Parameter-free classification of the Cosmic Web using the k-means algorithm

Edoardo Carlesi

December 11, 2020

1 Introduction

The classification of the different environmental types in cosmological simulations and observations is a problem that can be approached using several techinques. So far, there is no widespread consensus on which classification scheme works better and also which kind of metric should be used to gauge its goodness.
Here we will use the shear velocity tensor $\Sigma_{\mu\nu}$ to classify the environmental types that make up the cosmic web (the so called V-Web), which is defined as:

$$\Sigma_{\mu\nu} = -\frac{1}{H_0}\left(\frac{\partial v_\nu}{\partial x_\mu} - \frac{\partial v_\mu}{\partial x_\nu}\right) \tag{1}$$

and its three eigenvalues ($\lambda_1 > \lambda_2 > \lambda_3$) are the building blocks of our classification schemes. So far, the approach has been to identify four different environmental classes introducing a threshold λ_{th}

- $\lambda_1 < \lambda_{th}$ for voids
- $\lambda_2 < \lambda_{th}$ and $\lambda_3 > \lambda_{th}$, for sheets
- $\lambda_2 > \lambda_{th}$ and $\lambda_1 < \lambda_{th}$, for filaments
- $\lambda_3 > \lambda_{th}$ for knots

where each eigenvalue is associated with one direction of collapse (expansion) if its value is above (below) a given threshold λ_{th}. However, this threshold is a priori unknown and its value can be hard to justify on physical grounds.

This is what drives us to look for alternative, parameter-free classification methods that might help distinguishing between different environments without relying on specific values of λ_{th}. In what follows, we will describe the results based on one of the most popular *unsupervised learning* methods, the k-means clustering.

2 k-means clustering

The k-means algorithm is a method that can be applied to a dataset for classification purposes. Given a set of N points $x_1, ..., x_N$, each of which has m features, we aim at subdividing the data into k classes by looking at the clustering of the points in m-dimensional space (in general also called feature space or, in our specific case, λ space). What the k-means algorithm does is to identify a set of k-centers, where k is a parameter that needs to be specified by the user.

After randomly initializing the initial centers' positions in m-dimensional feature space, the algorithm proceeds iteratively with the following steps:

- For each of the k centers, compute the Euclidean distance of each data point x_p
- Assign each point x_p to the closest cluster center
- Once all of the N points have been assigned, re-compute each cluster center's position as the barycenter of all the points that were assigned to it

The number of iterations can be specified in advance or some convergence criterion can be implemented, and the algorithm also needs to be tested using different random initializations to ensure convergence of the results. In practice, defining as S_i the k target subsets and c_i as its center in m dimensional feature space, at each iteration t of the algorithm we have:

$$S_i^t = \{x_p : ||x_p - c_i^t||^2 \leq ||x_p - c_j||^2 \quad \forall j, 1 \leq j \leq k\} \tag{2}$$

and the cluster centers have to be recomputed as:

$$c_i^{(t+1)} = \frac{1}{N_{S_i^t}} \sum_{j=1}^{N_{S_i^t}} x_j \tag{3}$$

where $N_{S_i^t}$ is the number of points belonging to the i-th cluster at step t. Here we started assuming $k = 4$, just like in the λ_{th}-based approach and the subdivision into voids, sheets, filaments and knots; then moved to see the effects of assuming different k values, using several criteria to evaluate whether an optimal k can be established.

3 Dataset

The k means was tested on 5 constrained DM only simulations with the following properties

- 512^3 particles
- 100 h^{-1}Mpc box
- 128^3 grid, $1h^{-1}$Mpc smoothing length for the V-Web calculation

However, in our analysis we focused on a single simulation only. In fact, as shown in Table 1 and Table 2, the median values and standard deviations of the three eigenvalues as well as the volume filling fractions of the distinct kinds of environment across the simulations show a substantial quantitative agreement, with negligible quantitative difference from one another.
Thus, in what follows, we will focus on the results drawn from a single simulation, as the conclusions will not depend on the specific simulation selected. Some properties of this simulation - labeled SIMU04 - are shown in Table 3, where the median and standard deviations for the three eigenvalues as well as the median densities and the volume filling fraction are shown per environment type and per varying threshold values. We notice that

- $\lambda_1, \lambda 2, \lambda_3$ distributions in voids have always the smallest scatter around the mean
- median Δ values are increasing from voids to knots
- Knots and filaments consistently occupy a smaller fraction of volume than sheets and voids

Whereas these properties hold in the specific range of values chosen for λ_{th}, they are important for comparison with the results of k-means classification presented in the following section.

