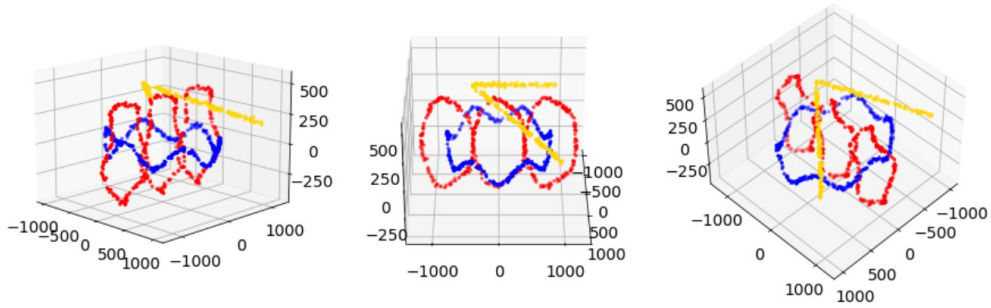
LCPB 21-22 Exercise 4, data visualization and clustering

You can choose to develop exercise 4A (recommended) or 4B

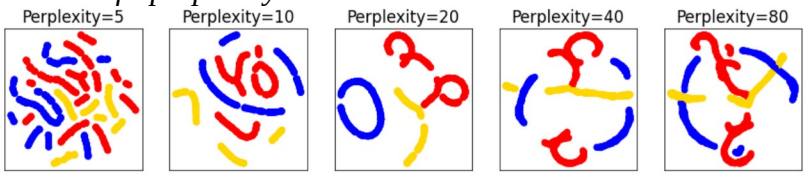
Exercise 4A

Consider data generated during the lesson.



1. *The role of dimensions*
   In each sample increase the number of dimensions from 3 to L>3, by introducing L-3 additional dimensions with noisy inputs. Study how visualization with t-SNE and clustering with DBSCAN are affected by this increase in dimensionality.
   To mix the information between all L dimensions while preserving the distances between points, one can also perform some rotation of data with orthonormal random matrices M in O(L): x → M.x

2. *The role of "perplexity" in t-SNE*



   Provide your explanation on the patterns observed by varying the perplexity of t-SNE.
   Do they vary if t-SNE is initialized by using principal component analysis (init='pca')?

3. *Tuning of "eps" and "minPts" in DBSCAN algorithm for clustering*
   The grid with several values of "eps" and "minPts" shows that the normalized mutual information (NMI) between true and predicted clusters is varying. Is there a correlation between these two parameters in providing a high NMI? Is there a way of guessing good values for "eps" and "minPts".
   Note: in the lesson we have looked at the typical distance between a point and its closest neighbor, but this does not say what is the typical distance from the 2nd, 3rd, …, "minPts"-neighbor.
   Furthermore, a possibility to consider is the plotting of ranked distances.

4. *VERY VERY OPTIONAL: t-SNE for clustering?*
   We know that t-SNE is stochastic and may converge to a different result if the random seed is varied. Moreover, visibly the result depends also on perplexity. Possibly, by checking which points are more likely to stay close to each other in different runs of t-SNE, one is able to assess the connectivity of the points in the original space, with implications for their clustering.

Exercise 4B

Do this version of the exercise at your own risk, it has not been tested earlier.

In this case we want to compare probability distributions *p(i)*, namely each sample is a set of probabilities, x=(p(0), p(1), p(2), … p(L-1)). The metrics to consider is the Jensen-Shannon divergence (JSD), which is in fact a true distance with all good properties of a metrics. It is based on the Kullback-Leibler divergence (KLD),

$$KL(p\|q) \equiv \sum_{i=0}^{L} p(i)\log\frac{p(i)}{q(i)}$$

which is not a symmetric function,

$$KL(p\|q) \neq KL(q\|p)$$

Considering the average distribution,

$$m(i) = \frac{1}{2}[p(i)+q(i)]$$

the JSD is defined as

$$JS(p,q) \equiv \frac{1}{2}KL(p\|m)+\frac{1}{2}KL(q\|m) = JS(q,p)$$

The JSD is always well-defined, while the KLD can "explode" if a *q(i)=0*.

For the exercise, chose say Y=4 different probability laws, $P_{y=1}(i)$, ..., $P_{y=Y}(i)$, and their cumulative distributions F_y's (for the sampling). For each data sample pick at random one of the P_y's (saving its label in the vector *y* for the test of performances) and generate M=1000 samples according to its law, with the constraint of limiting values to L bins corresponding to the integers *i=0,…,L-1* (larger extracted values from a *P* are discarded and re-extracted). Normalize the histogram in a *p(i)* and save the sample $x_n = p$.

The choice of probabilities can include, for example, exponentials $P\sim exp(-i/z)$ and power laws $P\sim(i+1)^{-z}$, where *z* is a fixed parameter for the class *y*.

With this database perform the same t-SNE, DBSCAN, and agglomerative clustering analysis done in the class, and eventually touch some of the points in exercise 4A.

Note that now one needs to impose the JS metrics in t-SNE and in the clustering algorithms. Hopefully, the description of this step is present in the documentation of the packages.