Nonnegative Matrix Factorization and Financial Applications

Zélia Cazalet
Research & Development
Lyxor Asset Management, Paris
zelia.cazalet@lyxor.com

Thierry Roncalli
Research & Development
Lyxor Asset Management, Paris
thierry.roncalli@lyxor.com

May 2011

Abstract

Nonnegative matrix factorization (NMF) is a recent tool to analyse multivariate data. It can be compared to other decomposition methods like principal component analysis (PCA) or independent component analysis (ICA). However, NMF differs from them because it requires and imposes the nonnegativity of matrices. In this paper, we use this special feature in order to identify patterns in stock market data. Indeed, we may use NMF to estimate common factors from the dynamics of stock prices. In this perspective, we compare NMF and clustering algorithms to identify endogenous equity sectors.

Keywords: Nonnegative matrix factorization, principal component analysis, clustering, sparsity.

JEL classification: G1, C5.

1 Introduction

NMF is a recent technique which knows success not only in data analysis but also in image and audio processing. It is an alternative approach to decomposition methods like PCA and ICA with the special feature to consider nonnegative matrices. Let A be a nonnegative matrix $m \times p$. We define a NMF decomposition as follows:

$$A \approx BC$$

with B and C two nonnegative matrices with respective dimensions $m \times n$ and $n \times p$. Compared to classic decomposition algorithms, we remark that BC is an approximation of A. There are also different ways to obtain this approximation meaning that B and C are not necessarily unique. Because the dimensions m, n and p may be very large, one of the difficulty of NMF is to derive a numerical algorithm with reasonable time of computation. In 1999, Lee and Seung develop a simple algorithm with strong performance and apply it to pattern recognition with success. Since this seminal work, this algorithm has been improved and there are today several ways to obtain a nonnegative matrix factorization.

Drakakis *et al.* (2008) apply NMF to analyse financial data. From a machine learning point of view, NMF could be viewed as a procedure to reduce the dimensionality of data. That's why we could consider NMF in different fields: time series denoising, blind source deconvolution, pattern recognition, etc. One of the most interesting application concerns data classification. Indeed, Drakakis *et al.* (2008) show a clustering example with stocks. In this paper, we explore this field of research. The paper is organized as follows. In Section 2, we present different NMF algorithms and compare them to PCA and ICA decomposition. In Section 3, we apply NMF to stocks classification and compare the results with these obtained by clustering methods. Section 4 describes other applications to financial modeling. Finally, we conclude in Section 5.

2 Nonnegative matrix factorization

2.1 Interpretation of the NMF decomposition

We first notice that the decomposition $A \approx BC$ is equivalent to $A^\top \approx C^\top B^\top$. It means that the storage of the data is not important. Rows of A may represent either the observations or the variables, but the interpretation of the B and C matrices depend on the choice of the storage. We remark that:

$$A_{i,j} = \sum_{k=1}^{n} B_{i,k} C_{k,j}$$

Suppose that we consider a variable/observation storage. Therefore, $B_{i,k}$ depend on the variable i whereas $C_{k,j}$ depend on the observation j. In this case, we may interpret B as a matrix of weights. In factor analysis, B is called the *loading matrix* and C is the *factor matrix*. $B_{i,k}$ is then the weight of the factor k for the variable i and $C_{k,j}$ is the value taken by the factor k for the observation j. If we use an observation/variable storage which is the common way to store data in statistics, C becomes the *loading matrix* and B the *factor matrix*.

Remark 1 *In the original work of Lee and Seung, the NMF decomposition is noted $V \approx WH$ with W a matrix of weights meaning that the data are stored in a variable/observation order.*

Let D be a nonnegative matrix such that D^{-1} is nonnegative too. For example, D may be a permutation of a diagonal matrix. In this case, we have:

$$A \approx BD^{-1}DC \approx B^\star C^\star$$

with $B^\star = BD^{-1}$ and $C^\star = DC$. It shows that the decomposition is not unique. Moreover, the decomposition may be rescaled by the matrix $D = \operatorname{diag}\left(B^\top \mathbf{1}\right)$. In this case, $B^\star$ is a nonnegative matrix such that:

$$\sum_{i=1}^{m} B^\star_{i,k} = 1$$

and $B^\star$ is a matrix of weights.

2.2 Some algorithms

In order to find the approximate factorization, we need to define the cost function f which quantify the quality of the factorization. The optimization program is then:

$$\begin{aligned} \left\{\hat{B}, \hat{C}\right\} &= \arg\min f\left(A, BC\right) \\ \text{u.c.} & \left\{ \begin{array}{l} B \geq 0 \\ C \geq 0 \end{array} \right. \end{aligned} \tag{1}$$

Lee and Seung (2001) consider two cost functions. The first one is the Frobenious norm:

$$f(A, BC) = \sum_{i=1}^{m} \sum_{j=1}^{p} \left(A_{i,j} - (BC)_{i,j} \right)^2$$

whereas the second one is Kullback-Leibler divergence:

$$f(A, BC) = \sum_{i=1}^{m} \sum_{j=1}^{p} \left(A_{i,j} \ln \frac{A_{i,j}}{(BC)_{i,j}} - A_{i,j} + (BC)_{i,j} \right)$$

To solve the problem (1), Lee and Seung (2001) propose to use the multiplicative update algorithm. Let $B_{(t)}$ and $C_{(t)}$ be the matrices at iteration t. For the Frobenious norm, we have:

$$\begin{array}{rcl} B_{(t+1)} & = & B_{(t)} \odot \left(A C_{(t)}^{\top} \right) \oslash \left(B_{(t)} C_{(t)} C_{(t)}^{\top} \right) \\ C_{(t+1)} & = & C_{(t)} \odot \left(B_{(t+1)}^{\top} A \right) \oslash \left(B_{(t+1)}^{\top} B_{(t+1)} C_{(t)} \right) \end{array}$$

where $\odot$ and $\oslash$ are respectively the element-wise multiplication and division operators[1]. A similar algorithm may be derived for the Kullback-Leibler divergence. Under some assumption, we may show that $\hat{B} = B_{(\infty)}$ and $\hat{C} = C_{(\infty)}$ meaning that the multiplicative update algorithm converges to the optimal solution.

For large datasets, the computational time to find the optimal solution may be large with the previous algorithm. Since the seminal work of Lee and Seung, a lot of methods have also been proposed to improve the multiplicative update algorithm and speed the converge. Among these methods, we may mention the algorithm developed by Lin (2007). The idea is to to use the alternating nonnegative least squares[2]:

$$\left\{ \begin{array}{rcl} B_{(t+1)} & = & \arg\min f\left(A, BC_{(t)}\right) \\ C_{(t+1)} & = & \arg\min f\left(A, B_{(t+1)} C\right) \end{array} \right. \tag{2}$$

with the constraints $B_{(t+1)} \geq 0$ and $C_{(t+1)} \geq 0$. To solve the previous problem, Lin (2007) uses the projected gradient method for bound-constrained optimization. We first remark that the two optimization problems (2) are symmetric because we may cast the first problem in the form of the second problem:

$$B_{(t+1)}^{\top} = \arg\min f\left(A^{\top}, C_{(t)}^{\top} B^{\top} \right)$$

So, we may only focus on the following optimization problem:

$$\begin{array}{rcl} C^{\star} & = & \arg\min f(A, BC) \\ & \text{u.c.} & C \geq 0 \end{array}$$

[1] To prevent problems with denominators close to zero, Pauca *et al.* (2006) propose to add a small positive number in the two previous denominators.

[2] We notice that the algorithm of Lee and Seung is a special case of this one where the optimality criterion is replaced by a sufficient criterion:

$$\left\{ \begin{array}{rcl} f\left(A, B_{(t+1)} C_{(t)}\right) & \leq & f\left(A, B_{(t)} C_{(t)}\right) \\ f\left(A, B_{(t+1)} C_{(t+1)}\right) & \leq & f\left(A, B_{(t+1)} C_{(t)}\right) \end{array} \right.$$

Let us consider the case the Frobenious norm. We have:

$$\partial_C f(A, BC) = 2B^\top (BC - A)$$

The projected gradient method consists in the following iterating scheme:

$$C \leftarrow C - \alpha \partial_C f(A, BC)$$

with α the length of descent. In place of finding at each iteration the optimal value of α, Lin (2007) proposes to update α in a very simple way depending on the inequality equation:

$$(1-\sigma)\,\partial_C f(A, BC)^\top \left(\tilde{C} - C\right) + \frac{1}{2}\left(\tilde{C} - C\right)^\top \partial_C^2 f(A, BC)\left(\tilde{C} - C\right) \leq 0$$

with $\tilde{C}$ the update of C. If this inequality equation is verified, α is increased whereas we decrease α otherwise.

2.3 Comparing NMF method with other factor decompositions

In order to understand why NMF is different from other factor methods, we consider a simulation study. We consider a basket of 4 financial assets. The asset prices are driven by a multi-dimensional geometric brownian motion. The drift parameter is equal to 5% whereas the diffusion parameter is 20%. The cross-correlation $\rho_{i,j}$ between assets i and j is equal to 20%, but $\rho_{1,2} = 70\%$ and $\rho_{3,4} = 50\%$. In order to preserve the time homogeneity, the data correspond to $x_{i,t} = \ln S_{i,t}$ where $S_{i,t}$ is the price of the asset i at time t. In Figure 1, we report the time series $x_{i,t}$ for the 4 assets and the first factor estimated by NMF and PCA methods. We remark that the NMF factors is not scaled in the same way than the PCA factor. However, the correlation between the first difference is equal to 98.8%.

Figure 1: Estimating the first factor of a basket of financial assets

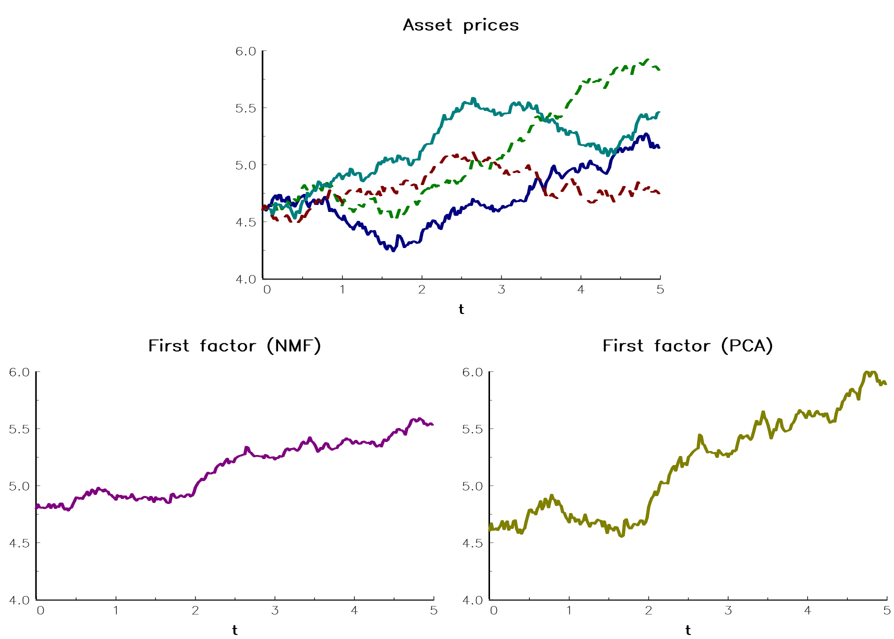


Table 1: Loading matrix of the PCA factors

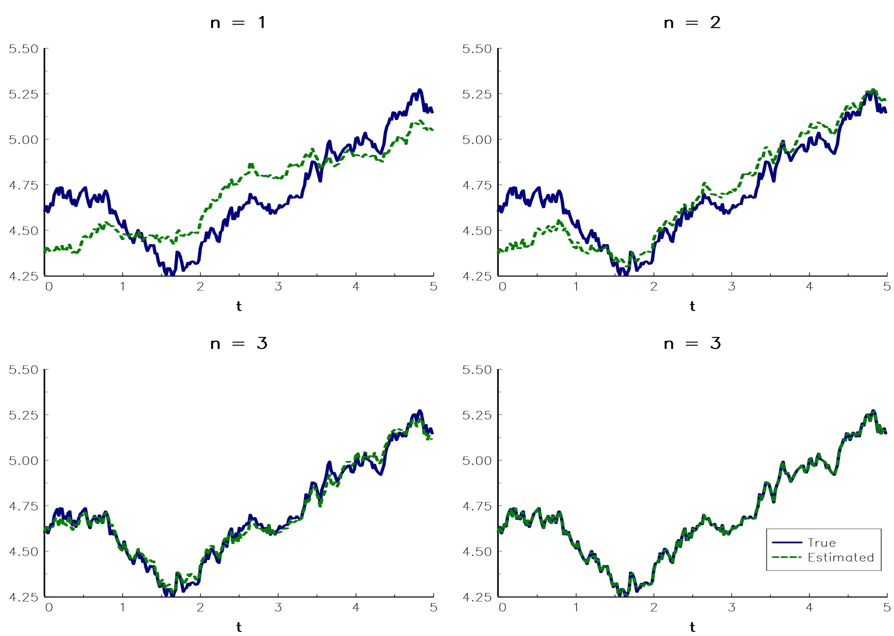
Asset	F_1	F_2	F_3	F_4
#1	0.55	−0.47	0.19	0.66
#2	0.60	−0.35	−0.33	−0.65
#3	0.45	0.51	0.70	−0.22
#4	0.37	0.63	−0.60	0.32