Table 1: Median and standard deviation values for $\lambda_1, \lambda 2, \lambda_3$ in the five different simulations.

	SIMU00	SIMU01	SIMU02	SIMU03	SIMU04
λ_1	0.203 ± 0.355	0.203 ± 0.299	0.201 ± 0.312	0.199 ± 0.347	0.205 ± 0.306
λ_2	-0.033 ± 0.193	-0.043 ± 0.174	-0.037 ± 0.181	-0.041 ± 0.189	-0.040 ± 0.191
λ_3	-0.309 ± 0.169	-0.308 ± 0.162	-0.301 ± 0.171	-0.311 ± 0.172	-0.297 ± 0.179

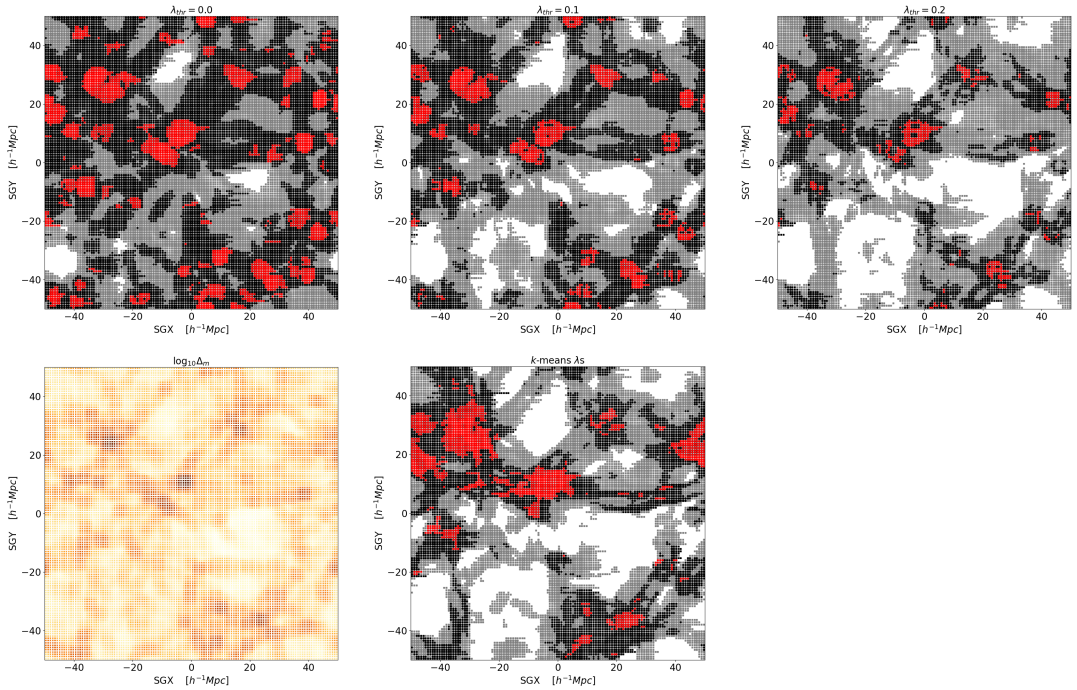
Table 2: Volume filling fractions of the four environment types defined using $\lambda_{th} = 0.2$ across the five simulations.

type	SIMU00	SIMU01	SIMU02	SIMU03	SIMU04
void	0.494	0.493	0.488	0.490	0.498
filament	0.387	0.406	0.395	0.401	0.397
sheet	0.109	0.093	0.107	0.099	0.101
knot	0.007	0.005	0.007	0.008	0.006

Table 3: Median and standard deviation values for $\lambda_1, \lambda_2, \lambda_3$, median Δ and volume filling fraction f obtained using three different threshold values: 0.0 (upper rows), 0.1 (central rows) and 0.2. (lower rows). The simulation is SIMU04.

