

Unsupervised learning: from big data to low-dimensional representations

Vincent MATTHYS, Pirashanth RATNAMOGAN, Othmane SAYEM

PROJECT 2

Face Clustering and Motion Segmentation Challenge

Vincent MATTHYS
Pirashanth RATNAMOGAN
Othmane SAYEM

{vincent.matthys, pirashanth.ratnamogan, othmane.sayem}@ens-paris-saclay.fr

MASTER MVA
ENS PARIS SACLAY

UNSUPERVISED LEARNING
Instructor : René VIDAL

- 0 This project is to be done in MATLAB or Python and in groups of three students. Please report which student did which part. Each student should also email me a grade (0-100%=best) for the work of his/her mates.**

We did the projet in Python. We have tried to share the work fairly. Hence during the first phase of the project, Vincent focused in the implementation of the Spectral Clustering and in the Clustering error while Pirashanth worked in the Sparse Subspace Clustering with noisy data implementation and Othmane worked in the K-Subspaces algorithm implementation. Because we have faced lot of troubles during the implementation we helped each other in view of successfully produce operational implementations of the three algorithms. After this first part made essentially of implementations and debugging, Othmane and Pirashanth dealed essentially with the Face clustering (question 2) and Vincent worked on Motion Segmentation (question 3).

- 1 Clustering Algorithms. Implement the Spectral Clustering (SC) algorithm (Algorithm 4.5), K-Subspaces algorithm (Algorithm 6.1) and Sparse Subspace Clustering (SSC) algorithm with noisy data (Algorithm 8.5 and 8.6). You can use the MATLAB kmeans function (or python) inside SC.**

We have implemented the three algorithms and an implementation of the "best" precision rate using the hungarian algorithm implemented within the function.

We also tried to improve the outcome that we obtained in the face clustering challenge, by using the methods described in the book "Generalized Principal Component Analysis". Indeed, we tried to use a preprocessing using PCA when applying the K-subspaces algorithm (that allows to cut the computation time) and we try to use SSC with sparse outlying data (we doesn't succeed to make it converge properly).

2 Face clustering

- 2.1 Report the clustering error for individuals 1-2 for different choices of the parameters σ and K for SC (when constructing a Gaussian affinity with K-NN), number of restarts for K-subspaces, and λ for SSC.**

Here we report the clustering errors for individuals 1-2 for different choice of the parameters in the three algorithms that we have implemented.

k	sigma	error rate
10	100	49%
10	500	49%
10	1000	49%
10	10000	49%
10	100000	50 %
5	50	49%
5	100	37.5%
5	500	49%
5	1000	49%
3	100	43.75%
3	500	46%
3	1000	48.43%
1	100	49%
1	500	48%
1	1000	48 %

TABLE 1 – Clustering precision for individuals 1-2 using Spectral Clustering

d	replicates	error rate
3	1	48.4%
3	2	6.2%
3	3	5.4%
2	3	25.8%
4	3	49.2%
5	3	50%
6	3	49.2%
10	3	46.8%

TABLE 2 – Clustering precision for individuals 1-2 using K-Subspaces

τ	μ_2	error rate
1	500	46%
0.2	500	45%
10^{-4}	20	33.5%
10^{-4}	30	33.5%
10^{-4}	40	33.6%
10^{-4}	50	33.6%
10^{-5}	20	10.1%
10^{-5}	30	8.5%
10^{-5}	40	8.5%
10^{-5}	50	8.5%
10^{-5}	1000	8.6%
10^{-6}	$D^{1/2}$ not invertible

TABLE 3 – Clustering precision for individuals 1-2 using SSC

As we can see parameters choice have an important role in the algorithms performance. To set smartly the parameters we can use some basic rule of thumbs :

1. For the SC one can set the parameter σ using the intra-cluster standard deviation
2. For the K-Subspaces, the higher the number of replicates is, the better the final outcome will be. However because each restart are costly we will never exceed a number of 3 restarts. As described in class images of the same face under varying illumination lie near an approximately nine-dimensional linear subspace. However we get better outcome by setting this linear subspace very low. Indeed it allows to find a more discriminative model. The big dimension of each vector (D) makes the algorithm costly in terms of computation time, one alternative would be to reduce the dimension using PCA, by taking only d principal components (depends on the dimension of clusters) and applying the algorithm to the resulting data matrix.
3. For the SSC algorithm, using **Lemma 8.28** we set the τ as a multiple of the calculated τ_{\min} . For the parameter μ_2 , we used a large choice of values then deduced an interval of values that assures the best performance.

The Tables 1,2 and 3 allow to have an overview of how our three algorithms could perform in the face clustering task.

Spectral Clustering is performing really poorly in the face clustering task. That could be explain, by the fact that to compute the affinity matrix we are using K-NN based on the euclidean distance. This distance isn't the more adequate for this kind of high dimensional data.

K-subspaces and SSC are performing well in some cases but the error rate is really sensitive to how we set the parameters.

