Nonnegative Matrix Factorization

Elena Geminiani
Ph.D. in Statistical Sciences
Academic year 2016/2017
elena.geminiani4@unibo.it

Summary

In the era of "Big Data", there is the urgent need of correctly processing enormous amount of data. If they take nonnegative values, the analysis cannot be accomplished by means of classical tools since they do not guarantee to preserve nonnegativity. In this connection, nonnegative matrix factorization (NMF) can be applied, since it provides a low-rank approximation of a target matrix, while extracting its meaningful features. This essay presents the theoretical details about NMF, the main numerical algorithms employed to solve the factorization problem, an overview of the most recent extensions, and empirical applications to image processing and text mining.

Table of Contents

Table of Contents **3**

1 Nonnegative Matrix Factorization **5**

1.1 Introduction 5
1.2 The method 7
1.3 Matrix factorization 9
1.3.1 Objective functions 10
1.3.2 Numerical algorithms 11
1.3.3 Initialization and rank choice 15
1.4 An ill-posed problem 18
1.5 Extensions 19

2 Application to Text Mining **21**

2.1 Introduction 21
2.2 Analysis 22

Bibliography **31**

1 Nonnegative Matrix Factorization

1.1 Introduction

The recent developments in technology have resulted in increasing quantities of data, rapidly overwhelming many of the classical analysis tools available. Processing this large amount of data requires powerful methods for their representation and dimension reduction.

In several real-world applications, it is frequent to deal with nonnegative data, for example the pixel intensities that produce an image, the amplitude spectra, the occurrence counts of a certain event, the scores performed by some users, the stock market values, and so on. The analysis of this kind of data, however, cannot be accomplished by means of the well-known matrix factorization techniques - like *singular value decomposition* (SVD) - and of the popular tools for reducing dimensionality and detecting latent structures - like *factor analysis, principal component analysis* and *cluster analysis* - that have been extensively studied in numerical linear algebra, since they cannot guarantee to preserve nonnegativity. As a matter of fact, despite its various strengths (optimality property, fast and robust computation, unique factorization and orthogonality), the SVD does not reveal anything about the data collection since the factors provide no interpretability. Such constraint clearly makes much of the previous work inapplicable to the present case (Press et al. 1989) and opened the way to the development of **nonnegative matrix factorization** (NMF).

This method has appeared in slightly different variants firstly in Jeter and Pye 1981, under the name of *nonnegative rank factorization*, then in Paatero and Tapper

1994, under the historically unfortunate name of *positive matrix factorization*. However, it was only after the publication of the article *"Learning the parts of objects by non-negative matrix factorization"* by Lee and Seung 1999 that this technique gathered the deserved credit and attention.

Nowadays, NMF stands out as one of the most popular low-rank approximations, used for compression, visualization, feature selection and noise filtering. It aims at replacing the original data by a lower dimensional representation obtained via subspace approximation, that is, it constructs a set of basis elements such that the linear space that they span approximates the data points as closely as possible. Such basis elements can then be used for identification and classification, which makes NMF a powerful unsupervised learning technique allowing to automatically cluster similar elements into groups, and hence a valuable alternative to common cluster algorithms, like *k-means*.

Furthermore, it proposes itself as a possible substitute of *principal component analysis* in the context of feature learning, of *hidden Markov models* in the field of temporal segmentation, of *independent component analysis* in the area of filtering and source separation, and of *vector quantization* in coding applications. By providing a low-rank approximation, the nonnegative decomposition allows to maximize hardware efficiency, since the low-rank matrix requires less storage than the original one, besides providing a cleaner and more efficient representation of the relationship between data elements.

The fields where NMF has been successfully applied for the analysis of high dimensional data are sky-high. Some examples include image processing (e.g. to recognize the main traits of a face), text mining (to recover the main topics of a set of documents), bioinformatics and computational biology (e.g. for molecular pattern discovery), signal processing, air emission control, blind-source separation (e.g. to separate voices in speech mixtures or voice from background, or separate singing voice from accompaniment or musical instruments in polyphonic mixtures), music analysis and transcription (e.g. to recognize musical notes played by piano, drums or multiple instruments), collaborative filtering, community detection and portfolio diversification. However, many other disciplines are expected to start employing NMF in the near future, due to its innate ability of automatically extracting sparse, meaningful, and easily-interpretable features from a set of nonnegative data.

1.2 The method

Nonnegative matrix factorization is a tool that permits the decomposition of multivariate data (arranged in a matrix-like format), and is concerned with the search of latent features that are presumed to be responsible for the generation of the directly observable variables (Nakayama and Shimojo 1992, Hinton et al. 1995). One suitable way to characterize the interrelationships between multiple variables contributing to the observed data is assuming each element of the data matrix to be a linearly weighted score by a specific entity based on several factors. Specifically, a set of multivariate n-dimensional nonnegative data vectors are placed in the columns of an $n \times m$ matrix $\boldsymbol{V}$ (*target matrix*), where m is the number of entities. The target matrix is then approximately factorized into the product of two nonnegative factor matrices, an $n \times r$ matrix $\boldsymbol{W}$ and an $r \times m$ matrix $\boldsymbol{H}$, such that:

$$\boldsymbol{V} \approx \boldsymbol{W}\boldsymbol{H} \qquad \text{with } (\boldsymbol{W}, \boldsymbol{H}) \geq 0$$

where the columns in $\boldsymbol{V} = \left[\boldsymbol{v}_1 \ldots \boldsymbol{v}_\mu \ldots \boldsymbol{v}_m\right]$ represent the data points, those in $\boldsymbol{W} = \left[\boldsymbol{w}_1 \ldots \boldsymbol{w}_a \ldots \boldsymbol{w}_r\right]$ the latent features, and those in $\boldsymbol{H} = \left[\boldsymbol{h}_1 \ldots \boldsymbol{h}_\mu \ldots \boldsymbol{h}_m\right]$ the coordinates of each data point in the factor matrix $\boldsymbol{W}$.

The rank of the factorization r is generally chosen to be smaller than n or m (i.e., $r << min(n, m)$), and such that $(n+m)r < nm$, so that $\boldsymbol{W}$ and $\boldsymbol{H}$ are smaller than $\boldsymbol{V}$ and hence their product can be regarded as a compressed form of the original data matrix. In this way the information contained in $\boldsymbol{V}$ can be summarized and split into r factors. It is thus clear the crucial role played by this parameter: too high values could result in potentially serious overfitting problems, whereas too small values could lead to a bad representation of the data.

The matrix $\boldsymbol{W}$ is called *basis matrix* since its column vectors approximately form a basis (or better, a pseudo-basis since the relation is approximate) for the vector space spanned by the columns of $\boldsymbol{V}$ (or at least for the nonnegative orthant in that space, i.e., all vectors in that space with nonnegative components). Since relatively few basis vectors are used to represent many data vectors, a good approximation can only be achieved if the factors discover the true latent structure present in the data. The matrix $\boldsymbol{H}$ is called *mixture coefficients matrix*, since its column vectors provide the weights that are used in the linear combination of the columns of the

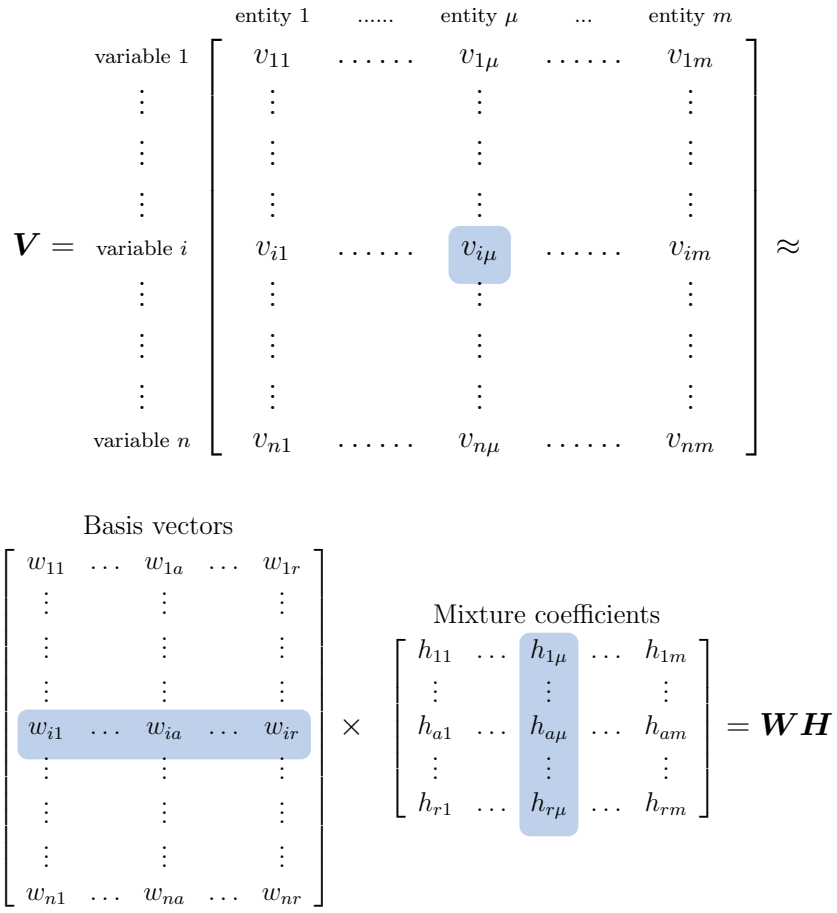


Figure 1.1: Decomposition of the target matrix in nonnegative matrix factorization

basis matrix to reconstruct the target matrix.

