

# *k*-Times Markov Sampling for SVMC

Bin Zou, Chen Xu, Yang Lu, Yuan Yan Tang, *Fellow, IEEE*, Jie Xu, and Xinge You, *Senior Member, IEEE*

**Abstract**—Support vector machine (SVM) is one of the most widely used learning algorithms for classification problems. Although SVM has good performance in practical applications, it has high algorithmic complexity as the size of training samples is large. In this paper, we introduce SVM classification (SVMC) algorithm based on *k*-times Markov sampling and present the numerical studies on the learning performance of SVMC with *k*-times Markov sampling for benchmark data sets. The experimental results show that the SVMC algorithm with *k*-times Markov sampling not only have smaller misclassification rates, less time of sampling and training, but also the obtained classifier is more sparse compared with the classical SVMC and the previously known SVMC algorithm based on Markov sampling. We also give some discussions on the performance of SVMC with *k*-times Markov sampling for the case of unbalanced training samples and large-scale training samples.

**Index Terms**—*k*-times Markov sampling, learning performance, support vector machine classification (SVMC), uniform ergodic Markov chain (u.e.M.c.).

## I. INTRODUCTION

SUPPORT vector machine (SVM) is one of the most widely used learning algorithms for pattern recognition problems [1]. Besides its good performance in practical applications, SVM classification (SVMC) also has a good theoretical property in universal consistency [2]–[5] and learning rates [4], [5] if the training samples come from an independent and identically distributed (i.i.d.) process. Since independence is a very restrictive concept [6]–[14], such i.i.d. assumption cannot be strictly validated in real-world problems. For example, many machine learning

Manuscript received December 15, 2015; revised May 20, 2016; accepted September 6, 2016. This work was supported in part by NSFC Project under Grant 61403132, Grant 61370002, and Grant 61272203, in part by Hubei Key Laboratory of Applied Mathematics, Hubei University, Hubei Province Technologies RD Program under Grant XYJ2014000459, in part by the Natural Sciences and Engineering Research Council of Canada under Grant RGPIN-2016-05024, in part by the Shenzhen Research Council under Grant CXZZ20150814155434903 and Grant JCYJ20140819154343378, and in part by NFS of Hubei Provience under Grant 2015CFB404. (*Corresponding Author: Jie Xu*)

B. Zou is with the Faculty of Mathematics and Statistics, Hubei University, Wuhan 430062, China (e-mail: zoubin0502@gmail.com).

C. Xu is with the Department of Mathematics and Statistics, University of Ottawa, Ottawa K1N 6N5, Canada (e-mail: cx3@uottawa.ca).

Y. Lu is with the Faculty of Science, Hong Kong Baptist University, Hong Kong (e-mail: lylylyte@gmail.com).

Y. Y. Tang is with the Faculty of Science and Technology, University of Macau, Macau 999078, China (e-mail: yytang@umac.mo).

J. Xu is with the Faculty of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China (e-mail: jie xu027@gmail.com).

X. You is with the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: youxg@mail.hust.edu.cn).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TNNLS.2016.2609441

applications, such as market prediction, system diagnosis, and speech recognition, are inherently temporal in nature, and consequently not i.i.d. processes [7]. Therefore, the relaxations of such i.i.d. assumption for SVMC have to be considered. Zou *et al.* [15] studied the generalization ability of SVMC with uniformly ergodic Markov chain (u.e.M.c.) samples, but the obtained learning rate is not optimal. Xu *et al.* [16] established the optimal learning rate of Gaussian kernels SVMC with u.e.M.c. samples by using the strongly mixing property of Markov chain. Xu *et al.* [17] obtained the optimal learning rate of SVMC with u.e.M.c. samples and presented the numerical studies on the performance of SVMC with Markov sampling. Although the SVMC with Markov sampling introduced in [17] has smaller misclassification rates, its total time of sampling and training is longer compared with the classical SVMC based on randomly independent sampling. Thus, a problem is posed: *How to reduce the sampling and training time of SVMC with Markov sampling introduced in [17], at the same time keeping its smaller classification rates?*

To answer this problem and to improve the learning performance of the classical SVMC, in this paper, we introduce SVMC algorithm based on *k*-times Markov sampling and present the numerical studies on the learning performance of SVMC with *k*-times Markov sampling for benchmark data sets. We compare the SVMC based on *k*-times Markov sampling with the classical SVMC and the SVMC based on Markov sampling introduced in [17]. These comparisons show that the SVMC with *k*-times (*k* = 1, 2), Markov sampling has three advantages at the same time compared with the classical SVMC and the SVMC with Markov sampling in [17]: 1) the misclassification rates are smaller; 2) the total time of sampling and training is less; and 3) the obtained classifiers are more sparse. To have a better showing the performance of SVMC with *k*-times Markov sampling, we also give some discussions for the cases of unbalanced training samples and large-scale training samples.

This paper is organized as follows. In Section II, we introduce some notions and notations used in this paper. In Section III, we introduce SVMC with *k*-times Markov sampling. Section IV compares the SVMC based on *k*-times Markov sampling with the classical SVMC and the SVMC based on Markov sampling introduced in [17]. Section V explains the learning performance of SVMC with *k*-times Markov sampling. In Section VI, we give some discussions on the learning ability of SVMC with *k*-times Markov sampling for the cases of unbalanced training samples and large-scale training samples. Finally, we conclude this paper in Section VII.

## II. PRELIMINARIES

In this section, we present some definitions and notations used in this paper.

### A. SVM Classification Algorithm

Let  $(\mathcal{X}, d)$  be a compact metric space and  $\mathcal{Y} = \{-1, 1\}$ . A binary classifier is a function  $h : \mathcal{X} \rightarrow \mathcal{Y}$  which labels every point  $x \in \mathcal{X}$  with some  $y \in \mathcal{Y}$ . Let  $\varphi$  be a probability distribution on  $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$  and  $(X, Y)$  be the corresponding random variable. The misclassification error for a classifier  $h : \mathcal{X} \rightarrow \mathcal{Y}$  is defined to be the probability of the event  $\{h(X) \neq Y\}$ , that is,  $\mathcal{R}(h) = P\{h(X) \neq Y\}$ . The SVM classifier [1] is constructed from samples and depends on a reproducing kernel Hilbert space (RKHS) associated with a Mercer kernel  $K$  [20].

The RKHS  $\mathcal{H}_K$  associated with the kernel  $K$  is defined to be the closure of the linear span of the set of functions  $\{K_x = K(x, \cdot) : x \in \mathcal{X}\}$  with the inner product  $\langle \cdot, \cdot \rangle_{\mathcal{H}_K} = \langle \cdot, \cdot \rangle_K$  satisfying  $\langle K_{x_i}, K_{x_j} \rangle_K = K(x_i, x_j)$ ,  $\langle \sum_i \alpha_i K_{x_i}, \sum_j \beta_j K_{x_j} \rangle_K = \sum_{i,j} \alpha_i \beta_j K(x_i, x_j)$  [19]. Denote  $\mathcal{C}(\mathcal{X})$  as the space of continuous functions on  $\mathcal{X}$  with the norm  $\|f\|_\infty = \sup_{x \in \mathcal{X}} |f(x)|$ . Let  $\kappa = \sup_{x \in \mathcal{X}} \sqrt{K(x, x)}$ , then the above reproducing property tells us that  $\|f\|_\infty \leq \kappa \|f\|_K$ ,  $\forall f \in \mathcal{H}_K$ . For a function  $f : \mathcal{X} \rightarrow \mathbb{R}$ , the sign function is defined as  $\text{sign}[f(x)] = 1$  if  $f(x) \geq 0$  and  $\text{sign}[f(x)] = -1$  if  $f(x) < 0$ . Then, the SVM classifier associated with the kernel  $K$  is defined as  $\text{sign}(f_S)$ , where  $f_S$  is a minimizer of the following optimization problem involving a sample set  $S = \{z_i\}_{i=1}^m$ :

$$\begin{aligned} f_S = \arg \min_{f \in \mathcal{H}_K} & \frac{1}{2} \|f\|_K^2 + \frac{C}{m} \sum_{i=1}^m \xi_i \\ \text{s.t. } & y_i f(x_i) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad 1 \leq i \leq m \end{aligned} \quad (1)$$

where  $C$  is a constant which depends on  $m$ :  $C = C(m)$  and often  $\lim_{m \rightarrow \infty} C(m) = \infty$  [4], [5]. We can rewrite the optimization problem (1) as a regularization scheme [19]: Define the loss function  $\ell(f, z) = (1 - f(x)y)_+$ , where  $(u)_+ = u$  if  $u \geq 0$ ,  $(u)_+ = 0$  if  $u < 0$ . The corresponding generalization error is the expectation of loss function  $\ell(f, z)$  with respect to  $z$ , i.e.,  $\mathcal{E}(f) = E[\ell(f, z)]$ . If we define the empirical error as  $\mathcal{E}_m(f) = \frac{1}{m} \sum_{i=1}^m \ell(f, z_i)$ , then (1) can be written as

$$f_S = \arg \min_{f \in \mathcal{H}_K} \{\mathcal{E}_m(f) + \lambda \|f\|_K^2\} \quad (2)$$

where  $\lambda = 1/(2C)$  is the regularization parameter [19].

Differ from the previously known works in [4] and [5] for i.i.d. samples, in this paper, we consider SVM algorithm with u.e.M.c. samples.

### B. Uniformly Ergodic Markov Chains

Suppose  $(\mathcal{Z}, \mathcal{S})$  is a measurable space, a Markov chain is a sequence of random variables  $\{Z_t\}_{t \geq 1}$  together with a set of transition probability measures  $P^n(A|z_i)$ ,  $A \in \mathcal{S}$ ,  $z_i \in \mathcal{Z}$ . It is assumed that  $P^n(A|z_i) := P\{Z_{n+i} \in A | Z_j, j < i, Z_i = z_i\}$ . Thus,  $P^n(A|z_i)$  denotes the probability that the state  $z_{n+i}$  will belong to the set  $A$  after  $n$  steps, starting from the initial

state  $z_i$  at time  $i$ . The fact that the transition probability does not depend on the values of  $Z_j$  prior to time  $i$  is the Markov property, that is,  $P^n(A|z_i) = P\{Z_{n+i} \in A | Z_i = z_i\}$ . This is expressed in words as “given the present state, the future and past states are independent.” Given two probabilities  $u_1, u_2$  on the measure space  $(\mathcal{Z}, \mathcal{S})$ , we define the total variation distance between the measures  $u_1$  and  $u_2$  as  $\|u_1 - u_2\|_{TV} = \sup_{A \in \mathcal{S}} |u_1(A) - u_2(A)|$ . Thus, we have the following definition of u.e.M.c. [8].

