УДК 004.8
DOI: 10.20535/SRIT.2308-8893.2018.2.04

BIG DATA ANALYSIS VIA MODEL REDUCTION METHODS

STANISLAV ZABIELIN

Abstract. The enormous growth in the size of data has been observed in recent years being a key factor of the Big Data scenario. Big Data require a new high-performance processing. The use of big data preprocessing methods for data mining in big data is reviewed in this paper. The definition, attributes and categorization of data preprocessing approaches in big data are introduced. The relation between big data and data preprocessing throughout all families of methods and advanced data technologies are likewise analyzed. Furthermore, research challenges are discussed, while concentrating on improvements in certain families of data preprocessing methods and applications based on new big data learning paradigms.

Keywords: nonlinear mapping, dimension reduction, big data, modelling, non-linear dynamic objects, dimensional reduction, diffusion maps, kernel method of main components

INTRODUCTION

Vast amounts of raw data are encompassing us in our world, information that can't be directly treated by humans or manual applications. Technologies as the Internet, engineering and science applications and networks, business services and much more create data in exponential development because of the development of powerful storage and connection tools. Organized knowledge and information can't be easily obtained because of this enormous data growth and neither one of its can be effectively understood or automatically extracted. These premises have prompted to the development of data science or data mining, a well-known discipline which is more and more present in the current world of the Information Age [1].

Nowadays, the current amount of data managed by systems all around the globe have surpassed the processing capacity of more traditional systems, and this applies to data mining as well. The arising of new technologies and services (like Cloud computing [9]) as well as the reduction in hardware price are leading to an ever-growing rate of information on the Internet. This phenomenon certainly represents a "Big" challenge for the data analytics community. Big Data can be thus defined as very high volume, velocity and variety of data that require a new high-performance processing [10].

Distributed computing has been widely used by data scientists before the advent of Big Data phenomenon. Many standard and time-consuming algorithms were replaced by their distributed versions with the aim of agilizing the learning process. However, for most of the current massive problems, a distributed approach becomes mandatory nowadays since no centralized architecture that is able to tackle these huge problems [12].

Nonlinear dimensionality reduction (NLDR) is an attractive topic in many scientific fields. The task of NLDR is to recover the latent low-dimensional structures hidden in high dimensional data. In many areas of artificial intelligence and data mining, the encountered high-dimensional data are intrinsically distributed on a smooth, low-dimensional manifold [8].

The NLDR problem on such data is specifically called "manifold learning" problem [13]. In recent years, there have emerged many manifold learning approaches which are applied to many real-world application problems (e.g., hyperspectral imaging classification and object tracking), aiming at discovering the intrinsic geometric representations of the nonlinear data manifolds. Based on the intrinsic construction principles, these approaches can be divided into two categories: global and local approaches. Global approaches, such as Isomap and CDA, attempt to preserve geometry at both local and global scales, essentially constructing entire isometric corresponding between all datapairs in the original and latent spaces [14]. Local approaches, such as LLE and Laplacian eigenmaps, attempt to preserve the local geometry of the data, intrinsically keeping invariance between all local areas in the original and latent spaces [13].

PROBLEM

Suppose the data consist of N real-valued vectors X_i, each of dimensionality D, sampled from some underlying manifold. Provided there is sufficient data (such that the manifold is well-sampled), we expect each data point and its neighbors to lie on or close to a locally linear patch of the manifold. We characterize the local geometry of these patches by linear coefficients that reconstruct each data point from its neighbors. Reconstruction errors are measured by the cost function:

$$\varepsilon(W) = \sum_i \left| X_i - \sum_j W_{ij} X_j \right|^2 .$$

Which adds up the squared distances between all the data points and their reconstructions. The weights W_{ij} summarize the contribution of the jth data point to the ith reconstruction. To compute the weights W_{ij}, we minimize the cost function subject to two constraints: first, that each data point X_i is reconstructed only from its neighbors, enforcing $W_{ij} = 0$, if X_j does not belong to the set of neighbors of X_i;second, that the rows of the weight matrix sum to one: $\sum_j W_{ij} = 1$.