The NMF approximation statement can be equivalently rewritten elementwise as $v_{i\mu} \approx (\boldsymbol{WH})_{i\mu} = \sum_{a=1}^{r} w_{ia} h_{a\mu}$, for $i = 1, \ldots, n$, and $\mu = 1, \ldots, m$ (Figure 1.1), where the element $v_{i\mu}$ of the target matrix represents the score obtained by entity μ on variable i, w_{ia} denotes the loading of variable i on factor a, whereas $h_{a\mu}$ is the response of entity μ to factor a.

The nonnegativity constraints placed on the elements of the factor matrices are compatible with the intuitive notion of additively combining parts (that is, subsets of the visible variables activated by specific factors) to generate the whole representation, which is how NMF learns a parts-based representation.

The columns of the matrices $\boldsymbol{W}$ and $\boldsymbol{H}$ are often sparse since they generally contain a large portion of vanishing coefficients. *Sparseness* is one of the greatest

strengths of NMF, since a sparse representation represents the majority of data by using only few active factors, making the feature set and the coefficients easier to interpret.

An Example: Image Processing

The first field of application to which nonnegative matrix factorization has been applied was image processing and face recognition. Lee and Seung 1999 claimed that this innovative method could provide a convenient way to learn parts of faces (like the nose, the eyes, and the mouth), that could successively be used to represent a visage.
In this context, the columns of the target matrix are made up by the pixel intensities of different facial images, the columns of $\boldsymbol{W}$ constitute the so called *basis images*, whereas the columns of $\boldsymbol{H}$ are named *encodings*, that is, the coefficients by which a face is represented with a linear combination of basis images.

1.3 Matrix factorization

The factorization of the matrix $\boldsymbol{V}$ can be achieved in several ways using numerical algorithms. They are usually iterative algorithms, which means that the new values of $\boldsymbol{W}$ and $\boldsymbol{H}$ minimizing a certain cost function are found at each iteration.

The majority of them use a block coordinate descendent scheme, namely, they optimize a loss function with respect to alternatively one of the two matrices, $\boldsymbol{W}$ or $\boldsymbol{H}$, while keeping the other fixed. The justification of the use of such approach is that whereas the nonlinear optimization problem is not simultaneously convex in both $\boldsymbol{W}$ and $\boldsymbol{H}$, the subproblem in either the basis matrix or the coefficients matrix is indeed convex. Since the problem is also generally symmetric in $\boldsymbol{W}$ and $\boldsymbol{H}$, most algorithms actually focus on updating only one among the two matrices, and use the same update also for the other one.

Every algorithm of this type is repeated until a convergence criterion is met, which for nearly all NMF algorithms is performing the numerical operations for a fixed number of iterations. Although the natural stopping criterion would be terminating the procedure whenever $\|\boldsymbol{V} - \boldsymbol{W}\boldsymbol{H}\| \leq \epsilon$, for any positive ϵ, the com-

putational effort to check it is huge, therefore the widely shared choice is fixing a priori a number of iterations, which is very simple to implement. Most NMF algorithms adhere to the following framework.

General framework of NMF algorithms

Input: A nonnegative matrix $\boldsymbol{V} \in \mathbb{R}_+^{n \times m}$ and the factorization rank r

- Generate some initial matrices $\boldsymbol{W}^{(0)} \geq 0, \boldsymbol{H}^{(0)} \geq 0$
- **for** $t = 1, 2, \ldots,$ stopping time
  $\boldsymbol{W}^{(t)} = \text{update } (\boldsymbol{V}, \boldsymbol{H}^{(t-1)}, \boldsymbol{W}^{(t-1)})$
  $\boldsymbol{H}^{(t)^T} = \text{update } (\boldsymbol{V}^T, \boldsymbol{W}^{(t)^T}, \boldsymbol{H}^{(t-1)^T})$
  end for

Output: A rank-r NMF of $\boldsymbol{V} \approx \boldsymbol{W}\boldsymbol{H}$, for $(\boldsymbol{W}, \boldsymbol{H}) \geq 0$

There are two main things that allow to differentiate between the numerous estimation procedures, namely, the choice of the *objective function* that represents the quality of the data representation, and the actual *numerical algorithm* employed to optimize it.

1.3.1 Objective functions

To find an approximate factorization $\boldsymbol{V} \approx \boldsymbol{W}\boldsymbol{H}$, one needs to define loss functions $L(\boldsymbol{V}, \boldsymbol{W}\boldsymbol{H})$ that quantify the quality of the approximation. Common cost functions are based on the square Euclidean distance between the two nonnegative matrices $\boldsymbol{V}$ and $\boldsymbol{W}\boldsymbol{H}$ (Lee and Seung 1997, Paatero 1997):

$$L(\boldsymbol{V}, \boldsymbol{W}\boldsymbol{H}) = \|\boldsymbol{V} - \boldsymbol{W}\boldsymbol{H}\|^2 = \sum_{i=1}^{n} \sum_{\mu=1}^{m} \left(v_{i\mu} - (\boldsymbol{W}\boldsymbol{H})_{i\mu}\right)^2$$

This loss function equivalently represents the square Froebenius norm of the error, and it implicitly assumes that the noise $\boldsymbol{N}$ present in the matrix $\boldsymbol{V} = \boldsymbol{W}\boldsymbol{H} + \boldsymbol{N}$ is Gaussian[1]. Though reasonable in many practical situations, a Gaussian noise

[1] From a statistical viewpoint, minimizing the Euclidean distance is equivalent to the optimization of the maximum likelihood criterion for a Gaussian probability distribution, where

may not be the best choice in presence of sparse nonnegative data. For this reason, other objective functions are more used in practice, like the generalized Kullback-Leibler divergence of $\boldsymbol{V}$ from $\boldsymbol{WH}$[2]:

$$L(\boldsymbol{V}, \boldsymbol{WH}) = D(\boldsymbol{V}||\boldsymbol{WH}) = \sum_{i=1}^{n}\sum_{\mu=1}^{m}\left(v_{i\mu}\log\frac{v_{i\mu}}{(\boldsymbol{WH})_{i\mu}} - v_{i\mu} + (\boldsymbol{WH})_{i\mu}\right)$$

Both measures are lower bounded by zero, and vanish if and only if $\boldsymbol{V} = \boldsymbol{WH}$, but whereas the former is a distance, the latter is not, since it is not symmetric in $\boldsymbol{V}$ and $\boldsymbol{WH}$; in particular, it reduces to the relative entropy (Kullback and Leibler 1951, Kullback 1997) when $\sum_{i=1}^{n}\sum_{\mu=1}^{m} v_{i\mu} = \sum_{i=1}^{n}\sum_{\mu=1}^{m}(\boldsymbol{WH})_{i\mu} = 1$, so that $\boldsymbol{V}$ and $\boldsymbol{WH}$ can be regarded as normalized probability distributions.

