Effective MVU via Central Prototypes and Kernel Ridge Regression

Carlotta Orsenigo(✉)

Dept. of Management, Economics and Industrial Engineering,
Politecnico di Milano, via Lambruschini 4b, 20156 Milano, Italy
carlotta.orsenigo@polimi.it

Abstract. Maximum variance unfolding (MVU) is one of the most prominent manifold learning techniques for nonlinear dimensionality reduction. Despite its effectiveness it has proven to be considerably slow on large data sets, for which fast extensions have been developed. In this paper we present a novel algorithm which combines classical MVU and multi-output kernel ridge regression (KRR). The proposed method, called Selective MVU, is based on a three-step procedure. First, a subset of distinguished points indicated as central prototypes is selected. Then, MVU is applied to find the prototypes embedding in the low-dimensional space. Finally, KRR is used to reconstruct the projections of the remaining samples. Preliminary results on benchmark data sets highlight the usefulness of Selective MVU which exhibits promising performances in terms of quality of the data embedding compared to renowned MVU variants and other state-of-the-art nonlinear methods.

Keywords: Nonlinear dimensionality reduction · Manifold learning · Maximum variance unfolding · Prototype selection

1 Introduction

Dimensionality reduction is the process of converting high dimensional data into meaningful representations of reduced dimensionality. As a preliminary step it plays a fundamental role in several machine learning tasks by favoring data visualization, clustering and classification. Dimensionality reduction techniques are usually divided into linear and nonlinear approaches. Within the family of nonlinear methods manifold learning algorithms have drawn great interest by attempting to recover the low dimensional manifold along which data are supposed to lie. These include, among others, Isometric feature mapping [1], Locally linear embedding [2], Laplacian eigenmaps [3], Local tangent space alignment [4] and Maximum variance unfolding [5].

Maximum variance unfolding (MVU), also known as Semidefinite embedding, relies on the notion of isometry which can be defined as a smooth invertible mapping that behaves locally like a rotation plus a translation. The final low-dimensional embedding is therefore locally-distance preserving, since it is derived

V. Torra and Y. Narakawa (Eds.): MDAI 2015, LNAI 9321, pp. 143–154, 2015.
DOI: 10.1007/978-3-319-23240-9_12

by keeping unchanged the distances and angles between neighboring points. The unfolding process requires the solution of a semidefinite program (SDP) which maximizes the variance of the points in the feature space, represented by the trace of the corresponding Gram matrix, under linear equality constraints which impose the local-isometry conditions.

Despite its effectiveness MVU turns out to be considerably slow when the number of points increases since solving large semidefinite programs is time-consuming. To overcome this drawback different approaches can be followed. One may resort to greedy optimization procedures, as those described in [6] and [7], able to efficiently achieve the global optimum of semidefinite programs on large data sets, or to iterative algorithms which transform graph embeddings into MVU feasible solutions [8]. A further strategy to speed up the algorithm is to reduce the original SDP to a smaller problem by means of the Gram matrix factorization. Based on this last approach two fast variants have been developed. In the first the Gram matrix is reconstructed from a smaller submatrix of inner products between randomly chosen landmarks [9]. In the second variant matrix factorization is obtained by expanding the solution of the initial SDP in terms of the bottom eigenvectors of the graph Laplacian [10].

In this paper we present a novel method for nonlinear dimensionality reduction which combines MVU and kernel ridge regression [11] and [12]. The proposed algorithm, called Selective MVU, is based on a three-step procedure. A subset of distinguished points, indicated as prototypes, is first selected from the original data set. Classical MVU is then applied on the collection of prototypes to find their embedding in the low d-dimensional space. The projections of the remaining points are finally derived by learning the nonlinear mapping through multi-output kernel ridge regression (KRR), which has been successfully used as out-of-sample extension for manifold learning [13]. The proposed algorithm enables significant computational savings compared to classical MVU that is in this case applied only to the set of representative points. It draws inspiration from both landmark methods for fast embedding [14] and [9], which place a point in the feature space according to its distance from the projected landmarks, and the spectral regression paradigm [15], in which the subspace learning problem is cast into a regression framework. To designate the collection of prototypes we also propose a novel method based on K-means algorithm that behaves more effectively than random selection. Experiments on eight benchmark data sets highlight the usefulness of Selective MVU which provides promising results compared to well-known MVU fast variants and other prominent nonlinear dimensionality reduction techniques.

