Vilniaus Universitetas
Matematikos ir informatikos institutas
L I E T U V A

INFORMATIKA (09 P)

DIDELĖS APIMTIES DUOMENŲ VIZUALI ANALIZĖ

Jelena Liutvinavičienė

2017 m. spalis

Mokslinė ataskaita MII-DS-09P-17-7

Matematikos ir informatikos institutas, Akademijos g. 4, Vilnius LT 08663

http://www.mii.lt/

Abstract

This research focuses on big data visualization that is based on dimensionality reduction methods. We propose a multi-level method for data clustering and visualization. Whole data mining process is divided into separate steps. For each step particular dimensionality reduction and visualization method is applied considering to data volume and type. The selection of methods is based on their speed and accuracy. Therefore the comparison of the selected methods is made according to these two criteria. Three groups of datasets containing different kind of data are used for methods evaluation. The factors that influence speed or accuracy are determined. The rank of investigated methods based on research results is presented in this paper.

Keywords: big data, dimensionality reduction, data visualization

Turinys

1 Introduction

Big data analytics is the process of examining big data to uncover hidden and useful information for better decisions. It involves visual presentation of data that enables to see hidden relations between objects which cannot be detected using conventional data analysis methods [15]. This particular research focuses on big data visualization that is based on dimensionality reduction methods. Our goal is to find the most effective ways to analyse and visualize data of such type. Dimensionality reduction refers to the process of taking a data set with a usually large number of dimensions, and then creating a new data set with a fewer number of dimensions, which are in some sense "important". The idea here is that we want to preserve as much "structure" of the data as possible, while reducing the number of dimensions [7].

In our approach, whole data mining process is divided into separate steps. For each step a particular dimensionality reduction and visualization method is applied considering to data volume and type. The selection of methods is based on their speed and accuracy. Therefore, the comparison of dimensionality reduction methods is presented in this work.

We propose a method where data is clustered and visualized on the surface of sphere. There is ability to see the parameters of each data group. The further analysis is performed only for the selected data cluster.

At the initial stage the accuracy of method is not so important, so the fastest visualization method can be used. For the following dimensionality reduction steps the demand for accuracy gradually increases. This requires using more accurate, but possibly slower methods. During each step, the selected data cluster is divided into smaller sets. At the end the most accurate method processes the data. It would require too many resources at the beginning of dimensionality reduction, but at the end the data set is small enough to be processed in the most accurate way.

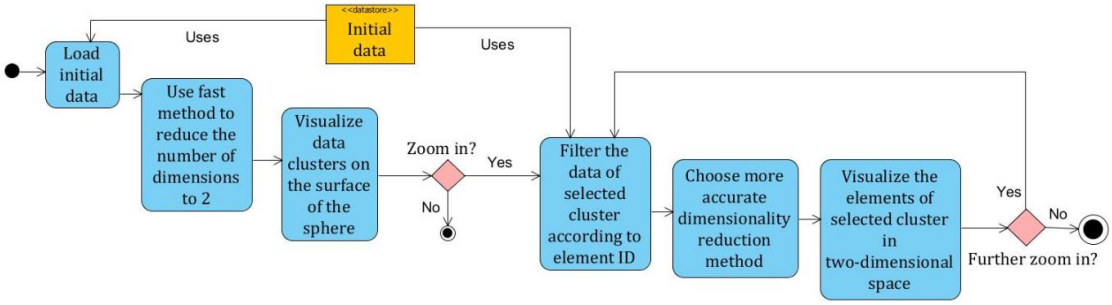

Figure 1. Multi-level method for big data visualisation

Most often there are just qualitative comparisons of different dimensionality reduction methods [2][13][12]. In some papers [7][5], we can also find speed or accuracy comparisons of selected methods. The review of such researches leads to insight that some methods are faster, but slower and that other ones have opposite characteristics. However, there is no general quantitative research of most popular methods that would compare both speed and accuracy.

Further in this paper, we investigate these well-known methods: Multidimensional Scaling (MDS), Principal Component Analysis (PCA), Independent Component Analysis (ICA), Principal Curves, Locally Linear Embedding (LLE), Isometric Mapping (Isomap).

2 Review of dimensionality reduction methods

The detailed reviews of dimensionality reduction methods were done by I. K. Fodor (2002), M. Mizuta (2004), C.O.S. Sorzano, J. Vargas et. al. (2014). In this section, we present a brief summary of the most popular methods. The demand for such methods rises, because various sources generate enormous amount of data, e.g. laboratory instruments can report thousands measurements for a single experiment, and the statistical methods face challenging tasks when dealing with such high-dimensional data. However, according to C.O.S. Sorzano, J. Vargas et. al. (2014), much of the data are highly redundant and can be efficiently brought down to a much smaller number of variables without a significant loss of information.

Multidimensional scaling

Given n items in a *d*-dimensional space and *n x n* matrix of proximity measures among the data items, multidimensional scaling (MDS) produces a *k*-dimensional, $k \leq d$, representation of the items such that the distances among the points in the new space reflect the proximities in the data [2].

The proximity measures the (dis)similarities among the items, and in general, it is a distance measure: the more similar two items are, the smaller their distance is. Popular distance measures are Euclidean distance, the Manhattan distance and the maximum norm [2].

In this research we use *mds()* function from R package 'smacof', which solves the stress target function for symmetric dissimiliarities by means of the majorization approach (SMACOF) and reports the Stress-1 value (normalized). This function allows for fitting three basic types of MDS: ratio MDS (used in our case), interval MDS (polynomial transformation), and ordinal MDS (also known as nonmetric MDS) [11].

Principal components analysis (PCA)

As long as data have a near-linear structure, the singularities of the data can be pointed out using Principal Component Analysis (PCA) [8]. PCA is by far one of the most popular algorithms for dimensionality reduction [13]. PCA finds components that make projections uncorrelated by selecting the highest eigenvalues of the covariance matrix and maximizes retained variance [1]. The theoretical idea behind PCA is that we find the principal components of the data, which correspond to the components along which there is the most variation [7].

Independent component analysis (ICA)

ICA is a higher-order method that seeks linear projections, not necessarily orthogonal to each other, that are as nearly statistically independent as possible. Statistical independence is a much stronger condition than uncorrelatedness. ICA can be considered as a generalization of the PCA and the Projection pursuit concepts. While PCA seeks uncorrelated variables, ICA seeks independent variables [2].

Principal curves, surfaces and manifolds

PCA is a perfect tool to reduce data that in their original k-dimensional space lie in some linear manifold. However, there are situations at which the data follow some curved structure. In this case, approximating the curve by a straight line will not perform a good approximation of the original data. For such type data the solution is to use principal curves, surfaces and manifolds [13].

Curve fitting to data is an important method for data analysis. When we obtain a fitting curve for data, the dimension of the data is nonlinearly reduced to one dimension [8].

