SLT coding exercise #1

Locally Linear Embedding

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

Due on Monday, March 6th, 2017

Sahan Ayvaz
16-950-834

Contents

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 equations governing the locally linear embedding. The first one is $E(W) = \sum_i \mid X_i \sum_j W_{ij} X_j \mid^2$ where X_i is the 1 by D data vector and W is the N by N weight vector. The second one is $\Omega(Y) = \sum_i \mid Y_i + \sum_j W_{ij} Y_j$ where Y_i is the 1 by d transformed vector.

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

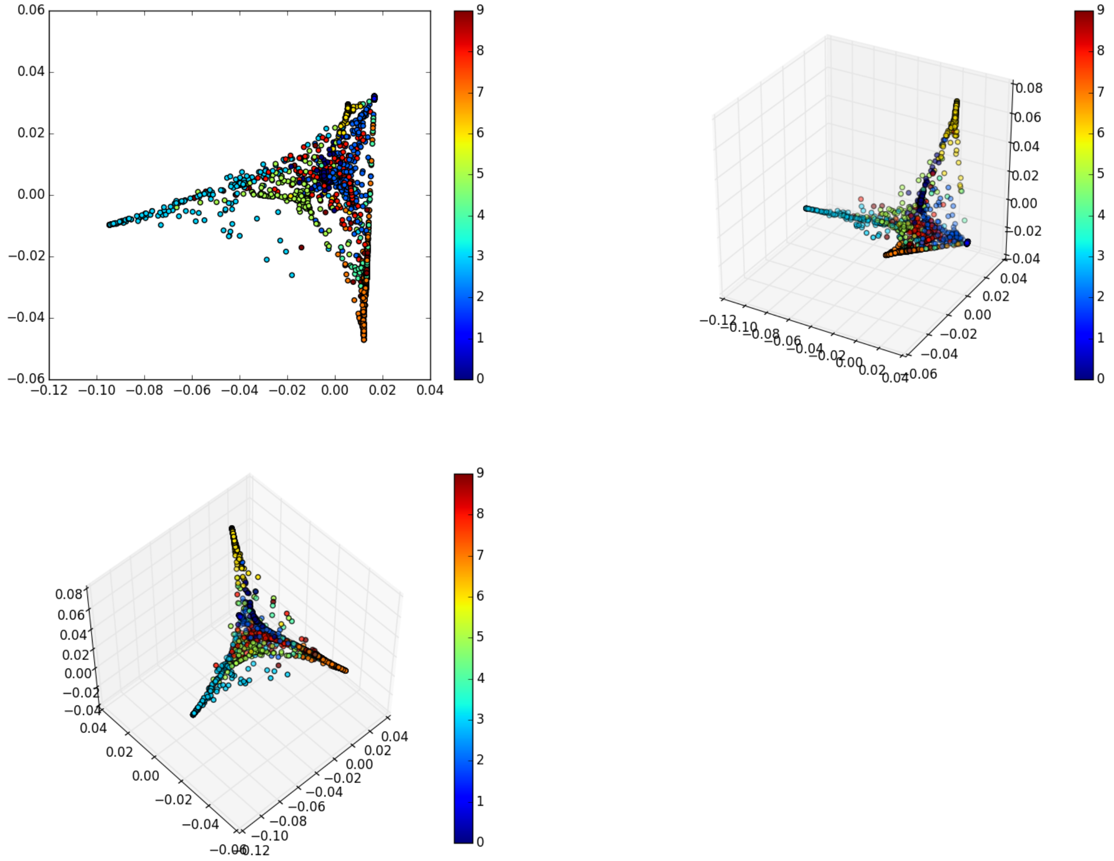
(a) Get the data

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/.

This project uses only the first 2000 numbers and their labels for testing LLE and plotting figures.

(b) Locally linear embedding

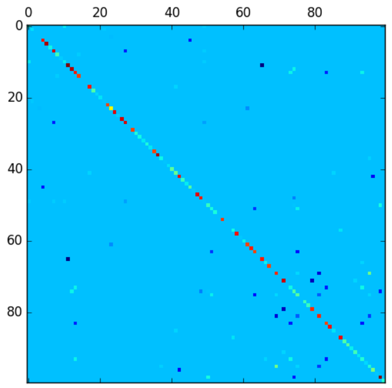
Implement the LLE algorithm and apply it to the MNIST data set. Provide descriptive visualizations for 2D & 3D embedding spaces. Is it possible to see clusters?

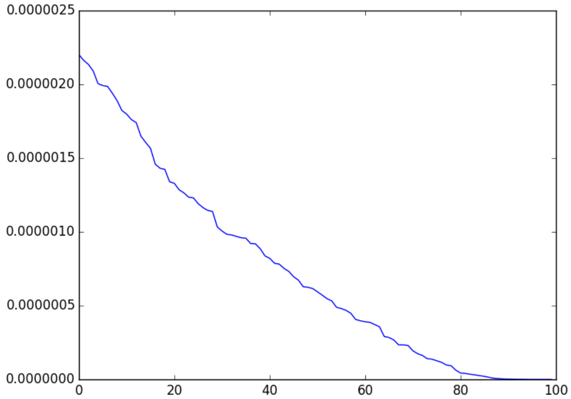


The first graph shows 2D embedding with 5 nearest neighbors, and the other graphs show 3D embedding with 5 nearest neighbors. In the 2D embedding, there are definitely some clusters for some numbers, such as 0, 3, 8 and 9; however, the cluster structure is not clearly separable. In the 3D embedding, the clusters are more clear to see by changing the angles of the graph. The two figures showing 3D embedding with different angles aims to show the cluster structure.

