AVERAGING VERSUS VOTING: A COMPARATIVE STUDY OF STRATEGIES FOR DISTRIBUTED CLASSIFICATION

Donglin Wang, Honglan Xu and Qiang Wu*

Department of Mathematical Sciences
Middle Tennessee State University
1301 E Main Street, Murfreesboro, TN 37132, USA

(Communicated by Ting Hu)

Abstract. In this paper we proposed two strategies, averaging and voting, to implement distributed classification via the divide and conquer approach. When a data set is too big to be processed by one processor or is naturally stored in different locations, the method partitions the whole data into multiple subsets randomly or according to their locations. Then a base classification algorithm is applied to each subset to produce a local classification model. Finally, averaging or voting is used to couple the local models together to produce the final classification model. We performed thorough empirical studies to compare the two strategies. The results show that averaging is more effective in most scenarios.

1. Introduction. Classification is a critical research field in machine learning, data mining, and pattern recognition. The objective of a classification problem is to build a model that assigns each observation to one of several predefined categories. It finds tremendous successful applications in modern data science, for instance, bioinformatics, social networks, text recognition, image processing, and computer vision. As such, it has received extensive research in the literature and many effective approaches have been developed. Examples of widely used classification algorithms include support vector machines (SVMs), logistic regression, decision trees, deep and convolutional neural networks; see e.g. [22, 6, 7] and many references therein.

Due to the rapid development of information and network technology, we are in a big data era and the growth of computing facility power falls way behind the growth of the scale of data. Effective use of this rich data information has become one of the focuses of modern statistics and machine learning research. Distributed algorithms have been receiving increasing attentions for their power to handle large scale data; see e.g. [19, 27, 14, 9, 11, 12].

Among various distributed learning paradigms, the divide and conquer approach is efficient and provable with theoretical guarantees. It first partitions a big data set (which is too big to be processed by a single machine or is naturally located in different machines for privacy or confidentiality reasons) into multiple subsets. Then a base algorithm is applied to each subset to generate a local model. The final model will then be obtained by coupling the local models together. It has been studied

* Corresponding author: Qiang Wu.

2020 Mathematics Subject Classification. Primary: 68T05, 68T09, 68W15.

Key words and phrases. Distributed classification, divide and conquer, averaging, voting, support vector machine, logistic regression.

Corresponding author: Qiang Wu.
for a variety of learning tasks and proved asymptotically rate optimal, for instance, M-estimation [18], kernel ridge regression [27, 14], kernel spectral regression [9, 11], bias corrected regularization kernel network [10] and the minimum error entropy [12, 8].

In this paper we consider classification problems and propose to implement distributed classification via the divide and conquer approach. As we have so many choices for the base classification algorithms to generate local models for each subset, we will focus on the logistic regression and SVM in this paper. As for the coupling stage, we propose two strategies, averaging and voting. One of our purposes is to do a thorough comparison of these two strategies.

The rest of this paper is organized as follows. In Section 2 we describe the binary classification problem setting and the two algorithms, namely, the logistic regression and SVM. In Section 3 we describe our distributed classification methods in detail. In Section 4 we describe the data sets used for this study and perform thorough empirical studies on the effectiveness of the proposed methods. We close with conclusions and discussions in Section 5.

2. Classification: problem setting and algorithms. Let us focus on the binary classification problems where there are only two categories. Each observation is associated with a \( p \)-dimensional vector, with each element measuring a feature of the observation. The task of classification is to build a model that allows to assign each observation to one of the two categories with minimal errors. In machine learning, we use \( X \subset \mathbb{R}^p \) to represent the feature space (or input space) and \( Y = \{1, -1\} \) as the labels of two classes. A function \( f: X \rightarrow Y \) is called the classification model (or classifier). For each \( x \in X \), the occurrence of \( Y \) follows a conditional distribution \( P(y|x) \).

The distribution is unknown and what we have is a set of \( n \) paired observations \( D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \). The purpose of a classification problem is to build an accurate classifier from the data \( D \). The optimal classifier is known to be the Bayes rule:

\[
f^*(x) = \begin{cases} 
1, & \text{if } P(y = 1|x) > P(y = -1|x), \\
-1, & \text{if } P(Y = 1|x) < P(y = -1|x).
\end{cases}
\]

A classification algorithm is Bayes consistent if the classification accuracy of the obtained classifier converges to that of the Bayes rule as \( n \rightarrow \infty \). Many efficient algorithms have been proposed in the literature. In this paper we focus on the use of two famous ones, the logistic regression and SVM.