Because all the tests that we did are really time consuming, we couldn't do the same type of test for each subset of individuals 1-10, and 1-20, 1-30, 1-38. Hence, as suggested we will set the parameters for the remaining tests that we have to compute. We will take the parameters that better performs in the part with the individuals 1-2.

2.2 Once you have chosen the best parameters for each method, report the clustering errors for individuals 1-2, 1-10, and 1-20, 1-30, 1-38, and analyze the clustering error as a function of the number of groups.

Here we report the clustering errors for individuals 1-2, 1-10, and 1-20, 1-30, 1-38. In those computations, the parameters are set by using the parameters that performs best in the clustering for individuals 1-2.

nb of individuals	error rate
1-2	37.5%
1-10	80%
1-20	86%
1-30	89%
1-38	88%

TABLE 4 – Clustering precision for various number of individuals using SC

nb of individuals	error rate
1-2	5.4%
1-10	64%
1-20	70%
1-30	72%
1-38	75%

TABLE 5 – Clustering precision for various number of individuals using K-subspaces

nb of individuals	error rate
1-2	8.5%
1-10	56%
1-20	65%
1-30	67%
1-38	-

TABLE 6 – Clustering precision for various number of individuals using SSC

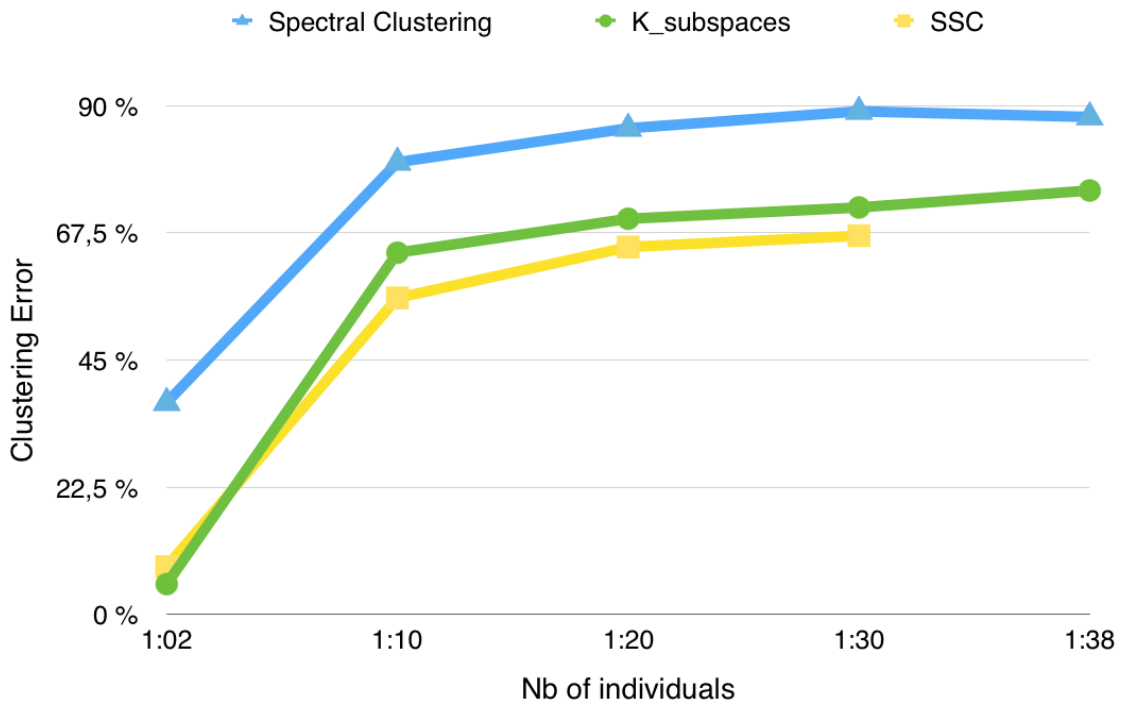


FIGURE 1 – Clustering error for each method, applied to various numbers of compared individuals

As we can see in the Table 4,5,6 the performance of the three algorithms are not quite good. Yet, Figure 1 for the chosen parameters, we notice that for all numbers of individuals, SSC gets the best results compared to the two other methods. Unsupervised face clustering is a really complex task. One can remember the following plots presented by R.Vidal :

