Dual-Constrained and Primal-Constrained Principal Component Analysis

Erik Andries * [1,2] and Ramin Nikzad-Langerodi † [3]

[1] *Center for Advanced Research Computing*
University of University Mexico
Albuquerque, NM, USA

[2] *Central New Mexico Community College*
Albuquerque, NM, USA

[3] *Data Science Group*
Software Competence Center Hagenberg (SCCH) GmbH,
Hagenberg, Austria

May 5, 2022

Abstract

Uniform Manifold Approximation and Projection (UMAP) and related manifold embedding techniques generate nonlinear projections of high-dimensional data onto low-dimensional subspaces. However, due to the non-linearity of these methods, there is no analogous transformation matrix (i.e., the loadings or eigenvectors) that allows one to see the corresponding relationship between each sample location and each feature (e.g., wavelength). We propose a mechanism that embeds the pairwise distance structure of UMAP-type embeddings into the machinery of Principal Component Analysis such that one can recover an approximate set of loading vectors associated with the intrinsic topology of high-dimensional data.

Keywords— Principal Component Analysis, UMAP, Embedding

1 Introduction

Most examples of high-dimensional data sets such as those in spectroscopy are truly not high-dimensional. Each data point or spectrum typically lies on a low-dimensional manifold embedded in a high-dimensional space spanned by all available variables or features (e.g., wavelengths). Linear projection techniques, such as Principal Component Analysis (PCA), try to approximate a subspace in which this manifold is embedded by taking linear combinations of all of the variables. While classical PCA is good at capturing global trends in data by finding a low-dimensional projection that maximizes variance, it largely neglects relevant aspects that are highly localized in structure. For example, in many imaging data sets such as those generated by functional Magnetic Resonance Imaging where one

*Corresponding author: erik.andries@gmail.com
†ramin.nikzad-langerodi@scch.at

measures neuronal activity at a particular location in the brain, PCA performs poorly in terms of recovering true low-rank signals [1].

Recent nonlinear methods such as t-SNE and UMAP create low-dimensional projections that attempt to preserve the underlying topological structure of the manifold [2, 3]. In most cases, the preservation of the underlying structure is *intrinsically-motivated*: data points that are close in the high-dimensional space should also be close in the low-dimensional projection. In some cases, the manifold structure can also be *extrinsically-motivated*, e.g., one can impose that two samples be deemed close together or far away on the basis of expert domain knowledge. In the analysis of human microbiome data, for example, external information from various bioinformatic pipelines are routinely used to quantify information about samples and features, e.g., phylogenetic dissimilarity across a pair of samples [4]. Nonlinear projection methods such as UMAP provide valuable and complementary insight on how samples are co-localized. However, such embeddings provide little guidance on how a particular feature (wavelength) impacts the co-localization of samples in the low-dimensional projection. When applying UMAP to a spectroscopic data set, is the impact of i-th wavelength negligible on the projection, or is it significant? In this paper, we embed a sample-to-sample dissimilarity matrix into PCA such that we aim to preserve the embedded nearest-neighbor structure provided by the UMAP projection. Moreover, we obtain an unsupervised model (a linear transformation matrix) by which we 1) obtain useful wavelength discriminating information and 2) make projections on new samples.

Generalizations of PCA have a long history in chemometrics [5–8]. The primary thrust of these PCA variants is to either deferentially weight samples or variables such that general noise structures (e.g., heteroscedasticity) can be accommodated, or to up-weight more recently acquired samples and down-weight older samples. In this paper, we do not concern ourselves with these issues. As already mentioned, our aim is to instead incorporate a priori nearest-neighbor structure via pairwise dissimilarities between samples. However, this aspect has not garnered much attention within chemometrics as it has outside of chemometrics. Of particular interest are the Locality Preserving Projections or LPPs [9–11]. LPPs linearly project high-dimensional data onto low-dimensional subspaces such that the Euclidean distances between data points within local neighborhoods are better preserved compared to methods that focus on global variance maximization such as PCA. Other approaches extend LPPs for classification purposes in supervised or semi-supervised scenarios [12–14]. In contrast, we investigate the behavior of linear dimension reduction where the nearest-neighbor topology has been externally-supplied to us by a nonlinear manifold embedding (e.g. UMAP). In this sense, this paper shares common topology-preserving mechanisms that have been explored in chemometrics, in particular domain adaptation and calibration transfer [15–18].

Section 2 examines various classical PCA formulations. In chemometrics, the most common way of explaining PCA does not easily admit generalizations that allow the embedding of a priori information (e.g. sample-to-sample dissimilarity). Instead, we utilize a less well-known notion of PCA that does admit such generalizations: find the projection that maximizes the sum of all squared pairwise distances between the projected samples in the low-dimensional subspace. Section 3 describes the nonlinear projection technique known as UMAP. We show that it provides useful insights with respect to sample clustering that would be difficult to obtain from PCA alone. Section 4 describes two extensions of PCA known as Dual-Constrained PCA (DC-PCA) and Primal-Constrained PCA (PC-PCA). Mathematically, both DC-PCA and PC-PCA can be expressed as a generalized eigenvalue problem. Section 5 examines additional projections and the resulting eigenvectors from DC-PCA and PC-PCA. Finally, Section 6 concludes the paper and proposes future research.

Notation. In this paper, matrices and vectors are denoted by boldfaced uppercase (e.g., $\mathbf{X}$) and lowercase (e.g., $\mathbf{x}$) letters, respectively. The superscripts $^{\mathrm{T}}$, $^{+}$ and $^{-1}$ indicate the transpose, pseudoinverse and inverse of a matrix or vector. Unless otherwise noted, all vectors are assumed to be column vectors. The columns of a matrix are denoted by *parenthetical* subscripts, e.g., $\mathbf{x}_{(\mathrm{j})}$ is the jth column of $\mathbf{X}$, while $\mathbf{x}_{\mathrm{i}}^{\mathrm{T}} = [x_{i1}, x_{i2}, \ldots, x_{in}]$ indicates the ith sample or row of $\mathbf{X}$. The matrix of spectra $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{\mathrm{m}}]^{\mathrm{T}} = [\mathbf{x}_{(1)}, \mathbf{x}_{(2)}, \ldots, \mathbf{x}_{(\mathrm{n})}]$ of size $m \times n$ consists of m samples (rows) and n variables (columns, e.g., wavelengths). We assume that the data has already been mean-centered such that

$$\mathbf{X} \leftarrow \mathbf{J}\mathbf{X}, \quad \mathbf{J} = \mathbf{I}_m - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^{\mathrm{T}}. \tag{1}$$

The first k columns of a matrix $\mathbf{M}$ will be denoted as $\mathbf{M}_k$.

2 PCA Formulations

PCA projects high-dimensional data (e.g., spectra) onto a low-dimensional subspace of uncorrelated variables (scores) called principal components (PCs). The first PC accounts for as much of the variance in the data as possible, and each succeeding PC accounts for as much of the remaining variance as possible. By using only the first few PCs, PCA makes it possible to effectively reduce the number of meaningful dimensions of the spectra, while simultaneously maximizing as much variance as possible. Traditionally, the projection is accomplished by a mapping via a $n \times k$ linear transformation matrix containing k orthonormal direction vectors $\mathbf{V}_k = [\mathbf{v}_{(1)}, \mathbf{v}_{(2)}, \ldots, \mathbf{v}_{(k)}]$ whereby the number of PCs we seek is small in number: $k \ll \min(m-1, n)$. The PCs or matrix of score vectors $\mathbf{T}_k$ are then obtained by the application of $\mathbf{V}_k$ which in turn shatters each spectrum $\mathbf{x}_i$ into perpendicular pieces $\mathbf{x}_i^T \mathbf{v}_{(j)}$:

$$\mathbf{T}_k = [\mathbf{t}_1, \ldots, \mathbf{t}_m]^T = [\mathbf{t}_{(1)}, \ldots, \mathbf{t}_{(k)}] = \mathbf{X}\mathbf{V}_k. \tag{2}$$

Here, each projected sample $\mathbf{t}_i$ in (2) can be expressed row-wise as $\mathbf{t}_i^T = [t_{i1}, \ldots, t_{ik}] = \mathbf{x}_i^T \mathbf{V}_k$. Similarly, each score vector in (2) is a column vector of $\mathbf{X}$ such that $\mathbf{t}_{(j)} = \mathbf{X}\mathbf{v}_{(j)}$. Knowing the value of each loading vector component v_{ij} of $\mathbf{v}_{(j)} = [v_{1j}, v_{2j}, \ldots, v_{nj}]^T$ is desirable since each feature component corresponds to a variable such as a wavelength. Since each column of $\mathbf{T}_k$ can be expressed as a linear combination of the columns of $\mathbf{X}$, a loading vector component v_{ij} with small magnitude indicates that the ith wavelength of $\mathbf{X}$ has a negligible impact on the construction of the jth PC.

Australian Rainforest Leaf Litter Data Set: In this paper, to enable the comparison of different projection methods as they are introduced, we will use the Australian Rainforest Leaf Litter (ARLL) data set to illustrate how samples are projected[19]. In brief, the ARLL data set consists of 702 NIR samples of leaf-litter collected at various stages of decomposition across various rainforest sites in Queensland, Australia. The spectra are displayed in absorbance units. Each spectrum consists of 1153 wavelengths ranging from 800nm to 2773nm with non-equal spacing between wavelength intervals. In addition to different collection sites, there were two different collection treatments: one used leaves collected in litter traps and exposed at their respective sites (in situ), and the other used leaf litter from the deciduous tree Archidendron vaillantii (control). Moreover, the sample collection times were allowed to vary, i.e., the amount of time leaves were left exposed on the soil surface. These times varied from 0 days to well over a year.

Figure 1 shows the absorbance spectra. The top subplot displays the absorbance waveforms as a function of wavelength—one curve for each spectrum. The bottom subplot displays the same absorbance values as a heatmap. To better see differences in absorbance across samples (rows) and wavelengths (columns), the dynamic range of absorbance has been truncated in the heatmap—only absorbance values below 0.004 are shown. (All absorbance values above 0.004 are color-coded as pink.) The samples in the heatmap are grouped into 11 unique sample pairs (i, j) of sample treatments ($i \in \{1, 2\}$) and collection times ($j \in \{1, 2, 3, 4, 5, 6\}$), i.e.,

$$\{(1,1), (1,2), (1,3), (1,4), (1,5), (1,6), (2,2), (2,3), (2,4), (2,5), (2.6\}.$$

(There are no samples associated with $i = 2$ and $j = 1$.) With respect to samples, there are are pronounced differences between sample treatments $i = 1$ and $i = 2$.

Both the top and bottom subplots show two main absorption bands due to water: one band located around 1900nm, and another—and the largest—at 2720nm due to atmospheric absorption. As a consequence, we will examine two versions of the ARLL data set: one using all wavelengths (denoted as *full*), and another that truncates all wavelengths greater than 2500nm (denoted as *truncated*). Without this truncation, subsequent projections will be unduly influenced by the atmospheric absorption of water. This can be seen in Table 1 where the explained variances are given for the two-dimensional (2D) and three-dimensional (3D) PCA projections across the full and truncated ARLL versions. By including the dominant waterband, the explained variances are higher relative to the explained variances associated with the truncated ARLL data set. In short, we want to assess how projections are affected by the inclusion or exclusion of the dominant waterband.

Figure 2 displays various PCA projections. Column 1 displays three-dimensional projections, while columns 2, 3 and 4 display two-dimensional projections associated with PCs 1 and 2, PCs 1 and 3 and PCs 2 and 3, respectively.

