WORKSHOP ON SUPPORT VECTOR MACHINES: THEORY AND APPLICATIONS

Theodoros Evgeniou and Massimiliano Pontil
Center for Biological and Computational Learning, and
Artificial Intelligence Laboratory,
MIT, E25-201,
Cambridge, MA 02139,
USA

Abstract

This paper presents a summary of the issues discussed during the one day workshop on "Support Vector Machines (SVM) Theory and Applications" organized as part of the Advanced Course on Artificial Intelligence (ACAI 99) in Chania, Greece. The goal of the paper is twofold: to present an overview of the background theory and current understanding of SVM, and to discuss the papers presented as well as the issues that arose during the workshop.

INTRODUCTION

Support Vector Machines (SVM) have been recently developed in the framework of statistical learning theory (Vapnik, 1998) (Cortes and Vapnik, 1995), and have been successfully applied to a number of applications, ranging from time series prediction (Fernandez, 1999), to face recognition (Tefas et al., 1999), to biological data processing for medical diagnosis (Veropoulos et al., 1999). Their theoretical foundations and their experimental success encourage further research on their characteristics, as well as their further use.

In this report we present a brief introduction to the theory and implementation of SVM, and we discuss the five papers presented during the workshop. The report is organized as follows: section 2 presents the theoretical foundations of SVM. A brief overview of statistical learning theory also using the discussion in (Vayatis and Azencott, 1999) is given. The mathematical formulation of SVM is presented, and theory for the implementation of SVM, as in (Trafalis, 1999), is briefly discussed. Section 3 summarizes the experimental work of (Veropoulos et al., 1999), (Fernandez, 1999) and (Tefas et al., 1999) and the variations of the standard SVM proposed in these papers. Finally section 4 presents some conclusions and suggestions for future research.

A BRIEF OVERVIEW OF THE SVM THEORY

Support Vector Machines have been developed in the framework of Statistical Learning Theory - see for example (Vapnik, 1998). We first briefly discuss some basic ideas of the theory.

Statistical Learning Theory: a primer

In statistical learning theory (SLT) the problem of supervised learning is formulated as follows. We are given a set of l training data $\{(\mathbf{x}_1,y_1)...(\mathbf{x}_l,y_l)\}$ in $R^n \times R$ sampled according to unknown probability distribution $P(\mathbf{x},y)$, and a loss function $V(y,f(\mathbf{x}))$ that measures the error done when, for a given $\mathbf{x}$, $f(\mathbf{x})$ is "predicted" instead of the actual value y. The problem consists in finding a function f that minimizes the expectation of the error on new data, that is, find a function f that minimizes the expected error:

$$\int V(y,f(\mathbf{x}))\,P(\mathbf{x},y)\,d\mathbf{x}\,dy$$

Since $P(\mathbf{x},y)$ in unknown, we need to use some induction principle in order to infer from the l available training examples a function that minimizes the expected error. The principle used is Empirical Risk Minimization (ERM) over a set of possible functions, called hypothesis space. Formally this can be written as minimizing the empirical error:

$$\frac{1}{l}\sum_{i=1}^{l} V(y_i, f(\mathbf{x}_i))$$

with f being restricted to be in a space of functions - hypothesis space - say H. An important question is how close the empirical error of the solution (minimizer of the empirical error) is to the minimum of the expected error that can be achieved with functions from H. A central result of the theory states the conditions under which the two errors are close to each other, and provides probabilistic bounds on the distance between empirical and expected errors (see theorem 1 below). These bounds are given in terms of a measure of complexity of the hypothesis space H: the more "complex" H is, the larger the distance between the empirical and expected errors is in probability (see theorem 1 below).

From the bounds that the theory provides, it occurs that it is possible to improve the ERM inductive principle by considering a structure of hypothesis spaces $H_1 \subset H_2 \subset ... \subset H_m$, with ordered "complexity" (i.e. H_{i+1} is more "complex" than H_i). ERM is performed in each of these spaces, and the choice of the final solution can be done using the aforementioned bounds. This principle of performing ERM over a structure (sequence) of nested hypothesis spaces is known as Structural Risk Minimization (SRM) (Vapnik, 1998).

