Dimensionality reduction for financial data visualization

Jelena Zubova
Institute of mathematics and informatics,
Vilnius University,
Vilnius, Lithuania
e-mail: jelena.zubova@mii.vu.lt

Olga Kurasova
Institute of mathematics and informatics,
Vilnius University,
Vilnius, Lithuania
e-mail: olga.kurasova@mii.vu.lt

Marius Liutvinavičius
Kaunas faculty,
Vilnius University,
Kaunas, Lithuania
e-mail: marius.liutvinavicius@khf.vu.lt

***Abstract*—Various data mining methods are used for examining large financial data sets to uncover hidden and useful information. Ability to access big data sources raises new challenges related with capabilities to handle such enormous amounts of data. This research focuses on big financial data visualization that is based on dimensionality reduction methods. We use data set that contains financial ratios of stocks traded on NASDAQ stock exchange. A brief overview of the most popular dimensionality reduction and visualization methods is presented in this paper. We also show how to adjust the algorithms of these methods for parallel computing. The MPI technology is applied in computer cluster to perform dimensionality reduction. The results show that Random projection and Multidimensional scaling methods can effectively classify data and find the most promising stocks.**

Keywords—financial data; dimensionality reduction; visualization

I. Introduction

This template, We use big data analytics for examining large financial data sets to uncover hidden and useful information. Ability to access new data sources provides many opportunities, for example create new investors sentiment indexes from online social media streams. But it also raises new challenges related with capabilities to handle such enormous amounts of data. We need effective methods and powerful environments to complete large and complex tasks.

This research focuses on big financial data visualization that is based on dimensionality reduction methods. We use data set that contains financial ratios of stocks traded on NASDAQ stock exchange. Our goal is to find the most effective ways to analyze and visualize such type data. In the second section, we present a brief overview of the most popular dimensionality reduction and visualization methods. In the third section we present how to adjust the algorithms of these methods for parallel computing. We use MPI technology in computer cluster to perform dimensionality reduction. In the last section we present the classification and visualization results which where get while using Random projection and Multidimensional scaling methods. The current data set is relatively small, but the results give suggestions what techniques to choose for future works with big online data streams.

II. 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 the brief summary of most popular methods in order to choose best suitable in our case.

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 meant to capture only those dimensions that 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 it possesses [6].

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 is highly redundant and can be efficiently brought down to a much smaller number of variables without a significant loss of information.

Fig. 1. Dimensionality reduction methods

A. 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) [11]. PCA is by far one of the most popular algorithms for dimensionality reduction [5].

PCA find components that make projections uncorrelated by selecting the highest eigenvalues of the covariance matrix and maximizes retained variance [2]. 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 [6].

B. Random Projection

Random Projection finds components that make projections uncorrelated by multiplying by a random matrix and minimizes computations for a particular dimension size [2].

This method involves taking a high-dimensional data set and then mapping it into a lower-dimensional space, while providing some guarantees on the approximate preservation of distance. If the input data is an $n \times d$ matrix, then to do a projection, we choose a "suitable" $d \times k$ matrix R, and then define the projection of A to be $E = AR$, which stores k-dimensional approximations for our n points (matrix E is $n \times k$) [6]. R is a matrix with elements r_{ij}, where r_{ij} = random Gaussian. R can also be constructed in one of the following ways [2]:

$r_{ij} = \pm 1$ with probability of 0.5 each

$r_{ij} = \pm 1$ with probability of 1/6 each, or 0 with a probability of 2/3

Steve Vincent (2004) made comparison of principal component analysis and random projection in text mining. He found that in general Random projection is faster by many orders of magnitude over PCA, but in most cases produced lower accuracy. This lead to suggestion to use Random projection method if speed of processing is most important. [2]

C. Factor analysis

Like PCA, factor analysis is also a linear method, based on the second-order data summaries. Factor analysis assumes that the measured variables depend on some unknown, and often immeasurable, common factors. The goal of this method is to uncover such relations, and thus can be used to reduce the dimension of data sets following the factor model. [3]

Factors are assumed to follow a multivariate normal distribution, and to be uncorrelated to noise [5].