Table 2: Loading matrix of the NMF factors

	$n = 1$	$n = 2$		$n = 3$		
Asset	F_1	F_1	F_2	F_1	F_2	F_3
#1	0.91	0.85	0.68	0.86	0.93	0.29
#2	0.99	0.89	1.00	0.85	1.00	1.00
#3	1.00	1.00	0.18	1.00	0.05	0.69
#4	0.93	0.95	0.03	0.99	0.13	0.13

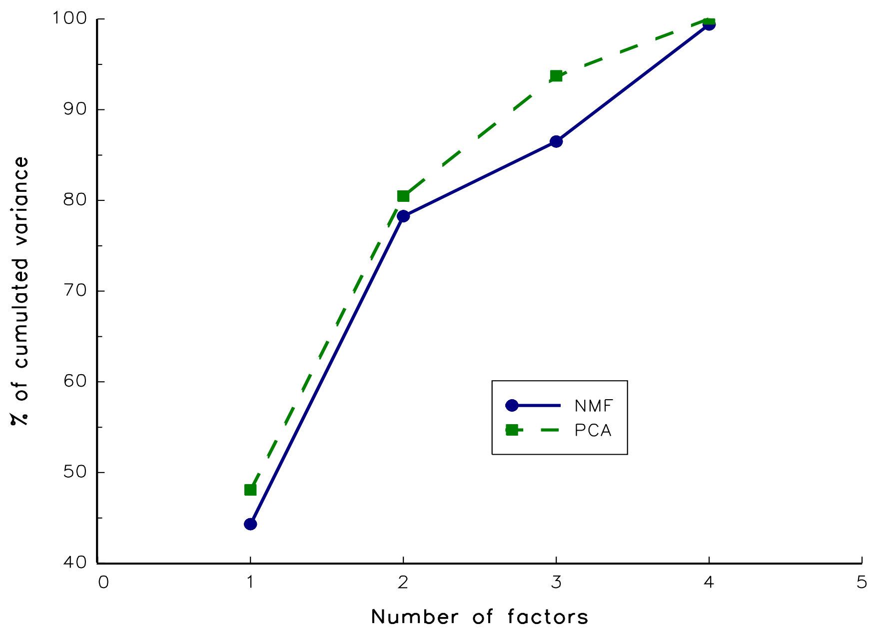
If we compare now the loading matrices of implied factors, we obtain different results (see Tables 1 and 2). The first factor is comparable, which is not the case of the other factors, because all the others factors of the PCA decomposition are long/short factors. By construction, NMF factors are long-only and depend on the choice of the number of factors. Therefore, their interpretation is more difficult. In Figure 2, we compare the dynamic of

Figure 2: Reconstruction of the asset prices



the first asset with the dynamic given by the NMF factors. Another interesting result is the decomposition of the variance according to the factors. In Figure 3, we notice that

Figure 3: Variance explained by each factor



PCA explains more variance than NMF for a given number of factors. We may explain this result easily because NMF may be viewed as a constrained principal component analysis with nonnegative matrices. However, it does not mean that PCA explains more variance than NMF from a marginal point of view. For illustration, the second NMF factor explains more variance than the second PCA factor in Figure 3.

3 Financial applications with NMF

3.1 Factor extraction of an equity universe

In what follows, we consider the EuroStoxx 50 index. Using the composition of this index at the end of 2010, we compute nonnegative matrix factorization on the logarithm of the stock prices. We have represented the first NMF factor in Figure 4. We may compare it with the logarithm of the corresponding index. Even if the decomposition is done just for one date and does not take into account of entry/exit in the index, we notice that the first NMF factor is highly correlated with the index[3]. One interesting thing is the sensibility values of the stocks with respect to this factor[4]. In Table 3, we indicate the stocks corresponding to the 5 smallest and largest sensibility to the NMF factor. We also indicate their corresponding rank and sensibility in the case of PCA. We notice that NMF and PCA results are very different. First, the range of the sensibility is more important for PCA than for NMF. Second, the correlation between NMF and PCA rankings is low. For the first PCA factor, the largest contribution comes from the financial and insurance sectors. This is not the case for NMF.

[3]The correlation between the index returns and the factor returns is equal to 97.2%. This is a little bit higher than those for the PCA, which is equal to 96.3%.

[4]In order to compare results, we normalize the sensibility such that the largest value is equal to 1.

Figure 4: Comparison between the EuroStoxx 50 and the first NMF factor

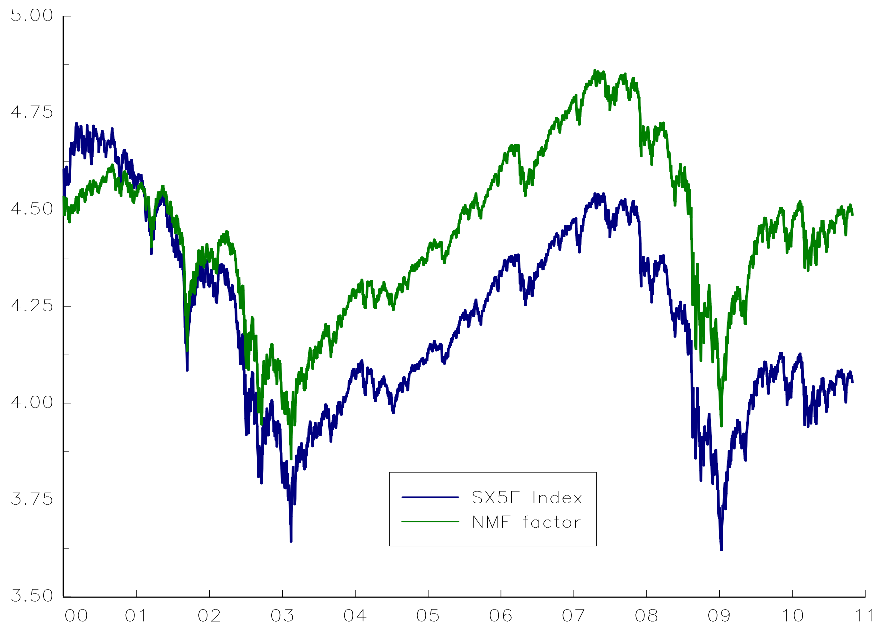


Table 3: Sensibility of the stock prices to the first factor

	NMF		PCA	
Name	Rank	Sensibility	Rank	Sensibility
KONINKLIJKE	1	0.56	12	0.46
DEUTSCHE TELEKOM	2	0.56	23	0.57
FRANCE TELECOM	3	0.57	30	0.62
VIVENDI	4	0.64	31	0.64
NOKIA	5	0.65	38	0.72
IBERDROLA	46	0.92	5	0.39
ENI	47	0.94	10	0.44
VINCI	48	0.99	13	0.46
UNIBAIL-RODAMCO	49	1.00	2	0.31
PERNOD-RICARD	50	1.00	1	0.29

If we consider now two factors, results are those given in Figure 5. In this case, we may interpret them as a factor of bear market and a factor of bull market. Another way to convince us about the nature of these factors is to consider the statistic β_i defined as follows[5]:

$$\beta_i = B^{\star}_{i,2} - B^{\star}_{i,1}$$

We have $\beta_i \in [0, 1]$. A large value of β_i indicates a stock for which the sensibility to the second factor is much more important than the sensibility to the first factor. If we rank the stocks according to this statistic, we verify that high (resp. low) values of β_i are associated to stocks which have outperformed (resp. underperformed) the Eurostoxx 50 index.

In the case when the number n of factors increase, it is more and more difficult to interpret the factors. For example, Figure 6 corresponds to the case $n = 4$. They are some similarities between the third or fourth factor with some sectors (see Table 4). For the first and second factors, we don't find an interpretation in terms of sectors. And the endogenous characteristic of these factors increases when n is large.

Table 4: Correlation of NMF factors with ICB sectors

Sector	0001	1000	2000	3000	4000	5000	6000	7000	8000	9000
#1	−3.5	11.9	−13.6	4.8	−38.5	−16.6	−29.6	−7.3	−31.4	−25.7
#2	0.8	6.4	13.0	−3.1	−24.2	2.0	6.5	−11.3	−13.0	17.0
#3	2.8	52.2	52.0	43.7	70.5	61.5	73.8	45.8	68.7	70.7
#4	4.7	34.0	28.5	14.7	59.9	36.2	38.0	37.9	53.9	24.7

We have the following correspondance between codes and sectors: 0001 = Oil & Gas, 1000 = Basic Materials, 2000 = Industrials, 3000 = Consumer Goods, 4000 = Health Care, 5000 = Consumer Services, 6000 = Telecommunications, 7000 = Utilities, 8000 = Financials, 9000 = Technology.

3.2 Pattern recognition of asset returns

Let $S_{t,i}$ be the price of the asset i at time t. We define the one-period return $R_{t,i}$ as follows:

$$R_{t,i} = \frac{S_{t,i}}{S_{t-1,i}} - 1$$

We may decompose this return as the sum of a positive part and a negative part. We obtain:

$$R_{t,i} = \underbrace{\max\left(R_{t,i}, 0\right)}_{R^{+}_{t,i}} - \underbrace{\max\left(-R_{t,i}, 0\right)}_{R^{-}_{t,i}}$$

Let R be a $T \times N$ matrix containing the return $R_{t,i}$ of the asset i at time t. We define the matrices R^{+} and R^{-} in the same way by replacing the elements $R_{t,i}$ respectively by $R^{+}_{t,i}$ and $R^{-}_{t,i}$. We may apply the nonnegative matrix factorization to the matrices R^{+} and R^{-}. We have:

$$\begin{aligned} R &= R^{+} - R^{-} \\ &= B^{+}C^{+} - B^{-}C^{-} \end{aligned} \tag{3}$$

In this case, the dimension of the C^{+} and C^{-} matrices is $K \times N$. We may interpret the decomposition of R as a projection of T periods to K patterns.

[5] We remind that we have rescaled the B and C matrices with $D = \operatorname{diag}\left(\max\left(B\right)\right)$. In this case, $B^{\star}$ satisfies the property $0 \leq B^{\star}_{i,k} \leq 1$.

