

SUPPORT VECTOR MACHINES - A FIRST LOOK

Aravind Ganapathiraju

Institute for Signal and Information Processing
Department of Electrical and Computer Engineering
Mississippi State University
Box 9571

Mississippi State, MS-39769

Tel: 601-325-8335

Fax: 601-325-3149

email: ganapath@isip.msstate.edu



1. Introduction

Support Vector Machine (SVM) is a new learning machine for a 2-group classification problem. In this paradigm, input vectors are mapped into a higher-dimensional feature space through an *a priori* chosen non-linear mapping, where a linear classification decision region is constructed. This decision region when mapped back into the original feature space can take a non-linear form. This is one of the greatest assets of SVMs where constructing non-linear decision regions has better classification ability than traditional classification techniques like Linear Discriminant Analysis. Also, special properties of the decision surface ensures good generalization ability of this learning machine. In many 2-class problems SVMs have been found to be more effective than other non-linear classifiers like Neural Networks, k-Nearest Neighbor classifiers.

2. Concept

The technical problem involved in moving to a higher dimensional space for classification is computation effort. For example a polynomial of degree 4 in a 30 dimensional space will have to be transformed to a 30^4 dimensional feature space which is prohibitive even for modern day computers. The conceptual problem involved is the lack of a definitive method of choosing a separating hyperplane which generalizes well. A solution to this problem was suggested by Vapnik in 1965. It was observed that to construct an optimal hyperplane one only has to take into account a small part of the training set called the *support vectors*. It was shown that if the training vectors are separated without errors by an optimal hyperplane, the expected error rate on a test sample is bounded by the ratio of the expectation of the support vectors to the number of training vectors. Since this ratio is independent of the dimension of the problem, if we can find a small set of support vectors, good generalization is guaranteed.

Let the optimal hyperplane be represented as,

$$w_0 \cdot z + b_0 = 0 \quad (1)$$

It can be shown that the weights w_0 for the optimal hyperplane in the feature space can be written as a linear combination of the support vectors as

$$w_0 = \sum \alpha_i \cdot z_i \quad (2)$$

Note the dot product involved in the above computations. It was shown that by interchanging some of the operations in the process, SVM can be made computationally tractable. One suggested method was to do the dot product operation in the input space and then making a non-linear transformation to the feature space.

3. Features of Support Vector Machines

1. Constructing the Decision Rules by SVMs is efficient - Quadratic optimization problem. Most implementations use specific quadratic programming packages for this purpose. Matlab's optimization toolbox supports this too.
2. SVMs are very general - By choosing the right dot product definition, one can construct polynomial decision regions of any order (systems with order 7 have also been reported).
3. Control of Open loop vs. Closed loop performance - principle of structural risk minimization []. One can theoretically sacrifice some errors on the training data for better generalizability. This trade-off is controlled by a single parameter which turns out to be the confidence interval of the decision region.

4. Current Applications

Most applications of SVMs so far have been in the area of image identification, face detection in images etc. Some reported SVM applications include,

1. Digit Recognition - 18 X 18 images of digits. Needs 10 different classifiers. SVM performs better than CART,

C4.5, and even a 5-layer NN.

