Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 83 (2016) 1025 – 1031
Procedia Computer Science

International Workshop on Data Mining for Decision Making Support (DMDMS 2016)

Dimensionality Reduction Method's Comparison Based On Statistical Dependencies

Tomas Vantuch[a,], Vaclav Snasel[a], Ivan Zelinka[a]

[a] *Department of Computer Science, VSB-Technical University of Ostrava, 17. listopadu 15 708 33, Ostrava-Poruba, Czech Republic.*

Abstract

The field of machine learning deals with a huge amount of various algorithms, which are able to transform the observed data into many forms and dimensionality reduction (DR) is one of such transformations. There are many high quality papers which compares some of the DR's approaches and of course there other experiments which applies them with success. Not everyone is focused on information lost, increase of relevance or decrease of uncertainty during the transformation, which is hard to estimate and only few studies remark it briefly. This study aims to explain these inner features of four different DR's algorithms. These algorithms were not chosen randomly, but in purpose. It is chosen some representative from all of the major DR's groups. The comparison criteria are based on statistical dependencies, such as Correlation Coefficient, Euclidean Distance, Mutual Information and Granger causality. The winning algorithm should reasonably transform the input dataset with keeping the most of the inner dependencies.

Peer-review under responsibility of the Conference Program Chairs

Keywords:
Principal Component Analysis, Non-negative Matrix Factorization, Autoencoder, Neighborhood Preserving Embedding, Granger Causality, Mutual Information

1. Introduction

The dimensionality reduction (DR) is a general title for mapping of larger dataset, which are compound of many variables into the smaller subset of uncorrelated features representing the maximal amount of information from the original data [1,2]. This methods can be categorized into two main groups. The first one is the linear DR whose basic idea comes from statistic, linear algebra and information theory and the second group consists of the non-linear algorithms which are mostly manifold-oriented or based on encoding driven by artificial neural network. It is obvious, that approaches which are so different can not produce results of the same meaning or quality and it has to be considered, which one is better for application on a given circumstances.

There are many high quality reviews, comparisons and a lot of applications where one kind of the DRs is significantly more suitable than another. The comparative approaches in this field of study can be categorized into the

* Corresponding author. Tel.: +420-597-325-278 ; fax: +420-596-918-507.
E-mail address: tomas.vantuch@vsb.cz

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2016.04.218

empiricaly based comparisons[3] which reveal advantages of one algorithm against another. They are focused on computational complexity, memory requirement, number of adjustable parameters or possible application of out-of-sample extension[4].

The other very frequently applied comparative method is the evaluation of performance of compared DRs by accuracy of applied classification algorithm[5,6].

The quantitative comparison of reconstructed time series is not frequently discussed topic because most of the non-linear and manifold based DRs are not able of strait-forward data reconstruction. On the other hand, the large reconstructing error does not imply the low quality of reduction[3].

These quantitative comparisons are the motivation of this paper. It attempts to measure the changes of the dependencies between the reconstructed time series (from their dimensionaly reduced state) and their original versions. The information criteria which serve as criteria to compare are Mutual Information, Granger causality, Correlation and Euclidean distance (section 3).

The input dataset is a matrix of a stock market prices. It contains 1548 rows which represent the time series of 387 investment symbols (open, high, low and close prices for each of them) and 1500 columns representing daily observations. The length of time series (1500 observations) represents 4 years and 40 days of daily prices and this dimension was continuously reduced and reconstructed for purposes of this experiment. The dataset was obtained from a public source (yahoo.finance).

In the next subsection, there are briefly described four applied approaches of dimensionality reduction. Section 3 covers some of the major notes about criteria that were controlled in this experiment. The sections 4 and 5 describe the results and conclusions of this paper.

2. Brief overview of dimensionality reduction methods

To reduce dimension of the dataset means to obtain dataset of lower dimension, where each of the observations is described by lower amount of un-correlated features (variables). These new extracted features are usually some linear or non-linear combinations of previous variables[1,2].

