A comparison between two manifold Techniques

Furrer Stanislas
School of Engineering (STI)
École polytechnique fédérale de Lausanne (EPFL)
Lausanne, Switzerland
Email: stanislas.furrer@epfl.ch

Carvalho Luis
School of Engineering (STI)
École polytechnique fédérale de Lausanne (EPFL)
Lausanne, Switzerland
Email: luis.carvalho@epfl.ch

***Abstract*—Locally linear embedding (LLE) is a classic method of nonlinear dimensional reduction and it has become more and more attractive to researchers due to its ability to deal with large amounts of high dimensional data and its non-iterative way of finding the embeddings. However, several problems in the LLE algorithm remain open, such as it's inevitable ill-conditioned eigenproblems. A modified version of the algorithm (MLLE) have been developed in order to improve the stability of LLE using multiple reconstruction weights. This paper comprehensively reviews and discuss LLE and its modified version. Their stability with various data and hyper parameters is discussed as well as their performance of topology preservation and classification. With numerical examples we will show how MLLE exceeds LLE stability in complex cases and when the parameters are weakly chosen.**

I. Introduction

Nowadays, scientists are faced with the necessity of exploring high-dimensional data more often than ever since information impacting on life and the evolution of mankind is growing extremely quickly. Dimensionality reduction is an important operation for dealing with multi-dimensional data. Most real data lie on a low dimensional manifold embedded in a high dimensional space. Frequently, high-dimensional data bear a lot of redundancies and correlations hiding important relationships. Therefore, data analysis can be used to eliminate these redundancies and reduce data complexities. A dimensionality reduction algorithm maps high dimensional data into a low dimensional space, revealing the underlying structure in the data. One such common dimension reduction technique is known as principal components analysis (PCA) [1]. PCA is limited in that it requires that the data lie on or near a linear subspace, which is an assumption that is often not satisfied in the real world. When the linearity assumption is not met, we turn to nonlinear dimensional reduction techniques, which do not require the linearity assumption and have been successfully in various applications. [2] [3] [4].

Recently, there have been advances in developing effective and efficient algorithms, such as isometric mapping (Isomap) [5], locally linear embedding (LLE) [6], Laplacian eigenmap (LE) [7], Hessian LLE [8] and local tangent space alignment (LTSA) [9]. All those methods can reduce the redundancies while retaining the primary characteristics. LLE is an effective nonlinear dimensionality reduction algorithm proposed first by Roweis in 2000 [6]. LLE is an unsupervised non-iterative method, which avoids the local minima problems plaguing many competing methods (e.g., those based on the expectation maximization (EM) algorithm). Compared to the other methods, the LLE algorithm requires only two parameters to be determined. The two parameters that have to be specified are the intrinsic dimension d and the number of nearest neighbors K. Improper values of these parameters greatly influence the results. On one hand, a large value of the intrinsic dimension d amplifies noise effects while a low value leads to overlaps in mapping results (excessively reduced) [11]. On the other hand, a low number of K nearest neighbors cannot make the reconstruction to reveal the global features of the original data [10], while a large K causes a manifold to lose the nonlinear feature and behave like the traditional PCA [1].

However, the original LLE has some intrinsic drawbacks. LLE was found to be sensitive to the amount of the initial data [12]. When there is insufficient data (poorly-sampled manifolds), local characteristics are lost. An excessive data amount results in an incomplete reconstruction and a long computational time. In the presence of noise, LLE is not anymore effective. When Gaussaen noise is added to each data point, the local linearity assumption becomes violated and LLE no longer handles the dimension reduction well. Furthermore, LLE is not robust against outliers. To overcome these limitations, some efforts have been recently made to develop various extensions of the original LLE. These include Robust Locally Linear Embedding (RLLE) which was developed to handle outliers [13], a version of LLE based on Hessian eigenmaps to handle high-dimensional data (HLLE) [8], an incremental version of LLE to preserve topology (ILLE) [14] and Locally Linear embedding with Additive Noise (LLEAN) which is designed to handle data that was corrupted by additive noise [15].

It was also reported that LLE may not be stable since the constrained least squares (LS) problem involved for determining the local weights may be ill-conditioned [16]. In recent years, some extensions of LLE are also proposed to obtain more reasonable reconstruction weights. To avoid the ill-conditioning problem in solving the least squares problem, a regularization parameter is introduced for solving the LS problem manually [17] or automatically [18]. Wang and Zhang (2010) [19] have proved that there are multiple sets of local construction weights that are approximately optimal for solving the LS problem. A recent algorithm called modified LLE is then proposed which exploits the local geometry by

constructing multiple weights to improve the stability of LLE. The study presents a comparison between LLE and a modified version (MLLE) proposed by Wang and Zhang (2010) [19]. The remainder of this paper is organized as follows : In section 2 briefly review the original LLE algorithm and show how the expected underlying manifold geometry is preserved. In section 3 we illustrate the instability of LLE resulted from the uncertain local weight and present the modified LLE algorithm. The key observation is that if a manifold has dimension d $>$ 1, a single set of reconstruction weights may not be able to determine the whole local linearity. The lack of control on the local linearity may result in instability in numerical embedding. In Section 4, we present the different evaluation criteria used for the comparison between LLE and MLLE. In Section 5, experimental results of the multi-class dataset Fashion MNIST and its augmented version are presented. Finally, section 6 offers conclusions and discussions.

II. A BRIEF REVIEW OF LOCALLY LINEAR EMBEDDING

In this section, we first outline the basic steps of the LLE algorithm, then discuss about the preservation of the manifold geometry and finally speak about the intrinsic value d and the current way to find the optimal parameter K.

