A Review of Non-Linear Dimensionality Reduction Methods and Their Applications in Medical Data Analysis

Christopher Erwin, San Francisco State University

17 December 2016

Abstract. A core group of kernel-based non-linear dimensionality reduction algorithms are reviewed. Applications of these algorithms to medical image analysis and genetic data analysis comprises the remainder of the report, along with a brief discussion of the future of the subject.

1. Introduction

A major challenge of working with complex data sets such as images or genome sequences is the so-called “curse of dimensionality”. This issue occurs when a data set contains a number of dimensions or independent variables which greatly exceeds the number of available observations. Dimensionality reduction offers a solution to this problem, by extracting only the most descriptive dimensions and thus reducing the number of dimensions required to adequately describe the data. This review covers a subset of such algorithms which utilize kernel methods to extract dimensions which may be non-linear, and may not necessarily exist in the original space..

High-dimensional data sets which benefit from dimensionality reduction can be found in many areas of medical data analysis. Image analysis in other areas of computer vision has dealt with this issue for years, and similar techniques which are used in other industries can be applied to medical imagery. For example, MRI or CT scans may have thousands of dimensions representing pixel intensities in the resulting images. Issues such as segmentation of cancerous

tumors, organs of some medical interest, or other structures may be more efficiently and accurately extracted from these images by first applying non-linear dimensionality reduction. We will see later how this occurs by using NLDR algorithms as a pre-processing step before applying traditional segmentation methods, or by more sophisticated techniques which incorporate a concept of non-linearity directly.

Another area of medicine within which the curse of dimensionality may be defeated by the discovery of non-linear low-dimensional embeddings is gene and protein expression profiling. These studies often aim to characterize the genetic and biochemical composition of cancers and other disease indicators. Due to the high complexity of human biology, these studies often contain "several thousand dimensional gene and protein profiles...[, yet] most protein and gene expression databases contain no more than a few thousand patient samples." [1]. We will explore examples in which linear clustering and classification is merely sufficient, but non-linear methods allow for much more informative visualizations which reveal novel classes otherwise hidden in the data.

In two or three dimensions, identifying classes and manifolds is simple to the human eye. For example, we can look at the data set below and see the highly non-linear spiral structure in the 3D space (Fig. 1). However, to project that spiral into 2D space with the constraint of maintaining linear separability of the four classes (colors) of data is a problem which can only be solved using non-linear dimensionality reduction algorithms. This data set is used in most of the algorithms in Section 2 to demonstrate their ability to visualize data points in a manifold space on a non-linear surface, like the surface of the spiral, rather than linear axes already present in the input space.

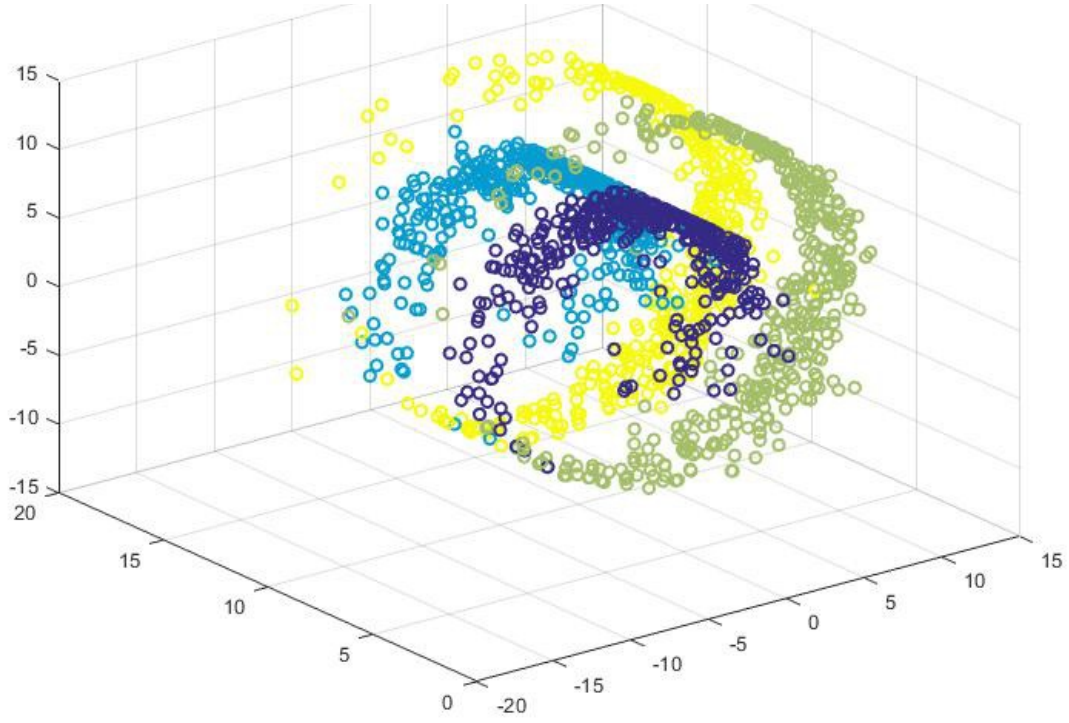

Fig. 1 -- Spiral data set in 3 dimensional space.

The problems most commonly found in medical literature involve segmentation (which can be described as clustering or classifying pixels), and classification. We will also explore the problem of identifying new clusters or classes hidden in high-dimensional data through the context of gene and protein expression data sets.

Clustering is a critical problem in medical data analysis, and a significant use case of dimensionality reduction. In a review of clustering algorithms in biomedical research, one property identified as one of "the major challenges in clustering that must be addressed by the next generation of clustering technologies" [2] is the capability to tackle the problem of high dimensionality. This property is further noted to be "particularly urgent in gene expression data analysis and image analysis. The identification of relevant features or the capture of the intrinsic dimension is important for describing the real data structure." [2]. Non-linear dimensionality

reduction methods often out-perform comparable linear methods, helping to solve these important issues and leading to more accurate analyses on high dimensional data.

