Memory Efficient Principal Component Analysis for the Dimensionality Reduction of Large Mass Spectrometry Imaging Datasets

Race, Alan; Steven, Rory; Palmer, Andrew; Styles, Iain; Bunch, Josephine

DOI:
10.1021/ac302528v

Document Version
Peer reviewed version

Citation for published version (Harvard):
Race, A, Steven, R, Palmer, A, Styles, I & Bunch, J 2013, 'Memory Efficient Principal Component Analysis for the Dimensionality Reduction of Large Mass Spectrometry Imaging Datasets' Analytical Chemistry, vol 85, no. 6, pp. 3071-3078. DOI: 10.1021/ac302528v

Memory Efficient Principal Component Analysis for the Dimensionality Reduction of Large Mass Spectrometry Imaging Datasets

Alan M. Race,[†,‡,¶] Rory T. Steven,[†,¶] Andrew D. Palmer,[†,‡,¶] Iain B. Styles,[*,‡] and Josephine Bunch[*,†,¶]

Centre for Physical Sciences of Imaging in the Biomedical Sciences, School of Chemistry, University of Birmingham, Edgbaston, Birmingham, B15 2TT, United Kingdom., School of Computer Science, University of Birmingham, Edgbaston, Birmingham, B15 2TT, United Kingdom., and School of Chemistry, University of Birmingham, Edgbaston, Birmingham, B15 2TT, United Kingdom.

E-mail: i.b.styles@cs.bham.ac.uk; j.bunch@bham.ac.uk

Abstract

A memory-efficient algorithm for the computation of Principal Component Analysis (PCA) of large mass spectrometry imaging data sets is presented. Mass Spectrometry Imaging (MSI) enables two- and three- dimensional overviews of hundreds of unlabeled molecular species in complex samples such as intact tissue. PCA, in combination with data binning or other reduction algorithms, has been widely used in the unsupervised processing of MSI data and as a dimentionality reduction method prior to clustering and spatial segmentation. Standard

[*]To whom correspondence should be addressed
[†]PSIBS, University of Birmingham
[‡]School of Computer Science, University of Birmingham
[¶]School of Chemistry, University of Birmingham

implementations of PCA require the data to be stored in random access memory. This imposes an upper limit on the amount of data that can be processed, necessitating a compromise between the number of pixels and the number of peaks to include. With increasing interest in multivariate analysis of large 3D multi-slice datasets and ongoing improvements in instrumentation, the ability to retain all pixels and many more peaks is increasingly important. We present a new method which has no limitation on the number of pixels and allows an increased number of peaks to be retained. The new technique was validated against the MATLAB (The MathWorks Inc., Natick, Massachusetts) implementation of PCA (*princomp*) and then used to reduce, without discarding peaks or pixels, multiple serial sections acquired from a single mouse brain which was too large to be analysed with princomp. *k*-means clustering was then performed on the reduced dataset. We further demonstrate with simulated data of 83 slices, comprising 20535 pixels per slice and equalling 44 GB of data, that the new method can be used in combination with existing tools to process an entire organ. MATLAB code implementing the memory efficient PCA algorithm is provided.

Introduction

Matrix assisted laser desorption/ionisation (MALDI) mass spectrometry imaging (MSI) is a sensitive technique allowing localisation and identification of unlabelled molecules in samples. The technique has been applied to a large range of analytes, such as drugs,[1,2] lipids,[3,4] peptides,[5] proteins[6] and metabolites[1,2] from many different tissue types, which are often single organ sections. MSI data are stored as a grid of spectra, where each pixel has an associated mass spectrum. Distributions of single analytes (single *m/z* peaks) of interest can be visualised by false colour ion images, created by assigning the intensity at each pixel to the value of the area under the peak in the spectrum.

MSI experiments can produce extremely large datasets, for example a 4 cm $\times$ 4 cm MALDI target plate imaged at a pixel size of 100 $\times$ 100 μm results in 160k pixels and if 100 kB per spectrum is assumed (6400 *m/z*-intensity pairs) then the dataset would be approximately 15.26

GB. Imaging the same area at high resolution (10 μm)[7] would result in over 16M pixels, and a potential raw data size of approximately 1.49 TB (1529 GB). Several applications require imaging to be performed on even larger sample areas, such as whole animal sections[1,6,8] or 3D volumes.[9–17]

The data size of a single spectrum is dependent on the mass resolving power of the instrument and the mass range of interest. Instruments such as the MALDI-FTICR can have mass resolutions of orders of magnitude greater than that of MALDI-TOF instruments and so when acquiring data over the same *m/z* range, can produce significantly more data.[2,18,19] A common way to reduce the size of data stored per spectrum is to store the data as *m/z*-intensity pairs rather than storing a value at every possible *m/z* location to reduce the amount of redundancy. This then introduces a further variable which determines the size of a single spectrum, the number of species detected (which is, in turn, a function of the sample type, the ionisation efficiency and/or the degree of fragmentation).

The problem of handling large secondary ion mass spectrometry (SIMS) datasets has been tackled by compressing the data to ensure that it fits in RAM,[20,21] but data size limits will continue to exist for algorithms which cannot utilise such compression and will again become problematic as improvements in imaging technology further increase the data size. Alternative data reduction strategies aim to isolate only peaks of interest and eliminate noise to reduce the amount of uninformative data used in further analysis.[22,23] However, hundreds or thousands of peaks can be detected, and so further reduction is often necessary. Principal component analysis (PCA) is a mathematical technique that can be used to solve this problem by reducing dimensionality while retaining variance within the data.[24]

When performing PCA via conventional means, the following steps are typically followed. (i) The dataset (N pixels, M peak intensities) is read into RAM as an $N \times M$ matrix. (ii) The mean spectrum (over the whole dataset) is subtracted from each spectrum in the dataset. (iii) Singular value decomposition (SVD) of the data matrix is then performed to determine eigenvalues and eigenvectors (also referred to as the loadings or coefficients in PCA). (iv) The data are then projected onto the space defined by the eigenvectors to determine the scores. This is exactly how the often used implementation of PCA *princomp* (as supplied by MATLAB Statistics Toolbox)

is performed. Implementations like these require the data matrix, along with multiple additional variables which are the same size as the data matrix, to be stored simultaneously in RAM in order to perform the full calculation. The finite size of RAM can easily be exceeded by the size of MALDI MSI datasets and this implementation severely restricts the size of the dataset that can be processed.

The memory limit can be reached through having a large number of peaks, a large number of pixels, or both and so a tradeoff has to be made that is dependent on the amount of RAM available for analysis.[25] Reducing the *m/z* dimension is commonly achieved through binning, however peak detection and alignment is a much more robust method of avoiding the loss of information while reducing the data.[14] Dependent on the size of the data, further reduction can be necessary prior to multivariate analysis, and so methods of selectively discarding peaks and pixels have also been developed.[23,26] This reduces both spectral (potentially merging peaks) and spatial (potentially merging features) resolution to a point which may be deemed unacceptable.[27] An alternative method to solve the memory issues implicit in PCA is to utilise sparse matrix storage.[25] However, it is entirely possible that a data set in sparse matrix form is still too large to be stored in RAM and so in these cases some form of data reduction will still be required prior to the sparse storage such as removing intensity values below a user-defined threshold or performing binning spectrally or both spatially and spectrally.