Environment	λ_1	λ_2	λ_3	$\bar{\Delta}$	f
void	-0.051 ± 0.046	-0.177 ± 0.060	-0.316 ± 0.062	0.154	0.114
sheet	0.146 ± 0.198	-0.107 ± 0.078	-0.333 ± 0.104	0.356	0.456
filament	0.346 ± 0.369	0.099 ± 0.144	-0.283 ± 0.185	0.984	0.366
knot	0.572 ± 0.558	0.227 ± 0.253	0.070 ± 0.107	3.404	0.061
void	0.022 ± 0.066	-0.138 ± 0.082	-0.313 ± 0.084	0.230	0.293
sheet	0.232 ± 0.212	-0.039 ± 0.103	-0.326 ± 0.138	0.521	0.469
filament	0.500 ± 0.415	0.189 ± 0.154	-0.244 ± 0.234	1.415	0.213
knot	0.881 ± 0.663	0.376 ± 0.313	0.160 ± 0.128	4.060	0.021
void	0.077 ± 0.087	-0.107 ± 0.100	-0.311 ± 0.105	0.305	0.494
sheet	0.346 ± 0.240	0.027 ± 0.125	-0.320 ± 0.172	0.695	0.387
filament	0.698 ± 0.470	0.299 ± 0.172	-0.211 ± 0.277	1.668	0.109
knot	1.288 ± 0.768	0.586 ± 0.376	0.279 ± 0.154	3.615	0.007

Figure 1: Comparison of the SGX-SGY plane on a 128^3 grid, $\lambda_{thr} = 0.0, 0.1, 0.2$, matter density and k-means with $k = 4$



4 k-means classification: results

The k-means algorithm was first applied using 20 iterations and $k = 4$. The so-identified cluster centers are shown in Table 4 together with the median overdensity per cluster type $\bar{\Delta}$ as well as the fraction of volume occupied by each environment type f. Environment types have been assigned looking at $\bar{\Delta}$ ($\bar{\Delta}$ is defined as the ratio of ρ over the median density $\bar{\rho}$) in the following way: once the four clusters have been identified we computed $\bar{\Delta}$ for the cells belonging to them; and labeled them as voids, sheets, filaments and knots by increasing median Δ value. This allowed us to create some common ground for comparison with the results with the standard, threshold based classification of the V-Web. The distributions of Δ within each environmental class are shown in Fig. 3.

In Fig. 1 we show the two-dimensional projections on the X and Y axes of the matter density as well as the usual V-Web classification schemes with three different values for λ_{th}, $0.0, 0.1$ and 0.2, these plots are compared with the k-means results. Overall, we notice that the k-means clustering produces a pattern remarkably close to the λ_{th}= 0.2 V-Web. The correspondence between the k-means and the λ_{th} classification of the V-Web can be quantified for the three thresholds noting that for λ_{th}= 0.0, 31% of the nodes identify the same kind of environment; this share rises to 61% λ_{th}= 0.1 and 78% for λ_{th}= 0.2. In particular, in the latter case, 89% of the voids and 59% of the knots are identified in the same location in physical space.

We notice that the volume filling fraction follows the same pattern shown by the usual classification scheme, with a decreasing f value inversely correlated with median density. However, it is clear by looking at the clusters' positions in λ space and their distributions show in Fig. 4 that the

Figure 2: k-means classification of points in 3d λ space (128^3 grids). Red dots are associated with knots, black dots with filaments, grey dots with sheets and white dots with voids.

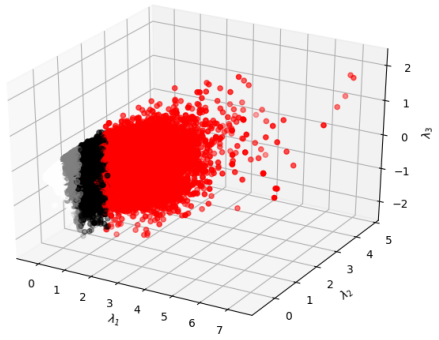


Figure 3: Matter density distributions for environment types identified using k-means and λs (grid = 128^3)

