SLT coding exercise #1

Locally Linear Embedding

https://gitlab.vis.ethz.ch/vwegmayr/slt-coding-exercises

Due on Monday, March 6th, 2017

Birkir Snaer Sigfusson
16-932-048

Contents

The Model 3

The Questions 4

(a) Get the data 4

(b) Locally linear embedding 4

(c) Cluster structure 4

(d) Nearest Neighbors 4

(e) Linear manifold interpolation 4

The Implementation 7

Your Page 8

The Model

The model section is intended to allow you to recapitulate the essential ingredients used in Locally Linear Embedding. Write down the *necessary* equations to specify Locally Linear Embedding and and shortly explain the variables that are involved. This section should only introduce the equations, their solution should be outlined in the implementation section.
Hard limit: One page

There are two main equation we are working with are:

$$\varepsilon(W) = \sum_i \left| \vec{X}_i - \sum_i W_{ij} \vec{X}_j \right|^2$$

$$\Phi(Y) = \sum_i \left| \vec{Y}_i - \sum_i W_{ij} \vec{Y}_j \right|^2$$

The first is a cost function for the reconstruction error, i.e. the sum of squared errors between each vector $\vec{X}_i$ and its reconstruction using all other vectors $\vec{X}_j,\ j = 1, ...i-1, i+1, ...N$. The $\vec{X}$ vectors have dimension D. W is an NxN matrix of weights. As we will see, W is really sparse because each $\vec{X}_i$ will only be reconstructed from its nearest neighbors. The second equation is a cost function for the embedding vectors $\vec{Y}_i$ which have dimensionality $d < D$ with the weights W_{ij} found using the first equation. Naturally the solution to the problem is found by minimizing the first equation, solving for the weights, then minimizing the second equation thus finding the embedding space.

The Questions

This is the core section of your report, which contains the tasks for this exercise and your respective solutions. Make sure you present your results in an illustrative way by making use of graphics, plots, tables, etc. so that a reader can understand the results with a single glance. Check that your graphics have enough resolution or are vector graphics. Consider the use of GIFs when appropriate.
Hard limit: Two pages

(a) Get the data

For this exercise we will work with the MNIST data set. In order to learn more about it and download it, go to http://yann.lecun.com/exdb/mnist/.

(b) Locally linear embedding

Implement the LLE algorithm and apply it to the MNIST data set. Provide descriptive visualizations for 2D & 3D embedding spaces. Is it possible to see clusters?

(c) Cluster structure

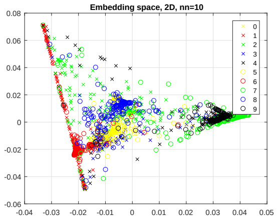
Investigate the cluster structure of the data. Can you observe block structures in the M matrix (use matrix plots)? Also plot the singular values of M. Do you notice something? Can you think of ways to determine the optimal embedding dimension?

(d) Nearest Neighbors

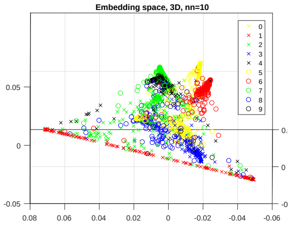
Investigate the influence of the choice of how many nearest neighbors you take into account. Additionally, try different metrics to find the nearest neighbors (we are dealing with images!).

(e) Linear manifold interpolation

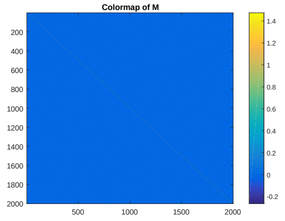
Assume you pick some point in the embedding space. How can you map it back to the original (high dimensional) space? Investigate how well this works for points within and outside the manifold (does it depend on the dimensionality of the embedding space?) Try things like linearly interpolating between two embedding vectors and plot the sequence of images along that line. What happens if you do that in the original space?



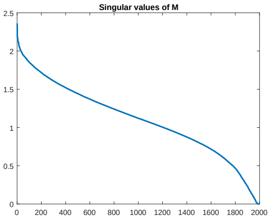
(a) 2-D embedding space. The clusters are visible but overlap is significant in various areas.



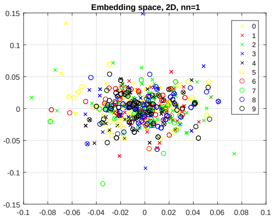
(b) 3-D embedding space. Adding a dimension reveals clearer serperation between clusters. E.g. the blue and yellow.



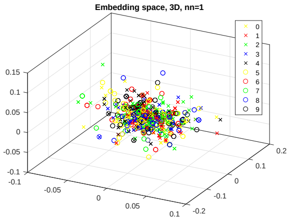
(c) M matrix for the embedding space in (a). I can't notice any clusters in the matrix. That's because the pictures for different numbers are placed randomly around the matrix so we shouldn't expect any noticable clusters to form.



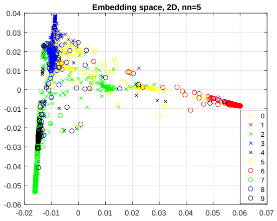
(d) The singular values of M in (c). Perharps the optimal dimension is where the graph plateaus? (going from the right).