Suppose the data lie on or near a smooth nonlinear manifold of lower dimensionality $d << D$.To a good approximation then, there exists a linear mapping-consisting of a translation, rotation, and rescaling-that maps the high-dimensional coordinates of each neighborhood to global internal coordinates on the manifold. By design, the reconstruction weights W_{ij} reflect intrinsic geometric properties of the data that are invariant to exactly such transformations. We therefore expect their characterization of local geometry in the original data space to be equally valid for local patches on the manifold. In particular, the same weights W_{ij} that

reconstruct the ith data point in D dimensions should also reconstruct its embedded manifold coordinates in d dimensions.

DISTANCE PRESERVATION

This part discusses methods that reduce the dimensionality of data using the distance preservation criterion. Ideally, saving the pairwise distances measured in the dataset ensures that the low-dimensional nesting inherits the basic geometric properties of the data, such as the global shape or local neighborhood relations. Unfortunately, in the nonlinear case the distances cannot be completely preserved. We will discuss various methods that try to overcome this difficulty. These methods use different types of distances; they also rely on various algorithms or optimization procedures to determine the nesting [15].

Multidimensional scaling

The term multidimensional scaling (MDS) actually hides a family of methods, rather than one clearly defined procedure. Scaling refers to methods that build a configuration of points in the target metric space from information about intermediate distances, and MDS scales when the target space is Euclidean [7].

The optimization method is exact and purely algebraic: the optimal solution is obtained in a closed form. Metric MDS is also called the spectral method, since the kernel operation in its procedure is the EVD of the Gram matrix. The model is continuous and strictly linear. The mapping is implicit.

MDS is simple, durable, but strictly linear. But the metric MDS is flexible: it takes coordinates, as well as scalar products or Euclidean distances. On the other hand, the MDS metric requires memory to store the N-by-N Gram matrix. Another limitation is the generalization to new data points, which includes an approximate formula for the double centering phase [16].

Sammon's nonlinear mapping

In 1969, Sammon proposed a method for establishing a mapping between a high-order space and a lower one. The model is nonlinear and discrete, creating an explicit mapping [6].

NLM Sammon uses an approximate optimization procedure that can get stuck at a local minimum. The method does not include an estimate of its own dimension; the dimension size is actually fixed by the user. Incremental or layered attachments are not possible: the method must be run separately for each given dimension [11].

Compared to the classical MDS metric, NLM can efficiently handle nonlinear varieties, at least if they are not too complex. Among other non-linear versions of the metric MDS, the NLM remains relatively simple and elegant. As a major drawback, NLM does not have the ability to generalize mapping to new points.

Just like many other methods of preserving distance, NLM in its original version works with a full distance matrix, therefore, contains $O(N^2)$ records. This can be an obstacle when embedding very large data sets. Another drawback of NLM is its optimization procedure, which can be slow and/or inefficient for

some data sets. In particular, the function of the stress of Sammon is not guaranteed concave; therefore, the optimization process may depend on the local minimum.

Graph distances. In short, these methods try to overcome some of the shortcomings of spatial metrics, such as Euclidean distance. The next subsection presents both geodetic and graphical distances, explains how they relate to one another, and motivates their use in the context of diminishing dimensionality. The following subsections describe methods for reducing the non-linear dimension, which use graph distances.

Isomap

Isomap is the simplest method of NLDR, which uses the distance of the graph as an approximation of the geodetic distance. Isomap is an autonomous batch method that works with precise algebraic optimization. Since Isomap operates as a metric MDS, decomposing the Gram matrix into eigenvalues and eigenvectors is often qualified as a spectral method. In the literature, Isomap is described without preliminary processing of data, such as vector quantization [5]. The isomap relies on a nonlinear model. In fact, if the Euclidean distance can be considered as a "linear" metric, then the Isomap's ability to embed nonlinear varieties arises only because of the use of the distance of the graph [17]; Other parts of the method, such as the basic model of the optimization procedure, are based on the classical metric MDS and remain purely linear. Therefore, the Isomap data model is hybrid: the geodetic distance approximation along the face distances is discrete, whereas the subsequent MDS-like step can be considered continuous [18].

Isomap extends metric MDS in a very elegant way. However, the data model of Isomap, which relies on developable manifolds, still remains too rigid. Indeed, when the manifold to be embedded is not developable, Isomap yields disappointing results. In this case, the guarantee of determining a global error does not really matter.

