Faculty of Economics, University of Niš, 18 October 2012

International Scientific Conference
SERBIA AND THE EUROPEAN UNION

USING ORACLE DATA MINER FOR FINANCIAL TIME SERIES FORECASTING

Jovica Stanković, MSc[*]

Jelena Stanković, MSc[*]

Ognjen Radović, PhD[*]

Abstract: *In recent years, Support Vector Machine (SVM), a novel artificial neural network (ANN) technique, has been successfully used as a nonparametric tool for regression estimation and forecasting time series data. In this paper, authors use Support Vector Regression (SVR) tool Oracle DM in forecasting volatility of Belex 15 index and estimation of Value-at-Risk (VaR). Two SVR models are implemented: the first one based on an estimation of the absolute value of returns, and the second based on an estimation of 5-day standard deviation. These models are compared with the standard GARCH (1,1) method for volatility estimation. Data that are used for calculation in this study are historical data of yields on Belex15 index from April 2005 to December 2011. The results show that the SVR tools give better estimates of VaR comparing to GARCH (1,1) method. However, authors recommend daily updating of the model and adjusting parameters of SVR in a high quality manner.*

Keywords: *Support Vector Regression (SVR), Oracle Data Miner, financial time series*

1. Introduction

The classical financial theory, based on the random walk process, approximates and simplifies the dynamics of the financial markets. Nevertheless, the micro simulations of the market dynamics, represented by agent-based models, provide a more realistic picture. These simulations require significant computing resources and generate complex models, whose behavior can become just as complex as the financial market. Therefore, the technical analysis, is still significantly used in professional purposes, although its' implementation is disputed in academic circles. The basis of this analysis is the study of the movement of technical indicators based on historical trends of financial time series. The analysis of historical series' trends of yield and volatility of Belex 15 index represents a starting point for predicting the future movement of this index.

[*] University of Niš, Faculty of Economics; e-mail: jovica.stankovic@eknfak.ni.ac.rs, jelenas@eknfak.ni.ac.rs, ognjen.radovic@eknfak.ni.ac.rs

UDC 519.246.8:004

During the last few years the most used models are the ones that apply data mining tools to predict the yield and volatility movements, or in general, financial time series modeling. The most commonly implemented models use artificial neural networks (ANN). The advantage of their usage lies in their universal approximation capabilities any non-linear function arbitrarily well without a priori assumptions about the properties of the data (Hornik et al., 1989). Today, models based on support vector machine (SVM) algorithm are becoming increasingly popular. SVM is a special class of ANN algorithms developed by Vapnik (Vapnik, 1995, Vapnik et al., 1997). The advantage of SVM over traditional ANN is that SVM implements the structural risk minimization principle, while ANN minimizes the error on the in-sample data estimating. Therefore, SVM often give better results in predicting comparing to ANN (Tay and Lijuan, 2001; Sansom et al., 2003, Huang et al., 2004, Shin et al., 2005, Huang et al., 2005, Chen et al ., 2006). SVM is originally designed for classification problems; however, it is successfully used for regression problems. Gavrishchaka and Ganguli (2003) used SVM to predict volatility in foreign-exchange data, and show that the SVM gives better results than the GARCH model.

In this paper, we will show the application of ODM based on SVR tool. We will use this tool to forecast future volatility of Belex15 index. The results of the prediction will be the basis for the assessment of risk values of VaR parametric model. The constructed models will be compared with the standard parametric VaR based on GARCH volatility forecasting models.

This paper is organized as follows: Part 2 provides an introduction to the tools and the Oracle Data Miner theory of SVR. In Part 3 will be shown a statistical analysis of time series Belex 15 index. The research methodology based on SVR and implemented to estimate VaR will be described in Part 4. In Part 5 the results of research will be presented and in Part 6 conclusions.

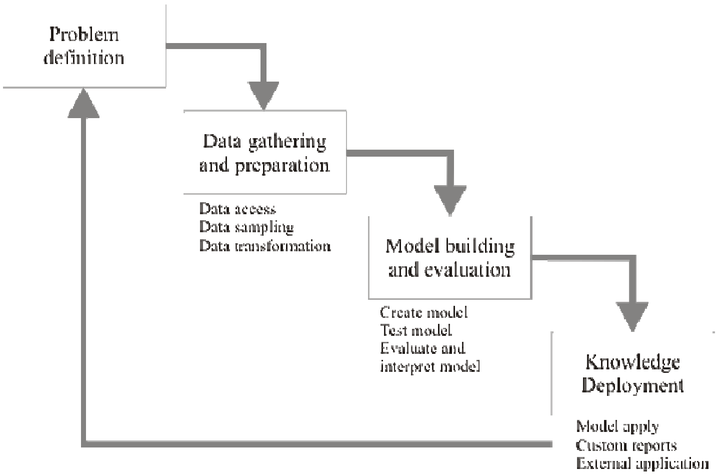
2. Oracle Data Miner and SVM

Data mining (DM) uses sophisticated mathematical algorithms for discovering data patterns and predicting future events, based on large data sets. Results of the DM process are models which are used for mining new data, or probability of outcomes and rules – conditions needed and probabilities of their outcome. Data mining is also known as Knowledge Discovery in Data (KDD).

