A Tale of Two Matrix Factorizations

Paul Fogel, Consultant, Paris
Douglas M. Hawkins, Professor, School of Statistics, University of Minnesota
Chris Beecher, Chief Science Officer, NextGen Metabolomics
George Luta, Assistant Professor, Department of Biostatistics, Bioinformatics, and Biomathematics, Georgetown University
S. Stanley Young, Assistant Director of Bioinformatics, National Institute of Statistical Sciences

ABSTRACT

In statistical practice, rectangular tables of numeric data are commonplace, and are often analyzed using dimension reduction methods like the singular value decomposition (SVD) and its close cousin, principal component analysis (PCA). This analysis produces score and loading matrices representing the rows and the columns of the original table and these matrices may be used for both prediction purposes and to gain structural understanding of the data. In some tables, the data entries are necessarily non-negative (apart, perhaps, from some small random noise), and so the matrix factors meant to represent them should arguably also contain only non-negative elements. This thinking, and the desire for parsimony, underlies such techniques as rotating factors in a search for "simple structure." These attempts to transform score or loading matrices of mixed sign into non-negative, parsimonious forms are however indirect and at best imperfect. The recent development of non-negative matrix factorization, or NMF, is an attractive alternative. Rather than attempt to transform a loading or score matrix of mixed signs into one with only non-negative elements, it directly seeks matrix factors containing only non-negative elements. The resulting factorization often leads to substantial improvements in interpretability of the factors. We illustrate this potential by synthetic examples and a real data set. The question of exactly when NMF is effective is not fully resolved, but some indicators of its domain of success are given. It is pointed out that the NMF factors can be used in much the same way as those coming from PCA for such tasks as ordination, clustering and prediction.

KEY WORDS: Principal component analysis, PCA, Singular value decomposition, SVD, Non-negative matrix factorization, NMF, latent dimensions.

1. INTRODUCTION

Rectangular tables of numeric data are wide-spread in statistical practice – for example in psychometrics where n subjects are scored on p items in a test; in microarrays where n tissues are tested with p probes; in the geosciences where p constituents are measured in n strata. Each of these settings gives rise to an $n \times p$ data matrix **X**. Whereas in the past, p was typically small, many emerging areas give rise to data matrices where n and/or p may be in the thousands or tens of thousands, challenging traditional multivariate analysis approaches.

A lower-rank matrix approximation is

$$\mathbf{X} = \mathbf{L}\mathbf{R}^{\mathbf{T}} + \mathbf{E}$$

where the left matrix **L** has n rows and k columns, the right matrix **R** has p rows and k columns, and **E** is a matrix of "errors". Superscript T indicates the transpose of a matrix. Each row of **L** represents one "case"; each row of **R** represents one "variable" and the k columns of both **L** and **R** represent k underlying latent variables or dimensions that relate the rows and the columns of **X.** The hope of this approximation is that a k value much smaller than n and p will nevertheless be enough to give a small **E** and so to capture nearly all the structure in **X**. Then the data matrix **X** itself can be discarded, and interpretation focused on **L** to explicate relationships between the cases, and on **R** to explicate those between the variables.

The most familiar way of getting a lower-rank approximation **(Greenacre and Underhill 1982)** is through the Singular Value Decomposition, or SVD and its close cousin Principal Component Analysis, or PCA. Starting with the exact spectral decomposition of **X**

$$\mathbf{X} = \mathbf{A}\,\boldsymbol{\Lambda}\,\mathbf{B}^{\mathrm{T}}$$

where **A** are the row singular vectors, **B** the column singular vectors, and **Λ** a diagonal matrix of the singular values, defining

$$\mathbf{L} = \mathbf{A}\boldsymbol{\Lambda}^{r}; \quad \mathbf{R} = \mathbf{B}\boldsymbol{\Lambda}^{1-r}$$

for any selected r gives an exact representation $\mathbf{X} = \mathbf{L}\mathbf{R}^{\mathrm{T}}$.

The eigenvalues of principal component analysis are the squares of the singular values in $\mathbf{\Lambda}$ and PCA's eigenvectors are the column singular vectors. Retaining just the first k columns of **L** and **R** of the exact spectral decomposition then gives a lower rank approximation to **X**. This approximation is optimal in a least squares sense– there is no other approximation using k latent variables that more accurately represents **X** as measured by sum of squared deviations**.**

Factor analysis (FA), a somewhat more distant cousin, also relies on the equation $\mathbf{X = LR^T + E}$. However, unlike the setting with the SVD, the **E** matrix of FA is modeled as having independent normal elements, conceptually making FA a quite different methodology from the SVD, despite their superficial similarity.

The representation of the approximation given by the SVD is not unique. If we retain the first k columns of **L** and **R** and let **C** be any non-singular $k \times k$ matrix, replacing **L** by $\mathbf{L}^* = \mathbf{LC}$ and **R** by $\mathbf{R}^* = \mathbf{R}(\mathbf{C}^{\mathrm{T}})^{-1}$ or more compactly $\mathbf{R}^* = \mathbf{RC}^{\mathrm{-T}}$ gives an approximation

$$\mathbf{X} \approx \mathbf{L}^*\mathbf{R}^{*\mathbf{T}} = (\mathbf{LC})\ (\mathbf{RC}^{\mathrm{-T}}) = \mathbf{LR}^{\mathbf{T}}$$

This factorization gives identical approximations to all elements of **X**, but uses left and right factors $\mathbf{L}^*$ and $\mathbf{R}^*$ that may look very different from **L** and **R**. This fact underlies the variety of methods used for example in factor analysis to rotate a hard-to-interpret loading matrix to one whose elements are easier to interpret, as summarized in the mantra of "simple structure."

When performing a PCA or a SVD, it is common to center the columns of **X** by subtracting a column (or sometimes a global) mean from each element of **X**, however this centering is not central to the methods. Frequently, leaving the data uncentered and adding one latent dimension to the fit to accommodate the mean leads to essentially the same structure as is obtained from the centered data, as will be illustrated later in this paper.

In some settings **X** consists of non-negative elements, apart, perhaps, from a few negative elements resulting from measurement error. For example intensities in microarrays, chemical compositions in geology and biological chemistry, and test scores in psychometrics are usually non-negative. When this is the case, it is generally desirable to use matrix approximations

$$\mathbf{X} \approx \mathbf{LR}^{\mathbf{T}}$$

whose **L** and **R** factors are also comprised of non-negative numbers. This will greatly simplify interpretation, since negative values in **L** and **R** are hard to make sense of. In psychometric testing, for example, a negative loading in the R matrix would imply that item was negatively associated with the latent aptitude being tested which, unless the item was designed with a reversed scale, would make no sense. But the SVD approximation does not give non-negative factors **L** and **R** – the orthogonality of the successive singular vectors makes this impossible except in the degenerate case that the singular vectors are some permutation of the identity matrix. This challenge leads to the rotation methods that take a loading matrix with mixed signs and attempt to find a non-singular transformation that will give an equally explanatory matrix consisting of more interpretable numbers – for example with few or no negative elements. But this post-processing of a SVD approximation to something even approximating non-negativity is a daunting task.

Note that if **X** is non-negative, mean-centering will create a matrix of mixed sign, and so mean centering is not performed when one is interested in non-negative factorization. An alternative method that will remove column effects but not destroy the non-negativity is to subtract the column minimum from all elements of each column. However this approach is not widely used.

Focusing on the issue of diagnosing underlying structure, suppose that a matrix **X** that is non-negative, except perhaps for some random noise, is in fact generated by two non-negative *k*-column matrices:

$$\mathbf{X} = \mathbf{L}\mathbf{R}^{T} + \mathbf{E}$$

To uncover the generating mechanism, we would like to recover **L** and **R**, the generating vector pairs. The fact that the SVD is a minimum-variance approximation of any given rank is a mixed blessing, as it can lead to multiple mechanisms being conflated in a single latent variable, as is shown in Outbox 1. Suppose that we are looking at a data matrix of subjects and their gene expression data and we have a diagnosis for each person: diseased or not. Now, we are given one named disease, but there may be multiple etiologies that lead to the same symptoms and hence the one named disease/condition; metabolic syndrome is a good current example. It is quite possible that any of several etiologies might lead to the same clinical picture, for instance, overweight especially in the upper body, insulin resistance, metabolic abnormalities, clinical failure, or other clinical indications. SVD will tend to conflate the different etiologies, putting all of them into the first component with positive coefficients, and then differentiating them in subsequent components by giving one etiology positive coefficients and another negative coefficients. While the true nature of the different etiologies can in principle be recovered by rotating the different components to separate them out, this can be an uncertain and tedious operation.

Non-negative matrix factorization, (**Lee and Seung 1999**), where the elements of the factoring matrices are also non-negative, addresses this problem head-on. Unlike the SVD, NMF is not able to conflate the mechanisms in different components with mixed signs, and will instead tend to identify each of the syndromes with its own component, making interpretation transparent.

Not all data matrices are conceptually generated by the product of non-negative matrices plus random noise. We have mentioned many examples where this is the case, but there are many others where the signs in X are genuinely mixed; when this is the case, NMF would not be a good choice for factorization and could lead to seriously misleading results.

