PERGAMON

Neural Networks 11 (1998) 637–649

Neural
Networks

Contributed article

The connection between regularization operators and support vector kernels

Alex J. Smola*, Bernhard Schölkopf, Klaus-Robert Müller

GMD First, Rudower Chaussee 5, 12489 Berlin, Germany

Received 6 August 1997; accepted 22 December 1997

Abstract

In this paper a correspondence is derived between regularization operators used in regularization networks and support vector kernels. We prove that the Green's Functions associated with regularization operators are suitable support vector kernels with equivalent regularization properties. Moreover, the paper provides an analysis of currently used support vector kernels in the view of regularization theory and corresponding operators associated with the classes of both polynomial kernels and translation invariant kernels. The latter are also analyzed on periodical domains. As a by-product we show that a large number of radial basis functions, namely conditionally positive definite functions, may be used as support vector kernels. © 1998 Elsevier Science Ltd. All rights reserved.

Keywords: Support vector machines; Mercer kernel; Regularization networks; Ridge regression; Green's functions; Conditionally positive definite functions; Polynomial kernels; Radial basis functions

Nomenclature

$\mathbb{R}$ = set of real numbers
$\mathbb{C}$ = set of complex numbers
$\mathbb{N}$ = set of integers
x = (lowercase latin) scalars
$\mathbf{x}$ = (boldface) elements of $\mathbb{R}^n$
α_i, α_i^*, β_i, β_i^* = Lagrange multipliers and expansion coefficients
$\langle\cdot\cdot\rangle$ = dot product in Hilbert space
$\|.\|$ = norm, induced by a dot product
f = functions
$\bar{f}$ = complex conjugate (of a function or a scalar)
$\tilde{f}$ = Fourier transform of f
$\mathcal{F}$ = feature space
Φ,$\Phi(\mathbf{x})$ = elements and mappings into F
$\hat{P}$ = operators
D, D_{ij} = matrices, matrix elements
$\delta_{x_0}(\mathbf{x})$, δ_{ij} = delta distribution, Kronecker delta
(λ_i, ϕ_i), (Λ_i, Ψ_i) = (eigenvector, eigenvalue) pairs
$\prod_{i=1}^{n}$ = product
$\bigotimes_{i=1}^{n}$ = convolution
$\sum_{i=1}^{n}$ = summation
1_I = indicator function on a set I
$\mathbf{1}$ = identity map
$\vec{1}$ = vector with all entries equal to 1

1. Introduction

Support vector (SV) machines for pattern recognition, regression estimation, and operator inversion exploit the idea of mapping data into a high dimensional feature space where they perform a linear algorithm. Instead of evaluating this mapping explicitly, one uses integral operator kernels $k(\mathbf{x}, \mathbf{y})$ which correspond to dot products of the mapped data in high dimensional space, Aizerman et al. (1964); Boser et al. (1992), i.e.

$$k(\mathbf{x},\mathbf{y}) = \langle\Phi(x)\cdot\Phi(y)\rangle \tag{1}$$

with Φ: $\mathbb{R}^n \rightarrow \mathcal{F}$ denoting the map into feature space $\mathcal{F}$. Mostly, this map and many of its properties are unknown. Even worse, so far no general rule was available which kernel should be used, or why mapping into a very high dimensional space often provides good results, seemingly defying the curse of dimensionality. In order to clarify this dilemma we show how these kernels $k(\mathbf{x}, \mathbf{y})$ correspond to regularization operators $\hat{P}$, the link being that k is the Green's function of $\hat{P}^*\hat{P}$ (with $\hat{P}^*$ denoting the adjoint operator of $\hat{P}$). In other words — given a support vector

* Corresponding author. Tel: (++49)-30-682-1833; Fax: (++49)-30-682-1805; E-mail: smola@first.gmd.de. URL: http://svm.first.gmd.de.

0893-6080/98/$19.00 © 1998 Elsevier Science Ltd. All rights reserved.
PII S0893-6080(98)00032-X

kernel we show how to find the corresponding regularization operator and vice versa. For the sake of simplicity, we shall limit ourselves to the case of regression — our considerations, however, also hold true for the other cases mentioned earlier.

This paper[1] starts by briefly reviewing the concepts of SV Machines (Section 2) and regularization networks (Section 3). Section 4 contains the main result, the derivation of a correspondence between regularization operators used in regularization networks and SV kernels. In Section 5 applications of this finding to translation invariant kernels for both unbounded and bounded support are presented. Section 6 presents the operators corresponding to polynomial kernels, another frequently used class of SV kernels. Subsequently Section 7 introduces a new class of possible SV kernels which do not necessarily satisfy Mercer's condition, namely kernels derived from conditionally positive definite functions. Section 8 concludes the paper with a discussion. Finally Appendix A contains a worked through example and Appendix B applies the methods presented in this paper to find a connection between ridge regression and SV machines. Due to its specific setting, however, only a less formal exposition is possible.

2. Support vector machines

The SV algorithm for regression estimation, as described in Vapnik (1995); Vapnik et al. (1997), exploits the idea of computing a linear function in high dimensional feature space $\mathcal{F}$ (furnished with a dot product). Thereby this algorithm can compute a nonlinear function in the space of the input data $\mathbb{R}^n$. The functions take the form

$$f(\mathbf{x}) = \langle w \cdot \Phi(\mathbf{x}) \rangle + b \tag{2}$$

with $\Phi:\mathbb{R}^n \to \mathcal{F}$ being the map into feature space and $w \in \mathcal{F}$.

In order to estimate f from a given training set $\{(\mathbf{x}_i, y_i) | i = 1,\ldots,\ell,\ \mathbf{x}_i \in \mathbb{R}^n, y_i \in \mathbb{R}\}$, one tries to minimize the regularized risk functional

$$\mathbb{R}_{\text{reg}}[f] = \mathbb{R}_{\text{emp}}[f] + \frac{\lambda}{2}\|w\|^2 = \frac{1}{l}\sum_{i=1}^{\ell} c(f(\mathbf{x}_i), y_i) + \frac{\lambda}{2}\|w\|^2 \tag{3}$$

i.e. the empirical risk functional $\mathbb{R}_{\text{emp}}[f]$ together with a complexity term $\|w\|^2$, thereby enforcing flatness in feature space. Here $c(f(\mathbf{x}_i), y_i)$ is the cost function determining how the distance between $f(\mathbf{x}_i)$ and the target values y_i should be penalized, and $\lambda \in \mathbb{R}^+$ is a regularization constant. The idea of *flatness* is derived from pattern recognition where this corresponds to finding a hyperplane that has maximum distance in $\mathcal{F}$ from the classes to be separated Boser et al. (1992); Cortes and Vapnik (1995). As shown in Vapnik (1995) for the case of ϵ-insensitive cost functions,

$$c(f(\mathbf{x}), y) = \begin{cases} |f(\mathbf{x}) - y| - \epsilon & \text{for } |f(x) - y| \geq \epsilon \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Eq. (3) can be minimized by solving a quadratic programming problem formulated in terms of dot products in $\mathcal{F}$. It turns out that the solution w can be expressed in terms of *support vectors*. Note that the representation can be sparse. Therefore, the points corresponding to nonzero α_i, which suffice for describing *f*, are called *support vectors*.

$$w = \sum_{i=1}^{\ell} \alpha_i \Phi(\mathbf{x}_i). \tag{5}$$

Therefore, via Eq. (1)

$$f(\mathbf{x}) = \langle w \cdot \Phi(\mathbf{x}) \rangle + b = \sum_{i=1}^{\ell} \alpha_i \langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) \rangle + b$$

$$= \sum_{i=1}^{\ell} \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b \tag{6}$$

where $k(\mathbf{x}_i, \mathbf{x})$ is a kernel function computing a dot product in feature space (a concept introduced by Aizerman et al., 1964). The coefficients α_i can be found by solving a quadratic programming problem (with $K_{ij}: = k(\mathbf{x}_i, \mathbf{x}_j)$, $\alpha_i = \beta_i - \beta_i^*$ and β_i, β_i^* being the solution of the optimization problem below):

$$\text{minimize } \frac{1}{2}\sum_{i,j=1}^{\ell} (\beta_i^* - \beta_i)(\beta_j^* - \beta_j)K_{ij}$$

$$- \sum_{i=1}^{\ell} ((\beta_i^* - \beta_i)y_i - (\beta_i^* + \beta_i)\epsilon)$$

$$\text{subject to } \sum_{i=1}^{\ell} (\beta_i - \beta_i^*) = 0,\ \beta_i, \beta_i^* \in \left[0, \frac{1}{\lambda\ell}\right] \tag{7}$$

