Randomized Approximation Methods for the Efficient Compression and Analysis of Hyperspectral Data

ARTICLE *in* ANALYTICAL CHEMISTRY · MARCH 2013

Impact Factor: 5.64 · DOI: 10.1021/ac400184g · Source: PubMed

analytical chemistry

Article

pubs.acs.org/ac

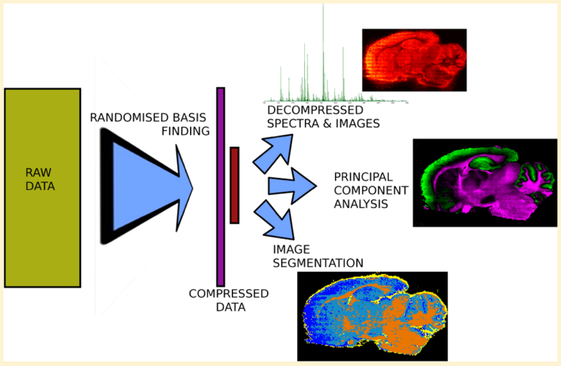
Randomized Approximation Methods for the Efficient Compression and Analysis of Hyperspectral Data

Andrew D. Palmer,[†] Josephine Bunch,[‡] and Iain B. Styles*[,§]

[†]PSIBS Doctoral Training Centre, [‡]School of Chemistry, and [§]School of Computer Science, University of Birmingham, Edgbaston, United Kingdom

Supporting Information

ABSTRACT: Hyperspectral imaging techniques such as matrix-assisted laser desorption ionization (MALDI) mass spectrometry imaging produce large, information-rich datasets that are frequently too large to be analyzed as a whole. In addition, the "curse of dimensionality" adds fundamental limits to what can be done with such data, regardless of the resources available. We propose and evaluate random matrix-based methods for the analysis of such data, in this case, a MALDI mass spectrometry image from a section of rat brain. By constructing a randomized orthornormal basis for the data, we are able to achieve reductions in dimensionality and data size of over 100 times. Furthermore, this compression is reversible to within noise limits. This allows more-conventional multivariate analysis techniques such as principal component analysis (PCA) and clustering methods to be directly applied to the compressed data such that the results can easily be back-projected and interpreted in the original measurement space. PCA on the compressed data is shown to be nearly identical to the same analysis on the original data but the run time was reduced from over an hour to 8 seconds. We also demonstrate the generality of the method to other data sets, namely, a hyperspectral optical image of leaves, and a Raman spectroscopy image of an artificial ligament. In order to allow for the full evaluation of these methods on a wide range of data, we have made all software and sample data freely available.



INTRODUCTION

Hyperspectral Imaging. Spectroscopic imaging is an increasingly popular tool in chemical analysis, with a variety of techniques available for the imaging and analysis of different sample types and target molecules. These include optical imaging, magnetic resonance spectroscopy, Raman spectroscopy, energy-dispersive X-ray energy, coherent anti-Stokes scattering, and mass spectrometry. These probe methods are commonly used to yield images of chemical distributions within samples, with specific techniques being used for particular classes of molecule or imaging requirements (for example, optical spectroscopy for remote chemical sensing,[1] magnetic resonance spectroscopy for *in vivo* measurements,[2] and mass spectrometry for unguided biomarker discovery[3]).

Spectral imaging methods collect image data in many (m) discrete measurement channels (for example, m/z ratio in mass spectrometry) at each of n spatial locations (pixels), and when the number of channels is large, the method may be called "hyperspectral". The sophistication of modern instrumentation has led to a rapid increase in the amount of data that can be collected and computational resources and methods have not kept pace.[4,5] There has been a steady move away from a strategy of collecting observations from only a few carefully chosen channels to measuring to the limits of the instruments and processing the data afterward.[4] Such strategies are problematic, since they mean that large amounts of data must be stored and processed, which is time-consuming and costly, even for simple tasks.

When computations are required in order to evaluate spectral profiles in distinct regions, or to classify different regions of the image, then the size of the data is even more restrictive. Hyperspectral methods may generate datasets that are tens of gigabytes in size and cannot be loaded into fast memory unless high-performance computing facilities are available.

There is a fundamental difficulty in processing very high-dimensional data that cannot be simply overcome via increased computational effort. As the dimensionality of a dataset increases, the variance in the distances between pairs of points tends toward zero—the so-called "curse of dimensionality". This is problematic because many data analysis methods require the calculation of distances between data points (e.g., to determine nearest neighbors or compare data to reference models). Furthermore, the number of data points required to generate reliable statistics over the dataset grows exponentially with the number of dimensions, and machine learning techniques cannot be reliably applied to extract patterns from the data. An empirical rule is that, for effective classification, a *sample to feature ratio* of 5−10 is necessary.[5]

Received: February 4, 2013
Accepted: March 27, 2013

The typical strategy for dealing with high-dimensional data is to try to take advantage of inherent redundancy due to nonvarying channels, or channels that covary strongly. Other types of data reduction use binning in the spatial[6] and/or spectral[7] domains at the expense of data resolution. There are also a number of sophisticated modality-specific methods.[8−10] These may require several passes through the data and this may be costly when the data cannot be held in memory, because hard disk access is substantially slower.

Random Projections for Data Analysis. One approach to dimensionality reduction that has seen a recent increase in popularity is random projection,[11] which reduces the dimensionality of the data by projecting the data onto a set of randomly chosen vectors. This process removes some data redundancy while approximately preserving distances between points, and angles between vectors,[11−13] which is highly desirable for certain types of machine learning and pattern analysis algorithms. For example, classification methods that compute distances between points can be used on the randomly projected data.[14] Random projections have been used for rapid filtering of secondary ion mass spectra,[15] for data transmission in compressive-projection principal component analysis,[16] and for data compression for transmission using random basis generation.[17] Other recent applications of random projections have included work on text mining,[11] semantic indexing,[18] and classification of gene array data.[19,20] Random projections have been applied to hyperspectral image scenes with particular interest in projection at the point of measurement for compressive sensing.[21,22] Image processing has also benefited from random projections with applications in unmixing,[23] clustering,[24] and nearest-neighbor finding.[11] In all cases, random projections were found to be computationally more efficient, with little or no degradation in the quality of results.

However, a simple random projection does not readily allow for interpretation of the results in the original measurement space (which is easily understood and physically meaningful). A disadvantage is that, in general, the reverse transformation is a probabilistic recovery problem.[16] Recent work by Halko et al.[25] has shown how random projections can be used to construct a low-dimensional orthogonal basis for a dataset. In this paper, we employ these ideas to construct a method for compressing hyperspectral data by reducing the dimensionality of the data down to approximately that of the subspace occupied by the data. We show how further computations can be performed on the compressed dataset, and how the results can be easily interpreted in the original (physical) measurement space (e.g., m/z values in mass spectrometry). We show how this method leads to a dramatic reduction in the size of hyperspectral datasets without a significant loss of information. Furthermore, we demonstrate that the method is applicable to other types of hyperspectral data by applying it to optical and Raman microscopy datasets. Demonstration code for employing basis approximation for spectral compression and a sample dataset have been made freely available in the Supporting Information.