2. Text Categorization - Differentiate between business vs. non-business reports. SVMs perform at 97.5% accuracy.

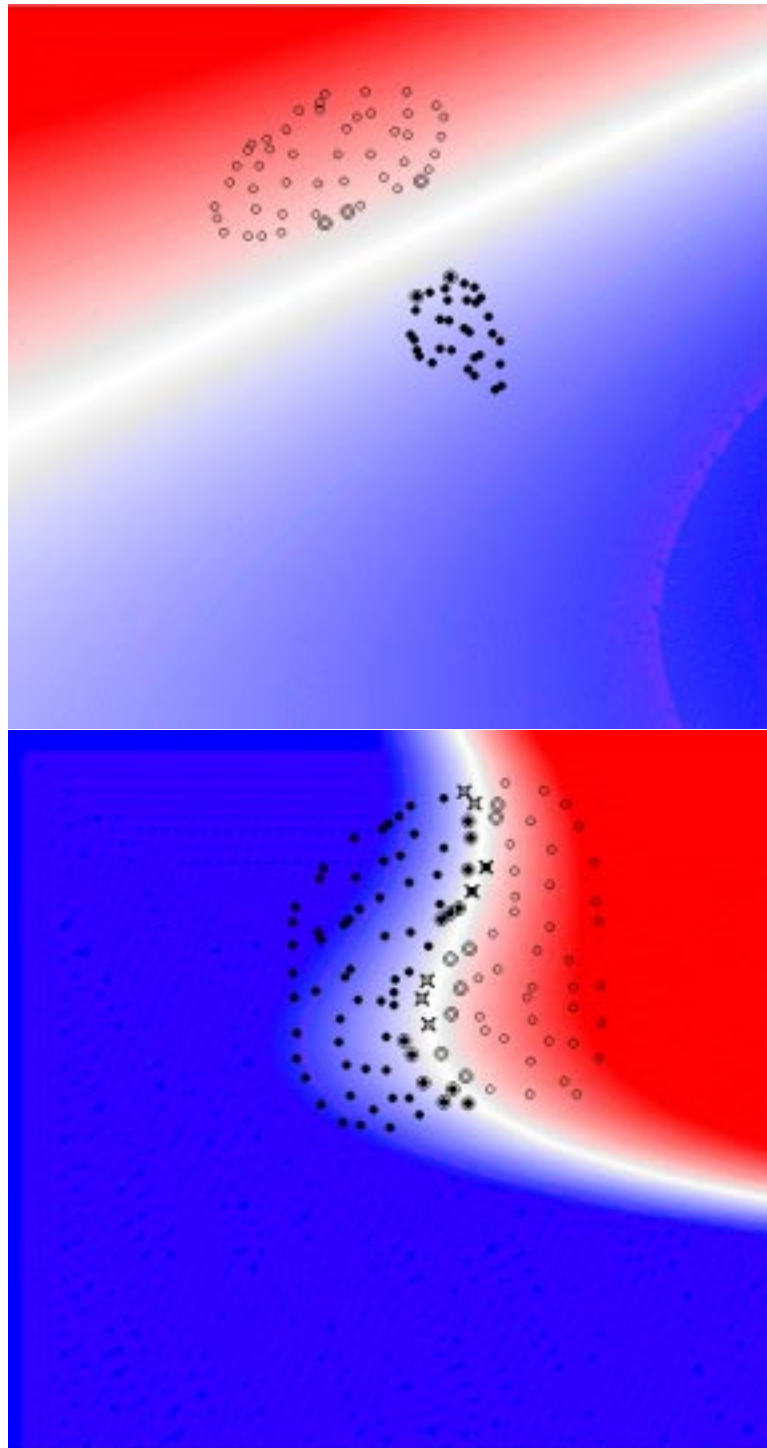


Figure 1: Sample classifications generated by the AT&T applet. Note the top figure was generated using a simple dot matrix metric and ends up as a linear classifier (similar to LDA)

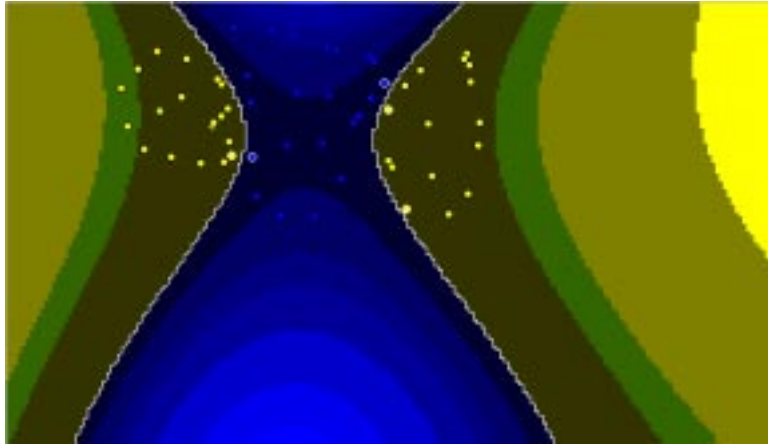


Figure 2: Sample classification by the Royal Holloway SVM applet. A 3rd degree polynomial was used in this case. This is interesting in the sense that SVMs handle multi-modal data effectively.

5. References

- [1] Corrina Cortes, Vladimir Vapnik, "Support Vector Networks", Machine Learning, vol. 20, September 1995.
- [2] Vladimir Vapnik, Estimation of Dependences Based on Empirical Data, Springer-Verlag, New York, New York, 1982.
- [3] Vladimir Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, New York, New York, 1995.
- [4] <http://www.isis.ecs.soton.ac.uk/research/svm>, Support Vector Machines, March 1998.
- [5] <http://svm.first.gmd.de>, Support Vector Machines, March 1998.
- [6] <http://svm.research.bell-labs.com/SVT/SVMsvt.html>, Demonstration 2-D Pattern Recognition Applet., March 1998.
- [7] <http://svm.dcs.rhnc.ac.uk>, Pattern Recognition SVM-Applet, March 1998.