Годишник на секция “Информатика”
Съюз на учените в България
Том 7, 2014, 28–40

Annual of “Informatics” Section
Union of Scientists in Bulgaria
Volume 7, 2014, 28–40

GENERALIZED NET OF THE PROCESS OF SUPPORT VECTOR MACHINES CLASSIFIER CONSTRUCTION

Veselina Bureva[1], Mirela Puleva[2] and Evdokia Sotirova[1]

[1] Prof. Asen Zlatarov University
Burgas-8000, Bulgaria
e-mails: vbureva@btu.bg, esotirova@btu.bg

[2] School of Mathematics and Natural Sciences Acad. Nikola Obreshkov
Burgas-8000, Bulgaria
e-mail: mirrellaa@gmail.com

Abstract: In the current paper is presented a generalized net model of the process of construction of a support vector machine classifier. It can be helpful in analizing, managing and optimizing the process. The implementation of the technique is applied to weather data.
Keywords: Artificial intelligence, Generalized nets, Data mining, Machine learning, Support vector machines, Weather databases.

1 Introduction

Support vector machines (SVMs) is a statistical method frequently used in the area of machine learning. In data mining SVMs is applied in order to classifying data and predict the future trends in different areas – medical diagnosis, bioinformatics, pattern recognition, image processing and text mining. The support vector classifiers have to be trained and after that they can be used for predicting future patterns. SVM algorithms try to find an “optimal hyper-plane” that can maximize the distance two groups of samples. The hyper-plane is optimal when it stands at maximum distance to the points of two classes. The two parallel planes to the hyper-plane, which are placed the nearest to the points of the separating sets are called “support vectors” (Fig. 1). The distance between the support vectors for each class is called “margin”. In the case, when there are many hyper-planes it searches for the “maximum hyper-plane”. It separates the classes the best. The elements from the classes that are closest to the hyperplane are called support vectors**.** The support vectors must be maximal near the points of the each class and maximal the farthest from the hyper-plane [9–12]. When the distance between the support vectors and the hyper-plane is too little, then the classifier has a "hard margin". The classes may include the single points, which are mapped on the margin place or they can be mapped on the place for the other class. In this case, these points are perceived as "noise" in the data and the margin technique is called "soft margin" [13, 15]. The optimization is made by applying the other methods (quadratic optimization) and introducing new parameters – "slack variables ε". When the sets cannot separate linearly it is needed to be mapped the input points on high dimensional space and the technique "kernel trick" to be applied. This is made while the classes become linearly separable. In this high dimensional feature space, linear classification can be performed. Therefore, the SVM is used to successfully classify linearly separable and non-linearly separable datasets [4–8].

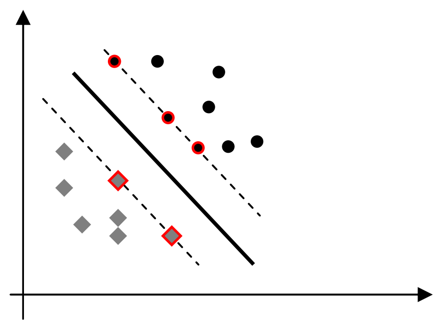

Figure 1. The separating hyperplane and support vectors

In the following paragraph are described the simple steps of algorithm for SVM in non-linear case:

- Preprocess the data – (missing values, presumably incorrect values, noise in the data, Numerisize the data, Normalize the data);
- Develop the model(s) – Select kernel type, Determine kernel parameters based on the selected kernel type (This is a hard problem. One should consider using cross validation and experimentation to determine the appropriate values for these parameters);
- If the results are satisfactory, finalize the model, otherwise change the kernel type and/or kernel parameters to achieve the desired accuracy level;
- Extract and deploy the model.

In Multi-class classification problems can be included algorithms based thoroughly on the two-class classification [19, 20]. On the area of these techniques belong all other techniques that have and own process of training the data. Widely-used methods are One-against-all and One-against-one, Decision tree SVM, DAG SVM (directed acyclic graph SVM), Fuzzy SVM. While they training the data, they group all the classes in pairs.

2 Generalized network model

The concept of Generalized nets (GNs) is introduced in [1, 2]. The Generalized Net model of the process of support vector machines classifier construction is presented in Fig. 2. In [3, 14, 18] are presented other methods for knowledge discovery, which are modelled by generalized nets. The implementation of these techniques is also made by using a weather data.