Randomized Matrix Factorizations. Matrix factorizations are the basis of many classical data analysis methods; however, they often cannot be applied to high-dimensional data, because of the computational cost, which prevents well-understood techniques from being used to provide insight into these datasets. Random projections can serve additional purposes other than simple projection/reduction; recent work has investigated their use to enable efficient matrix factorization.[25]

The general motivation for performing a matrix factorization is to express the input data in a particular form that clearly exposes its structure and properties. A common example of matrix factorization is principal component analysis (PCA), which simultaneously reveals patterns in data variance and suppresses noise.[7,26,27] For very large matrices, the computations required to perform many types of matrix factorization are impossible, or extremely slow due to the resources required to perform them. It has been shown that a carefully constructed random projection can be used to find a subspace which retains most of the action of the matrix, and to construct an orthonormal basis for this subspace, providing a compressed representation of the matrix that retains nearly all of the information present in the original matrix.[25] Matrix factorization can then be performed on the compressed matrix. Furthermore, the compression scheme allows for simple and high-quality decompression back into the original space. These are the ideas that form the basis of this work.

■ METHODS

Randomized Basis Finding. Hyperspectral imaging data tends to be highly rank-deficient; that is, there are many linear dependencies within the data. This is because channels in the dataset will frequently covary with each other, implying that the data exists in a lower-dimensional space than the measurements. If the transformation required to represent the data within this subspace could be determined, then the data could be represented and manipulated much more efficiently. In general, this is a computationally expensive task, and the size of many hyperspectral datasets often prohibits a deterministic calculation of the required transformation.

However, it is possible to find this subspace probabilistically. The central idea is that projecting the data onto a set of random vectors preserves (with high probability) the majority of the information within the data set, provided that the number of random vectors is slightly larger than the rank of the full dataset $\mathbf{X}_{m\times n}$.[25] The random projections can then be orthogonalized (for example, by QR decomposition or Gramm−Schmidt orthogonalization) to generate an orthonormal basis, $\mathbf{Q}$, of greatly reduced dimension for the dataset that preserves most of the information content. A simple procedure for this has been proposed[25] and is described in Algorithm 1 and graphically in Figure 1.

Algorithm 1.

Aim: Generate an approximate basis for a spectral image.
Input: Spectral image, $\mathbf{X}$; integer k.
Output: Approximate basis for $\mathbf{X}$, $\mathbf{Q}$.

(1) Consider a data set $\mathbf{X}_{m\times n}$ containing n pixels in each of m spectral channels.

(2) Generate random vectors $\{\mathbf{v}^{(i)}\}_{i=1:k}$ of length n (with $k \gtrsim \mathrm{rank}(\mathbf{X})$) by drawing values from a normal distribution with a mean of 0 and a standard deviation of 1: $\mathcal{N}(0, 1)$.

(3) Form the random projection matrix $\mathbf{\Omega}_{n\times k} = [\mathbf{v}^{(1)}|...|\mathbf{v}^{(k)}]$

(4) Project $\mathbf{X}$ onto the random vectors to create a random sampling of the data $\mathbf{S}_{m\times k} = \mathbf{X\Omega}$.

(5) Create orthonormal matrix $\mathbf{Q}_{m\times k}$ from $\mathbf{S}$ by factorizing $\mathbf{S} = \mathbf{QR}$, with $\mathbf{R}$ being an upper triangular matrix.

The matrix $\mathbf{Q}$ can be used to project the data onto the lower dimensional subspace and form a reduced data matrix $\mathbf{A}_{k\times n} = \mathbf{Q}^{\mathrm{T}}\mathbf{X}$. This provides a high-quality compressed representation of X, provided that a suitable value of the reduced

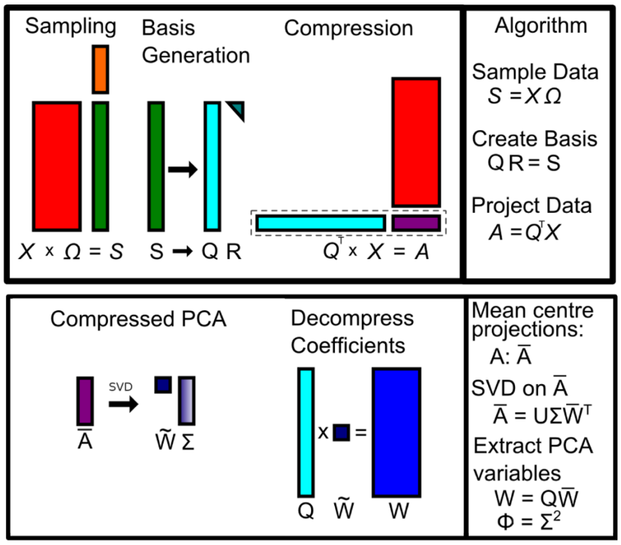


Figure 1. (Top) Diagram showing the sampling and compression scheme. Form the sampling matrix (**S**) by multiplying the data (**X**) by a random matrix (**Ω**): **S** = **XΩ**. An orthonormal basis is formed by QR decomposition of the sampling matrix. Compress **X** onto **Q** by leaving data stored as a pair of matrices: the basis **Q** and scores **A**. (Bottom) Diagram showing the compressed PCA process. The compressed data **A** is mean-centered and SVD is performed to produce the compressed PCA coefficients (loadings) $\tilde{\mathbf{W}}$ and variances Φ. The coefficients are decompressed using the compression basis $\mathbf{W} = \mathbf{Q}\tilde{\mathbf{W}}$. Multiply Φ by $\tilde{\mathbf{W}}$ to obtain the scores (**Y**).

dimensionality k has been chosen. This can be checked by computing the error metric, $M = \|\mathbf{X} - \mathbf{QA}\|$.

The reduced representation is good, provided that $M \leq \varepsilon$, with ε being a small positive real number, i.e., projecting the data onto the low-rank subspace, and then back-projecting into the original space recreates the original dataset to a high degree of accuracy. In this simple procedure, k was selected manually; iterative algorithms exist to determine this automatically[25] but are not practical on data stored out of main memory. In practice, we have found that it is straightforward to estimate a suitable value of k. The consequence of these results is that, by storing only the matrices **Q** and $\mathbf{A} = \mathbf{Q}^T\mathbf{X}$, we can recreate the original dataset via $\mathbf{X} = \mathbf{QA}$. For datasets of low rank, this can be a very considerable saving in memory. We will define the compression ratio $R_c = B_c/B_o$, where $B_o = m \times n$ is the original data size (**X**), and $B_c = k(m + n)$ is the combined size of the basis matrix **Q** and the reduced data matrix (**A**).

Computations on Projected Data. Randomized basis finding allows us to perform a reversible dimensionality reduction on a dataset. Since this reduction is essentially an orthogonalized random projection, it has the same properties as a conventional random projection. That is, it preserves distances between points and angles between vectors, and therefore is suitable for further analysis (for example, classification).[24,28] The orthogonality of the projection allows the analysis of the reduced data to be taken one step further: algorithms that depend on some type of matrix factorization can also be used. As an example of this, we will consider how to perform PCA on the dataset. In general, PCA is performed by computing the eigenvalues and eigenvectors of the covariance matrix as in Algorithm 2 but given the reduced dataset $\mathbf{A}_{k\times n}$, we follow Algorithm 3, which computes the eigenvalue decomposition of the compressed covariance matrix and subsequently decompresses the eigenvectors. A graphical summary of this algorithm is presented in Figure 1. In the literature, the eigenvectors are often referred to as PCA coefficients or PCA loadings; we use the term "coefficients" in this paper, but all three are interchangeable in this context.