*Definition 1:* A Markov chain  $\{Z_t\}_{t \geq 1}$  is said to be uniformly ergodic if for some  $0 < \gamma < \infty$  and  $0 < \rho < 1$

$$\|P^n(\cdot|z) - \pi(\cdot)\|_{TV} \leq \gamma \rho^n \quad \forall n \geq 1, \quad n \in \mathbb{N}$$

where  $\pi(\cdot)$  is the stationary distribution of  $\{Z_t\}_{t \geq 1}$ .

*Remark 1:* By [21, Th. 3.8], we have that if the state space of Markov chain is finite, and the transition probabilities of any two states are always positive, then this Markov chain is u.e.M.c..

### C. Generalization Ability of SVM With u.e.M.c. Samples

To measure the generalization ability of SVM, we should bound how  $\text{sign}(f_S)$  converges (with respect to the misclassification error) to the best classifier, the Bayes rule  $f_c = \text{sign}(f_\varphi)$  as  $m$  and hence  $C(m)$  tend to infinity, where  $f_\varphi$  is the regression function of  $\varphi$ ,  $f_\varphi(x) = \int_{\mathcal{Y}} y d\varphi(y|x)$ ,  $x \in \mathcal{X}$ . Since deciphering how close  $\text{sign}(f_S)$  is from  $f_c$  is a very difficult issue in general, we usually estimate the excess misclassification error  $\mathcal{R}(\text{sign}(f_S)) - \mathcal{R}(f_c)$  in statistical learning [1].

Chen *et al.* [5] and Steinwart and Scovel [22] estimated the excess misclassification error of SVM with i.i.d. samples and obtained the optimal learning rate. Xu *et al.* [17] extended the works in [5] and [22] with i.i.d. samples to the case of non-i.i.d. samples, u.e.M.c. samples, and established the optimal rate for the SVM with u.e.M.c. samples.

*Proposition 1* [17]: Let  $\{z_i\}_{i=1}^m$  be a u.e.M.c. sample set. Taking  $\lambda = (1/m)^\vartheta$ . For any  $\epsilon > 0$  and  $0 < \delta < 1$ , there exists a constant  $C_1$  independent of  $m$  such that

$$\mathcal{R}(\text{sign}(f_S)) - \mathcal{R}(f_c) \leq C_1 \left( \frac{1}{m} \right)^\theta$$

holds true with probability at least  $1 - \delta$  provided  $m \geq 112(\kappa + 1) \|\Gamma\|^2 \ln(1/\delta) (\ln(1/\delta)/C_s)^{1/s}$ , where  $\vartheta = 2/(1 + \beta)(1 + s)$ , and  $\theta = 2\beta/[(1 + \beta)(1 + s)] - \epsilon$ ,  $\|\Gamma\|$ ,  $C_s > 0$ ,  $1 \geq \beta > 0$ ,  $s > 0$  are constants, which are defined as Lemma 2 and Definitions 3 and 4 in the Appendix.

By Proposition 1, we can conclude that for  $\beta = 1$ ,  $\theta > \frac{1}{2}$  (up to a  $\epsilon$ ). In particular, when  $\beta = 1$ ,  $s \rightarrow 0$ ,  $\theta$  is arbitrarily close to 1. This implies that the learning rate in Proposition 1 is arbitrarily close  $m^{-1}$ , which is the same as the optimal rate obtained in [5], [22], and [23] for the i.i.d. setting.

Although the SVM with u.e.M.c. samples has the optimal learning rate and the SVM based on Markov sampling introduced in [17] has smaller misclassification rates, it is usually very time-consuming compared with the classical SVM. Therefore, in this paper, we introduce a new SVM algorithm based on  $k$ -times Markov sampling.

**Algorithm 1** SVMC Algorithm Based on  $k$  Times Markov Sampling for Balanced Training Samples

**Input:**  $S_T$ ,  $N$ ,  $k$ ,  $q$ ,  $n_2$ 
**Output:**  $\text{sign}(f_k)$ 

- 1:** Draw randomly  $N$  samples  $S_{iid} := \{z_j\}_{j=1}^N$  from  $S_T$ . Train  $S_{iid}$  by SVMC and obtain a preliminary learning model  $f_0$ . Let  $i = 0$ .
- 2:** Let  $N_+ = 0$ ,  $N_- = 0$ ,  $t = 1$ .
- 3:** Draw randomly a sample  $z_t$  from  $S_T$ , called it the current sample. Let  $N_+ = N_+ + 1$  if the label of  $z_t$  is  $+1$ , or let  $N_- = N_- + 1$  if the label of  $z_t$  is  $-1$ .
- 4:** Draw randomly another sample  $z_*$  from  $S_T$ , called it the candidate sample, and calculate the ratio  $\alpha$ ,  $\alpha = e^{-\ell(f_i, z_*)}/e^{-\ell(f_i, z_t)}$ .
- 5:** If  $\alpha \geq 1$ ,  $y_t y_* = 1$  accept  $z_*$  with probability  $\alpha_1 = e^{-y_t f_i}/e^{-y_* f_i}$ . If  $\alpha = 1$  and  $y_t y_* = -1$  or  $\alpha < 1$ , accept  $z_*$  with probability  $\alpha$ . If there are  $n_2$  candidate samples can not be accepted continually, then set  $\alpha_2 = q\alpha$  and accept  $z_*$  with probability  $\alpha_2$ . If  $z_*$  is not accepted, go to Step 4, else let  $z_{t+1} = z_*$ ,  $N_+ = N_+ + 1$  if the label of  $z_{t+1}$  is  $+1$  and  $N_+ < N/2$ , or let  $z_{t+1} = z_*$ ,  $N_- = N_- + 1$  if the label of  $z_{t+1}$  is  $-1$  and  $N_- < N/2$  (if the value  $\alpha$  (or  $\alpha_1, \alpha_2$ ) is bigger than 1, accept the candidate sample  $z_*$  with probability 1).
- 6:** If  $N_+ + N_- < N$ , return to Step 4, else we obtain  $N$  Markov chain samples  $S_{Mar}$ . Let  $i = i + 1$ . Train  $S_{Mar}$  by SVMC and obtain a learning model  $f_i$ .
- 7:** If  $i < k$ , go to Step 2, else output  $\text{sign}(f_k)$ .

### III. SVMC WITH $k$ -TIMES MARKOV SAMPLING

Let  $S_T$  be a given training set,  $m$  be the size of  $S_T$ , and  $N$  be the size of i.i.d. training subset  $S_{iid}$  and the size of Markov training subset  $S_{Mar}$ . Let  $N_+$  and  $N_-$  be the sizes of training samples which label are  $+1$  and  $-1$ , respectively. That is, we first consider the case of balanced training samples.  $q$  and  $n_2$  are two technical parameters, which will be remarked in Remark 2. Then, SVMC with  $k$ -times Markov sampling is stated as follows (see Algorithm 1).

*Remark 2:* By Remark 1, we can conclude that the Markov chain samples  $S_{Mar}$  generated in Step 6 of Algorithm 1 are u.e.M.c. samples, since the acceptance probabilities  $\alpha, \alpha_1$ , and  $\alpha_2$  defined in Algorithm 1 are positive. To generate quickly the Markov chain samples  $S_{Mar}$ , we introduce two technical parameters  $q$  and  $n_2$  in Algorithm 1: since as the value  $\ell(f_i, z_t)$  of the current sample  $z_t$  is smaller, the acceptance probability  $\alpha = e^{-\ell(f_i, z_*)}/e^{-\ell(f_i, z_t)}$  will be smaller, which implies that the candidate sample  $z_*$  will be accepted with a smaller probability. Thus, generating Markov samples  $S_{Mar}$  will be very time-consuming. For the parameter  $n_2$ , we consider  $n_2 = 10, 30, 50$ . For the parameter  $q$ , we consider  $q = 0.2, 1.2, 2.2$ . In experiment, we find that the standard deviations of misclassification rates and the number

TABLE I  
11 REAL-WORLD DATA SETS

Dataset	Size of $D_{train}$	Size of $D_{test}$	Input dimension
Skin	163371	81686	3
Nursery	8640	4320	8
Shuttle	43500	14500	9
Poker	768757	256253	10
Letter	13333	6667	16
Image	26000	20200	18
Statlog	70000	30000	20
SDD	39006	19503	48
Acoustic	65686	32842	50
Covtype	435759	145253	54
Mnist	40000	20000	780

of support vector have a tendency of increase, while the total time of sampling and training has a tendency of decrease as  $q$  increases. The standard deviations of misclassification rates and the number of support vector have a tendency of decrease, while the total time of sampling and training has a tendency of increase as  $n_2$  increases. To have a tradeoff between the standard deviations of misclassification rates, the number of support vector, and the total time of sampling and training, all of the following experimental results are based on  $n_2 = 30$  and  $q = 1.2$ .

*Remark 3:* Compared Algorithm 1 with the classical SVMC, the SVMC based on Markov sampling introduced in [17], we can find that the differences are obvious: the classical SVMC with a given training set  $S$  is the “batch learning” of  $m$  training samples ( $m$  is the size of the training set  $S$ ). The SVMC with Markov sampling introduced in [17] is two times “batch learning” of  $m$  training samples ( $m$  i.i.d. samples and  $m$  Markov chain samples). While Algorithm 1 is  $k + 1$  times “batch learning” of  $N$  training samples and the total number of training samples is  $(k + 1)N \leq m$ .

### IV. EXPERIMENTS AND COMPARISONS

We present the numerical studies on the performance of SVMC with  $k$ -times Markov sampling for 11 real-world data sets. Table I summarizes the properties of the selected data sets. Here, multiclass data sets (e.g., Acoustic and Mnist data sets) were modified randomly to be binary class data sets, such that the size of  $+1$  class equals the size of  $-1$  class as much as possible. For every data set, we break randomly down it into two parts: an original training set  $D_{train}$  and a test set  $D_{test}$ . The parameter  $C$  of SVMC is chosen by the method of five-fold cross validation.

#### A. Comparisons With the Classical SVMC

We first compare Algorithm 1 with the classical SVMC. To have a better showing the performance of Algorithm 1, the training samples of Algorithm 1 are drawn from the training samples trained by the classical SVMC. We simply state our experimental procedure as follows.

- 1) We draw randomly a training set  $S$  from the original training set  $D_{train}$ , and the size of the training set  $S$  is  $m$ . We train  $S$  by SVMC and test it on the given test set  $D_{test}$ .