There are many forms of both linear and non-linear dimensionality reduction algorithms. Most are based on the spectral decomposition of a similarity matrix which describes the distance between data points. This can be a literal distance matrix, composed of the euclidean distance between data points in the input space, or it can be composed of distances in some constructed feature space. This distinction between decomposition of a similarity matrix in input space and a matrix of distances in feature space forms the key difference between the algorithms described here. All of the methods described serve as a sort of preprocessing step, and clustering or classification occurs on the low-dimensional output space (hence the efficiency improvements).

There are of course other methods of dimensionality reduction, but in order to minimize scope, those treated here are a limited class based on spectral decomposition of a similarity matrix constructed using a special "kernel" function. This function describes the inner product in a high-dimensional feature space. By using a kernel function to represent inner products, the data points do not need to be explicitly mapped to the feature space, even though the required information concerning the distance between points in the space is obtainable.

The remainder of this report will continue as follows. Section 2 describes a selection of non-linear dimensionality reduction methods based on spectral decomposition of a similarity matrix. Section 3 investigates applications in genetics. Section 4 explores applications in image processing with an emphasis on MRI analysis. Section 5 discusses and critiques the methods described, and makes suggestions concerning the future of the field substantiated by published research. Section 6 concludes the review.

2. Algorithms for Non-Linear Dimensionality Reduction

2.1 Kernel PCA

Kernel PCA is a non-linear extension of Principal Component Analysis. To understand KPCA, let us first review traditional linear PCA. PCA seeks to diagonalize the covariance matrix of a set of centered input data points. This can be accomplished by solving the eigenvalue equation for the covariance matrix. We then select the resulting eigenvectors corresponding to the *n* largest eigenvalues. These eigenvectors form an orthogonal basis for the *n* axes which explain the most variance in input space.

The downside to linear PCA is the guarantee that these new dimensions are some combination of axes in the original input space because PCA is merely a basis transformation. The number of dimensions can indeed be reduced, and the new dimensions are rotated to agree with the axes of maximum explained variance. However, this paradigm cannot support the extraction of non-linear subspaces which may be far more informative.

Kernel PCA leverages a technique called the kernel trick to effectively construct a distance matrix in some unknown, very high dimensional feature space. The major difference between linear and non-linear PCA is the substitution of a kernel function (chosen a priori) for all dot products used in the algorithm. We use the output of the kernel function to represent a dot product in the very high dimensional feature space:

$$k(x, y) = (\Phi(x) \cdot \Phi(y)) \tag{1}$$

This allows us "to compute the value of the dot product [distance] in F without having to carry out the map Φ," [3] which is much more computationally expensive.

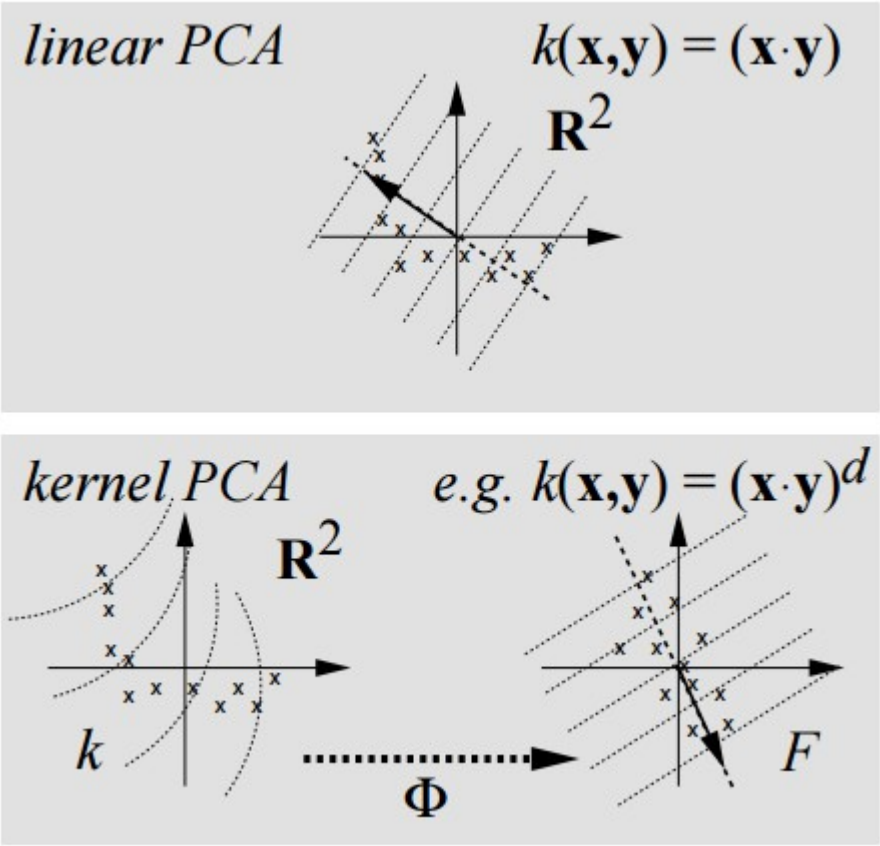

Fig. 2: Comparison of Linear and Kernel PCA. Note that even when maintaining the same number of dimensions, Kernel PCA is capable of extracting a new dimension which optimally describes the data. This dimension may not exist in the original input space. Image from [3].

The assertion that equation (1) is accurate for our purposes has been substantiated by other research, notably "Mercer's theorem of functional analysis [which] implies that if k is a continuous kernel of a positive integral operator, there exists a mapping into a space where k acts as a dot product". Furthermore, "The choice of *k* then *implicitly* determines the mapping Φ and the feature space F." [3]. These facts form much of the groundwork of the algorithms we explore in section 2.

The original paper noted that during experimentation "nonlinear principal components afforded better recognition rates," [3] however, the 1998 paper only discussed its results in

comparison to the 1989 LeNet1 and a handful of other nonlinearizations of PCA. It should be noted that linear Support Vector Machines were also amenable to the same kernel-based non-linear extension as linear PCA at that time. In fact, the paper notes that this fact was known from personal correspondence, but the technique was not widely adopted by the machine learning community. This makes Kernel PCA a significant breakthrough in the acceptance of non-linear techniques for classification.

