Journal of the Korean Data & Information Science Society
2015, **26**(6), 1557–1563

http://dx.doi.org/10.7465/jkdi.2015.26.6.1557
한국데이터정보과학회지

Multiclass LS-SVM ensemble for large data[†]

Hyungtae Hwang[1]

[1]Department of Statistics, Dankook University

Received 28 August 2015, revised 23 November 2015, accepted 24 November 2015

Abstract

Multiclass classification is typically performed using the voting scheme method based on combining binary classifications. In this paper we propose multiclass classification method for large data, which can be regarded as the revised one-vs-all method. The multiclass classification is performed by using the hat matrix of least squares support vector machine (LS-SVM) ensemble, which is obtained by aggregating individual LS-SVM trained on each subset of whole large data. The cross validation function is defined to select the optimal values of hyperparameters which affect the performance of multiclass LS-SVM proposed. We obtain the generalized cross validation function to reduce computational burden of cross validation function. Experimental results are then presented which indicate the performance of the proposed method.

Keywords: Ensemble, generalized cross validation function, hat matrix, least squares support vector machine, multiclass classification.

1. Introduction

Vapnik (1995, 1998) originally designed Support vector machine (SVM), which solves the weak point of neural network such as the existence of local minima in the area of statistical learning theory and structural risk minimization. SVM solutions are characterized by convex optimization problems. Despite of many successful application of SVM in classification and regression problem, training an SVM requires to solve a quadratic program (QP) problem. QP is to optimize a quadratic function over a polyhedron, defined by linear equations and/or inequalities, which is time memory expensive. Suykens and Vanderwalle (1999) proposed a modified version of SVM in a least squares (LS) sense for classification. In LS- SVM the solution is given by a linear system instead of QP problem. The fact that LS-SVM has explicit primal-dual formulations has lots of advantages including easy computation. For applications of SVM and LS-SVM see Shim and Hwang (2013) and Seok (2014).

The binary classification using SVM or LS-SVM is known to be well developed. Multiclass classification is typically performed using the voting scheme method based on combining binary classifications (Schölkopf *et al.*, 1995). Suykens and Vanderwalle (1999) proposed multiclass classification method using LS-SVM in a step but its linear equation are composed of linear equations corresponding to each of binary classifications. Weston and Watkins (1998)

[†] The present research was conducted by the research fund of Dankook University in 2015.
[1] Professor, Department of Applied Statistics, Dankook University, Gyeonggido 448-701, Korea. E-mail: hthwang@dankook.ac.kr

proposed the multiclass classification method using SVM which does not use a combination of binary classifications.

For large scale problems, Espinoza *et al.* (2005) proposed the fixed size LS-SVM by using the sparse approximation of nonlinear feature mapping function induced by kernel function, whose computation is based on the Nyström approximation (Williams and Seeger, 2001) and the quadratic Renyi entropy (Girolami, 2003).

In this paper we propose the multiclass classification method for large data, which is performed by using the hat matrix of LS-SVM ensemble. We define the cross validation (CV) function to select the optimal hyperparameters which affects the performance of multiclass classification and obtain the generalized cross validation (GCV) function for CV function.

The rest of paper is organized as follows. In Section 2 we propose LS-SVM ensemble for large data. In Section 3 we propose the multiclass classification method for large data by LS-SVM ensemble and GCV function, respectively. In Section 4 we perform the numerical studies with real data sets. In Section 5 we give the conclusions.

2. LS-SVM ensemble for large data

2.1. LS-SVM

Let the training data set be denoted by $\boldsymbol{x} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, with each input variable $\boldsymbol{x}_i \in R^d$, the output $y_i \in R$, and the test data by $\boldsymbol{x}_t$. We consider the case of nonlinear regression. Then, we take the form,

$$f(\boldsymbol{x}) = \boldsymbol{w}'\phi(\boldsymbol{x}) + b$$

where the term b is a bias term. Here the feature mapping function $\phi(\cdot) : R^d \rightarrow R^{d_f}$ maps the input space to the higher dimensional feature space where the dimension d_f is defined in an implicit way.