The remainder of the paper is organized as follows. Section 2 briefly recalls maximum variance unfolding. Section 3 presents the novel Selective MVU algorithm and the prototype selection method. Computational experiments and results are described in Sect. 4. Conclusions and future developments are discussed in Sect. 5.

2 Maximum Variance Unfolding

Maximum variance unfolding imposes local-isometry constraints aimed at preserving both distances and angles between points and their neighbors, in order to find low-dimensional projections which faithfully represent the input data.

Let $S_m = \{\mathbf{x}_i,\ i \in \mathcal{M} = \{1, 2, \ldots, m\}\} \subset \Re^n$ be a set of m points approximately confined to a nonlinear manifold of intrinsic dimension d ($d \ll n$). The unfolding process starts with the construction of the neighborhood graph in which nodes represent data points and edges neighborhood relations. Then, it requires the solution of a quadratic optimization problem which maximizes the variance of the embedding subject to the local-isometry conditions. In practice, the problem is reformulated as the following semidefinite program over the Gram matrix $\mathbf{G}_m = [g_{ij}]$ of the points in the feature space, with $g_{ij} = \langle \mathbf{z}_i, \mathbf{z}_j \rangle, \forall i, j \in \mathcal{M}$,

$$\max_{\mathbf{G}} \ \operatorname{tr}(\mathbf{G}) \tag{SD}$$

$$\text{s.to} \quad g_{ii} + g_{jj} - 2g_{ij} = d_{ij}^2 \qquad \forall i, j \in \mathcal{M},\ \eta_{ij} = 1, \tag{1}$$

$$\sum_{i,j \in \mathcal{M}} g_{ij} = 0, \tag{2}$$

$$\mathbf{G} \succeq 0, \tag{3}$$

where $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$ and the coefficient $\eta_{ij} \in \{0, 1\}$ takes the value 1 if $\mathbf{x}_j$ is among the k-nearest neighbors of $\mathbf{x}_i$ or $\mathbf{x}_i$ and $\mathbf{x}_j$ are common neighbors of another point in the data set. The first constraints of problem SD preserve the distances between neighboring points. The second yields a unique solution by centering the projections on the origin and the third forces the Gram matrix to be positive semidefinite. The objective function, finally, maximizes the trace of $\mathbf{G}$ which is tantamount to maximizing the total variance of the points in the low-dimensional space.

Once the matrix $\mathbf{G}$ is learned via semidefinite programming the final embedding is obtained by computing its d largest eigenvalues and setting the projections to $\mathbf{Z} = \mathbf{V}\mathbf{\Lambda}^{1/2}$, where $\mathbf{Z}_{m \times d}$ is the matrix of embedded vectors $\mathbf{z}_i$, $\mathbf{\Lambda}_d$ is the square diagonal matrix of leading eigenvalues and $\mathbf{V}_{m \times d}$ is the matrix of corresponding eigenvectors.

Although efficient solvers for semidefinite programming exist, problem SD hardly scales to large data sets. The computational effort increases with the number of constraints and the size of $\mathbf{G}$. It is possible to show, however, that for well-sampled manifolds the Gram matrix can be reasonably approximated as the product of smaller matrices $\mathbf{G} \approx \mathbf{Q}\mathbf{Y}\mathbf{Q}'$, where $\mathbf{Q}_{m \times l}$ ($l \ll m$) must be properly determined. This results in a semidefinite program over the square matrix $\mathbf{Y}$ of size l, which has to be optimized under the local distance constraints. The low-rank expansion of $\mathbf{G}$ represents the key point of two fast MVU extensions given by Landmark MVU (L-MVU) [9] and Graph Laplacian Regularized MVU (GL-MVU) [10].

3 Selective MVU

The distinctive traits of MVU are the maximization of the variance of the embedding and the preservation of the distances between neighboring points. L-MVU and GL-MVU algorithms represent a substantial improvement over classical MVU from the computational viewpoint. However, they both diverge from the original paradigm due to the Gram matrix factorization.