2.1. Logistic regression. Logistic regression belongs to the generalized linear model family. It models the quantity \( \log \left( \frac{P(y=1|x)}{P(y=-1|x)} \right) \) by a family of real valued functions \( f(x, \theta) \) where \( \theta \) is the parameter of the function. If linear models are used, we have \( f(x, \theta) = w^\top x + b \) with \( w \in \mathbb{R}^p \), \( b \in \mathbb{R} \) and \( \theta = (w, b) \in \mathbb{R}^{p+1} \). In logistic regression the parameter \( \theta \) is estimated by the maximum likelihood estimation. Note that \( f(x, \theta) = \log \left( \frac{P(y=1|x)}{P(y=-1|x)} \right) \) implies

\[
P(y|x) = \frac{1}{1 + e^{-f(x, \theta)}}.
\]

For the given data \( D \) the log-likelihood function is

\[
\ell(\theta|D) = -\sum_{i=1}^{n} \log \left( 1 + e^{-y_if(x_i, \theta)} \right).
\]
Note that
\[ f(x, \theta) > 0 \iff P(y = 1|x) > P(y = -1|x), \]
\[ f(x, \theta) < 0 \iff P(y = -1|x) > P(y = 1|x). \]

Let \( \hat{\theta}_D \) be the estimator that maximizes \( \ell(\theta|D) \). The classification of new observations can be decided by the sign of the function \( \hat{f}_D = f(x, \hat{\theta}_D) \).\(^1\) Note this is equivalent to the empirical risk minimization estimation \[ \hat{f}_D = \arg \min_{f(x, \theta)} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i, \theta)) \]
with a loss function \( L(y, t) = \log(1 + e^{-yt}) \).

2.2. Support vector machine. The support vector machine (SVM) for binary classification was first formulated in [4]. It was originally motivated by maximizing the margin between the two classes and proved to be an empirical minimization estimator with hinge loss \( L(y, t) = (1 - yt)_+ = \max(0, 1 - yt) \). It became popular quickly after its invention because of its excellent generalization performance and ability to handle high dimensional problems. Its computational and theoretical properties were extensively studied in the last two decades; see [22, 16, 20, 26, 24, 23, 5] and references therein for the sequential minimal optimization (SMO) algorithm, margin bounds, consistency and convergence analysis, and so on.

If the two classes are separable by hyperplanes, an optimal separating hyperplane is selected by maximizing the margin between the two classes, which corresponds to a hard margin optimization problem

\[
\min \frac{1}{2} \|w\|^2 \quad \text{subject to } y_i(w^\top x_i + b) \geq 1.
\]

If the two classes are not separable, slack variables are introduced and soft margin problem is solved:

\[
\min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} \xi_i \quad \text{subject to } y_i(w^\top x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0,
\]

where \( \xi_i \) are slack variables and \( C > 0 \) is a cost parameter. This can be rewritten as a Tikhonov regularized empirical risk minimization problem with hinge loss:

\[
(\hat{w}_D, \hat{b}_D) = \arg \min \frac{1}{n} \sum_{i=1}^{n} \left(1 - y_i(w^\top x_i + b)\right)_+ + \lambda \|w\|^2
\]

where \( \lambda = \frac{1}{2nC} \) is the regularization parameter. It is usually solved by considering its dual problem

\[
\max \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^\top x_j \quad \text{subject to } 0 \leq \alpha_i \leq C, \sum_{i=1}^{n} \alpha_i y_i = 0.
\]

The solution gives \( \hat{w}_D = \sum_{i=1}^{n} \alpha_i y_i x_i \). Then \( \hat{b}_D \) is determined by using support vectors and the estimated classifier is the sign of the function \( \hat{f}_D(x) = \hat{w}_D^\top x + \hat{b}_D \).