D. 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. In contrast with PCA, the goal of ICA is not necessarily dimension reduction. [3]

E. Maximum likelihood

This method specifies the likelihood of the noise-free ICA model, and uses the maximum likelihood principle to estimate the parameters. The advantages of this method include the asymptotic efficiency of maximum likelihood estimates under regularity conditions. However, it is computationally intensive, which make it undesirable in many practical situations. [3]

F. Vector quantization [3]

Probably the simplest way of reducing dimensionality is by assigning a class (among a total of K classes) to each one of the observations x_n. This can be seen as an extreme case of dimensionality reduction in which we go from M dimensions to 1 (the discrete class label K). Each class, K, has a representative x_K which is the average of all the observations assigned to that class. If a vector x_n has been assigned to the K_n-th class, then its approximation after the dimensionality reduction is simply $x_n = x_{Kn.}$ Widely used K-means method also belongs to this group of dimensionality reduction methods. [5]

G. Principal curves, surfaces and manifolds

PCA is the perfect tool to reduce data that in their original M-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. [5]

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. [11]

H. Projection pursuit

PCA is ineffective in analyzing nonlinear structures, i.e. curves, surfaces or clusters. In such cases Projection pursuit method might be a solution. The goal of projection pursuit is to find a projection that reveals interesting structures in the data. "Interesting" is defined as being "far from the normal distribution", i.e. the normal distribution is assumed to be the most uninteresting. The degree of "far from the normal distribution" is defined as being a projection index. There are various standards for interestingness. [11]

I. Multidimensional scaling

Given n items in a d-dimensional space and $n \times n$ matrix of proximity measures among the 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. [3]

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 Euclidian distance, the Manhattan distance and the maximum norm. [3]

J. Kohonen's self-organizing maps

Self-Organizing Maps (SOM) are generalizations of the vector quantization approaches presented above. SOM work by assigning to each input vector a label K_n corresponding to the class closest to its representative vector. Kohonen's SOMs start by creating a set of labels on a given manifold (usually a plane). Labels are distributed in a regular grid and the topological neighborhood is defined as the neighbors in the plane of each point of the grid. [5]

K. *Support Vector machines*

Support vector machines (SVMs) have been recognized as one of the most successful classification methods [8]. SVMs are a powerful machine learning technique not only for classification, but also for regression. It works by finding a hyperplane that best splits the target values. SVM performs classification by finding the hyperplane that maximizes the margin between the two classes. The vectors (cases) that define the hyperplane are the support vectors. The margin is the largest distance between the borderline data points. SVMs handle nonlinearity by mapping the data into a higher-dimensional space using kernel tricks. [9]

L. *Other methods*

Important comparison of different methods was made by R. S. Rosaria, I. Adae et. al. (2014). They concentrated on a few state-of-the-art methods to reduce input dimensionality and examined how they might affect the final classification accuracy. In particular, they tried removing data columns with too many missing values and used low variance filter that calculates each column variance and removes those columns with a variance value below a given threshold. They also reduced highly correlated columns and implemented dimensionality reduction via tree ensembles (Random forests). But the best results were got by using backward feature elimination and forward feature construction together with removing data columns with too many missing.

The Backward Feature Elimination loop performs dimensionality reduction against a particular machine learning algorithm. Similarly to the Backward Feature Elimination approach, a Forward Feature Construction loop builds a number of pre-selected classifiers using an incremental number of input features. The Forward Feature Construction loop starts from 1 feature and adds one more feature at a time in the subsequent iterations. [12]

III. Algorithm Adoption for Parallel Computing

In this section we briefly describe the MPI technology that is used for parallel computing in computer cluster. We also present the algorithm of selected dimensionality reduction method which is adapted for MPI.

