CH5650 - Molecular Data Science and Informatics - Quiz - Report

Datapreprocessing:

The data file 'Polymer.data' is read as a text file in python. The ATOM, id, molecule type, atomic positions, and velocities alone are stored in a numpy array for each timeframe. The other 9 lines of data in every timeframe are ignored. Since the objective is to carry out clustering to find the number of phases, the order of the data in timeframes is not necessary. Now, the data contains positions and velocities of individual atoms in every timestep. But the focus of analysis here is to find the number of phases of the molecule throughout the trajectory. So, it is important to coalesce all the atomic features into a representative for the entire molecule for each timestep. Hence, the atomic coordinates and velocities are averaged out for each timeframe. However, this gave extremely low orders of velocities compared to position coordinates. This caused imbalanced data values which led to poor PCA results. Scaling and normalizing also were tried, however the results of dimensionality reduction deteriorated. Median was thus used to coalesce the individual atomic features for each timestep. This produced higher order values for velocities, improving the results of kernel PCA and clustering.

Also, we do not take 'ATOMs id' for further analysis because no additional data about the atoms were given, which could have been used to incorporate atomic properties as additional predictors. Since the 'mol' and 'type' attributes always contain the value 1 throughout the trajectory, we drop them from the features list as they make no difference. The featurs for the molecule per timestep were thus obtained.

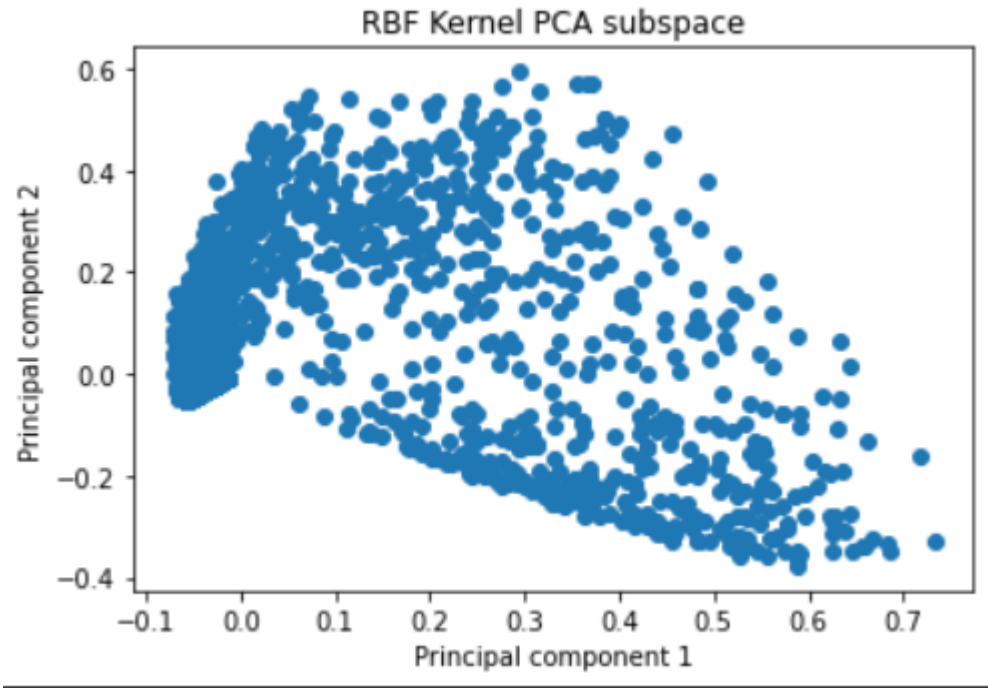

Fig 1: Dimensionality reduction - Kernel PCA subspace

Dimensionality reduction & Clustering:

Principal Component Analysis is employed to carry out dimensionality reduction. Linear PCA was first tried, however the subspace produced failed to depict any meaningful clusters. Kernel PCA was then employed, with

'radial basis function' kernel. Several orders of gamma values were taken and the gamma value = 20, produced subspace containing minimal discernable clusters. It was taken for further analysis.