The GN model of the process of support vector machines contains 11 transitions and 48 places. It contains the following set of transitions:

$$A = \{Z_1, Z_2, Z_3, Z_4, Z_5, Z_6, Z_7, Z_8, Z_9, Z_{10}, Z_{11}\},$$

where the transitions describe these process:

- Z_1 – “preprocessing the input data”;
- Z_2 – “a multi-class classification”;
- Z_3 – “separating the input data on training, validation and testing sets”;
- Z_4 – “choosing a method for multi-class classification”;
- Z_5 – “mapping data on R^2, finding the best hyper-plane, the support vectors, and the margin”;
- Z_6 – “choosing the another method for multi-class classification or one multi-class classifier needs additional parameters (Decision Tree SVM, directed acyclic graph SVM (DAGSVM))”;
- Z_7 – “optimization of the classifier”;
- Z_8 – “finding nonlinear classifier”;
- Z_9 – “finding a multi-class classifier”;
- Z_{10} – “validating the classifier”;
- Z_{11} – “testing the classifier”.

Initially, there is one α_1-token that is located in place L_6 with initial characteristic: “*data for preprocessing*”.

In the next time-moments this token is split into two or more. The original α_1-token will continue to stay in place L_6, while the other α-tokens will move to the next transitions.

Initially, there is one β_1-token that is located in place L_5 with initial characteristic: “*preprocessing techniques*”.

In the next time-moments this token is split into two. The original β_1-token will continue to stay in place L_5, while the other β-token will move to transition Z_2 via place L_3.

From places L_1, L_2, and L_7 the α_2-, β_3- and β_2-tokens enter the net with initial characteristic respectively: “*new data for classifying*” in place L_1, “*new preprocessing technique*” in place L_2, and “*method for multi-class classification*” in place L_7.