A. LLE Algorithm

LLE is an unsupervised learning algorithm. It preserves the relationships between neighbors in manifold data and represents high dimensional data in a lower dimensional Euclidean space. LLE maps a dataset $X = \{X_1, X_2, \ldots, X_N\}$, $X_i \in R^D$ globally to a lower dimensional set $Y = \{Y_1, Y_2, \ldots, Y_N\}$, $Y_i \in R^d$ with $d < D$. The algorithm has three steps:

1) Obtain the set of K nearest neighbors for each $\mathcal{X}_i$. Denote this set $\mathcal{N}_i$.
2) Compute constrained weights matrix $W = (w_{ij})_{i,j=1,\ldots,n}$ that best linearly reconstruct X_i from its neighbors $X_i \leftarrow \sum W_{ij} X_j$. The optimal weights are determined by solving the following constrained LS problem :

$$\begin{aligned} &\min \left\| x_i - \sum_{j \in J_i} w_{ij} x_j \right\|^2 \\ &\text{s.t.} \sum_{j \in J_i} w_{ij} = 1 \end{aligned} \tag{1}$$

3) Compute low dimensional embedding vectors $Y_i \in R^d$ best reconstructed by minimizing the cost function

$$\begin{aligned} &\min \sum_{i=1}^{N} \left\| y_i - \sum_{j \in J_i} w_{ij} y_j \right\|^2 \\ &\text{s.t.} \sum_{i=1}^{N} y_i = 0, \frac{1}{N} \sum_{i=1}^{N} y_i y_i^T = I \end{aligned} \tag{2}$$

The two constraints make the embedding cost function be invariant to translations and rescaling. To find the matrix $Y = [y_i, \ldots, y_N]$ under this constraints, a new sparse symmetric and positive semi-definite matrix M is constructed based on the matrix W : $M = (I - W)^T (I - W)$. Now the LLE embedding problem is transformed into the computation of the bottom d nonzero eigenvalues of matrix M.

Let us denote matrix $G_i = [\ldots, x_j - x_i, \ldots]_{j \in J_i}$, we can rewrite the constrained LS problem (2) as

$$\min \|G_i w\|, \quad \text{s.t.} \quad w^T \mathbf{1}_{k_i} = 1 \tag{3}$$

where $\mathbf{1}_{k_i}$ denotes the k_i -dimensional vector of all 1's. This problem is not stable if $G^T G$ is singular (has zero eigenvalues) or nearly singular (has relative small eigenvalues). Typically, eigenvalues and/or eigenvectors of a particular matrix are very sensitive to small perturbations of the matrix which means it is hard to accurately derive the eigenvectors, which correspond to the smallest nonzero eigenvalues. This issue is called ill-conditioned eigenproblem.

In that case it's suggested [17] to solve the regularized linear system with a regularization constant $\gamma \ll 1$:

$$\left(G^T G + \gamma \|G\|_F^2 I\right) y = \mathbf{1}_k, \quad w = y / \mathbf{1}_k^T y \tag{4}$$

One factor that results in the instability of LLE is that the learned linear structure, by using single weight vector at each point, is brittle. LLE may give a wrong embedding even if all weight vectors are well approximated in a high accuracy. It is imaginable if G_i is rank reducible since multiple optimal weight vectors exist in that case. It's from this observation that Wang and Zhang [19] prove that though the exact optimal weight vector may be unique, multiple approximately optimal weight vectors exist.

B. Manifold Geometry

The basic assumption of LLE is that the data is well-sampled and lies on or near a smooth non-linear manifold of lower dimensionality $d \ll D$. There exists then a linear mapping consisting of translation, rotation, and rescaling that maps the high dimensional coordinates of each neighborhood to global internal coordinates on the manifold. In the case of noisy data and outliers, the assumption of local linearity fails and LLE will provide a very poor embedding.

C. Optimal number of nearest neighbors

The original LLE has two parameters to be adjusted: the number K of nearest neighbors for each data point and the dimensionality of the embedded space, d (intrinsic dimensionality of the data manifold or,equivalently, the minimal number of degrees of freedom needed to generate the original data). Visualization means that d is fixed(it is either 1, 2 or 3), so that the only parameter to be estimated is K. The reason for choosing the right K is that a large number of nearest neighbors causes smoothing or elimination of small-scale structures in the manifold. In contrast, small neighborhoods can falsely divide the continuous manifold into disjointed sub-manifolds. [10]

Most extensions of LLE differ in the neighborhood selection since it is the only nonlinear step of the LLE. Some are using other distance metrics [20] [21] [22] and others use other rules to select neighbors [23] [13]. Each method aims to solve an intrinsic drawback of LLE such as robustness to outliers and noise.

In this study we will use the automatic hierarchical [**?**] procedure for finding K_{opt}. The main steps are summarized as follows:

1) Calculate $\varepsilon(\mathbf{W})$ for each $K, K \in [1, K_{\max}]$ according to Eq.1 where K_{max} will be arbitrary chosen
2) Find all minima of $\varepsilon(\mathbf{W})$ and corresponding K 's which compose the set S of initial candidates.
3) For each $K^* \in S$, run LLE and compute a quantitative measure.
4) Select K_{opt} according to :

$$K_{opt} = \arg_i \min_{K_i^*} \left(1 - \rho^2_{\mathrm{p}_x \mathrm{D}_y}\right) \tag{5}$$

This quantitative measure called residual variance (Tenenbaum et al. 2000 [5]), illustrates how well the distance information is preserved. It is a measure of general global topology preservation. The hierarchical method for finding the optimal number of nearest neighbors has the major advantage of computing the eigenvectors only N_S times where $N_S << K_{\max}$. It is clear that this method is faster than the straight forward one that computes all the steps of LLE for each $K \in [1, K_{\max}]$.

D. Estimating the intrinsic dimensionality of data

Many studies proposed different strategies to estimate the intrinsic dimension d [28] [29]. In this study we are more interested in the visualization of the embedded data in lower spaces. Therefore we will choose $d = 2$ for most of the following experimentations.

III. Modified Locally Linear Embedding

The LLE algorithm has a major problem when the number of neighbors is greater than the number of input dimensions as the matrix defining each local neighborhood becomes rank-deficient. LLE solves this by using an arbitrary regularization parameter as in equation 4, which may or may not yield an optimal solution. MLLE addresses this regularization problem by using multiple weight vectors in each neighborhood. In this section we will illustrate the instability of LLE with a toy example and summarize the algorithm of MLLE. Finally, we will compare the computational complexity of both algorithm

A. Illustration of the instability of LLE

Outside the consideration of noise and outliers, LLE does not behave well in datasets with high curvature and/or non-convexity property. To illustrate this argument we consider a synthetic dataset 1 generated in a non-convex domain to highlight the essential improvement of MLLE versus LLE.