A. *MPI*

MPI is message-passing library interface specification. MPI addresses primarily the message-passing parallel programming model, in which data is moved from the address space of one process to that of another process through cooperative operations on each process. [7]

Figure 2 shows typical structure of multi-core cluster. In the pure MPI case there is one MPI proccess per core. H. Jin, D. Jespersen (2011) stated that the increasing number of cores in modern microprocessors is pushing the current high performance computing systems into the petascale and exascale era.

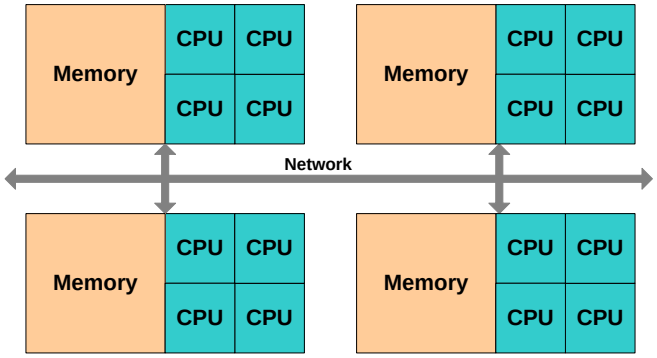

Fig. 2. MPI scheme

According to R. Rabenseifner, G. Hager, etc. (2009), most systems in high-performance computing feature a hierarchical hardware design: shared memory nodes with several multi-core CPUs are connected via a network infrastructure.

The comparison between MPI and hybrid models was made by F. Cappello, D. Etiemble (2000). Their test results showed that a unified MPI approach is better for most of the benchmarks.

Rifat Chowdhury (2010) designed an algorithm using the libraries of OpenMP to compute Matrix Multiplication efficiently. The speedup was close to the ideal 4 times using four cores of a processor to compute the final matrix instead of one. This is an example how parallel computing can increase the performance.

B. *Random Projection method*

Our goal is to combine speed and accuracy. For initial data dimensionality reduction we selected Random projection method because it is very fast. Moreover we adapted this method for parallel computing. This method can be applied for many purposes. For example, G. E. Dahl, J. W. Stokes et. al. successfully used Random projections for large-scale malware classification to further reduce the dimensionality of the original input space.

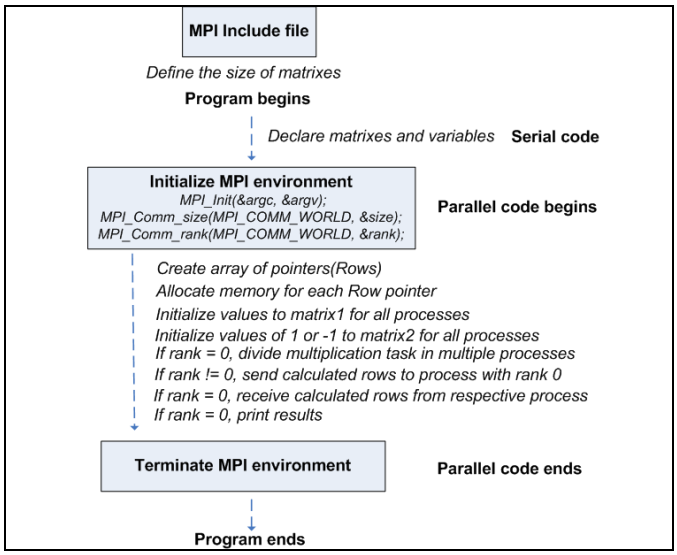

Fig. 3. Random projection algorithm for parallelization with MPI

E. Bingham and H. Mannila used Random projection as dimensionality reduction method tool in cases of processing both noisy and noiseless images, and information retrieval in text documents. In our case we use this method for financial data mining.