Over the years, there has been accumulating evidence from many different fields that NMF is capable of finding parts - see the NMF review paper of **Devarajan (2008).** All this evidence raises a question of why and when NMF is so much better at finding parts than, say, principal components analysis.

The paper by **Donoho and Stodden (2003)** (D&S) considers the "why" question, producing a set of rules describing “Separable Factorial Articulation Families”, and shows that if these rules are satisfied there is a unique exact non-negative factorization. These are sufficient, but not necessary conditions. Corresponding necessary conditions, however, seem to be lacking. Their “swimmer” example provides a case in which the sufficient conditions are violated, yet NMF can recover the underlying structure, leaving the issue of necessary conditions a topic for further research.

An additional issue is that D&S’s sufficient conditions related to the underlying true generating model, whereas all an analyst typically has is a data matrix, whose underlying generating model is unknown, and for whom the sufficient conditions can not be checked. This uncertainty may be tolerable for the data analyst who can fit the NMF model and decide whether it makes sense in subject-matter terms, but it is an issue one would like to see resolved.

In this paper, we first describe a real Near InfraRed (NIR) spectroscopy data set, which we will return to. Next we review and summarize what researchers have uncovered so far and provide some clean synthetic examples and counter-examples that contrast SVD and NMF. Then we return to an analysis of the NIR dataset using things we have learned about NMF. Finally we make some summary points in a discussion and point out some intriguing open questions about NMF.

2. A REPRESENTATIVE REAL DATA SET

NIR spectroscopy can be used to measure vibrational energy of chemical bonds and is often used to estimate the concentrations of molecular subtypes, e.g. protein, carbohydrate, etc., in complex samples. We consider a designed experiment run by **Ellekjær, Isaksson, and Solheim (1994)**, and used by **Langsrud (2006)**, where the three constituents fat, salt, and starch added in making sausage were varied according to a single-replicate 6x3x3 factorial design. In the 54 sausages analyzed, fat has six levels (8%, 12%, 16%, 20%, 24%, 28%); salt has three levels (1.3%, 1.6%, 1.9%); and starch has three levels (1.5%, 4.5%, 7.5%). The 54 sausages were measured by both sensory analysis and NIR spectroscopy. We look at the spectral data only. NIR spectra were measured in 4 nm increments from wavelengths 1100 nm to 2498 nm, thus there are 351 measurements per sample. Accordingly, the NIR matrix has 54 rows and 351 columns (**Figure 1**). It is conceptually plausible, at least as a starting model, that each spectrum in the data set will be a superposition of pure "archetypal" spectra corresponding to fat, salt and starch, weighted according to the composition of the sausage meat. As spectral energy is necessarily positive, this data set falls within our conceptual framework of a non-negative matrix plausibly generated (apart from random error) as a product of non-negative matrices **L** and **R** that reflect, respectively, the composition of each sausage and the spectrum of each pure archetype.

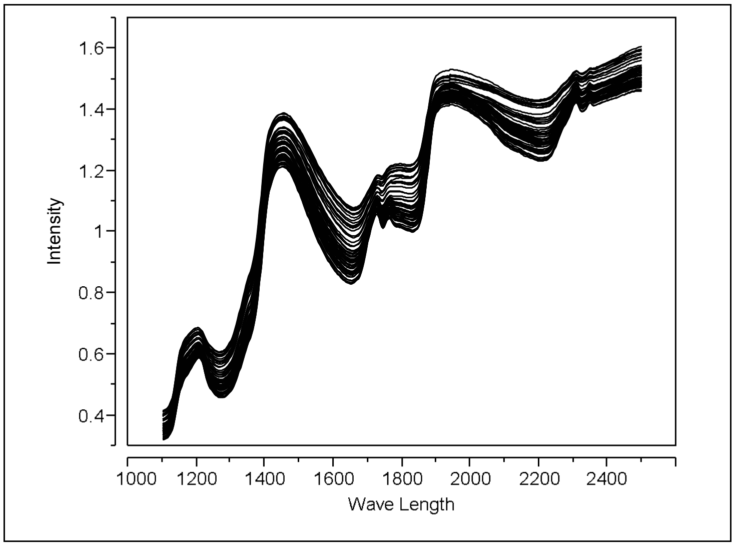


Figure 1: NIR spectra of 54 sausages in 4 nm increments from 1100 nm to 2498 nm.

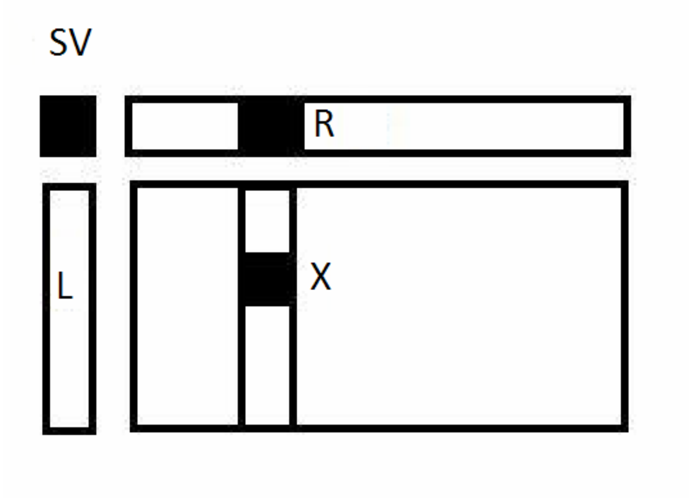
As the samples are mixtures of fat, salt and starch, along with other ingredients, it is reasonable to expect that their spectra would be mixtures of “archetype” spectra of fat, salt and starch and the other ingredients. We would like to extract the archetype curves that correspond to these pure components. We will return to this example once we have briefly covered the computation of a SVD and NMF and applied SVD and NMF to some synthetic datasets.

3. METHODS

3.1 Singular Value Decomposition via Alternating Least Squares (ALS)

Good (1969) points out that the SVD of a matrix is central to the computing of many statistical methods. He gives the alternating least squares algorithm for computing a SVD in a few sentences. See **Outbox 1**. Modifications to ALS by **Gabriel and Zamir (1979)** and by **Liu et al. (2003)** can be used to compute an analog of SVD that is both robust to outliers and accommodates missing information. Common practice in fitting the SVD is to center the columns by subtracting their means. This is not an essential feature of the SVD, and so to better highlight the similarities and differences of the SVD and NMF, we will not center the data in what follows.

The ALS algorithm is a key to understanding both how to obtain a good approximation to a given data matrix and to why, in the case of SVD, that the approximation can lead to confusion. In addition to mixing generating vectors, which restricts interpretability, SVD can pull in noise as if it were a feature (**Faber *et al.,* 1995**). This is in addition to the interpretability problems of having a possibly user-unfriendly basis in *k*-dimension space that can be transformed in uncountably many ways.



0. (Optionally) Mean center each column of X.

1. Fill the Left and Right vectors, L and R, with initial guesses. L and R will become the left and right singular vectors.

2. Fixing on a particular column, fit a no-intercept least squares linear regression of this column on L. If any cell(s) in the column are missing, omit them from the regression.

3. Put the estimated slope of this regression in R.

4. Repeat this for all X columns, filling out the entire R vector.

5. Normalize the R vector, divide by the sum of squares of its elements.

6. Repeat steps 2 to 5, regressing each row of X on the trial row R.

7. Alternate row and column regressions until convergence.

8. Form the outer product of the row and column singular vectors, L and R, and fit a no-intercept regression of all non-missing values in X on this outer product. The slope of this regression estimates the singular value, SV.

9. Replace the matrix X by the residuals of the regression on the outer product.

10. Repeat steps 1-9 to get the next singular triple, and repeat the entire procedure until the desired number of components has been found.outer product.

10. Repeat steps 1-9 to get the next singular triple, and repeat the entire procedure until the desired number of components has been found.

Outbox 1. Alternating Least Squares SVD algorithm.

The mean centering is described as optional. If omitted, the first singular triplet (left and right singular vector and singular value) tends to accommodate the data location and then the more interesting structure emerges from the second singular triplet on, so operationally, centering tends to remove one component which captures the general mean.

Thanks to the orthogonality issue mentioned earlier, SVD produces factors that include both positive and negative elements. We can also see this in algorithmic terms – as each successive component is deflated from **X,** it will generally introduce some negative residuals leading to negative elements in the next axis pair fitted.

3.2 Non-negative matrix factorization

3.2.1 A simple algorithm

The ALS algorithm for the SVD described above finds the latent dimensions sequentially, one at a time. **Lee and Seung (1999, 2001)** describe an algorithm for NMF based on multiplicative update rules, but fitting all *k* terms of the factorization at the same time. The algorithm does not require that all elements of the data matrix X be non-negative, and so can accommodate the situation that the original matrix **X** contains a few negative elements. As long as these negatives are few and moderate, the factor matrices will contain only non-negative elements. See **Outbox** 2. Suggestions for the necessary initial values are given later.

For a given rank k, we start by initializing the elements of the nonnegative matrices **L** and **R**: $\mathbf{L}_{ia} > 0$, $\mathbf{R}_{aj} > 0$, $\forall\ i, j, a,$ where i, j and a are the row, column and component indexes.

