Factorisation matricielle non-négative et classification d'images

MOHAMED BEN HAMDOUNE[1] et YANNIS TANNIER[1]

Université Paris-Descartes, 12 Rue de l'École de Médecine, 75006 Paris, France,
mohamed.ben_hamdoune@etu.parisdescartes.fr
yannis.tannier@etu.parisdescartes.fr
https://www.univ-paris5.fr/

Abstract. La classification automatique ou *clustering* consiste à partionner un ensemble d'objets (instances) décrits par un ensemble de variables en groupes (classes) homogènes. Avec l'avènement du *BigData* et de la science des données, le clustering est devenu une tâche encore très importante dans divers domaines dont l'imagerie. Les images sont des données très répandues notamment sur le web et les réseaux sociaux (Instagram, Pinterest, Flickr, Google,etc).Le but sera de proposer un systéme de classification pour des images provenant de divers bases de données (photos, peintures, bandes dessinés, etc). La factorisation matricielle non-négative permet d'approximer une matrice de données positive par le produit de deux matrices de dimensions inférieures et positives. Par sa simplicité, cette méthode est devenue populaire et est utilisée à la fois dans la réduction de la dimension et également dans la classification automatique (clustering) en un nombre de classes k fixé par l'utilisateur.

Keywords: Apprentissage Non Supervisé, Spherical K-means, K-means, R, NMF, Python, Scikit-Learn, Nimfa, NmfGpu4R, Cuda

1 Introduction

1.1 La factorisation matricelle non-négative

Cette section donne une définition formelle des problèmes de factorisation matricielle non-négative et définit les notations utilisées dans le cadre de ce TER (**Travail d'études et de recherches**).

Soit X une matrice non-négative $n \times p$, (i.e avec $x_{ij} \geq 0$, tel que $X \geq 0$), et $r > 0$ un entier positif. La factorisation matricielle non-négative (**NMF**) consiste à trouver une approximation

$$X \approx WH \ , \tag{1}$$

où W, H sont $n \times r$ et $r \times p$ matrices non-négativs respectivement. En pratique, le rang de factorisation r est souvent choisis de tel sorte que $r \ll \min(n, p)$. L'objectif derrire ce choix est de résumer et diviser l'information contenue dans X en facteurs r: les colonnes de W.

Selon le domaine d'application, ces facteurs ont des noms différents: images de base, métagènes, signaux source. Dans cette vignette, nous utilisons de manière équivalente et alternative les termes *base matrice* ou *metagènes* pour faire référence à la matrice W, et *matrice de coefficients de mélange* et *profils d'expression de métagène* pour se référer á la matrice H. La principale approche du **NMF** consiste à estimer les matrices W et H comme un minimum local:

$$\min_{W,H\geq 0} \underbrace{[D(X,WH)+R(W,H)]}_{=F(W,H)} \tag{2}$$

où

- D est une fonction de perte qui mesure la qualité de l'approximation. Les fonctions de perte communes sont basées soit sur la distance de Frobenius

$$D : A, B \mapsto \frac{Tr(AB^t)}{2} = \frac{1}{2}\sum_{ij}(a_{ij}-b_{ij})^2,$$

  ou la divergence Kullback-Leibler.

$$D : A, B \mapsto KL(A||B) = \sum_{i,j} a_{ij}\log\frac{a_{ij}}{b_{ij}} - a_{ij} + b_{ij}.$$

- R est une fonction de régularisation facultative, définie pour appliquer les propriétés sur les matrices W et H, telles que la *smoothness* ou la *sparsité* [2].

Sample Heading (Fourth Level) The contribution should contain no more than four levels of headings. Table **??** gives a summary of all heading levels.

1.2 Les méthodes de partitionnement

on parle de clustering

2 Construction de la matrice numérique

3 Estimation du rang de la factorisation

4 Méthodes d'initialisation

5 Classification

6 Conclusion

Un param´etre critique dans la **NMF** est le rang de la factorisation r. Il définit le nombre de variables utilisés pour approcher la matrice cible. Étant donné

une méthode **NMF** et la matrice cible, une faon courante de décider de r est d'essayer différentes valeurs, de calculer une mesure de qualité des résultats et de choisir la meilleure valeur en fonction de ces critres de qualité.

Several approaches have then been proposed to choose the optimal value of r. For example, [1] proposed to take the first value of r for which the cophenetic coefficient starts decreasing, [3] suggested to choose the first value where the RSS curve presents an inflection point, and [4] considered the smallest value at which the decrease in the RSS is lower than the decrease of the RSS obtained from random data.