2.2 Isomap

The Isometric Feature Mapping algorithm (Isomap) is a similar non-linear extension of metric Multidimensional Scaling (MDS). This algorithm has a number of similarities to Kernel PCA (which are explored later in this section), but on its surface this procedure is quite different in its implementation.

Isomap is based on the estimation of geodesic distances over a manifold embedded in a higher dimensional space by creating a graph of either the K nearest neighbors of each input point (called K-Isomap), or the points within some distance ε (ε-Isomap). Then the shortest paths between all points in this graph are computed. Dimensionality reduction is achieved when classical MDS is run on this graph distance matrix, "constructing an embedding of the data in a d-dimensional Euclidean space Y that best preserves the manifold's estimated intrinsic geometry" [4]. If MDS was run on the regular input data, normal straight-line distances would appear more accurate than the desired distance on the manifold. As seen in Figure 3, the restriction of MDS to only the distances between points near each other in some local neighborhood greatly improves the outcome. However, the K or ε parameter must be chosen carefully to avoid short-circuiting when the desired manifold has sections separated by a narrow margin. This may be true of any

similar algorithm dependent upon local distances.

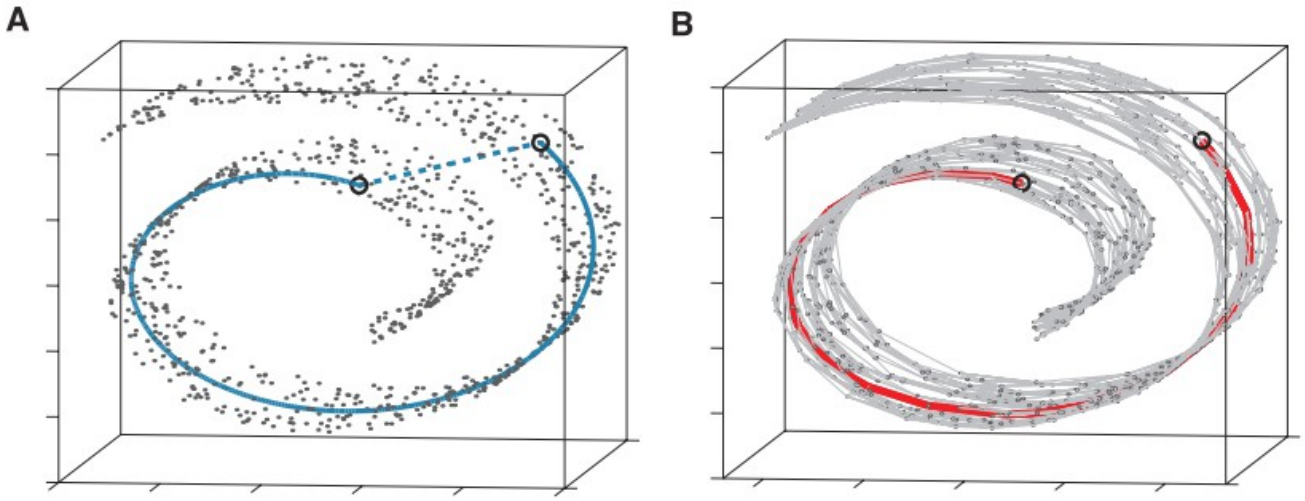

Fig. 3: A comparison between straight-line and geodesic distance along the spiral. From [4].

Due to this graph construction approach, Isomap will converge and find the true low-dimensional structure of non-linear manifolds. "These guarantees of asymptotic convergence rest on a proof that as the number of data points increases, the graph distances...provide increasingly better approximations to the intrinsic geodesic distances..." [4]. As a result, Isomap is particularly suited to applications in which highly complex data is well-structured, but does not have a well-defined manifold geometry, such as OCR and gesture recognition.

2.3 Locally Linear Embedding

Locally Linear Embedding places its emphasis on the preservation of the neighborhood of each data point such that each neighborhood appears linear on the manifold of interest. This approach "eliminates the need to estimate pairwise distances between widely separated data points..." enabling the algorithm to tend to "accumulate very sparse matrices, whose structure can be exploited for savings in time and space." [5]. This construction of sparse matrices makes LLE more versatile and efficient than the above algorithms when properly implemented.

The algorithm initially looks similar to Isomap. It first uses K-nearest neighbors to create a neighborhood for each data point. Then it diverges by reconstructing each point using a set of weights to form a linear combination with each of the surrounding data points. This step is where the "locally linear" guarantee of the algorithm comes into play. The low-dimensional embedding best reconstructed by the weights is defined by the eigenvectors corresponding to the smallest non-zero eigenvalues of the weight matrix. The points in the new space will have a different set of weights with the same locally linear constraint as in the input space, forming a new projection of the points in a lower dimensional space.

Due to the definition of the local neighborhood being to critical to the operation of LLE, the selection of a K variable is critical. The paper notes that this is an acceptable requirement, considering the advantages of LLE compared to other methods.

> "Many popular learning algorithms for nonlinear dimensionality reduction do not share the favorable properties of LLE. Iterative hill-climbing methods for autoencoder neural networks... self-organizing maps... and latent variable models... do not have the same guarantees of global optimality or convergence; they also tend to involve many more free parameters, such as learning rates, convergence criteria, and architectural specifications." [5]

While this supremacy is true for the complex methods listed above, other algorithms such as Isomap and KPCA are barely mentioned. A paragraph does state the differences between the Isomap and LLE approaches before claiming LLE can be optimized to handle its sparse matrices with higher efficiency than Isomap can handle its graph distances.