Then we apply the following multiplicative update rules until the difference between two iterations is small:

$$\mathbf{L}_{ia} \leftarrow \mathbf{L}_{ia} \, (\, \mathbf{XR}\,)_{ia} \,/\, (\, \mathbf{LR}^{\mathrm{T}}\mathbf{R})_{ia} \,, \ \forall\ i, a$$

$$\mathbf{R}_{ja} \leftarrow \mathbf{R}_{ja} \, (\, \mathbf{L}^{\mathrm{T}}\mathbf{X}\,)_{aj} \,/\, (\mathbf{L}^{\mathrm{T}}\, \mathbf{LR}^{\mathrm{T}})_{aj} \,, \ \forall\ j, a$$

At each iteration we update the current elements of **L** and **R** using specific multiplicative factors that relate to the current quality of the intended approximation, see Lin (2005) for further details regarding the properties of this algorithm and extensions using projected gradient methods.

Outbox 2. Multiplicative Update Rules, NMF algorithm.

3.2.2 Additional computational refinements

This steepest descent method using multiplicative updates is known to be very efficient during initial iterations but then converge slowly. Also, if **X** contains some negative values due to noise, multiplicative rules **may** yield negative values in the factoring vectors elements as well. A combination of initial multiplicative updates (“preliminary warm-up”) followed by an implementation of the projected gradient as described by **Lin (2005)** can speed up convergence substantially. Projected gradient follows the same principle as steepest descent, however takes the positive part of the solution found after each iteration in order to obey the non-negativity constraint on L and H. This combination will also tolerate negative values in **X** due to noise **(Badeau et al. 2011)**. This will be illustrated later through synthetic examples.

3.2.3 Initialization

Conventional SVD, and NMF as sketched in Outbox 2, involve least squares fits. However in contrast to SVD, the solution to NMF may not be unique, due to the non convexity of the residual sum of squares and so the choice of initial values affects not only speed of convergence, but also the quality of the local optimum to which the algorithm converges. To take the problem back to the relevant convex geometry which offers an essentially unique NMF, **D&S's** synthetic "Swimmer" dataset illustrates the impact of initialization. This data set depicts a figure with a fixed torso and four moving parts (limbs), each able to exhibit four articulations (different positions). Each image contains 12 pixels in the center for the fixed torso and four "limbs" of 6 pixels that can be in one of 4 positions. All combinations of all possible limb positions gives us 256 images. **Figure 2** shows the first 16 images from the Swimmer dataset. The dataset is therefore generated by 17 elements – the 16 limb positions and the invariant torso which one would hope to recover from a matrix factorization.

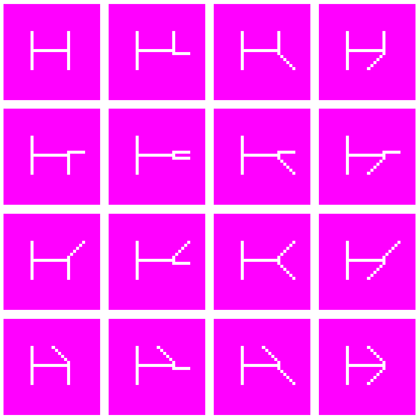


Figure 2: The first 16 images of the swimmer dataset, which depicts a figure with four moving parts (limbs), each able to exhibit four articulations (different positions).

The swimmer data set deviates from D&S's sufficient conditions as the torso is unchanged in all 256 images. Thus the existence of a unique NMF is not a given. And indeed, when applying our combination of multiplicative update rules and projected gradient with a random initialization, 15 of the 16 articulated parts are properly resolved, but one articulated part is missing. We instead find a "ghost" of the torso and this ghost also appears in some of the other parts (**Figure 3**). In D&S's original paper, all 16 limb parts were properly resolved but the torso was not. Furthermore, "ghosts" of the torso were seen in a number of the other parts, along with the legitimate limbs.

This picture comes from a random initialization, other random initializations lead to different solutions, none of which is fully satisfactory.

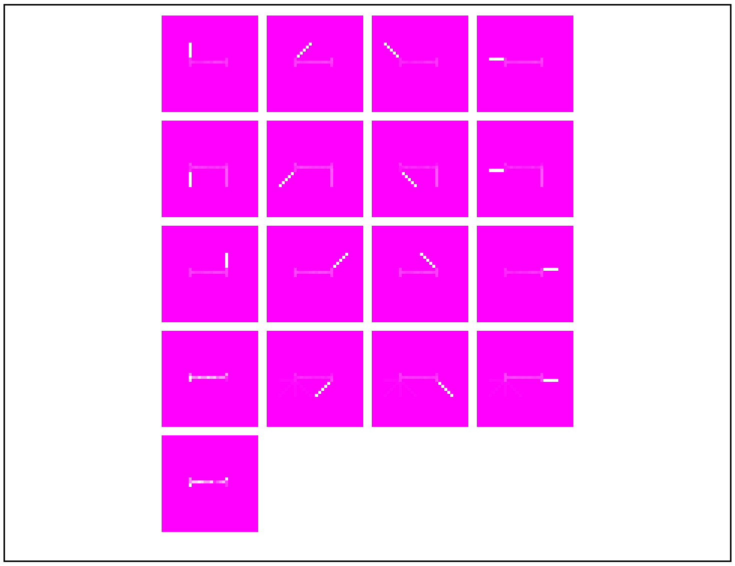


Figure 3: The pure archetypes of the Swimmer dataset, as found by NMF with random initialization. The two left bottom images show that one part is not properly resolved and is confounded with the torso. The “ghost” of the torso and this particular part appears in some of the other parts.

These local optima illustrate the importance of good initial values. **Boutsidis, 2007,** proposed starting from the singular value decomposition of **X**. Each pair of singular vectors can be decomposed into a sum of positive and negative pairs: $\mathbf{L} = \mathbf{L}^+ + \mathbf{L}^-$ and $\mathbf{R} = \mathbf{R}^+ + \mathbf{R}^-$. For each pair of singular vectors, the part ($\mathbf{L}^+$, $\mathbf{R}^+$) or ($-\mathbf{L}^-$,$-$ $\mathbf{R}^-$) that carries more variance is selected to start NMF. Boutsidis benchmarked this approach with a clustering (k-means) based approach (**Wild, 2004**) and showed that it performed comparatively well. Indeed, when applied to the Swimmer Dataset, all articulated parts and the torso are perfectly identified **(Figure 4)**.

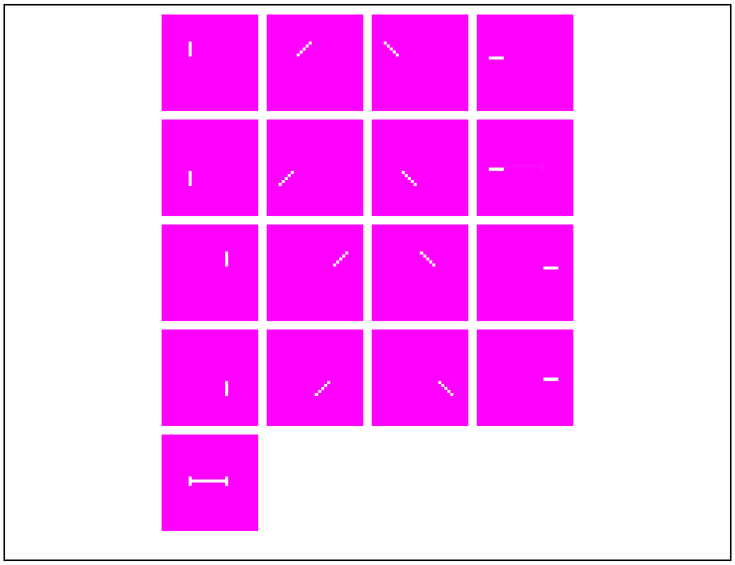


Figure 4: The parts of the swimmer dataset, as found by NMF with SVD-based initialization. All parts are perfectly resolved, including the torso, whose "ghost" has disappeared from other parts.

Most NMF algorithms use random initialization and then try to "bag" the results in some way (**Brunet et al. 2004**). Our experience suggests that the SVD-based initialization scheme usually gives good results. Nevertheless, we will give later a counter-example showing how deviation from D&S's separability rule can not always be surmounted by this initialization scheme.

4. SYNTHETIC EXAMPLES

In all of the synthetic examples to follow, the data matrices were constructed by matrix multiplication of left and right generating vectors, $\mathbf{X=LR^T}$. The left vectors are termed weighting vectors, and the right vectors are called spectral vectors. The elements of both the generating matrices **L** and **R** are either zero or uniformly distributed, U(1,2). A small, normally-distributed noise with mean 0 and standard deviation 0.1 was added to each cell. The generating vectors and the data matrices are presented visually as heatmaps. Our goal will be to retrieve the generating vectors of the synthesized matrices. Thus, for all synthetic examples we will compute SVD and NMF on the original, uncentered data matrices with their known rank. The choice of the factorization rank, which in real situations is unknown, will be discussed later in the analysis of our representative real sausage example.

