Cluster analysis of soft X-ray spectromicroscopy data

M. Lerotic\textsuperscript{a,}\textsuperscript{*}, C. Jacobsen\textsuperscript{a}, T. Schäfer\textsuperscript{b}, S. Vogt\textsuperscript{c}

\textsuperscript{a}Department of Physics and Astronomy, State University of New York at Stony Brook, Stony Brook, NY 11794-3800, USA
\textsuperscript{b}Forschungszentrum Karlsruhe, INE, P.O. Box 3640, 76021 Karlsruhe, Germany
\textsuperscript{c}Argonne National Laboratory, Argonne, IL 60439-4836, USA

Received 13 August 2003; received in revised form 6 January 2004; accepted 26 January 2004

Abstract

Soft X-ray spectromicroscopy provides spectral data on the chemical speciation of light elements at sub-100 nm spatial resolution. When all chemical species in a specimen are known and separately characterized, existing approaches can be used to measure the concentration of each component at each pixel. In other cases (such as often occur in biology or environmental science), some spectral signatures may not be known in advance so other approaches must be used. We describe here an approach that uses principal component analysis to orthogonalize and noise-filter spectromicroscopy data. We then use cluster analysis (a form of unsupervised pattern matching) to classify pixels according to spectral similarity, to extract representative, cluster-averaged spectra with good signal-to-noise ratio, and to obtain gradations of concentration of these representative spectra at each pixel. The method is illustrated with a simulated data set of organic compounds, and a mixture of lutetium in hematite used to understand colloidal transport properties of radionuclides.

\textcopyright 2004 Elsevier B.V. All rights reserved.

PACS: 07.05.Kf; 07.85.Tt; 61.10.Ht; 78.70.Dm

Keywords: X-ray microscopy; X-ray spectromicroscopy; Principal component analysis; Cluster analysis

1. Introduction

Spectromicroscopy is a powerful tool which provides a way to see chemical speciation with the spatial resolution of a microscope. It can be carried out using photon absorption (from X rays to the infrared), X-ray fluorescence excitation, or electron energy loss, especially when plural inelastic scattering can be ignored. In studies of specimens where only a few chemical species are present (such as polymer blends), the data can be interpreted straightforwardly using reference spectra of pure components. In other situations such as in biology or environmental science, this approach may not be possible due to compositional complexity.

We describe here a method to find natural groupings of data without prior knowledge of the spectra of all components \cite{1}. Drawing upon prior discussions of multivariate statistical analyses of energy loss electron microscopy data \cite{2,3}, we
describe a particular approach that provides experimentally useful information for X-ray spectromicroscopy. We use principal component analysis [4] to orthogonalize spectromicroscopy data, and discard much of the noise present in the data. We then use cluster analysis or unsupervised pattern matching [5] to classify pixels according to the similarity of their spectra, and then recover gradations of thicknesses of representative materials using these observable spectra. This approach can nicely visualize nanoscale speciation in complex specimens.

1.1. Data sets

In order to illustrate clustering methods, we will make use of two soft X-ray spectromicroscopy data sets. One is an experimental data set acquired at the oxygen K absorption edge using the Stony Brook STXM IV microscope [6,7] at the beamline X1A2 at the National Synchrotron Light Source. The other is a simulated data set, using experimentally determined carbon K edge spectra of several amino acids [8]. The experimental data are acquired as a series of transmission images in \((x, y)\) at nearby photon energies \(E\) [9], which provides the same \((x, y, E)\) data as would be obtained by acquiring a series of spectra at adjacent pixels.

The experimental data come from the application of soft X-ray spectromicroscopy to studies in environmental science. It is drawn from a study of lutetium structural incorporation in hematite, which has already been intensively characterized including Lu L-edge EXAFS and colloid migration studies [10–12]. Lutetium is commonly used as lanthanide homologue for the trivalent actinide americium. The understanding of Am(III) crystal structure entrapment and the maximum incorporation capacity in either stationary (e.g., canister corrosion) or mobile (e.g., colloidal transport at large distances to the “far field”) iron phases is of paramount importance to the reliable prediction of radionuclide mobility in deep geological nuclear waste repositories. Studies on the oxygen K-edge pre-peak intensity, attributed to the unoccupied bands of primary O(2p)–Fe(3d) character, have shown that this pre-peak can be directly correlated with the proportion of Fe–O–Fe bonds or Fe substitution present in the mineral structure [13]. This pre-peak feature shows in addition a splitting (with absorption peaks at 530.0 and 531.7 eV) due to a crystal-field-induced difference between the energy levels of the \(t_{2g}\) and \(e_g\) orbitals. To investigate the maximum trivalent lanthanide/actinide entrapment capacity and possible recrystallization kinetics which would remove structurally incorporated Lu out of the hematite lattice, samples were prepared by crystallizing various mixtures (0–100 mol% Lu per mol Fe) precipitates of ferrihydrite according to the hematite synthesis conditions described by Schwertmann et al. [14]. Only the data of 5 mol% Lu per mol Fe substituted hematite will be discussed within this paper. The washed and freeze dried sample was re-suspended in purified, deionized water and directly dried on a Si3N4 window for measurement. This preparation procedure produced a sample with variations in both thickness and composition.

In order to explore the characteristics of the method in an example where the specimen composition and component spectra are known in advance, we will also use a simulated data set. This was formed by using experimentally-measured absorption spectra of pure thin films of collagen and the amino acids leucine and tyrosine [8; see Fig. 1]. With the exception of a 10 × 10 pixel hole at the lower left corner, the specimen (in a 128 × 128 pixel array, giving \(P = 16,384\) pixels) was assumed to have a uniform thickness of 200 nm: Most of the specimen was assumed to be 100% collagen; however, specific regions (in the shape of letters) were assumed to have different compositions. The letters A, B, and C were given a composition of 90%, 50%, and 10% leucine, respectively, and the letters D, E, and F were given a composition of 90%, 50%, and 10% tyrosine, respectively, with collagen making up the rest of the composition in each case. This artificial specimen was then “illuminated” with 1000 photons per pixel at each of 133 photon energies evenly spaced between 282 and 302 eV, corresponding to the near-edge absorption region of carbon. At each pixel and energy, the square root of the “transmitted” photons was multiplied by a normally distributed random number (Gaussian
distribution with a mean of zero and a standard deviation of one) and added to the transmitted photons (with the result constrained to be \(\geq 0\)) to approximate the Poisson statistics of photon counting.

2. Analysis of X-ray spectromicroscopy data

When an X-ray flux \(I_0(E)\) is incident on a thin film with thickness \(t\), the transmitted flux \(I(E)\) is given by the Lambert–Beer law as

\[
I(E) = I_0(E) \exp[-\mu(E)t].
\]

The linear absorption coefficient \(\mu(E)\) can be written as

\[
\mu(E) = 2r_e \frac{hc}{E} n_a f_2(E),
\]

where \(f_2\) represents the complex number of effective electrons per atom \([f_1(E) + if_2(E)]\) [15]. In this expression, \(r_e = 2.818 \times 10^{-15}\) m is the classical radius of the electron, and \(hc = 1239.852\) eV nm. The number density of atoms \(n_a = (\rho N_A)/M\) is found from the mass density \(\rho\), Avogadro’s number \(N_A\), and molar mass \(M\); compound mixtures can be represented by calculating an element-weighted average of the product \(n_a f_2(E)\). We define optical density \(D(E)\) of the film to be

\[
D(E) = -\ln \left( \frac{I(E)}{I_0(E)} \right) = \mu(E)t
\]

so that we can obtain the thickness from \(t = D(E)/\mu(E)\) if \(\mu(E)\) is known.

