International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307, Volume-1, Issue-5, November 2011

Performance Evaluation of Kernels in Multiclass Support Vector Machines

R. Sangeetha, B. Kalpana

***Abstract* — In recent years, Kernel based learning algorithm has been receiving increasing attention in the research domain. Kernel based learning algorithms are related internally with the kernel functions as a key factor. Support Vector Machines are gaining popularity because of their promising performance in classification and prediction. The success of SVM lies in suitable kernel design and selection of its parameters. SVM is theoretically well-defined and exhibits good generalization result for many real world problems. SVM is extended from binary classification to multiclass classification since many real-life datasets involve multiclass data. In this paper, we propose an optimal kernel for one-versus-one (OAO) and one-versus-all (OAA) multiclass support vector machines. The performance of the OAO and OAA are evaluated using the metrics like accuracy, support vectors, support vector percentage, classification error, and speed. The empirical results demonstrate the ability to use more generalized kernel functions and it goes to prove that the polynomial kernel's performance is consistently better than other kernels in SVM for these datasets.**

***Index Terms*— Support Vector Machine, Multiclass Classification, Kernel function, One versus One, One versus All.**

I. INTRODUCTION

Improving efficacy of classifiers have been an extensive research area in machine learning over the past two decades, which led to state-of-the-art classifiers like support vector machines ,neural networks and many more. Support Vector Machine is a robust classification tool, effectively overcomes many traditional classification problems like local optimum and curse of dimensionality. Three major issues of SVM are ***Kernel Mapping***, ***Quadratic Optimization*** and ***Maximum Margin Classifiers***. This paper focuses in the first issue. Multiclass SVM decomposes multiclass labels into several two class labels and it trains a svm classifier to solve the problems and then reconstruct the solution of the multiclass problem from outputs of the classifiers [9], such as OAO-SVM and OAA-SVM.

The paper is organized as follows. Section 2 and 3 describe SVM and Multiclass SVM. Section 4 explains the kernels and its parameters. Section 5 elucidates the experimental results. Lastly, Section 6 concludes with future work.

Sangeetha.R, Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India. (e-mail:sangeethadj@gmail.com).

Dr.B.Kalpan, Department of Computer Science, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India.(e-mail:kalpanabsekar@gmail.com)

II. SUPPORT VECTOR MACHINES [12, 13]

Support Vector Machine has been a new and important tool for classification and regression. In dealing with large data classification, traditional optimization algorithms such as Newton Method or Quasi-Newton Method cannot work any more due to the memory problem. SVMs belong to a family of generalized linear classification. A special property of SVM [3-6] is it simultaneously minimizes the empirical classification error and maximizes the geometric margin. So SVM is called as Maximum Margin Classifiers. SVM maps input vector to a higher dimensional space where a maximal separating hyperplane is constructed. Two parallel hyperplanes are constructed on each side of the hyperplane that separates the data. The separating hyperplane is a hyperplane that maximize the distance between the two parallel hyperplanes. An assumption is made that the larger the margin or distance between these parallel hyperplanes then better the generalization error of the classifier.

Consider the problem of separating the set of training vectors belonging to binary classes or dichotomization (x_i, y_i), $i = 1,\ldots. l$, $x_i \in R^n$, $y_i \in \{+1, -1\}$, where the R^n is the input space, x_i is the feature vector and y_i is the class label of x_i. The separating hyperplanes are linear discriminating functions as follows,

$$f(x) = w^T x + b \ , \qquad (1)$$

where w is a weight vector and b is called the bias value. One of the hyperplanes that maximizes the margin $\frac{2}{\|w\|^2}$ is named as the optimal separating. The optimal separating hyperplane [4] can be found by solving the following optimization problem:

$$\min_{\omega, b, \xi} \frac{1}{2}|\omega| + C \sum_{i=1}^{l} \xi_i \ , \qquad (2)$$

subject to

$$y_i(\omega^T x_i) + b \geq 1 - \xi_i, \xi_i \geq 0 \qquad (3)$$

or its dual problem

$$\min_{\alpha} \quad \frac{1}{2}\alpha^T Q \alpha - e^T \alpha \ , \qquad (4)$$

subject to

$$0 \leq \alpha_i \leq C, i = 1,\ldots,l, y^T \alpha = 0, \qquad (5)$$

where e is the vector of all ones, C is the penalty of error which is positive; Q_{ij} is $y_i y_j \langle x_i , x_j \rangle$ and ξ_i is the relaxation parameter. Thus if we obtain α and b then we can classify the decision function as follows

$$f(x) = \sum_{i=1}^{l} \alpha_i y_i \langle x_i . x_j \rangle + b \qquad (6)$$

Most optimization problems involve terms that are unknown and are usually not directly obtainable from the training data and they are not easy to guess, e.g., ξ_i in above equation. Thus, it becomes convenient to formulate an equivalent optimization problem that has the same solution as the original one, but does not involve any other information than what is provided by the training samples. This involves the use of Karush-Kuhn-Tucker conditions. The former problem is then called the Primal problem, and the latter is called as Dual.