Locally linear embedding (LLE)

Locally linear embedding (LLE) and Isomap (see below) together with MDS and Kernel PCA are called spectral dimensionality reduction techniques, because they are based on the eigenvalue decomposition of some matrix.

LLE method is used to learn manifolds close to the data and project them onto it. For each item we look for the K-nearest neighbours and produce a set of weights for its approximation. This optimization is performed simultaneously for all items. Once the weights have been determined, we look for points of lower dimension. The new points have to be reconstructed from its neighbours in the same way (with the same weights) as the items they represent. This latest problem is solved by solving an eigenvalue problem and also keeping the smallest eigenvalues [2].

Isometric mapping (Isomap)

If the distances between objects are measured as geodesic distances, then the MDS method is called Isomap. The geodesic distance between two points in a manifold is the one measured along the manifold itself; in practical terms it is computed as the shortest path in a neighborhood graph connecting each observation to its K-nearest neighbors [2].

3 Research methodology

The main goal of this research is to compare the speed and accuracy of the selected methods of visualization based on dimensionality reduction. R was chosen as a basis for analysis, because there are various open source packages that enable to execute and evaluate different dimensionality reductions methods. RStudio environment was used to perform the tasks.

3.1 Data

Three groups of different kind datasets were created for testing purposes.

Randomly generated nonclustered data

First of all, 50 different datasets containing randomly generated numbers were created with R function *sample()*. The number of columns is from 10 to 50. The number of items is from 1000 to 10000. So the smallest dataset is 1000x10 and the largest one is 10000x50.

Randomly generated clustered data

The second group contains 25 datasets of clustered data. The function *genRandomClust* from R package 'clusterGeneration' was used to generate cluster datasets with specified degree of separation [9]. Each dataset has 4 clusters. The number of columns is from 10 to 50. The number of items is from 1000 to 9000. The smallest dataset is 1000x10 and the largest one is 9000x50.

Real financial data

The third group contains 20 datasets of real financial data – stock ratios from finviz.com [14]. In total there are information about 7000 companies. Each company is described by 50 parameters, which can be grouped into 6 categories: overview (market capitalization, price, volume etc.), valuation (P/E, PEG, P/B, EPS etc.), financial (ROA, ROE, ROI etc.), performance (price changes, volatility, recommendations), technical (ATR, Beta, SMA etc.), ownership.

The number of columns in datasets is from 10 to 50. The number of items is from 1000 to 7000. The smallest dataset is 1000x10 and the largest one is 7000x50. In all cases of our research the initial number of dimensions is reduced to 2.

3.2 Evaluation criteria

We use 2 main criteria to compare different methods:

- **Speed.** It is measured as execution time of dimensionality reduction process.
- **Accuracy.** We use 3 different measures to evaluate the accuracy:
    - **Stress** – the measure got by solving the square loss function of MDS method. We used R function *mds()* from package ‘smacof’ to find the stress value.
    - **Spearman coefficient** (The Spearman's Rank Correlation Coefficient). It is a statistical measure used to discover the strength of a link between two sets of data [3]. This ratio uses the ranks of variables instead of their values. Possible values range from -1 (strong negative relation) to 1 (strong positive relation). If ratio is equal to zero, this means there is no statistical link between datasets. To calculate this ratio R function *cor()* with method “spearman” was used.
    - **Shannon entropy**. We used R function *entropy* from package ‘entropy’ that estimates the Shannon entropy H of the random variable Y from the corresponding observed items [10][4]. This estimator shows how accurate the projection got by using particular dimensionality reduction method retains the initial amount of information. Less value of this measure means better accuracy.

4 Research results

In this section, we present the results of speed and accuracy comparison for each group of data. At the end the overall comparison is made.

4.1 Randomly generated nonclustered data

In the first case randomly generated nonclustered datasets are used for investigation.

Speed of methods

As results show, MDS (smacof), Isomap and LLE methods have the same characteristics:

- When number of instances increases the execution time also increase.
- The initial amount of dimensions doesn't have significant effect for the time of execution.

Fig. 2 shows the execution time of MDS (smacof) method for datasets that contain 10 columns, but differ in number of rows. The charts of execution time for the datasets having more columns look the same, because this factor has no influence. However Isomap is much slower, this can be seen in Fig. 6.

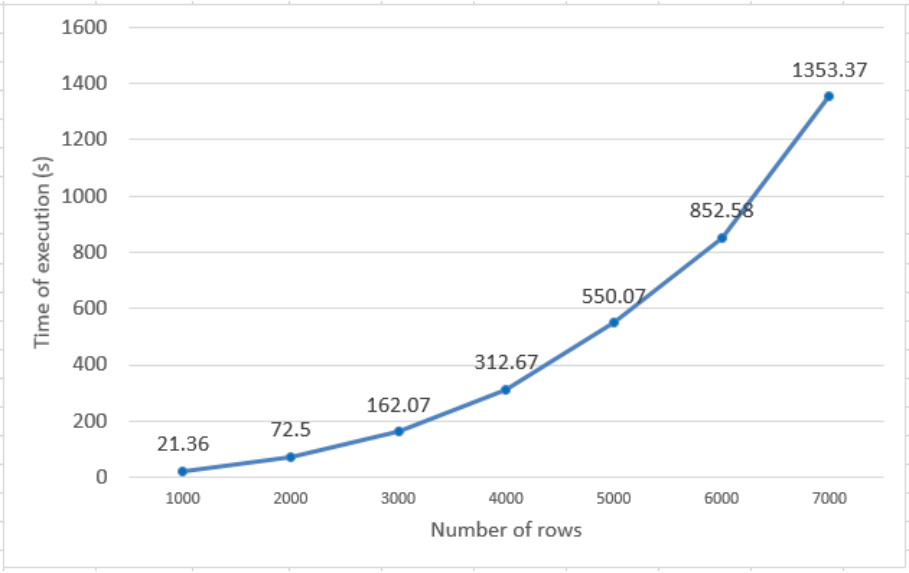

Figure 2. Execution time of MDS (smacof) method

For PCA the execution time just slightly increases in both cases: when number of rows increases and when number of columns increases (Fig. 3):