Algorithm 2.

Aim: Compute the principal component eigenvectors and eigenvalues from a data matrix.

Input: A data matrix, **X**.

Output: eigenvectors,**W**; eigenvalues, **Φ**.

(1) Given a data matrix $\mathbf{X}_{m\times n}$, compute the mean of each row (data channel) and subtract to form matrix $\bar{\mathbf{X}}$, such that each row of $\bar{\mathbf{X}}$ has a zero-mean.
(2) Compute the covariance matrix $\mathbf{C} = \bar{\mathbf{X}}\bar{\mathbf{X}}^T$.
(3) Calculate the eigenvalue/vectors of **C**. Noting that **C** is a symmetric real matrix, this is equivalent to performing SVD on **C**: $\mathbf{C} = \mathbf{W\Phi W}^T$, The columns of **W** are the principal component vectors, and the diagonal of **Φ** contains the variance along each of the principal components.

Algorithm 3.

Aim: Compute the principal component eigenvectors and eigenvalues from a compressed data matrix.

Input: Compressed data matrix, **A**; and its approximate basis, **Q**.

Output: eigenvectors,**W**; eigenvalues, **Φ**.

(1) Form the zero-mean reduced data matrix $\bar{\mathbf{A}}$, using Algorithms 1 and 2.1.
(2) Form the covariance matrix in the reduced subspace, $\tilde{\mathbf{C}} = \bar{\mathbf{A}}\bar{\mathbf{A}}^T$.
(3) Perform SVD to diagonalize: $\tilde{\mathbf{C}} = \tilde{\mathbf{W}}\tilde{\mathbf{\Phi}}\tilde{\mathbf{W}}^T$, giving the principal components and variances in the reduced subspace.
(4) Since $\bar{\mathbf{A}} = \mathbf{Q}^T\bar{\mathbf{X}}$, we observe that $\tilde{\mathbf{C}} = \bar{\mathbf{A}}\bar{\mathbf{A}}^T = \mathbf{Q}^T\bar{\mathbf{X}}\bar{\mathbf{X}}\mathbf{Q}$.
(5) It then follows that $\mathbf{Q}\tilde{\mathbf{C}}\mathbf{Q}^T = \mathbf{Q}\mathbf{Q}^T\bar{\mathbf{X}}\bar{\mathbf{X}}^T\mathbf{Q}\mathbf{Q}^T \approx \bar{\mathbf{X}}\bar{\mathbf{X}}^T = \mathbf{C}$.
(6) Since $\mathbf{C} = \mathbf{W\Phi W}^T$ and $\tilde{\mathbf{C}} = \tilde{\mathbf{W}}\tilde{\mathbf{\Phi}}\tilde{\mathbf{W}}^T$, we have $\mathbf{WFW}^T = \mathbf{Q}\tilde{\mathbf{W}}\tilde{\mathbf{\Phi}}\tilde{\mathbf{W}}^T\mathbf{Q}^T$.
(7) The principal components of **X** can be computed directly from the principal components of **A**: $\mathbf{W} = \mathbf{Q}\tilde{\mathbf{W}}$ and $\mathbf{\Phi} = \tilde{\mathbf{\Phi}}$.

The use of an orthogonalized random projection has two important consequences. First, it has reduced the dimensionality of the problem. We can perform PCA on the reduced data matrix **A**, instead of on the full data matrix **X**. On a full dataset, this calculation requires the formation and diagonalization of the $m \times m$ covariance matrix, with $m \approx 10^5$ for MALDI datasets. This is intractable by normal means, although special data reduction techniques can be employed in combination with memory-efficient algorithms, which construct the principal components (PCs) without requiring the data to be in memory.[29] All of these techniques require data channels to be explicitly removed from the dataset. Here, we need only diagonalize the $k \times k$ covariance matrix from the reduced subspace, from which no data channels have been fully removed. Second, the orthogonalization procedure allows the PC vectors to be projected back into the full high-dimensional measurement space, so that they can be analyzed in terms of physically meaningful quantities. This is the key advantage of this approach over regular random projection methods.

It should be noted that an equivalent calculation can be performed by SVD factorization of the data matrix directly, rather than forming the covariance matrix. A similar argument can be applied to this approach, and the same transformations can be used to recover the principal components in the original space, as described in Algorithm 3.

The projection of the data onto the PC vectors, the "scores", are denoted by $\mathbf{Y}_L$ where the number of components maintained (L) is determined by the fraction of variance maintained. Columns from $\mathbf{Y}_L$ can be plotted as an image to examine trends extracted by PCA, and the spectral origin of these trends can be deduced from the PC vectors.

Therefore, the use of randomized algorithms for basis finding has two important consequences for the processing of high-dimensional hyperspectral datasets. First, they permit the data to be compressed/decompressed via a simple matrix multiplication, Second, they make it possible to perform computations on the data *while it is compressed.*

Pixel Subsampling. The procedure for computing a randomized basis can be further refined by the following observation. The computation forms random linear combinations of the data points, and there is very likely to be some degree of homogeneity in the data points, since, for example, neighboring pixels are likely to have very similar spectra. Therefore, it is not necessary to include all data points in the basis computation, and we can *spatially randomly subsample* the data to increase the efficiency of the computation, as explained in Algorithm 4. The basis generated from a randomly chosen subset of the data will, with high probability, form a basis for the full dataset as long as the sampling density is sufficient. This can easily be checked by ensuring that the error metric $M < \varepsilon$. The use of spatial subsampling reduces the amount of computation that must be performed in order to compute a random basis for the dataset and is particularly advantageous for datasets with many pixels and a high degree of spatial homogeneity.

Algorithm 4.

```
Aim: Generate an approximate basis for a spectral image.
Input: Spectral image, $\mathbf{X}$; integer $k$; integer $r$.
Output: Approximate basis for $\mathbf{X}$, $\mathbf{Q}$.
(1) Consider a dataset $\mathbf{X}_{m\times n}$ containing $n$ pixels in each of $m$ spectral channels.
(2) Randomly select $r$ columns from $\mathbf{X}$ to form a spatially randomly subsampled data matrix $\mathbf{Y}_{m\times r}$.
(3) Generate random vectors $\{\mathbf{v}^{(i)}\}_{i=1:k}$ of length $r$ (with $k \gtrsim \mathrm{ran}(\mathbf{X})$) by drawing values from a normal distribution with a mean of zero and a standard deviation of one ($\mathcal{N}(0, 1)$).
(4) Form the random projection matrix $\mathbf{\Omega}_{r\times k} = [\mathbf{v}^{(1)}|...|\mathbf{v}^{(k)}]$.
(5) Project $\mathbf{Y}$ onto the random vectors to create a sampling matrix $\mathbf{S}_{m\times k} = \mathbf{Y\Omega}$.
(6) Create orthonormal matrix $\mathbf{Q}_{m\times k}$ from $\mathbf{S}$ by factorizing $\mathbf{S} = \mathbf{QR}$, with $\mathbf{R}$ being an upper triangular matrix.
```

Evaluation of Compression. The error metric M is a useful measure of the ability of the randomized basis to represent the data, but it is not a measure of how well a specific spectrum can be reconstructed from its compressed representation. For this purpose, we adopt two measures that are commonly used to evaluate compression quality and are used to compare individual decompressed spectra from a compressed image with the corresponding single pixel spectrum from the input data. Signal-to-noise ratio (SNR) is commonly used in spectral imaging communities to evaluate the quality of a compression and compares the average numerical deviation between the input data and data that has undergone a compression−decompression cycle, with reference to the input data. It is computed as[30]

$$\mathrm{SNR} = 10 \log_{10}\left(\frac{\sigma^2}{\mathrm{MSE}}\right)$$

where $\sigma^2 = 1/n \sum_{i=1}^{n} \bar{\mathbf{X}}_i^{\mathrm{T}} \bar{\mathbf{X}}_i$ and the mean square error of the data points $\mathrm{MSE} = 1/n\sum_{i=1}^{n} \mathbf{d}_i^{\mathrm{T}}\mathbf{d}_i$, where $\mathbf{d}_i = \mathbf{X}_i - \mathbf{QA}_i$ and subscript i denotes the ith column of the matrix.