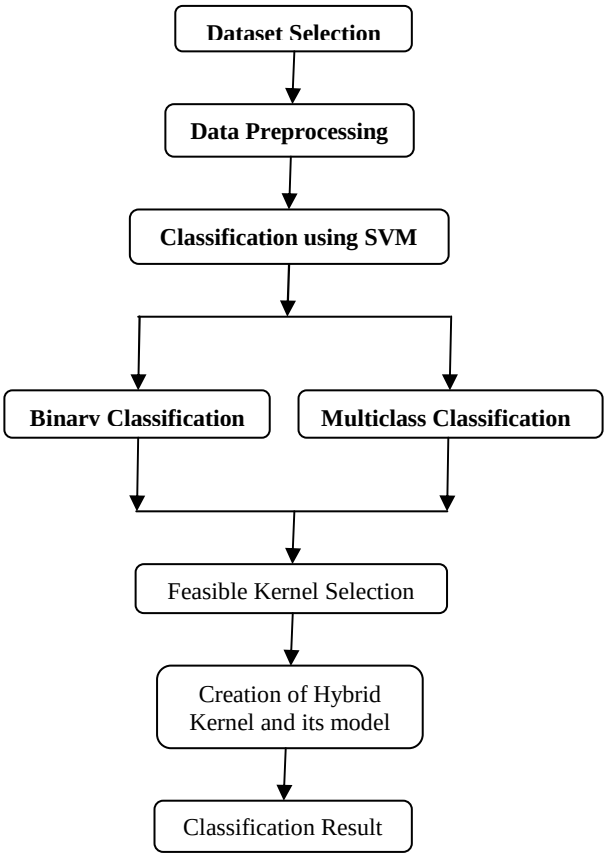

Fig.1 Flow of Proposed work.

III. MULTICLASS SUPPORT VECTOR MACHINES [15]

Support Vector Machines are based on variational-calculus which constrained to have structural risk minimization (SRM) principle and it uses convex optimization with unique optimum solution. In SVM, hyperplanes are derived to separate the class labels in feature space. One of the hyperplanes that maximizes the margin is an optimal separating hyperplane. Binary classification is explicated in [12, 13].

Figure 1 represents the flow of the proposed work. Multiclass SVM can be solved by combining the binary classification decision functions. Multiclass SVM is of two types namely, **One versus One** decomposition and ***One versus All*** decomposition. The OAA decomposition [10] transforms the multiclass problem into a series of *c* binary subtasks that can be trained by the binary SVM. Let the training set $T_{XY}^{y} = \{(x_1, y_1^{'}),....,(x_l, y_{l}^{'})\}$ contain the modified hidden states defined as

$$y_i^{'} = \begin{cases} 1 & for \quad y = y_i , \\ 2 & for \quad y \neq y_i \end{cases} \qquad (7)$$

The discriminant functions

$$f_y(x) = \langle \alpha_y \bullet K_s(x) \rangle + b_y , \quad y \in Y , \qquad (8)$$

are trained by the binary SVM solver from the set $T_{XY}^{y}, y \in Y$

The OAO decomposition [10] transforms the multi-class problem into a series of $g = c(c-1)/2$ binary subtasks that can be trained by the binary SVM. Let the training set $T_{XY}^{y} = \{(x_1^{'}, y_1^{'}),....,(x_{l_j}^{'}, y_{l_j}^{'})\}$ contain the training vectors $\boldsymbol{x}_{i \in} I^{j}=\{i: y_i = y^1 \text{ V } y_i = y^2\}$ and the modified the hidden states defined as

$$y_i^{'} = \begin{cases} 1 & for \quad y_j^1 = y_i , \\ 2 & for \quad y_j^2 \neq y_i , \end{cases} \quad i \in I^j \qquad (9)$$

The training set $T_{XY}^{j}, j = 1,2,... g$ is constructed for all $g=c(c-1)/2$ combinations of classes $y_j^1 \in Y \ \& \ y_j^2 \in Y \setminus \{y_j^1\}$ The binary SVM rules $q_j , j = 1, \ldots , g$ are trained on the data T_{XY}^{j} .

IV. KERNELS IN MULTICLASS SUPPORT VECTOR MACHINES

Kernel functions establish the characteristics of SVM model and level of non linearity. A necessary and sufficient condition for a simple inner product kernel to be valid is that it must satisfy Mercer's theorem [11]. In general, kernels are of two types namely *Local* and *Global* kernels. Data that are close to each other in local kernels influence on the kernel points and data that are far away from each other in global kernels influence on the kernel points. Commonly used kernels like polynomial, RBF, linear are used in this paper. Few other kernels are shown in Table1.

In existing statistical learning theory, when kernels are positive definite, there is one approach to obtain the mapping from original data set to feature space i.e. the kernels are demanded to satisfy Mercer's condition [16] and as a result they can be seen as dot product in some Hilbert space. Mercer's conditions seriously confine the wider application of SVM. Almost all the current review on kernel methods in machine learning focuses on kernels which are positive definite.

A. *Theorem. (Mercer's)*

Suppose that $K : \chi \times X \rightarrow R$ is symmetric and satisfies $\sup_{x,y} K(x, y) < \infty$,and define

$$T_K f(x) = \int_x K(x, y) f(y) dy \quad (10)$$

suppose that $T_K : L^2(\chi) \rightarrow L^2(X)$ is positive semi-definite; thus,

$$\int_\chi \int_\chi K(x, y) f(x) f(y) dxdy \geq 0 \quad (11)$$

for any, $f \in L^2(\chi)$. Let λ_i, ψi be the Eigen functions and Eigen vectors of $T_{K,}$ with

$$\int_\chi K(x, y)\psi_i(y) dy = \lambda_i \psi_i(x) \quad (12)$$

Then

1. $\sum_i \lambda_i < \infty$
2. $\sup_x \psi_i(x) < \infty$
3. $K(x, y) = \sum_{i=1}^{\infty} \lambda_i \psi_i(x)\psi_i(y)$

where the convergence is uniform in x, y.

