Kernel Optimization Using a Generalized Eigenvalue Approach

Jayadeva, Sameena Shah, and Suresh Chandra

Indian Institute of Technology Delhi, Hauz Khas,
New Delhi -110016, India

Abstract. There is no single generic kernel that suits all estimation tasks. Kernels that are learnt from the data are known to yield better classification. The coefficients of the optimal kernel that maximizes the class separability in the empirical feature space had been previously obtained by a gradient-based procedure. In this paper, we show how these coefficients can be learnt from the data by simply solving a generalized eigenvalue problem. Our approach yields a significant reduction in classification errors on selected UCI benchmarks.

Keywords: Data dependent kernel, Fisher's coefficient, generalized eigenvalue, kernel optimization, Rayleigh quotient.

1 Introduction

SVMs aim to minimize the upper bound on the generalization error by maximizing the margin between the separating hyperplane and the data. If a linear separator in the input space does not suffice, a non-linear mapping is used to map the input space into a higher dimensional Euclidean or Hilbert feature space. This embedding induces a Riemannian metric in the input space, which is either enlarged or reduced in the feature space. Thus, by choosing an "appropriate" mapping $\phi(\cdot)$, the data points become linearly separable or mostly linearly separable in the high dimensional feature space, enabling easy application of structural risk minimization. Each kernel induces a different mapping or structure of the data in the embedding space, which may or may not be suitable from the classification perspective. To obtain a good quality embedding, that is more appropriate for the estimation task, the choice of the kernel should be "learned" from the structure of the data rather than chosen through some trial and error heuristics. An improper choice of kernel may lead to worse classification. In this paper, we extend the approach of [3], which tries to optimize the data dependent kernel by maximizing the class separability criterion in the empirical feature space. The transformation from the input space to an r-dimensional Euclidean space is given by $x \longrightarrow D^{-1/2}P^T(k(x,x_1),k(x,x_2),\ldots,k(x,x_m))^T$, where, $K_{m\times m} = P_{m\times r}D_{r\times r}P_{r\times m}$. Their approach has led to obtaining kernels that yield a significantly better classification performance compared to primary Gaussian or polynomial kernels for k-Nearest Neighbor (KNN), Kernel Fisher Discriminant (KFD), Kernel Minimum Squared Error machine (KMSE), and SVM on the UCI benchmark data sets [4]. However, only a slight improvement

S. Chaudhury et al. (Eds.): PReMI 2009, LNCS 5909, pp. 32–37, 2009.

in performance is observed on using the optimal kernels for SVMs. Our approach reduces to solving a single generalized eigenvalue problem. Our approach also has a number of other advantages, namely,

1. It avoids using an iterative algorithm to update alpha (coefficients of the optimal kernel) in each step, since now only a single generalized eigenvalue problem needs to be solved.
2. It avoids an ascent procedure that can not only potentially run into numerical problems when the Hessian matrices in question are not well conditioned, but also gets stuck in local maxima.
3. It also avoids tuning parameters such as learning rate and the number of iterations.

Section 2 formally defines our notion of optimality and develops the procedure to obtain the data dependent optimal kernel as a generalized eigenvalue solution. Section 3 gives the classification performance evaluation of our optimal kernel with a Gaussian kernel, and with the optimal kernel generated by the iterative update approach of [3] on UCI Benchmark data sets and section 4 concludes the paper.

2 Kernel Optimization

Maximizing the separation between the different classes is a natural step in the hope of obtaining better classification. To evaluate the measure of separability, we choose the Fisher Discriminant function [5]. It attempts to find a natural measure between examples induced by the generative model. In terms of the between class scatter matrix S_B and the within class scatter matrix S_W, the Fisher discriminant criterion $J(\cdot)$ can be written as (cf. [6]) $J(w) = \frac{w^T\ S_B\ w}{w^T\ S_W\ w}$. Substituting $\nabla J(w) = 0$ for maximizing $J(\cdot)$we get,

$$S_B w = J(w) S_W w. \tag{1}$$

Thus the extremum points w^* of the Rayleigh Quotient $J(w)$ are obtained as the eigenvectors of the corresponding generalized eigenvalue problem.