The optimization problem is defined with a penalty parameter $C > 0$ as follows:

$$\text{minimize } \frac{1}{2}\boldsymbol{w}'\boldsymbol{w} + \frac{C}{2}\sum_{i=1}^{n} e_i^2 \tag{2.1}$$

over $\{\boldsymbol{\omega}, b, \boldsymbol{e}\}$ subject to equality constraints,

$$y_i = \boldsymbol{w}'\phi(\boldsymbol{x}_i) + b + e_i, \ i = 1, \cdots, n.$$

The Lagrangian function can be constructed as follows:

$$L(\boldsymbol{w}, b, e : \alpha) = \frac{1}{2}\boldsymbol{w}'\boldsymbol{w} + \frac{C}{2}\sum_{i=1}^{n} e_i^2 - \sum_{i=1}^{n} \alpha_i(\boldsymbol{w}'\phi(\boldsymbol{x}_i) + b + e_i - y_i), \tag{2.2}$$

where α_i's are the Lagrange multipliers. The conditions for optimality given by

$$\begin{aligned}
&\frac{\partial L}{\partial \boldsymbol{w}} = 0 \ \rightarrow \ \boldsymbol{w} = \sum_{i=1}^{n} \alpha_i \phi(\boldsymbol{x}_i) \\
&\frac{\partial L}{\partial b} = 0 \ \rightarrow \ \sum_{i=1}^{n} \alpha_i = 0 \\
&\frac{\partial L}{\partial e_i} = 0 \ \rightarrow \ \alpha_i = Ce_i, \ i = 1, \cdots, n \\
&\frac{\partial L}{\partial e_i} = 0 \ \rightarrow \ \boldsymbol{w}'\phi(\boldsymbol{x}_i) + b + e_i - y_i = 0, \ i = 1, \cdots, n,
\end{aligned}$$

lead to the linear equation,

$$\begin{bmatrix} K + \boldsymbol{I}/C & \mathbf{1} \\ \mathbf{1}' & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} \ = \ \begin{bmatrix} \boldsymbol{y} \\ 0 \end{bmatrix}, \tag{2.3}$$

where $\boldsymbol{y} = (y_1, \cdots, y_n)', \mathbf{1} = (1, \cdots, 1)', \boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_n)'$, and $K = \phi(\boldsymbol{x})'\phi(\boldsymbol{x}) = K(\boldsymbol{x}, \boldsymbol{x})$, which are obtained from the application of Mercer's conditions (1909). Several choices of the kernel $K(\cdot, \cdot)$ are possible.

Solving the linear equation (2.3) the optimal values of bias and Lagrange multipliers, b and α_i's are obtained, then the optimal regression function for the test data $\boldsymbol{x}_t \in R^d$ is obtained as follows:

$$\widehat{y}(\boldsymbol{x}_t) = \sum_{i=1}^{n} K(\boldsymbol{x}_t, \boldsymbol{x}_i)\alpha_i + b = K(\boldsymbol{x}_t, \boldsymbol{x})\boldsymbol{\alpha} + b. \tag{2.4}$$

Note that it can be easily shown that Lagrange multipliers of LS-SVM for binary classification are identical to product of diagonal matrix of $\boldsymbol{y}$ and Lagrange multipliers of LS-SVM for regression obtained from equation (2.3), when $\boldsymbol{y}$ consists of class labels -1 and 1. That is, if $\boldsymbol{y}$ consists of class labels -1 and 1, $\widehat{y}(\boldsymbol{x}_{ti})$'s obtained by LS-SVM for regression and classification are identical.

Thus, for the binary classification, each observation of the test data $\boldsymbol{x}_t$ can be classified into either class according to the sign of $\widehat{y}(\boldsymbol{x}_t)$ in (2.4). We use LS-SVM for regression, instead of LS-SVM for classification, to approximate the cross validation function easily.

