

# Base Density Support Vector Machine for Optimization the Stock Company

Adrian D. Oлару, Serban A. Oлару, Mihaela B. Ciorei and Nicolae F. Mihai

**Abstract**— The base density of the support vector for machine learning (BDSVM) is important to establish the basic data for neural network learning. This paper presents a method to find the most important data and establish the objective function (FO) for the vector support machine learning (SVM). SVM is the most successful algorithm for classification problems. SVM establishes the decision limit between two classes of collected data (in the binary classification variant), this constituting the training points for the neural network. However, sometimes there are some less significant data among the training points, which are far from the training limit, being considered noise or data misplaced in Cartesian space from this training limit, called outliers. These outliers affect the performance of the learning stage of the neural network, and the proposed BDSVM algorithm eliminates them. Many research results prove this sensitivity of SVM to outliers, which is a weak point for SVM. Various researchers have proposed different methods approaches to reduce the effect of outliers, but no one method is suitable for all types of data sets. In this paper, the new method of collected data base density for support vector machine (BDSVM) is introduced.

**Index Terms**—Assisted research, Base density, Classification, LabView software, Machine learning, Optimization stock, Support vector machine.

## I. INTRODUCTION

Support Vector Machine (SVM) is a supervised classification method derived from statistical learning theory that often provides good classification results from complex data that includes both correct data and noise or outliers. SVM separates the collected data into classes through a decision limit, which maximizes the margin of obtaining correct data between the classes addressed. The surface is called the optimal hyperplane [1]-[3], and the data points closest to the hyperplane are called support vectors. The support vectors are the critical elements of the training set. You can adapt SVM to become a non-linear classifier by using non-linear kernels. While SVM is a binary classifier in its simplest form, it can work as a multiclass classifier by combining multiple binary SVM classifiers (creating a binary classifier for each possible pair of classes). SVM includes a penalty parameter that allows for some degree of misclassification, which is particularly important for non-separable training sets. The

penalty parameter controls the optimal trade-off between the data producing training errors and the correct data in the vicinity of the separation zone (minimum distances of the points from the points on the separation curve). Classification is a type of statistical problem, where we want to predict with accuracy, a membership in a predefined class, based on the characteristics of that class. In general, the main purpose of classification is to learn the discrimination rule to achieve the minimum error rate. In the statistical literature, linear discriminant analysis (LDA) Fisher's and quadratic discriminant analysis (QDA) are classic examples of a discriminant rule, and modern statistical tools include classification trees, logistic regression, neural networks, and methods based on support vector density, see [4]-[6]. Support Vector Machines is an important example of computational methods, one of the key areas in machine learning. It originates in the theoretical foundations of Statistical Learning Theory and Structural Risk Minimization (SRM) [7]-[8]. SVM was introduced by Vapnik and colleagues in the 1970s, but its major developments were formulated in the 1990s. The main idea behind SVM is to find an optimal separating hyperplane with a maximized optimization margin. The maximum margin reduces the empirical risks (training errors) and leads to a very good generalization performance. SVM has become very famous due to its high generalization ability and good performance in pattern recognition (digit recognition, computer vision, and text and speech categorization, etc.) and has found applications in a wide variety of fields [8]. The main objectives of the base density of SVM are: reducing the effects of outliers, maximizing the decision to obtain the optimal boundary between data sets, ensuring better generalization and adjusting the decision boundary according to the density of the data sets. Meanwhile, base density SVM reduces the number of support vectors, which decreases the computational complexity. It is worth noting that in the base density SVM, the input vectors are those that are in the highest confidence region of the dataset than other input vectors. Base density SVM can detect outliers or data points that are outside the densely populated area, singular or erroneous points. To detect these outliers, the densely populated area of a data set must first be determined. Data points that are located in the densely populated area will be considered important (meaningful) points and others as less important (meaningless), which may be misclassified or ignored. Although the concept of population density is used to develop density-based SVM, the formula is different. In

Manuscript received January 30, 2024. This work was supported in part by the Ella Icon company (sponsor and financial support acknowledgment goes here).