In this paper we present a novel MVU extension, called Selective MVU (S-MVU), in which the required computing effort is reduced according to a different framework. Instead of resorting to modified MVU formulations applied to the entire set of data, the proposed method uses classical MVU to find the embedding of a collection of distinguished points indicated as prototypes. The low-dimensional coordinates of the remaining samples are then reconstructed via multi-output kernel ridge regression (KRR) [13]. The aim of this study, therefore, is to empirically investigate whether the solution of the original MVU model over a subset of representative points combined with an accurate regression method for learning the nonlinear mapping may provide higher quality low-dimensional projections compared to both MVU fast variants. The proposed Selective MVU algorithm can be summarized as follows.

Procedure. Selective MVU (S-MVU)

1. Define a collection $P \subseteq S_m$ of prototypes, where $\text{card}(P) = p$. Let $\mathcal{P} \subseteq \mathcal{M}$ be the set of their indices.
2. Find the embedding of P by solving problem SD over the corresponding Gram matrix $\mathbf{G}_p = [g_{ij}]$, where $g_{ij} = \langle \mathbf{z}_i, \mathbf{z}_j \rangle, \forall i, j \in \mathcal{P}$. Then, set $\mathbf{Z} = \mathbf{V}\mathbf{\Lambda}^{1/2}$, where $\mathbf{Z}_{p \times d}$ contains the projections of the prototypes in the feature space, $\mathbf{\Lambda}_d$ collects the d leading eigenvalues of $\mathbf{G}$ and $\mathbf{V}_{p \times d}$ the corresponding eigenvectors.
3. Learn the mapping via multi-output kernel ridge regression. To this aim, define a Mercer kernel $\rho : \Re^n \times \Re^n \mapsto \Re$ inducing a nonlinear projection $\phi : \Re^n \mapsto \mathcal{H}$ from the original input space $\Re^n$ to a Hilbert space $\mathcal{H}$. Formulate the kernel ridge regression model as $\min \|\mathbf{Z} - \langle \mathbf{\Phi}, \mathbf{W} \rangle\|_F^2 + \lambda \|\mathbf{W}\|_{\mathcal{H}}^2$, where $\|\cdot\|_F$ is the Frobenius norm of a matrix, the vector $\mathbf{\Phi}$ collects the images $\phi(\mathbf{x}_i)$, $i \in \mathcal{P}$, in $\mathcal{H}$ and the parameter λ controls the trade-off between the error and the penalty term. The regression coefficients can be computed in close form as $\mathbf{W} = \mathbf{\Phi}' (\mathbf{U} + \lambda \mathbf{I}_p)^{-1} \mathbf{Z}$, where $\mathbf{I}_p$ is the identity matrix of size p and $\mathbf{U}_p = [u_{ij}]$ is the kernel matrix associated to ρ, with $u_{ij} = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$, $\forall i, j \in \mathcal{P}$.
4. Embed the other points $\mathbf{x}_k, k \in \mathcal{M} \backslash \{\mathcal{P}\}$, by setting $\mathbf{z}_k = \langle \mathbf{W}', \phi(\mathbf{x}_k) \rangle = \mathbf{Z}' (\mathbf{U} + \lambda \mathbf{I}_p)^{-1} \mathbf{T}(\mathbf{x}_k)$, where the generic element of the p-dimensional vector $\mathbf{T}$ is given by $t_j = \langle \phi(\mathbf{x}_j), \phi(\mathbf{x}_k) \rangle$, $j \in \mathcal{P}$.

Notice that, the distance-preserving constraints of problem SD are in this case imposed only to the collection of prototypes, which are the pivotal elements for data embedding.

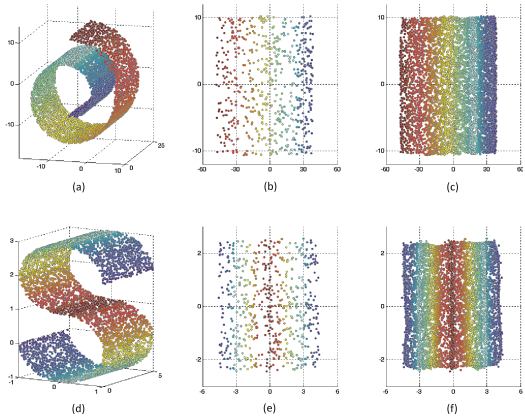

Fig. 1. Embedding of two artificial data sets. Panels (a) and (d) illustrate the Swiss roll and the S-curve data sets in the original three-dimensional space. Panels (b) and (e) show the two-dimensional projection of the randomly selected prototypes by means of MVU. Panels (c) and (f) depict the final mapping obtained by applying KRR on the embedded prototypes.