2.1. Principal Component Analysis

(PCA) is a linear DR technique[7]. PCA reduces the data by finding a few orthogonal linear combinations (principal components - PC) of the original variables with largest variance. The number of PCs is equal to the number of original variables, but only few of them holds the maximal variance. It is the reason why the rest of the principal components can be disregarded with minimal loss of information[8].

The mapping matrix M is found by equation $cov(X)M = \lambda M$, where $cov(X)$ is a covariance matrix of the input data and the matrix λ contains eigenvalues (PCs) of the covariance matrix on diagonal. The columns of matrix M are sorted according to the eigenvalues of the matrix λ. Reduced representation of the input data (matrix Y) is than computed from the first d principal components by equation $Y = XM_d$.

2.2. Non-Negative Matrix Factorization

(NMF) is a linear, non-negative matrix representation[9]. The idea of this method is to obtain two matrices W and H as a decomposition of the given matrix V of N vectors. The matrix W is $N \times M$ matrix of basis vectors and H is the new low-dimensional representation of the given matrix V.

The Alternating Least Squares (ALS) algorithm is one of the simplest approaches for obtaining such $W \times H$ representation for the given matrix[10]. The key point of this approach is the switching between two phases - once the H is fixed and W is found by a non-negative least squares solver and than W is fixed and H is found analogously. This methodology is based on the knowledge that NMF optimization function is not convex in both W and H properties, but it is convex in either W or H.

2.3. Autoencoder

The autoasociative neural network encoder (autoencoder) is the DR model based on Multi-layered perceptron model[11]. Based on the adjusted activation function of the network, this model is linear (the solution is strongly related to PCA) or non-linear (in case of commonly used sigmoid activation function)[12].

The structure of the ANN consists of two identical input and output layers and a variable number of the hidden layers with adjustable number of neurons. This similarity between input and output layer is required because the observations of the given matrix serve at the same time as the input and desired output values. The middle part - the bottleneck is made of smaller dimension and occurs in one hidden layer. The output values of this hidden layer are the low-dimensional representation of the input values.

The autoencoder can be trained by the Backpropagation learning algorithm[13], which is based on a gradient descent method.

2.4. Neighborhood Preserving Embedding

(NPE) is the linear approximation to the Local Linear Embedding (LLE)[14], which is the manifold oriented non-linear DR method[15]. NPE shares some aspects with Locality Preserving Projection[16] like both of them are focused to discover the local structure of the manifold. On the other hand, the objective function of these approaches is very different.

The algorithm comes through three steps. In the first step, it is a construction of the adjacency graph. There are two conditions which decides about putting the edges between the nodes. It could be decided by KNN (if the point i is the near neighbor of the point j) or by maximal distance adjusted by a threshold value (i and j are connected if $||x_j - x_i|| \leq \epsilon$).

In the second step of the algorithm, it has to be computed the weights of the connections. Let W denote the weight matrix where W_{ij} represents the weight of the edge between points i and j. The computation of the weights is achieved by the following objective function:

$$min \sum_i ||x_i - \sum_j W_{ij} x_j||^2 \tag{1}$$

where $\sum_j W_{ij} = 1, j = 1, 2, ...m$.

The last step is the computing of the projection, which is performed by solving of the generalized eigenvector problem.

$$\begin{gathered} XMX^T a = \lambda XX^T a \\ where \;\; X = (x_1, ...x_m); \;\; M = (I - W)^T (I - W); \;\; I = diag(1, ..., 1) \end{gathered} \tag{2}$$

The column vectors $a_0, a_1, ...a_{d-1}$ are the solution of equation (2), ordered according to their eigenvalues ($\lambda_0 \leq \lambda_1 ... \leq \lambda_{d-1}$) and it is compound into transformation matrix $A = (a_0, a_1, ...a_{d-1})$ so the formula of the reduction can be written as $y_i = A^T x_i$.