We also compute the Pearson product−moment correlation coefficient (PCC) which is a measure of the linear dependence between an input spectrum and the same spectrum that has undergone a compression−decompression cycle. A value close to 1 indicates a strong linear dependence and, hence, a high quality of signal recovery.

Image Segmentation with *k*-Means. After compression, *k*-means cluster-based segmentation was performed using the function *kmeans* in MATLAB (32 bit, Mathworks, Natick, MA). The basis generated using Algorithm 1 preserves relative point-to-point distances; therefore, clustering directly on the compressed data is possible.

Computational Hardware. All computation was performed using Matlab R2009a (32 bit, Mathworks, Natick, MA) on an AMD Athlon II X4 620 processor (2.6 GHz, with 5Gb RAM, running Windows 7 Professional (64 bit)).

Data Collection. A range of datasets have been selected that fully illustrate the fundamental properties of these algorithms. The datasets are summarized in Table S1 in the Supporting Information and consist of hyperspectral image datacubes acquired from visible light reflectance, confocal Raman microscopy, and matrix-assisted laser desorption ionization (MALDI) mass spectrometry. A schematic illustration of each dataset is shown in Figure S1 in the Supporting Information. In general, a hyperspectral imaging dataset is represented as a three-dimensional (3D) "datacube" with two spatial axes and one spectral axis. For this work, we reorganize this into the two-dimensional (2D) matrix $\mathbf{X}$ described previously with each column containing the spectrum for a single pixel. Depending on the imaging methods used, the data are collected row-wise (by channel) or column-wise (by pixel).

The focus of this paper is on data processing methods rather than particular features within individual datasets. The data were chosen to illustrate the application to different type of hyperspectral data, but all are common analytical techniques that rapidly collect large hyperspectral data volumes.

■ RESULTS AND DISCUSSION

Spectral Images. Mass spectrometry is one of the highest-dimensional analytical techniques routinely used in analytical chemical imaging, and data reduction techniques are routinely employed in its analysis.[8,27,31] We will first demonstrate the application of the randomized algorithms to this type of data and then show its further application to optical and Raman datasets. The dataset used here has a dimensionality (number of mass channels) of 129 796 (from Table S1 in the Supporting Information). The mass spectrometry image was collected in the small molecule (m/z 50−1000) region, using an orthogonal quadrupole time-of-flight instrument (QStar Elite QqTOF, AB Sciex, Warrington, U.K.). This configuration means that the detector was the main source of noise in the data. No assumptions are made regarding peak shape or noise characteristics during compression: just that there is some degree of spectral degeneracy.

We investigated the performance of randomized methods in four ways. First, we investigated their use as a compression scheme, by examining the degree of compression that could be obtained and the quality of decompressed data versus the original data. Second, we studied the use of matrix factorization on the data, in the context of PCA. In particular, we verified that PCA on the compressed data yields identical results to PCA on raw data. Third, we investigated the effect of pixel subsampling on the quality of compression. Finally, we applied k-means clustering in order to demonstrate a simple segmentation on the compressed data.

Data Compression. We investigated the efficiency and quality of compression on the mass spectrometry dataset. Algorithm 1 was applied to the data to generate a randomized basis matrix. Since the size of the data set prevented it from being loaded into memory as a whole, the data matrix $\mathbf{X}$ and its random projections $\mathbf{S}$ were formed piecewise by loading in a portion of the dataset at a time. This was done using the imzMLConverter tool.[32] The resulting randomized basis matrix $\mathbf{Q}$ was used to generate a reduced data matrix $\mathbf{A} = \mathbf{Q}^T\mathbf{X}$. The reduced data was then decompressed to form $\mathbf{X}' = \mathbf{QA}$. A projection dimension of $k = 100$, producing a compression ratio of 0.01, was chosen and was found to adequately represent the data, giving a PCC of >0.99 and an SNR of 45. The result of this process is shown alongside the original data in Figure 2.

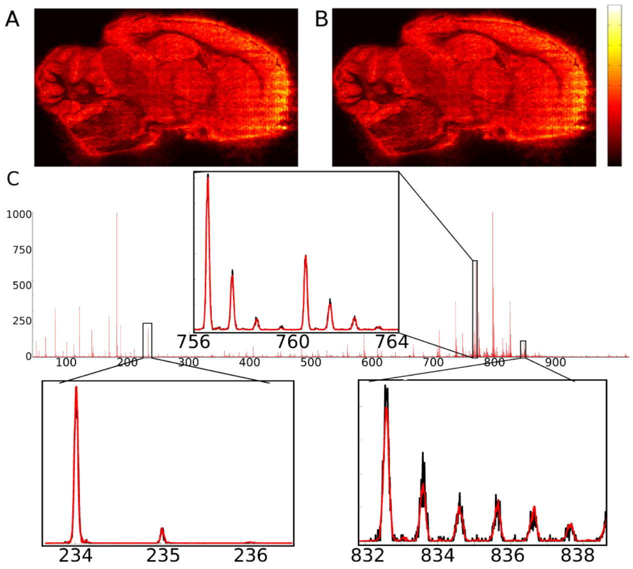


Figure 2. Decompressed MALDI image. (A) Channel map formed from raw data of m/z = 782.55 ± 0.05 showing the distribution of a common lipid, PC(34:1).[33] (B) Channel map showing the distribution following a single compress–decompress cycle of PC(34:1) with k = 100. (C) Overlay of raw (solid black) and decompressed (red) spectra from a single pixel, showing that there is no substantial deviation following decompression; enlargements of specific peaks show that peak shape and intensity is maintained regardless of the initial peak m/z or intensity and the two spectra are still all but indistinguishable.

Qualitatively, this figure shows that the image representation of the selected ion channel is visually indistinguishable from the raw data following a compression–decompression cycle and that decompression of individual spectra show only small differences at the level of the noise. The compression procedure reduced the dataset from the single matrix $\mathbf{X}$ with $m \times n$ = 129 796 × 20 535 = 2 665 360 860 elements to a pair of matrices $\mathbf{Q}$ and $\mathbf{A}$ with $k(m+n)$ = 100 × (129 796 + 20 535) = 15 033 100 elements, giving a compression ratio of R_c = 0.0056. The raw data is ~20 GB in size, which is reduced to ~115 MB with this method.