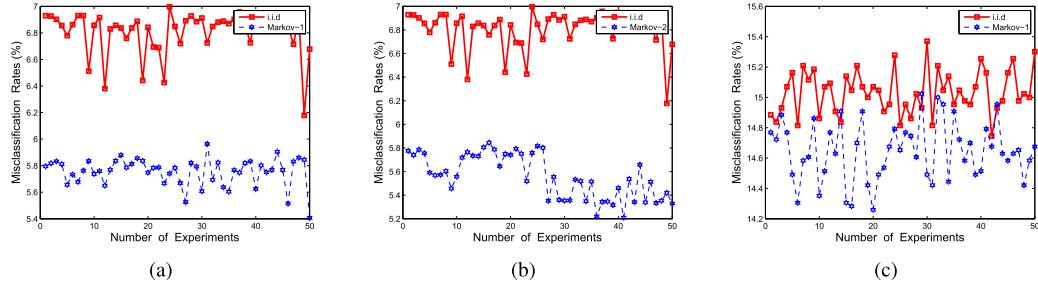


Fig. 1. 50 times experimental misclassification rates. (a) Skin:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ . (b) Skin:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ . (c) Nursery:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ .

TABLE II  
MISCLASSIFICATION RATES (%) FOR  $m = 3000$

Dataset	MR(i.i.d.)	MR(Markov-1)	MR(Markov-2)
Skin	$6.81 \pm 0.16$	$5.75 \pm 0.10$	$5.56 \pm 0.19$
Nursery	$15.03 \pm 0.15$	$14.64 \pm 0.19$	$14.07 \pm 0.25$
Shuttle	$3.27 \pm 0.22$	$2.81 \pm 0.13$	$2.99 \pm 0.83$
Poker	$50.06 \pm 3.70$	$42.89 \pm 3.79$	$37.00 \pm 2.66$
Letter	$27.15 \pm 0.34$	$26.46 \pm 0.48$	$25.75 \pm 0.56$
Image	$15.18 \pm 0.40$	$13.87 \pm 0.40$	$13.26 \pm 0.25$
Statlog	$27.94 \pm 0.74$	$26.30 \pm 0.82$	$26.89 \pm 0.52$
SDD	$40.83 \pm 0.55$	$38.33 \pm 0.72$	$36.48 \pm 0.77$
Acoustic	$26.23 \pm 0.39$	$25.53 \pm 0.37$	$24.65 \pm 0.43$
Covtype	$28.33 \pm 0.32$	$28.02 \pm 0.25$	$27.68 \pm 0.32$
Mnist	$18.51 \pm 0.49$	$15.94 \pm 0.72$	$15.70 \pm 0.52$

- 2) For the training set  $S$ , we set  $S_T = S$  in Algorithm 1, and obtain two classifiers  $\text{sign}(f_k)$  ( $k = 1, 2$ ) by Algorithm 1. Then, we test them on the same test set  $D_{\text{test}}$ .
- 3) We repeat procedures 1) and 2) for 50-times. Since the training set  $S$  is drawn randomly from  $D_{\text{train}}$ , we use “MR (i.i.d.)” to denote the (average) misclassification rates of the classical SVM. We use “MR (Markov- $k$ )” to denote the misclassification rates of Algorithm 1.

For simplicity, in this section, we consider the case of  $k = 1, 2$  and  $m = 3N$ . For example, for the case of  $m = 3000$ , we take  $N = 1000$ . Then, the size of training sample for the SVM with 1-time (or 2-times) Markov sampling is 2000 (or 3000).

1) *Comparison of Misclassification Rates:* We first consider the case of linear prediction models and present the misclassification rates of SVM with  $k$ -times ( $k = 1, 2$ ) Markov sampling for  $m = 3000$  as follows.

In Table II, we can find that for  $m = 3000$ , all the means of the misclassification rates of SVM with  $k$ -times ( $k = 1, 2$ ) Markov sampling are smaller than that of the classical SVM. To compare Algorithm 1 with the classical SVM, we use the Wilcoxon signed-rank test (we show the ranks for each method and whether the hypothesis is rejected with a significance value of  $\alpha = 0.05$ ) [18] to find out whether there exist significant differences between the two methods based on the means of misclassification rates presented in Table II. In Table III, we observe that SVM with  $k$ -times ( $k = 1, 2$ ) Markov sampling (S-M-1 and S-M-2) have a better performance compared with the classical SVM (S-IID) and the SVM with 2-times Markov sampling (S-M-2) has a better

TABLE III  
WILCOXON TESTS FOR S-IID, S-M-1, AND S-M-2

Comparison	$R^+$	$R^-$	Hypothesis ( $\alpha = 0.05$ )	Selected
S-M-1 vs. S-IID	0	66	Rejected for S-M-1	S-M-1
S-M-2 vs. S-IID	0	66	Rejected for S-M-2	S-M-2
S-M-2 vs. S-M-1	7	59	Rejected for S-M-2	S-M-2

TABLE IV  
TOTAL TIME (s) OF SAMPLING AND TRAINING FOR  $m = 3000$

Dataset	Time(i.i.d.)	Time(Markov-1)	Time(Markov-2)
Skin	212.9134	15.1619	20.1091
Nursery	508.2205	38.1275	53.3043
Shuttle	154.08	15.23	28.36
Poker	1203.2	107.72	148.96
Letter	1138.2	85.98	105.99
Image	758.59	48.39	61.19
Statlog	1235.1	92.1809	111.0899
SDD	2207.4	172.848	215.786
Acoustic	4102.8	198.08	227.94
Covtype	1700.2	123.93	148.71
Mnist	1724.8	45.62	63.27

performance compared with the SVM with 1-time Markov sampling (S-M-1).

To have a better understanding the performance of SVM with  $k$ -times ( $k = 1, 2$ ) Markov sampling, we present Figs. 1(a)–8(a) to compare 50-times experimental results of the classical SVM with that of SVM based on  $k$ -times ( $k = 1, 2$ ) Markov sampling. Here, “red square” and “blue hexagon” denote the experimental results of the classical SVM and the SVM with  $k$ -times ( $k = 1, 2$ ) Markov sampling, respectively. The numbers on the vertical axis and the horizontal axis of figures denote the misclassification rates and the experimental times, respectively.

In Figs. 1(a)–8(a), we can find that for 3000 (or 4500 and 6000) training samples, the 50-times misclassification rates of SVM with  $k$ -times Markov sampling are smaller than that of the classical SVM except at most 2-times results for Nursery with  $m = 3000$  and  $k = 1$ .

2) *Comparison of Sampling Time and Training Time:* We use “Time (i.i.d.)”, “Time (Markov- $k$ )” to denote the total time of sampling and training of the classical SVM and the SVM with  $k$ -times Markov sampling, respectively.

In Table IV, we can find that for 3000 training samples, the total time of sampling and training of the SVM with

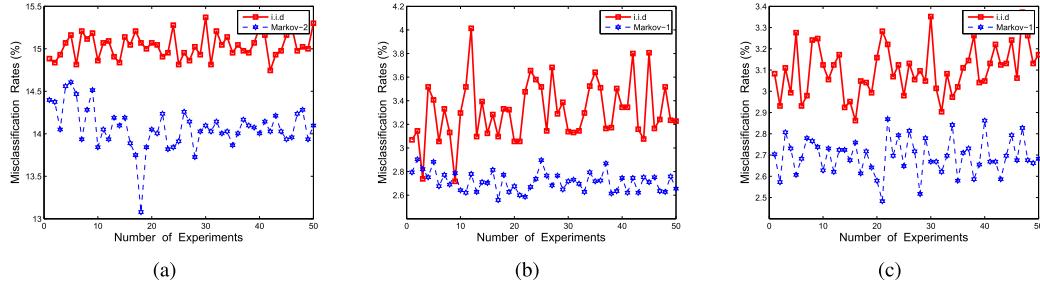


Fig. 2. 50 times experimental misclassification rates. (a) Nursery:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ . (b) Shuttle:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ . (c) Shuttle:  $m = 6000$ ,  $N = 2000$ , and  $k = 1$ .

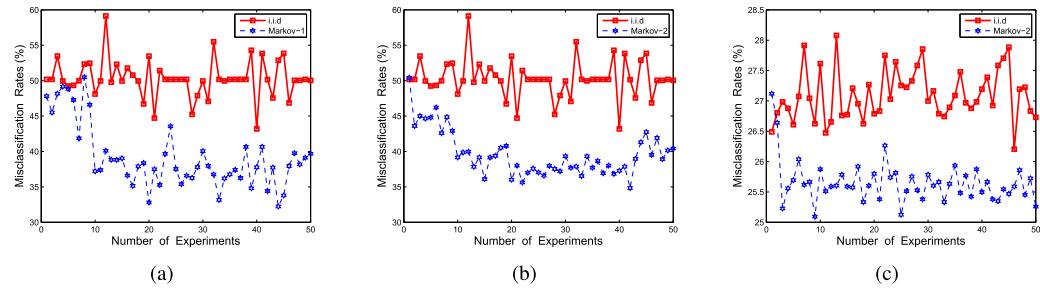


Fig. 3. 50 times experimental misclassification rates. (a) Poker:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ . (b) Poker:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ . (c) Letter:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ .

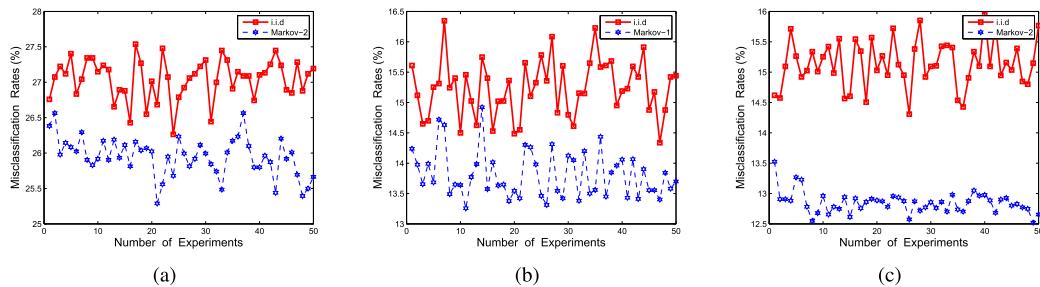


Fig. 4. 50 times experimental misclassification rates. (a) Letter:  $m = 4500$ ,  $N = 1500$ , and  $k = 2$ . (b) Image:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ . (c) Image:  $m = 4500$ ,  $N = 1500$ , and  $k = 2$ .