Figure 5: NMF with two factors

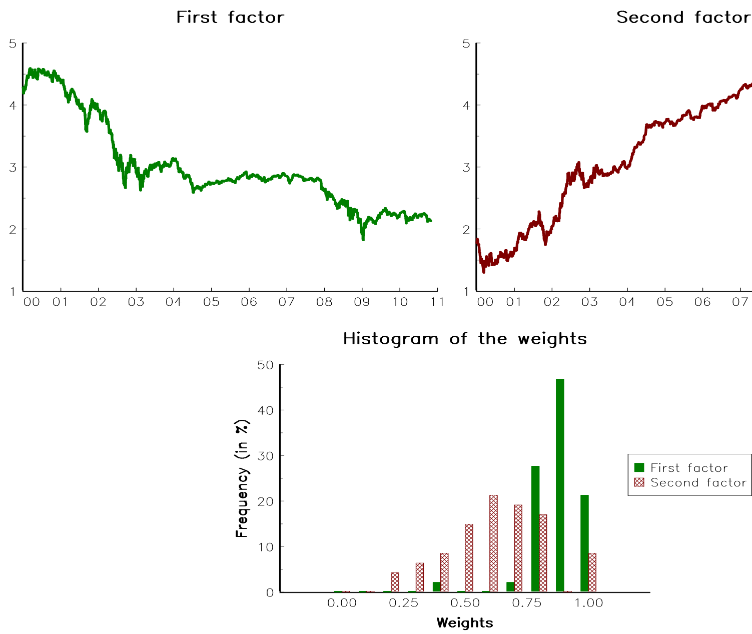


Figure 6: NMF with four factors

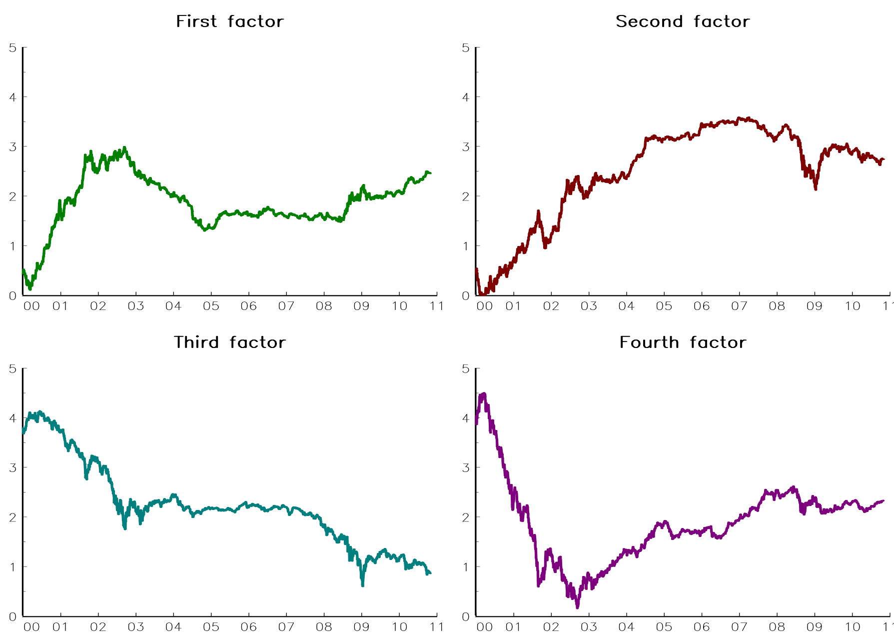
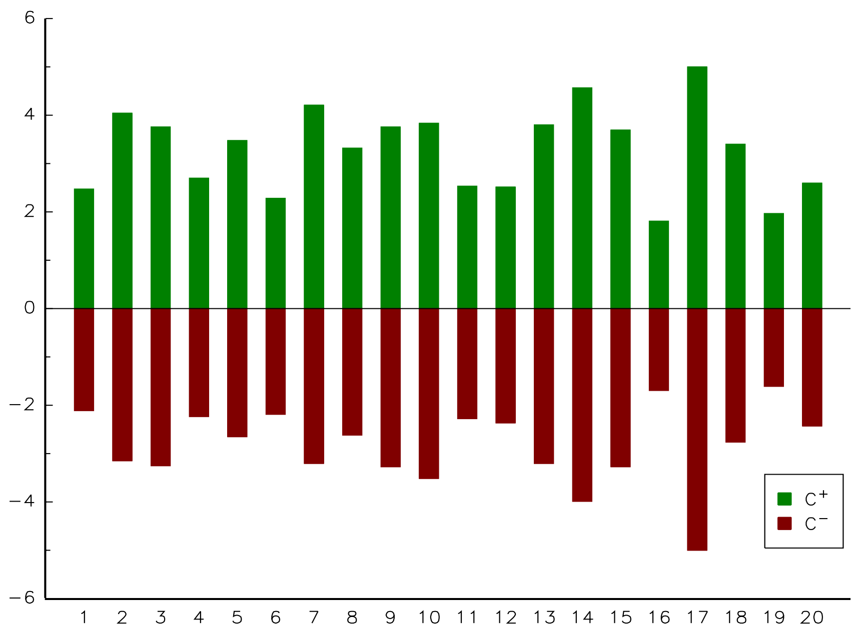


Figure 7: The case of one pattern



Remark 2 *Another way to do pattern recognition is to consider the following factorization:*

$$\begin{pmatrix} R^+ & R^- \end{pmatrix} = BC \tag{4}$$

In this case, C is a $K \times 2N$ matrix. Decompositions (3) and (4) are very similar. But the last one is more restrictive. Indeed, we have:

$$\begin{pmatrix} R^+ & R^- \end{pmatrix} = B \begin{pmatrix} C^+ & C^- \end{pmatrix}$$

This factorization is a restrictive case of the decomposition(3) with $B^+ = B^-$.

In what follows, we consider the weekly returns of 20 stocks[6] by considering the period from January 2000 to December 2010. The dimension of the R matrix is then 574×20 (number of periods T $\times$ number of stocks N). Let us consider the case of one pattern ($K = 1$). In Figure 7, we have reported the C matrix, that is the values of the pattern. Figure 8 corresponds to the B matrix, that is the sensibility of each weekly period to the pattern. If we compute the R^2 statistic associated to this pattern model, we obtain a value closed to 50% (see Figure 9). If we consider more patterns, we may of course obtain better R^2. For example, with 12 patterns, the R^2 statistic is equal to 90%.

To give an example of patterns, we have reported the case $K = 4$ in Figure 10. We notice that all the stocks are not always represented to define a pattern. For example, the second pattern to describe the positive part of returns concerns mainly 5 stocks (Siemens, Telefonica, Allianz, SAP, Deutsche Telekom). With these 4 patterns, the R^2 statistic is equal to 70% for these 20 stocks and the entire period. We remark however that the R^2 statistic differs between stocks (see 5). Indeed, it is equal to 90% for SAP whereas it is equal to 43% for Danone.

[6]Total, Siemens, Banco Santander, Telefonica, BASF, Sanofi, BNP Paribas, Bayer, Daimler, Allianz, ENI, E.ON, SAP, Deutsche Bank, BBVA, Unilever, ING, Schneider, Danone, Deustche Telekom.

Figure 8: Sensibility to the pattern

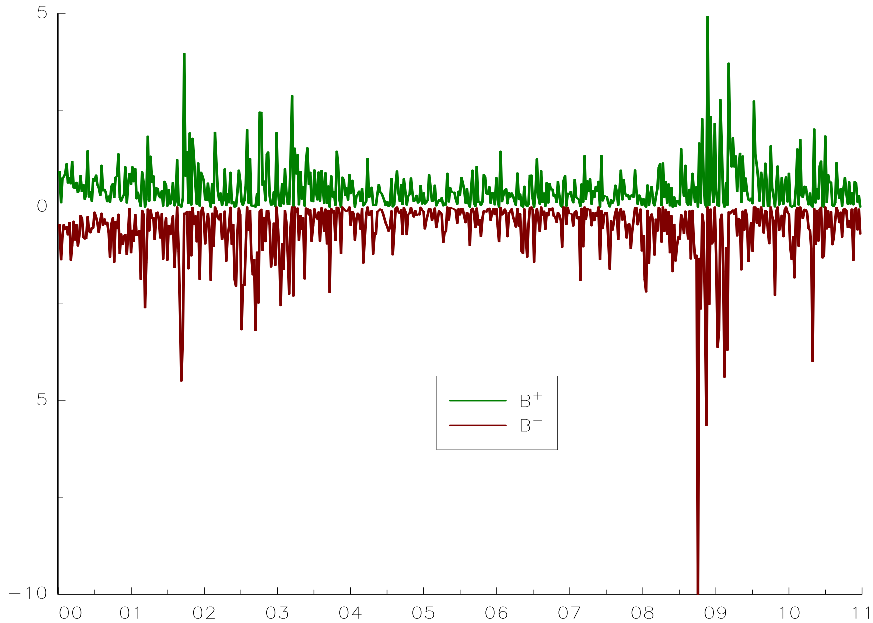


Figure 9: R^2 (in %) of the pattern model

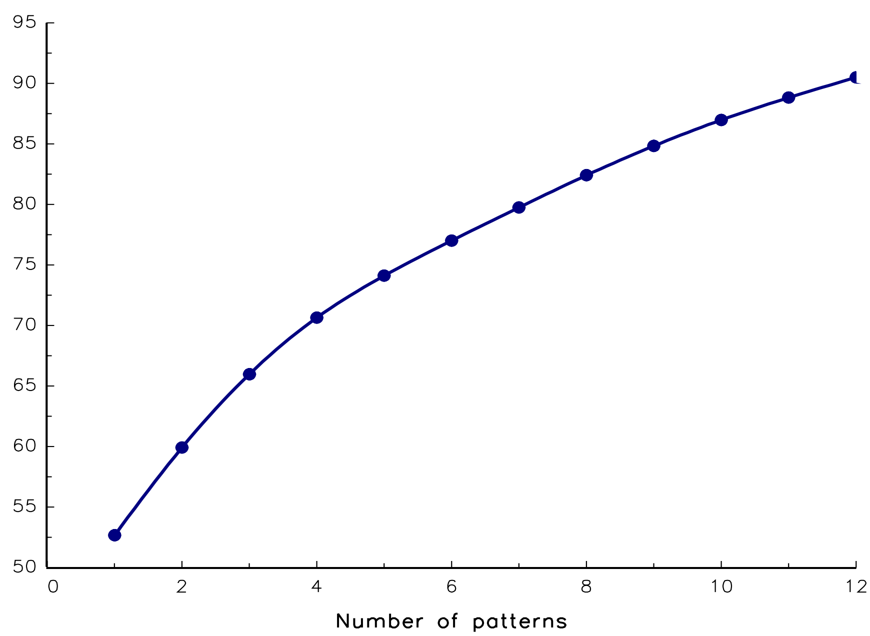


Figure 10: The case of 4 patterns

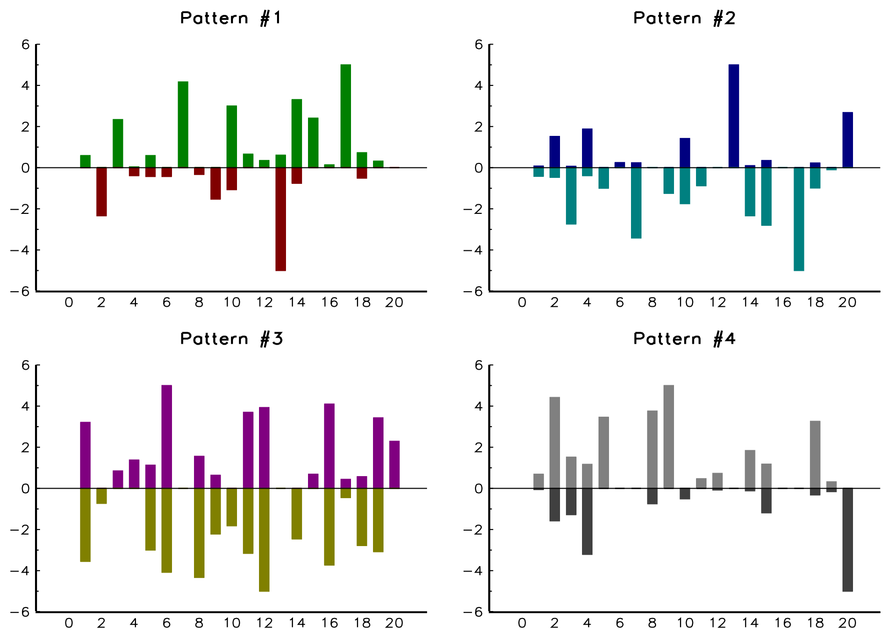


Table 5: R^2 (in %) for each stock

	Number of patterns						
Stock	1	2	3	4	5	10	20
Total	46	46	57	59	59	83	100
Siemens	57	70	70	76	76	79	100
Banco Santander	70	73	75	76	81	90	100
Telefonica	36	54	57	66	64	73	97
BASF	64	63	69	71	74	76	98
Sanofi	27	28	45	54	56	67	100
BNP Paribas	58	69	74	75	74	93	100
Bayer	44	44	54	56	76	97	100
Daimler	57	57	60	69	74	87	100
Allianz	62	63	66	67	68	95	100
ENI	44	48	57	61	62	83	100
E.ON	35	36	58	59	60	65	100
SAP	38	65	76	90	92	99	100
Deutsche Bank	72	75	75	75	75	91	99
BBVA	72	75	77	78	84	91	100
Unilever	20	20	43	51	51	68	99
ING	71	82	84	86	94	97	100
Schneider Electric	49	49	50	52	56	87	100
Danone	26	25	40	43	42	62	100
Deutsche Telekom	22	54	56	77	80	91	98

