67750: Advanced Practical Machine Learning	2017/8

Exercise 1 — Manifold Learning Practical Solution

TA: Daniel Gissin

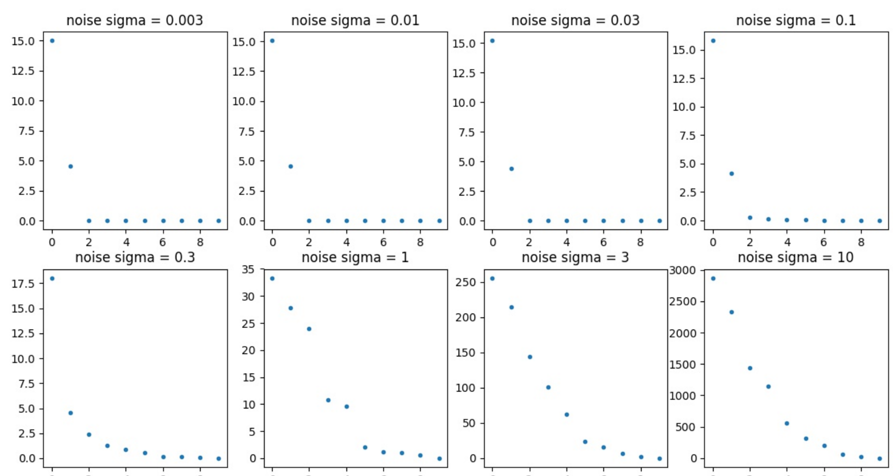
1 Practical Solution

The code for the practical solution can be viewed on the moodle, for those of you who had trouble with any of the tasks.

We request that you **do not publish the solutions of the exercise or pass them on to people outside of this class**. Please allow future students of this course to learn these subjects without the temptation of ruining their learning experience.

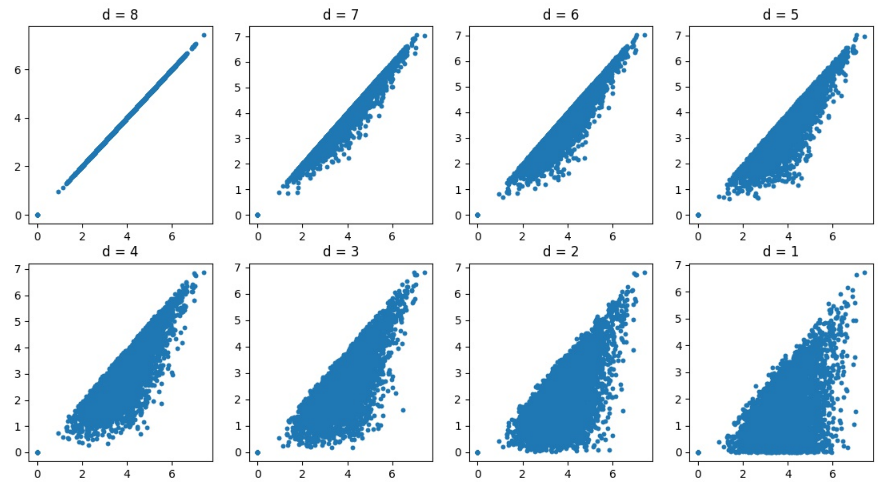
1.1 Scree Plot

When we have no noise in the data set, we expect the data to be completely on a 2d linear subspace. The moment we add a little bit of noise, the data matrix becomes full-rank, but we can still recognize that it is "mostly" 2 dimensional using the scree plot. In the figure you can see how the noise effects the scree plot - the more noisy the data set, the less we are able to see that it is 2 dimensional...