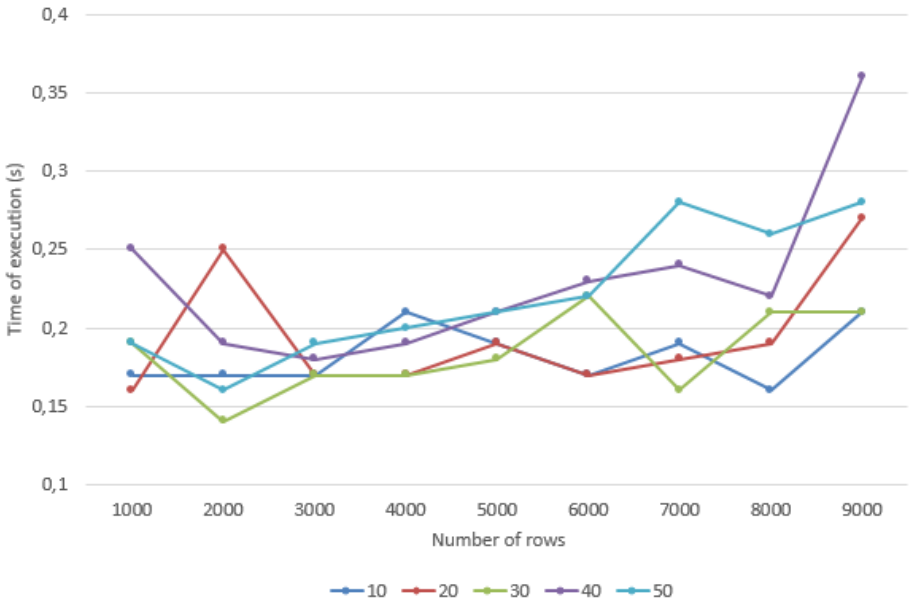

Figure 3. Execution time of PCA

The execution time of ICA is similar to PCA. Only Principal curves distinguish by regular increase of execution time in both cases (when dimensions and instances increase) (Fig. 4).

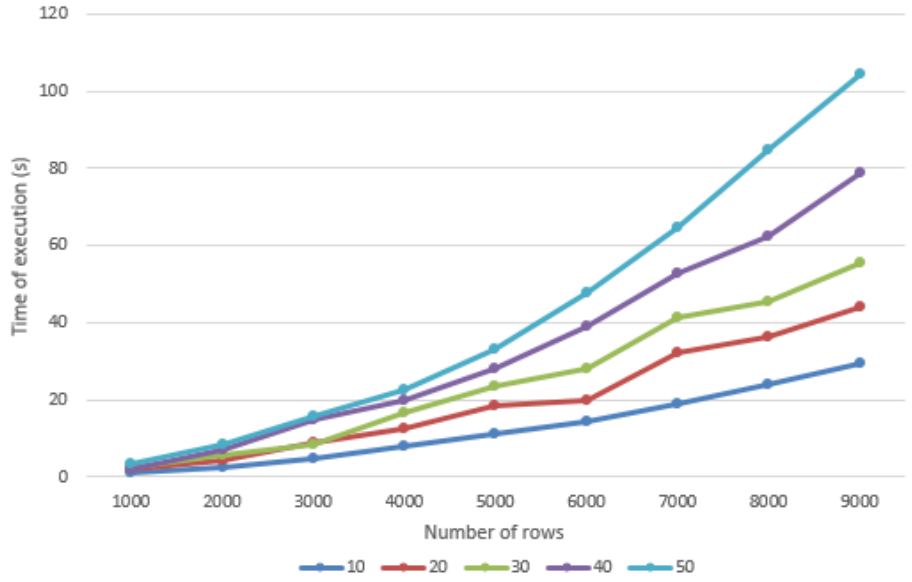

Figure 4. Execution time of Principal curves

In Fig. 5 the execution times of tested methods are compared. The number of instances is from 1000 to 10000. The initial number of dimensions doesn't have significant influence for any method, so only one case with 40 dimensions is presented.

	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000
PCA	0.25	0.19	0.18	0.19	0.21	0.23	0.24	0.22	0.36	0.28
ICA	0.18	0.22	0.19	0.22	0.18	0.18	0.16	0.25	0.29	0.23
Principal Curves	2.21	6.89	14.73	19.98	28.29	39.22	52.55	62.47	78.98	105.88
LLE	2.00	13.22	41.51	93.66	178.67	304.12	486.33	727.57	1029.79	-
MDS (smacof)	15.76	65.64	173.64	340.48	624.07	981.79	1482.42	-	-	-
Isomap	39.75	350.41	1229.72	3168	6222.66	11078.34	17710.56	41077.29	-	-

Figure 5. Comparison of execution times

It should be noted, that LLE couldn't process the datasets with more than 9000 rows, Isomap couldn't process more than 8000 rows and for MDS (smacof) the maximum was 7000 rows. This was due to the lack of RAM. In Fig. 6 the speed of methods is presented in logarithmic scale. PCA and ICA are the fastest. Principal curves, MDS and LLE are much slower. However, Isomap is the slowest (its execution time is significantly longer than others).

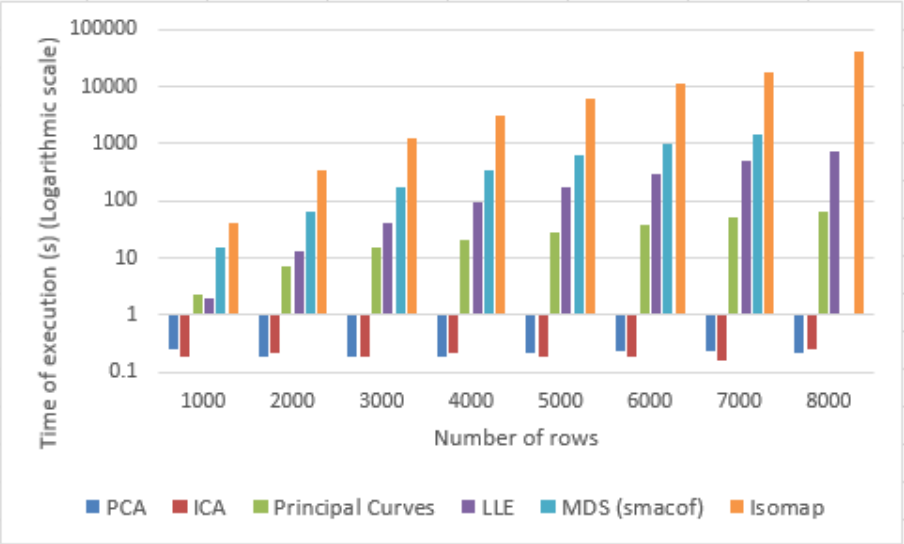

Figure 6. Comparison of execution times

Accuracy of methods

For all investigated methods we found that the same rules apply:

- When number of instances increases, the accuracy doesn't change
- When number of initial dimensions increases, this leads to worse accuracy

This was confirmed by all measures. However, the level of accuracy reduction is not the same for different methods. The Fig. 7-8 compare the accuracy of all analysed methods. As the number of instances doesn't make significant influence, we show only the cases with 7000 instances. In all cases with different number of initial dimensions, the results are similar. Therefore we present only two of them: 20 dimensions and 40 dimensions.

