

# Entropy-Based Support Matrix Machine

Changming Zhu<sup>(✉)</sup>

College of Information Engineering, Shanghai Maritime University,  
Shanghai 201306, People's Republic of China  
[cmzhu@shmtu.edu.cn](mailto:cmzhu@shmtu.edu.cn)

**Abstract.** Support Vector Machine (SVM) cannot process imbalanced problem and matrix patterns. Thus, Fuzzy SVM (FSVM) is proposed to process imbalanced problem while Support Matrix Machine (SMM) is proposed to process matrix patterns. FSVM applies a fuzzy membership to each training pattern such that different patterns can make different contributions to the learning machine. However, how to evaluate fuzzy membership becomes the key point to FSVM. Although SMM can process matrix patterns, it still has no ability to process imbalanced problem. This paper adopts SMM as the basic and proposes an entropy-based support matrix machine for imbalanced data sets, i.e., ESMM. The contributions of ESMM are: (1) proposing an entropy-based fuzzy membership evaluation approach which enhances importance of certainty patterns, (2) guaranteeing importance of positive patterns and getting a more flexible decision surface. Experiments on real-world imbalanced data sets and matrix patterns validate the effectiveness of ESMM.

**Keywords:** Support matrix machine · Entropy · Fuzzy membership · Imbalanced data set · Pattern recognition

## 1 Introduction

Support Vector Machine (SVM) constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier [1]. Conventional SVM can be used in many tasks including text and hypertext categorization, classification of images, classifying proteins in medical science, and recognizing hand-written characters. Although SVM has been validated effect on these applications, it still has two disadvantages. One is that SVM cannot process matrix patterns and another is that it cannot process imbalanced problem.

- Matrix patterns which dimensions are  $m \times n$  (where  $m$  and  $n$  are both larger than 1) are the basic of matrix learning. For example, video and images are both matrix patterns. In order to process matrix patterns, matrix-pattern-oriented

learning machine (MatC), i.e., matrix learning machine, has been developed. Classical learning machines include matrix-pattern-oriented Ho-Kashyap learning machine with regularization learning (MatMHKS) [2], new least squares support vector classification based on matrix patterns (MatLSSVC) [3], and one-class support vector machines based on matrix patterns (OCSVM) [4]. Besides those matrix learning machines, Xie et al. have proposed a Support Matrix Machine (SMM) [5] so as to replace SVM. SMM can leverage the structure of the data matrices and has the grouping effect property. But all of these matrix learning machines cannot process imbalanced problem.

- As is known to all, in many real-world classification problems, such as e-mail foldering [6], fault diagnosis [7], detection of oil spills [8], and medical diagnosis [9], we can always divide a data set into two classes, one is positive class and the other is negative class. When the size of positive class is much smaller than that of negative class, imbalanced problem occurs. Since most standard classification learning machines including Support Vector Machine (SVM) and Neural Network (NN) are proposed with the assumption on the balanced class distributions or equal misclassification costs [10], so they fail to properly represent the distributive characteristics of patterns and result in the unfavorable performance when they are adopted to process imbalanced problem. In order to overcome such a disadvantage, Fuzzy SVM (FSVM) [11] and Bilateral-weighted FSVM (B-FSVM) [12] are proposed. FSVM applies a fuzzy membership to each input pattern and reformulates SVM such that different input patterns can make different contributions to the learning of decision surface. B-FSVM treats every pattern as both positive and negative classes, but with different memberships due to we can not say one pattern belongs to one class absolutely. But for both of them, how to determine the fuzzy membership function is the key point. Furthermore, both of them cannot process matrix patterns.

In this paper, we try to propose a learning machine which can process matrix patterns and imbalanced problem. First, in order to process matrix patterns, we adopt SMM as a basic. Then in order to process imbalanced problem, we adopt the notion of FSVM, namely, applies a fuzzy membership to each input pattern. Furthermore, for the fuzzy membership, we propose a new fuzzy membership evaluation approach which assigns the fuzzy membership of each pattern based on its class certainty. In this paper, class certainty demonstrates the certainty of pattern labeled to a certain class. Due to the entropy is an effective measure of certainty, we adopt the entropy to evaluate the class certainty of each pattern. In doing so, the entropy-based fuzzy membership evaluation approach is proposed. This approach determines the fuzzy membership of training patterns based on their corresponding entropies. By adopting the entropy-based fuzzy membership evaluation and SMM, the Entropy-based Support Matrix Machine (ESMM) is proposed to process the imbalanced data sets. In practice, as the importance of positive class is higher than that of negative class in imbalanced data sets, i.e., the learning machine should pay more attention to positive patterns than negative ones. Thus, positive patterns are assigned to relatively large fuzzy memberships to guarantee the importance of positive class here. While, the

fuzzy memberships of negative patterns are determined by the entropy-based fuzzy membership evaluation approach, i.e., patterns with lower class certainty are assigned to small fuzzy memberships based on the criterion that patterns with lower class certainty are more insensitive to noise, and easily mislead the decision surface, thus their importance should be weakened on imbalanced data sets. After evaluating the fuzzy membership of all training patterns, ESMM is adopted to classify imbalanced data sets.