Le package **NMF** provides functions to help implement such procedures and plot the relevant quality measures. Note that this can be a lengthy computation, depending on the data size. Whereas the standard NMF procedure usually involves several hundreds of random initialization, performing 30-50 runs is considered sufficient to get a robust estimate of the factorization rank [1, **?**]. For performance reason, we perform here only 10 runs for each value of the rank.

The result is a S3 object of class NMF.rank, that contains a data.frame with the quality measures in column, and the values of r in row. It also contains a list of the consensus matrix for each value of r.

All the measures can be plotted at once with the method, and the function generates heatmaps of the consensus matrix for each value of the rank. In the context of class discovery, it is useful to see if the clusters obtained correspond to known classes. This is why in the particular case of the Golub dataset, we added annotation tracks for the two covariates available ('Cell' and 'ALL.AML'). Since we removed the variable 'Sample' in the preliminaries, these are the only variables in the phenotypic data.frame embedded within the ExpressionSet object, and we can simply pass the whole object to argument annCol. One can see that at rank 2, the clusters correspond to the ALL and AML samples respectively, while rank 3 separates AML from ALL/T-cell and ALL/B-cell[1].

Overfitting Even on random data, increasing the factorization rank would lead to decreasing residuals, as more variables are available to better fit the data. In other words, there is potentially an overfitting problem.

In this context, the approach from [4] may be useful to prevent or detect overfitting as it takes into account the results for unstructured data. However it requires to compute the quality measure(s) for the random data. The NMF package provides a function that shuffles the original data, by permuting the rows of each column, using each time a different permutation. The rank estimation procedure can then be applied to the randomized data, and the random measures added to the plot for comparison.

[1] Remember that the plots shown in come from only 10 runs, using the 200 first genes in the dataset, which explains the somewhat not so clean clusters. The results are in fact much cleaner when using the full dataset.

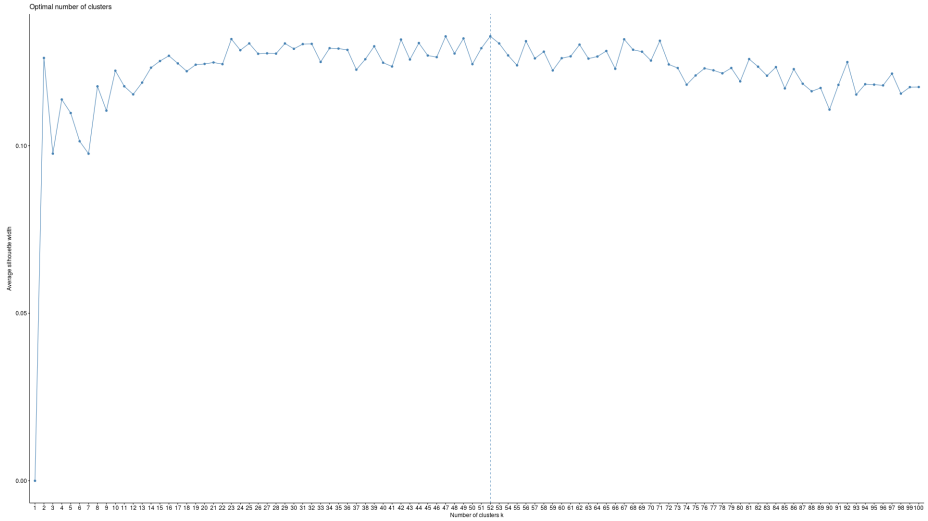

Fig. 1. La largeur moyenne de la silhouette en utilisant K-means pour un intervalle de 1 á 100, le nombre de clust optimal est 51.