2.1. Matrix treatment of spectromicroscopy

Spectromicroscopy data consist of a series of energy spectra at adjacent positions forming an image [16], or a “stack” of images [9,17] over a series of spectroscopically interesting energies. From these data indexed at \(n = 1, \ldots, N\) energies, we form a data matrix \(D_{N \times P}\) with columns indexed by \(p = 1, \ldots, P\) for pixels, which in our case correspond to image positions as \(p = i_{\text{col}} + (i_{\text{row}} - 1)m_{\text{rows}}\) where \(i_{\text{col}}\) and \(i_{\text{rows}}\) are both indexed from a starting value of 1. If we have \(s = 1, \ldots, S\) spectroscopically distinct components in the specimen, we can express the optical density at particular energy \(n\) and pixel \(p\) summed over all components as the sum of the thicknesses \(t_{sp}\) of all components \(s\) at the pixels \(p\), and the spectra \(\mu_{ns}\) of all components \(s\) at the energies \(n\), as

\[
D_{np} = \mu_{n1} t_{1p} + \mu_{n2} t_{2p} + \cdots + \mu_{ns} t_{sp} = \sum_{s=1}^{S} \mu_{ns} t_{sp}
\]

or, for all \(N\) energies indexing rows and \(P\) pixels indexing columns in matrix notation,

\[
D_{N \times P} = \mu_{N \times S} \cdot t_{S \times P}.
\]

If we know the set of exact absorption spectra \(\mu_{N \times S}\) for all of the \(s = 1 \ldots S\) known components, we can calculate spatially-resolved thickness maps \(t_{S \times P}\) of the components by matrix inversion of Eq. (5) as

\[
t_{S \times P} = (\mu_{N \times S})^{-1} \cdot D_{N \times P}.
\]
The inversion of the matrix $\mu_{N \times S}$ can be accomplished using singular value decomposition (see Appendix A), leading to quantitative maps of specimen composition [18,19]. Equivalent results have also been obtained using curve-fitting methods to obtain thickness maps based on known spectra (A. Hitchcock, personal communication).

2.2. Principal component analysis

In many cases, particularly in biology or environmental science, the specimen cannot be assumed to be made up of a simple combination of a limited number of components for which reference spectra are known a priori. One approach to handle these cases involves the use of principal component analysis (PCA) to characterize the data set in terms of its most significant variations without prior knowledge of their characteristics. From its origin in the social sciences, it has been used extensively in chemistry and, more recently, in X-ray absorption spectroscopy [20], electron energy-loss spectrum imaging [3], and X-ray spectromicroscopy [21,22].

The goal in PCA is to describe the specimen by a set of $s = 1, \ldots, S_{\text{abstract}}$ abstract components (where $S_{\text{abstract}} \ll N$) [4]. These abstract components describe the main spectroscopic signatures in the data; each signature may in fact arise from a linear combination of several different chemical species, so that there is not a simple, direct relationship between one particular abstract component and one particular chemical component of the specimen. As a result, in PCA we seek to characterize the specimen not in terms of known spectra $\mu_{N \times S_{\text{physical}}}$ and thicknesses $t_{S \times P}$, but in terms of column and row matrices

$$D_{N \times P} = C_{N \times S_{\text{abstract}}} \cdot R_{S_{\text{abstract}} \times P},$$

where the column matrix $C_{N \times S_{\text{abstract}}}$ contains in each column a spectrum (with $N$ points) of one of the $S_{\text{abstract}}$ components, while the row matrix $R_{S_{\text{abstract}} \times P}$ contains in each row an image (with $P$ pixels) of one of the $S_{\text{abstract}}$ components.

One method for calculating the column matrix $C_{N \times S_{\text{abstract}}}$ is to use the covariance of the data (singular value decomposition can also be used, as described in Appendix B, but at a much greater cost in terms of computer storage and calculation time). One can calculate either a spectral covariance $Z_{N \times N}$ or a spatial covariance $Z_{P \times P}$ (see Appendix C); we describe here the calculation based on spectral covariance $Z_{N \times N}$ which is preferred when $N < P$. The spectral covariance matrix is formed from the data matrix as

$$Z_{N \times N} = D_{N \times P} \cdot D_{N \times P}^\top,$$

so that it measures the correlation between images at various energies. Because the correlation of the image at energy $n_1$ with the image at energy $n_2$ is the same as the correlation of $n_2$ with $n_1$, the covariance matrix $Z_{N \times N}$ is symmetric. We then wish to find the eigenvectors (which we will henceforth call eigenspectra) and eigenvalues $\lambda(s)$ that fully span the covariance matrix:

$$Z_{N \times N} \cdot C_{N \times S_{\text{abstract}}} = C_{N \times S_{\text{abstract}}} \cdot A_{N \times N},$$

where $S_{\text{abstract}} = N$, and $A_{N \times N}$ is a diagonal matrix whose diagonal elements are given by the eigenvalues $\lambda(s)$ for $s = 1, \ldots, N$. Columns of the eigenspectra matrix $C_{N \times S_{\text{abstract}}}$ consist of the $N$ eigenspectra sorted in the order of decreasing magnitude of corresponding eigenvalues $\lambda(s)$, as anticipated by Eq. (7), with $S_{\text{abstract}} = N$. We can also find a corresponding eigenimage matrix $R_{S_{\text{abstract}} \times P}$ from

$$R_{S_{\text{abstract}} \times P} = C_{S_{\text{abstract}} \times N}^\top \cdot D_{N \times P},$$

where we have used the fact that $C$ is orthogonal (being composed of eigenvectors) so that its inverse is its transpose, $C^{-1} = C^\top$. It should be emphasized that the eigenspectra $C_{N \times S_{\text{abstract}}}$ and eigenimages $R_{S_{\text{abstract}} \times P}$ are calculated directly from the data with no prior assumptions.

Examination of the eigenspectra and eigenimages by themselves can provide considerable insight into the data [21,22]. As an example, the first few eigenspectra of the lutetium/hematite data are shown in Fig. 2. The first eigenspectrum is essentially an average of the spectra at all $P$ pixels, so it looks like a recognizable X-ray absorption spectrum, and the first eigenimage shows an average of all $N$ images corresponding roughly to a thickness map of the specimen. The second eigenspectrum gives the first correction to that average, and the third eigenspectrum gives the next...
When reproducing the covariance matrix, $Z_{N \times N}$ are shown. Of the eigenspectra, also for a restricted 525–550 eV energy range with random fluctuations from one energy point where the eigenspectra show increasingly shade of grey exactly halfway between black and white. Black corresponds to maximum negative values, and zero is the scale where white corresponds to maximum positive values, positive and negative values; they are shown here on a grey energy range. It should be noted that the eigenimages can have variations in noise from pixel to pixel. The eigenvalues of the data until eigenspectra 5 and 6 begin to represent average that are required to represent the spectra present in the subsequent eigenspectra represent successive corrections to this average that are required to represent the spectra present in the pixels of the data until eigenspectra 5 and 6 begin to represent mostly variations in noise from pixel to pixel. The eigenvalues $\lambda(s)$ of Eq. (9) shown in (B), which multiply the eigenspectra when reproducing the covariance matrix $Z_{N \times N}$, also indicate the decreasing significance of successive components. The eigenimages $R_{s \times P}$ (from Eq. (10)) shown in (C) go from showing something that is close to total thickness at $s = 1$ to only noise at $s = 6$ (full energy range) or $s = 5$ (restricted energy range). It should be noted that the eigenimages can have positive and negative values; they are shown here on a grey scale where white corresponds to maximum positive values, black corresponds to maximum negative values, and zero is the shade of grey exactly halfway between black and white.