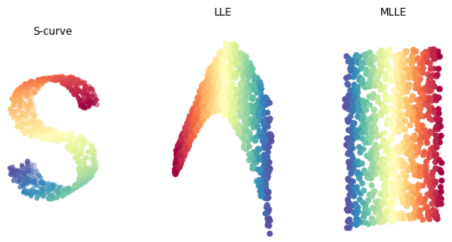

Fig. 1. 3-dimensional S curve with 1000 data points and its 2D embedding using LLE and MLLE. This toy example illustrates the distortion behaviour of LLE with non-convex and high curvature data. MLLE Solves the regularization problem of LLE by using multiple weight vectors in each neighborhood and therefore increasing the stability of the standard LLE. Embedding parameters : $K_{opt} = 10$, $d = 2$

B. MLLE Algorithm

According to Wang and Zhang paper [19] the modified locally linear embedding algorithm can be summarize as follows :

Algorithm 1: Modified Locally linear Embedding

Result: d-dimensional embedding $\{t_1, \ldots, t_N\}$
for *each* $i = 1, \cdots, N$ **do**
  Determine $\mathcal{N}_i = \{x_j, j \in J_i\}$ of $x_i, i \notin J_i$
  Compute the regularized solution $w_i(\gamma)$ by (4)
  Compute the eigenvalues and eigenvectors
  Set $\rho_i = \sum_{j=d+1}^{k_i} \lambda_j^{(i)} / \sum_{j=1}^{d} \lambda_j^{(i)}$
end
Sort $\{\rho_i\}$ to be $\{\rho_{\pi_i}\}$ in increasing order
Set $\eta = \rho_{\pi_{\lceil N/2 \rceil}}$
for *each* $i = 1, \cdots, N$ **do**
  Set s_i by (6)
  Set $V_i = \left[v^{(i)}_{k_i - s_i + 1}, \ldots, v^{(i)}_{k_i}\right], \alpha_i = \left\|\mathbf{1}_{k_i}^T V_i\right\|$
  Construct Φ by using $W_i = w_i(\gamma)\mathbf{1}_{s_i}^T + V_i$
end
pick up the eigenvector matrix corresponding to the 2nd to $(d+1)$th smallest eigenvalues
Set $T = [u_2, \ldots, u_{d+1}]^T$

The number S_i of approximation optimal weight vectors is determined by

$$s_i = \max_{\ell} \left\{ \ell \leq k_i - d, \quad \frac{\sum_{j=k_i-\ell+1}^{k_i} \lambda_j^{(i)}}{\sum_{j=1}^{k_i-\ell} \lambda_j^{(i)}} < \eta \right\} \tag{6}$$

C. Complexity of both Algorithm

The complexity of the regular LLE with N training data points of dimension D for individual steps can be expressed as follows :

1) Finding nearest neighbors - $O(D \log(k) N \log(N))$
2) Computing reconstruction weights - $O\left(DNK^3\right)$
3) Computing bottom eigenvectors - $O\left(dN^2\right)$

where k is the number of nearest neighbors and d the output dimension. The computational cost of MLLE is almost the same as the one of LLE. The additional cost of MLLE comes from the computation of the eigendecomposition of $G_i^T G_i$

and is approximately equal to $O\left(N(k-D)k^2\right)$. In practice, the added cost of constructing the MLLE weight matrix is relatively small compared to the cost of stages of the nearest neighbors and the computation of the bottom eigenvectors because $k \ll N$.

IV. Evaluation Criteria

Historically, distance preservation has been the first criterion used to achieve a dimension reduction in a nonlinear way. From the point of view of an ideal case, the preservation of the pairwise distances measured in a dataset ensures that the low-dimensional embedding inherits the main geometric properties of the data, such as the overall shape. However, techniques such as LLE, MLLE, LE, t-SNE reduce the dimensionality of the data by preserving their topology rather than their pairwise distances. Therefore, to compare the performance of LLE and MLLE we will look at local and global topology preservation. We will also measure the classification capability of the embedded data obtained.

A. Topology Preservation

Spearman's Rho Siegel and Castellan presented one of the first measurements to estimate the topology preservation [25]. This quantitative numerical measure estimates the correlation of rank order data. It tries to assess how well the corresponding projection preserves the order of pairwise distances between data points in a high-dimensional space. Spearman's rho is computed by using the following equation :

$$\rho_{Sp} = 1 - \frac{6\sum_{i=1}^{T}\left(r_x(i) - r_y(i)\right)^2}{T^3 - T} \tag{7}$$

where T is the number of distances to be compared, $r_x(i)$ are the ranks of pairwise distances calculated from the original (n-dimensional) data points and sorted in ascending order, $r_y(i)$ are the ranks of pairwise distances calculated for the projected (d-dimensional) data points and sorted in ascending order. The interval of ρ_{Sp} is $[-1, 1]$. When $\rho_{Sp} = 1$, or $\rho_{Sp} = -1$ there is a perfect positive or negative correlation between the two sets of variables. Therefore the closer ρ_{Sp} is to 1, the better the data topology is preserved in the projected space.

Karbauskaite et al.(2007) [10] demonstrated that ρ_{Sp} is suitable to estimate the topology preservation after visualizing the data by the LLE algorithm. To make this statement true it is necessary to calculate $r_x(i)$ using geodesic distances with a small number of neighbours (≤ 10) and Euclidean or geodesic distances for the calculation of $r_y(i)$. Spearman's Rho is a local neighborhood preservation approach and can also be used to choose the optimal k.

Tenenbaum et al.(2000) [5] used the residual variance (Equ 4) for assessing the overall quality of an embedding and is the value commonly used to choose the optimal K. This criteria is a global structure holding approach.