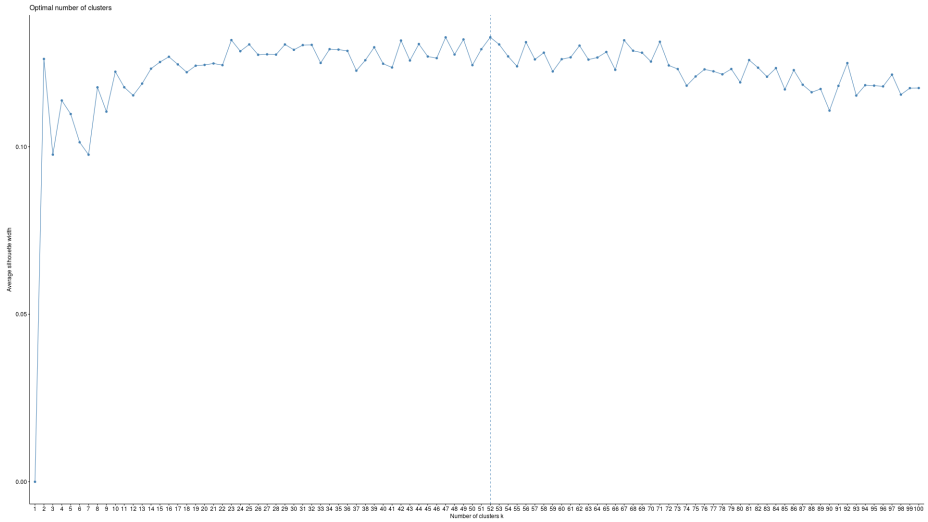

Fig. 2. La largeur moyenne de la silhouette en utilisant K-means pour un intervalle de 1 à 100, le nombre de clust optimal est 52 pour des blocs 16x16 avec un filtre gris.

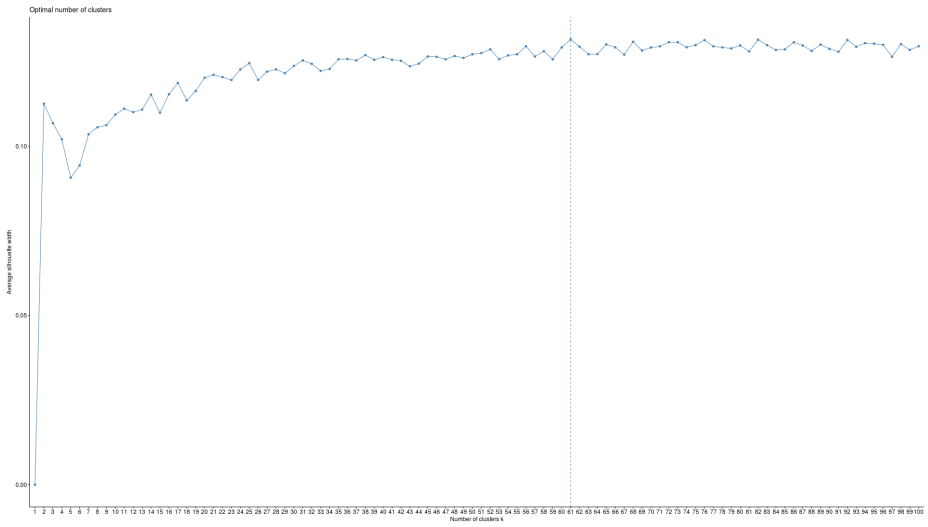

Fig. 3. La largeur moyenne de la silhouette en utilisant K-means pour un intervalle de 1 à 100, le nombre de clust optimal est 61 pour des blocs 32x32 avec un filtre gris.

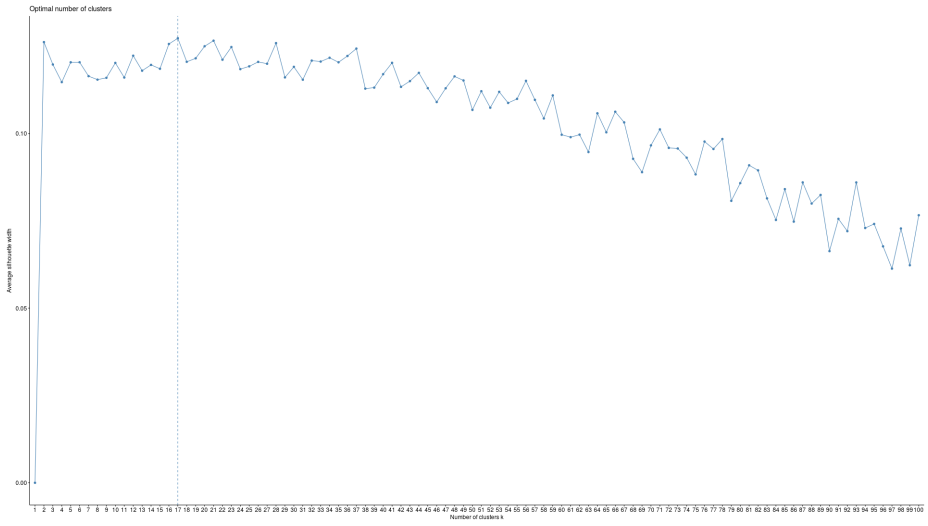

Fig. 4. La largeur moyenne de la silhouette en utilisant Spherical K-means pour un intervalle de 1 à 100, le nombre de clust optimal est 61 pour des blocs 16x16 avec un filtre gris.