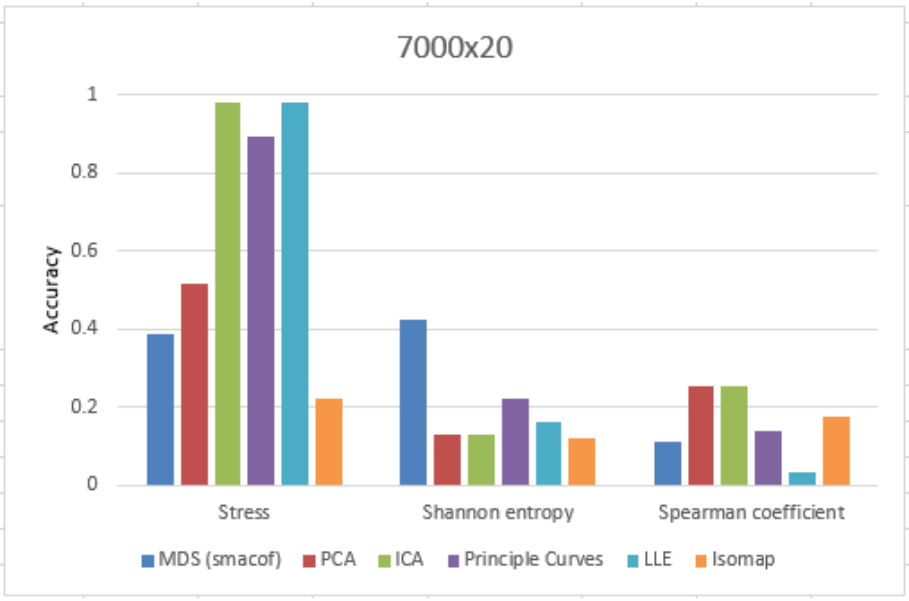

Figure 7. The comparison of accuracy measures

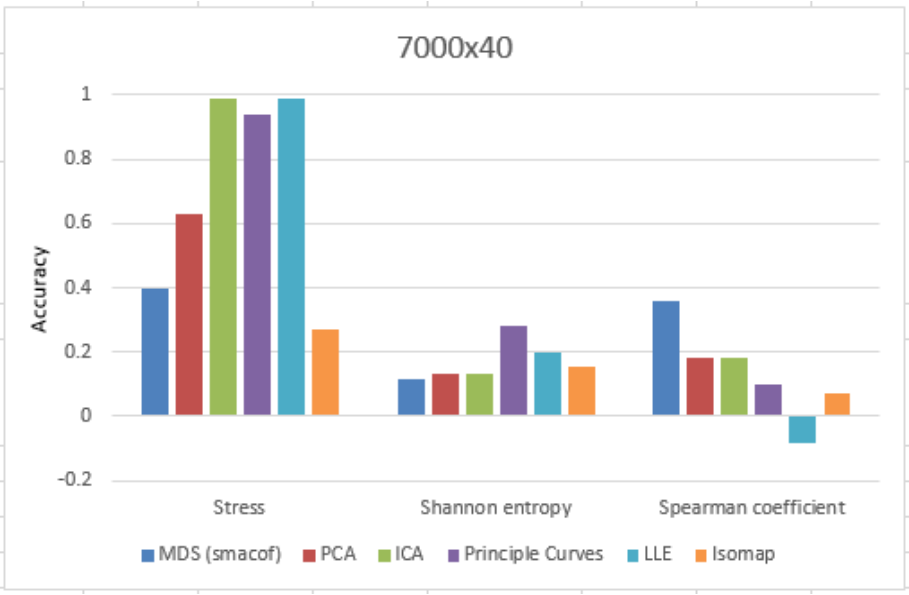

Figure 8. The comparison of accuracy measures

The results show that PCA and MDS were the most accurate with our datasets. LLE showed the worst accuracy.

The rank of methods

The Fig. 9 summarizes the results of nonclustered data case. We ranked all investigated methods by their speed and accuracy ("6" means the highest score and "1" stands for the worst score).

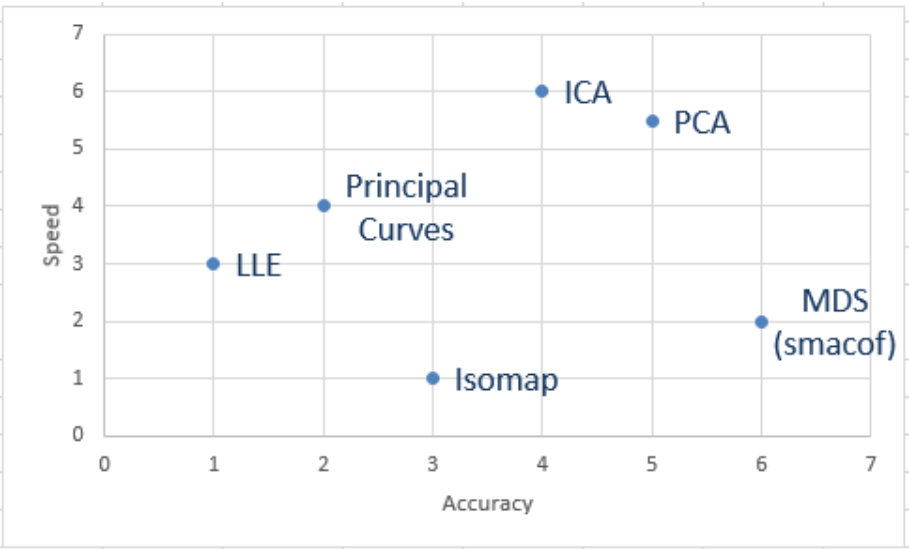

Figure 9. The comparison of methods by speed and accuracy

PCA and ICA are the fastest methods. MDS is the most accurate, but slower. Principal curves showed moderate results. The results of LLE and Isomap are the worst. Although Isomap is significantly slower, but its accuracy in some cases can be the best.

4.2 Randomly generated clustered data

In the second case randomly generated clustered datasets are used for investigation.

Speed of methods

For MDS (smacof), Isomap and LLE the same trends as with nonclustered data can be seen:

- When number of instances increases the execution time also increase.
- The initial amount of dimensions doesn't have significant effect for the time of execution.

Fig. 10 shows the execution times of these methods for datasets from 1000x10 to 7000x10.