Note that Eq. (4) is not the only possible choice of cost functions resulting in a quadratic programming problem (many convex cost function, in particular quadratic parts and infinities are admissible, too). For a detailed discussion see Smola and Schölkopf (1997); Smola et al. (1998). Also note that any continuous symmetric function $k(\mathbf{x}, \mathbf{y}) \in L^2 \otimes L^2$ may be used as an admissible kernel if it satisfies a weak form of Mercer's condition (Riesz and Nagy, 1955)

$$\int\int k(\mathbf{x}, \mathbf{y}) g(\mathbf{x}) g(\mathbf{y}) \mathrm{d}x \mathrm{d}y \geq 0 \text{ for all } g \in L^2(\mathbb{R}^n) \tag{8}$$

3. Regularization networks

Here again we start with minimizing the empirical risk functional $R_{\text{emp}}[f]$ plus a regularization term $\|\hat{P}f\|^2$ defined by a regularization operator $\hat{P}$ in the sense of Tikhonov and Arsenin (1977), i.e. $\hat{P}$ is a positive semidefinite operator

[1] Portions of this work have been published in Smola and Schölkopf (1998).

mapping from the Hilbert Space $\mathcal{H}$ of functions f under consideration to a dot product space D such that the expression $\langle \hat{P}f\cdot\hat{P}g\rangle$ is well defined. For instance by choosing a suitable operator that penalizes large variations of f one can reduce the well-known overfitting effect. Another possible setting also might be an operator $\hat{P}$ mapping from $L^2(\mathbb{R}^n)$ into some reproducing kernel Hilbert space (Kimeldorf and Wahba, 1971; Girosi, 1997). In Appendix A, we provide a worked through example (mainly taken from Girosi et al., 1993) for a simple regularization operator to illustrate our reasoning.

Similar to Eq. (3), we minimize

$$R_{\text{reg}}[f] = R_{\text{emp}} + \frac{\lambda}{2}\|\hat{P}f\|^2 = \frac{1}{l}\sum_{i=1}^{\ell} c(f(\mathbf{x}_i), y_i) + \frac{\lambda}{2}\|\hat{P}f\|^2 \quad (9)$$

Using an expansion of f in terms of some symmetric function $k(\mathbf{x}_i,\ \mathbf{x}_j)$ (note here, that k need not fulfill Mercer's condition),

$$f(\mathbf{x}) = \sum_i \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b, \quad (10)$$

and the cost function defined in Eq. (4), this leads to a quadratic programming problem similar to the one for SVs. By computing Wolfe's dual (for details of the calculations see Smola and Schölkopf, 1997), and using

$$D_{ij} := \langle (\hat{P}k)(\mathbf{x}_i, .)\cdot(\hat{P}k)(\mathbf{x}_j, .)\rangle \quad (11)$$

($\langle f\cdot g\rangle$ denotes the dot product of the functions f and g in Hilbert Space, e.g. $\int \bar{f}(\mathbf{x})g(\mathbf{x})\mathrm{d}\mathbf{x}$), we get $\vec{\alpha} = D^{-1}K(\vec{\beta} - \vec{\beta}^*)$, with β_i, β_i^* being the solution of

$$\text{minimize } \frac{1}{2}\sum_{i,j=1}^{\ell}(\beta_i^* - \beta_i)(\beta_j^* - \beta_j)(KD^{-1}K)_{ij}$$

$$-\sum_{i=1}^{\ell}((\beta_i^* - \beta_i)y_i - (\beta_i^* + \beta_i)\epsilon)$$

$$\text{subject to } \sum_{i=1}^{\ell}(\beta_i - \beta_i^*) = 0,\ \beta_i, \beta_i^* \in \left[0, \frac{1}{\lambda l}\right]. \quad (12)$$

Unfortunately, this setting of the problem does not preserve sparsity in terms of the coefficients, as a potentially sparse decomposition in terms of β_i and β_i^* is spoiled by $D^{-1}K$, which in general is not diagonal (Eq. (6), on the other hand, does typically have many vanishing coefficients, see e.g. Schölkopf et al., 1995; Vapnik, 1995).

4. The relation between both methods

Comparing Eq. (7) with Eq. (12) leads to the question if and under which condition the two methods might be equivalent and, therefore, also under which conditions, given a suitable cost function, regularization networks might lead to sparse decompositions (i.e. only a few of the expansion coefficients α_i in f would differ from zero). A sufficient[2] condition is $D = K$ (thus $KD^{-1}K = K$), i.e.

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle (\hat{P}k)(\mathbf{x}_i, .)\cdot(\hat{P}k)(\mathbf{x}_j, .)\rangle \text{ (self consistency).} \quad (13)$$

This is the main equation of this paper. Our goal now is to solve the following two problems:

1. Given a regularization operator $\hat{P}$, find a kernel k such that a SV machine using k will not only enforce flatness in feature space, but also correspond to minimizing a regularized risk functional with $\hat{P}$ as regularization operator;
2. Given a SV kernel k, find a regularization operator $\hat{P}$ such that a SV machine using this kernel can be viewed as a regularization network using $\hat{P}$.

These two problems can be solved by employing the concept of Green's functions as described in Girosi et al. (1993). These functions had been introduced in the context of solving differential equations. For our purpose, it is sufficient to know that the Green's functions $G_{\mathbf{x}_i}(\mathbf{x})$ of $\hat{P}^*\hat{P}$ satisfy

$$(\hat{P}^*\hat{P}G_{\mathbf{x}_i})(\mathbf{x}) = \delta_{\mathbf{x}_i}(\mathbf{x}) \quad (14)$$

Here, $\delta_{\mathbf{x}_i}(\mathbf{x})$ is the δ-distribution (not to be confused with the Kronecker symbol δ_{ij}) which has the property that $\langle f\cdot\delta_{\mathbf{x}_i}\rangle = f(\mathbf{x}_i)$. The relationship between kernels and regularization operators is formalized in the following proposition:

Proposition 1 (Green's functions and Mercer kernels). *Let $\hat{P}$ be a regularization operator, and G be the Green's function of $\hat{P}^*\hat{P}$. Then G is a Mercer kernel such that $D = K$. SV machines using G minimize risk functional Eq. (9) with $\hat{P}$ as regularization operator*.[3]

Proof. Substituting Eq. (14) into $G_{\mathbf{x}_j}(\mathbf{x}_i) = \langle G_{\mathbf{x}_j}(.)\cdot\delta_{\mathbf{x}_i}(.)\rangle$ yields

$$G_{\mathbf{x}_j}(\mathbf{x}_i) = \langle (\hat{P}G_{\mathbf{x}_i})(.)\cdot(\hat{P}G_{\mathbf{x}_j})(.)\rangle \quad (15)$$

hence $G(\mathbf{x}_i,\mathbf{x}_j) := G_{\mathbf{x}_i}(\mathbf{x}_j)$ is symmetric and satisfies Eq. (13). Thus the SV optimization problem Eq. (7) is equivalent to the regularization network counterpart Eq. (12). Furthermore, G is an admissible non-negative kernel, as it can be written as a dot product in Hilbert space, namely

$$G(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}_j)\rangle \text{ with } \Phi : \mathbf{x}_i \mapsto (\hat{P}G_{\mathbf{x}_i})(.). \quad (16)$$

■

A similar result can be obtained by exploiting Mercer's theorem in a more straightforward manner, by using the fact that a Mercer kernel k can be expanded into a convergent

[2] In the case of K not having of full rank D is only required to be the inverse on the image of K. The pseudoinverse for instance is such a matrix.

[3] This condition is sufficient but not necessary for satisfying Eq. (13). Any projection of G onto an invariant subspace of $\hat{P}^*P$ would also satisfy this equation. Note that as $G(.,.)$ being a function on $\mathbb{R}^n \otimes \mathbb{R}^n$ the projection operator has to be applied to it as a function of both the first and the second argument.

series of its eigensystem ($\phi_\ell(\mathbf{x})$,λ_ℓ) with $\lambda_\ell \geq 0$,

$$k(\mathbf{x}_i,\mathbf{x}_j)=\sum_\ell \lambda_\ell \phi_\ell(\mathbf{x}_i)\phi_\ell(\mathbf{x}_j) \tag{17}$$

This is particularly useful for the approximation of periodical functions and will come handy in example 6 as we will have to deal with a discrete eigensystem in this case.