To use Random projection method, we firstly declare 3 matrixes: $n \times d$ matrix1 (contains initial financial data), $d \times k$ matrix2 (operating matrix) and $n \times k$ matrix3, which is our results matrix. During the next step we initialize random integer values to all elements of matrix1. Then we assign values of *1* or *-1* to matrix2 with probability 0,5 each. Then matrix1 and matrix2 are multiplied and results are assigned to the elements of matrix3.

The code is constructed accordingly to MPI requirements (Fig. 3). Program includes MPI library. All matrices and variables are declared in serial code region, but MPI environment is initialized by special functions before parallel region begins. The program uses determined amount of threads ("ranks"). "Zero" thread is used to divide multiplication task into multiple processes and handle incoming results from the remaining threads. All computing operations are simultaneously done by multiple threads.

IV. Research Results

In this section we present the results of applying Random projection and Multidimensional scaling methods for our financial market data set.

The data set contains information about 1400 companies that are traded on NASDAQ stock exchange. Every company is described by 54 different parameters. All these parameters are grouped into 6 categories: overview, valuation, financial, performance, technical and ownership. All data is from finviz.com website.

In the *Overview* group there are such parameters as sector, industry, market capitalization, price, volume. In *Valuation* group we have P/E, PEG, P/B. EPS and sales parameters. *Financial* parameters *group* contains ROA, ROE, ROI, earnings, debt and margins parameters. *Performance* indicators present price changes during different periods of time, volatility and recommendation values.

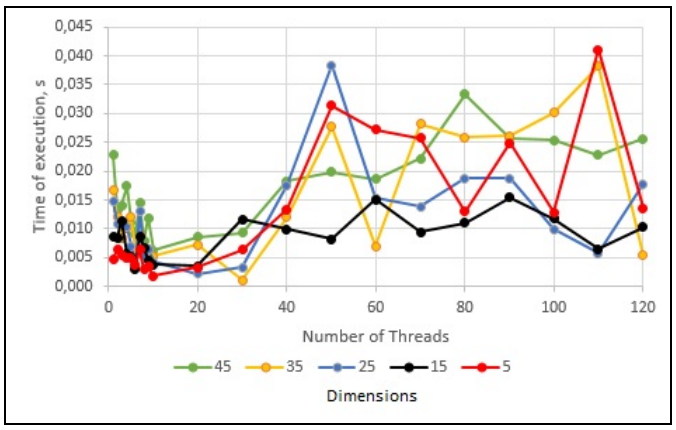

Fig. 4. The execution times of Random projection algorithm

Technical indicators are ATR, Beta, Simple moving averages of different periods of time, RSI, the lowest and highest prices at which a stock has traded in the previous 52 weeks. And the last group contains parameters of stock ownership.

Usually our goal is to visualize data and uncover hidden information. This means that for example in this case we have to reduce the dimensions from 54 to 2 or 3. It's complicated to quickly perform such tasks with single machine. So in the first step we executed Random projection algorithm in the cluster having 120 nodes to reduce the dimensions from initial 54 to smaller amount: 25, 15, 10 and 5 (got several new data sets). When the most suitable number of threads were chosen it took less than 0,04 seconds to complete each task (Figure 4). It should be noted that using more threads not always leads to better performance. As this data set was relative small, 10 to 30 threads was optimal choice to execute the algorithm in the fastest way. In comparison with serial code, MPI code worked from 10 to 20 times faster.

In the second step we applied Multidimensional scaling method (with Sammon) to visualize the reduced data set containing 5 variables. We investigated two stock classification cases: based on analysts' recommendation and based on multicriteria indicator.