4.1 Orthogonal generating vectors

There are four generating left and right vectors. The vectors are orthogonal (**Figure 5**). This is as simple a situation as imaginable, so matrix factorization should return the generating vectors.

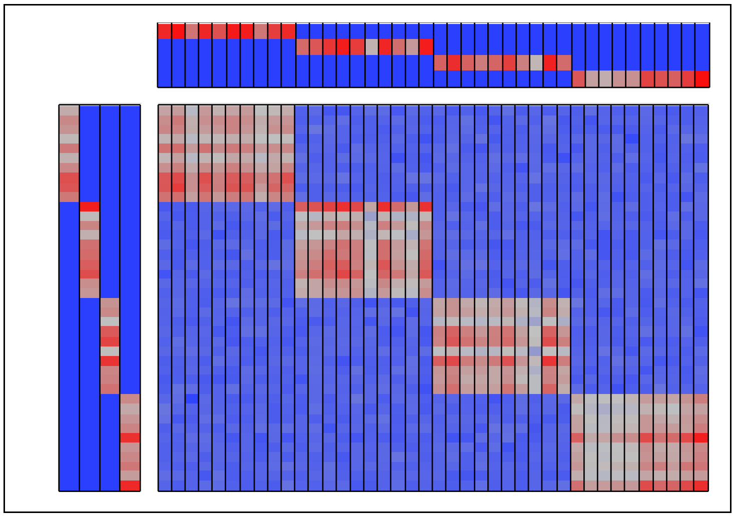


Figure 5: Simple synthetic example with four generating left and right vectors. The vectors are orthogonal.

We computed the SVD and NMF on this synthetic dataset. SVD recovered four vector pairs with singular values of 25.2, 24.7, 23.3 and 23.1. We also computed the NMF with four vector pairs. For this fully orthogonal example, both SVD and NMF (data not shown) recover the correct left and right generating vectors.

4.2 Realistic synthetic mixture

We now turn to another synthetic but more realistic example. There are two right and left generating vectors, but there are four kinds of individuals (**Figure 6**). Group 1 individuals are normal controls. Groups 2, 3 and 4 are diseased, but although there is but one named disease, say diabetes, there are two different etiologies, E1 and E2. Groups 2 and 3 suffer from one of these two “pure” etiologies. The unfortunate people in Group 4 suffer from both etiologies. Finally, there are a number of genes that do not participate in either etiology. These non-participatory columns add a “real” non-participatory noise component. There are two right generating vectors, each having ten active genes, one set for etiology 1 and the other for etiology 2. There are 20 inactive genes. A small normal noise with mean 0 and standard deviation 0.1 was added to each cell in the table.

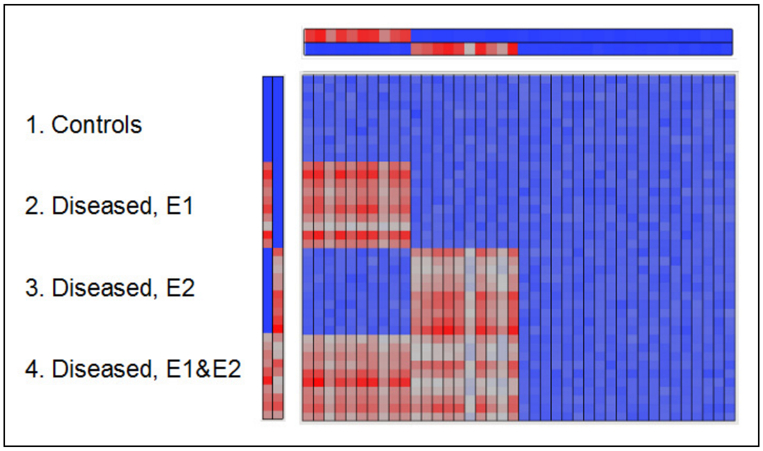


Figure 6: More realistic synthetic example. Two generating vectors lead to four kinds of individuals - normal controls and three disease groups with two etiologies.

Although synthetic, this model is based on a real situation. The Pima Indians in Arizona have a very high incidence of diabetes (**Baier and Hanson 2004).** The genetics is complicated and it is still not well understood, but is believed to involve multiple mechanisms. Furthermore, it is believed that some people have more than one mechanism; therefore, this situation would correspond to this synthetic example.

4.2.1 SVD analysis

How well does SVD treat this data set? Two vector pairs capture nearly all of the variance in the matrix, 99.5%, with singular values of 45 and 27.8.

The score plot clearly shows the four groups, but it has oriented the groups along a non-diseased/diseased principal axis. Component 2 contrasts the two etiologies, but does not provide useful information about how they differ (**Figure 7**).

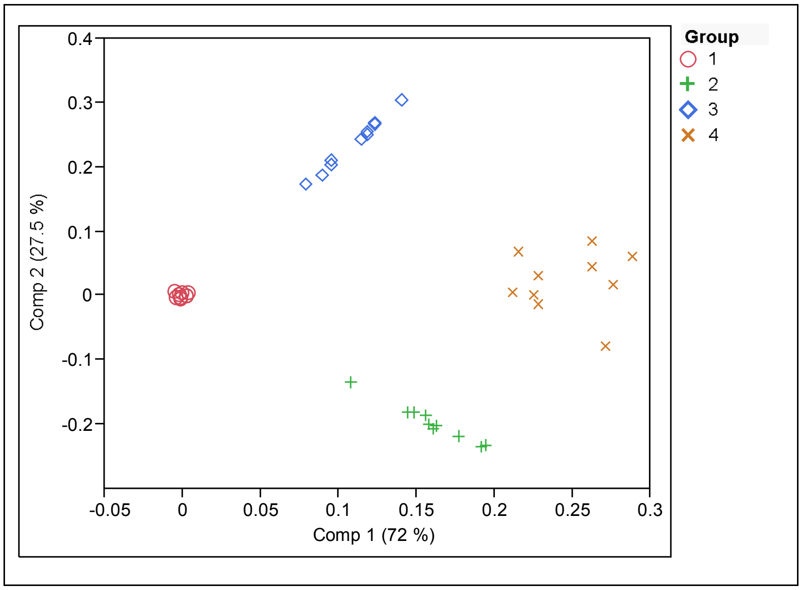


Figure 7: Two dimensions capture nearly all variance. Score plot shows four groups, but is oriented unhelpfully. “0” are disease free; “+” are Etiology 1; “◊” are Etiology 2; “x” have both diseases.

It is a simple fact that the two generating vector pairs are equally important, yet SVD has very unequal singular values. SVD’s first component (**Figure 8)** simply contrasts all affected and non-affected samples with no clues to the deeper structure of the data. Its second component does provide the additional insights on the structure of the data, but only by rotating the two components is there a clear picture of the two mechanisms.

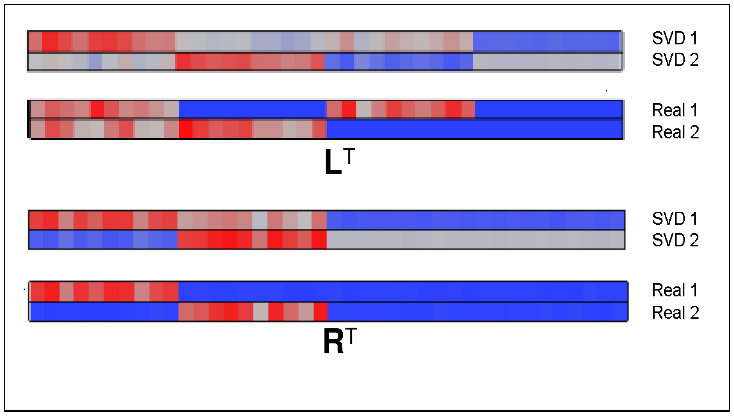


Figure 8: Heatmaps of SVD left and right singular vectors. Generating vectors are not recovered.

Plots of the scores, elements of the left factoring vectors, correctly show four distinct groups, however, assessing the nature of the data set is complicated. Singular vector pair 1 contrasts diseased versus non-diseased and singular vector pair 2 seems to contrast one etiology with the other, Groups 2 and 3, but the contrast is, at best, not clear.

4.2.2 NMF analysis

NMF analysis has no such difficulties of interpretation. The left vectors clearly show that there are four types of people. The right vectors show that there are two etiologies (**Figure 9**). This exactly matches the generating mechanism.

Regarding variance explained, SVD is a least squares method and gives the minimum residual variance possible from any *k* component approximation. NMF as we are using it here is also a least-squares method, but with non-negativity constraints, and so its residual variance is necessarily at least as large as that of a SVD with the same number of components. In this data set, NMF recovers 99.5% of the variance, almost as high as the SVD and providing an indication that the non-negativity constrained NMF solution is an adequate fit to the data.