Data mining provides answers to complex questions when results provided by data analysis, query and reporting techniques are not good enough. DM algorithms often require large data sets for the creation of quality models and discovering patterns, hidden information and relationships within data. Data mining algorithms are very sensitive to data quality problems such as missing values, outliers (data values that are very different from the typical values in data set), irrelevant columns, data coding, and data that is included or excluded from data set. Data mining process demands highly trained specialist who must understand the problem and data used for building the model in order to properly interpret the results when the model is applied.

Figure 1 illustrates iterative nature of DM project and its phases. The process flow shows that a data mining project does not stop when a particular solution is deployed. The results of data mining trigger new business questions, which in turn can be used to develop more focused models.

Figure 1 The Data Mining Process source



Source : Oracle Data Mining Concepts 11g Release 2

First phase of a data mining project focuses on defining DM problem. Next step is data gathering and preparation, removing some of the data or adding additional data, so as needed data transformations. Data quality problems, such as missing or wrong data values, are also solved in this phase. Good data preparation can significantly improve results and the information that can be discovered through data mining. Third phase is model building and evaluation, choosing algorithm and adjusting parameters to optimal values. If the algorithm requires additional data transformations than some steps from phase 2 are repeated. DM process is time and resource consuming, so usual practice is to work with a reduced set of data – training data. Stage 3 also evaluates how well the model satisfies the originally-stated business goal. If the results are not good enough, some steps of previous phases are repeated. Last phase is knowledge deployment which can involve applying of models to new data or the integration of DM models within information or business intelligence systems, such as data warehouse infrastructure, query and reporting tools.

2.1 Oracle Data Mining

Oracle Data Mining is implemented as part of Oracle Database, which maximizes scalability and makes efficient use of system resources. Implementation within Oracle Database offers many advantages such as increased security, reduced possible errors caused by data movement from original data bases, and easier data preparation and administration. Since data is in the same data base, data mining process can access current and relevant data. Oracle Database offers many features for advanced statistical analysis, Online Analytic Processing (OLAP) and business intelligence. (Stackowiak, R., Rayman, J., Greenwald, R. 2007)

DM function specifies a class of problems that can be modeled and solved. Oracle data mining supports both supervised and unsupervised data mining. (www.oracle.com, Oracle Data Mining Concepts 11g Release 2)

In the supervised data mining, also known as direct learning, learning process is directed by a previously known dependent attribute, and it explains the behavior of the target as a function of a set of independent attributes or predictors. Results of supervised learning are predictive models. This is in contrast to unsupervised learning where the goal is pattern detection. Table 1. presents typical DM supervised and unsupervised functions.

First phase of supervised model building is training when the software analyzes test data with known target value and "learns" the logic for making the prediction. Second phase of the model building is testing on separate data sets that were not used for building a model. This phase determines whether the model is generalized enough. Last phase is scoring which means applying built model to actual data set. Scoring is the purpose of classification and regression.

Unsupervised learning is non-directed; there is no previously-known result to guide the algorithm in building the model. There is no distinction between dependent and independent attributes. Unsupervised learning is generally used for descriptive purposes. Typical examples of unsupervised learning are: clustering, anomaly detection (detecting atypical values or events), feature extraction, and association rules (relations between data or events).

Table 1. Supervised and unsupervised DM functions

	Function	**Description**
Supervised Functions	Attribute Importance	Identifies the attributes that are most important in predicting a target attribute
	Classification	Assigns items to discrete classes and predicts the class to which an item belongs
	Regression	Approximates and forecasts continuous values
Unsupervised Functions	Anomaly Detection	Identifies items (outliers) that do not satisfy the characteristics of "normal" data, implemented through one-class classification
	Association Rules	Finds items that tend to co-occur in the data and specifies the rules that govern their co-occurrence
	Clustering	Finds natural groupings in the data
	Feature Extraction	Creates new attributes (features) using linear combinations of the original attribute

Source : www.oracle.com

2.2 Data Mining Algorithms Supported by Oracle Data Mining

Oracle Data Mining supports at least one algorithm for each data mining function. Supported algorithms are listed in the Table 2. (www.oracle.com, Oracle Data Mining Concepts 11g Release 2). For some data mining functions several algorithms are supported, so there is possibility to choose the most appropriate algorithm for concrete problem. Choosing appropriate algorithm demands several models building, and choosing algorithm parameters for best results. It is time and resource consuming process, and demands great expertise in data mining and good understanding of the problem being solved. Graphical user interface offers excellent support in all phases of model building and testing.

