GLOBAL JOURNAL OF ENGINEERING, DESIGN & TECHNOLOGY
(Published By: Global Institute for Research & Education)
www.gifre.org

# Support Vector Approach for Classification and Regression problems in Misclassified Data produce sparse solution

M. Premalatha[1] & Dr. C. Vijayalakshmi[2]
[1]Research scholar, Department of Mathematics, Sathyabama University, Chennai
[2] School of Advanced Sciences (SAS), Department of Mathematics, VIT University, Chennai

## Abstract
Machine Learning is considered as a subfield of Artificial Intelligence and it is concerned with the development of techniques and methods which enable the computer to learn. Hence, the goal of learning was to output a hypothesis that performed the correct classification of the training data and early learning algorithms were designed to find such an accurate fit to the data. Since then SVMs have been successfully applied to real-world data analysis problems, often providing improved results compared with other techniques. It gives the clear idea for the advantage of the support vector approach is that sparse solutions to classification and regression problems in misclassified data. This fact facilitates the application of SVMs to problems that involve a large amount of data.

***Keywords***— *SVM Margin, Kernel Function and Mapping, Classification and Regression, One Class SVM.*

## 1. Introduction
The SVMs provide a compromise between the parametric and the nonparametric approaches: As in linear classifiers, SVMs estimate a linear decision function; mapping of the data into a higher-dimensional feature space may be needed. This mapping is characterized by the choice of a class of functions known as kernels. The foundations of Support Vector Machines (SVM) have been developed by Vapnik. Classification in SVM is an example of Supervised Learning. This information points to a desired response, validating the accuracy of the system, or be used to help the system learn to act correctly. A step in SVM classification involves identification as which are intimately connected to the known classes. This is called feature selection or feature extraction. Support Vector Machine (SVM) is a classification and regression prediction tool that uses machine learning theory to maximize predictive accuracy while automatically avoiding over-fit to the data. Each instance in the training set contains one target values and several attributes. The goal of SVM is to produce a model which predicts target value of data instances in the testing set which are given only the attributes.

## 2. SVM Margin
*Geometric margin (distance to the hyperplane):*
This is related to the norm of the weight vector, so that maximizing the margin corresponds to minimizing the norm.
*Numerical margin:*
The main reason to maximize this margin is because the hinge loss is a convex non increasing upper bound of the classification loss, so that making $y_i f(x_i)$ large will ensure that the loss is small and thus that we minimize the number of misclassification errors, but does not guarantee that the expected misclassification error will be minimized as well. For instance, if one minimizes the loss over linear combinations of kernels and if there exist a combination such that the total loss on the training set is zero, then this combination is not unique: we can multiply it by an arbitrary positive scale factor. It introduces a coupling between the numerical and the geometric margins: maximizing the geometric margin leads to regularization which prevents over fitting by complex functions, while maximizing the numerical margin leads to minimization of the empirical error.

### *2.1. SVM –Linear-Nonlinear-Regression*
The goal of learning was to output a hypothesis that performed the correct classification of the training data and early learning algorithms were designed to find such an accurate fit to the data. The ability of a hypothesis to correctly classify data not in the training set is known as its generalization.

$$margin = \arg\min d(x) = \arg\min \frac{|x.w + b|}{\sqrt{\sum_{i=1} w_i^2}}$$

The point that lies closest to the separating hyper plane, i.e. the Support Vectors, then the two planes $H_1$ and $H_2$ that these points lie on can be states as follows
$x_i$. w + b = +1 for $H_1$
$x_i$. w + b = -1 for $H_2$
Plus plane = $x_i$. w + b = +1
Minus plane = $x_i$. w + b = -1
Classify as: -1< $x_i$. w + b < 1
We define $d_1$ as being the distance from $H_1$ to the hyperplane and $d_2$ from $H_2$ to it. The hyper plane's equidistance from $H_1$ and $H_2$ means that $d_1 = d_2$ - a quantity known as the SVM's margin.