Correction to the first two, and so on, so that eigenspectra beyond the first do not look like recognizable X-ray absorption spectra. This emphasizes the point that the components we have found are abstract rather than physical. As the eigenspectrum index $s$ is increased, we reach a point where the eigenspectra show increasingly random fluctuations from one energy $n$ to another, and the eigenimages have the “salt and pepper” appearance of noise images. At this point, the variations being represented are no longer those of significant spectral variations in the data, but simply represent random variations due to noise. We therefore conclude that there is a subset of significant components $\tilde{S}_{\text{abstract}}$ that fully represent the meaningful variations in the data. There may in fact still be imbedded errors due to experimental errors (such as detector nonlinearities) that are mixed in with the $\tilde{S}_{\text{abstract}}$ correct factors, but the significant components $\tilde{S}_{\text{abstract}}$ exclude the extracted error [23,4, Chapter 4].

It is of course desirable to find a measure of $\tilde{S}_{\text{abstract}}$, and one such measure that is said to be particularly robust is the factor indicator function $\text{IND}(s)$ [4, Eq. (4.63)] which reaches a minimum when $s = \tilde{S}_{\text{abstract}}$. However, in our experience the factor indicator function is not a good predictor of $\tilde{S}_{\text{abstract}}$ for X-ray spectromicroscopy data sets, and one must instead determine an appropriate value of $\tilde{S}_{\text{abstract}}$ by examining several factors:

1. Examination of the eigenvalues $\lambda(s)$. The first few eigenvalues decrease rapidly as they measure increasingly subtle variations in spectral signature. One then enters a regime where there is a slow decrease in the eigenvalues associated with successive components of noise. The correct number of reduced components $\tilde{S}_{\text{abstract}}$ is approximately at the “knee” of the eigenvalue plot.

2. Examination of the quality of the reproduction of an experimental spectrum using only $\tilde{S}_{\text{abstract}}$ eigenspectra, as will be discussed below (see Figs. 3 and 14).

3. Examination of the eigenimages to see if there appears to be significant structure present, or if only random pixel-to-pixel variations (“salt and pepper” noise) appear.

In the lutetium/hematite data set of Fig. 2, examination of the data over the entire energy range of 520–580 eV suggests a value of $\tilde{S}_{\text{abstract}} = 5$ so as to fully represent all nonnoise variations of the data. However, the small amount of structure shown in the $s = 5$ eigenimage seems to be primarily due to slight differences in absorption in the spectroscopically uninteresting range of...
550–580 eV, as illustrated by the \( s = 5 \) eigenspectrum for the full energy range data. We have therefore re-calculated the components using only the data in the energy range 525–550 eV, where most of the oxygen near-edge structure is contained. The resulting components, also shown in Fig. 2, indicate that the important near-edge variations in the data are adequately represented by \( S_{\text{abstract}} = 4 \). It is our experience that restricting the energy range of the data in this manner is usually desirable. We will use this restricted energy range with \( S_{\text{abstract}} = 4 \) in subsequent analysis of the lutetium/hematite data.

We can now determine a reduced version of our data which we define by

\[
D'_{N \times P} = C_{N \times S_{\text{abstract}}} \cdot R_{S_{\text{abstract}} \times P}, \tag{11}
\]

which of course differs from Eq. (7) only by the restriction of using only \( S_{\text{abstract}} \) significant components rather than the full set of \( S_{\text{abstract}} = N \) components. If we have been careful in our choice of \( S_{\text{abstract}} \), this reduced data matrix should represent all the meaningful information of our original data, with the extracted error \([23, 4, \text{Chapter 4}]\) removed. This reduced data matrix has an additional important feature that will be exploited in cluster analysis: it is formed out of orthogonal eigenspectra ordered in degree of their significance (as determined by their eigenvalues \( \lambda(s) \) in Eq. (9)), separating successively important variations in the data into successive indices \( s \) of \( S_{\text{abstract}} \). This orthogonalized, noise-filtered representation of the data is a good “space” to search for patterns in the data.

2.3. Fitting physical spectra using principal components

Having found a reduced set of eigenspectra and eigенимages that describes the data, we assume that there must exist a transformation matrix \( T \) that allows one to re-create actual spectra \( M'_{N \times S_{\text{physical}}} \) and thickness images \( I_{S_{\text{physical}} \times P} \) from the \( S_{\text{abstract}} \) number of principal components. This transformation must satisfy

\[
D'_{N \times P} = (C_{N \times S_{\text{abstract}}} \cdot T_{S_{\text{abstract}} \times S_{\text{physical}}}) \cdot (T_{S_{\text{physical}} \times S_{\text{abstract}}}^{-1} \cdot R_{S_{\text{abstract}} \times P}), \tag{12}
\]

whereas the reduced data matrix of Eq. (11) is assumed to be formed from \( D'_{N \times P} = C_{N \times S_{\text{abstract}}} \cdot R_{S_{\text{abstract}} \times P} \). Comparing Eq. (12) with Eq. (5), it can
be seen that physical spectra can be associated with eigenspectra by

$$\mu_N \times S_{\text{physical}} = C_{N \times S_{\text{abstract}}} \cdot T_{S_{\text{abstract}} \times S_{\text{physical}}} \cdot \mu_N \times S_{\text{physical}}.$$  \hspace{1cm} (13)

The transformation matrix $T$ can therefore be determined from the eigenspectra and physical spectra to be

$$T_{S_{\text{abstract}} \times S_{\text{physical}}} = C^T_{S_{\text{abstract}} \times N} \cdot \mu_N \times S_{\text{physical}},$$  \hspace{1cm} (14)

where we have again used the fact that $C$ is orthogonal so $C^{-1} = C^T$. With $T$ thus determined, we can also represent the thickness maps $t_{S_{\text{physical}}} \times p$ from the eigenimages by

$$t_{S_{\text{physical}}} \times p = T^{-1}_{S_{\text{physical}}} \cdot R_{S_{\text{abstract}}} \times p.$$  \hspace{1cm} (15)

While the transformation matrix $T_{S_{\text{abstract}} \times S_{\text{physical}}}$ involves the matrix of orthogonal eigenspectra $C^T_{S_{\text{abstract}} \times N}$, it also involves the matrix of target spectra $\mu_N \times S_{\text{physical}}$ which has no guarantee of being orthogonal. We therefore must invert the transformation matrix without assuming orthogonality; this can be accomplished using singular value decomposition as described in Appendix A.