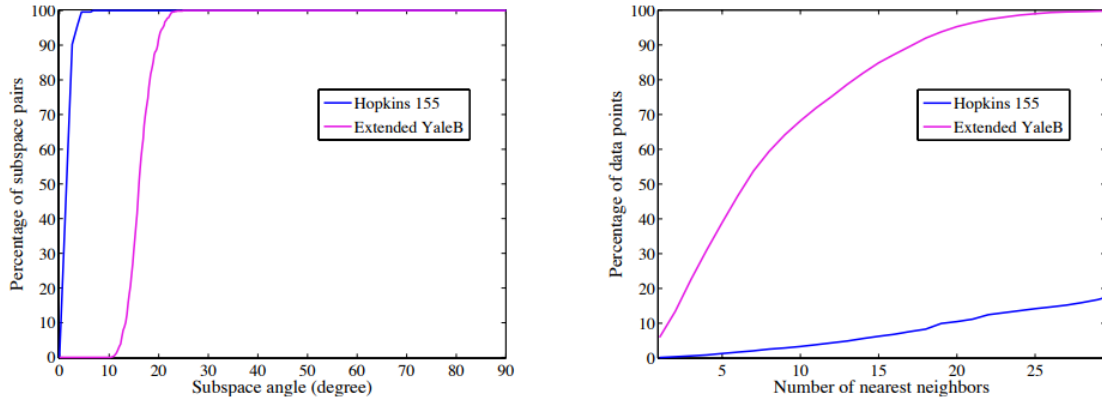


FIGURE 2 – Challenge face clustering, from R.Vidal [1] page

We have two difficulties in the clustering task in Extended YaleB database : angles between subspaces are small, Nearby points are in different subspaces.

These difficulties make that it will be almost impossible for the SC and the K-subspaces algorithms to perform well because they are based on these points. Hence, before running the tests, we could only hope the SSC with noising data algorithm to perform well. K-subspaces, perform well with individuals 1-2 but its performance decreases when increasing the number of individuals. SSC also performs badly even if it produces the best outcome, as shown in the Figure1. As suggested in the book, another possibility is to use the SSC with sparse outlying entries that seems to perform well in the face clustering task.

3 Motion Segmentation

By applying the algorithms in section 1 to the feature point trajectories from the Hopkins155 dataset, we aim to quantify the results obtained. The clustering error for different choices of the parameters σ and K for SC (when constructing a Gaussian affinity with K-NN), number of restarts for K-subspaces, and λ for SSC are compared next.

The figure 2 shows the difference in terms of neighborhood between the face clustering problem and the motion segmentation problem. One can expects the parameters and the results to be different in both cases. The percentage of sequences for which the clustering error is below a given value is represented to quantify the results obtained by qualitatively looking at the area under this curve. In order to perform the clustering part, the algorithms were given the true number of clusters, available in the Hopkins155 dataset as the maximum of the label in each *ground_truth.mat*.

3.1 Spectral clustering

For the spectral clustering, the influence of K in the computation of the affinity matrix is presented in figure 3. The area under the curve is larger with K , except for the case $K = 1$, but the effect decrease with K . Moreover, two different behaviours can be observed : $K \leq 3$ and $K \geq 4$ where a substantial early gap in term of misclassification appears. To conclude, as long as the complexity allows it, a bigger choice of K has to be preferred.