3.3 Classification of stocks

In what follows, we consider the universe of the 100 largest stocks of the DJ Eurostoxx Index. The study period begins at January 2000 and ends at December 2010. Let $P_{i,t}$ be the price of the stock i for the period t. We consider the matrix A_0 with elements $\ln P_{i,t}$. We perform a NMF of the matrix A_0 with several initialization methods. They are described in Appendix B:

1. RAND corresponds to the random method;
2. NNDSVD uses the algorithm of nonnegative double singular value decomposition proposed by Boutsidis and Gallopoulos (2008);
3. NNDSVD^a and NNDSVD^{ar} are two variants of NNDSVD where the zeros are replaced by the average value a of the matrix A or random numbers from the uniform distribution $\mathcal{U}_{[0,a/100]}$;
4. KM is based on the K-means algorithm proposed by Bertin (2009);
5. CRO corresponds to the closeness to rank-one algorithm described by Kim and Choi (2007).

3.3.1 Relationship between NMF factors and sectors

Remark 3 *Let A_0 be the $m_0 \times p$ matrix which represents the initial data. With NMF, we estimate the factorization $A_0 \approx B_0 C_0$ with $\{B_0, C_0\} = \arg\min f(A_0, BC)$ under the constraints $B \geq 0$ and $C \geq 0$. Suppose now that the $m_1 \times p$ matrix A_1 represents other data. We may estimate a new NMF and obtain $A_1 \approx B_1 C_1$. In some application, we would like to have the same factors for A_0 and A_1, that is $C_1 = C_0$. In this case, the second optimization program becomes then $B_1 = \arg\min f(A_1, BC_0)$ u.c. $B \geq 0$. For the Frobenious norm, this program may be easily solved with quadratic programming. The optimal solution is $B_1 = \begin{bmatrix} \beta_1 & \cdots & \beta_{m_1} \end{bmatrix}^\top$ with:*

$$\begin{aligned} \beta_i &= \arg\min \frac{1}{2} \beta^\top \left(C_0 C_0^\top \right) \beta - \beta^\top \left(C_0 A_1^\top \mathbf{e}_i \right) \\ & \text{u.c.} \quad \beta \geq 0 \end{aligned}$$

Of course, this optimal solution is valid even if the matrix C_0 is not given by NMF, but corresponds to exogenous factors.

One may wonder if the NMF factors are linked to sectors. We use the ICB classification with 10 sectors. Let $I_{j,t}$ be the value of the j^{th} sector index. A first idea is to apply NMF to the matrix A_1 with elements $\ln I_{i,t}$, but by imposing that the matrix of factors C corresponds to the matrix C_0 estimated with the stocks. Using the results of the previous remark, the estimation of the matrix B_1 with $B_1 \geq 0$ is straightforward. If we have a one-to-one correspondance between NMF factors and ICB sectors, B_1 must be a permutation of a diagonal matrix. Results are reported in Appendix C.1 with $n = 10$ factors. We notice that results depend on the initialization method. By construction, two methods (NNDSVD and CRO) produce sparse factorization. For the four others, the factorization is very dense. Except for some very specific cases, a sector is generally linked to several factors.

Another way to see if sectors could be related to NMF factors is to consider the C_2 matrix with elements $\ln I_{i,t}$. In this case, we impose exogenous factors and we could use the

Table 6: Frequencies of the p_i statistic

p_i	Frequency
0%	39
$]0, 10\%]$	4
$]10\%, 20\%]$	7
$]20\%, 30\%]$	4
$]30\%, 40\%]$	9
$]40\%, 50\%]$	7
$]50\%, 60\%]$	5
$]60\%, 70\%]$	7
$]70\%, 80\%]$	3
$]80\%, 90\%]$	4
$]90\%, 100\%[$	2
100%	9

previous technique to estimate the B_2 matrix such that $A_0 = B_2 C_2$ and $B_2 \geq 0$. If sectors are the natural factors, we must verify that $(B_2)_{i,j} > 0$ if the stock i belongs to the sector j and $(B_2)_{i,j} \simeq 0$ otherwise. Let $\mathcal{S}(i) = j$ be the mapping function between stocks and sectors. For each stock i, we may compute the proportion of factor weights explained by the sector $\mathcal{S}(i)$ with respect to the sum of weights:

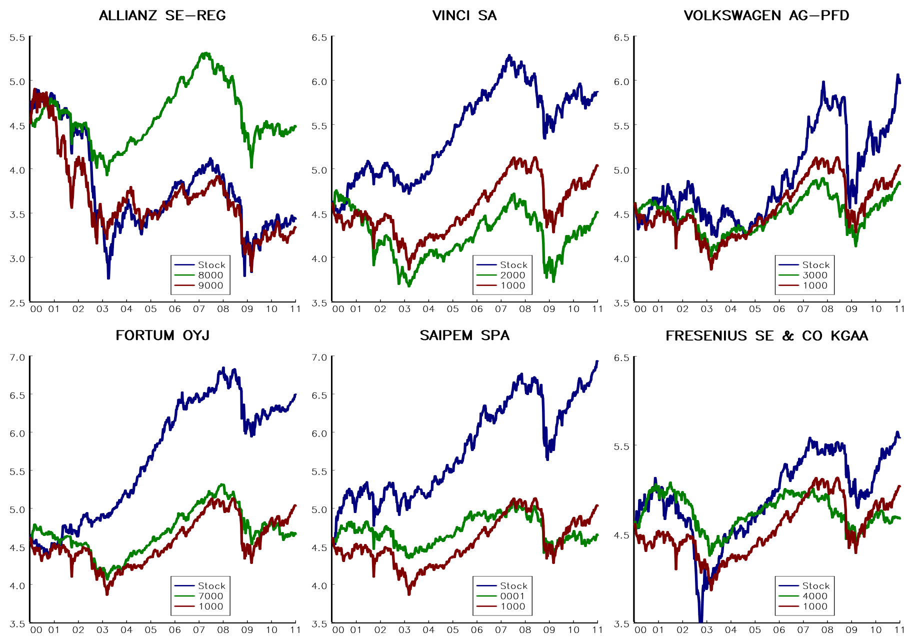
$$p_i = \frac{(B_2)_{i,\mathcal{S}(i)}}{\sum_{j=1}^{n} (B_2)_{i,j}}$$

If $p_i = 100\%$, it means that the sector $\mathcal{S}(i)$ of the stock explains all the factor component. If $p_i = 0\%$, it means that the sector $\mathcal{S}(i)$ of the stock explains nothing. In Table 6, we have reported the distribution of the p_i statistic. We notice that, among the 100 stocks, only 11 stocks have a statistic larger than 90%. These stocks are TOTAL (Oil & Gas), BASF (Basic Materials), SANOFI (Health Care), ARCELORMITTAL (Basic Materials), LINDE (Basic Materials), NOKIA (Technology), KONINKLIJKE (Telecommunications), ALCATEL-LUCENT (Technology), K+S (Basic Materials), CAP GEMINI (Technology) and STMICROELECTRONICS (Technology). In the same time, 39 stocks are not explained by their sectors. We have reported the 25 largest stocks with $p_i = 0$ in Table 15 (see Appendix C.2 page 30). In Figure 11, we have reported some of these stocks[7]. For each stock, we also report its corresponding sector (green line) and the “representative" sector which presents the largest weight in the B_2 matrix (red line). Most of the times, we confirm that the behavior of the stock is closer to the behavior of the representative sector than to the behavior of the corresponding sector.

Remark 4 *The previous analysis is done for a long period (11 years). If we consider a shorter period (for example 1 or 2 years), difference between the corresponding sector and the representative sector is more important.*

[7] We remind that the data are the logarithm of the prices $\ln P_{i,t}$. Moreover, we have normalized the prices such that $P_{i,0} = 100$.

Figure 11: Some stocks which have a behavior different of their sector index



3.3.2 NMF classifiers

Before doing a classification based on NMF, we apply the K-means procedure directly on the stocks returns. The results set a benchmark for the future results of NMF classifiers. For that, we consider 10 clusters. In Table 7, we remark that the cluster #1 groups together a large part of stocks with 8 sectors represented. This cluster has certainly no economic signification. It contains the stocks which are difficult to classify in the other clusters.

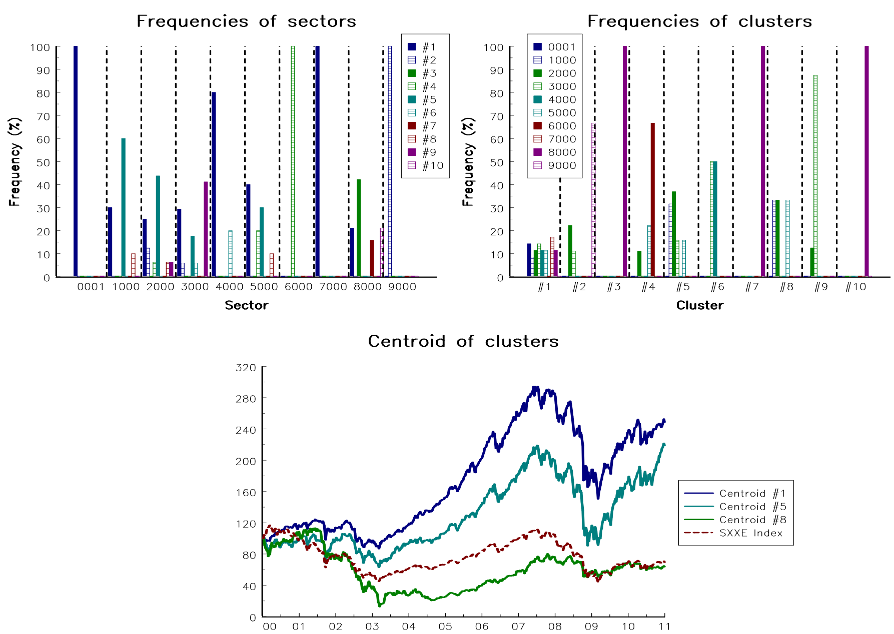
Table 7: Number of stocks by sector/cluster

	cluster										
sector	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	total
0001	5										5
1000	3				6			1			10
2000	4	2		1	7			1	1		16
3000	5	1			3	1			7		17
4000	4					1					5
5000	4			2	3			1			10
6000				6							6
7000	6										6
8000	4		8				3			4	19
9000		6									6
total	35	9	8	9	19	2	3	3	8	4	100

In Figure 12, we report the frequency of each sector in each cluster. We do not notice an exact correspondence between sectors and clusters because several clusters are represented by the same sector. Indeed, this classification highlights the Financials sector which has a frequency of 100% for 3 of the 10 clusters (#3, #7 and #10). In addition, the clusters #2, #4 and #9, respectively represented by Technology, Telecommunication and Consumer Goods sectors, are pointed up. It seems logical that the classification brings to light these sectors because the studied period is strongly influenced by the dot.com and financial bubbles. However, we notice that some clusters, strongly characterized by some sectors, only represent a small part of them. For instance, clusters #7 and #10 represent each one less than 25% of the Financials sector. As a consequence, some clusters represent more a subset of a sector than the average behavior of the sector. Figure 13 confirms this idea. In some cases, the dynamics of centroids differ from the dynamics of the most represented sectors. For instance, the evolution in cluster #10 differs from the global evolution of the Financials sector. Indeed, the centroid owns an evolution flatter than the evolution of the sector. So, the cluster #10 highlights a subset of Financials stocks which has a behavior which is very different than the Financials sector, especially during the burst of the dot.com bubble.