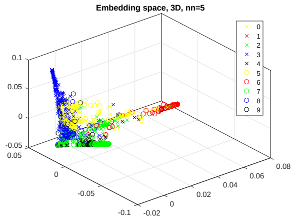
(e) 2-D embedding space using only one nearest neighbor. All datapoints are mixed together



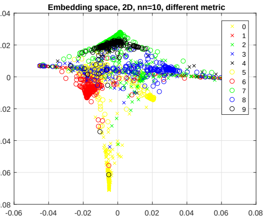
(f) 3-D embedding space using only one nearest neighbor. Adding a dimension hasn't revealed anything.



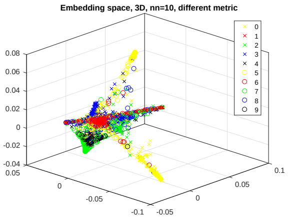
(g) 2-D embedding space using 5 nearest neighbors. The clusters are now becoming visible.



(h) 3-D embedding space using 5 nearest neighbors

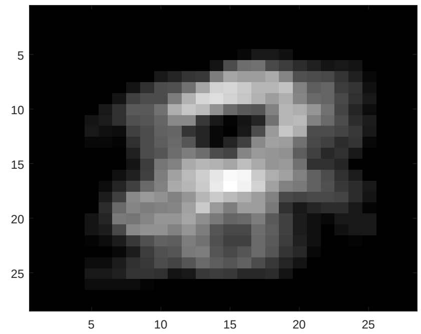


(i) 2-D embedding space using absolute distance. It works but I wouldn't say its better.

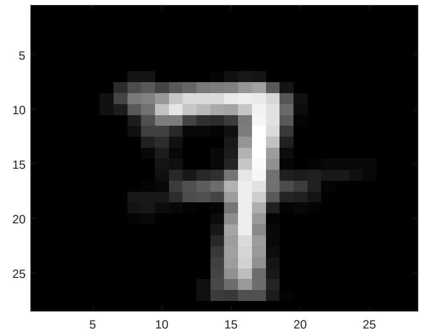


(j) 3-D embedding space using absolute distance. Again I'm more happy about using L2 distance.

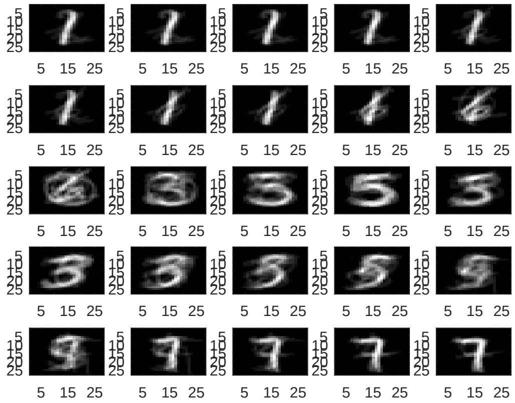
The Questions continued on next page. . .



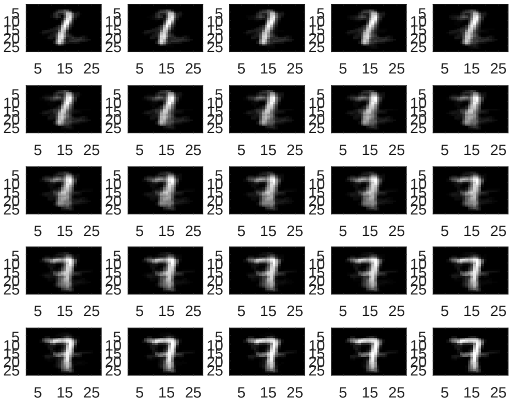
(a) Point on manifold mapped into the original space. This point was generated randomly so it seems to be in a place with a lot of numbers



(b) Point outside manifold mapped into higher dimension. The result is clearly a seven.



(a) linearly interpolating between two points in the embedding space. The set of pictures get a bit messy in the beginning.



(b) Linearly interpolating between two points in the original space. The transition between images is much smoother in this case

The Implementation

In the implementation section you give a concise insight to the practical aspects of this coding exercise. It mainly mentions the optimization methods used to solve the model equations. Did you encounter numerical or efficiency problems? If yes, how did you solve them? Provide the link to your git branch of this coding exercise.
Hard limit: One page

1. Find the nearest neighbors of each vector in $\vec{X}$. Simple for loop and calculate L2 distance.

2. Find the optimal weights. Here I used the followed the suggestion of the paper by solving $Cw = 1$.

3. Find the eigenvectors of M and solve for the embedding vectors $\vec{Y}$. Because of having to compute the eigenvectors of M I couldn't use too much data.

Overall there weren't any complications. Just followed the outline of the paper.

Your Page

Your page gives you space to include ideas, observations and results which do not fall into the categories provided by us. You can also use it as an appendix to include things which did not have space in the other sections.
No page limit.

Your Answer
YOUR GIT BRANCH