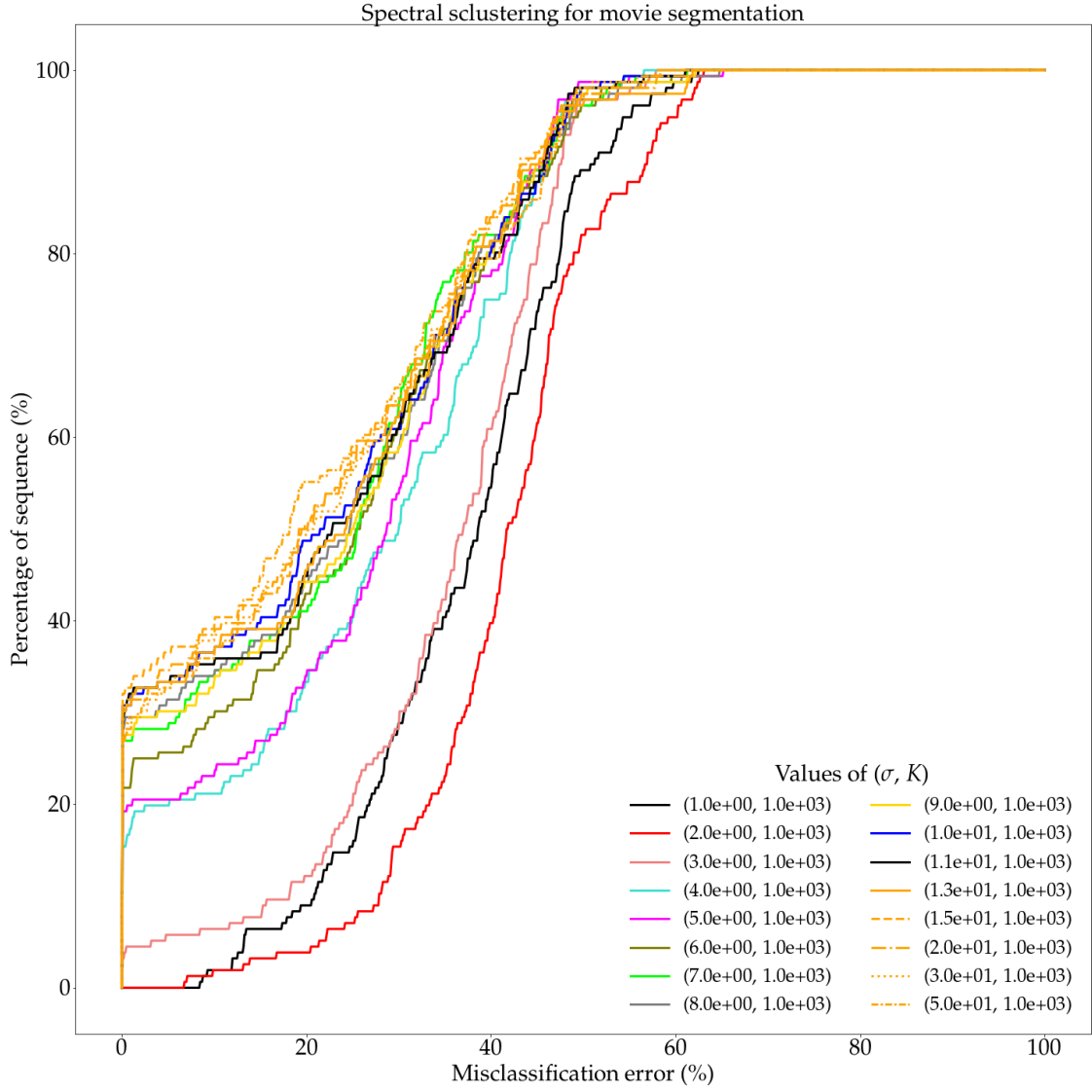


FIGURE 3 – Case of spectral clustering, with different parameters K for the computation of the affinity matrix. The percentage of sequences for which the clustering error is less than or equal to a given percentage of error is represented in different colors corresponding to a given value of K

The influence of σ is studied in figure 4. Here, two different values of K are represented for $\sigma \in \{10^i\}_{i=0..5}$. There is no noticeable behaviour here, and we can approximately say that the major parameter is K .