To quantify the quality of compression, we computed the signal-to-noise (SNR) of the spectra and the Pearson's product–moment correlation coefficient (PCC) between the raw and decompressed data. This was done for a range of k-values in order to investigate the trade-off between data size and compression quality. The results of this are shown in Figure 3. The SNR and PCC are both seen to be positively correlated with the compression ratio, indicating that taking a large value of k does increase the data quality, but the PCC increases rapidly at first and then flattens out very quickly. The SNR continues to increase, but reaches acceptable values at low compression ratios. In optical hyperspectral imaging, SNR values of >30 for lossy compression are typically considered to be good, and values of ≥50 are considered to be excellent.[30,34,35] An SNR of 30 is achieved for a compression ratio of <0.002, corresponding to k < 35 on this dataset. For the example in Figure 2 with k = 100, the SNR is ~45 and the PCC is >0.99. This suggests that the information lost from the data is at the level of the noise.

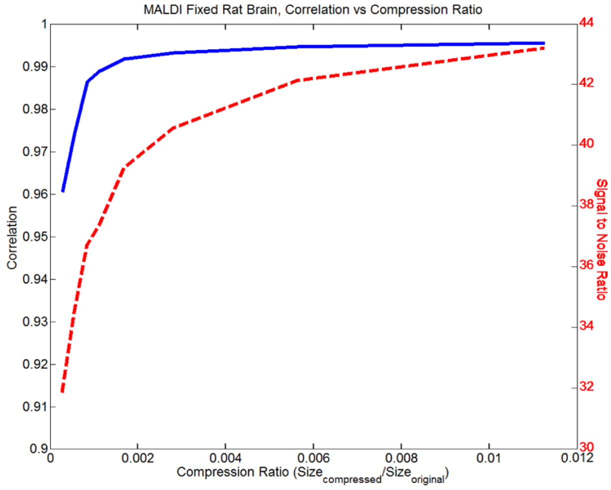


Figure 3. Evaluation metrics for the MALDI mass spectrometry image, shpowing the Pearson product–moment correlation coefficient (PCC) (solid blue, left axis) and the signal-to-noise ratio (SNR) (dashed red, right axis). Since the PCC values tended toward one, high-quality signal recovery is achieved.

Comparisons to other mass spectrometry data reduction schemes are difficult, because they tend to be based on peak-picking procedures, which can be tuned to pick an appropriate number of peaks to compress the data to the size dictated by the computer's memory. Measures of compression quality for the peak-picking methods are not known, but it is commonly accepted that the process of rebinning and peak picking does cause information loss and most efforts have focused on not discarding "informative" peaks.[8] Extracting 50–200 peaks has been suggested to be appropriate for further analysis such as segmentation,[36] which corresponds to a sample-to-feature ratio of ~10 for image sizes typically collected from MALDI time-of-flight experiments. For high-resolution instruments, this may require discarding the majority of detectable peaks. The main advantage of the randomized methods is that the dimensionality of the data can be reduced to a similar level as that obtained by peak-picking methods, but no part of the data is removed completely. Furthermore, the dimensionality reduction obtained via these means is reversible. The discrete wavelet transform, which is a more comparable method, has been employed in an attempt to preserve the spectral integrity of the data.[37] This was shown to reduce the dimensionality from 6490 to 819; however, details of the total data size and metrics of compression quality were not presented. On this limited basis, we believe that randomized basis approximation is able to achieve superior compression ratio to wavelet-based methods while maintaining the quality of the data.

Compressed Factorization. We now investigate the application of a matrix factorization algorithm on compressed hyperspectral image data. There is keen interest in multivariate analysis on mass spectrometry imaging data, with PCA being a popular choice.[7,26] All applications of PCA for mass spectrometry data have required reducing the dimensionality of the data to a manageable level before PCA can be performed. Recent work on making PCA tractable for mass spectrometry images has included methods for peak selection,[8,38] spatial summation,[26] and memory-efficient methods for PCA.[29] As a result, PCA is not routinely performed on entire mass spectrometry images and the effects of data reduction cannot be quantitatively evaluated.

We have evaluated compressed PCA (Algorithm 3) against PCA on uncompressed data to demonstrate that equivalent results are obtained. This is not possible on the raw MALDI mass spectrometry image summarized in Table S1 in the Supporting Information, because of the large image size, so the MALDI rat brain image was loaded into memory after being spectrally rescaled[8,31] at $\Delta m/z$ = 0.2, resulting in 4808 spectral channels. Note that it is necessary to do this only for the purposes of comparison with standard PCA; the proposed compressed PCA can process the full dataset. PCA was performed on this directly using the *princomp* function in MATLAB. We then performed PCA on the compressed data, using Algorithm 1 and Algorithm 3, for a range of *k*-values, corresponding to different compression ratios. The first five principal components were compared, corresponding to 95% of the variance in the data.

The error metrics of SNR and PCC were calculated between the PCA coefficients generated directly and those generated using compressed PCA. Figure 4 shows how the SNR and PCC values vary with the compression ratio, and a direct comparison of the scores and coefficients between PCA on the original data and compressed PCA using a compression ratio of 0.0026. Both the PCC and SNR metrics show an improvement as the compression ratio increases, but the PCC increases more rapidly and plateaus at a value of >0.99. Once the data have been sufficiently highly sampled, the variance from the median PCC tends to zero and the probability of isolated outliers is low; below this value, both the variance and the PCC value increase smoothly. The SNR continues to improve, even after the PCC plateau is reached, suggesting that any discrepancies thereafter are on the order of the noise.

With the exception of the very lowest compression ratios chosen (corresponding to k < 10), the median value of the average PCC between the PC coefficients obtained from the two methods over 10 repeats was >0.99. We conclude that, provided that a sufficient value of k has been chosen (this can be determined by evaluating the error metric M), PCA performed on the compressed data gives the same results as PCA performed on the raw dataset, within the noise limits of the data.

It is also instructive to investigate the computation time required by the two methods. Using PCA on the uncompressed data, the computation took 1.5 h, using a mass spectrometry workstation (described in the Methods section). This did not include time taken to load/save the data. Using the compressed method, PCA took <8 s. This included generation of the basis, compression of the data and back-projection of the eigenvector coefficients, i.e., all stages of Algorithm 1 and Algorithm 3, such that the input and output of the two methods are directly comparable. Therefore, we conclude that the additional overhead required to compress and decompress the data is far less than the benefits gained from being able to perform PCA on a much smaller data matrix.