Objective function

$\min\|w\|$

*such that* $y_i(x_i \cdot w + b) - 1 \geq 0$

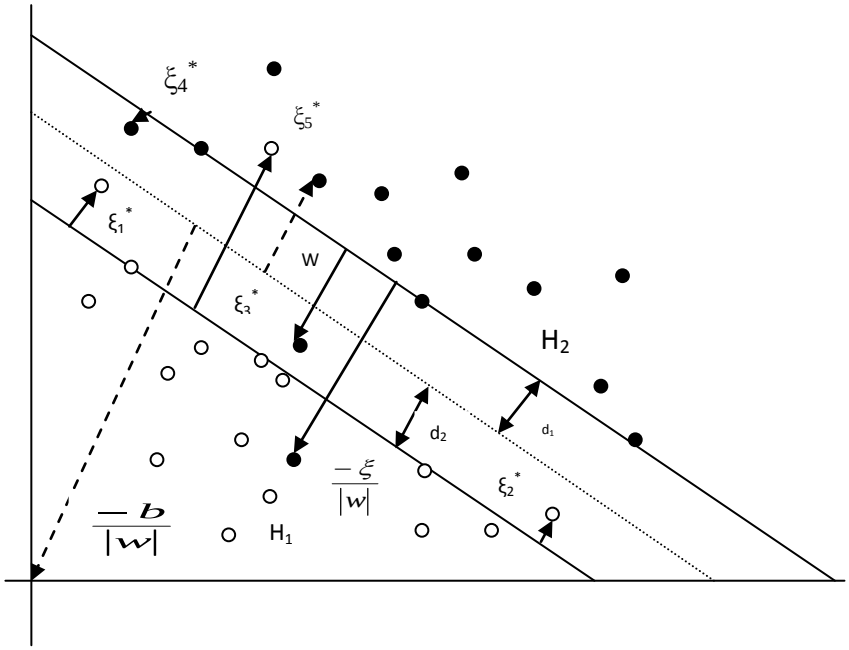



Fig: 1 SVM Linear Regression

### *2.2. Primal Problem*

Let $x_1=1$, $x_2=2$ $y = [-1, 1]^T$

Min ½($w^2$)

Subject to

1(w.1+b) ≥1; -1(w.2+b) ≥1

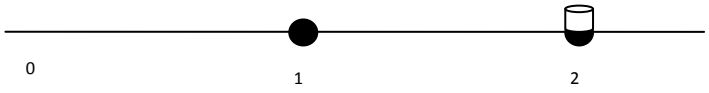



Fig: 2 Optimal Solution of Primal Problem

W ≥ -1; b ≥ 1 – w ≥ 2; b ≥ 2

Smallest possibility b = 2

Optimal solution is (w, b) = (-1, 2)

The separating hyper plane - x +2 =0 →x - 2 =0

### *2.3. Non Separable case*

Introduce slack variables $\xi_i \geq 0$

$y_i(x_i \cdot w + b) - 1 \geq 0; \quad y_i(x_i \cdot w + b) - 1 + \xi_i \geq 0$

**Objective Function: Soft Margin**

$$\min \phi(w, \xi) = \frac{1}{2}(w \cdot w) + C(\sum_{i=1}^{n} \xi_i$$

Constant C controls the tradeoff between margin and errors

**Similar Dual Optimization Problem:**

$$\max Q(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j}^{n} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j)$$

$$\sum_{i=1}^{n} y_i \alpha_i = 0; \qquad 0 \leq \alpha_i \leq C; \quad i = 1,2,....,n$$

**Extended to non linear Boundary:**

$$f(x) = g(x) + b = \sum_{i=1}^{n} w_i K(x_i.x) + b \leftarrow f(x) = (w.x) + b$$

The solution has to be a linear combination of the training instance.

**Multiclass SVMs:** *One-versus-all*

Train *n* binary classifiers, one for each class against all other classes. Predicted class is the class of the most confident classifier.

*One-versus-one*

Train *n (n-1)/2* classifiers, each discriminating between a pair of classes. Several strategies for selecting the final classification based on the output of the binary SVMs.

## 3. Relation Between Kernel function

*Kernel Trick Assumption*

**IDEA:** Map to higher dimension so the boundary is linear in that space but non linear in current space separation may be easier in higher dimensions.

