Discrimination of Outer Membrane Proteins using Reformulated Support Vector Machine based on Neutrosophic Set

Wen Ju and H. D. Cheng

wen.ju@aggiemail.usu.edu, hengda.cheng@usu.edu
Department of Computer Science, Utah State University, Logan, UT 84311-4205, U.S.A

Abstract

Neutrosophic logic is introduced in 1995 as a generalization of fuzzy logic. It includes a new component as neutralities. In this paper, we propose a novel neutronsophic set for SVM inputs and combine it with the reformulated SVM which treats samples differently according to the weighting function. The proposed classifier helps reducing the effects of outliers. We test it on discriminating outer membrane proteins (OMPs) from globular proteins and α-helical membrane proteins using amino acid composition and residue pair information. The experiment results show that the proposed method outperforms the traditional SVM in both classification accuracy and MCC.

Keywords: Neutrosophic set, reformulated support vector machine, outer membrane proteins (OMPs).

1. Introduction

Neutrosophic logic was introduced by Florentin Smarandache in 1995 as a generalization of fuzzy logic. It studies the neutrosophic logical values of the propositions. Each proposition is estimated to have three components: the percentage of truth in a subset *T,* the percentage of indeterminacy in a subset *I*, and the percentage of falsity in a subset *F* [1]. Compared with all other logics, neutrosophic logic introduces a percentage of "indeterminacy" due to unexpected parameters hidden in some propositions. The main distinction between neutrosophic logic (NL) and fuzzy logic (FL) is that the sum of neutrosophic components in NL is not necessarily 1 as in FL but any number from $^{-}0$ and 3^{+} [2].

Support Vector Machine (SVM) developed by Vapnik and Cortes has superior features such as avoiding over-fitting and obtaining global optimal [3]. It has been applied to many problems in bioinformatics. Kim and Park [4] used it to predict protein relative solvent accessibility. Nguyen and Rajapakse [5] applied SVM to predict protein secondary structures. It has also been applied to protein domains identification (Vlahovicek *et al.*, [6]), protein-protein binding sites prediction (Brandford and Westhead [7]), remote protein homology detection (Busuttil *et al.* [8]) and protein subcellular localization (Nair and Rost [9]).

Outer membrane proteins (OMPs) perform a variety of functions, such as selectively allowing the passage of molecules, mediating non-specific, passive transport of ions and small molecules [10]. Discriminating OMPs from globular proteins and α-helical membrane proteins is an important task both for dissecting OMPs from genomic sequences and for the successful prediction of their secondary and tertiary structures. Park *et al* used SVM to discriminate OMPs based on amino acid composition and residue pair information in [10].

In this paper, we propose a novel neutrosophic set for the input samples of SVM. Combining the neutrosophic set with the reformulated SVM, we discriminate OMPs from globular proteins and α-helical membrane proteins. We use the same dataset in [10], which is composed of 208 OMPs, 673 globular proteins and 206 α-helical membrane proteins. The experimental results show that the proposed method outperforms the traditional SVM in both accuracy and MCC.

The rest of the paper is organized as follows. Section 2 introduces the reformulated support vector machine. Section 3 describes the proposed neutrosophic set for SVM input and how the neutrosophic set is integrated into the reformulated SVM in detail. The experiment results are listed in section 4 and conclusions are drawn in section 5.

2. Reformulated Support Vector Machine

SVM uses hypothesis space of linear functions in a high-dimensional feature space, and it is trained with a learning algorithm based on optimization theory [11].