The contributions of this paper can be highlighted as follows:

- (1) A new entropy-based fuzzy membership evaluation approach is proposed. This approach adopts entropy to evaluate class certainty of a pattern and determines the corresponding fuzzy membership based on class certainty. In doing so, the learning machine can pay more attention to the patterns with higher class certainty to result in more robust decision surface.
- (2) To guarantee the importance of positive class, the positive patterns are assigned to the relatively large fuzzy memberships, which results in the decision surface paying more attention to the positive class so as to increase generalization of learning machine.

The rest of this paper is given below. Section 2 introduces the proposed entropy-based fuzzy membership evaluation approach, then give the details of ESMM. In Sect. 2.1, several experiments on real-world imbalanced data sets and matrix patterns including images are conducted to validate the effectiveness of ESMM. Following that, conclusions are given in Sect. 4.

## 2 Entropy-Based Matrix Learning Machine (ESMM)

### 2.1 Entropy-Based Fuzzy Membership

When we process imbalanced data sets, positive class is always more important than negative class. Thus, in the proposed entropy-based fuzzy membership evaluation approach, we assign positive patterns to a relatively high fuzzy membership, e.g., 1.0, to guarantee the importance of positive class. As to the negative class, we fuzzify negative patterns based on their class certainties.

In information theory, entropy is always used to characterize the certainty of source of information. If entropy is smaller, then information is more certain [13]. By employing this character of entropy, we can evaluate the class certainty of a pattern in the training class. After getting the class certainty of each pattern, we assign fuzzy membership of each training pattern based on class certainty. In practice, the patterns with higher class certainty, i.e., lower entropy, are assigned to higher fuzzy memberships to enhance their contributions to the decision surface, and vice versa.

Suppose that there are  $N$  training patterns,  $\{x_i, y_i\}$  where  $i = 1, 2, \dots, N$  and  $y_i \in \{+1, -1\}$  is the class label. When  $y_i = 1$ , pattern  $x_i$  belongs to the positive class, otherwise, it belongs to the negative class. The probabilities of  $x_i$  belonging to positive and negative class are  $p_{+i}$  and  $p_{-i}$  respectively. The entropy of  $x_i$  is  $H_i = -p_{+i}\ln(p_{+i}) - p_{-i}\ln(p_{-i})$  where  $\ln$  represents the natural logarithm

operator. Due to neighbors of a pattern can determine local information of it, thus the probability evaluation is based on its  $k$  nearest neighbors. For a pattern  $x_i$ , we select its  $k$  nearest neighbors  $\{x_{i1}, x_{i2}, \dots, x_{ik}\}$  at first. Then we count the number of both positive and negative class in these  $k$  selected patterns and denote the numbers of patterns belonging to positive and negative class are  $num_{+i}$  and  $num_{-i}$  respectively. Finally, the probabilities of  $x_i$  belonging to positive and negative class are calculated with  $p_{+i} = \frac{num_{+i}}{k}$  and  $p_{-i} = \frac{num_{-i}}{k}$ . After evaluating the class probabilities of  $x_i$ , we can calculate its entropy.

By adopting the above entropy evaluation approach, the entropy of the negative patterns are  $H = \{H_{-1}, H_{-2}, \dots, H_{-n_-}\}$ , where  $n_-$  is the number of the negative patterns.  $H_{min}$  and  $H_{max}$  are the minimum and maximum entropy of  $H$ . The entropy-based fuzzy memberships for negative patterns are evaluated as follows.

Firstly, separate the negative patterns into  $m$  subsets based on their entropy as described in Table 1, i.e.,  $\{Sub_k\}$  where  $k = 1, 2, \dots, m$ .

**Table 1.** Algorithm of negative class separation.

---

```

For k=1:m
  Up= $H_{max} - \frac{k-1}{m} (H_{max} - H_{min})$ 
  Low= $H_{max} - \frac{k}{m} (H_{max} - H_{min})$ 
  For i=1:n-
    if Low <  $H_{-i}$  ≤ Up
      negative pattern  $x_i$  is distributed into the subset  $Sub_k$ .
    End
  End
End

```

---

Then, fuzzy memberships of patterns in each subset are set as:

$$FM_k = 1.0 - \alpha \times (k - 1), \quad k = 1, 2, 3, \dots, m \quad (1)$$

where  $FM_k$  is the fuzzy membership for patterns distributed in subset  $Sub_k$ , the fuzzy membership parameter  $\alpha \in (0, \frac{1}{m-1})$  since  $FM_k$  is positive and not larger than 1.0. It should be declared that patterns in the same subset are set to same fuzzy membership so that these patterns selected in the same subsets have same importance to the decision surface. Finally, the fuzzy membership  $s_i$  for a training pattern  $x_i$  is assigned as: if  $y_i = +1$ , then  $s_i = 1.0$ , else if  $y_i = -1$  and  $x_i \in Sub_k$ , then  $s_i = FM_k$ . So far, the entropy-based fuzzy membership for the training patterns are evaluated.