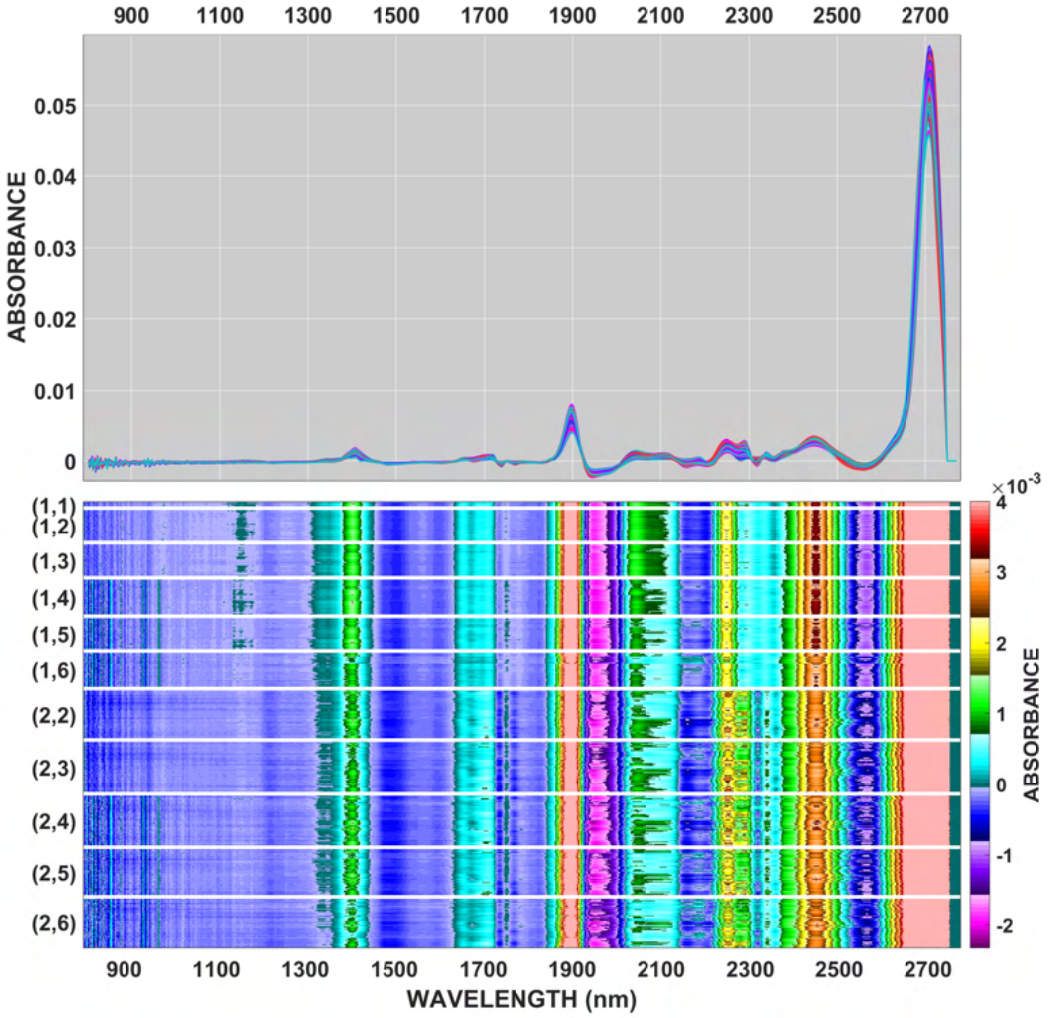

Figure 1: Spectra associated with the ARLL data set. The top subplot displays the spectra—one curve for each spectrum. The bottom subplot displays absorbance as a heatmap (only absorbance values below 0.004 are shown; all values above 0.004 are color-coded as pink). The samples in the heatmap are grouped into 11 unique sample pairs (i, j) of sample treatments ($i \in \{1, 2\}$) and collection times ($j \in \{1, 2, 3, 4, 5, 6\}$).

EXPLAINED VARIANCE		
	2D	3D
FULL	83.9%	89.5%
TRUNCATED	58.6%	80.3%

Table 1: The two- and three-dimensional explained variance associated with PCA.

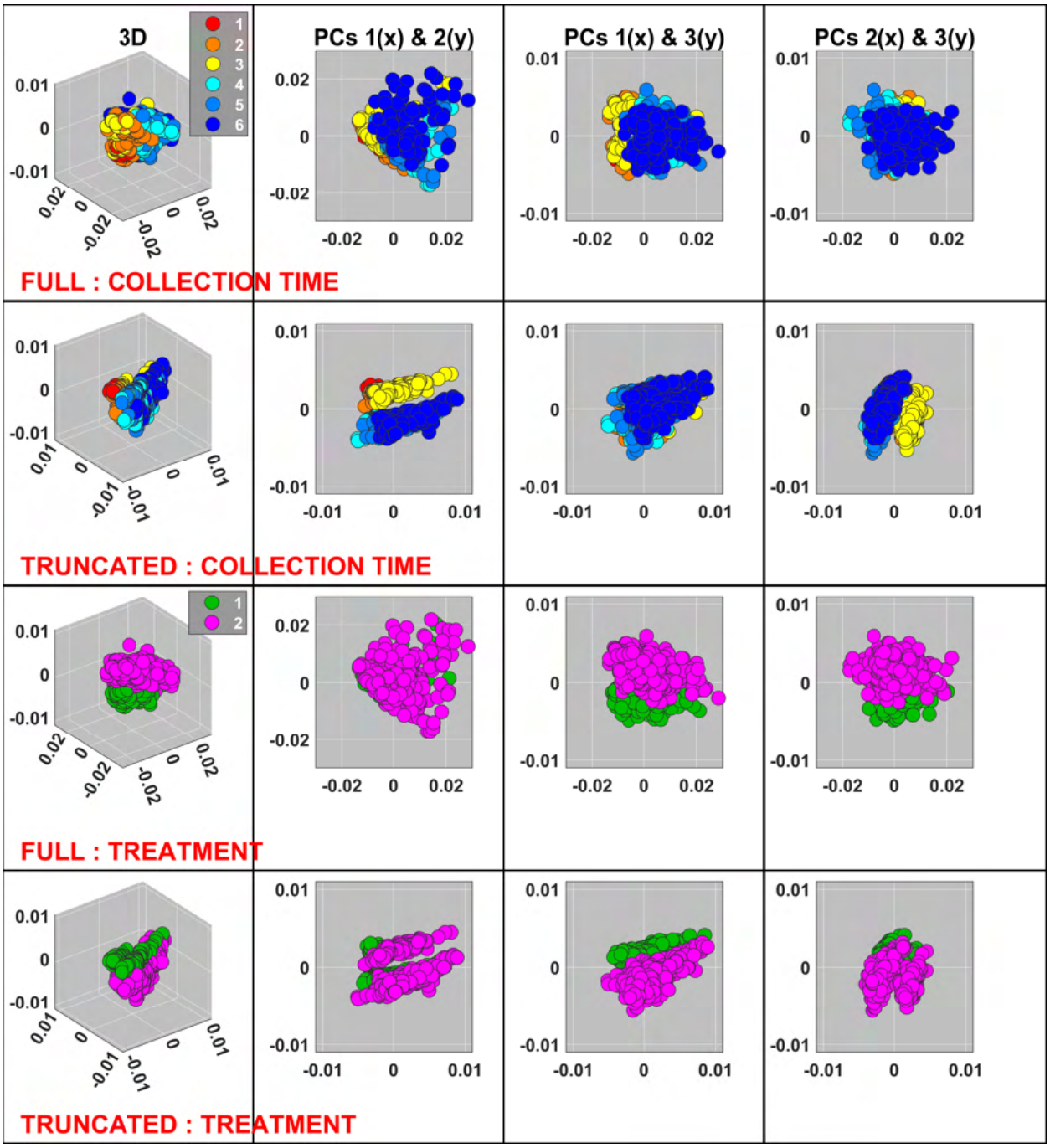

Figure 2: PCA applied to the full and truncated ARLL data sets. Column 1 displays three-dimensional PCA projections. Columns 2, 3 and 4 display two-dimensional PCA projections across PCs 1 and 2, PCs 1 and 3 and PCs 2 and 3, respectively. The first two rows and the last two rows correspond to samples that are color-coded according to the six collection times and two treatments, respectively. Rows 1 and 3 correspond to the full ARLL data set while the rows 2 and 4 correspond to the truncated ARLL data set.

The first two rows and the last two rows correspond to samples color-coded according to the six collection times and two treatments, respectively. (Rows 1 and 3 are associated with the full ARLL data set while the rows 2 and 4 are associated with the truncated ARLL data set.) For the PCA projections associated with the full ARLL data set, three dimensions are required to show appreciable sample separation across both sample collection times and sample treatments. (There is separation but no clear separation between samples of different types.) For the PCA projections associated with the truncated ARLL data set, two latent dimensions suffice for a distinct sample separation according to sample collection time (PCs 1&2 and PCs 2&3). However, a third latent dimension associated with PC 3 is required to show separation with respect to sample treatment. In short, all three PCs are required to show discernible separation in samples across both collection time and treatment. It is important to note that it is difficult to detect clusters in spectroscopic data with PCA (or any other unsupervised data analysis technique) without knowing in advance the colors associated with sample membership.

2.1 Singular Value Decomposition (SVD)

Perhaps the most common way to extract the direction vectors, or loadings, in PCA is to apply the Singular Value Decomposition (SVD) to the $m \times n$ matrix of mean-centered spectra $\mathbf{X}$

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_{(\mathrm{i})} \mathbf{v}_{(\mathrm{i})}^{\mathrm{T}}, \quad \begin{cases} \mathbf{U} = [\mathbf{u}_{(1)}, \ldots, \mathbf{u}_{(\mathrm{r})}] \\ \mathbf{\Sigma} = \mathrm{diag}(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_r) \\ \mathbf{V} = [\mathbf{v}_{(1)}, \ldots, \mathbf{v}_{(\mathrm{r})}]. \end{cases} \tag{3}$$

Here r indicates the rank of $\mathbf{X}$ such that $1 \leq r \leq \min(m-1, n)$. We use only the first k ($k \ll r$) right singular vectors such that $\mathbf{T}_k = \mathbf{U}_k\mathbf{\Sigma}_k = \mathbf{X}\mathbf{V}_k$.

2.2 Eigenvectors of the Covariance Matrix

The variance of a projection onto a direction of unit length can be expressed as the following function[20]:

$$f(\mathbf{v}) = \mathbf{v}^{\mathrm{T}} \left(\frac{1}{m} \mathbf{X}^{\mathrm{T}} \mathbf{X} \right) \mathbf{v} \tag{4}$$

As a result, finding the direction of maximal variance is also the direction that satisfies

$$\max_{\mathbf{v}} f(\mathbf{v}) \;\; \text{subject to} \;\; \mathbf{v}^{\mathrm{T}}\mathbf{v} = ||\mathbf{v}||_2 = 1. \tag{5}$$

Using the standard calculus-based approach of Lagrangian multipliers, maximizing (5) is equivalent to finding the eigenvector associated with the largest eigenvalue of the following eigenvalue problem:

$$\mathbf{C}\mathbf{v} = \lambda\mathbf{v}, \quad \mathbf{C} = \mathbf{X}^{\mathrm{T}}\mathbf{X}. \tag{6}$$

Note that re-scaling a matrix does not alter the eigenvectors of $\mathbf{C}$ but simply re-scales the corresponding eigenvalues. Moreover, the eigendecomposition of a symmetric positive semi-definite matrix (characterized by having all eigenvalues being non-negative, among all other properties), such as $\mathbf{C}$, yields an orthogonal basis of eigenvectors.

By extension, PCA finds multiple direction vectors $\{\mathbf{v}_{(1)}, \ldots, \mathbf{v}_{(\mathrm{k})}\}$ contained in $\mathbf{V}_k$ that maximize the sum of variance terms:

$$\max_{\mathbf{v}_{(1)}, \ldots, \mathbf{v}_{(\mathrm{k})}} \sum_{j=1}^{k} \mathbf{v}_{(\mathrm{j})}^{\mathrm{T}} \mathbf{C} \mathbf{v}_{(\mathrm{j})} \;\; \text{subject to} \;\; \mathbf{v}_{(\mathrm{i})}^{\mathrm{T}} \mathbf{v}_{(\mathrm{j})} = \delta_{ij}, \; i, j = 1, \ldots, k. \tag{7}$$

The "delta" function δ_{ij} in (7) is 1 if $i = j$ and 0 otherwise, and this indicates that the direction vectors $\mathbf{v}_{(\mathrm{j})}$ are orthonormal whereby $\mathbf{V}_k^{\mathrm{T}}\mathbf{V}_k = \mathbf{I}$. This maximization problem can also be recast as the eigenvalue problem $\mathbf{C}\mathbf{v} = \lambda\mathbf{v}$ where we instead find the eigenvectors associated with the k largest eigenvalues. The score vectors are then obtained by the transformation $\mathbf{T}_k = \mathbf{X}\mathbf{V}_k$.

2.3 Maximizing Pairwise Distances

This PCA formulation is the least used but it is the one that we are the most interested in: find a k-dimensional transformation matrix that maximizes the sum of all squared pairwise distances between projected samples[21]. As before, the projected vectors are denoted as $\mathbf{T}_k$ where $\mathbf{t}_{\mathrm{i}}^{\mathrm{T}} = \mathbf{x}_{\mathrm{i}}^{\mathrm{T}}\mathbf{V}_k$. Under this framework, we find the transformation matrix $\mathbf{V}_k$ that maximizes

$$\sum_{i,j=1}^{m} ||\mathbf{t}_{\mathrm{i}} - \mathbf{t}_{\mathrm{j}}||_2^2 = \sum_{i,j=1}^{m} \left|\left|\mathbf{V}_k^{\mathrm{T}}(\mathbf{x}_{\mathrm{i}} - \mathbf{x}_{\mathrm{j}})\right|\right|_2^2. \tag{8}$$

Note that (8) is an unweighted sum of squared pairwise distances. Hence, this formulation naturally lends itself to many *weighted sum* generalizations that we will explore in Section 4.