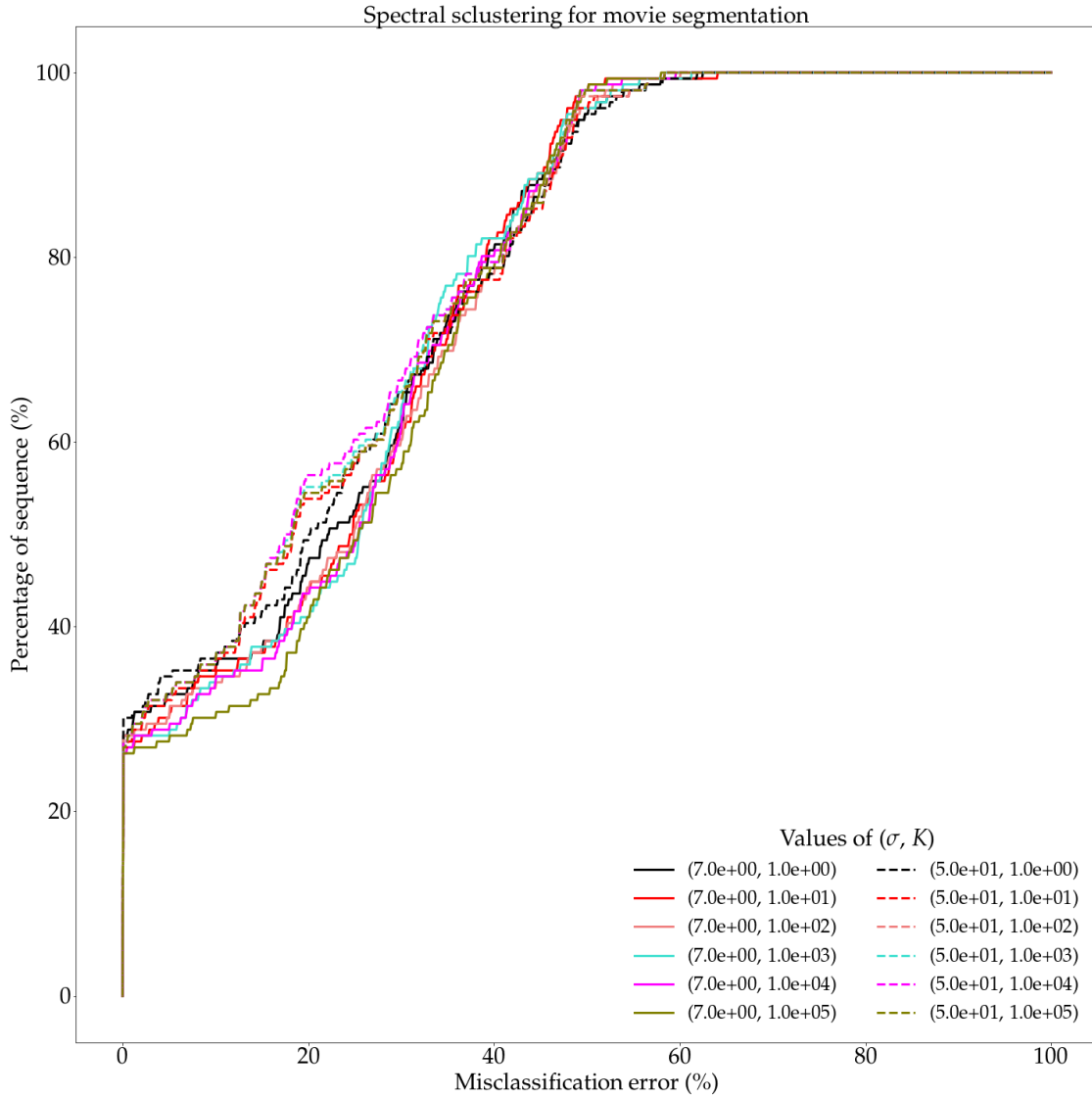


FIGURE 4 – Case of spectral clustering, with different parameters σ for the computation of the affinity matrix. The percentage of sequences for which the clustering error is less than or equal to a given percentage of error is represented in different colors corresponding to a given value of σ . Two values of K , 7 and 50 are represented.

3.2 Ksubspaces

When applying Ksubspaces to motion segmentation, the dimensions of the target clusters have to be known. We therefore can look at the singular values of the matrix of feature point trajectories, as proposed by R.Vidal in [1], to get a good approximation with 4 dimensional subspace for every motion. In case of motion segmentation the influence of the number of restarts is observed in figure 5. The single restart profile differs from the other, due to the randomness of the initialization in that case which is not counterbalanced by the number of restarts. The larger the number of restarts, the larger the area under the curve : it couldn't have been another way. To conclude, a largest number of restarts has to be preferred, to smooth the initial states and to get higher probabilities to get the global maximum.

