Notes from class discussion on Isomap

Dear all, coming back to you after your questions of Friday on Isomap.

Here are some important considerations you should be taking into account.

- First of all, the geodesic distance matrix you generate must be symmetric!

  Considering that `sklearn.neighbors.NearestNeighbors`(as some of you correctly pointed out in class) only provides the distance in a directed graph (ie. it returns the distance d_{ij} from node i to node j but does not associate the same distance to the edge from node j to node i) you can do one of the two following things.

  – If you use the result of `sklearn.neighbours.NearestNeighbours` directly, you can force symmetry in the Floyd Warshall algorithm by adding in the inner most loop (or in the if-statement more correctly) that `d[j, i] = d[i,j]` , after the update of `d[i,j]`

  – You can symmetrize first the pairwise distance matrix and then use the Floyd Warshall algorithm as seen in class.

- Check that the resulting matrix is symmetric as this property also ensures that you won't find any complex eigenvalues (which was the problem for some of you).

Let me remind also how to select the number k of Nearest Neighbours: it should be chosen as the minimal value of k that ensures a fully-connected graph. Depending on the size of your dataset (ei. during the first try with $n = 100$) you may need to consider a larger value for k.

If you have any more questions we can discuss them during the next labs!