Evaluating the quality of the reproduction of an experimental spectrum $\mu_N \times S_{\text{physical}}$ from $S_{\text{abstract}}$ eigenspectra according to Eq. (13) provides a very good means of judging the proper choice of $S_{\text{abstract}}$ in X-ray spectromicroscopy. As Fig. 3 shows, selection of a reasonable value of $S_{\text{abstract}}$ allows one to obtain a fitted spectrum that recreates the physically significant elements of the experimental spectrum while rejecting noise. Of course, if the physical spectrum is not well represented by either the full $S_{\text{abstract}}$ or restricted $S_{\text{abstract}}$ set of eigenspectra (meaning the eigenspectra do not fully span the spectral set in which the physical spectrum lies), it will be impossible to fully recreate its spectral signature. This can happen if the physical spectrum is acquired in a separate measurement where different systematic errors apply. This effect can be seen in the residual to the physical spectrum fit in Fig. 3, but it is absent when reconstructing cluster spectra as will be shown in Fig. 14.

### 2.4. Comments on data preprocessing

It is not uncommon for researchers in spectromicroscopy or spectrum imaging analysis to carry out a number of preprocessing operations on their data. We therefore comment on them from a point of view of applying them to X-ray spectromicroscopy data, and illustrate results using some of these approaches in Fig. 4:

- In infrared spectroscopy, it is common to take the second derivative of spectra prior to classifying them [24] to increase their visual distinguishability. This is a less desirable step in X-ray and electron approaches because radiation damage considerations lead the experimentalist to acquire quantum-noise-limited spectra which result in very noisy derivatives. In addition, it is reasonable to expect that a covariance test of spectral difference might work just as well on raw data as well as on second-derivative data, since the same energy-to-energy difference information is contained in both representations.

- It is tempting to consider applying some sort of spectral smoothing to the data prior to principal component analysis. We have succumbed to this temptation in explorations of different approaches, but we have subsequently rejected it. Spectral filtering slightly alters the shape of the eigenvalue versus component curves such as are shown in Fig. 2B, but the “knee” in these curves still exists and the characteristics of the eigenspectra at the transition from $S_{\text{abstract}}$ significant to $S > S_{\text{abstract}}$ insignificant components are much the same. Indeed, the goal of working with a reduced data representation based on $S_{\text{abstract}}$ components is to remove those components with poor correlation, which naturally include quantum noise, and pre-smoothing of the data may be counterproductive by removing some of the noise that would otherwise be removed as extracted error [23,4, Chapter 4].

- Many researchers use covariance about the mean or mean centering. This involves subtracting the average spectrum from the spectrum of each
Fig. 4. Comparison of eigenvalues and eigenspectra obtained by PCA after various preprocessing operations have been applied to the data (see Section 2.4). We have chosen to work with the optical density $D_{N_x,P}$. 
pixel according to
\[
D^b_{N \times P} = D_{N \times P} - B_{N \times P},
\]
where \(B_{N \times P}\) is a matrix where each row \(n\) consists of repeats of the value
\[
b_n = \frac{1}{P} \sum_{p=1}^{P} D_{np}.
\]

If mean centering is applied, information about the zero point of the experimental scale is lost [4] which is useful information in X-ray spectromicroscopy where one wants to distinguish absorbing from nonabsorbing regions. Indeed, the shapes of the eigenspectra obtained with and without mean centering are identical (Fig. 4), with differences only in pixel weights for the first eigenspectrum (which in the mean centered case have partly negative values, caused by centering the origin of the eigenspace at the average of the data points). As a result, we do not use mean centering in our analysis.

- In electron energy loss spectroscopy, plural inelastic scattering effects mean that one cannot do a simple normalization of the data to obtain a linear optical density \(D(E) = -\ln[I(E)/I_0(E)]\). As a result, several authors renormalize the raw data \(I(E)\) according to the square root of row or column averages as required for factorial analysis of correspondence [25] rather than covariance analysis. This weights individual pieces of data according to their statistical significance. We have not chosen to apply this approach because of our desire to preserve a data matrix \(D_{N \times P}\) or reduced, noise-filtered data matrix \(D'_{N \times P}\) which can be used in proportional equations (Eqs. (6) and (15)) to obtain thickness maps. Different adjustments applied to each energy \(n\) or pixel \(p\) would violate that simple proportionality. In particular, by using the reduced data matrix \(D'_{N \times P}\) in principal component analysis, we have removed the spectral components with poor covariance (such as uncorrelated quantum statistical noise) as extracted error, which may provide much of the same effect of emphasizing data with good statistics.

For these reasons we have chosen to adopt the “first analyze, then process” philosophy advocated by Trebbia and Bonnet [25].

3. Cluster analysis

Application of principal component analysis to spectromicroscopy data has given us a very useful intermediate result: we can now gain insight into our data matrix \(D_{N \times P}\) in terms of a reduced set of principal eigenspectra \(C_{N \times S_{abstract}}\) and eigenimages \(R_{S_{abstract} \times P}\). These eigenspectra and eigenimages are noise-filtered and orthogonalized into components sorted by their degree of covariance, and thus significance. However, it is only in the case where we know of the spectra of all physical components of the specimen, and thus the full matrix \(\mu_{N \times S_{physical}}\), that we are able to calculate the transformation matrix \(T_{S_{abstract} \times S_{physical}}\) of Eq. (14) to allow us to interpret our eigenspectra as real spectra, and our eigenimages as real images. In other words, we have to know the answer in order to interpret the answer, which of course is unsatisfying if one has an unknown specimen. Other strategies exist; for example, one can use oblique analysis [26] to seek a transformation matrix \(T\) from the \(S_{abstract}\) orthogonal coordinates provided by principal component analysis to a set of \(S\) coordinates that have the properties of being pure positive and lying near groupings of data.

We adopt here an alternative strategy: we seek a method of grouping pixels with similar experimentally-determined spectra together, and then analyze the entire data according to these major spectral themes found in the data. We implement this theme-with-variations approach using cluster analysis or unsupervised pattern matching algorithms [5].

Cluster analysis typically involves evaluation of groupings of data points in some data representation, followed by classification or assignment of subsets of data to specific clusters. What data representation might be best for seeking clusters? Let us compare our set of principal eigenimages \(R_{S_{abstract} \times P}\) of Eq. (10) with the original data matrix \(D_{N \times P}\) of Eq. (5) or even its reduced version \(D'_{N \times P}\) of Eq. (11). The data matrix expresses the signal at
each pixel \( p \) in terms of its spectral response over \( N \) energies, whereas the eigenimage matrix expresses the signal in terms of its degree of incorporation of each of \( S_{\text{abstract}} \) orthogonal components; of course, \( S_{\text{abstract}} < N \). It is natural therefore that we look for clustering of the data in the eigenimage matrix \( R_{S_{\text{abstract}} \times P} \) over the set of dimensions \( S_{\text{abstract}} \), so that we can take advantage of the eigenimage matrix properties of orthogonality and reduced dimensionality. Each pixel \( p \) is then represented by a weighting \( R_{s,p} \) in each of the \( s = 1 \ldots S_{\text{abstract}} \) components. We can then attempt to locate cluster centers in this \( S_{\text{abstract}} \)-dimensional space, and classify pixels according to their distances from these cluster centers. While a great many distance metrics are available [5], distances from these cluster centers may be undesirable, since we may want to give greater or lesser weighting to the components with increasing index \( s \) that describe increasingly subtle variations in the spectral signatures of the data, or ever decreasing contribution of a spectral component that is present in few pixels.