Table 2. Data mining algorithms supported by Oracle data mining

	Algorithm	**Function**
Supervised Functions	Decision Tree (DT)	Classification
	Generalized Linear Models (GLM)	Classification and Regression
	Minimum Description Length (MDL)	Attribute Importance
	Naive Bayes (NB)	Classification
	Support Vector Machine (SVM)	Classification and Regression
Unsupervised Functions	Apriori (AP)	Association
	k-Means (KM)	Clustering
	Non-Negative Matrix Factorization (NMF)	Feature Extraction
	One Class Support Vector Machine (One-Class SVM)	Anomaly Detection
	Orthogonal Partitioning Clustering (O-Cluster or OC)	Clustering

Source : www.oracle.com

2.3 Support Vector Machines (SVM)

Support Vector Machines (SVM) is effective and promising techniques for DM. Some recent studies showed that SVM achieves better results comparing to traditional algorithms like decision trees, neural networks, and Bayesian approaches.

The theoretical foundations of SVM are based on the Vapnik-Chervonenkis (VC) theory (Vapnik, 1995, Vapnik et al., 1997). While traditional neural network models had implemented the empirical risk minimization principle, SVM implements the structural risk minimization principle. In SVR, an upper bound on the generalization error is minimized as opposed to the ERM, which minimizes the prediction error on the training data. Thus, over-fitting is unlikely during the training phase, enabling better predictions and generalization of the model to new data. In the model build (training) process, a regression algorithm estimates the value of the target as a function of the predictors for each case in the build data. These relationships between predictors and target are summarized in a model, which can then be applied to a different data set in which the target values are unknown.

The SVM model learns a linear regression function. In the linear kernel case the linear function is defined in the original input space. In the non-linear kernel case, input vectors are transformed into the high-dimensional space and then linear separable classification problem in solved in a new space by constructing optimal separating hyper-plane. Linear model is constructed in the new space that represents a nonlinear decision boundary in the original space. (Milenova, B., Yarmus, J., Campos, M., 2008)

SVM hinges on the extraction of a subset of the training data that serves as support vectors and that represents a stable characteristic of the data.

SVM regression uses an ε-insensitive loss function, that is, the loss function ignores values that are within a certain distance from the true value. The epsilon parameter (ε) in SVM regression represents the width of the insensitivity tube around the actual target value. Predictions falling within ε distance of the true target are not penalized as errors (Milenova, B., Yarmus, J., Campos, M., (2008)).

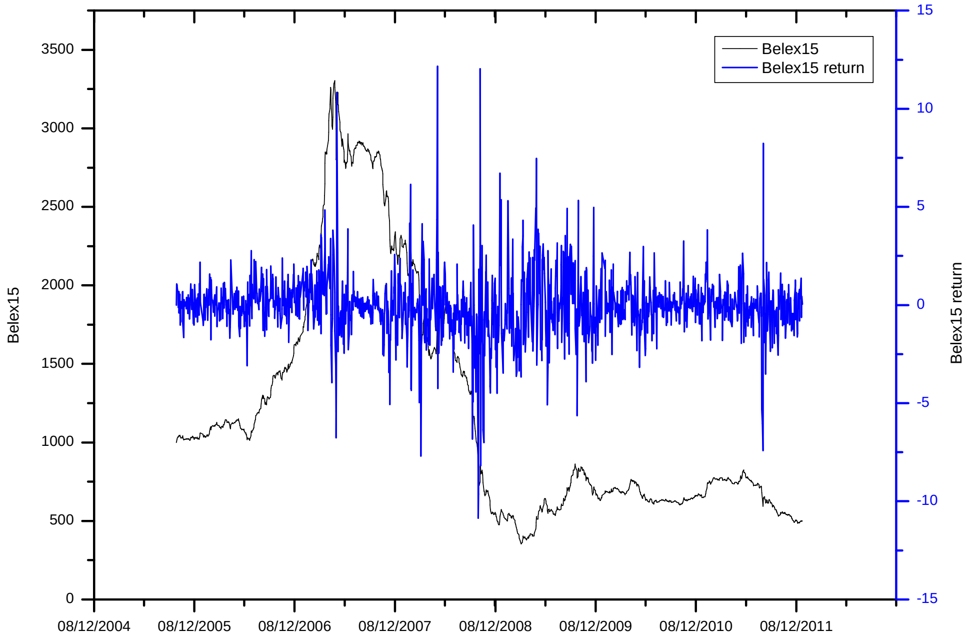
3. Data and Statistical Analysis

3.1 Research Data

In this analysis we used stock index Belex 15 and financial time series are obtained from a historical dataset provided by Belgrade Stock Exchange. The total number of dataset is 1575 observations, from April 2005 to December 2011.