	Overview										
	No.	Ticker	Company	Sector	Industry	Country	Market Cap	Price	Change	Volume	
Valuation											
P/E	Fwd P/E	PEG	P/S	P/B	P/C	P/FCF	EPS this Y	EPS next Y	EPS past 5Y	EPS next 5Y	Sales past 5Y
Financial											
Dividend	ROA	ROE	ROI	Curr R	Quick R	LTDebt/ Eq	Debt/Eq	Gross M	Oper M	Profit M	Earnings
	Performance										
	Perf Week	Perf Month	Perf Quart	Perf Half	Perf Year	Perf YTD	Volatility W	Volatility M	Recom	Rel Volume	
	Technical										
	Beta	ATR	SMA20	SMA50	SMA200	52W High	52W Low	RSI	from Open	Gap	
		Ownership									
		Outstanding	Float	Insider Own	Insider Trans	Inst Own	Inst Trans	Float Short	Short Ratio		

Fig. 5. Parameters of analysed companies

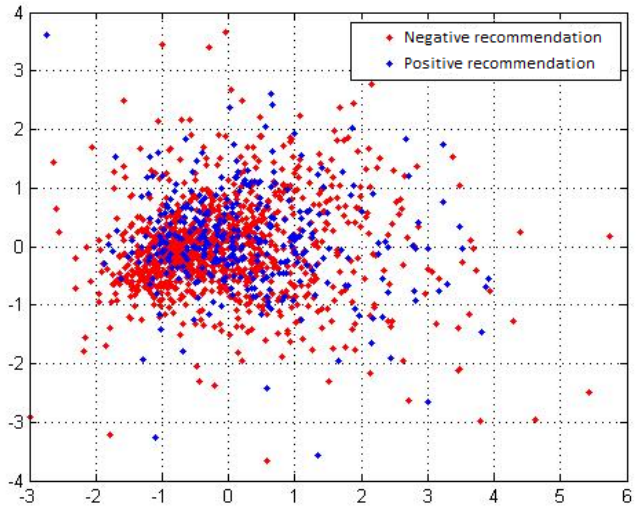

Fig. 6. Data visualization using Multidimensional scaling

A. *Stock classification based on analysts' recommendation*

One of our initial 54 factors was the recommendation of financial analysts. This variable ranges from *1* (strong recommendation to buy stock) to *5* (strong recommendation to sell stock). We raised hypothesis that "good" companies (having recommendation ratio from *1* to *2*) should be separated from the rest companies after dimensionality reduction and visualization processes.

However, figure 6 shows that the combination of Random projection and Multidimensional scaling methods couldn't separate these two classes. They just overlapped each other. We tried to use 3D plot, but it was uninformative too (Fig. 7).

The first thought could be that the methods are not efficient enough. But it also could be that in reality the potentiality of "good" and "bad" companies (as determined by analysts) don't differ so much. This would explain why using just the opinion of financial advisors often leads to unstable returns or even losses. It also suggests that opinions might be more based on intuition than various stock ratios.

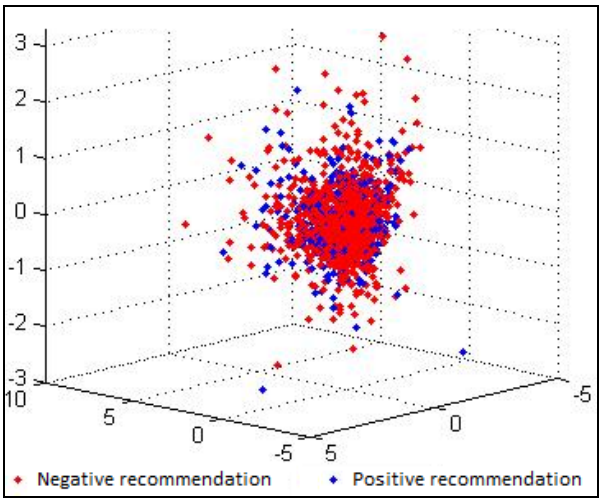

Fig. 7. Data visualization using Multidimensional scaling (3D plot)

TABLE I. MULTICRITERIA INDICATOR FOR STOCK CLASSIFICATION

	P/E	EPS this Year %	ROE	SMA50 %	RSI
„Good stocks"	< 20	> 5	> 5	> 0	> 50
„Bad stocks"	> 20	< 5	< 5	< 0	< 50

B. *Stock classification based on multicriteria indicator*

In the second case we constructed multicriteria indicator for separating stocks that are worth investing ("good"), neutral stocks and "bad" stocks.