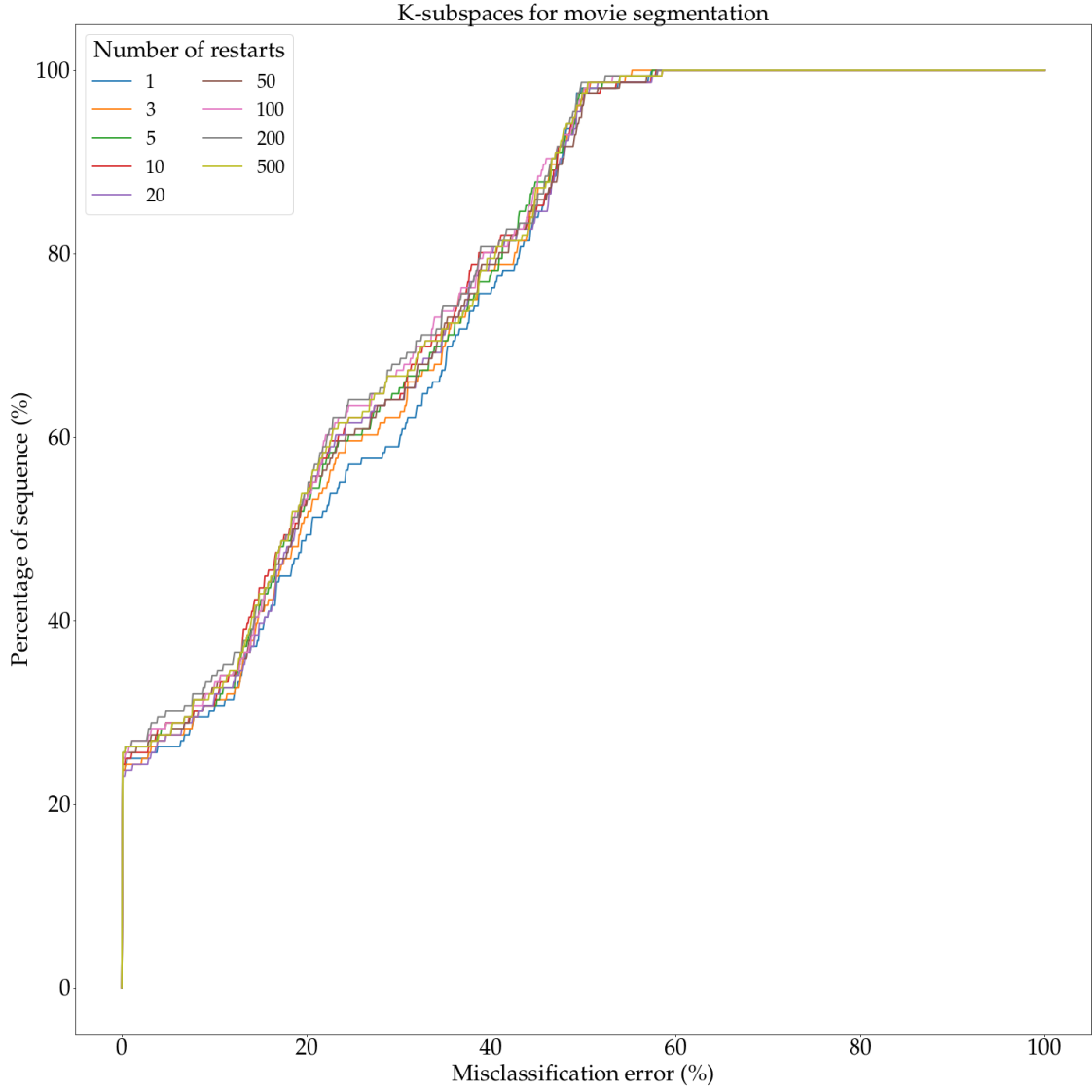


FIGURE 5 – Case of k-subspaces, with different number of restarts. The percentage of sequences for which the clustering error is less than or equal to a given percentage of error is represented in different colors corresponding to a given number of restarts.

3.3 Sparse spectral clustering

The influence of τ parameter for sparse spectral clustering applying to motion segmentation with the Hopkins155 data-set is observed in figure 6. The larger area under the curve is obtained for $\tau = 10^4$ which is very different for the values obtained in the face clustering problem, with 10^9 order of magnitude.

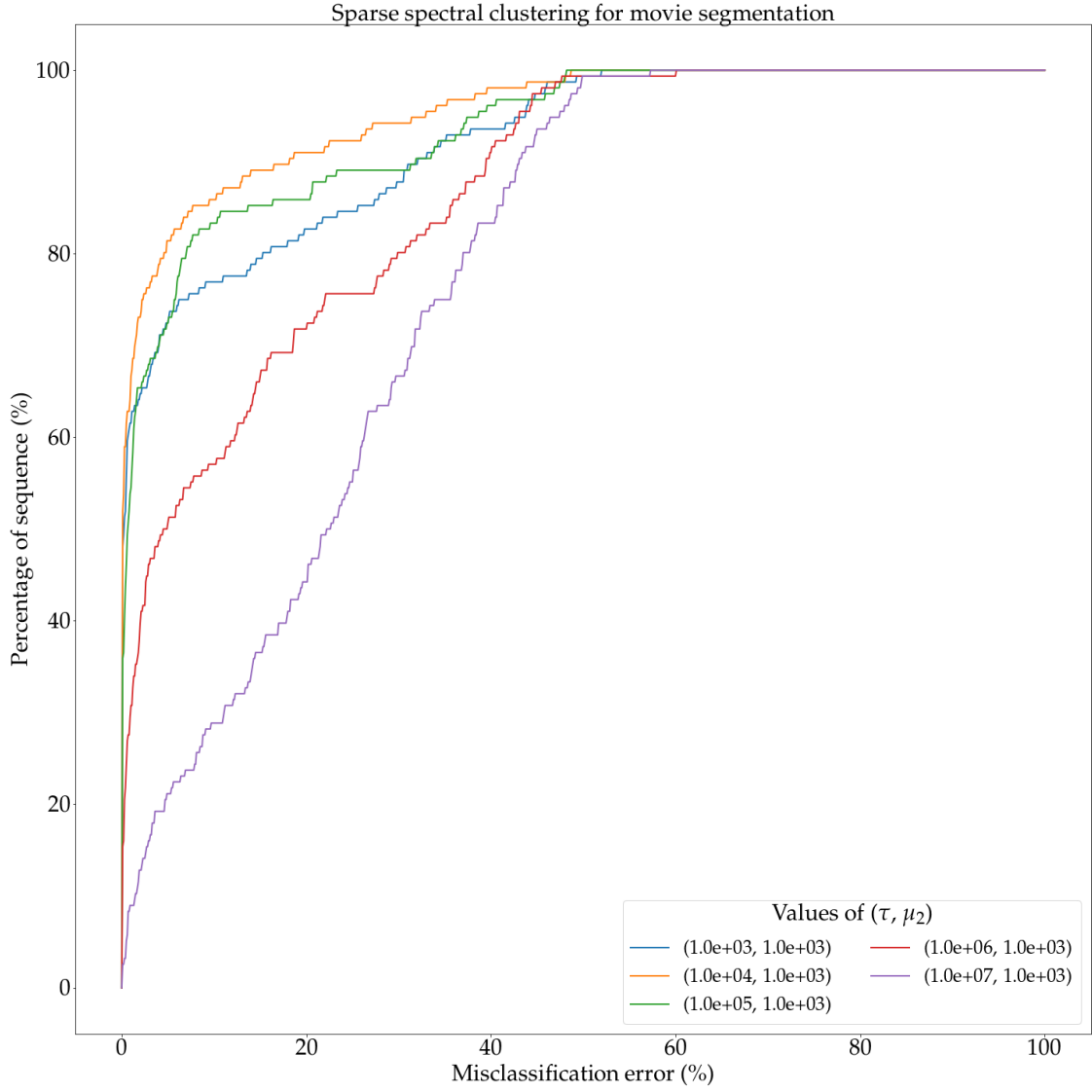


FIGURE 6 – Case of sparse spectral clustering for $\mu_2 = 10^3$ and for different values of τ . The percentage of sequences for which the clustering error is less than or equal to a given percentage of error is represented in different colors corresponding to a given τ .