The theoretical justification and examples are in the original paper[14].

3. Information Criteria Measures

As it was mentioned before, this experiment is focused on comparison between the time series of reconstructed dataset (after reduction) and the time series from the original dataset. The comparisons are described in the following chapter. This section describes criteria that are applied in this paper.

The first simple comparison metric was the Euclidean distance, which simply computes the distance between two given coordinates in their N-dimensional space.

The other widely known measurements compare these time series by their statistical dependencies. The correlation coefficient as the covariance between X and Y divided by product of their standard deviation is very familiar linear dependency evaluation. The significance of the dependency is considered by the distance of the resulted value from zero. The negative value of the correlation coefficient indicates the anti-correlation - the progress of time series demonstrate opposite moves.

3.1. Mutual Information

(MI) is the non-linear evaluation of the dependency between two random variables[17]. In other words, it could be said that MI reveals how the presence of X decrease the level of uncertainty of the Y. This estimation is frequently applied for feature selection in the field of machine learning.

In case of discrete variables, the MI is defined by joint probability ($p(x, y)$) and marginal probabilities ($p(x)$,$p(y)$) of X and Y.

$$I(X; Y) = \sum_{x,y} p(x, y) log \frac{p(x, y)}{p(x)p(y)} \tag{3}$$

In case of continuous variables, when the probability distribution function (PDF) is unknown, the MI's estimation is not an easy task and it can be performed by various ways[18].

In this paper, it was used the Kernel Density estimation (KDE)[19], which can return more precise results with lower amount of adjustment.

3.2. Granger causality

(GC) is well known contribution proposed by Clive Granger[20] for the evaluation of causal interaction between two time series. This procedure quantifies how the variable Y helps to predict the variable X ($\mathcal{F}_{Y->X}$) by estimating their vector-autoregressive (VAR) model and computing covariance matrices of their residuals.

In this paper there was employed the simplest unconditional, time-domain form. The calculation of G-causality follows this simple steps provided by MVGC Matlab toolbox developed by Seth[21]. Let's suppose we have two stationary and trend-free variables X and Y and their VAR model is decomposed as

$$X_t = \sum_{k=1}^{p} A_{xx,k} \cdot X_{t-k} + \sum_{k=1}^{p} A_{xy,k} \cdot Y_{t-k} + \varepsilon_{xx,t} \tag{4}$$

The Acaice Information Criteria (AIC)[22] performs the proper adjustment of the number of lagged values of X and Y. To obtain stable VAR model[23], it has to be tested for colinearity, stationarity, heteroscedasticity, etc. In this equation, the dependency of X on Y is captured in the coefficient matrix A_{xy}. If all of these coefficients are equal to zero, we can say that X is independent from Y and the following regression of X possibly has the similar forecasting performance

$$X_t = \sum_{k=1}^{p} A'_{xx,k} \cdot X_{t-k} + \varepsilon'_{xx,t} \tag{5}$$

By this regression (5), it is written that variable X depends only on its past values and the added white noise. The unconditional, time-domain Granger Causality is than computed by following equation

$$\mathcal{F}_{Y->X} = ln \frac{|\Sigma'_{xx}|}{|\Sigma_{xx}|} \tag{6}$$

where Σ'_{xx} is the covariance of the residuals of the regression model (5) and Σ_{xx} is the covariance of the residuals of the VAR model (4).

Finally it is computed the p-value that that could reject the null hypothesis of zero causality, which is the following:

$$H_0 : A_{xy,1} = A_{xy,2} = ... = A_{xy,p} = 0 \tag{7}$$

In case of conditions of this paper (univariate causal target and smaller samples), the F-cumulative distribution function was applied to obtain the p-value (details in[21]).

4. Adjustments and results

This section describes the results of this experiment. There was applied one perspective of comparisons between original and reconstructed data. The entire experiment was focused on measuring changes in dependencies between the time series. It was achieved by computing dependency between the random time set x_i from the original dataset and its reconstructed version x_i^r from the reduced dataset.