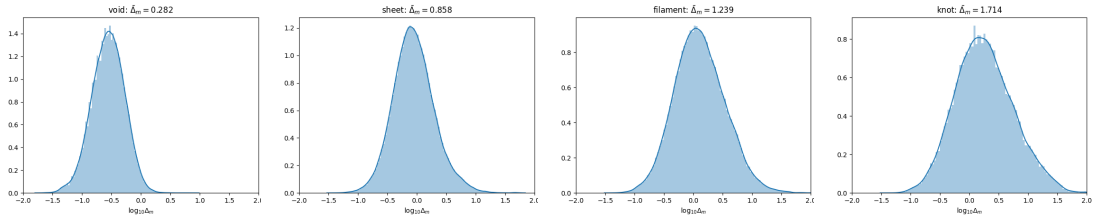


Table 4: Eigenvalue-only k-means, grid: 128^3. Cluster centers, median density $\bar{\Delta}$ value and volume filling fraction f for each kind of environment.

Environment	λ_1	λ_2	λ_3	$\bar{\Delta}$	f
void	0.075	-0.125	-0.324	0.283	0.488
sheet	0.331	0.052	-0.265	0.861	0.312
filament	0.801	0.212	-0.265	1.250	0.125
knot	1.739	0.545	-0.127	1.718	0.023

k-means algorithm does not produce a straightforward correspondence with the threshold based classification, since no single equivalent λ_{th} can be identified from the results.
Fig. 2 shows the distribution of the nodes in λ space, that is, each point is associated with the three eigenvalues at a node of the 128^3 grid. This shows that the four environmental types have very different clustering properties, with voids being narrowly distributed in a small subvolume of the eigenvalues' and - on the other extreme - knots being spread around with a larger dispersion, as can also be seen in the distributions of λs in Fig. 4 and Fig. 5. This patterns characteristics were already noted previously for the case of standard V-Web, in the results shown in Table 3. In general it is therefore possible to state that while the fraction of volume occupied by voids is larger than the one of knots *in physical space*, in λ space the opposite is true, with voids being distributed compactly in a very small region while knots spread around a larger one. This is due to the fact that whereas we have a lower limit to the matter density in a given region of space (zero), the upper value is in principle unbounded and thus can be associated with a larger spectrum of (positive) λ values.

5 Choosing the optimal k

So far we have assumed $k = 4$ as is common practice when dealing with the cosmic web, as there are four possible combination of eigenvalues given a single threshold λ_{th}. However, as no other parameter is assumed within the framework of the k-means algorithm, in principle we do not need to restrict to $k = 4$ but we can instead look at what happens when using different numbers of classes.

It turns out that there is no unique solution to this problem, as several different metrics can be used to estimate the optimal number of ks. Moreover the distribution in λ does not show evident peaks or clustering patterns (see Fig. 2) as $\lambda_1, \lambda 2, \lambda_3$ are continuously spread and show no clear boundary for the cluster distribution. In what follows we have used the elbow method, the Calinski-Harabasz score, the Silhouette score, the center scatter, the entropy score and the gap statistics to estimate the goodness of different ks, the results of the applications of these scores are plotted in Fig. 6. Here we will briefly summarize the properties of each one of these estimators.

The Elbow method: we look at the within-cluster-dispersion (WCD) as a function of k, that is, at how tightly distributed are the points around each cluster center:

$$WCD(k) = \frac{1}{k}\sum_{j=0}^{k}\frac{1}{N_j}\sum_{i=0}^{N_j}\sqrt{(x_j^i - c_j)^2} \tag{4}$$

where x_j^i are the coordinates (in λ space) of the points belonging to the j-th cluster, N_j is the number of points in that cluster and c^j is the cluster center. Intuitively, larger ks will lead to a decreasing dispersion and hence - if the goal is to minimize the WCD - looking at decreasing $WCD(k)$ alone would lead us to favour $k = N_{tot}$, equal to the total number of points. However, we notice that the WCD decreases sharply for the firs k values but there are diminishing returns on using increasingly larger ks. The Elbow method then recognizes as optimal k value the one at the point of maximum curvature, when the diminishing returns regime kicks in. In our case we found $k = 4$, which is equal to what we would expect from the threshold classification approach.