All of these data reduction methods do provide a useful temporary solution, but as the move is made towards high resolution 3D data sets the problem will again return and so algorithms that are explicitly designed to handle large datasets will become more desirable. The importance of memory optimised algorithms applied to MSI data was recently commented upon by Alexandrov and Kobarg[28].

Clustering techniques are becoming an invaluable tool in the processing and interpretation of MSI data sets and have been the focus of many recent articles.[17,24,28,29] PCA has been shown to be a useful technique prior to clustering due to the reduced dimensionality and noise suppression,[24,29] however if meaningful peaks are discarded during the data reduction step prior to PCA then the

subsequent clustering process may lose the ability to separate subtly distinct regions. PCA is also commonly employed as an unsupervised technique to objectively determine trends within the data,[30–35] which again may result in the inability to detect subtly different trends if the initial data is incomplete. This use of PCA is often considered to be controversial due to the inability to relate negative principal component loading values to experimental *m/z* signals and so other multivariate analysis techniques such as probabilistic latent semantic analysis (pLSA) and non-negative matrix factorisation (NNMF) have been applied to MSI data.[27] However, despite the fact that the workflow presented is applicable to both uses of PCA, the focus in this work is on the use as a dimensionality reduction technique only.

An efficient implementation of PCA has been developed for large datasets, stored as databases, without the requirement for the entire data matrix to be stored in memory at once.[36] In this paper we develop this method for the analysis of MSI data stored in the open mass spectrometry imaging format imzML.[37] We use the new technique to analyse a previously prohibitively large dataset, multiple mouse brain sections forming a 3D volume, while retaining all detected peaks and all pixels in the image.

Methods

Detailed materials, sample preparation and mass spectrometry experimental description are given in the Supporting Information. Briefly, a single rat brain section was coated in α-cyano-4-hydroxycinnamic acid using an automated matrix deposition system (TM sprayer from HTX Technologies, NC, U.S.A.)[3] and 12 sections of mouse brain were coated in *para*-Nitroaniline using an artistic air-brush.[38] MALDI MSI data were acquired on a QSTAR Elite QqTOF (AB Sciex, Warrington, UK) for the single section rat brain image and a QSTAR XL QqTOF (AB Sciex) for the 3D mouse brain image.

Data Processing

Data processing was performed using an Intel i7 2600 (3.40 GHz) with 8 GB RAM. All data were converted from proprietary format to the open mass spectrometry format mzML using AB SCIEX MS Data Converter beta version 1.1 (AB Sciex). Data in the mzML format was converted to the open mass spectrometry imaging format imzML using imzMLConverter.[39] Further data processing was performed in MATLAB version 7.12.0.635 with Statistics Toolbox (The MathWorks Inc., Natick, Massachusetts). *k*-means clustering was performed on the first 40 principal components of the PCA reduced data (explaining 99.14% of the variance). The variance cut-off is an arbitrary choice, where the assumption is that the remaining 0.86% variance within the dataset is noise. This value works well in practice, however the noise level should always be underestimated (and therefore as much variance as possible is retained), to avoid discarding components that capture information rather than purely noise. The *k*-means algorithm supplied as part of MATLAB was used, with k=2...10 (shown in Figure S2) with k=7 selected as optimal.[40]

The method developed in this paper is based on a technique developed for performing PCA on large databases,[36] where it is frequently impossible to load an entire dataset due to memory limitations, but is easy to sequentially load the individual data points (spectra, in the case of MSI). The method is based on the formation of two "summarisation" matrices

$$\mathbf{L} = \sum_{i=1}^{N} \mathbf{x}^{(i)} \tag{1}$$

$$\mathbf{Q} = \sum_{i=1}^{N} \mathbf{x}^{(i)} (\mathbf{x}^{(i)})^T \tag{2}$$

where the column vector $\mathbf{x}^{(i)}$ is the spectrum at the i^{th} pixel and the sums are over the N points (pixels) in the dataset. From the summarisation matrices, the covariance matrix can be computed as

$$\Sigma = \frac{1}{N}\mathbf{Q} + \frac{1}{N^2}\mathbf{L}\mathbf{L}^T. \tag{3}$$

The important feature of this formulation is that the summarisation matrices can be formed incrementally and require only one data point to be in memory. This is especially advantageous for datasets containing large numbers of pixels. Note that the full covariance matrix must be constructed and very high-dimensional datasets may still prove to be intractable. For this reason, we employ peak detection methods in order to reduce the dimensionality of the data. The full algorithm is presented in Figure 1 and a MATLAB implementation is provided in the Supporting Information.