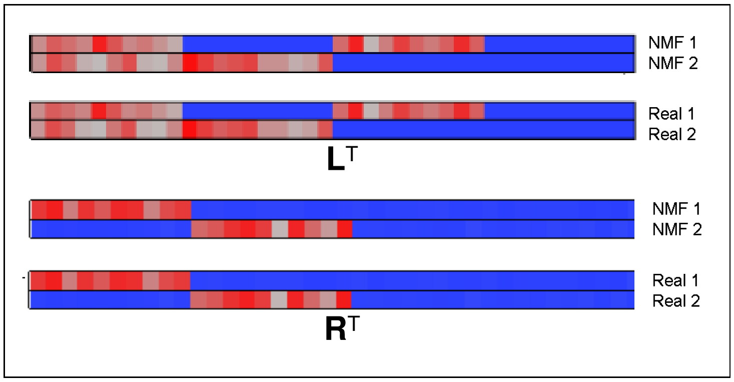


Figure 9: Heatmaps of the left and right vectors by NMF, showing perfect agreement with the real generating vectors.

The NMF score plot **(Figure 10)** gives a very satisfying 2x2 factorial layout. Controls are at the origin, the single etiologies are on each axis and the double etiologies are just where they should be.

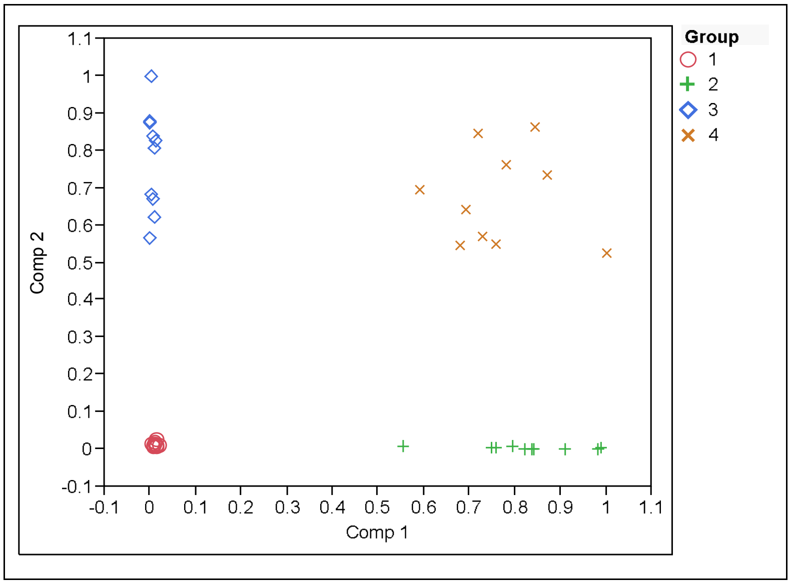


Figure 10: NMF score plot, showing 2x2 factorial layout with controls at the origin, single etiologies along each axis, and double etiologies loading on both axes.

4.2.3 Extended realistic mixture

We extend our realistic example by adding a set of 10 genes that are activated in either or both etiologies E1 and E2 (**Figure 11**). This situation can arise with a number of medical conditions. When a person gets sick from any of a number of diseases, multiple genes – for example for inflammation – are turned on. These general response genes are not specific to the disease at issue and can cause great confusion as they can be taken as markers for the disease.

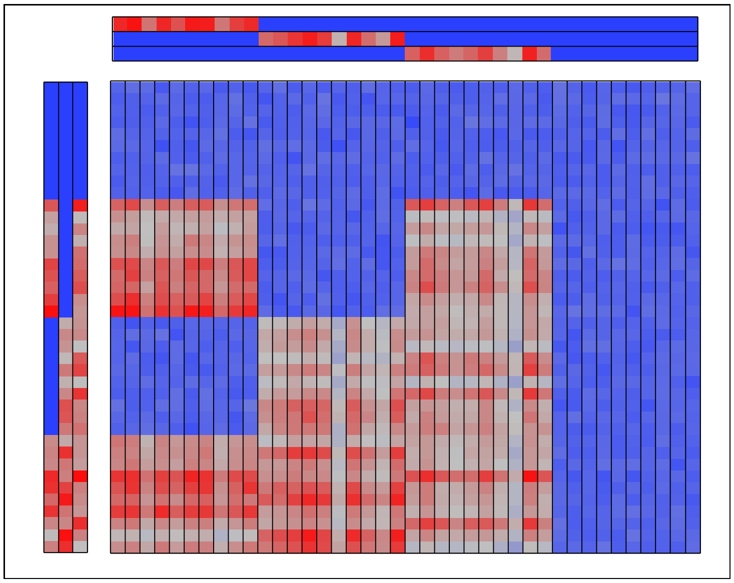


Figure 11: Even more realistic synthetic example. Two generating vectors lead to four kinds of individuals - normal controls and three disease groups with two etiologies. A third generating vector leads to a group of genes which are activated in multiple disease groups, e.g. inflammation genes, including both etiologies of the disease at issue.

Three SVD vector pairs capture nearly all of the variance in the matrix, 99.7%, with singular values of 60.8, 26.2 and 13.1. NMF also recovers 99.7% of the variance. However, now that "inflammation genes" are turned on for all groups except the control group, NMF does not find the groups well and has a “ghost” in one of the NMF vectors (**Figure 12**). This clearly illustrates how deviation from D&S’s sufficient conditions can not always be surmounted, even with a SVD-based initialization scheme.

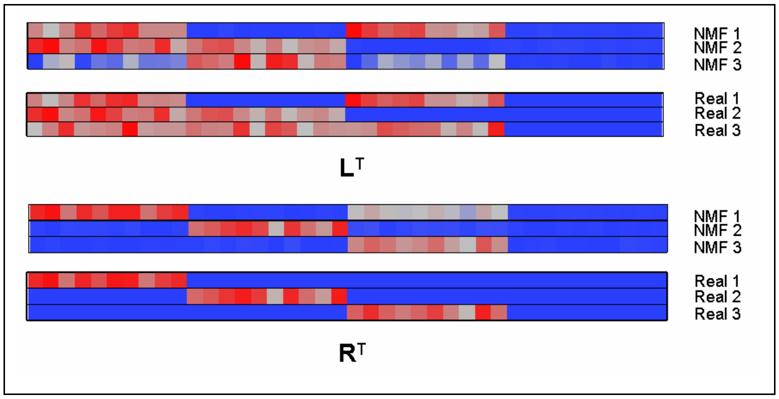


Figure 12: Heatmaps of NMF left and right singular vectors. The generating vectors are not recovered. Note the "ghost" of the third set of genes in the first right NMF vector.

4.2.4 Some Discussion of the Synthetic Samples

"Ghosts" could in principle be removed by a method that zeroed out unimportant terms in the alternating regressions. One familiar method of doing this is the LASSO, which adds an L1 penalty to a regression which has the effect, as the penalty coefficient increases, of getting progressively more sparse models. We implemented and tested a version of LASSO for NMF-LASSO by slightly adapting Lin's projected gradient algorithm (**Lin 2005**) to include the L1 penalty in its alternating regressions – see also **Hoyer (2004).**

Using our version of NMF-LASSO, the "ghost" of inflammation genes can be made to disappear (data not shown), but finding the right penalty was far from trivial: We increased it progressively until the ghost completely disappeared. Obviously this strategy only works when you know exactly what you want. In regression, the L1 penalty is commonly found using cross-validation. However, such a strategy is not viable for NMF where there is not a single regression but a long sequence of regressions with different dependent and predictor variables.

5. RETURN TO THE SAUSAGE EXAMPLE

Let us return now to the sausage example. First, salt is something of a stand in for water. NIR does not detect salt, but salt can modify hydration of proteins and carbohydrates. Consequently we focus on the first two factors of the experiment.

The spectrum formed by the minimum intensities at each wave length can be interpreted as a baseline spectrum, a "general factor", which will not tell us anything regarding the different levels of fat or starch. This general factor being a substantial part of the signal, we choose to take it off in a preliminary step and then proceed to the analysis of the resulting matrix, for both SVD and NMF. Note that this treatment is similar to subtracting the mean intensity done in the standard PCA analysis, while ensuring that the transformed **X** remains non-negative. See Laurberg and Hansen (2007) where they attempt to estimate a general baseline.

Our strategy will be to select the approximation rank *k* from SVD (we will come back to the choice of the approximation rank later in the discussion) then proceed to a rank *k* approximation of **X** from NMF. Finally, we will apply a rank *k*+1 approximation to illustrate the consequences of rank selection on results and interpretation.

5.1 SVD results

As noted in section 3.1, fitting the SVD to the uncentered matrix of this data set added one extra singular triple with right singular vector perfectly correlated with the overall mean spectrum (Pearson r = 1) and left singular vector highly correlated with a vector of constants (coefficient of variation = 3.7%). So we chose to fit the SVD to the centered matrix, where the first two components explain 99.2% of the total variance. In the score plot (**Figure 13**), the size of the "bubbles" is proportional to the level of fat, indicating that the first component is negatively correlated with fat. The factorial design appears underneath the score plot with reversed sign for fat to adjust for this negative correlation. On the left side of the score plot, red points (high level of starch) tend to appear on the top and blue points (low level of starch) at the bottom. On the right side of the score plot, the points – all of them corresponding to low fat samples and labeled by their row number in the matrix – form a separate cluster. Most noticeable, the direction of the relationship between starch and the second component reverses for these particular points – red points are low, blue and pink points are high, suggesting an interaction between fat and starch. We will come back to this particular point a little later.

