Density Estimation using Support Vector Machines

J. Weston, A. Gammerman, M. Stitson, V. Vapnik, V. Vovk, C. Watkins.

Technical Report

CSD-TR-97-23

February 5, 1998

Department of Computer Science
Egham, Surrey TW20 0EX, England

1 Introduction

In this report we describe how the Support Vector (SV) technique of solving linear operator equations can be applied to the problem of density estimation [4]. We present a new optimization procedure and set of kernels closely related to current SV techniques that guarantee the monotonicity of the approximation. This technique estimates densities with a mixture of bumps (Gaussian-like shapes), with the usual SV property that only some coefficients are non-zero. Both the width and the height of each bump is chosen adaptively, by considering a dictionary of several kernel functions. There is empirical evidence that the regularization parameter gives good control of the approximation, and that the choice of this parameter is universal with respect to the number of sample points, and can even be fixed with good results.

2 The density estimation problem

We wish to approximate the density function $p(x)$ where

$$F(x) = P(X \leq x) = \int_{-\infty}^{x} p(t)dt. \tag{1}$$

We consider densities in the interval [0,1]. Finding the required density means solving the linear operator equation [1]

$$\int_{0}^{1} \theta(x-t)p(t)dt = F(x) \tag{2}$$

where instead of knowing the distribution function $F(x)$ we are given the iid (independently and identically distributed) data

$$x_1, \ldots, x_\ell. \tag{3}$$

We consider the multi-dimensional case where the data $x_1, \ldots, x_\ell$ are vectors of dimension d.

The problem of density estimation is known to be ill-posed. That is, when finding $f \in \mathcal{F}$ that satisfies the equality $Af = F, f$ can have large deviations in solution with small deviations in the right hand side. In our terms, a small change in the cumulative distribution function of the continuous random variable x can cause large changes in the derivative, the density function. One can choose regularization techniques to obtain a sequence of solutions that converge to the desired one.

Using the data (3) we construct the empirical distribution function

$$F_\ell(x) = \frac{1}{\ell}\sum_{i=1}^{\ell} \theta(x^1 - x_i^1)\ldots\theta(x^d - x_i^d)$$

where $x = (x^1, \ldots, x^d)$ instead of the right hand side of (2) which is unknown. We use the SV method to solve the regression problem of approximating the right hand side, using the data

$$(x_1, F_\ell(x_1)), \ldots, (x_l, F_l(x_\ell)).$$

Applying the SV method of solving linear operator equations [3], the parameters of the regression function can then be used to express the corresponding density.

[1]
$$\theta(x) = \begin{cases} 1, & x > 0 \\ 0, & \text{otherwise} \end{cases}$$

The advantage of this approach is that we can control regularization through the free parameter ϵ in the SVM approach. For any point x_i the random value $F_l(x_i)$ is unbiased and has the standard deviation

$$\sigma_i = \sqrt{\frac{1}{\ell}F(x_i)(1-F(x_i))}$$

so we characterize our approximation with

$$\epsilon_i = \lambda\sigma_i = \lambda\sqrt{\frac{1}{\ell}F_\ell(x_i)(1-F_\ell(x_i))}$$

where λ is usually chosen to be $1+\delta$, where δ is some small value.

Therefore one constructs triples

$$(x_1, F_\ell(x_1), \epsilon_1), \ldots, (x_l, F_l(x_\ell), \epsilon_\ell). \tag{4}$$

3 SV Method for Solving Linear Operator Equations

To solve the density estimation problem we use the SV method for solving linear operator equations

$$Af(t) = F(x) \tag{5}$$

where operator A is a one to one mapping between two Hilbert spaces. We solve a regression problem in image space (the right hand side of the equation) and this solution which is an expansion on the support vectors can be used to describe the solution in pre-image space (the left hand side of the equation before the operator A has been applied). The method is as follows: choose a set of functions to solve the problem in pre-image space $f(t,w) \in \Phi$ that are linear in some flattening space:

$$f(t,w) = \sum_{r=0}^{\infty} w_r\phi_r(t) = (W * \Phi(t)). \tag{6}$$

That is, the set of functions is a linear combination of the functions

$$\Phi(t) = (\phi_0(t), \ldots, \phi_N(t), \ldots) \tag{7}$$

which can be thought of as a hyperplane in some flattening space, where the linear combination W that defines the parameters of the set of functions can also be viewed as the coefficients to the hyperplane

$$W = (w_1, \ldots, w_N, \ldots). \tag{8}$$

The mapping from pre-image to image space by the operator A can then be expressed as a linear combination of functions in another Hilbert space defined thus:

$$F(x,w) = Af(x,w) = \sum_{r=0}^{\infty} w_r\psi_r(x) = (W * \Psi(x)) \tag{9}$$

where ψ_r is the r^{th} function from our set of functions after the linear operator A has been applied, i.e $\psi_r(x) = A\phi_r(x)$.