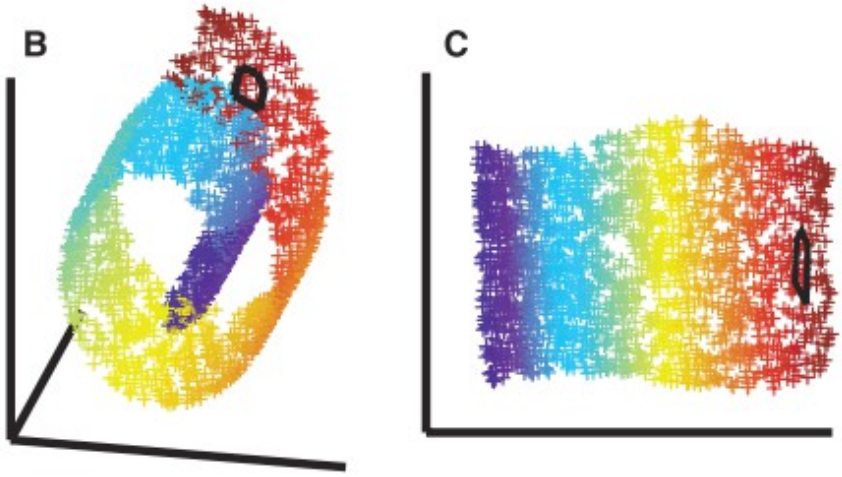

Fig. 4: Demonstration of LLE with a neighborhood circled. From [5].

We can see in Figure 4 a projection of the spiral manifold onto a 2D space, which is similar to the Isomap output in Figure 3. The competing algorithms' routes to this projection are somewhat different due to their different goals when constructing a similarity matrix. Isomap's goal is to preserve geodesic distances between data points, and LLE focuses on "analyzing local symmetries, linear coefficients, and reconstruction errors" [5]. The differences between these algorithms will be explored in more detail in section 5.

2.4 Maximum Variance Unfolding

Maximum Variance Unfolding (MVU) shares traits of both Isomap and LLE. Like Isomap, MVU is a non-linear extension of MDS, and begins by constructing a graph from the K-nearest neighbors of each data point in input space. However, the method in which this graph is processed to extract the dimensions of maximum variance differs drastically from Isomap. Isomap directly performs MDS on the matrix of shortest path distances between graph nodes. MVU uses an iterative approach, treating the discovery of a distance matrix that maximizes the distance between points as a semidefinite programming problem before applying MDS to this matrix.

By preserving the local neighborhoods of each point, MVU parallels LLE by guaranteeing the local linearity of each region of the manifold. It alters the points' positions while maintaining the distances between points, creating a low-dimensional projection which preserves local neighborhoods in a fashion comparable to the output of LLE. We must note that "neighborhoods of inputs and outputs will be related by translation and rotation if and only if all the distances and angles between points and their neighborhoods are preserved." [6]. The lack of a sparse distance matrix like that computed in LLE makes MVU computationally very different to LLE in terms of time efficiency. In fact, MVU is similar to Isomap in the time and space required to construct, process, and perform MDS on a graph.

The algorithm begins by constructing a graph of the K-nearest neighbors of each data point. The graph is then modified to maximize the sum of squared distances between pairs of output points:

$$\Phi = \frac{1}{2\mathrm{n}} \sum_{ij} \| y_i - y_j \|^2 \tag{2}$$

This equation can be maximized to "pull the outputs as far apart as possible, *subject to the constraints*" previously noted concerning local neighborhood geometry [6]. This optimization can be expressed as a matrix K representing dot products (distances) in the input space, much like the above algorithms.

The local isometry constraint forces the addition of an extra step compared to the aforementioned algorithms. The problem is construed as optimization of the trace of the distance matrix, rather than simply selecting eigenvectors of the distance matrix. The effect of this optimization is immediately clear: the inner products between points in the output space are maximized while maintaining the geometry (angles and distances) of the neighborhoods. This

maximization "is an instance of semidefinite programming...[whose] objective function is linear in the matrix elements." [6]. It is from this resulting matrix that the top eigenvectors are selected, yielding the desired low-dimensional embedding.

The following Figure 4 visualizes several iterations of the algorithm with varying levels of "normalized variance" of data points. When this value reaches the maximum value of 1, the reorganization of points is complete and spectral decomposition can be used to isolate the desired dimensions.

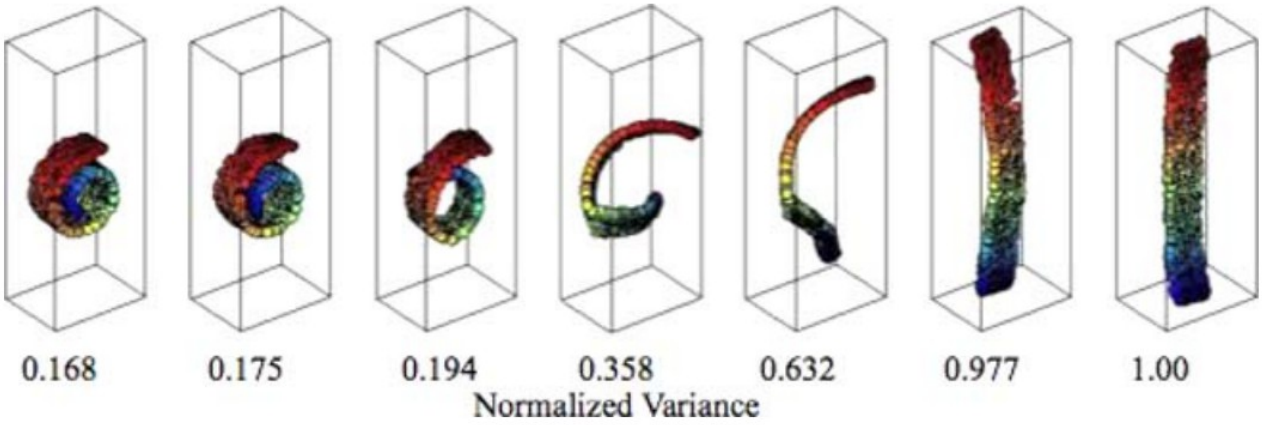

Fig. 4: MVU through a varying number of iterations, resulting in different levels of normalized variance. From [6].