3 Uniform Manifold Approximation and Projection

Uniform Manifold Approximation and Projection, or UMAP, corresponds to a nonlinear projection that has gained traction in the last few years with respect to visualizing data[3]. In brief, the matrix $\mathbf{X}$ gets nonlinearly mapped to a low-dimensional subspace denoted by $\mathbf{A}$:

$$\mathbf{A} = [\mathbf{a}_1, ..., \mathbf{a}_{\mathrm{m}}]^{\mathrm{T}} = [\mathbf{a}_{(1)}, ..., \mathbf{a}_{(\mathrm{k})}]. \tag{9}$$

Although we will not discuss the inner workings of the nonlinear projection machinery, an excellent discussion or "walk through" demonstration of how UMAP works can be found in [22]. In (9), each row of $\mathbf{A}$ (denoted by $\mathbf{a}_{\mathrm{i}}^{\mathrm{T}}$) indicates the ith projected sample, and each column of $\mathbf{A}$ (denoted by $\mathbf{a}_{(\mathrm{j})}$) is analogous to a "nonlinear" PC or score vector. Instead of maximizing variance, UMAP instead preserves topology, i.e., maximizing local nearest-neighbor sample associations along a manifold. Unfortunately, and as a consequence of the nonlinear optimization associated with the low-dimensional mapping, there is no linear transformation matrix that relates the impact or contribution of the ith wavelength of $\mathbf{X}$ to the jth nonlinearly mapped dimension or $\mathbf{a}_{(\mathrm{j})}$.

UMAP, like other nonlinear projection methods such as t-SNE [2], is a stochastic algorithm. Without fixing a random seed, UMAP will naturally differ between runs. Randomness is also exploited to speed up the underlying UMAP algorithm itself. Fortunately, UMAP has been shown to be relatively stable such that the the variance between runs are relatively small on average[3, 22]. All of the UMAP-derived results examined in this paper are based upon the same UMAP projection.

Figure 3 displays the two-dimensional UMAP projections for the ARLL data set. Columns 1 and 2 correspond to the full and truncated data sets, while rows 1 and 2 correspond to samples color-coded according to collection and treatment. For the UMAP projections, we observe clear sample separations with respect to collection time (early times $\{1, 2, 3\}$ versus later times $\{4, 5, 6\}$ across both the full and truncated data sets). For treatment, we also observe sample separation for both the full and truncated data sets but it is not as pronounced as the difference between early and later collection times. UMAP reveals, in two latent dimensions, clear sample distinctions across collection times and treatments. In contrast, PCA required at least three latent dimensions to reveal these separations, and even in three dimensions, the sample separations were fuzzy and overlapping. In short, UMAP compresses the ARLL data more efficiently with respect to capturing low-rank signals associated with physically relevant phenomena. However, UMAP provides no transformation matrix that could shed insight as to which wavelengths are most responsible for the two-dimensional separation of these samples.

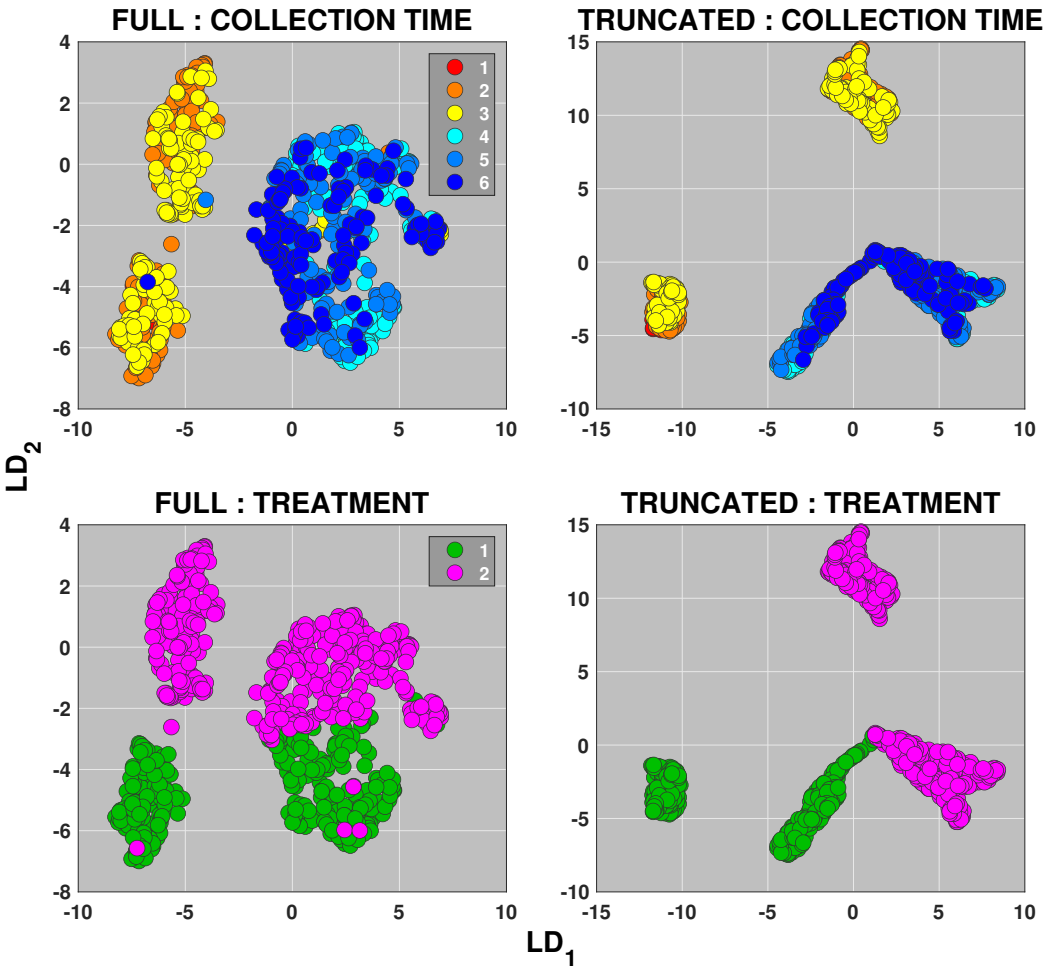

Figure 3: Two-dimensional UMAP projections associated with the full and truncated ARLL data sets. Columns 1 and 2 display the projections associated with the full and truncated ARLL data sets, respectively. Rows 1 and 2 correspond to samples that are color-coded according to collection time and treatment, respectively. For UMAP, the axes labels LD_1 and LD_2 indicate the first two latent dimensions.

4 Dissimilarity-Embedded PCA

We now want to extend the formulation of PCA in (8) such that the weighted sum will not be uniform. If $\mathbf{D}$ is a symmetric $m \times m$ matrix with non-negative weight entries d_{ij}, then the maximization of (8) can be re-written as

$$\sum_{i=1}^{m}\sum_{j=1}^{m} d_{ij}\,||\mathbf{t}_i - \mathbf{t}_j||_2^2, \quad \mathbf{D} = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & \ldots & d_{mm} \end{bmatrix} \tag{10}$$

When d_{ij} is large, the term $d_{ij}\,||\mathbf{t}_i - \mathbf{t}_j||_2^2$ will increase the overall weighted sum, and the distance between the projected samples $\mathbf{t}_i$ and $\mathbf{t}_j$ is not preserved, i.e., the projected samples are pushed further apart. When d_{ij} is small and near zero, the weighted sum will decrease, and this forces $\mathbf{t}_i \approx \mathbf{t}_j$, i.e., the projected samples are forced to be close together. As a result, the weight d_{ij} encodes *dissimilarity* prior knowledge, which is manifested by how the projected samples $\mathbf{t}_\mathrm{i}$ and $\mathbf{t}_\mathrm{j}$ are pushed or pulled apart by the value of d_{ij}.

4.1 Dual- and Primal-Constrained PCA

Suppose each projected vector $\mathbf{t}_\mathrm{i}$ is a transformation of each spectrum $\mathbf{x}_\mathrm{i}$ by a single eigenvector denoted by $\mathbf{v}$, i.e., $t_i = \mathbf{x}_\mathrm{i}^\mathrm{T}\mathbf{v}$ such that $\mathbf{t} = [t_1, \ldots, t_m]^\mathrm{T} = \mathbf{X}\mathbf{v}$. In this unidimensional case, the projected sample $\mathbf{t}_\mathrm{i}$ is in fact a scalar (i.e., t_i) and the norm $||\mathbf{t}_i - \mathbf{t}_j||_2^2$ simplifies to $(t_i - t_j)^2$. As a result, the weighted sum in (10) for the unidimensional case can be re-written (after some algebra) as follows:

$$\begin{aligned} \sum_{i=1}^{m}\sum_{j=1}^{m} d_{ij}(t_i - t_j)^2 &= 2\mathbf{t}^\mathrm{T}\mathbf{H}\mathbf{t} - 2\mathbf{t}^\mathrm{T}\mathbf{D}\mathbf{t} \qquad \left(\mathbf{H} = \mathrm{diag}(h_1, \ldots, h_m), \quad h_i = \sum_{k=1}^{m} d_{ik}\right) \\ &= 2\mathbf{t}^\mathrm{T}\mathbf{L}\mathbf{t} = 2\mathbf{v}^\mathrm{T}\mathbf{C}_\mathrm{L}\mathbf{v} \qquad \left(\mathbf{L} = \mathbf{H} - \mathbf{D}, \quad \mathbf{C}_\mathrm{L} = \mathbf{X}^\mathrm{T}\mathbf{L}\mathbf{X}\right) \end{aligned} \tag{11}$$

The matrix $\mathbf{L} = \mathbf{H} - \mathbf{D}$ is commonly referred to as a Laplacian matrix. All Laplacian matrices are characterized by 1) having zero sum for all its of rows and columns, 2) being positive-semidefinite, and 3) having non-negative diagonal entries and non-positive off-diagonal entries. The matrix $\mathbf{C}_\mathrm{L} = \mathbf{X}^\mathrm{T}\mathbf{L}\mathbf{X}$ can be interpreted as a Laplacian-weighted covariance matrix.

We now consider an expansion of (11) to multiple terms involving the k score vectors $\mathbf{t}_{(\mathrm{i})}$ (or the k eigenvectors $\mathbf{v}_{(\mathrm{i})}$):

$$f(\mathbf{t}_{(1)}, \ldots, \mathbf{t}_{(\mathrm{k})}; \mathbf{L}) = \sum_{i=1}^{k} \mathbf{t}_\mathrm{i}^\mathrm{T}\mathbf{L}\mathbf{t}_i \;=\; \sum_{i=1}^{k} \mathbf{v}_{(\mathrm{i})}^\mathrm{T}\mathbf{C}_\mathrm{L}\mathbf{v}_{(\mathrm{i})} = g(\mathbf{v}_{(1)}, \ldots, \mathbf{v}_{(\mathrm{k})}; \mathbf{L}). \tag{12}$$

To make the function notation in (12) more compact, we will re-write the arguments of f and g as follows: $f(\mathbf{t}_{(1)}, \ldots, \mathbf{t}_{(\mathrm{k})}; \mathbf{L}) = f(\mathbf{T}_k; \mathbf{L})$ and $g(\mathbf{v}_{(1)}, \ldots, \mathbf{v}_{(\mathrm{k})}; \mathbf{L}) = g(\mathbf{V}_k; \mathbf{L})$. Borrowing from terminology common in optimization, the functions f and g can be interpreted as objective functions of dual vectors (the score vectors in $\mathbf{T}_k$) and primal vectors (the eigenvectors in $\mathbf{V}_k$), respectively. Furthermore, f and g can accommodate, in principle, other $m \times m$ symmetric matrices $\mathbf{S}$ as well:

$$f(\mathbf{T}_k; \mathbf{S}) = \sum_{i=1}^{k} \mathbf{t}_\mathrm{i}^\mathrm{T}\mathbf{S}\mathbf{t}_i \;=\; \sum_{i=1}^{k} \mathbf{v}_{(\mathrm{i})}^\mathrm{T}\mathbf{C}_\mathrm{S}\mathbf{v}_{(\mathrm{i})} = g(\mathbf{V}_k; \mathbf{S}) \tag{13}$$

where $\mathbf{C}_\mathrm{S} = \mathbf{X}^\mathrm{T}\mathbf{S}\mathbf{X}$.