Another problem encountered when running Isomap is the practical computation of the geodesic distances. The approximations given by the graph distances may be very rough, and their quality depends on both the data (number of points, noise) and the method parameters.

TOPOLOGY PRESERVATION

This part discusses methods that reduce dimensionality while preserving the data topology, rather than their pairwise distances. Preservation of topology seems more powerful, but more difficult to implement than keeping distance. The methods described are divided into two classes, depending on the type of topology used. The simplest methods are based on a predefined topology, whereas more modern methods prefer a topology built in accordance with a set of data that will be re-embedded.

Self-Organizing Maps

Along with multilayer perceptron (MLP), the self-organizing map is perhaps the most widely known method in the field of artificial neural networks. The SOM is

basically a vector quantization method. This means that vector quantization is mandatory in the SOM. As for the reduction in dimension, SOM models the data in a nonlinear and discrete way, representing it with a deformed lattice [4].

SOM is a neural network with learning without a teacher, performing the tasks of visualization and clustering. The idea of the network was proposed by the Finnish scientist T. Kohonen.

Most of the time, SOMs are implemented by standalone algorithms, similar to the Robbins-Monro procedure. Online versions can be easily obtained. There is a so-called "batch" version of SOM: instead of updating the prototypes, one after another, they all move simultaneously at the end of each epoch, as in the standard gradient descents [19].

The wide success of the SOM can be explained by the following advantages. The method is very simple from an algorithmic point of view, and its main idea, once understood, is intuitively attractive. Zombies are reliable enough and work very well in many situations, such as visualization of tagged data.

Nevertheless, the SOM have some known shortcomings, especially when they are used to reduce the dimension. Most implementations process only one or two-dimensional lattices. Vector quantization is mandatory, which means that the SOM does not really embed data points: base coordinates are calculated only for prototypes. Moreover, the form of the embedding is identical to the lattice, which, in turn, is determined in advance, arbitrarily. This means that the SOM cannot capture the shape of the data cloud in a low-dimensional attachment. From a computational point of view, it is very difficult to assess the convergence of the SOM, since an explicit objective function or an error criterion has been optimized [20]. In fact, it is proved that such a criterion cannot be determined, with the exception of some very special cases.

Generative Topographic Mapping

The generic topographic mapping (GTM) was put forward by Bishop, Swensen and Williams as a fundamental alternative to the SOM. In fact, GTM is a specific density network based on generative modeling, as indicated by its name. [3]

The essential difference between GTM and almost all the other methods is that the GTM is based on the Bayesian learning principle [21]. This probabilistic approach leads to another optimization method: instead of using (stochastic) gradient descent or spectral decomposition, the EM algorithm is used. As described above, GTM is a batch method, but there is also a version that works with the stochastic EM procedure. Since GTM defines the parameters of the generating model of the data, the diminution of the dimension is easily generalized to new points. Therefore, we can say that GTM defines an implicit mapping, although the hidden space is discrete.

If the implementation does not impose a two-dimensional grid, an external procedure is needed to evaluate the internal dimension of the data to determine the correct measurement for the hidden space.

Compared to SOM, GTM provides a generative data model. Moreover, the probabilistic approach that has come has a number of advantages. First, apart from finding the hidden coordinates x of the point y, GTM can also approximate p

(x | y), that is, the probability that the attachment will be located in x coordinates in a hidden space. This allows us to identify problems in diminishing dimensionality when, for example, the probability distribution is not unimodal [3].

Secondly, from an algorithmic point of view, GTM optimizes a clearly defined objective function. GTM optimizes the probability of using the EM algorithm. In comparison with these classical optimization methods, EM is guaranteed to maximize the probability monotonically and converge to a maximum after several dozen iterations [22].

CONCLUSIONS

Dimension reduction often plays an important role in the analysis, interpretation and understanding of numerical data. In practice, reducing the dimension can help to extract some information from arrays of numbers that would otherwise remain useless because of their large size. To a certain extent, the goal is to improve the readability of the data. This can be achieved by visualizing the data in diagrams, diagrams, graphs and other graphical representations.