\(^1\) Another traditional way to label the two categories in a binary classification problem is to use 0 and 1. In this case, the representations of the conditional probability and the log-likelihood function are more complicated. Moreover, the label is not predicted directly by the sign of the classifier but a map from positive to 1 and negative to 0 is needed.
Notice that the dual problem (3) depends only on the pairwise inner product of input data. Such a property allows to extend SVM to nonlinear models via kernel tricks. Let $K : X \times X \to \mathbb{R}$ be a Mercer kernel, meaning that it is continuous, symmetric and semi positive definite. The kernel $K$ induces a reproducing kernel Hilbert space $\mathcal{H}_K$ which is the closure of the linear span $\{ f(\mathbf{x}) = \sum_{i=1}^n c_i K(\mathbf{x}_i, \mathbf{x}) : c_i \in \mathbb{R}, \mathbf{x}_i \in X \}$ equipped with the inner product satisfying $\langle K(\mathbf{x}_i, \mathbf{x}), K(\mathbf{x}_j, \mathbf{x}) \rangle_K = K(\mathbf{x}_i, \mathbf{x}_j)$. See [2] for more properties of Mercer kernels and reproducing kernel Hilbert spaces. The kernel SVM is given by

$$(\hat{g}_D, \hat{b}_D) = \arg\min_{g, b} \frac{1}{n} \sum_{i=1}^n (1 - y_i(g(\mathbf{x}_i) + b))_+ + \lambda \|g\|^2_K, \quad (4)$$

whose solution is $\hat{g}_D(\mathbf{x}) = \sum_{i=1}^n \alpha_i y_i K(\mathbf{x}_i, \mathbf{x})$ with $\alpha_i$ solved from its dual problem

$$\max \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i y_i \alpha_j y_j K(\mathbf{x}_i, \mathbf{x}_j) \quad \text{subject to } 0 \leq \alpha_i \leq \frac{1}{2n\lambda}, \sum_{i=1}^n \alpha_i y_i = 0.$$ 

Similar to the linear SVM case the offset term $\hat{b}_D$ needs to be solved separately using support vectors. The kernel based classifier is then given by the sign of $\hat{f}_D = \hat{g}_D + \hat{b}_D$.

Popular choices of the kernel $K$ include linear kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^\top \mathbf{x}_j$ and the Gaussian kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\sigma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$. It is obvious that if a linear kernel is used, the algorithm gives back the linear SVM (2). If the kernel $K$ is universal meaning that $\mathcal{H}_K$ is dense in the space $C(X)$ of continuous functions, the offset term $b$ is not necessary [21].

3. Distributed classification. Now we turn to implement distributed classification for large scaled data by the divide and conquer approach. For a given big data set $D$, we partition it into $k$ subsets $D = \bigcup_{j=1}^k D_j$ and train a local model $\hat{f}_j = \hat{f}_{D_j}$ for each subset $D_j$ using either logistic regression or SVM. Next we need to generate a final classifier by coupling $\hat{f}_j$ together. We propose two strategies as follows.

- **Voting.** Note that each $\hat{f}_j$ induces a classifier via its sign. Consequently, for each new observation $\mathbf{x} \in X$ we are able to label it $k$ times by applying all $k$ local classifiers. This gives $k$ predicted labels $\{\text{sign}(\hat{f}_j(\mathbf{x})) : j = 1, \ldots, k\}$. The voting strategy makes the final decision by counting which label appears more times. Mathematically if we use $I(E)$ to denote the indicator function meaning that $I(E) = 1$ if $E$ is true and $I(E) = 0$ if it is false, then the voting strategy produces the final classifier as

$$F_v(\mathbf{x}) = \begin{cases} 
1, & \text{if } \sum_{j=1}^k I(\text{sign}(\hat{f}_j(\mathbf{x})) = 1) > \sum_{j=1}^k I(\text{sign}(\hat{f}_j(\mathbf{x})) = -1), \\
-1, & \text{if } \sum_{j=1}^k I(\text{sign}(\hat{f}_j(\mathbf{x})) = 1) < \sum_{j=1}^k I(\text{sign}(\hat{f}_j(\mathbf{x})) = -1).
\end{cases}$$