(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?

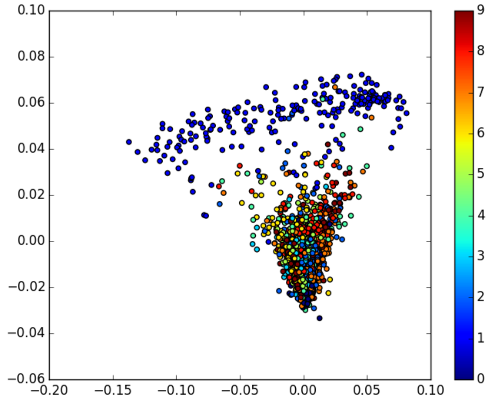


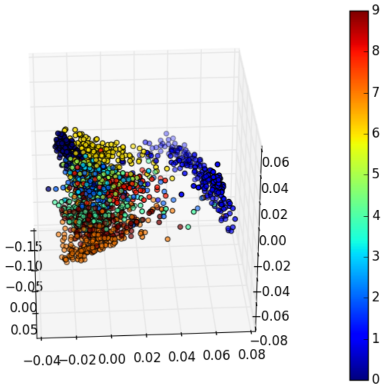


The first figure shows the matrix plot of M matrix for its first 100 entries, and the second figure plots its singular values for the last 100 singular values. The matrix plot reveals that the block structure is concentrated on the diagonal elements of the matrix. The singular value plot reveals that the singular values forms a steep decrease which stops at around 90. Since we are trying to minimize the cost, the smallest eigenvalues would be needed. Therefore, the optimal embedding dimension could be taken as 10.

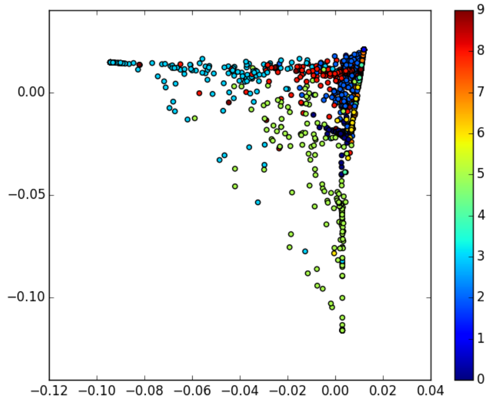
(d) Nearest Neighbors

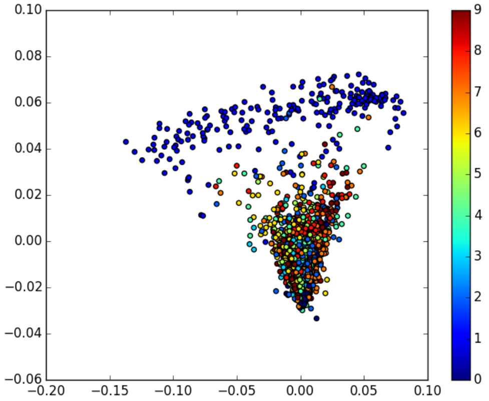
Investigate the influence of the choice of how many nearest neighbors you take into account. Additionally, try different metrics to find the nearest neighbors (we are dealing with images!).

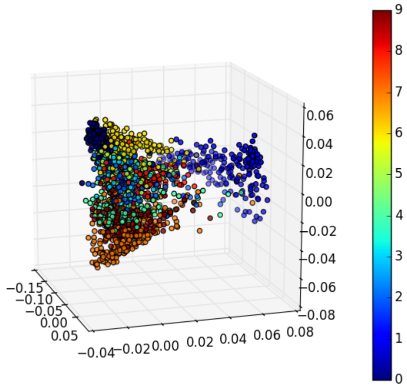




The figures show LLE with 30 nearest neighbors with euclidean distance. Compared to LLE with 5 nearest neighbors, the clusters are more distinct. The 3D embedding shows more clearly that the numbers could easily be clustered into their labels by even looking at the graph. Therefore, increasing the number of nearest numbers definitely increases the cluster distinction.







The figures show LLE with Minkowski distance and 5 nearest neighbors and Minkowski distance and 30 nearest neighbors with 2D and 3D embedding. Using a different distance metric did not affect the overall result, i.e. clustering, much compared to Euclidean distance. Although Euclidean distance is quite useless in high dimensional data, due to the 2D and 3D representation of the data, i.e. lower dimensionality, the distance metric did not influence the overall result. Yet we can clearly see that increasing the number of neighbors affected the clustering by yielding a better clustering again.

(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?

Your Answer

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

Link to git branch: `https://gitlab.vis.ethz.ch/vwegmayr/slt-coding-exercises/tree/16-950-834/1_locally_linear_embedding`
The LLE algorithm requires to find the K nearest neighbors of a data point. However, using a brute-force Knn search is very slow. Therefore, I used the scikit-learn's NearestNeighbors implementation with kd-tree search in order to optimize the Knn search. The rest of the code is vectorized as much as possible. For example, the C matrix is calculated by finding a k by D matrix A where each row j is calculated by $X_i - \nu_j$ where ν_j is the one of the neighbors, and performing $A^T A$. A similar approach is applied to finding the rows of W and M; therefore, the performance is close to the scikit-learn's LocallyLinearEmbedding implementation.
The two things could have been improved in my LLE implementation. The eigenvalues are not sorted by their absolute values. The inverse of C is calculated instead of solving the systems of linear equation. Those two things might cause small deviations from the scikit-learn's implementation and decrease the performance for high dimensional LLE.

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
16-950-834