2.1 Data Dependent Optimal Kernel

Similar to Amari [2], we choose the data dependent kernel function, K, as a conformal transformation of the form $K(x, y) = q(x).q(y).k_0(x, y)$, where,

$$q(x) = \alpha_0 + \sum_{i=1}^{n_e} \alpha_i k_1(x, a_i) \tag{2}$$

In (2), k_1 denotes a primary Gaussian kernel of the form $k_1(x, a_i) = \exp(-\gamma\|x - a_i\|)^2$ and $\alpha = \{\alpha_0, \alpha_1, \ldots, \alpha_{n_e}\}$ is the vector of linear combination coefficients. The total number of empirical cores is denoted by n_e and a_i denotes the i-th empirical core. We choose a random set of one-third of the data points as the empirical cores. If the data is written as the first m_1 data points class C_1, followed by the rest m_2 data points of class C_2, then, the kernel matrix K_0 , corresponding to the primary kernel, can be written as

$$K_0 = \begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} \tag{3}$$

where, K_{11}, K_{12}, K_{21} and K_{22} are sub-matrices of order $m_1 \times m_1$, $m_1 \times m_2$, $m_2 \times m_1$, and $m_2 \times m_2$ respectively. The "between-class" kernel scatter matrix B_0, and the "within - class" kernel scatter matrix W_0, can then be written as

$$B_0 = \begin{pmatrix} \frac{1}{m}K_{11} & 0 \\ 0 & \frac{1}{m}K_{22} \end{pmatrix} - \begin{pmatrix} \frac{1}{m}K_{11} & \frac{1}{m}K_{12} \\ \frac{1}{m}K_{21} & \frac{1}{m}K_{22} \end{pmatrix}, \tag{4}$$

and,

$$W_0 = \begin{pmatrix} k_{11} & 0 & \dots & 0 \\ 0 & k_{22} & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & k_{mm} \end{pmatrix} - \begin{pmatrix} \frac{1}{m}K_{11} & 0 \\ 0 & \frac{1}{m}K_{22} \end{pmatrix}. \tag{5}$$

Using (4) and (5), the Fisher's coefficient is equivalent to $J = \frac{q^T B_0 q}{q^T W_0 q}$, where,

$$q = \begin{pmatrix} 1 & k_1(x_1, a_1) & \dots & k_1(x_1, a_n) \\ 1 & k_1(x_2, a_1) & \dots & k_1(x_2, a_n) \\ \vdots & \dots & \dots & \vdots \\ 1 & k_1(x_m, a_1) & \dots & k_1(x_m, a_n) \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = K_1 \alpha \quad . \tag{6}$$

Here, K_1 is a matrix of order $m \times (n_e + 1)$ and α is as before. Hence J can now be used as a measure of separability of the data in the empirical feature space that is easy to compute. Maximizing J, to obtain the maximum separation of data requires finding the optimal q. Optimizing the projection q is in turn equivalent to optimizing α with respect to the data (cf. (2)). Maximization of this separability measure, to obtain the optimal kernel, thus amounts to finding the optimal set of coefficients of the vector α. These coefficients would optimally enlarge or reduce the spaces around the empirical cores to reduce the error maximally. To obtain this optimal α, an iterative procedure to maximize Rayleigh quotient J was proposed in [3]. The algorithm based on the general gradient descent method updates in all iterations. While a decreasing learning rate ensures convergence, the algorithm still requires an iterative update solution and is susceptible to getting trapped in local maxima. In the sequel we solve (11) as a generalized eigenvalue problem and obtain the optimal data dependent kernel.

2.2 The Generalized Eigenvalue Solution

On substituting (6), J can be written as

$$J = \frac{q^T B_0 q}{q^T W_0 q} = \frac{\alpha^T K_1^T B_0 K_1 \alpha}{\alpha^T K_1^T W_0 K_1 \alpha} \quad . \tag{7}$$

As a generalized eigenvalue problem, (6) can be written as $B_0 q = \lambda W_0 q$, where λ are the set of eigenvalues. This is solved by Lanczos method. For the optimal kernel, the eigenvector α, corresponding to the maximum eigenvalue is used. It also avoids the need for tuning parameters like number of iterations and learning rate. Further, it also avoids the possibility of getting stuck in local minima which is a generic problem associated with gradient descent based methods. The regularized measure of separability is then given by

$$J = \frac{\alpha^T K_1^T B_0 K_1 \alpha}{\alpha^T K_1^T (W_0 + DI) K_1 \alpha} \tag{8}$$

where D is a small constant and I is the identity matrix of the appropriate dimension.