Proposition 2 (a discrete counterpart). *Given a regularization operator $\hat{P}$ with an expansion of $\hat{P}^*\hat{P}$ into a discrete eigensystem (Λ_ℓ, Ψ_ℓ) and a kernel k with*

$$k(\mathbf{x}_i,\mathbf{x}_j)=\sum_\ell \frac{\mathrm{d}_\ell}{\Lambda_\ell}\Psi_\ell(\mathbf{x}_i)\Psi_\ell(\mathbf{x}_j) \tag{18}$$

where $\mathrm{d}_\ell \in \{0,1\}$ *for all* ℓ, *and* $\Sigma_\ell(\mathrm{d}_\ell/\Lambda_\ell)$ *convergent. Then k satisfies Eq. (13).*

Proof. Evaluating Eq. (13) and using orthonormality of the system ($\mathrm{d}_\ell/\Lambda_\ell$, Ψ_ℓ), yields:

$$\begin{aligned}
&\langle k(\mathbf{x}_i,.)\cdot(\hat{P}^*\hat{P}k)(\mathbf{x}_j,.)\rangle \\
&=\left\langle \sum_\ell \frac{\mathrm{d}_\ell}{\Lambda_\ell}\Psi_\ell(\mathbf{x}_i)\Psi_\ell(.)\cdot\hat{P}^*\hat{P}\left(\sum_{\ell'}\frac{\mathrm{d}_{\ell'}}{\Lambda_{\ell'}}\Psi_{\ell'}(\mathbf{x}_j)\Psi_{\ell'}(.)\right)\right\rangle \\
&=\sum_{\ell,\ell'}\frac{\mathrm{d}_\ell}{\Lambda_\ell}\frac{\mathrm{d}_{\ell'}}{\Lambda_{\ell'}}\Psi_\ell(\mathbf{x}_i)\Psi_{\ell'}(\mathbf{x}_j)\langle\Psi_\ell(.)\cdot\hat{P}^*\hat{P}\Psi_{\ell'}(.)\rangle \\
&=\sum_\ell \frac{\mathrm{d}_\ell}{\Lambda_\ell}\Psi_\ell(\mathbf{x}_i)\Psi_\ell(\mathbf{x}_j)=k(\mathbf{x}_i,\mathbf{x}_j)
\end{aligned} \tag{19}$$

■

Rearranging of the summation coefficients is allowed as the eigenfunctions are orthonormal and the series $\Sigma_\ell(\mathrm{d}_\ell/\Lambda_\ell)$ converges. Consequently a large class of kernels can be associated with a given regularization operator (and vice versa) thereby restricting ourselves to some subspace of the eigensystem of $\hat{P}^*\hat{P}$.[4]

Excluding eigenfunctions of $\hat{P}^*\hat{P}$ from the kernel expansion effectively decreases the expressive power of the set of approximating functions, i.e. we limit the capacity of the system of functions. Removing low capacity (i.e. very flat) eigenfunctions from the expansion will have an adverse effect, though, as the data will have to be approximated by the higher capacity functions.

In the following we will exploit this relationship in both ways: to compute Green's functions for a given regularization operator $\hat{P}$ and to infer the regularization operator from a given kernel *k*.

Note that a similar reasoning can be applied to connect ridge regression schemes with support vector kernels as shown in Appendix B.

5. Translation invariant kernels

Let us now more specifically consider regularization operators $\hat{P}$ that may be written as multiplications in Fourier space

$$\langle \hat{P}f\cdot\hat{P}g\rangle=\frac{1}{(2\pi)^{n/2}}\int_\Omega \frac{\overline{\tilde{f}(w)}\tilde{g}(w)}{P(w)}\mathrm{d}w \tag{20}$$

with $\tilde{f}(w)$ denoting the Fourier transform of $f(\mathbf{x})$, and $P(w)=P(-w)$ real valued, nonnegative and converging to 0 for $|w|\to\infty$ and Ω = supp[$P(w)$]. Small values of $P(w)$ correspond to a strong attenuation of the corresponding frequencies. Hence small values of $P(w)$ for large w are desirable since high frequency components of $\tilde{f}$ correspond to rapid changes in *f*. $P(w)$ describes the filter properties of $\hat{P}^*\hat{P}$ — note that no attenuation takes place for $P(w)=0$ as these frequencies have been excluded from the integration domain.

For regularization operators defined in Fourier space by Eq. (20) it can be shown by exploiting $P(w)=P(-w)=\overline{P(w)}$ that

$$G(\mathbf{x}_i,\mathbf{x})=\frac{1}{(2\pi)^{n/2}}\int_{\mathbb{R}^n} \mathrm{e}^{\mathrm{i}w(\mathbf{x}_i-x)}P(w)\mathrm{d}w \tag{21}$$

is a corresponding Green's function satisfying translational invariance, i.e.

$$G(\mathbf{x}_i,\mathbf{x}_j)=G(\mathbf{x}_i-\mathbf{x}_j) \text{ and } \tilde{G}(w)=P(w)$$

For the proof, one only has to show that *G* satisfies Eq. (13). This provides us with an efficient tool for analyzing SV kernels and the types of capacity control they exhibit. In fact the above is a special case of Bochner's theorem (Bochner, 1959) stating that the Fourier transform of a positive measure constitutes a positive Hilbert Schmidt kernel.

Example 3 ($\mathbf{B_q}$-splines). In Vapnik et al. (1997) the use of B_q-splines was proposed (see Fig. 1) as building blocks for kernels, i.e.

$$k(\mathbf{x})=\prod_{i=1}^{n} B_q(\mathbf{x}_i) \tag{22}$$

with $\mathbf{x}\in\mathbb{R}^n$. For the sake of simplicity, we consider the case n = 1. Recalling the definition (up to scaling factors) by Unser et al. (1991)

$$B_q=\otimes^{q+1}1_{[-0.5,0.5]} \tag{23}$$

we can utilize the above result and the Fourier–Plancherel identity to construct the Fourier representation of the corresponding regularization operator. Up to a multiplicative constant, it equals

$$P(w)=\tilde{k}(w)=\mathrm{sinc}^{(q+1)}\left(\frac{w_i}{2}\right) \tag{24}$$

This answers the question why only *B*-splines of odd order are admissible although both even and odd order *B*-splines

[4] The intuition of this reasoning is that there exists a one to one correspondence between kernels and regularization operators only on the image of $\mathcal{H}$ under the integral operator ($\hat{O}$f)(**x**): = ∫ k(**x**, **y**)f(**y**) d**y**, namely that $\hat{O}$ and $\hat{P}^*\hat{P}$ are inverse to another. On the null space of $\hat{O}$, however, the regularization operator $\hat{P}^*\hat{P}$ may take on an arbitrary form. In this case *k* still will fulfill the self consistency condition.

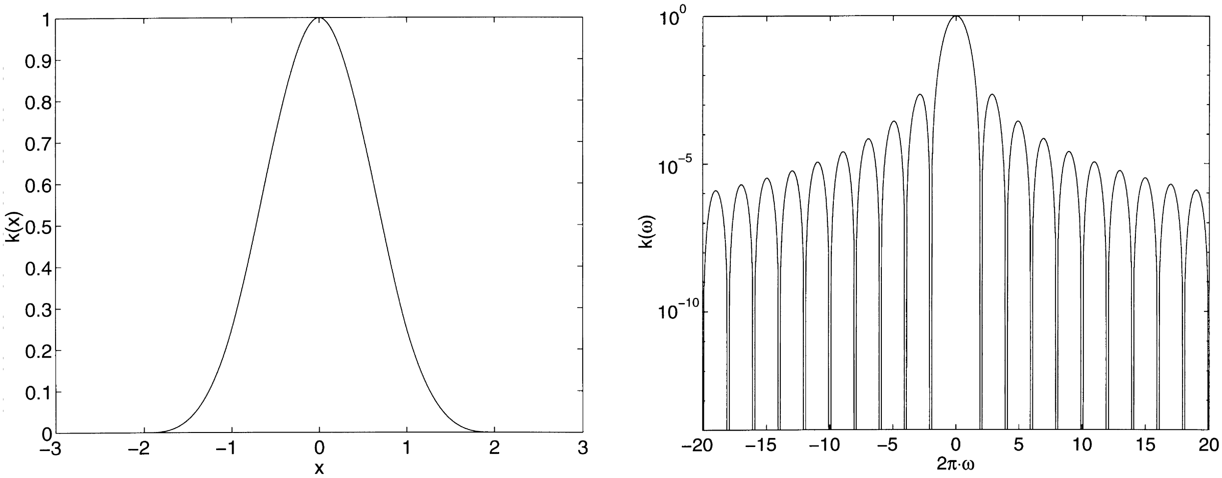

Fig. 1. Left: B_3-spline kernel. Right: Fourier transform of the kernel.

converge to a Gaussian for $q \to \infty$ due to the law of large numbers: The even ones have negative parts in the Fourier spectrum (which would result in an amplification of the corresponding frequency components). The zeros in $\tilde{k}$ stem from the fact that B_q has only compact support $[-(q + 1)/2, (q + 1)/2]$. By using this kernel we trade reduced computational complexity in calculating f (we only have to take points into account with $\|\mathbf{x}_i - \mathbf{x}_j\| \le c$ from some limited neighborhood determined by c) for a possibly worse performance of the regularization operator as it completely removes frequencies w_p with $\tilde{k}(w_p) = 0$.