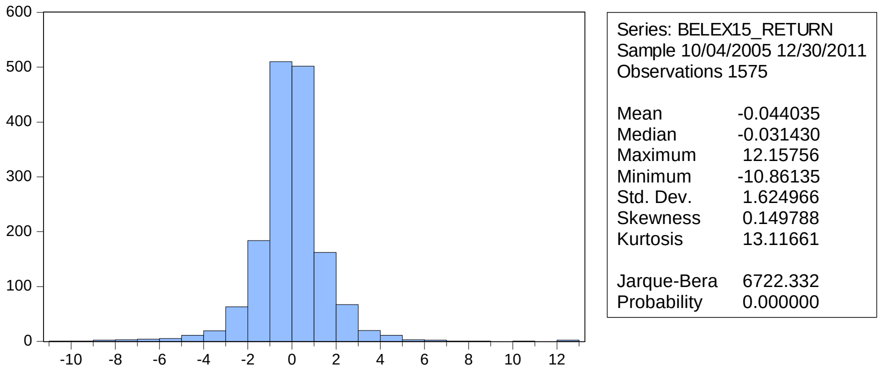
As it can be seen from the Figure 2., which shows the value and log-normal return, Belex 15 has significant trend oscillations during the observed period. The maximum value of over 3000 RSD Belex 15 reached in 2006 and the minimum value of less than 500 RSD it reached during the period of financial crises in 2008 and stayed at very low level until now.

Figure 2 Price and return of Belex 15 index (01.4.2005. - 31.12.2011.)



The summary statistics of the Belex 15 return series reveal a significant kurtosis of the distribution and slight skewness, what indicates that distribution of this serial is not normal. The results of Jarque-Bera tests confirm that the observed return serial is non-normal at the 99% confidence interval.

Figure 3 Descriptive statistics of logarithmic returns of BELEX 15 index



```
Series: BELEX15_RETURN
Sample 10/04/2005 12/30/2011
Observations 1575

Mean        -0.044035
Median      -0.031430
Maximum      12.15756
Minimum     -10.86135
Std. Dev.    1.624966
Skewness     0.149788
Kurtosis     13.11661

Jarque-Bera  6722.332
Probability  0.000000
```

The results obtained by the ARCH test of volatility of normalized logarithmic returns of selected financial time series (Table 3) indicate return autocorrelation and heteroscedasticity, what makes these series inadequate for the implementation of many VaR models. These characteristics are, however, common to many emerging markets (Žiković, 2007).

Table 3. The results of ARCH volatility tests of normalized logarithmic returns of BELEX 15 index (lag = 10*)*

	ARCH (CV=18.3070)	
	Test statistics	p-value
Belex15	268.4089	0.00000

Previously conducted tests on dataset of the returns on BELEX15 index showed the following:

- Distribution of log-returns is generally moderately skewed with high peak and „fat tails“; because of what it can not be considered normal;
- There is positive serial correlation of returns and heteroscedasticity, as evidenced by ARCH test.

The transformation of the empirical returns of the observed market portfolio to meet the assumption of independent and identically distributed returns (IID) is obtained by ARMA-GARCH model. ARMA-GARCH model successfully captured the dynamics of stock indexes from EU new member and candidate states and produced standardised innovations that proved to be independently and identically distributed (Žiković, 2007). Regarding the observed volatility modelling, the achieved results show that GARCH representation will be necessary to capture adequately the dynamics of data generating processes of analysed indexes.

Observed dataset is divided into two segments: in-sample dataset of 1325 observations – from April 2005 to June 2011, and out-sample data set of 250 observations – from Jun to December 2011. GARCH coefficients of the out-sample data set are presented in Table 4.

Table 4. GARCH (1,1) coefficients of the BELEX 15 index during the period from 10.01.2011. to 30.12.2011.

Dependent Variable: BELEX15_OUT_SAMPLE
Method: ML - ARCH (Marquardt) - Normal distribution
Sample: 1/10/2011 12/30/2011
Included observations: 250
GARCH = C(2) + C(3)*RESID(-1)^2 + C(4)*GARCH(-1)

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	-0.151281	0.072589	-2.084063	0.0372
Variance Equation				
C	0.181451	0.063701	2.848503	0.0044
RESID(-1)^2	0.180913	0.046722	3.872101	0.0001
GARCH(-1)	0.665615	0.084817	7.847670	0.0000

Various VaR methods are applied on in-sample dataset, while the out-sample data set is used for back testing the achieved results. In the previous research (Radović, Marinković, Stanković, 2011) we have shown that VaR model based on GARCH volatility modelling gives better results in comparison to Historical simulation and Monte Carlo simulation VaR models.