A. D. Oлару is with the National University of Science and Technologies POLITEHNICA Bucharest, RO 060042 Romania (e-mail: aolaru\_51@ymail.com).

S. A. Oлару is with the Aerospace Engineering Department, Military Equipment and Technology Research Agency, Romania (e-mail: serban1978@yahoo.com).

M. B. Ciorei was with Ella Icon company, Bucharest, RO 60317 Romania (e-mail: mihaela.ciorei89@gmail.com).

N. F. Mihai is with the Techno Accord robotics company, Leuval 030042, Canada (e-mail: mniculae@yahoo.com).

this method, the distance (Euclidean and Mahalanobis) between the data points of a data set, plays a major role in determining the area of high population density.

## II. ALGORITHM OF BDSVM

The distance between two points calculated in Euclidean space is presented in (6) [9]-[11]. Suppose a two-dimensional data set is given by  $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ . First, the Euclidean distance between all data points of a class is calculated. For example, the Euclidean distance between point 1 and 2, 3, ...,  $n$  and the Euclidean distance between point 2 at 1, 3, ...,  $n$  and so on.

$$\begin{aligned} d_{12} &= \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \\ d_{13} &= \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} \\ &\dots \\ d_{1n} &= \sqrt{(x_1 - x_n)^2 + (y_1 - y_n)^2} \end{aligned} \quad (6)$$

The distance average of  $d_i$  is:

$$d_1 = \frac{d_{12} + d_{13} + \dots + d_{1n}}{n-1} \quad (7)$$

The total distance average of the group is:

$$d = \frac{d_1 + d_2 + \dots + d_n}{n} \quad (8)$$

If  $d_i > d$ ,  
the point  $i$  is in outlier group;  
Else  
the point  $i$  will be considered important (meaningful) points in BDSVM;  
End.

From these meaningful points will be construct the optimization function (FO) that will be used in the training neural network to obtain the optimal stock.

The distance between the points representing the input data calculated by the Mahalanobis method [9] instead of the Euclidean distance is calculated with (9). The Mahalanobis distance is calculated from the quantity  $\mu$  which represents the average of the points' distances, to each point. The  $cov^{-1}$  represents the inverse covariance matrix. This distance is based on the correlation between the variables or the variance-covariance matrix [11]-[12]. The Mahalanobis distance is the smaller unit and takes into account the correlation of the data set and does not depend on the measurement scale [9]-[10]. The Mahalanobis distance from the point to the mean of the distribution  $\mu$  can be calculated by (9), and the Mahalanobis distance from one point to another can be calculated by (10):

$$d = \sqrt{(x - \mu)^T cov^{-1} (x - \mu)} \quad (9)$$

$$d = \sqrt{(x - y)^T cov^{-1} (x - y)} \quad (10)$$

Where the population variance is calculated with [12]:

$$var(x_n) = \frac{\sum_1^n (x - \mu)^2}{n} \quad (11)$$

and population covariance with:

$$cov(x_n, y_n) = \frac{\sum_1^n (x_i - \mu_x)(y_i - \mu_y)}{n} \quad (12)$$

If  $cov(x_i) \& cov(y_i) > 0$   
both of them increase or decrease;  
If  $cov(x_i) \& cov(y_i) < 0$   
when  $x_i$  increase  $y_i$  decrease or vice-versa;  
If  $cov(x_i) \& cov(y_i) = 0$   
not exist any relation between  $x_i$  &  $y_i$ ;  
If  $var(x_i) > var(y_i)$   
 $x_i$  increase or decrease faster than  $y_i$ ;  
End.

Average of  $d$  is:

$$average\_d = \frac{\sum_1^n \sqrt{(x_i - \mu)^T cov^{-1} (x_i - \mu)}}{n} \quad (13)$$

If  $d_i > d$ ,  
the point  $i$  is in outlier group;  
Else  
the point  $i$  will be considered important (meaningful) points in BDSVM;  
End.