Example 4 (Gaussian kernels). Following the exposition of Yuille and Grzywacz (1988) as described in Girosi et al. (1993), one can see that for

$$\|\hat{P}f\|^2 = \int d\mathbf{x} \sum_m \frac{\sigma^{2m}}{m!2^m}(\hat{O}^m f(\mathbf{x}))^2 \tag{25}$$

with $\hat{O}^{2m} = \Delta^m$ and $\hat{O}^{2m+1} = \Delta\nabla^m$, Δ being the Laplacian and ∇ the gradient operator, we get Gaussians kernels (see Fig. 2)

$$k(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x}\|^2}{2\sigma^2}\right) \tag{26}$$

Moreover, we can provide an equivalent representation of $\hat{P}$ in terms of its Fourier properties, i.e. $P(w) = \exp[-(\sigma^2\|w\|^2)/2)$ up to a multiplicative constant. Training a SV machine with Gaussian RBF kernels (Schölkopf *et al.*, 1997) corresponds to minimizing the specific cost function with a regularization operator of type Eq. (25).

Recall that Eq. (25) means that all derivatives of f are penalized (we have a pseudodifferential operator) to obtain a very smooth estimate. This also explains the good performance of SV machines in this case, as it is by no means obvious that choosing a flat function in some high dimensional space will correspond to a simple function in low dimensional space, as shown in example 5.

Gaussian kernels tend to yield good performance under general smoothness assumptions and should be considered especially if no additional knowledge of the data is available.

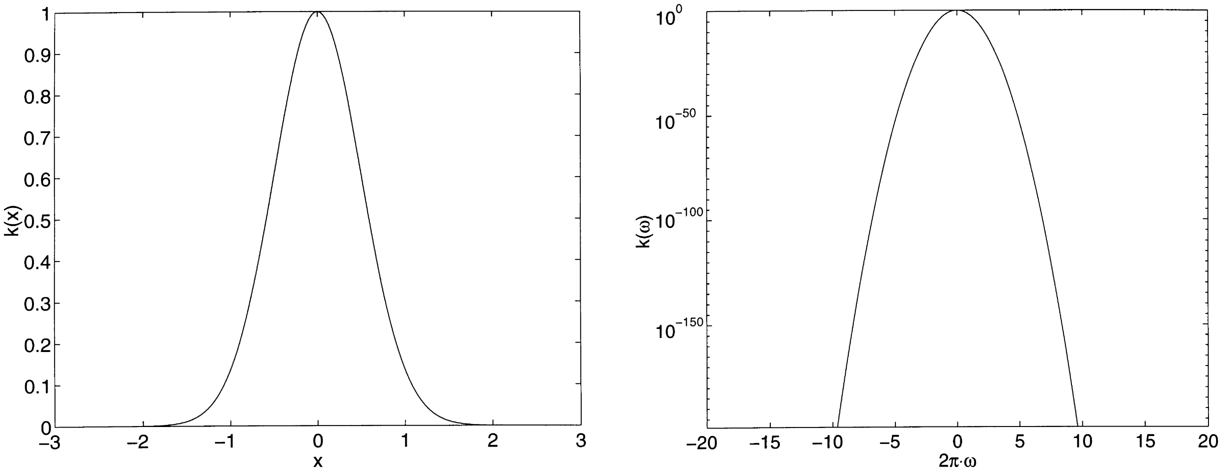

Fig. 2. Left: Gaussian kernel with standard deviation 0.5. Right: Fourier transform of the kernel.

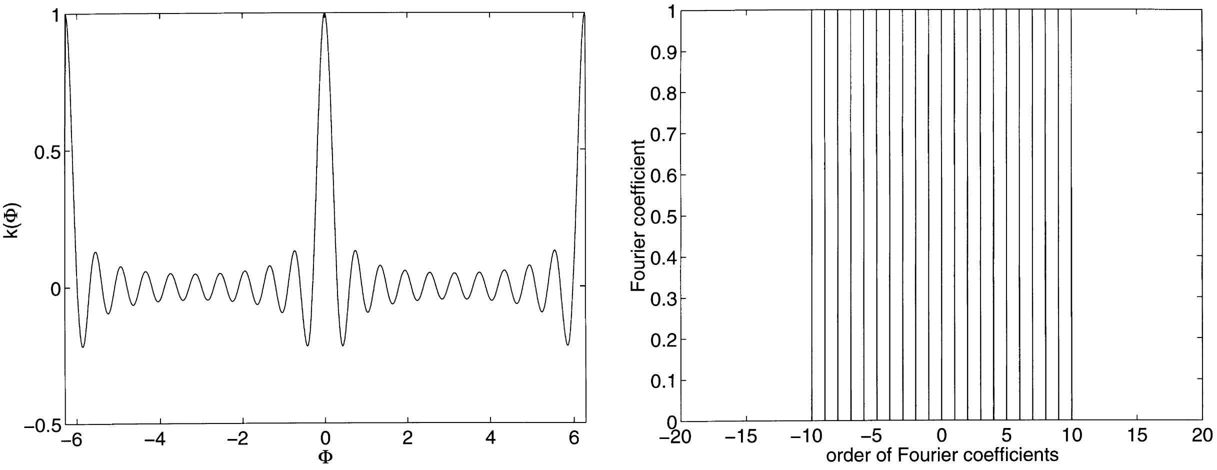

Fig. 3. Left: Dirichlet kernel of order 10. Note that this kernel is periodical. Right: Fourier transform of the kernel.

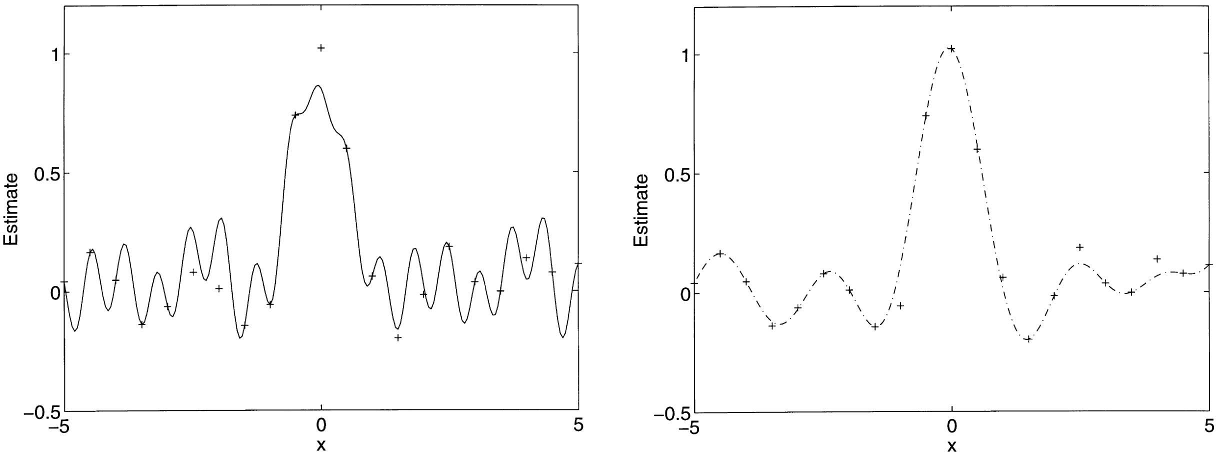

Fig. 4. Left: Regression with a Dirichlet Kernel of order $N = 10$. One can clearly observe the overfitting. Right: regression of the same data with a Gaussian Kernel of width $\sigma^2 = 1$.

