# SUPPORT VECTOR MACHINES

**Anshu Bharadwaj**
**Indian Agricultural Statistics Research Institute, New Delhi-11012**

## Introduction

SVMs deliver state-of-the-art performance in real-world applications such as text categorisation, hand-written character recognition, image classification, biosequences analysis, etc., and are now established as one of the standard tools for machine learning and data mining. A support vector machine (SVM) is a concept in computer science for a set of related supervised learning methods that analyze data and recognize patterns, used for classification and regression analysis. The standard SVM takes a set of input data and predicts, for each given input, which of two possible classes comprises the input, making the SVM a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

## Motivation

Classifying data is a common task in machine learning. Suppose some given data points each belong to one of two classes, and the goal is to decide which class a new data point will be in. In the case of support vector machines, a data point is viewed as a $p$-dimensional vector (a list of $p$ numbers), and we want to know whether we can separate such points with a $(p - 1)$-dimensional hyperplane. This is called a linear classifier. There are many hyperplanes that might classify the data. One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes. So we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized. If such a hyperplane exists, it is known as the maximum-margin hyperplane and the linear classifier it defines is known as a maximum margin classifier; or equivalently, the perceptron of optimal stability.

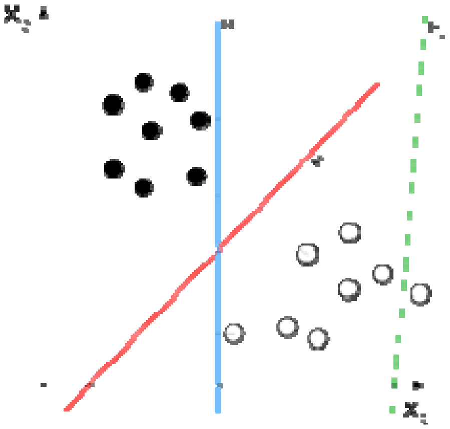

**Figure 1**: H3 (green) doesn't separate the two classes. H1 (blue) does, with a small margin and H2 (red) with the maximum margin.

## Support Vector Machines

The foundations of Support Vector Machines (SVMs) based on statistical learning theory have been developed by Vapnik (1998), Burges (1998), to solve the classification problem. The support vector machine (SVM) is the recent addition to the toolbox of data mining practitioners and are gaining popularity due to many attractive features, and promising empirical performance. They are a new generation learning system based on the latest advances in statistical learning theory (figure 2). The formulation embodies the Structural Risk Minimization (SRM) principle, which has been shown to be superior (Gunn, Brown & Bossely 1997), to traditional Empirical Risk Minimization (ERM) principle, employed by conventional neural networks. SRM minimizes an upper bound on the expected risk, as opposed to ERM that minimizes the error on the training data. It is this difference which equips SVM with a greater ability to generalize, which is the goal in statistical learning. The principle of SRM provides an upper bound to the generalization error of a classifier (R) in terms of its training error ($R_e$), the number of training examples (N), and the model complexity, otherwise known as its capacity (h). More specifically with a probability of $1-\eta$, the generalization error of the classifier can be worst at

$$R \leq R_e + \psi\left(\frac{h}{N}, \frac{\log(\eta}{N}\right) \qquad (1)$$

where $\psi$ is a monotone increasing function of the capacity *h.* SRM is also another way to express generalization error as a tradeoff between training error and model complexity. The capacity of a linear model is inversely related to its margin. Models with small margins have higher capacities because they are more flexible and can fit many training sets, unlike models with large margins. However, according to SRM principle, as the capacity increases, the generalization bound will also increase.