Suppose we are given a training set S containing n labeled points $(x_1, y_1), \ldots, (x_n, y_n)$, where $x_i \in R^N$ and $y_i \in \{-1, 1\}$, $i=1, \ldots, n$. $\Phi(x)$ denotes the mapping from R^N to a feature space Z. We want to find the hyperplane with maximum margin as:

$$w \cdot z + b = 0 \qquad (1)$$

such that for each point (z_i, y_i), where $z_i = \Phi(x_i)$,

$$y_i(w \cdot z_i + b) \geq 1, \quad i = 1, \ldots, n. \qquad (2)$$

When the data set is not linearly separable, the soft margin is allowed by introduction of n non-negative variables, denoted by $\xi = (\xi_{1,} \xi_{2,} \ldots \xi_n)$, such that the constraint for each sample in Eq. (2) is rewritten as:

$$y_i(w \cdot z_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n. \qquad (3)$$

The optimal hyperplane problem is the solution to the problem

$$\text{minimize} \frac{1}{2} w \cdot w + C \sum_{i=1}^{k} \xi_i \qquad (4)$$

subject to

$$y_i(w \cdot z_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n. \qquad (5)$$

where the first term in equation (4) measures the margin between support vectors and the second term measures the amount of misclassifications. C is a constant parameter that tunes the balance between the maximum margin and the minimum classification error.

Lin and Wang proposed fuzzy support vector machine in [12]. A membership s_i is assigned for each input sample (x_i, y_i), where $0 < s_i < 1$. Since the membership s_i is the attitude of the corresponding point x_i toward one class and the parameter ξ_i is a measure of error in the SVM, the term $s_i \xi_i$ is a measure of error with different weighting. The optimal hyperplane problem is then regarded as the solution to

$$\text{minimize } \frac{1}{2} w \cdot w + C \sum_{i=1}^{k} s_i \xi_i \qquad (6)$$

subject to

$$y_i(w \cdot z_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n. \qquad (7)$$

We use the similar idea in the reformulated SVM. The difference is that the membership s_i is substituted by weighting function g_i where $g_i > 0$. Different inputs contribute differently to the training procedure, and we use weighting function g_i to evaluate the degree of importance for each input. The value of g_i is a positive number and is unnecessary to be smaller than 1. Now the optimal hyperplane problem in the reformulated SVM is the solution to

$$\text{minimize } \frac{1}{2} w \cdot w + C \sum_{i=1}^{k} g_i \xi_i \quad (8)$$

subject to

$$y_i(w \cdot z_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n. \quad (9)$$

3. Integrating Neutrosophic Set with Reformulated SVM

3.1. Neutrosophic Set

Neutrosophic set is a generalization of the intuitionistic set, classical set, fuzzy set, paraconsistent set, dialetheist set, paradoxist set and tautological set [2].

In classical theory, there are only <A> and <Non-A>. The degree of neutralities <Neut-A> is introduced and added in neutrosophic theory. Generally a neutrosophic set is denoted as <*T, I, F*>. An element *x* (*t*, *i*, *f*) belongs to the set in the following way: it is *t* true in the set, *i* indeterminate in the set, and *f* false, where *t*, *i*, and *f* are real numbers taken from the sets *T*, *I*, and *F* with no restriction on *T*, *I*, *F*, nor on their sum *m=t+i+f.* The major difference between neutrosophic set (NS) and fuzzy set (FS) is that there is no limit on the sum *m* in NL while in FS *m* must be equal to 1.

3.2. Proposed Neutrosophic Set for SVM input

Many research results have shown that the SVM is very sensitive to noises and outliers. Here we propose a neutrosophic set for the input samples of SVM based on the distances between the sample and the class centers. The reformulated SVM integrated with the proposed neutrosophic set can help solving the problems of noise and outliers.

Using the same notations in section 2, the neutrosophic set for input samples are denoted as a sequence of points:

$$(x_j, y_j, t_j, i_j, f_j), \quad j = 1, \ldots, n.$$

Here for a sample x_j belongs to class y_j, it is t_j true, i_j indeterminate and f_j false. We define the center of positive samples C_+, the center of negative samples C_- and the center of all samples C_{all} as following:

$$C_+ = \sum_{k=1}^{n_+} x_k, C_- = \sum_{k=1}^{n_-} x_k, \qquad C_{all} = \sum_{k=1}^{n} x_k \quad (10)$$

where n_+ is the number of positive samples and n_- is the number of negative samples.