The outlier points  $x_3$  will be removed from the weight matrix of the neural network, also the row and column that contents the outlier point  $x_3$  will be removed from the matrix, because it is necessary that the kernel matrix must be quadratic (14):

$$\begin{bmatrix} k(x_1, x_1) & k(x_1, x_2) & k(x_1, x_3) & \dots & k(x_1, x_n) \\ k(x_2, x_1) & k(x_2, x_2) & k(x_2, x_3) & \dots & k(x_2, x_n) \\ k(x_3, x_1) & k(x_3, x_2) & k(x_3, x_3) & \dots & k(x_3, x_n) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ k(x_n, x_1) & k(x_n, x_2) & k(x_n, x_3) & \dots & k(x_n, x_n) \end{bmatrix} \quad (14)$$

The block schema to define the base density support vector machine learning BDSVM are shown in fig.1 [11].

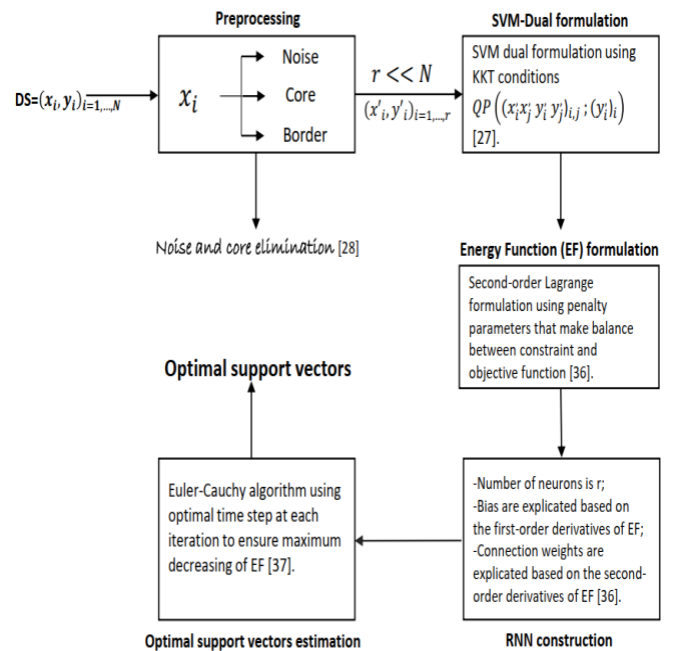


Fig.1. Block schema to define the BDSVM, by courtesy of authors of paper [11].

The optimization function (FO) was proposed one third polynomial functions with real coefficients:

$$FO = a_1 * x^3 + a_2 * x^2 + a_3 * x + a_4 \quad (12)$$

where the real coefficients  $a_i$  were determined by using the matrix form of equation:

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{pmatrix} = \begin{bmatrix} x_1^3 & \dots & x_1 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ x_4^3 & \dots & x_4 & 1 \end{bmatrix}^{-1} \cdot \begin{pmatrix} FO_1 \\ \dots \\ FO_4 \end{pmatrix} \quad (13)$$

with the following constraints:

- $x_i > 0$ ;
- $x_i$  must be meaningful points,  $x_i \in \text{group } I$ ;
- $x_i \in \text{BDSVM}$ ;
- $x_i \in \text{group A, B, C}$ .

To establish the optimal number of the stock products was used the A,B,C theory that impose the following data [13]:

- A- contents the products with 20% from the total number of the products and 80% of total value of products;
- B- contents the products with 30% from the total number of the products and 15% of total value of products;
- C- contents the products with 50% from the total number of the products and 5% of total value of products;

### III. LABVIEW INSTRUMENTATION

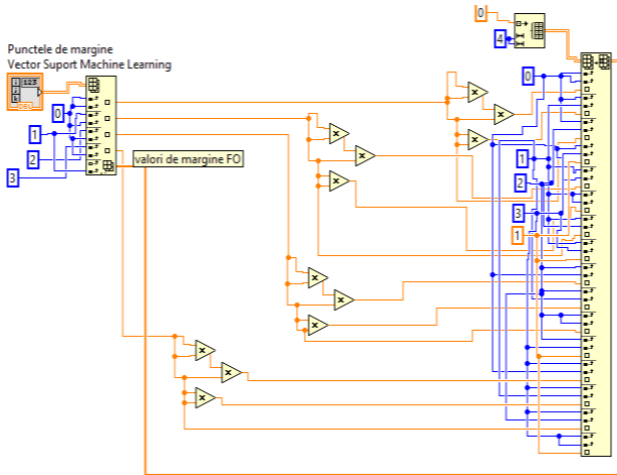


Fig.2. Part of the block schema of the LabView virtual instrument to find the optimal function's real coefficients.