SVM belongs to the class of supervised learning algorithms in which the learning machine is given a set of examples (or inputs) with the associated labels (or output values). Like in decision trees, the examples are in the form of attribute vectors, so that the input space is a subset of $R^n$. SVM is a classifier that searches for a hyperplane with the largest margin, which is why it is known as maximum margin classifier. SVMs create a hyperplane that separates two classes (this can be extended to multi class problems). While doing so, SVM algorithm tries to achieve maximum separation between the classes. Separating the classes with a large margin minimizes a bound on the expected generalization error. By "minimum generalization error", it means that when new examples (data points with unknown class values) arrive for classification, the chance of making error in the prediction (of the class to which it belongs) based on the learned classifier (hyperplane) should be minimum. Intuitively, such a classifier is one which achieves maximum separation-margin between the classes. The two planes parallel to the plane are called bounding planes. The distance between these bounding planes is called margin and by SVM "learning", i.e. finding hyperplane which maximizes this margin. The points (in the dataset) falling on the bounding planes are called the support vectors. "Machine" in Support Vector Machines is nothing but the algorithm (Soman, Diwakar & Ajay 2006) (figure 3). SVM has greater advantages over ANNs and other classifiers since they are independent of the dimensionality of the feature space. Use of quadratic programming in SVM has an edge over ANNs classifier which gives only local minima whereas SVM provides global minima. SVM was designed initially as binary classifier i.e. it classifies the data into two classes but researchers have extended its boundaries to be a multi-class classifier. SVM was first introduced as a training algorithm (Boser, Guyon & Vapnik 1992) that automatically tunes the capacity of the classification function maximizing the margin between the training patterns and the decision boundary (Cristianini & Shawe-Taylor 2000). This algorithm operates with large class of decision functions that are linear in their parameters but not restricted to linear dependences in the input components. For the computational considerations, SVM works well on the two important practical considerations of classification algorithms i.e. speed and convergence.

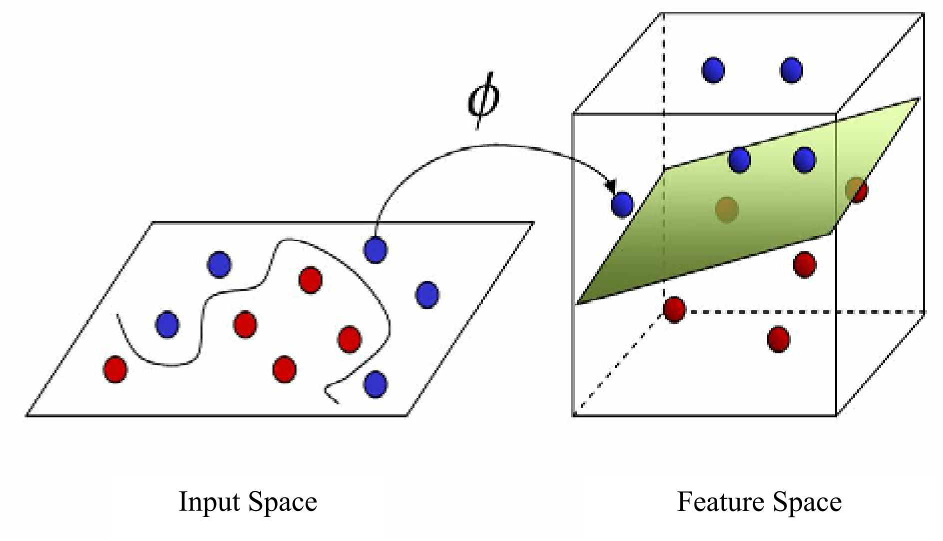

**Figure 2:** Principles of SVM

## Theoretical Development of SVM

There are a number of publications detailing the mathematical formulation of the SVM. The inductive principle behind SVM is structural risk minimization (SRM). According to Vapnik (1995), the risk of a learning machine (R) is bounded by the sum of the empirical risk estimated from training samples $(R_{emp})$ and a confidence interval $(\Psi)$:$R \leq R_{emp} + \Psi$. The strategy of $SVM$ is to keep the empirical risk $(R_{emp})$ fixed and to minimize the confidence interval $(\Psi)$, or to maximize the margin between a separating hyper plane and closet data points. A separating hyper plane refers to a plane in a multi-dimensional space that separates the data samples of two classes. The optimal separating hyper plane is the separating hyper plane that maximizes the margin from closest data points to the plane. Currently, one $SVM$ classifier can only separate two classes. Integration strategies are needed to extend this method to classifying multiple classes.