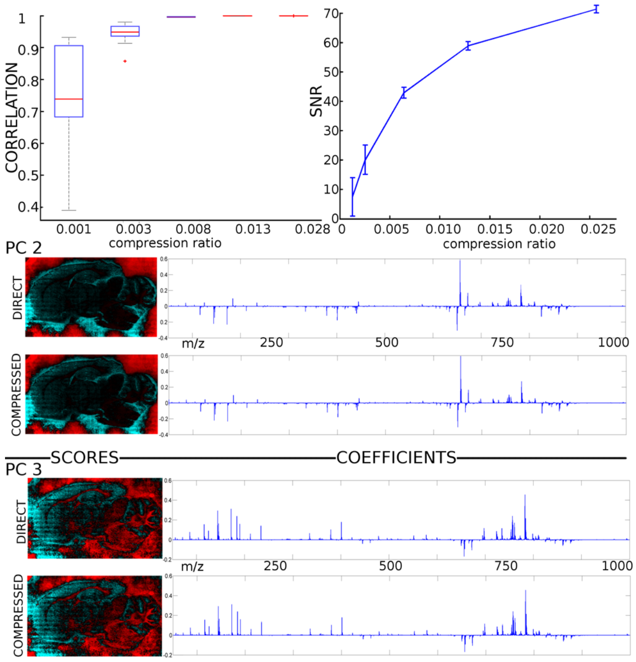


Figure 4. Comparison of direct and compressed PCA (compression ratio 0.0026) on a single dataset. Top: PCC and SNR error metrics (boxplot and error bars show variance from 10 repeats). Bottom: Scores and coefficients for principal components 2 and 3. High SNR and PCC coeffients of >0.99 show that the PCA coefficients generated on compressed data and decompressed are almost identical to coefficients generated directly. This can be seen visibly by examining the scores and coefficients from PC2 and PC3 (PC1 is also identical; see Figure S4 in the Supporting Information).

We also note that the mean-centring of the data that is required can be performed while the data are compressed, because of the choice of a symmetric distribution for the random matrix. This is useful as the data must be passed twice (once to calculate the mean and again to subtract it), and being able to do this on the compressed data is a considerable advantage.

Segmentation with *k*-Means Clustering. The size and complexity of mass spectrometry images, particularly from biological samples, has led to concerted efforts to provide user-friendly "at a glance" information content visualizations. The most commonly used technique is segmentation, which divides an image into color-coded regions that are "similar", sacrificing specific spectral information for spatial clarity. Visualization approaches have been demonstrated for extracting subtle data features;[31] however, an advantage of segmentation is that a finite number of distinct image regions are produced, with characteristic spectra that are directly physically interpretable.[36] Clustering provides an automated method for segmenting an image by grouping pixels with similar spectra together. Multiple similarity measures exist but the Euclidean distance is most commonly used and recommended for clustering.[39,40] Pixels are usually considered independently during clustering, so groups can be spatially discontinuous; however, the use of spatial preprocessing has been shown to improve the appearance of clustering results.[41]

We have investigated segmentation of the MALDI rat brain image directly from the compressed data, employing the popular k-means clustering,[6,7,42] with 3−6 clusters being tested. Algorithm 1 was applied for compression, using k = 100, then clustered using the MATLAB function *kmeans*. Since this is a projection of the data that preserves distances between points, k-means clustering should be effective on this directly.[14]

The results of the segmentation using six clusters is shown in Figure 5. The tissue is divided along borders corresponding to known anatomical features, including the cerebellum in cluster 5 (yellow); the frontal cortex in cluster 1 (dark blue); the striatum in cluster 4 (green); and disconnected brain regions including the white matter of the cerebellum, optic chasm, and corpus collossum in cluster six (orange). The mean spectrum for each cluster is generated without total data decompression by simply backprojecting the mean projection for a set of pixels, this would not be possible with nonorthogonal random projections. The mean spectra from clusters 1 and 6, which are the two most dissimilar clusters determined by the Euclidean distance between the cluster centers, are shown in Figure 5, and distinct spectral differences between the two regions can be seen alongside some common peaks. An investigation into the specific molecular differences between these regions is beyond the scope of this paper but the workflow is seen to provide a rapid route to visual inspection of data distributions and spectral profiles from important image regions.

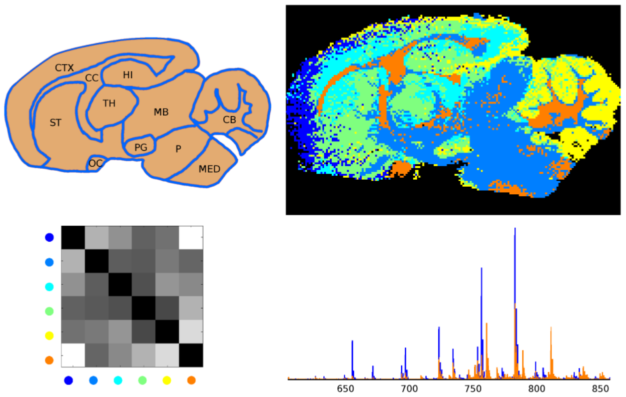


Figure 5. MALDI mass spectrometry image segmentation: k-means segmentation was used to segment the image into six regions and provide a simple map of spectral heterogeneity. (Top left) Key histological structures (cerebellum (cb), cortex (ctx), hippocampus (hi), medulla (med), midbrain (mb), optic chiasm, pituitary gland (pg), pons (p), striatum (st), and thalamus (th)). (Top right) Segmentation map dividing the brain region into six areas. (Bottom left) Euclidean distance between the cluster centroids shown in the map (colored accordingly). (Bottom right) Mean mass-spectra from clusters 1 and 6 (blue and orange areas, respectively), showing clear differences in the peaks present from the lipid region.

Other clustering methods using the Euclidean distance such as hierarchical clustering[36,40] could also potentially benefit from random matrix-based compression, and future work will include a thorough investigation of this and other methods.

Performance Improvements with Pixel Subsampling. Algorithm 4 describes a performance improvement for basis approximation that allows a small subset of the data to be used to generate the reduced basis for the data. This method takes advantage of spatial homogeneity to reduce the number of points that are required to span the subspace occupied by the data. We investigated the use of Algorithm 4, using a range of spatial subsampling rates and compression ratios to generate the randomized basis. When subsampling spatially, a (randomly chosen) subset of the pixels is used instead of the entire data matrix to generate the projection basis. The PCC and SNR values are shown in Figure 6. We see that the compression quality stabilizes at a sampling rate of <20% in all cases. The compression ratio has little effect on the sampling rate needed, but improves the quality of compression independently.