To illustrate the projection based on Selective MVU we applied the proposed algorithm to two artificial data sets obtained by sampling 6000 points from a Swiss roll and a S-curve surface, respectively. In particular, we computed the two-dimensional embedding from the three-dimensional space by setting $k = 6$, using the radial basis function (RBF) kernel for KRR and randomly choosing 10 % of the available points as prototypes. The projections obtained by S-MVU are depicted in Fig. 1. As we may observe, although based on a very small number of representative points the final mapping of both data sets faithfully correspond to the structure of the manifold in the native three-dimensional space.

3.1 Central Prototypes Selection

The most straightforward way to designate the set of prototypes is to select them randomly. Random selection is usually applied to identify landmark points in landmarks-based manifold learning algorithms [14] and [9]. However, it may generate misleading data projections [16] and [17], especially when data are affected by noise.

To find the collection of representative points we resorted to a simple but effective procedure based on clustering. In particular, we first applied K-means algorithm to partition the points into K distinct clusters. From each cluster we then selected a predefined number of central prototypes, defined as the points for which the maximum distance from the other points in the cluster is minimized. The algorithm can be summarized as follows:

Procedure. Central Prototypes Selection (CPS)

1. Let $\gamma = p/m$ be the fraction of points in S_m to use as prototypes.
2. Identify a set of K points as initial seeds and partition S_m into K clusters by applying K-means algorithm.

3. For each cluster $C_h, h = 1, 2, \ldots, K$, sort the points in ascending order based on their maximum distance from the other points in the cluster. Select the first $\lfloor \gamma \cdot \text{card}(C_h) \rfloor$ points from the list, where $\lfloor \cdot \rfloor$ denotes the integer part, and insert them in the set P of desired prototypes.

The initial seeds for K-means clustering at step 2 were computed through a multivariate variant of the algorithm proposed in [18], which introduces a measure of distance between cluster centers and virtually reduces to zero the variance of different runs. Indeed, it provides the same initial clusters across multiple experiments by excluding any form of randomness. The algorithm was originally conceived for clustering along a single dimension. The multivariate extension considered in this study is described by the following procedure.

Procedure. Seeds Selection (SS)

1. Sort the points in S_m in terms of increasing magnitude, given by their norms $\|\mathbf{x}_i\|$, $i \in \mathcal{M}$. Let F be the set of sorted points.
2. Compute the distances $D_j = \left\|\mathbf{x}^{j+1} - \mathbf{x}^j\right\|$, $j = 1, 2, \ldots, m-1$, between all pairs of consecutive points, where $\mathbf{x}^j$ denotes point $\mathbf{x}$ at position j in F.
3. Identify the indices $\left\{j_1, j_2, \ldots, j_{(K-1)}\right\}$ corresponding to the $K-1$ highest distance values and sort them in ascending order. Define the sets of indices $\mathcal{U} = \left\{j_1, j_2, \ldots, j_{(K-1)}, j_K\right\}$ and $\mathcal{V} = \left\{j_0, j_1 + 1, \ldots, j_{(K-1)} + 1\right\}$ of the points serving as upper and lower bounds, respectively, where $j_K = m$ and $j_0 = 1$.
4. Compute the K initial seeds as the mean vectors between the upper and lower bound points defined above.

To highlight the usefulness of central prototypes selection we considered a data set composed by 6000 points randomly sampled from a Swiss roll manifold with 1% of uniform distributed outliers, and computed the low-dimensional embedding by means of alternative techniques. In particular, we analyzed the effect of randomness in the worst case scenario when outliers are used as landmarks in L-MVU and prototypes in S-MVU. The different embeddings, obtained by setting $k = 6$ for all methods, are illustrated in Fig. 2. As one may notice, the presence of noise interferes with the unfolding process and induces a major distortion when the projections are based on the outliers (Panels d and e). The use of central prototypes in S-MVU, however, mitigates this effect preserving the structure of the underlying manifold (Panel f). As shown in Fig. 2, a major robustness of S-MVU compared to GL-MVU and L-MVU was also observed when injecting 10% of outliers (Panels g, h and i).