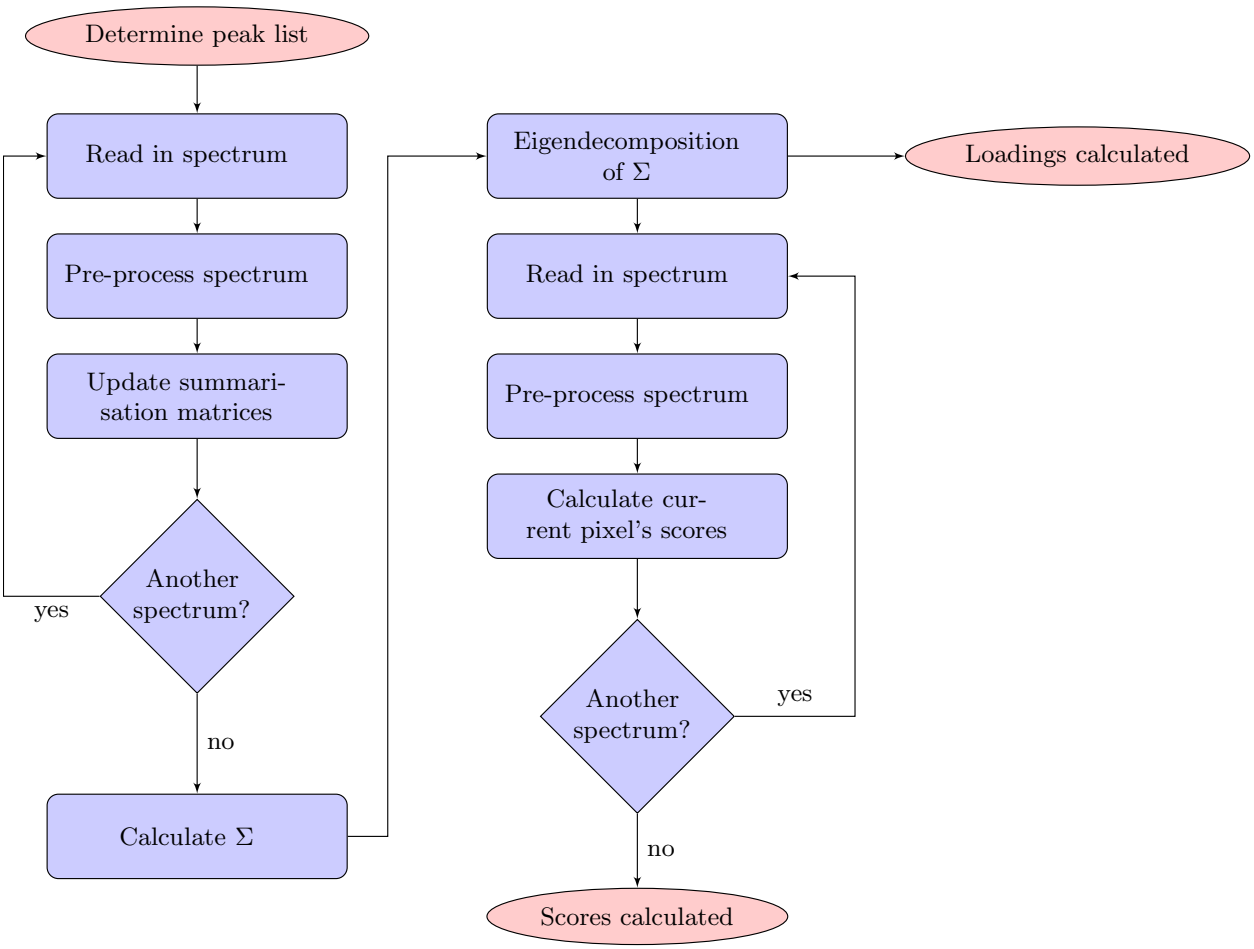

Figure 1: Workflow for memory efficient principal component analysis, only requiring a single spectrum plus the summarisation matrices in RAM at any single point in time.

Description of the Algorithm

With reference to Figure 1, the algorithm can be decomposed into the following steps:

Determine peak list: For combining and comparing spectra, each spectrum must be on the same axis. This is often achieved through binning in the *m/z* domain. In the case of TOF MS, data are acquired at fixed time intervals which are then converted into the *m/z* domain, so here we utilise constant time intervals resulting in *m/z* intervals which follow a square root function. To determine the number of time intervals (or bins) each *m/z* was converted into the time domain (by calculating the square root of each *m/z* value due to the square relation between time of flight and *m/z*[41]) and the difference between adjacent time points was calculated. The minimum time difference was then taken as the time interval at which the detector acquired data. This was then used to reconstruct a common axis for all spectra acquired during the same imaging experiment. For data acquired using the QSTAR Elite (AB Sciex) over the *m/z* range 50-1000, each spectrum had 130861 time bins. The same process was performed for calculating the time interval of data acquired on the QSTAR XL. These bins contain a large amount of redundant data (such as zeros and multiple bins representing a single peak), so to remove this redundancy it was beneficial to determine a peak list of all possible *m/z* centroids, which may either correspond to a molecule or to noise, contained in the data. The peak list was determined from the 'basepeak spectrum' (the maximum intensity at each *m/z* bin within the entire dataset),[23] by first smoothing using a Savitzky-Golay filter (with a window size of 25 time bins) and then performing peak detection using a second derivative gradient method (described in detail in the Supporting Information).

Read in spectrum: Spectra, consisting of (m/z, intensity) pairs, were accessed sequentially from the binary portion of the imzML format and loaded into RAM.

Pre-process spectrum: After loading, spectra were zero-filled, so that the axis was linear in the time domain, and then smoothed using the same Savitzky-Golay filter previously applied to the basepeak spectrum. The intensity values at each of the m/z locations in the peak list were extracted.

Update summarisation matrices: The summarisation matrices are updated with the current spectrum $\mathbf{x}^{(i)}$

$$\mathbf{L} \mapsto \mathbf{L} + \mathbf{x}^{(i)} \tag{4}$$

$$\mathbf{Q} \mapsto \mathbf{Q} + \mathbf{x}^{(i)}(\mathbf{x}^{(i)})^T \tag{5}$$

$\mathbf{L}$ and $\mathbf{Q}$ have been initialised to zero.

Calculate Σ: Once the summarisation matrices have been updated with all spectra, the covariance matrix can be computed as

$$\Sigma = \frac{1}{N}\mathbf{Q} + \frac{1}{N^2}\mathbf{L}\mathbf{L}^T \tag{6}$$

Eigendecomposition of Σ: Eigendecomposition was performed by first reducing the covariance matrix to a tridiagonal matrix followed by QR decomposition of the tridiagonal matrix to calculate the eigenvalues and eigenvectors.

$$\mathbf{U}\mathbf{S}\mathbf{U}^T = \Sigma \tag{7}$$

where $\mathbf{U}$ contains the eigenvectors (also referred to as the loadings or coefficients in PCA) and the diagonal of $\mathbf{S}$ contains the eigenvalues.

Calculate Scores: The scores of the data points against the principal components are computed point-by-point (only one spectrum is required in memory). The score of the spectrum from pixel i against principal component j is computed as

$$s_{ij} = (\mathbf{x}^{(i)} - \frac{\mathbf{L}}{N})^T \mathbf{U}_j \tag{8}$$

where $\mathbf{U}_j$ is the j^{th} column of $\mathbf{U}$ (the j^{th} principal component) and $\mathbf{L}/N$ is subtracted in order to mean-centre the data. Score images can then be generated for a principal component of choice by arranging the score values in the same two dimensional grid as the spectra were collected.

In certain cases, it may be necessary to construct the correlation matrix instead of the covariance matrix. This is necessary in cases where the variables are on different scales and so is not generally applicable for MSI data, but may be useful for liquid chromatography ion mobility spectrometry-mass spectrometry (LC-IMS-MS) data where elution time, drift time and m/z are on different

axes.[42] The correlation matrix ρ can be formed as

$$\rho = \Sigma \circ \frac{1}{\sigma\sigma^T} \tag{9}$$

where $\circ$ denotes the element-wise (Hadamard) product of the matrices and $\sigma = \sqrt{\text{diag}\,(\Sigma)}$ is a vector containing the standard deviation of each dimension (peak). Overwriting the covariance matrix with the correlation matrix will ensure that no extra memory is required and then the subsequent steps of eigendecomposition of ρ ($\mathbf{USU}^T = \rho$) and scoring can be followed as described previously, with Equation 8 replaced by

$$s_{ij} = \left[\left(\mathbf{x}^{(i)} - \frac{\mathbf{L}}{N}\right) \circ \frac{1}{\sigma}\right]^T \mathbf{U}_j \tag{10}$$

Numerical Optimisations

We have noted that whilst this method does not require the whole dataset to be loaded simultaneously, the full covariance matrix ($M \times M$, where M is the number of peaks) must be formed. This limits the dimensionality (number of peaks) that can be processed. However, there are several numerical optimisations that can be made in order to increase this limit. We first observe that in the computation of $\mathbf{Q}$, the product $\mathbf{x}^{(i)}(\mathbf{x}^{(i)})^T$ is symmetric and hence $\mathbf{Q}$ is also symmetric and one need only compute and store the upper triangular part of $\mathbf{Q}$. The formation of both $\mathbf{x}^{(i)}(\mathbf{x}^{(i)})^T$ and $\mathbf{LL}^T$ can be performed element-by-element, updating the relevant variable ($\mathbf{Q}$ and Σ respectively), removing the need to allocate any extra memory for storing temporary variables the same size as the covariance matrix. Furthermore, the covariance matrix can be computed "in-place" of $\mathbf{Q}$ to further reduce the memory requirements. With the covariance matrix stored in packed triangular form, the necessary eigendecomposition can be performed first by tridiagonalisation, and then decomposing with optimised numerical methods from LAPACK designed for this case.[43] It is also possible to discard latter columns of $\mathbf{U}$ once the number of principal components (P) to retain has been determined, reducing $\mathbf{U}$ from $M \times M$ to $M \times P$.

