Optimizing Kernel Parameters by Second-Order Methods

Shigeo Abe

Kobe University - Graduate School of Engineering
Kobe, Japan

Abstract. Radial basis function network (RBF) kernels are widely used for support vector machines (SVMs). But for model selection of an SVM, we need to optimize the kernel parameter and the margin parameter by time-consuming cross validation. In this paper we propose determining parameters for RBF and Mahalanobis kernels by maximizing the class separability by the second-order optimization. For multi-class problems, we determine the kernel parameters for all the two-class problems and set the average value of the parameter values to all the kernel parameters. Then we determine the margin parameter by cross-validation. By computer experiments of multi-class problems we show that the proposed method works to select optimal or near optimal parameters.

1 Introduction

Support vector machines (SVMs) have been used for various applications as a powerful tool for pattern classification. The success of SVMs is based on (1) mapping the input space to a high-dimensional feature space, and (2) the maximization of the margin between two classes in the feature space.

One of the advantages of SVMs is that we can improve generalization ability by proper selection of kernels. In most cases polynomial kernels and radial basis function network (RBF) kernels are used. Mahalanobis kernels [1, 2, 3], which exploit the data distribution information more than RBF kernels do are also used.

The major problem is that we need to optimize the structure of an SVM, namely, model selection. There are many approaches to easy model selection using error bounds [4, 5, 6]. Another approach uses a criterion such as kernel discriminant analysis (KDA) [7] and the distance between class centers [8], which is a simplified version of KDA. In [3], fast model selection of Mahalanobis kernels by line search is discussed for two-class problems.

In this paper, we use the distance between class centers [8] as a criterion and determine the kernel parameters for RBF and Mahalanobis kernels by the second-order method. For a multi-class problem, we generally use one-against-all or pairwise classification, each of which is composed of several two-class problems. If we set different values to the kernels for the two-class problems, the feature spaces associated with the two-class problems are different. Thus, comparison of the values of the decision functions becomes meaningless. Thus, to avoid this, we set the average of the kernel parameters determined by the second-order method to kernel parameters. Then, we determine the margin parameters by cross-validation.

In Section 2, we review SVMs and RBF and Mahalanobis kernels for pattern classification. And in Section 3 we discuss model selection of RBF kernels and Mahalanobis kernels and an extension to multi-class problems. Finally in Section 4, we evaluate the proposed method for RBF and Mahalanobis kernels.

2 Support Vector Machines

Let m-dimensional inputs $\mathbf{x}_i$ $(i = 1, \ldots, M)$ belong to Class 1 or 2 and the associated labels be $y_i = 1$ for Class 1 and -1 for Class 2. To enhance separability, the input space is mapped into the high-dimensional dot-product space called feature space. Let the l-dimensional mapping function be $\mathbf{g}(\mathbf{x})$. If the dot product in the feature space is expressed by $H(\mathbf{x}, \mathbf{x}') = \mathbf{g}^T(\mathbf{x})\mathbf{g}(\mathbf{x})$, $H(\mathbf{x}, \mathbf{x}')$ is called kernel function, and we do not need to explicitly treat the feature space.

The decision function in the feature space is given by

$$D(\mathbf{x}) = \sum_{i=1}^{M} \alpha_i y_i H(\mathbf{x}_i, \mathbf{x}) + b, \tag{1}$$

where b is a bias term and α_i are dual variables associated with $\mathbf{x}_i$ and are obtained by solving the following quadratic program: Maximize

$$Q(\boldsymbol{\alpha}) = \sum_{i=1}^{M} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{M} \alpha_i \, \alpha_j \, y_i \, y_j \, H(\mathbf{x}_i, \, \mathbf{x}_j) \tag{2}$$

subject to the constraints

$$\sum_{i=1}^{M} y_i \, \alpha_i = 0, \quad C \geq \alpha_i \geq 0 \quad \text{for} \quad i = 1, \ldots, M, \tag{3}$$

where C is the margin parameter that determines the tradeoff between the maximization of the margin and minimization of the classification error.