2.2. LS-SVM ensemble for large data

In LS-SVM, we need the inverse of $(K(\boldsymbol{x}, \boldsymbol{x}) + I/C)^{-1}$, which is almost impossible for very large data. Here we propose LS-SVM ensemble for large data. Instead training one LS-SVM on whole data at a time, we train individual LS-SVM on each subsets which are obtained by randomly dividing whole data and aggregate them to obtain the optimal regression function for the test data $\boldsymbol{x}_t \in R^d$ as follows:

$$\widehat{y}(\boldsymbol{x}_t) = \frac{1}{M}\sum_{j=1}^{M}(K(\boldsymbol{x}_t, \boldsymbol{x}^j)\boldsymbol{\alpha}^j + b^j), \tag{2.5}$$

where $\boldsymbol{\alpha}^j$ and b^j are computed from the linear equation (2.3) using the jth randomly chosen subset $(\boldsymbol{x}^j, \boldsymbol{y}^j)$ of whole data such that $\boldsymbol{x} = \cup_{j=1}^{M} \boldsymbol{x}^j$, $\boldsymbol{x}^j \cap \boldsymbol{x}^l = \phi$ if $j \neq l$. LS-SVM ensemble is inspired by the idea of the bagging (Breiman, 1996), which is known to improve the stability and reduces variance and help to avoid overfitting.

In LS-SVM ensemble, we need the inverse of $\left(K(\boldsymbol{x}^j, \boldsymbol{x}^j) + I/C\right)^{-1}$ for $n_j << n$, which enables to train LS-SVM on large data.

3. Multiclass Classification

3.1. Multiclass classification by LS-SVM

In this section we give simple overview on multiclass classification by LS-SVM using one-against-all method. Let the training data set be denoted by $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, with each input vector $\boldsymbol{x}_i \in R^d$ and the class label $y_i \in \{1, 2, \cdots, m\}$, where m is number of classes. For multiclass classification using one-against-all method, we transform $\boldsymbol{y}$ into $n \times m$ matrix $\boldsymbol{Y}$ which consists of 1 and -1 such that $Y_{ik} = 1$ and $Y_{il} = -1$ for $k \neq l$ implies ith observation belongs to the kth class. We have m LS-SVMs for binary classification with $\{(\boldsymbol{x}_i, Y_{ik})\}_{i=1}^n$ for $k = 1, \cdots, m$.

From the linear equation,

$$\begin{bmatrix} K + \boldsymbol{I}/C & \mathbf{1} \\ \mathbf{1}' & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_{\cdot k} \\ b_k \end{bmatrix} = \begin{bmatrix} \boldsymbol{Y}_{\cdot k} \\ 0 \end{bmatrix}, \ k = 1, 2, \cdots, m, \tag{3.1}$$

where $\boldsymbol{Y}_{\cdot k}$ is the kth column of $\boldsymbol{Y}$, the optimal bias and Lagrange multipliers, b_k and $\boldsymbol{\alpha}_{\cdot k}$ are obtained. For the test data $\boldsymbol{x}_t$,

$$\widehat{Y}_k(\boldsymbol{x}_t) = K(\boldsymbol{x}_t, \boldsymbol{x})\boldsymbol{\alpha}_{\cdot k} + b_k, \ k = 1, \cdots, m. \tag{3.2}$$

If $\widehat{Y}_k(\boldsymbol{x}_t) > 0$ and $\widehat{Y}_l(\boldsymbol{x}_t) < 0$ for $k \neq l$then the test data $\boldsymbol{x}_t$ is classified into the kth class.

