Anna M. BARTKOWIAK

University of Wrocław

doi:10.15199/48.2017.01.14

Classic and convex non-negative matrix visualization in clustering two benchmark data

Abstract. *Both the classic and the convex NMF (Nonnegative Matrix Factorization) yield a parsimonious, lower rank representation of the data. They may yield also an indication on a soft clustering of the data vectors, We analyze two sets of diagnostic data, wine and sonar, for which the classic and convex nonnegative matrix factorization (NMF) behave differently when indicating group membership of the data vectors. The data are given as mxn matrices, with columns denoting objects, and rows - their attributes. We assess the clustering by multivariate graphical visualization methods.*

Streszczenie. *Dla wybranych danych 'wine' i 'sonar' znajdujemy – za pomocą NMF (nieujemna faktoryzacja macierzy) – ukrytą strukturę tych macierzy oraz wskazania co do klasteryzacji obiektów przedstawianych w kolumnach danych. Otrzymaną klasteryzację potwierdzamy trzema metodami wielozmiennej wizualizacji wektorów danych.* ***(Klasteryzacja przy użyciu klasycznej i typu convex nieujemnej faktoryzacji macierzy na przykładzie dwóch zbiorów danych)***

Keywords: non-negative matrix factorization, matrix approximation, reduction of dimensionality, multivariate data space, multivariate graphical visualization, clustering of data vectors
Słowa kluczowe: nieujemna faktoryzacja macierzy, aproksymacja macierzy, redukcja wymiarowości,wielozmienna przestrzeń danych, wizualizacja graficzna wielozmiennych danych, wyznaczanie skupień wektorów danych

1. Introduction

We consider a size $m \times n$ data matrix $\mathbf{X}$ composed from n column data vectors denoting objects or individuals:

$$\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \text{ with } \mathbf{x}_j \in \mathbb{R}^m,\ j = 1, \ldots n.$$

Each data vector $\mathbf{x}_i$ is considered as an object (individual, data sample) characterized by m attributes. The same data vector $\mathbf{x}_i$ may be also viewed as a data point in the m-dimensional data space $\mathbb{R}^m$. The specific assumption about the data matrix **X** is that **its elements are nonnegative**: $\mathbf{X} \geq 0$; which, equivalently, is indicated by the notation $\mathbf{X}_+$, or with more details: $\mathbf{X} \in \mathbb{R}_+^{m \times n}$.

Notice, that the notation $m \times n$ in the data matrix **X** is different as those used in statistics, where the data matrix is typically denoted as $\mathbf{X} \in \mathbb{R}_{\pm}^{n \times m}$, with its data vectors $i = 1, \ldots, n$ being row vectors located in the rows of $\mathbf{X}$ (The notation $M_{\pm}$ means that the elements of that matrix may be of mixed sign, that is both positive and negative).

The NMF method (short for Non-negative Matrix Factorization) provides for the given matrix $\mathbf{X}_{m \times n}$, and an assumed constant integer k – usually much less than $min(m, n)$ – a kind of approximation by lower rank matrices $\mathbf{A}$ and $\mathbf{H}$:

$$\mathbf{X}_{m \times n} \approx \mathbf{A}_{m \times k} * \mathbf{H}_{k \times n}, \quad s.t. \quad \mathbf{A}, \mathbf{H} \geq 0 \tag{1}$$

where the factorizing matrices $\mathbf{A}$ and $\mathbf{H}$ are non-negative matrices $\mathbf{A}_+$ and $\mathbf{H}_+$, that is, they are subjected to the constraints $\mathbf{A}, \mathbf{H} \geq 0$. The matrices **A** and **H** are called the **NMF factors** of $\mathbf{X}$.

The factor **A** of size $m \times k$ is called the *basis*. Its columns constitute basis vectors and act as representatives (prototypes) of all the column vectors $\mathbf{x}_j,\ j = 1, \ldots, n$ contained in $\mathbf{X}$.

The factor **H** of size $k \times n$ is referred to as *encoder*, and its columns provide coefficients permitting to reproduce (approximate) from the basis **A** the respective columns of **X**. For example, to approximate (reconstruct) the column vector $\mathbf{x}_j \in \mathbf{X}_{m \times n}$, one uses the formula:

$$\mathbf{x}_j \approx \hat{\mathbf{x}}_j = \mathbf{A} * \mathbf{h}_j,\ \ j = 1, \ldots, n. \tag{2}$$

The full **X** matrix is approximated (reconstructed) by performing the simple multiplication:

$$\mathbf{X} \approx [\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_n] = \hat{\mathbf{X}} = \mathbf{A} * [\mathbf{h}_1, \ldots, \mathbf{h}_n]. \tag{3}$$

The quality (faithfulness) of the approximation is usually expressed by residuals calculated by the LS (Least Squares) method, or by an index called divergence (e.g. Kullback-Leibler divergence) derived from quasi-likelihood of the distributions of $\mathbf{X}$ and $\hat{\mathbf{X}}$.

The developments of NMF started since the appearance of the seminal paper [15] and are continuously growing. The NMF method is a cousin of PCA; however thanks to the non-negativity of its constituents it is gaining more and more popularity. It is reported that NMF, as opposed, e.g., to PCA, yields more interesting approximations and is more flexible because of the many ways the discrepancy between the observed matrix (**X**) and the reconstructed matrix $\widehat{\mathbf{X}}$ can be defined and deployed. For the developments, see [3, 7, 8, 27] and references therein. For particular comparison with PCA, see [1] and references therein. Generally, NMF compares favorably with PCA.

The main applications of NMF are: matrix decomposition, matrix approximation and reduction of dimensionality. From the perspective of data analysis, the NMF yields also data reduction. It has been also notified, that the NMF (or its extended 'sparse' versions) may provide in its factors also some information on clustering of the data vectors $\mathbf{x}_j \in \mathbf{X}$; moreover, the clustering indications are somehow similar to results obtained by the k-means and the spectral clustering algorithms [16, 5, 4, 17, 25, 26]. It was also stated, that NMF acts as a soft clustering algorithm.