3.2 Research Methodology

Value-at-risk (VaR) is a rather simple method for assessing and controlling market risk. Assuming that the structure of the portfolio remains the same over the chosen time horizon, this method calculates the maximum losses caused by the changes in selected market variables for a given confidence level. VaR is suggested as a measure of market risk for the financial institutions, although it can be used to evaluate many other types of risks. In this paper, authors will assess market risk using analytical VaR (using GARCH (1,1) Risk Metrics VaR) and compare its effectiveness with Oracle SVM tools.

Analytical VaR is also known as parametric or variance-covariance method because its fundamental assumptions are that returns on risk factors are normally distributed, the correlations between risk factors are constant and the delta (or price sensitivity to changes in a risk factor) of each portfolio constituent is constant. (Choudhry, 2006)

Let r_t represents the daily logarithmic return of the stock index in the holding period *t*. If returns are normally distributed in the specific period of time *t*, the dynamics of r_t can be expressed as follows:

$$r_t = \mu_t + \sigma_t Z_\alpha \tag{1}$$

where μ_t stands for mean, σ_t for standard deviation and Z_α is the lower α percentile of the standard normal distribution, which is IID with zero mean, unit variance and marginal distribution function $F_z(z)$. By assuming normal distribution, a simple analytic formula of VaR can be derived:

$$VaR_{t,\alpha} = -Z_{\alpha}\sigma_t + \mu_t \tag{2}$$

Since the authors try to estimate the risk on the typical emerging financial market, results of statistical analysis, that show that distribution of returns of BELEX15 do not follow normal Gaussian distribution, have to be considered and particular model for the dynamics of mean and volatility has to be selected. Many different models for volatility dynamics have been proposed in econometric literature including models from the ARCH/GARCH family (Bollerslev, 1992). In this paper we used GARCH (1,1) to estimate conditional mean and variance of the returns on stock index assuming that distribution is normal. Assumed GARCH (1,1) process can be expressed as follows:

$$r_t = \mu + \varepsilon_t \, , \; \varepsilon_t = \sqrt{h_t Z_t} \tag{3}$$

$$h_t = \omega + \alpha\varepsilon_{t-1}^2 + \beta h_{t-1} \tag{4}$$

where used symbols present following variables: r_t - return at time t, μ - mean return, residuals (error terms), $z_t \sim iid\ N(0,1)$ - normally distributed random variable, and ω, α, β - parameters of the model.

Rather than using analytical VaR, the authors use the same GARCH (1,1) model but, instead of setting prespecified parameter values, the authors estimate them (Table 4.).

The application of SVR is based on the assumption that volatility can be represented as a non-linear function of returns' time series, absolute values of returns and past estimations of volatility. Concretely, volatility can be calculated using following formula:

$$\sigma_{t+1} = f\left(r_t, |r_t|, |r_{t-1}|, r_t^2, r_{t-1}^2, \sigma_{10}, \sigma_{25}, \sigma_{50}, \sigma_{250}\right) \tag{5}$$

where used symbols are: r_t – return, $|r_t|$ - absolute return, r_t^2 – squared return, σ – 10-, 25-, 50-, 250-day average standard deviation.

We used this approach to generate the SVR volatility forecasting model as a model for forecasting volatility of 5-day average standard deviation of the following form:

$$SVM_{std5} = \sigma_{5,t+1} = f\left(r_t, |r_t|, |r_{t-1}|, r_t^2, r_{t-1}^2, \sigma_{10}, \sigma_{25}, \sigma_{50}, \sigma_{250}\right) \tag{6}$$

Another approach and the corresponding model assume that the volatility or standard deviation can be estimated on the basis of the absolute return $\sigma_i \approx |r_i|$. In this case, the model can be represented in the following form:

$$SVM_{abs} = |r_{t+1}| = f\left(r_t, |r_t|, |r_{t-1}|, r_t^2, r_{t-1}^2, \sigma_{10}, \sigma_{25}, \sigma_{50}, \sigma_{250}\right) \tag{7}$$

All models that we have used in the experiment, ARMA (1,1)-GARCH (1,1)-RM, Oracle SVM Abs RM VaR and Oracle SVM Std5 RM VaR, have been first trained on the in-sample dataset (1325 observations - from April 2005 to June 2011), and tested on the out-sample data set (250 observations - from June to December 2011). The models implemented on the in-sample dataset are fixed and used without modification in forecasting on the out-sample data set. The exception to this approach is the ARMA (1,1)-

GARCH (1,1)-RM-OL model, which is constantly modified in the out-sample data set. This model simulates the daily updating of the GARCH model as a basis for assessing the volatility (and VaR) for the next day.