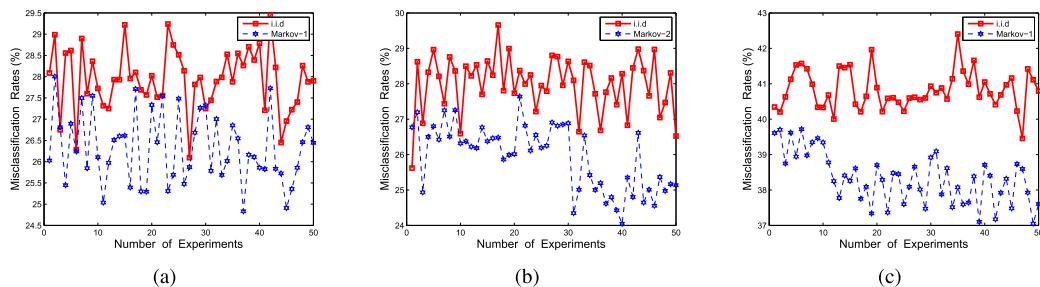


Fig. 5. 50 times experimental misclassification rates. (a) Statlog:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ . (b) Statlog:  $m = 6000$ ,  $N = 2000$ , and  $k = 2$ . (c) SDD:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ .

$k$ -times ( $k = 1, 2$ ) Markov sampling is less than that of the classical SVMC. To have a better showing the performance of SVMC with  $k$ -times Markov sampling, we also present Figs. 8(b)–11(c) to compare the total time of sampling and training of the classical SVMC with that of SVMC based on  $k$ -times ( $k = 1, 2$ ) Markov sampling for a different size  $m$  of the training set  $S$ . The numbers on the vertical axis and

horizontal axis denote the total time of sampling and training for 50-times experiments and the size  $m$  of the training set  $S$ , respectively.

In Figs. 8(b)–11(c), we can find that for different size  $m$  of the training set  $S$ , the total time of sampling and training of SVMC with  $k$ -times ( $k = 1, 2$ ) Markov sampling is less than that of the classical SVMC.

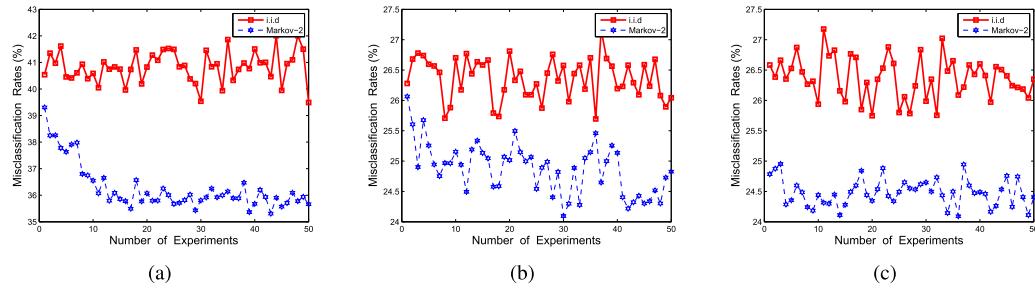


Fig. 6. 50 times experimental misclassification rates. (a) SDD:  $m = 4500$ ,  $N = 1500$ , and  $k = 2$ . (b) Acoustic:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ . (c) Acoustic:  $m = 4500$ ,  $N = 1500$ , and  $k = 2$ .

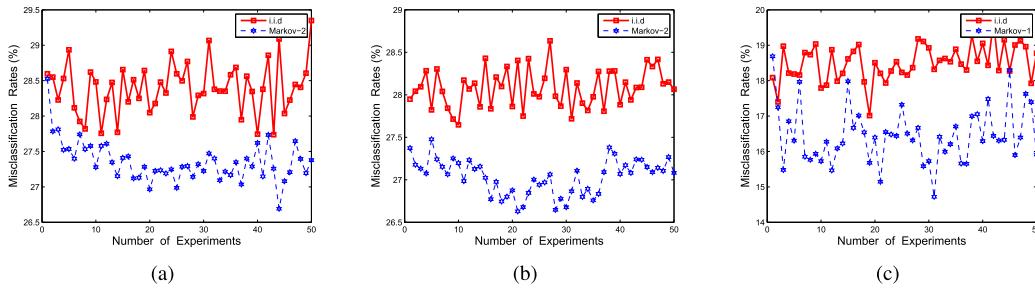


Fig. 7. 50 times experimental misclassification rates. (a) Covtype:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ . (b) Covtype:  $m = 6000$ ,  $N = 2000$ , and  $k = 2$ . (c) Mnist:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ .

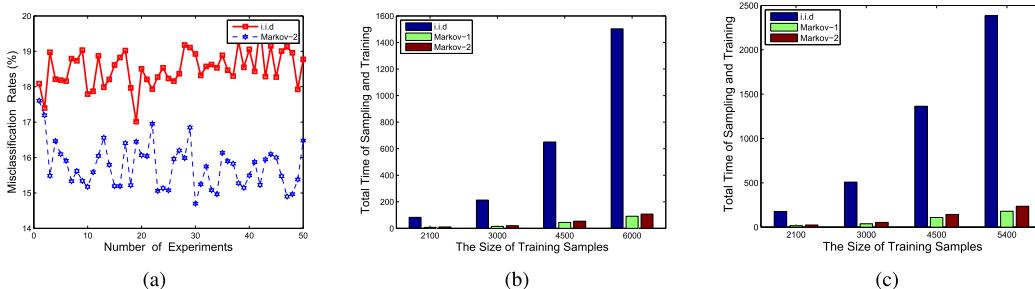


Fig. 8. (a) 50 times experimental misclassification rates for Mnist ( $m = 3000$ ,  $N = 1000$ , and  $k = 2$ ). (b) Total time (s) of sampling and training for Skin with different numbers of training samples. (c) Total time (s) of sampling and training for Nursery with different numbers of training samples.

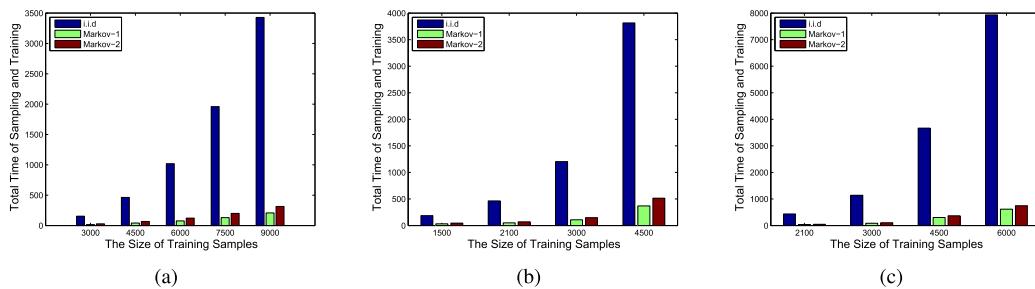


Fig. 9. Total time (s) of sampling and training for different numbers of training samples. (a) Shuttle. (b) Poker. (c) Letter.

3) *Comparison of Support Vector Numbers:* For SVMC, the optimal separating function  $f_S(x)$  reduces to a linear combination of kernels on the training samples [1]

$$f_S(x) = \sum_i \theta_i y_i K(x_i, x) + b, \quad (x_i, y_i) \in S. \quad (3)$$

In (3),  $x_i$  that corresponds to the nonzero coefficients  $\theta_i$  is called to be support vector [1]. If the numbers of support vector

are smaller, then expression (3) is said to be “more sparse.” In Table V, we present the average support vector numbers of the classifier obtained by SVMC with  $k$ -times ( $k = 1, 2$ ) Markov sampling and the classical SVMC for 50 times experiments. Here, “SVs(i.i.d)” and “SVs(Markov- $k$ ) ( $k = 1, 2$ )” denote the (average) support vector numbers of the classical SVMC and the SVMC with  $k$ -times ( $k = 1, 2$ ) Markov sampling, respectively.

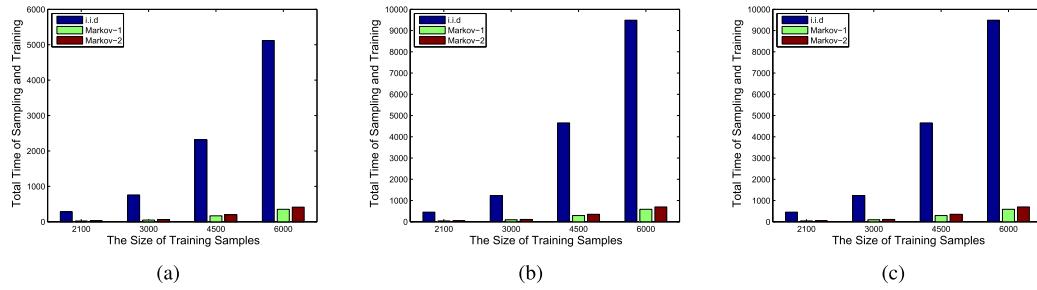


Fig. 10. Total time (s) of sampling and training for different numbers of training samples. (a) Image. (b) Statlog. (c) SDD.

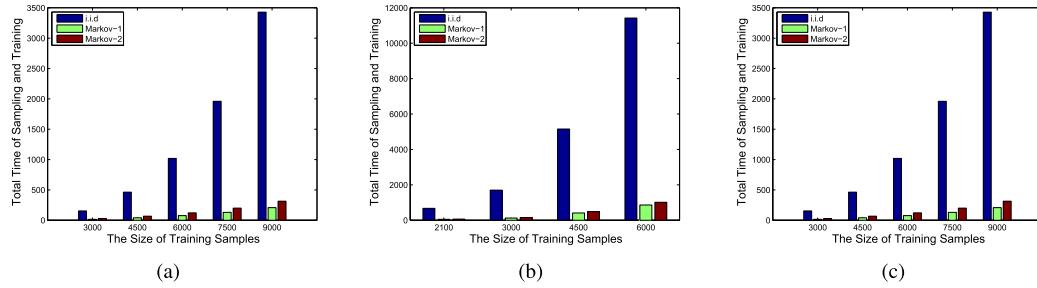


Fig. 11. Total time (s) of sampling and training for different numbers of training samples. (a) Acoustic. (b) Covtype. (c) Mnist.

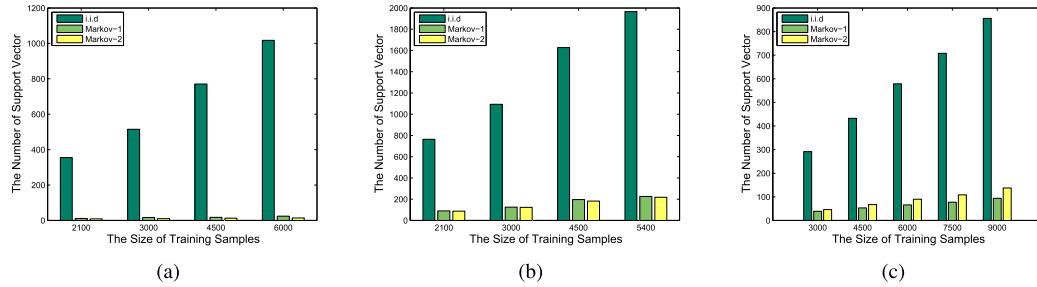


Fig. 12. Average numbers of support vector for different numbers of training samples. (a) Skin. (b) Nursery. (c) Shuttle.