The problem of finding the required density (finding the vector W in pre-image space) is equivalent to finding the vector of coefficients W in image space, where W is an expansion on the support vectors

$$w = \sum_{i=1}^{N} \beta_i\Psi(x_i),$$

giving the approximation to the desired density

$$f(t,w) = \sum_{i=1}^{N} \beta_i \Psi(x_i)\Phi(t).$$

To find the required density we solve a linear regression problem in image space by minimizing the same functional we used to solve standard regression problems ([2, 3]). Instead of directly finding the infinite dimensional vector W which is equivalent to finding the parameters which describe the density function, we use kernel functions to describe the mapping from input space to the image and pre-image Hilbert spaces.

In image space we use the kernel

$$K(x_i, x_j) = \sum_{r=0}^{\infty} \psi_r(x_i)\psi_r(x_j)$$

to define an inner product defined by the set of functions Ψ. We solve the corresponding regression problem in image space, using the coefficients to define the density function in pre-image space:

$$f(t, \alpha, \alpha^*) = \sum_{i=1}^{N} \beta_i \mathcal{K}(x_i, t)$$

where

$$\mathcal{K}(x_i, t) = \sum_{r=0}^{\infty} \psi_r(x_i)\phi_r(x_j). \tag{10}$$

4 Spline Approximation of a Density

We can look for the solution to equation (2) in any set of functions where one can construct a corresponding kernel and cross kernel. For example, consider the set of constant splines with infinite number of nodes. That is we approximate the unknown density by the function:

$$\begin{aligned} p(x) &= \int_0^1 g(\tau)[\int_0^x \theta(t-\tau)dt]d\tau + a_0 x \\ &= \int_0^1 g(\tau)[(x-\tau)_+]d\tau + a_0 x \end{aligned} \tag{11}$$

where function $g(\tau)$ and parameter a_0 are to be estimated.

So the corresponding kernel is

$$K(x_i, x_j) = \int_0^x (x_i - \tau)_+ (x_j - \tau)_+ d\tau + x_i x_j$$

$$= (x_i \wedge x_j)^2 (x_i \vee x_j) - \frac{1}{2}(x_i \wedge x_j)^3 - \frac{1}{2}(x_i \wedge x_j)^2 (x_i \vee x_j) + \frac{1}{3}(x_i \wedge x_j)^3 + x_i x_j \tag{12}$$

and the corresponding cross kernel is

$$\begin{aligned} \mathcal{K}(x,t) &= \int_0^1 \theta(t-\tau)(x-\tau)_+ d\tau + x \\ &= x(x \wedge t) - \frac{(x \wedge t)^2}{2} + x \end{aligned}$$

Using kernel (12) and triples (4) we obtain the support vector coefficients $\beta_i = \alpha_i^* - \alpha_i$, only some of which are non-zero, from the standard SV regression approximation with generalized ϵ-insensitive loss function, by maximizing the quadratic form

$$W(\alpha^*, \alpha) = -\sum_{i=1}^{\ell} \epsilon_i(\alpha_i^* + \alpha_i) + \sum_{i=1}^{\ell} y_i(\alpha_i^* - \alpha_i) - \frac{1}{2}\sum_{i,j=1}^{\ell} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(x_i, x_j)$$

subject to the constraints

$$\sum_{i=1}^{\ell} \alpha_i^* = \sum_{i=1}^{\ell} \alpha_i$$
$$0 \leq \alpha_i^* \leq C, \quad i = 1, \ldots, \ell$$
$$0 \leq \alpha_i \leq C, \quad i = 1, \ldots, \ell. \tag{13}$$

These coefficients define the approximation to the density

$$f(t) = \sum_{i=1}^{N} \beta_i^0 \mathcal{K}(x_i^0, t)$$

where x_i^0 are the support vectors with corresponding non zero coefficients β_i^0.