The Calinski-Harabasz score: looks at the ratio of WSC to the between cluster dispersion (BCD) value, that expresses how far and separated the different clusters are. In principle, the ratio that produces a maximum k should be the optimal one, because it is the one that combines

Figure 4: Eigenvalues distributions for environment types identified using k-means and λs (grid = 128^3)

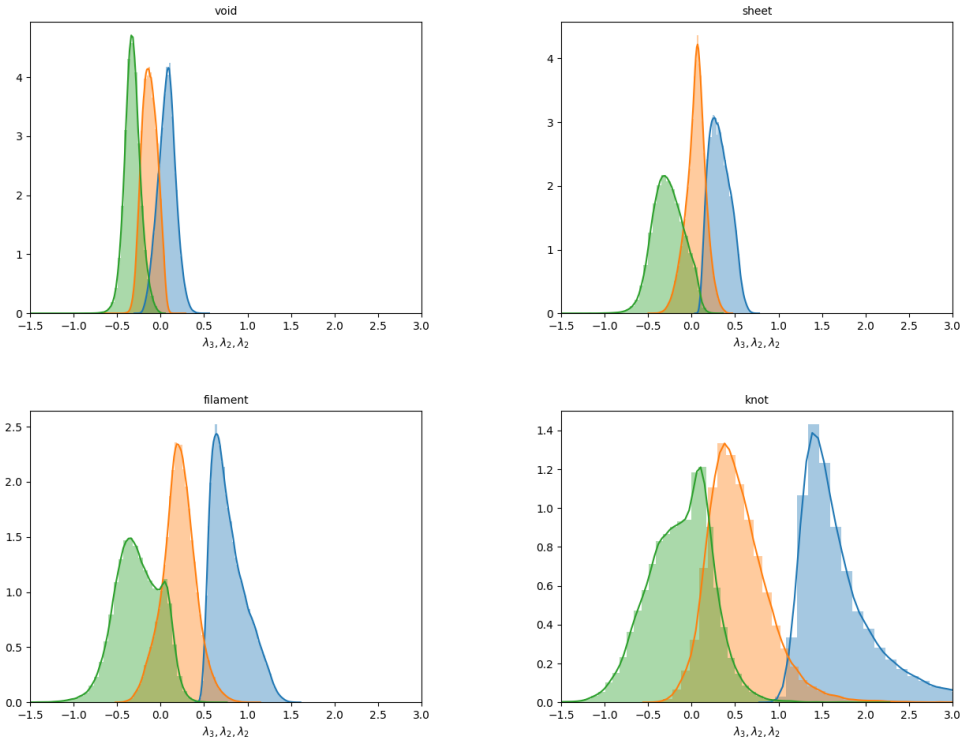


Figure 5: Distributions of $\lambda_1, \lambda_2, \lambda_3$ color coded by environment type, we notice that dispersion increases with increasing median density of each cluster type.