B. Classification Capability

In the case of labeled data, the classification rate reduction gives a good measurement of the capability of classification of the projected data. It compares the performance of classification with and without dimension reduction. The measurement is given by the following formula :

$$R = \frac{N_x^{\text{correct}} - N_y^{\text{correct}}}{N_x^{\text{correct}}} \tag{8}$$

where N_x^{correct} and N_y^{correct} define the number of correctly classified data samples in the original and projected spaces, correspondingly. To get this value k nearest neighbor classifier is used with different k. The smaller the R, the better the classification capability of the projected data.

To measure the classification quality, the f1-score is a widely used technique that gives a measure of classification accuracy.

V. Experimental Result with Toys dataset and fashion NIMST

In this section we will compare the performance of LLE and MLLE in different aspects.

Firstly, we will use a toy dataset to experiment the research of the optimal K number of neighbours according to local and global measurements of topology preservation. Then we will compute the evolution of the quality of the embedding when increasing the number of data points.

Secondly, we will experiment LLE and MLLE on a more complex dataset; fashion-MNIST from Zalando-research. We will find the optimal K thanks to local topology preservation measurement and visualise the embedding on the full dataset. Then we will compare the classification capability of both algorithm on the full data-set and observe their improvements when increasing the intrinsic parameter d. Afterwards, we will add different types of noise and rotation to the dataset in order to observe the robustness of both algorithm in these particular cases.

Finally, we will compare the clustering properties of both algorithms with a convolutional neural network.

A. The Swiss roll and S-curve

To illustrate the research of the optimal parameter K, we will use the Swiss-Roll with 1000 data points. We compute Spearsman's Rho and the residual variance for each $K \in [4, 40]$ for ten random generations of the Swiss-Roll and compute the average value for each K. The optimal K gives the best embedding. It can be quantified by the highest value ρ_{Sp} and the smallest residual variance as illustrate in the Fig. 2. We observe that ρ_{Sp} and the residual variance are correlated. They are sufficient to find the optimal k. The Fig. 2 compares the best embedding with the worst one according to Spearman's value. Generally, we observe that MLLE has a better topology preservation measure than LLE. It is corroborate by the visualisation of both embedding. The behaviour of the ρ_{Sp} and the residual variance for LLE and MLLE are similar, therefore in that case the range of K_{opt} are almost equivalent for both algorithm.

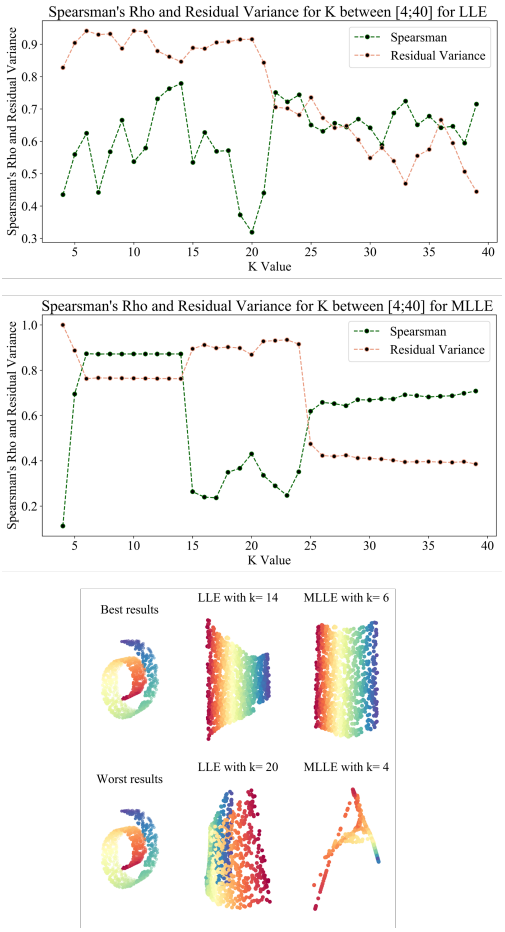

Fig. 2. 3-dimensional Swiss Roll and its 2D embedding using LLE and MLLE. Both graphs show the evolution of Spearsman's Rho and the residual variance for each $K \in [4, 40]$. The Swill Roll has been randomly generated 10 times and the average measure was computed for each value of K. The best embedding is obtained when ρ_{Sp} is high and the residual variance small. In average, MLLE gives a better topology preservation measure than LLE and more smooth embedding visualization

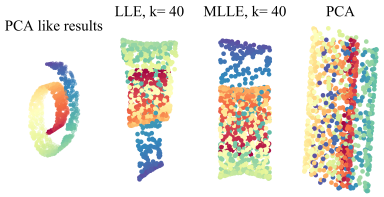

Fig. 3. 3-dimensional Swiss Roll and its 2D embedding using LLE, MLLE and PCA on the two last eigenvectors. The number of nearest neighbours is set high. The mapping of both LLE and MLLE behaves like a regular PCA. In fact, when k is set too high, it causes smoothing or elimination of small-scale structures in the manifold. The mapping loses its nonlinear character

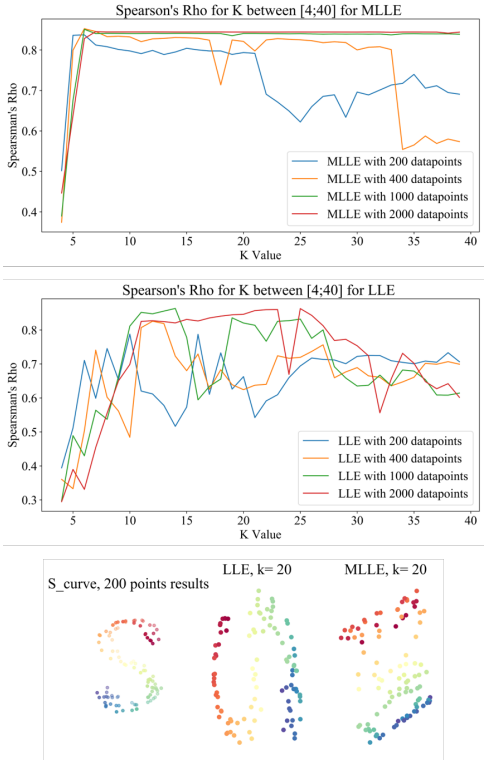

Fig. 4. 3-dimensional S-curve with variation of data points and its 2D embedding using LLE, MLLE. Both graphs show the evolution of Spearsman's Rho with variation of the number of data points. The measure of ρ_{Sp} for each k is the average of 10 random generations. ρ_{Sp} gets higher when the number of data points increases. The visualisation of the mapping with a very low number of points shows how both algorithms struggle with low density samples. In the visualisation, the low density embedding of LLE treats the S-curve as a non-continuous manifold and maps it with almost the curvature of an S.