Determination of maximum data size

The maximum size of data that could be processed with *princomp* was determined by selecting a value for the number of peaks, M, and then iteratively altering the number of pixels, N, (using binary search) to produce a random data matrix of size $N \times M$ and then performing *princomp* on the matrix. If this resulted in an 'Out of Memory' error then the number of pixels was reduced, otherwise the number of pixels was increased. When the maximum number of pixels was determined for a specific number of peaks, the number of peaks was altered and the process was repeated.

Image registration and 3D visualisation

Image registration was performed on ion images of *m/z* 826 using StackReg as part of the Fiji package (http://fiji.sc/) which was modified to enable exportation of the calculated affine transform. The affine transform was imported into MATLAB and applied to all image stacks prior to visualisation. 3D data were visualised using vol3d v2 written by Oliver Woodford (http://www.mathworks.com/matlabcentral/file
vol3d-v2).

Results and Discussion

Data size limits for PCA performed using *princomp* on a computer with 8GB RAM are shown in Figure 2. Assuming a large, high resolution image (of 32M pixels[44]) is to be processed using *princomp*, only 8 peaks could be retained. With such small numbers of peaks, it is feasible to examine all ion images manually and there is no longer any requirement to reduce the dimensionality prior to further analysis. Furthermore, and more importantly, reducing a complex dataset to a small number of peaks will likely discard a significant amount of informative data. For data sizes which exceed the memory limit, the data must be reduced prior to PCA by discarding some information, introducing a compromise between the number of pixels and the number of peaks to retain, and is often remedied through the use of data reduction techniques that discard peaks, pixels or both.[23,26] Discarding pixels generally requires some form of prior knowledge about the dataset, specifically

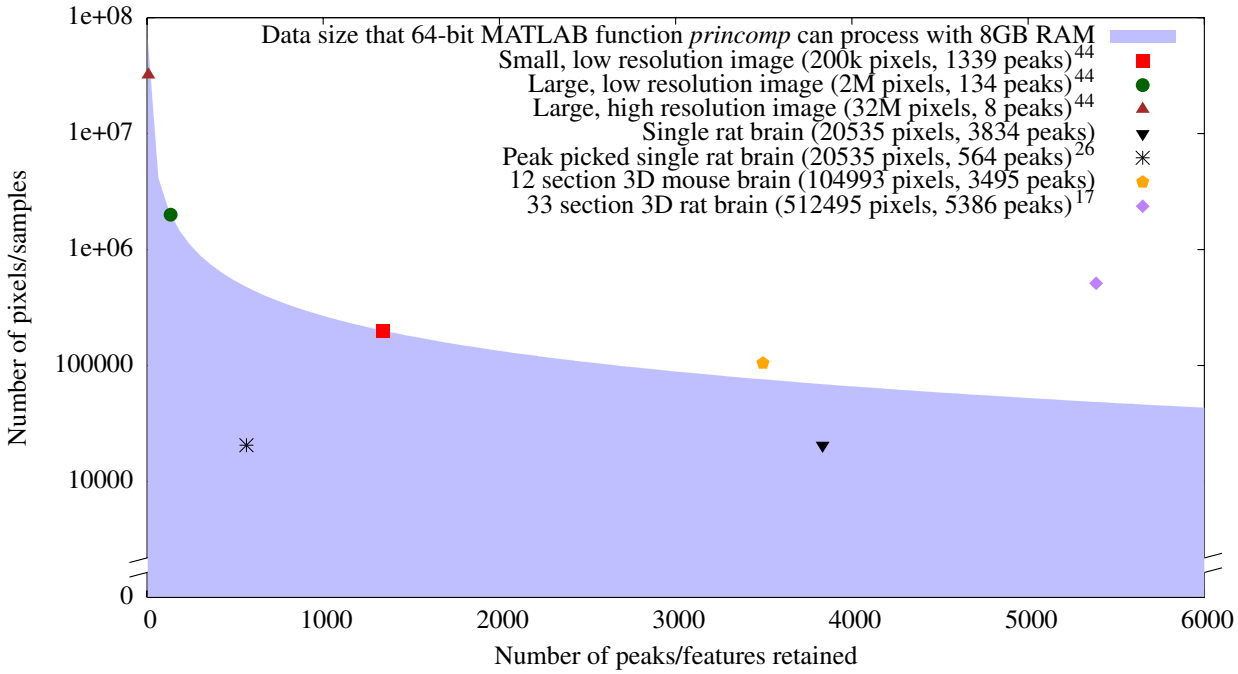

Figure 2: Data sizes that *princomp* (MATLAB Statistics Toolbox version 7.12.0.635) can process using 8GB RAM shown as the area under the curve, demonstrating the compromise between either number of pixels (or samples) or peaks retained in the mass spectrometry data reduction step. All combinations of number of peaks and pixels shown can be processed with the new workflow.

which pixels are relevant. This has the opportunity to introduce analyst bias if the pixel selection method requires user input to aid the generation of a mask to separate matrix related pixels and sample related pixels. Discarding peaks is a much more widely accepted form of data reduction, but dependent on the method used it can introduce the same disadvantages as discarding pixels. However, it should be noted that if uninteresting or uninformative peaks and pixels can be discarded in an unbiased and correct way then the results of PCA will provide much more insightful information on the nature of the variations in the data and potentially reveal variation that was previously masked by the uninformative differences.

An alternative method to reduce the amount of memory required by the PCA algorithm is to use the NIPALS implementation of PCA.[45] This algorithm iteratively calculates the principal components in order of largest variance explained and so only the required principal components can be calculated, reducing computation time and memory requirements. However, the requirement to store the data matrix as well as the residual matrix (which is the same size as the data matrix) in memory means that for large datasets this method will still be prohibitive and the proposed method will still out perform NIPALS in terms of memory savings.

In some cases of MSI, for example small image areas acquired using a high resolution mass analyser such as FTICR, the number of pixels N can be smaller than the number of detected peaks M, and it is then possible to perform PCA on the covariance matrix computed from the transpose of the data matrix $\mathbf{X}^T$, giving a covariance matrix of size $N \times N$ instead of $M \times M$. The eigenvectors of the original covariance matrix can then be calculated by multiplying the principal component vectors by $\mathbf{X}^T$. The method presented here can be applied to this calculation in order that the whole data set need not be loaded into memory, however, this will require a third pass through the data to do the final multiplication. Such requirements are uncommon in MALDI MSI as typically the number of pixels greatly exceeds the number of detected peaks, although this case may be more useful in MS studies where the number of samples is less than that of the detected peaks.