It appears that the clustering indications of NMF do not always conform with the known subgroups of the data. The indications of NMF may point to some clustering of the data, which indeed shows some new structure of the data - and this means a success for the data analyst who can explain the indicated partition. However, it may happen, that the indication is so soft that it is practically useless.

Our question is: Can this strange behavior - that is finding good or useless clusters – be connected with some properties of the data discovered independently of the NMF analysis? Our thesis is, that this behavior is connected with geometry (topology) of the data vectors/points observed in the multivariate data space. To our best knowledge, this problem has not been directly addressed so far. In the following we present an experiment carried out to confirm our supposition. In our considerations, we will concentrate on the classic NMF (as formulated in [15]) and its extended version, called convex NMF (as formulated in [5]). The parameters of the mod-

els will be obtained using the Frobenius norm as goodness of fit of the approximation of $\mathbf{X}$ by the rank k approximation $\mathbf{AH}$ appearing in left hand expression in eq. (1). The ALS (Alternating Least Square) algorithm will be applied to obtain the factor matrices $\mathbf{A}$ and $\mathbf{H}$. We will use the implementation of the algorithms described in [18].

Our presented experiment has as goal to throw light on the circumstances of the encountered situation: various clustering abilities of the NMF observed for various data sets. We will investigate in detail two known real data sets (wine and sonar) taken from the UCI data repository. Both data sets have known a priori number of subgroups ($k = 3$ for wine and $k = 2$ for sonar). We will try to find the (known a priori) subgroups of the analyzed data using classic and convex NMF. For comparison, we will apply also the standard k-means algorithm.

Independently, in search of the topological geometry of the data, we will visualize both data sets by three multivariate visualization methods: Kohonen self-organizing maps, t-distributed stochastic embedding, and canonical discriminant functions derived from Fisher's LDA. We will observe and compare the results.

Our thesis is, that the ability of clustering of the NMF algorithm depends on the geometry of the analyzed data points. If the geometry of the data points conforms with the clusters devised a priori, then NMF will recover the devised clusters. Otherwise the applicability of NMF for clustering will be uncertain and its applicability (accuracy of correct classification) will be modest or none.

In next Section 2 we present shortly the two data sets chosen for analysis. Section 3 introduces shortly methods using for the analysis. Sections 4 and 5 are devoted to detailed analysis and presentation of results obtained for the two selected two data sets. Finally, Section 6 contains summary of the results and closing remarks.

2. The data sets 'wine' and 'sonar'

In the following we will consider two real data sets: the wine and the sonar data. Both sets were downloaded from the UCI Data Repository and constitute a kind of benchmark data for clustering.

Both data sets – as downloaded from UCI – are nonnegative. The wine variables are expressed on a much differentiated scale. Therefore they need some standardization. The sonar data exhibits real values from [0,1], hence this data set was used for analysis in its original scale.

The **wine** data set is given statistically as a real data matrix $\mathbf{X}$ of size 178×13.comprises k=3 groups of data, coming from 3 vineyards. Each data vector (row of $\mathbf{X}$) is characterized by 13 analytical (chemical) variables describing some properties of the given wine samples. The data are composed from 3 groups of data, coming from 3 different vineyards denoted by us as *A, B, C*. The cardinalities of these (sub)groups are: $n_A = 59, n_B = 71, n_C = 47$. Their subsequent no.s in the wine data set are:
A, no.s $1 - 59$; *B*, no.s $60 - 130$; and *C*, no.s 131-178.

The detailed description of the data and their 13 variables are at the UCI site. An analysis of the data and graphs showing the meaningfulness of the clustering into 3 subgroups may be found in [2] .

The **sonar** data set comprises k=2 groups of data containing reflectance signals from rock and mine. The data set is given statistically as a real data matrix $\mathbf{X}_{n\times m}$ of size 208×60. The m=60 variables are psd (power spectrum density) characteristics obtained from Fourier analysis of waveform signals (chirps) beamed at two objects: piece of rock and a cylindrical mine, located on the bottom of a water reservoir. The reflectance of the signals is different for the two objects. The data matrix $\mathbf{X}$ contains as its subgroups firstly the reflectance signals returned from the 'rock' object, and next the reflectance signals returned from the 'mine' object. Thus we have two sub-groups of data labelled 'rock' and 'wine'. The subgroups will be referred to as G1 and G2 appropriately. The sequence no.s (#-es) of the data vectors in the data set are: G1, no-s $1 - 97$; and G2, no-s $98 - 208$.

The data were donated to UCI data repository by Gorman and Sejnowski, who designed the experiment, gathered the data, and working with Neural Networks obtained some decision functions for recognition the group membership ('rock' or 'mine') on the base of the returned chirp signal [9, 10].

We will show that the considered NMF and k-means algorithms when considered with k equal to the proper number of subgroups (that is: $k = 3$ for wine and $k = 2$ for sonar) yield very useful clustering information for the wine data, but truly no clustering information for the sonar data. Why? Could it be expected? On what ground?

To find out, what's going on, we will concentrate on three tasks answering the following three questions:

1. How affine are the respective NMF bases A_{wine} or A_{sonar} to the corresponding K-means centroids?
2. How close are the group assignment obtained from the NMF encoders given in the matrices H_{wine} and H_{sonar} to the respective group assignments obtained by the corresponding k-means results?
3. Is it possible to recognize by the considered NMF techniques the vineyard of the wine data vectors or the type (rock-or-mine?) of object belonging to the sonar data?

3. Methods used for analysis

We define the notation '**d.v.**' for the expression 'data vector', and the notation **d.p.** for 'data point'.

We will use in our experiment the following analytical methods:

For computing NMF and k-means:
- Ordinary NMF with LS estimation method [18]
- Convex NMF with LS estimation method [18]
- kmeans [20, 24]

For Multivariate visualization of the *d.p.*'s:
- Kohonen Self-Organizing Maps [12, 24]
- t-distributed stochastic embedding [23]
- Canonical discriminant functions (CDF) [6, 14, 20].