The nonlinear optimization problem is then minimizing the objective function with respect to the matrix factors, subject to the nonnegativity constraint. In practice, an optional regularization function $R(\boldsymbol{V}, \boldsymbol{WH})$ is usually added to the cost function in order to enforce desirable properties on the factor matrices, such as smoothness or sparsity (further details in section 1.5 and in Cichocki, Zdunek, Phan et al. 2009):

$$\min_{\boldsymbol{W},\boldsymbol{H}\geq 0}\Big[L(\boldsymbol{V}, \boldsymbol{WH}) + R(\boldsymbol{V}, \boldsymbol{WH})\Big]$$

As a consequence of the optimization problem, a sequence of matrices $(\boldsymbol{W}^{(t)}, \boldsymbol{H}^{(t)})$ is built such that at each step the value of the objective function is reduced. Lee and Seung 1999 pointed out that the exact form of the objective function is not as crucial as the nonnegativity constraints for the success of the factorization.

1.3.2 Numerical algorithms

The possible sequences of matrices $(\boldsymbol{W}^{(t)}, \boldsymbol{H}^{(t)})$, however, may differ not only in the specification of the objective function, but also in the choice of the optimization technique used to update them. As earlier mentioned, the cost functions are convex

$v_{i\mu} \sim \mathcal{N}\Big((\boldsymbol{WH})_{i\mu}, \sigma^2\Big)$

[2] The minimization of the generalized Kullback-Leibler divergence is equivalent to the optimization of the maximum likelihood criterion for a Poisson probability distribution, where $v_{i\mu} \sim \mathrm{Poi}\Big((\boldsymbol{WH})_{i\mu}\Big)$

in either $\boldsymbol{W}$ or $\boldsymbol{H}$, and not simultaneously in both variables, which makes unrealistic expecting the algorithms to find global minima, since only convergence to stationary points may eventually be guaranteed. Unlike the unconstrained optimization problem which can efficiently be solved using SVD, the complexity of NMF is generally *NP-hard*[3] (Vavasis 2009). For real-world problems, however, even local minima are useful as they provide as well the desirable properties of data compression and feature extraction.

There are many techniques from numerical optimization that can be applied to find local minima, and they are mainly classified in three categories:

1. **Gradient descent algorithms** (Cauchy 1847): they are perhaps the simplest technique to implement, but they are very slow to converge (if at all). Other methods such as conjugate gradient (Hestenes and Stiefel 1952, Straeter 1971) have faster convergence, at least in the vicinity of local minima, but are more complicated to implement. Unfortunately, the convergence of gradient-based methods is very sensitive to the choice of the step size - a serious inconvenient for large applications - and a convergence supporting theory is missing;

2. **Multiplicative update rules**: they are extensively used since they are simple to implement, applicable to sparse matrices, and they were proposed as numerical algorithm for solving NMF by Lee and Seung 2001, whose article launched the research on this topic;

3. **Alternating least squares algorithms** (Kim and Park 2007): the name comes from the fact that a least squares step is followed by another least squares step in an alternating fashion. They exploit the convexity of the objective function in either $\boldsymbol{W}$ or $\boldsymbol{H}$, which implies that, given one matrix, the other one can be found with a simple least squares computation. They are very fast, quick to convergence, and give accurate factors.

Due to the inefficiency in this framework of gradient descendent algorithms, the attention has been devoted to multiplicative rules and alternating least squares.

[3]Recently, Arora et al. 2013 described a subclass of nonnegative matrices (near separable matrices) for which NMF can be solved efficiently.

Multiplicative rules

Multiplicative update rules are by now the most employed numerical algorithm to solve NMF due to their tradeoff between speed, ease of implementation for solving the optimization problems, and desirable properties. They can be divided in two main classes, depending on the choice of the objective function. Specifically, if the loss function is based on the divergence measure, then $D(\boldsymbol{V}||\boldsymbol{WH})$ is non increasing under the following updates:

$$h_{a\mu} \leftarrow h_{a\mu}\frac{\sum_{i=1}^{n} w_{ia}\frac{v_{i\mu}}{(\boldsymbol{WH})_{i\mu}}}{\sum_{i=1}^{n} w_{ia}} \quad a=1,\dots,r \quad \mu=1,\dots,m$$

$$w_{ia} \leftarrow w_{ia}\frac{\sum_{\mu=1}^{m} h_{a\mu}\frac{v_{i\mu}}{(\boldsymbol{WH})_{i\mu}}}{\sum_{\mu=1}^{m} h_{a\mu}} \quad i=1,\dots,n \quad a=1,\dots,r$$

and it is invariant if and only if $\boldsymbol{W}$ and $\boldsymbol{H}$ are at a stationary point of the divergence. Similarly, if the loss function is based on the square Euclidean distance, then $\|\boldsymbol{V}-\boldsymbol{WH}\|$ is non increasing under the update rules[4]:

$$h_{a\mu} \leftarrow h_{a\mu}\frac{\left(\boldsymbol{W}^T\boldsymbol{V}\right)_{a\mu}}{\left(\boldsymbol{W}^T\boldsymbol{WH}\right)_{a\mu}} \quad a=1,\dots,r \quad \mu=1,\dots,m$$

$$w_{ia} \leftarrow w_{ia}\frac{\left(\boldsymbol{VH}^T\right)_{ia}}{\left(\boldsymbol{WHH}^T\right)_{ia}} \quad i=1,\dots,n \quad a=1,\dots,r$$

and it is invariant if and only if $\boldsymbol{W}$ and $\boldsymbol{H}$ are at a stationary point of the distance. It can be noticed that each update consists of a multiplication by a term, which is unity when $\boldsymbol{V}=\boldsymbol{WH}$, so that the perfect reconstruction of the data is necessarily a fixed point of the update rules. The fidelity of the approximation enters the updates through the normalized quotient $\frac{v_{i\mu}}{(\boldsymbol{WH})_{i\mu}}$.

The multiplicative rules are based on the *majorization-minimization framework*, since the estimates are the global minimizer of a quadratic function majorizing L, that is, a function that is larger than L everywhere and is equal to L at the current iteration. Hence minimizing that function ensures L to decrease, and therefore, leads to an algorithm for which the objective function monotonically decreases. Un-

[4]In practice 10^{-9} is added at the denominator to avoid division by zero

fortunately, there is no guarantee that the algorithm converges to a local minimum (Chu et al. 2004, Finesso and Spreij 2006). In particular, Berry et al. 2007 stated that the stationarity of a limit point occurred (be it a local minimum or not) when the point lay inside the feasible region, whereas it could not even be determined when the point lay on the boundary region.

Alternating least squares

The alternating least squares (ALS) NMF algorithms were developed with the intention of solving a couple of issues affecting the multiplicative rules. The first one is related to the fact that even though the multiplicative rules in practice often converge, they are relatively slow to reach convergence: in fact, the matrix $\boldsymbol{W}$ is only updated once before updating $\boldsymbol{H}$. They can hence be significantly accelerated using a more efficient alternation strategy.

In the basic ALS framework, one matrix is taken as fixed and the other is computed using least squares, then the former is updated while the other is keeping fixed. After each least squares step, a projection step ensuring nonnegativity follows, so that the possibly negative elements resulting from the numerical computation are set to zero. For example, if $\boldsymbol{W}$ is taken as fixed, then one solves for $\boldsymbol{H}$ in the matrix equation $\boldsymbol{W}^T\boldsymbol{W}\boldsymbol{H} = \boldsymbol{W}^T\boldsymbol{V}$, and sets all negative elements in $\boldsymbol{H}$ to zero. Afterwards, the newly computed matrix $\boldsymbol{H}$ is taken as fixed, and one solves for $\boldsymbol{W}$ in the equation $\boldsymbol{H}\boldsymbol{H}^T\boldsymbol{W}^T = \boldsymbol{H}\boldsymbol{V}^T$, setting to null all the resulting negative elements. Some extensions of ALS algorithms (like hierarchical alternating least squares NMF; Cichocki, Zdunek and Amari 2007) update one matrix several times before updating the other, taking advantage of the fact that some of the products need not be recomputed, giving rise to faster convergence techniques.

The other drawback of multiplicative rules concerns the so-called *locking property*, that is, once an element in $\boldsymbol{W}$ or $\boldsymbol{H}$ becomes zero, it must remain zero, since NMF only allows non-subtractive combination of parts to represent a whole. This implies that once the algorithm starts heading down a path towards a fixed point, even if it is a poor fixed point, it must continue in that vein. On the contrary, the ALS algorithms are more flexible, allowing the iterative process to escape from a path heading towards a poor local minimum. They were proved to converge to a local minimum, but the additional constraint on least squares generally makes the optimization computationally demanding.