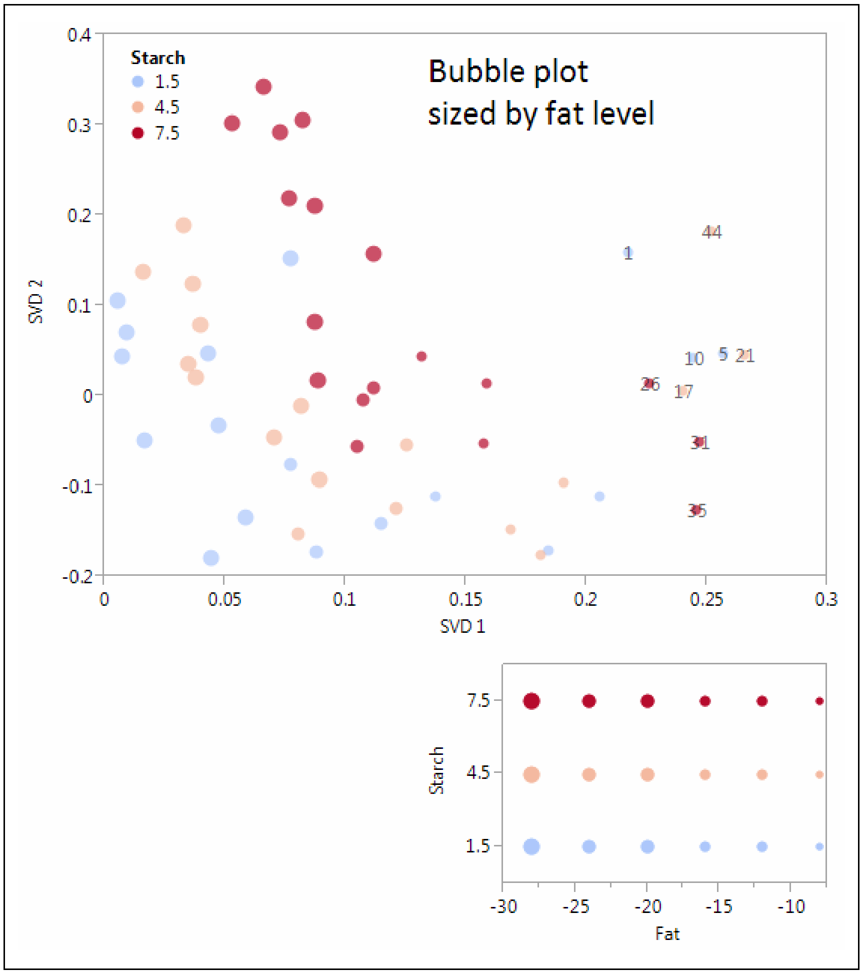


Figure 13: SVD score plot of sausage dataset. Size of "bubbles" is proportional to fat level; it decreases from left to right. In general, the starch level increases from lower left to upper right. Remember, we are looking at data from a 6 × 3 factorial design.

5.2 NMF results

Following our strategy, we chose to run NMF first with two components, which explain 99.7% of the total variance, slightly more than SVD.

Again, the size of the "bubbles" in the score plot is proportional to the level of fat (**Figure 14**). In the case of NMF the factorial design (which appears underneath the score plot) is more apparent, indicating that the components more likely represent the real factors involved. Given this, it is interesting that the plot of the wave vectors closely approximate the absorbencies of fat and starch.

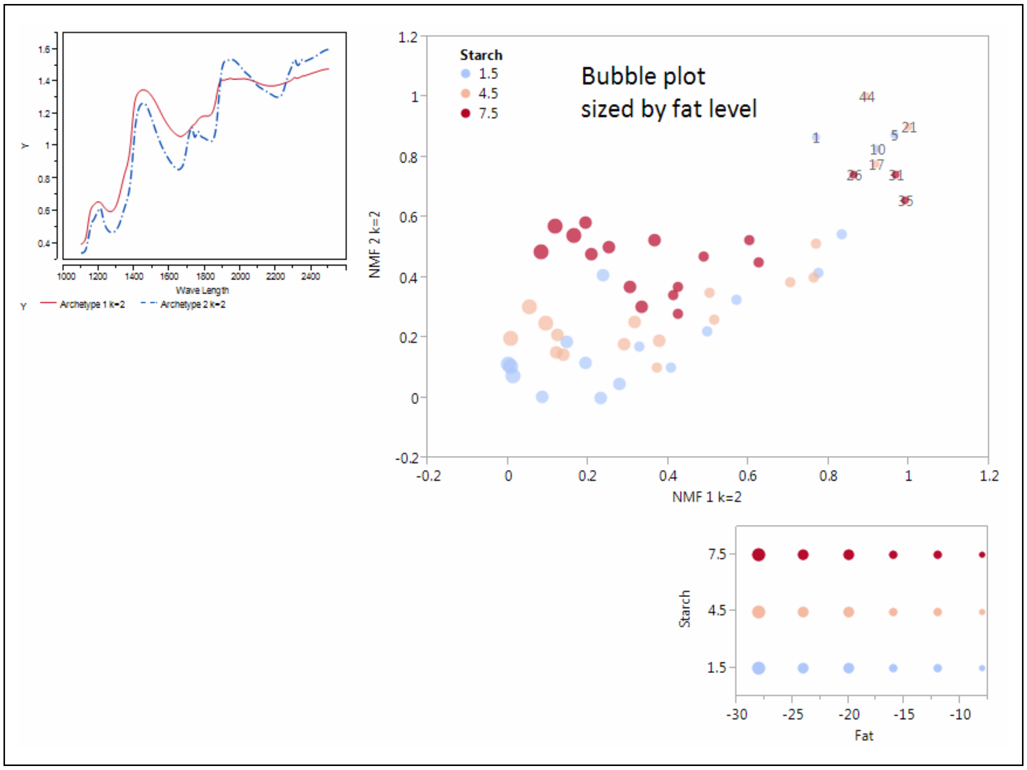


Figure 14: NMF score plot of sausage dataset with *k*=2. The factorial design is more apparent, so components more likely represent real factors.

As already noted in the SVD score plot, the direction of the relationship between starch and the second component reverses for low fat samples on the right of the plot. Most noticeably, these low fat samples have high loadings on both components, meaning that none of the archetypal spectra really fit them. Would one more archetype help explain these samples specifically? Actually it does, as illustrated by the 3D plot (**Figure 15**), in which labeled points are clearly identified in the top region of the third axis.

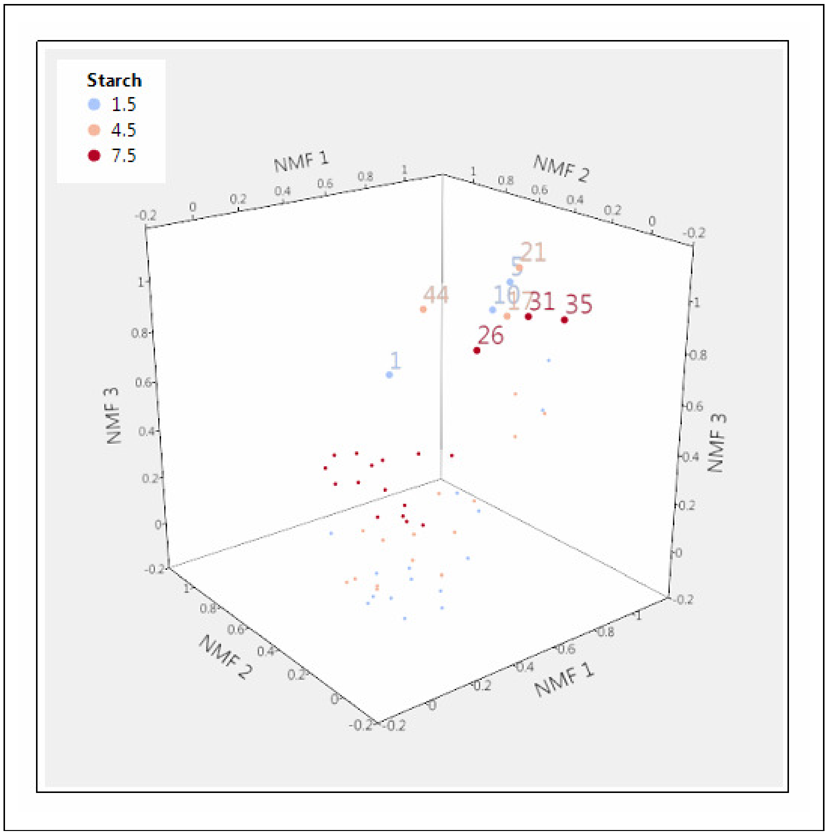


Figure 15: 3D score plot of NMF, *k*=3 . Labeled points are clearly identified in the top region of the third axis.

If we look back at the score plot of the first two components, we see now that the factorial design (which appears underneath the score plot) is even more apparent than with *k* = 2 (**Figure 16**).