For a given data set, the methods will be carried out in the following order

(a) classic and convex NMF,
(b) clustering by k-means,
(c) exploratory data visualization by SOM, t-sne, CDF.

Let's state clearly, that all the used in our analysis methods – except canonical discriminant functions – work in an unsupervised mode. The group colors in the presented figures were added *ex post* after obtaining the results/projections.

Most of the techniques we listed above are well known and do not need explaining; except, perhaps, the convex NMF and the t-sne. Therefore, these two methods are briefly introduced below in the paragraphs headed 'Variants of NMF' and 'Exploratory data visualization'.

Variants of NMF

The **classic NMF** introduced in formula (1) considers the model

$$\mathbf{X}_+ \approx \mathbf{A}_+ * \mathbf{H}_+ \tag{4}$$

Thus the data matrix and both NMF factors must be non-negative. It is supposed that the factorization in (4) is a technique for representation of intrinsically non-negative data as a linear combination of spatially localized components, called parts-based representation [15].

The factors **A** and **H** may be obtained from the optimization criterion

$$\min_{A,H} \frac{1}{2}||\mathbf{X} - \mathbf{AH}||_{\mathbf{F}}^{\mathbf{2}} \quad \text{s.t. } \mathbf{A}, \mathbf{H} \geq \mathbf{0}. \tag{5}$$

Let us mention, that the criterion for finding the factors **A** and **H** can be also obtained using stochastic Bayesian reasoning. It can be derived from the log-posterior probability under the assumptions of Gaussian error, Gaussian distributed basis vectors, and Laplace-distributed coefficient vectors [18].

The classic NMF has been successfully applied in several applications, among others in pattern recognition, image processing, object detection in images, classification and clustering, however rather it does not result in 'parts-based' representation, as emphasized in [15]. Also the classification and clustering by the classic NMF, as presented in [15], appeared often to be problematic. It was notified [11] that more sparse matrices $\mathbf{A}$ and $\mathbf{H}$ yield 'better' clustering results. Therefore several more constrained algoritms (as compared to the model in eq. (5) were elaborated [11, 22, 18]. Among them are algorithms called **sparse NMF models**; they were reported to have better clustering properties. However, it was also observed that adding too much emphasized sparseness constraints resulted in a degradation of the approximation accuracy of the analyzed data matrix [21].

The **convex NMF** is a specific sparse ZMN model introduced by Ding et al. in [5], see also ([18]). At first glance it follows the model shown in eq. (1), however the basis $\mathbf{A}$ is computed in a different way:

$$\mathbf{X}_\pm \approx \mathbf{A}_\pm * \mathbf{H}_+, \text{ where } \mathbf{A}_\pm = \mathbf{X}_\pm * \mathbf{W}_+. \tag{6}$$

In the convex NMF model both the data matrix **X** and the factor **A** are allowed to have mixed sign elements. Moreover, what is perhaps more important, the basis $\mathbf{A}$ is supposed to be constructed as linear combination of the column vectors from **X**. This means that the basis is spanned by the data vectors $\mathbf{x}_j$ from **X**, and as such it belongs to the same space. The linear combinations spanning the basis **A** are designated by the matrix $\mathbf{W}_+$ of size $n \times k$. The comments of the authors of the convex NMF methods may highlight more the idea of the convex NMF. The authors [5] write in Section 2.2 of their paper: 'for reasons of interpretability, we may wish to restrict ourselves to convex combinations of the column of **X**. This constraint has the advantage that we could interpret the columns of **A** as weighted sum of certain data points; in particular, these columns would capture a notion of *centroids*.. . . As we will see, convex NMF has an interesting property: its factors $\mathbf{A} = \mathbf{XW}$ and $\mathbf{H}$ both tend to be very sparse.'

Both algorithms (that is, classic and convex NMF) work in unsupervised mode. The results by NMM reported in this paper were obtained using the software [18].

Note, that for calculations with the NMF algorithms, the transpose of the data matrix downloaded from UCI should be taken.

Exploratory Data Visualization

Here we will use three methods: SOM, t-sne, CDF (alias: Fisher's LDA) .

The principles of **Kohonen's self-organizing map (SOM)** are described in [12]. For calculations, we have used the software [24].

The t-sne (t-distributed stochastic neighbor embedding) [23] aims at the preservation of the similarities $\mathcal{S}$ among objects (d.v.'s) in a high-dimensional space and translation them to lower dimensional (e.g. 2D) space. The similarities are evaluated by considering stochastic neighborhoods in both spaces.

Let's first consider the data space R^m where the multivariate data points $\mathbf{x}_1, \ldots, \mathbf{x}_n$ reside. The authors [23] assume here that the data points $\mathbf{x}_i$ are generated by a probability function with Gaussian kernel centered at $\mathbf{x}_i$, with an free parameter σ (to be estimated by the user). Then the conditional probabilities $p_{j|i}$ that $\mathbf{x}_j$ is a neighbor of given $\mathbf{x}_i$ is expressed as follows:

$$p_{j|i} = \frac{exp\{-||\mathbf{x}_i - \mathbf{x}_j||^2/2\sigma^2\}}{\sum_{k \neq i} exp\{-||\mathbf{x}_i - \mathbf{x}_k||^2/2\sigma^2\}} \tag{7}$$

Of course, $p_{j|i} \neq p_{i|j}$. However, the values p_{ij} of symmetrized conditional probabilities can be defined as $p_{ij} = (p_{j|i} + p_{i|j})/2n$.

Analogous probabilities q_{ij} for the projection pairs $\mathbf{y}_i, \mathbf{y}_j$ are defined via Student-t distribution with one degree of freedom , which is equivalent to a Cauchy distribution (motivation explained in [23]):

$$q_{ij} = \frac{(1 + ||\mathbf{y}_i - \mathbf{y}_j||^2)^{-1}}{\sum_{k \neq i} (1 + ||\mathbf{y}_i - \mathbf{y}_k||^2)^{-1}} \tag{8}$$

Both p_{ii} and q_{ii} $(i = 1, \ldots, n)$ are set to zero.