In our assumption "good" stocks meet all following criteria: *P/E < 20*, *EPS this year > 5*, *ROE > 5*, *50-period SMA > 0*, *RSI > 50*. Table 1 shows the ratios by which "good" and "bad" classes of stocks are constructed. "Neutral" stocks are those which don't fall within any of previously defined two classes.

We used the same Multidimensional scaling method to visualize the reduced data set, so the arrangement of data points is the same as presented in figure 6. But in this case we found that the three different classes of stocks were separated quite well.

Figure 8 shows that "good" companies were presented in the left side of the plot. The "bad" and "neutral" companies overlapped, but "bad" companies separated from "good" companies.

This leads to suggestion, that dimensionality reduction and visualization methods can effectively classify data and find the most promising stocks. However, in order to explain the differences between classes we need to use several different ratios.

We used the combination of Multidimensional scaling and Random projection methods because the latter one is very fast. However it might be not so accurate. So we tried to apply MDS method for full initial data set and it actually led to better results. 3D plot in figure 9 shows that "bad" stocks were separated from "neutral" stocks.

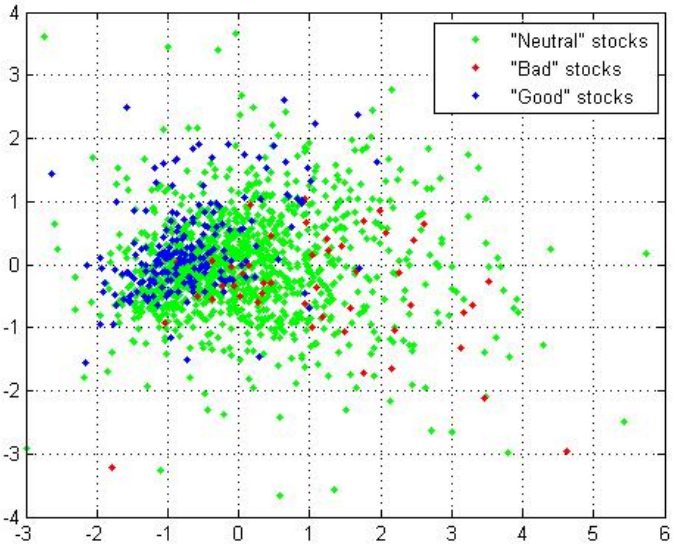

Fig. 8. Visualizing three classes of stocks

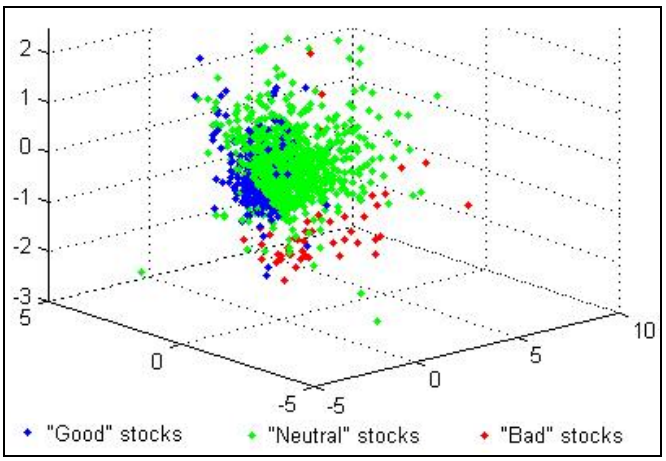

Fig. 9. Visualizing initial data set using MDS

V. CONCLUSIONS

Various data mining methods are used for examining large financial data sets to uncover hidden and useful information. This research focused on financial data visualization that is based on dimensionality reduction methods. We used data set that contained financial ratios of stocks traded on NASDAQ stock exchange. A brief overview of the most popular data dimensionality reduction and visualization methods was presented in this paper. We also showed how to adjust the algorithm of Random projection method for parallel computing. The MPI technology was applied in computer cluster to perform dimensionality reduction. The performance results revealed the advantages of parallel computing. Our goal was to visualize data and uncover hidden information. In order to do this we had to reduce the dimensions to 2 or 3. In the first step we executed Random projection algorithm in the cluster to reduce the dimensions from initial 54 to smaller amount. In the second step we applied Multidimensional scaling method to visualize the reduced data set.