5 Considering a Monotonic Set of Functions

Unfortunately, the described technique does not guarantee the chosen density will always be positive (recall that a probability is always nonnegative, and the distribution function monotonically increases). This is because the set of functions $F(x, w) \in \Phi$ from which we choose our regression in image space can contain non-monotonic functions.

We can choose a set of monotonic regression functions and require that the coefficients β_i, $i = 1, \ldots, \ell$ are all positive. However, many sets of monotonic functions expressed with Mercer Kernels are too weak in their expressive power to find the desired regression - for example if we choose from the set of polynomials with only positive coefficients.

A set of linear constraints can be introduced into the optimization problem to guarantee monotonicity. This becomes computationally unacceptable in the multi-dimensional case, and in our experiments did not give good results even in the one dimensional case.

We require a set of functions that has high VC dimension but is guaranteed to approximate the distribution function with a monotonically increasing function.

6 Another Approach to SV Regression Estimation

In the SV approach, regression estimation problems are solved as a quadratic optimization problem (13), giving the approximation

$$F(x) = \sum_{i=1}^{N} \beta_i^0 K(x_i^0, x).$$

If we choose to solve this problem directly, (to choose a function from this set of functions) without the regularizing term $(w * w)$ (minimizing the norm of coefficients), we are only required to solve a Linear Programming (LP) problem

[4]. In this alternative approach we can choose our regularizing term as the sum of the support vector weights. This is justified by bounds obtained in the problem of Pattern Recognition that probability of test error is less than the minimum of three terms:

1. **A function of the number of free parameters** — this can be billions, and although is often used as a regularizer in classical theory, is in fact ignored in the SV method.
2. **A function of the size of the margin** — this is the justification for the regularizer used in the usual SV approach (maximizing the margin).
3. **A function of the number of support Vectors** — the justification for the new approach.

So, to solve regression problems we can minimize the following:

$$\sum_{i=1}^{\ell} \alpha_i + \sum_{i=1}^{\ell} \alpha_i^* + C \sum_{i=1}^{\ell} \xi_i + C \sum_{i=1}^{\ell} \xi_i^* \tag{14}$$

under constraints

$$y_i - \epsilon - \xi_i \leq (\sum_{j=1}^{\ell} (\alpha_i^* - \alpha_j) K(x_i, x_j)) + b \leq y_i + \epsilon + \xi_i^*, \quad i = 1, \ldots, \ell \tag{15}$$

$$\alpha_i \geq 0, \quad \xi_i \geq 0, \quad \alpha_i^* \geq 0, \quad \xi_i^* \geq 0, \quad i = 1, \ldots, \ell. \tag{16}$$

This regularizing term can also be seen as a measure in some sense of smoothness in input space; a small number of support vectors will mean a less complex decision function. So minimizing the sum of coefficients can be seen as an approximation of minimizing the number of support vectors.

7 Linear Programming Approach to SV Density Estimation

There exist powerful basis functions that will guarantee the monotonicity of the regression but are non-symmetrical, thus violating Mercer's condition of describing an inner product in some Hilbert space. If we do not describe an inner product, we cannot use the quadratic programming approach to the support vector machine, which minimizes the norm of coefficients of a hyperplane in feature space.

Using the linear programming approach to the Support Vector Machine we do not have this restriction. In fact in this approach $K(x, y)$ can be any function from $L_1(P)$. To estimate a density we choose a monotonic basis function $K(x, y)$ which has a corresponding cross kernel $\mathcal{K}(x, t)$, and then minimize

$$\sum_{i=1}^{\ell} \alpha_i + C \sum_{i=1}^{\ell} \xi_i + C \sum_{i=1}^{\ell} \xi_i^* \tag{17}$$

under constraints

$$y_i - \epsilon_i - \xi_i \leq \sum_{j=1}^{\ell} \alpha_j K(x_i, x_j) \leq y_i + \epsilon_i + \xi_i^*, \quad i = 1, \ldots, \ell \tag{18}$$

$$\sum_{i=1}^{\ell} \alpha_i K(x_i, 0) = 0 \tag{19}$$

$$\sum_{i=1}^{\ell} \alpha_i K(x_i, 1) = 1 \tag{20}$$

$$\alpha_i \geq 0, \quad \xi_i \geq 0, \quad \xi_i^* \geq 0, \quad i = 1, \ldots, \ell. \tag{21}$$