1.3.3 Initialization and rank choice

All NMF algorithms need to be initialized with a value for $\boldsymbol{W}^{(0)}$ and/or (depending on the method) $\boldsymbol{H}^{(0)}$, from which the iteration process can start. Due to the high dimensionality of the problem, and since there is no global minimization algorithm, the choice of the initialization turns out to be very important in order to get meaningful results. A good initialization, in fact, can increase the speed and accuracy of the algorithms, as it can produce faster convergence to an improved local minimum (Smilde, Bro and Geladi 2005).

In the standard NMF algorithm the factor matrices are initialized randomly, and their elements are drawn from a uniform distribution defined on the same range of the target matrix's entries. This has the advantage of being very simple, however it has the drawback that multiple runs with different starting points should be performed in order to achieve a decent stability, which significantly increases the computation times.

To overcome this issue, several seeding methods requiring to be performed only once have been proposed. We recall, for example, the ones based on *independent component analysis* (ICA, which is like NMF, but where independence between the columns of $\boldsymbol{W}$ has been assumed; Marchini, Heaton and Ripley 2013), on *nonnegative double singular value decomposition* (particularly suited to initialize NMF algorithms with sparse factors; Boutsidis and Gallopoulos 2008), or on *clustering techniques* (the basis matrix is initialized using the centroids computed with some clustering methods, like k-means, whereas the mixture coefficients matrix is initialized as a proper scaling of the cluster indicator matrix; Wild, Curry and Dougherty 2004, Xue et al. 2008). In practice, one may use several diverse initializations via a Monte Carlo type approach, and keep the best solution obtained, even if this may be prohibitive on large and realistically-sized problems.

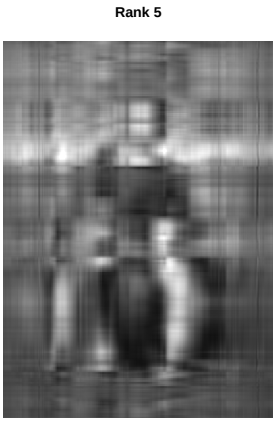
The final decision to be made concerns the choice of the value for the factorization rank r. In particular, when it takes too high values, a potential risk of *overfitting* may occur, since more variables can better fit the data, and therefore, the residuals decrease.

To illustrate the cruciality of such a choice, let us consider the following application to image recognition, specifically to the painting *Madonna del cardellino* (Madonna of the Goldfinch) by the Italian artist Raffaello (the image was converted

Figure 1.2: *The original painting in grey-scale format; target matrix rank = 400*

into a grey-scale format; Figure 1.2). By analyzing the pixel intensities matrix with NMF models having different complexities, a sequence of low-rank matrices (that is, approximated images) was obtained (Figure 1.3). As it can be seen, low values of r do not allow to recognize the image, whereas as the factorization rank increases the image becomes less blurry, and more defined, but a too big value may risk to overfit the image.

Unless one has prior knowledge based on the specific domain theory, one common way of deciding on its value is through trial and error, that is, different values are tried, some quality measures of the results are computed, and the best value according to such measures for the application at hand is chosen. Nevertheless, several approaches have been proposed to further help users find the optimal value of such parameter. For example, applying the SVD technique and looking at the decay of the singular values of the input data matrix; or taking the first value of r for which the cophenetic coefficient starts decreasing (Brunet et al. 2004); or choosing the first value where the residual sum of squares curve presents an inflection point (Hutchins et al. 2008); or yet considering the smallest value at which the decrease in the residual sum of squares is lower than the decrease of the analogous curve obtained from random data (Frigyesi and Höglund 2008).



(a) *Factorization rank=5.*

(b) *Factorization rank=20.*

(c) *Factorization rank=40.*

(d) *Factorization rank=80.*

Figure 1.3: The approximated representations of the original painting with different factorization ranks: 5, 20, 40, 80.

1.4 An ill-posed problem

When proposing the NMF factorization, Lee and Seung did not specify under which assumptions such approximation was unique (i.e., well defined) and when it recovered the correct decomposition. To this end, Donoho and Stodden 2004 pointed out a potentially serious problem with nonnegative matrix factorization and questioned when the above discussed generative model held. They discovered that even in situations where $\boldsymbol{V} = \boldsymbol{W}\boldsymbol{H}$ held exactly, the decomposition might not be unique.

Specifically, they interpreted NMF geometrically as the problem of finding a simplicial convex cone $C_{\boldsymbol{W}} = \left\{ \sum_{a=1}^{r} \lambda_a \boldsymbol{w}_a, \lambda_a \geq 0 \right\}$ which contains the cloud of data points and which is contained in the positive orthant, and they showed that if the data values are strictly positive, i.e., $v_{i\mu} \geq \epsilon \, \forall \epsilon > 0$, for $i = 1, \ldots, n$ and $\mu = 1, \ldots, m$, the column vectors of $\boldsymbol{V}$ are placed well inside the interior of the positive orthant of $\mathbb{R}^n$, and there are many simplicial cones containing the data because nonnegative matrices form a cone with many facets making hard to characterize which and when a facet is active or not in the optimization (Chu et al. 2004). As an illustration of the problem, let us consider the data points in two dimensions depicted in Figure 1.4. Since there is open "space" between the data and the coordinate axes, one can choose the basis vectors anywhere in this open space, and represent each data point exactly with a nonnegative linear combination of these vectors.

It can hence be concluded that under such strict positivity condition, there are many distinct representations of the form $\boldsymbol{V} = \boldsymbol{W}\boldsymbol{H}$ where $\boldsymbol{W} \geq 0$ and $\boldsymbol{H} \geq 0$, which basically means that the solution found by the algorithms is not unique, but depends on starting values. In other words, any minimum solution given by the matrices $\boldsymbol{W}, \boldsymbol{H}$ can be also given by an infinite number of equally good solution pairs. In fact, any nonnegative invertible matrix $\boldsymbol{Q}$ satisfying $\widetilde{\boldsymbol{W}} = \boldsymbol{W}\boldsymbol{Q} \geq 0$ and $\widetilde{\boldsymbol{H}} = \boldsymbol{Q}^{-1}\boldsymbol{H} \geq 0$ generates an equivalent factorization[5] $\widetilde{\boldsymbol{W}}\widetilde{\boldsymbol{H}} = \boldsymbol{W}\boldsymbol{H}$. This issue is addressed by saying that NMF is an ill-posed problem. These complications are usually alleviated by imposing additional constraints to confine the feasible solution set (e.g. smoothness constraints, shape constraints, geometric constraints, cross-

[5] It is worth mentioning, however, that there do exist conditions under which NMF is unique, irrespectively of the employed algorithms, namely if data points "fill out" the positive orthant or a proper subset of it, or, more specifically, if the generative model applies, separability conditions are complied with, and a complete factorial sampling is observed (details in Donoho and Stodden 2004).

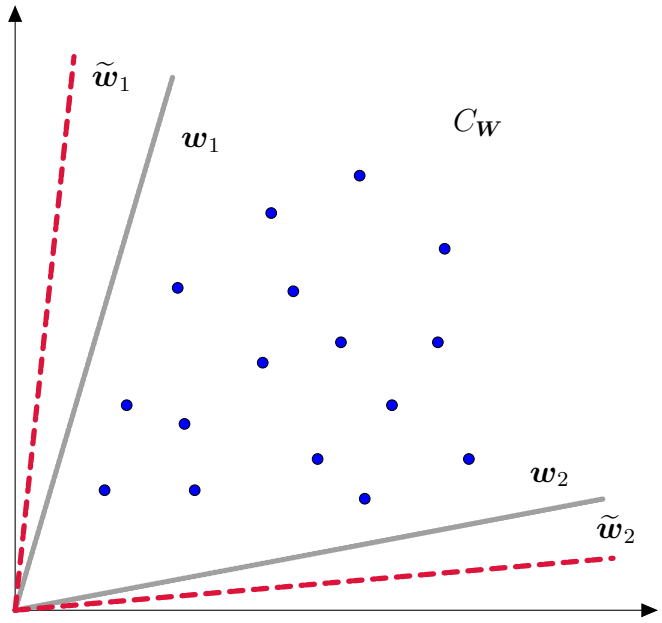


Figure 1.4: Non-uniqueness of nonnegative matrix factorization

modal correspondence constraints).

1.5 Extensions

Over the years, the standard NMF model has been extended from its earliest formulation to include auxiliary constraints on $\boldsymbol{W}$ and/or $\boldsymbol{H}$ explicitly to improve the factorization problem. These additions may be applied for several reasons: to compensate uncertainties in the data, to enforce desired characteristics (like robustness) in the computed solution, or to impose prior knowledge about the application at hand. The auxiliary constraints are typically enforced through penalty terms that are summed to the cost function, and incorporated in the objective function. The most common extensions pertain to the introduction of a smoothness and/or a sparseness term.