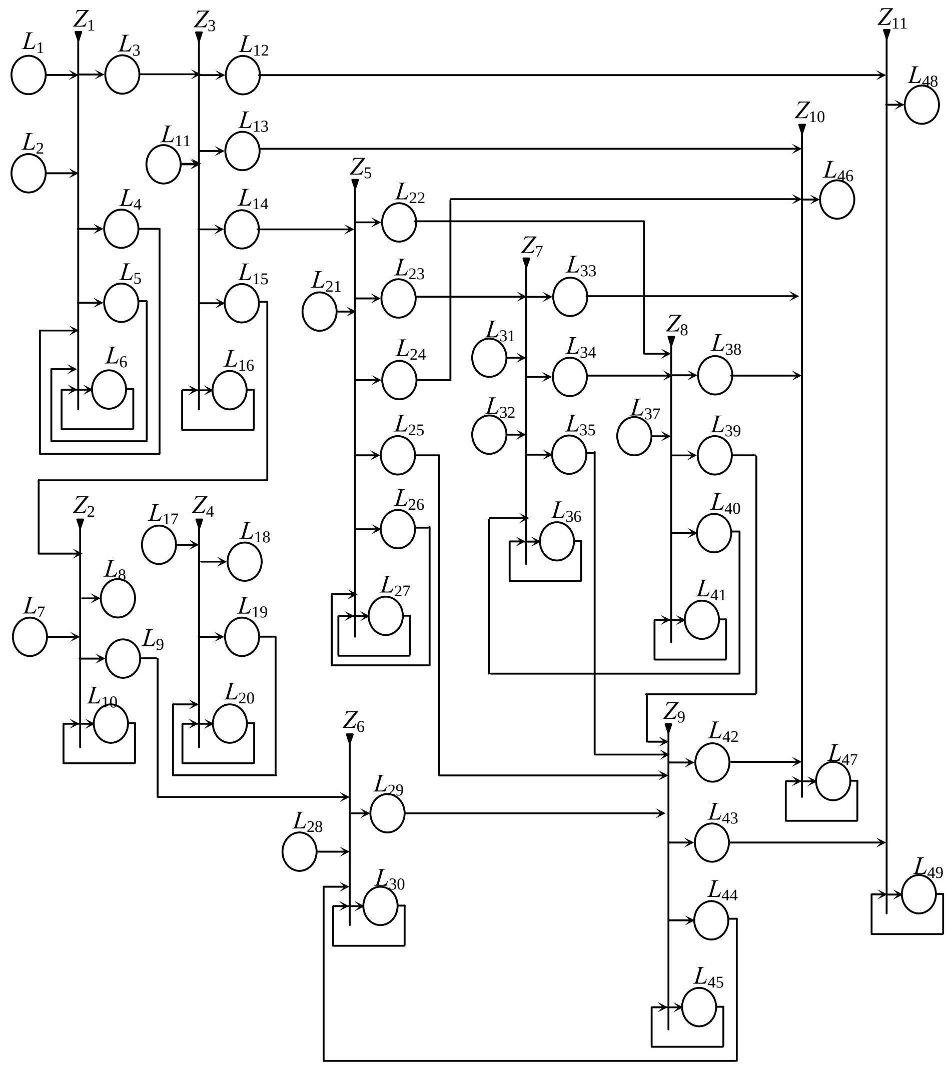

Figure 2. GN model of the process of Support vector machines classifier construction

The transition Z_1 has the form:

$$Z_1 = \langle \{L_1, L_2, L_4, L_5, L_6\}, \{L_3, L_4, L_5, L_6\}, R_1, \vee(\wedge (L_1, L_2), L_4, L_5, L_6) \rangle,$$

where:

$$R_1 = \begin{array}{c|cccc} & L_3 & L_4 & L_5 & L_6 \\ \hline L_1 & false & false & false & true \\ L_2 & false & false & true & false \\ L_4 & W_{4,3} & W_{4,4} & false & false \\ L_5 & false & W_{5,4} & W_{5,5} & false \\ L_6 & false & W_{6,4} & false & W_{6,6} \end{array},$$

and:

- $W_{4,3}$ = "The data is preprocessed",
- $W_{4,4} = \neg W_{4,3}$.
- $W_{5,4}$ = "The method for preprocessing is selected",
- $W_{5,5} = \neg W_{5,4}$,
- $W_{6,4}$ = "The data to be preprocessed is selected",
- $W_{6,6} = \neg W_{6,4}$,

The α_3-tokens that enter in place L_3 obtain the characteristic: "*Data for partitioning*".

The transition Z_2 has the form:

$$Z_2 = \langle\{L_7, L_{10}, L_{15}\}, \{L_8, L_9, L_{10}\}, R_2, \vee(\wedge(L_7, L_{15}), L_{10})\rangle,$$

where:

$$R_2 = \begin{array}{c|ccc} & L_8 & L_9 & L_{10} \\ \hline L_7 & false & false & true \\ L_{10} & W_{10,8} & W_{10,9} & W_{10,10} \\ L_{15} & false & false & true \end{array},$$

and:

- $W_{10,8}$ = "The method for the input data, based on two-class classification, is chosen",
- $W_{10,9}$ ="The other method for the input data is chosen",
- $W_{10,10} = \neg (W_{10,8} \wedge W_{10,9})$.

The α-tokens that enter in the places L_8 and L_9 have the following characteristics: "*the method based on two-class classification and input data*" in place L_8, and "*other method and input data*" in place L_9.

The token β_4 enters the net via place L_{17} with initial characteristics: "*method for grouping the classes*".

The transition Z_3 has the form:

$$Z_3 = \langle L_3, L_{11}, L_{16}\}, \{L_{12}, L_{13}, L_{14}, L_{15}, L_{16}\}, R_3, \vee(\wedge(L_3, L_{11}), L_{16})\rangle,$$

where:

$$R_3 = \begin{array}{c|ccccc} & L_{12} & L_{13} & L_{14} & L_{15} & L_{16} \\ \hline L_3 & false & false & false & false & true \\ L_{11} & false & false & false & false & true \\ L_{16} & W_{16,12} & W_{16,13} & W_{16,14} & W_{16,15} & W_{16,16} \end{array},$$

and:

- $W_{16,12} = W_{16,13} = W_{16,14} = W_{16,15} =$ "The data is partitioned",
- $W_{16,16} = \neg\,(W_{16,12} \wedge W_{16,13} \wedge W_{16,14} \wedge W_{16,15})$.

The α-tokens that enter in the places L_{12}, L_{13}, L_{14}, L_{15} have the following characteristics: "*Training set for two-class problem*", "*Validation set*", "*Testing set*", "*Training set for multi-class problem*".