Example 5 (Dirichlet kernels). In Vapnik et al. (1997), a class of kernels generating Fourier expansions was introduced for interpolating data on $\mathbb{R}^n$,

$$k(x) = \frac{\sin(2N+1)x/2}{\sin x/2} \tag{27}$$

(As in example 3 consider $\mathbf{x} \in \mathbb{R}^1$ to avoid tedious notation.) By construction, this kernel corresponds to $P(w) = 1/2\sum_{i=-N}^{N}\delta_i(w)$. A regularization operator with these properties, however, may not be desirable as it only damps a finite number of frequencies (cf. Fig. 3) and leaves all other frequencies unchanged which can lead to overfitting (Fig. 4).

In some cases it might be useful to approximate periodical functions, e.g. functions defined on a circle. This leads to the second possible type of translation invariant[5] kernel functions, namely functions defined on factor spaces. Without loss of generality assume the period to be 2π — consequently one gets translation invariance on $\mathbb{R}/2\pi$.

In the following we will show the consequences of this setting for the operator defined in example 4.

Example 6 (periodical Gaussian kernels). Analogously to Eq. (25), define a regularization operator on functions on $[0, 2\pi]^n$ by

$$\|\hat{P}f\|^2 = \pi^{-n}\int_{[0,2\pi]^n} \mathrm{d}\mathbf{x}\sum_m \frac{\sigma^{2m}}{m!2^m}(\hat{O}^m f(\mathbf{x}))^2 \tag{28}$$

with $\hat{O}$ as in example 4. For the sake of simplicity assume $n = 1$. A generalization to multidimensional kernels is straightforward.

It is easy to check that the Fourier basis $\{1/2, \sin(\ell x), \cos(\ell x), \ell \in \mathbb{N}\}$ is an eigensystem of the operator defined above, with eigenvalues $\exp((\ell^2\sigma^2)/2)$. Now apply proposition 2, taking into account all eigenfunctions except $\ell = 0$.

[5] Obviously defining translation invariant kernels on a bounded interval is not a reasonable concept as the data would hit the bounds of the interval when translated by a large amount. Therefore, only unbounded intervals and factor spaces are possible domains.

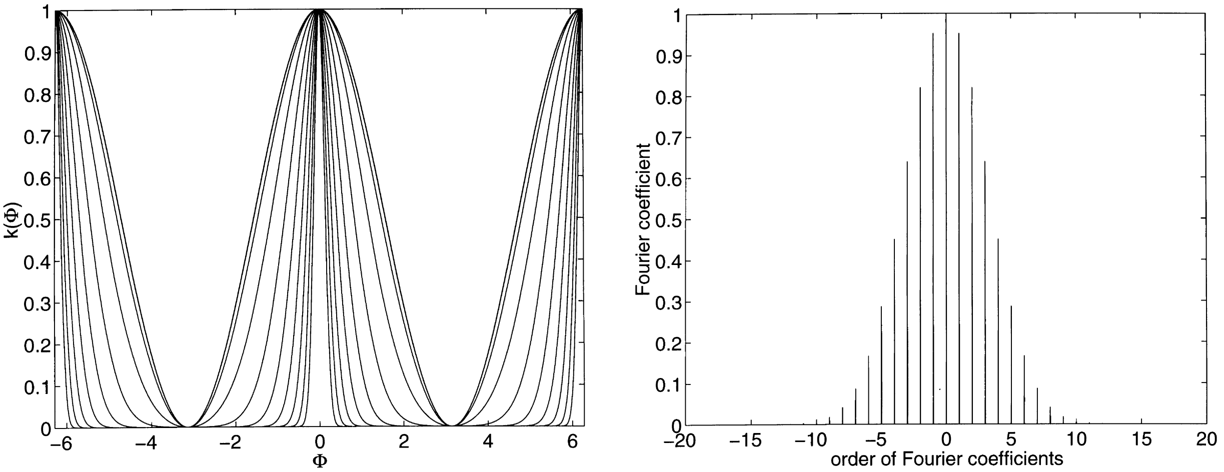

Fig. 5. Left: periodical Gaussian kernel for several values of σ (normalized to 1 as its maximum and 0 as its minimum value). Peaked functions correspond to small σ. Right: Fourier coefficients of the kernel for $\sigma^2 = 0.1$

This yields the following kernel:

$$k(x,x') = \sum_{\ell=1}^{\infty} e^{-\frac{\ell^2\sigma^2}{2}}(\sin(\ell x)\sin(\ell x') + \cos(\ell x)\cos(\ell x'))$$

$$= \sum_{\ell=1}^{\infty} e^{-\frac{\ell^2\sigma^2}{2}}\cos(\ell(x-x')) \tag{29}$$

For practical purposes one may truncate the expansion after a finite number of terms. Moreover we rescale k to have a range of exactly [0, 1] by using the positive offset $\sum_{\ell=1}^{\infty}(-1)^{(l-1)}e^{-((\ell^2\sigma^2)/2)}$ and the scaling factor $1/2\sum_{\ell=1}^{\infty}e^{-(((2\ell-1)^2\sigma^2)/2)}$ (cf. Fig. 5).

In the context of periodical functions, the difference between this kernel and the Dirichlet kernel of example 5 is that the latter does not distinguish between the different frequency components in $w \in \{-\mathrm{N}\pi,\ldots,\mathrm{N}\pi\}$. However, it effectively limits the maximum capacity of the system to approximating the data with a Fourier expansion up to the order N.

The question that arises now is which kernel to choose. Let us think about two extreme situations.

- Suppose we already knew the shape of the power spectrum Pow(w) of the function we would like to estimate. In this case we choose k such that $\tilde{k}$ matches the power spectrum.
- If we happen to know very little about the given data a general smoothness assumption is a reasonable choice. Hence we might want to choose one of the Gaussian kernels in example 4 or 6. If computing time is important one might moreover consider kernels with compact support, e.g. using the B-spline kernels of example 3. This choice will cause many matrix elements $k_{ij} = k(\mathbf{x}_i - \mathbf{x}_j)$ to vanish.

The usual scenario will be in between the two extreme cases and we will have some limited prior knowledge available. For more information on using prior knowledge for choosing kernels see Schölkopf et al. (1998).

Prior knowledge can also be used to determine the free parameters of the kernel, e.g. its width (σ) in the examples 4 and 6. Besides that model selection principles like structural risk minimization (Vapnik, 1982), cross validation (Bishop, 1995; Amari et al., 1997; Kearns, 1997), MDL (Rissanen, 1985), Bayesian methods (MacKay, 1991; Bishop, 1995), etc. can be employed. Choosing a small width of the kernels leads to high generalization error as it effectively decouples the separate basis functions of the kernel expansion into very localized functions which is equivalent to memorizing the data, whereas a wide kernel tends to oversmooth.

Note that the choice of the width may be more important than the actual functional form of the kernel. There may be little difference in the relevant part of the filter properties between e.g. a B-spline and a Gaussian kernel (cf. Fig. 6).

The invariance of the kernels presented so far has been exploited only in the context of invariance with respect to the translation symmetry group in $\mathbb{R}^n$. Yet they could also be applied to other symmetry transformations corresponding to other canonical coordinate systems such as the rotation and scaling group as proposed by Segman et al. (1992); Ferraro and Caelli (1994), i.e. to a logpolar parametrization of $\mathbb{R}^n$ or the parametrization of manifolds.

6. Kernels of dot-product type

There exists a large class of support vector kernels which are not translation invariant, namely kernels of the type

$$k(\mathbf{x},\mathbf{x}') = t(\langle\mathbf{x}\cdot\mathbf{x}'\rangle) \tag{30}$$

For instance, polynomial kernels $(\langle\mathbf{x}\cdot\mathbf{x}'\rangle + c)^p$ of homogeneous ($c = 0$) or inhomogeneous type ($c > 0$) belong to this class. It follows directly from Poggio (1975) that polynomial kernels satisfy Mercer's condition. Now the question

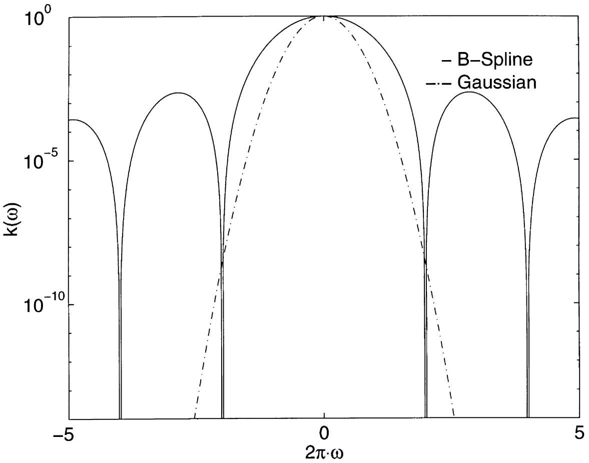

Fig. 6. Comparison of regularization properties in the low frequency domain of B_3-spline kernel and Gaussian kernel ($\sigma^2 = 20$). Up to an attenuation factor of $5 \cdot 10^{-3}$ both types of kernels exhibit qualitatively similar filter characteristics.

arises which regularization operator $\hat{P}$ these kernels might correspond to, and which functions t might be admissible ones. Obviously $\hat{P}$ can not be translation invariant, as this is not the case for k. Note that although lacking translation invariance, these kernels still exhibit (by construction) the property of rotation invariance — orthogonal transformations R are isometries of the Euclidean dot product: $\langle \mathbf{x} \cdot \mathbf{y} \rangle = \langle R\mathbf{x} \cdot R\mathbf{y} \rangle$.