Such a kernel defines a Mercer Kernel according to Mercer theorem given in [16]. This gives the mapping in to feature space as

$$x \mapsto \phi(x) = (\sqrt{\lambda_1}\psi_1(x), \sqrt{\lambda_2}\psi_2(y), ...)^T \quad (13)$$

B. *Reproducing Kernel Hilbert Spaces [16]*

Let us consider an inner product $\langle u, v \rangle$ as

1. A usual dot product: $\langle u, v \rangle = v'w = \sum_i v_i w_i$
2. A kernel product: $\langle u, v \rangle = k(v, w) = \psi(v)'\psi(w)$ where $\psi(u)$ may have infinite dimension.

However, an inner product $\langle ., . \rangle$ must satisfy the following conditions

1. Symmetry $\langle u, v \rangle = \langle v, u \rangle \forall u, v \in \chi$
2. Bilinearity $\langle \alpha u + \beta v, w \rangle = \alpha \langle u, w \rangle + \beta \langle v, w \rangle$ $\forall u, v, w \in \chi, \forall \alpha, \beta \in R$
3. Positive definiteness $\langle u, u \rangle \geq 0, \forall u \in \chi$ $\langle u, u \rangle = 0 \Leftrightarrow u = 0$

Definition 1

A Hilbert Space is an inner product space that is complete and separable with respect to the norm defined by the inner product.

Definition 2

K (,.) is a reproducing kernel Hilbert spaces H if $\forall$ f$\in$ H, $f(x) = \langle k(x, .), f(.) \rangle$ .A Reproducing Kernel Hilbert Space (RKHS) is a Hilbert space H with a reproducing kernel whose span is dense in H. We could equivalently define an RKHS as a Hilbert space of function with all evaluation functionals bounded and linear.

From the above definition and theorem, kernel function K must be continuous, symmetric, and have a positive definite gram matrix. Such a K means that there exists a mapping to a reproducing kernel Hilbert space such that the dot product there gives the same value as the function K. If a kernel does not satisfy Mercer's condition, then the corresponding Quadratic Problem has no solution. Hence, if any new kernel is proposed it should be checked with mercer kernel.

Table 1. Types of Kernels

Kernels	Function
Laplacian	$K(x, y) = \exp\left(-\frac{\|x-y\|}{\sigma}\right)$
Rational Quadratic	$K(x, y) = 1 - \frac{\|x-y\|^2}{\|x-y\|^2 + c}$
Multiquadratic	$k(x, y) = \sqrt{\|x-y\|^2 + c}$
Log	$K(x, y) = -\log\|x-y\|^d + 1)$
Bessel	$K(x, y) = \frac{J_{V+1}(\sigma\|x-y\|)}{\|x-y\|^{-n(v+1)}}$
Cauchy	$K(x, y) = \frac{1}{1 + \frac{\|x-y\|^d}{d}}$
Wavelet	$K(x, y) = \prod_{m=1}^{N} h\left(\frac{x_i - c}{a}\right) h\left(\frac{y_i - c}{a}\right)$

Table 2. Data Sets Used

Datasets	Size	Features	Class
Pentagon	99	2	5
Iris	150	4	3
Wine	270	13	3

V. RESULTS AND DISCUSSIONS

In this section, ***OAO*** and ***OAA*** SVM's kernel functions are evaluated using the metrics like accuracy, support vectors, support vector percentage, training error, classification error and time taken .For experimentation, two benchmark datasets (Iris, Wine) are taken from the UCI machine learning repository and one synthetic dataset from [10].Brief sketch of the datasets is given in table 2. In multiclass SVM, the optimal regularization parameter *C* and the kernel parameters are estimated by repeating classifications.

Linear kernel $K(x_i, x_j) = 1 + x_i^T x_j$ is a simple kernel function based on the penalty parameter ***C***, since parameter *C* controls the trade-off between frequency of error c and complexity of decision rule [7]. Also, it reduces the support vectors, training error and classification error by incrementing the parameter C.But it is not suitable for large datasets.

Polynomial kernel $K(x_i, x_j) = (1 + x_i^T x_j)^p$ also known as ***global kernel***, is non-stochastic kernel estimate with two parameters i.e. ***C*** and polynomial degree ***p***. Each data from the set $\mathbf{x_i}$ has an influence on the kernel point of the test value $\mathbf{x_j}$, irrespective of its the actual distance from $\mathbf{x_j}$ [14], It gives good classification accuracy with minimum number of support vectors and low classification error.

Radial basis function $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$ also known as ***local kernel***, is equivalent to transforming the data into an infinite dimensional Hilbert space .Thus, it can easily solve the non-linear classification problem. It has an effect on the data points in the neighborhood of the test value [14]. RBF gives similar result as polynomial with minimum training error but for some cases the number of support vector and classification error increases.

Exponential radial basis function $K(x_i, x_j) = \exp(-\frac{\|x_i - x_j\|}{2\sigma^2})$ gives piecewise linear solution.