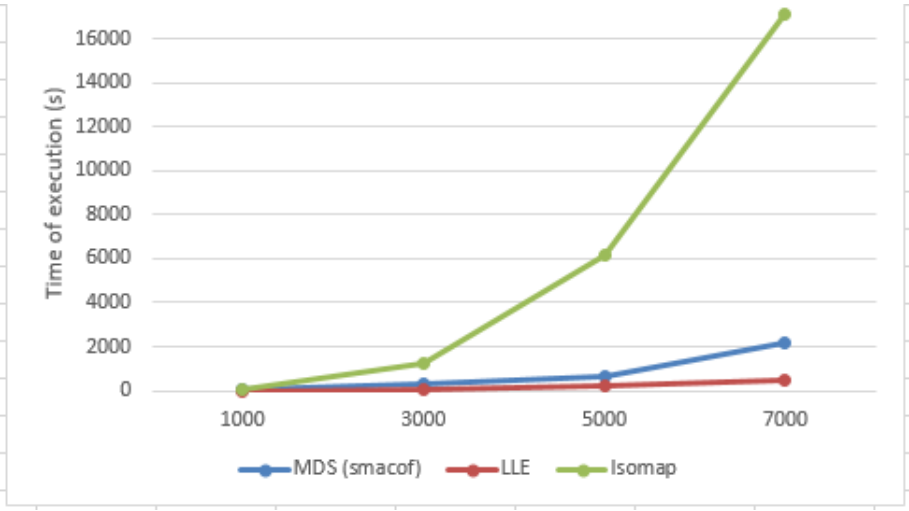

Figure 10. Execution time of MDS (smacof), Isomap and LLE methods

For PCA the execution time slightly increases (with some exceptions) when both number of rows and number of columns increases. In this case the execution time of ICA is also similar to PCA. With clustered data there is no such obvious regular increase of the execution time when using Principal curves method (Fig. 11), which can be seen in Fig. 4.

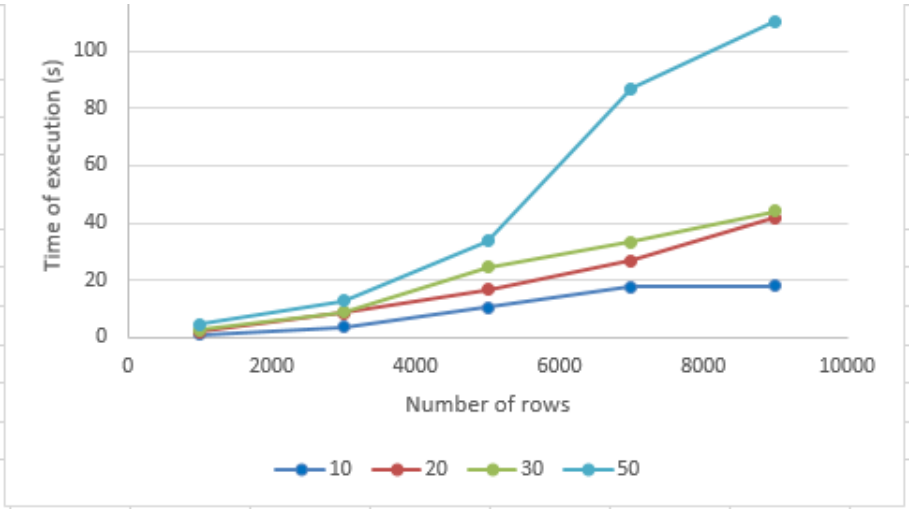

Figure 11. Execution time of Principle curves method

In Fig. 12 the execution times of different methods are compared. The number of instances is 1000, 3000, 5000, 7000 and 9000. In this case the number of dimensions doesn't have significant influence for any method, so only one case with 40 dimensions is presented.

	1000	3000	5000	7000	9000
PCA	0.21	0.2	0.21	0.25	0.23
ICA	0.19	0.17	0.19	0.19	0.17
Principal Curves	2.62	11.91	22.81	49.08	109.98
LLE	2.14	41.45	184.04	484.42	-
MDS (smacof)	14.81	185.36	42.125	1750.67	-
Isomap	38.33	1219.35	6297.34	18008.4	-

Figure 12. Comparison of execution times (with 40 initial dimensions)

The results are similar to that got previously with nonclustered data (Fig. 6). However, in this case LLE, MDS (smacof) and Isomap couldn‘t process the datasets with 9000 rows.

Accuracy of methods

For all methods we found that the same rules apply as with nonclustered data:

- When number of instances increases, the accuracy doesn‘t change
- When number of initial dimensions increases, this leads to worse accuracy

The Fig. 13 shows the accuracy values got with dataset that contains 7000 rows and 40 columns. MDS (smacof) is the most accurate by two measures: Shannon entropy and Spearman coefficient. However, according to Stress, Isomap is more accurate than MDS (smacof). PCA and ICA showed the moderate results. The accuracy of LLE and Principle curves is the worst.

The results of speed and accuracy with clustered data are almost the same as with nonclustered data.

4.3 Real financial data

In the third case the real stock data are used for comparison of dimensionality reduction methods.

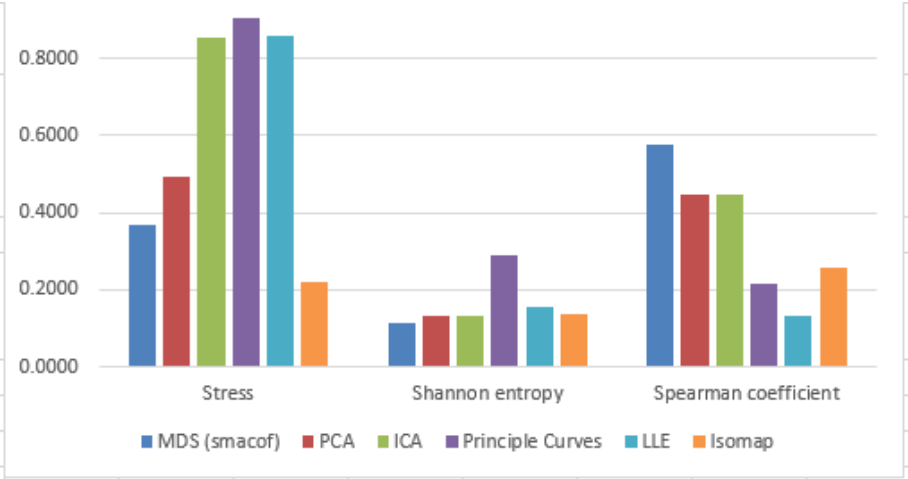

Figure 13. The comparison of accuracy ratios (dataset 7000x40)

Speed of methods

It may seem that MDS (smacof) has the same characteristics (when number of instances increases the execution time also increase; the initial amount of dimensions doesn't have significant effect for the time of execution). However, in the case with real data, we found that execution time slightly increases when number of initial dimensions increases (Fig. 14). This contrary relationship is unusual and needs further investigation.