- **Averaging.** Notice that all local models $\hat{f}_j$ learned from logistic regression or SVM are real valued functions. The averaging strategy will first compute the average of these functions

$$\bar{f}_D = \frac{1}{k} \sum_{j=1}^k \hat{f}_j.$$
It is assumed to be a good approximation to the global estimator $\hat{f}_D$ which is learned as if we are able to apply logistic regression or SVM on the whole data $D$. Then the sign of $\hat{f}_D$ is used as the final classifier for new observations, that is,

$$F_a(x) = \text{sign} \left( \hat{f}_D(x) \right).$$

Before moving forward, it is worth to remark that we have

$$F_v(x) = \text{sign} \left( \sum_{j=1}^{k} \text{sign}(f_j(x)) \right).$$

Therefore, voting strategy is actually averaging after local decisions while averaging strategy corresponds to averaging without local decisions. Consequently, if the base classification algorithm outputs a classifier directly instead of a real valued function (such as in some rule based algorithms or k-nearest neighbor method), both voting and averaging strategies will define the same final classifier, that is, $F_v = F_a$.

4. **Experiments.** In this section we test the effectiveness of the distributed classification approaches and make a thorough comparison between the two coupling strategies by applying them in real world applications. We selected eight different data sets from the UC Irvine Machine Learning Repository (https://archive.ics.uci.edu/ml/index.php) and the famous MNIST handwritten digits recognition data (http://yann.lecun.com/exdb/mnist/) for this purpose. They cover a variety of different fields, including finance, communications, education, medical treatment, transportation, imaging process, etc. This makes our comparative study representative and convincing.

The Default of Credit Card Clients data set displays the customers’ default payments in Taiwan where the task is to predict the credibility of customers based on their personal information such as the amount of given credit, education, gender, age, marital status, history of past payment, etc. [25]. The Wilt Data Set consists of image segments which are generated by segmenting pansharpened images and contain spectral information from the Quickbird multispectral image bands and texture information from the panchromatic image band [13]. The purpose is to detect diseased trees in high-resolution multispectral satellite imagery. This data is heavily imbalanced (74 observations for the “diseased trees” class vs 4,265 observations for “other land cover” class). APS Failure data set was collected from heavy scania trucks in everyday usage [15]. The task is to diagnose whether a truck failure is caused by a component of the air pressure system (APS) from 170 anonymous factors. This data set is heavily imbalanced and contains many missing values. We preprocess the data by removing 10 features with missing values about 50% or more and filling other missing values with mean value of known data. The MAGIC Gamma Telescope Data Set contains 10 image parameters obtained by processing pixel images generated by a Monte Carlo program for Ground-based imaging Cherenkov telescopes and is used to discriminate signal (class “gamma”) and background (class “hadrons”) [3]. The Spam Data Set contains 4,601 emails that use word frequencies or other statistics to detect whether an email is a spam or not. The Epileptic Seizures data contains 11,500 records, each representing one second EEG signal for an individual, and is used to classify the patients with epileptic seizure from health people [1]. The Wireless Indoor Localization Data Set is collected to perform experiments on how to use wifi signal strength on smartphones to determine indoor location [17]. There are four classes in this data. Since we focus
Table 1. Description of Data Sets and Classification Tasks

| Classification Task                  | Number of Observations | Number of Features |
|--------------------------------------|-------------------------|--------------------|
| Default of Credit Card Clients       | 30,000                  | 23                 |
| Wilt Diseased Tree Detection         | 4,889                   | 5                  |
| APS Failure                          | 60,000                  | 170                |
| MAGIC Gamma Telescope                | 19,020                  | 10                 |
| Spam Email Detection                 | 4,601                   | 57                 |
| Epileptic Seizures                   | 9,200                   | 178                |
| Wireless Localization \{1,2\} vs \{3,4\} | 2,000                   | 7                  |
| Student Evaluation \{1,2\} vs \{3,4,5\} | 5,046                   | 32                 |
| Handwritten Digits 5 vs 8            | 12,017                  | 786                |

on the binary classification problem in this paper, we group the four classes and consider the binary problem of classes \{1,2\} vs classes \{3,4\}. The Turkey Student Evaluation Data Set includes 5,046 useful student evaluation scores given by Gazi University in Ankara. The task is to predict the evaluation score based on four attributes of instructors/courses and feedbacks on 28 evaluation questions. This is an ordered classification problem with score values \{1,2,3,4,5\}. We create a binary classification problem by considering the below average class \{1,2\} vs average or above class \{3,4,5\}. The MNIST dataset contains 60,000 images of handwritten digits as training data and 10,000 images as test data. Each image consists of $28 \times 28 = 784$ gray-scale pixel intensities. We only consider the subproblem of digit 5 vs digit 8 in this study. We summarized in Table 1 the nine data sets and corresponding binary classification tasks.