2.5 Laplacian Eigenmaps

The Laplacian Eigenmap algorithm (LEM) again leverages properties of a graph constructed from the input data to map the data to a low-dimensional manifold. Like Isomap and MVU, the procedure begins by building a weighted graph using either the K-nearest neighbors or a threshold distance ε. This algorithm begins to differ from the others reviewed so far in the manner in which the graph's edges are weighted. LEM supports two options: the heat kernel:

$$W_{ij} = e^{\frac{-\|x_i - x_j\|^2}{t}} \tag{3}$$

where t is a real-valued parameter for connected nodes, or W = 0 for nodes not connected. Or, a simplified version where the edge weights are equal to 1 if nodes are connected and 0 otherwise. This is equivalent to setting the t parameter to infinity, and this removes the necessity of selecting an appropriate t value. The results of LEM using different values of t (including infinity) can be seen in figure 5 below.

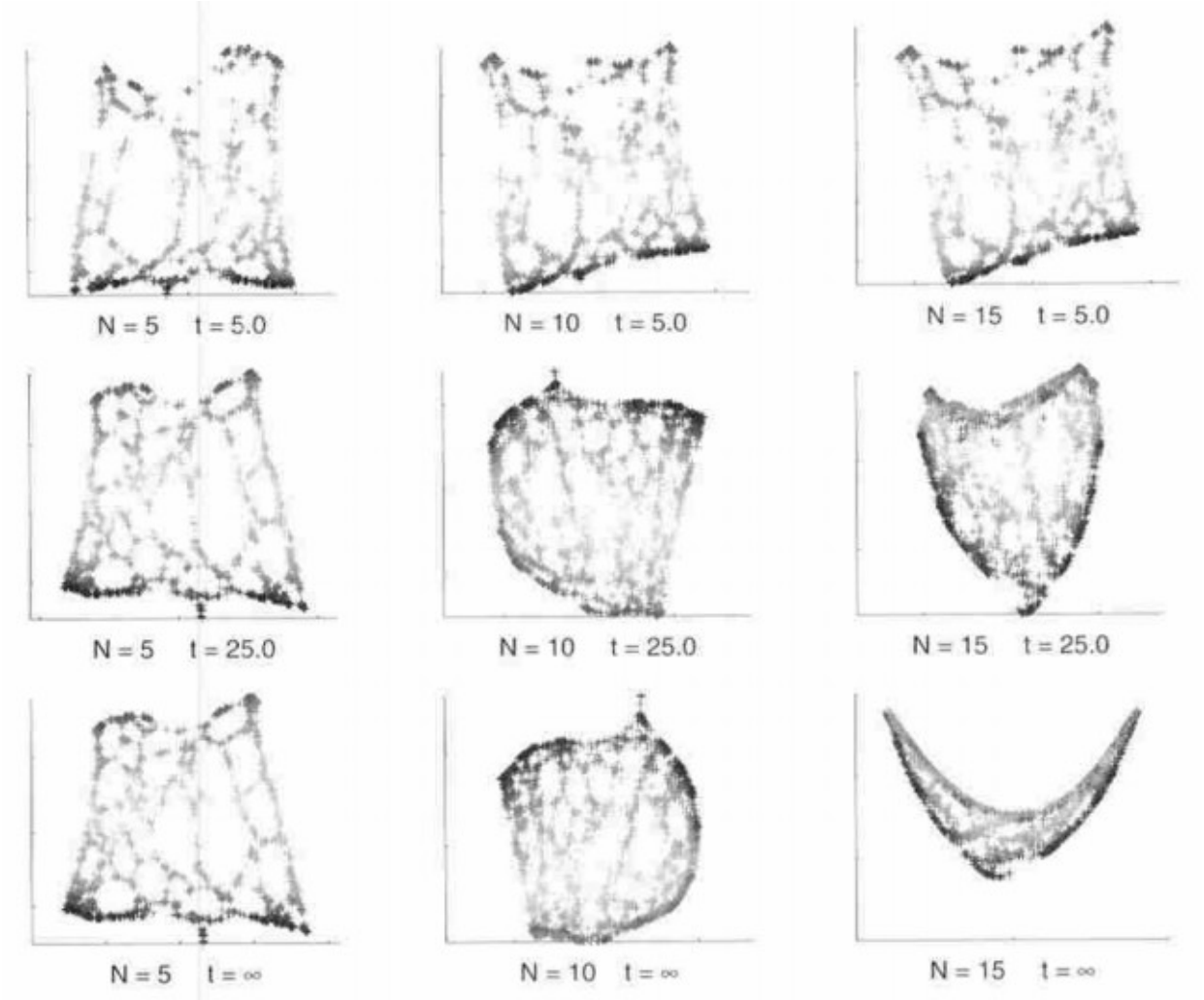

Fig. 5: Output of LEM given the swiss roll data set as input, and varying values of N and t. From [7].

We then solve the generalized eigenvector problem:

$$Lf = \lambda D f \tag{4}$$

The weight matrix W is summed along its columns or rows to form the diagonal matrix D in the above equation. L is the laplacian matrix L = D - W. The eigenvectors corresponding to the smallest non-zero eigenvalues represent the embedded subspace in the same way as the other algorithms utilized the decomposition.

The use of the graph laplacian that makes LEM even more interesting. This approach to dimensionality reduction "may also be interpreted in the framework of clustering and has very close ties to spectrally based clustering techniques such as those used for image segmentation..." [7]. This concept will be explored in more detail in section 5, but the use of local dimensionality reduction methods yield the same results as common spectral clustering algorithms.

3. Genetics Applications

One area of medical data analysis ripe for improvement via the application of non-linear dimensionality reduction is the analysis of gene and protein expression data sets. These studies often contain many thousands of dimensions (the individual gene or protein indicators), but very few data points (corresponding to individuals participating in the study). This situation, a massive number of dimensions and few data points to work with, is practically the definition of the Curse of Dimensionality. Classification and analysis in this high dimensional input space is very difficult, and many studies have found "classifier performance with linear DR schemes for biomedical data [to be] a mixed bag." [1]. To better understand how non-linear techniques stack up when compared to traditional linear methods, this section will explore the results of one particular survey and touch on the results of some publications which support the survey's conclusion of better performance when non-linear dimensionality reduction algorithms are used.