When maximizing the objective functions $f(\mathbf{T}_k; \mathbf{S})$ and $g(\mathbf{V}_k; \mathbf{S})$, we also desire that the direction vectors associated with the new coordinate axes in the low-dimensional subspace have unit length and are orthogonal. As a result, we now separately maximize f and g subject to orthonormality constraints on the respective dual and primal vectors:

$$\max_{\mathbf{t}_{(1)}, \ldots, \mathbf{t}_{(\mathrm{k})}} f(\mathbf{T}_k; \mathbf{S}) \quad \text{subject to} \quad \mathbf{t}_{(\mathrm{i})}^\mathrm{T}\mathbf{t}_{(\mathrm{j})} = \mathbf{v}_{(\mathrm{i})}^\mathrm{T}(\mathbf{X}^\mathrm{T}\mathbf{X})\mathbf{v}_{(\mathrm{j})} = \mathbf{v}_{(\mathrm{i})}^\mathrm{T}\mathbf{C}\mathbf{v}_{(\mathrm{j})} = \delta_{ij}, \tag{14}$$

$$\max_{\mathbf{v}_{(1)}, \ldots, \mathbf{v}_{(\mathrm{k})}} g(\mathbf{V}_k; \mathbf{S}) \quad \text{subject to} \quad \mathbf{v}_{(\mathrm{i})}^\mathrm{T}\mathbf{v}_{(\mathrm{j})} = \delta_{ij} \tag{15}$$

where $\mathbf{C} = \mathbf{X}^\mathrm{T}\mathbf{X}$ is the same covariance matrix used in (6). Recall that the delta function introduced in (7), and used again in (14) and (15), indicates orthonormality. What makes (14) and (15) numerically appealing is that they are equivalent to solving the following eigenvalue problems [21]:

$$(14) \quad \Rightarrow \quad \mathbf{C}_\mathrm{S}\mathbf{v} = \lambda\mathbf{C}\mathbf{v} \tag{16}$$

$$(15) \quad \Rightarrow \quad \mathbf{C}_\mathrm{S}\mathbf{v} = \lambda\mathbf{v}. \tag{17}$$

If $\mathbf{C}_\mathrm{S}$ is symmetric and positive (or negative) semi-definite, i.e., all eigenvalues are real and are non-negative (or non-positive), then we seek the eigenvectors associated with the k largest eigenvalues *in magnitude*. Note that the eigenvectors obtained from (16) are the result of a generalized eigenvalue problem of the matrix pair $(\mathbf{C}_\mathrm{S}, \mathbf{C})$. When $\mathbf{S}$ is the identity matrix, then (17) reduces to the ordinary PCA. The projections associated with (16) and (17) will be referred to as *Dual-Constrained PCA* and *Primal-Constrained PCA*, respectively, and will be referred to by the acronyms DC-PCA and PC-PCA.

The orthogonality condition $\mathbf{t}_{(\mathrm{i})}^\mathrm{T}\mathbf{t}_{(\mathrm{j})} = \mathbf{v}_{(\mathrm{i})}^\mathrm{T}\mathbf{C}\mathbf{v}_{(\mathrm{j})} = \delta_{ij}$ in (14) is a generalization of the standard Euclidean notion of orthogonality: two vectors $\mathbf{v}$ and $\mathbf{w}$ are defined to be orthogonal with respect to the symmetric matrix $\mathbf{B}$ if

$\mathbf{v}^{\mathrm{T}}\mathbf{B}\mathbf{w} = \mathbf{w}^{\mathrm{T}}\mathbf{B}\mathbf{v} = 0$. When $\mathbf{B} = \mathbf{I}$, we say that $\mathbf{v}$ and $\mathbf{w}$ are orthogonal with respect to the Euclidean distance metric; otherwise $\mathbf{v}$ and $\mathbf{w}$ are orthogonal via a non-Euclidean distance metric defined by $\mathbf{B}$. For our purposes, we will stick with the standard notion of orthogonality in Euclidean distances. In the context of DC-PCA, we want to emphasize that it is the score vectors $\mathbf{t}_{(\mathrm{i})}$ that are orthogonal and not the eigenvectors $\mathbf{v}_{(\mathrm{i})}$.

We now apply the DC-PCA projection to the ARLL data set. The intent is to construct a Laplacian matrix that preserves the cluster structure associated with the UMAP projection (see Figure 3). First, we define the $m \times m$ dissimilarity matrix $\mathbf{D}$ in (10) to contain the squared pairwise Euclidean distances between the UMAP-projected samples in (9), i.e., each matrix entry d_{ij} is the squared Euclidean distance between the UMAP-projected samples $\mathbf{a}_{\mathrm{i}}$ and $\mathbf{a}_{\mathrm{j}}$. Second, we compute the Laplacian matrix using (11):

$$\mathbf{L} = \mathbf{H} - \mathbf{D}, \quad \mathbf{H} = \mathrm{diag}(h_1, \ldots, h_m), \quad h_i = \sum_{j=1}^{m} d_{ij}, \quad d_{ij} = ||\mathbf{a}_{\mathrm{i}} - \mathbf{a}_{\mathrm{j}}||_2^2 \,. \tag{18}$$

The first column of Figure 4 shows the two-dimensional DC-PCA projections using this Laplacian matrix $\mathbf{L}$. In the top subplot (row 1) of column 1, we observe a temporal ordering of the samples, but unlike the UMAP projection in Figure 3, these projections do not show a clear separation between temporally-ordered samples, or separation according to treatment (row 3 of column 1). However, in rows 2 and 4 of column 1 (associated with the truncated ARLL data set), we do observe separation with respect to collection time and treatment (although the separation in treatment is not as pronounced as the UMAP projections). Like the SVD, the eigenvalue problem (or generalized eigenvalue problem in general) yields direction vectors that are non-unique with respect to reflection in the x- or y-axes, i.e., the resulting eigenvector can point in either the positive or negative eigenaxis direction. For the full ARLL data set color-coded according to treatment, the DC-PCA projection shows a clear x-axis reflection compared to the corresponding UMAP projection in Figure 3. In Table 2, the first column shows the explained variances for DC-PCA via the Laplacian embedding matrix for two PCs: 0.27% and 0.29% across the full and truncated ARLL data sets, respectively (compared to PCA with 83.9% and 58.6% across the full and truncated data sets).

Next, we will explore DC-PCA when the embedding matrix $\mathbf{S}$ is set to the matrix of squared Euclidean distances $\mathbf{D}$ in (18). In Section 4.3, we will observe that PC-PCA is in fact a special case of a more general PCA introduced by [23] and later developed by [24]. Furthermore, in Section 4.3, we will discuss the discrepancy between the explained variances of PCA and the explained variances of DC-PCA and PC-PCA in Section 4.4.

4.2 Euclidean Distances as the Dissimilarity Matrix

Perhaps a more natural choice for the embedded dissimilarity matrix $\mathbf{S}$ would be to only use the Euclidean distance matrix $\mathbf{D}$ instead of the Laplacian matrix $\mathbf{L} = \mathbf{H} - \mathbf{D}$ used in (18). If so, we would instead solve the generalized eigenvalue problem for the matrix pair $(\mathbf{C}_{\mathrm{D}}, \mathbf{C})$ where

$$\mathbf{C}_{\mathrm{D}}\mathbf{v} = \lambda\mathbf{C}\mathbf{v}, \quad \mathbf{C}_{\mathrm{D}} = \mathbf{X}^{\mathrm{T}}\mathbf{D}\mathbf{X}. \tag{19}$$

However, a Euclidean distance matrix has different properties than that of a Laplacian matrix, and is characterized by the following: 1) symmetric with trace (or sum of eigenvalues) equal to zero, and 2) has rank m with exactly one positive eigenvalue while the remaining $m-1$ eigenvalues are negative[25, 26]. But for purposes of solving (19), the more meaningful question is the following: instead of the matrix properties of $\mathbf{D}$, what are the matrix properties of the weighted covariance matrix $\mathbf{C}_{\mathrm{D}}$?

Any matrix of squared Euclidean distances can be re-written in terms of the original data matrix and its sample norms[27]. Recall that in our case, the Euclidean distance matrix $\mathbf{D}$ is based upon the UMAP-projected samples in $\mathbf{A}$:

$$\mathbf{q} = [\,||\mathbf{a}_1||_2^2, \ldots, ||\mathbf{a}_{\mathrm{m}}||_2^2\,]^{\mathrm{T}}, \quad \mathbf{D} = \mathbf{q}\mathbf{1}_{\mathrm{m}}^{\mathrm{T}} + \mathbf{1}_{\mathrm{m}}\mathbf{q}^{\mathrm{T}} - 2\mathbf{A}\mathbf{A}^{\mathrm{T}}. \tag{20}$$

Note that the ones vector $\mathbf{1}_{\mathrm{m}}$ is in the null space of the mean-centered spectra such that $\mathbf{X}^{\mathrm{T}}\mathbf{1}_m = \mathbf{0}_m$. When we substitute the three-term expansion of $\mathbf{D}$ in (20) into the weighted covariance matrix $\mathbf{C}_{\mathrm{D}} = \mathbf{X}^{\mathrm{T}}\mathbf{D}\mathbf{X}$, the weighted

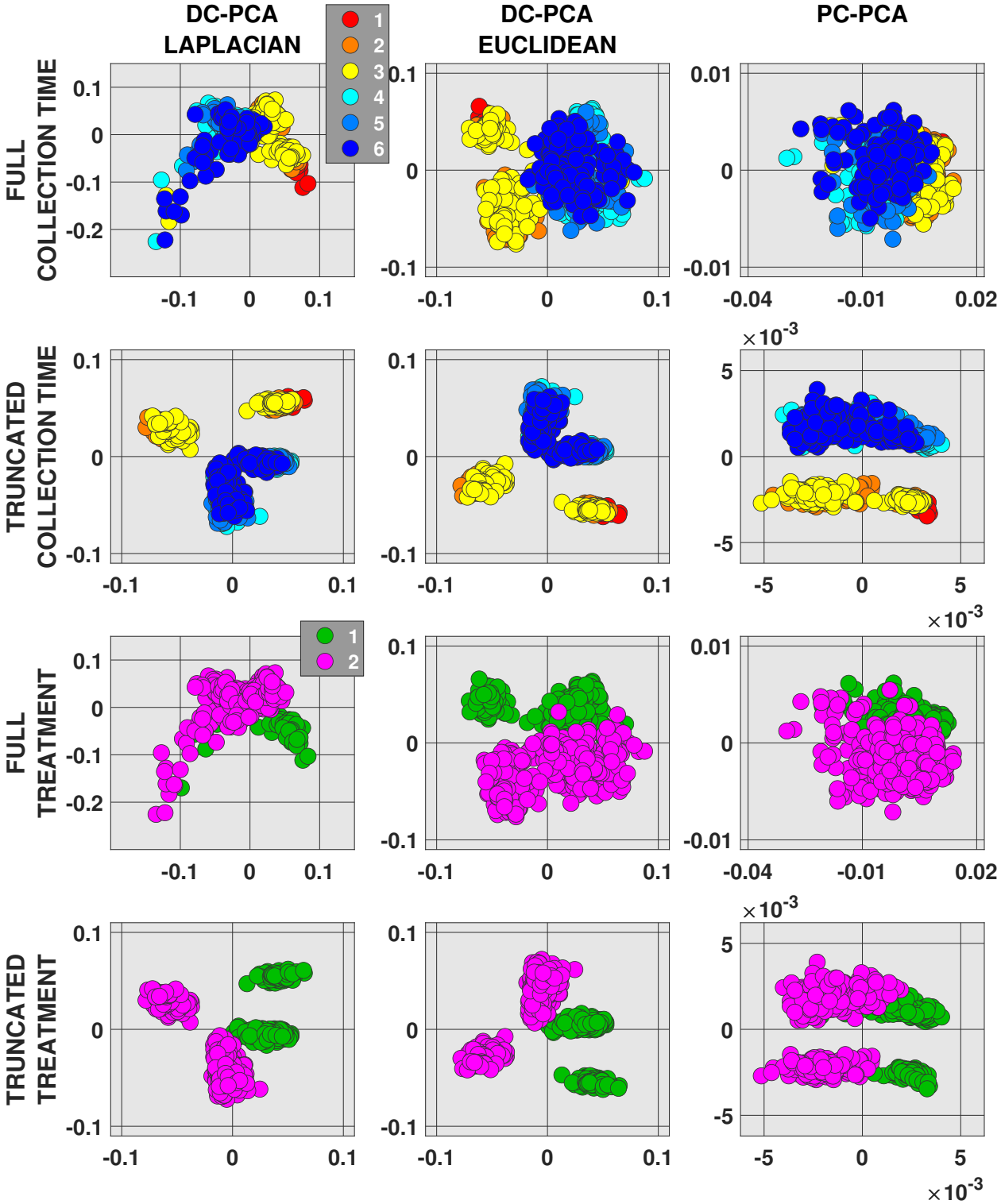

Figure 4: DC-PCA via Laplacian and Euclidean matrices, and PC-PCA applied to the ARLL data sets. Columns 1, 2 and 3 correspond to DC-PCA via the Laplacian embedding matrix, DC-PCA via the Euclidean distance matrix and PC-PCA, respectively. The top two rows and the last two rows correspond to the samples color-coded according to collection time and treatment, respectively. Rows 1 and 3 are associated with the full ARLL data set while rows 2 and 4 are associated with the truncated data set. The explained variances are shown in Table 2. The x- and y-axes in each subplot correspond to PC 1 and PC 2, respectively.

covariance matrix is found to be negative semi-definite (using the property that $\mathbf{v}\mathbf{C}_{\mathrm{D}}\mathbf{v} \leq 0$ for all vectors $\mathbf{v}$):

$$\begin{aligned}
\mathbf{v}^{\mathrm{T}}\mathbf{C}_{\mathrm{D}}\mathbf{v} &= \mathbf{v}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{D}\mathbf{X}\mathbf{v} \\
&= \mathbf{v}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\left(\mathbf{q}\mathbf{1}_{\mathrm{m}}^{\mathrm{T}} + \mathbf{1}_{\mathrm{m}}\mathbf{q}^{\mathrm{T}} - 2\mathbf{A}\mathbf{A}^{\mathrm{T}}\right)\mathbf{X}\mathbf{v} \\
&= \mathbf{v}^{\mathrm{T}}\left[\,(\mathbf{X}^{\mathrm{T}}\mathbf{q})(\mathbf{1}_{\mathrm{m}}^{\mathrm{T}}\mathbf{X}) + (\mathbf{X}^{\mathrm{T}}\mathbf{1}_{\mathrm{m}})(\mathbf{q}^{\mathrm{T}}\mathbf{X}) - 2(\mathbf{X}^{\mathrm{T}}\mathbf{A})(\mathbf{A}^{\mathrm{T}}\mathbf{X})\,\right]\mathbf{v} \\
&= -2\mathbf{v}^{\mathrm{T}}(\mathbf{X}^{\mathrm{T}}\mathbf{A})(\mathbf{A}^{\mathrm{T}}\mathbf{X})\mathbf{v} \\
&= -2\left|\left|(\mathbf{A}^{\mathrm{T}}\mathbf{X})\mathbf{v}\right|\right|_2^2 \leq 0
\end{aligned} \tag{21}$$

Since $\mathbf{C}_{\mathrm{D}}$ is negative semi-definite, all of its eigenvalues are non-positive, and as a result, we seek the eigenvectors in (19) associated with the largest k eigenvalues in magnitude or absolute value. As with the DC-PCA approach via the Laplacian matrix, the resulting eigenvectors (the primal vectors) obtained here are also not orthogonal.