An important question that arises in SLT is that of measuring the "complexity" of a hypothesis space - which, as we discussed, we need in order to choose the final optimal solution to the learning problem. In (Vayatis and Azencott, 1999) quantities measuring this "complexity" of a hypothesis space are discussed, and suggestions for how to measure such quantities experimentally are made. We briefly describe the "complexity" quantities discussed in (Vayatis and Azencott, 1999).

The first quantity discussed is a standard one in SLT (Vapnik, 1998), and is called the VC dimension of a set of functions. This is a combinatorial quantity that characterizes the capacity of the set of functions to shatter a set of points (for more information we refer the reader to the literature). Using the VC dimension of a hypothesis spaces H, the aforementioned distance between empirical and expected errors can be bound as follows:

Theorem 1 (Vapnik and Chervonenkis, 1971) *If V is the VC-dimension of a hypothesis space H, then with probability 1-η, the minimum of the expected error that can be achieved with functions from H, say L, and the minimum empirical error, say L_{emp}, satisfy the constraint*:

$$L_{emp} - 4\sqrt{2}\sqrt{\frac{V\left(1+\log\left(\frac{2l}{V}\right)\right)-\log\left(\frac{\eta}{4}\right)}{l}} \leq L \leq L_{emp} + 4\sqrt{2}\sqrt{\frac{V\left(1+\log\left(\frac{2l}{V}\right)\right)-\log\left(\frac{\eta}{4}\right)}{l}}$$

*independent of the distribution of the data P(**x**,y).*

Theorem 1 holds independently of the probability distribution of the data P(**x**,y). In (Vayatis and Azencott) the distribution P(**x**,y) is also taken into account (similar work has already been done in the past - for example see (Vapnik 1998) and references therein), and the distance between empirical and expected error is bounded using a "complexity" quantity that takes P(**x**,y) into account. The bounds presented are tighter than that of theorem 1, but require the knowledge of the distribution dependent complexity quantity. The goal of (Vayatis and Azencott, 1999) was to introduce a possible experimental setup to compute the distribution-depended complexity quantity they suggest. To summarize, (Vayatis and Azencott, 1999) discuss the basic theoretical framework in which learning machines such as SVM have been developed, and suggest possible directions of research that can lead to improvements of the theory and so possibly to improvements of SVM (for example the theory can be used to choose the parameters of an SVM, such as the kernel and the regularization parameter C - see below).

We now discuss how SVM emerge from the theoretical framework we outlined.

Support Vector Machines formulation

Support Vector machines realize the ideas outlined above. To see why, we need specify two things: the hypothesis spaces used by SVM, and the loss functions used. The folklore view of SVM is that they find an "optimal" hyperplane as the solution to the learning problem. The simplest formulation of SVM is the linear one, where the hyperplane lies on the space of the input data **x**. In this case the hypothesis space is a subset of all hyperplanes of the form:

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b.$$

In their most general formulation, SVM find a hyperplane in a space different from that of the input data x. It is a hyperplane in a feature space induced by a kernel K (the kernel defines a dot product in that space (Wahba, 1990)). Through the kernel K the hypothesis space is defined as a set of "hyperplanes" in the feature space induced by K. This can also be seen as a set of functions in a Reproducing Kernel Hilbert Space (RKHS) defined by K (Wahba, 1990), (Vapnik, 1998). We do not discuss RKHS here and refer the reader to the literature.

So to summarize, the hypothesis space used by SVM is a subset of the set of hyperplanes defined in some space - an RKHS. This space can be formally written as

$$\left\{ f : \|f\|_K^2 < \infty \right\}$$

where K is the kernel that defines the RKHS, and $\|f\|_K^2$ is the RKHS norm of the function (Wahba, 1990). For example, for the linear case mentioned above, K is the kernel $K(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1 \cdot \mathbf{x}_2$, the functions considered are of the form $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$, and the RKHS norm of these functions is simply the norm of **w**, namely $\|f\|_K^2 = \|\mathbf{w}\|^2$.

In fact SVM consider subsets of this space, namely sets of the form

$$\left\{ f : \|f\|_K^2 \leq A^2 \right\}$$

for some constant A. In the SLT framework discussed above, the constant A is used to define a structure of hypothesis spaces (the larger A is, the more complex the hypothesis space is). The goal of SVM is to find the solution with the "optimal" RKHS norm, that is, to find the optimal A.