The transition Z_4 has the form:

$$Z_4 = \langle\{L_8, L_{17}, L_{19}, L_{20}\}, \{L_{18}, L_{19}, L_{20}\}, R_4, \vee(\wedge(L_8, L_{17}), L_{19}, L_{20})\rangle,$$

where:

$$R_4 = \begin{array}{c|ccc} & L_{18} & L_{19} & L_{20} \\ \hline L_8 & false & false & true \\ L_{17} & false & true & false \\ L_{19} & W_{19,18} & W_{19,19} & false \\ L_{20} & false & W_{20,19} & W_{20,20} \end{array},$$

and:

- $W_{19,18} =$ "There is a two-class combination for all classes",
- $W_{19,19} = \neg\, W_{19,18}$,
- $W_{20,19} =$ "There are classes for combining",
- $W_{20,20} = \neg\, W_{20,19}$.

The α-tokens that enter in the places L_{18} and L_{19} have the following characteristics: "*two classes for classification"* in the place L_{18} and "*there is a classes for grouping"* in the place L_{19} .

The token β_4 enters the net via place L_{21} with initial characteristics: "*labels for the classes:* 1 *and* –1".

The transition Z_5 has the form:

$$Z_5 = \langle\{L_{13}, L_{14}, L_{21}, L_{26}, L_{27}\}, \{L_{22}, L_{23}, L_{24}, L_{25}, L_{26}, L_{27}\}, R_5, \vee(\wedge(L_{13}, L_{14}, L_{21}), L_{26}, L_{27})\rangle,$$

where:

$$R_5 = \begin{array}{c|cccccc} & L_{22} & L_{23} & L_{24} & L_{25} & L_{26} & L_{27} \\ \hline L_{13} & false & false & false & false & false & true \\ L_{14} & false & false & false & false & false & true \\ L_{21} & false & false & false & false & false & true \\ L_{26} & W_{26,22} & W_{26,23} & W_{26,24} & W_{26,25} & W_{26,26} & W_{26,27} \\ l_{27} & false & false & false & false & W_{27,26} & W_{27,27} \end{array},$$

and:

- $W_{26,22}$ = "The best hyper-plane, the support vectors and the margin are not determined and the model have to be constructed in higher dimension (nonlinear separable case)",
- $W_{26,23}$ = "The best hyper-plane, the support vectors and the margin are determined and the model will be optimized (soft margin - noise, outliers)",
- $W_{26,24}$ = "The best hyper-plane, the support vectors and the margin are determined and the model will be validated",
- $W_{26,25}$ = "The best hyper-plane, the support vectors and the margin are determined and the constructed model will be for multi-class classification",
- $W_{26,26} = \neg (W_{26,22} \wedge W_{26,23} \wedge W_{26,24} \wedge W_{26,25})$,
- $W_{27,26}$ = "There is the input data for mapping on the R^2",
- $W_{27,27} = \neg W_{27,26}$.

The α-tokens that enter in places L_{22}, L_{23}, L_{24}, L_{25}, L_{26} have the following characteristics: "*model for constructing in higher dimension*" in place L_{22}, "*the support vectors and the margin for model optimization*" in place L_{23}, "*the support vectors and the margin for multi-class model*" in place L_{24}, "*linear model for validation*" in place L_{25}, and "*input data for mapping in R^2*" in place L_{26}.

The tokens β_5 enters the net via places L_{23} with initial characteristic: "*other parameters for multi-class classification*".

The transition Z_6 has the form:

$$Z_6 = \langle \{L_9, L_{28}, L_{30}, L_{44}\}, \{L_{29}, L_{30}\}, R_6, \vee(\wedge(L_9, L_{28}, L_{44}), L_{30})\rangle,$$

where:

$$R_6 = \begin{array}{c|cc} & L_{29} & L_{30} \\ \hline L_9 & false & true \\ L_{28} & false & true \\ L_{30} & W_{30,29} & W_{30,30} \\ L_{44} & false & true \end{array},$$

and:

- $W_{30,29}$ = "There is a model ready",
- $W_{30,30} = \neg W_{30,29}$.

The α-token that enters in the place L_{29} has the following characteristic: *"multi-class model ready"*.