We should note that it is not strictly necessary to work with the Euclidean matrix $\mathbf{D}$ as an embedding dissimilarity matrix. In some applications, it is preferable to work with a similarity matrix instead. With this in mind, and if we assume that the projected samples $\mathbf{A}$ of UMAP have been mean-centered, then we can convert $\mathbf{D} = \mathbf{q}\mathbf{1}_{\mathrm{m}}^{\mathrm{T}} + \mathbf{1}_{\mathrm{m}}\mathbf{q}^{\mathrm{T}} - 2\mathbf{A}\mathbf{A}^{\mathrm{T}}$ in (20) to a kernel similarity matrix $\mathbf{K}$ via the following double centering technique [25, 27]:

$$\mathbf{K} = -\frac{1}{2}\mathbf{J}\mathbf{D}\mathbf{J} = -\frac{1}{2}\mathbf{J}(\mathbf{q}\mathbf{1}_{\mathrm{m}}^{\mathrm{T}} + \mathbf{1}_{\mathrm{m}}\mathbf{q}^{\mathrm{T}} - 2\mathbf{A}\mathbf{A}^{\mathrm{T}})\mathbf{J} = \mathbf{A}\mathbf{A}^{\mathrm{T}} \tag{22}$$

where $\mathbf{J}$ is the mean-centering matrix used in (1). To see this, first note that, for any matrix $\mathbf{M}$, $\mathbf{J}\mathbf{M}\mathbf{J}$ centers the rows and columns to have mean $\mathbf{0}$. Consequently, $\mathbf{J}(\mathbf{q}\mathbf{1}^{\mathrm{T}})\mathbf{J} = \mathbf{J}(\mathbf{1}\mathbf{q}^{\mathrm{T}})\mathbf{J} = \mathbf{0}$ since the rows of $\mathbf{q}\mathbf{1}^{\mathrm{T}}$ and columns of $\mathbf{1}\mathbf{q}^{\mathrm{T}}$ are constant. If we replace $\mathbf{D}$ with $\mathbf{K} = \mathbf{A}\mathbf{A}^{\mathrm{T}}$ as the embedding matrix in (19), then $\mathbf{C}_{\mathrm{K}} = \mathbf{X}^{\mathrm{T}}\mathbf{K}\mathbf{X}$ is symmetric positive semi-definite. The resulting generalized eigenvalues of the matrix pair $(\mathbf{C}_{\mathrm{K}}, \mathbf{C})$ will be different from the generalized eigenvalues of $(\mathbf{C}_{\mathrm{D}}, \mathbf{C})$ but the eigenvectors will be the same (except for a ± 1 multiplicative factor associated with the non-uniqueness of eigenaxis directions).

The second column of Figure 4 illustrates the two-dimensional DC-PCA projections via the Euclidean matrix approach. In contrast to DC-PCA via the Laplacian matrix, DC-PCA via the Euclidean matrix strongly preserves the UMAP-based topology of the ARLL data set. Moreover, there is an increased separation of samples due to treatment compared with PCA in three dimensions and UMAP in two dimensions (see Figures 2 and 3). As with UMAP, the effect of including the large absorption band for the full ARLL data set clearly inhibits the ability of DC-PCA to fully separate the samples. Now that we appear to have an approach that preserves the UMAP pairwise distance structure, we will later examine the behavior of the corresponding eigenvectors in Section 5. Although DC-PCA via the Euclidean embedding matrix has superior topological preservation than DC-PCA via the Laplacian matrix, the two-dimensional explained variances for the full and truncated ARLL data sets are marginal at best—see column 2 of Table 2.

4.3 Generalized Matrix Decomposition and PC-PCA

There are recent generalizations of PCA that do consider structural dependencies [4, 23, 24, 28]). We focus on the most related generalization based upon the Generalized Matrix Decomposition (GMD). It was first developed by [23] and later expanded by [24]. Let $\mathbf{S}$ and $\mathbf{F}$ be two positive semi-definite matrices of size $m \times m$ and $n \times n$, respectively. GMD finds the direction vectors that solve the following optimization problem:

$$\max_{\mathbf{v}_{(1)},\ldots,\mathbf{v}_{(\mathrm{k})}} \sum_{j=1}^{k} \mathbf{v}_{(\mathrm{j})}^{\mathrm{T}}\mathbf{F}\mathbf{C}_{\mathrm{S}}\mathbf{F}\mathbf{v}_{(\mathrm{j})} \quad \text{subject to} \quad \mathbf{v}_{(\mathrm{i})}^{\mathrm{T}}\mathbf{F}\mathbf{v}_{(\mathrm{j})} = \delta_{ij}, i, j = 1, \ldots, k. \tag{23}$$

where $\mathbf{C}_{\mathrm{S}} = \mathbf{X}^{\mathrm{T}}\mathbf{S}\mathbf{X}$ and the ith PC is given by $\mathbf{t}_{(\mathrm{i})} = \mathbf{X}\mathbf{F}\mathbf{v}_{(\mathrm{i})}$. When $\mathbf{S} = \mathbf{I}$ and $\mathbf{F} = \mathbf{I}$, GMD reduces to the ordinary PCA. Although [24] uses the power method as the eigensolver, it is illustrative to recast (23) as a generalized eigenvalue problem[1]:

$$(\mathbf{F}\mathbf{C}_{\mathrm{S}}\mathbf{F})\mathbf{v} = \lambda\mathbf{F}\mathbf{v}. \tag{24}$$

[1]Although the power method has many desirable advantages, e.g., easy to implement and computationally lightweight, it is not recommended for large number of PCs k unless re-orthogonalization is performed to ensure that the current eigenvector $\mathbf{v}_{(\mathrm{k})}$ remains orthogonal to previous eigenvectors.

Of particular interest is when the feature similarity matrix $\mathbf{F}$ is set to the identity matrix. When $\mathbf{F} = \mathbf{I}$, the generalized eigenvalue problem in (24) reduces to the PC-PCA in (17). Moreover, we are interested in the PC-PCA case when we set both $\mathbf{S} = \mathbf{D}$ and $\mathbf{F} = \mathbf{I}$ where $\mathbf{D}$ is the matrix containing all squared pairwise distances between the UMAP-projected samples in $\mathbf{A}$.

The third column of Figure 4 illustrates the PC-PCA projections via embedded Euclidean distances (i.e., (24) with $\mathbf{S} = \mathbf{D}$) on the ARLL data sets. Compared with DC-PCA via embedded Euclidean distances, PC-PCA does not achieve an appreciable separation of samples on the full ARLL data set with respect to collection times and treatment. However, on the truncated ARLL data set, PC-PCA does show sample separation between early and later collection times. In the presence of the large water absorption band, PC-PCA has a more difficult time showing phenomenologically meaningful clusters of a physical nature compared to the truncated data set. Recall that the crucial projection difference between DC-PCA and PC-PCA is the difference in the orthogonality constraints: we either enforce that the dual (score) vectors are orthogonal, or we enforce that the primal (eigenvectors) are orthogonal. Compared to DC-PCA, the explained variances are significantly higher for PC-PCA—see column 3 of Table 2. However, when we compare the PC-PCA projections in Figure 4 and the PCA projection in Figure 2 using only the first and third PCs, there is not much qualitative difference between PC-PCA and PCA. (The second PC for PCA has little or no variation with respect to ARLL sample separation.) Compared to PCA, PC-PCA does have *slightly* more collection time separation (early collection times $\{1, 2, 3\}$) versus later collection times $\{4, 5, 6\}$) in the direction of the first PC.

EXPLAINED VARIANCE			
	DC-PCA (Laplacian)	DC-PCA (Euclidean)	PC-PCA
FULL	0.3%	1.5%	58.8%
TRUNCATED	0.4%	2.2%	49.0%

Table 2: The two- and three-dimensional explained variance associated with DC-PCA and PC-PCA.

4.4 Approximation Error and Topological Preservation

In the discussion so far regarding DC-PCA and PC-PCA projections, we have not defined a metric which measures how close these projections preserve or reconstruct the UMAP sample topology. Our discussion in this regard has been intentionally qualitative as opposed to quantitative, to reproduce the sample-to-sample associations revealed by UMAP. Historically, the concept of explained variance in the case of PCA has been deemed the gold standard by which one judges how close or true a low-rank approximation is to the original data. For example, in the case of PCA, the low-rank approximation is typically expressed in terms of scores and loadings: $\mathbf{X}^{\text{approx}} = \mathbf{T}_k\mathbf{V}_k^{\text{T}}$, or $\mathbf{X}^{\text{approx}} = \mathbf{T}_k(\mathbf{V}_k^{\text{T}}\mathbf{V}_k)^{-1}\mathbf{V}_k^{\text{T}}$ when $\mathbf{V}_k$ is not orthonormal[29]. The explained variance associated with a low-rank approximation is often described via a Frobenius norm:

$$\text{expvar} = \left(1 - \frac{||\mathbf{X} - \mathbf{X}^{\text{approx}}||_{\text{F}}^2}{||\mathbf{X}||_{\text{F}}^2}\right) \times 100\% \quad \text{where} \quad ||\mathbf{X}||_{\text{F}}^2 = \sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}^2. \tag{25}$$

By its very construction, PCA will always be best with respect to minimizing this norm in an unsupervised manner. Traditionally, one computes explained variance using (25). However, this approach is based upon using Euclidean distances $||\mathbf{x}||_2^2 = \mathbf{x}^{\text{T}}\mathbf{x}$ as opposed to non-Euclidean distances $\mathbf{x}^{\text{T}}\mathbf{S}\mathbf{x}$ where $\mathbf{S}$ is the embedding matrix. For consistency, we will use (25) as the default metric by which we compute explained variance across all approaches.[2]

For the full and truncated ARLL data sets, and for the first two PCs, the explained variances for DC-PCA and PC-PCA are significantly lower than the explained variances of PCA. Based upon the small values of explained variance, both DC-PCA and PC-PCA would appear to convey little trust and are akin to visualizing noise or inchoate blobs.

[2]For a comparison, the two-dimensional explained variances on the full ARLL data set for PC-PCA using Euclidean and non-Euclidean distances (where $\mathbf{S} = \mathbf{D}$) is 58.9% and 4.4%, respectively.

However, the interesting thing is the following: it is the *unexplained variance* that is also very relevant. As both UMAP and DC-PCA (and PC-PCA to a lesser extent) clearly demonstrate, PCA does a relatively poor job of compressing the relevant low-rank signals in the first two latent dimensions. Hence, the classical description of explained variance is ill-equipped to describe topological approximation via nearest-neighbor embedding. Although topological preservation metrics are not uncommon in disciplines like computer vision—see [30] and references therein, they are not common in chemometrics. Much as explained variance is the companion error metric to PCA, new accuracy or companion metrics are likely needed in chemometrics for topologically-minded projection methods, especially as data sets get larger and more locally structured.