Considering the SVM (SVR) model, in our experiment we have defined no optimal model parameters, such as C and kernel parameters. Our models have taken the standard values for the parameters set by Oracle DM. We have used Gaussian kernel function and system determined values for tolerance, epsilon value, and standard deviation. Usually, these parameters are defined by deeper (and more complex) analysis of the problem, and often parameters values are determined in the iterative process of building and testing models. Since the values of these parameters can significantly affect the quality of the predictions, the SVR models need further improvement.

3.2 Results and Discussion

Different methods for VaR calculation are applied to selected financial dataset in order to compare results and get a conclusion about performance of SVM and GARCH methods. The results on average VaR values at 90, 95 and 99 percents confidence level are presented in table 5. At all confidence level Oracle SVM-Std5 RM gives the lowest value of the average VaR, but at 90% and 95% confidence level ARMA(1,1)-GARCH(1,1)-RM-OL gives a slightly better result then this method. In general, results achieved by GARCH method that is updated everyday are better than the ones achieved by SVM methods.

Table 5. Average VaR values at 90, 95 and 99% confidence levels, for VaR models.

	Average VaR		
	90%	95%	99%
ARMA(1,1)-GARCH(1,1)-RM	-1.61627%	-2.04407%	-2.84654%
Oracle SVM-Abs RM VaR	-1.47253%	-1.85955%	-2.58554%
Oracle SVM-Std5 RM VaR	-1.43938%	-1.81700%	-2.52535%
ARMA(1,1)-GARCH(1,1)-RM-OL	-1.40350%	-1.79411%	-2.52682%

Respectivly, ARMA(1,1)-GARCH(1,1)-RM-OL gives the lowest values for required capital at 90 and 95% confidence level, while Oracle SVM-Std5 RM VaR outperformed it at 99% confidence level, as it can be seen in table 6.

Table 6. Capital requirements at 90, 95 and 99% confidence levels, for VaR models.

	Capital Requirements		
	90%	95%	99%
ARMA(1,1)-GARCH(1,1)-RM	6.46509%	8.17626%	8.53961%
Oracle SVM-Abs RM VaR	5.89014%	6.69439%	7.75661%
Oracle SVM-Std5 RM VaR	5.75751%	5.45099%	7.57605%
ARMA(1,1)-GARCH(1,1)-RM-OL	5.05260%	5.38232%	7.58045%

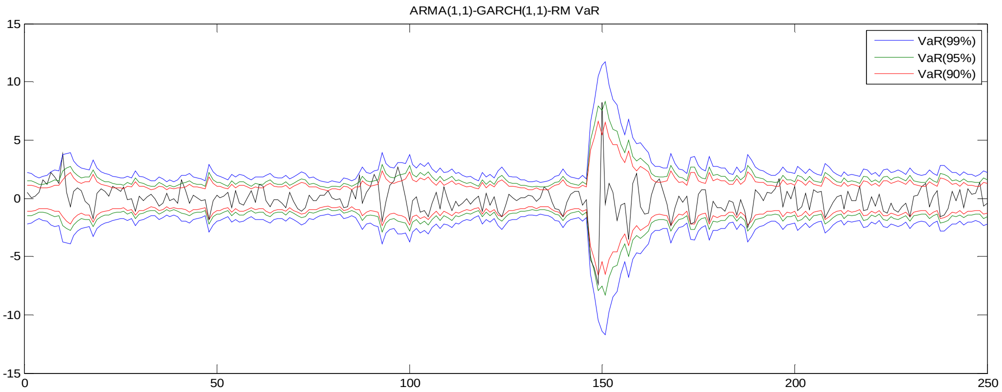
To identify what method of calculating VaR is acceptable to the regulators, the authors use Kupiec test at 5% significance level for backtesting the achieved results. VaR values are estimated by setting a fixed window length and making a forecast one day in advance. Backtesting is implemented on the out-sample dataset by comparing achieved results with real data. The results of the overall acceptance, according to Kupiec test are presented in table 7.