The tokens β_6 and β_7 enter the net via places L_{31} and L_{32} with initial characteristics: "*values for slack variables* (ε_1,..., ε_n)" in place L_{31}, and "*values for an additional parameter* C" in place L_{32}.

The transition Z_7 has the form:

$$Z_7 = \langle\{L_{23}, L_{31}, L_{32}, L_{40}, L_{36}\}, \{L_{33}, L_{34}, L_{35}, L_{36}\}, R_7, \vee(\wedge(L_{23}, L_{31}, L_{32}, L_{40}), L_{36})\rangle,$$

where:

$$R_7 = \begin{array}{c|cccc} & L_{33} & L_{34} & L_{35} & L_{36} \\ \hline L_{23} & false & false & false & true \\ L_{31} & false & false & false & true \\ L_{32} & false & false & false & true \\ L_{36} & L_{36,33} & L_{36,34} & L_{36,35} & L_{36,36} \\ L_{40} & false & false & false & true \end{array},$$

and:

- $W_{36,33}$ = "The model is optimized",
- $W_{36,34}$ = "The model cannot be optimized and it needs to be mapped on higher dimension",
- $W_{36,35}$ = "The model is part of the multi-class problem (model)",
- $W_{36,36} = \neg(W_{36,33} \wedge W_{36,34} \wedge W_{36,35})$.

The α-tokens that enter in the places L_{33}, L_{34}, L_{35} have the following characteristics: "*optimized linear model for validation*" in place L_{33}, "*model for mapping in higher dimension*" in place L_{34}, and "*model for multi-class problem*" in place L_{35}.

The token β_8 enters the net via place L_{37} with initial characteristics: "*nonlinear φ-functions that map the input on a higher dimensional space and kernel trick (choosing the kernel*)".

The transition Z_8 has the form:

$$Z_8 = \langle\{L_{22}, L_{34}, L_{37}, L_{41}\}, \{L_{38}, L_{39}, L_{40}, L_{41}\}, R_8, \vee(L_{22}, L_{37}, L_{34}, L_{41})\rangle,$$

where:

$$R_8 = \begin{array}{c|cccc} & L_{38} & L_{39} & L_{40} & L_{41} \\ \hline L_{22} & false & false & false & true \\ L_{34} & false & false & false & true \\ L_{37} & false & false & false & true \\ L_{41} & W_{41,38} & W_{41,39} & W_{41,40} & W_{41,41} \end{array},$$

and:

- $W_{41,38}$ = "The nonlinear classifier is constructed",
- $W_{41,39}$ = "The nonlinear classifier for multi-class problem is constructed",

- $W_{41,40}$ = "The nonlinear classifier needs optimization",
- $W_{41,41} = \neg (W_{41,38} \wedge W_{41,39} \wedge W_{41,40})$.

The α-tokens that enters in the places L_{38}, L_{39} and L_{40} have the following characteristic: "*nonlinear model for validation*" in place L_{38}, "*nonlinear model for multi-class problem*" in place L_{39}, and "*nonlinear model for optimization*" in place L_{40}.

The transition Z_9 has the form:

$$Z_9 = \langle \{L_{25}, L_{29}, L_{35}, L_{39}, L_{45}\}, \{L_{42}, L_{43}, L_{44}, L_{45}\}, R_9, \vee(L_{25}, L_{29}, L_{35}, L_{39}, L_{45})\rangle,$$

where:

$$R_9 = \begin{array}{c|cccc} & L_{42} & L_{43} & L_{44} & L_{45} \\ \hline L_{25} & false & false & false & true \\ L_{29} & false & false & false & true \\ L_{35} & false & false & false & true \\ L_{39} & false & false & false & true \\ L_{45} & W_{45,42} & W_{45,43} & W_{45,44} & W_{45,45} \end{array},$$

and:

- $W_{45,42}$ = "The classifier have to be validated",
- $W_{45,43}$ = "The classifier have to be tested",
- $W_{45,44}$ = "The classifier is needed from additional processing",
- $W_{45,45} = \neg(W_{45,42} \wedge W_{45,43} \wedge W_{45,44})$.

The α-tokens that enter in places L_{42}, L_{43} and L_{44} have the following characteristic: "*multi-class model for validation*" in place L_{42}, "*multi-class model for testing*" in place L_{43}, and "*multi-class model for additional processing*" in place L_{44}.