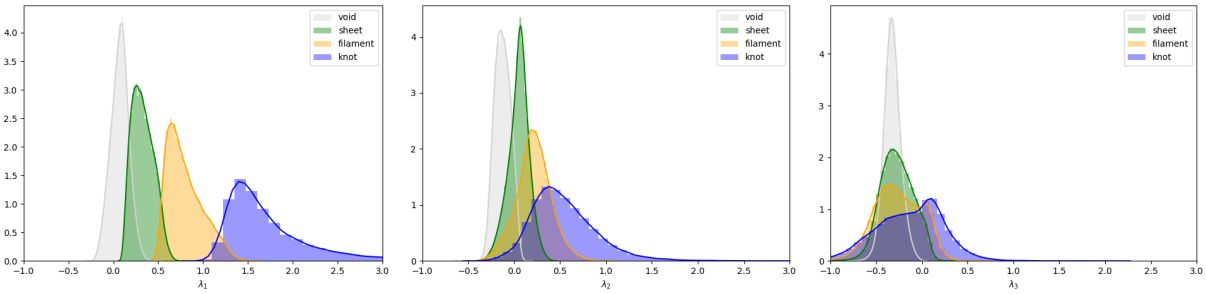
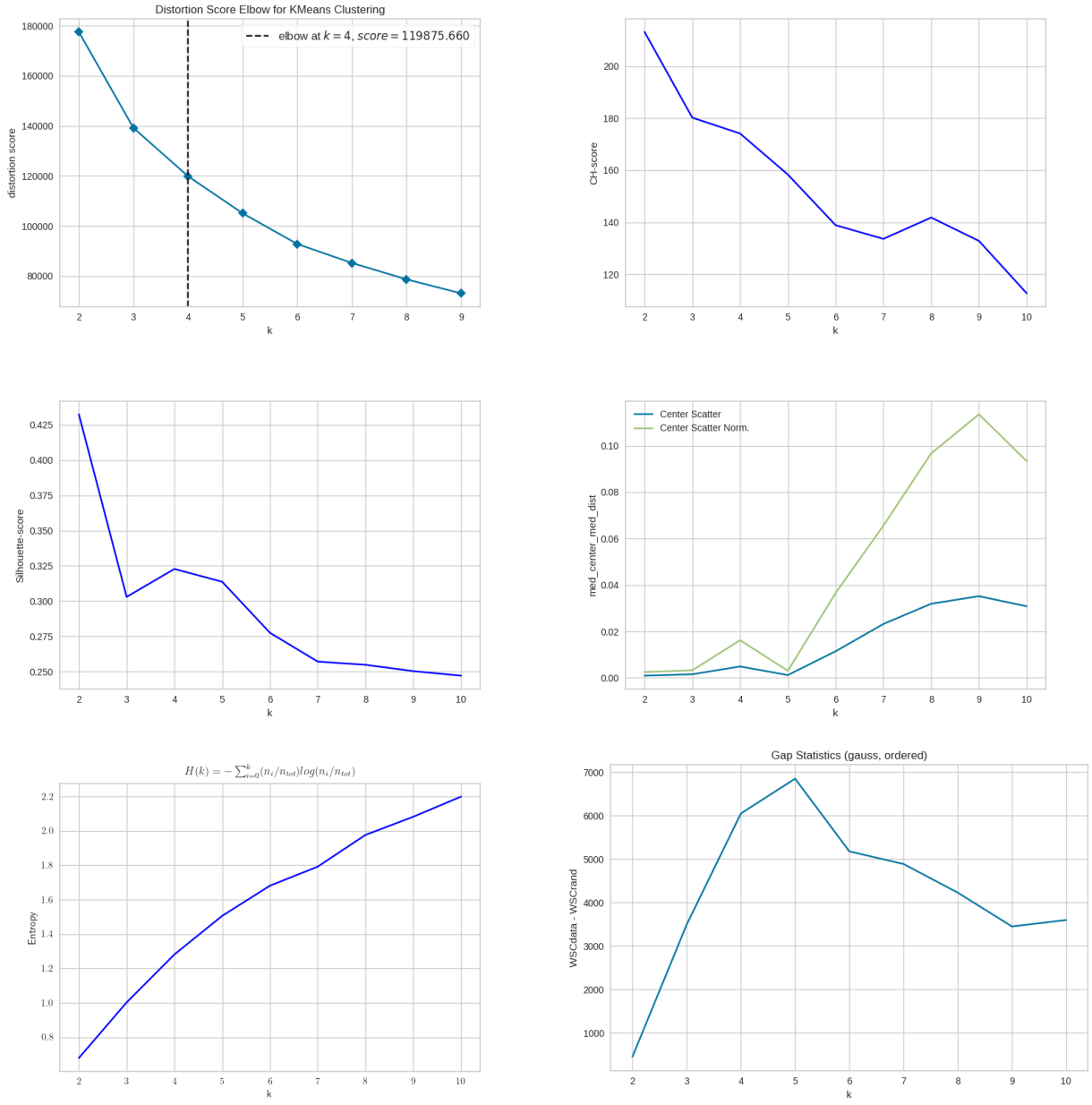


Figure 6: Optimal ks estimation using different metrics, from left to right in descending order: Elbow method, Calinski-Harabasz score, Silhouette score, center scatter, Entropy and Gap Statistics.



a maximal BCD (as we aim for well defined, separated clusters) and minimal WSC (as we want the points within each cluster to be close to the clusters centers). WCD is defined as in Eq. (4) and introduce

$$BCD(k) = \frac{1}{k}\sum_{i=0}^{k-1}\frac{1}{k-i}\sum_{j=i+1}^{k}\sqrt{(c_j - c_i)^2} \tag{5}$$

with c_i, c_j being again the cluster centers; we then define the CH score as:

$$s(k) = \frac{BCD(k)}{WCD(k)}\frac{N_{tot}-k}{k-1} \tag{6}$$

In our case $s(k)$ is a decreasing function of k which can be only said, in general, to favour smaller k values.