In such a way one obtains two probability distributions P and Q of symmetrized conditional probabilities, which may be compared by the Kullback-Leibler (KL) divergence denoted in the equation below as *C*. The KL divergence is a widely used distance measure for accounting the difference between two probability distributions.:

$$C = \sum_i KL(P||Q) = \sum_i \sum_j p_{ij} log \frac{p_{ij}}{q_{ij}}. \tag{9}$$

For the calculation of t-sne we have used the Matlab function implemented by van der Maaten downloaded from his homepage [23].

Canonical Discriminant Analysis (CDA) is also known as **Fisher's LDA** (Linear Discriminant Analysis) [14, 6, 20]. It builds canonical discriminant functions derived on the base of maximizing the Fisher's criterion being the ratio of the between group to the within group scatter. This is a supervised method of analysis; it needs information of the group membership of the learning sample. The presented hereafter results were obtained using own Matlab function.

The **k-means** calculations were performed using the Matlab *STATS* toolbox. The k-means method is very popular method; it is described, for example, in described in [20]. Principles of cluster analysis are described in [6, 13].

4. Analysis of the wine data

We proceed here along the points (a), (b) and (c) announced at the begin of Section 3.

(a) Results of classic and convex NMF

Note, that for calculations with the NMF algorithms, the transposes of the data downloaded from UCI should be taken.

Let $\mathbf{X}$ of size 13×178 denote the analyzed data matrix. Because of large scale differences for the analyzed variables, the data matrix $\mathbf{X}$ was firstly standardized 'by range'. The model for the **classic NMF** given as eq. (4) yields the matrix decomposition $\mathbf{X}_+ = \mathbf{A}_+ * \mathbf{H}_+ + \mathbf{E}$, with the basis matrix $\mathbf{A}$, the encoder $\mathbf{H}$ and error matrix $\mathbf{E}$ of the approximation.

Assuming the rank of approximation $k = 3$, we obtained, as factors of the decomposition, the size 13×3 matrix $\mathbf{A}$, and the size 13×178 matrix $\mathbf{H}$. They are displayed in Fig. 1.

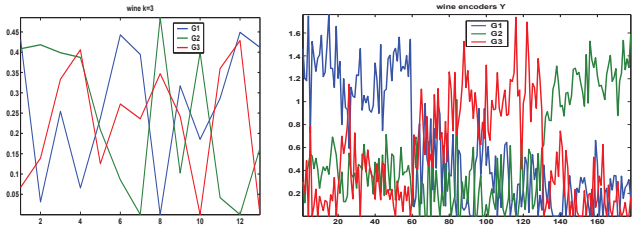

Fig. 1. Wine by classic NMF. Left: Column profiles of the basis A. Right: Row profiles of encoder H.

The left exhibit of Fig. 1 shows The profiles of the three basic vectors contained in $\mathbf{A}$. They are deemed to be representatives of three clusters of data, indicated by the encoder $\mathbf{H}$. Looking at that exhibit one may state, that $\mathbf{A}$ is not specially sparse, none-the-less the three profiles are well distinguishable. The rows of the encoders from $\mathbf{H}$ are shown in the right exhibit of Fig. 3. It may be noticed, that the 3 row-profiles of $\mathbf{H}^T$ are decidedly more sparse. Moreover, the dominating colors of the profiles conform (with few exceptions) with the vineyard groups. We observe:
for no.s 1-59 the dominating color is green (1st row of $\mathbf{H}$),
for no.s 60-130 the dominating color is red (3rd row of $\mathbf{H}$),
for no.s 131-178 the dominating color is blue (2-nd row of $\mathbf{H}$).
Thus, in this case the classic NMF indicates really the clusters contained in the data, that is, it indicates the 3 vineyards wherefrom the samples come.

Fig. 2 shows analogous basis $\mathbf{A} = \mathbf{X} * \mathbf{W}$ and the encoders $\mathbf{H}$ obtained from the convex NMF (see eq. 6) when assuming rank $k = 3$. Additionally, it provides the matrix $\mathbf{W}$, which shows which data vectors contribute mostly in establishing the basic vectors in $\mathbf{A}$. The $n \times k$ weight matrix **W** is shown separately in Fig. 3.

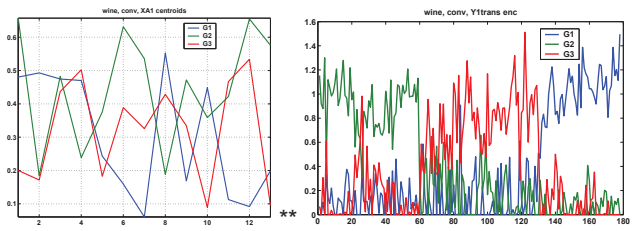

Fig. 2. Wine by convex NMF. Left: Profiles of column vectors of basis A obtained as A=XW. Right: Profiles of row vectors of encoder H.

Looking at Fig. 1 and Fig. 2 one may state that they look much alike, both the bases and the encoders. The encoders differ for *d.v.*'s belonging to the three sub-groups of the wine data (the change is visible for *d.v.*'s no. 60 and 131.

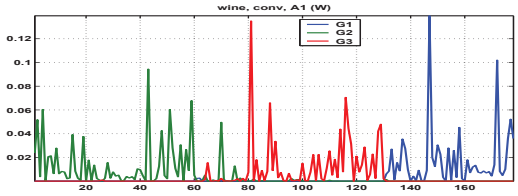

Fig. 3. Wine. Factorization using convex NMF. Profile of weighting matrix W size 178×3. Notice that the matrix is very sparse. The sparseness conforms with the groups of the data.