3 Experimental Results

In this section we apply the algorithm to datasets of different level of complexity in terms of number of instances (m) and number of attributes (d). Ionosphere ($m = 351$ and d=34), Monks ($m = 432$ and d=6), Heart($m = 297$ and d=13), Wisconsin ($m = 569$ and d=30). The data points with missing attributes, if any, have been removed. In the Ionosphere dataset, the column with zero variance has been removed. Each dataset has been normalized to lie between zero mean and unit variance. For all our experiments we have used (8), where D have been chosen through cross validation.

For evaluating our approach's performance, in each run, we randomly divide the data set into three disjoint sets. The data points in the first set are used for training while those in the second set are used for testing. The data points in the remaining third set are used as the empirical cores. Tables 1 through 4 compare the performance in terms of the classification error incurred (both training and testing separately) by using a primary Gaussian kernel, the kernel optimized using the approach proposed by Xiong et al. and the kernel optimized using the suggested approach illustrated in this paper. γ_0 corresponds to the parameter for the primary kernel while γ_1 is the parameter for the data dependent kernel. For the sake of comparison we choose the same set of values for γ_0 , γ_1 and C (for SVMs) as those in [3]. Each entry in the table, for a particular choice of kernel and a particular choice of value for γ_0 and γ_1 has been computed as the average of 20 random runs. The parameter for C has been chosen to be 1000 for all runs.

For most cases, the kernel optimized via the generalized eigenvalue approach gives significantly better classification errors than the corresponding other two methods. The comparison between the generalized eigenvalue approach based optimal kernel and using an arbitrary kernel clearly shows the huge difference in performance. This is because in [3], only an approximation to the optimal value of α is obtained for the equation $\frac{J_1}{J_2}\alpha = N_0^{-1} M_0 \alpha$ using an update of the form $\alpha^{n+1} = \alpha^n + \eta \left(\frac{1}{J_2} M_0 - \frac{J_1}{J_2^2} N_0 \right) \alpha^n$. On the other hand we mapped the problem to a generalized eigenvector problem. The generalized eigenvector problem is a quasiconvex optimization problem, since the constraint is convex and

Table 1. Training and testing error rates (%) using SVM classifier for the different kernels on Ionosphere data

(γ, γ_0)	Gaussian Kernel		Iterative update		Generalized eigenvalue	
	Training Error	Testing Error	Training Error	Testing Error	Training Error	Testing Error
(0.01, 0.0005)	1.4103	16.5812	0.3846	17.6923	0.2564	16.4530
(0.05, 0.0001)	3.8034	13.5470	5.3419	12.4786	0.5556	4.7009
(0.01, 0.0001)	4.5299	12.2650	3.9744	11.9658	2.7778	8.3761

Table 2. Training and testing error rates (%) using SVM classifier for the different kernels on the Monks-I dataset

(γ, γ_0)	Gaussian Kernel		Iterative update		Generalized eigenvalue	
	Training Error	Testing Error	Training Error	Testing Error	Training Error	Testing Error
(0.01, 0.0005)	32.7957	32.9655	33.5314	33.1353	9.2530	14.7142
(0.05, 0.0001)	31.5054	31.9624	33.4140	33.2258	7.7957	13.7366
(0.01, 0.0001)	31.2097	31.6129	33.4946	33.0645	6.2634	13.9785

Table 3. Training and testing error rates (%) using SVM classifier for the different kernels on the CLEVELAND HEART dataset