3.2 Complexity of Selective MVU

Classical MVU runs in $O\left(m^3 + c^3\right)$ over a set of m points, where c is the number of constraints in the semidefinite program [5]. The time-complexity of KRR is $O\left(p^3\right)$ [13] whereas the prototype selection procedure runs in $O\left(m^2 + mKIn\right)$, where I is the number of iterations in the K-means algorithm and the quadratic

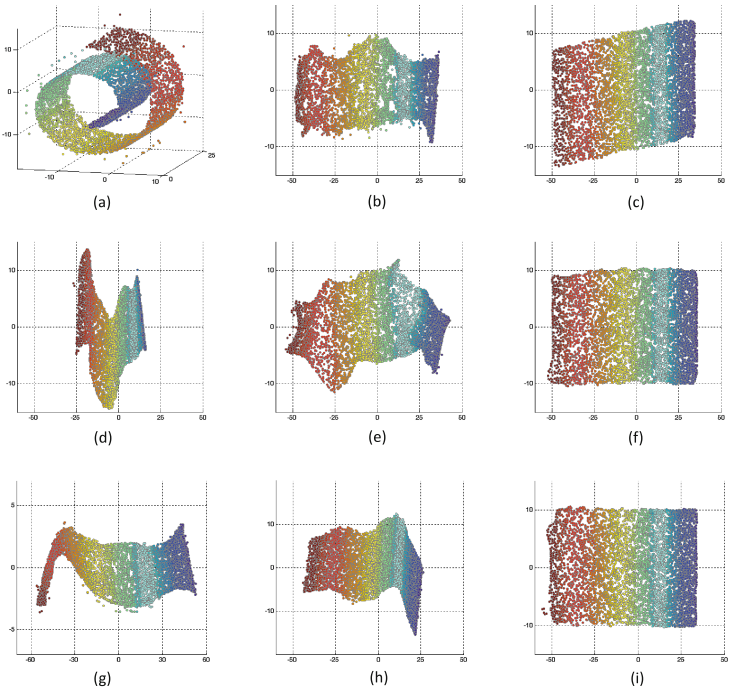

Fig. 2. Embedding of a Swiss roll with noise. Panel (a) represents the Swiss roll composed by 6000 points with 1% of uniform distributed outliers (60 outliers). Panels (b) and (c) describe the projections obtained by GL-MVU (12 Laplacian eigenvectors used) and L-MVU (30 landmarks used). Panels (d) and (e) illustrate the mapping of L-MVU (30 outliers used as landmarks) and S-MVU (600 prototypes composed by the 60 outliers and 540 randomly chosen points). Panel (f) shows the embedding of S-MVU based on 600 central prototypes. Finally, panels (g), (h) and (i) display the unfolding of GL-MVU, L-MVU and S-MVU, respectively, on the Swiss roll data set with 10% of outliers.

term refers to the intra-clusters distances computation. The overall complexity of S-MVU is, therefore, $O\left(m^2 + mKIn + p^3 + c^3\right)$. Major computational advantages are obtained when $p \ll m$, where p can be naturally expressed as a fraction $\gamma \in (0, 1]$ of the available points, $p = \lfloor \gamma m \rfloor$. Experiments on artificial manifold data sets and on medium-size data sets from the UCI Repository [19] empirically showed that γ can be fixed to a very small value ($\gamma \approx 0.1$) to obtain a fast low-dimensional unfolding at the expense of a reduced loss in the quality of the embedding.

4 Experiments and Results

To evaluate the usefulness of Selective MVU we resorted to a variety of criteria based on the concept of loss of quality, which is supposed to be strongly related to the preservation of the data geometry [20]. Most of these criteria can be

divided into local and global approaches which are focused, respectively, on the local neighborhood and the global structure preservation.

Computational tests were performed on eight benchmark data sets mostly available at [19] and given by Page Blocks, Wall-Following Robot Navigation (Wall), Gisette, Isolet, U.S. Postal Service Handwritten Digits (USPS), Human Activity Recognition Using Smartphones (Smartphones), Pen-Based Recognition of Handwritten Digits (Penbased) and EEG Eye State (EEG). Prior to the experiments missing values were removed and data were standardized. These data sets are described in Table 1 in terms of number of points, attributes and dimensionality d^* of the embedding space. This last was estimated by analyzing the difference between consecutive eigenvalues in the eigenspectrum of the landmark Gram matrix in L-MVU, as suggested in [9].