In our study we use RBF kernels:

$$H(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\gamma}{m} \|\mathbf{x} - \mathbf{x}'\|^2\right), \tag{4}$$

where γ is a slope control parameter and Mahalanobis kernels:

$$H(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\delta}{m} \, (\mathbf{x} - \mathbf{x}')^T Q^{-1} \, (\mathbf{x} - \mathbf{x}')\right), \tag{5}$$

where $\delta \, (> 0)$ is the scaling factor to control the Mahalanobis distance, Q is the covariance matrix given by

$$Q = \frac{1}{M} \sum_{i=1}^{M} (\mathbf{x}_i - \mathbf{c}) \, (\mathbf{x}_i - \mathbf{c})^T, \qquad \mathbf{c} = \frac{1}{M} \sum_{i=1}^{M} \mathbf{x}_i. \tag{6}$$

If we use the full covariance matrix, it will be time-consuming for a large number of input variables. Thus in the following we only consider Mahalanobis kernels with diagonal covariance matrices.

3 Model Selection

3.1 Idea

Since a multi-class problem is converted to several two-class problems, first we consider a two-class problem. For a given classification problem, it is expected that if the kernel parameter is chosen so that the class centers are maximally separated, the margin between classes will also be maximized. And it will improve the generalization ability if the SVM is used. Therefore, instead of the cross-validation for two parameters simultaneously, we consider optimizing parameters one by one; namely, first we determine the kernel parameter by maximizing the distance between class centers and then we determine the margin parameter by cross-validation. We call this line search with the second-order method.

For multi-class problems, if we set different values of kernel parameters to different two-class problems, feature spaces associated with two-class problems are different. Thus, comparison of the outputs of the decision functions will become useless. To avoid this problem, we need to set the same parameter value for all the two-class problems. The value needs to be as near as possible to each kernel parameter, which is the average of the kernel parameters.

3.2 Maximizing the Inter-Class Distance

3.2.1 Inter-Class Distance

The square distance between the centers of Classes 1 and 2, $\mathbf{c_1}$ and $\mathbf{c}_2$, is given by

$$\begin{aligned} d^2(p) &= \|\mathbf{c}_1 - \mathbf{c}_2\|^2 \\ &= \left(\frac{1}{N_1}\sum_{i\in X_1}\mathbf{g}(\mathbf{x}_i) - \frac{1}{N_2}\sum_{i\in X_2}\mathbf{g}(\mathbf{x}_i)\right)^T \left(\frac{1}{N_1}\sum_{i\in X_1}\mathbf{g}(\mathbf{x}_i) - \frac{1}{N_2}\sum_{i\in X_2}\mathbf{g}(\mathbf{x}_i)\right) \\ &= \frac{1}{N_1^2}\sum_{i,j\in X_1} H(\mathbf{x}_i,\mathbf{x}_j) - \frac{2}{N_1\,N_2}\sum_{\substack{i\in X_1\\ j\in X_2}} H(\mathbf{x}_i,\mathbf{x}_j) \\ &\quad + \frac{1}{N_2^2}\sum_{i,j\in X_2} H(\mathbf{x}_i,\mathbf{x}_j), \end{aligned} \tag{7}$$

where p is a kernel parameter: either δ or γ, X_1 and X_2 are sets of the training data for Classes 1 and 2, respectively, and N_1 and N_2 are the numbers of data for Classes 1 and 2, respectively.

Now consider how $d^2(p)$ behaves as p changes from 0 to infinity. Because the behaviors of $d^2(\gamma)$ and $d^2(\delta)$ are similar, in the following we investigate $d^2(\gamma)$.

We assume that training data are distinct. Namely, $\mathbf{x}_i \neq \mathbf{x}_j$ for $i \neq j$. Then for $\gamma = 0$, $d^2(\gamma) = 0$ and

$$\lim_{\gamma \to \infty} d^2(\gamma) = \frac{1}{N_1} + \frac{1}{N_2}. \tag{8}$$

The first and the third terms on the right-hand side of (7) monotonically decrease while the second term monotonically increases. If the sum of the first and the third terms decays faster than the second term increases, $d^2(\gamma)$ monotonically increases and reaches a plateau as the value of γ increases (Notice that $d^2(\gamma) \geq 0$). But if the sum of the first and third terms decays slower, there is a peak in $[0, \infty]$. This occurs if the distance of two data in a class is in general shorter than the distance of two data belonging to different classes. In classification problems, this condition is usually satisfied. Thus, there is a maximum distance of $d(\gamma)$ for $\gamma \in [0, \infty)$.