## 2.2 Entropy-Based Support Matrix Machine

By adopting the evaluated entropy-based fuzzy membership which is given before, we propose the entropy-based support matrix machine (ESMM). The detailed description on ESMM is given below.

Suppose that there is a binary-class classification problem with  $N$  matrix patterns  $(A_i, y_i, s_i)$ ,  $i = 1, 2, \dots, N$ . Here  $A_i \in R^{m \times n}$  is the matrix representation of  $x_i$  and its class label is  $y_i \in \{+1, -1\}$ . If  $y_i = +1$ ,  $x_i$  or  $A_i$  belongs to class +1 or positive class, and then if  $y_i = -1$ , the pattern belongs to class -1 or negative class.  $s_i$  is the entropy-based fuzzy membership.

The corresponding criterion function of ESMM is defined below.

$$\begin{aligned} \min L(W, b) = & \frac{1}{2} \text{tr}(W^T W) + \theta \|W\|_* + \\ & C \sum_{i=1}^N (s_i (1 - y_i [\text{tr}(W^T A_i) + b_i])) \end{aligned} \quad (2)$$

where  $W \in R^{m \times n}$  is the matrix of regression coefficients and its nuclear norm is  $\|W\|_*$ .  $\theta$  is the coefficient.  $\text{tr}$  is the trace of matrix.  $b_i$  is a loose variable for pattern  $A_i$ .  $C$  ( $C \in R, C \geq 0$ ) is the regularization parameter that adjusts the trade-off between model complexity and training error. Here,  $b = [b_1, \dots, b_i, \dots, b_N]^T$  and  $b_i$  is started as  $b_i \geq 0$ . The iteration for  $b$  is given in Eq. (3).

$$b_i(k+1) = b_i(k) + \rho(e_i(k) + |e_i(k)|) \quad (3)$$

where the error vector  $e$  at  $k$ -th iteration of  $A_i$ , i.e.,  $e_i(k)$  should be  $W(k)^T A_i - 1 - b_i(k)$  and  $W(k)$  and  $b_i(k)$  are  $k$ -th component of  $W$  and  $b_i$ . Then we adopt the similar method of SMM to get the optimal  $W$  and  $b$  here. After that, we can get the discriminant function of ESMM as below.

$$g(A_i) = W^T A_i \quad (4)$$

If  $g(A_i) > 0$ , then we label  $A_i$  as a positive pattern, then if  $g(A_i) < 0$ , we label  $A_i$  as a negative pattern.

### 3 Experiments

In this section, we adopt 25 real-world imbalanced data sets and 5 image data sets for examples and the compared learning machines are SVM, FSVM, MatMHKS, B-FSVM, SMM. The used 25 real-world imbalanced data sets are selected from the KEEL imbalanced benchmark ones [14, 15]. Information of these data sets are given in Tables 2 and 3.

#### 3.1 Experimental Settings

In terms of the compared learning machines and ESMM, the experimental settings are given here. For the SVM-based learning machines, the used kernel is Radial Basis Function (RBF) kernel  $\text{ker}(x_i, x_j) = \exp(-\frac{\|x_i - x_j\|_2^2}{\sigma^2})$  where  $\sigma$  is selected from the set  $\{10^{-3}, 10^{-2}, \dots, 100, 1000\}$ . For MatMHKS, the experimental setting can be found in [2]. For SMM, the experimental setting can be found in [5]. For ESMM, its setting is similar with SMM. Moreover, for ESMM, the number of the separated subsets  $m = 10$  and the fuzzy membership parameter  $\alpha = 0.05$  which results in the fuzzy membership  $0.5 \leq s_i \leq 1.0$ . The reason of restricting  $s_i \in [0.5, 1.0]$  is that the class label of  $x_i$  should not be neglected when determining the fuzzy membership, i.e.,  $x_i$  is more likely to be classified to the class which is indicated by its class label. As to negative class patterns,

**Table 2.** Information of real-world imbalanced data sets (IR represents the imbalanced ratio of the corresponding data set).