The DR's methods implementation was provided by Matlab Toolbox for Dimensionality Reduction[3], except the implementation of NMF, which is already included in Matlab.

The PCA does not need any adjustable parameters, but every other methods needs to be properly adjusted.

The Autoencoder was applied with one input, one output and one hidden layer. The number of units inside of the hidden layer reflect the dimension of mapped matrix. The activation function was the sigmoid and then number of learning iterations was 5000.

The NPE method obtains the k value (number of nearest neighbors) equal to 6 and the NMF gets only increased number of iterations into 300.

4.1. Reconstruction of the dataset

Reconstruction of the dataset's matrix was provided differently for each of the DR's method according to its character. The PCA's reconstruction was performed in two steps. The first one was simply the computation of the product of the mapping (matrix of eigenvalues) and mapped matrices and the second step was adding a matrix of the mean values of the original dataset.

The NMF's reconstruction was computed as a product of W and H matrices as it is defined in the method's description.

The reconstruction of mapped data from Autoencoder is performed during each of the learning's iteration, because the neural network attempts to recreate the input sequence on the output layer. To perform the reconstruction, it is necessary to apply the learned network weights and put the mapped data as input values into the hidden layer. The output layer of the network will return the reconstructed data.

In case of NPE, the reconstruction was performed the same linear way as it was computed in case of PCA. The product of eigenvalues and mapped matrix is summed with the vector of means of the original matrix. This results into the lower quality of reconstruction, because the eigenvalues does not reflect the statistical behavior of the dataset, but the neighbor connections (adjagency matrix).

4.2. Dependencies between x_i and its reconstructed version

There was taken 100 random time series of the original dataset compared to their reconstructed versions and the results are the medians of the measured values. The measurement of the correlation (Fig. 1) reveals that in case of PCA and NMF the reduction of dimension does not significantly affect the level of correlation until the reduction reaches the level of 95%. These DR's methods were dealing with this feature reasonably well, but on the other hand the reconstructed time series from Autoencoder and NPE were absolutely uncorrelated with their original versions.

From the compared methods, the PCA also seems to be the ideal choice in times of keeping the maximal amount of Granger causality (see in Fig. 1). The focus on the variance through the eigenvalues was able to capture the behavior of the time series mostly until the reduction reaches the border near to 4% of original data set. The area between 50-25 was the most frequent place where the p-value drops under the level of significance - so there was no more presence of causality between compared time series.

In cases of other DR algorithms, the return values of GC were obtained very low and p-values were mostly confirming the null hypothesis of zero causality during all of the iterations.

The Fig. 2 describes the amount of MI between original and reconstructed time series. The DR methods like PCA and NMF were more successful than the rest of the methods in this criteria. The success of the Autoencoder strictly depends on its learning ability by its adjusted structure. This is the reason why most of its results are so inconsistent.

The values of Euclidean distance (Fig. 2) reveal that the maximal similarity (lowest values) was obtained again only by PCA and NMF algorithms. These methods obtain the smallest reconstruction error during the most of the reduction levels. The Autoencoder was not able to learn the given dataset for correct reduction, even if the number

Figure 1. Correlation coef. (left) and Granger causality (right) between original time set and its reconstructed version