TABLE V  
AVERAGE NUMBERS OF SUPPORT VECTOR FOR  $m = 3000$

Dataset	SVs(i.i.d.)	SVs(Markov-1)	SVs(Markov-2)
Skin	515.06	16.64	9.86
Nursery	1093.6	125.34	123.06
Shuttle	291.78	38.8	46.12
Poker	2950.3	769.62	565.98
Letter	1845.3	297.86	186.06
Image	1356.4	127.14	88.12
Statlog	1878.3	274.88	174.10
SDD	2511.8	512.24	329.4
Acoustic	1770.2	208.24	157.38
Covtype	1963.2	286.08	185.40
Mnist	780.62	207.10	198.74

Table V shows that for  $m = 3000$ , the numbers of support vector for SVMC with  $k$ -times ( $k = 1, 2$ ) Markov sampling are less than that of the classical SVMC. To have a better showing the performance of SVMC based on  $k$ -times Markov sampling, we present Figs. 12(a)–15(b) to compare the (average) numbers of support vector of the classical SVMC with that of SVMC based on  $k$ -times ( $k = 1, 2$ ) Markov sampling for different sizes  $m$  of training set  $S$ . Here, the numbers on the vertical axis and the horizontal axis of figures denote the

(average) numbers of support vector and the size  $m$  of training set  $S$ , respectively.

In Figs. 12(a)–15(b), we can find that for different size  $m$  of the training set  $S$ , the average numbers of support vector of SVMC with  $k$ -times ( $k = 1, 2$ ) Markov sampling are much less than that of the classical SVMC, which implies that the classifiers obtained by SVMC with  $k$ -times ( $k = 1, 2$ ) Markov sampling are more sparse compared with the classical SVMC.

4) *Case of Nonlinear Prediction Models:* The above-mentioned experimental results are based on the linear prediction models. For the case of nonlinear prediction models, we consider Gaussian kernels [23] and present Figs. 15(c)–17(c) to show the performance of Gaussian kernels SVMC with  $k$ -times ( $k = 1, 2$ ) Markov sampling for Acoustic, SDD, and Mnist data sets (the parameter  $\sigma$  is choosed by the method of five-fold cross validation, and we use the same parameter  $\sigma$  of kernel,  $\sigma = 30$  for three data sets).

In Figs. 15(c)–17(c), we find that for Acoustic, SDD, and Minst data sets with 3000 (or 4500) training samples, almost all the 50-times experimental results of Gaussian kernels SVMC with  $k$ -times Markov sampling are better than that of the classical SVMC with Gaussian kernels prediction models except at most 2-times results for Acoustic with

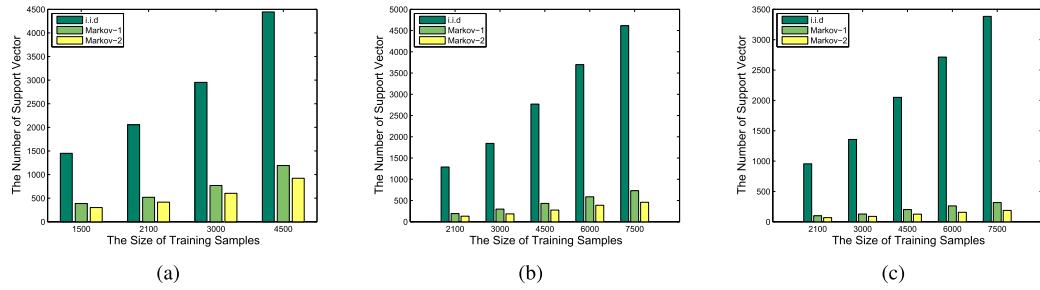


Fig. 13. Average numbers of support vector for different numbers of training samples. (a) Poker. (b) Letter. (c) Image.

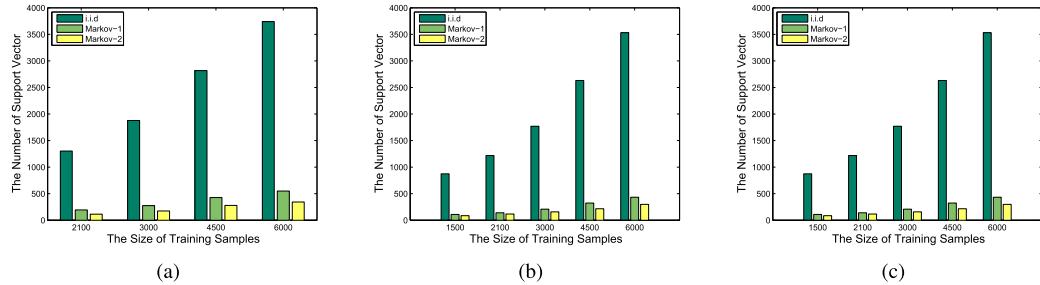
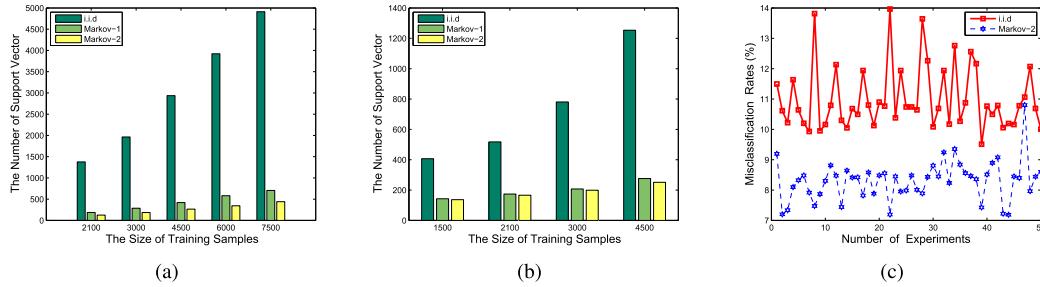
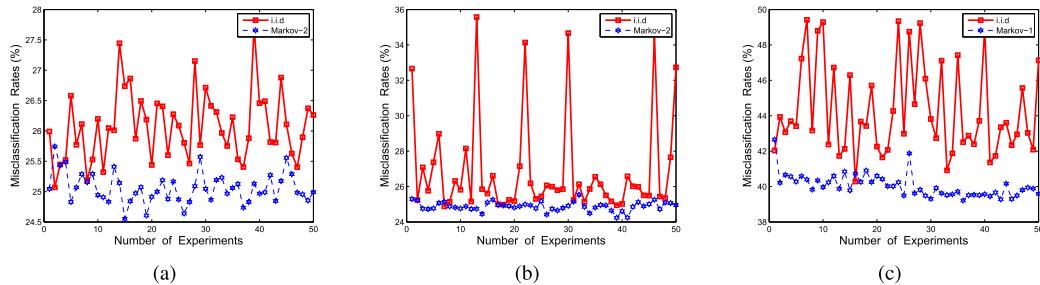


Fig. 14. Average numbers of support vector for different numbers of training samples. (a) Statlog. (b) SDD. (c) Acoustic.

Fig. 15. Average numbers of support vector for different numbers of training samples. (a) Covtype. (b) Mnist. (c) 50 times experimental misclassification rates for Shuttle with Gaussian kernel and  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ .Fig. 16. 50 times misclassification rates for Gaussian kernel. (a) Acoustic:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ . (b) Acoustic:  $m = 4500$ ,  $N = 1500$ , and  $k = 2$ . (c) SDD:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ .

$m = 4500$  and  $k = 2$ . The other experimental results (the total time of sampling and training, the average numbers of support vector) are similar to that of linear prediction models presented in the last section.

#### B. Comparisons With the SVMC Based on Markov Sampling Introduced in [17]

In this section, we compare our method (Algorithm 1) with the SVMC based on Markov sampling introduced in [17].

Since the training samples of SVMC based on Markov sampling introduced in [17] are drawn from the original training set  $D_{\text{train}}$ , we take  $S_T = D_{\text{train}}$  in Algorithm 1, and compare the learning performance of the two methods under the same size of training samples. We use ‘‘MR (Markov),’’ ‘‘Time (Markov),’’ ‘‘SVs (Markov)’’ [‘‘MR (Markov- $k$ ),’’ ‘‘Time (Markov- $k$ ),’’ and ‘‘SVs (Markov- $k$ )’’] to denote the misclassification rates, the total time of sampling and training, and the support vector numbers of the SVMC based on Markov

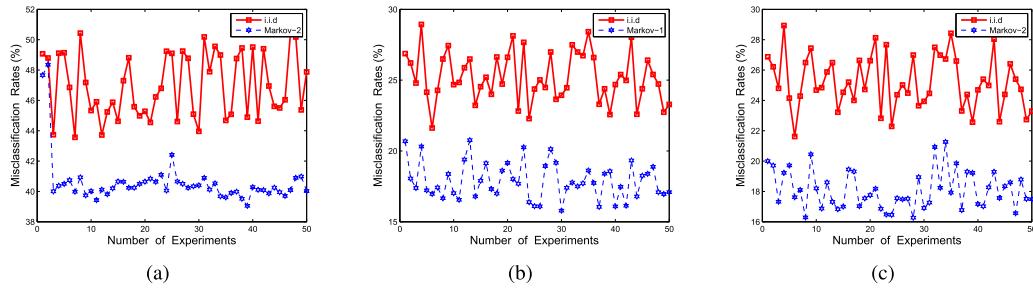


Fig. 17. 50 times misclassification rates for Gaussian kernel. (a) SDD:  $m = 4500$ ,  $N = 1500$ , and  $k = 2$ . (b) Mnist:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ . (c) Mnist:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ .

TABLE VI  
ION RATES (%) FOR  $m = 6000$

Dataset	MR(Markov)	MR(Markov-1)	MR(Markov-2)
Skin	$5.76 \pm 0.62$	$5.76 \pm 0.08$	$5.47 \pm 0.13$
Nursery	$15.11 \pm 0.17$	$15.21 \pm 0.19$	$15.11 \pm 0.15$
Shuttle	$2.70 \pm 0.06$	$2.78 \pm 0.07$	$2.68 \pm 0.10$
Poker	$0.37 \pm 0.02$	$0.36 \pm 0.01$	$0.36 \pm 0.04$
Letter	$26.20 \pm 0.22$	$26.23 \pm 0.26$	$25.54 \pm 0.15$
Image	$13.80 \pm 0.40$	$13.92 \pm 0.31$	$13.11 \pm 0.21$
Statlog	$25.21 \pm 0.06$	$24.95 \pm 0.63$	$23.79 \pm 0.50$
SDD	$38.81 \pm 0.29$	$38.96 \pm 0.34$	$37.46 \pm 0.17$
Acoustic	$24.77 \pm 0.16$	$24.79 \pm 0.24$	$24.15 \pm 0.33$
Covtype	$27.43 \pm 0.13$	$27.55 \pm 0.16$	$27.27 \pm 0.09$
Mnist	$15.18 \pm 0.42$	$15.24 \pm 0.34$	$14.83 \pm 0.35$

sampling introduced in [17] [the SVMC based on  $k$ -times ( $k = 1, 2$ ) Markov sampling], respectively. All the experimental results are the average of 50-times experiments and  $m$  is the size of training samples.