Order	Data set	Attributes	Training	Testing	IR
1	Wisconsin	9	546	137	1.86
2	Haberman	3	245	61	2.78
3	Newthyroid2	5	172	43	5.14
4	Ecoli3	7	269	67	8.6
5	Ecoli046vs5	6	162	41	9.15
6	Yeast2vs4	8	411	103	9.08
7	Ecoli067vs5	6	176	44	10
8	Led7digit02456789vs1	7	354	89	10.97
9	Glass2	9	171	43	11.59
10	Yeast1vs7	7	367	92	14.3
11	Pageblocks13vs4	10	378	94	15.86
12	Glass016vs5	9	147	37	19.44
13	Yeast2vs8	8	386	96	23.1
14	Yeast1289vs7	8	758	189	30.57
15	Yeast6	8	1187	297	41.4
16	Pima	8	614	154	1.87
17	Ecoli1	7	269	67	3.36
18	Segment0	19	1846	462	6.02
19	Ecoli034vs5	7	160	40	9
20	Ecoli01vs235	7	195	49	9.17
21	Yeast05679vs4	8	422	106	9.35
22	Glass016vs2	9	154	38	10.29
23	Glass0146vs2	9	164	41	11.06
24	Cleveland0vs4	13	138	35	12.31
25	Ecoli4	7	269	67	15.8

**Table 3.** Information of image data sets. Imbalanced ratio of each image data set is 9.0.

Order	Data set	Attributes	Training	Testing
26	COIL-20 [16]	$32 \times 32$	1296	144
27	Letter [17]	$24 \times 18$	450	50
28	ORL [18]	$32 \times 20$	360	40
29	PubFig [19]	$74 \times 74$	52917	5880
30	Gaze [20]	$5184 \times 3456$	5292	588

we set  $s_i > 0.5$  to indicate that a negative pattern  $x_i$  is more likely to belong to the negative class. Moreover, the number of nearest neighbors  $k$  for calculating the class probability is selected from  $\{1, 2, 3, \dots, 18, 19, 20\}$ . In order to measure the performance of compared learning machines on imbalanced data sets, the values of Area Under the ROC Curve (AUC) [25] is given. Besides those,  $maxIter = 500$  represents the maximal size of the iteration. One-against-one classification strategy is used for multi-class problems here [21–24]. The 10-fold cross validation approach [26] is adopted for the parameter selection. The computations are performed on Intel Core 2 processors with 2.66 GHz, 8G RAM, Microsoft Windows 7, and Matlab environment.

### 3.2 Experiments on Real-World Imbalanced Data Sets

For experiments, the AUC values are presented in Table 4. The average AUC on all used data sets are presented. Moreover, the average ranks of the learning machines on the used data sets are listed.

**Table 4.** AUC values (%) of the compared learning machines on real-world imbalanced data sets and image data sets. The best  $k$  for each data sets of ESMM is presented. (Note that the average AUC values and the average ranks of the compared learning machines are listed in the last two rows. Best result for each data set is in bold.)