To verify that the new workflow produced the same results as simply using *princomp* a dataset with a sufficiently small enough number of pixels and detected peaks was chosen, and the coeffi-

cients and scores produced were compared. The difference between the two resulting coefficient matrices is on the order of 10^{-6} which is a difference on the order of $10^{-4}\%$. The matrix $\mathbf{Q}$ (Equation 2) summarises $\mathbf{X}\mathbf{X}^T$, where $\mathbf{X}$ is the data matrix[36] and any differences between these two matrices propagate through the algorithm. Such small differences do not significantly affect any of the observed results; the correlation between any principal component's coefficients calculated by the two methods is 1, therefore they describe identical distributions. A visual comparison of score images produced using both methods is shown in Figure 3 with the memory requirements at each step of the process shown in Table 1. Although the memory sizes included in Table 1 focus on the most common case where the number of pixels is larger than the number of peaks, memory savings will still be achieved in the inverse case but will be less significant. For a system with 8GB RAM, a dataset with 100000 pixels and assuming the first 50 principal components will be calculated, the maximum number of peaks that can be included in the proposed workflow is greater than 26000. This is in contrast to the maximum of 2666 peaks included when using *princomp* shown in Figure 2.

Verification on large data sets was performed by simulating a full 3D MALDI MSI experiment by replicating the single section data used previously. Binning the data at 0.2 *m/z*, the standard bin width used in BIOMAP (Novartis), resulted in 4751 bins (31 kB per spectrum) and a data size of 744 MB (for 20535 pixels). Only two full slices of this size could be retained and processed using *princomp* and 8GB of RAM. Assuming 12 μm sagittal sections are taken of a rat brain of size 2cm x 1cm, the distance between the two full sections would be 325 μm, over which distance the internal structure can change significantly. Using Fonville et al.[26]'s method of selecting informative peaks reduced the number of bins to 564 (4 kB per spectrum and 88 MB per slice) which would allow 23 sections to be processed, with 30 μm between each section. Clearly acquiring data with higher lateral, axial or mass resolution would increase the data size, regardless of applying peak picking or not, and therefore decrease the number of sections that it is possible to process while also increasing the distance between sections. However use of the proposed method would enable the entire brain to be processed at any lateral or axial resolution, either with or without discarding

Table 1: Memory size requirements and the corresponding intermediate variable sizes at each step of the principal component analysis algorithm using N (number of pixels) = 100000, M (number of peaks) = 3000 and assuming P (principal components to calculate) = 50 (assuming 99% variance is explained in the first 50 principal components, however commonly fewer principal components are required in practice). Steps 1-4) for *princomp* correspond to steps i-iv) as described in the introduction and are summarised in the lower table. Steps 1-4) for the proposed method refer to the named algorithm steps described in the methods section.

Step	*princomp*	Memory (MB)	Proposed Method	Memory (MB)
1)	Data matrix ($N \times M$)	2288.82	$\mathbf{x}^{(i)}$ ($M \times 1$)	0.02
			$\mathbf{L}$ ($M \times 1$)	0.02
			$\mathbf{Q}$ ($[M(M+1)/2] \times 1$)	34.34
2)	Data matrix ($N \times M$)	2288.82	$\mathbf{L}$ ($M \times 1$)	0.02
	Mean centred data ($N \times M$)	2288.82	Σ ($[M(M+1)/2] \times 1$)	34.34
3)	Data matrix ($N \times M$)	2288.82	$\mathbf{L}$ ($M \times 1$)	0.02
	Mean centred data ($N \times M$)	2288.82	Σ ($[M(M+1)/2] \times 1$)	34.34
	$\mathbf{U}$ ($N \times M$)	2288.82	$\mathbf{U}$ ($M \times M$)	68.66
	$\mathbf{S}$ ($M \times M$)	68.66	$\mathbf{S}$ ($M \times 1$)	0.02
	$\mathbf{V}$ ($M \times M$)	68.66	Working ($[4M-4] \times 1$)	0.09
4)	Data matrix ($N \times M$)	2288.82	$\mathbf{L}$ ($M \times 1$)	0.02
	Mean centred data ($N \times M$)	2288.82	Σ ($[M(M+1)/2] \times 1$)	34.34
	$\mathbf{U}$ ($N \times M$)	2288.82	$\mathbf{U}$ ($M \times M$)	68.66
	$\mathbf{S}$ ($M \times 1$)	0.02	$\mathbf{S}$ ($M \times 1$)	0.02
	$\mathbf{V}$ ($M \times M$)	68.66	Scores ($N \times P$)	38.15
	Scores ($N \times M$)	2288.82		
	Max. RAM Usage (MB)	9223.96	Max. RAM Usage (MB)	141.18

Step	*princomp*	Proposed Method
1)	Read dataset into RAM Pre-process and reduce if necessary	'Read in spectrum' 'Pre-process spectrum' 'Update summarisation matrices'
2)	Mean center data matrix	'Calculate Σ'
3)	SVD of data matrix	'Eigendecomposition of Σ'
4)	Data matrix projected using eigenvectors	'Calculate Scores'

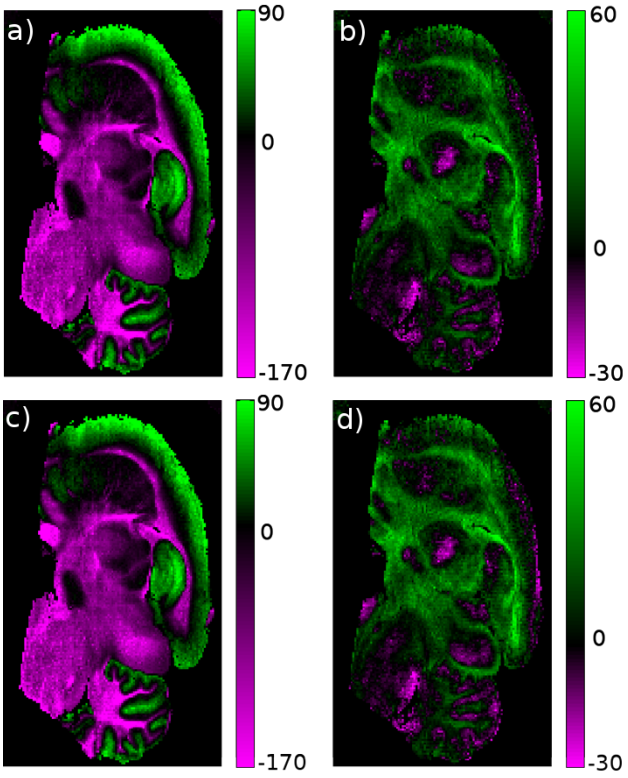

Figure 3: Comparison of principal component score images of a MALDI MS image of a single rat brain section using *princomp* (a, b) and the memory efficient PCA method (c, d). a) and c) show principal component 5 (demonstrating a significant amount of variance between grey and white matter regions) and b) and d) show principal component 19 (demonstrating that information is still contained in high principal components). The total difference between the coefficient matrices (which are used to calculate score images) produced by each method is on the order of 10^{-6} which is a difference on the order of $10^{-4}\%$, resulting in a small, but insignificant (and visibly indistinguishable), difference in the score images.

peaks detected on either binned or raw data acquired using a QSTAR XL or QSTAR Elite over the mass range of *m/z* 50-1000, shown in Figure S1.