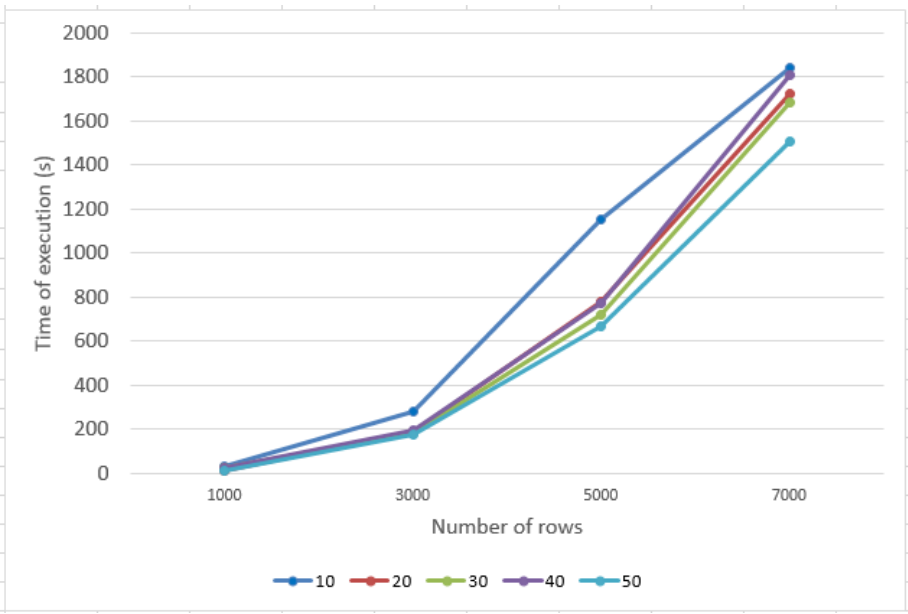

Figure 14. Execution time of MDS (smacof) method

For the remaining methods the trends of speed are the same as in previous cases. However, it was impossible to process the real data with LLE method. It found data too much correlated.

	1000	3000	5000	7000
PCA	0.18	0.15	0.19	0.23
ICA	0.19	0.17	0.16	0.20
Principal Curves	4.43	17.27	38.89	81.25
MDS (smacof)	22.58	198.82	774.7	1809.37
Isomap	42.37	1130.76	6175.44	16599.5
LLE	-	-	-	-

Figure 15. Comparison of execution times

Accuracy of methods

With real data we couldn't get the measures not only for LLE method, but also the Stress value of ICA. This confirms that all methods can cope with generated data, but the real world situations may cause issues to them.

Fig. 16 shows the results in case with 7000 instances and 40 dimensions. MDS (smacof), PCA, ICA and Isomap show similar results with all datasets (accuracy depends on the initial amount of dimensions, but the trends remain the same).

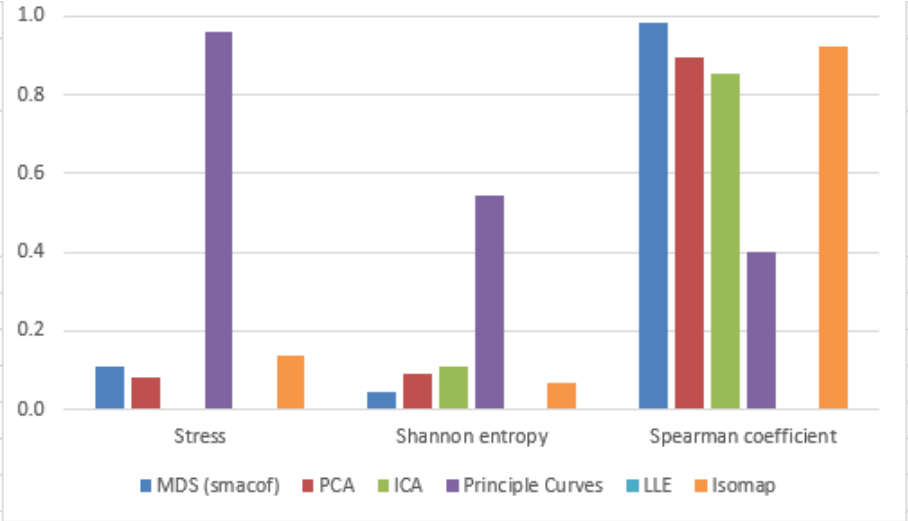

Figure 16. Comparison of accuracy (dataset 7000x40)

However, with Principle curves we couldn't confirm one rule for all datasets. Fig. 17-18 show that when the number of initial dimensions constantly increases the values of Spearman coefficient and Shannon entropy fluctuates. This leads to suggestion, that information which can be extracted from data has impact for the accuracy of dimensionality reduction. This is why adding more columns of randomly generated data is not the same as adding more real data, which can add completely different aspects for analyzed subject.

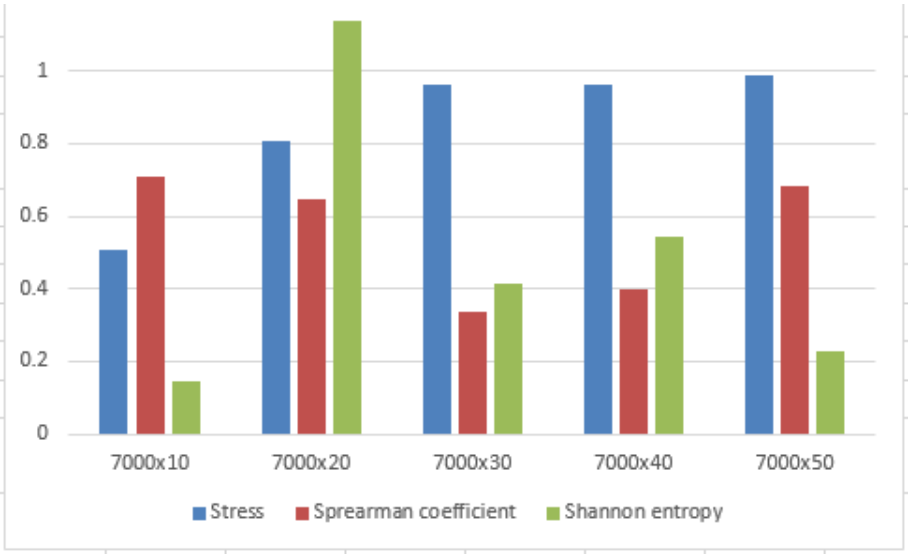

Figure 17. Accuracy of Principal curves

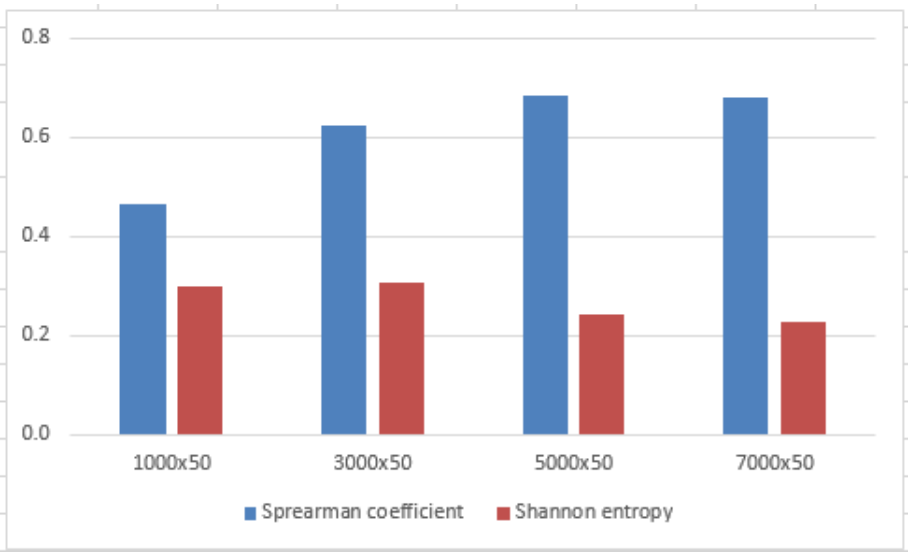

Figure 18. Accuracy of Principal curves

Fig. 17-18 shows that more items lead to better accuracy. This feature is seen only with real data.