We denote *U* as the whole input samples set, *P* as the positive samples subset and *N* as the negative samples subset. For positive samples where $y_j = 1$, the neutrosophic components are defined as:

$$t_j = 1 - \frac{\|x_j - C_+\|}{\max_{x_k \in P} \|x_k - C_+\|}$$
$$i_j = 1 - \frac{\|x_j - C_{all}\|}{\max_{x_k \in U} \|x_k - C_{all}\|} \quad (11)$$
$$f_j = 1 - \frac{\|x_j - C_-\|}{\max_{x_k \in P} \|x_k - C_-\|}$$

where $\|x\|$ denotes the Euclidean distance of variable *x*.

For negative samples where $y_j = -1$, the neutrosophic components are defined as:

$$t_j = 1 - \frac{\|x_j - C_-\|}{\max_{x_k \in N} \|x_k - C_-\|}$$
$$i_j = 1 - \frac{\|x_j - C_{all}\|}{\max_{x_k \in U} \|x_k - C_{all}\|} \quad (12)$$
$$f_j = 1 - \frac{\|x_j - C_+\|}{\max_{x_k \in N} \|x_k - C_+\|}$$

With the above definition, every input sample is associated with a triple <t_j, i_j, f_j> as its neutrosophic components. The larger t_j it has, the more probability it belongs to the labeled class. The larger i_j it has, the more probability it is indeterminate. The larger f_j it has, the more probability it belongs to the opposite of the labeled class.

3.3. Integrating Neutrosophic Set with Reformulated SVM

In order to use the reformulated SVM, we should define a weighting function for input samples.

Following the steps in section 3.2, every sample has been associated with a triple <t_j, i_j, f_j> as its neutrosophic components. Larger t_j means the sample is nearer to the center of the labeled class and is less likely an outlier. So t_j should be emphasized in the weighting function. Larger i_j means the sample is harder to be discriminated between two classes. This factor should also be emphasized in the weighting function in order to classify the indeterminate samples more accurately. Larger f_j means the sample is more likely an outlier. This sample should be treated less importantly in the training procedure. Based on these analyses, we define the weighting function g_j as:

$$g_j = t_j + i_j - f_j \tag{13}$$

The reformulated SVM combined with the proposed weighting function treats samples differently in the training procedure and can help reducing the effects of outliers in the training samples.

4. Experimental Results

The same four measures in [10] are used to evaluate the classification performance. Sensitivity, specificity, overall accuracy and MCC are defined as

$$sensitivity = \frac{TP}{TP + FN} \tag{14}$$

$$specificity = \frac{TN}{TN + FP} \tag{15}$$

$$overall\ accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{16}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{17}$$

where TP, FP, TN and FN refer to the number of true positives, false positives, true negatives and false negatives proteins, respectively.

We use the same dataset and repeat all the experiments with the same parameter γ as stated in [10]. There are three categories of experiments: discrimination of OMPs and globular proteins, discrimination of OMPs and α-helical membrane proteins and discrimination of OMPs and non-OMPs. All of the experiments are results of 5-fold cross-validation test as in [10]. The results are listed in Tables 1, 2 and 3 respectively. The original results of [10] are listed in normal font while our results using FSVM are listed in Italic and Bold font with '***F***' as postfix.

Table 1. Discrimination of OMPs and globular proteins.

Input	Sen (%)	Spe (%)	Overall (%)	MCC
20D	82.7	93.8	91.1	0.757
20DF	***82.9***	***95.4***	***92.5***	***0.793***
400D	83.2	97.3	94.0	0.830
400DF	***85.1***	***97.7***	***95.2***	***0.863***
420D	63.5	100	91.4	0.755
420DF	***70.6***	***98.8***	***93.4***	***0.805***
17D	87.5	94.1	92.5	0.798
17DF	***91.6***	***95.3***	***92.9***	***0.829***

Table 2. Discrimination of OMPs and α-helical membrane proteins.

Input	Sen	Spe	Overall	MCC
	(%)	(%)	(%)	
20D	98.6	91.3	94.9	0.901
20DF	***98.8***	***92.9***	***95.7***	***0.913***
400D	99.0	90.3	94.7	0.897
400DF	***98.7***	***92.6***	***95.6***	***0.918***
420D	95.7	89.8	92.8	0.856
420DF	***95.1***	***92.9***	***93.8***	***0.881***
15D	99.0	92.7	95.9	0.920
15DF	***99.1***	***93.4***	***96.6***	***0.928***

Table 3. Discrimination of OMPs and non-OMPs.