Smoothness constraints are often applied to regularize the computed solutions in presence of noise in the data. As far as sparseness is concerned, it is often of interest to get a sparse (i.e., many of the elements of the basis matrix and/or of the coefficients matrix are exactly equal to zero) parts-based representation of the

data because it means that only a few units are effectively used to represent the data points. However, in the original formulation of the NMF model, sparseness appeared more as a side-effect, rather than a goal, and the degree to which a representation was sparse could not be controlled in any way. To this end, Hoyer 2004 derived the *NMF model with sparseness constraints* (known also as AHCLS, i.e., alternating Hoyer-constrained least squares), and showed that explicitly incorporating the notion of sparseness[6] as a penalty term in the objective function definitely improved the found decomposition.

It is worth noticing that other more arduous extensions of standard NMF have been proposed in recent years, among which we recall: the *non-smooth NMF (ns-NMF)* using a modified version of multiplicative updates for Kullback-Leibler divergence to fit a NMF-like model, explicitly meant to give sparser results (Pascual-Montano et al. 2006); the *NMF with offset*, using a modified version of multiplicative updates for Euclidean distance to fit a NMF model including an intercept term (Badea 2008); the *pattern-expression NMF*, using multiplicative rules to minimize an objective function based on Euclidean distance and regularized for effective expression of patterns with basis vectors (Zhang et al. 2008).

[6] The measure used to induce sparseness, called Hoyer's measure, was $\text{sparseness}(\boldsymbol{x}) = \frac{\sqrt{n} - \frac{\|\boldsymbol{x}\|_1}{\|\boldsymbol{x}\|_2}}{\sqrt{n}-1}$.

2

Application to Text Mining

Over the years, nonnegative matrix factorization has been successfully applied to a variety of fields. In this chapter an application to text analysis for topic recovery and document classification is considered. Text analysis is a discipline that applies analytic tools to learn from collections of text data like books, newspapers, emails, social media, and so on.

2.1 Introduction

In text mining, the target matrix is the so called *term-document* matrix, where each column corresponds to a specific document (the set of all documents compose the *corpus*) and each row to a certain word (the set of all words make up the *dictionary*). The generic entry of the matrix is a word frequency, that is, the number of times a certain word appears in a specific document. Thus, each column of $\boldsymbol{V}$ contains the word count for a particular document, whereas each row contains the counts of a particular word in different documents. The data matrix is generally very sparse, since most of the documents use only a small subset of the dictionary. As far as the factor matrices are concerned, the columns of the basis matrix $\boldsymbol{W}$ are the sets of words simultaneously found in different documents, and hence can be interpreted as the (latent) *topics*, whereas the weights in the linear combinations appearing in $\boldsymbol{H}$ quantify the *importance* of each topic in each document, and therefore identify which documents discuss which topics.

Performing nonnegative matrix factorization to this kind of data leads to two primary advantages. The first one is the dimension reduction, since it allows to

project the document corpus onto an r-dimensional semantic space where documents are represented as an additive (rather than a subtractive, as would happen with SVD) linear combination of topics. The second one concerns the clustering of the data elements: whether a document completely belongs to a particular topic, or instead is more or less related to several topics, its cluster membership can be easily determined by finding the base topic for which the document has the largest projection value. But the benefits extend: once a good clustering method has been obtained, computers can automatically organize a document corpus into a meaningful cluster hierarchy, which enables its efficient browsing and navigation. In this way, NMF proposes itself as a valuable alternative to *probabilistic latent semantic analysis* (PLSA; Hofmann 1999) and traditional *information retrieval technologies* aiming at automatically detecting the latent salient semantic structure for a document corpus, and at identifying document clusters in the derived latent semantic space.

2.2 Analysis

The analyzed texts constitute the complete published works by the English poet and playwright William Shakespeare, and were freely downloaded at Project Gutenberg, a digital library collecting thousands of cultural e-books. There are 38 works - divided in comedies, tragedies and history plays - and generally each of them has been split into a series of documents according to the number of acts or another proper subdivision (Figure 2.1). This splitting gave rise to 182 different documents composing the corpus and, thus, the target matrix. Since a single work can be characterized by a variety of underlying topics and features, the *acts*, and not the *works*, were considered the documents of the term-document matrix, so that each subpart could be classified independently from the allocation of other acts within the same work.

A variety of preprocessing operations were necessary to prepare the document corpus for the text analysis. Firstly, special characters, numbers, punctuation and stopwords (i.e., the common words) were removed from the texts, secondly letters were converted to lower case, and finally the document was stemmed (i.e., common word endings, such as “es”, “ed” and “s” were removed). After that, the term-document matrix was constructed. It was composed by 18646 words, 182 documents,

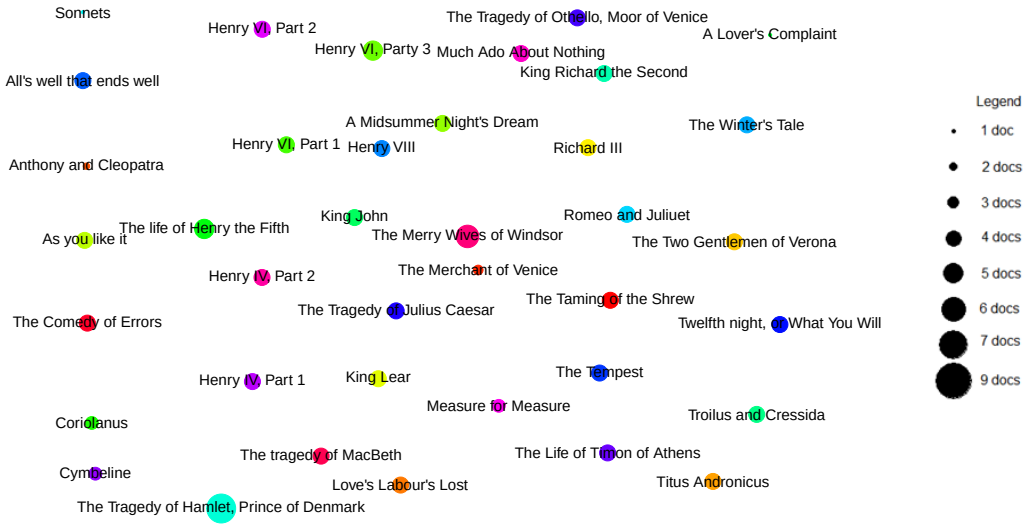


Figure 2.1: The works present in the collection. The size of the points quantifies the number of documents within each work

and the sparseness degree was approximately 95%.

Before applying the NMF method, the target matrix was explored. A common visualization tool in text analysis is the *word cloud* (Figure 2.2) since it gives a quick overview of the frequency of words in the corpus: the larger the font and warmer the color, the more frequent the word.

In the target matrix there were many terms which did not occur often, therefore they were removed from the analysis. The final result consisted in a term-document matrix considering only the 63 most frequent terms[1], which was decomposed by means of nonnegative matrix factorization.

The analysis was conducted using the **R** package **NMF** (Gaujoux and Seoighe 2010; Gaujoux and Seoighe 2015a; Gaujoux and Seoighe 2015b). The first step was the choice of the factorization rank. Supposing not to have any prior knowledge about the main themes discussed by William Shakespeare in his works, the choice of the value of the factorization rank was made through trial and error. Therefore, different values of r (namely, ranging from 2 to 6) were tested. By inspecting the plots of quality measures (Figure 2.3), in particular cophenetic coefficients and residual sum of squares curves, the most appropriate values appeared to be either 3

[1]The original term-document matrix was in fact so big that an immense computational effort would have been necessary to perform the analysis.