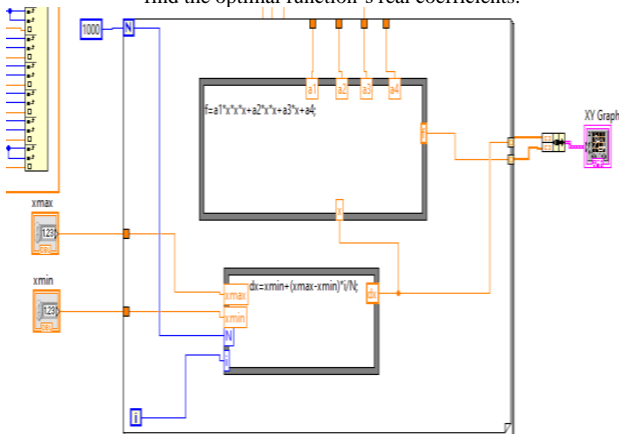


Fig.3. Part of block schema of the LabView virtual instrument to design the characteristic of the FO.

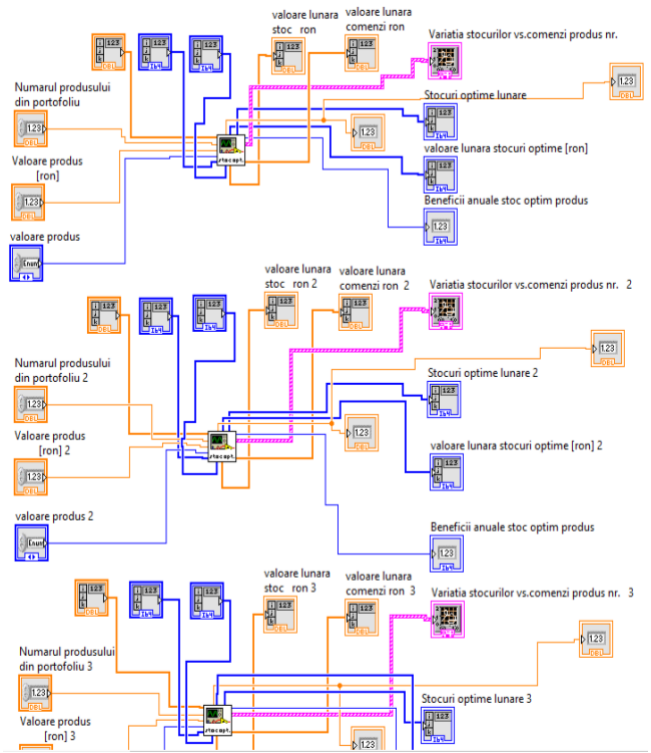


Fig.4. Block schema of the virtual LabView instrument to compare the stock values from diverse A,B,C categories.