For this specific value of τ we looked for the best μ_2 parameter. The results obtained are visible in figure 7. We can clearly distinguish the profile obtaining the higher area under the curve, for $\mu_2 = 10^4$. It leads to our conclusion, that optimal parameters for SSC applied to motion segmentation, based on this dataset, is $\tau = 10^4, \mu_2 = 10^4$.

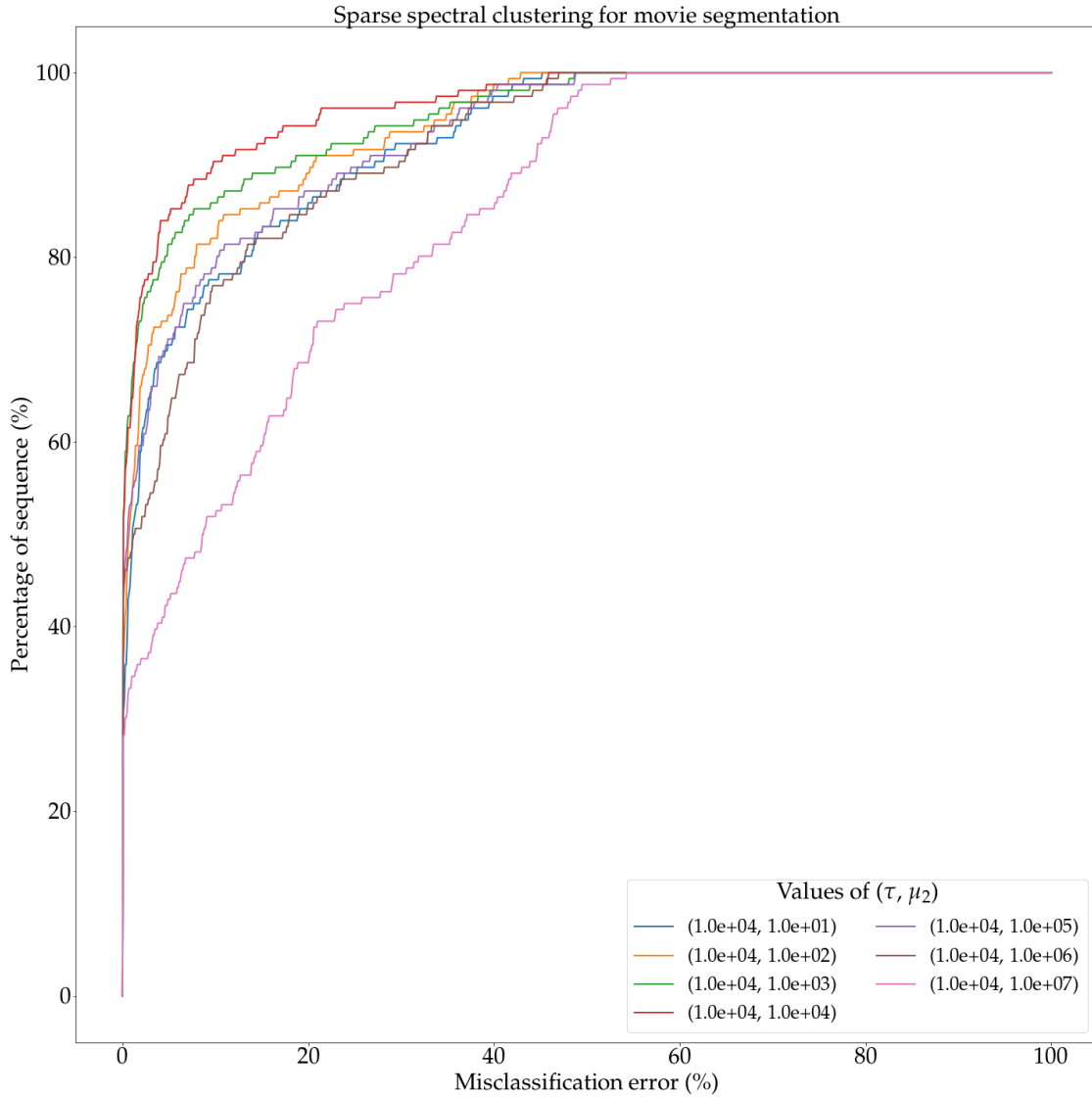


FIGURE 7 – Case of sparse spectral clustering for $\tau = 10^4$ and for different values of μ_2 . The percentage of sequences for which the clustering error is less than or equal to a given percentage of error is represented in different colors corresponding to a given μ_2 .