(γ, γ_0)	Gaussian Kernel		Iterative update		Generalized eigenvalue	
	Training Error	Testing Error	Training Error	Testing Error	Training Error	Testing Error
(0.01, 0.0005)	9.0377	17.9160	29.4524	30.8347	8.5061	21.1590
(0.05, 0.0001)	9.8990	16.5657	9.7475	16.4646	1.3131	28.9899
(0.01, 0.0001)	10.0000	16.3131	17.3737	22.0202	7.3232	21.6667
(0.05, 0.001)	7.3737	18.2323	12.2727	21.3131	1.1111	29.1919

Table 4. Training and testing error rates (%) using SVM classifier for the different kernels on the WISCONSIN BREAST CANCER dataset

(γ, γ_0)	Gaussian Kernel		Iterative update		Generalized eigenvalue	
	Training Error	Testing Error	Training Error	Testing Error	Training Error	Testing Error
(0.01, 0.0005)	0.6925	2.9640	9.4460	12.1330	0.8587	2.9640
(0.05, 0.0001)	1.1579	3.5789	1.8684	4.1053	0.6316	3.7632
(0.01, 0.0001)	1.4474	3.0526	10.2105	13.2632	1.1053	2.8684

the objective function is quasiconvex [7]. The Generalized eigenvalue problem is tractable and can be solved in polynomial time [7]. In practice they can be solved (a feasible point with an objective value that exceeds the global minimum by less than the prescribed accuracy can be obtained) very efficiently. Matlab uses semi-definite programming to obtain the global optimal of the GEVP problem [8]. Hence we could obtain the optimal solution, which is not possible in the other case. The approach in [3] was adopted to remove the problem of singularity. We corrected this problem using regularization. The generalized eigenvalue problem thus allows us to obtain the optimal solution without the need of any tuning of parameters or stopping conditions etc and without compromising the quality of the optimum. Overall one can conclude that if the empirical cores have the same distribution as the training and testing set, then classification accuracy can be improved significantly, sometimes by an order of magnitude, by the use of such data dependent kernels optimized by the generalized eigenvalue approach.

4 Conclusions and Discussion

The low classification error rates obtained by using the generalized eigenvalue approach to obtain the data dependent kernel implies that this approach yields a kernel that is better suited for the classification task because it can better adapt to the structure of the data and leads to a significant improvement in SVM's performance. This is because the generalized eigenvalue approach yields an exact solution that avoids getting stuck in a local maximum, and consequently leads us to obtain better kernels that yield a lower classification error. Moreover, we did not face any numerical problems in computing the generalized eigenvalue solution for the regularized Fisher's Discriminant function for various data sets. The performance of the proposed approach was compared with that of [3]. The experimental results convincingly demonstrated the effectiveness of our approach. Thus, it is evident that using the exact solution gives much better solutions that not only fit well, but also generalize well. In future, it would be interesting to explore the effect that the number of empirical cores would have on the classifier performance. We also plan to test our approach on other classifiers.

References

1. Aizerman, M.A., Braverman, E.M., Rozonoer, L.I.: Theoretical foundations of the potential function method in pattern recognition learning. Automation and Remote Control 25, 821–837 (1964)
2. Amari, S., Wu, S.: Improving support vector machine classifiers by modifying kernel functions. Neural Networks 12(6), 783–789 (1999)
3. Xiong, H., Swamy, M.N.S., Ahmad, M.O.: Optimizing the Kernel in the Empirical Feature Space. IEEE Trans. Neural Networks 16(2), 460–474 (2005)
4. Murphy, P.M., Aha, D.W.: UCI machine learning repository (1992), http://www.ics.uci.edu/~mlearn/MLRepository.html
5. Jaakkola, T.S., Haussler, D.: Exploiting generating models in discriminative classifiers. In: Proc. of Tenth Conference on Advances in Neural Information Processing Systems, Denver (December 1998)
6. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn., pp. 117–124. John Wiley and Sons, Inc., Chichester (2001)
7. Boyd, S., El Ghaoui, L., Feron, E., Balakrishnan, V.: Linear Matrix Inequalities in System and Control Theory. Studies in Applied Mathematics, vol. 15. SIAM, Philadelphia (1994)
8. GEVP, Robust Control Toolbox. The Mathworks, http://www.mathworks.com/access/helpdesk/help/toolbox/robust/gevp.html