In Table VI, we can find that for 6000 training samples, almost all the means of misclassification rates of SVMC based on 2-times Markov sampling are smaller than that of the SVMC based on Markov sampling introduced in [17]. To compare SVMC based on  $k$ -times Markov sampling with the SVMC based on Markov sampling introduced in [17], we use the Wilcoxon signed-rank test (we show the ranks for each method and whether the hypothesis is rejected with a significance value of  $\alpha = 0.05$ ) [18] to find out whether there exist significant differences between two methods based on the means of misclassification rates presented in Table VI.

In Table VII, we observe that for  $m = 6000$ , the SVMC with 2-times Markov sampling (S-M-2) has a better performance compared with the SVMC based on Markov sampling (S-M) introduced in [17] and the SVMC with 1-time Markov sampling (S-M-1) and the SVMC based on Markov sampling (S-M) introduced in [17] has a better performance compared with the SVMC with 1-time Markov sampling (S-M-1).

In Tables VIII and IX, we can find that for 6000 training samples, the total time of sampling and training of SVMC with  $k$ -times ( $k = 1, 2$ ) Markov sampling is far less than that of the SVMC based on Markov sampling introduced in [17]. The support vector numbers of SVMC based on  $k$ -times ( $k = 1, 2$ ) Markov sampling are smaller than that of SVMC based on Markov sampling introduced in [17].

TABLE VII  
WILCOXON TESTS FOR S-M, S-M-1, AND S-M-2

Comparison	$R^+$	$R^-$	Hypothesis( $\alpha = 0.05$ )	Selected
S-M-1 vs. S-M	52.5	13.5	Not Rejected	S-M
S-M-2 vs. S-M	3	63	Rejected for S-M-2	S-M-2
S-M-2 vs. S-M-1	0.5	65.5	Rejected for S-M-2	S-M-2

TABLE VIII  
TOTAL TIME (s) OF SAMPLING AND TRAINING FOR  $m = 6000$

Dataset	Time(Markov)	Time(Markov-1)	Time(Markov-2)
Skin	254.3568	90.8987	109.2663
Nursery	672.696	237.8498	283.4012
Shuttle	214.01	77.11	103.71
Poker	1020.7462	330.4296	588.3988
Letter	1879.91	651.86	766.76
Image	1059.11	352.96	406.08
Statlog	5132.8	647.1861	795.2471
SDD	3448	1162.9368	1461.7573
Acoustic	4897.60	1460.82	1641.96
Covtype	2465.51	887.71	1046.85
Mnist	1935.72	305.72	392.63

TABLE IX

Dataset	SVs(Markov)	SVs(Markov-1)	SVs(Markov-2)
Skin	47.84	38.16	21.32
Nursery	361.10	225.76	135.88
Shuttle	88.38	62.02	63.74
Poker	65.96	64.14	34.44
Letter	899.06	602.86	309.88
Image	487.50	322.70	170.34
Statlog	1001.9	684.66	442.60
SDD	1707	1129	725.8
Acoustic	755.58	461.06	287.12
Covtype	883.92	632.74	348.52
Mnist	363.40	305.44	315.42

## V. EXPLANATIONS OF LEARNING PERFORMANCE

In this section, we explain the performance of SVMC with  $k$ -times Markov sampling as follows.

- 1) By Algorithm 1, we can find that the size of training samples used by each time training is  $N$ , which is smaller than the size of the training samples trained

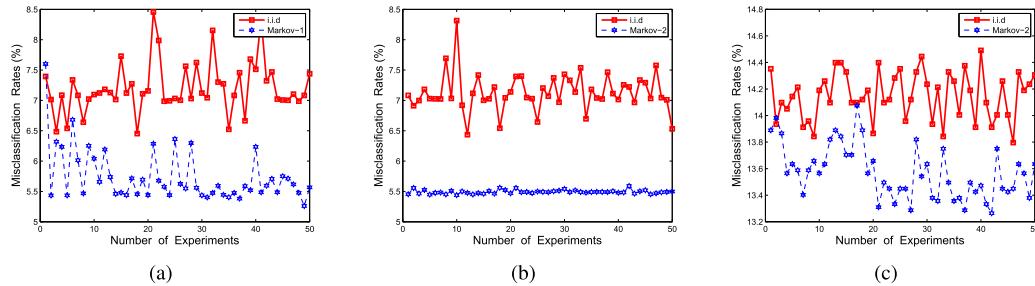


Fig. 18. 50 times experimental misclassification rates for unbalanced training samples. (a) Skin:  $m = 3000$ ,  $N = 1000$ , and  $k = 1$ . (b) Skin:  $m = 4500$ ,  $N = 1500$ , and  $k = 2$ . (c) Nursery:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ .

by the classical SVM, and the SVM with Markov sampling introduced in [17]. The algorithmic complexity of SVM is about  $O(m^3)$  as the size of training samples trained by SVM is  $m$  [24]. Therefore, although SVM with  $k$ -Markov sampling consists of  $k+1$  times training SVM, which is more than the training times of the classical SVM and the SVM with Markov sampling introduced in [17], the total time of sampling and training of SVM with  $k$ -times Markov sampling is less than that of the classical SVM and the SVM with Markov sampling introduced in [17].

- 2) By the statistical learning theory in [19], the samples that closest to the interface of two classes data are the most “important” samples for classification problems. In other words, many samples in the original training set  $D_{\text{train}}$  are “redundant” for classification problems. Since we know only the training samples (instead of the distribution of training samples), to define the transition probabilities of Markov chain samples, we draw randomly  $N$  samples from  $S_T$  and obtain a preliminary learning model  $f_0$ . Thus,  $f_0$  has the structure information of the given data set and the Markov chain samples  $S_{\text{Mar}}$  that used  $f_0$  to define the transition probabilities are “representative and good samples” compared with the i.i.d. samples. Thus, the learning model  $f_1$  can be considered as a “revision” of  $f_0$ . Similarly, the Markov samples that used  $f_1$  to define the transition probabilities are more “representative and good samples” compared with the Markov chain samples that used  $f_0$  to define the transition probabilities. Therefore, although the size  $N$  of training samples for each training of the SVM with  $k$ -times Markov sampling is smaller than the size of training samples of the classical SVM, and the SVM with Markov sampling introduced in [17], the misclassification rates of SVM with  $k$ -times Markov sampling can be smaller compared with the classical SVM and the SVM based on Markov sampling introduced in [17].

In 1) and 2), we have that for the SVM with  $k$ -times Markov sampling, since the size of training samples of each time training is smaller and these training samples are “representative and good samples,” the support vector number of the SVM with  $k$ -times Markov sampling is smaller than that of the classical SVM and the SVM based on Markov sampling introduced in [17].

---

#### Algorithm 2 SVM Algorithm Based on $k$ Times Markov Sampling for Unbalanced Training Samples

---

**Input:**  $S_T$ ,  $N$ ,  $k$ ,  $q$ ,  $n_2$   
**Output:**  $\text{sign}(f_k)$

- 1: Draw randomly  $N$  samples  $S_{\text{iid}} := \{z_j\}_{j=1}^N$  from  $S_T$ . Train  $S_{\text{iid}}$  by SVM and obtain a preliminary learning model  $f_0$ . Let  $i = 0$ .
  - 2: Let  $N_i = 0$ ,  $t = 1$ .
  - 3: Draw randomly a sample  $z_t$  from  $S_T$ , called it the current sample. Let  $N_i = N_i + 1$ .
  - 4: Draw randomly another sample  $z_*$  from  $S_T$ , called it the candidate sample. Calculate the ratio  $\alpha$ ,  $\alpha = e^{-\ell(f_i, z_*)}/e^{-\ell(f_i, z_t)}$ .
  - 5: If  $\alpha = 1$ ,  $y_t y_* = 1$  accept  $z_*$  with probability  $\alpha_1 = e^{-y_* f_i}/e^{-y_t f_i}$ . If  $\alpha = 1$  and  $y_t y_* = -1$  or  $\alpha < 1$ , accept  $z_*$  with probability  $\alpha$ . If there are  $n_2$  candidate samples can not be accepted continually, then set  $\alpha_2 = q\alpha$  and accept  $z_*$  with probability  $\alpha_2$ . If  $z_*$  is not accepted, go to Step 4, else let  $z_{t+1} = z_*$ ,  $N_i = N_i + 1$  (if  $\alpha$  (or  $\alpha_1$ ,  $\alpha_2$ ) is greater than 1, accept  $z_*$  with probability 1).
  - 6: If  $N_i < N$ , return to Step 4, else we obtain  $N$  Markov chain samples  $S_{\text{Mar}}$ . Let  $i = i + 1$ . Train  $S_{\text{Mar}}$  by SVM and obtain a learning model  $f_i$ .
  - 7: If  $i < k$ , go to Step 2, else output  $\text{sign}(f_k)$ .
- 

## VI. DISCUSSION

We give some discussions on the performance of SVM with  $k$ -times Markov sampling for the cases of unbalanced training sample and large-scale training samples.

### A. Case of Unbalanced Training Sample

For simplicity, all the experimental results in the last section are based on the case that the training samples of +1 class and -1 class are balanced. However, the training samples of many real-world data sets are unbalanced, such as Skin, Covtype, Statlog, and Nursery data sets. Then, we introduce a new algorithm (Algorithm 2) and compare it with the classical SVM. Here, the training samples of Algorithm 2 are also drawn from the training set of the classical SVM.