## The Optimal Separating Hyperplane

Let the training data of two separable classes with $k$ samples be represented by $(x_1, y_1), \dots, (x_k, y_k)$ where $x \in R^n$ is an n-dimensional space,

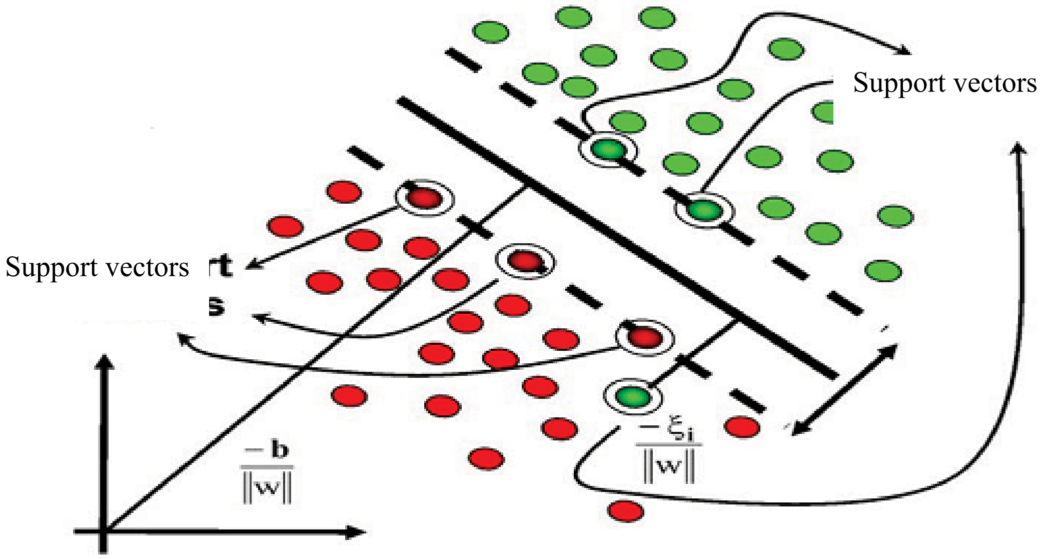

**Figure 3:** Schematic Diagram of a Support Vector Machine

and $y \in \{+1, -1\}$ is class label. Suppose the two classes can be presented by two hyper planes parallel to the optimal hyper plane (figure 3):

$$w.x_i + b \geq 1 \qquad for\ y_i = 1, i = 1, 2, \dots, k \qquad (2)$$

$$w.x_i + b \leq -1 \qquad for\ y_i = -1 \qquad (3)$$

where $w = (w_1, \dots, w_n)$ is a vector of *n* elements. Inequalities (2) and (3) can be combined into a single inequality:

$$y_i[w'x_i + b] \geq 1 \qquad i = 1, \dots, k \qquad (4)$$

As shown in figure 3, the optimal separating hyperplane is the one that separates the data with maximum margin. This hyperplane can be found by minimizing the norm of *w*, or the following function:

$$F(w) = \frac{1}{2}(w'w) \qquad (5)$$

Under inequality constraint (4)

The saddle point of the following Lagrangian gives solutions to the above optimization problem:

$$L(w, b, \alpha) = \frac{1}{2}(w'w) - \sum_{i=1}^{k} \alpha_i \{y_i[w'x_i + b] - 1\} \qquad (6)$$

Where $\alpha_i \geq 0$ are Lagrange multipliers (Sundaram 1996). The solution to this optimization problem requires that the gradient of $L(w, b, \alpha)$ with respect to *w* and *b* vanishes, giving the following conditions:

$$w - \sum_{i=1}^{k} y_i \alpha_i x_i \qquad (7)$$

$$\sum_{i=1}^{k} \alpha_i y_i = 0 \qquad (8)$$

By substituting (7) and (8) into (6), the optimization problem becomes:

Maximize

$$L(\alpha) = \sum_{i=1}^{k} \alpha_i - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i \alpha_j y_i y_j (x_i' x_j) \qquad (9)$$

Under constraints $\alpha_i \geq 0, i = 1, \dots, k$

Given an optional solution $\alpha^0 = (\alpha_1^0, \ldots, \alpha_k^0)$ to (8), the solution $w^0$ to (7) is a linear combination of training samples:

$$w^0 = \sum_{i=1}^{k} y_i \alpha_i^0 x_i \qquad (10)$$

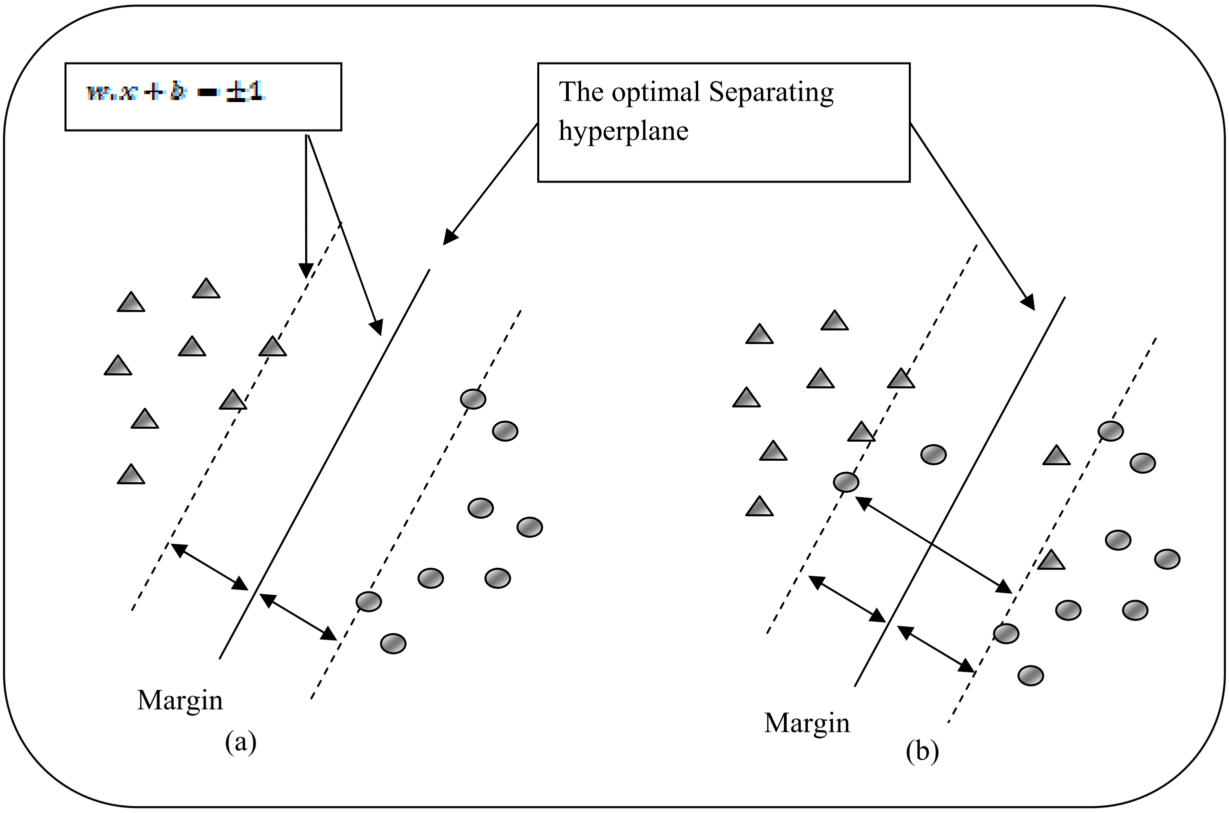

**Figure 4:** SVM classification

According to the Kuhn-Tucker theory (Sundaram 1996), only points that satisfy the equalities in (2) and (3) can have non-zero coefficients $\alpha_i^0$. These points lie on the two parallel hyperplane and are called support vectors (figure 4). Let $x^0(1)$ be a support vector of one class and $x^0(-1)$ of the other, then the constant $b^0$ can be calculated as follows:

$$b^0 = \frac{1}{2}\left[w^{0'} x^0(1) + w^{0'} x^0(-1)\right] \qquad (11)$$

The decision rule that separates the two classes can be written as:

$$f(x) = sign\left(\sum_{support\ vector} y_i \alpha_i^0 (x_i{'}x) - b^0\right) \qquad (12)$$