Instead of searching many hypothesis spaces one by one by performing ERM for each choice of A, SVM search for an A (or the optimal RKHS norm $\|f\|_K^2$) in a different way, as it will be obvious from the SVM formulation presented below. This "search method" for the optimal $\|f\|_K^2$ has been extensively discussed in the literature (see for example (Bartlett and Shawe-Taylor, 1998), (Burges, 1998), (Evgeniou et al., 1999)), and we do not discuss it here any further.

The second choice is that of the loss function. For this we have to distinguish between SVM classifiers and SVM regressors. For classification ideally the misclassification error needs to be minimized, so a loss function of the form sign(-yf($\mathbf{x}$)) should be used (in classification y takes binary values ±1, and classification is done by taking the sign of function f($\mathbf{x}$)). However because of scaling as well as computational reasons (Vapnik, 1998), the actual loss function used for SVM classification is $|1-yf(\mathbf{x})|_+$ (that is, 0 if $1-yf(\mathbf{x}) < 0$, and $1-yf(\mathbf{x})$ otherwise). This is also called the "soft margin" loss function because of its standard "margin" interpretation: the points for which the loss function is zero are the ones that have "margin"

$$yf(\mathbf{x})/\|f\|_K^2$$

at least $1/\|f\|_K^2$ (that is $1 - yf(\mathbf{x}) \leq 0 \Rightarrow yf(\mathbf{x})/\|f\|_K^2 \geq 1/\|f\|_K^2$). The margin is an important geometric quantity associated with SVM classification. For more information we refer the reader to the literature.

For regression the loss function used is the so-called epsilon-insensitive loss function $|y-f(\mathbf{x})|_\varepsilon$ which is equal to $|y-f(\mathbf{x})|-\varepsilon$ if $|y-f(\mathbf{x})| > \varepsilon$, and 0 otherwise.

To summarize, following the SLT ideas outlined above for the given choices of the loss function and the hypothesis spaces, SVM are learning machines that minimize the empirical error while taking into account the "complexity" of the hypothesis space used by also minimizing the RKHS norm of the solution $\|f\|_K^2$. SVM in practice minimize a trade off between empirical error and complexity of hypothesis space. Formally this is done by solving the following minimization problems:

SVM classification

$$\min_f \|f\|_K^2 + C\sum_{i=1}^{l} |1 - y_i f(\mathbf{x}_i)|_+ \qquad (1)$$

SVM regression

$$\min_f \|f\|_K^2 + C\sum_{i=1}^{l} |y_i - f(\mathbf{x}_i)|_\varepsilon \qquad (2)$$

where C is a so called "regularization parameter" that controls the trade off between empirical error and complexity of the hypothesis space used.

Having discussed how SVM stem out of the theory outlined above, we now turn to their actual implementation. The next section briefly discusses how the minimization problems (1) and (2) can be done, taking also into account (Trafalis, 1999).

SVM IMPLEMENTATION

As mentioned above, training an SVM means solving problems (1) or (2). It turns out that both problems can be rewritten as constrained quadratic programming (QP) problems. We present the QP formulation for SVM classification, and regarding regression we refer the reader to the literature (Vapnik, 1998).

Problem (1) can be rewritten as follows:

SV classification:

$$\min_{f,\xi_i} \|f\|_K^2 + C\sum_{i=1}^{l} \xi_i \qquad (3)$$

$$\text{subject to: } y_i f(\mathbf{x}_i) \geq 1 - \xi_i, \text{ for all i}$$
$$\xi_i \geq 0$$

Variables ξ_i are called slack variables and they measure the error made at point $(\mathbf{x}_i,y_i)$. We see that the number of constraints is equal to the number of training data, l. Training SVM, that is, solving constrained QP problem (3), becomes quite challenging when the number of training points is large. A number of methods for fast SVM training have been proposed in the literature. Decomposing the QP problem into a number of smaller ones through chunking or decomposition algorithms is one approach suggested (for example see (Osuna et al., 1997), (Vapnik, 1998)). A sequential optimization method has also recently been proposed (Platt, 1998).