We therefore introduce a power-law scaling parameter \( \gamma \) that will multiply the origin-centered eigenimage weightings \( R_{s,x,P} \) for the \( P \) pixels at each component \( s = 1 \ldots S_{\text{abstract}} \) according to

\[
R_{s,x,P}^{\text{scaled}} = (R_{s,x,P} - \langle R_s \rangle \left( \frac{\lambda(1)}{\lambda(s)} \right)^\gamma)
\]

Setting \( \gamma = 0 \) will eliminate any rescaling of eigenimage weightings \( R_{s,x,P} \). The sensitivity to higher component indices will be increased with \( \gamma > 0 \), and decreased with \( \gamma < 0 \). Setting \( \gamma \) to too large a value opens one to the risk of clustering pixels according to variations in their imbedded error [23,4, Chapter 4] which becomes more significant in higher index component indices \( s \); in practice, we find that values of \( \gamma \leq 0.5 \) work well.

The combined result of these two adjustments gives us a scaled set of eigenimages \( R_{S_{\text{abstract}} \times P}^{\text{scaled}} \) where the \( s = 1 \) component can optionally be removed from consideration in classifying the data.
3.1. Cluster analysis algorithm: learning vector quantization

In the ideal situation, data points are arranged in a few tightly packed, well separated groups. The goal of the clustering algorithm is to find a partitioning that minimizes distances within the groups and maximizes distances between them. While a great number of clustering algorithms exist, we use here a learning vector quantization (LVQ) algorithm [27,28] (closely related to self-organizing map algorithms) over K iterations as illustrated on Fig. 5:

1. We begin by assigning random positions to each of G cluster centers. The number of cluster centers G will usually be larger than the number of significant components \( \hat{S}_{\text{abstract}} \); determination of the number of clusters G will be discussed later. The starting position or component “weight”

\[
W_{\hat{S}_{\text{abstract}}} = [w_{1,g}, \ldots, w_{\hat{S}_{\text{abstract}},g}] \tag{20}
\]

of the gth cluster center is randomly assigned on a uniform distribution over the range \(-1\) to \(+1\) in each dimension \(s\).

2. We now choose one pixel \(p^*\) at random and calculate the distance from that pixel to each of the G cluster centers. The “winning” cluster center \(g^*\) which is closest to the pixel in question is then moved toward the pixel by adjusting its weights according to

\[
w_{s,g*} = w_{s,g*} + \beta_k [R_{s,g*} - w_{s,g*}] \tag{21}
\]

where \([R_{s,g*} - w_{s,g*}]\) is a component of the Euclidian vector distance from the old cluster center position \(W_{\hat{S}_{\text{abstract}}}\) to the pixel position \(R_{\hat{S}_{\text{abstract}} \times g^*}\). The coefficient \(\beta_k\) is a learning rate, which we adjust linearly from 0.3 to 0.1 over \(K = 20\) iterations in the present work. Cluster centers other than the “winning” cluster \(g^*\) are not adjusted.

3. We now repeat step 2 for all remaining pixels \(P\), and then for all iterations \(K\) as the outer index of a nested loop with the learning rate \(\beta_k\) adjusted as noted above. We have found it important to iterate over a randomized ordering of pixels \(P\); otherwise, cluster centers will acquire a bias towards the pixels at the lower left corner of the image (in our case, the starting pixel locations) if these pixels are used in succession at the start of the algorithm.

4. After \(K\) iterations have been completed, we now assign each pixel \(p\) to be a member of the cluster \(g\) which the pixel is closest to the center of.

We note that it is possible that some cluster centers \(g\) might be distant from all pixels \(p\) and thus never be chosen as the “winning” cluster \(g^*\) to be moved closer to a pixel. At the end of the algorithm, these cluster centers have no members, and they are removed from the list of clusters and the number of clusters \(G\) is adjusted accordingly. That is, bad initial guesses of cluster center positions will be abandoned rather than bias the clustering. We note that while in principle one can get different clustering results each time the algorithm is run due to different random choices of initial cluster positions, in practice we find good consistency between different runs of the algorithm on the same data. Finally, we have also implemented a K-means algorithm for locating cluster centers [5]. The results we have obtained
using K-means are quite similar to those obtained using the learning vector quantization method, except that we find that the boundaries of cluster regions in images are slightly smoother when using the learning vector quantization algorithm. As noted before, both algorithms use a measure of the distance from cluster centers, so in both cases the data are implicitly assumed to be clustered in hyperspheres.

Having assigned a cluster index $g$ to all pixels $P$, we can visualize our result in several standard ways. A pseudo-color image of the specimen can be generated where each cluster index $g$ is assigned a different color (see e.g., Fig. 6), and we can also display the number of pixels assigned to each cluster center. Histograms of distances of pixels from their respective cluster centers can be generated. These distances can also be shown for each pixel in a greyscale image where bright regions are those which are poorly classified by the number of cluster centers $G$ chosen (see Fig. 7). Scatterplots provide another means to view the result of cluster analysis. For any pair of significant components $i$ and $j$, one can plot the position of each pixel $p$ based on its eigenimage weightings $R_{i,p}^{scaled}$ and $R_{j,p}^{scaled}$. The pixels can be color-coded based on their assigned cluster index $g$, and a number for the cluster index can be plotted based on its coordinates $(w_{i,g}, w_{j,g})$ in these two components. An example of such a scatterplot is shown in Fig. 8, which demonstrates that it is not always possible to recognize cluster center positions based on only two components. Fortunately, the clustering algorithm is able to “see” the data in all $S_{abstract}$ components, and cluster the data accordingly.

### 3.2. Dendrograms and the number of clusters $G$

We now wish to determine the number of clusters $G$ that should be chosen for classifying the data. A commonly used method for aiding this choice is to examine hierarchical trees or dendrograms of the clustered data [5, 29]. These hierarchical
methods can be used as clustering algorithms in their own right, as well. Divisive hierarchical methods start with one cluster which contains all pixels \( P \), and successively splits the pixels into increasing numbers of clusters based on distances from current cluster centers. Agglomerative hierarchical methods start out with each pixel in its own cluster; the two closest pixels are merged into a new common cluster, and this process is continued until all pixels are merged into one cluster. In our case, we will use hierarchical agglomeration not at the starting point of individual clusters for each pixel, but at a starting point of having already classified the data into \( G \) clusters. In other words, we use hierarchical agglomeration not to cluster individual pixels (which would be very time consuming); instead, we use hierarchical agglomeration on cluster

Fig. 7. Error maps from cluster analysis. These images show the distance from a particular pixel to its cluster center. All displayed images are shown on the same scale, with the largest distance error shown as pure white. Regions which are not well described by a given number of clusters stand out clearly as being far from any cluster center.