One of the data set ratios was the recommendation of financial analysts. We raised hypothesis that companies having recommendation to buy them should be separated from the rest of companies. But the results showed that the combination of Random projection and Multidimensional scaling methods couldn't do this. This might have happened because in reality the potentiality of "good" and "bad" companies (as they are determined by analysts' recommendations) doesn't differ so much. However, in the second case of stock classification based on multicriteria indicator "good" and "bad" stocks were separated quite well. This leads to suggestion, that dimensionality reduction and visualization methods can effectively classify data and find the most promising stocks. But in order to explain the differences between classes we need to use several different ratios. It should be also noted, that MDS method alone was more accurate than combination of two methods.

REFERENCES

[1] R. Chowdhury, "Parallel Computing with OpenMP to solve matrix Multiplication," UCONN BIOGRID REU Summer 2010. Department of Computer Science & Engineering. University of Connecticut, Storrs, CT 06269.

[2] Dr. Domeniconi, "Comparison of Principal Component Analysis and Random Projection in Text Mining," April 29, 2004. INFS 795.

[3] I. K. Fodor, "A survey of dimension reduction techniques," Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, June 2002

[4] Intel Parallel studio. Access: https://software.intel.com/en-us/intel-parallel-studio-xe

[5] C. O.S. Sorzano, J. Vargas, A. Pascual Montano, "A survey of dimensionality reduction techniques," 2014. Access: arXiv:1403.2877.

[6] A. K. Menon. Random projections and applications to dimensionality reduction. School of Information Technologies, The University of Sydney, 2007.

[7] MPI technology. Access: http://www.mpi-forum.org/docs/mpi-3.1/mpi31-report.pdf

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

[9] S. Kudyba. Big Data, Mining, and Analytics–Components of Strategic Decision Making. March 12, 2014 by Auerbach Publications. Reference - 325 Pages - 89 B/W Illustrations. ISBN 9781466568709 - CAT# K16400

[10] Message passing interface. Access: https://computing.llnl.gov/tutorials/mpi/

[11] M. Mizuta, "Dimension Reduction Methods, Papers," Humboldt-Universität Berlin, Center for Applied Statistics and Economics (CASE), 15, 2004.

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

[13] H. Jin, D. Jespersen etc. High performance computing using MPI and OpenMP on multi-core parallel systems. Parallel Computing 37 (2011) 562–575

[14] R. Rabenseifner, G. Hager, G. Jost, "Hybrid MPI/OpenMP Parallel Programming on Clusters of Multi-Core SMP Nodes," Conference: Proceedings of the 17th Euromicro International Conference on Parallel, Distributed and Network-Based Processing, PDP 2009, Weimar, Germany, 18-20 Febuary 2009

[15] F. Cappello, D. Etiemble, "MPI versus MPI+OpenMP on the IBM SP for the NAS Benchmarks," Supercomputing, ACM/IEEE 2000 Conference. ISSN: 1063-9535.

[16] G. E. Dahl, J. W. Stokes, L. Deng, D. Yu, "Large-scale malware classification using random projections and neural network," Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference, pp. 3422-3426, 2013.

[17] E. Bingham, H. Mannila, "Random projection in dimensionality reduction: Applications to image and text data," Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 245-250, 2001.