In (Trafalis, 1999) the approach suggested to solve the QP problem (3) was that of Interior Point Methods (IPM). Trafalis (Trafalis, 1999) presents an overview of IPM, concentrating on primal-dual optimization methods. These methods consist of solving iteratively the QP problem by moving between the formulation (3) (called the "primal" formulation) and its dual formulation, which can be found to be (see (Vapnik, 1998), (Trafalis, 1999)):

SVM classification, Dual formulation:

$$\min_{\alpha_i} \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{4}$$

$$\text{subject to: } 0 \leq \alpha_i \leq C, \text{ for all i}$$

$$\sum_{i=1}^{l} \alpha_i y_i = 0$$

Typically in the literature SVM are trained by solving the dual optimization problem (4) ((Osuna et al., 1997), (Vapnik, 1998), (Burges, 1998)). Trafalis (Trafalis, 1999) proposes primal-dual IPM methods for SVM training which differ from the ones typically used.

In (Trafalis, 1999) the IPM discussed are also used to train learning machines other than SVM. In particular, (Trafalis, 1999) shows how the proposed primal-dual IPM can be used to train Artificial Neural Networks (ANN) typically trained using backpropagation One of the main differences between ANN and SVM is that, as mentioned in (Trafalis, 1999), while for ANN there can be many local optimal solutions, for SVM there is only one optimal solution for problem (3), since SVM are trained by solving a QP problem which has one global optimal solution. This is one practical "advantage" of SVM when compared with ANN.

EXPERIMENTS WITH SVM AND SOME VARIATIONS

Application of SVM to medical decision support.

The paper (Veropoulos et al., 1999) proposed an application of SVM classifiers to medical diagnosis of Tuberculosis from photomicrographs of Sputum smears. Except the fact that this is the first time that SVMs are used in a medical problem, another interesting point was the introduction of two methods that can be used for controlling the performance of the system on a particular class of the data (that is, force the SVM to better classify the data from one of the two classes of the classification task). In most medical problems, medical experts must have the ability to put more weight on one of the classes of the problem (usually the class on which the diagnosis is 'heavily' based). Another common problem in a wider area of applications is the presence of unbalanced data sets (the set of examples from one class is significantly larger than the set of examples from the other class). For these reasons, controlling the performance of a system on a particular class of the data is practically very useful. To do so, (Veropoulos et al., 1999) used a slightly modified version of the standard SVM formulation (3) – the same idea was suggested in (Osuna et al., 1997). The idea is to use different regularization parameter C for each of the two classes. This translates in the following SVM formulation:

$$\min_{f,\xi_i} \|f\|_K^2 + C_1 \sum_{i \in class_1} \xi_i + C_2 \sum_{i \in class_2} \xi_i \tag{5}$$

$$\text{subject to: } y_i f(\mathbf{x}_i) \geq 1 - \xi_i, \text{ for all i}$$

$$\xi_i \geq 0$$

By changing the ratio C_1/C_2, (Veropoulos et al., 1999) showed how to influence the performance of the SVM on one of the classes, therefore altering the false negative vs false positive ratio for one of the classes.

A different approach to dealing with the problem of unbalanced data or to putting more weight on one of the classes is also discussed in (Veropoulos et al., 1999). This second approach is based on a version of SVM slightly different from the one described above, so we do not discuss it here and refer the reader to (Veropoulos et al., 1999).

Time series prediction using Local SVMs

An application of SVM regression was discussed in (Fernandez, 1999). The problem was time series prediction. The approach taken was the use of SVM regression to model the dynamics of the time series and subsequently predict future values of the series using the constructed model. Instead of using the standard SVM regression formulation described above, a variation developed in (Scholkopf et al., 1998) was used. Using this variation the ε parameter of the SVM regression loss function (see above) is automatically estimated. Furthermore, (Fernandez,

1999) used an approach to learning which is different from the standard one: instead of developing one global regression model from all the available training data, (Fernandez, 1999) develops a number of SVM regression models, each one trained using only part of the initial training data. The idea, which has been suggested in (Bottou and Vapnik, 1992), is to split the initial training data set into parts, each part consisting only of training data that are close to each other (in a Euclidean distance sense). Then a "local" SVM is trained for each subset of the data. The claim in (Bottou and Vapnik, 1992) is that such an approach can lead to a number of simple (low complexity, in the SLT sense outlined above) learning machines, instead of a single machine that is required to fit all data.