3.2. Model selection of multiclass LS-SVM

The functional structures of multiclass LS-SVM is characterized by hyper parameters, the penalty parameter C and the kernel parameters. To select the optimal hyperparameters of multiclass LS-SVM, we define the cross validation (CV) function as follows:

$$CV(\boldsymbol{\lambda}) = \frac{1}{n} \sum_{i=1}^{n} (Y_{ik_i} - \widehat{Y}_{ik_i}^{(-i)}(\boldsymbol{\lambda}))^2, \tag{3.3}$$

where $\boldsymbol{\lambda}$ is the set of hyperparameters and $\widehat{Y}_{im_i}^{(-i)}(\boldsymbol{\lambda})$ is the predicted value of Y_{ik_i} obtained from data without ith observation. Here k_i is the column number of the ith row of Y such that $Y_{ik_i} = 1$, which implies that the ith observation belongs to k_ith class. Since for each candidates of hyperparameters, $\widehat{Y}_{ik_i}^{(-i)}(\boldsymbol{\lambda})$ for $i = 1, \cdots, n$, should be evaluated, selecting parameters using CV function is computationally formidable.

By leaving-out-one lemma (Kimeldorf and Wahba, 1971),

$$(Y_{ik_i} - \widehat{Y}_{ik_i}^{(-i)}) - (Y_{ik_i} - \widehat{Y}_{ik_i}) = \widehat{Y}_{ik_i} - \widehat{Y}_{ik_i}^{(-i)} \approx \frac{\partial \widehat{Y}_{ik_i}}{\partial Y_{ik_i}} (Y_{ik_i} - \widehat{Y}_{ik_i}^{(-i)})$$

we have

$$(Y_{ik_i} - \widehat{Y}_{ik_i}^{(-i)}) \approx \frac{Y_{ik_i} - \widehat{Y}_{ik_i}}{1 - \dfrac{\partial \widehat{Y}_{ik_i}}{\partial Y_{ik_i}}} \text{ and } \widehat{Y}_{ik_i} = H_i \boldsymbol{Y}_{.k_i},$$

where H_i is the ith row of hat matrix $H(\boldsymbol{x}, \boldsymbol{x})$ such that $\widehat{\boldsymbol{Y}}_{.k} = H(\boldsymbol{x}, \boldsymbol{x})\boldsymbol{Y}_{.k}$,

$$H(\boldsymbol{x}, \boldsymbol{x}) = (K, \mathbf{1})\begin{pmatrix} (K + I/C)^{-1} - (K + I/C)^{-1}\mathbf{1}(\mathbf{1}'(K + I/C)^{-1}\mathbf{1})\mathbf{1}'(K + I/C)^{-1} \\ (\mathbf{1}'(K + I/C)^{-1}\mathbf{1})^{-1}\mathbf{1}'(K + I/C)^{-1} \end{pmatrix}.$$

Then the ordinary cross validation (OCV) function can be obtained as follows:

$$OCV(\boldsymbol{\lambda}) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1 - \widehat{Y}_{ik_i}(\boldsymbol{\lambda})}{1 - \dfrac{\partial \widehat{Y}_{ik_i}}{\partial Y_{im_i}}}\right)^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1 - \widehat{Y}_{ik_i}(\boldsymbol{\lambda})}{1 - h_{ii}(\boldsymbol{\lambda})}\right)^2, \tag{3.4}$$

where $h_{ii}(\boldsymbol{\lambda})$ for $i = 1, \cdots, n$, is the ith diagonal element of the hat matrix $H = H(\boldsymbol{x}, \boldsymbol{x})$. By replacing $h_{ii}(\boldsymbol{\lambda})$ in (3,4) by $tr(H)/n$, the generalized cross validation (GCV) function can be then obtained as follows:

$$GCV(\boldsymbol{\lambda}) = \frac{n\sum\limits_{i=1}^{n}(1 - \widehat{Y}_{ik_i}(\boldsymbol{\lambda}))^2}{(n - tr(H))^2}. \tag{3.5}$$