Skipping tedious calculations, we give an example of an operator satisfying

$$k(\mathbf{x}_1, \mathbf{x}_2) = \langle \hat{P}k(\mathbf{x}_1, .) \cdot \hat{P}k(\mathbf{x}_2, .) \rangle \tag{31}$$

for homogeneous polynomials. We then use this result to give an analogous expansion for the inhomogeneous case, and present a sufficient condition for $t(\langle \mathbf{x} \cdot \mathbf{x}' \rangle)$ to be an admissible Mercer kernel.

Let $\mathbf{m} = (m_1, \ldots, m_n) \in \mathbb{N}_0^n$ be a multi index and denote

$$|\mathbf{m}| : = \sum_{i=1}^{n} m_i \text{ and } \begin{pmatrix} p \\ \mathbf{m} \end{pmatrix} : = \frac{p!}{(p - |\mathbf{m}|)! \prod_{i=1}^{n} m_i!} \tag{32}$$

the multinomial coefficient. Moreover, let

$$D_0^{\mathbf{m}} f : = \frac{1}{m_1!}\partial_{x_1}^{m_1}, \ldots, \frac{1}{m_n!}\partial_{x_n}^{m_n} f(\mathbf{x})|_{\mathbf{x}=0} \tag{33}$$

and $\mathrm{e}_{\mathbf{m}}$ be an orthonormal basis, i.e. $\langle \mathrm{e}_{\mathbf{m}} \cdot \mathrm{e}_{\mathbf{m}'} \rangle = \delta_{\mathbf{m}\mathbf{m}'}$. Observe how for each $\mathbf{m}'$ $D_0^{\mathbf{m}}$ extracts exactly one coefficient from the monomials of degree $\mathbf{m}$. Now we can define an operator $\hat{P}_p$ which will act as a regularization operator and satisfy Eq. (31), namely

$$\hat{P}_p = \sum_{|\mathbf{m}|=p} e_{\mathbf{m}} \begin{pmatrix} p \\ \mathbf{m} \end{pmatrix}^{\frac{1}{2}} D_0^{\mathbf{m}} \tag{34}$$

Example 7 (Vapnik, 1995). A simple example of an operator of this type can be obtained for degree 2 homogeneous polynomials on $\mathbb{R}^2$ i.e. for the kernel

$$k(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x} \cdot \mathbf{y} \rangle^2. \tag{35}$$

Denoting $(1/2)\partial_{x_1}^2, \partial_{x_1}\partial_{x_2}, (1/2)\partial_{x_2}^2$ the projectors onto the corresponding monomials we have

$$\hat{P} = \mathrm{e}_1 \frac{1}{2}\partial_{x_1}^2 + \mathrm{e}_2 \sqrt{2}\partial_{x_1}\partial_{x_2} + \mathrm{e}_3 \frac{1}{2}\partial_{x_1}^2 \tag{36}$$

corresponding to

$$\langle (x_1, x_2) \cdot (y_1, y_2) \rangle^2 = \langle (x_1^2, \sqrt{2}x_1x_2, x_2^2) \cdot (y_1^2, \sqrt{2}y_1y_2, y_2^2) \rangle \tag{37}$$

An intuitive description of $\hat{P}$ would be that the data is mapped from $\mathbb{R}^2$ into 3-dimensional feature space ($\mathcal{F} = \mathbb{R}^3$) by computing monomials of degree 2. Subsequently one seeks to compute the *flattest* function in this new space.

Note that $\hat{P}_p$ is only well-defined on functions that are p times differentiable. Accordingly, we will have to restrict the space of functions under consideration to C^p. This is not a major restriction as polynomial kernels are in C^∞ by construction.

It is interesting that the homogeneous polynomial kernel also satisfies the self consistency condition Eq. (13) for the following operator

$$\hat{P} = \sum_{i=0}^{\infty} \hat{P}_i \tag{38}$$

In order to construct an operator for inhomogeneous polynomials, we make use of the expansion

$$(\langle \mathbf{x} \cdot \mathbf{y} \rangle + c)^p = \sum_{i=1}^{p} \begin{pmatrix} p \\ i \end{pmatrix} c^{p-|m|} \langle \mathbf{x} \cdot \mathbf{y} \rangle^i \tag{39}$$

(for convenience set $c = 1$). Hence one may decompose the inhomogeneous polynomial kernel into a series of homogeneous kernels and construct the corresponding operator by

$$\hat{P}_{\mathrm{inh}} = \sum_{i=0}^{p} \begin{pmatrix} p \\ i \end{pmatrix}^{\frac{1}{2}} \hat{P}_i = \sum_{|\mathbf{m}| \le p} \mathrm{e}_{\mathbf{m}} \begin{pmatrix} p \\ \mathbf{m} \end{pmatrix}^{\frac{1}{2}} D_0^{\mathbf{m}} \tag{40}$$

Exploiting this idea even further allows us to state a sufficient condition for $t(\langle \mathbf{x} \cdot \mathbf{y} \rangle)$ to be a Mercer kernel. As homogeneous polynomial kernels satisfy Mercer's condition so does any positive linear combination of them.

Corollary 8 (functions with non-negative power-series). *For every function t(x) that can be expanded into a uniformly convergent power series on $\mathbb{R}$ with nonnegative expansion coefficients, i.e.*

$$t(x) = \sum_{i=0}^{\infty} a_i x^i \text{ with } a_i \ge 0 \tag{41}$$

the kernel $k(\mathbf{x}, \mathbf{y}) := t(\langle \mathbf{x}\cdot\mathbf{y}\rangle)$ is a Mercer kernel and a corresponding regularization operator is

$$\hat{P}_t = \sum_{i=0}^{\infty} a_i^{1/2}\hat{P}_p \tag{42}$$

Consequently, functions like e^x, $\cosh(x)$, $\sinh(x)$, etc. could be used as possible Mercer kernels. Moreover, note that the same argument applies for $t(k(\mathbf{x}, \mathbf{y}))$: if k is any Mercer kernel, and t satisfies the conditions of Corollary 8 then

$$t(k(\mathbf{x},\mathbf{y})) = t(\langle \Phi(\mathbf{x})\cdot\Phi(\mathbf{y})\rangle) \tag{43}$$

is a Mercer kernel. So Eq. (43) provides further means to construct more general kernels, e.g. $\sinh(e^{\langle \mathbf{x}\cdot\mathbf{y}\rangle})$.

7. A new class of support vector kernels

We will follow the lines of Madych and Nelson (1990) as pointed out by Girosi et al. (1993). The main statement is that conditionally positive definite (cpd) functions generate admissible SV kernels. This is very useful as the property of being cpd often is easier to verify than Mercer's condition, especially when combined with the results of Schoenberg and Micchelli on the connection between cpd and completely monotonic functions Schoenberg (1938a); Schoenberg (1938b); Micchelli (1986). Moreover, cpd functions lead to a class of SV kernels that do not necessarily satisfy Mercer's condition.