3.2.2 *Determining Parameters by the Second-Order Method*

Assuming that there is a peak of $d^2(p)$ in $p \in [0, \infty]$, we obtain the value of p that satisfies $\partial d^2(p)/\partial p = 0$ by the second-order method. Let $p + \Delta p$ satisfies $\partial d^2(p + \Delta p)/\partial p = 0$. The second-order approximation of $\partial d^2(p + \Delta p)/\partial p = 0$ is given by

$$\frac{\partial d^2(p)}{\partial p} + \frac{\partial^2 d^2(p)}{\partial p^2} \Delta p + \frac{1}{2} \frac{\partial^3 d^2(p)}{\partial p^3} (\Delta p)^2 = 0. \tag{9}$$

Solving (9) for Δp, we obtain

$$\Delta p^{\pm} = \left(-\frac{\partial^2 d^2(p)}{\partial p^2} \pm \sqrt{\left(\frac{\partial^2 d^2(p)}{\partial p^2} \right)^2 - 2 \frac{\partial d^2(p)}{\partial p} \frac{\partial^3 d^2(p)}{\partial p^3}} \right) / \frac{\partial^3 d^2(p)}{\partial p^3}. \tag{10}$$

Now we discuss which of $\Delta p^{\pm}$ to take. Let $\partial d^2(p)/\partial p = 0$ at $a\,(> 0)$. For the whole range of p, $\partial^3 d^2(p)/\partial p^3 > 0$. We divide the range of p into two intervals: $(0, a)$ and $[a, \infty)$. Now we consider the correction Δp for each of the intervals:

1. Interval $(0, a)$. In this interval $\partial d^2(p)/\partial p \geq 0$ and $\partial^2 d^2(p)/\partial p^2 < 0$. Thus, if the term in the square root in (10) is non-negative, $\Delta p^{\pm}$ are non-negative and we take the smaller solution, Δp^-, as the correction since $p + \Delta p^-$ is nearer to the peak than $p + \Delta p^+$.

2. Interval (a, ∞). Because $\partial d^2(p)/\partial p < 0$, the term in the square root in (10) is positive. Thus, $\Delta p^- < 0$ and $\Delta p^+ > 0$ and we take Δp^- as the correction.

Starting from a small positive value of p (e.g., 0.01), we calculate the correction according to the above procedure and modify p until the correction is sufficiently small (e.g., less than 10^{-5}).

Table 1: Parameter setting.

Data	RBF(g)		RBF(s)		Diag (l)		Diag (s)	
	γ	C	γ	C	C	δ	C	δ
Iris	0.1	5000	14.1	1	1	0.5	1	0.975
Numeral	5	10	15.2	1	10	0.3	1	1.58
Thyroid	5	10^5	95.2	7000	100	0.5	50	1.18
Blood cell	10	1000	57.7	10	10	0.6	10	1.20
H-50	10	500	20.3	50	50	0.5	50	1.08
H-13	10	3000	47.7	50	100	0.9	50	1.14
H-105	10	50	15.6	2	10	1.1	4	0.967

3.2.3 Multi-class Problems

For multi-class problems, we usually use one-against-all or pairwise classification. In one-against-all classification, we separate one class from the others. Therefore, there are n two-classes and as the kernel parameter value we calculate the average of the n kernel parameters determined by the second-order method. In pairwise classification, we separate one class from another class. Thus, there are $n(n-1)/2$ two-classes and we calculate the average of the $n(n-1)/2$ kernel parameters determined by the second-order method.