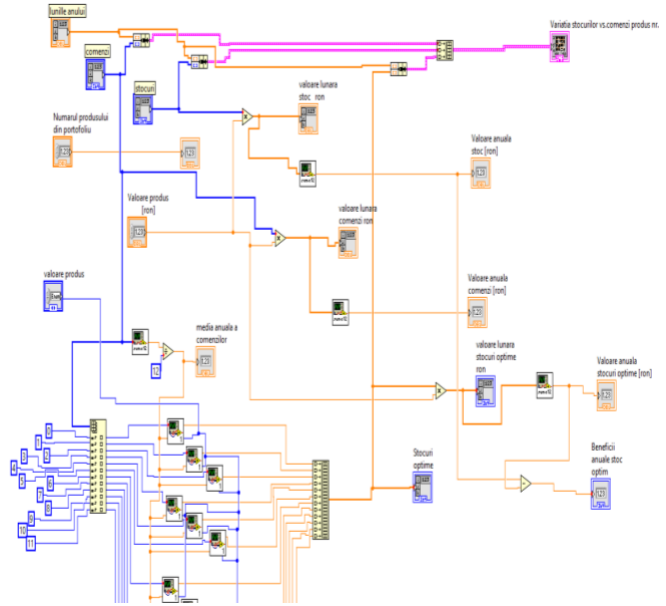


Fig.5. Block schema of the virtual LabView instrument to determine the optimal stock by using the BDSVM.

Virtual instruments  $V_i$  with LabView soft cover the assisted research to find the polynomial optimizing function (FO) after core and noise elimination. We proposed to develop one polynomial function with three order with constant coefficients, FO. In the figs.2-3 are shown the coefficients results values and the characteristic of the FO. After the first step of the block schema solved to define BDSVM, fig.1, and used the mathematical proposed model (1)- (13) were find four points from the border that we used to find the real coefficients  $a_i$  of the FO. By using the FO and the ABC theory were construct the  $V_i$ -s for the calculus of the optimized stock of the products in three variant with value<200, 200<value<500 and value>500, fig.4. To compare the initial stocks, the command product and the optimized stock was designed LabView  $V_i$ -s from the fig.5.

#### IV. CASE STUDY- STOCK OPTIMIZATION

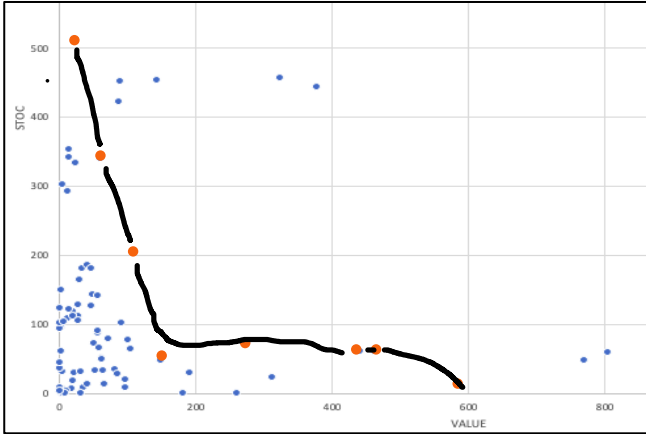


Fig.4. Border position of all analyzed data value- vs. stock in the analyzed company; optimal established points with red color.

TABLE I: THE DATA OF VALUES AND STOCKS OF THE ANALYZED COMPANY

Total Sales Value	STOC
28,874	111
50,851	73
47,76	126
108,82	204
87,037	422
90,691	452
36,726	8
56,306	141
97,389	8
61,686	339
63,256	51
91,676	102
101,202	77
150,048	48
192,232	30
21,485	19
4,022	95
5,385	6
144,322	453
323,698	457
378,174	443
52,732	33
56,291	87
82,647	35
64,631	33
57,241	91
96,791	20
4,523	31
1,368	9
71,88	79
62,89	50
67,253	13
441	61
277	74

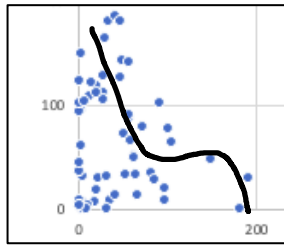


Fig.5. Zoom of the data in the first field value<200 with border values.