3.4 Comparison

Finally, the best parameters for the three algorithms are chosen, and the resulting curves are presented in figure 8. It's clear that the only algorithm presenting $AUC \gg 0.5$ is the SSC one. In fact, as we have seen in figures 6, 7, the only algorithm with such an AUC is SSC, supposing that the spatial data structure is not convenient for the two others clustering algorithms. In fact, if SSC is working, it does mean that the subspaces are independants, or disjoint, and that the dimension of the subspaces are very closed. On the other hand, as the local distance, on which is based spectral clustering fails to capture structure nearby closed subspaces, we could expect the poor results obtained by this algorithm, as the subspace angles, shown in figure 1 in Hopkins155 data-set are really small in comparison with the Extended YaleB dataset.

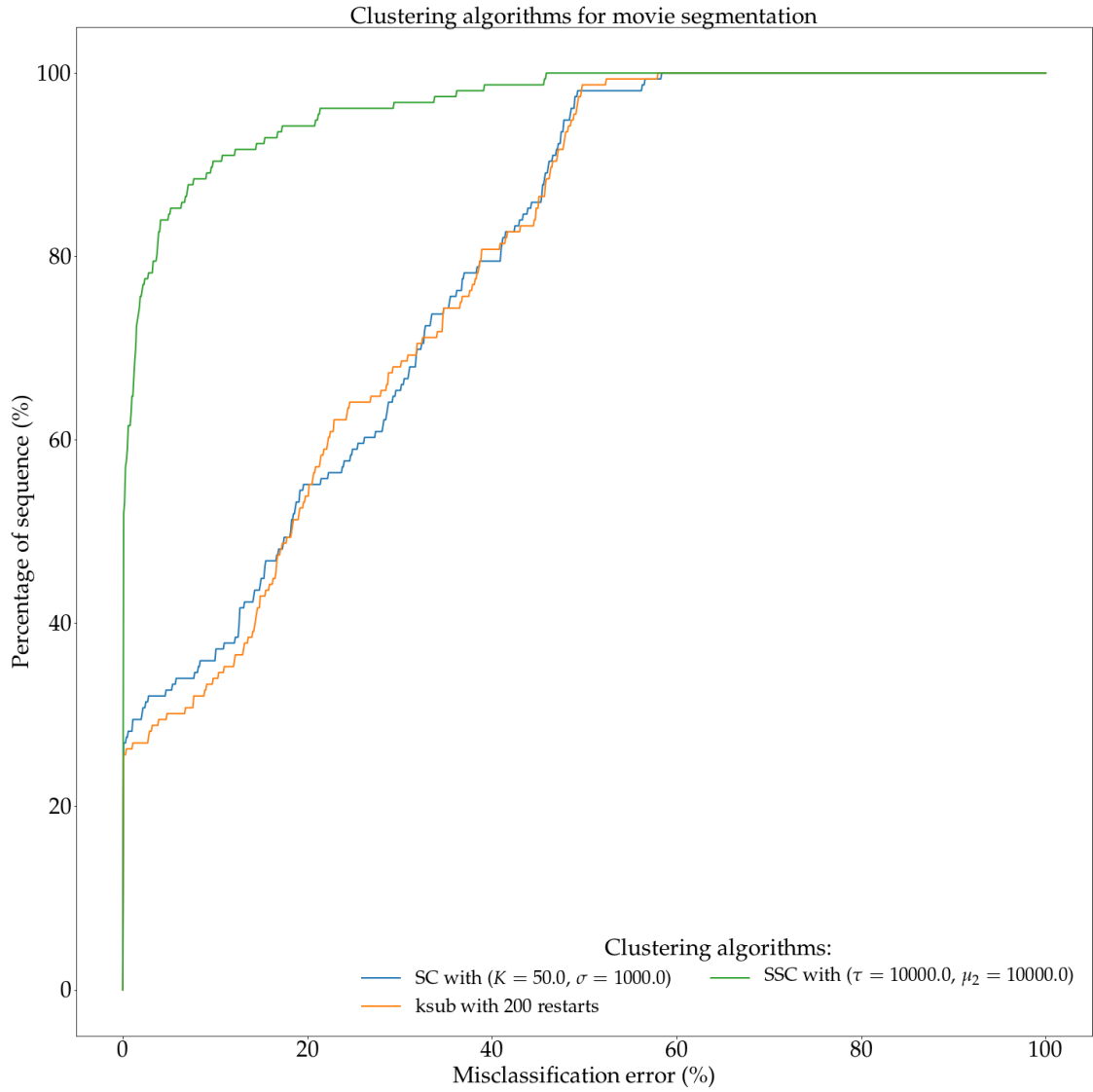


FIGURE 8 – Applications to motion segmentation with the best parameters for three clustering algorithms. The percentage of sequences for which the clustering error is less than or equal to a given percentage of error.

Références

- [1] R. VIDAL, Y. MA et S. S. SASTRY, « Principal Component Analysis », in *Generalized Principal Component Analysis*, Springer, 2016.