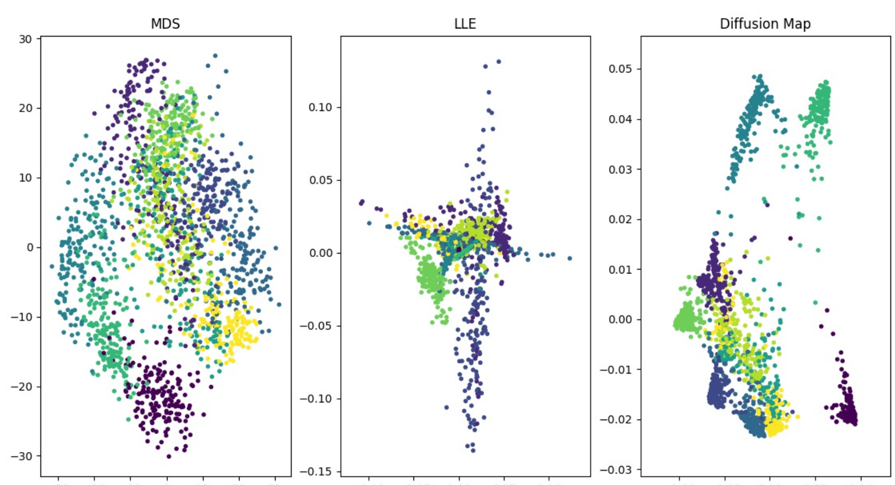
1.2 Distance Scatter Plot

When plotting the pairwise distances before and after dimensionality reduction, we can see as we reduce the dimension more and more, the distances between points become less correlated. Since the data really was full-rank, each dimension had information that we lost when we went to lower and lower dimensions.



1.3 MNIST

When we compare the algorithms on the MNIST data set, we can see that all of them really do detect the structure of the data relatively well. That being said, had we not had the labels of the data, we would have trouble picking out clusters from the results of MDS and LLE. On the other hand, DM is quite successful here, and we would have been able to detect at least some of the clusters.

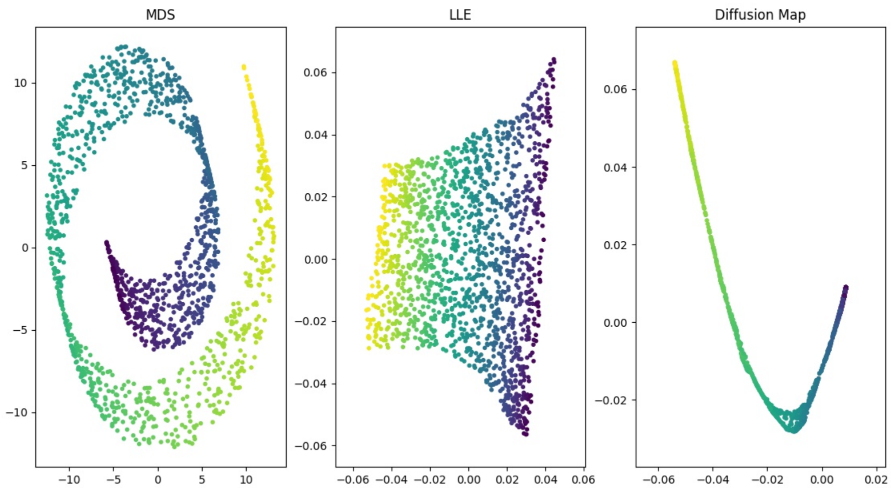


Note that tweaking the parameters of both LLE and DM could potentially give us even better results.

1.4 Swiss Roll

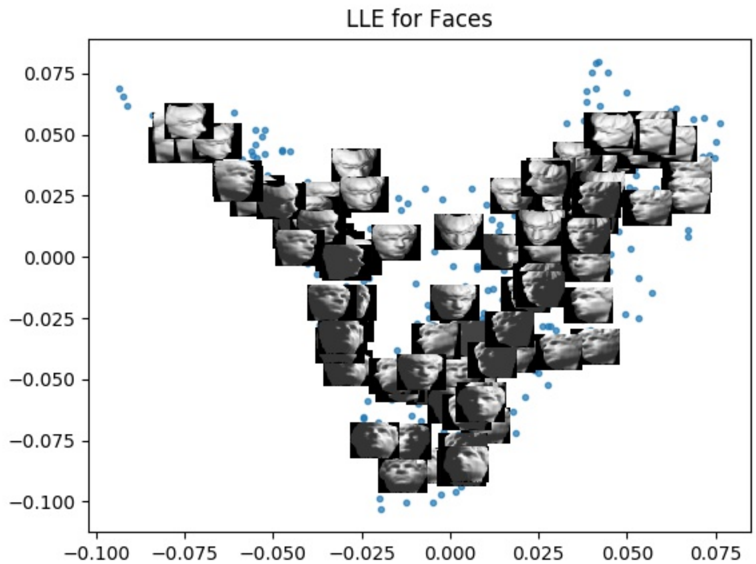
The swiss roll data set is relatively difficult to get to work well, and requires parameter tweaking. Since LLE has only one integer parameter, tweaking it should have been easier for you.