The goal of this particular investigation into the "Efficacy of Nonlinear Dimensionality

Reduction Schemes in Classifying Gene and Protein Expression Studies" was to "systematically and quantitatively compare and evaluate the performance of" PCA, LDA, and linear MDS against Isomap, LLE, and LEM [1] (all of which were described in Section 2). This comparison was made on the basis of a variety of measures indicating cluster quality, including intercentroid distance and cluster tightness, as well as the performance of a linear classifier (SVM) on the low-dimensional data. The performance of the algorithms reviewed in section 2 continued to exceed the linear DR algorithms for aiding classification accuracy on a low-dimensional manifold.

However, there are some cases in which linear methods successfully perform dimensionality reduction, but fail to adequately visualize clusters of data present in the input space. For example, Figure 6 below compares the performance of LDA, LLE, and LEM on a data set of lung cancer patients (and those without lung cancer).

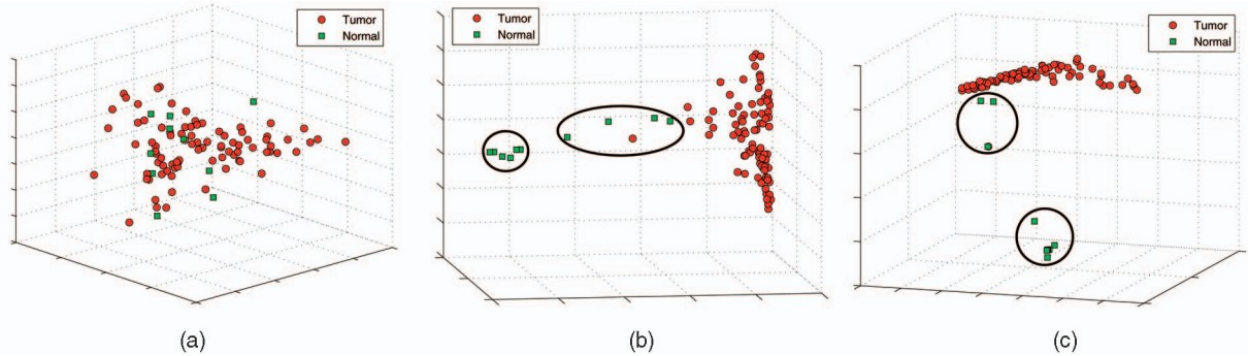

Fig. 6: Lung cancer data set in 3 dimensions via a) LDA, b) LLE, and c) LEM. Note the distinct classes of non-tumor data points which appear in the non-linear embeddings (circled). From [1].

In the linear LDA example, the resulting data are difficult to work with for either task. However, the non-linear methods not only reveal useful classification boundaries, they also reveal two distinct classes of non-tumor data points. This phenomenon is repeated in Figure 7. Even though metric MDS appears to provide a reasonable classifier, the clusters (representing novel classes of

prostate cancer) are not readily apparent despite existing within the high-dimensional data.

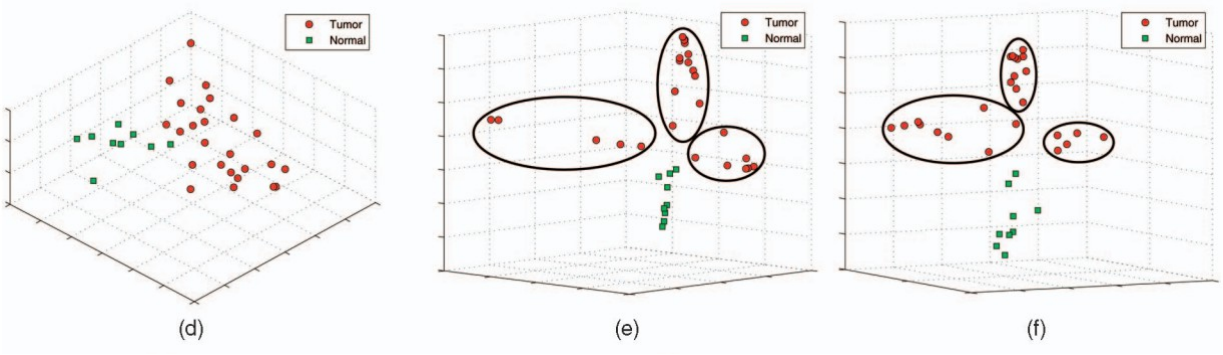

Fig. 7: Prostate cancer data set in 3 dimensions via d) MDS, e) LLE, and f) LEM. Note the distinct classes of cancer present in the non-linear projections despite the reasonable boundary found in MDS. From [1].

Furthermore, a separate study of microarray data sets [8] "found that there were biologically significant elements of the gene expression profile that were not seen with linear MDS." [1]. The survey cites other research which also "...found that LDA gave poor results in distinguishing disease classes on a cohort of nine gene expression studies...found limited use of LDA and PCA for classifying gene and protein expression profiles of a diffuse large B-cell lymphoma data set since the classes appeared to be linearly inseparable..." and so on. The large amount of research finding that non-linear methods consistently out-perform linear methods seems to suggest that biomedical data inherently contains informative non-linear structures [1].

4. Image Analysis

Non-linear dimensionality reduction and closely related non-linear classification methods can also be applied to problems in image segmentation and the identification of new classes of images. Several studies have taken advantage of non-linear methods such as LLE, graph embedding, and spectral clustering to improve both supervised and unsupervised classification of MRI and MRS data to aid in the detection of cancerous tissue and segmentation of organs. Here

we will briefly review one application of LLE as a precursor to MRI segmentation, and only touch upon the related methods.

There are many ways of performing image segmentation, and most simply consider the image as a whole. The restriction of segmentation algorithms to a lower dimensional manifold can improve performance compared to their operation within the original image space. For example, Zhang and Pless took on the task of segmenting the left ventricle from cardiopulmonary MR images. Figure 8 below compares the original segmentation by level set evolution in the input space (top row) and the result of segmentation on data in the manifold space (bottom row).