Gaussian radial basis function $K(x_i, x_j) = \exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2})$ deals with data that has conditional probability distribution approaching gaussian function. ***RBF kernels*** perform better than the linear and polynomial kernel. However, it is difficult to find an optimum parameters σ and equivalent ***C*** that gives better result for a given problem.

Sigmoid kernel $K(x_i, x_j) = \tanh(k x_i^T x_j - \delta)$ is not efficient as other kernel function, because it lacks the necessary condition of a valid kernel. Parameters κ and δ must be chosen properly to obtain high classification accuracy.

The performance metrics of several kernels are compared to find an optimal and efficient kernel and it is carried out using **MATLAB** and **C++.** The tables 3.1, 3.2, 3.3 show training error, classification error and time taken for different kernels in OAO and OAA SVM on three datasets.

And, they are graphically depicted in figures 2, 3, 4 for OAO and figures 5, 6, 7 for OAA using kernel parameters in X axis and range of values in Y axis. Similarly support vectors, support vector percentage, accuracy are illustrated in tables 4.1, 4.2, 4.3. Also, they are visually portrayed in figures 8, 9, 10 for OAO and figures 11, 12, 13 for OAA.

In table 3.1 (i) *Exponential RBF* kernel's training error, classification error rate and time are lesser than the other kernels for OAO SVM, (ii) *Polynomial* and *Exponential RBF* Kernels training time, error rate and time are lesser than the other kernels for OAA SVM. In table 3.2, *Polynomial*, *ERBF* and *RBF kernels* training error, classification error and time are better compared to other kernels for OAO and OAA. In table 3.3,*Polynomial*, *ERBF* and *RBF* kernels training error, classification error and time are better compared to other kernels for OAO and OAA.Similarly, from tables [4.1- 4.3] *Polynomial kernel* and *RBF kernels* give better result. In kernel function, number of support vector increases then the classification accuracy diminishes. After analyzing all the features of the kernel function, appropriate and optimal kernels for our datasets are polynomial kernel and RBF kernels. They have minimum number of support vectors, minimum value as classification error and good classification accuracy which is shown in Figure 2-13.

VI. CONCLUSION

Classification time and Computational complexity for the multiclass SVM classifier depend on the number of support vectors required. In SVM classification, the required memory to store the support vectors is directly proportional to the number of support vectors. Hence, support vectors must be reduced to speed up the classification and to minimize the computational and hardware resources required for classification. Here, performance metrics of different kernels in multiclass SVM on three datasets are compared. As a result, the efficient kernel for multiclass SVM classifier is polynomial kernel for these datasets. Hybrid kernels can be created using systematic methodology and optimization technique. Therefore, the best method to combine the optimal feasible kernels would be our research work in future.

APPENDIX

Table 3.1 Training and Test Error Rate for Iris Dataset

Kernels	Parameter	One versus One			Parameter	One versus All		
		TE	CE	Time(s)		TE	CE	Time(s)
Linear	C=10	0.0167	0.5	0.03	C=10	0.0583	0.3333	0.14
	C=100	0.0	0.6333	0.01	C=10000	0.0167	0.3000	28
Polynomial	C=1, p = 1.5	0.0167	0.3333	0.04	C=10,p=2	0.08	0.1	0.34
	C=1,p=2.5	0.1416	0.4333	0.125	C=100,p=2	0.0	0.2	0.09
RBF	C=1, γ = 0.5	0.058	0.2667	0.03	C=10, γ =1.5	0.033	0.0667	0.04
	C=1, γ =1.5	0.025	0.5333	0.05	C=10, γ =1	0.025	0.1333	0.04
ERBF	C=1,σ=1.5	0.0167	0.0333	0.031	C=1, σ =0.5	0.008	0.0667	0.015
	C=10, σ =2.5	0.0167	0.2667	0.03	C=100, σ =2	0.1667	0.1	0.06
GRBF	C=10, σ =2	0.025	0.1333	0.03	C=10, σ =0.05	0.008	0.2333	0.12
	C=10, σ =1.5	0.0167	0.4	0.03	C=100, σ =0.05	0.008	0.2	0.28
Sigmoid	C=1, k=1,δ=2	0.0583	0.2667	0.063	C=1000, k=1, δ =3	0.0	0.2	0.218
	C=1000, k =5, δ =2	0.3083	0.0333	0.016	C=1000,k=2, δ =5	0.0	0.2667	0.313

Table 3.2 Training and Test Error Rate for Pentagon Dataset

Kernels	Parameter	One versus One			Parameter	One versus All		
		TE	CE	Time (s)		TE	CE	Time (s)
Linear	C=10	0.0	0.25	0.015	C=10	0.013	0.05	0.109
	C=100	0.0	0.25	0.03	C=1000	0.0	0.05	0.078
Polynomial	C=1000,p=3	0.0	0.25	0.0	C=100,p=1.5	0.0	0.1	0.1
	C=1000,p=6	0.0	0.25	0.031	C=1000,p=1.5	0.0	0.1	0.171
RBF	C=10, γ = 0.005	0.0	0.4	0.0	C=100, γ =0 .5	0.367	0.3	0.156
	C=100, γ =0.5	0.0	0.3	0.12	C=100, γ =6	0.025	0.1	0.09
ERBF	C=100, σ =1.5	0.0	0.25	0.02	C=100, σ =0.5	0.0	0.05	0.046
	C=1000, σ =0.5	0.02	0.8	0.016	C=inf, σ =2	0.0	0.1	0.109
GRBF	C=100, σ =0.05	0.0	0.5	0.031	C=10, σ =0.5	0.101	0.45	0.031
	C=1000, σ =2	0.04	0.3	0.015	C=inf, σ =0.5	0.316	0.25	0.031
Sigmoid	C=10, k=1, δ =2	0.01	0.3	0.0	C=100,k=1,δ=1	0.0	0.1	0.046
	C=100, k=0.5, δ=1	0.03	0.25	0.031	C=inf, k=2, δ =0.5	0.0	0.1	0.187