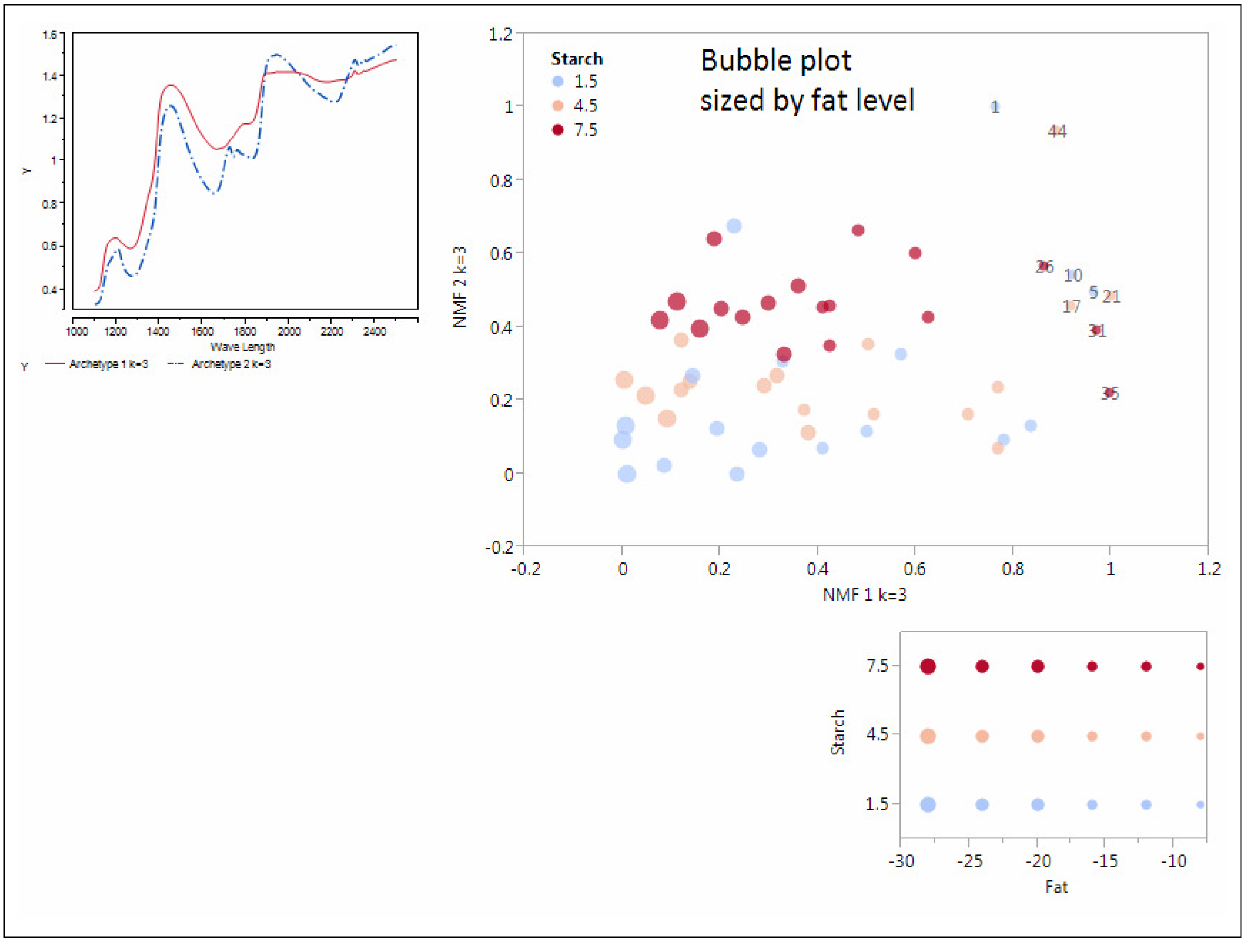


Figure 16: Score plot of the first two components with k = 3. Factorial design clearer than with k = 2.

We also note that the first archetypal spectrum obtained with $k = 2$ remains almost unchanged with $k = 3$. In contrast, the second archetypal spectrum effectively splits the second dimension into two parts, an example of the “multiple mechanisms” capability of the NMF. The wave lengths that are responsible for this split are most visible around 1700 and between 1900 and 2500 nm (**Figure 17**).

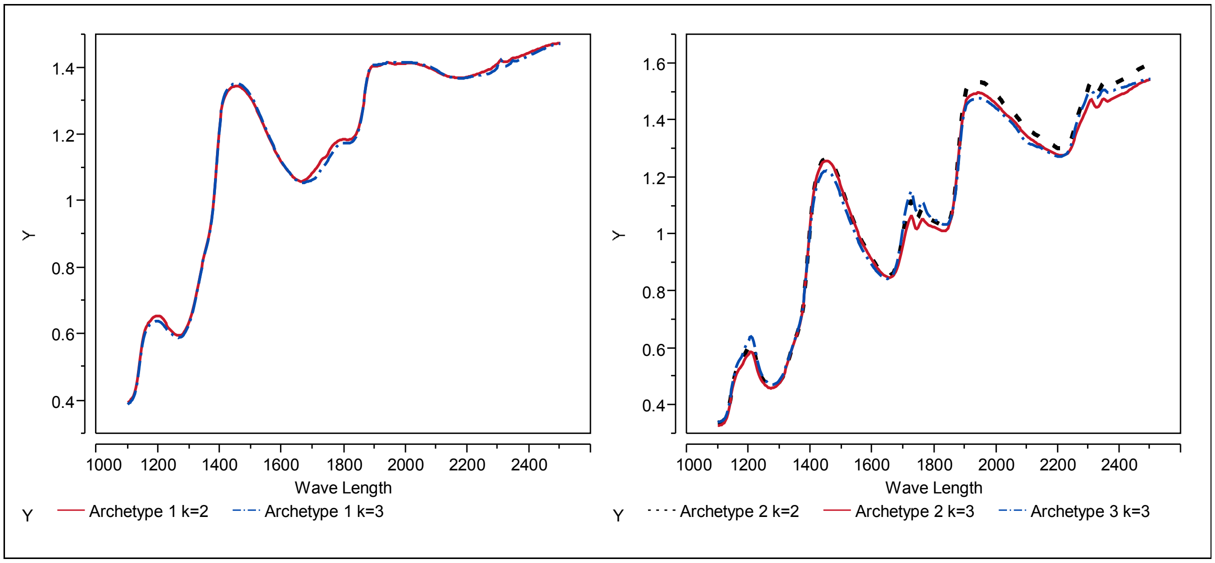


Figure 17: Archetypal spectra with *k*=3. *k*=2. Left: First archetype almost unchanged with *k* = 3. Right: Third dimension splits second archetype, k=2, into two archetypes. Wave lengths for split most visible around 1700 and between 1900 and 2500 nm.

Noting that the percentage of the explained variance is already high with *k* = 2, we assume that adding a component should help explain, not predict. Indeed, let us come back now to the interaction between starch and fat mentioned earlier. When fat is at its lowest level, the starch exists in an aqueous environment and so the salt, soluble in water, but not oil, has access to it. The salt itself will have no vibrational structure, but will alter the starch vibrations as it alters the waters of hydration. The fact that the differences are so slight indicates that only certain types of bonds are affected, namely those that show up at the specific vibrations around 1700 and between 1900 and 2500 nm thanks to the addition of a third component.

As with the synthetic examples, a heatmap of the first two components (SVD and NMF), starch and fat levels will illustrate these results (**Figure 18**). Note that we skipped the low fat samples that reversed the relationship between starch and the second component in both SVD and NMF, due to the intrusion of salt in starch vibrations at such low levels of fat. We also took the negative sign of the first component to achieve positive correlations between fat and the found components (SVD and NMF). The correlation between fat, starch and NMF factors appears strikingly clearer than with SVD. Correlation levels confirm this observation (**Table 1**).

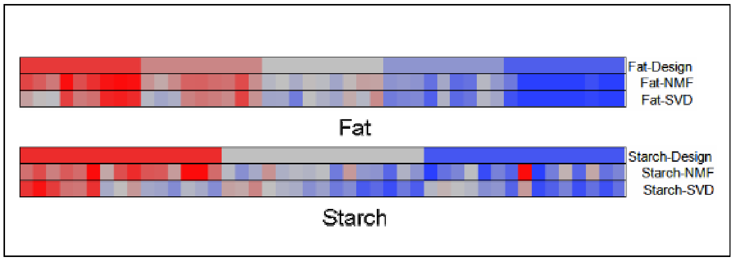


Figure 18: Heatmaps of the design, NMF and SVD vectors. Top: Fat, 1st components; Bottom: Starch, 2nd components.

	Fat-Design	Fat-NMF	Fat-SVD
Fat-Design	1.00	0.94	0.88
Fat-NMF	0.94	1.00	0.97
Fat-SVD	0.88	0.97	1.00

Table 1a : Correlation between fat and 1st component (NMF and SVD)

	Starch-Design	Starch-NMF	Starch-SVD
Starch-Design	1.00	0.70	0.51
Starch-NMF	0.70	1.00	0.58
Starch-SVD	0.51	0.58	1.00

Table 1b : Correlation between starch and 2nd component (NMF and SVD)

Finally, we note that SVD applied on the centered matrix gives similar results to SVD applied on the baseline-free matrix (data not shown), showing that baseline-centering led to the same diagnosis as mean-centering, but without losing non-negativity.