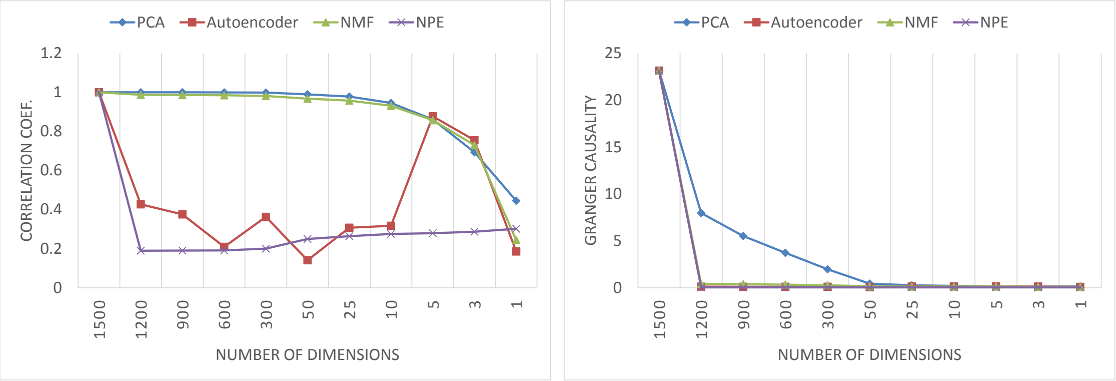
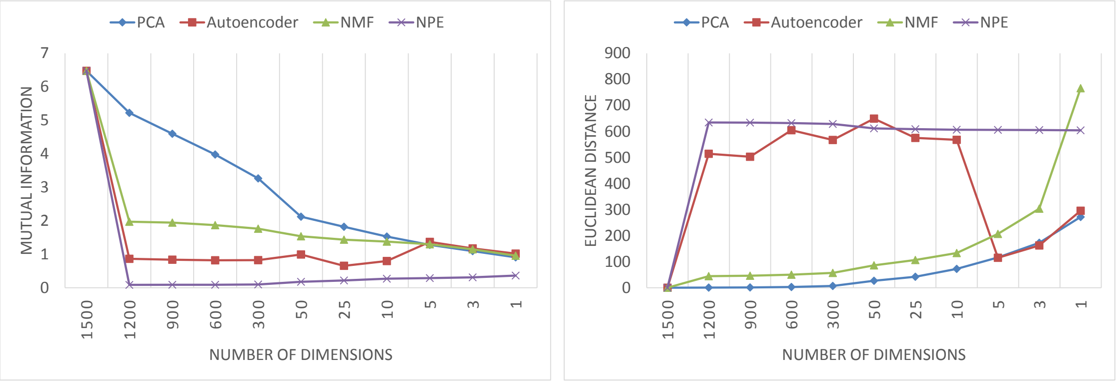


Figure 2. Mutual information (left) and Euclidean distances (right) between original time set and its reconstructed version



of its iterations was 5000. The NPE method deals with very high reconstruction errors, but as it was mentioned previously, it does not indicate the low quality of reduction, only different aim.

5. Conclusions

This experiment compares four ideologically different methods of DR. The comparison was made by measurement of the statistical dependencies between time series of the original and reconstructed dataset.

The described results reveal that algorithm based on manifold learning is not able to reflect inner basic statistical dependencies on its reconstructed data. This problem can be on the side of the reduction, which probably does not encode enough necessary features. On the other hand, it can also be the problem of the reconstruction, which recomputes the dataset in inappropriate way. This questions are the aim of the future work.

The only one applied non-linear algorithm, the Autoencoder, which is basically DR by ANN, seems harder to be maintained and properly adjusted due to higher number of parameters like number of layers, units per layer, type of learning algorithm, etc.. Its quality of the reduction is very sensitive to the quality of the ANN's learning ability.

The statistical dependencies were kept in reasonable quality by Principal Component analysis, due to its aim on statistical properties of the input dataset. The reasonably well results were obtained by Non-negative matrix factorization too, this method was able to keep correlation between compared time series. On the other hand the values of causality and mutual information were lower than in case of PCA.

The future work will be interested in Autoencoder based on multiple Restricted Boltzman Machines[24], in adding DR methods focused on statistical properties and on different manifold oriented algorithms.