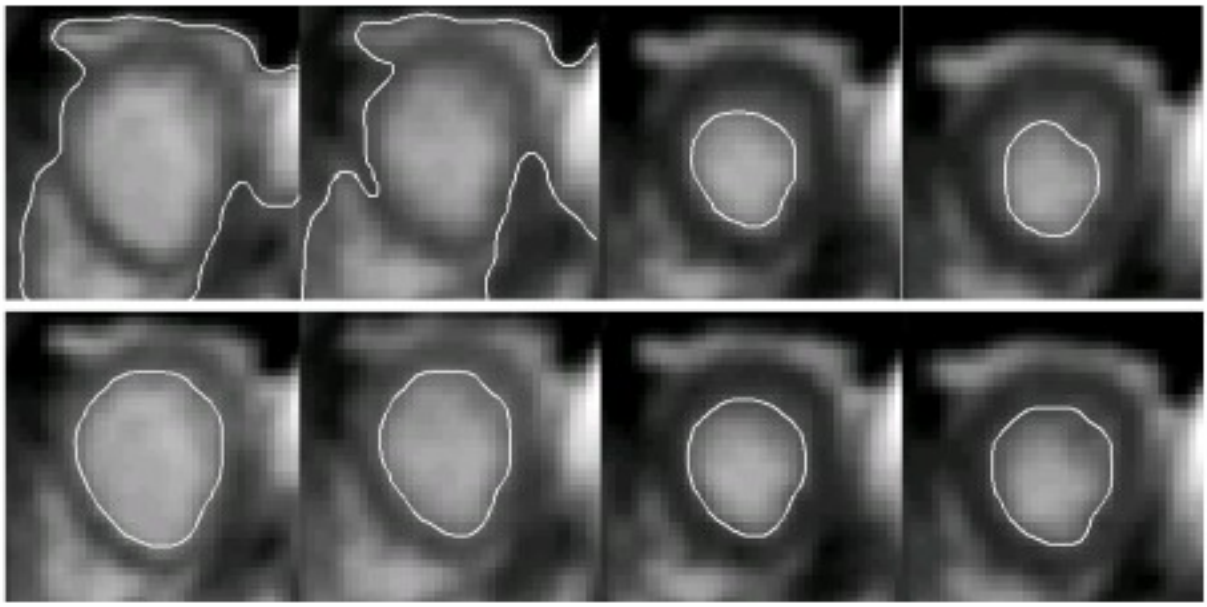

Fig. 8: Left ventricle segmentation before (top) and after (bottom) taking into consideration the manifold extracted by non-linear dimensionality reduction. From [9].

The addition of LLE to the image processing pipeline proved effective in increasing the system's segmentation accuracy. The "additional constraints that the manifold imposes on the level set evolution allows segmentation of the left ventricle in images that are too low contrast to support

single image segmentation." [9]. The applicability of restricting traditional image processing algorithms to low-dimensional manifolds can also be seen in other studies.

Another team [10] recently "demonstrated the use of graph embedding to detect the presence of new tissue classes on high-dimensional prostate MRI studies." [1]. In this study, the accuracy of prostate cancer detection from MRI was improved, as well as the discovery of new class information which may lead to an improved understanding of prostate cancer.

An additional study [11] utilized spectral clustering in "distinguishing between cancerous and benign magnetic resonance spectra (MRS) in the prostate and in discriminating between different cancer grades on digitized tissue histopathology." [1]. While spectral clustering is not technically a non-linear dimensionality reduction algorithm, it does perform dimensionality reduction as a preparatory step before clustering. Also, the clustering method utilizes the eigenvalues of a similarity matrix, which is similar to the algorithms referred to in this review. The algorithm can be recast as a special case of non-linear kernel K-Means clustering, and can be improved by the prior application of LLE or other non-linear dimensionality reduction algorithms as seen in this section.

5. Discussion

The algorithms reviewed in section 2 are closely related. All of the algorithms are concerned with the preservation of some property of the local neighborhood, and they can all be reformulated as a variation of Kernel PCA with some appropriately constructed kernel matrix. Furthermore, while the derivation is outside the scope of this review, the results of such locally-oriented non-linear dimensionality reduction methods like LLE and LEM yield the same results as spectral clustering algorithms [7].

All of the articles referenced so far competently describe the operation of the algorithm in question, but rarely does an article include more than a paragraph comparing the algorithm to others in the same class. Granted this sort of comparison is not the goal of the reviewed papers, but it would be beneficial to see such an analysis.

A fair common ground would seem to be Kernel PCA, whose concepts form a basis for the other algorithms in Section 2. Each algorithm is based on the spectral decomposition of a similarity matrix formed by computing some property of each data point's local neighborhood. In Kernel PCA, this similarity matrix is some kernel function chosen a priori. The later algorithms can be construed as Kernel PCA by selecting the kernel matrix to agree with the philosophy of the algorithm in question.

First off, Isomap can be seen as a variant of Kernel PCA on a matrix of graph distances. While Isomap is a non-linear extension of metric MDS, it should be noted that MDS can itself be interpreted as Kernel PCA and vice versa, enabling this analogy [12]. The algorithm first constructs a graph of the K-nearest neighbors or neighbors within some distance ε. Instead of euclidean distances between points, the distance matrix is composed of the shortest paths along the graph from point to point. The centered matrix of squared distances forms the kernel for KPCA [13].

Locally Linear Embedding requires a slightly more complex kernel matrix. We first define a matrix $M = (I - W^T)(I - W)$ [13]. The Kernel matrix is then defined as $K = (\lambda_{max} I - M)$ [13]. Although LLE ordinarily uses the eigenvectors corresponding to the smallest non-zero eigenvalues of the matrix of reconstruction weights W, the Kernel PCA variant uses the largest eigenvectors, as PCA does. The matrix does not necessarily have a corresponding

numerically computable kernel function or feature map.

Interestingly, LLE can also be interpreted in a manner very close to LEM. It is possible to construct "an alternative kernel for LLE that is analagous to the heat kernel for the graph Laplacian..." defined as $K_t = e^{-Mt}$ where t is a real-valued parameter [13]. This kernel can then be used in kernel PCA to yield the same results, but the use of this matrix allows for the interpretation of LLE as a graph operator, similar to LEM.

Naturally, the Laplacian Eigenmap can also be formulated as a special case of Kernel PCA. LEM also constructs a graph similar to Isomap, but defines its connection weights, and therefore the algorithm's representation of a data point's neighborhood, using the heat kernel seen in equation 3. Equation 4 then calls for spectral decomposition of the diagonal matrix constructed by summing the weight matrix over its rows or columns.