**Kernel Trick**

To generalize the above method to non-linear decision functions, the support vector machine implements the following idea: it maps the input vector *x* into a high-dimensional feature space *H* and constructs the optimal separating hyperplane in the space. Suppose the data are mapped into a high-dimensional space *H* through mapping function $\Phi$ :

$$\Phi : R^{n} \rightarrow H \qquad (13)$$

A vector *x* in the feature space can be represented as $\Phi(\mathbf{x})$ in the high-dimensional space *H*. Since the only way in which the data appear in the training program are in the form of dot product of the two vectors, the training algorithm in the high-dimensional space *H* would only depend on data in this space through a dot product, i.e. on functions of the form $\Phi(x_i)^t\Phi(\mathbf{x}_j)$. Now, if there is a kernel function *K* such that

$$k(x_i, x_j) - \Phi(x_i)^t\Phi(x_j) \qquad (14)$$

*K* has to be used in the training program without knowing the explicit form of $\Phi$. The same trick can be applied to the decision function because the only form in which the data appear are in the form of dot products. Thus, if a kernel function *K* can be found, a classifier can be trained and used in the high-dimensional space without knowing the explicit form of the mapping function. The optimization problem (3.9) can be rewritten as:

$$L(\alpha) = \sum_{i=1}^{k} \alpha_i - \frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{k} \alpha_i\alpha_j y_i y_j K(x_i{}' x_j) \qquad (15)$$

And the decision rule expressed in equation (12) becomes:

$$f(x) = sign\left(\sum_{support\ vector} y_i\alpha_i^0 K(x_i{}' x) - b^0\right) \qquad (16)$$

A kernel that can be used to construct a SVM must meet Mercer's condition (Courant and Hilbert 1953). Polynomial kernels and  Radial basis functions (RBF) kernel meet this condition (Vapnik 1995). Throughout the course of this research, Radial Basis Kernel has been used in SVM classifier which is explained as follows:

A Radial Basis Function (RBF) is a real-valued function whose value depends only on the distance from the origin, so that$\emptyset(x_1) = \emptyset(\|x_1\|)$; or alternatively on the distance from some other point $x_2$, called a center, so that $\emptyset(x_1 x_2) = \emptyset(\|x_1 - x_2\|)$. Any function $\phi$ that satisfies the property $\emptyset(x_1)$=$\emptyset(\|x_1\|)$ is a radial function. The norm is usually used, although other distance functions are also possible. The following expression describes the Radial Basis Function Kernel for SVM:

$$\phi = \exp\left\{-\gamma|x_1 - x_2|^2\right\} \quad , \quad \text{where } \gamma > 0 \qquad (17)$$

$\gamma$ is called the RBF kernel parameter. The RBF kernel is the most popular kernel type due to its localized and finite response across the entire range of real x-axis.

### Properties

SVMs belong to a family of generalized linear classifiers and can be interpreted as an extension of the perceptron. They can also be considered a special case of Tikhonov regularization. A special property is that they simultaneously minimize the empirical *classification error* and maximize the *geometric margin*; hence they are also known as maximum margin classifiers.

### Parameter selection

The effectiveness of SVM depends on the selection of kernel, the kernel's parameters, and soft margin parameter C.

A common choice is a Gaussian kernel, which has a single parameter $\gamma$. Best combination of $C$ and $\gamma$ is often selected by a grid-search with exponentially growing sequences of $C$ and $\gamma$, for example, $C \in \{2^{-5}, 2^{-3}, \ldots, 2^{13}, 2^{15}\}$ ; $\gamma \in \{2^{-15}, 2^{-13}, \ldots, 2^{1}, 2^{3}\}$ . Typically, each combination of parameter choices is checked using cross validation, and the parameters with best cross-validation accuracy are picked. The final model, which is used for testing and for classifying new data, is then trained on the whole training set using the selected parameters.

## Issues

Potential drawbacks of the SVM are the following two aspects:

- Uncalibrated class membership probabilities
- The SVM is only directly applicable for two-class tasks. Therefore, algorithms that reduce the multi-class task to several binary problems have to be applied; see the multi-class SVM section.
- Parameters of a solved model are difficult to interpret.

### Summary and Conclusions

The support vector machine has been introduced as a robust tool for many aspects of data mining including classification, regression and outlier detection. The SVM for classification has been detailed and some practical considerations mentioned. The SVM uses statistical learning theory to search for a regularized hypothesis that fits the available data well without over-fitting. The SVM has very few free parameters, and these can be optimized using generalisation theory without the need for a separate validation set during training. The SVM does not fall into the class of 'just another algorithm' as it is based on firm statistical and mathematical foundations concerning generalisation and optimisation theory. Moreover, it has been shown to outperform existing techniques on a wide variety of real world problems. SVMs will not solve all of the problems, but as kernel methods and maximum margin methods are further improved and taken up by the data mining community they will become an essential tool in any researcher's toolkit who deals with data mining particularly classification.