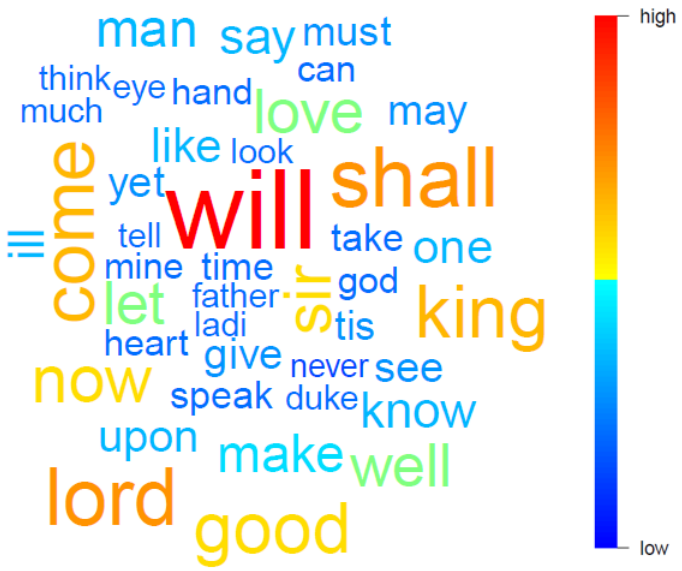


Figure 2.2: Word cloud of the published works by William Shakespeare. Only words occurring more than 1000 times were depicted

or 4. Since the change in the fit statistics was not relevant when moving from rank 3 to rank 4, the decision was to go for a simpler model, and hence the rank equal to three was chosen.

A rank-3 NMF model was fitted to the data using multiplicative update rules minimizing the Kullback-Leibler divergence[2]. To prevent the algorithm from heading towards a local minimum, the model was estimated 200 times (as suggested by Gaujoux 2014), each time starting from a different set of initial values. The results reported below refer to the model showing the best performances. The estimation of the NMF model lead to the estimate of the 63×3 basis matrix $\boldsymbol{W}$ and of the 3×182 mixture coefficients matrix $\boldsymbol{H}$, thus obtaining an approximation of the target matrix.

To explore factor matrices, a common practice in this framework is drawing their heatmaps. The heatmap of $\boldsymbol{W}$ is almost impossible to visualize since it is composed by many rows which conceal the identification of a possible pattern. To this end, since NMF factors are sparse (documents are usually characterized by a small set of salient words), Kim and Park 2007 derived a procedure allowing to extract the relevant words for each basis vector, based on a scoring scheme. By applying such

[2]Other algorithms with different objective functions were fitted, but the results did not change significantly.

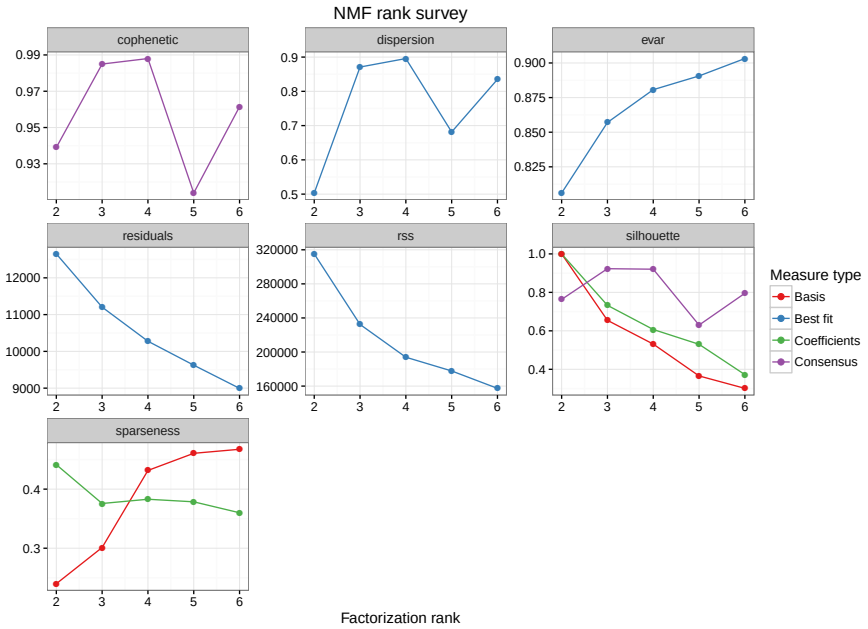


Figure 2.3: Plots of quality measures for values of the factorization rank ranging from 2 to 6

method to the heatmap of the basis matrix, one can obtain a reduced plot where only the most important rows were retained (Figure 2.4).

The columns of the heatmap represent the latent topics of the document corpus. The annotation tracks in the rows and columns define the division into the basis components, whereas the dendrogram on the left-hand side orders the rows by hierarchical clustering using the Euclidean distance and a complete linkage method.

The first column, i.e., the first basis vector, is clearly strongly activated by the words *eye*, *love* and, to a slightly less extent, *live*. This entails to the common topic of **love, beauty**, and **eternity**. The second basis vector is activated by words *lord, first, great, fear* and *day*, with a strong connection also with *live*. These words may stand for a variety of themes: for example, the word *lord* reminds to a sort of **obeisance**, and referential respect towards authorities; the words *great* and *first* give an idea of **grandeur**, magnificence, but also of **power** and influence; and finally the word *day*, together with *live* and *fear*, may entail to a sentiment of **illusiveness of life** and **impending death**. The third basis component is represented only by the word *sir*. It may be that when the writer wanted to address nobles and gentlemen used the word *lord*, whereas when he wished to turn to middle-class men in a more

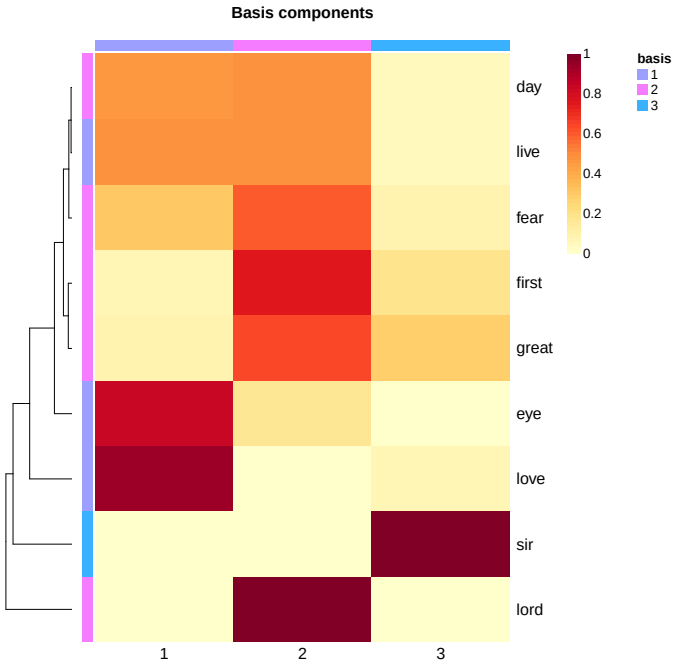


Figure 2.4: Heatmap of the basis matrix where only the relevant rows were selected