This differs from the usual SV LP Regression (Section 6) in the following ways:

1. We require only positive coefficients, so we only have one set of Lagrange multipliers.
2. We require the constraints (19) and (20) to guarantee $F(0) = 0$ and $F(1) = 1$.
3. We no longer need a threshold b for our hyperplane because we require that $F(0) = 0$, and $F(1) = 1$, and choose kernels that can satisfy these conditions.
4. We allow each vector x_i, $i = 1, \ldots, \ell$ to have a unique ϵ-insensitivity so we can find the corresponding support vectors for our triples (4).

To show that there exist powerful linear combinations of monotonic non-symmetrical kernels, consider the following basis function, the so-called step kernel

$$K(x, y) = \prod_{i=1}^{d} \theta(x_i - y_i).$$

For each support vector (basis function) this gives the effect of a step around the point of the vector in input space (we can also think of this as constant splines with a node at each vector). The coefficient or weight of the vector can make the step arbitrarily large. This kernel can approximate any distribution and is known to converge to the desired one as the number of training examples increases. However, the non-smoothness of the function renders it useless for estimating densities.

8 Gaussian-like Approximation of a Density

We would like to approximate the unknown density from a mixture of bumps (Gaussian-like shapes). This means approximating the regression in image space with a mixture of sigmoidal functions. Consider sigmoids of the form:

$$K(x, y) = \frac{1}{1 + e^{\gamma(x-y)}}$$

In particular, if $\gamma = \frac{2\pi}{\eta}$ then η is a good approximation to the distance from the support vector at which the sigmoid approaches flatness.

The approximation of the density then is:

$$f(x) = \sum_{i=1}^{N} \alpha_i \mathcal{K}(x_i, x)$$

where

$$\mathcal{K}(x, y) = \frac{\gamma}{2 + e^{\gamma(x-y)} + e^{-\gamma(x-y)}}. \tag{22}$$

The distribution function is approximated with a linear combination of sigmoidal shapes which can estimate well the desired regression. The derivative of the regression function (the approximation of the density) is a mixture of Gaussian-like shapes. The chosen centres for the bumps are defined by the support vectors, and their heights by the size of their corresponding weights.

9 Constant Spline Approximation of a Density

We can also estimate our density using a mixture of uniform densities. This means approximating the regression in image space with a mixture of linear pieces, using a kernel of the form:

$$K(x,y) = \begin{cases} 0, & (x-y) < -\gamma \\ \frac{(x-y)+\gamma}{2\gamma}, & -\gamma \leq (x-y) \leq \gamma \\ 1, & (x-y) > \gamma \end{cases}$$

where γ is the width parameter.

The approximation of the density is then:

$$f(x) = \sum_{i=1}^{N} \alpha_i \mathcal{K}(x_i, x)$$

where

$$\mathcal{K}(x,y) = \begin{cases} 0, & (x-y) < -\gamma \\ 1, & -\gamma \leq (x-y) \leq \gamma \\ 0, & (x-y) > \gamma \end{cases}$$

10 Adaptive kernel width

This technique does not allow us to choose the width of each piece of our density function, but only its height and centre. The width of the pieces is decided by the free parameter γ, but all of the widths have to be the same. We would like to remove this free parameter, and allow our SV technique to choose these widths adaptively. This can be achieved if for each centre x_i we have a dictionary of k kernels, giving the approximation to the density:

$$f(x) = \sum_{i=1}^{\ell} (\alpha_i^1 \mathcal{K}_1(x_i, x) + \alpha_i^2 \mathcal{K}_2(x_i, x) + \ldots + \alpha_i^k \mathcal{K}_k(x_i, x))$$

where each vector x_i,$i = 1, \ldots, \ell$ has coefficients α_i^j, $j = 1, \ldots, k$. We then have a corresponding dictionary of k cross kernels, where $\mathcal{K}_i$ has the width γ_i, a chosen dictionary of widths, for example

$$\gamma_1 = \frac{1}{2}, \quad \gamma_2 = \frac{1}{3} \ldots.$$

As usual, many of these coefficients will be zero. This technique allows us to remove the free parameter γ, and also tends to reduce the number of Support Vectors required to approximate a function because of the power of the dictionary.