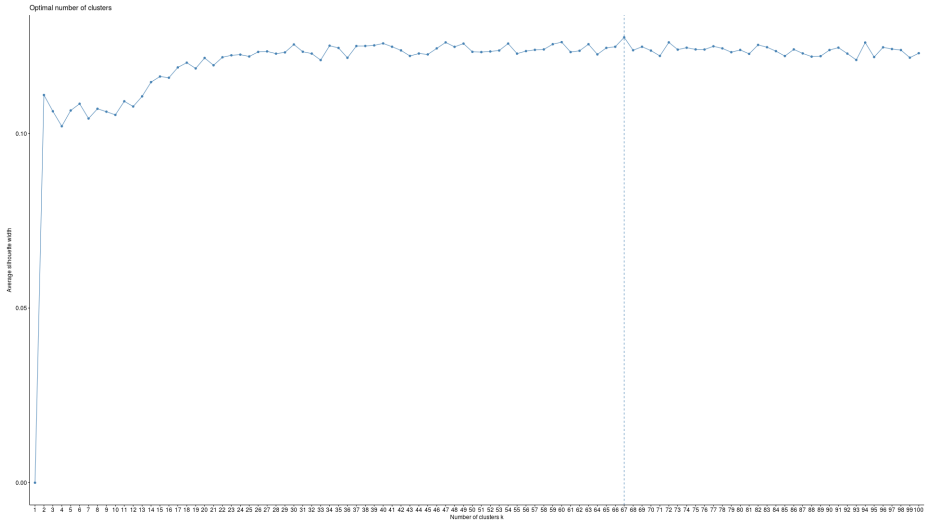

Fig. 5. La largeur moyenne de la silhouette en utilisant Spherical K-means pour un intervalle de 1 à 100, le nombre de clust optimal est 67 pour des blocs 32x32 avec un filtre gris.

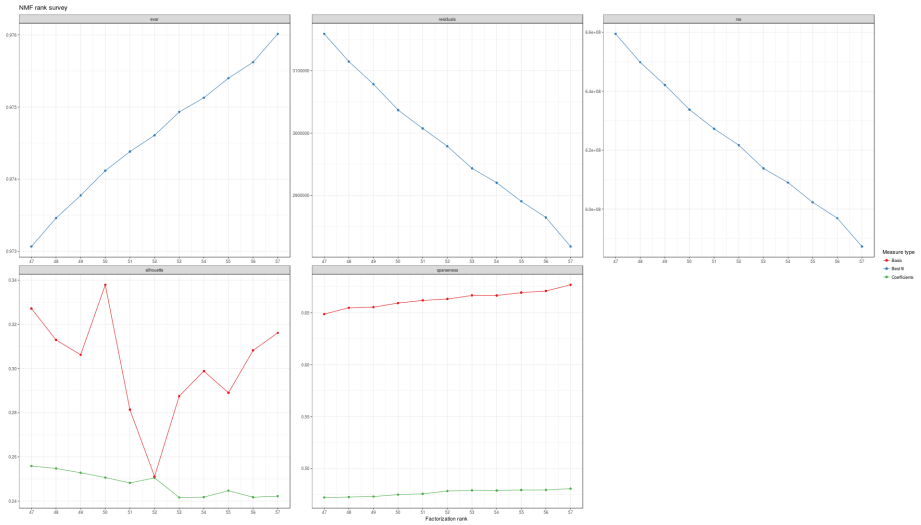

Fig. 6. Diffrents résultats significatifs pour 16x16 avec un filtre gris, l'intervalle de 47 57 avec l'algorithme KL et une initialisation avec une NNDSVD.