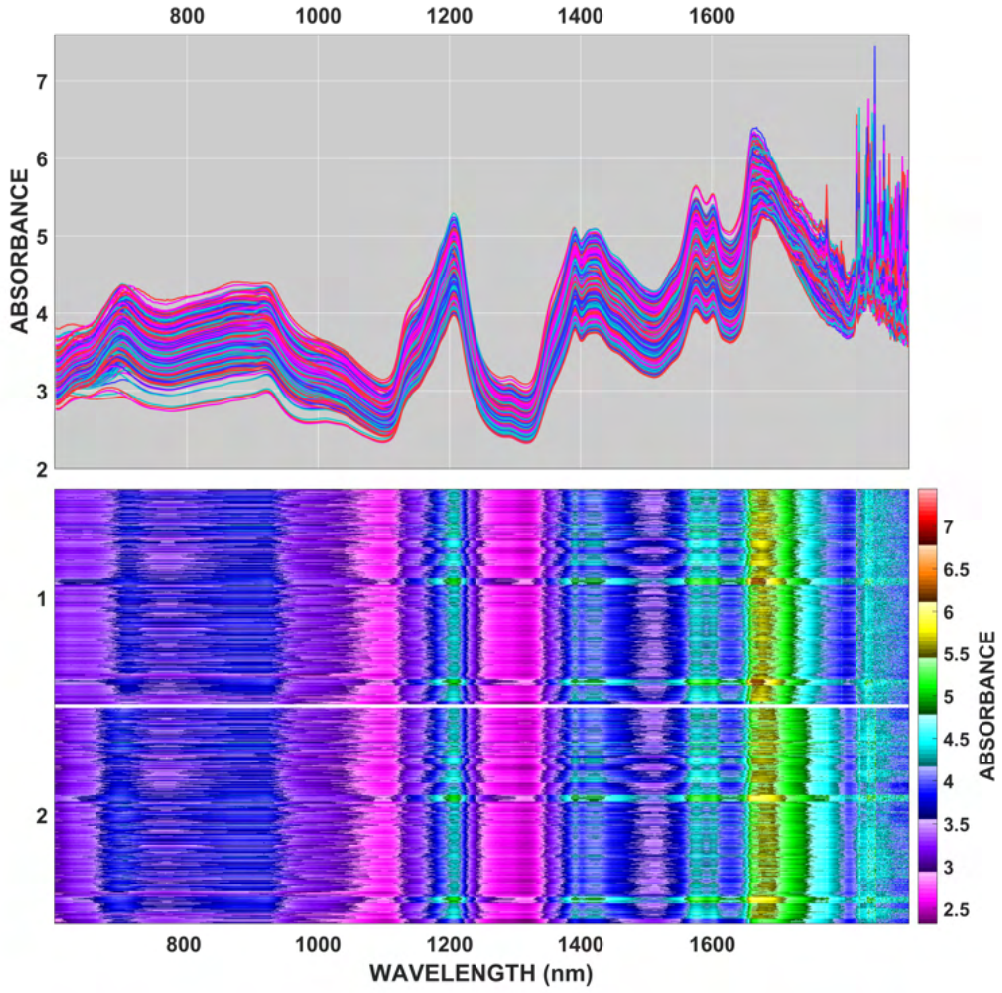

Figure 5: Spectra associated with the Tablet data set. The top subplot displays the spectra—one curve for each spectrum. The bottom subplot displays absorbance as a heatmap. The samples in the heatmap are grouped according to spectrometer (1 and 2).

5 Projections and Associated Eigenvectors

We introduce another data set known as Tablet. In 2002, the International Diffuse Reflectance Conference published a "Shootout" data set consisting of spectra from 655 pharmaceutical tablets measured on two separate spectrometers using a range of 600-1898nm at 2nm intervals for a total of 650 wavelengths[31]. Combining the data from both spectrometers, we have 1310 samples (655 samples from each instrument). Our goal is to assess whether PCA,

UMAP, PC-PCA and DC-PCA can find meaningful differences in the spectra between the two spectrometers via their respective projections. Figure 5 displays the absorbance spectra. The top subplot shows the absorbance waveforms as a function of wavelength—one curve per spectrum. The bottom subplot shows the absorbance values as a heatmap. The samples are grouped according to spectrometer. The differences between spectrometers occur mainly at two locations: the wavelength band around 700nm and the wavelength band around 1700nm.

5.1 Projections on the Tablet Data Set

Figure 6 illustrates the two-dimensional projections associated with PCA, UMAP, DC-PCA and PC-PCA on the Tablet data set. The first row corresponds to PCA. Column 1 is a three-dimensional PCA projection, with columns 2, 3, and 4 displaying all two-dimensional PC combinations. The second row corresponds to the projections associated wit UMAP, DC-PCA and PC-PCA. The samples are color-coded according to spectrometer (1 or 2). Both DC-PCA and PC-PCA use the squared Euclidean distances as the embedding matrix derived from the corresponding UMAP projections. For PCA, it takes the inclusion of the third latent dimension to show separation between spectrometer membership. Both DC-PCA and PC-PCA show separation across spectrometers using only two latent dimensions.

The DC-PCA projection for the Tablet data set is a faithful reproduction of UMAP (modulo reflection) but perhaps it is too faithful of a reproduction. In the next subsection, we will next examine the two leading eigenvectors of both the ARLL and Tablet data sets and the linear projection methods across PCA, DC-PCA and PC-PCA, paying attention to the amount of oscillation (or parsimony) across wavelengths.

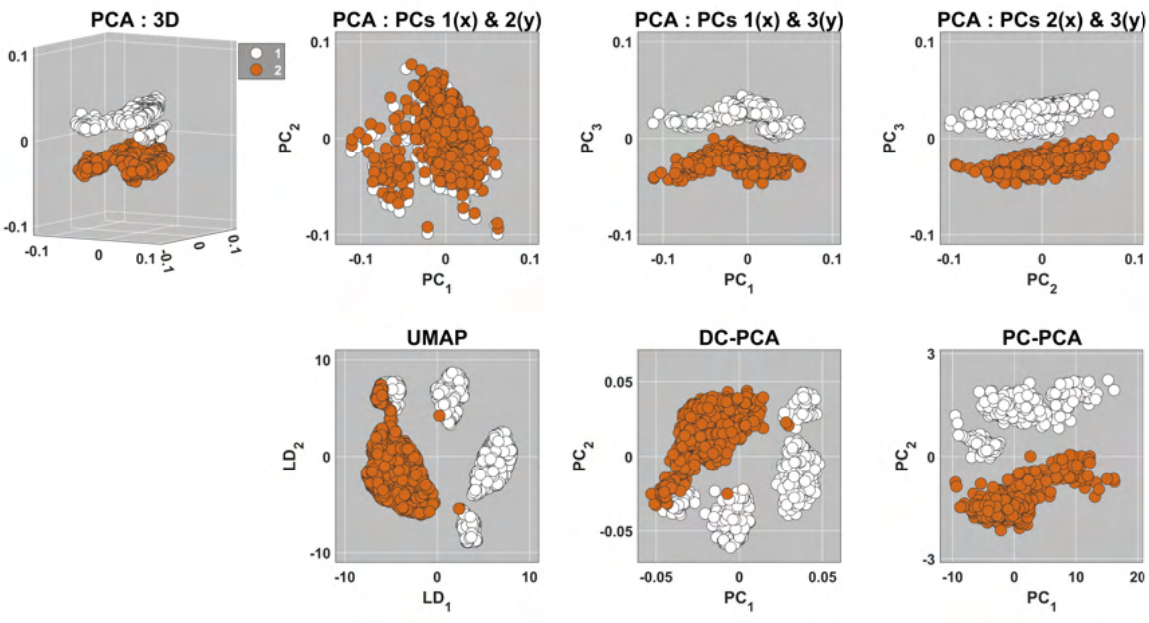

Figure 6: Projections associated with Tablet data set across various projection methods. With the exception of the subplot at the upper left (which is a three-dimensional projection), two-dimensional projections associated with PCA, UMAP, DC-PCA and PC-PCA are displayed. The first row corresponds to PCA, with columns 2, 3, and 4 displaying all two-dimensional PC combinations. The second row corresponds to the UMAP, DC-PCA (with the Euclidean embedding matrix) and PC-PCA. The samples are color-coded according to spectrometer (1 or 2).

5.2 Eigenvectors of ARLL and Tablet

For each data set, Figure 7 displays the eigenvectors associated PCA, DC-PCA and PC-PCA, and the correlation between eigenvectors. Columns 1 through 4 correspond to PCA, DC-PCA and PC-PCA, respectively. Rows 1 through 3, rows 4 through 6, and rows 7 through 9 correspond to the ARLL (full), ARLL (truncated) and Tablet data sets. For each data set, there is a heatmap that displays correlation coefficients between each of the three PCA eigenvectors and each of the two eigenvectors of DC-PCA and PC-PCA. We will highlight two aspects of eigenvector behavior that stand out: 1) parsimony, and in particular, the high-frequency oscillations associated with DC-PCA eigenvectors, and 2) correlation, and in particular, the correlation of PC-PCA eigenvectors with certain PCA eigenvectors.

The first two DC-PCA eigenvectors exhibit large amounts of high-frequency oscillations, i.e. there are more alternating sign changes in the eigenvector elements v_{ij} compared to those observed with PCA and PC-PCA. In short, the eigenvectors of DC-PCA are much less parsimonious than those of PC-PCA and PCA. In contrast, for PCA, one would typically observe this amount of high-frequency oscillation only for eigenvectors associated with PCs that are large in number. (And, for PCA, the corresponding high-frequency eigenvectors would be associated with PCs with a small amount of explained variance.)

From an interpretation perspective, we have a tension between an ingrained desirability for eigenvector smoothness (like those obtained by PCA and PC-PCA) and the topological explainability of non-smooth eigenvectors (like those obtained by DC-PCA). The most likely explanation of the non-smoothness phenomena is the difference in the orthogonality constraints, i.e., orthogonality in non-Euclidean distances for DC-PCA versus orthogonality in Euclidean distances for PC-PCA. Each constraint in the Euclidean space ($\mathbf{v}_{(\mathrm{i})}^{\mathrm{T}}\mathbf{v}_{(\mathrm{j})} = \delta$) involves the sum of n terms (n is the number of wavelengths) while each constraint in the non-Euclidean space ($\mathbf{v}_{(\mathrm{i})}^{\mathrm{T}}\mathbf{C}\mathbf{v}_{(\mathrm{j})} = \delta$) involves n^2 terms. For example, in the case of the full ARLL data set, the orthogonality constraints for PC-PCA and DC-PCA involve the sum of $n = 1153$ and $n^2 = 1153^2$ terms, respectively. As a result, the size of the possibility space that can satisfy the non-Euclidean orthogonality constraints of DC-PCA is much, much larger than the possibility space associated with PC-PCA. In short, parsimony of eigenvector shape for the first few PCs is not explicitly encoded by DC-PCA. But that is not the purpose of the UMAP-infused DC-PCA scheme. The purpose is to optimize an embedding in a low-dimensional space that preserves nearest-neighbor sample structure. Although the lack of shape parsimony for DC-PCA may be a "bridge too far" for some, simply having the eigenvectors themselves provides more information than it is possible to obtain from UMAP alone. For example, for the full ARLL data set in Figure 4, the sample collection times (early versus later) for DC-PCA are separated along the first PC. Hence, the wavelengths associated with the largest components in magnitude for the first eigenvector would be the most responsible for collection time separation. Likewise, the wavelengths associated with the largest components in magnitude for the second eigenvector would be the most responsible for samples separated by treatment.

The second aspect of eigenvector behavior that stands out is the correlation between certain eigenvectors of PCA and PC-PCA. For the full ARLL data set, the eigenvectors associated with the first and third PCs of PCA are highly correlated with the eigenvectors associated with first and second PCs of PC-PCA, respectively; see the heatmap of eigenvector correlations in the third row in Figure 7. For the truncated ARLL data set, the eigenvectors associated with the second and third PCs of PCA are highly correlated (positive and negative) with the eigenvectors associated with the second and first PCs of PC-PCA, respectively; see the heatmap of eigenvector correlations in the sixth row in Figure 7. (The highly negative correlation is due to the non-uniqueness of eigenaxis direction.) A similar story also holds for the Tablet data set—the eigenvectors associated with the first and third PCs of PCA are highly correlated (negative and positive) with the eigenvectors of the first and second PCs of PC-PCA, respectively; see the heatmap in the ninth row of Figure 7. In effect, PC-PCA is functionally equivalent to principal component selection, i.e., PC-PCA removes a PC that it does not need to encode and preserve topological information (the second PC in the case of full ARLL and Tablet data sets, and the first PC in the case of truncated ARLL data set). From an interpretation perspective, the close connection between the parsimonious eigenvectors of PCA and PC-PCA is likely to be deemed highly desirable. However, as evidenced by the projections in Figure 6, the striving for parsimony in eigenvector shape in PC-PCA is done at the expense of preserving UMAP-based topological structure.