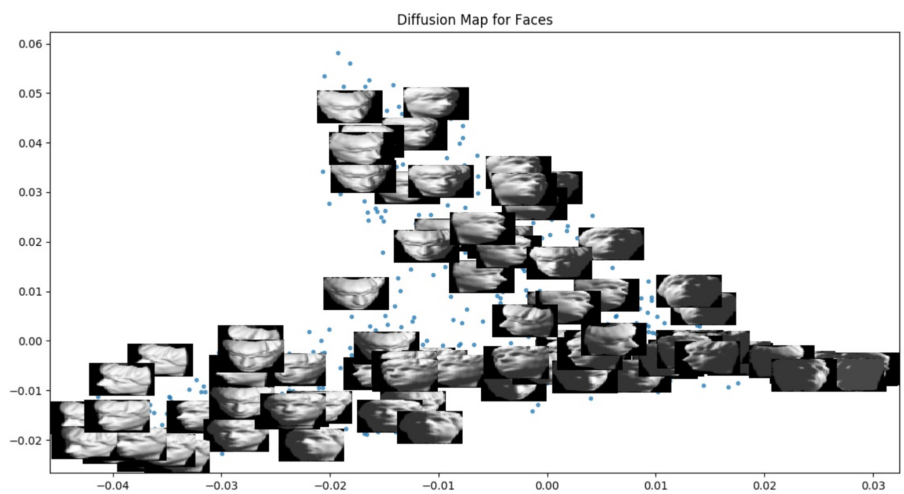
We can see that LLE performs better than DM on this sort of data. As for MDS, since our data is intrinsically non linear, trying to fit a linear subspace to it fails completely, and we end up having data points that are far apart on the manifold be close after MDS...



1.5 Faces

While MDS doesn't fail miserably on the Faces data, LLE and DM are able to recover the structure of the data really well, seperating the data set into two clear degrees of freedom corresponding to direction and/or lighting:

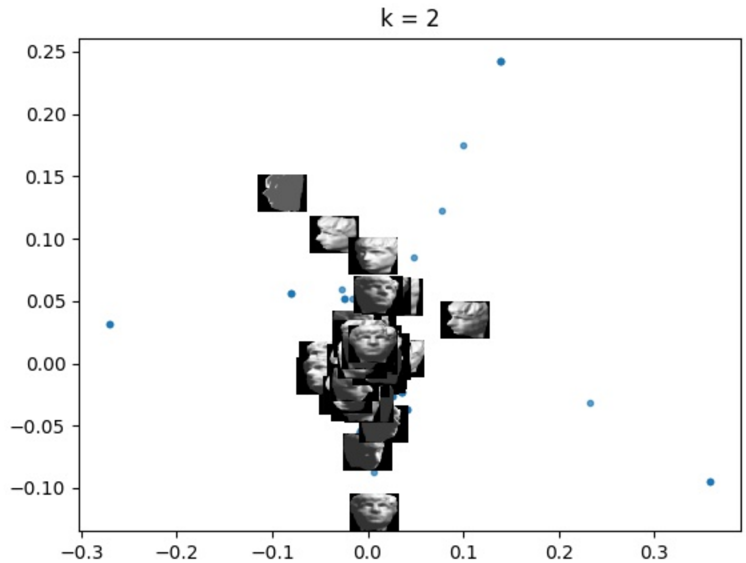


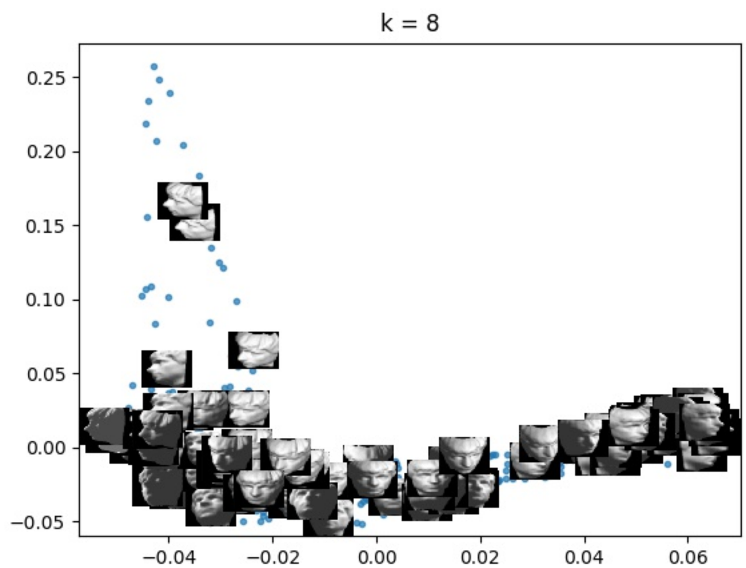


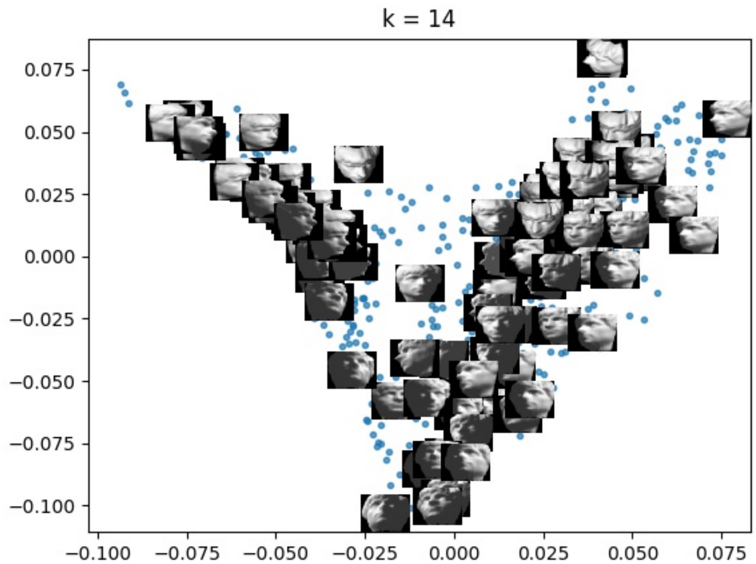
This is quite remarkable! While the data originally sat in a very high dimensional (4069) ambient space, we were able to reduce it all the way down to two dimensions and still maintain a lot of the information that the original data had.

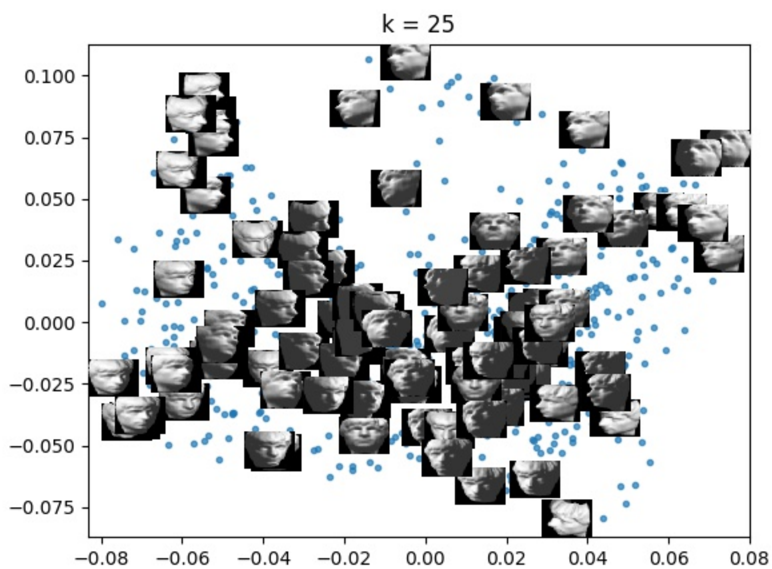
1.6 Parameter Tweaking

The number of neighbors we use in LLE determines how large the neighborhood of the data points should be. When k is too small, we aren't looking at enough neighbors to successfully reconstruct each data point. It is also possible that the small neighborhoods create disconnected portions of the data set, causing them to become much closer than they really should be. On the other hand, when k is too large we end up having relatively distant points effecting the location of each point (when they are far away on the manifold, or just not appropriate for a linear approximation). This makes the LLE algorithm be non-local and act more and more like PCA or MDS:



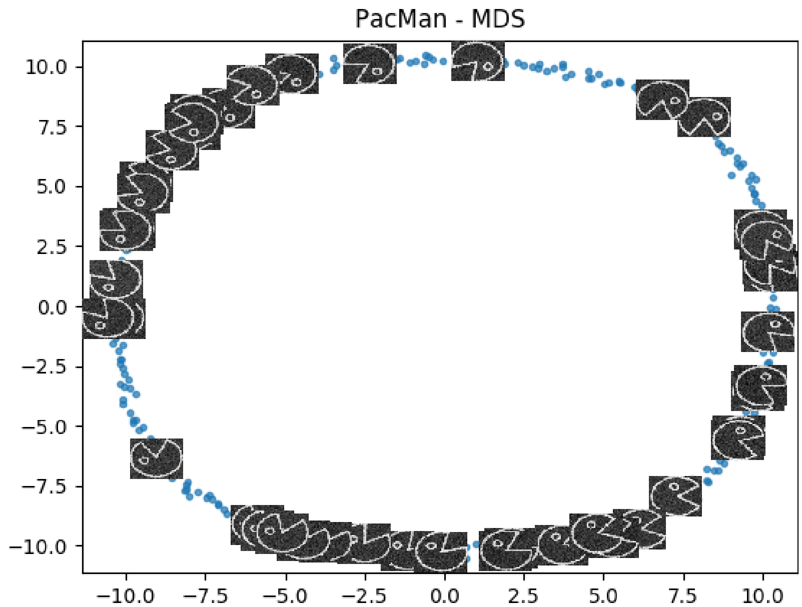


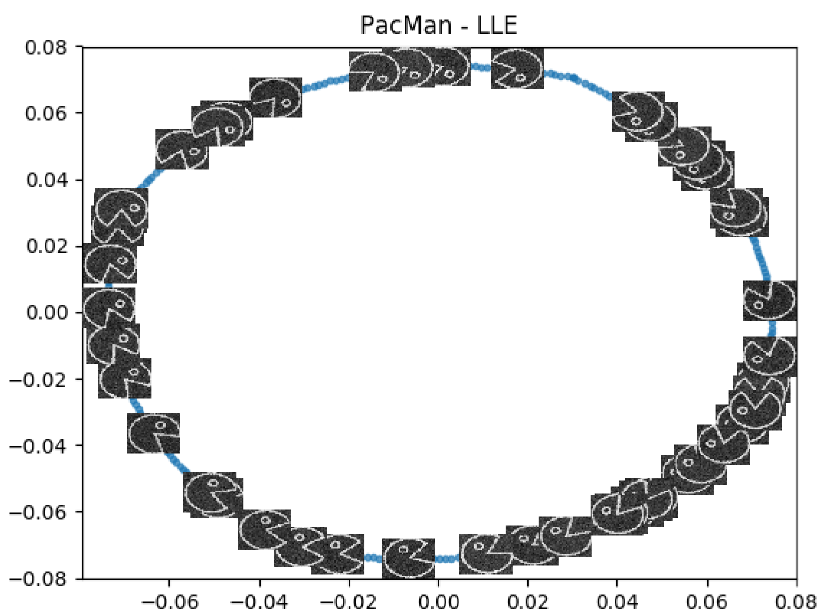


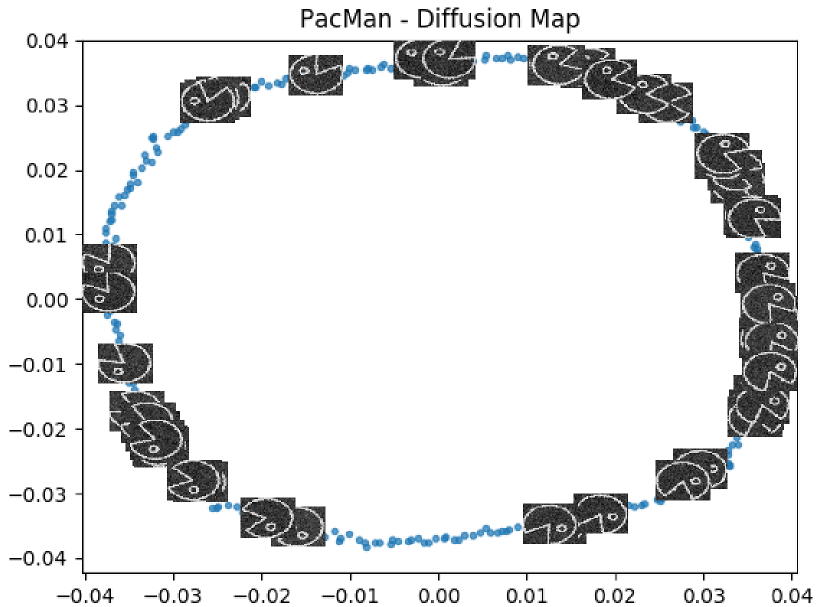


1.7 Bonus

Taking the picture and rotating it gives us a data set similar to the Faces. Surprisingly, even MDS works really well on this simple example (while we know that examples with higher degrees of freedom are hard for MDS):


PacMan - MDS
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−10.0 −7.5 −5.0 −2.5 0.0 2.5 5.0 7.5 10.0


PacMan - LLE
0.08
0.06
0.04
0.02
0.00
−0.02
−0.04
−0.06
−0.08
−0.06 −0.04 −0.02 0.00 0.02 0.04 0.06 0.08



1.8 In Summary

We hope you were able to appreciate the potential and importance of Manifold Learning techniques when working on high dimensional data.

Since the early 2000s the field of Manifold Learning has taken off, but there are still many open questions. The algorithms that are around today aren't perfect, and have different strengths and weaknesses - it is wise to try out different algorithms to see what works best for your specific problem.