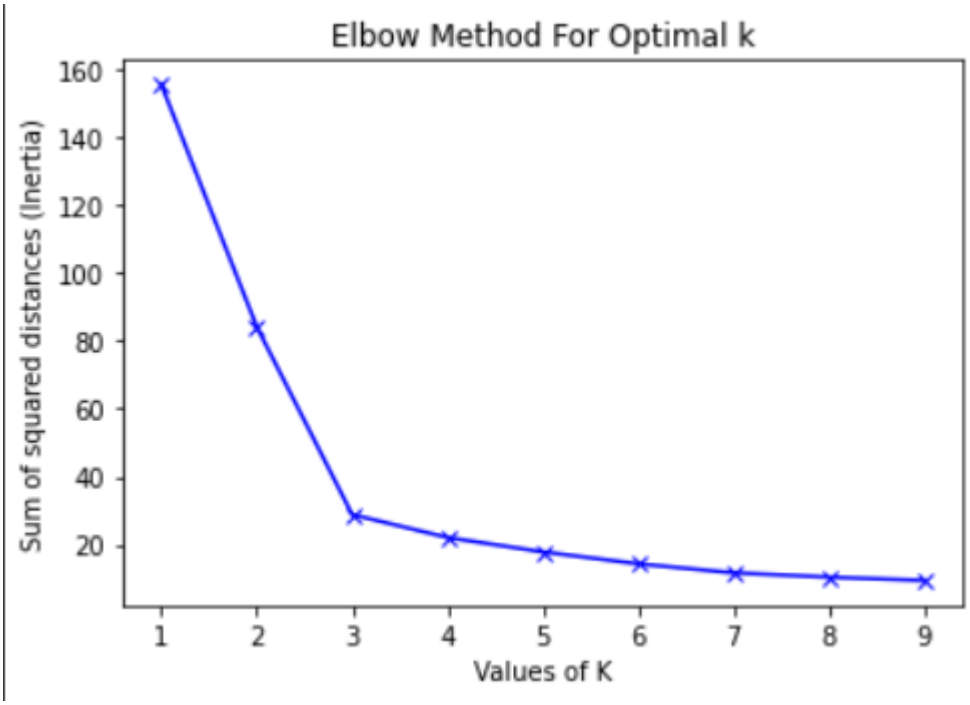

Fig 2: Calculating the optimal number of clusters using elbow method

To find the number of phases present in the trajectory, we take the help of clustering. Two types of clustering techniques were utilized for the analysis; k-means and DBSCAN. Since, we do not know the number of clusters beforehand, we use the 'elbow' method to find the optimal number of clusters, which turned out to be 3 (figure 2). K-means is performed using the cluster module in scikit library, with random initialization. The results of k-means are shown in figure 3.

We also perform DBSCAN for 2 reasons: (1) the subspace did not exhibit exclusive clusters of data, (2) it provides a way of finding the number of clusters without providing it apriori in the algorithm. The hyperparameters, 'eps' and 'minimum samples' were fixed by after several efforts of trial and error. The 'eps' and 'min_samples' values were tuned in such a way so as to keep the noise points less and also to ignore trivial clusters (small clusters within the large cluster). This way, three different clusters were obtained (figure 4).