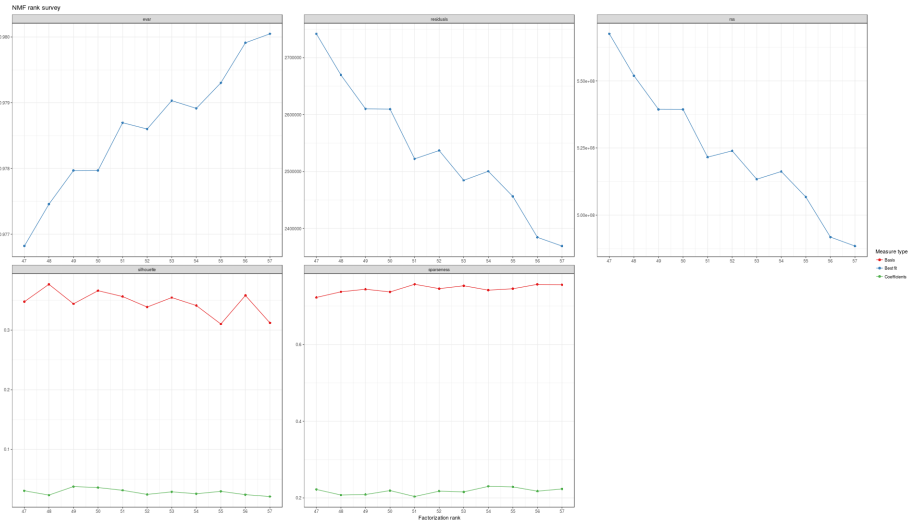

Fig. 7. Diffrents résultats significatifs pour 16x16 avec un filtre gris, l'intervalle de 47 57 avec l'algorithme KL et une initialisation de manire alatoire.

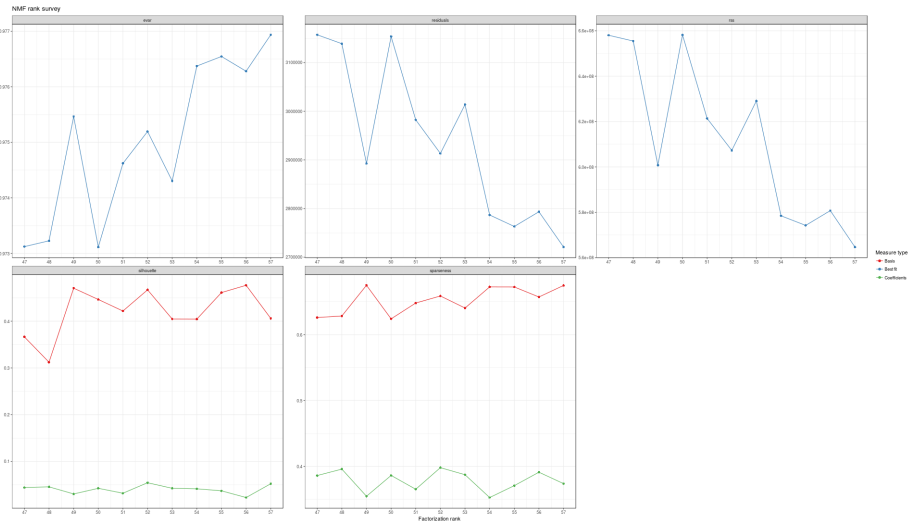

Fig. 8. Diffrents résultats significatifs pour 16x16 avec un filtre gris, l'intervalle de 47 57 avec l'algorithme KL et une initialisation avec une ICA.

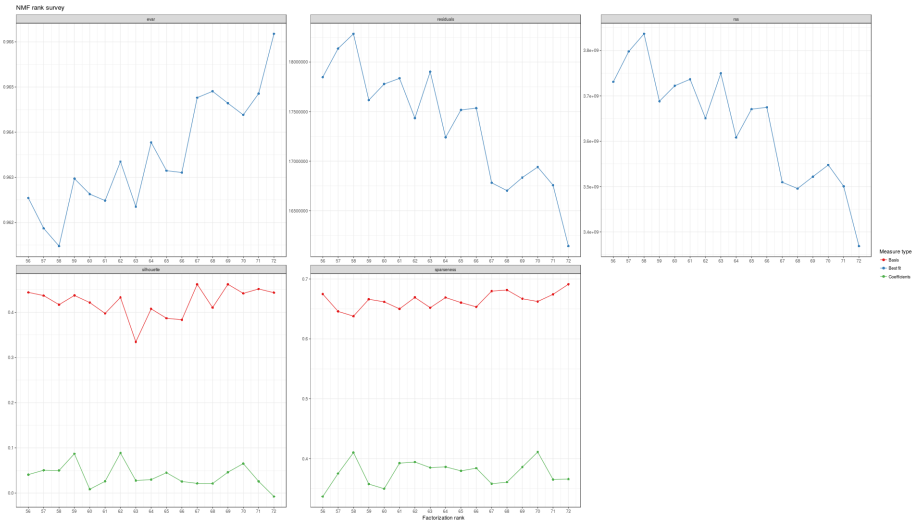

Fig. 9. Diffrents résultats significatifs pour 32x32 avec un filtre gris, l'intervalle de 56 72 avec l'algorithme KL et une initialisation avec une ICA.