When k is set too small, a continuous manifold can falsely be divided into disjoint sub manifolds, and thus, the mapping does not reflect any global properties. In contrast, if k is too high, the algorithm will lose its nonlinear character and behave like traditional Principal Component Analysis as in Fig. 3 (Jolliffe, 1989, [1]). Setting a high k also tends to cause a data point to have neighbors that are actually very distant. More intuitively, this can be seen as a short circuit.

When we variate the number of points of the S-curve dataset (Fig. 4), we observe that the overall Spearman's Rho value increases for both algorithms. In fact if the sample density is low, LLE and MLLE are unavoidable to derive the non-uniform warps and folds. Both algorithms need enough nearest neighbours to catch the non linearity of the dataset. In Fig. 4 we observe that K_{opt} is getting bigger as we increase the number of points and it is smaller for MLLE. It shows the robustness of MLLE with low density datasets.

In summary of our analysis on LLE and MLLE with toy datasets, we conclude that MLLE regularisation is stronger than LLE to map datasets into a lower dimension embedding. MLLE is not limited to convex and low curvature data. Overall, MLLE preserves better the local and global topology of the dataset regarding Spearman's Rho and the residual variance. Moreover, it shows a stronger performance with low density dataset. The choice of K_{opt} depends on the measure of the topology preservation. Generally, it appears that a range of values are optimal. On the other hand, K_{opt} should not be chosen too high otherwise the algorithm will lose its nonlinear character and behave like a traditional PCA.

B. *Fashion-MNIST*

Fashion MNIST [27] is a dataset with 70'000 images of clothes from Zalando in low resolution and in greyscale (Fig.5). The images of clothes are labeled in 10 distinct classes : T-shirt/top, Trouser, Pullover, Dress, Coat, Sandal, Shirt, Sneaker, Bag, Ankle boot. Zalando intends Fashion-MNIST to serve as a direct drop-in replacement for the original MNIST dataset for bench marking machine learning algorithms.

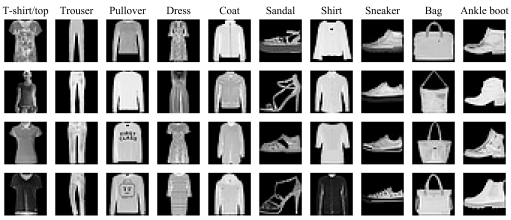

Fig. 5. Samples from Fashion MNIST dataset.

Each sample is a 784 dimension data point (28x28 greyscale image) with the class label. To get a better visualisation and overcome the curse of dimensionality for classification technique, we apply a non linear dimensional reduction.

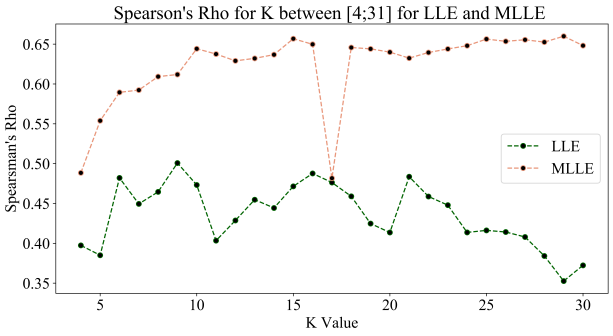

Fig. 6. Spearsman's Rho in function of k for MLLE and LLE for the fashion MNIST dataset. For each k we compute 10 times LLE on the dataset and measure the average ρ_{Sp} . For the rest of the paper we will take K_{opt} = 22 because it has given good performances

Fig.7, shows the embedding of a subset of 2000 data points of fashion MNIST on the first two coordinates of LLE and MLLE. The ten classes have 200 samples of different elements. The embedding has been processed with $K_{opt} = 22$ and $d = 2$ according to the results obtained in Fig. 6. The visualization of both algorithms shows many characteristics of the dataset. We observe three peaks represented by trousers, shoes and pullover labels. In fact shoes have mostly grey pixels in the horizontal, Trouser in a thin region in the vertical and Pullover in a big region of the vertical.

In terms of visualisation (Fig.7) and topology preservation (Fig.6), MLLE gives better results than LLE. To proceed with the classification using KNN, we did a 10 fold cross-validation with a split ratio of 0.2 (training : 10'000 data points / testing : 2'000 data points.). The classification rate reduction for LLE and MLLE are respectively 0.141 and 0.129. The F1 measurement analysis in (Fig.8) shows better results for MLLE versus LLE. The confusion matrix (Fig.9) shows that wrong classification concerns the same couple of classes for both algorithms.

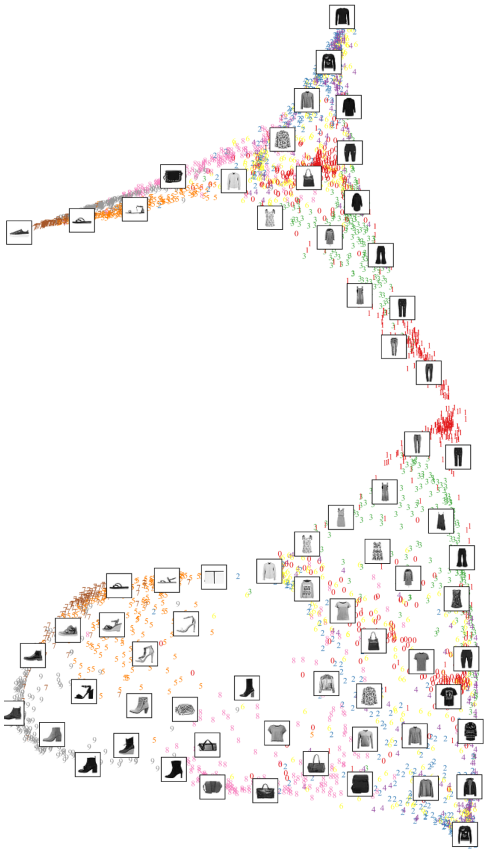

Fig. 7. Images of clothes (Fig.5) mapped into the embedding space described by the first two coordinates of LLE (top image) and MLLE (bottom image). Each colored number represents the class label. Images of clothes have been shown for few samples. Embedding parameters : $K_{opt} = 22$, $d = 2$