An important motivation (NL) DR is the prevention of its harmful consequences. Paradoxically, however, many NLDR methods do not completely solve the problem, but only shy away from it. Many NLDR methods give poor results when the internal dimension of the underlying variety exceeds four or five. In such cases, the size of the embedding space becomes high enough to observe undesirable effects associated with the curse of dimension, for example, the phenomenon of empty space. The future will show whether new methods can solve this problem.

REFERENCES

1. *Big* Data prediction for 2013. Blog by Mike Gualtieri. (n.d.) — Available at: http://blogs.forrester.com/mike_gualtieri
2. *Big* Data prediction for 2013. Blog by Mike Gualtieri. (n.d.) — Available at: http://blogs.forrester.com/mike_gualtieri
3. *Horvath D.* Generative Topographic Mapping of Conformational Space / D. Horvath, I. Baskin, G. Marcou, A. Varnek // Molecular Informatics. — 2017. — **36** (10). — P. 22.
4. *Kohonen T.* Essentials of the self-organizing map / T. Kohonen // Neural Networks. — 2013. — N 37. — P. 52–65.
5. *Wang L.* The Isomap Algorithm and Topological Stability / L. Wang // Science. — 2002. — **295** (5552). — P. 81.
6. *Lerner B.* On pattern classification with Sammons nonlinear mapping an experimental study / B. Lerner // Pattern Recognition. — 1998. — **31**(4). — P. 371–381.
7. *Young F.* Multidimensional Scaling: History, Theory, and Applications / B. Lerner // Psychology Press. — 2017. — N 11. — P. 13.
8. *Lee J.* Nonlinear dimensionality reduction / J. Lee, M. Verleysen // NY: Springer. — 2010. — 29. — P. 110.
9. *Marinescu D.* Cloud Computing: Theory and Practice / D. Marinescu // Elsevier Science & Technology Books. — 2017. — 2. — P. 66.

10. Beyond the hype: Big data concepts, methods, and analytics. Egyptian Journal of Medical Human Genetics. Available at: https://www.sciencedirect.com/science/article/pii/ S0268401214001066
11. *Ewing R.* Visualization of expression clusters using Sammons non-linear mapping / R. Ewing R., J. Cherry // Bioinformatics. — 2001. — **17**(7). — P. 658–659.
12. *Dinh H.* A survey of mobile cloud computing: architecture, applications, and approaches / H. Dinh // Wireless Communications and Mobile Computing. — 2011. — **13**(18). — P. 1587–1611.
13. *Wang Q.* Combining local and global information for nonlinear dimensionality reduction / Q. Wang, J. Li // Neurocomputing. — 2009. — **72**(10–12). — P. 2235–2241.
14. *You S.* Think locally, fit globally: Robust and fast 3D shape matching via adaptive algebraic fitting / S. You, D. Zhang // Neurocomputing. — 2017. — N 89. — P. 119–129.
15. *Lee J.* Nonlinear projection with curvilinear distances: Isomap versus curvilinear distance analysis / J. Lee J., A. Lendasse, M. Verleysen // Neurocomputing. — 2004. — N 57. — P. 49–76.
16. *Cox T.* Multidimensional scaling / T. Cox, M. Cox // Boca Raton. — 2001. — 11. — P. 22.
17. *Law M.* Incremental nonlinear dimensionality reduction by manifold learning / M. Law, A. Jain // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2006. — **28**(3). — P. 377–391.
18. *Lee T.* Improved criteria for sampled-data synchronization of chaotic Lur'e systems using two new approaches / T. Lee, J. Park // Nonlinear Analysis: Hybrid Systems. — 2017. — 24. — P. 132–145.
19. *Du K.* Clustering: A neural network approach / K. Du // Neural Networks. — 2010. — **23**(1). — P. 89–107.
20. *Wang L.* Local Dynamic Modeling with SelfOrganizing Maps and Applications to Nonlinear System Identification and Control / L. Wang // Intelligent Signal Processing. — 2009. — 15. — P. 21.
21. *Svensen J.* GTM: the Generative Topographic Mapping / J. Svensen // University of Aston in Birmingham. — 1998. — 12. — P. 981.
22. *Ghahramani Z.* Unsupervised Learning / Z. Ghahramani // Advanced Lectures on Machine Learning Lecture Notes in Computer Science. — 2004. — 15. — P. 72–112.

Recieved 08.11.2017

From the Editorial Board: the article corresponds completely to submitted manuscript.