Table 3.3 Training and Test Error Rate for Wine Dataset

Kernels	Parameter	One versus One			Parameter	One versus All		
		TE	CE	Time(s)		TE	CE	Time(s)
Linear	C=1	0.0625	0.9412	1.288	C=10	0.528	0.8542	2.676
	C=100	0.0694	0.9118	1.355	C=100	0.253	0.8574	2.897
Polynomial	C=10,p=2	0.1458	0.1471	0.687	C=10,p=2	0.319	0.0882	0.156
	C=100,p=3	0.2656	0.1471	0.153	C=100,p=2	0.319	0.0882	0.171
RBF	C=100, γ =0.0005	0.0486	0.4118	0.703	C=100, γ = 0.00005	0.09	0.6765	2.359
	C=100, γ =0.05	0.0069	0.0588	0.859	C=1000, γ=0.00005	0.09	0.6765	2.567
ERBF	C=100, σ =8	0.0277	0.5588	0.812	C=100, σ =6	0.006	0.705	1.987
	C=100, σ =2.5	0.0138	0.2647	0.328	C=100, σ =10	0.006	0.6765	2.555
GRBF	C=1000, σ =8	0.0277	0.1765	0.593	C=1000, σ =8	0.013	0.8529	2.234
	C=1000, σ =6	0.0208	0.0882	0.965	C=1000, σ =6	0.0	0.8876	1.187
Sigmoid	C=100, k=2, δ =4	0.9	0.5902	0.562	C=100, σ=2,δ=4	0.58	0.8532	1.234
	C=100, k=2,δ =2	0.91	0.5902	0.531	C=100 k=2,δ=2	0.59	0.8532	1.25

Table 4.1 Accuracy, Support Vector and Support Vector % for Iris Dataset

Kernels	Parameter	One versus One			Parameter	One versus All		
		SV	SV%	Accuracy%		SV	SV%	Accuracy %
Linear	C=10	16	13.33	50	C=10	70	58.33	66.67
	C=100	11	7.5	36.67	C=10000	63	52.5	70
Polynomial	C=1, p = 1.5	23	19.1	66.67	C=10,p=2	15	12.5	90
	C=1,p=2.5	16	13.33	56.67	C=100,p=2	10	8.3	80
RBF	C=1, γ = 0.5	40	33.33	73.33	C=10, γ =1.5	20	16.67	93.33
	C=1, γ =1.5	31	25.8	46.67	C=10, γ =1	23	19.1	86.67
ERBF	C=1,σ=1.5	47	39	96.67	C=1, σ =0.5	31	25.8	93.33
	C=10, σ =2.5	28	23.33	73.33	C=100, σ =2	17	14.16	90
GRBF	C=10, σ =2	31	25.8	86.67	C=10, σ =0.05	45	37.5	76.67
	C=10, σ =1.5	26	21.6	60	C=100, σ =0.05	44	36.67	80
Sigmoid	C=1, k=1, δ=2	46	38.3	73.33	C=1000, k=1, δ =3	12	10	83.33
	C=1000, k =5, δ =2	40	33.33	96.66	C=1000,k=2, δ =5	11	9.16	76.67

Table 4.2 Accuracy, Support Vector and Support Vector % for Pentagon Dataset

Kernels	Parameter	One versus One			Parameter	One versus All		
		SV	SV%	Accuracy%		SV	SV%	Accuracy%
Linear	C=10	25	31.65	75	C=10	50	63.3	95
	C=100	20	25.32	75	C=1000	40	50.63	95
Polynomial	C=1000,p=3	18	22.78	75	C=100,p=1.5	19	24.05	90
	C=1000,p=6	15	18.98	75	C=1000,p=1.5	17	21.51	90
RBF	C=10, γ = 0.005	15	18.98	60	C=100, γ =0 .5	43	54.43	70
	C=100, γ =0.5	20	25.32	70	C=100, γ =6	21	26.58	90
ERBF	C=100, σ =1.5	21	26.58	75	C=100, σ =0.5	30	37.97	95
	C=1000, σ =0.5	28	35.44	20	C=inf, σ =2	19	24.05	90
GRBF	C=100, σ =0.05	38	48.1	50	C=10, σ =0.5	32	40.5	55
	C=1000, σ =2	18	22.78	70	C=inf, σ =0.5	17	21.51	75
Sigmoid	C=10, k=1, δ =2	28	35.44	70	C=100,k=1,δ=1	20	25.32	90
	C=100, k=0.5, δ =1	20	25.32	75	C=inf, k=2, δ =0.5	19	24.05	90