In Figs. 18(a)–20(a), we can find that almost all the 50 times misclassification rates of SVM with  $k$ -Markov sampling are smaller than that of the classical SVM except at most 3 times

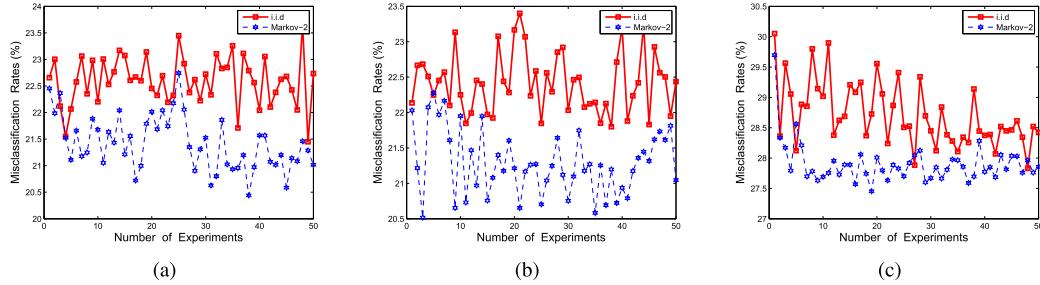


Fig. 19. 50 times experimental misclassification rates for unbalanced training samples. (a) Statlog:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ . (b) Statlog:  $m = 4500$ ,  $N = 1500$ , and  $k = 2$ . (c) Covtype:  $m = 3000$ ,  $N = 1000$ , and  $k = 2$ .

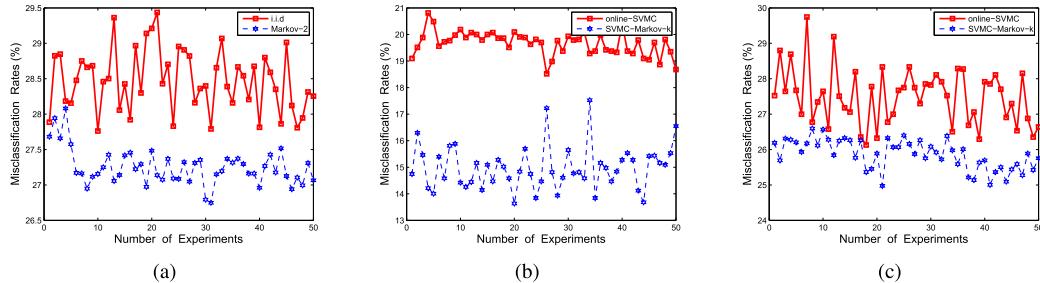


Fig. 20. (a) 50 times experimental misclassification rates of unbalanced training samples for Covtype ( $m = 4500$ ,  $N = 1500$ , and  $k = 2$ ). 50 times experimental misclassification rates for large training samples. (b) Nursery:  $m = 8500$ ,  $N = 150$ , and  $k = 2$ . (c) Letter:  $m = 13000$ ,  $N = 450$ , and  $k = 2$ .

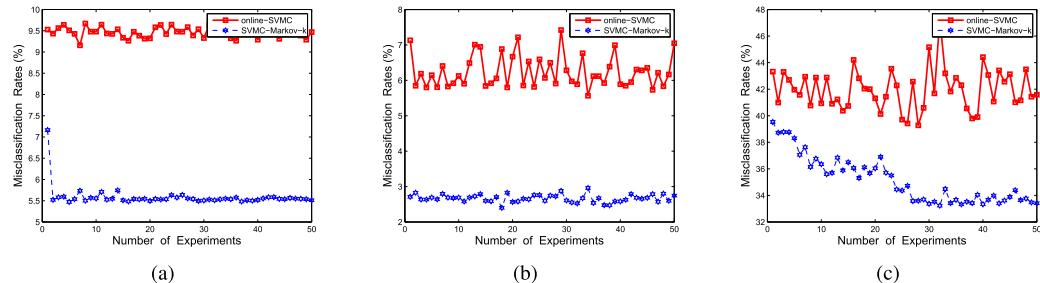


Fig. 21. 50 times experimental misclassification rates for large-scale training samples. (a) Skin:  $m = 150000$ ,  $N = 600$ , and  $k = 3$ . (b) Shuttle:  $m = 40000$ ,  $N = 900$ , and  $k = 3$ . (c) SDD:  $m = 39000$ ,  $N = 500$ , and  $k = 3$ .

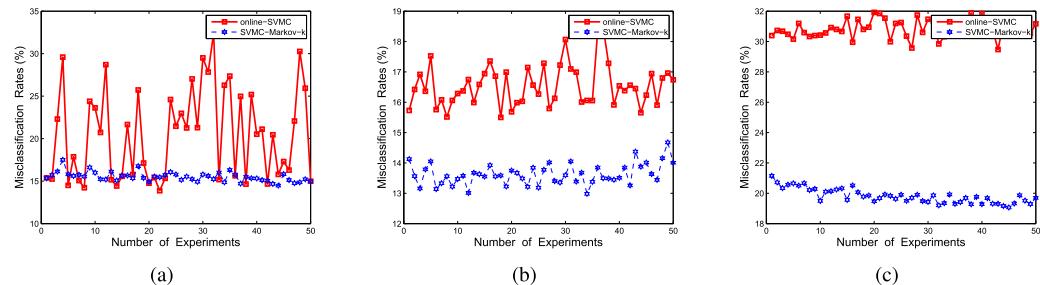


Fig. 22. 50 times experimental misclassification rates for large-scale training samples. (a) Mnist:  $m = 40000$ ,  $N = 900$ , and  $k = 4$ . (b) Image:  $m = 20000$ ,  $N = 350$ , and  $k = 2$ . (c) Statlog:  $m = 65000$ ,  $N = 600$ , and  $k = 3$ .

results for Covtype with  $m = 3000$ . In addition, other experimental results (the total time of sampling and training and the numbers of support vectors) of Algorithm 2 for these data sets are similar to that of the case of balanced training samples presented in the last section.

#### B. Case of Large-Scale Training Samples

SVM might be practically challenging when the size  $m$  of training samples is large. In particular, when the size  $m$

of training samples satisfies  $m \geq 10000$ , SVM is hard to implement [24]. Thus, when the size  $m$  of training samples is large, online SVM algorithms can be applied to provide efficient classifiers and its complexity is about  $O(m)$ . Therefore, we compare Algorithm 2 with online SVM [24] based on randomly independent sampling. Since the size of training samples of online SVMC is large, the training samples of Algorithm 2 are drawn from the original training set  $D_{\text{train}}$ .

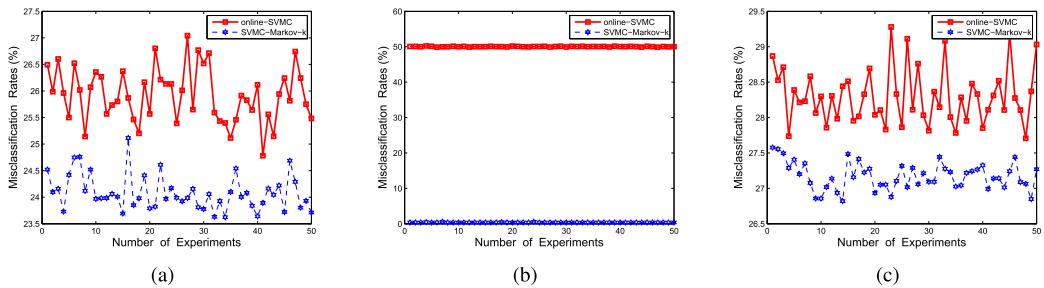


Fig. 23. 50 times experimental misclassification rates for large-scale training samples. (a) Acoustic:  $m = 70\,000$ ,  $N = 700$ , and  $k = 3$ . (b) Poker:  $m = 400\,000$ ,  $N = 1000$ , and  $k = 8$ . (c) Covtype:  $m = 400\,000$ ,  $N = 1000$ , and  $k = 17$ .

TABLE X  
TOTAL TIME (s) OF SAMPLING AND TRAINING FOR  
LARGE-SCALE TRAINING SAMPLES

Dataset	Time(online-SVMC)	Time(Markov-k)	# Figure
Skin	78.6436	11.9147	Fig. 21(a)
Nursery	4.6935	3.8066	Fig. 20(b)
Shuttle	15.1843	18.0310	Fig. 21(b)
Letter	8.7804	25.9339	Fig. 20(c)
Poker	371.1514	54.8161	Fig. 23(b)
Image	10.9732	11.3782	Fig. 22(b)
Statlog	44.9254	39.6226	Fig. 22(c)
SDD	28.1083	52.7678	Fig. 21(c)
Acoustic	48.1018	95.6211	Fig. 23(a)
Covtype	301.2292	338.2878	Fig. 23(c)
Mnist	65.6599	72.8723	Fig. 22(a)

In Figs. 20(b)–23(c), we can find that almost all the 50-times misclassification rates of SVM with  $k$ -times Markov sampling (SVM-Markov- $k$ ) are smaller than that of online SVM with random sampling (online-SVMC) except about 8-times experimental results for Mnist data set. In Table X, we can find that the total time of sampling and training of SVM with  $k$ -times Markov sampling [Time(Markov- $k$ )] is close to that of online SVM with randomly sampling [Time(online-SVMC)].

## VII. CONCLUSION

To improve the learning performance of the classical SVM and the SVM with Markov sampling introduced in [17], in this paper, we introduced a new SVM algorithm based on  $k$ -times Markov sampling (Algorithm 1) for the case of balanced training samples, and compared our algorithm with the classical SVM and the SVM based on Markov sampling introduced in [17]. The experimental results indicated that the learning performance (the misclassification rates, the total time of sampling and training, and the numbers of support vector) of the SVM with  $k$ -times ( $k = 1, 2$ ) Markov sampling is better than that of the classical SVM and the SVM with Markov sampling introduced in [17]. Since many real-world data sets of two-class classification problem are unbalanced, we presented another SVM algorithm with  $k$ -times Markov sampling (Algorithm 2) for the case of unbalanced samples. In addition, although SVM is one of the most widely used algorithms for classification problem, the algorithmic

complexity of SVM is higher as the size of training samples is larger. Therefore, we also compared SVM based on  $k$ -times Markov sampling (Algorithm 2) with the online SVM based on random sampling. To the best of our knowledge, these studies here are the first works on this topic.

Along the line of this paper, several open problems deserves further research, for example, studying the performance of SVM for regression based on  $k$ -times Markov sampling and establishing the bounds on the support vector numbers for the SVM with  $k$ -times Markov sampling. All these problems are under our current investigation.

## APPENDIX

**Lemma 1 (Doeblin Condition [25]):** Let  $\{Z_t\}_{t \geq 1}$  be a Markov chain with transition probability measure  $P^k(\cdot|\cdot)$ , and let  $\mu$  be some nonnegative measure with nonzero mass  $\mu_0$ . If there is some integer  $t$  such that for all  $z$  in  $\mathcal{Z}$ , and all measurable sets  $A$ ,  $P^t(A|z) \leq \mu(A)$ , then for any integer  $k$  and for any  $z, z'$  in  $\mathcal{Z}$ ,  $\|P^k(\cdot|z) - P^k(\cdot|z')\|_{TV} \leq 2\beta_1^{k/t}$ , where  $\beta_1 = 1 - \mu_0$ .