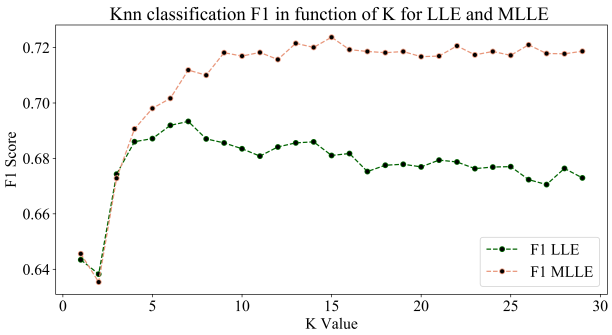

Fig. 8. F1 score on the embedding of the full dataset in function of K for LLE and MLLE. A 10 fold cross validation has been done to calculate accurately the F1 score for each K. MLLE mapping gives better classification performance in terms of F1. Embedding parameters : $K_{opt} = 22$, $d = 2$

Let's consider the performance of classification with different values of the intrinsic dimension d. Fig.10 shows that both algorithms get better F1 scores with the increase of d. Nevertheless, d is strictly upper-bounded by $K_{opt} = 22$. In

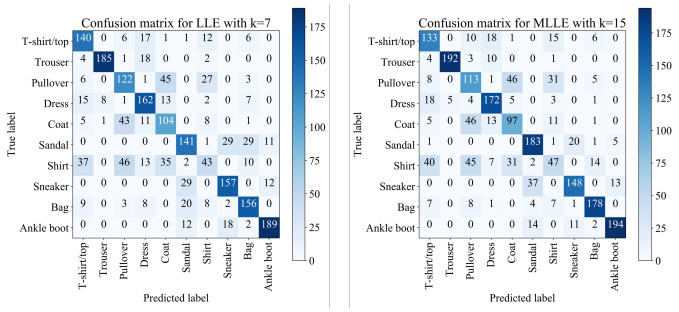

Fig. 9. Confusion matrix for the best classification situation of both algorithms. We observe that the wrong classification concerns more or less the same couples of classes for both algorithms. Embedding parameters : $K_{opt} = 22$, $d = 2$

fact, the performance of both algorithms with respect to F1 measurements gets very similar when the intrinsic dimension increase. Considering that our dataset is dense enough, this result shows that MLLE is more robust for classification than LLE when the chosen d in far from reflecting the real intrinsic dimension of the dataset. On opposite, when d reflects better the intrinsic dimension, both algorithms have the same performance in terms of classification on the Fashion MNIST dataset.

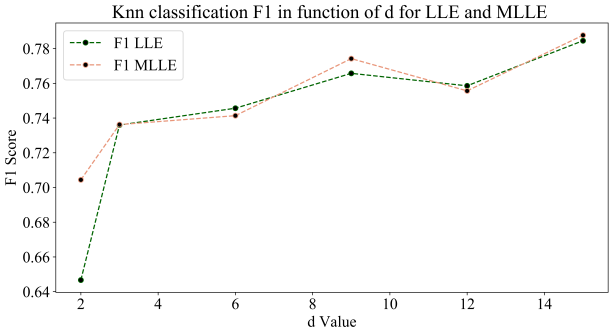

Fig. 10. F1 score on the embedding of the full dataset in function of d for LLE and MLLE. A 10 fold cross validation has been done to calculate accurately the F1 for each d. Both mappings give better classification performances in terms of F1 as d gets higher. Embedding parameters : $K_{opt} = 22$, $K_{knn} = 15$

We now compare the classification performance of LLE and MLLE with fashion MNIST under three types of noise. In Table I we measured the F1 score and the classification rate reduction with a KNN classifier. For every experience, the K nearest neighbours for the classifier have been chosen in order to get the highest F1 score. We observe in table I that both algorithms behave similarly with noisy data. The mapping is weaker with noisy data. Concerning the classification rate, it depends on the classifier performance with noise. To overcome noise, image filtering is generally used before processing.

Next we consider the class sneakers with 10 elements and make 100 rotations of 3.6 degrees to each element of the class. This example explores how strong both algorithms are when applying linear transformations on a dataset with few different items. The Spearman's rho measure for LLE and MLLE are

Noise Type	Embedding	F1	classification rate
None	LLE—MLLE	**0.695—0.721**	**0.141—0.129**
Salt and Paper 10%	LLE—MLLE	0.677—0.711	0.170—0.132
Gaussian $\sigma^2 = 100$	LLE—MLLE	0.670—0.670	0.090—0.080
Gaussian $\sigma^2 = 256$	LLE—MLLE	0.500—0.489	0.090—0.130

Table I. Classification performance with noise. The classification has been done 10 times for each noise and the average result has been plotted. The ratio salt and pepper is 0.5 and it has been applied on 10% of the pixels of each image. The Gaussian is centered with $\sigma^2 = 100$ and $\sigma^2 = 256$. Embedding parameters : $K_{opt} = 22$, $d = 2$

respectively 0.94 and 1. Regarding the topology preservation results and the visualisation of the mapping (Fig.11) MLLE has a better performance. It is difficult for LLE to handle rotation on different items. The perfect circle of the mapping of MLLE in Fig.11 illustrates well the better preservation of local linearity. In fact, MLLE is proven to preserve better local linearity and being more stable than the original LLE.