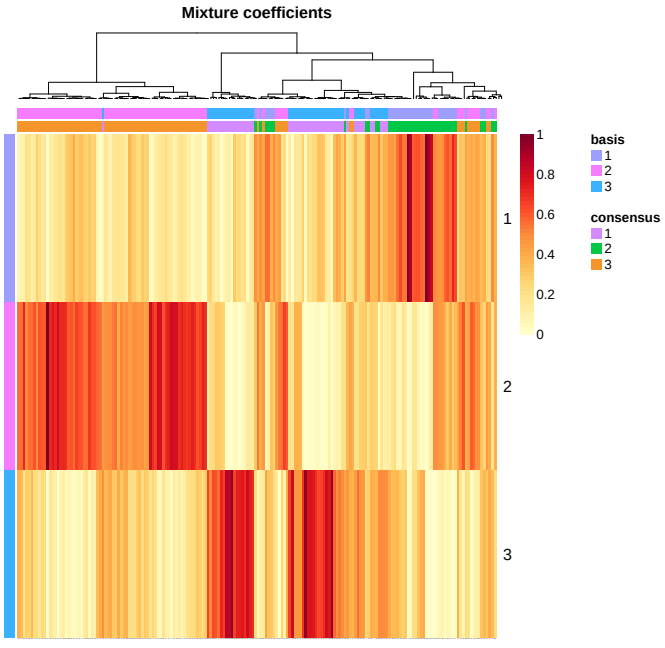


Figure 2.5: Heatmap of the mixture coefficients matrix

Cluster 1 *love, eye, live*	Cluster 2 *lord, first, fear, great, day*	Cluster 3 *sir*
Sonnets	All's Well That Ends Well (4/5)	Anthony and Cleopatra (1/2)
All's Well That Ends Well (1/5)	Anthony and Cleopatra (1/2)	As You Like It (2/5)
As You Like It (3/5)	Coriolanus (2/4)	The Comedy of Errors
The Life of Henry the Fifth (1/6)	Cymbeline	Coriolanus (2/4)
Henry VI, Part 3 (3/5)	The Tragedy of Hamlet (5/9)	The tragedy of Hamlet (4/9)
King John (1/5)	Henry IV, Part 1	Henry IV, Part 2 (3/5)
The Tragedy of Julius Caesar (1/5)	Henry IV, Part 2 (2/5)	King John (1/5)
King Lear (1/5)	The Life of Henry the Fifth (5/6)	King Lear (2/5)
Love's Labour's Lost (4/5)	Henry VI, Part 1	Love's Labour's Lost
A Midsummer Night's Dream (4/5)	Henry VI, Part 2	The tragedy of MacBeth (4/5)
Much Ado About Nothing (3/5)	Henry VI, Part 3 (5/6)	Measure for Measure (2/4)
Romeo and Juliet	Henry VIII	The Merchant of Venice (2/3)
The Tempest (1/5)	King John (1/5)	The Merry Wives of Windsor
Titus Andronicus (2/5)	The Tragedy of Julius Caesar (4/5)	Much Ado About Nothing (2/5)
Troilus and Cressida (1/5)	King Lear (2/5)	The Tragedy of Othello (2/5)
The Two Gentlemen of Verona (4/5)	The Tragedy of Macbeth (4/5)	The Taming of the Shrew
A Lover's Complaint	Measure for Measure (2/4)	The Tempest (3/5)
	The Merchant of Venice (1/3)	Twelfth Night, or What You Will
	A Midsummer Night's Dream (1/3)	The Two Gentlemen of Verona (1/5)
	The Tragedy of Othello (3/5)	The Winter's Tale (1/5)
	King Richard the Second	
	Richard III	
	The Tempest (1/5)	
	The life of Timon of Athens	
	Titus Andronicus (3/5)	
	Troilus and Cressida (4/5)	
	The Winter's Tale (4/5)	

Table 2.1: The clustering of Shakespeare's work provided by a rank-3 NMF model. The numbers in brackets represent how many acts of the same play have been grouped within a specific cluster

conversational and familiar tone the word *sir* was employed. It is also possible that the term *sir* had been used sarcastically. All in all, the possible vein given by this word may be related to comicality, **sarcasm** and **irony**.

Let us consider now the heatmap of the mixture coefficients matrix (Figure 2.5). Each coefficient quantifies the importance of a specific topic on a certain document. Therefore, a clustering of the documents can be obtained by allocating each document to the basis component with respect to which it presents the highest mixture coefficient. In this way, the documents are classified according to the semantic topic they discuss. The obtained partition is listed in Table 2.1. The second cluster contains approximately half of the documents, whereas the third group has a size of 30%, and the first one contains the remaining 20% of the documents.

One can look at which documents have been allocated to which group in order to understand whether NMF performed a good clustering. Among the works falling

in the first cluster we can find the collection of the "Sonnets", the tragedy "Romeo and Juliet", the comedies "Love's Labour's Lost", "A Midsummer Night's Dream" and "The two gentlemen of Verona", the poem "A Lover's Complaint" and others. All these works unquestionably share the same feature, that is they all intensively explore the topic of love in many diverse connotations: be it an overpowering romantic strength in "Romeo and Juliet", or a dream-like enchanted condition in "A Midsummer Night's Dream", or a dark, physical force in the last "Sonnets".

In the second group we find mostly history plays, for example the two tetralogies ("Henry VI, Part 1", "Henry VI, Part 2", "Henry VI, Part 3", "Richard III"; "Richard II", "Henry IV, Part 1", "Henry IV, Part 2", "Henry V"), and tragedies, like "Othello, Moor of Venice", "Hamlet, Prince of Denmark" and "MacBeth". These works have been joined because they shared the same motifs of obeisance, power, grandeur and life-death conflict. It turns out that this splitting is very accurate. The history plays, in fact, illustrate the ascension to the throne and the lives of five generations of Medieval kings, and therefore, are characterized by the recurrent arguments of rulership and military and political control, whereas the story of Macbeth, for instance, gets across the subtle boundary between kingship and tyranny, and warns readers from the dangers caused by ambition, lust for power and greed for wealth. But there is more: some of the plays of this category confront with intense speculations on the complexities of life and death (e.g. Hamlet, *"To be, or not to be, that is the question"*), spirituality and uncertainty. All things considered, nonnegative matrix factorization has exactly picked the right keywords addressing all the main topics of these documents.

The final cluster contains documents which we guessed shared the same comic and sarcastic vein. This finds a confirmation in the results of the partition since we find mostly comedies, like "The Comedy of Errors", "The Merry Wives of Windsor" and "The Taming of the Shrew". These works are characterized by talks full of sarcasm, humour, irony, puns and wordplays that produce a comic effect that entertains and engages the audience.

It should be noticed, however, that for some plays, like "The Tempest", the composing acts were allocated in more than one category. This should not be surprising: although one could try to identify a single recurrent theme within each work, comedies may have dark motifs and tragic situations, as well as tragedies may have some high comic moments, and history plays may contain comedy, tragedy

and everything in between. The truth is that Shakespeare's works have undeniably many facets, and they go over such an extensive variety of motifs that a trivial binary classification of his works would not do justice to the genius and immense talent of this author.

Conclusion

In the era of "Big Data" science and "data analytics" there is the urgent need of correctly processing enormous amount of data by identifying the features of interest in order to make informed decisions. It is very frequent to deal with nonnegative data (for example, in image recognition and text mining), whose analysis cannot be accomplished by means of classical analysis tools since they do not guarantee to preserve nonnegativity.

Starting from such a problem, the nonnegative matrix factorization framework was derived, allowing to brilliantly factorize also nonnegative data. The method provides a low-rank approximation of a possibly huge target matrix, while extracting its meaningful characteristics. Despite its relatively young age, several algorithms optimizing different objective functions were proposed over the years, each of which with strengths and weaknesses. Furthermore, numerous extensions of the standard algorithm were derived to produce improved solutions. Although the constrained factorization problem was shown to be ill-posed, in most practical situations this did not constitute a serious issue.

The empirical application that was presented consisted in the text analysis of the complete works of William Shakespeare. The estimation of the NMF model allowed to select the salient topics from the texts corpus, and thus giving insight on the most recurrent themes and motifs. Moreover, it enabled to efficiently cluster the documents according to the semantic topics they discussed.

In conclusion, this powerful method has made great strides and a huge impact in the last few years, and, by looking at the number of yearly citations received by the founding paper of Lee and Seung 1999 (Figure 2.6), it is very likely that it will continue to have a promising and bright future.

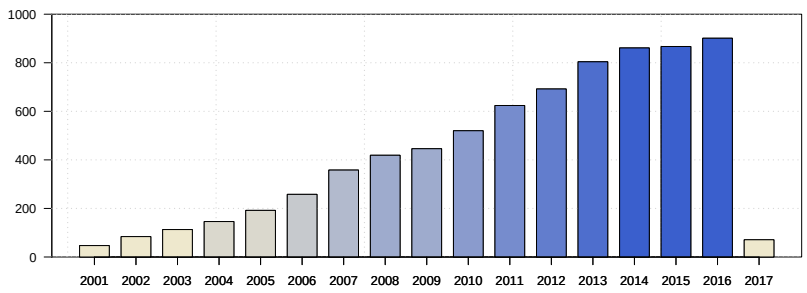


Figure 2.6: Citations per year of Lee and Seung's paper that launched NMF

Bibliography

Arora, Sanjeev et al. (2013). 'A Practical Algorithm for Topic Modeling with Provable Guarantees.' In: *ICML (2)*, pp. 280–288 (cit. on p. 12).

Badea, Liviu (2008). 'Extracting gene expression profiles common to colon and pancreatic adenocarcinoma using simultaneous nonnegative matrix factorization.' In: *Pacific Symposium on Biocomputing*. Vol. 290. 13. Citeseer, pp. 279–290 (cit. on p. 20).

Berry, Michael W et al. (2007). 'Algorithms and applications for approximate nonnegative matrix factorization'. In: *Computational statistics & data analysis* 52.1, pp. 155–173 (cit. on p. 14).

Boutsidis, Christos and Gallopoulos, Efstratios (2008). 'SVD based initialization: A head start for nonnegative matrix factorization'. In: *Pattern Recognition* 41.4, pp. 1350–1362 (cit. on p. 15).