In (Fernandez, 1999) each of the individual SVM machines had its ε parameter estimated independently. The ε parameter of the SVM loss function is known to be related to the noise of the data (Pontil et al., 1998). So, in a sense, the approach of (Fernandez, 1999) leads to local SVMs each having an ε parameter that depends on the noise of the data in particular regions of the space (instead of a single ε that needs to "model" the noise of all the data).

The experiments described in (Fernandez, 1999) show that training many local SVMs instead of one global learning machine leads to significant improvements in performance. In fact, this was also the finding of (Bottou and Vapnik, 1992) who first showed experiments with local learning machines.

An Application of SVM to face authentication

Starting from Fisher Linear Discriminant (FLT) (Duda and Hart, 1973), (Tefas et al., 1999) develop variations of this standard classification method, and compare them with SVM classification. In (Tefas et al., 1999) ideas behind the formulation of FLT are used to, effectively, choose a kernel for SVM classification. We briefly review FLT, and we then show the SVM used in (Tefas et al.).

Given data from two classes, FTL leads to a hyperplane separating the two classes using the following criterion: the projections of the data on the hyperplane are such that the between-class variance of the projections is maximized, while the within-class variance is minimized (Duda and Hart, 1973). If $\mathbf{m}_1$ and $\mathbf{m}_2$ are the means of the two classes, and S_b and S_w are the between-class and within class scatter matrices, then FLT yields the hyperplane $\mathbf{w}$ that maximizes the so called Fisher discriminant ratio $\mathbf{w}^T S_b \mathbf{w} / \mathbf{w}^T S_w \mathbf{w}$.

In (Tefas et al., 1999) the solution $\mathbf{w}^{FLT}$ of FLT is also shown to be the optimal solution of a constrained QP optimization problem. It is noted in (Tefas et al., 1999) that SVM are also formulated as constrained QP problems (as we discussed in the previous section), and a comparison between the SVM formulation with the one yielding $\mathbf{w}^{FLT}$ is made.

Furthermore, (Tefas et al., 1999) use the idea of FLT of maximizing the within class variance in order to design a kernel K for SVM classification. To see how this is done, we state again the SVM formulation for linear kernels, in which case, as discussed above, the RKHS norm of the functions $f(\mathbf{x}) = \mathbf{w}\cdot\mathbf{x} + f$ is simply $\|\mathbf{w}\|^2$. In this case SVM is formulated as:

SV linear classification

$$\min_{\mathbf{w},\xi_i} \|\mathbf{w}\|^2 + C\sum_{i=1}^{l}\xi_i$$

$$\text{subject to: } y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 - \xi_i, \text{ for all i}$$

$$\xi_i \geq 0$$

Instead of using this linear SVM machine, (Tefas et al., 1999) use the following machine:

SV classification with "FLT" kernel:

$$\min_{\mathbf{w},\xi_i} \mathbf{w}^T S_w \mathbf{w} + C\sum_{i=1}^{l}\xi_i \tag{6}$$

$$\text{subject to: } y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 - \xi_i, \text{ for all i}$$

$$\xi_i \geq 0$$

It turns out that (Tefas et al., 1999) use an SVM classifier with kernel $K(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^{\mathbf{T}} S_w^{-1} \mathbf{x}_2$ (this can be seen through the dual formulation of problem (6)). In (Tefas et al., 1999) the "FLT kernel" SVM was compared with the standard linear SVM for the task of face recognition from images. They show that the "FLT" SVM outperforms the standard SVM. On the other hand, they mention than using non-liner SVMs they can further improve performance.

CONCLUSIONS

The report presented an overview of the theory of SVM in parallel with a summary of the papers presented in the ACAI 99 workshop on "Support Vector Machines: theory and applications". Some of the important conclusions of this report as well as of the workshop are summarized below:

(i) SVM are motivated through statistical learning theory. The theory characterizes the performance of learning machines using bounds on their ability to predict future data. One of the papers in the workshop (Vayatis and Azencott, 1999) presented new bounds on the performance of learning machines, and suggested a method to use them experimentally in order to better understand the learning machines (including SVM).