The Silouhette score: also looks at BCD and WCD but in a different combination, namely

$$s(k) = \frac{BCD(k) - WCD(k)}{max\{BCD(k), WCD(k)\}} \tag{7}$$

which automatically results in an index normalized between -1 and 1. In this case we also find a decreasing dependence on k which results in no specific k being preferred by the score (apart from a generic preference of smaller over larger k values).

The center scatter: in this approach we ran the k-means algorithm several times for each k using a different random initialization and limiting the cluster center iteration steps to 10. We used $n_{max} = 25$ different random initializations per k, each of the k clusters was labeled consistently across all runs, ensuring that they all approximately referred to the same region of λ space. In other words for each i-th cluster's we computed how its center position would fluctuate:

$$s(k) = \frac{1}{k}\frac{1}{n_{max}}\sum_{i=0}^{k}\sum_{j=0}^{n_{max}}\sqrt{(c_j - \bar{c}_i)^2}$$

where $\bar{c}_i$ is the median value of the center across all i clusters. To have a (rough) idea of how much this fluctuation corresponds in terms of actual distance between the clusters, we further compute the $BCD(k)$ value:

$$BCD(k) = \frac{1}{k}\sum_{i=0}^{k-1}\frac{1}{k-i}\sum_{j=i+1}^{k}\sqrt{(c_j - \bar{c}_i)^2} \tag{8}$$

where we now have substituted c_i with $\bar{c}_i$, the median value of the i-th center across the n_{max} realizations. We used the BCD value to normalize $s(k)$ and determine the mean scatter of the centers over the mean scatter across clusters. This index shows that values of $k < 5$ produces a scatter below 2% of the BCD, meaning that the centers are well defined and can be reliably identified even with random initializations, whereas for larger ks the fluctuations of individual cluster positions can be as high as 10% of the BCD. However, also in this case a single, optimal k cannot be identified.

The entropy score: uses the Shannon entropy definition

$$H(k) = -\sum_{i=0}^{k}\frac{n_i}{n_{tot}}\log\frac{n_i}{n_{tot}} \tag{9}$$

Figure 7: Three-dimensional clustering of the eigenvalues for $k = (3, 5)$

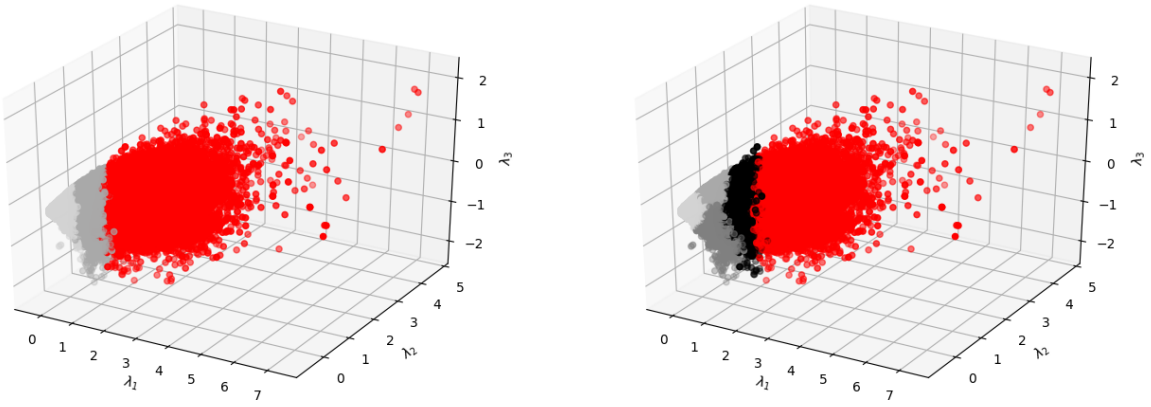
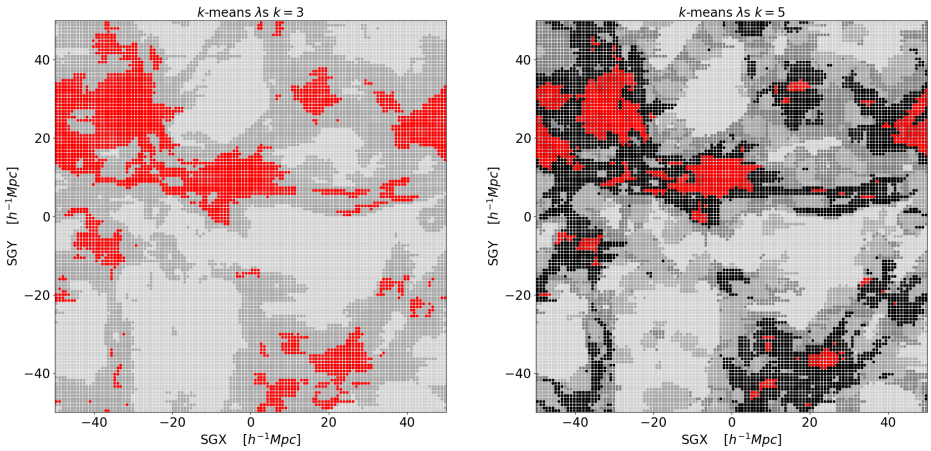