Fig. 3 shows how the basis vectors $A = [a_1, a_2, a_3]$ were composed. First basis vector a_1 was composed mainly from d.v.'s belonging to group 1; second basis vector a_2 was composed mainly from d.v.'s belonging to group 2; and third basis vector a_3 was composed mainly from d.v.'s belonging to group 3. The contributions to the bases are clearly visible. It is interesting to note that only a part of the *d.v.*'s contributes markedly to the contents of the basic vectors. The weight matrix $\mathbf{W}$ is very sparse indeed.

The conclusion from this part of analysis is: Both classic and convex NMF may serve for clustering the wine data. Both classic and convex are good in this respect. The convex NMF is a little better by showing in more detail which data vectors participate mostly when constructing the basis.

(b) Wine. Clustering by k-means for k=3

The results of the computations by **k-means** are shown in Fig. 4. In the top left exhibit of the figure one finds the centroids of the clusters. They correspond to the basic vectors in $\mathbf{A}$ obtained from the classic and the convex NMF. Left exhibit shows the group assignments indicated by the k-means algorithm, the first and last group have only correct assignments, without any error.

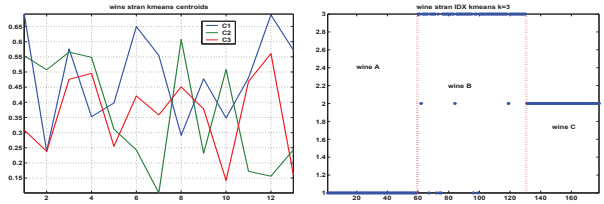

Fig. 4. Wine. Results of k-means for k=3. Left: Profiles of the three centroids. Right: Group assignment by k-means. The middle group has 8 erroneous assignments.

In the top right exhibit one finds the group assignment of subsequent data vectors. The assignment is crisp: it can be only expressed by integers 1, 2, or 3. One may note, that the groups containing the data vectors no.s 1-59 got assignment '1' and those with no.s 131-168 got assignment '2'. Thus these two groups are pure.
What concerns data vectors no.s 60-130, the majority of them got assignment '3', yet there are 3 dv's with assignment '3' and 5 d.v.'s with assignment '1'. Thus 8 items (wine samples) belonging to this group was not recognized and erroneously indicated as belonging to other groups. assignments.

Summarizing, the k-means performs only crisp group assignments, saying yes or no, while the NMF provides a 'soft' assignment by indicating how likely it is that the given item belongs to one of the putative groups. Such kind of assignments is more informative and may be preferable in real situations.

(c) Wine. Exploratory visualization

We visualize the data using Kohonens' self-organizing maps, t-distributed stochastic neighbors embedding, and canonical discriminants functions . For the exploratory data

analysis we used the $n \times m = 178 \times 13$ data matrix, which was standardized to means equal zero and std=1 for all 13 variables. The results are shown in Fig. 5 and Fig. 6.

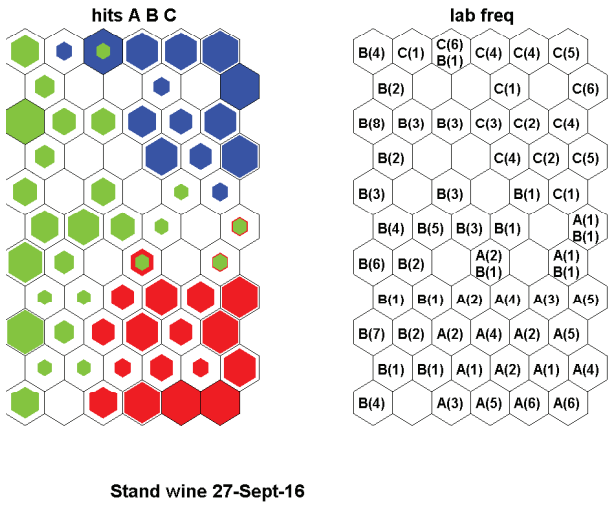

Fig. 5. Visualization of the wine data using self-organizing maps of size 11×6. Left: Som-hits into the hexagons of the map – by data vectors belonging to 3 subgroups of the wine data. The number of hits into each hexagon (its frequency) is shown by the magnitude of the painted (in red, green or blue) area inside each hexagon. Right: The same, as at left, however now the frequency of hits is printed directly as integer number.

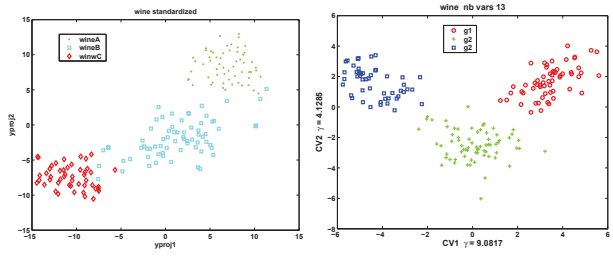

Fig. 6. Wine. Left: Visualization using tsne. Points from the three groups appear practically separated. Right: Visualization using canonical discriminant functions. The 3 groups appear completely separated

Fig. 5 shows a hexagonal map of size 11×6. It yields a faithful visualization of the data, because the quantization_error = 1.883, and the topological_error = 0.017. For explanations, see [24]. Looking at the maps displayed in Fig. 5 one states that only 4 hexagons are impure: these are hexagons at positions (d.v. short for data vector)
{1,3}: 1 green *d.v.* from group B and 6 *blue d.v.*'s from C);
{6,6}: 1 green *d.v.* from group B and 1 red *d.v.* from A;
{7,4}: 1 green *d.v.* from group B and 1 red *d.v.* from A;
{7,6}: 1 green *d.v.* from group B and 1 red *d.v.* from A;

Fig. 6 shows the wine data as viewed by the t-sne method (left panel) and the canonical discriminant analysis method (right panel). The t-sne is an unsupervised method and does not use any outside information; yet it recognizes the three subgroups nearly perfectly. The CDA, a supervised method, recognizes the 3 subgroups even better.

Generally, this part of analysis permits to state that the wine groups (vineyards wherefrom the wine samples come) are well discernible in the 13-dimensional data space.