Data set	k	ESMM	SMM	MatMHKS	B-FSVM	FSVM	SVM
1	9	94.05 $\pm$ 2.37	94.02 $\pm$ 2.30	89.98 $\pm$ 3.33	91.30 $\pm$ 3.03	<b>96.10 <math>\pm</math> 2.41</b>	93.64 $\pm$ 3.67
2	13	64.42 $\pm$ 1.96	64.47 $\pm$ 1.98	<b>67.06 <math>\pm</math> 1.40</b>	61.58 $\pm$ 1.44	58.00 $\pm$ 2.05	54.08 $\pm$ 1.65
3	11	93.28 $\pm$ 1.87	91.47 $\pm$ 1.89	89.78 $\pm$ 2.57	92.44 $\pm$ 2.32	93.43 $\pm$ 2.33	<b>97.12 <math>\pm</math> 2.90</b>
4	17	87.66 $\pm$ 2.41	87.37 $\pm$ 2.41	93.67 $\pm$ 2.70	<b>94.55 <math>\pm</math> 2.54</b>	76.83 $\pm$ 2.36	65.74 $\pm$ 1.80
5	14	<b>98.61 <math>\pm</math> 2.73</b>	97.15 $\pm$ 2.74	92.59 $\pm$ 2.91	89.07 $\pm$ 3.15	86.61 $\pm$ 1.06	82.43 $\pm$ 1.91
6	7	93.84 $\pm$ 2.95	<b>94.33 <math>\pm</math> 2.94</b>	91.79 $\pm$ 1.47	87.37 $\pm$ 2.11	87.32 $\pm$ 1.85	83.77 $\pm$ 2.14
7	5	97.12 $\pm$ 2.94	<b>98.67 <math>\pm</math> 2.87</b>	90.79 $\pm$ 1.71	94.39 $\pm$ 1.97	78.90 $\pm$ 2.28	74.15 $\pm$ 2.52
8	17	92.80 $\pm$ 2.62	90.53 $\pm$ 2.58	88.30 $\pm$ 2.29	<b>93.43 <math>\pm</math> 2.35</b>	87.66 $\pm$ 2.14	81.64 $\pm$ 2.87
9	18	<b>81.33 <math>\pm</math> 2.22</b>	80.07 $\pm$ 2.20	68.12 $\pm$ 1.66	66.99 $\pm$ 1.74	69.21 $\pm$ 2.09	51.76 $\pm$ 1.51
10	18	<b>78.41 <math>\pm</math> 2.53</b>	76.58 $\pm$ 2.50	58.24 $\pm$ 1.95	68.95 $\pm$ 1.85	61.56 $\pm$ 2.14	67.10 $\pm$ 2.20
11	15	79.86 $\pm$ 2.36	79.29 $\pm$ 2.29	77.65 $\pm$ 1.93	80.99 $\pm$ 2.37	<b>92.11 <math>\pm</math> 2.67</b>	91.22 $\pm$ 3.32
12	14	<b>87.24 <math>\pm</math> 2.27</b>	85.75 $\pm$ 2.21	73.53 $\pm$ 2.07	80.66 $\pm$ 1.91	74.85 $\pm$ 2.18	75.31 $\pm$ 1.74
13	15	<b>71.30 <math>\pm</math> 1.67</b>	69.28 $\pm$ 1.69	67.12 $\pm$ 1.55	56.86 $\pm$ 1.20	57.62 $\pm$ 1.87	62.45 $\pm$ 1.52
14	10	91.00 $\pm$ 2.24	91.62 $\pm$ 2.22	<b>92.81 <math>\pm</math> 3.17</b>	87.97 $\pm$ 2.80	71.63 $\pm$ 1.54	71.64 $\pm$ 2.13
15	5	72.58 $\pm$ 1.31	<b>72.84 <math>\pm</math> 1.32</b>	71.15 $\pm$ 2.13	69.81 $\pm$ 2.38	68.65 $\pm$ 1.53	71.50 $\pm$ 1.75
16	3	90.36 $\pm$ 1.72	89.80 $\pm$ 1.72	81.80 $\pm$ 2.18	<b>93.74 <math>\pm</math> 2.44</b>	85.40 $\pm$ 2.59	78.11 $\pm$ 2.36
17	10	80.62 $\pm$ 2.19	78.88 $\pm$ 2.20	82.32 $\pm$ 2.82	83.40 $\pm$ 2.01	98.24 $\pm$ 2.03	<b>99.72 <math>\pm</math> 2.25</b>
18	6	<b>93.52 <math>\pm</math> 3.00</b>	91.50 $\pm$ 2.89	87.46 $\pm$ 1.91	83.81 $\pm$ 1.88	89.61 $\pm$ 2.98	77.26 $\pm$ 2.20
19	18	92.35 $\pm$ 2.69	<b>93.72 <math>\pm</math> 2.73</b>	92.22 $\pm$ 3.10	91.20 $\pm$ 2.19	74.69 $\pm$ 2.30	82.43 $\pm$ 2.18
20	19	<b>83.26 <math>\pm</math> 1.96</b>	81.00 $\pm$ 1.99	78.59 $\pm$ 2.28	79.42 $\pm$ 2.16	64.48 $\pm$ 2.02	66.34 $\pm$ 1.74
21	5	<b>75.38 <math>\pm</math> 2.08</b>	75.26 $\pm$ 2.06	69.36 $\pm$ 2.09	55.51 $\pm$ 1.57	61.42 $\pm$ 1.84	53.73 $\pm$ 1.52
22	16	91.84 $\pm$ 2.19	91.08 $\pm$ 2.18	85.21 $\pm$ 2.01	<b>95.55 <math>\pm</math> 3.40</b>	87.28 $\pm$ 2.53	94.66 $\pm$ 2.85
23	4	<b>81.49 <math>\pm</math> 2.00</b>	80.37 $\pm$ 2.00	59.32 $\pm$ 1.60	71.62 $\pm$ 2.15	66.51 $\pm$ 2.05	55.52 $\pm$ 1.35
24	1	74.72 $\pm$ 2.07	74.44 $\pm$ 2.01	63.24 $\pm$ 1.72	60.41 $\pm$ 1.21	68.05 $\pm$ 1.76	<b>77.90 <math>\pm</math> 1.86</b>
25	12	95.59 $\pm$ 2.79	<b>96.65 <math>\pm</math> 2.81</b>	95.98 $\pm$ 2.51	92.53 $\pm$ 2.80	84.58 $\pm$ 1.72	77.94 $\pm$ 2.37
26	5	65.40 $\pm$ 1.78	64.90 $\pm$ 1.74	59.33 $\pm$ 1.97	<b>71.77 <math>\pm</math> 1.76</b>	71.07 $\pm$ 1.76	69.50 $\pm$ 2.37
27	18	<b>66.08 <math>\pm</math> 2.15</b>	65.88 $\pm$ 2.15	57.58 $\pm$ 1.01	60.62 $\pm$ 1.50	49.37 $\pm$ 1.45	48.94 $\pm$ 1.32
28	20	<b>90.19 <math>\pm</math> 2.06</b>	88.54 $\pm$ 2.01	86.21 $\pm$ 3.07	79.27 $\pm$ 2.02	79.82 $\pm$ 1.71	82.30 $\pm$ 2.10
29	8	92.89 $\pm$ 2.01	93.59 $\pm$ 1.99	88.12 $\pm$ 2.36	<b>95.14 <math>\pm</math> 3.19</b>	89.48 $\pm$ 2.26	81.30 $\pm$ 2.45
30	18	<b>94.78 <math>\pm</math> 1.70</b>	92.15 $\pm$ 1.68	90.61 $\pm$ 2.56	87.60 $\pm$ 2.02	66.13 $\pm$ 1.23	85.69 $\pm$ 2.21
AVG	11.72	<b>85.06 <math>\pm</math> 2.23</b>	84.37 $\pm$ 2.21	79.66 $\pm$ 2.20	80.60 $\pm$ 2.18	76.55 $\pm$ 2.02	75.16 $\pm$ 2.16
AVG.rank		2.07	2.53	4.03	3.50	4.30	4.57