To conclude, this preliminary study shows the heterogeneity into sectors and points up some special stocks whose behaviors differs from the behaviors of their sectors.

Figure 12: Results of the cluster analysis



Let us see now if NMF classifiers can represent an alternative of the sectors classification. Before performing the classification, we have to do two choices: the initialization method and the order n of the factorization. In Figure 14, we see the evolution of the NMF error with respect to the dimension n. We notice that NNDSVD and CRO methods converge more slowly than the others. It is explained by the fact that these methods produce sparse factorization. The other initialization methods provide similar results. However, NNDSDV^{ar} has the lowest euclidean error in most cases. As a result, in what follows, we choose to analyse results with the NNDSDV^{ar} initialization method.

Figure 13: Centroid of some clusters

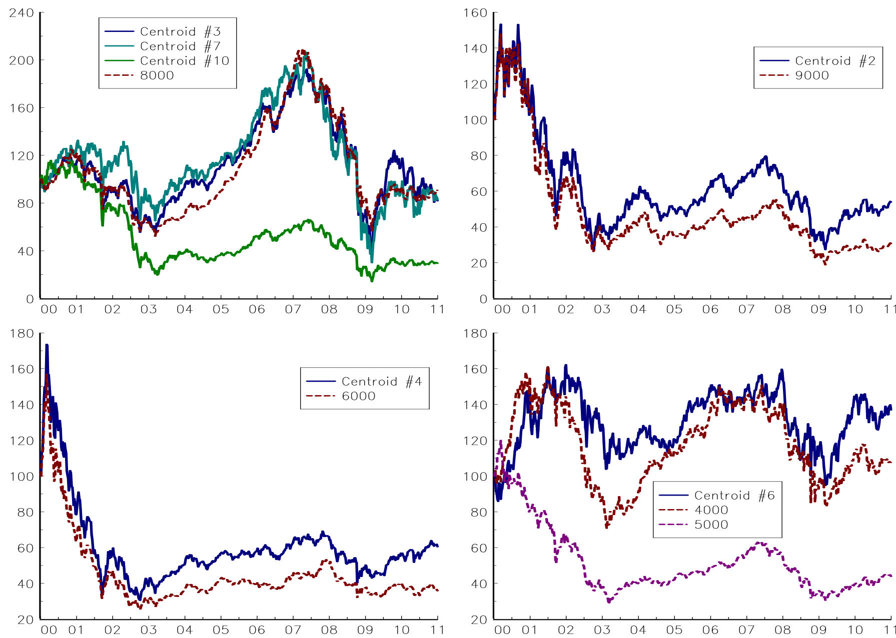
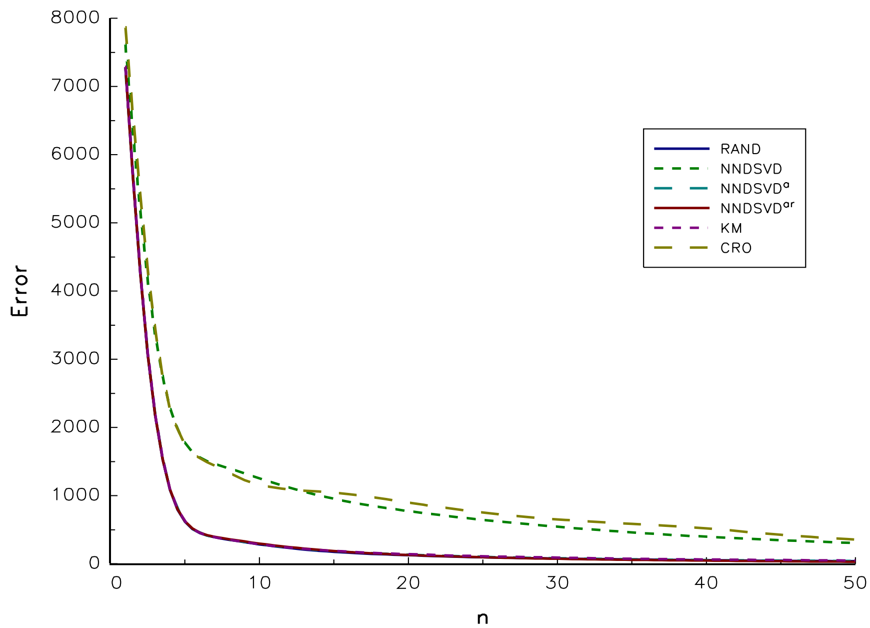
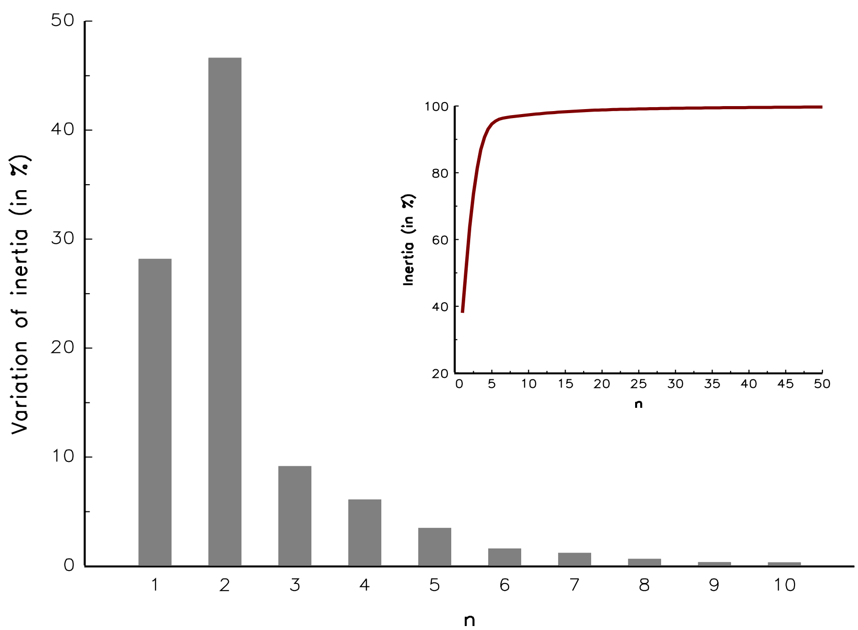


Figure 14: Evolution of the error with respect to initialization method



The next step consists to fix the smallest order n of the factorization which does not lead to serious information loss. For this, we study the inertia retrieved by the NMF method and we select the order which permits to keep more than 90% of the information. According to Figure 15, $n = 4$ is sufficient to retrieve 90% of the information. But $n = 5$ permits to gain 3.47% supplementary information. As a consequence, we select $n = 5$ which represents the number of factors resulting from NMF.

Figure 15: NMF inertia

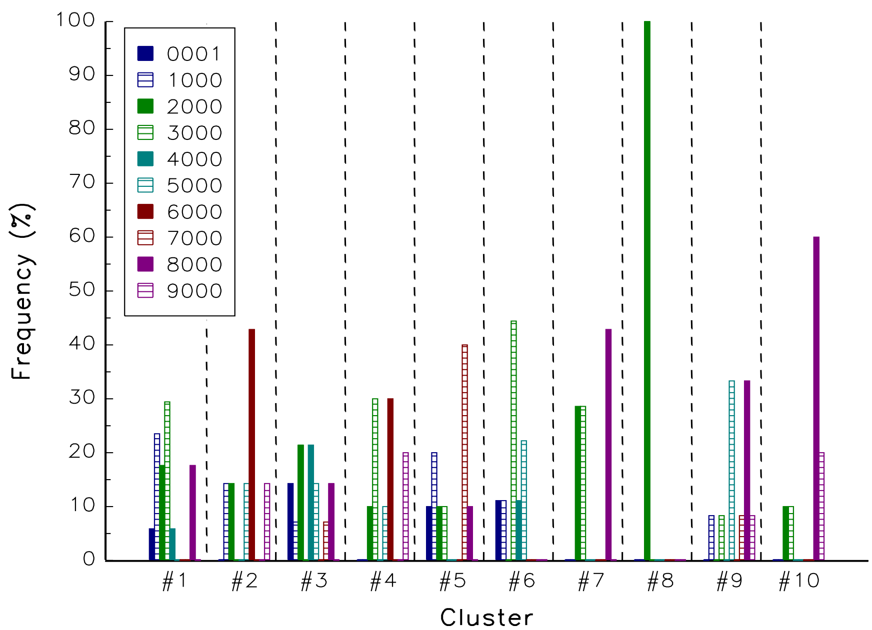


We then proceed to classify stocks by applying an unsupervised algorithm to the normalized matrix $B^{\star}$. The idea is to group stocks that are influenced by the same factors. For this, we consider the K-means algorithm with 10 clusters. According to Table 8, we notice that clusters are well distributed. Contrary to the previous classification, we de not observe one cluster which owns a lot of stocks. Figure 16 indicates that these clusters do not represent sectors. An exception is the cluster #8 which is only represented by the Industrials sector.

Table 8: Number of stocks by sector/cluster

	cluster										
sector	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	total
0001	1		2	0	1	1					5
1000	4	1	1		2	1			1		10
2000	3	1	3	1	1		2	4		1	16
3000	5			3	1	4	2		1	1	17
4000	1		3			1					5
5000		1	2	1		2			4		10
6000		3		3							6
7000			1		4				1		6
8000	3		2		1		3		4	6	19
9000		1		2					1	2	6
total	17	7	14	10	10	9	7	4	12	10	100

Figure 16: Frequencies of sectors in each cluster



In Figure 17, we report the evolution of the centroid of each cluster k. It is computed as follows:

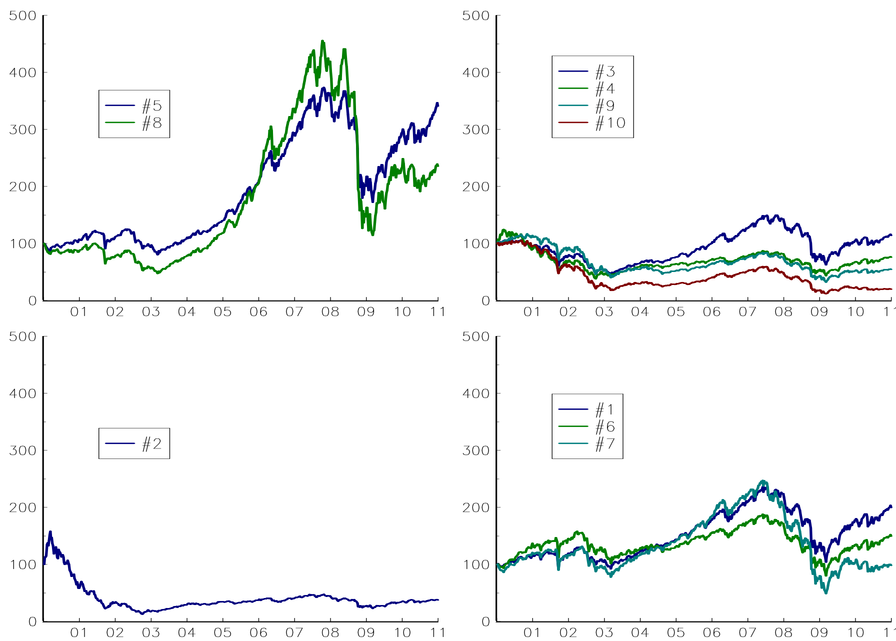
$$\bar{x}_k = w_k^\top C^\star$$

with:

$$w_{k,j} = \frac{1}{N_k} \sum_{i=1}^{m} \mathbf{1}\left\{\mathcal{C}(i) = k\right\} \cdot B_{i,j}^\star$$

where $N_k = \sum_{i=1}^{m} \mathbf{1}\left\{\mathcal{C}(i) = k\right\}$ is the number of stocks in the cluster k.