**Lemma 2 [26]:** For u.e.M.c. sample  $Z_1, \dots, Z_m$ , we have  $\|\Gamma\| \leq \sqrt{2}/(1 - \beta_1^{1/2t})$ , where  $\beta_1$  and  $t$  are defined as that in Lemma 1.

**Definition 2:** For a subset  $\mathcal{F}$  of a metric space and  $\epsilon > 0$ , the covering number  $\mathcal{N}(\mathcal{F}, \epsilon)$  of the function set  $\mathcal{F}$  is the minimal  $\tau \in \mathbb{N}$ , such that there exist  $\tau$  disks in  $\mathcal{F}$  with radius  $\epsilon$  covering  $\mathcal{F}$ .

For  $r > 0$ , let  $\mathcal{B}_r = \{f \in \mathcal{H}_K : \|f\|_K \leq r\}$ . It is a subset of  $\mathcal{C}(X)$  and the covering number is well defined [27]. We denote the covering number of  $\mathcal{B}_1$  as  $\mathcal{N}(\epsilon) = \mathcal{N}(\mathcal{B}_1, \epsilon)$ ,  $\epsilon > 0$ .

**Definition 3 [5]:** The RKHS is said to have polynomial complexity exponent  $s > 0$  if there exists some  $C_s > 0$  such that  $\ln \mathcal{N}(\epsilon) \leq C_s(1/\epsilon)^s$ ,  $\forall \epsilon > 0$ .

**Remark 4:** The covering number  $\mathcal{N}(\epsilon)$  has been extensively studied (please see [28]–[31]). It was shown in [30] that Definition 3 holds if  $K$  is  $C^{2n/s}$  on a subset  $\mathcal{X}$  of  $\mathbb{R}^n$ . In particular, for a  $C^\infty$  kernel (such as Gaussians), Definition 3 is valid for any  $s > 0$  [30].

**Definition 4 [32]:** Let  $f_\lambda = \arg \min_{f \in \mathcal{H}_K} \{\mathcal{E}(f) + \lambda \|f\|_K^2\}$ . We say the function  $f_\psi$  can be approximated by  $\mathcal{H}_K$  with exponent  $0 < \beta \leq 1$  if there exists a constant  $C_\beta$  such that for any  $\lambda > 0$ ,  $D(\lambda) \leq C_\beta \lambda^\beta$ , where  $D(\lambda) := \mathcal{E}(f_\lambda) - \mathcal{E}(f_\psi) + \lambda \|f_\lambda\|_K^2$ .

## ACKNOWLEDGMENT

The authors would like to thank the Editor-in-Chief, the handling Associate Editor, and three anonymous referees, whose careful comments and valuable suggestions led to a significant improvement of the presentation of this paper.

## REFERENCES

- [1] V. Vapnik, *Statistical Learning Theory*. Hoboken, NJ, USA: Wiley, 1998.
- [2] T. Zhang, "Statistical behaviour and consistency of classification methods based on convex risk minimization," *Ann. Statist.*, vol. 32, no. 1, pp. 56–85, Mar. 2004.
- [3] I. Steinwart, "Consistency of support vector machines and other regularized kernel classifiers," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 128–142, Jan. 2005.
- [4] I. Steinwart and A. Christmann, *Support Vector Machines*. New York, NY, USA.: Springer, 2008.
- [5] D. R. Chen, Q. Wu, Y. M. Ying, and D. X. Zhou, "Support vector machine soft margin classifiers: Error analysis," *J. Mach. Learn. Res.*, vol. 5, pp. 1143–1175, Sep. 2004.
- [6] I. Steinwart and A. Christmann, "Fast learning from non-i.i.d. observations," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, Dec. 2009, pp. 1768–1776.
- [7] T. Steinwart, D. Hush, and C. Scovel, "Learning from dependent observations," *J. Multivariate Anal.*, vol. 100, no. 1, pp. 175–194, Jan. 2009.
- [8] M. Vidyasagar, *Learning and Generalization With Application to Neural Networks*, 2nd ed. London, U.K.: Springer, 2003.
- [9] B. Zou, L. Q. Li, and Z. B. Xu, "The generalization performance of ERM algorithm with strongly mixing observations," *Mach. Learn.*, vol. 75, no. 3, pp. 275–295, Jun. 2009.
- [10] B. Yu, "Rates of convergence for empirical processes of stationary mixing sequences," *Ann. Probab.*, vol. 22, no. 1, pp. 94–116, Jan. 1994.
- [11] M. Mohri and A. Rostamizadeh, "Stability bounds for stationary  $\varphi$ -mixing and  $\beta$ -mixing processes," *J. Mach. Learn. Res.*, vol. 11, pp. 798–814, Feb. 2010.
- [12] S. Smale and D. X. Zhou, "Online learning with Markov sampling," *Anal. Appl.*, vol. 7, pp. 87–113, Jan. 2009.
- [13] J. Xu, Y. Y. Tang, B. Zou, Z. Xu, L. Li, and Y. Lu, "The generalization ability of online SVM classification based on Markov sampling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 628–639, Mar. 2015.
- [14] B. Zou, L. Q. Li, Z. B. Xu, T. Luo, and Y. Y. Tang, "The generalization performance of Fisher linear discriminant based on Markov sampling," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 2, pp. 288–300, Feb. 2013.
- [15] B. Zou, Z. Peng, and Z. B. Xu, "The learning performance of support vector machine classification based on Markov sampling," *Sci. China Inf. Sci.*, vol. 56, no. 3, pp. 1–16, Mar. 2013.
- [16] J. Xu, Y. Y. Tang, B. Zou, Z. B. Xu, L. Q. Li, and Y. Lu, "Generalization performance of Gaussian kernels SVMC based on Markov sampling," *Neural Netw.*, vol. 53, pp. 40–51, May 2014.
- [17] J. Xu *et al.*, "The generalization ability of SVM classification based on Markov sampling," *IEEE Trans. Cybern.*, vol. 45, no. 6, pp. 1169–1179, Jun. 2015.
- [18] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bull.*, vol. 1, no. 6, pp. 80–83, 1945.
- [19] T. Evgeniou, M. Pontil, and T. Poggio, "Regularization networks and support vector machines," *Adv. Comput. Math.*, vol. 13, no. 1, pp. 1–50, 2000.
- [20] N. Aronszajn, "Theory of reproducing kernels," *Trans. Amer. Math. Soc.*, vol. 68, no. 3, pp. 337–404, 1950.
- [21] M. P. Qian and G. L. Gong, *Application of Random Processes*. Beijing, China: Peking Univ. Press, 1998.
- [22] I. Steinwart and C. Scovel, "Fast rates for support vector machines," in *Learning Theory: COLT 2005* (Lecture Notes in Computer Science), vol. 3559, P. Auer and R. Meir, Eds. Berlin, Germany: Springer, 2005, pp. 279–294.
- [23] I. Steinwart and C. Scovel, "Fast rates for support vector machines using Gaussian kernels," *Ann. Statist.*, vol. 35, no. 2, pp. 575–607, Jun. 2007.
- [24] Y. Ying and D. X. Zhou, "Online regularized classification algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 4775–4788, Nov. 2006.
- [25] S. P. Meyn and R. L. Tweedie, *Markov chains Stochastic Stability*. New York, NY, USA: Springer-Verlag, 1993.
- [26] P. M. Samson, "Concentration of measure inequalities for Markov chains and  $\Phi$ -mixing processes," *Ann. Probab.*, vol. 28, no. 1, pp. 416–461, Jan. 2000.
- [27] A. W. van der Vaart and J. A. Wellner, *Weak Convergence Empirical Processes*. New York, NY, USA: Springer-Verlag, 1996.
- [28] T. Zhang, "Covering number bounds of certain regularized linear function classes," *J. Mach. Learn. Res.*, vol. 2, pp. 527–550, Mar. 2002.
- [29] P. Doukhan, *Mixing: Properties Examples* (Lecture Notes in Statistics). Berlin, Germany: Springer, 1995.
- [30] D. X. Zhou, "Capacity of reproducing kernel spaces in learning theory," *IEEE Trans. Inf. Theory*, vol. 49, no. 7, pp. 1743–1752, Jul. 2003.
- [31] P. L. Bartlett, "The sample complexity of pattern classification with neural networks: The size of the weights is more important than the size of the network," *IEEE Trans. Inf. Theory*, vol. 44, no. 2, pp. 525–536, Mar. 1998.
- [32] F. Cucker and S. Smale, "On the mathematical foundations of learning," *Bull. Amer. Math. Soc.*, vol. 39, no. 1, pp. 1–49, Jan. 2001.



**Bin Zou** is currently a Professor with the Key Laboratory of Applied Mathematics, and the Faculty of Mathematics and Statistics, Hubei University, Wuhan, China. His current research interests include statistical learning theory, machine learning, and pattern recognition.



**Chen Xu** is currently an Assistant Professor with the Department of Mathematics and Statistics, University of Ottawa, Ottawa, ON, Canada.



**Yang Lu** received the M.S. and B.S. degrees in software engineering from the University of Macau, Zhuhai, China, in 2012 and 2014, respectively. He is currently pursuing the Ph.D. degree in computer science with Hong Kong Baptist University, Hong Kong.



**Yuan Yan Tang** (S'88–M'88–SM'96–F'04) received the B. Sc. degree in electrical and computer engineering from Chongqing University, Chongqing, China, the M.Eng. degree in electrical engineering from the Beijing Institute of Post and Telecommunications, Beijing, China, and the Ph.D. degree in computer science from Concordia University, Montréal, QC, Canada.

He is currently a Chair Professor with the Faculty of Science and Technology, University of Macau, Zhuhai, China, and a Professor, an Adjunct Professor, an Honorary Professor with several institutes including Chongqing University, Concordia University, and Hong Kong Baptist University, Hong Kong. His current research interests include wavelet theory and applications, pattern recognition, image processing, document processing, artificial intelligence, and Chinese computing.

Dr. Tang is the IAPR Fellow. He is the Founder and the General Chair of the series International Conferences on Wavelets Analysis and Pattern Recognition. He is the Founder and an Editor-in-Chief of the International Journal on *Wavelets, Multiresolution, and Information Processing*, an Associate Editor in-Chief of the International Journal on *Frontiers of Computer Science*. He is the Founder and the Chair of the pattern Recognition Committee in the IEEE SMC.



**Jie Xu** is currently a Professor with the Faculty of Computer Science and Information Engineering, Hubei University, Wuhan, China. Her current research interests include machine learning and pattern recognition.



**Xinge You** (M'08–SM'10) is currently a Professor with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China. His current research interests include wavelets and its application, signal and image processing, and pattern recognition.