Definition 9 (conditionally positive definite functions). *A continuous function h, defined on* $[0, \infty)$, *is said to be conditionally positive definite (cpd) of order m on* $\mathbb{R}^n$ *if for any distinct points* $\mathbf{x}_1,\ldots\mathbf{x}_\ell \in \mathbb{R}^n$ *the quadratic form*

$$\sum_{i,j=1}^{\ell} c_i c_j h(\|\mathbf{x}_i - \mathbf{x}_j\|^2) \tag{44}$$

is non-negative provided that the scalars $c_1,\ldots,c_\ell$ *satisfy* $\sum_{i=1}^{\ell} c_i p(\mathbf{x}_i) = 0$ *for all polynomials p on* $\mathbb{R}^n$ *of degree lower than m.*

Definition 10 (completely monotonic functions). *A function h(x) is called completely monotonic of order m if*

$$(-1)^n \frac{d^n}{dx^n} h(x) \geq 0 \text{ for } x \in \mathbb{R}_0^+ \text{ and } n \geq m \tag{45}$$

It can be shown (Schoenberg, 1938a; Schoenberg, 1938b; Micchelli, 1986) that a function $h(x^2)$ is conditionally positive definite if and only if $h(x)$ is completely monotonic of the same order. This gives a (sometimes simpler) criterion for checking whether a function is cpd or not.

Proposition 11 (cpd functions and admissible kernels). *Define* Π_m^n *to be the space of polynomials of degree lower than m on* $\mathbb{R}^n$. *Every cpd function h of order m generates an admissible Kernel for SV expansions on the space of functions f orthogonal to* Π_m^n *by setting* $k(\mathbf{x}_i, \mathbf{x}_j) := h(\|\mathbf{x}_i - \mathbf{x}_j\|^2)$.

Proof. In Dyn (1991); Madych and Nelson (1990) it was shown that cpd functions h of order m generate semi-norms $\|.\|_h$ by

$$\|f\|_h^2 := \int d\mathbf{x}_i d\mathbf{x}_j h(\|\mathbf{x}_i - \mathbf{x}_j\|^2) f(\mathbf{x}_i) f(\mathbf{x}_j) \tag{46}$$

provided that the projection of f onto Π_m^n is zero. For these functions, this, however, also defines a dot product in some feature space. Hence they can be used as SV kernels. ■

Consequently, one may use kernels like those proposed in the context of regularization networks by Girosi et al. (1993) as SV kernels:

$$k(\mathbf{x},\mathbf{y}) = e^{-\beta\|\mathbf{x}-\mathbf{y}\|^2} \text{ Gaussian, } (m=0) \tag{47}$$

$$k(\mathbf{x},\mathbf{y}) = -\sqrt{\|\mathbf{x}-\mathbf{y}\|^2 + c^2} \text{ multiquadric, } (m=1) \tag{48}$$

$$k(\mathbf{x},\mathbf{y}) = \frac{1}{\sqrt{\|\mathbf{x}-\mathbf{y}\|^2 + c^2}} \text{ inverse multiquadric, } (m=0) \tag{49}$$

$$k(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|^2 \ln \|\mathbf{x}-\mathbf{y}\| \text{ thin plate splines, } (m=2) \tag{50}$$

Here the corresponding regularization operator $\hat{P}$ is given implicitly by the seminorm (Eq. (46)) as

$$\|\hat{P}f\|^2 = \|f\|_h^2 \tag{51}$$

However, one has to ensure the orthogonality of our estimate with respect to Π_m^n, i.e. ensure that $\sum_{i=1}^{\ell} c_i p(\mathbf{x}_i) = 0$ for all polynomials p on $\mathbb{R}^n$ of degree lower than m with c_i being the expansion coefficients of the estimate, i.e. α_i.

We proceed with algorithmic details how to actually compute the expansion. In order not to loose expressive power in the estimate f it is necessary to take the polynomials separately into account, i.e. modify Eq. (10) to get

$$f(\mathbf{x}) = \sum_{i=1}^{\ell} c_i k(\mathbf{x}_i, \mathbf{x}) + p(\mathbf{x}) \text{ with } p(\mathbf{x}) \in \Pi_m^n \tag{52}$$

Both of these issues can be addressed by splitting f into a term $f\perp$ orthogonal to Π_m^n for which $\|f^\perp\|_h^2$ is well defined and a polynomial term which will not be regularized at all. (Of course one could define an additional regularization operator for the polynomial part but this would without need render the notation more tedious.) Hence, the regularized risk functional (Eq. (9)) takes on the following form

$$R_{\text{reg}}[f] = R_{\text{emp}}[f] + \frac{\lambda}{2}\|f^\perp\|_h^2 \tag{53}$$

with $f\perp := (1 - \text{Proj}[\Pi_m^n])f$ and Proj[.] denoting the projection operator. Repeating the calculations that led to Eq. (7), yields a similar optimization problem with the difference being that the equality constraint

$$\sum_{i=1}^{l} (\beta_i - \beta_i^*) = 0 \tag{54}$$

has been replaced with

$$\sum_{i=1}^{l} (\beta_i - \beta_i^*) p(\mathbf{x}_i) = 0 \text{ for all } p \in \prod_m^n \tag{55}$$

Note that for the $m = 1$ condition, Eq. (55) reduces to Eq. (54) as Π_1^n contains only the constant function. The resulting optimization problem is positive semidefinite, however, only in the feasible region given by the equality constraints. Some of the eigenvalues of the matrix K may be negative in the space of coefficients not satisfying Eq. (55). It can be seen very easily for the multiquadric case (Eq. (48)) — all entries in K_{ij} are negative. This can lead to numerical instabilities for quadratic programming codes as they usually assume the quadratic matrix to be positive semidefinite not only in the feasible region of the parameters but on the whole space (cf. More and Toraldo, 1991; Vanderbei, 1994). A practical solution to this problem is to remove the space $\mathcal{S}$ spanned by all polynomials Π_m^n on the data $\{\mathbf{x}_1,\ldots,\mathbf{x}_\ell\}$ from the image of K_{ij} while keeping it symmetric by substituting K_{ij} with $((1 - \Pi_{\text{Proj}}[\mathcal{S}])^t K (1 - \Pi_{\text{Proj}}[\mathcal{S}]))_{ij}$. Here Π_{Proj} is the projection matrix on $\mathbb{R}^l$ corresponding to Proj[$\mathcal{S}$].

Example 12 (projecting out Π_1^n). The space Π_1^n consists of all polynomials on $\mathbb{R}^n$ of degree lower than 1, i.e. only of the constant function. Hence $\mathcal{S}$, the span of Π_1^n on any nonempty set $\{\mathbf{x}_1,\ldots,\mathbf{x}_\ell\} \subset \mathbb{R}^n$ *is* span$\{\vec{1}\}$. Consequently, $(1/\ell)\vec{1}\vec{1}^t$ is a projector onto that space and we get[6]

$$K_{ij} \mapsto \left(\left(1 - \frac{1}{\ell}\vec{1}\vec{1}^t \right) K \left(1 - \frac{1}{\ell}\vec{1}\vec{1}^t \right) \right)_{ij} \tag{56}$$

Note that in the standard SV problem this modification of k_{ij} leads to the same solution due to the constraint $\Sigma_i(\alpha_i - \alpha_i^* = 0$.

Example 13 (projecting out Π_2^n). Π_2^n consists of all constant and linear functions on $\{\mathbf{x}_1,\ldots,\mathbf{x}_\ell\}$. Here $\mathcal{S} = \text{span}(\{\mathbf{v}_0,\ldots,\mathbf{v}_n\})$ with

$$\mathbf{v}_0 := (1, \ldots, 1)$$

$$\mathbf{v}_i := (\mathbf{x}_{i1}, \ldots, \mathbf{x}_{i\ell}) \text{ for } i \in 1, \ldots, n$$

In the case of $\ell \leq n + 1$ already a linear model will suffice for reducing $R_{\text{reg}}[f]$ to 0. In this case the solution of the quadratic optimization problem is just 0 as K_{ij} will have rank 0 after the projection.

For $\ell > n + 1$ we will have to transform $\mathbf{v}_0,\ldots,\mathbf{v}_n$ into an orthonormal basis $\mathrm{e}_0,\ldots,\mathrm{e}_n$ of $\mathcal{S}$, e.g. by applying the Gram–Schmidt procedure. This in turn allows us to construct an orthogonal projector onto $\mathcal{S}$ and the corresponding modified matrix from K_{ij}.

As one can observe, only cpd functions of order up to 2 are of practical interest for SV methods as the number of additional constraints and projection operations increases in a combinatoric way thereby rendering the calculations computationally infeasible for $m > 2$.

8. Discussion

A connection between SV kernels and regularization operators has been shown, which may provide one key to understanding why SV machines have been found to exhibit high generalization ability. In particular for the common choices of kernels, the mapping into feature space is not arbitrary but corresponds to good regularization operators (see examples 3, 4 and 6). For kernels, however, where this is not the case, SV machines may show poor performance (example 5). Consequently the regularization framework enables us to analyze the regularization properties of kernels used in practice.