ACKNOWLEDGMENT

Work is partially supported by Grant of SGS No. SP2016/97, SP2016/175, VSB - Technical University of Ostrava, Czech Republic

References

1. A. Rajaraman, J. D. Ullman, J. D. Ullman, J. D. Ullman, Mining of massive datasets, Vol. 77, Cambridge University Press Cambridge, 2012.
2. M. Kantardzic, Data mining: concepts, models, methods, and algorithms, John Wiley & Sons, 2011.
3. L. van der Maaten, E. O. Postma, H. J. van den Herik, Dimensionality reduction: A comparative review (2008).
4. Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. Le Roux, M. Ouimet, Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering, Advances in neural information processing systems 16 (2004) 177–184.
5. J. M. Banda, R. A. Angryk, P. C. Martens, Quantitative comparison of linear and non-linear dimensionality reduction techniques for solar image archives., in: FLAIRS Conference, 2012.
6. G. Lee, C. Rodriguez, A. Madabhushi, An empirical comparison of dimensionality reduction methods for classifying gene and protein expression datasets, in: Bioinformatics Research and Applications, Springer, 2007, pp. 170–181.
7. I. Jolliffe, Principal component analysis, Springer series in statistics, Springer-Verlang, 1986.
8. J. Shlens, A tutorial on principal component analysis, in: Systems Neurobiology Laboratory, Salk Institute for Biological Studies, 2005.
9. I. Buciu, Non-negative matrix factorization, a new tool for feature extraction: Theory and applications.
10. D. D. Lee, H. S. Seung, Algorithms for non-negative matrix factorization, in: T. Leen, T. Dietterich, V. Tresp (Eds.), Advances in Neural Information Processing Systems 13, MIT Press, 2001, pp. 556–562.
11. G. E. Hinton, P. Dayan, M. Revow, Modeling the manifolds of images of handwritten digits, Neural Networks, IEEE Transactions on 8 (1) (1997) 65–74.
12. P. Baldi, Autoencoders, unsupervised learning, and deep architectures, Unsupervised and Transfer Learning Challenges in Machine Learning, Volume 7 (2012) 43.
13. D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back-propagating errors, in: J. A. Anderson, E. Rosenfeld (Eds.), Neurocomputing: Foundations of Research, MIT Press, Cambridge, MA, USA, 1988, pp. 696–699.
14. X. He, D. Cai, S. Yan, H.-J. Zhang, Neighborhood preserving embedding, in: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on, Vol. 2, 2005, pp. 1208–1213 Vol. 2.
15. S. Roweis, L. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326, cited By 5799.
16. X. Niyogi, Locality preserving projections, in: Neural information processing systems, Vol. 16, MIT, 2004, p. 153.
17. D. J. C. MacKay, Information Theory, Inference & Learning Algorithms, Cambridge University Press, New York, NY, USA, 2002.
18. J. Walters-Williams, Y. Li, Estimation of mutual information: A survey, in: P. Wen, Y. Li, L. Polkowski, Y. Yao, S. Tsumoto, G. Wang (Eds.), Rough Sets and Knowledge Technology, Vol. 5589 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2009, pp. 389–396.
19. Y.-I. Moon, B. Rajagopalan, U. Lall, Estimation of mutual information using kernel density estimators, Physical Review E 52 (3) (1995) 2318–2321.
20. C. W. J. Granger, Essays in econometrics, Harvard University Press, Cambridge, MA, USA, 2001, Ch. Investigating Causal Relations by Econometric Models and Cross-spectral Methods, pp. 31–47.
21. L. Barnett, A. K. Seth, The mvgc multivariate granger causality toolbox: A new approach to granger-causal inference.
22. H. Akaike, A new look at the statistical model identification, Automatic Control, IEEE Transactions on 19 (6) (1974) 716–723.
23. J. D. Hamilton, Time series analysis, Vol. 2, Princeton university press Princeton, 1994.
24. T. N. Sainath, B. Kingsbury, B. Ramabhadran, Auto-encoder bottleneck features using deep belief networks, in: Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on, IEEE, 2012, pp. 4153–4156.