Fig. 11. Left and right are respectively LLE and MLLE embeddings of 10 different sneakers with data augmentation. Each sneaker has been repeated 100 times with small rotations. Embedding parameters : $K_{opt} = 22$, $d = 2$

VI. LLE AND MLLE IN CONVOLUTIONAL NEURAL NETWORKS

To complete our study on these two manifold methods, we will compare the classification performance of both algorithms with a completely different method : a convolutional neural network. Here we will variate the intrinsic parameter d and compute the accuracy of the classification. The CNN model has been applied with the following properties : a first fully connected 32 neurons dense layer with ReLU as the activation function and a 10 softmax layer. The model is trained and validated with respectively 48000 and 12000 data points. The performance is evaluated on a testing set of 10000 data points. Our CNN model has an accuracy of 0.91 when trained and tested on the dataset without embedding. The Fig. 12 shows the same properties than the classification with KNN (Fig. 10); both LLE and MLLE follow the same growth pattern when increasing the dimension. This time, LLE shows a slightly better performance even for a weak choice of parameter (d =2). For the following we will focus on the overall classification results with d = 14. The table II shows how each class impacts the overall results of the classification. As in the confusion

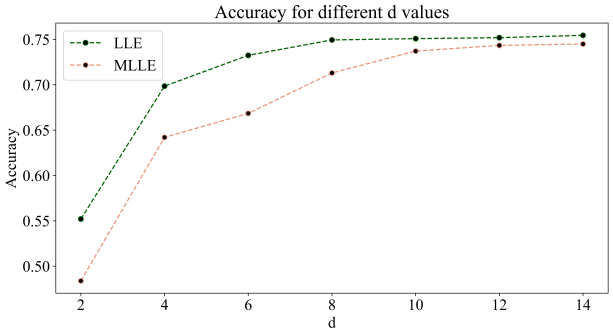

Fig. 12. Prediction accuracy growth according to intrinsic dimension hyperparameter d for LLE and MLLE. Embedding parameters : $K_{opt} = 22$

matrix in Fig. 9, we observe that the wrong classification concerns more or less the same couples of classes for both algorithms, which are Pullover, Shirt and Coat. In comparison, without embedding the CNN model has a score of 0.7 for shirt and 0.8 for Coat and Shirt.

Class	Embedding	precision	recall	F1 score
T-shirt/top	LLE—MLLE	0.72—0.71	0.76—0.77	0.74—0.74
Trouser	LLE—MLLE	0.98—0.99	0.94—0.94	0.96—0.96
Pullover	LLE—MLLE	**0.51—0.50**	**0.68—0.65**	**0.58—0.56**
Dress	LLE—MLLE	0.72—0.74	0.88—0.85	0.79—0.79
Coat	LLE—MLLE	**0.53—0.56**	**0.46—0.52**	**0.49—0.54**
Sandal	LLE—MLLE	0.90—0.85	0.84—0.82	0.87—0.83
Shirt	LLE—MLLE	**0.42—0.39**	**0.23—0.22**	**0.30—0.28**
Sneaker	LLE—MLLE	0.83—0.79	0.84—0.80	0.83—0.80
Bag	LLE—MLLE	0.95—0.94	0.94—0.94	0.94—0.94
Ankle boot	LLE—MLLE	0.86—0.87	0.92—0.90	0.89—0.88

Table II. Classification report of both algorithms. Embedding parameters : $K_{opt} = 22$, $d = 14$

d	Embedding	Embedding time	Training time	F1
784	None	00:00:00	00:15:23	0.92
14	LLE	03:08:03	00:00:01	0.75
	MLLE	03:10:19	00:00:01	0.74

Table III. CNN Computational time with and without embedding. Embedding parameters : $K_{opt} = 22$, $d = 14$

The results obtained in table III shows that it took up to 12 times longer to complete the experience with embedding than without it. The values itself may differ from a machine to another but we clearly observe that classification is almost instantaneous after embedding. It's also interesting to observe that the embedding time of both algorithms are almost equal as the complexity of both algorithms are almost the same. Even though the embedding takes a huge time, it has to be done only once. It can allow to test and optimize faster deep learning parameters. It means that we may want to use the embedded version of the dataset to test several convolutional neural network models with very fast epochs until we are satisfied to eventually test the model with the original data.

VII. Conclusion and discussion

The classic non linear reduction method LLE proved its performance and convenience requiring only two hyperparameters to be determined. However, some intrinsic drawbacks such as its assumptions on a low curvature and convex manifold, inevitable ill-conditioned eigenproblems and sensitivity to noise restricts its application. Wang and Zhang (2010) [19] have propose an extension of LLE called modified locally linear embedding which propose a more reasonable reconstruction weights to avoid the ill-conditioning problem in solving the least square problems. In this paper, With the help of toy datasets and Fashion-MNIST dataset, we illustrated how MLLE exploits the local geometry by constructing multiples weights to improve the stability of LLE. The comparison has been shown trough measure of topology preservation and visualisation. However, both algorithms are very sensitive to the choice of K. When too small, a continuous manifold can falsely be divided into disjoint sub manifolds, and thus, the mapping does not reflect any global properties. In contrast, if K is set too high, the algorithm will lose it's nonlinear character and behave like a traditional Principal Component Analysis. The number of data points is as well crucial since both algorithms need dense datasets to catch the local linearity. Nevertheless, we observe that MLLE is stronger than LLE as the standard method is more sensitive to the choice of parameters than its modified version. In terms of visualisation and topology preservation, MLLE gets a better score than LLE although both algorithms were able to extract the main features of the dataset. This has been seen through the good classification performance of the mapping with KNN classifier. Some experiences have shown that the classification performance of both algorithms are highly impacted by the choice of the hyperparameter d. With weak choice of d, MLLE has a better performance in terms of F1 score but when the parameter d reflects better the real intrinsic dimension of the data, both algorithm show the same performance. When adding noise to the dataset, both algorithms get weaker and give almost same performance in terms of classification. To illustrate how MLLE is more stable than LLE in complex situations, we applied rotations on the sneakers class. We observed that MLLE has a better visual and topology preservation than LLE when more than one item are rotated. Finally, the classification performance of LLE with a Convolutional Neural Network using embeddings on the full data shows slightly better performance even for weak choice of parameter d than MLLE. The initial computational cost of LLE and MLLE was higher than the total training time of the CNN model but they might be useful to test several models on the embedded dataset with a faster training before using the whole dataset with all of its dimensions.