From the experimental results, it is found that: (1) ESMM results in the best classification performance on 12 of 30 imbalanced data sets which indicates that the our proposal outperforms the compared learning machines; (2) ESMM outperforms the traditional MatMHKS on 26 of 30 imbalanced data sets; (3) The average AUC of ESMM is respectively greater than B-FSVM, FSVM, and SVM on about 5%, 9%, and 10%, which demonstrates that ESMM is of significant advantage in processing imbalanced data sets compared to SVM, FSVM, and B-FSVM; (4) It is found that for some data sets, for example, COIL-20, the three matrix-pattern-oriented approaches perform worse than SVM. We think it is the coincidence due to the training part is gotten in random. But according to the average result, we still find that our proposed ESMM outperforms others in average; (5) For the used 25 vector data sets, ESMM performs best on the 9 data sets of them. For others, ESMM performs better in average and it does not perform worst on any vector data set. This phenomenon can explain the superiority of the ESMM on the vector data sets; (6) As we said before, ESMM and FSVM both adopt fuzzy membership to each input pattern. Now from the experiments, it is found that compared with FSVM, the better performance of ESMM validates that the proposed entropy-based fuzzy membership evaluation approach boosts the performance of a learning machine.

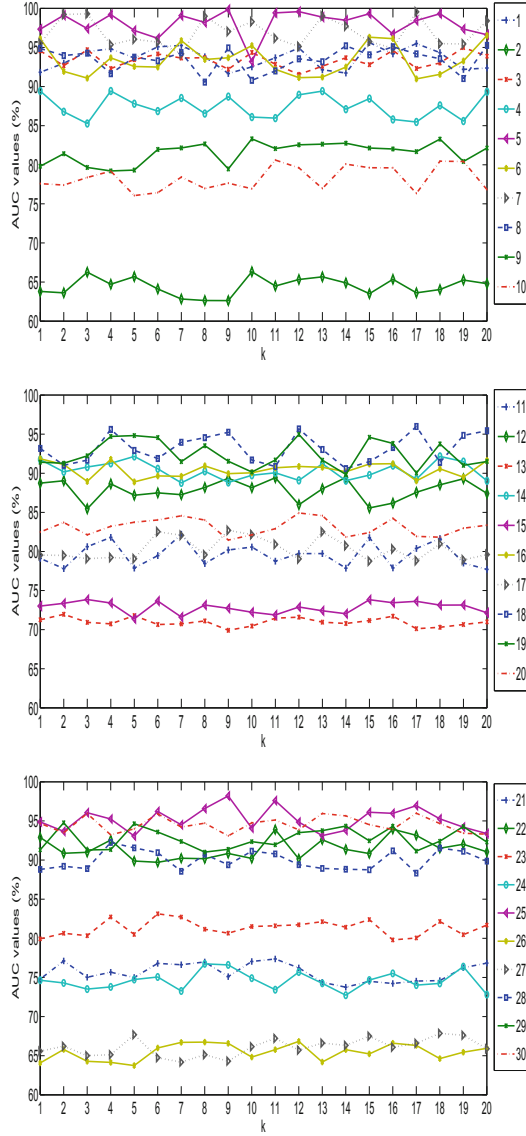
### 3.3 Influence of Parameter $k$ on the Performance of ESMM

In ESMM, the entropy-based fuzzy membership is evaluated based on the class probability of each training pattern. Thus, the number of nearest neighbors  $k$  might have some influence on the class probability. To further investigate the effectiveness of ESMM, here, we study the influence of  $k$  on the classification performance. All data sets given in Table 2 are used and related experimental results are given in Fig. 1. The figure shows AUC on the testing sets of the adopted real-world imbalanced data sets and image data sets with respect to  $k$ . It is found that: (1) the number of the selected nearest neighbors for calculating the class probability, i.e.,  $k$ , has some influence on the classification performance since AUC curves fluctuate with respect to  $k$  on most data sets; (2) on some data sets, the classification performances are sensitive to the variation of  $k$ , since AUC curves on these data sets fluctuate greatly while on some data sets, are not; (3) in generally, with  $k$  from 8 to 12, ESMM always gets best performance. Such a result can give us a guidance that how to determine an appropriate  $k$  in practical use.