A general conclusion from the present section is that the wine groups are well recognized both by the classic and convex NMF, by k-means, and also by three independent multivariate visualization methods.

5. Analysis of the sonar data

The data were obtained from waveform signals elaborated by Fourier transforms. The return of each signal is characterized by 60 parameters (variables) obtained from the respective power spectrum. As such, the recorded data - by their nature - are non-negative. They are reals contained in the interval [0, 1].

(a) Sonar. Classic and convex NMF

We proceed here similarly, as for the wine data. The results from the classic NMF are shown in Fig. 7.

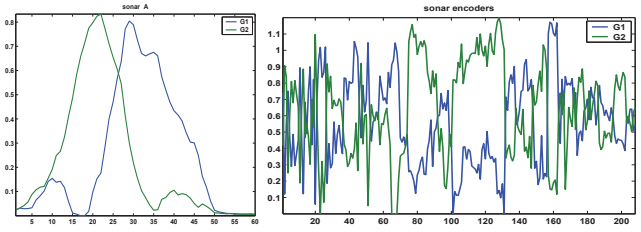

Fig. 7. Sonar by classic NMF. Left: Column profiles of the basis A. Right: Row profiles of encoder H.

One may notice there in the left exhibit that the basic vectors are fairy differentiated. The encoders - except the middle part for no-s 60–120 are very soft and do not yield any useful information with respect of clustering the rock and the mine objects.

The convex NMF yields similar results - what concerns the base vectors and the encoders. They are shown in Fig. 8. The composing of basic vectors from the recorded data vectors is shown in Fig. 9.

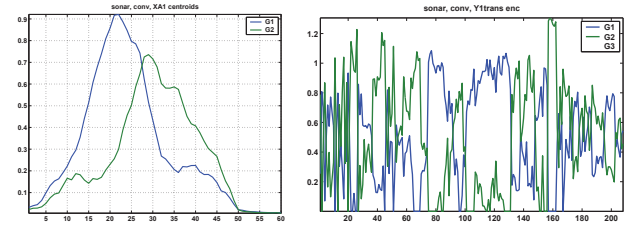

Fig. 8. Sonar by convex NMF. Left: Column profiles of basis A obtained as A=XW. Right: Row profiles of encoder H.

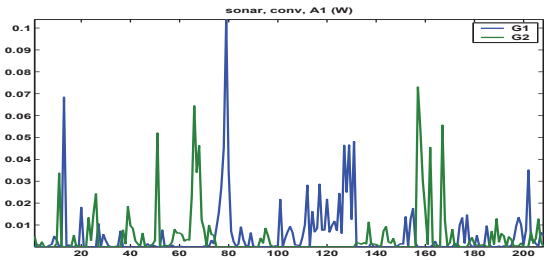

Fig. 9. Sonar by convex NMF. Column profile of weighting matrix W contributing for constructing the basis A=XW.

Looking at these figures it is clear that the eventual information on the indicated two clusters has very little in common with the pure clusters containing the two kind of objects.

(b) Sonar. Clustering by k-means for k=2

The results of *k-means* are shown in Fig. 10. At the left there are centroids of the clusters yielded by applying the k-means, and at the right the cluster assignments.

Comparing the profiles of the k-means centroids (left exhibit of Fig. 10) with the profiles of basis vectors yielded by the classic and the convex NMF (left exhibits in Fig. 8 and Fig. 9), one sees there a great similarity. There are two profiles shown in each exhibit and they have clearly differentiated shapes. The two profiles of the k-means centroids

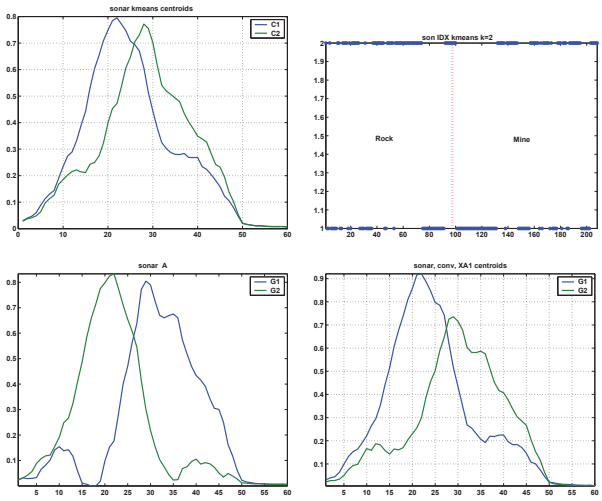

Fig. 10. Sonar. Results of k-means. Left: Profiles of centroids. Right: Group assignment by k-means. Bottom: Corresponding bases from both NMFs. Bottom left: The A base from classic NMF. Bottom right: The A base from convex NMF.

are little closer each other as those yielded by both NMFs. One might say that the centroids obtained by k-means are the most close ones, those from the convex NMF are a little bit more shifted each from the other, and the most separated are those obtained by the classic NMF.

What concerns the group assignments, one may see there some granularity, yet this has nothing in common with the external group assignment: rock or mine.

(c) Sonar. Exploratory visualization

We show here results obtained using using Kohonens' self-organizing maps, t-distributed stochastic neighbors embedding and canonical discriminant functions.

Fig. 11 shows the results displayed by Kohonen's self-organizing maps. Fig. 12 shows the visualization of the sonar data by the t-sne and CDF methods.