Figure 17: Centroid of the clusters



We notice that NMF classifiers permit to underline distinct behaviors. Indeed, the centroids of clusters #5 and #8 are strongly influenced by the financial bubble whereas the centroid of cluster #2 is influenced by the dot.com bubble. Stocks which belong to clusters #1, #6 and #7 don't suffer from the dot.com bubble, whereas it is not the case of stocks which belong to clusters #3, #4 #9 and #10. We notice also that the centroids of clusters #1, #6 and #7 are very different at the end of the study period.

To conclude, NMF classifiers are useful for pattern recognition but generally, clusters do not have economic signification. An other drawback is the clusters dependence on the selected initialization procedure for NMF.

4 Conclusion

Knowing its results in image recognition or audio processing, NMF seems to be a useful decomposition method for financial applications. Imposing the non-negativity of matrices permits to consider the factorization as a decomposition with a loading matrix and a matrix of long-only factors. This special feature is interesting to do pattern recognition but the interpretation is more and more difficult with the increasing number of factors. We can also apply a clustering algorithm on the loading matrix in order to group together stocks with same patterns. According to our results, NMF classifiers set apart different behaviors but the economical interpretation of clusters is difficult. A direct classification on stock returns with the K-means procedure seems more robust and highlights some special stocks whose behaviors differs from the behaviors of their sectors.

References

[1] Bertin N. (2009), *Les factorisations en matrices non-négatives : approches contraintes et probabilistes, application à la transcription automatique de musique polyphonique*, Mémoire de thèse, Télécom ParisTech.

[2] Boutsidis C. and Gallopoulos E. (2008), SVD Based Initialization: A Head Start for Nonnegative Matrix Factorization, *Pattern Recognition*, 41(4), pp. 1350-1362.

[3] Drakakis K., Rickard S., de Fréin R. and Cichocki A. (2008), Analysis of Financial Data Using Non-negative Matrix Factorization, *International Mathematical Forum*, 3(38), pp. 1853-1870.

[4] Hastie T., Tibshirani R. and Friedman R. (2009), *The Elements of Statistical Learning*, Second Edition, Springer.

[5] Hoyer P.O. (2004), Non-negative Matrix Factorization with Sparseness Constraints, *Journal of Machine Learning Research*, 5, pp. 1457-1469.

[6] Kim Y-D. and Choi S. (2007), A Method of Initialization for Nonnegative Matrix Factorization, *IEEE International Conference on Acoustics Speech and Signal Processing*, 2(1), pp. 537-540.

[7] Lee D.D. and Seung H.S. (1999), Learning the Parts of Objects by Non-negative Matrix Factorization, *Nature*, 401, pp. 788-791.

[8] Lee D.D. and Seung H.S. (2001), Algorithms for Non-negative Matrix Factorization, *Advances in Neural Information Processing Systems*, 13, pp. 556-562.

[9] Lin C.J. (2007), Projected Gradient Methods for Non-negative Matrix Factorization, *Neural Computation*, 19(10), pp. 2756-2779.

[10] Pauca V.P., Piper J. and Plemmons J (2006), Nonnegative Matrix Factorization for Spectral Data Analysis, *Linear Algebra and its Applications*, 416(1), pp. 29-47.

[11] Sonneveld P., van Kan J.J.I.M., Huang X. and Oosterlee C.W. (2009), Nonnegative Matrix Factorization of a Correlation Matrix, *Linear Algebra and its Applications*, 431(3-4), pp. 334-349.

A Cluster analysis

Cluster analysis is a method for the assignment of observations into groups (or clusters). It is then an exploratory data analysis which allows to group similar observations together. As a result, the objective of clustering methods is to maximize the pairwise proximity between observations of a same cluster and to maximize the dissimilarity between observations which belong to different clusters. In what follows, we are concerned by unsupervised learning algorithms, that is segmentation methods with no information on the groups.

A.1 The K-means algorithm

It is a special case of combinatorial algorithms. This kind of algorithm does not use a probability distribution but works directly on observed data. We consider m objects with n attributes $x_{i,j}$ ($i = 1, \ldots, m$ and $j = 1, \ldots, n$). We would like to build K clusters defined by the index k ($k = 1, \ldots, K$). Let $\mathcal{C}$ be the mapping function which permits to assign an object to a cluster[8]. The principe of combinatorial algorithms is to adjust the mapping function $\mathcal{C}$ in order to minimize the following loss function:

$$\mathcal{L}(\mathcal{C}) = \frac{1}{2} \sum_{k=1}^{K} \sum_{\mathcal{C}(i)=k} \sum_{\mathcal{C}(i')=k} d(x_i, x_{i'})$$

where $d(x_i, x_{i'})$ is the dissimilarity measure between the objects i and i'. As a result, the optimal mapping function is denoted $\mathcal{C}^\star = \arg\min \mathcal{L}(\mathcal{C})$.

In the case of the K-means algorithm, the dissimilarity measure is the Frobenius distance:

$$\begin{aligned} d(x_i, x_{i'}) &= \sum_{j=1}^{n} (x_{i,j} - x_{i',j})^2 \\ &= \|x_i - x_j\|^2 \end{aligned}$$

Therefore, the loss function becomes (Hastie *et al.*, 2009):

$$\mathcal{L}(\mathcal{C}) = \sum_{k=1}^{K} N_k \sum_{\mathcal{C}(i)=k} \|x_i - \bar{x}_k\|^2$$

where $\bar{x}_k = (\bar{x}_{1,k}, ..., \bar{x}_{m,k})$ is the mean vector associated with the k^{th} cluster and $N_k = \sum_{i=1}^{m} \mathbf{1}\{\mathcal{C}(i) = k\}$. Because $m^\star_\Omega = \arg\min \sum_{i\in\Omega} \|x_i - m\|^2$, another from of the previous minimization problem is:

$$\{\mathcal{C}^\star, m_1^\star, \ldots, m_K^\star\} = \arg\min \sum_{k=1}^{K} N_k \sum_{\mathcal{C}(i)=k} \|x_i - m_k\|^2$$

This problem is solved by iterations. At the iteration s, we compute the optimal means of the clusters $\left\{m_1^{(s)}, \ldots, m_K^{(s)}\right\}$ for the given mapping function $\mathcal{C}^{(s-1)}$. Then, we update the mapping function using the following rule:

$$k = \mathcal{C}^{(s)}(i) = \arg\min \left\|x_i - m_k^{(s)}\right\|^2$$

We repeat these two steps until the convergence of the algorithm $\mathcal{C}^\star = \mathcal{C}^{(s)} = \mathcal{C}^{(s-1)}$.

[8] For example, $\mathcal{C}(i) = k$ assigns the i^{th} observation to the k^{th} cluster.

A.2 Hierarchical clustering

This algorithm creates a hierarchy of clusters which may be represented in a tree structure. Unlike the K-means algorithm, this method depends on neither the number of clusters nor the starting configuration assignment. The lowest assignment is the individual objects whereas the highest corresponds to one cluster containing all the objects. We generally distinguish two methods:

- in the *agglomerative* method, the algorithm starts with the individual clusters and recursively merge the closest pair of clusters into one single cluster;
- in the *divise* method, the algorithm starts with the single cluster and recursively split the cluster into two new clusters which presents the maximum dissimilarity.

In this study, we only consider the agglomerative method.

Let k and k' be two clusters. We define the dissimilarity measure $d(k,k')$ as a linkage function of pairwise dissimilarities $d(x_i,x_{i'})$ where $\mathcal{C}(i)=k$ and $\mathcal{C}(i')=k'$:

$$d(k,k') = \ell\left(\left\{d(x_i,x_{i'}), \mathcal{C}(i)=k\right\}, \mathcal{C}(i')=k'\right)$$

There exists different ways to define the function ℓ:

- *Single linkage*
$$d(k,k') = \min_{x_i\in k, x_{i'}\in k'} d(x_i,x_{i'})$$
- *Complete linkage*
$$d(k,k') = \max_{x_i\in k, x_{i'}\in k'} d(x_i,x_{i'})$$
- *Average linkage*
$$d(k,k') = \frac{1}{N_k N_{k'}} \sum_{x_i\in k} \sum_{x_{i'}\in k'} d(x_i,x_{i'})$$

At each iteration, we search the clusters k and k' which minimize the dissimilarity measure and we merge them into one single cluster. When we have merged all the objects into only one cluster, we obtain a tree which is called a dendrogram. It is also easy to perform a segmentation by considering a particular level of the tree. In Figure 18, we report a dendrogram on simulated data using the single linkage rule. We consider 20 objects divided into two groups. The attributes of the first (resp. second) one correspond to simulated Gaussian variates with a mean 20% (resp. 30%) and a standard deviation 5% (resp. 10%). The intra-group cross-correlation is set to 75% whereas the inter-group correlation is equal to 0. We obtain very good results. In practice, hierarchical clustering may produce concentrated segmentation as illustrated in Figure 19. We use the same simulated data as previously except that the standard deviation for the second group is set to 25%. In this case, if we would like to consider two clusters, we obtain a cluster with 19 elements and another cluster with only one element (the 18^{th} object).

B Initialization methods for NMF algorithms

We have to use starting matrices $B_{(0)}$ and $C_{(0)}$ to initialize NMF algorithms. In this section, we present the most popular methods to define them.

Figure 18: An example of dendrogram

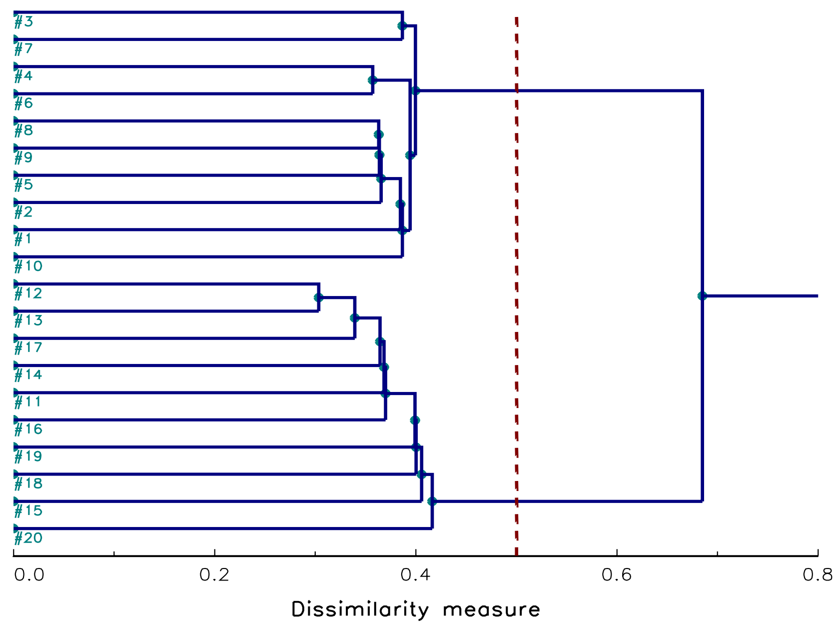
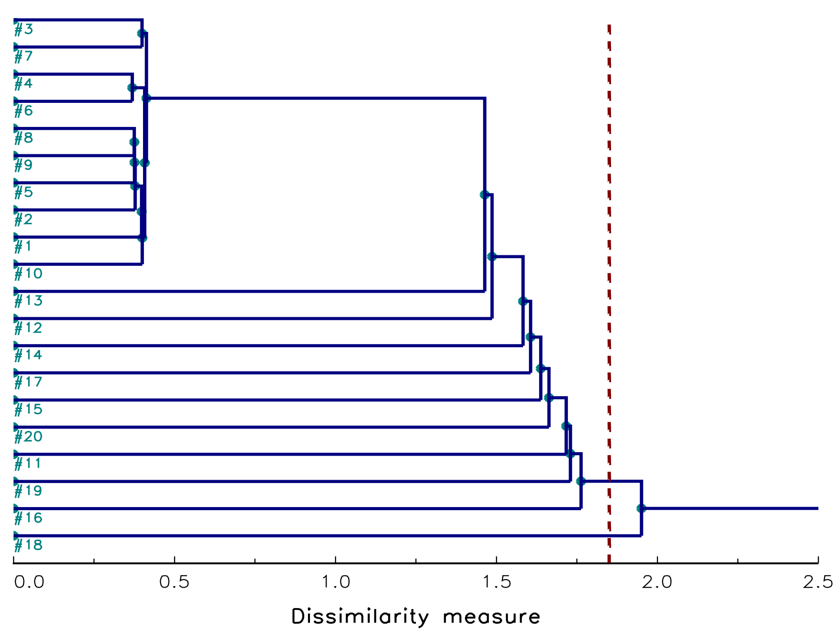


Figure 19: Bad classification



B.1 Random method

The first idea is to consider matrices of positive random numbers. Generally, one use uniform distribution $\mathcal{U}_{[0,1]}$ or the absolute value of Gaussian distribution $|\mathcal{N}(0,1)|$.