Fig. 8. Scatterplots of the simulated test data. Each pixel \( p \) has a weighting \( R_{ij}^{\text{scaled}} \) (Eq. (19)) in each of the \( S_{\text{abstract}} \) components. These scatterplots show the position of all pixels (colored according to their assigned cluster in online version only) in any two of these components or dimensions, and the locations of the respective cluster centers (numbers offset slightly from their true locations for clarity) in these same dimensions. Clustering results A, B, and D from Fig. 6 are shown.
centers that have already been obtained using the LVQ or K-means algorithm.

For agglomerating clusters, distances between clusters can be determined by methods including measuring the distance between cluster centers (the centroid linkage), or determining the average of distances between all pairings of pixels from the two clusters (the average linkage). Having measured the distance between all clusters, one then merges the two clusters that are closest to each other into one new cluster with a position that is then set by using a distance matrix updated using the average linkage version of a recurrence formula [5, Eq. (4.2)].

Distances between clusters are then re-measured, and the process is repeated until there is only one cluster left. When using centroid linkage to calculate dendrograms, inversions or reversals can occur (see e.g., Fig. 4.8 of Everitt et al. [5]) which make interpretation more difficult. Such inversions do not occur when the average linkage is used, so this is the measure we have used for calculating the dendrograms shown here.

Agglomerative clustering is commonly visualized using a dendrogram (see e.g., Fig. 6), which illustrates the merger made at each step of the analysis. At the bottom are shown all G clusters provided by the initial cluster analysis; these are called the terminal nodes because they represent where division of the dendrogram ends. As two clusters are merged together into one, the distance between them serves as the branch distance along the vertical. In the final step, there is only one cluster left, which is called the root of the dendrogram.

This dendrogram can often be used to gain insight into a better choice of the number of clusters G. When many clusters are merged over a short distance along the dendrogram the distances between them are not very large so their characteristics are fairly similar. However, when the distance along the abscissa is large before two clusters are merged, then their characteristics are rather different. As a result, if one has a dendrogram which resembles several “arms” reaching down which eventually branch out into many “fingers” and “thumbs,” a good estimate of the number of clusters is the number of “wrists” present in the dendrogram. This measure can be used along with other information the user has about the specimen.

3.3. Cluster spectra and thickness maps

Cluster analysis has provided us with a means to classify our data based on similarities between spectra. However, a disadvantage of the approach is that it provides an either/or answer: a pixel is determined to be a member of either one cluster or another. (We note that fuzzy clustering methods, where pixels are assigned a weight for membership in more than one cluster, also exist.) This is often helpful for understanding X-ray spectromicroscopy data, but of course a real specimen may well have gradations of composition which gradually change from one position to another. These gradations in composition will be masked by nonfuzzy cluster analysis, unless one uses a very large number G of clusters in which case the simplification one originally sought through cluster analysis is lost!

In order to reach our final goal of characterizing a sample based on continuous thicknesses of representative spectral signatures, we first obtain these signatures by calculating the average spectrum $$\bar{D}_{N \times g}$$ of all pixels within a cluster as

$$\bar{D}_{N \times g} = \frac{1}{P_g} \sum_{j=1}^{P_g} D_{N \times j} / P_g,$$  \hspace{1cm} (22)

where j indexes the P_g pixels p that are members of cluster g, and N is the array of photon energies. Because each cluster’s spectrum $$\bar{D}_{N \times g}$$ will represent an average of all pixels with nearly identical spectra, it will have a signal-to-noise ratio that is greatly improved relative to individual pixel spectra. Now that we have this set of physical spectra that are present within the specimen, we can define a transformation matrix where the “signature” spectra matrix $$\bar{D}_{N \times G}$$ stands in for the measured physical spectra matrix $$\mu_{N \times S_{\text{physical}}}$$ to allow us to define a transformation matrix analogous to Eq. (14) of

$$T_{S_{\text{abstract}}} = C_{S_{\text{abstract}}}^T \cdot \bar{D}_{N \times G},$$  \hspace{1cm} (23)
which in turn allows us to follow Eq. (15) to obtain pseudo-thickness maps $t_{G \times P}$ for the set of $G$ "signature" spectra as

$$t_{G \times P} = T_{G \times S_{\text{abstract}}}^{-1} R_{S_{\text{abstract}} \times P} = T_{G \times S_{\text{abstract}}}^{-1} S_{\text{abstract}}^{T} \cdot C_{N \times N} \cdot D_{N \times P}^{T},$$

where we will use singular value decomposition to invert the matrix $T_{G \times S_{\text{abstract}}}$. We note that $\mu_{N \times S_{\text{physical}}}$ could be assumed to contain spectra expressed as linear absorption coefficients in a reciprocal physical length, allowing $t_{S_{\text{physical}} \times P}$ to be interpreted as thicknesses in physical units. Because the cluster "signature" spectra $D_{N \times G}$ are due to unknown thicknesses of unknown compounds, we cannot directly interpret the pseudo-thickness maps $t_{G \times P}$ in terms of physical thicknesses. (The same can be said of oblique analysis, where one determines a set of oblique spectra to be mapped [26].) Even so, the pseudo-thickness maps $t_{G \times P}$ and cluster spectra $D_{N \times G}$ are immensely useful in allowing us to view continuous transitions from one "signature" spectrum to another at cluster boundaries.

4. Cluster analysis of simulated data

In order to illustrate the performance of the analysis methods described above, we first consider the case of the simulated specimen of Fig. 1. In Fig. 2 we saw that it is better to carry out PCA and subsequent cluster analysis only over a narrow, near-edge energy range. Examination of the spectra in Fig. 1 would suggest that for the test data a restricted energy range of 284–292 eV would be appropriate for examination; however, as a more demanding test we used the full energy range of 282–302 eV for the analyses shown here. Principal component analysis gave $S_{\text{abstract}} = 3$ components, as one would expect for this simulated specimen composed of collagen, leucine, and tyrosine. We then carried out cluster analysis using the variations discussed in Section 3: we chose the eigenvalue power law scaling term $\gamma$ of Eq. (19) to be either 0 or 0.3, we changed the number of clusters $G$ sought, and we chose in one case to seek clusters only among components $s = 2, 3$ rather than the full set $s = 1, 2, 3$. As Fig. 6 shows, one can get slightly different clustering results with each choice of parameters, and indeed even with repeated clusterings with the same choice of parameters due to the fact that random cluster center positions are used as the starting point for each calculation. This would seem to indicate a lack of robustness in clustering, but examination of the dendrograms of Fig. 6 shows that in fact all clustering examples give the same classification of the essentials of the data: letters A and B are either grouped together or closely spaced on the same dendrogram branch, and the same applies to letters D and E. That is, the algorithm finds regions based on the similarity of their spectroscopic components, and also to a lesser extent based on their fractional thickness. In the case where overly aggressive clustering led to "salt and pepper" noise in the collagen background region, the erroneous result is readily recognizable, and the "salt and pepper" regions represent the last branch on the dendrogram which would be merged in the first step of agglomeration.