4 Performance Evaluation

We compared the generalization ability of the proposed method with grid search for RBF kernels and two-stage line search for Mahalanobis kernels using multi-class data sets used in [9]. Two-stage line search for Mahalanobis kernels is done as follows: Setting $\delta = 1$ we determine the margin parameter by 5-fold cross-validation. Then we determine the value of δ by cross-validation [3].

In all studies, we normalized the input ranges into $[0, 1]$. The value of C was selected from among $\{1, 10, 50, 100, 500, 1000, 2000, 3000, 5000, 8000, 10000, 50000, 100000\}$, the value of γ by grid search from among $\{0.1, 0.5, 1, 5, 10, 15\}$, and the value of δ by line search from among $[0.1, 0.2, \ldots, 1.9, 2]$.

We used one-against-all SVMs. To resolve unclassifiable regions, we used fuzzy SVMs with minimum operators [9]. Table 1 lists the parameters determined by each method. In the table, g, s, and l denote grid search, line search with the second-order method, and two-stage line search. For RBF kernels, the values of γ for the proposed method are larger than those for the grid search but the values of the margin parameter are smaller. But for Mahalanobis kernels, margin parameters for two-stage line search and the proposed method are comparable. The same is true for the values of δ.

Table 2 lists the classification errors of the test data sets. The numeral in parentheses shows the classification error of the training data set if the classification error is not 0%. For each classification problem, the minimum classification

Table 2: Performance comparison of Mahalanobis and RBF Kernels.

Data	RBF(g)	RBF (s)	Diag (l)	Diag (s)
Iris	**5.33**	**5.33** (2.67)	**5.33** (1.33)	**5.33** (1.33)
Numeral	**0.61**	0.98 (0.25)	0.63 (0.12)	0.98 (0.25)
Thyroid	2.71 (0.50)	4.26 (0.05)	2.80 (0.74)	**2.61** (0.61)
Blood cell	7.16 (2.36)	**6.58** (2.52)	7.65 (3.58)	7.29 (2.13)
H-50	0.74	**0.59**	0.72	0.74
H-13	0.37	0.41 (0.04)	**0.22** (0.02)	0.41 (0.04)
H-105	**0**	**0**	**0**	**0**

error of the test data set is shown in boldface. Except for the thyroid data by the proposed method, the four methods show the comparable performance.

For all the cases, the second-order method converged within 10 iterations. The calculation time of γ for H-105 using a workstation (3.6GHz, 2GB memory, Linux operating system) was 3209 seconds but if we used pairwise classification it reduced to 188 seconds.

5 Conclusions

We discussed optimizing the kernel parameter by maximizing the distance between classes. For multi-class problems kernel parameters obtained for the two-class problems are averaged and set as a kernel parameter. Then the margin parameter is determined by cross-validation.

According to the computer experiments, except for one case generalization abilities of the proposed method are comparable with those by grid search and two-stage line search for RBF and the Mahalanobis kernels, respectively.

References

[1] R. Herbrich, *Learning Kernel Classifiers: Theory and Algorithms*, MIT Press, 2002.

[2] Y. Grandvalet and S. Canu. Adaptive scaling for feature selection in SVMs, In *Advances in Neural Information Processing Systems 15*, pp. 569–576. MIT Press, 2003.

[3] S. Abe, Training of support vector machines with Mahalanobis kernels, *Proc. ICANN 2005*, pp. 571–576, 2005.

[4] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, 1995.

[5] T. Joachims, Estimating the generalization performance of an SVM efficiently, *Proc. ICML-2000*, pp. 431–438, 2000.

[6] O. Chapelle and V. Vapnik, Model selection for support vector machines, *Advances in Neural Information Processing Systems 12*, pp. 230–236. MIT Press, 2000.

[7] L. Wang and K. L. Chan, Learning kernel parameters by using class separability measure, In *Kernel Machines Workshop, NIPS*, 2002.

[8] K.-P. Wu and S.-D. Wang, Choosing the kernel parameters of support vector machines according to the inter-cluster distance, *Proc. IJCNN 2006*, pp. 2184–2190, 2006.

[9] S. Abe, *Support Vector Machines for Pattern Classification*, Springer-Verlag, 2005.