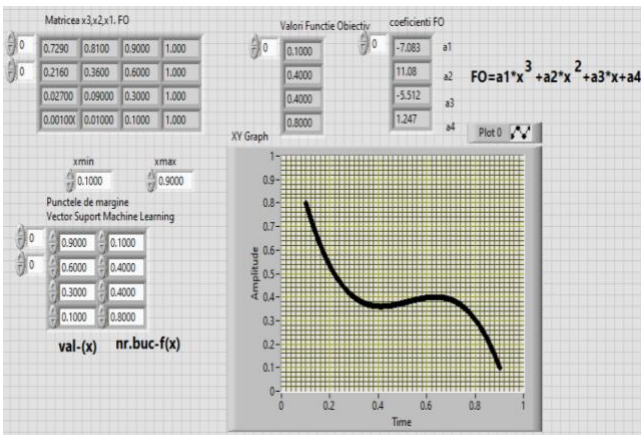
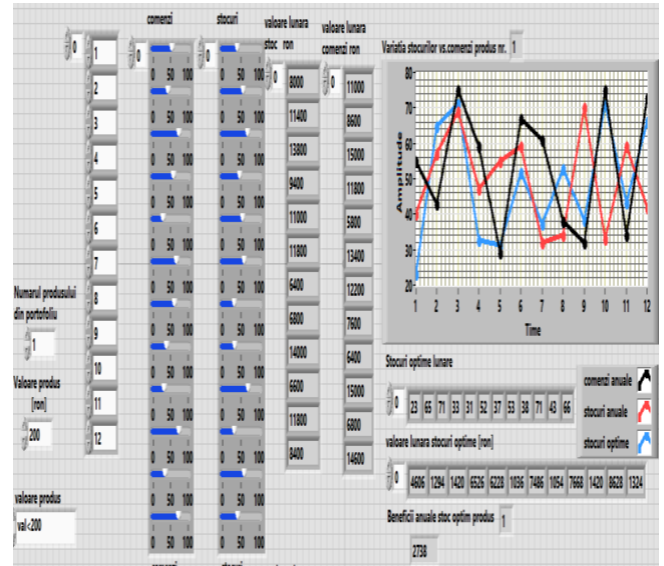
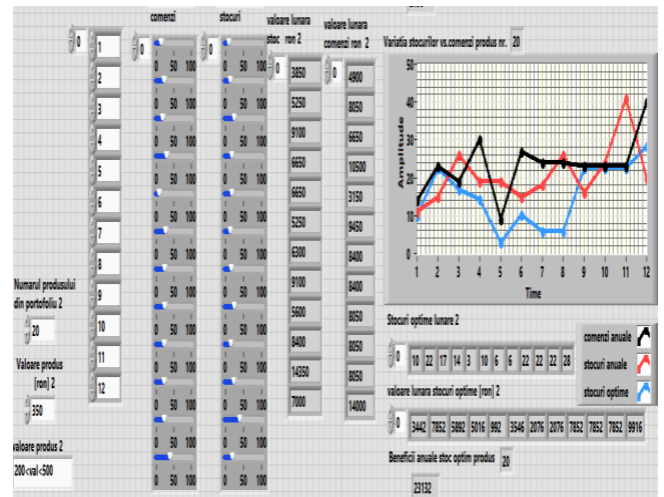


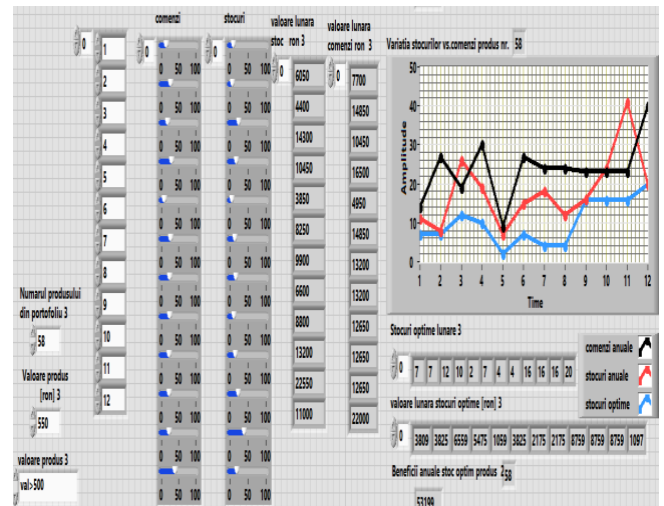
Fig.6. The front panel of the LabView virtual instrument with results for Optimal Function (FO) established by using the ABC theory and BDSVM of data.



a



b



c

Fig.7. Front panel with results after was applied the FO to establish the optimal stock by using the BDSVM and ABC theory; a-the results for the value < 200; b- the results for the 200<value< 500; c-the results for the value >500.