Despite MVU behaves like local methods by preserving distances and angles between neighboring points it can be also regarded as a global technique since it maximizes the overall variance of the embedding. Besides L-MVU and GL-MVU the proposed algorithm was therefore compared to three state-of-the-art local and global approaches. Among local methods we considered Locally linear embedding (LLE) and Local tangent space alignment (LTSA). The former emerged as the most effective manifold learning algorithm for microarray data embedding [21]; the latter received great attention for its simple geometric intuition and straightforward implementation. Among global techniques we focused on Kernel PCA (KPCA) with RBF kernel, which has been related to MVU in a recent taxonomy proposed in [22]. Indeed, both methods are spectral techniques which convert the dimensionality reduction problem into the eigendecomposition of a kernel matrix.

In the following experiments some parameters were fixed. The number of Laplacian eigenvectors in GL-MVU was set to 12 whereas 30 landmarks were used in L-MVU to limit the computing time. The percentage of points taken as prototypes in S-MVU was fixed to 0.1 for all data sets except for EEG, for which it was set to 0.05. The number of clusters in procedure CPS was found through

Table 1. Description of the data sets. The last column indicates the estimated dimensionality of the embedding space.

ID	Data set	Points	Attributes	d^*
1	*Pageblocks*	5406	10	5
2	*Wall*	5456	24	6
3	*Gisette*	7000	5000	10
4	*Isolet*	7797	617	6
5	*USPS*	9298	256	6
6	*Smartphones*	10299	561	5
7	*Penbased*	10992	16	6
8	*EEG*	14980	14	6

a grid search so to minimize the neighborhood size for which the connection of the neighborhood graph was achieved. Finally, the RBF kernel was applied in the KRR model by setting $\lambda = 0.1$ and fixing the RBF parameter to 10^j for a given j in the interval $[-3, -1]$. The same RBF parameter's values were tested for KPCA. All methods were implemented in MATLAB. Computations were run on a 3.40 GHz quad-core processor with 16 GB RAM.

To analyse the local neighborhood preservation we resorted to two local criteria which measure the degree of overlap between the neighboring sets of a point and of its embedding. The first is represented by the Local-Continuity Meta-Criterion (Q_L) [23], which is defined as the average size of the overlap of the neighboring sets. The second is given by Trustworthiness and Continuity (Q_{TC}) [24], which is based on the exchange of indices of neighboring samples in the input and the feature space according to the pairwise Euclidean

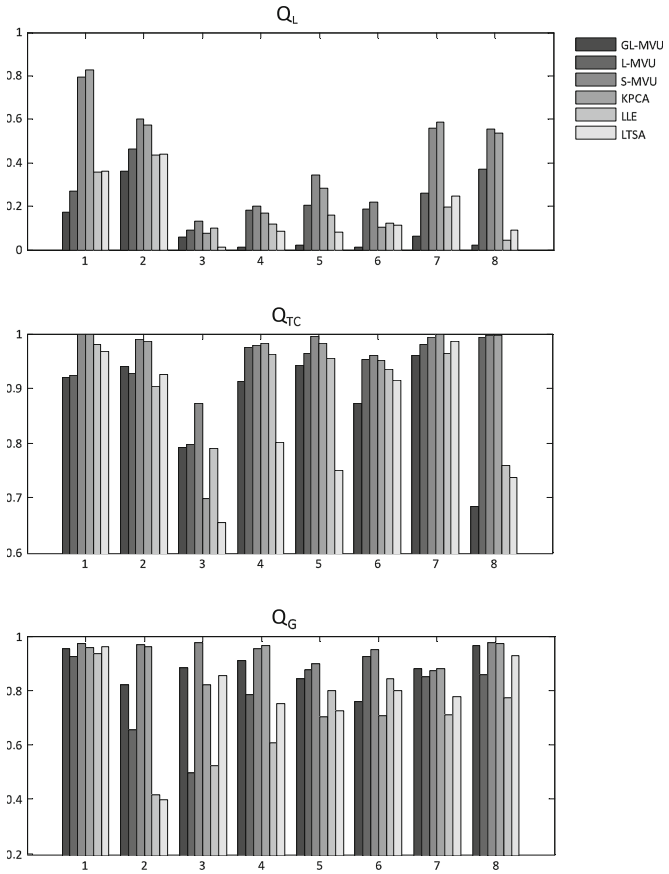

Fig. 3. Local and global quality assessment. Each panel indicates the performance of the competing algorithms on the eight data sets.