3.3. Multiclass classification by LS-SVM ensemble for large data

In this section we use the hat matrix for the test data $\boldsymbol{x}_t$ to avoid to solve m linear equations in (3.1). Here (3.2) can be rewritten as

$$\begin{aligned} \widehat{Y}_k(\boldsymbol{x}_t) =& K(\boldsymbol{x}_t, \boldsymbol{x})\boldsymbol{\alpha}_{,k} + b_k = (K_t, 1)\begin{pmatrix} \boldsymbol{\alpha}_{.k} \\ b_k \end{pmatrix} \\ =& (K_t, 1)\begin{pmatrix} (K+I/C)^{-1} - (K+I/C)^{-1}\mathbf{1}(\mathbf{1}'(K+I/C)^{-1}\mathbf{1})\mathbf{1}'(K+I/C)^{-1} \\ (\mathbf{1}'(K+I/C)^{-1}\mathbf{1})^{-1}\mathbf{1}'(K+I/C)^{-1} \end{pmatrix}\boldsymbol{Y}_{.k} \\ =& (K_t, 1)H_0(\boldsymbol{x})\boldsymbol{Y}_{.k} = H(\boldsymbol{x}_t, \boldsymbol{x})\boldsymbol{Y}_{.k} \text{ for } k = 1, \cdots, m. \end{aligned} \tag{3.6}$$

Since $H(\boldsymbol{x}_t, \boldsymbol{x})$ does not depend on $\boldsymbol{Y}_{.k}$ we can write $\widehat{Y}(\boldsymbol{x}_t)$ for the test data $\boldsymbol{x}_t$,

$$\widehat{Y}(\boldsymbol{x}_t) = H(\boldsymbol{x}_t, \boldsymbol{x})\boldsymbol{Y}, \tag{3.7}$$

where

$$\widehat{Y}(\boldsymbol{x}_t) = (\widehat{Y}_1(\boldsymbol{x}_t), \cdots, \widehat{Y}_m(\boldsymbol{x}_t)).$$

Thus, we need not to solve m linear equations in (3.1) but once as (3.6).

For large data we use $(\boldsymbol{x}^j, \boldsymbol{Y}^j)$ for $j = 1, \cdots, M$then $\widehat{Y}(\boldsymbol{x}_t)$ for the test data $\boldsymbol{x}_t$ can expressed as follows:

$$\widehat{Y}(\boldsymbol{x}_t) = \frac{1}{M}(K_t, 1) \sum_{j=1}^{M} H_0(\boldsymbol{x}^j)\boldsymbol{Y}^j. \tag{3.8}$$

In training the individual LS-SVM on $(\boldsymbol{x}^j, \boldsymbol{Y}^j)$ for $j = 1, \cdots, M$, the optimal values of hyperparameters are chosen form GCV function (3.5).

4. Numerical Studies

We illustrate the performance of the proposed method through 3 real data sets available from UCI Machine Learning Depository (http://archive.ics.uci.edu/ml), which are wine data set, cardiotocography data set, and Sensor readings4 data set. For the numerical studies we implement MATLAB R2006b over Core (TM) running at 3.60GHz. The radial basis function (RBF) kernel are used for LS-SVM, where the RBF kernel is defined as

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left(-\frac{1}{\sigma^2}(\boldsymbol{x}_1 - \boldsymbol{x}_2)'(\boldsymbol{x}_1 - \boldsymbol{x}_2)\right),$$

where σ^2 is a bandwidth parameter.

To illustrate the multiclass classification performance of multiclass LS-SVM ensemble, we run multiclass LS-SVM and the Naive Bayes classifier, and compare misclassification rates each other. We randomly divide the whole data set into the training data set and the test data set. The averages of 100 misclassification rates from multiclass LS-SVM, the Naive Bayes classifier and multiclass LS-SVM ensemble are obtained from each test data set. The penalty parameter and bandwidth parameter, (C, σ^2) are obtained from training data set by GCV function (3.5).

Wine data set of 3 classes, which is from results of wines grown in the same region in Italy but derived from three different cultivars, has 12 variables and 178 observations. The training data consist of 144 observations and the test data consist of 34 observations. We use 3 random subsets of whole training data set for multiclass LS-SVM ensemble.

Cardiotocography data set of 3 classes, which consists of measurements of fetal heart rate and uterine contraction features on cardiotocograms classified by expert obstetricians. There are 21 input variables and 2126 observations. The training data consist of 1800 observations and the test data consist of 326 observations. Since the within-class variance in the sixth input variable is not always positive, we cannot train the Naive Bayes classifier on whole training data set. We use 3 random subsets of whole training data set for multiclass LS-SVM ensemble.