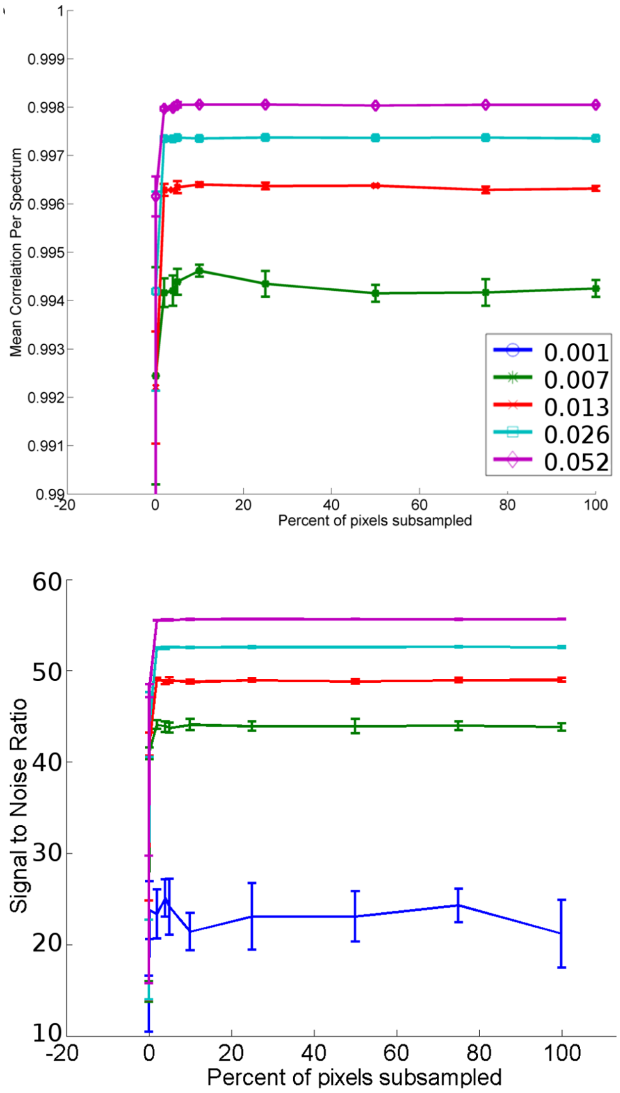


Figure 6. Analysis of compression quality obtained using spatial subsampling from the reduced MALDI mass spectrometry data set with a range of compression ratios: (top) Pearson correlation coefficient (PCC) and (bottom) signal-to-noise ratio (SNR). A random chosen subset of pixels was used in every trial. Error bars show variation (one standard deviation) from 10 trials. Both the PCC and SNR stabilize after the subsampling exceeds 10%, regardless of the compression ratio; again, a higher compression ratio provides an improvement in compression quality.

These results are sample-specific, being dependent on the degree of spatial homogeneity in the sample.

Therefore, it is not necessary to read in the entire dataset in order to generate the randomized basis. It will, of course, still be necessary to read the entire dataset in order to perform the projection, but the use of random sampling means that all of the data need only be read in once. This is likely to reduce the time required to process very large datasets significantly, where disc access time is a significant cost.

Application to Spectral Optical and Raman Imaging. While MALDI mass spectrometry is a particularly challenging hyperspectral imaging method, because of the very high dimensionality of its datasets, other spectral imaging methods may also benefit from randomized methods. We applied the randomized compression method to two further hyperspectral datasets: an optical spectroscopy image of leaves undergoing autumnal changes (361 920 pixels in 320 spectral channels), and a Raman microscopy image of an artificial ligament bound to a bruschite anchor (22 500 pixels in 1024 channels) (see Table S1 in the Supporting Information). Data compression was performed on both datasets using Algorithm 1 with $k = 100$. The same software was used for all datasets, with the exception of the module that reads in the datasets. The decompression is presented for visual comparison in Figure 7, and the SNR and PCC values are shown in Figure S3 in the Supporting Information. Using a value of $k = 100$ gave compression ratios of 0.31 for the optical data and 0.10 for the Raman data, SNRs of ~45 in both cases, and correlations of >0.99 for the Raman data and >0.97 for the optical data. A visual comparison between raw and decompressed channel images (panels 1A and 1B in Figure 7 for the optical data and panels 2A and 2B in Figure 7 for the Raman data) shows no visible differences for either the optical or Raman images. Comparing overlaid single pixel spectra (panels 1C and 2C in Figure 7), we see some noise in the decompressed optical signal, when compared to the raw signal. The Raman spectra are virtually indistinguishable, even at high magnification.

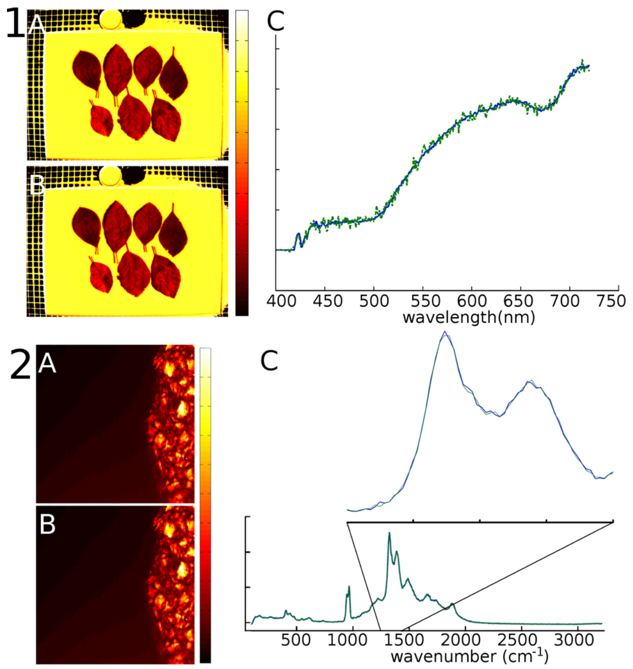


Figure 7. Data compression quality for (1) optical and (2) Raman hyperspectral images. Shown are (A) raw channel image (550 ± 20 nm and 1320 ± 20 cm^{-1}, respectively); (B) decompressed channel image; and (C) overlaid raw and decompressed spectra from a single pixel. Data compressed using Algorithm 1 and $k = 100$.

We then investigated compressed factorization on these two datasets using PCA. Algorithm 4 was used to calculate the PCA coefficients and scores for each image, features within the score images were used to highlight image regions and the coefficients for this region then examined. The scores and coefficient plots are shown in Figure S2 in the Supporting Information. In the PCA scores from the optical dataset, all leaves are clearly separated from the background in the first component, where they score negatively, and leaves at advanced stages of the autumn process are identifiable in the second component, where they score positively. Green leaves score negatively in both. Comparing the corresponding regions of the coefficient plot suggest that this is due to a feature at a wavelength near 550 nm, which corresponds to the chlorophyll absorption peak. In the Raman dataset, the ligament is clearly separated from the bruschite scaffold which itself is visible as a mixture of grains of phosphorus-containing material surrounded by the calcium matrix. The phosphor-containing portion is positive in both components, while the ligament is negative in the second component. Examining this region in the coefficients shows peaks in the range 700−900 cm^{-1}, which corresponds to the known scattering from the phosphor−oxygen (P−O) bond.

■ CONCLUSION

We have demonstrated a random matrix-based method for the compression and analysis of hyperspectral images on datasets from MALDI mass spectrometry, hyperspectral optical imaging, and Raman microscopy.

The results presented in this paper demonstrate that randomized basis approximation and factorization are a powerful tool in the analysis of the sorts of large numerical datasets that hyperspectral imaging can provide. The key advantages of these methods are that no data are explicitly discarded, the compression is reversible, and computations can be performed on the compressed data without any loss of information. The randomized methods give particularly good results when the dataset is of low numerical rank; that is, there is significant homogeneity within the data. Very heterogeneous samples will have a lesser degree of linear dependence and will be less amenable to this approach, although one would still expect some improvement. No knowledge of instrument response or noise characteristics was required for effective compression, and including these would be expected to improve the compression achieved, by increasing intraspectral covariance. Recent work[29] has proposed a memory-efficient method for performing PCA. This method still required the number of spectral channels to be reduced, but can handle many pixels arbitrarily (the methods of this paper will have a large, but finite, limit on the number of pixels that can be processed). We believe that it should be possible to combine these strategies in order to effectively remove all limits on the size of the datasets that can be processed.

In order to facilitate further evaluation and uptake of these methods, the demonstration code for employing basis approximation for spectral compression and a sample dataset have been included in the Supporting Information.

■ ASSOCIATED CONTENT

Ⓢ Supporting Information

This material is available free of charge via the Internet at http://pubs.acs.org.