Find ($w_1$, $w_2$, $w_3$,…,$w_n$, b)

$$\min \frac{1}{2}\|r\|_k^2 + C\left(\sum_{i=1}^{n}\xi_i\right)$$

Higher dimensional (may be infinite) feature space

$\phi(x) = (\phi_1(x), \phi 2(x)\ldots)$

$$\min z = \|w\|^2$$

$$y_i(w^T\phi(x_i) + b \geq 1$$

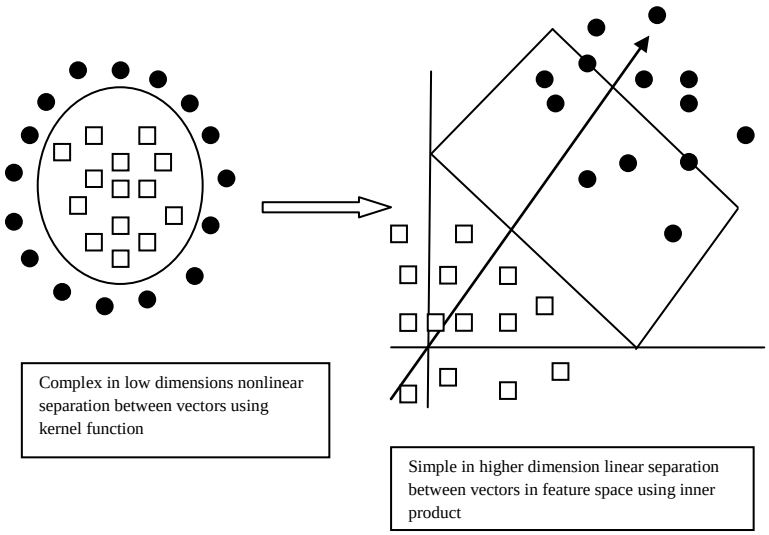



Fig: 3 Linear and Non linear Separation of low and high Dimension

Square in the norm of w has been introduced to make the problem quadratic. Given its convexity this optimization problem has no local minima. Consider this solution of the problem w* and b*. This solution determines the hyperplane in the feature space $D^*(x) = (w^*)^T\phi(x_i) + b = 0$ points $\phi(x_i)$ that satisfy the equalities $y_i (w^*)^T\phi(x_i)+b^* = 1$ are called support vectors, the SV can be automatically determined from the solution of the optimization problem. SV represents a small fraction of the sample, and the solution is said to be sparse. The hyperplane $D^*(x) = 0$ is completely determined by the subsample made up of the SV, the evaluation of the decision function $D^*(x)$ is computationally efficient, allowing the use of SVMs on large data sets. We have divided the data set into a training set (80% of the data points) and a test set (20% of the data points).

Table: 1 Test Error in various Method

| Method | Test error |
|---|---|
| FLDA | 3.123 |
| kNN | 1.423 |
| Linear SVM | 0.03 |

Since the sample is relatively small with respect to the space dimension, it should be easy for any method to find a criterion that separates the training set into two classes, but this does not necessarily imply the ability to correctly classify the test data. The construction depends on inner products → we will have to evaluate inner products in the feature space. This can be computationally intractable, if the dimensions become too large. It is apparent that the three methods have been able to find a method that perfectly separates the training data set into two classes, but only the linear SVM shows good performance when classifying new data points.

### *3.1. Algorithm of Classification and Regression*

The basics of a classification algorithm which has the following features:

1. Reduction of the classification problem to the computation of a linear decision function.
2. Absence of local minima in the SVM optimization problem.
3. A computationally efficient decision function (sparse solution).

*Algorithm of Classification*

Step: 1 Create H

Step: 2 select the parameters C, to find the large insensitive loss region

Step: 3 find α Objective function is maximized

$$\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\alpha^T H\alpha; \quad 0 \le \alpha_i \le C; \quad \sum_{i=1}^{n}\alpha_i y_i = 0$$

Step: 4 Calculate $w = \sum_{i=1}^{N}\alpha_i y_i \phi(x_i)$

Step: 5Determine the set of support vectors S by finding the indices i $0 \le \alpha_i \le C$

Step: 6 $b = \frac{1}{N_s}\sum_{s\in S}^{N}\left(y_s - \sum_{n\notin S}\alpha_n y_n \phi(X_n)\right)$