We can instead use the pseudoinverse of the matrix L, related to the graph laplacian and seen in equation 4, as the kernel matrix in KPCA. This matrix is already centered. The downside to this interpretation is that it is practically more work to construe LEM as KPCA than it is to simply perform LEM. However, there is an alternative construction which leverages the heat kernel $K_t = e^{-Mt}$ which performs a similar duty. The derivation of this can be found in [13], but is outside the scope of this review.

Lastly, MVU can be described as a non-linear extension of MDS and PCA. The algorithm begins in a familiar way, by constructing a graph of the K-nearest neighbors of each data point. The sum of squared distances (equation 2) between each point in the input space is then used to construct the distance matrix K. Normally, after maximization of the trace of this matrix via semidefinite programming, MDS is applied to find the manifold of interest. However, the matrix

can also be used as input to kernel PCA. As such, the "inner product matrix K computed by maximum variance unfolding can be viewed as a kernel matrix between inputs" [6]. This conversion to kernel PCA is more straight-forward.

The goal of performing non-linear dimensionality reduction is two-fold. Firstly, this class of algorithms is clearly capable of reducing the number of independent dimensions when converting from the input space to the output space, thereby reducing the complexity of the data with minimal loss of information. As a side-effect, these methods often enable traditional linear classifiers to operate effectively when considering non-linear data. This would have been impossible in the original input space.

However, if non-linear classification is the only goal, there is another way. The Kernel Trick may be applied to linear classifiers such as SVM directly. The construction of such non-linear classifiers carries out the same task without the overhead of the assumptions made in order to facilitate dimensionality reduction [14].

The path forward, whether it involves classification, segmentation, or dimensionality reduction for data visualization, will only bring larger and more complex data sets. Current state of the art non-linear dimensionality reduction algorithms perform well even on large and high-dimensional data. To make the situation appear even better, the selection of large numbers of parameters like learning rates is rarely a big concern. In other areas of machine learning parameter selection can be a time-consuming issue.

However, most of the algorithms described in this review do rely upon an indicator for how neighborhoods should be constructed. As the k or ε value gets higher, the algorithm loses its effectiveness and eventually becomes indistinguishable from linear algorithms. As the value

decreases to zero, the algorithm also becomes completely ineffective. The user of the algorithm must still tailor the neighborhood measure to the specific problem at hand and continue running tests until an appropriate value is found. Only a rough estimate can be made without eventually inspecting the algorithm's output for validity.

Computational systems which seek to bridge the gap between dimensionality reduction algorithms, clustering and segmentation algorithms, and domain experts who need to solve problems using these tools, are slowly developing [15]. It is tools such as this that will bring non-linear dimensionality reduction and other machine learning algorithms into common use. As ease of use increases, domain experts without in-depth education in machine learning will be able to utilize these new advancements in data analysis to solve real problems.

6. Conclusion

Non-linear dimensionality reduction methods are interesting and useful extensions of the linear methods for data analysis. The algorithms discussed in this review are only a small subset of those available. This review explored a core segment of methods grounded in spectral decomposition, and they are all closely related to principal component analysis.

Data sets in the medical field are often highly complex and contain non-linear manifolds which are not readily apparent. The application of non-linear dimensionality reduction algorithms can not only help alleviate the curse of dimensionality, but can also reveal new knowledge which would otherwise be hidden in the non-linearity of the data. Non-linear dimensionality reduction algorithms have real and important significance in medical and other areas of scientific data analysis.

7. References

[1] G. Lee, C. Rodriguez, and A. Madabhushi, "Investigating the Efficacy of Nonlinear Dimensionality Reduction Schemes in Classifying Gene and Protein Expression Studies," *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 5, no. 3, pp. 368–384, Jul. 2008.

[2] Rui Xu and D. C. Wunsch, "Clustering Algorithms in Biomedical Research: A Review," *IEEE Rev. Biomed. Eng.*, vol. 3, pp. 120–154, 2010.

[3] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.

[4] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[5] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[6] K. Q. Weinberger and L. K. Saul, "Unsupervised Learning of Image Manifolds by Semidefinite Programming," *Int. J. Comput. Vis.*, vol. 70, no. 1, pp. 77–90, Oct. 2006.

[7] M. Belkin and P. Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.

[8] K. Dawson, R. L. Rodriguez, and W. Malyj, "Sample phenotype clusters in high-density oligonucleotide microarray data sets are revealed using Isomap, a nonlinear algorithm," *BMC Bioinformatics*, vol. 6, no. 1, p. 1, 2005.

[9] Q. Zhang and R. Pless, "Segmenting cardiopulmonary images using manifold learning with level sets," in *International Workshop on Computer Vision for Biomedical Image Applications*, 2005, pp. 479–488.

[10] A. Madabhushi, J. Shi, M. Rosen, J. E. Tomaszeweski, and M. D. Feldman, "Graph Embedding to Improve Supervised Classification and Novel Class Detection: Application to Prostate Cancer," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2005*, vol. 3749, J. S. Duncan and G. Gerig, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 729–737.

[11] P. Tiwari, A. Madabhushi, and M. Rosen, "A Hierarchical Unsupervised Spectral Clustering Scheme for Detection of Prostate Cancer from Magnetic Resonance Spectroscopy (MRS)," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2007*, vol. 4792, N. Ayache, S. Ourselin, and A. Maeder, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 278–286.

[12] C. K. Williams, "On a connection between kernel PCA and metric multidimensional scaling," *Mach. Learn.*, vol. 46, no. 1–3, pp. 11–19, 2002.

[13] J. Ham, D. D. Lee, S. Mika, and B. Schölkopf, "A Kernel View of the Dimensionality Reduction of Manifolds," in *Proceedings of the Twenty-first International Conference on Machine Learning*, New York, NY, USA, 2004, p. 47–.

[14] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[15] T. Schultz and G. L. Kindlmann, "Open-box spectral clustering: applications to medical image analysis," *IEEE Trans. Vis. Comput. Graph.*, vol. 19, no. 12, pp. 2100–2108, 2013.