Following verification, the new methodology was applied to a dataset containing multiple serial sections taken from a single mouse brain which was too large to be handled with *princomp*. The new methodology was used to reduce the 3D dataset without the requirement of discarding peaks or pixels. Clustering was then performed, using the *k*-means algorithm (k=7), on the reduced dataset, revealing 3 clusters on the tissue region and 4 clusters describing the matrix region. All three clusters in the tissue region are visualised in Figure 6, highlighting white matter in yellow (including the corpus callosum and arbor vitae), grey matter (including the cerebral cortex) in blue

and a tissue edge region in red.

Molecules which correlate with each cluster were determined by calculating the Pearson product-moment correlation coefficient between every image produced at each *m/z* bin in the raw data and a binary image of the cluster of interest.[17] The distribution of the highest correlating molecules with each cluster, tentatively identified as PC 36:1[46] for the white matter (yellow) cluster, PC 32:0[46] for the grey matter (blue) cluster and haem[8] for the tissue edge (red) cluster, in both 2D and 3D are shown in Figure 5.

The selection of only informative m/z bins[26] would have reduced the data set sufficiently to be analysed with *princomp*. Clustering with *k*-means (*k*=3) on the reduced dataset produced the same clusters as shown in Figure 6, however a review of the peak lists showed that *m/z* 616 was discarded by this reduction method due to the noisy background correlating with the matrix region. This peak was actually found to contribute significantly to the red cluster in Figure 6. Examination of ion images of this peak showed that it was found predominantly in a region in which commonly detected endogenous species were observed to be of unusually low ion counts. This regional suppression is likely to have been caused by contamination with blood, during either the organ excision, or sectioning/mounting procedures. This observation would have been difficult to make without an unsupervised tool such as PCA (specifically one which can handle retention of the entire peak list), requiring manual inspection of every possible ion image and comparing to the reduced dataset while considering the anatomy of the sample.

The benefits of applying clustering algorithms to a large (50 GB) 3D kidney dataset have been described recently.[17] The described data contained a very large (512495) number of pixels and so peak picking was performed prior to clustering to reduce the data sufficiently, which involved discarding peaks if they appeared in less than 1% of the spectra. An alternative method of data reduction would be to use the method presented here, which can cope with arbitrarily many pixels, to reduce the entire dataset of 7677 *m/z* bins to a small number of principal components that explain at least 99% of the variance. This would reduce the chance of discarding informative peaks that are only present in a small, localised feature that is smaller than 1% of the image size (5124 pixels).

As such, this work provides new ways to evaluate the effects, suitability and robustness of peak picking on larger datasets.

Spatial binning[25] may be a useful tool for the reduction in memory requirement (as well as enhancing imaging signal-to-noise and increasing contrast) however this comes at a cost of spatial detail. For cases where spatial binning is performed solely for the benefit of memory reduction the method proposed here will prove valuable. A recent article stated that the choice to only retain 650 *m/z* bins was "a pragmatic desire to use manageable covariance matrices",[47] however the method presented here has demonstrated that covariance matrices far exceeding 650 *m/z* bins can be handled with ease, and use of the proposed method would enable handling of the entire 11280 processed *m/z* bins. Binning in the *m/z* domain combined with a sparse data matrix representation in order to compute PCA on a 3D ToFSIMS dataset was recently described.[9] In order to preserve the sparsity, and therefore the memory reduction achieved, mean centering and scaling were omitted, however the work presented here does not require the entire dataset to be stored in RAM and so mean centering and scaling can be applied if required.

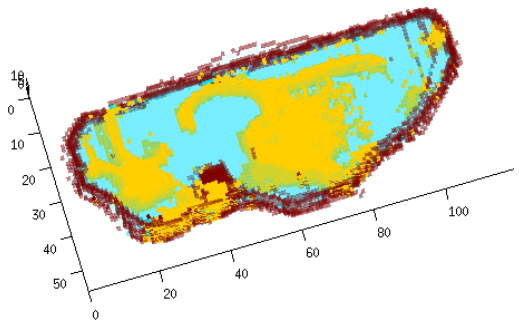

Figure 4: 3D representation of the three on tissue clusters determined with *k*-means (*k*=7) applied to a MALDI mass spectrometry image of 12 serial sections of mouse brain after being reduced by PCA (with 99.14% of the variance retained in 40 principal components).

The iterative accessing and processing of the data increases the computation time required to process each dataset when compared with methods which retain the entire dataset in RAM, like *princomp*. However, if the dataset has already been pre-processed and reduced to peak lists, either

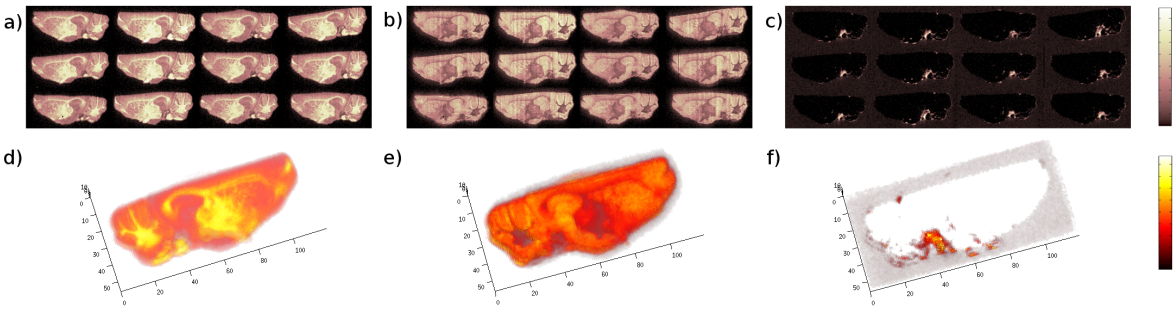

Figure 5: (a-c) Selected ion images which correlate highly with each on tissue cluster distribution determined from *k*-means of serial sections, shown in 2D representation. (d-f) 3D representations of ion images where the alpha channel (transparency) is proportional to the intensity. (a,d) *m/z* 826 (PC 36:1 $[M+K]^+$). (b,e) *m/z* 734 (PC 32:0 $[M+H]^+$). (c,f) *m/z* 616 (haem $[M+H]^+$).