The rank of methods

Fig. 19 shows the rank of methods according to their speed and accuracy while processing the real data. The results show that MDS is the most accurate method. However, it's not as fast as PCA or ICA. These are the fastest methods, but they showed moderate accuracy values. The speed of ICA is the same as PCA, but it's not so accurate.

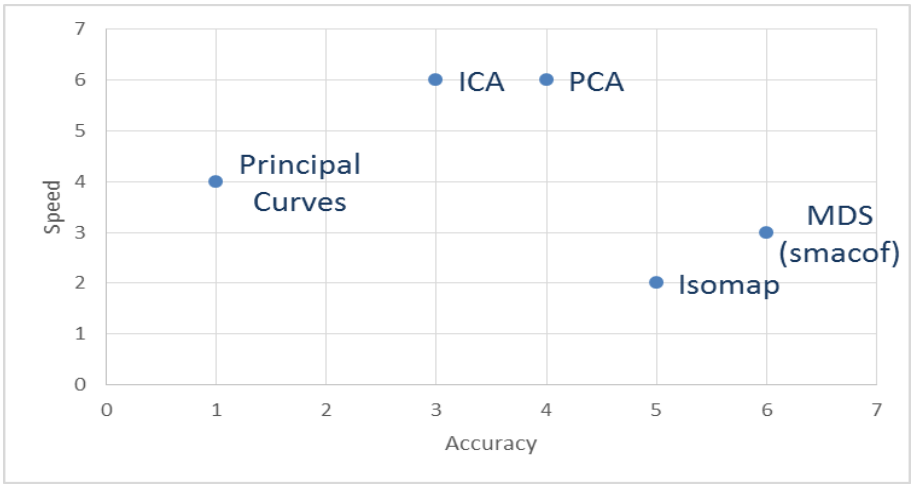

Figure 19. The comparison of speed and accuracy

4.4 Overall comparison

In this section, we present how the speed and accuracy of dimensionality reduction methods depend on the kind of data. Fig. 20 shows that the kind of data is not important for the speed of methods. It doesn‘t affect the time of execution.

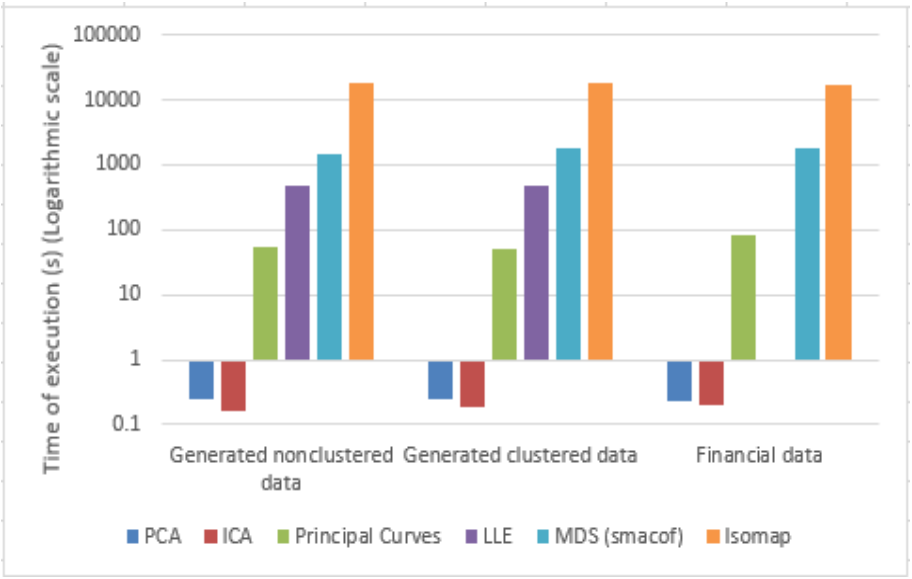

Figure 20. Execution times for different kind of data

However, it has infuence on the accuracy. Clustered data has better Stress values than nonclustered data. Moreover, PCA, MDS (smacof) and Isomap showed best accuracy exactly with real stock data (Fig. 21).

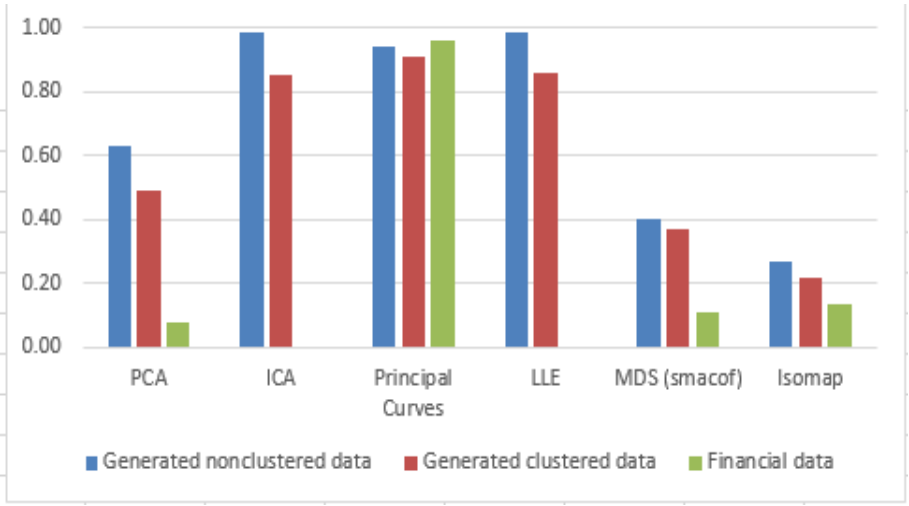

Figure 21. Comparison of accuracy (Stress)

According to Spearman coefficient (Fig. 22) the best accuracy is also achieved when processing the real data. Clustered data also has higher accuracy values than nonclustered data.