The transition Z_{10} has the form:

$$Z_{10} = \langle \{L_{13}, L_{24}, L_{33}, L_{38}, L_{42}, L_{47}\}, \{L_{46}, L_{47}\}, R_{10}, \vee (L_{13}, L_{24}, L_{33}, L_{38}, L_{42}, L_{47})\rangle,$$

where:

$$R_{10} = \begin{array}{c|cc} & L_{46} & L_{47} \\ \hline L_{13} & false & true \\ L_{24} & false & true \\ L_{33} & false & true \\ L_{38} & false & true \\ L_{42} & false & true \\ L_{47} & W_{47,46} & W_{47,47} \end{array},$$

and:

- $W_{47,46}$ = "The classifier is validated",
- $W_{47,47} = \neg W_{47,46}$.

The α-token that enters in place L_{46} has the characteristic "*validated model*".

The transition Z_{11} has the form:

$$Z_{11} = \langle \{L_{12}, L_{46}, L_{49}\}, \{L_{48}, L_{49}\}, R_{11}, \vee (L_{12}, L_{46}, L_{49})\rangle,$$

where:

$$R_{11} = \begin{array}{c|cc} & L_{48} & L_{49} \\ \hline L_{12} & \mathit{false} & \mathit{true} \\ L_{46} & \mathit{false} & \mathit{true} \\ L_{49} & W_{49,48} & W_{49,49} \end{array},$$

and:

- $W_{49,48}$ = "The classifier is tested",
- $W_{49,49} = \neg\, W_{49,48}$.

The α-token that enters in place L_{48} has the characteristic "*tested model*".

3 Implementation of the algorithm

The implementation of the technique is made using weather datasets [3, 17–19]. The database includes following attributes: Wind, Temperature, Outlook, Humidity, Smoke detector, Infra-red detector and Date/Time. The interest represents the cases when the Infra-red detector has value "yes". Simple example for analysis based on weather data is given using RapidMiner software. Firstly, it is necessary to have a connection with the database. The preprocessing step is made using "Select Attributes" and "Nominal to Numerical" operators. The classification task demands to have a label attribute - this is the attribute "Infra-red detector". The following steps are generating a model and its validation.

The kernel model is presented on the Fig. 3. Its accuracy and performance is shown in Fig. 4 and Fig. 5.

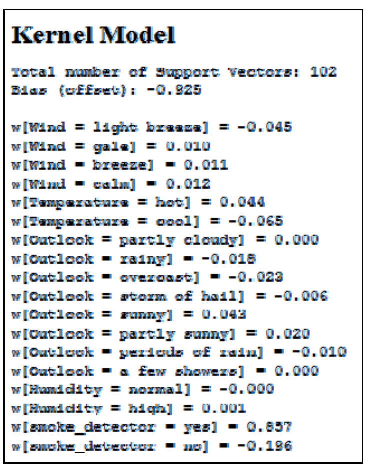

Figure 3. The kernel model

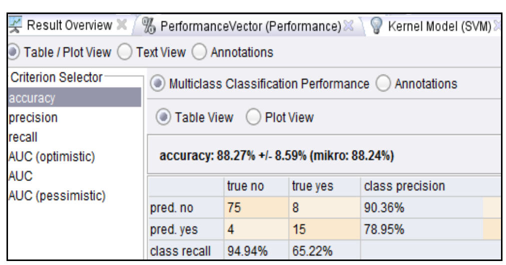

Figure 4. The accuracy of the model

PerformanceVector

```
PerformanceVector:
accuracy: 88.27% +/- 8.59% (mikro: 88.24%)
ConfusionMatrix:
True:   no      yes
no:     75      8
yes:    4       15
precision: 78.95% (positive class: yes)
ConfusionMatrix:
True:   no      yes
no:     75      8
yes:    4       15
recall: 65.00% +/- 32.87% (mikro: 65.22%) (positive class: yes)
ConfusionMatrix:
True:   no      yes
no:     75      8
yes:    4       15
AUC (optimistic): 0.873 +/- 0.158 (mikro: 0.873) (positive class: yes)
AUC: 0.871 +/- 0.161 (mikro: 0.871) (positive class: yes)
AUC (pessimistic): 0.869 +/- 0.164 (mikro: 0.869) (positive class: yes)
```

Figure 5. The performance of the model

4 Conclusion

In the presented paper is constructed a generalized net model for the process of classifying the datasets by support vector machine technique. The classifier is discussed in the cases of two classes and multi-class classification. The version(s) of linearly separable data and the non-linearly separable data is (are) explained. The extensions to the SVM model - soft margin classification and non-linear SVMs are presented. In addition to this article two generalized nets can be constructed: a generalized net for Multiclass SVMs algorithm or algorithms and a generalized net for the process of classification.