(ii) SVM are trained by solving a constrained quadratic optimization problem. Among others, this implies that there is a unique optimal solution for each choice of the SVM parameters. This is unlike other learning machines, such as standard Neural Networks trained using backpropagation.

(iii) Primal dual interior point optimization methods may be used to efficiently train SVM with large data sets, as described in (Trafalis, 1999).

(iv) Training many local SVMs instead of a single global one can lead to significant improvement in the performance of a learning machine, as shown in (Fernandez, 1999).

(v) SVM has been successfully used for medical diagnosis (Veropoulos et al., 1999). Methods for dealing with unbalanced training data, or for biasing the performance of an SVM towards one of the classes during classification were suggested and used in (Veropoulos et al., 1999).

(vi) An SVM using a kernel motivated from Fisher Linear Discriminant was shown to outperform the standard linear SVM for a face recognition task in (Tefas et al., 1999).

The ideas presented in the papers and discussed in the workshop suggest a number of future research directions: from tuning the basic statistical learning theory results, to developing efficient training methods for SVM, to designing variations of the standard SVM for practical usage. Some of the main issues regarding the design and use of SVMs are, among others, the choice of the kernel of the SVM (as (Tefas et al., 1999) showed), and the choice of the regularization parameter (as (Veropoulos et al., 1999) discussed). On the other hand, significant improvements in the performance of SVM may be achieved if ensembles of SVMs are used (like in (Fernandez, 1999))

Acknowledgements: The authors would like to thank the organizers of ACAI 99, and the co-organizers of the workshop: Constantine Papageorgiou, Tomaso Poggio, and Ioannis Pitas.

REFERENCES

Bartlett P. and Shawe-Taylor J., “Generalization performance of support vector machine and other pattern classifiers”, In C.~Burges B.~Scholkopf, editor, “Advances in Kernel Methods--Support Vector Learning”. MIT press, 1998.

Bottou L. and Vapnik V., “Local learning algorithms”, Neural Computation, 4(6): 888--900, November 1992.

Burges C., “A tutorial on support vector machines for pattern recognition”, In “Data Mining and Knowledge Discovery”. Kluwer Academic Publishers, Boston, 1998, (Volume 2).

Cortes C. and Vapnik V., “Support vector networks”, Machine Learning, 20:1--25, 1995.

Duda R. and Hart P., "Pattern Classification and Scene Analysis", Wiley, New York 1973.

Evgeniou T., Pontil M., and Poggio T., “A unified framework for regularization networks and support vector machines” A.I. Memo No. 1654, Artificial Intelligence Laboratory, MIT, 1999.

Fernandez R., “Predicting time series with a local support vector regression machine", ACAI99.

Osuna E., Freund R., and Girosi F., “Support Vector Machines: Training and Applications”, A.I. Memo No. 1602, Artificial Intelligence Laboratory, MIT, 1997.

Platt J., “Fast training of Support Vector Machines using sequential minimal optimization”, In C.~Burges B.~Scholkopf, editor, “Advances in Kernel Methods--Support Vector Learning”. MIT press, 1998.

Pontil M., Mukherjee S., and Girosi F., “On the noise model of Support Vector Machine regression” A.I. Memo, MIT Artificial Intelligence Laboratory, 1998.

Tefas A., Kotropoulos C., and Pitas I., "Enhancing the performance of elastic graph matching for face authentications by using Support Vector Machines", ACAI99.

Trafalis T., "Primal-dual optimization methods in neural networks and support vector machines training", ACAI99.

Vapnik V., ”Statistical Learning Theory”, Wiley, New York, 1998.

Vapnik V. and Chervonenkis A., “On the uniform convergence of relative frequencies of events to their probabilities”, in ”Th. Prob. and its Applications”, 17(2): 264--280, 1971.

Vayatis N. and Azencott R., "How to estimate the Vapnik-Chervonenkis Dimension of Support Vector Machines through simulations", ACAI99.

Veropoulos K., Cristianini N., and Campbell C., "The Application of Support Vector Machines to Medical Decision Support: A Case Study", ACAI99.

Wahba G., “Splines Models for Observational Data”, Series in Applied Mathematics, Vol. 59, SIAM.