distances. Q_{TC} is defined as a linear combination of two measures which evaluate, respectively, the degree of trustworthiness that points farther away enter the neighborhood of a sample in the embedding and the degree of continuity that points originally included in the neighborhood are pushed farther away. The coefficient of the linear combination was here fixed to 0.5. The above criteria have proven to be good estimates of the embedding quality. In particular, the greater their values are in the interval $[0, 1]$, the better is the projection.

The global structure holding performance was, instead, analyzed by means of a global metric recently proposed in [25]. This metric, here denoted as Q_G, evaluates the difference of the transforming scales of the embedding set compared to the original data manifold along various directions. This is achieved by computing a shortest path tree of the neighborhood graph and using the Spearman's rank order correlation coefficient defined on the rankings of the main branches lengths. The original global manifold is well preserved in the data embedding as the value of Q_G approaches 1.

The results obtained by the competing techniques are depicted in Fig. 3, where each panel collects the performances for a given measure. The computing time for data embedding recorded by selecting for each method the minimum number of neighbors generating a connected neighborhood graph (provided $k \geq 4$), and once K has been fixed for S-MVU, is shown in Fig. 4. Since the aforementioned criteria are highly sensitive to the neighborhood size, to perform a fair comparison we computed Q_L, Q_{TC} and Q_G for a fixed value of k that was set equal to the number of neighbors for which the corresponding neighborhood graph turned out to be connected for all methods.

According to the local quality assessment the embedding generated by S-MVU and KPCA more faithfully preserved the local neighborhood structure of

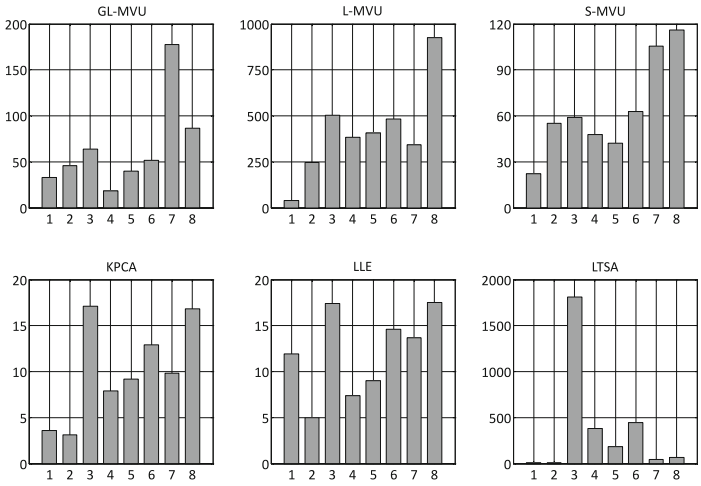

Fig. 4. Computing time (secs) for embedding each of the eight data sets.

the original manifold. In particular, S-MVU dominated both MVU extensions, LLE and LTSA, and provided better results on the majority of the data sets (6 out of 8) compared to KPCA. The proposed algorithm exhibited also notable performances in terms of global structure preservation, as indicated by Q_G. Therefore, the embedding set of S-MVU encountered a smaller distortion of the global shape of the manifold on most data sets. It is worth to notice that, whereas L-MVU and LLE performed generally better than GL-MVU and LTSA according to the local measures, they were often dominated by the latter based on the global metric. The proposed S-MVU algorithm, instead, behaved well both in terms of local and global quality assessment.

5 Conclusions and Future Extensions

In this paper we described a novel method for nonlinear dimensionality reduction indicated as Selective MVU (S-MVU). In the proposed algorithm the unfolding process is guided by a subset of distinguished points called central prototypes, whose embedding is computed by means of classical MVU. The projections of the remaining samples are thereafter reconstructed via multi-output kernel ridge regression. S-MVU was empirically compared to two well-known fast MVU extensions and to three prominent nonlinear dimensionality reduction methods. On several benchmark data sets it achieved noteworthy performances and emerged as a valid alternative to state-of-the-art techniques in terms of quality of the data projection.