Acknowledgment

The first and the third authors are grateful for the support provided by the project DFNI-I-01/0006, funded by the National Science Fund, Bulgarian Ministry of Education, Youth and Science.

References

[1] Atanassov, K. Generalized Nets. World Scientific, Singapore, 1991.

[2] Atanassov, K. On Generalized Nets Theory. “Prof. M. Drinov” Academic Publishing House, Sofia, 2007.

[3] Bureva, V., E. Sotirova, Generalized net of the process of association rules discovery by Eclat algorithm using weather databases, 14th Int. Workshop on Generalized Nets Burgas, 29–30 November 2013, 1–10.

[4] Cortes, C., V. Vapnik, Support-Vector Networks, Machine Learning Kluwer Academic Publishers, Boston. Manufactured in The Netherlands, 20, 1995, 273-297.

[5] Duda, R., P. Hart, D. Stork, Pattern Classification, 2nd Edition, Wiley-Interscience, 2000.

[6] Friedman, J., T. Hastie, R. Tibshirani, The Elements of Statistical Learning: Data Mining, Inference and Prediction, Friedman, Hastie & Tibshirani, 2008.

[7] Gonen, M., E. Alpaydın, Multiple Kernel Learning Algorithms, Journal of Machine Learning Research 12 (2011), 2211-2268.

[8] Han, J., M. Kamber, Data Mining: Concepts and Techniques, Second Edition, Elsevier Inc, 2006.

[9] Support vector machines and machine learning on documents, http://nlp.stanford.edu/IR-book/html/htmledition/support-vector-machines-and-machine-learning-on-documents-1.html

[10] SVM – Support Vector Machines, http://www.support-vector-machines.org/.

[11] Izenman, J., Modern Multivariate Statistical Techniques: Regression, Classification and Manifold Learning, Springer, 2008.

[12] Kecman, V., Learning and Soft Computing. Support Vector Machines, Neural Networks and Fuzzy Logic Models, Massachusetts Institute of Technology, 2001.

[13] Olson, L., D. Delen, Advanced Data Mining Techniques, Springer-Verlag Berlin Heidelberg, 2008.

[14] Petkov T., S. Surchev, S. Sotirov, Generalized net model of the forest-fire detection with ART2 neural network, 15-th International Workshop on Generalized Nets, IWGN’2013,Burgas, 28-33.

[15] Radhika, Y., M. Shashi, Atmospheric Temperature Prediction using Support Vector Machines, International Journal of Computer Theory and Engineering, Vol. 1, No. 1, April 2009.

[16] Rani, R., R. Rao, An Enhanced Support Vector Regression Model for Weather Forecasting, IOSR Journal of Computer Engineering, Vol. 12, Issue 2, 2013, 21-24.

[17] Rao, T., N. Rajasekhar, T. Rajinikanth, An efficient approach for Weather forecasting using Support Vector Machines, 2012 Int. Conf. on Comp. Technology and Science, Vol. 47, IACSIT Press, Singapore, 2012.

[18] Sotirov S., I. Vardeva, M. Krawczak, Iintuitionistic fuzzy multilayer perceptron as a part of integrated systems for early forest-fire detection, NIFS, Vol. 19, N: 3, Sofia, 2013, 81-89.

[19] Statnikov, A., P. Aliferis, P. Hardin, I. Guyon, A gentle introduction to support vector machines in biomedicine. Vol. 1. Theory and Methods, World Scientific Publ. Co, 2011.

[20] Wang, L., X. Fu, Data Mining with Computational Intelligence, Springer-Verlag Berlin Heidelberg, Printed in Germany, 2005, ISBN-10 3-540-24522-7, ISBN-13 978-3-540-24522-3.