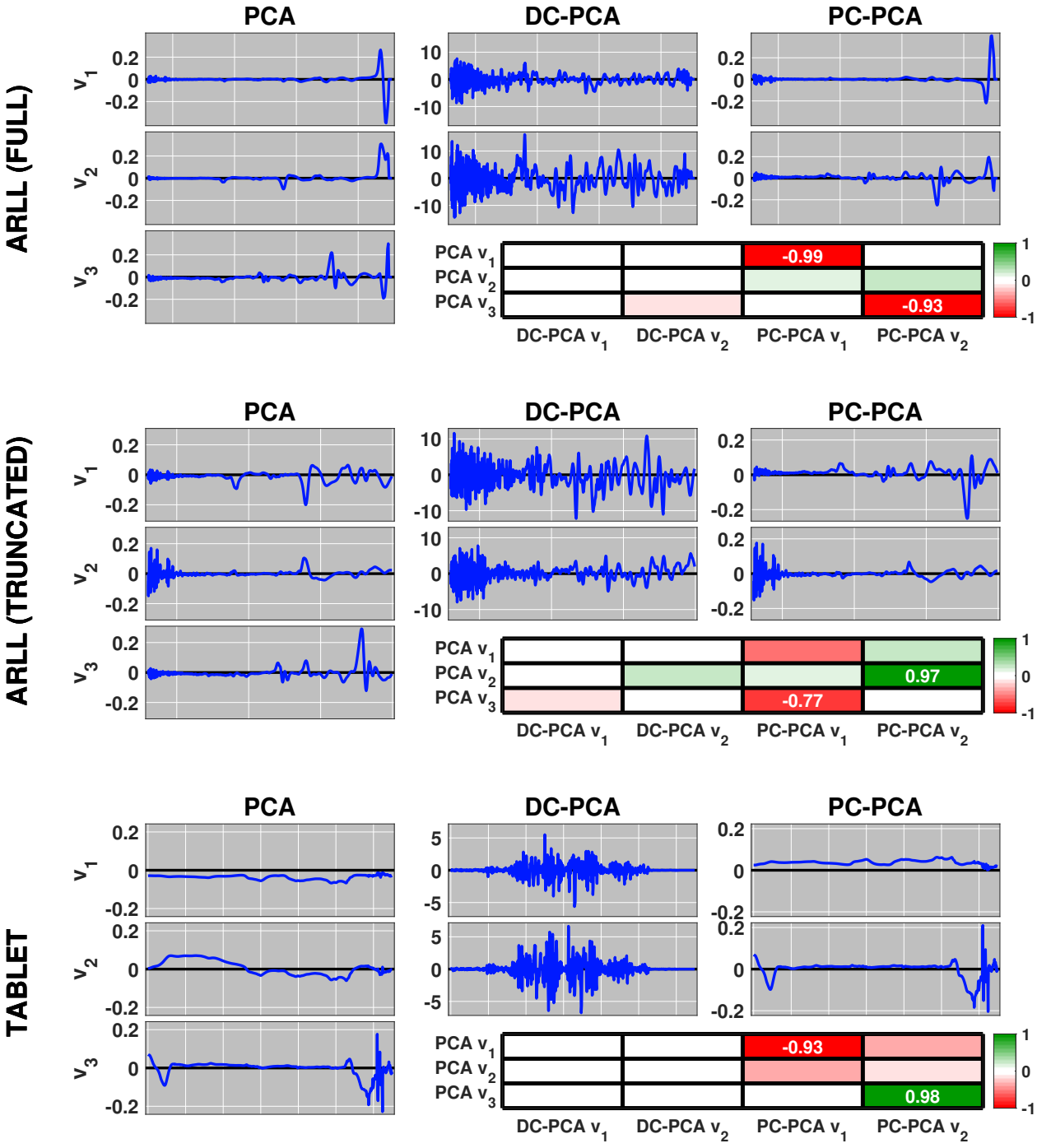

Figure 7: Eigenvector plots and correlation coefficients for PCA, DC-PCA and PC-PCA. Rows 1 through 3, rows 4 through 6, and rows 7 through 9 correspond to the full ARLL, truncated ARLL and Tablet data sets. Columns 1 through 3 correspond to PCA, DC-PCA and PC-PCA, respectively. Each curve in blue is associated with one of the eigenvectors ($\mathbf{v}_1$, $\mathbf{v}_2$ and $\mathbf{v}_3$). Associated with each data set is a heatmap (rows 3, 6 and 9): a correlation coefficient is computed for each of the three PCA eigenvectors against each of the first two eigenvectors of DC-PCA and PC-PCA.

5.3 Projection of New Samples

DC-PCA and PC-PCA prove useful for preserving local nearest-neighbor associations and generating visualizations for the ARLL and Tablet data sets. However, if one wants to make these projections more useful for multivariate calibration purposes, then it is important to create a calibration model and then apply that model to new spectra (e.g., as one would do via cross-validation). If we want to use DC-PCA and PC-PCA to learn a latent subspace, then we should be able to easily project new samples onto that subspace such that the new samples appear in locations close to their respective members in the calibration set. Fortunately, DC-PCA and PC-PCA make this easily possible via a simple post-multiplication by the eigenvector matrix.

UMAP and other nonlinear projections can also map new samples to their respective latent spaces, but the nonlinear and non-parametric transformation processes that act on these new samples can be time-consuming. Recently, though, a parametric version of UMAP has been created, and this has a beneficial side-effect of accelerating the nonlinear projection of new samples onto the UMAP latent space[32]. As mentioned before, UMAP proceeds in two steps. First, UMAP constructs a graph of local relationships between data points. Second, it then optimizes an embedding in a low-dimensional space that preserves the structure of the graph. The parametric UMAP approach replaces the second step of this process with an optimization of parameters from of a deep neural network. However, the parametric UMAP, like its non-parametric counterpart, still does not provide information that relates the importance of a given wavelength on a sample spectrum.

The protocol for creating the calibration set of samples (denoted as CAL, i.e., the training set) and the validation set of samples (denoted as VAL, i.e., the test set) is as follows. The ARLL samples within each of the six sample collection levels are split into two groups. We use a 90%:10% split for each level: 90% of the within-level samples belong to the CAL subset while the remaining 10% are set aside for the VAL subset. Across the levels, all of the CAL subsets are pooled together while the new samples consist of the pooled VAL subsets. The DC-PCA and PC-PCA models are all derived from the pooled CAL samples and are comprised on the following: 1) the mean spectrum, 2) the embedding matrix of squared Euclidean distances between UMAP projected samples, and 3) the eigenvectors for the first two PCs. The pooled VAL spectra are then mean-centered with respect to the CAL spectral mean and post-multiplied by the CAL-derived eigenvector matrix.

For the Tablet data set, we also follow a (90%,10%) split: With each spectrometer, 90% of the samples belong to the CAL subset while the remaining 10% are set aside for the VAL subset. Across the two spectrometers, the CAL subsets are pooled together and the new samples consist of the remaining pooled VAL subsets. The DC-PCA and PC-PCA models are created from the pooled CAL samples, and the CAL and VAL samples are projected using their respective DC-PCA and PC-PCA models.

Figure 8 displays the projections for both the calibration and validation sets. The VAL samples are displayed as thick-bordered triangles. Rows 1 and 2, rows 3 and 4 and row 5 correspond to the full ARLL, the truncated ARLL, and the Tablet data sets. Row 2 is the same as row 1 (and row 4 is the same as row 3) except that the ARLL samples have been color-coded according to sample treatment. Both DC-PCA and PC-PCA are both effective at learning a latent subspace, and projecting new samples into that subspace. All projections show new samples from the VAL set being projected into locations that are co-localized with the CAL set (with the possible exception of PC-PCA on the full ARLL data set since PC-PCA did not show pronounced separation of samples in this case). Across the data sets, DC-PCA learns a latent model that generalizes well to new samples, and this is in spite of the non-parsimonious and oscillatory nature of the DC-PCA eigenvectors. This stands in contrast to the perspective of ordinary PCA where the inclusion of highly oscillatory eigenvectors is deemed undesirable since these eigenvectors are associated with overfitting. Given that the DC-PCA models do generalize well to new samples, we surmise that these eigenvectors are capturing real information regarding low-rank signals associated with physically relevant phenomena (i.e., differences across collection times, treatments, or spectrometers). Moreover, this capturing by DC-PCA of low-rank signals is likely attributable to the accommodation of nearest-neighbor structure (via the embedding of pairwise distances associated with UMAP-projected samples). Given the quite parsimonious eigenvector behavior of PC-PCA in Figure 7, it is encouraging that there is as much unsupervised separation between samples of different type as there is. (And this separation is achieved by effectively removing the first or second PC, a PC that is deemed influential by PCA).

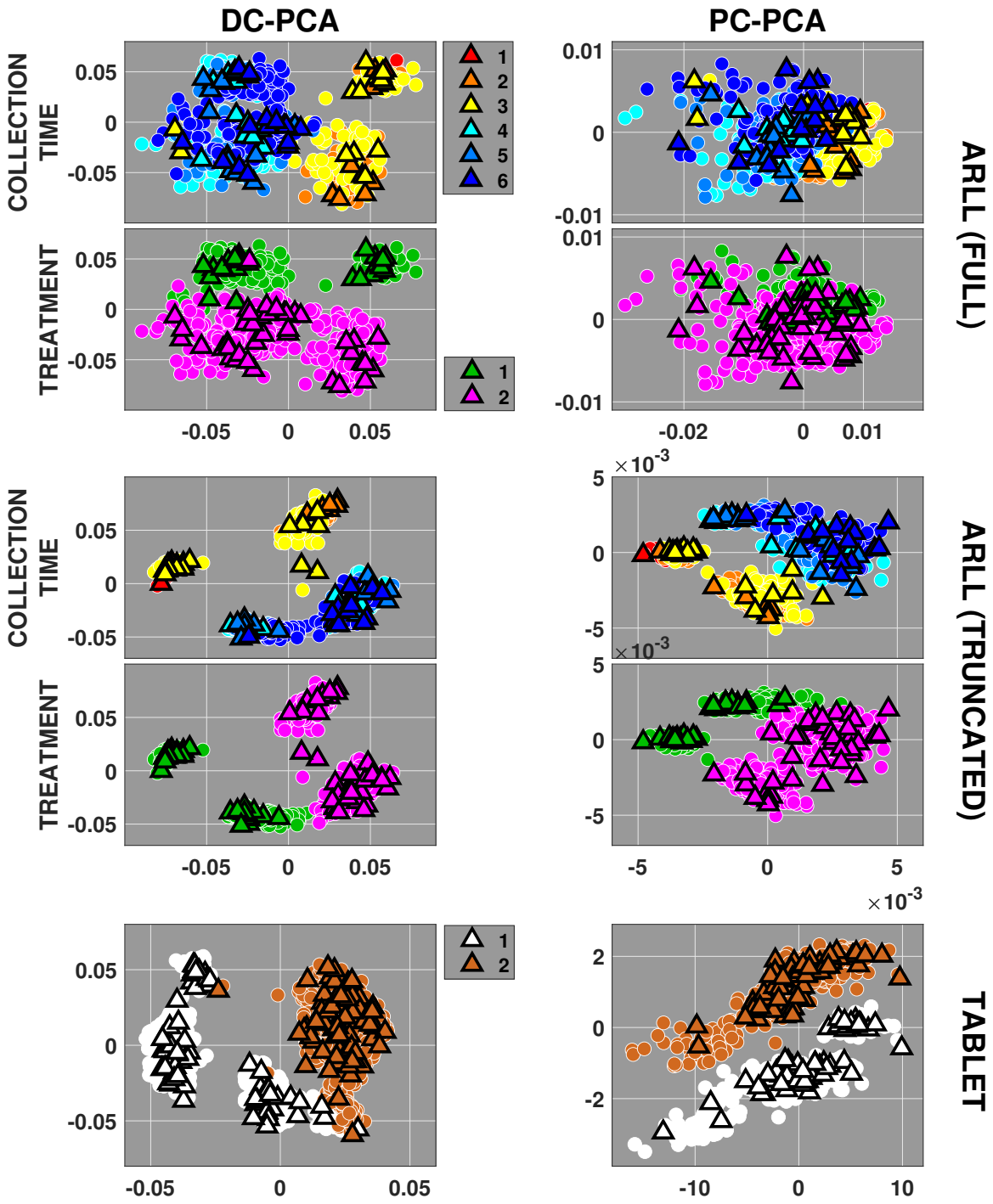

Figure 8: DC-PCA and PC-PCA models applied to new samples. The full ARLL, the truncated ARLL and Tablet data sets correspond to rows 1 and 2, rows 3 and 4, and row 5, respectively. The first and second columns correspond to the DC-PCA and PC-PCA projections, respectively. For each factor (e.g., collection times, treatments or spectrometers), the samples within each factor level are split into two groups: 90% (CAL) and 10% (VAL). The DC-PCA and PC-PCA models are constructed from the pooled CAL subset of samples across all factor levels, and the new samples consist of the pooled VAL samples across all factor levels. Both the pooled CAL and VAL samples are projected using only the CAL models for DC-PCA and PC-PCA. The projected CAL are displayed as circles, while the the projected VAL samples are displayed as thick-bordered triangles.

5.4 Computational Considerations

The UMAP implementation used in this paper is from an implementation developed in the Herzenberg Lab at Stanford University[33]. The code is written in MATLAB with external program interfaces (which compiles and links C++ source files), and is a faithful rewrite (except for the nearest neighbor search) of the original Python implementation from [3]. This MATLAB/C++ results in superior speed-up performance over the default Python implementation. We used the default settings.

The primary purpose of the UMAP-embedded PCA variants mentioned in this paper is to provide eigenvectors: eigenvectors for elucidating the relationship between wavelength and sample location, eigenvectors that make it easy to project new samples onto a learned latent space, and eigenvectors that encode nearest-neighbor structure. However, these eigenvectors can numerically be provided in a number of ways. This paper proposes the following: perform UMAP to acquire pairwise Euclidean distances of projected samples, followed by a generalized eigenvalue problem. If one wants to accelerate PC-PCA or DC-PCA, then one could use a power method approach—as was mentioned in Section 4.3 with the GMD scheme of [24]. However, the computational bottleneck for DC-PCA and PC-PCA will still be the initial UMAP computation. Although UMAP is reasonably fast, it is not yet applicable to massive data sets. For context, MNIST, a classic toy data set in the machine learning community—70000 samples by 784 features—can be solved in minutes using the classical UMAP implementation [34]. As nonlinear topological projection methods such as t-SNE and UMAP continue to evolve, their computational speed will become faster—much in the same way that classical SVD and PCA continue to evolve with respect to computational speed-up (e.g., from deterministic PCA to its stochastic variants involving randomized projections).

