. A., HAMADY, M., KELLEY, S. T. and KNIGHT, R. (2007). Quantitative and qualitative β diversity measures lead to different insights into factors that structure microbial communities. *Applied and Environmental Microbiology* **73** 1576–1585.

MATSEN, F. A. and EVANS, S. N. (2013). Edge principal components and squash clustering: Using the special structure of phylogenetic placement data for sample comparison. *PLoS ONE*.

MCMURDIE, P. J. and HOLMES, S. (2014). Waste not, want not: Why rarefying microbiome data is inadmissible. *PLoS Comput. Biol.* **10** e1003531.

PARADIS, E., CLAUDE, J. and STRIMMER, K. (2004). APE: Analyses of phylogenetics and evolution in R language. *Bioinformatics* **20** 289–290.

PAVOINE, S., DUFOUR, A.-B. and CHESSEL, D. (2004). From dissimilarities among species to dissimilarities among communities: A double principal coordinate analysis. *J. Theoret. Biol.* **228** 523–537. MR2080909

PENROSE, R. (1955). A generalized inverse for matrices. *Proc. Camb. Philos. Soc.* **51** 406–413. MR0069793

PURDOM, E. (2011). Analysis of a data matrix and a graph: Metagenomic data and the phylogenetic tree. *Ann. Appl. Stat.* **5** 2326–2358. MR2907117

QUAST, C., PRUESSE, E., YILMAZ, P., GERKEN, J., SCHWEER, T., YARZA, P., PEPLIES, J. and GLÖCKNER, F. O. (2013). The silva ribosomal rna gene database project: Improved data processing and web-based tools. *Nucleic Acids Res.* **41** D590–D596.

R CORE TEAM (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

RANDOLPH, T. W., ZHAO, S., COPELAND, W., HULLAR, M. and SHOJAIE, A. (2018). Kernel-penalized regression for analysis of microbiome data. *Ann. Appl. Stat.* **12** 540–566. MR3773404

RAPAPORT, F., ZINOVYEV, A., DUTREIX, M., BARILLOT, E. and VERT, J.-P. (2007). Classification of microarray data using gene networks. *BMC Bioinform.* **8** 35.

RINALDO, A. (2009). Properties and refinements of the fused lasso. *Ann. Statist.* **37** 2922–2952. MR2541451

SUBRAMANIAN, A., TAMAYO, P., MOOTHA, V. K., MUKHERJEE, S., EBERT, B. L., GILLETTE, M. A., PAULOVICH, A., POMEROY, S. L., GOLUB, T. R. et al. (2005). Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. USA* **102** 15545–15550.

TIBSHIRANI, R. and WANG, P. (2008). Spatial smoothing and hot spot detection for cgh data using the fused lasso. *Biostatistics* **9** 18–29.

TIBSHIRANI, R., SAUNDERS, M., ROSSET, S., ZHU, J. and KNIGHT, K. (2005). Sparsity and smoothness via the fused lasso. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **67** 91–108. MR2136641

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.

DEPARTMENT OF STATISTICS
INDIANA UNIVERSITY
BLOOMINGTON, INDIANA 47408
USA
E-MAIL: jfukuyam@iu.edu