Input	Sen	Spe	Overall	MCC
	(%)	(%)	(%)	
20D	87.5	92.6	91.6	0.752
20DF	***87.8***	***93.7***	***93.1***	***0.792***
400D	86.5	96.4	94.5	0.823
400DF	***86.8***	***97.2***	***94.9***	***0.848***
18D	89.9	92.5	92.0	0.767
18DF	***90.3***	***94.9***	***94.1***	***0.821***
28D	90.9	94.7	93.9	0.816
28DF	***89.7***	***96.3***	***94.8***	***0.831***

Due to the page settings, some notations are abbreviated in the tables. Sensitivity is truncated as *sen* while specificity is denoted by *spe*. The results show that the proposed method outperforms the traditional SVM in both overall classification accuracy and MCC. The increase of overall accuracy is 1%-2% and the MCC is increased by 3%-4% in most cases. The improvement is significant and adequately validates the correctness and effectiveness of the proposed method.

5. Conclusion

SVM is sensitive to outliers and noises in the input samples. In order to eliminate the effects of outliers, we integrate reformulated SVM with neutrosophic set derived from input samples. The reformulated SVM treats samples differently according to the weighting function in the training procedure. The weighting function is based on the neutrosophic set. We apply the proposed method to the discrimination of outer membrane proteins and compare the results with that of the traditional SVM. The experimental results have shown that the proposed classifier achieves higher accuracy and MCC than the traditional SVM method does.

6. References

[1] F. Smarandache, J. Dezert, A. Buller, M. Khoshnevisan, S. Bhattacharya, S. Singh, F. Liu, Gh. C. Dinulescu-Campina, C. Lucas, C.Gershenson, *Proceedings of the First International Conference on Neutrsophy, Neutrosophic Logic, Neutrosophic Set, Neutrosophic Probability and Statistics*, 2001.

[2] F. Smarandache, “A unifiying Filed in Logics: Neutrosophic Logic. Neutrosophy, Neutrosophic Set, Neutrosohpic Probability and Statistics,” third edition, Xiquan, Phoenix, 2003.

[3] V. Vapnik and C.Cortes, “Support vector networks”, *Machine learning*, vol. 20, pp. 273-293, 1995.

[4] H. Kim and H. Park, “Prediction of protein relative solvent accessibility with support vector machines and long-range interaction 2D local descriptor”, *Proteins*, vol. 54, pp. 557-562, 2004.

[5] M. N. Nguyen and J. C. Rajapakse, “Two-stage multi-class support vector machines to protein secondary structure prediction”, *Pac, Symp. Biocomput.*, pp. 346-357, 2005.

[6] K. Valhovicek, *et al.*, “Prediction of protein domain-architecture using

support vector machine”, *Nucleic Acids, Res.,* vol. 33, pp. 223-225, 2005.

[7] J. R. Bradford and D. R. Westhead, “Improved prediction of protein-protein binding sites using a support vector machines approach”, *Bioinformatics*, vol. 21, pp. 1487-1494, 2005.

[8] S. Busuttil, *et al*., “Support vector machines with profile-based kernels for remote protein homology detection”, *Genome Inform. Ser. Workshop Genome Inform.,* vol. 15, pp. 191-200, 2004.

[9] R. Nair and B. Rost, “Mimicking cellular sorting improves prediction of sub-cellular localization”, *J. Mol. Biol.,* vol. 348, pp. 85-100, 2005.

[10] K. Park, M. M. Gromiha, P. Horton and M. Suwa, “Discrimination of outer membrane proteins using support vector machines”, *Bioinformatics*, vol. 21, pp. 4223-4229, 2005.

[11] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*. Cambridge, U.K. Cambridge Univ. Press, 2000.

[12] C. F. Lin and S. D. Wang, “Fuzzy Support Vector Machines”, *IEEE transactions on Neural Networks*, vol. 13, pp.464-471, 2002.