through using automatic data reduction methods described by McDonnell et al.[23] and Fonville et al.[26] or by manually selecting known peaks,[48] then the amount of processing required is reduced by eliminating the 'Pre-process spectrum' steps from the workflow shown in Figure 1. For small MSI datasets, where the number of pixels and detected/retained peaks falls below the curve in Figure 2, standard implementations of PCA will outperform (in terms of speed) the approach described here. As a comparison, the time required to load, pre-process and perform *princomp* on the single rat brain image (shown in Figure 3) was 5.7 minutes whereas the time required to perform the new methodology was 13.2 minutes. The reason that the time is doubled is due to the requirement to read in and process the data a second time to calculate the scores. Eliminating the pre-processing step reduced the time to 2.7 minutes and 8.2 minutes for *princomp* and the proposed method respectively. Processing time for the 3D dataset using the proposed method was 24 minutes. Even if the processing time is lengthened, as the size of the data increases (in both the number of pixels and in mass resolution) the need for data processing routines that can handle increased data sizes becomes more apparent. Furthermore, general purpose programming on graphical processing units (GPGPU) could be employed to reduce the time taken as the generation of the summarisation matrices is highly parallelisable.[44]

This method inherently provides the ability to compute the covariance and correlation matrices in a more memory efficient way, enabling rapid determination of co-localised *m/z* peaks on larger

Table 2: Comparison of the time required to load the MSI data and perform *princomp* and the proposed method including and excluding the time taken to pre-process (spectral smoothing and peak detection) the data. The 2D rat brain image contained 20535 pixels and 564 detected peaks. The 3D mouse brain image contained 104993 pixels and 3495 detected peaks.

Dataset	*princomp*	*princomp* (no pre-processing)	Proposed Method	Proposed Method (no pre-processing)
2D Rat brain	5.7 mins	2.7 mins	13.2 mins	8.2 mins
3D Mouse brain	X	X	24.0 mins	12.0 mins

data sets than previously reported.[49] This type of investigation requires prior knowledge of an *m/z* of interest and so would be beneficial in pharmaceutical studies where the *m/z* of the drug is known and molecules, such as metabolites, which co-localise with the drug are of specific interest.

Despite this article being focused solely on MSI data, the method presented here can be applied to any analytical technique that has sufficiently large data such that memory limitations have become problematic.

Conclusion

Data processing is an essential and extremely challenging aspect of mass spectrometry imaging research. The highly multivariate nature of the technique poses challenges in both the limits of data size and the ease of information extraction from imaging experiments. We have presented a means of handling the complete data without discarding potentially useful information. These methods will become increasingly important as efforts towards complete and unsupervised review of large 3D image datasets continues. The memory efficient workflow described here provides, for the first time, a means of performing PCA on extremely large MS image data. The methods also allow for a comparison of data reduction techniques and a means of comparing discarded (or retained) peaks with a complete list. This will facilitate ongoing efforts in the development and evaluation of data reduction tools and computational techniques for the improved processing and interpretation of MSI data.

References

(1) Stoeckli, M.; Staab, D.; Schweitzer, A. *International Journal of Mass Spectrometry* **2007**, *260*, 195–202.

(2) Cornett, D.; Frappier, S.; Caprioli, R. *Analytical Chemistry* **2008**, *80*, 5648–5653.

(3) Carter, C.; McLeod, C.; Bunch, J. *Journal of the American Society for Mass Spectrometry* **2011**, *22*, 1991–1998.

(4) Palmer, A. D.; Griffiths, R.; Styles, I.; Claridge, E.; Calcagni, A.; Bunch, J. *Journal of Mass Spectrometry* **2012**, *47*, 237–241.

(5) Taban, I.; Altelaar, A.; van der Burgt, Y.; McDonnell, L.; Heeren, R.; Fuchser, J.; Baykut, G. *Journal of the American Society for Mass Spectrometry* **2007**, *18*, 145–151.

(6) Khatib-Shahidi, S.; Andersson, M.; Herman, J.; Gillespie, T.; Caprioli, R. *Analytical Chemistry* **2006**, *78*, 6448–6456.

(7) Römpp, A.; Guenther, S.; Takats, Z.; Spengler, B. *Analytical and Bioanalytical Chemistry* **2011**, *401*, 65–73.

(8) Stoeckli, M.; Staab, D.; Schweitzer, A.; Gardiner, J.; Seebach, D. *Journal of the American Society for Mass Spectrometry* **2007**, *18*, 1921–1924.

(9) Fletcher, J.; Rabbani, S.; Henderson, A.; Lockyer, N.; Vickerman, J. *Rapid Communications in Mass Spectrometry* **2011**, *25*, 925–932.

(10) Ghosal, S.; Fallon, S.; Leighton, T.; Wheeler, K.; Kristo, M.; Hutcheon, I.; Weber, P. *Analytical Chemistry* **2008**, *80*, 5986–5992.

(11) Breitenstein, D.; Rommel, C.; Möllers, R.; Wegener, J.; Hagenhoff, B. *Angewandte Chemie International Edition* **2007**, *46*, 5332–5335.

(12) Fletcher, J.; Lockyer, N.; Vickerman, J. *Mass Spectrometry Reviews* **2011**, *30*, 142–174.

(13) Sinha, T.; Khatib-Shahidi, S.; Yankeelov, T.; Mapara, K.; Ehtesham, M.; Cornett, D.; Dawant, B.; Caprioli, R.; Gore, J. *Nature Methods* **2007**, *5*, 57–59.

(14) Xiong, X.; Xu, W.; Eberlin, L.; Wiseman, J.; Fang, X.; Jiang, Y.; Huang, Z.; Zhang, Y.; Cooks, R.; Ouyang, Z. *Journal of The American Society for Mass Spectrometry* **2012**, *23*, 1147–1156.

(15) Crecelius, A.; Cornett, D.; Caprioli, R.; Williams, B.; Dawant, B.; Bodenheimer, B. *Journal of the American Society for Mass Spectrometry* **2005**, *16*, 1093–1099.

(16) Seeley, E.; Caprioli, R. *Analytical Chemistry* **2012**, *84*, 2105–2110.

(17) Trede, D.; Schiffler, S.; Becker, M.; Wirtz, S.; Steinhorst, K.; Strehlow, J.; Aichler, M.; Kobarg, J. H.; Oetjen, J.; Dyatlov, A.; Heldmann, S.; Walch, A.; Thiele, H.; Maass, P.; Alexandrov, T. *Analytical Chemistry* **2012**, *84*, 6079–6087.

(18) Goodwin, R.; Pitt, A.; Harrison, D.; Weidt, S.; Langridge-Smith, P.; Barrett, M.; Logan Mackay, C. *Rapid Communications in Mass Spectrometry* **2011**, *25*, 969–972.

(19) Smith, D.; Aizikov, K.; Duursma, M.; Giskes, F.; Spaanderman, D.; McDonnell, L.; O'Connor, P.; Heeren, R. *Journal of the American Society for Mass Spectrometry* **2011**, *22*, 130–137.

(20) Reichenbach, S.; Henderson, A.; Lindquist, R.; Tao, Q. *Rapid Communications in Mass Spectrometry* **2009**, *23*, 1229–1233.

(21) Reichenbach, S.; Tian, X.; Lindquist, R.; Tao, Q.; Henderson, A.; Vickerman, J. Visualization and analysis of large three-dimensional hyperspectral images. *SPIE Defense, Security, and Sensing*, 2009; pp 734108–734108.