To conclude our study, we showed that MLLE is stronger than LLE with weak parameters and in complex situations. With well adjusted parameters and not too complex data set, the performance of both algorithms is very similar.

VIII. Appendix

References

[1] I. T. Jolliffe, *Principal Component Analysis.* New York, NY: Springer-Verlag, 2002.

[2] Lei, Y., Xu, Y., Yang, J. et al. *Feature extraction using orthogonal discriminant local tangent space alignment.* Pattern Anal Applic 15, 249–259 (2012). https://doi.org/10.1007/s10044-011-0231-0

[3] Zhu-Hong You, Ying-Ke Lei, Jie Gui, De-Shuang Huang, Xiaobo Zhou, *Using manifold embedding for assessing and predicting protein interactions from high-throughput experimental data*, Bioinformatics, Volume 26, Issue 21, 1 November 2010, Pages 2744–2751, https://doi.org/10.1093/bioinformatics/btq510

[4] Gui J., Wang C., Zhu L. (2009) *Locality Preserving Discriminant Projections*. In: Huang DS., Jo KH., Lee HH., Kang HJ., Bevilacqua V. (eds) Emerging Intelligent Computing Technology and Applications. With Aspects of Artificial Intelligence. ICIC 2009. Lecture Notes in Computer Science, vol 5755. Springer, Berlin, Heidelberg

[5] Tenenbaum, J.B., V. Silva and J.C. Langford (2000). *A global geometric framework for nonlinear dimensionality reduction*. Science, 290, 2319–2323.

[6] Roweis, S.T., and L.K. Saul (2000). *Nonlinear dimensionality reduction by locally linear embedding.* Science, 290, 2323–2326.

[7] M. Belkin, P. Niyogi. *Laplacian eigenmaps and spectral techniques for embedding and clustering.* In T.G. Dietterich, S. Becker and Z. Ghahramani (eds.), Advances in Neural Information Processing Systems 14. MIT Press, 2002

[8] Donoho, D.L., and C. Grimes (2005). *Hessian eigenmaps: New locally linear embedding techniques for highdimensional data.* Proceedings of the National Academy of Sciences, 102(21), 7426–7431.

[9] Z. Zhang, H. Zha, *Principal manifolds and nonlinear dimensionality reduction via tangent space alignment*, SIAM J. Sci. Comput. 26 (1) (2005) 313–338.

[10] Karbauskait˙e, R., O. Kurasova and G. Dzemyda (2007). *Selection of the number of neighbours of each data point for the locally linear embedding algorithm.* Information Technology and Control, 36(4), 359–364.

[11] Yin, J., D. Hu and Z. Zhou (2007). *Growing locally linear embedding for manifold learning.* Journal of Pattern Recognition Research, 2(1), 1–16.

[12] J. Xiao, Z. Zhou, D. Hu, J. Yin, and S. Chen, *Self-organizing locally linear embedding for nonlinear dimensionality reduction*, in L. Wang, K. Chen, and Y.S. Ong (Eds.): ICNC 2005, LNCS 3610, pp. 101-109, 2005.

[13] Zhang, Yansheng Ye, Dong Liu, Yuanhong. (2017). *Robust locally linear embedding algorithm for machinery fault diagnosis.* Neurocomputing. 273. 10.1016/j.neucom.2017.07.048.

[14] Kouropteva O, Okun O, Pietikäinen M (2005) *Incremental locally linear embedding*. Pattern Recognit 38(10):1764–1767

[15] Wang, Justin Wong, Raymond Lee, Thomas. (2019). *Locally linear embedding with additive noise.* Pattern Recognition Letters. 123. 10.1016/j.patrec.2019.02.030.

[16] J. Wang, Z. Zhang,*Nonlinear embedding preserving multiple local-linearities*, Pattern Recognit. 43 (2010) 1257–1268.

[17] L. Saul, S.Roweis,*Think globally, fitlocally: unsupervised learning ofnonlinear manifolds*, J. Mach. Learn. Res. 4 (2003) 119–155.

[18] G. Daza-Santacoloma, C.D. Acosta-Medina, G. Castellanos-Domínguez, *Regularization parameter choice in locally linear embedding*, Neurocomputing 73 (2010) 1595–1605.

[19] Zhang, Zhenyue & Wang, Jing. (2006). *MLLE: Modified Locally Linear Embedding Using Multiple Weights.* Adv Neural Inf Process Syst. 19. 1593-1600.

[20] Varini C, Degenhard A, Nattkemper TW (2006) *ISOLLE: LLE with geodesic distance.* Neurocomputing 69(13–15):1768–1771

[21] Pan Y, Ge SS, Al Mamun A (2009) *Weighted locally linear embedding for dimension reduction.* Pattern Recognitn 42(5):798–811

[22] Zhou CY, Chen YQ (2006) *Improving nearest neighbor classification with cam weighted distance.* Pattern Recognit 39(4):635–645

[23] Park J, Zhang Z, Zha H, Kasturi R (2004) *Local smoothing for manifold learning.* CVPR I452-I459

[24] O. Kouropteva, O. Okun, M. Pietikainen. *Selection of the optimal parameter value for the locally linear embedding algorithm.* Proc. of 2002 International Conference on Fuzzy Systems and Knowledge Discovery, 2002, 359–363.

[25] S. Siegel, N. Castellan, Nonparametric *Statistics for the Behavioral* Sciences, McGraw-Hill Inc., 1988.

[26] Yin J, Hu D, Zhou Z (2008) *Noisy manifold learning using neighborhood smoothing embedding.* Pattern Recognit Lett 29(11):1613–1620

[27] Han X, Kashif R, Roland Vollgraf (2017) *Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms* arXiv:1708.07747

[28] Brand M (2003) *Charting a manifold.* In: Becker S, Thrun S Obermayer K (eds) Advances in Neural Information Processing Systems, volume 15, 961–968. Cambridge, MA, MIT Press.

[29] Kegl B (2003) Intrinsic dimension estimation using packing numbers. In: Becker S, Thrun S Obermayer K (eds) Advances in Neural Information Processing Systems, volume 15, 681–688. Cambridge, MA, MIT Press.