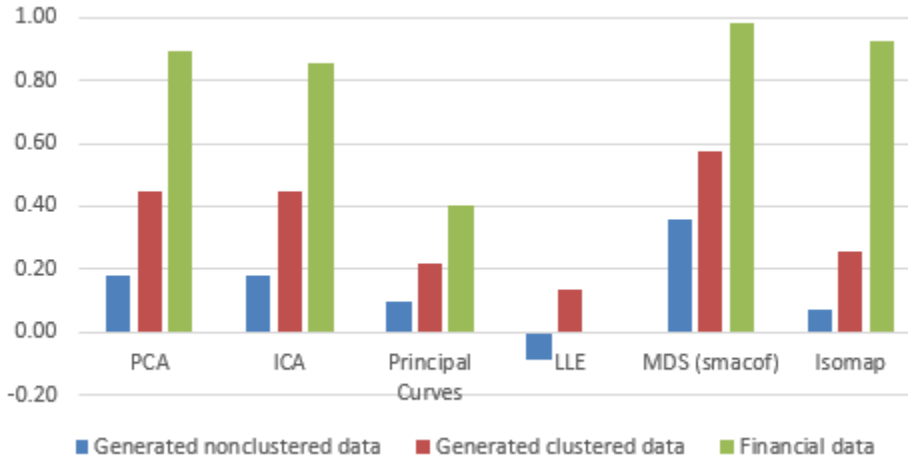

Figure 22. Accuracy measures: Spearman coefficient

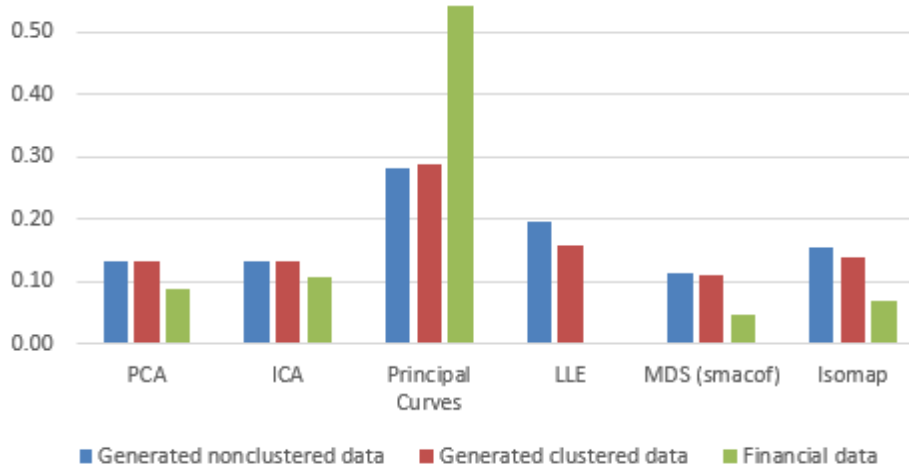

Figure 23. Accuracy measures: Shannon entropy

According to Shannon entropy (Fig. 23), there is no significant difference of accuracy between clustered and nonclustered data. But again, the accuracy is much better in case with the real data (except Principal curves method).

5 Conclusions

This particular research focuses on big data visualization that is based on dimensionality reduction methods. In our approach all data mining process is divided into separate steps. For each step individual dimensionality reduction and visualization method is applied considering to data volume and type. The selection of methods is based on their speed and accuracy. Therefore in this paper we presented the comparison of dimensionality reduction methods according to these two criteria. 3 different measures were used to evaluate the accuracy: Stress, Spearman coefficient and Shannon entropy. All methods were tested with 3 groups of different kind of data: nonclustered randomly generated data, clustered randomly generated data and real financial data.

Several rules were confirmed for randomly generated data (both clustered and nonclustered). When number of instances increases the execution time also increases. However, the initial amount of dimensions doesn't have significant effect for the time of execution. For accuracy there is opposite situation. When number of instances increases, the accuracy doesn't change, but when number of initial dimensions increases, this leads to worse accuracy.

Meanwhile in the case with real data we found that execution time can slightly increase when number of initial dimensions increases. It was also impossible to process the real data with LLE method and get Stress values of ICA. This shows that the real world situations may cause issues to particular methods. The results also show that more instances of real data lead to better accuracy. They also show that the kind of data is not important for the speed of methods. But it has infuence on the accuracy. Clustered data have better values of accuracy metrics than nonclustered data. And the best accuracy is achieved when processing the real data.

The results show that MDS is the most accurate method, but not so fast as PCA or ICA. These are the fastest methods, but they showed moderate accuracy values. Principal curves and LLE showed the worst results. Isomap was significantly slower, but its accuracy in some cases can be the best.

6 References

[1] **Domeniconi.** Comparison of Principal Component Analysis and Random Projection in Text Mining. INFS 795, (2004)

[2] **I. K. Fodor.** A survey of dimension reduction techniques. *Center for Applied Scientific Computing*, Lawrence Livermore National Laboratory. (2002).

[3] **J. Hauke, J. Kossowski.** Comparison of values of pearson's and spearman's correlation coefficients on the same sets of data. *Quaestiones geographicae*, vol. 30(2). (2011)

[4] **J. Hausser, K. Strimmer.** Entropy inference and the James-Stein estimator, with application to nonlinear gene association networks. *Journal of Machine Learning Research*, vol. 10, p. 1469-1484. (2009).

[5] **H. Kim, P. Howland, H. Park.** Dimension Reduction in Text Classification with Support Vector Machines. *Journal of Machine Learning Research*, vol. 6, p. 37–53. (2005).

[6] **S. Kudyba.** Big Data, Mining, and Analytics–Components of Strategic Decision Making. *CRC Press Taylor & Francis Group an Auerbach Book.* ISBN 9781466568709. (2014).

[7] **A. K. Menon**. Random projections and applications to dimensionality reduction. School of Information Technologies, The University of Sydney.

[8] **M. Mizuta.** Dimension Reduction Methods. Humboldt-Universität Berlin, *Center for Applied Statistics and Economics (CASE)*, vol. 15. (2007).

[9] R package 'clusterGeneration' - Random Cluster Generation (with Specified Degree of Separation). 2015.

[10] R package 'entropy' - Estimation of Entropy, Mutual Information and Related Quantities.

[11] R package 'smacof' - Multidimensional Scaling. 2017.

[12] **R. S. Rosaria, I. Adae, A. Hart, M. Berthold**. Seven Techniques for Dimensionality Reduction. Knime. (2014).

[13] **C. O. S. Sorzano, J. Vargas, A. P. Montano.** A survey of dimensionality reduction techniques. Access: <arXiv:1403.2877>. (2014).

[14] Stock ratios. Access: < http://finviz.com/>.

[15] **J. Zubova, O. Kurasova, M. Liutvinavicius.** Parallel computing for dimensionality reduction. Information and Software Technologies, Springer-Verlag. p. 230-241, ISBN 978-3-319-46254-7. (2016).