### 3.4 Comparison Between ESMM and Entropy-Based MatMHKS (EMatMHKS)

Here, we give the comparison between ESMM and our previous proposed learning machine, entropy-based MatMHKS (EMatMHKS) [27] which is also an entropy-based matrix learning machine. Table 5 shows the comparison between ESMM and EMatMHKS.  $\uparrow$  ( $\downarrow$ ) represents that ESMM performance better (worse) than





**Fig. 1.** Variation of AUC values of ESMM with respect to  $k$  on the adopted imbalanced data sets and image data sets. Here, in the legend, each number denotes one data set which is given in Tables 2 or 3.

EMatMHKS and  $(\star)$  represents that the AUC value of ESMM is  $\star$  larger or smaller than the one of EMatMHKS. From this table, we find that ESMM performs better than EMatMHKS in average.

**Table 5.** Comparison between ESMM and EMatMHKS on AUC values (%).

Data set		Data set		Data set		Data set		Data set		Data set	
1	↑(1.02)	2	↑(0.03)	3	↑(2.11)	4	↑(0.02)	5	↑(1.01)	6	↑(0.78)
7	↓(0.23)	8	↑(1.02)	9	↑(2.98)	10	↑(6.18)	11	↓(0.03)	12	↑(3.91)
13	↑(0.82)	14	↑(0.56)	15	↑(0.81)	16	↑(2.87)	17	↓(0.12)	18	↑(1.98)
19	↑(0.04)	20	↑(1.23)	21	↑(3.12)	22	↑(2.91)	23	↑(8.19)	24	↑(2.12)
25	↑(0.01)	26	↑(2.01)	27	↑(2.10)	28	↑(1.81)	29	↑(1.07)	30	↑(0.91)

## 4 Conclusion

There are two hot spots of present research, one is imbalanced problem and the other is matrix learning. Imbalanced problem occurs when the size of positive class is more smaller than that of negative class. However, most standard classification learning machines result in unfavorable performance on imbalanced data sets since they are originally designed for processing balanced problems. Although SVM can process imbalanced data sets in some extent, it assigns the same importance to each training pattern. This results in the decision surfaces biasing toward the negative class. In order to overcome the disadvantage of SVM, some researchers propose FSVM and B-FSVM by applying fuzzy memberships to the training patterns to reflect different importance of them. Since the key point in FSVM and B-FSVM is how to determine the fuzzy membership, so this paper presents an entropy-based fuzzy membership evaluation approach for imbalanced data sets. Moreover, matrix patterns cannot be solved by those traditional learning machines including SVM well, so some scholars have developed MatMHKS and SMM. This paper adopts the entropy-based fuzzy membership and SMM, and then proposes ESMM. ESMM can not only guarantee importance of the positive class, but also pay more attentions to the patterns with higher class certainties. Thus, ESMM can results in more flexible decision surfaces than both conventional SVM, FSVM, B-FSVM, and MatMHKS on the imbalanced data sets. To validate the effectiveness of ESMM, we adopt 25 real-world imbalanced data sets and 5 image data sets for experiments. Experimental results demonstrates that ESMM outperforms the compared learning machines on real-world imbalanced data sets and the images. Moreover, in the process of ESMM,  $k$  has some influence on the classification performance.

**Acknowledgment.** This work is supported by Natural Science Foundation of Shanghai under grant number 16ZR1414500 and National Natural Science Foundation of China under grant number 61602296 and 51575336, and the authors would like to thank their supports.