We can thus generalize the Linear Programming SV Regression technique (section 7) to the following optimization problem: minimize

$$\sum_{i=1}^{\ell} \sum_{n=1}^{k} \alpha_i^n + C \sum_{i=1}^{\ell} \xi_i + C \sum_{i=1}^{\ell} \xi_i^* \tag{23}$$

with constraints

$$y_i - \epsilon_i - \xi_i \leq \sum_{j=1}^{\ell} \sum_{n=1}^{k} \alpha_j^n K_n(x_i, x_j) \leq y_i + \epsilon_i + \xi_i^*,$$

$$\sum_{i=1}^{\ell}\sum_{n=1}^{k}\alpha_i^n K_n(x_i,0) = 0$$

$$\sum_{i=1}^{\ell}\sum_{n=1}^{k}\alpha_i^n K_n(x_i,1) = 1$$

$$\alpha_i \geq 0, \quad \xi_i \geq 0, \quad \xi_i^* \geq 0, \qquad i = 1, \ldots, \ell.$$

11 Choice of regularizer

When considering density estimates as mixtures of kernel functions, where all the coefficients are positive, the sum of coefficients must sum to 1 (if the kernel chosen has been scaled appropriately) to ensure that the integral of the estimated density is also 1. This can be easily translated into a linear constraint which can replace the boundary conditions $F(0) = 0$ and $F(1) = 1$ to give a density estimate without finite support (this is more suitable for estimates using a mixture of Gaussian shapes because of the tails of the distribution.) However, using the linear regularizer described in the optimization problem (23) is not a good approximation to minimizing the number of support vectors because of the coefficients summing to 1. This means we must choose another regularizer. This gives us the more general optimization problem:

$$\Omega(\alpha) + C\sum_{i=1}^{\ell}\xi_i + C\sum_{i=1}^{\ell}\xi_i^* \tag{24}$$

with constraints

$$y_i - \epsilon_i - \xi_i \leq \sum_{j=1}^{\ell}\sum_{n=1}^{k}\alpha_j^n K_n(x_i,x_j) \leq y_i + \epsilon_i + \xi_i^*,$$

$$\sum_{i=1}^{\ell}\sum_{n=1}^{k}\alpha_i^n = 1$$

$$\alpha_i \geq 0, \quad \xi_i \geq 0, \quad \xi_i^* \geq 0, \qquad i = 1, \ldots, \ell.$$

There are many choices of $\Omega(\alpha)$, for example we could try a more accurate minimization of the number of support vectors by approximating the θ function (because it has no useful gradients) with a sigmoid function:

$$\Omega(\alpha) = \sum_{i=1}^{\ell}\sum_{n=1}^{k}\frac{1}{1+\exp(-\lambda\alpha_i^n + 6)}.$$

However, this leads to a nonlinear, non-convex optimization problem.

In the case where one only has a single kernel function, if the cross kernel (the kernel that describes the density estimate rather than the estimate of the distribution function) satisfies Mercer's condition and describes a hyperplane in some feature space, then one can choose the regularizer

$$\Omega(\alpha) = \sum_{i,j=1}^{\ell}\alpha_i\alpha_j\mathcal{K}(x_i,x_j)$$

which is equivalent to minimizing $W \cdot W$, the norm of the vector of coefficients describing the hyperplane in feature space. This is interesting because we can use the same regularizer as the usual support vector pattern recognition and regression estimation cases, because we are regularizing the derivative of our function, even though we are using a kernel to solve our regression problem that does not satisfy Mercer's condition. So we can regularize our density estimate if we use the radial basis (Gaussian-like) kernel (22).

However, when considering a dictionary of kernels one must consider how the different kernels are combined in the regularizer, i.e they describe a set of hyperplanes in different Hilbert spaces.

In our experiments we chose a much simpler regularizer, a weighted sum of the support vector coefficients:

$$\Omega(\alpha) = \sum_{i=1}^{\ell} \sum_{n=1}^{k} w_n \alpha_i^n$$

where w_n is chosen to penalize kernels of small width ($w_n = n$ if the k kernels are in order, smallest width first). This gives a linear optimization problem, however the quality of the solution can probably be improved by choosing other regularizers.