Table 4.3 Accuracy, Support Vector and Support Vector % for Wine Dataset

Kernels	Parameter	One versus One			Parameter	One versus All		
		SV	SV%	Accuracy%		SV	SV%	Accuracy%
Linear	C=1	33	22.91	5.88	C=10	35	24.3	14.58
	C=100	31	21.52	8.882	C=100	39	27.08	14.26
Polynomial	C=10,p=2	44	30.55	85.29	C=10,p=2	8	5.55	91.18
	C=100,p=3	45	31.25	85.29	C=100,p=2	8	5.55	91.18
RBF	C=100, γ =0.0005	68	47.22	58.82	C=100, γ = 0.00005	65	45.13	32.35
	C=100, γ =0.05	75	52.08	94.12	C=1000, γ =0.00005	55	38.19	32.35
ERBF	C=100, σ =8	70	48.6	44.12	C=100, σ =6	78	54.16	29.45
	C=100, σ =2.5	85	59.02	73.53	C=100, σ =10	72	50	32.35
GRBF	C=1000, σ =8	80	55.55	82.35	C=1000, σ =8	90	62.5	14.71
	C=1000, σ =6	80	55.55	91.18	C=1000, σ =6	100	69.44	11.24
Sigmoid	C=100, k=2, δ =4	112	77.77	40.98	C=100, σ=2,δ=4	122	84.72	14.68
	C=100, k=2,δ =2	110	76.38	40.98	C=100 k=2,δ=2	122	84.72	14.68

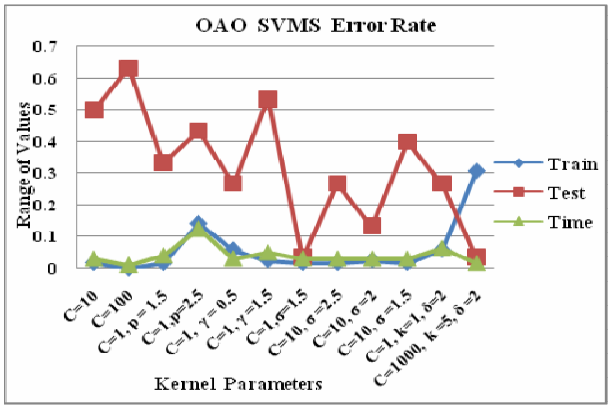

Fig.2 OA O- Error Rate for Iris dataset.

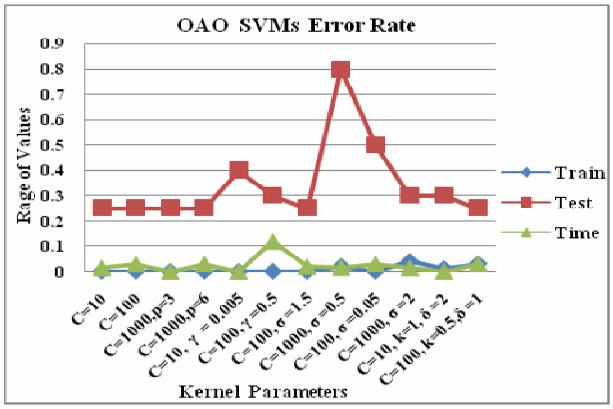

Fig.3 OAO- Error Rate for Pentagon dataset

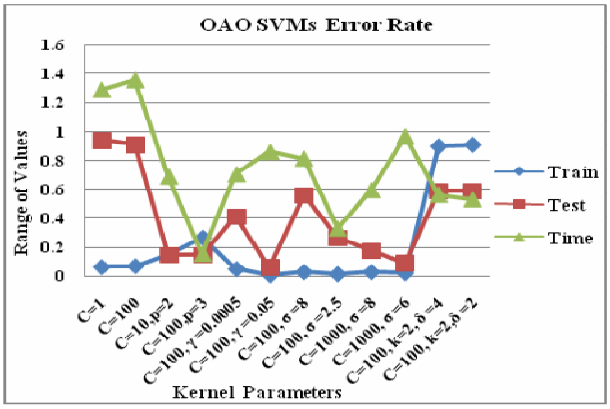

Fig.4 OAO- Error Rate for Wine dataset.

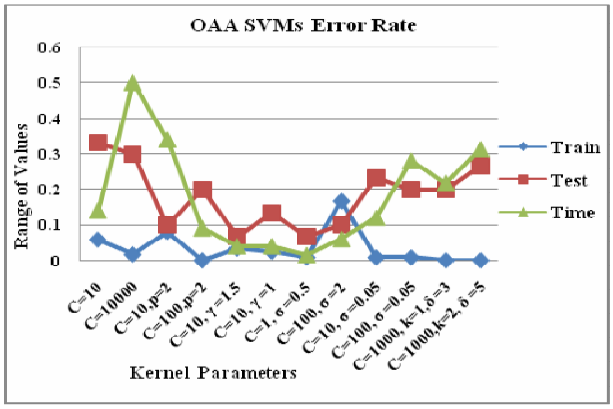

Fig.5 OAA- Error Rate for Iris dataset.