The results contents some characteristics after were applied the BDSVM mathematical complex model and was designed the value vs. stock points, see fig.4-5 and table I, to establish the core, the noises and optimal border. The optimal border,

that was nominated in the paper like optimizing function (FO), was established after choose some points and construct one polynomial function with three order with constant coefficients and were determined them, see fig.6. The optimizing stock characteristics were design in figs.7, for all analyzed cases with ABC theory, with a-the results for the value < 200; b- the results for the 200<value< 500 and c-the results for the value >500.

The obtained objective function (FO), by using the border points, considered important (meaningful) points is:

$$FO = -7.083x^3 + 11.08x^2 - 5.512x + 1.247 \quad (14)$$

Obs. All calculations were performed in normalized coordinates. All other choosing points must to respect strictly the constraint  $x_i \in FO$ .

After will be applied the proposed method will be obtained easily the convergence process for the used neural network that will be filtered the results.

## V. CONCLUSION

The method shown in the paper, the results, the LabView Vi-s cover the theory to optimize the SVM, to establish the base density for the support vector for the machine learning (BDSVM) and open the way to optimize the stocks, reduce the financial effort of the companies and assure one optimal repartition of the products for each month, during the year. The proposed algorithm can be improved with new conditions so that the obtained results will be better filtered and the convergence of the applied neural networks will be better ensured.

## CONFLICT OF INTEREST

The authors declare no conflict of interest.

## AUTHOR CONTRIBUTIONS

AO conducted the research and wrote some chapters of the paper; SO wrote some chapters of the paper and check the English language; MC analyzed the data, wrote some chapters of the paper and check the company stock; NM

wrote some chapters and check the stock results; all authors had approved the final version.

## ACKNOWLEDGMENT

All authors thanks to Ella Icon private company from Romania and to Techno Accord private company from Canada to help us with data for the case study and the stocks.

## REFERENCES

- [1] Chih Chung Chang and Chih Jen Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2, 27:1-27:27 (2011). Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [2] Hsu, C.-W., C.-C. Chang, and C.-J. Lin. (2010). A practical guide to support vector classification. National Taiwan University. <https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>.
- [3] Wu, T.-F., C.-J. Lin, and R. C. Weng. (2004). Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5:975-1005, <https://www.csie.ntu.edu.tw/~cjlin/papers/svmprob/svmprob.pdf>.
- [4] Hastie, T., Tibshirani, R., and Friedman, J. (2001) *The Elements of Statistical Learning*. Springer Verlag, New York.
- [5] Duda, R. O., Hart, P. E., and Stork, D. G. (2000) *Pattern Classification (2nd Edition)*. Wiley-Interscience, New York.
- [6] McLachlan, G. J. (2004) *Discriminant Analysis and Statistical Pattern Recognition*. Wiley-Interscience, New York.
- [7] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer 2000.
- [8] V. Kecman, *Learning and Soft Computing, Support Vector Machines, Neural Networks, and Fuzzy Logic Models*, The MIT Press 2001.
- [9] Y. Dodge, *The Concise Encyclopedia of Statistics*, Springer 2008.
- [10] Z. Nazari, D. Kang, and H. Endo, *Density Based Support Vector Machines, The 29<sup>th</sup> International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, pp.1-3, 2014.
- [11] El Moutaouakil K, El Ouissari A, Olaru A, Palade V, Ciorei M. OPT-RNN-DBSVM: OPTimal Recurrent Neural Network and Density-Based Support Vector Machine. *Mathematics*. 2023; 11(16):3555. <https://doi.org/10.3390/math11163555>.
- [12] Variance-covariance matrix- GeeksforGeeks.html
- [13] <https://www.netsuite.com/portal/resource/articles/inventory-management/abc-inventory-analysis.shtml>.

Copyright © 2020 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).