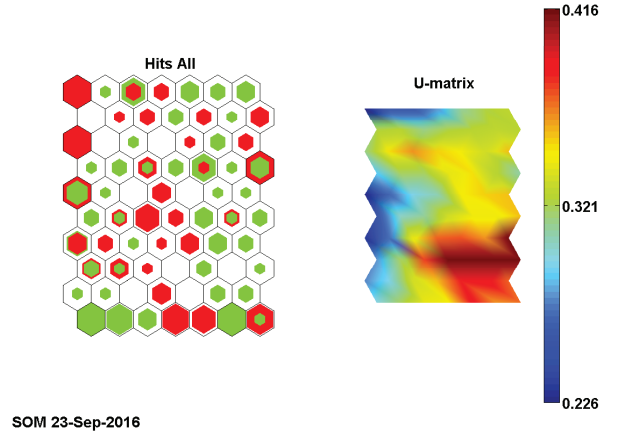

Fig. 11. Visualization of the sonar data using self-organizing maps of size 10×7. Left: Som-hits of the two groups of data 'rock' and 'mine' marked by colors red and green. Right: The topology of the data space where the sonar data vectors reside.

The map presented in Fig. 11 is composed from $10 \times 7 = 70$ hexagonal cells being in correspondence with the multivariate 60-D data space where the sonar data vectors reside. Nine of these hexagons are empty. Remaining 61 cells contain data vectors in variable amount. The fidelity of representation of the data vectors (data points) in the map is given by the quantization_error = 1.883, and the topological_error = 0.017. This means a fairy good representation of the vector quantization, and a very good representation of the topology of the data points. Unfortunately, what concerns the qualitative characteristics of the contents of the cells, both rocks and mines are there mixed together and no separated regions for rocks and mines can be observed.

Fig. 12, top exhibit, shows the 208 sonar data vectors visualized in 2D space using the t-sne method. Again, similarly as in Fig. 11, the rocks and mines are mixed together.

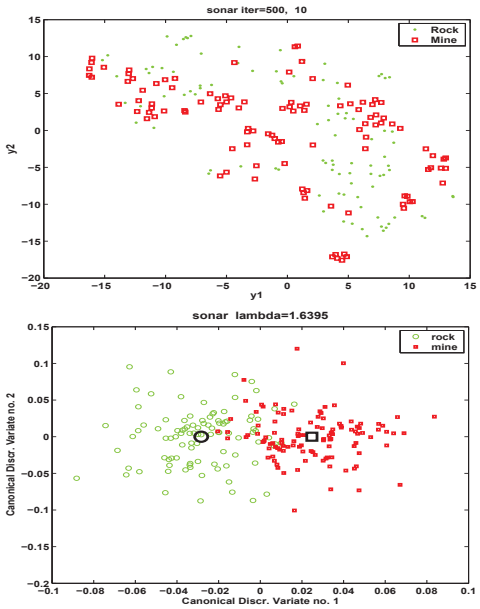

Fig. 12. Sonar data. Top: Visualization using t-sne. Points from the two groups are intermixed. Bottom: Visualization using canonical discriminant functions (CDA), that need information on group membership. The differentiation between groups is much better, as that when using t-sne, working without group membership information.

The bottom exhibit in Fig. 12 shows the 208 sonar data vectors visualized in the 2D space of canonical discriminant functions, constructed by Fisher's LDA, a supervised machine learning technique. Here the clustering effect is clearly visible, although the two groups are not fully separated.

The overall summary from this section is: The non-supervised methods like classic or convex NMF, k-means, t-sne do not recognize the group structure 'rock or mine' in the sonar data. However, the supervised canonical discriminant functions can do it fairly well.

6. Summary of the results and closing remarks

The NMF method was invented for decomposition of a real valued data matrix $\mathbf{X}$ as $\mathbf{X} = \mathbf{AH} + \mathbf{E}$, where the two factors $\mathbf{A}$ and $\mathbf{H}$ are real valued matrices usually of much smaller rank as $\mathbf{X}$, and $\mathbf{E}$ is the error matrix. All three involved matrices are non-negative. The decomposition has as goal dimensionality reduction which results in a parsimonious representation of the elaborated data. The methodology of NMF is in the eye of modern data analysis and is developing.

One of the actually trendy problems is: Can NMF serve also as a method for clustering? It follows from experimental research on various data sets, that for some of them the answer is 'yes', and for the others the answer is 'no'. Our question: is this a haphazard behavior?

Our thesis is the following: The clustering property of the NMF algorithm depends from the geometry of the data vectors/points observed in the multivariate data space. We should get insight into that space and observe there the topological neighborhood of the data vectors/points dealt with. If

we see there some clusters, then NMF will also indicate for them in the constructed decomposition. However, the clusters indicated by the NMF decomposition, may be different from those, we know from our knowledge based on some external sources we trust. In such a case other supervised machine learning technique may yield the sought clustering.

The thesis is demonstrated by the following experiment. We considered two multivariate benchmark data: wine (13 variables, 3 groups), and sonar (60 variables, 2 groups). For insight into the multi-variate data space we have employed three visualization techniques: two of them unsupervised (Kohonen's SOMs, and stochastic t-distributed embedding) and one supervised (CDA, i.e. Fisher's LDA).

Our analysis of the wine data resulted in the observation that all the applied visualization methods show quite clearly the subdivision of the wine data into 3 clusters corresponding to 3 vineyards the data come from. Also the k-means method sustains that observation. The overlap of the clusters is minimal. And this is sustained when working both with the classic and the convex NMF.

The results for the second investigated data set are quite different. The sonar data set is composed from two groups of data labelled 'rock' (1st group) and 'mine' (2nd group), and this is our external knowledge on these data. These groups have different geometry not recognized in the Euclidean data space. And neither classic nor convex NMF nor k-means recognize these groups. However, according [9], neural networks working in a supervised mode can do it.

In our investigation NMF has been working as a unsupervised linear method using the Least Square Error criterion. It does not recognize the two groups of the sonar data. Similarly, these sonar groups are not recognized by the unsupervised visualization methods (t-sne and SOM) and the k-means method. However, the supervised LDA (Canonical Discriminant Analysis), makes a big progress in the direction of recognizing the two groups of the sonar data.

Therefore our final conclusions are: Ordinary and convex NMF indicate only for a soft clustering of the data. The clustering indications in some cases may be so soft that they are practically useless; moreover, they are absolutely not conforming with the clusters we are concerned with. In such a case, we should switch to supervised learning. We may also try to define a different criterion of model fitness or add some external information on the data. Successful approaches of that kind are reported in the literature [7, 26, 25, 3].