In addition to the dendrograms, it is helpful to look at the cluster distance error maps of Fig. 7. These figures show on a greyscale image the distance from each pixel to its "winning" cluster center $g^*$, and also several metrics of cluster distances: the maximum, the root mean squared, and the distance within which 95% of all pixels are located to their respective cluster centers. In a clustering run where the lowest-concentration letters C and F were not found, the distance error map very clearly shows that there are regions which were not properly clustered. It is also useful to consider the scatterplots of Fig. 8 of clustering runs A ($\gamma = 0$, components $s = 1, 2, 3$, sought $G = 8$, found $G = 4$ clusters), B ($\gamma = 0.3$, components $s = 1, 2, 3$, sought $G = 8$, found $G = 8$ clusters), and D ($\gamma = 0$, components $s = 2, 3$, sought $G = 8$, found $G = 4$ clusters) along with their respective cluster maps of Fig. 6. The scatterplot for clustering run B shows most clearly how each letter of the test data can be separated in principal component space, and also how clusters 1 and 2 are strongly overlapping and thus are the first to merge in the dendrogram of Fig. 6.
Our ultimate goal is to be able to come up with representative spectra for an unknown data set, and map thicknesses corresponding to these spectra. For our simulated data, we will use the results of clustering run A of Fig. 6. With these clusters, we show in Fig. 9 the cluster spectra $D_N$ calculated according to Eq. (22), and pseudo-thickness maps $t_{G \times P}$ calculated according to Eq. (24). As can be seen, the cluster spectra are very close to the collagen, leucine, and tyrosine spectra used to build the simulated data, with slight differences due to the fact that the pixels that were clustered together involved mixtures of leucine and collagen, or tyrosine and collagen, rather than the respective pure substances. Because the collagen spectrum is mixed into the spectra of clusters 2 and 3, the cluster 1 pseudo-thickness map shows some negative values. However, we also note that the low-concentration letters C and F do indeed show up in the pseudo-thickness maps even though they were not found by the clustering algorithm. This figure indicates how cluster analysis can be used to recover representative spectra from the data, and pseudo-thicknesses corresponding to these representative spectra.

5. Cluster analysis of lutetium/hematite data

We have carried out the analysis methods described above on the lutetium/hematite data described in Section 1.1. As shown in Fig. 2, $\hat{S}_{\text{abstract}} = 4$ principal components were used to describe the data in the energy range 525–550 eV. Examination of clustering results and dendrograms analogous to those shown in Fig. 6 showed that $G = 5$ clusters calculated using $\gamma = 0$ provided a reasonable segmentation of the data, as shown in Fig. 10. Cluster 1 is a mostly open region with very little optical density, while the spectrum of cluster 5 is similar to the pure hematite spectrum shown in Fig. 3 with nearly equal optical density at 530.0 and 531.7 eV. We can gain further insight into the
data by consideration of the dendrogram in Fig. 10. This shows that clusters 1 and 2 are most similar to each other in terms of weak absorption, even though their spectral signatures are rather different. Cluster 3 is then merged with these first two clusters, and its spectrum appears to be quite similar to that of cluster 2. Since clusters 2 and 3 both have decreased absorption at 530.0 eV relative to 531.7 eV, they can be interpreted as representing an increasing degree of substitution of Lu$^{3+}$ for Fe$^{3+}$ in the hematite matrix. Cluster 4 is merged with cluster 5; since its spectral shape is more similar to cluster 2 than cluster 5, cluster 4 can also be assumed to have a lesser but nonnegligible degree of lutetium incorporation into the hematite matrix. Its similarity to the nearly-pure-hematite of cluster 5 therefore must be based primarily on its greater optical density rather than on its chemical characteristics.

One problem in the clustering results of Fig. 10 is that the structure in the upper left appears not in one but in several pseudo-thickness maps, either as a positive pseudo-thickness (clusters 1, 3, and 4; grey regions) or as a negative one (clusters 2 and 5; red regions; color scale in online version only). Spectroscopically, clusters 2, 3 and 4 are fairly similar to each other, except in their overall scale of optical density, as noted above. These effects

<table>
<thead>
<tr>
<th>Distance between clusters</th>
<th>Cluster indices</th>
<th>Cluster spectra</th>
<th>Cluster thicknesses</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>Cluster 1</td>
<td>Cluster 5</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Cluster 2</td>
<td>Cluster 4</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Cluster 3</td>
<td>Cluster 3</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Cluster 4</td>
<td>Cluster 4</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>Cluster 5</td>
<td>Cluster 5</td>
</tr>
</tbody>
</table>

Fig. 10. Results of cluster analysis of the lutetium/hematite data, showing the cluster index image (A), cluster spectra $D_{N+G}$ (B), cluster pseudo-thickness maps $t_{G,P}$ (C), and dendrogram (D). These results were obtained with the $s = 1$ component included in the cluster analysis, which was carried out with $\gamma = 0$. Cluster 1 shows very little absorption, cluster 5 is reminiscent of the pure hematite spectrum shown in Fig. 3, and clusters 2–4 show various degrees of incorporation of lutetium.
signal that the clustering algorithm classified pixels in part due to similarities in thickness, as can be confirmed by examination of the scatterplots involving component 1 in Fig. 11. A more desirable outcome might be to instead classify the data according to similarities in spectroscopic signature with thickness effects disregarded. As was noted in Section 2.2, the first or $s = 1$ component of principal component analysis is dominated by the average absorption spectrum of the entire specimen, and the $s = 1$ eigenimage is in some sense a map of thickness in the specimen without regard for composition.

We have therefore recalculated the clustering of the lutetium/hematite data with the first $s = 1$ eigenimage excluded, as described in Section 3. We again sought $G = 5$ clusters with an eigenvalue scaling factor of $\gamma = 0$. The results are shown in Figs. 12 and 13. The scatterplots of Fig. 13 now do not show the data separated based on one and only one component. The cluster spectra $\tilde{D}_{N \times g}$ of Fig. 12 now show differences that are more pronounced in spectral signature than in overall optical density. (One can also carry out a cross-check of the correct number of abstract components $S_{abstract}$ by examining the target spectrum fits of Eq. (13) for the real, physical spectra of selected clusters, as shown in Fig. 14.) It is perhaps even more informative to note that Fig. 12 now shows few red (online only), negative pseudo-thickness regions $t_{G \times P}$. As before, cluster 1 shows mostly residual weak absorption throughout the specimen. Cluster 2 has a spectrum very similar to that of the pure hematite spectrum shown in Fig. 3; cluster 5 has a spectrum which is quite similar but may show the onset of the well-known “thickness effect” in absorption spectroscopy where spectral shapes can become distorted due to less-strongly-absorbed, higher diffraction orders from the X-ray monochromator. What is particularly interesting is that clusters 3 and 4 show differing intensities of the 530 eV absorption peak indicating a changing degree of lutetium incorporation into the hematite matrix.

The cluster analysis of the lutetium/hematite oxygen K-edge spectromicroscopic data demonstrates that lutetium (5 mol% per mol Fe) is initially structurally incorporated in the hematite
Fig. 12. Cluster analysis of the lutetium/hematite data with the first component excluded, showing the cluster index image (A), cluster spectra (B), cluster pseudo-thickness maps (C), and dendrogram (D). These results were obtained with $\gamma = 0$. Compared to the $s = 1-4$ cluster results of Fig. 10, these $s = 2-4$ clustering results show few regions of negative pseudo-thickness $t_{G,P}$ and clearer classification of the data based on chemical speciation as opposed to thickness.