12 Approximating a mixture of normal distributions

We considered a density generated from a mixture of two normal distributions

$$p(x, \mu, \sigma) = \frac{1}{2\sigma\sqrt{(2\pi)}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} + \frac{1}{2\sqrt{(2\pi)}} \exp\left\{-\frac{x^2}{2}\right\} \tag{25}$$

where $\mu = -4$, $\sigma = \frac{1}{2}$. We drew 50 , 100 and 200 examples generated by this distribution, and estimated the density using our technique. Here, $\lambda = 1$, and a dictionary of four kernel widths was used - $\gamma_1 = 0.1, \gamma_2 = 0.2, \gamma_3 = 0.3$ and $\gamma_4 = 0.5$. The results are shown in figure 1.

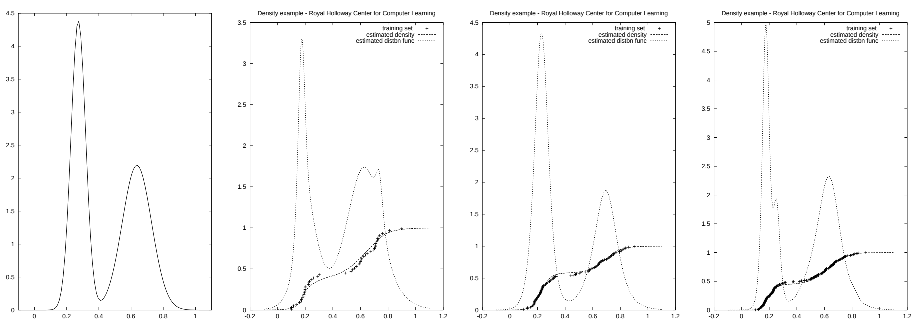

Figure 1 The real density (far left) scaled to the interval [0,1] is estimated using from left to right 50, 100 and 200 data points. The number of support vectors required were 9,10 and 13 respectively.

13 Conclusions and Further Research

We have described a new SV technique for estimating multi-dimensional densities. The estimation can be a mixture of bumps where the height, width and centres are chosen by the training algorithm. We can also use other sets of functions desscribed by other kernels, for example constant splines. As is usual for SV techniques, the description of the approximation can be short as it depends on the number of support vectors.

The control of the regularization in this method appears to be independent of the number of examples. Fixing λ to be greater than the estimated variance (say $\lambda = 1$) and C to a high value (close to infinity), then the technique becomes a parameterless estimator of densities.

The key idea of our method is the following: find the combination of weights that describe the smoothest function (according to some measure of smoothness which we choose) that lies inside the epsilon tube of the training data. Hypothetically we could choose to have infinitely many widths, giving us an infinite set of kernel functions at each point in our training set. Even with these infinite number of variables our optimization problem still has a finite (probably very small) number of coefficients because it will not require many bumps to describe the smoothest function inside the epsilon tube. This has a great advantage over other kernel methods, for example, the Parzen windows method. The Parzen method is a fixed width estimator (the width is the free parameter you must choose appropriately) and the number of coefficients is equal to the number of training points. Thus our method has two advantages over the Parzen method: we remove the free parameter of width and we introduce sparsity into our decision function.

The idea of dictionaries of kernel functions may also be useful in other problems (pattern recognition, regression estimation). Dictionaries of different sets of functions could also be considered, and not just different parameters of the same kernel type.

Although the method looks promising, when the sample size is small (say less than 50 points) the estimates could be improved. To estimate well one must choose a good measure of smoothness. In the current implementation this method is rather poor - a weighted sum of coefficients is minimized (this is not a good idea because the sum of coefficients must sum to 1.) This is a rather *ad hoc* method, and there are many other regularizers we could choose. For example, as the density estimate describes a hyperplane in feature space (it is a radial basis kernel) we could minimize the norm of the vector of coefficients describing this hyperplane, as in the usual support vector case.

References

[1] Cortes, C.; and Vapnik, V. 1995. Support Vector Networks. *Machine Learning* 20:273-297.

[2] Vapnik, V. N. *The Nature of Statistical Learning Theory.* Springer-Verlag, New York, 1995.

[3] Vapnik, V.; Golowich, S.; Smola, A. 1997. Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing. In: M. Mozer, M. Jordan, and T. Petsche (eds.): *Neural Information Processing Systems, Vol. 9.* MIT Press, Cambridge, MA, 1997 (in press).

[4] Vapnik, V. N. *Statistical Learning Theory.* J. Wiley, 1998 (forthcoming).