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: i.b.styles@cs.bham.ac.uk.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

We are grateful to the EPSRC, for funding through a studentship to A.P. in the PSIBS doctoral training centre (EP/F50053X/1); Alistair Bannerman and Liam Grover, for providing the Raman dataset; and Claire Carter, for providing the MALDI mass spectrometry dataset.

■ REFERENCES

(1) Govender, M.; Chetty, K.; Bulcock, H. *Water SA* **2007**, *33*, 145−152.

(2) Umbehr, M.; Bachmann, L.; Held, U.; Kessler, T.; Sulser, T.; Weishaupt, D.; Kurhanewicz, J.; Steurer, J. *Eur. Urol.* **2009**, *55*, 575−591.

(3) Amstalden van Hove, E.; Smith, D.; Heeren, R. *J. Chromatogr. A* **2010**, *1217*, 3946−3954.

(4) Donoho, D. *AMS Math Challenges Lectures*, 2000; pp 1−32.

(5) Somorjai, R.; Dolenko, B.; Baumgartner, R. *Bioinformatics* **2003**, *19*, 1484−1491.

(6) Gowen, A.; Marini, F.; Esquerre, C.; O'Donnell, C.; Downey, G.; Burger, J. *Anal. Chim. Acta* **2011**, *705*, 272−282.

(7) McCombie, G.; Staab, D.; Stoeckli, M.; Knochenmuss, R. *Anal. Chem.* **2005**, *77*, 6118−6124.

(8) Fonville, J. M.; Carter, C.; Cloarec, O.; Nicholson, J. K.; Lindon, J. C.; Bunch, J.; Holmes, E. *Anal. Chem.* **2011**, *84*, 1310−1319.

(9) Vidal, M.; Amigo, J. *Chemom. Intell. Lab. Syst.* **2012**, *117*, 138−148.

(10) Plaza, A.; Benediktsson, J.; Boardman, J.; Brazile, J.; Bruzzone, L.; Camps-Valls, G.; Chanussot, J.; Fauvel, M.; Gamba, P.; Gualtieri, A.; Marconcini, M.; Tilton, J.; Trianni, G. *Remote Sens. Environ.* **2009**, *113*, S110−S122.

(11) Bingham, E.; Mannila, H. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001; pp 245−250.

(12) Lindenstrauss, J.; Pelczynski, A. *Studia Math* **1968**, *29*, 193.

(13) Dasgupta, S.; Gupta, A. *An Elementary Proof of the Johnson−Lindenstrauss Lemma*, Technical Report TR-99-006; International Computer Science Institute, Berkeley, CA, 1999.

(14) Fowler, J.; Du, Q.; Zhu, W.; Younan, N. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, 2009 (IGARSS 2009)*, 2009; Vol. 5, Paper V-76.

(15) Varmuza, K.; Engrand, C.; Filzmoser, P.; Hilchenbach, M.; Kissel, J.; Krüger, H.; Silén, J.; Trieloff, M. *Anal. Chim. Acta* **2011**, *705*, 48−55.

(16) Fowler, J. *IEEE Trans. Image Process.* **2009**, *18*, 2230−2242.

(17) Zhang, J.; Erway, J.; Hu, X.; Zhang, Q.; Plemmons, R. *J. Electr. Comput. Eng.* **2012**, DOI: 10.1155/2012/409357.

(18) Lin, J.; Gunopulos, D. In *Proceedings of the Text Mining Workshop*, 3rd SIAM International Conference on Data Mining, 2003.

(19) Varmuza, K.; Filzmoser, P.; Liebmann, B. *J. Chemom.* **2010**, *24*, 209−217.

(20) Durrant, R.; Kabán, A. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010; pp 1119−1128.

(21) Yang, J.; Wright, J.; Huang, T.; Ma, Y. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008*; 2008; pp 1−8.

(22) Basevi, H.; Tichauer, K.; Leblond, F.; Dehghani, H.; Guggenheim, J.; Holt, R.; Styles, I. *Biomed. Opt. Express* **2012**, *3*, 2131−2141.

(23) Shi, Z.; Liu, L.; Zhai, X.; Jiang, Z. *Neural Comput. Appl.* **2012**, 1−13.

(24) Fern, X. Z.; Brodley, C. E. In *Machine Learning—International Workshop Then Conference*, 2003; Vol. 20, No. 1, p 186.

(25) Halko, N.; Martinsson, P.; Tropp, J. *SIAM Rev.* **2011**, *53*, 217−288.

(26) Henderson, A.; Fletcher, J.; Vickerman, J. *Surf. Interface Anal.* **2009**, *41*, 666−674.

(27) McDonnell, L.; Van Remoortere, A.; De Velde, N.; Van Zeijl, R.; Deelder, A. *J. Am. Soc. Mass Spectrom.* **2010**, *21*, 1969−1978.

(28) Deegalla, S.; Bostrom, H. In *5th International Conference on Machine Learning and Application (ICMLA '06)*, 2006; pp 245−250.

(29) Race, A. M.; Steven, R. T.; Palmer, A. D.; Styles, I. B.; Bunch, J. *Anal. Chem.* **2013**, *85*, 3071−3078.

(30) Tang, X.; Pearlman, W.; Modestino, J. *IEEE Int. Conf. Image Process.* **2004**, 1133−1136.

(31) Fonville, J. M.; Carter, C. L.; Pizarro, L.; Steven, R. T.; Palmer, A. D.; Griffiths, R. L.; Lalor, P. F.; Lindon, J. C.; Nicholson, J. K.; Holmes, E.; Bunch, J. *Anal. Chem.* **2012**, *85*, 1415−1423.

(32) Race, A.; Styles, I.; Bunch, J. *J. Proteom.* **2012**, *75*, 5111−5112.

(33) Carter, C.; McLeod, C.; Bunch, J. *J. Am. Soc. Mass Spectrom.* **2011**, 1−8.

(34) Du, Q.; Fowler, J. E. *Geoscience and Remote Sensing Letters, IEEE* **2007**, *4* (2), 201−205.

(35) Galli, L.; Salzo, S. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, IGARSS'04*, 2004; p *1*.

(36) Alexandrov, T. *BMC Bioinform.* **2012**, *13*, 1−13.

(37) Van de Plas, R.; De Moor, B.; Waelkens, E. *Proc. 2008 ACM Symp. Appl. Comput.* **2008**, 1307−1308.

(38) McDonnell, L.; van Remoortere, A.; van Zeijl, R.; Dalebout, H.; Bladergroen, M.; Deelder, A. *J. Proteom.* **2010**, *73*, 1279−1282.

(39) Deininger, S.; Ebert, M.; Fütterer, A.; Gerhard, M.; Röcken, C. *J. Proteom. Res.* **2008**, *7*, 5230−5236.

(40) Deininger, S.; Becker, M.; Suckau, D. *Methods Mol. Biol.* **2010**, *656*, 385−403.

(41) Alexandrov, T.; Becker, M.; Deininger, S.; Ernst, G.; Wehder, L.; Grasmair, M.; von Eggeling, F.; Thiele, H.; Maass, P. *J. Proteom. Res.* **2010**, *9*, 6535−6546.

(42) Jones, E.; van Remoortere, A.; van Zeijl, R.; Hogendoorn, P.; Bovée, J.; Deelder, A.; McDonnell, L. *PloS One* **2011**, *6*, No. e24913.