Fig. 13. Scatterplots of pixel weightings $R_{dp}^{\text{calc}}$ for the lutetium/hematite data clustered with the first component excluded. Compared to the $s = 1-4$ cluster results of Fig. 11, the data are now clustered by a combination of components, rather than based on one component only. This approach classifies the data more on its spectroscopic variations and less on its thickness.
crystal structure, as indicated by significant pre-edge changes in clusters 3 and 4 of Fig. 12. It also shows a nanoscale separation into regions compatible with pure hematite (clusters 2 and 5) and mixed Fe/Lu hematite crystals with distinct Lu concentrations (clusters 3 and 4). These results suggest that lutetium-substituted hematite might not be the thermodynamic stable phase, and that re-crystallization processes structurally exclude lutetium. Such kinetic information is of paramount importance to identify the essential mineral phases determining long-term radionuclide mobility, and further investigations of these phenomena are presently underway.

6. Conclusion

We have described the use of principal component analysis to orthogonalize and noise-filter spectromicroscopy data, and cluster analysis to classify the data into regions with similar spectra.
This allows one to obtain characteristic, physically-meaningful spectra, and pseudo-thicknesses associated with these spectra, from specimens with no prior information on composition. One can “tune” the degree to which chemical variations are weighted relative to thickness variations by excluding the first component from consideration by the clustering algorithm, by using an eigenvalue scaling parameter $\gamma$ to increase the sensitivity to higher components, or both. Ongoing investigations concern the use of distance metrics other than Euclidian, and the combination of clustering with oblique analysis.

This approach has been examined using simulated data where the known composition was recovered, and data obtained as part of a study of the incorporation of lutetium (a lanthanide homologue for the trivalent actinide americium) in hematite with implications for groundwater colloidal transport of radionuclides.

Acknowledgements

We wish to thank Michael Feser, Jörg Maser, Sue Wirick, and Kathy Dardenne for many helpful discussions, and Angelika Osanna for her role in bringing multivariate statistical analysis methods to our research. We gratefully acknowledge funding from the National Institutes for Health under Contract R01 EB00479-01A1, and from the National Science Foundation under Contracts OCE-0221029 and CHE-0221934. Data were acquired using the Stony Brook scanning transmission X-ray microscopes which operate at the National Synchrotron Light Source (NSLS) at Brookhaven National Laboratory, which is supported by the U.S. Department of Energy, Division of Materials Sciences and Division of Chemical Sciences, under Contract No. DE-AC02-98CH10886.

Appendix A. Matrix inversion using singular value decomposition (SVD)

Singular value decomposition is based on the Eckart–Young theorem of linear algebra, which states that an array $A$ with $N \geq S$ can be decomposed as

$$A_{N\times S} = U_{N\times S} \cdot W_{S\times S} \cdot V_{S\times S}^T,$$

(A.1)

where the matrix $U_{N\times S}$ has orthogonal columns, the matrix $W_{S\times S}$ is zero everywhere except for its diagonal elements which are all zero or positive (these diagonal elements are called the singular values), and the matrix $V_{S\times S}$ has orthonormal rows. That is, these matrices have the properties that $U_{N\times S}^T \cdot U_{N\times S} = I_{N\times N}$, and $V_{S\times S}^T \cdot V_{S\times S} = I_{S\times S}$. The singular value decomposition algorithm [30, Section 2.6] can be used to numerically construct these arrays. With them, one can find the inverse of $A_{N\times S}$ as $A_{N\times S}^{-1} = V_{S\times S} \cdot W_{S\times S}^{-1} \cdot U_{N\times S}^T$, (A.2) where the inverted matrix $W_{S\times S}^{-1}$ is again a diagonal matrix with elements $W_{i,i}^{-1}$ that are the inverse of the singular values $W_{i,i}$, or zero when $W_{i,i} = 0$.

Appendix B. Calculation of eigenspectra using singular value decomposition

Besides using covariance, the eigenspectrum matrix can also be determined from the singular values of the data matrix. Following [4, Eq. (3.81)] and realizing that we have defined the data matrix as (see Eq. (5))

$$D_{P\times N} = C_{N\times S_{\text{abstract}}} \cdot R_{S_{\text{abstract}} \times P}$$

rather than $D_{P\times N}$, we can express the data matrix using singular value decomposition (SVD) as

$$D_{P\times N} = U_{P\times S_{\text{abstract}}} \times W_{S_{\text{abstract}} \times S_{\text{abstract}}} \times V_{S_{\text{abstract}} \times N}^T.$$

(B.1)

The eigenvalues of the data matrix are then given by [4, Eq. (3.84)]

$$\lambda(s) = W_{S_{\text{abstract}}}^2$$

(B.2)

from diagonal matrix $W_{S_{\text{abstract}} \times S_{\text{abstract}}}$, the eigenspectra are given by [4, Eq. (3.83)]

$$C_{N\times S_{\text{abstract}}} = V_{N\times S_{\text{abstract}}}$$

(B.3)

and the eigenimages are given by [4, Eq. (3.82)]

$$R_{S_{\text{abstract}} \times P} = U_{S_{\text{abstract}} \times P} \times W_{S_{\text{abstract}} \times S_{\text{abstract}}}.$$

(B.4)
In case like ours, where we have very large number of pixels $P$ and small number of energies $N$, calculating eigenspectra and eigenimages using SVD takes significantly longer and uses much more data storage than calculating it by using the covariance matrix $Z_{N \times N}$. SVD generates square matrices with the larger dimension of $N$ or $P$, while $Z_{N \times N}$ or $Z_{P \times P}$ can be formed based on the lesser of $N$ or $P$.

### Appendix C. Eigenspectra and eigenimages from spatial covariance

In soft X-ray spectromicroscopy, the intrinsic width of near-absorption-edge resonances is about 0.06–0.2 eV, while the energy range over which they lie is typically 20–30 eV. Data sets with $N = 100–500$ photon energies are thus typical, while the number of pixels may be $P = 100 \times 100 = 10^4$ or more, so that $N \ll P$. It is for this reason that we have here used the spectral covariance $Z_{N \times N}$ as defined by Eq. (8). If, on the other hand, one has $P < N$ (such as in infrared microspectroscopy), the spatial covariance $Z_{P \times P}$ of the data can be obtained from

$$Z_{P \times P} = D_{P \times N} \cdot D_{N \times P}^T,$$

where data matrix $D_{P \times N}$ is given by

$$D_{P \times N} = C_{P \times S_{\text{abstract}}} \cdot R_{S_{\text{abstract}} \times N}.$$

Eigenimage matrix $C_{P \times S_{\text{abstract}}}$ is obtained from the spatial covariance as

$$Z_{P \times P} \cdot C_{P \times S_{\text{abstract}}} = C_{P \times S_{\text{abstract}}} \cdot A_{P \times P}.$$

Columns of the eigenimage matrix $C_{P \times S_{\text{abstract}}}$ now consist of the $P$ eigenimages. The row matrix $R_{S_{\text{abstract}} \times N}$ is now an eigenspectra matrix, and it can be obtained from

$$R_{S_{\text{abstract}} \times N} = C_{S_{\text{abstract}} \times P}^T \cdot D_{P \times N}^T.$$

The spatial and spectral covariances will yield the same conclusions about the number of principal components [4], target spectrum fitting, and so on, so it is sensible to choose whichever one gives the smaller and faster computation.

### References