B.2 K-means method

Bertin (2009) proposes to apply the K-means algorithm on the matrix $x = A^\top$, meaning that the clustering is done on the columns of the matrix. When the n clusters are determined, we compute the centroid matrix $\bar{x}_{n\times m}$:

$$\bar{x}_{n\times m} = \begin{pmatrix} \bar{x}_1^\top \\ \vdots \\ \bar{x}_n^\top \end{pmatrix}$$

where $\bar{x}_k = (\bar{x}_{k,1}, \ldots, \bar{x}_{k,m})$ is the mean vector associated with the k^{th} cluster. Finally, we have $B_{(0)} = \bar{x}_{n\times m}^\top$and $C_{(0)}$ is a matrix of positive random numbers.

B.3 SVD method

Let A be a $m \times p$ matrix. The singular value decomposition is given by:

$$A = usv^\top$$

where u is a $m \times m$ unitary matrix, s is a $m \times p$ diagonal matrix with nonnegative entries and v is a $p \times p$ unitary matrix. The rank-n approximation is then:

$$A \approx \sum_{k=1}^{n} u_k \sigma_k v_k^\top$$

with u_k and v_k the k^{th} left and right singular vectors and $\sigma_i = s_{i,i}$ the k^{th} largest singular value ($\sigma_1 \geq \ldots \geq \sigma_n$). An idea to initialize the NMF algorithm is to define $B_{(0)}$ and $C_{(0)}$ such that the k^{th} column of $B_{(0)}$ is equal to $u_k\sqrt{\sigma_k}$ and the k^{th} row of $C_{(0)}$ is equal to $\sqrt{\sigma_k}v_k^\top$. However, This method does not work because the vectors u_k and v_k are not necessarily nonnegative except for the largest singular value ($k = 1$) as explained in the following remark.

Remark 5 *We may show that the left singular vectors u of A are the eigenvectors of $AA^\top$, the right singular vectors v of A are the eigenvectors of $A^\top A$ and the non-zero singular values σ_i are the square roots of the non-zero eigenvalues of $AA^\top$. The Perron-Frobenius theorem implies that the first eigenvector of nonnegative irreducible matrices is nonnegative. We deduce that the first eigenvector of the $AA^\top$ or $A^\top A$ matrices is nonnegative if all entries of A are nonnegative. It proves that the singular vectors u_1 and v_1 are nonnegative.*

Boutsidis and Gallopoulos (2008) propose to modify the previous factorization:

$$A \approx \sum_{k=1}^{n} \tilde{u}_k \sigma_k \tilde{v}_k^\top$$

where $\tilde{u}_k$ and $\tilde{v}_k$ are nonnegative vectors. The algorithm is called nonnegative double singular value decomposition (NNDSVD) and is a simple modification of the singular value decomposition by considering only the nonnegative part of the singular values. For each singular value σ_k, we have to decide what is the largest nonnegative part by noticing that:

$$u_k \sigma_k v_k^\top = (-u_k)\,\sigma_k\,(-v_k)^\top$$

Let x^+ be the nonnegative part of the vector x. We define $u_k^- = (-u_k)^+$ and $v_k^- = (-v_k)^+$. It comes that:

$$\tilde{u}_k = \gamma_k \times \begin{cases} u_k^+ / \left\| u_k^+ \right\| & \text{if } m^+ > m^- \\ u_k^- / \left\| u_k^- \right\| & \text{otherwise} \end{cases}$$

and:

$$\tilde{v}_k = \gamma_k \times \begin{cases} v_k^+ / \left\| v_k^+ \right\| & \text{if } m^+ > m^- \\ v_k^- / \left\| v_k^- \right\| & \text{otherwise} \end{cases}$$

where $m^+ = \left\| u_k^+ \right\| \left\| v_k^+ \right\|$ and $m^- = \left\| u_k^- \right\| \left\| v_k^- \right\|$. In order to preserve the inertia, the singular vectors are scaled by the factor $\gamma_k = \max\left(\sqrt{m^+}, \sqrt{m^-}\right)$. The initialization of $B_{(0)}$ and $C_{(0)}$ is then done by using $\tilde{u}_k$ and $\tilde{v}_k$ in place of u_k and v_k.

B.4 CRO method

The CRO-based hierarchical clustering is an alternative method of the classical agglomerative hierarchical clustering when the similarity distance is given by the closeness to rank-one (CRO) measure:

$$\operatorname{cro}(X) = \frac{\sigma_1^2}{\sum_i \sigma_i^2}$$

with $\sigma_1 \geq \sigma_2 \geq \ldots \geq 0$ the singular values of the matrix X. Let k_1 and k_2 be two clusters. The CRO measure between two clusters k_1 and k_2 of the matrix A is defined by $\operatorname{cro}\left(A_{(k_1,k_2)}\right)$ where $A_{(k_1,k_2)}$ is the submatrix of A which contains the row vectors of the two clusters k_1 and k_2. Kim and Choi (2007) summarize the CRO-based hierarchical algorithm as follows. First, we assign each row vector of A into m clusters. Then, we merge the pairs of clusters with the largest CRO measure into one single cluster until n clusters remains.

When the CRO-based hierarchical clustering is applied to the matrix A, we obtain n clusters represented by the submatrices $A_1, \ldots, A_n$. Kim and Choi (2007) consider then the rank-one SVD approximation[9]:

$$\begin{pmatrix} A_1 \\ \vdots \\ A_n \end{pmatrix} \approx \begin{pmatrix} u_1 \sigma_1 v_1^\top \\ \vdots \\ u_n \sigma_n v_n^\top \end{pmatrix}$$

This decomposition leads to a very simple assignment of the matrix $B_{(0)}$ and $C_{(0)}$. The k^{th} column of $B_{(0)}$ corresponds to the vector u_k for the rows which belong the k^{th} cluster and the rest of elements of the k^{th} column are equal to zero (or a small number). The k^{th} row of $C_{(0)}$ corresponds to $\sigma_k v_k^\top$. By construction, this initialization method produces sparse factorization.

Remark 6 *We have proved previously that the singular vector u and v associated to the rank-one SVD approximation of a nonnegative matrix A are nonnegative. From a numerical analysis point of view, the rank-one SVD approximation may be $usv^\top$ or $(-u)\,s\,(-v)^\top$. In practice, we consider also $|u|$ and $|v|$ in place of u and v when working with the rank-one approximation.*

[9] We have $A_k = u_k \sigma_k v_k^\top$ where u_k and v_k are the left and right vectors associated with the largest singular value σ_k.

C Results

C.1 Estimation of the B_1 matrix

Table 9: B_1 matrix with sectors indexes (RAND)

	NMF factors									
Sector	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Oil & Gas	1.00	0.84	0.56	0.69	0.63	0.84	0.82	0.69	0.54	0.79
Basic Materials	0.58	1.00	0.53	0.69	0.64	0.86	1.00	0.49	0.49	1.00
Industrials	0.47	0.83	0.68	0.64	0.71	0.72	0.74	0.60	0.55	0.84
Consumer Goods	0.70	0.89	0.59	0.68	0.81	0.77	0.81	0.67	0.35	0.87
Health Care	0.95	0.95	0.44	0.80	0.71	0.79	0.66	1.00	0.51	0.56
Consumer Services	0.55	0.68	0.66	0.46	0.55	0.65	0.68	0.84	0.41	0.49
Telecommunications	0.69	0.66	1.00	0.30	0.47	0.78	0.43	0.62	0.27	0.49
Utilities	0.97	0.91	0.71	0.72	0.65	1.00	0.83	0.42	0.63	0.65
Financials	0.89	0.76	0.60	1.00	1.00	0.81	0.60	0.40	1.00	0.46
Technology	0.54	0.49	0.96	0.55	0.45	0.45	0.69	0.53	0.57	0.63

Table 10: B_1 matrix with sectors indexes (NNDSVD)

	NMF factors									
Sector	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Oil & Gas	0.99	0.24		0.02	0.21		0.34	0.20		0.36
Basic Materials	0.93	0.21	0.44	0.21	0.01	0.82	0.71	0.13	0.21	1.00
Industrials	0.79	0.52	0.55	0.64	0.31	1.00	0.46	0.34	0.59	0.64
Consumer Goods	0.93	0.26	0.18		0.11	0.86	0.32	0.10	0.24	0.60
Health Care	1.00	0.35		0.02	0.13			0.14		0.41
Consumer Services	0.71	0.68	0.34	0.53	0.38	0.60	0.18	0.33		0.12
Telecommunications	0.59	0.73	1.00	0.59	1.00		0.68	1.00		0.10
Utilities	0.92	0.33	0.64	0.41	0.30		1.00	0.56		0.30
Financials	0.88	0.47	0.49	1.00	0.11		0.50	0.51	0.67	0.43
Technology	0.59	1.00	0.63	0.97	0.65	0.23	0.70	0.73	1.00	0.16

Table 11: B_1 matrix with sectors indexes (NNDSVD^a)

	NMF factors									
Sector	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Oil & Gas	0.83	0.99	1.00	1.00	0.67	0.97	0.29	0.34	0.70	0.07
Basic Materials	0.79	1.00	0.56	0.84	0.50	0.57	0.86	1.00	0.64	0.99
Industrials	0.90	0.75	0.54	0.65	0.55	0.49	0.94	0.91	0.64	0.95
Consumer Goods	0.86	0.86	0.60	0.88	0.45	0.60	0.90	0.87	0.65	0.87
Health Care	0.83	0.90	0.85	0.99	0.66	0.92	0.77	0.30	0.81	0.37
Consumer Services	0.85	0.33	0.54	0.47	0.61	0.69	1.00	0.72	1.00	1.00
Telecommunications	1.00	0.35	0.57	0.42	0.33	0.93	0.69	0.52	0.78	0.71
Utilities	0.80	0.92	0.84	0.77	0.70	1.00	0.46	0.56	0.87	0.74
Financials	0.73	0.94	0.90	0.69	1.00	0.68	0.48	0.48	0.81	0.80
Technology	0.98	0.39	0.71	0.50	0.79	0.67	0.46	0.43	0.65	0.30

Table 12: B_1 matrix with sectors indexes (NNDSVD^{ar})

Sector	NMF factors									
	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Oil & Gas	0.94	0.38	0.32	0.25	0.27	0.16	0.83	0.28	0.33	1.00
Basic Materials	0.79	0.50	0.83	0.67	0.23	0.92	1.00	0.32	0.38	0.87
Industrials	0.65	0.65	0.73	0.79	0.57	1.00	0.97	0.56	0.55	0.64
Consumer Goods	0.82	0.46	0.55	0.36	0.36	0.78	0.97	0.36	0.42	0.56
Health Care	1.00	0.44	0.21	0.06	0.15	0.10	0.46	0.25	0.31	0.83
Consumer Services	0.63	0.71	0.49	0.51	0.56	0.50	0.86	0.49	0.39	0.29
Telecommunications	0.52	0.61	0.96	0.59	1.00	0.29	0.90	0.50	0.56	0.22
Utilities	0.82	0.53	1.00	0.70	0.31	0.16	0.95	0.53	0.41	0.90
Financials	0.74	0.69	0.84	1.00	0.20	0.32	0.78	1.00	0.68	0.94
Technology	0.43	1.00	0.83	0.99	0.78	0.54	0.99	0.76	1.00	0.51