6 Conclusion and Future Work

In this paper, we have successfully embedded externally-supplied topological information via UMAP (as a dissimilarity weighted covariance matrix) into PCA. One modified PCA model—DC-PCA–mimics the nearest-neighbor structure revealed by UMAP, while the other modified PCA—PC-PCA—more closely mimics PCA (with either the first and second PC of PCA being suppressed). Both DC-PCA and PC-PCA create models that allow for the projection of new samples into learned latent subspaces such that new samples are projected in locations that are co-localized with their calibration sample counterparts.

The intent in this paper was modest—to use UMAP as an external source of a priori domain information that encodes local nearest-neighbor sample associations. This encoding was accomplished by computing the squared Euclidean distances between all projected UMAP samples, and embedding this Euclidean distance matrix into PCA as a weighted covariance matrix, which was subsequently maximized subject to orthonormality constraints on the score or loading vectors. However, our unsupervised exploration of embedded matrices was limited and additional unsupervised exploration for the enhancement and detection of low-rank signals is warranted. For example, regression applications using non-Euclidean embedding matrices have shown success with the analysis of microbiome data involving nuclear magnetic resonance data[4, 35]. In the context of calibration transfer and maintenance, non-Euclidean Laplacian matrices were custom built for pulling together samples from different instruments into a common calibration domain[17].

Ideally, a careful design of experiment imbues the covariance matrix $\mathbf{C} = \mathbf{X}^{\mathrm{T}}\mathbf{X}$ with information about where and how often measurements were made, and what the standard deviations of those measurements were. However, the embedding of a dissimilarity matrix $\mathbf{S}$ into $\mathbf{C} = \mathbf{X}^{\mathrm{T}}\mathbf{X}$ cannot be done in an arbitrary or ad hoc fashion, otherwise such information will be lost or not be fully utilized. Numerically, the spectral properties of $\mathbf{S}$ have to be sufficiently understood such that the resulting eigenvalue problems involving $\mathbf{C}_{\mathrm{S}}$ have desirable properties, e.g., positive (or negative) semi-definiteness. Informationally, both $\mathbf{S}$ and the spectra $\mathbf{X}$ should be complementary yet *co-informative*, e.g., both are informative for clustering samples but from different view points. In the statistical and machine learning community, measures such as the Hilbert-Schmidt Information Criterion (HSIC; [36]) are commonly used for measuring the co-informativeness (or conversely, statistical independence) of two data sets. In chemometrics, such criteria are derived mostly from the data fusion community where one extracts common and distinct subspace

information—see [37] and references therein for a recent summary of approaches. In the future, a combination of approaches related to both HSIC and data fusion could be used to assess whether one should embed a dissimilarity matrix or not, and if so, measure approximately the information gained from such an embedding. This assessment via some embedding fitness criterion is likely to be application dependent, though. For example, embedding a Laplacian matrix based upon UMAP projected distances did not work well (enough) in this paper, but using embedded Laplacian matrices for calibration transfer purposes did work well in [15, 17, 18].

As mentioned in Section 5.2, one may want to mitigate the amount of high-frequency oscillation in the DC-PCA eigenvector profiles for the first two PCs. To rectify this, one could imagine a homotopy approach where we continuously vary between DC-PCA and PC-PCA:

$$\mathbf{C}_{\mathrm{D}}\mathbf{v} = \lambda \mathbf{C}_{\tau}\mathbf{v} \quad \text{where} \quad \mathbf{C}_{\tau} = (1-t)\mathbf{C} + \tau\mathbf{I}, \; 0 \leq \tau \leq 1. \tag{26}$$

As τ increases from 0 to 1, (26) transitions from DC-PCA to PC-PCA. Instead of a homotopy, another potential remedy would be to create a compact or sparse approximation to the matrix $\mathbf{C}$. By radically reducing the number of non-zero elements in $\mathbf{C}$, one could also reduce the number of terms used to satisfy the orthogonality constraints $\mathbf{v}_{(\mathrm{i})}^{\mathrm{T}}\mathbf{C}\mathbf{v}_{(\mathrm{j})} = \delta_{ij}$ associated with DC-PCA. These and other remedies would also go a long way to addressing a more fundamental question: is eigenvector smoothness compatible with the preservation of nearest-neighbor sample structure?

In Section 4.4, better metrics for computing the approximation error between the original data $\mathbf{X}$ and the low-rank reconstruction are warranted. Explained variance via Euclidean distances are well understood, but explained variance in non-Euclidean distances are not as well-understood as they could be for a general chemometric audience. Furthermore, the importing of alternative metrics outside of chemometrics that quantify topological preservation as opposed to variance preservation requires further research effort.

A follow-up numerical assessment of UMAP, DC-PCA, PC-PCA and PCA is also warranted. Such an assessment would explore a wide range of parameter settings. For example, varying the number of nearest neighbors (15 nearest neighbors was the default) used in UMAP results in a trade-off between prioritizing global and local structure. As the number of nearest neighbors increases, UMAP connects more and more samples when constructing the graph representation of the high-dimensional data, which in turn leads to a projection that more accurately reflects the global structure of the data. Moreover, one can also explore the following: a wider range of spectroscopic sets (not just the two near infrared data sets used in this paper), a wider range of data sizes within a data set (e.g., 20%, 40%, 60%, 80% and 100% of samples). All of the these assessments would be numerically timed to see where computational bottlenecks occur.

Acknowledgements

The first author kindly acknowledges Scott Anthony Parsons from Queensland, Australia regarding additional details on the Australian Rainforest Leaf Litter data set. The first author would also like to acknowledge Timothy Randolph at the Fred Hutchinson Cancer Research Center in Seattle regarding discussions about embedded similarity matrices for unsupervised and supervised data exploratory purposes. The second author acknowledges funding from the Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK), the Federal Ministry for Digital and Economic Affairs (BMDW), and the Province of Upper Austria in the frame of the COMET - Competence Centers for Excellent Technologies program managed by the Austrian Research Promotion Agency FFG, the COMET Center CHASE and the FFG project Interpretable and Interactive Transfer Learning in Process Analytical Technology (Grant No. 883856).

Bibliography

[1] B.P. Rogers et al. "Functional Connectivity in the Human Brain by fMRI". *Magn Reson Imaging* 25 (2007), pp. 1347–1357.

[2] L. van der Maaten and G. Hinton. "Vizualizing Data Using t-SNE". *J Mach Learn Res* 9 (2008), pp. 2579–2605.

[3] L. McInnes, J. Healy, and J. Melville. "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction". *arXiv1802.03426* (2018). Accessed March 23, 2022, pp. 2579–2605. URL: https://arxiv.org/abs/1802.03426.

[4] Y. Wang et al. "The Generalized Matrix Decomposition Biplot and Its Application to Microbiome Data". *mSystems* (2019), 4:e00504–19. URL: https://doi.org/10.1128/mSystems.00504-19.

[5] S. Wold. "Exponentially weighted moving principal components analysis and projections to latent structures". *Chemometr Intell Lab* 23 (1994), pp. 149–161.

[6] P.D. Wentzell et al. "Maximum likelihood principal component analysis". *J Chemometr* 11 (1997), pp. 339–366.

[7] J.A. Westerhuis et al. "Grey component analysis". *J Chemometr* 21 (2007), pp. 474–485. DOI: 10.1002/cem.1072.

[8] K. Van Deun et al. "Weighted sparse principal component analysis". *Chemometr Intell Lab* 195 (2019). DOI: 10.1016/j.chemolab.2019.103875.

[9] X. He and P. Niyogi. "Locality preserving projections". *Adv Neur In* 16 (2003), pp. 153–160.

[10] L. Luo et al. "Nonlocal and local structure preserving projection and its application to fault detection". *Chemometr and Intell Lab* 157 (2016), pp. 177–188.

[11] J Wang, B. Zhong, and J.L. Zhou. "Quality-Relevant Fault Monitoring Based on Locality-Preserving Partial Least-Squares Statistical Models". *Ind Eng Chem Res* 56.24 (2017), pp. 7009–7020. DOI: 10.1021/acs.iecr.7b00248.

[12] M. Sugiyama. "Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis". *J Mach Learn Res* 8 (2007), pp. 1027–1061.

[13] M. Sugiyama. "Semi-supervised local Fisher discriminant analysis for dimensionality reduction". *Mach Learn* 78.1-2 (2010), pp. 35–61.

[14] M. Sugiyama et al. "Information-maximization clustering based on squared-loss mutual information". *Neural Comput* 26.1 (2014), pp. 84–131.

[15] R. Nikzad-Langerodi et al. "Domain-invariant partial-least-squares regression". *Anal Chem* 90.11 (2018), pp. 6693–6701.

[16] G. Huang et al. "Domain adaptive partial least squares regression". *Chemometr Intell Lab* 201 (2020), p. 103986.

[17] R. Nikzad-Langerodi and E. Andries. "A chemometrician's guide to transfer learning". *J Chemometr* (2021), e3373. DOI: https://doi.org/10.1002/cem.3373.

[18] R. Nikzad-Langerodi and F. Sobieczky. "Graph-based calibration transfer". *J Chemometr* 35.4 (2021), e3319.

[19] S. Parsons. *Near Infrared Spectra of Australian Rainforest leaf litter (decomposition).* Accessed March 23, 2022. 2011. URL: https://research.jcu.edu.au/data/published/0ec8d54906187ecc7561f5fb0be82324.

[20] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. 2004, pp. 75–84.

[21] Y. Koren and L. Carmel. "Visualization of labeled data using linear transformations". *IEEE Infor Vis* (2003).

[22] L. McInnes. *How UMAP Works*. Accessed March 23, 2022. 2018. URL: https://umap-learn.readthedocs.io/en/latest/how_umap_works.html.

[23] Y. Escoufier. "Operator related to a data matrix: a survey". *Compstat 2006 - Proceedings in Computational Statistics*. Ed. by A. Rizzi and M. Vichi. Springer-Verlag, Berlin, Germany, 2006, pp. 285–297.

[24] G.I. Allen, L. Grosenick, and J. Taylor. "A generalized least-square matrix decomposition". *J Am Stat Assoc* 109 (2014), pp. 145–159. URL: https://doi.org/10.1080/01621459.2013.852978.

[25] J. Gower. "Properties of Euclidean and non-Euclidean distance matrices". *Linear Algebra Appl* 67 (1985), pp. 81–97.

[26] E. Dempsey. "Inverse Eigenvalue Problem for Euclidean Distance Matrices". MA thesis. Auburn University, 2014.

[27] P. Diaconis, S. Goel, and S. Holmes. "Horseshoes in multidimensional scaling and local kernel methods". *Ann Appl Stat* 2.3 (2008), pp. 777–807. DOI: 10.1214/08-AOAS165.

[28] G. Satten et al. "Restoring the duality between principal components of a distance matrix and linear combinations of predictors, with application to studies of the microbiome." *PLoS One* (2017), 12:e0168131. URL: https://doi.org/10.1371/journal.pone.0168131.

[29] J. Camacho et al. "All Sparse PCA Models Are Wrong, But Some Are Useful. Part I: Computation of Scores, Residuals and Explained Variance". *arXiv1907.03989* (2019).

[30] R. Karbauskaitė and G. Dzemyda. "Topology Preservation Measures in the Visualization of Manifold-Type Multidimensional Data". *Informatica* 20.2 (2009), pp. 235–254.

[31] IDRC. "International Diffuse Reflectance Conference (IDRC) Shootout" (2002). Accessed March 23, 2022. URL: https://eigenvector.com/resources/data-sets.

[32] T. Sainburg, L. McInnes, and T.Q. Gentner. "Parametric UMAP Embeddings for Representation and Semisupervised Learning". *Neural Comput* 33 (2021), pp. 2881–2907. DOI: 10.1162/neco_a_01434.

[33] C. Meehan et al. *MATLAB Central File Exchange: Uniform Manifold Approximation and Projection (UMAP)*. Accessed March 23, 2022. 2022. URL: https://www.mathworks.com/matlabcentral/fileexchange/71902.

[34] L. McInnes. *How UMAP Works*. Accessed March 23, 2022. 2018. URL: https://umap-learn.readthedocs.io/en/latest/precomputed_k-nn.html.

[35] T. W. Randolph et al. "Kernel-Penalized Regression for Analysis of Microbiome Data". *ArXiv1511.002972v2* (2017).

[36] A. Gretton et al. "Measuring statistical dependence with Hilbert-Schmidt norms". *International Conference on Algorithmic Learning Theory ALT'05*. Springer-Verlag, Berlin, Germany, 2005, pp. 63–77.

[37] A.K. Smilde et al. "Common and distinct components in data fusion". *J Chemometr* 31 (2017), e2900.