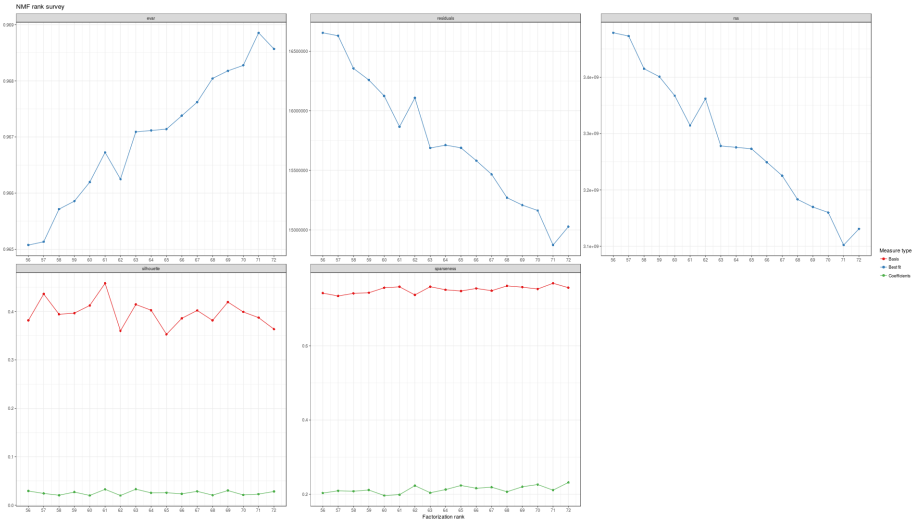

Fig. 10. Diffrents résultats significatifs pour 32x32 avec un filtre gris, l'intervalle de 56 72 avec l'algorithme KL et une initialisation avec de manire alatoire.

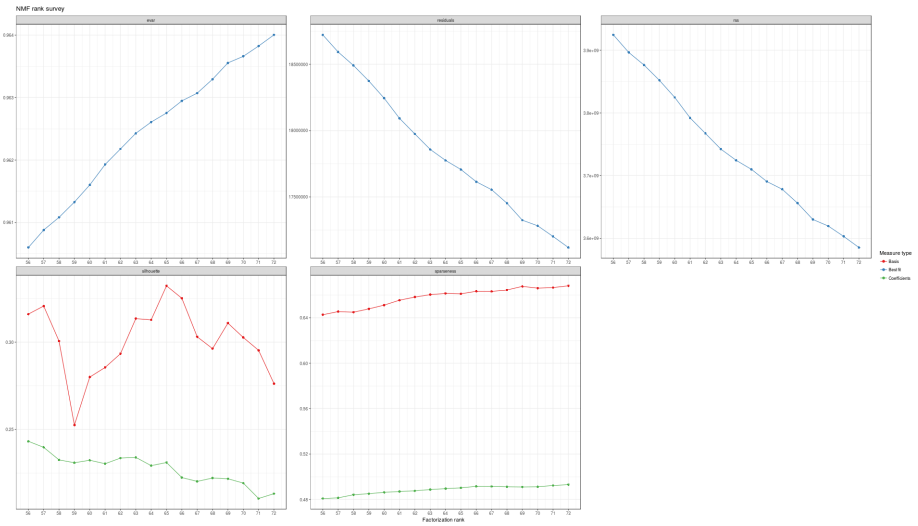

Fig. 11. Diffrents résultats significatifs pour 32x32 avec un filtre gris, l'intervalle de 56 72 avec l'algorithme KL et une initialisation avec une NNDSVD.

References

1. Author, F.: Article title. Journal **2**(5), 99–110 (2016)
2. Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) CONFERENCE 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016). https://doi.org/10.10007/1234567890
3. Author, F., Author, S., Author, T.: Book title. 2nd edn. Publisher, Location (1999)
4. Author, A.-B.: Contribution title. In: 9th International Proceedings on Proceedings, pp. 1–2. Publisher, Location (2010)
5. LNCS Homepage, http://www.springer.com/lncs. Last accessed 4 Oct 2017