The present study can be extended in several directions. First, novel procedures for selecting the representative points or embedding the set of non-prototypes samples could be developed. It would be also worthwhile to investigate the effectiveness of greedy optimization algorithms for solving problem SD in Selective MVU to speed up the unfolding process. Finally, further computational tests could be performed by comparing the accuracy that alternative classification algorithms achieve on the different data projections.

References

1. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science **290**, 2319–2323 (2000)
2. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science **290**, 2323–2326 (2000)
3. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput. **15**, 1373–1396 (2003)
4. Zhang, Z., Zha, H.: Principal manifolds and nonlinear dimension reduction via local tangent space alignment. SIAM J. Sci. Comput. **26**, 313–338 (2004)
5. Weinberger, K.Q., Saul, L.K.: Unsupervised learning of image manifolds by semidefinite programming. In: IEEE International Conference on Computer Vision and Pattern Recognition, pp. 988–995 (2004)
6. Kleiner, A., Rahimi, A., Jordan, M.I.: Random conic pursuit for semidefinite programming. In: Advances in Neural Information Processing Systems, pp. 1135–1143 (2010)

7. Hao, Z., Yuan, G., Ghanem, B.: Bilgo: Bilateral greedy optimization for large scale semidefinite programming. Neurocomputing **127**, 247–257 (2014)
8. Chen, W., Weinberger, K.Q., Chen, Y.: Maximum variance correction with application to A* search. In: Proceedings of the 30th International Conference on Machine Learning, pp. 302–310 (2013)
9. Weinberger, K.Q., Packer, B.D., Saul, L.K.: Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization. In: Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics, pp. 381–388 (2005)
10. Weinberger, K.Q., Sha, F., Zhu, Q., Saul, L.K.: Graph laplacian regularization for large-scale semidefinite programming. In: Advances in Neural Information Processing Systems, vol. 19, p. 1489 (2007)
11. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer, New York (2001)
12. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press, Cambridge (2004)
13. Orsenigo, C., Vercellis, C.: Kernel ridge regression for out-of-sample mapping in supervised manifold learning. Expert Syst. Appl. **39**, 7757–7762 (2012)
14. de Silva, V., Tenenbaum, J.B.: Global versus local methods in nonlinear dimensionality reduction. In: Advances in Neural Information Processing Systems, vol. 15, pp. 705–712 (2003)
15. Cai, D., He, X., Han, J.: Spectral regression for efficient regularized subspace learning. In: IEEE 11th International Conference on Computer Vision, pp. 1–8 (2007)
16. Chen, Y., Crawford, M. M., Ghosh, J.: Improved nonlinear manifold learning for land cover classification via intelligent landmark selection. In: IEEE International Geoscience & Remote Sensing Symposium, pp. 545–548 (2006)
17. Gu, R.J., Xu, W.B.: An improved manifold learning algorithm for data visualization. In: Proceedings of the 2006 International Conference on Machine Learning and Cybernetics, pp. 1170–1173 (2006)
18. Khan, F.: An initial seed selection algorithm for k-means clustering of georeferenced data to improve replicability of cluster assignments for mapping application. Appl. Soft Comput. **11**, 3698–3700 (2012)
19. Bache, K., Lichman, M.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine (2013). http://archive.ics.uci.edu/ml
20. Gracia, A., González, S., Robles, V., Menasalvas, E.: A methodology to compare dimensionality reduction algorithms in terms of loss of quality. Inf. Sci. **270**, 1–27 (2014)
21. Orsenigo, C., Vercellis, C.: A comparative study of nonlinear manifold learning methods for cancer microarray data classification. Expert Syst. Appl. **40**, 2189–2197 (2013)
22. van der Maaten, L., Postma, E., van den Herik, H.: Dimensionality reduction: A comparative review (2007)
23. Chen, L., Buja, A.: Local multidimensional scaling for nonlinear dimension reduction, and proximity analysis. J. Am. Stat. Assoc. **104**, 209–219 (2009)
24. Venna, J., Kaski, S.: Local multidimensional scaling. Neural Networks **19**, 889–899 (2006)
25. Meng, D., Leung, Y., Xu, Z.: A new quality assessment criterion for nonlinear dimensionality reduction. Neurocomputing **74**, 941–94 (2011)