Table 7. Kupiec test backtesting results at 90, 95 and 99 % confidence levels

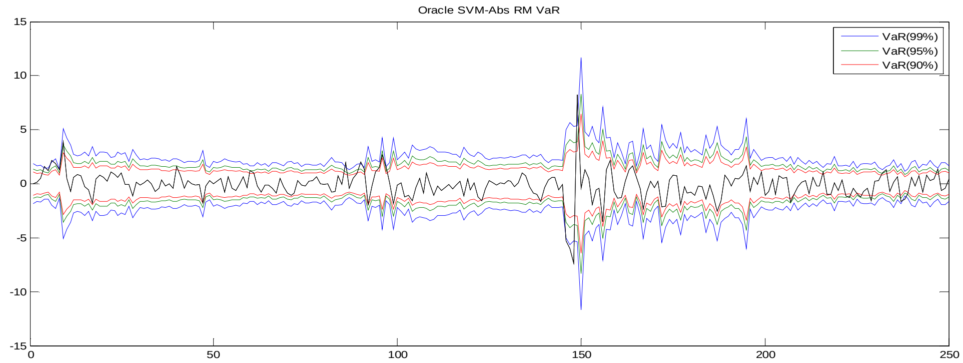
	Kupiec test		
	90%	95%	99%
ARMA(1,1)-GARCH(1,1)-RM	+ (20)	+ (10)	+ (2)
Oracle SVM-Abs RM VaR	+ (16)	+ (7)	+ (4)
Oracle SVM-Std5 RM VaR	+ (11)	+ (4)	+ (1)
ARMA(1,1)-GARCH(1,1)-RM-OL	+ (7)	+ (1)	+ (0)

The results of Kupiec test confirm acceptability of all tested methods for calculation VaR. However, SVM methods, particularly Oracle Std5 RM-SVM, are significantly better than the ARMA (1,1)-GARCH (1,1)-RM method. Again, updating daily GARCH model (ARMA (1,1)-GARCH (1,1)-RM-OL) achieved the best results on the test. Comparing the difference in the results of the ARMA (1,1)-GARCH (1,1)-RM and ARMA (1,1)-GARCH (1,1)-RM-OL we can see that the constant model adaptation (day by day) gives significantly better results (regardless of model) than a static model over time. On the other hand, daily model updating for each financial series requires significant expenditure of computing resources.

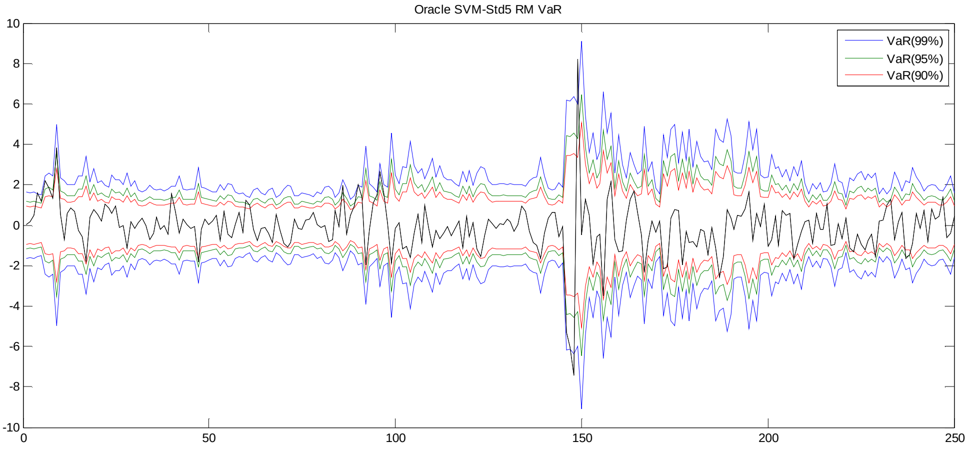
Figure 4. The estimated value of the VaR for 90%, 95% and 99% confidence levels of the following models: a) ARMA (1,1)-GARCH (1,1)-RM, b) Oracle SVM Abs RM VaR, c) Oracle SVM Std5 RM VaR.



a)



b)



c)

The tested SVM models were models with fixed model parameters. The parameter C and core function are chosen empirically without tuning and models optimization. Most of the research highlights the sensitivity of the SVM forecasting accuracy to the model parameters. This leaves additional space for improving the quality of the SVM models and will be part of our future research.

4. Conclusion

The experimental results show that the SVM outperformed the GARCH (1,1). But day-by-day model updating provides more accurate forecasting. In this paper, we investigate settings of SVM parameters (C, kernel), but we need to point out that forecasting performances of SVMs are sensitive to the value of the model parameters. This issue provides opportunities for significant improvement of the model. This study shows that SVM provides a promising alternative for forecasting financial time-series volatility.

Oracle's SVM implementation allows database users with relatively little knowledge on data mining tools with little effort in the use of tools to improve the SVM forecasting of financial time series and greatly improve the quality of the model based on these tools. However, the advance usage of implemented SVM tools is not an easy task and often depends on the user experience, which usually requires additional empirical research and the increased consumption of computing resources.

