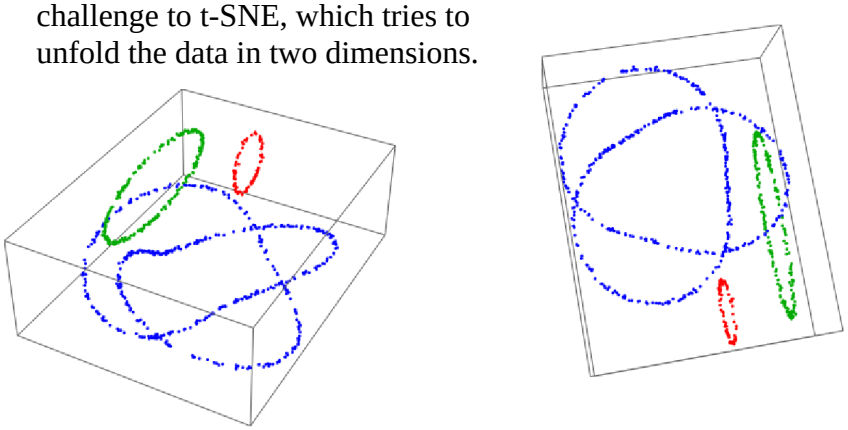
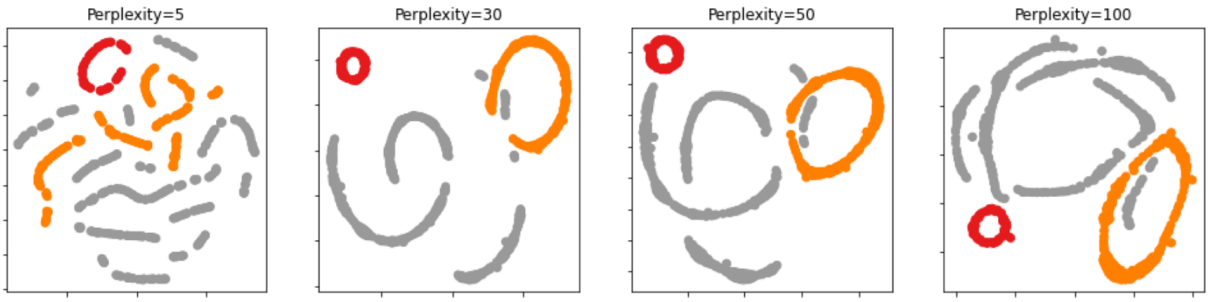
Exercise 4, clustering and data visualization

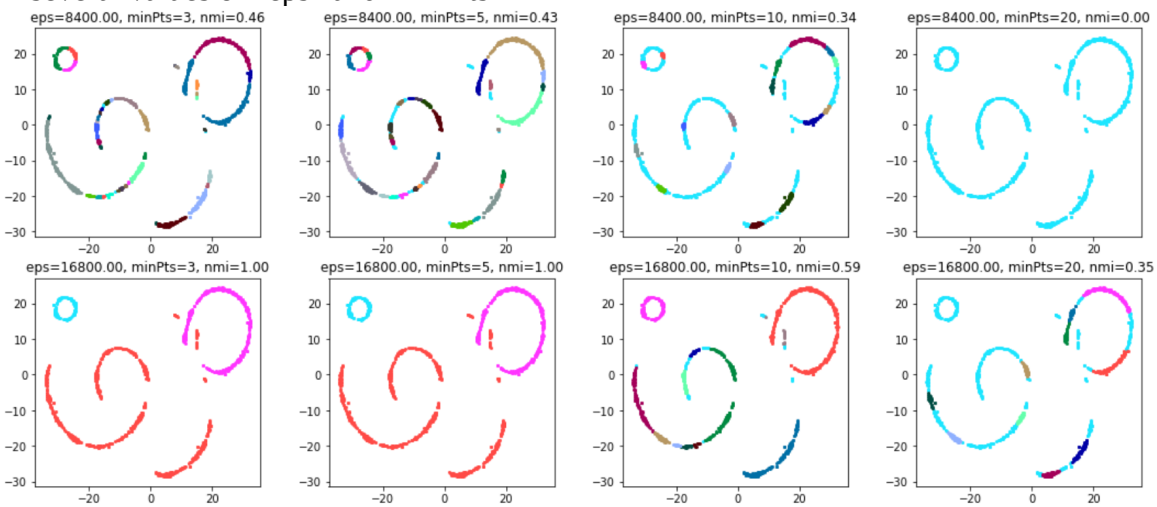
1. Read data files as “data_t-SNE_310101_d5_R100_e1_N800.dat” provided in the google folder, which contain high dimensional data (d=5 in this case, with columns separated by the tab “\t”) with embedded manifolds as those in the figure, which represent three clusters with a linear closed structure. Given N data points, the first 10% belongs to cluster “0” (red), the next 30% to cluster “1” (green) and the last 60% to cluster “2” (blue). The green cluster is linked to the blue one, in the 3d representation. This is expected to challenge the convergence of t-SNE. The knotted structure of the blue cluster should introduce a similar challenge to t-SNE, which tries to unfold the data in two dimensions.



2. Apply t-SNE with 4 different perplexities to the data. The method is one out of many methods available at scikit website: https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html The result should look like



3. From NB15 extract the part concerning the DBSCAN algorithm for clustering and embed it in the notebook to analyze the clustering in d=5 dimensions to generate predicted labels, then plot the results. Something like the next figure should arise. It includes a grid with several values of “eps” and “minPts”



The figure specifies, for each panel, the corresponding value of “nmi” defined in the notebook by Mehta et al, which quantifies the difference between true labels and predicted ones.
One needs to take care of introducing meaningful values of “eps” in DBSCAN: this parameter is the radius of the spheres around points, and one needs spheres to be reasonably full. In your notebook, find the typical scale separating points in d=5 from their first neighbors, and use it as a reference for “eps”, trying some multiples of that value.

4. As usual, play with parameters. For instance, in t-SNE one may initialize the algorithm by using principal component analysis (PCA). Also a 3D version of t-SNE could be used; it is available in the packages.