Figure 8: Two dimensional projections of the cosmic web using $k = (3, 5)$



where n_i is the total number of points belonging to the i-th cluster for a given k. Again, we find a slowly monotonical increasing function that gives no clear indication for a single k value.

The gap statistics: looks at the differences in WCD between the data at hand and a randomized data distribution. This was generated using a normal distribution for $\lambda_{1,2,3}$ with means and standard deviations as in Table 1 for SIMU04. For each k then we look at the difference in the two WCDs

$$g(k) = |WCD(k) - WCD^{rand}(k)| \tag{10}$$

The best k value is the one that maximizes this difference, in our case, this is found to be $k = 5$ by looking at Fig. 6.

Finally we show the cosmic web with k-means classification using $k = 3$ and $k = 5$ as an example, by looking at the distribution of eigenvalues in λ space and at the V-Web maps. Whereas

it can be argued that $k = 3$ produces a consistent picture of the universe, with three well defined areas, it is pretty clear by visual inspection that the additional class is associated with structures between filaments and sheets which, however, have no clear physical role or determination, and to not seem to convey any additional information about the distribution of matter.

6 Conclusions

We have analysed the cosmic web in cosmological N-body simulations using an unsupervised learning approach, that is, the classification into different environments was decided autonomously by the algorithm (the k-means in our case) without supervision.
We computed the shear tensor eigenvalues on a 128^3 grid for DMO simulations of 512^3 particles. We started assuming $k = 4$ as is customary in the V-Web analysis based on a single threshold λ_{th} which allows to label every node as either a void, a sheet, a filament and a knot depending on the number of eigenvalues above the given threshold. Our results based on this assumption showed that the k-means is able to produce a classification which is consistent with the other method on many levels. The main findings are:

- We have shown that the density distribution, the V-Web (with $\lambda_{th} = 0.2$) and the k-means algorithm with $k = 4$ produce similar maps, where the most prominent voids and clusters can be identified to be located at the same positions. In this case, 78% of the nodes could be identified as belonging to the same environmental type though they were classified with two different methods.

- We have shown that the eigenvalue distributions share also very interesting properties, with $\lambda_{1,2,3}$ in voids being sharply peaked with very small scatter whereas their distributions, especially in knots, show a much larger dispersion.

- Relaxing the assumption of four classes, we investigated the results of using a series of k ranging from $k_{min} = 2$ to $k_{max} = 10$ and used several different estimators to evaluate whether we could obtain an optimal value of k without further assumptions. Some of these methods cannot provide a single clear answer for the optimal k (for the entropy, Silhouette and CH scores for example), while we found $k = 4$ in the case of the Elbow method, $k = 5$ for the gap statistics and $k < 6$ by the cluster centers' scatter.

We have been able to show that the k-means algorithm is a viable parameter free alternative to the λ_{th} V-Web classification, which produces consistent results even though it's based on a radically different approach that neglects any physical prior. Even relaxing our assumptions on k it is remarkable to note how the Elbow method (the most used) converges to $k = 4$, as suggested by the standard approach based on a λ threshold. This has the implication that the k-means is revealing some intrinsic features of the topology of $\lambda_1, \lambda 2, \lambda_3$ distribution in λ space.