Step: 7 Each new point x' is classified be evaluating *y'= sign (w.ϕ(x')+b)*

*Algorithm of Regression*

Step: 1 select the parameters C and ε, to find the large insensitive loss region

Step: 2 find $\alpha^+$, $\alpha^-$ Objective function is maximized

$$\sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-)r_i - \varepsilon\sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-) - \frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-)\phi(x_i).(x_j)$$

$$0 \le \alpha_i^+ \le C \quad ;0 \le \alpha_i^- \le C; \qquad \sum_{i=1}^{N}\alpha_i^+ - \alpha_i^- = 0$$

Step: 3Calculate $w = \sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-)\phi(x_i)$

Step: 4

Determine the set of support vectors S by finding the indices i

$$0 \le \alpha_i \le C \quad and\ \xi_i = 0$$

Step: 5 $b = \frac{1}{N_s}\sum_{s\in S}^{N}\left[r_i - \varepsilon - \sum_{m,\varepsilon=1}^{N}(\alpha_i^+ - \alpha_i^-)x_i.x_m\right]$

Step: 6 Each new point x' is classified be evaluating *y'*= $\sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-)\phi(x_i).\phi(x') + b$

### *3.2. Percentage of misclassified data*

We have considered three kernels:

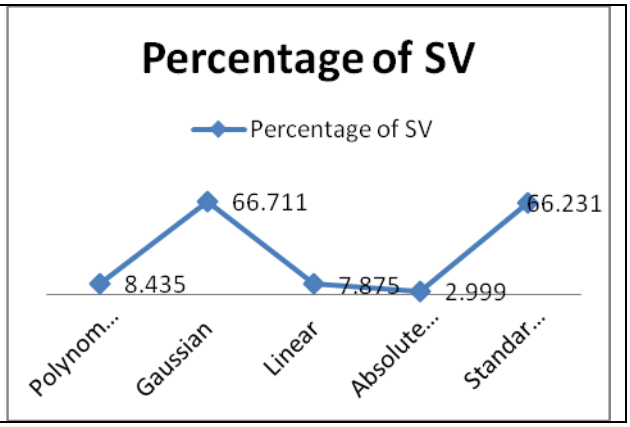
$$polynomial \quad \text{ker}\,nals : K_1(x,z) = (1 + x^T z)^2$$

$$Gaussian \quad Kernals : K_2(x,z) = \exp^{(-\|x-z\|^2)}; \quad Linear \quad Kernals : K_3(x,z) = x^T z$$

SVM is solving a classification or regression problem on data that is not linearly separable. We will compare SVMs using these kernels with the AV combination method and a semi definite programming (SDP) technique for building linear combinations of kernels. The data set has been randomly partitioned ten times into a training set and a test set, and for each method, a run of the experiment has been done over each partition. The AV method provides the best results (a test error of 3%), using significantly less support vectors than the other methods. The SDP method improves only the results of the Gaussian and the polynomial kernel.

Table: 2 Percentage of misclassified data and SV for the cancer data

| Method | Percentage of SV |
|---|---|
| Polynomial | 8.435 |
| Gaussian | 66.711 |
| Linear | 7.875 |
| Absolute value | 2.999 |
| Standard deviations in parentheses | 66.231 |




Get the objective function

$X_1^T X_1 = 0$, $X_1^T X_2 = 0$

$X_2^T X_1 = 0$, $X_2^T X_2 = 0$

Objective function

$$\frac{1}{2}\alpha_1^2 - (\alpha_1 + \alpha_2) = \frac{1}{2}\begin{bmatrix}\alpha_1 & \alpha_2\end{bmatrix}\begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\alpha_1\\ \alpha_2\end{bmatrix} - \begin{bmatrix}1 & 1\end{bmatrix}\begin{bmatrix}\alpha_1\\ \alpha_2\end{bmatrix}$$

Constraints
$\alpha_1 - \alpha_2 = 0$; $\alpha_2 = \alpha_1$
$1/2\alpha_1^2 - 2\alpha_2$
Smallest value $\alpha_1 = 2$
Considerations for supervised machine learning:
Prediction accuracy
Interpretability of the resulting model
Fitness of data to assumptions behind the method
Computation time
Accessibility of software to implement
Transform $\mathbf{x}_i$ to a higher dimensional space to "make classes linearly separable" Input space: the space $\mathbf{x}_i$; Feature space: the space of f ($\mathbf{x}_i$) after transformation. Linear operation in the feature space is equivalent to non-linear operation in input space.