(22) Yang, C.; He, Z.; Yu, W. *BMC Bioinformatics* **2009**, *10*, 4.

(23) McDonnell, L.; Van Remoortere, A.; De Velde, N.; Van Zeijl, R.; Deelder, A. *Journal of the American Society for Mass Spectrometry* **2010**, *21*, 1969–1978.

(24) Deininger, S.; Ebert, M.; Fütterer, A.; Gerhard, M.; Röcken, C. *Journal of Proteome Research* **2008**, *7*, 5230–5236.

(25) Klerk, L.; Broersen, A.; Fletcher, I.; van Liere, R.; Heeren, R. *International Journal of Mass Spectrometry* **2007**, *260*, 222–236.

(26) Fonville, J. M.; Carter, C.; Cloarec, O.; Nicholson, J. K.; Lindon, J. C.; Bunch, J.; Holmes, E. *Analytical Chemistry* **2012**, *84*, 1310–1319.

(27) Jones, E.; Deininger, S.; Hogendoorn, P.; Deelder, A.; McDonnell, L. *Journal of Proteomics* **2012**, *75*, 4962–4989.

(28) Alexandrov, T.; Kobarg, J. *Bioinformatics* **2011**, *27*, i230–i238.

(29) McCombie, G.; Staab, D.; Stoeckli, M.; Knochenmuss, R. *Analytical Chemistry* **2005**, *77*, 6118–6124.

(30) Biesinger, M.; Paepegaey, P.; McIntyre, N.; Harbottle, R.; Petersen, N. *Analytical Chemistry* **2002**, *74*, 5711–5716.

(31) Trim, P.; Atkinson, S.; Princivalle, A.; Marshall, P.; West, A.; Clench, M. *Rapid Communications in Mass Spectrometry* **2008**, *22*, 1503–1509.

(32) Broersen, A.; Van Liere, R. Transfer functions for imaging spectroscopy data using principal component analysis. *Proc. Eurographics/IEEE VGTC Symposium on Visualization*, 2005.

(33) Altelaar, A.; Luxembourg, S.; McDonnell, L.; Piersma, S.; Heeren, R. *Nature Protocols* **2007**, *2*, 1185–1196.

(34) Sjövall, P.; Lausmaa, J.; Johansson, B. *Analytical Chemistry* **2004**, *76*, 4271–4278.

(35) Van de Plas, R.; Ojeda, F.; Dewil, M.; Van Den Bosch, L.; De Moor, B.; Waelkens, E. Prospective exploration of biochemical tissue composition via imaging mass spectrometry

guided by principal component analysis. *Proceedings of the Pacific Symposium on Biocomputing*, 2007; pp 3–7.

(36) Ordonez, C. *IEEE Transactions on Knowledge and Data Engineering* **2010**, *22*, 1752–1765.

(37) Römpp, A.; Schramm, T.; Hester, A.; Klinkert, I.; Both, J.; Heeren, R.; Stöckli, M.; Spengler, B. *Methods Mol. Biol* **2011**, *696*, 205–224.

(38) Steven, R. T.; Race, A. M.; Bunch, J. *Journal of the American Society for Mass Spectrometry* DOI: 10.1007/s13361-013-0586-0.

(39) Race, A. M.; Styles, I. B.; Bunch, J. *Journal of Proteomics* **2012**, *75*, 5111–5112.

(40) Alexandrov, T.; Becker, M.; Guntinas-Lichius, O.; Ernst, G.; von Eggeling, F. *Journal of Cancer Research and Clinical Oncology* **2012**, 1–11.

(41) Hoffmann, E. *Mass spectrometry*; Wiley Online Library, 1996.

(42) Baker, E. S.; Livesay, E. A.; Orton, D. J.; Moore, R. J.; Danielson, W. F.; Prior, D. C.; Ibrahim, Y. M.; LaMarche, B. L.; Mayampurath, A. M.; Schepmoes, A. A.; Hopkins, D. F.; Tang, K.; Smith, R. D.; Belov, M. E. *Journal of Proteome Research* **2010**, *9*, 997–1006.

(43) Anderson, E.; Bai, Z.; Bischof, C.; Blackford, S.; Demmel, J.; Dongarra, J.; Du Croz, J.; Greenbaum, A.; Hammarling, S.; McKenney, A.; Sorensen, D. *LAPACK Users' Guide*, 3rd ed.; Society for Industrial and Applied Mathematics: Philadelphia, PA, 1999.

(44) Jones, E.; van Zeijl, R.; Andrén, P.; Deelder, A.; Wolters, L.; McDonnell, L. *Journal of The American Society for Mass Spectrometry* **2012**, *23*, 745–752.

(45) Geladi, P.; Kowalski, B. R. *Analytica Chimica Acta* **1986**, *185*, 1–17.

(46) Griffiths, R.; Bunch, J. *Rapid Communications in Mass Spectrometry* **2012**, *26*, 1557–1566.

(47) Stone, G.; Clifford, D.; Gustafsson, J.; McColl, S.; Hoffmann, P. *BMC Research Notes* **2012**, *5*, 419.

(48) Wagner, M.; Castner, D. *Langmuir* **2001**, *17*, 4649–4660.

(49) McDonnell, L.; van Remoortere, A.; van Zeijl, R.; Deelder, A. *Journal of Proteome Research* **2008**, *7*, 3619–3627.

Acknowledgement

This work was funded through studentships to AMR, RTS and ADP via the EPSRC through the PSIBS Doctoral Training Center, grant EP/F50053X/1.

Supporting Information Available

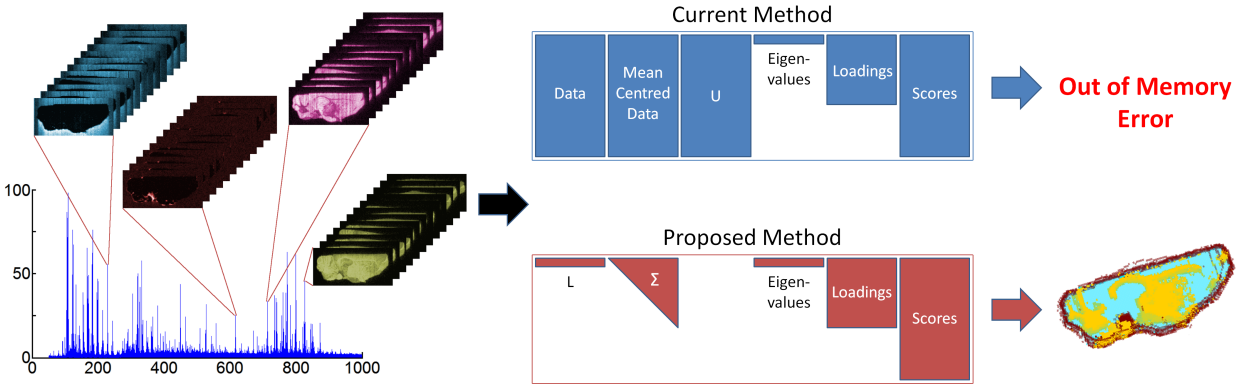

Figure 6: For Table of Contents Only.

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