The averages and standard errors of 100 misclassification rates on 100 test data sets by multiclass LS-SVM, multiclass LS-SVM ensemble and the Naive Bayes classifier are shown in Table 4.1. From the results of these 2 examples we can see that multiclass LS-SVM and multiclass LS-SVM ensemble have the similar classification performance.

Sensor readings4 data set of 4 classes, which consists of robot navigates through the room following the wall in a clockwise direction for 4 rounds using 24 ultrasound sensors arranged

circularly around its waist. There are 4 input variables and 5456 observations. The training data consist of 5000 observations and the test data consist of 456 observations. Due to out of memory of MATLAB R2006b we cannot train multiclass LS-SVM on whole training data set. We use 10 random subsets of whole training data set for multiclass LS-SVM ensemble.

The averages and standard errors of 100 misclassification rates on 100 test data sets by the Naive Bayes classifier and multiclass LS-SVM ensemble are shown in Table 4.1. From the results we can see that multiclass LS-SVM ensemble have the better classification performance than the Naive Bayes classifier on Sensor readings4 data set.

Table 4.1 The misclassification error rates for multiclass LS-SVM ensemble and other methods (standard error in parenthesis)

	LS-SVM	LS-SVM ensemble	Naive Bayes
Wine	0.0315 (0.0031)	0.0279 (0.0029)	0.0450 (0.0033)
Cardiotocography	0.2230 (0.0023)	0.2249 (0.0023)	-
Sensor readings4	-	0.0390 (0.0008)	0.1076 (0.0014)

5. Conclusions

Through the examples we showed that multiclass LS-SVM ensemble shows the satisfying results, which is simple approaches to modelling the multiclass classification problem for large data. In future work, we investigate the optimal number of random subsets of whole data for multiclass LS-SVM ensemble.

References

Breiman, L. (1996). Bagging predictors. *Machine Learning*, **24**, 123-140.

Espinoza, M., Suykens, J.A.K. and De Moor, B. (2005). Load forecasting using least squares Support vector machines. *Lecture Notes in Computer Science*, **3512**, 1018-1026.

Girolami, M. (2003). Orthogonal series density estimation and kernel eigenvalue problem. *Neural Computation*, **14**, 669-688.

Kimeldorf, G. S. and Wahba, G. (1971). Some results on Tchebycheffian spline functions. *Journal of Mathematical Analysis and its Applications*, **33**, 82-95.

Mercer, J. (1909). Functions of positive and negative type and their connection with theory of integral equations. *Philosophical Transactions of Royal Society of London A*, 415-446.

Schölkopf, B., Burges, C. and Vapnik, V. (1995). Extracting support data for a given task. *In Proceedings of First Conference on Knowledge Discovery and Data Mining* , 252-257, Menlo Park, CA.

Seok, K. H. (2014). Semi-supervised classification with LS-SVM formulation. *Journal of the Korean Data & Information Science Society*, **21**, 461-470.

Shim, J. and Hwang, C. (2013). Expected shortfall estimation using kernel machines. *Journal of the Korean Data & Information Science Society*, **24**, 625-636.

Suykens, J. A. K. and Vanderwalle, J. (1999). Least square support vector machine classifier. *Neural Processing Letters*, **9**, 293-300.

Suykens, J. A. K. and Vandewalle, J. (1999). Multiclass least squares support vector machines. In *Proceeding of the International Joint Conference on Neural Networks*,, 900-903, Washington DC.

Vapnik, V. N. (1995). *The nature of statistical learning theory*, Springer, New York.

Vapnik, V. N. (1998). *Statistical learning theory*, Springer, New York.

Weston, J. and Watkins, C. (1998). *Multi-class SVM*, Technical Report 98-04, Royal Holloway University, London.

Williams, C. K. I. and Seeger, M. (2001). Using the Nyström method to speed up kernel machines. In *Proceeding of Neural Information Processing Systems Conference 13*, 682-699, MIT press.