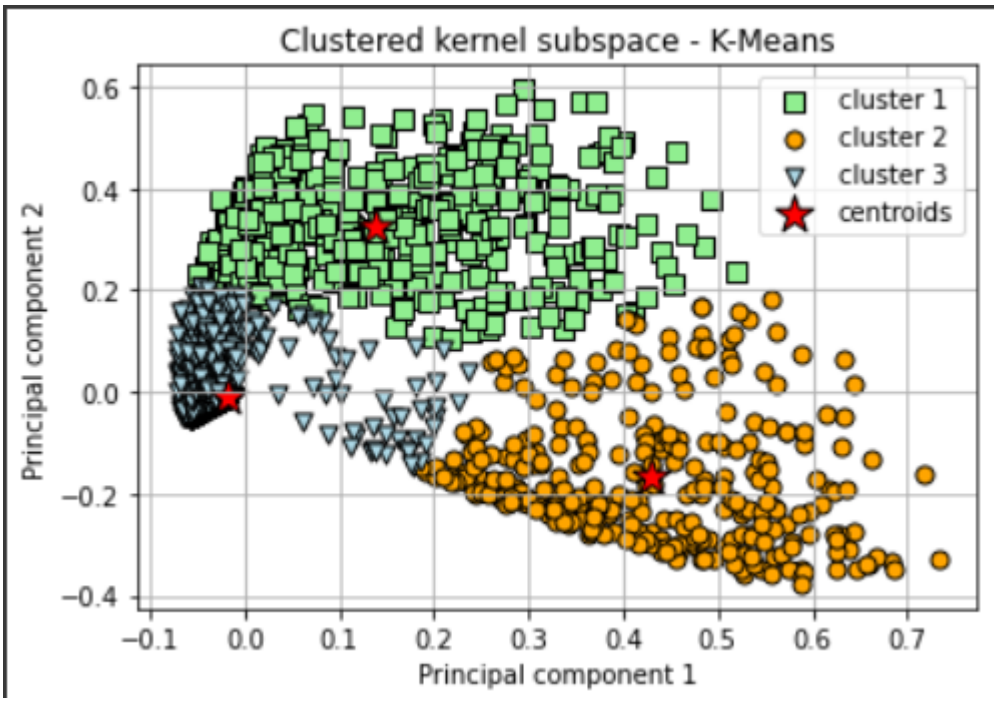

Fig 3: K-Means clustered subspace

The results of k-means and DBSCAN clusters were noticed to not be the same, albeit giving the same number of clusters. A possible reason could be sensitivity of k-means algorithm to initialization, as the centroids are initialized randomly here.

Conclusion:

Thus, the number of phases were found to be 3. The lower dimensional representation of the data showing all possible phases is shown in figure 4.

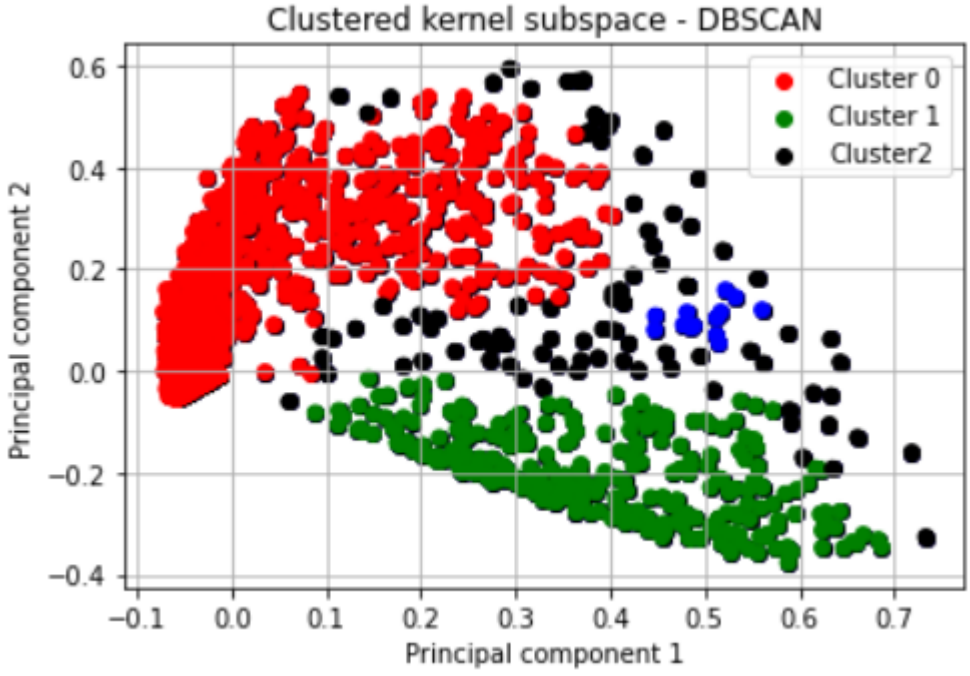

Fig 4: DBSCAN clustered subspace