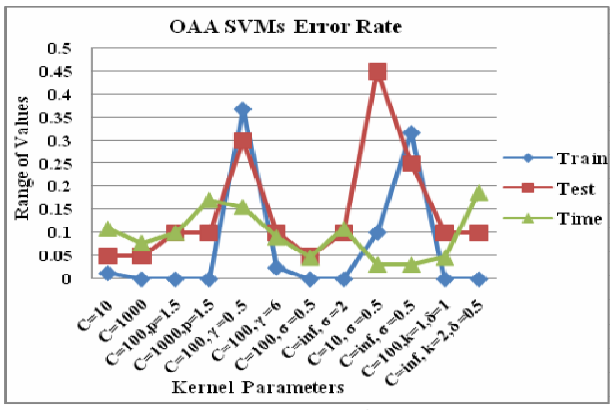

Fig.6 OAA- Error Rate for Pentagon dataset.

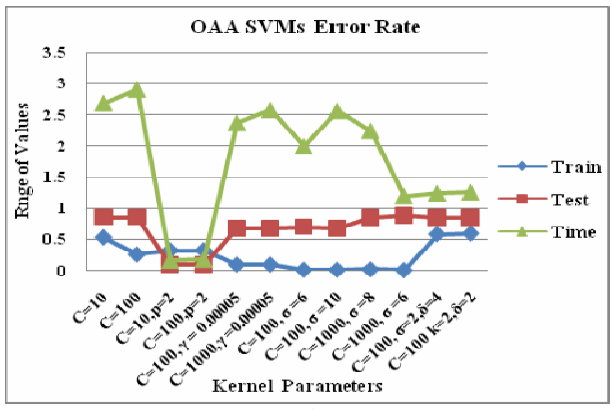

Fig.7 OAA- Error Rate for Wine dataset

Figure (2-4) represent OAO multiclass SVM Error Rate for Iris,Pentagon and Wine.Figure (5-7) represent OAA multiclass SVM Error Rate for Iris,Pentagon and Wine.

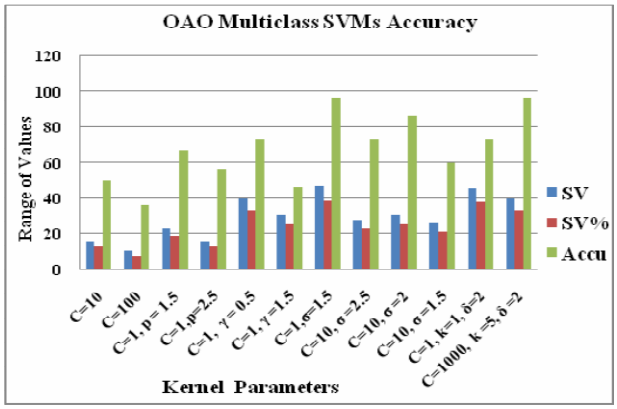

Fig.8 OAO- Accuracy for Iris dataset.

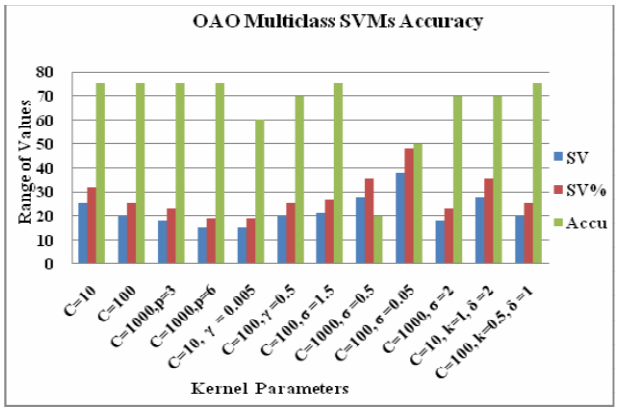

Fig.9 OAO- Accuracy for Pentagon dataset

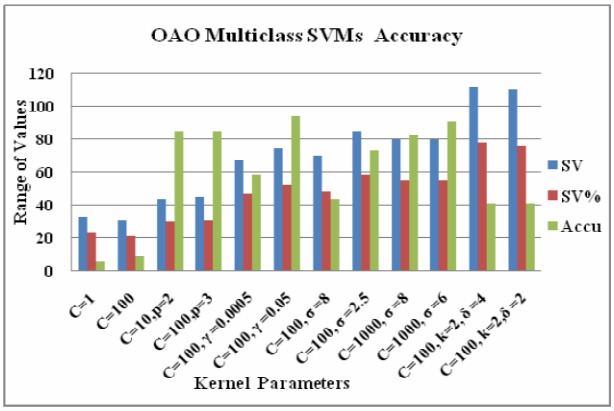

Fig.10 OAO- Accuracy for Wine dataset.

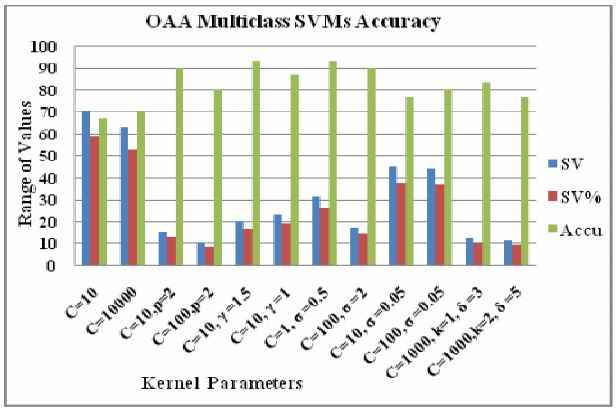

Fig. 11 OAA- Accuracy for Iris dataset.