Brunet, Jean-Philippe et al. (2004). 'Metagenes and molecular pattern discovery using matrix factorization'. In: *Proceedings of the national academy of sciences* 101.12, pp. 4164–4169 (cit. on p. 16).

Cauchy, Augustin (1847). 'Méthode générale pour la résolution des systemes d'équations simultanées'. In: *Comp. Rend. Sci. Paris* 25.1847, pp. 536–538 (cit. on p. 12).

Chu, Moody et al. (2004). 'Optimality, computation, and interpretation of nonnegative matrix factorizations'. In: *SIAM Journal on Matrix Analysis*. Citeseer (cit. on pp. 14, 18).

Cichocki, Andrzej, Zdunek, Rafal and Amari, Shun-ichi (2007). 'Hierarchical ALS algorithms for nonnegative matrix and 3D tensor factorization'. In: *International Conference on Independent Component Analysis and Signal Separation*. Springer, pp. 169–176 (cit. on p. 14).

Cichocki, Andrzej, Zdunek, Rafal, Phan, Anh Huy et al. (2009). *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons (cit. on p. 11).

Dempster, Arthur P, Laird, Nan M and Rubin, Donald B (1977). 'Maximum likelihood from incomplete data via the EM algorithm'. In: *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38.

Donoho, David L and Stodden, Victoria C (2004). 'When does non-negative matrix factorization give a correct decomposition into parts?' In: *Advances in neural information processing systems 16: proceedings of the 2003 conference*. Ed. by S Thrun, L Saul and B Schölkopf (cit. on p. 18).

Finesso, Lorenzo and Spreij, Peter (2006). 'Nonnegative matrix factorization and I-divergence alternating minimization'. In: *Linear Algebra and its Applications* 416.2-3, pp. 270–287 (cit. on p. 14).

Friedman, Jerome, Hastie, Trevor and Tibshirani, Robert (2001). *The elements of statistical learning*. Vol. 1. Springer series in statistics Springer, Berlin.

Frigyesi, Attila and Höglund, Mattias (2008). 'Non-negative matrix factorization for the analysis of complex gene expression data: identification of clinically relevant tumor subtypes'. In: *Cancer informatics* 6 (cit. on p. 16).

Gaujoux, Renaud (2014). *An introduction to NMF package* (cit. on p. 24).

Gaujoux, Renaud and Seoighe, Cathal (2010). 'A flexible R package for nonnegative matrix factorization'. In: *BMC Bioinformatics* 11.1, p. 367. ISSN: 1471-2105. DOI: 10.1186/1471-2105-11-367. URL: http://www.biomedcentral.com/1471-2105/11/367 (cit. on p. 23).

— (2015a). *The package NMF: manual pages*. R package version 0.20.6. CRAN. URL: http://cran.r-project.org/package=NMF (cit. on p. 23).

— (2015b). *Using the package NMF*. R package version 0.20.6. CRAN. URL: http://cran.r-project.org/package=NMF (cit. on p. 23).

Gersho, Allen and Gray, Robert M (1992). 'Vector Quantization I: Structure and Performance'. In: *Vector quantization and signal compression*. Springer, pp. 309–343.

Hebb, Donald Olding (2005). *The organization of behavior: A neuropsychological theory*. Psychology Press.

Hestenes, Magnus Rudolph and Stiefel, Eduard (1952). *Methods of conjugate gradients for solving linear systems*. Vol. 49. NBS (cit. on p. 12).

Hinton, Geoffrey E et al. (1995). 'The "wake-sleep" algorithm for unsupervised neural networks'. In: *Science* 268.5214, pp. 1158–1161 (cit. on p. 7).

Hofmann, Thomas (1999). 'Probabilistic latent semantic indexing'. In: *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pp. 50–57 (cit. on p. 22).

Hoyer, Patrik O (2004). 'Non-negative matrix factorization with sparseness constraints'. In: *Journal of machine learning research* 5.Nov, pp. 1457–1469 (cit. on p. 20).

Hutchins, Lucie N et al. (2008). 'Position-dependent motif characterization using non-negative matrix factorization'. In: *Bioinformatics* 24.23, pp. 2684–2690 (cit. on p. 16).

Jeter, MW and Pye, WC (1981). 'A note on nonnegative rank factorizations'. In: *Linear Algebra and its Applications* 38, pp. 171–173 (cit. on p. 5).

Jolliffe, Ian (2002). *Principal component analysis*. Wiley Online Library.

Kim, Hyunsoo and Park, Haesun (2007). 'Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis'. In: *Bioinformatics* 23.12, pp. 1495–1502 (cit. on pp. 12, 24).

Kullback, Solomon (1997). *Information theory and statistics*. Courier Corporation (cit. on p. 11).

Kullback, Solomon and Leibler, Richard A (1951). 'On information and sufficiency'. In: *The annals of mathematical statistics* 22.1, pp. 79–86 (cit. on p. 11).

Langville, Amy N et al. (2014). 'Algorithms, initializations, and convergence for the nonnegative matrix factorization'. In: *arXiv preprint arXiv:1407.7299*.

Lee, Daniel D and Seung, H Sebastian (1997). 'Unsupervised learning by convex and conic coding'. In: *Advances in neural information processing systems*, pp. 515–521 (cit. on p. 10).

— (1999). 'Learning the parts of objects by non-negative matrix factorization'. In: *Nature* 401.6755, pp. 788–791 (cit. on pp. 6, 9, 11, 30).

— (2001). 'Algorithms for non-negative matrix factorization'. In: *Advances in neural information processing systems*, pp. 556–562 (cit. on p. 12).

Marchini, JL, Heaton, C and Ripley, BD (2013). 'fastICA: FastICA algorithms to perform ICA and Projection Pursuit'. In: *R package version*, pp. 1–2 (cit. on p. 15).

McCulloch, Warren S and Pitts, Walter (1943). 'A logical calculus of the ideas immanent in nervous activity'. In: *The bulletin of mathematical biophysics* 5.4, pp. 115–133.

Nakayama, K and Shimojo, S (1992). 'Experiencing and perceiving visual surfaces'. In: *Science* 257.5075, pp. 1357–1363 (cit. on p. 7).

Paatero, Pentti (1997). 'Least squares formulation of robust non-negative factor analysis'. In: *Chemometrics and intelligent laboratory systems* 37.1, pp. 23–35 (cit. on p. 10).

Paatero, Pentti and Tapper, Unto (1994). 'Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values'. In: *Environmetrics* 5.2, pp. 111–126 (cit. on p. 5).

Pascual-Montano, Alberto et al. (2006). 'Nonsmooth nonnegative matrix factorization (nsNMF)'. In: *IEEE transactions on pattern analysis and machine intelligence* 28.3, pp. 403–415 (cit. on p. 20).

Press, William H et al. (1989). *Numerical recipes*. Vol. 3. Cambridge University Press, Cambridge, England (cit. on p. 5).

Salton, Gerard and McGill, Michael J (1986). *Introduction to modern information retrieval*. McGraw-Hill, New York.

Saul, Lawrence and Pereira, Fernando (1997). 'Aggregate and mixed-order Markov models for statistical language processing'. In: *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pp. 81–89.

Smilde, Age, Bro, Rasmus and Geladi, Paul (2005). *Multi-way analysis: applications in the chemical sciences*. John Wiley & Sons (cit. on p. 15).

Straeter, Terry Anthony (1971). 'On the extension of the Davidon-Broyden class of rank one, quasi-Newton minimization methods to an infinite dimensional Hilbert space with applications to optimal control problems'. PhD thesis. North Carolina State University at Raleigh. (cit. on p. 12).

Vavasis, Stephen A (2009). 'On the complexity of nonnegative matrix factorization'. In: *SIAM Journal on Optimization* 20.3, pp. 1364–1377 (cit. on p. 12).

Wild, Stefan, Curry, James and Dougherty, Anne (2004). 'Improving non-negative matrix factorizations through structured initialization'. In: *Pattern recognition* 37.11, pp. 2217–2232 (cit. on p. 15).

Xu, Wei, Liu, Xin and Gong, Yihong (2003). 'Document clustering based on non-negative matrix factorization'. In: ACM, pp. 267–273.

Xue, Yun et al. (2008). 'Clustering-based initialization for non-negative matrix factorization'. In: *Applied Mathematics and Computation* 205.2, pp. 525–536 (cit. on p. 15).

Zhang, Junying et al. (2008). 'Pattern expression nonnegative matrix factorization: algorithm and applications to blind source separation'. In: *Computational intelligence and neuroscience* 2008 (cit. on p. 20).