References

1. Andersen TG, Bollerslev T, 1998, Answering the Skeptics: Yes, Standard Volatility Models do Provide Accurate Forecasts, International Economic Review, 39, 885-905
2. Bollerslev, T., 1986, Generalized autoregressive conditional heteroskedasticity, Journal of Econometrics 31, 307-327.
3. Bollerslev, T., Chou R.Y., and Kroner K.F., 1992, ARCH modeling in finance: A review of the theory and empirical evidence, Journal of Econometrics 52, 5-59.
4. Chen WH, Jen-Ying S, Soushan WU (2006). Comparison of support vector machines and back propagation neural networks in forecasting the six major Asian stock markets. Int. J. Electron. Finan., 1: 49-67.
5. Gavrishchaka VV, Supriya BG (2003). Volatility forecasting from multiscale and high-dimensional market data. Neurocomputing, 55(1- 2): 285-305.
6. Huang W, Nakamori Y, Wang SY (2005). Forecasting stock market movement direction with support vector machine. Comput. Oper. Res., 32(10): 2513-2522.
7. Huang Z, Chen H, Hsu C-J, Chen W-H, Wu S (2004). Credit rating analysis with support vector machines and neural networks: a market comparative study. Decis. Support Syst., 37(4): 543-558.
8. Jorion P., 1995, Predicting volatility in the foreign exchange market, Journal of Finance, 50, 507–528.
9. Jorion, P. “Risk Measuring the Risk in Value at Risk.” Financial Analysts Journal, November-December 1996, pp. 47-56.
10. Milenova, B., Yarmus, J., Campos, M., (2008) SVM in Oracle Database 10g: Removing the Barriers to Widespread Adoption of Support Vector Machines, http://www.oracle.com/technetwork/database/options/advanced-analytics/odm/overview/support-vector-machines-paper-1205-129825.pdf
11. Moorad C., “An Introduction to Value-at-Risk”, 2006, John Wiley & Sons Ltd, England, pp. 33-34
12. Radović, O., Marinković, S., Stanković, J. (2011). Serbian Capital Market During and After the Crisis – VaR Estimation of a Hypothetical Portfolio, Proceedings of the Ninth International Conference: “Challenges of Europe: Growth and Competitiveness – Reversing the Trends”, Split, Croatia, 627 – 645
13. Sansom DC, Downs T, Saha TK (2003). Evaluation of support vector machine based forecasting tool in electricity price forecasting for Australian national electricity market participants. J. Electric Electron. Eng., 22(3): 227-234.
14. Shin K.S., Lee T.S., and Kim H.J. (2005), An application of support vector machines in bankruptcy prediction model, Expert Systems with Applications 28, 127–135.
15. Stackowiak, R., Rayman, J., Greenwald, R. (2007), Indianapolis, Wiley Publishing, Oracle Data Warehousing and Business Intelligence Solutions
16. Tay FEH, Lijuan C (2001). Application of support vector machines in financial time series forecasting. Omega, 29: 309–317.
17. Vapnik V, Golowich S, Smola A (1997). Support vector method for function approximation, regression estimation, and signal processing, in: Mozer M, Jordan M, and Petsche T, (Eds.), Advances in Neural Information

18. Vapnik VN (1995). The nature of statistical learning theory, Springer-Verlag, New York.
19. Žiković, S., (2007). Testing Popular VaR Models in EU New Member and Candidate States, Journal of Economics and Business, 25: 325 – 346.
20. Oracle Data Mining Concepts 11g Release 2 (11.2), http://docs.oracle.com/cd/E11882_01/datamine.112/e16808.pdf, (15.06.2012.)

PRIMENA ORACLE DATA MINER-a ZA PREDVIĐANJE FINANSIJSKIH VREMENSKIH SERIJA

Rezime: Poslednjih godina vektori podrške (SVM), nova tehnika veštačkih neuronskih mreža (ANN), uspešno se koriste kao neparametarski alat za regresionu procenu i predviđanje vremenskih serija. U ovom radu autori koriste Support Vector Regression (SVR) alat Oracle DM u predviđanju volatilnosti Belex 15 indeksa i procene Value-at-Risk (VaR). Implementirana su dva modela SVR, prvi, zasnovan na proceni apsolutne vrednosti prinosa, i drugi, zasnovan na proceni 5-dnevne standardne devijacije. Ovi modeli uporedjeni su sa standarnim GARCH(1,1) metodom procene volatilnosti. U ovom istraživanju, korišćeni su istorijski podaci o prinosu Belex 15 indeksa od aprila 2005 do decembra 2011. godine. Rezultati istraživanja pokazuju da SVR alati daju bolje procene VaR od GARCH(1,1) metoda. Međutim, preporučuje se svakodnevno ažuriranje modela i kvalitetnije podešavanje parametera SVR-a.

Ključne reči: Regresija vektorima podrške, Oracle Data Miner, finansijske vremenske serije