***Author**: Ph.D. DSc. (hab) Anna Bartkowiak, retired profesor of Wrocław University, Institute of Computer Science, ul. Joliot-Curie 15, 50-383 Wrocław, Poland, email: aba@cs.uni.wroc.pl*

REFERENCES

[1] Bartkowiak A.M. , Zimroz R., NMF and PCA as applied to gearbox data, K. Jackowski, et al., (Eds): Intelligent Data Engineering and Automated Learning - IDEAL 2015 LNCS 9375, pp. 199-206, 2015.

[2] Bartkowiak A. and Szustalewicz A., Kernel Discriminant Analysis – a Practice Using the UCI Wine Data. In: J. Hartmann, J. Michalek, Eds, Biometricke metody a modely w soucasne vede a vyskumu, Sbornik referatu. UKZUZ Brno, 2006, pp. 45-56. ISBN 80-86548-89-9.

[3] Cichocki A., Zdunek R., Phan A.H., Amari Sh.: Nonnegative matrix and tensor factorizations. Applications to exploratory multi-way data analysis and blind source separation. Wiley, Chichester U.K. 2009.

[4] Ding C., He X.: Matrix factorization and spectral clustering. Proceed. SIAM Data Mining Conf., 2005.

[5] Ding C., Li T., Jordan MI: Convex and semi-nonnegative matrix factorizations. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI), 32, pp. 45-55, 2010. [web page]

[6] Everitt B.S. and Dunn G.: Aplied Multivariate Data Analysis. Arnold, New York, Toronto 1991.

[7] Fevotte C., Bertin N., Durrieu J-L.: Nonnegative matrix factorization with the Itakura-Saito divergence. With application to music analysis. Neural Computation 21 (3),pp. 793-830, 2009.

[8] Gillis N.: The why and how of nonnegative matrix factorization. In: J.A.K. Suykens et al. (Eds), Regularization, Optimization, Kernels and Support Machines. Chapman & HallCRC, Chapter 1, pp. 3-39, 2014.

[9] Gorman R.P., Sejnowski T.L.: Analysis of Hidden Units in a Layered Neural Network Trained to Classify Sonar Targets. Neural Networks, Vol. 1,pp. 75-89, 1988

[10] Gorman R.P., Sejnowski T.J.: Learned Classification on Sonar Targets Using a Massively Parallel Network. IEEE Trans. on Acoustics, speech, and Signal Processing, 16(7), pp. 1135-1140, July 1988.

[11] Hoyer P.O.: Non-negative matrix factorization with sparseness constraints. Journal of Machine Learning Research, 5, 1457-1469, 2004.

[12] Kohonen T.: Self-organizing Maps, Springer, Heidelberg, Third Extended Edition 2001, 501 pages.

[13] Krzyśko M., Wolynski W., et.al.: Systemy uczące się, rozpoznawanie wzorców, analiza skupień i redukcja wymiarowości (Self-learning systems, pattern recognition, cluster analysis and reduction of dimensionality), WNT, Warszawa, 2008.

[14] Lachenbruch P.: Discriminant Analysis, Hafner Press, London, 1975.

[15] Lee D.D., Seung H.S.: Learning the parts of objects by nonnegative matrix factorization. Nature 401, 788–791, 1999.

[16] Lee H., Cichocki A., Choi S.: Kernel nonnegativematrix factorization for spectral EEG feature extraction. Neurocomputing 72, pp. 3182-3190, 2009.

[17] Li T.,Ding C.: The relationships among various non-negative matrix factorization methods for clustering. Proceedings of the Sixth Int. Conf. on Data Mining, © IEEE, pp. 362-371, 2006.

[18] Li Y., Ngom A.: The non-negative matrix factorization toolbox for biological data mining. BMC Source Code for Biology and Medicine, **8**:(10), pp. 1–15, 2013. [web page] `https://sites.google.com/site/nmftool/`. [Accessed on 28 Oct. 2016.]

[19] Mannevaara M., Jilderin J.: Experiments on Gorman and Sejnowski sonar data. Manuscript 2001. [web page] `https://notendur.hi.is/benedict/Courses/sonar.pdf`. [Accessed on 16 Oct. 2016.]

[20] Marsland S.: Machine Learning, An Algorithmic Perspective. CRC Press, Taylor & Francis Group, Boca Raton London New York, Chapman & Hall, 2009.

[21] Okun O.G.: Non-negative matrix factorization and classifiers: experimental study. Proc. Intrn. Conf. on Visualisation, Imaging, and Image Processing (VIIP2004) Marbella, Spain, 2004.

[22] Tujaka A., Sparseness and locality in Nonnegative Matrix Factorization. Polish J. of Environ. Stud. 16, no. 5B, pp. 286-293, 2007.

[23] van der Maaten L., Hinton G., Visualizing data using t-SNE, Journal of Machine Learning Research 1, pp. 1-48, 2008.

[24] Vesanto J., et al., SOM Toolbox for Matlab 5, Som Toolbox Team, HUT, Finland. Libella Oy, Espoo, Version 0beta 2.0, pp. 1-54 November 2001.

[25] Zdunek R.: Nieujemna faktoryzacja macierzy i tensorów: Zastosowanie do klasyfikacji i przetwarzania sygnałów (Nonnegative matrix and tensor factorization, Application in classification and signal processing.) Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2014.

[26] Zdunek R.: Convex non-negative matrix factorization with Rank-1 update for clustering. L. Rutkowski et al. (Eds.): ICAISC 2015, LNAI 9120, pp. 59Ű-68, Springer-Verlag Berlin Heidelberg (2015).

[27] Zurada J.M., Ensari T., Asi E.H., Chorowski J.: Nonnegative matrix factorization and its application to pattern recognition and text mining. Proc. of the 13th Federated Conference on Computer Science and Information Systems, Cracow pp. 11–16, 2013.