For each data set we select 60\% as training data and the remaining 40\% as testing data. The training data are further divided into 11 subsets for distributed classification. We used the glm function in R programming to implement logistic regression. It is sensitive to multicolinearity and singularity of the data. We had to perform principal component analysis for APS Failure data and the Epileptic Seizures data and keep 35 top principal components to prevent it from broken. For SVM we used the liquidSVM package in R. Gaussian kernel was used and both the cost parameter and bandwidth parameter were selected by cross validation. SVM is more computational stable due to the regularization. We used all available features for classification tasks and did not do any further preprocessing other than those built in the package. The experiment was repeated 100 times and the mean classification accuracy are reported in Table 2 and Table 3 for distributed logistic regression and distributed SVM, respectively. We see that in a majority scenarios averaging strategy performs better than voting strategy. The hypothesis tests show that the difference between these two strategies are statistically significant even in many scenarios where the difference looks minimal. More importantly, we do see scenarios in which the difference is not significant but there are no scenarios in which voting is significantly better. This supports the safe use of averaging strategy in practice.

A plausible reason for the superiority of averaging over voting is conjectured as follows. For observations far away from the boundary, all local classifiers are expected to give values far away from zero and most of them should have the same sign, regardless the classification is correct or incorrect. Consequently, both voting
and averaging should perform similarly on these observations. For observations near the boundary, however, poor but reasonable local classifiers may give values close to zero with wrong signs. Their impact on the average value of all local classifiers will be minimal and the final model is still likely to give correct classification provided that there are several good local classifiers. But their impact could be boosted by the voting strategy so that the final model gives wrong decision. As a result, the more data near the boundary, the more likely averaging performs better than voting.

The results also show that the p-values for distributed SVM in Table 3 are almost consistently larger than those for distributed logistic regression in Table 2, indicating the performance of distributed logistic regression are more severely affected by the coupling strategies while distributed SVM is less affected. This is probably because SVM is more robust and the local classifiers produce less predicted values near zero.

5. Conclusions. In this paper we proposed two coupling strategies, voting and averaging, to implement distributed logistic regression and distributed SVM classification. They are applied to a variety of real world problems. A comparative study shows that averaging is more effective in most scenarios and therefore is recommended for use in practice.
To close, we remark that in some classification algorithms the real valued functions have their probabilistic or geometrical interpretations. Averaging strategy may also be specifically designed according to these interpretations. For instance, in logistic regression an alternative averaging strategy is to average the local estimates for the conditional probability by

$$P_D(y|x) = \frac{1}{k} \sum_{i=1}^{k} P_j(y|x) = \frac{1}{k} \sum_{j=1}^{k} \frac{1}{1 + e^{-yf_j(x)}}.$$ 

Since this is not doable for all classification algorithms, it is out of the scope of this paper.

REFERENCES

[1] R. G. Andrzejak, K. Lehnertz, F. Mormann, C. Rieke, P. David and C. E. Elger, Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state, Physical Review E, 64 (2001), 061907.

[2] N. Aronszajn, Theory of reproducing kernels, Trans. Amer. Math. Soc., 68 (1950), 337–404.

[3] R. K. Bock, A. Chilingarian, M. Gaug, F. Hakl, T. Hengstebeck, M. Jiřina, J. Klaschka, E. Kotrě, P. Savický, S. Towers, A. Vaičiulis and W. Wittek, Methods for multidimensional event classification: a case study using images from a cherenkov gamma-ray telescope, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 516 (2004), 511–528.

[4] C. Cortes and V. Vapnik, Support-vector networks, Machine Learning, 20 (1995), 273–297.

[5] F. Cucker and D. X. Zhou, Learning Theory: An Approximation Theory Viewpoint, Cambridge University Press, Cambridge, 2007.