These results support the claim that NMF retrieves “mechanisms,” regions where fat or starch adsorb more strongly. We also added on the side of the score plot the corresponding archetypal spectra. This greatly facilitates the interpretation of the score plot and helps visually identify the wave lengths which differentiate the two archetypes. Again, we note that the factorization of **X** is not unique, since $\mathbf{DD}^{-1}$ where **D** is non-negative diagonal can be inserted between **L** and **R**. We scaled the elements in the vectors of **L** to have a maximum value of 1.00. This way, coordinates on the score plot can be viewed as weights on either first or second archetype.

6. DISCUSSION

The SVD and NMF are alternative ways to examine a data matrix. The major properties of SVD and PCA are well-known. The data matrix is decomposed into left (score) and right (loading) vector pairs. Adding more vector pairs allows better and better approximation to the elements of **X**. Conceptually the NMF is nothing more or less than an SVD in which negative elements in either the left or the right singular vectors are prohibited. As such, in settings where the data matrix **X** consists of non-negative elements (plus perhaps some zero-mean noise), the NMF factorization is likely to be more interpretable, and to more plausibly represent an actual generating mechanism. SVD can be looked at as optimizing the prediction of the elements of X, whereas NMF is attempting to explain the data via L and R, along the lines of **Shmueli G. (2010),** to predict or to explain**.**

As the left and right factors of the NMF are conceptual analogs of the left and right singular vectors in a SVD, they may be used for all the same purposes – for example cases may be clustered using the rows of **L**, and variables may be interpreted using the rows of **R.**

A graphic illustration of the power of the NMF method is in the dramatic picture in **Lee and Seung (1999)**, showing the decomposition of photographs of faces into ears, eyes, nose by the successive columns of the NMF. Other papers have used such terms as "meta genes" to describe groups of genes that group together **(Brunet et al. 2004).** Clearly the utility of NMF is in the direction of interpretation. The elements of the factoring vectors lead to an interpretation of how the internal components of X are combined in each sample. For example, in the NIR case, sets of contiguous wavelengths might be taken as response to a specific type of chemical bond. In the case of genes, sets of genes might be up or down regulated together. In the case of metabolites, molecules in a pathway might be up or down regulated together.

Mixtures deserve serious mention. NMF appears to be able to decompose mixtures into their component parts. The earliest chemistry example of what is effectively NMF that we have found is **Lawton and Sylvestre (1971).** There, on page 628, equation (31) is the separability rule (R2) of **D&S** in the context of non-binary, spectro photometric curves (but for k=2). In our NIR example, "pure" spectra are the vectors determined by NMF. The weights in the left factor **L** give how those spectra are put together to approximate the sample values as given in **X**. In psychometrics, NMF may prove a useful tool for resolving test data matrices into factor-like ability scores and matching subject loadings. In a more speculative vein for medical examples, vectors of **R** might correspond to different etiologies. For some people the disease may involve only one of the etiologies (i.e. right vectors), for other people it may involve multiple etiologies. The elements of **L** give how those etiologies are combined for a specific person. Alternatively, if we have people with different stages of a single disease, then the vector pairs in $\mathbf{LR}^{\mathrm{T}}$ could give how the disease is progressing when looked at over multiple people. The progression of metabolic syndrome to diabetes comes to mind.

The selection of the number of components *k* is a decision that has to be made. When you do a SVD, each new pairs of singular vectors is orthogonal to all the vectors that went before. If you overfit by choosing too large a rank, you will be trying to interpret pairs of junk coefficient vectors generated by noise, a process with high potential for misleading conclusions. A common approach is to make a scree plot, a plot of the series of successive eigenvalues against the integers. Conceptually, the scree plot drops steeply while the terms are still capturing "structure" and then flattens once they are capturing "noise." Guided by this idea the analyst makes an "arts and crafts" judgment call on the value of *k*, the number of signal components. The scree plot is open to many objections: like the fact that its shape can depend a lot on whether you plot on a natural, a square or a log scale. While these objections are legitimate, the scree plot is still widely used.

Unlike the SVD, the NMF with *k* components is not a concatenation of *k* successive optimal terms, but also seeks a best rank-*k* approximation to the original data matrix within the limitations that its left and right factors have only non-negative elements. So to the extent that NMF is finding a good rank-*k* approximation, it should be close to a linear transformation of the SVD. However, NMF does not have an orthogonality constraint, and so it is able to create a new dimension by, say, copying one of the row vectors and splitting the corresponding column vector into two parts. This has much less potential for confusion than does an over fitted SVD component. So getting *k* just right should not be as critical as with SVD.

Questions remain over how to pick the dimensionality of a NMF, and of numeric diagnostics that can be applied to **X** to decide whether a NMF is likely to succeed. By fitting a NMF and SVD of the same dimension to a data set and comparing their variance explained, the user will be alerted to the situation where the non-negativity constraints are not supported, but this still leaves open the possibility that some transformation of the left and right factors would lead to better interpretation. Also of concern is that, because of non-convexity, fitting algorithms are not guaranteed to converge to the global optimum, but are to some degree at the mercy of their initialization. These are all valuable avenues for further research.

Data Sets and Software

The data sets and SAS JMP scripts used in this paper can be downloaded from www.niss.org/irMF. The datasets are also available as supplemental material from TAS. Orange is an open source and free statistical analysis system based on Python, http://orange.biolab.si/. There is a module, Orange-NMF, for Orange that deals with various aspects of non-negative matrix factorization, http://orange.biolab.si/addons/.

REFERENCES

Badeau, R., Bertin, N., and Vincent, E. (2011), "Stability analysis of multiplicative update algorithms for non-negative matrix factorization." *ICASSP*, 2148-2151.

Baier, L.J., and Hanson, R.L. (2004), "Genetic studies of the etiology of type 2 diabetes in Pima Indians: hunting for pieces to a complicated puzzle." *Diabetes,* 53, 1181-1186.

Boutsidis, C., and Gallopoulos, E. (2007), "SVD based initialization: A head start for nonnegative matrix factorization". *Journal of Pattern Recognition*, http://dx.doi.org/10.1016/j.patcog.2007.09.010.

Brunet, J.P., Tamayo, P., Golub, T.R. and Mesirov, J.P. (2004), "Metagenes and molecular pattern discovery using matrix factorization." *Proceedings of the National Academy of Science,* 101, 4164–4169.

Devarajan, K., (2008), "Nonnegative matrix factorization: an analytical and interpretive tool in computational biology", *PLoS Comput Biol*. 25;4(7):e1000029.

Donoho, D., and Stodden, V. (2004), "When does non-negative matrix factorization give a correct decomposition into parts?" *Advances in Neural Information Processing System16*, Cambridge, Mass, USA: MIT Press. pages 1141-1148.

Ellekjær, M.R., Isaksson, T., and Solheim, R. (1994), "Assessment of sensory quality of meat sausages using near infrared spectroscopy." *Journal of Food Science*, 59, 456-464.

Faber, N. M., Meinders, M. J., Geladi, P., Sjöström, M., Buydens, L. M. C., and Kateman, G., (1995), "Random error bias in principal component analysis. Part I. derivation of theoretical predictions." *Analytica Chimica Acta*, 304, 257–271.

Gabriel, K.R. and Zamir, S. (1979), "Lower rank approximation of matrices by least squares with any choice of weights." *Technometrics*, 21, 489–498.

Good, I.J. (1969), "Some applications of the singular decomposition of a matrix." *Technometrics*, 11, 823-831.

Greenacre, M.J., and Underhill, L.G. (1982), "Scaling a data matrix in a low-dimensional Euclidean space." in DM Hawkins (ed) *Topics in Applied Multivariate Analysis*, Cambridge University Press.

Hoyer PO (2004), "Nonnegative matrix factorization with sparseness constraints". *Journal of Machine Learning Research,* 5, 1457–1469.

Lawton, W.H., and Sylvestre, E.A. (1971), "Self modeling curve resolution." *Technometrics*, 13, 617-633.

Lee, D.D., and Seung, H.S. (1999), "Learning the parts of objects by non-negative matrix factorization." *Nature,* 401, 788-791.

Lee, D.D., and Seung, H.S. (2001), "Algorithms for non-negative matrix factorization."
in Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 556-562. MIT Press.

Lin, C. J. (2005), "Projected gradient methods for non-negative matrix factorization". Tech. Report Information and Support Service ISSTECH-95-013, Department of Computer Science, National Taiwan University.

Liu, L., Hawkins, D.M., Ghosh, S., and Young, S.S. (2003), "Robust singular value decomposition analysis of microarray data." *Proceedings of the National. Academy of Science*, 100, 13167-13172.

Langsrud, Ø. (2006), "Explaining correlations by plotting orthogonal contrasts." *The American Statistician,* 60, 335-339.

Laurberg H., and L.K. Hansen (2007). "On affine non-negative matrix factorization." ICASSP

Shmueli G. (2010), “To Explain or to Predict?” *Statistical Science,* 25, 289-310.

Wild, S, Curry, J., and Dougherty, A. (2004), “Improving non-negative matrix factorizations through structured intitialization.” *Pattern Recognition,* 37, 2217–2222.