## 4. One Class SVM

With this aim, the one-class SVM algorithm solves quadratic optimization problem

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 - b + \frac{1}{v_n}\sum_{i=1}^{n}\xi_i$$

$$s.t\ w^T\phi(x_i) \geq b - \xi_i, \quad i = 1,...,n; \qquad \xi_i \geq 0, \quad i = 1,...,n$$

Where $\xi_i$ are slack variables, v belongs to [0, and 1] is an a priori fixed constant which represents the fraction of outside points and b is the decision value which determines whether a given point belongs to the estimated high density region. The decision function will take the form h(**x**) = sign (**w***$^T\phi$(**x**) - b$^*$), where **w**$^*$ and b$^*$ are the values of **w** and b at the solution of problem. The hyperplane **w***$^T$ $\phi$ (**x**) - b$^*$ = 0 separates from the origin the mapped data for which the decision function h(**x**) = +1. Problem is smooth and convex, and follows the SVM idea of building a hyperplane in a feature space.

## 5. Conclusion

They allow easy construction of a nonlinear algorithm from a linear one. In a different direction, one could try to extend the notion of kernel so as to handle higher level similarities, such as analogies (which can be considered as similarities between pairs of examples). In classification problems generalization control is obtained by maximizing the margin, which corresponds to minimization of the weight vector in a canonical framework. The minimization of the weight vector can be used as a criterion in regression problems, with a modified loss function. The hyperplane **w***T F(**x**) - b* = 0 separates from the origin the mapped data for which the decision function h(**x**) = +1. We will compare SVMs using kernels with the AV combination method and a semi definite programming (SDP) technique for building linear combinations of kernels. The differentiable formulation of the SVM problem allows its solution by the use of standard Newton-type methods for convex optimization.

## References


[1]. Blanshard, G., Bousquet, O., and Massart, P. (2006). Statistical performance of support vector machines.
[2]. Buja, A., Swayne, D., Littman, M., Hofmann, H. and Chen, L. (2008), Data visualization with multidimensional scaling, Journal of Computational and Graphical Statistics.
[3]. Colin Campbell, Yiming Ying. Learning with Support Vector Machines. (2011), Synthesis Lectures on Artificial Intelligence and Machine Learning **5**:1, 1-95.
[4]. Emilio Carrizosa, Belen Martin-Barragan, Dolores Romero Morales, (2014). A nested heuristic for parameter tuning in Support Vector Machines. Computers & Operations Research **43**, 328-334.
[5]. Fukumizu, K., Bach, F. R. and Jordan, M. I. (2006). Kernel dimension reduction for regression. Technical report, Dept. Statistics, Univ. California, Berkeley.
[6]. Fan, R-E. Chang, K.-W. Hsieh, C.-J. Wang X.-R and Lin. C.-J. LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research, 9:1871 - 1874, (2008).
[7]. Matthias Varewyck, Jean-Pierre Martens. (2011), A Practical Approach to Model Selection for Support Vector Machines with a Gaussian Kernel. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) **41**:2, 330-340.
[8]. Premalatha, M. Vijayalakshmi C. (2012), Analysis of Soft Computing in Neural Network, in Proc 2nd International Conference in Computer Applications'12, Associated with ASDF, ACM, SERSC, Pondicherry, Vol 2, PP.no.172-177
[9]. Shai Shalev-Shwartz, Yoram Singer, Nathan Srebro, and Andrew Cotter. Pegasos, (2011), Primal estimated sub-gradient solver for SVM. Mathematical Programming, 127(1):3-30.
[10]. Vijaya Saradhi, V. Girish. K. R. (2014), Effective Parameter Tuning of SVMs Using Radius/Margin Bound Through Data Envelopment Analysis.
[11]. Xu J. (2010), Constructing a Fast Algorithm for Multi-label Classification with Support Vector Data Description. IEEE International Conference on Granular Computing (GrC2010), Aug. 14-16, 2010, San Jose CA, USA, pp. 817-821.
[12]. Yuan, M. and Lin, Y. (2007), Model selection and estimation in regression with grouped variables, Journal of the Royal Statistical Society, Series B 68(1): 49–67.
[13]. Zhixia Yang, Yingjie Tian, Naiyang Deng. (2009), Leave-one-out bounds for support vector ordinal regression machine. Neural Computing and Applications **18**:7, 731-748.