## References

1. Support Vector Machine. [https://en.wikipedia.org/wiki/Support\\_vector\\_machine#Applications](https://en.wikipedia.org/wiki/Support_vector_machine#Applications)
2. Chen, S.C., Wang, Z., Tian, Y.J.: Matrix-pattern-oriented Ho-Kashyap classifier with regularization learning. *Pattern Recogn.* **40**(5), 533–1543 (2007)

3. Wang, Z., Chen, S.C.: New least squares support vector machines based on matrix patterns. *Neural Process. Lett.* **26**(1), 41–56 (2007)
4. Yan, Y., Wang, Q., Ni, G., Pan, Z., Kong, R.: One-class support vector machines based on matrix patterns. In: Jiang, L. (ed.) *Proceedings of the 2011, International Conference on Informatics, Cybernetics, and Computer Engineering (ICCE2011)*. AINSC, pp. 223–231. Springer, Heidelberg (2011). doi:[10.1007/978-3-642-25188-7\\_27](https://doi.org/10.1007/978-3-642-25188-7_27)
5. Xie, Y.B., Zhang, Z.H., Li, W.J.: Support Matrix Machines. In: *The 32nd International Conference on Machine Learning (ICML 2015)*
6. Dai, H.: Class Imbalance Learn. Fuzzy Total Margin Based Support Vector Mach. *Appl. Soft Comput.* **31**, 172–184 (2015)
7. Deng, X., Tian, X.: Nonlinear process fault pattern recognition using statistics kernel PCA similarity factor. *Neurocomputing.* **121**, 298–308 (2013)
8. Guo, Y., Zhang, H.Z.: Oil spill detection using synthetic aperture radar images and feature selection in shape space. *Int. J. Appl. Earth Obs. Geoinf.* **30**, 146–157 (2014)
9. Ozcift, A., Gulten, A.: Classifier ensemble construction with rotation forest to improve medical diagnosis performance of machine learning algorithms. *Comput. Methods Programs Biomed.* **104**(3), 443–451 (2011)
10. Brown, I., Mues, C.: An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Syst. Appl.* **39**(3), 3446–3453 (2012)
11. Lin, C., Wang, S.: Fuzzy support vector machines. *IEEE Trans. Neural Netw.* **13**(2), 464–471 (2002)
12. Wang, Y., Wang, S., Lai, K.K.: A new fuzzy support vector machine to evaluate credit risk. *IEEE Trans. Fuzzy Syst.* **13**(6), 820–831 (2005)
13. Shannon, C.E.: A mathematical theory of communication. *SIGMOBILE Mob. Comput. Commun. Rev.* **5**(1), 3–55 (2001)
14. Alcala-Fdez, J., Fernandez, A., Luengo, J., Derrac, J., Garcia, S., Sanchez, L., Herrera, F.: A software tool to assess evolutionary algorithms for data mining problems. *J. Multiple-Valued Logic Soft Comput.* **17**, 2–3 (2011)
15. Alcala-Fdez, J., Sanchez, L., Garcia, S., Jesus, M.J., Ventura, S., Garrell, J.M., Otero, J., Romero, C., Bacardit, J., Rivas, V.M.: A software tool to assess evolutionary algorithms for data mining problems. *Soft. Comput.* **13**(3), 307–318 (2009)
16. Nene, S.A., Nayar, S.K., Murase, H.: Columbia object image library (COIL-20). Technical report CUCS-005-96. Columbia University (1996)
17. Cun, B.B.L., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Handwritten digit recognition with a back-propagation network. In: *Advances in neural information processing systems* (1990)
18. Bennett, F., Richardson, T., Harter, A.: Teleporting-making applications mobile. In: *Mobile Computing Systems and Applications*, pp. 82–84 (1994)
19. Kumar, N., Berg, A.C., Belhumeur, P.N., Nayar, S.K.: Attribute and simile classifiers for face verification. In: *International Conference on Computer Vision*, pp. 365–372 (2009)
20. Smith, B.A., Yin, Q., Feiner, S.K., Nayar, S.K.: Gaze locking, passive eye contact detection for human-object interaction. In: *ACM Symposium on User Interface Software and Technology (UIST)*, pp. 271–280 (2013)
21. Milgram, J., Cheriet, M., Sabourin, R.: “One Against One” or “One Against All”, Which One is Better for Handwriting Recognition with SVMs? In: *Tenth International Workshop on Frontiers in Handwriting Recognition* (2013)

22. Debnath, R., Takahide, N., Takahashi, H.: A decision based one-against-one method for multi-class support vector machine. *Pattern Anal. Appl.* **7**(2), 164–175 (2004)
23. Hsu, C.W., Lin, C.J.: A comparison of methods for multiclass support vector machines. *IEEE Trans. Neural Netw.* **13**(2), 415–425 (2002)
24. Cortes, C., Vapnik, V.: Support vector machine. *Mach. Learn.* **20**(3), 273–297 (1995)
25. Huang, J., Ling, C.X.: Using AUC and accuracy in evaluating learning algorithms. *IEEE Trans. Knowl. Data Eng.* **17**(3), 299–310 (2005)
26. Braga-Neto, U.M., Dougherty, E.R.: Is cross-validation valid for small-sample microarray classification? *Bioinformatics* **20**(3), 374–380 (2004)
27. Zhu, C.M., Wang, Z.: Entropy-based matrix learning machine for imbalanced data sets. *Pattern Recogn. Lett.* **88**(1), 72–80 (2017)