[6] J. Friedman, T. Hastie and R. Tibshirani, The Elements of Statistical Learning, Data Mining, Inference, and Prediction, Springer Series in Statistics, Springer-Verlag, New York, 2001.

[7] I. Goodfellow, Y. Bengio and A. Courville, Deep learning, MIT Press, Cambridge, MA, 2016.

[8] X. Guo, T. Hu and Q. Wu, Distributed minimum error entropy algorithms, preprint, (2020).

[9] Z.-C. Guo, S.-B. Lin and D.-X. Zhou, Learning theory of distributed spectral algorithms, Inverse Problems, 33 (2017), 074009.

[10] Z.-C. Guo, L. Shi and Q. Wu, Learning theory of distributed regression with bias corrected regularization kernel network, Journal of Machine Learning Research, 18 (2017), 1–25.

[11] Z.-C. Guo, D.-H. Xiang, X. Guo and D.-X. Zhou, Thresholded spectral algorithms for sparse approximations, Analysis and Applications, 15 (2017), 433–455.

[12] T. Hu, Q. Wu and D.-X. Zhou, Distributed kernel gradient descent algorithm for minimum error entropy principle, Applied and Computational Harmonic Analysis, 49 (2020), 229–256.

[13] B. A. Johnson, R. Tateishi and N. T. Hoan, A hybrid pansharpening approach and multiscale object-based image analysis for mapping diseased pine and oak trees, International Journal of Remote Sensing, 34 (2013), 6969–6982.

[14] S.-B. Lin, X. Guo and D.-X. Zhou, Distributed learning with regularized least squares, Journal of Machine Learning Research, 18 (2017), 1–31.

[15] E. C. Ozan, E. Riabchenko, S. Kiranyaz and M. Gabbouj, An optimized k-nn approach for classification on imbalanced datasets with missing data, in International Symposium on Intelligent Data Analysis, Springer, (2016), 387–392.

[16] J. Platt, Fast training of support vector machines using sequential minimal optimization, Advances in Kernel Methods - Support Vector Learning, MIT Press, Cambridge, MA, (1999), 185–208.

[17] J. G. Rohra, B. Perumal, S. J. Narayanan, P. Thakur and R. B. Bhatt, User localization in an indoor environment using fuzzy hybrid of particle swarm optimization & gravitational search algorithm with neural networks, in Proceedings of Sixth International Conference on Soft Computing for Problem Solving, Springer, (2017), 286–295.

[18] J. D. Rosenblatt and B. Nadler, On the optimality of averaging in distributed statistical learning, Information and Inference: A Journal of the IMA, 5 (2016), 379–404.
[19] E. R. Sparks, A. Talwalkar, V. Smith, J. Kottalam, X. Pan, J. Gonzalez, M. J. Franklin, M. I. Jordan and T. Kraska, Mli: An api for distributed machine learning, in 2013 IEEE 13th International Conference on Data Mining, IEEE, (2013), 1187–1192.

[20] I. Steinwart, Support vector machines are universally consistent, Journal of Complexity, 18 (2002), 768–791.

[21] I. Steinwart and A. Christmann, Support Vector Machines, Springer Science & Business Media, 2008.

[22] V. Vapnik, Statistical learning theory, John Wiley & Sons, Inc., New York, 1998.

[23] Q. Wu, Y. Ying and D.-X. Zhou, Multi-kernel regularized classifiers, Journal of Complexity, 23 (2007), 108–134.

[24] Q. Wu and D.-X. Zhou, Analysis of support vector machine classification, Journal of Computational Analysis & Applications, 8 (2006), 99–119.

[25] I.-C. Yeh and C.-H. Lien, The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients, Expert Systems with Applications, 36 (2009), 2473–2480.

[26] T. Zhang, Statistical behavior and consistency of classification methods based on convex risk minimization, Annals of Statistics, 32 (2004), 56–85.

[27] Y. Zhang, J. C. Duchi and M. J. Wainwright, Divide and conquer kernel ridge regression: A distributed algorithm with minimax optimal rates, Journal of Machine Learning Research, 16 (2015), 3299–3340.

Received March 2020; revised May 2020.

E-mail address: donglin.wang@mtsu.edu
E-mail address: hx2g@mtmail.mtsu.edu
E-mail address: qwu@mtsu.edu