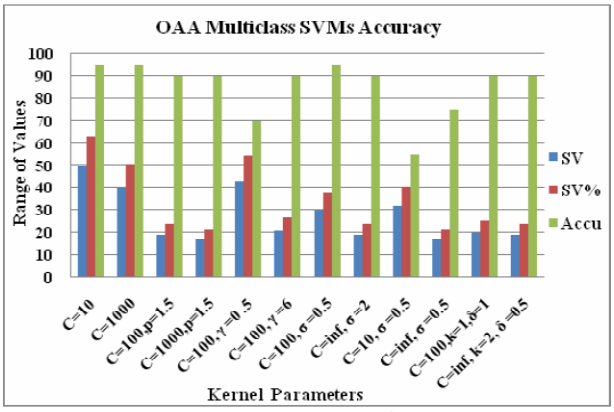

Fig.12 OAA- Accuracy for Pentagon dataset.

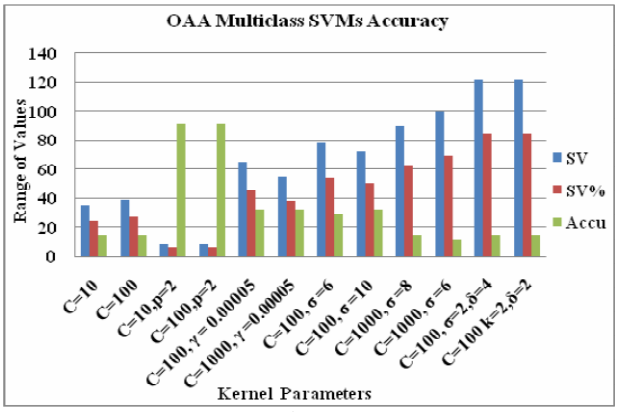

Fig.13 OAA- Accuracy for Wine dataset

Figure (8-10) represent OAO multiclass SVM Accuracy for Iris,Pentagon and Wine.Figure (11-13) represent OAA multiclass SVM Accuracy for Iris,Pentagon and Wine.

REFERENCES

[1] J. Han and M. Kamber, *Data Mining—Concepts and Technique*, 2nd ed. San Mateo, CA: Morgan Kaufmann, 2006.

[2] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining. Reading*, MA: Addison-Wesley, 2005.

[3] V. Vapnik, *An overview of statistical learning theory*, IEEE Trans. on Neural Networks, 1999.

[4] N. Cristianini and J. Shawe-Taylor, *Introduction to Support Vector Machines*, Cambridge University Press, 2000.

[5] B. Schölkopf and A. Smola, *Leaning with Kernels*, MIT Press, 2001.

[6] C. J. C. Burges, *A tutorial on support vector machines for pattern recognition*. Data Mining and Knowledge Discovery, 1998, pp 56–89.

[7] Corinna Cortes and V. Vapnik, *Support-Vector Networks*, Machine Learning, 1995.

[8] J. Manikandan , B.Venkataramani,*Study and evaluation of a multi-class SVM classifier using diminishing learning technique*, Neurocomputing, 2010.

[9] Anna Wang, Wenjing Yuan, Junfang Liu, Zhiguo Yu, Hua Li, *A novel pattern recognition algorithm: Combining ART network with SVM to reconstruct a multi-class classifier*, Computers and Mathematics with Applications, 2009.

[10] Vojtech Franc, Václav Hlavá, *Statistical Pattern Recognition Toolbox for Matlab*, 2009.

[11] Ralf Herbrich, *Learning kernel classifiers: theory and algorithms*, MIT Press, Cambridge, Mass, ISBN 026208306X, 2001.

[12] Sangeetha, R., Kalpana, B, *A comparative study and choice of an appropriate kernel for support vector machines*, In: Das, V.V., Vijaykumar, R. (eds.) ICT 2010. CCIS, vol. 101,pp. 549–553. Springer, Heidelberg (2010)

[13] Sangeetha, R., Kalpana, B, *Optimizing the Kernel Selection for Support Vector Machines using Performance Measures*, In: A2CWiC 2010, ISBN: 978-1-4503-0194-7,2010

[14] G.F. Smits , E.M Jordaan,*Improved SVM Regression using Mixtures of Kernels*, IJCNN '02. Proceedings of the International Joint Conference on Neural Networks, 2002.

[15] J Weston, C Watkins, *Multi class support vector machines*, Technical Report.

[16] XIA Guo-en *and* SHAO Pei-ji***"Factor Analysis Algorithm with Mercer Kernel",*** IEEE Second International Symposium on Intelligent Information Technology and Security Informatics, 2009.

Ms. R. Sangeetha completed her M.C.A from DJ Academy for Managerial Excellence, Coimbatore. Her area of interest is Data Mining. She is pursuing her Ph.D Full Time and working as a Research Assistant in Avinashilingam University, Coimbatore.

Dr. B. Kalpana received her Ph.D in Computer Science from Avinashilingam University, Coimbatore. She specializes in Data mining. She has around 20 years of teaching experience at the post graduate and under graduate level. She has published and presented papers in several refereed international journals and conferences. She is a member of the International Association of Engineers and Computer Scientists, Hongkong, Indian Association for Research in Computing Sciences (IARCS) and the Computer Society of India.