Table 13: B_1 matrix with sectors indexes (KM)

Sector	NMF factors									
	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Oil & Gas	0.93	0.98	0.84	0.76	0.95	0.99	0.87	0.88	0.79	0.71
Basic Materials	1.00	0.93	0.76	1.00	0.75	0.83	0.84	0.89	0.91	0.61
Industrials	0.80	0.83	0.72	0.88	0.75	0.74	0.78	0.74	0.83	0.79
Consumer Goods	0.87	1.00	0.72	0.95	0.82	0.88	0.82	0.83	0.76	0.72
Health Care	0.96	0.94	0.82	0.77	1.00	1.00	1.00	0.91	0.59	0.77
Consumer Services	0.66	0.69	0.59	0.62	0.74	0.78	0.78	0.67	0.60	0.88
Telecommunications	0.70	0.71	0.51	0.52	0.74	0.63	0.55	0.74	0.62	0.93
Utilities	1.00	0.89	0.85	0.77	0.84	0.86	0.86	1.00	0.95	0.67
Financials	0.81	0.82	1.00	0.78	0.81	0.80	0.97	0.85	1.00	0.65
Technology	0.57	0.73	0.58	0.44	0.73	0.70	0.76	0.55	0.64	1.00

Table 14: B_1 matrix with sectors indexes (CRO)

Sector	NMF factors									
	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Oil & Gas	0.22	0.93	0.34		0.09					0.67
Basic Materials	0.29	1.00				0.25		1.00		0.17
Industrials	0.75	0.05		0.60	0.30	0.71		0.80	0.20	
Consumer Goods	0.92	0.20			0.15			0.59		
Health Care	1.00	0.03	0.60						0.12	
Consumer Services	0.46		0.76	0.07	0.35	0.41		0.26	0.21	0.23
Telecommunications	0.04	0.27	0.75		1.00	0.78				
Utilities		0.77	1.00			1.00	0.26			0.27
Financials		0.38	0.70	1.00		0.02	1.00	0.05	1.00	0.13
Technology		0.28	0.10		0.66	0.17	0.18	0.76	0.65	1.00

C.2 Results of classification

Here are the clusters obtained with the K-means classifier:

- #1: TOTAL SA (1); ENI SPA (1); E.ON AG (7000); DANONE (3000); ENEL SPA (7000); AIR LIQUIDE SA (1000); IBERDROLA SA (7000); VINCI SA (2000); ARCELORMITTAL (1000); L'OREAL (3000); ASSICURAZIONI GENERALI (8000); REPSOL YPF SA (1); CARREFOUR SA (5000); RWE AG (7000); UNIBAIL-RODAMCO SE (8000); PERNOD-RICARD SA (3000); ESSILOR INTERNATIONAL (4000); SAMPO OYJ-A SHS (8000); FORTUM OYJ (7000); HEINEKEN NV (3000); FRESENIUS MEDICAL CARE AG & (4000); SAIPEM SPA (1); VALLOUREC (2000); HENKEL AG & CO KGAA VORZUG (3000); FRESENIUS SE & CO KGAA (4000); TECHNIP SA (1); REED ELSEVIER NV (5000); SOLVAY SA (1000); EDP-ENERGIAS DE PORTUGAL SA (7000); ABERTIS INFRAESTRUCTURAS SA (2000); DELHAIZE GROUP (5000); SODEXO (5000); MERCK KGAA (4000); GROUPE BRUXELLES LAMBERT SA (8000); ATLANTIA SPA (2000);
- #2: SIEMENS AG-REG (2000); SAP AG (9000); KONINKLIJKE PHILIPS ELECTRON (3000); NOKIA OYJ (9000); ASML HOLDING NV (9000); BOUYGUES SA (2000); ALCATEL-LUCENT (9000); CAP GEMINI (9000); STMICROELECTRONICS NV (9000);
- #3: BANCO SANTANDER SA (8000); DEUTSCHE BANK AG-REGISTERED (8000); BANCO BILBAO VIZCAYA ARGENTA (8000); AXA SA (8000); INTESA SANPAOLO (8000); UNICREDIT SPA (8000); ERSTE GROUP BANK AG (8000); NATIONAL BANK OF GREECE (8000);
- #4: TELEFONICA SA (6000); DEUTSCHE TELEKOM AG-REG (6000); FRANCE TELECOM SA (6000); VIVENDI (5000); KONINKLIJKE KPN NV (6000); TELECOM ITALIA SPA (6000); SAFRAN SA (2000); PUBLICIS GROUPE (5000); PORTUGAL TELECOM SGPS SA-REG (6000);
- #5: BASF SE (1000); SCHNEIDER ELECTRIC SA (2000); LVMH MOET HENNESSY LOUIS VUI (3000); LINDE AG (1000); COMPAGNIE DE SAINT-GOBAIN (2000); THYSSENKRUPP AG (2000); AKZO NOBEL (1000); CRH PLC (2000); ADIDAS AG (3000); PPR (5000); LAFARGE SA (2000); K+S AG (1000); KONINKLIJKE DSM NV (1000); HEIDELBERGCEMENT AG (2000); UPM-KYMMENE OYJ (1000); CHRISTIAN DIOR (3000); METRO AG (5000); RYANAIR HOLDINGS PLC (5000); METSO OYJ (2000);
- #6: SANOFI (4000); UNILEVER NV-CVA (3000);
- #7: BNP PARIBAS (8000); ING GROEP NV-CVA (8000); SOCIETE GENERALE (8000);
- #8: BAYER AG-REG (1000); KONINKLIJKE AHOLD NV (5000); ALSTOM (2000);
- #9: DAIMLER AG-REGISTERED SHARES (3000); BAYERISCHE MOTOREN WERKE AG (3000); VOLKSWAGEN AG-PFD (3000); MICHELIN (CGDE)-B (3000); MAN SE (2000); RENAULT SA (3000); PORSCHE AUTOMOBIL HLDG-PFD (3000); FIAT SPA (3000);
- #10: ALLIANZ SE-REG (8000); MUENCHENER RUECKVER AG-REG (8000); COMMERZBANK AG (8000); AEGON NV (8000);

The next table indicates some stocks which are not explained by their sector index.

Table 15: B_2 coefficients

Name	Sector	Factors									
		0001	1000	2000	3000	4000	5000	6000	7000	8000	9000
ALLIANZ	Financials						0.24				0.78
SAP	Technology		0.08	0.12	0.12	0.39		0.30			
ENEL	Utilities	0.66									0.34
VIVENDI	Consumer Services							0.03			0.91
VINCI	Industrials		1.17						0.02		
SAINT-GOBAIN	Industrials	0.48				0.01				0.47	
VOLKSWAGEN	Consumer Goods		1.08								
MUENCHENER RUECKVER	Financials						0.86				0.22
UNIBAIL-RODAMCO	Financials		1.19								
PERNOD-RICARD	Consumer Goods	0.12	1.08								
ESSILOR INTERNATIONAL	Health Care		1.13								
THYSSENKRUPP	Industrials		0.82							0.09	
MICHELIN	Consumer Goods	0.28	0.68							0.07	
CRH	Industrials	0.61	0.04						0.08	0.26	
ADIDAS	Consumer Goods	0.10	1.01								
FORTUM	Utilities		1.25								
TELECOM ITALIA	Telecommunications										1.02
FRESENIUS MEDICAL CARE	Health Care		0.86					0.17			
MAN	Industrials		1.02								
SAIPEM	Oil & Gas		1.27								
VALLOUREC	Industrials		1.37								
ALSTOM	Industrials										0.79
LAFARGE	Industrials	0.48								0.28	0.20
FRESENIUS SE & CO	Health Care		0.99						0.08	0.01	
AEGON	Financials										0.89

We have the following correspondance between codes and sectors: 0001 = Oil & Gas, 1000 = Basic Materials, 2000 = Industrials, 3000 = Consumer Goods, 4000 = Health Care, 5000 = Consumer Services, 6000 = Telecommunications, 7000 = Utilities, 8000 = Financials, 9000 = Technology.

Here are the clusters obtained with the NMF classifier (NNDSVD^{ar}, $n = 5$):

- #1: TOTAL SA (1); BASF SE (1000); BNP PARIBAS (8000); DANONE (3000); AIR LIQUIDE SA (1000); UNIBAIL-RODAMCO SE (8000); ESSILOR INTERNATIONAL (4000); MICHELIN (CGDE)-B (3000); CRH PLC (2000); ADIDAS AG (3000); HEINEKEN NV (3000); HENKEL AG & CO KGAA VORZUG (3000); KONINKLIJKE DSM NV (1000); SOLVAY SA (1000); ABERTIS INFRAESTRUCTURAS SA (2000); GROUPE BRUXELLES LAMBERT SA (8000); ATLANTIA SPA (2000);
- #2: DEUTSCHE TELEKOM AG-REG (6000); FRANCE TELECOM SA (6000); VIVENDI (5000); ARCELORMITTAL (1000); KONINKLIJKE KPN NV (6000); SAFRAN SA (2000); CAP GEMINI (9000);
- #3: BANCO SANTANDER SA (8000); BAYER AG-REG (1000); SCHNEIDER ELECTRIC SA (2000); INTESA SANPAOLO (8000); REPSOL YPF SA (1); THYSSENKRUPP AG (2000); FRESENIUS MEDICAL CARE AG & (4000); BOUYGUES SA (2000); FRESENIUS SE & CO KGAA (4000); TECHNIP SA (1); EDP-ENERGIAS DE PORTUGAL SA (7000); METRO AG (5000); DELHAIZE GROUP (5000); MERCK KGAA (4000);
- #4: SIEMENS AG-REG (2000); TELEFONICA SA (6000); SAP AG (9000); LVMH MOET HENNESSY LOUIS VUI (3000); KONINKLIJKE PHILIPS ELECTRON (3000); ASML HOLDING NV (9000); TELECOM ITALIA SPA (6000); CHRISTIAN DIOR (3000); PUBLICIS GROUPE (5000); PORTUGAL TELECOM SGPS SA-REG (6000);
- #5: E.ON AG (7000); IBERDROLA SA (7000); VINCI SA (2000); LINDE AG (1000); VOLKSWAGEN AG-PFD (3000); RWE AG (7000); SAMPO OYJ-A SHS (8000); FORTUM OYJ (7000); SAIPEM SPA (1); K+S AG (1000);
- #6: SANOFI (4000); ENI SPA (1); UNILEVER NV-CVA (3000); L'OREAL (3000); BAYERISCHE MOTOREN WERKE AG (3000); PERNOD-RICARD SA (3000); REED ELSEVIER NV (5000); UPM-KYMMENE OYJ (1000); RYANAIR HOLDINGS PLC (5000);
- #7: SOCIETE GENERALE (8000); UNICREDIT SPA (8000); COMPAGNIE DE SAINT-GOBAIN (2000); LAFARGE SA (2000); RENAULT SA (3000); ERSTE GROUP BANK AG (8000); PORSCHE AUTOMOBIL HLDG-PFD (3000);
- #8: MAN SE (2000); VALLOUREC (2000); HEIDELBERGCEMENT AG (2000); METSO OYJ (2000);
- #9: DAIMLER AG-REGISTERED SHARES (3000); DEUTSCHE BANK AG-REGISTERED (8000); BANCO BILBAO VIZCAYA ARGENTA (8000); ENEL SPA (7000); ASSICURAZIONI GENERALI (8000); CARREFOUR SA (5000); MUENCHENER RUECKVER AG-REG (8000); AKZO NOBEL (1000); KONINKLIJKE AHOLD NV (5000); PPR (5000); SODEXO (5000); STMICROELECTRONICS NV (9000);
- #10: ALLIANZ SE-REG (8000); ING GROEP NV-CVA (8000); AXA SA (8000); NOKIA OYJ (9000); COMMERZBANK AG (8000); ALSTOM (2000); ALCATEL-LUCENT (9000); AEGON NV (8000); FIAT SPA (3000); NATIONAL BANK OF GREECE (8000);