Capacity control is one of the strengths of SV machines; however, this does not mean that the structure of the learning machine, i.e. the choice of a suitable kernel for a given task, should be disregarded. On the contrary, the rather general class of admissible SV kernels should be seen as another strength, provided that we have a means of choosing the right kernel. The newly established link to regularization theory can thus be seen as a tool for constructing the structure consisting of sets of functions in which the SV machine (approximately) performs structural risk minimization (e.g. Vapnik, 1995). In other words it allows to choose an appropriate kernel given the data and the problem specific knowledge.

For completeness an explicit construction of the regularization operators for polynomial kernels has been given in order to provide corresponding operators not only for translation invariant kernels. To make things more transparent Appendix A contains a worked through example for computing a SV kernel for a specific choice of regularization operators.

Note that the regularized risk approach can also be dealt with in a reproducing kernel Hilbert space (RKHS) approach which may lead to sometimes more elegant exposition of the subject, see Kimeldorf and Wahba (1971); Micchelli (1986); Wahba (1990); Girosi (1997); Schölkopf (1997).

Finally the regularization framework made it possible to extend the class of admissible kernels to those defined by conditionally positive definite functions — a class of kernels that do not necessarily have to satisfy Mercer's condition.

A simple consequence of the proposed link is a Bayesian interpretation of support vector machines. In this case the choice of a special kernel can be regarded as a prior on the hypothesis space with $P[f] \propto \exp(-\frac{\lambda}{2}\|\hat{P}f\|^2)$.

[6] Curiously enough the matrix we obtain by this method is identical to the one that is being diagonalized in Kernel PCA (Schölkopf et al., 1996). This is clear as projecting out the span of constant polynomials is equivalent to centering in feature space.

Future work will be necessary for understanding Vapnik's capacity bounds (Vapnik, 1995) from a regularization network point of view.

Acknowledgements

The authors thank Chris Burges, Federico Girosi, Leo van Hemmen, Takashi Onoda, John Shawe-Taylor, Vladimir Vapnik, Grace Wahba, and Alan Yuille for helpful discussions and comments. A.J. Smola is supported by a grant from the DFG (JA 379/71), and B. Schölkopf is supported by a grant from the Studienstiftung des deutschen Volkes.

Appendix A A worked through example

In this section we will construct a support vector kernel for the regularization operator

$$\|\hat{P}f\|^2 = \langle \hat{P}f \cdot \hat{P}f \rangle = \langle f \cdot \hat{P}^* \hat{P} f \rangle = \|f\|_2^2 + \sum_{i=1}^{n} \|\partial_{\mathbf{x}_i} f\|_2^2 \qquad \text{(A1)}$$

This example is taken from Girosi et al. (1993) and used to illustrate our reasoning in detail. For ease of notation assume $f: \mathbb{R} \to \mathbb{R}$.

A corresponding representation of $\hat{P}^* \hat{P}$ in Fourier space ($\tilde{f}$ denoting the Fourier transform of f) yields

$$\|\hat{P}f\|^2 = \int_{\mathbb{R}} \mathrm{d}w |\tilde{f}(w)|^2 (1 + w^2) \qquad \text{(A2)}$$

or equivalently (cf. Section 5, Eq. (20)) $P(w) = 1/(1 + w^2)$. In order to satisfy the self consistency condition (Eq. (13)), we have to compute the inverse Fourier transform of $P(w)$ to obtain the Green's functions of $\hat{P}^* \hat{P}$ (cf. Eq. (21)). This leads to a kernel of the form

$$k(x, x') = \mathrm{e}^{-|x - x'|} \qquad \text{(A3)}$$

A function expansion in terms of this Laplacian kernel (it has the same shape as a Laplacian distribution but should not be confused with the latter at all) however, may not always be desirable as it is by far not as smooth as regressions using a Gaussian kernel (see Fig. 7).

Appendix B Ridge regression

Another frequently used method for selecting the regularization operator is to select D (see Eq. (11)) to be the unit-matrix ($D_{ij} = \delta_{ij}$). This approach often is called ridge regression and is a very popular, method in the context of shrinkage estimators. Now one may pose a similar question as in Section 4, namely regarding the equivalence of ridge regression and support vectors. No answer is available for a direct equivalence, however, we will show that one may obtain models generated by the same type of regularization operators. The requirement for an equivalence of the latter type would be

$$D_{ij} = D(\mathbf{x}_i, \mathbf{x}_j) = \langle (\hat{P}k)(\mathbf{x}_i, .) \cdot (\hat{P}k)(\mathbf{x}_j, .) \rangle = \delta_{ij} \qquad \text{(B1)}$$

for all possible choices of $\mathbf{x}_i \in \mathbb{R}^n$. Unfortunately this requirement cannot be met for the case of the Kronecker δ, as Eq. (B1) implies the function $D(\mathbf{x}_0, .)$ to be nonzero only on a set with (Lebesgue) measure 0. The solution is to change the finite Kronecker δ into the more appropriate δ-distribution, i.e. $\delta(\mathbf{x}_i - \mathbf{x}_j)$.

By a similar reasoning as in Proposition 1, one can see that this is true for $k(\mathbf{x}, \mathbf{y})$ being the Green's function of $\hat{P}$. Note that as a regularization operator, $(\hat{P}^* \hat{P})^{1/2}$ is equivalent to $\hat{P}$, as we can always replace the latter by the former without any difference in the regularization properties. Therefore, without loss of generality, we will assume that $\hat{P}$ is a positive semidefinite endomorphism. Formally we hence require

$$\langle (\hat{P}k)(\mathbf{x}_i, .) \cdot (\hat{P}k)(\mathbf{x}_j, .) \rangle = \langle \delta_{\mathbf{x}_i}(.) \cdot \delta_{\mathbf{x}_j}(.) \rangle = \delta_{\mathbf{x}_i, \mathbf{x}_j} \qquad \text{(B2)}$$

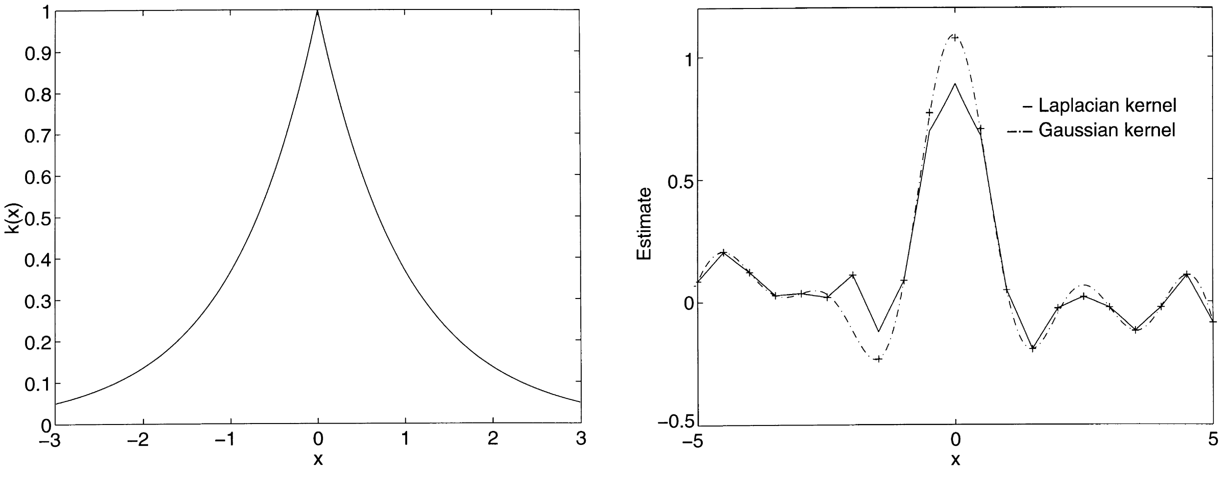

Fig. 7. Left: Laplacian kernel. Right: regression with a Gaussian ($\sigma = 1$) and a Laplacian kernel (kernel width 2) of the data shown in Fig. 4.

Again, this allows us to connect regularization operators and kernels (we have to find the Green's function of $\hat{P}$ to satisfy the equation above). For the special case of translation invariant operators denoted in Fourier space we can associate $\hat{P}$ with $P_{\mathrm{ridge}}(w)$, leading to

$$\|\hat{P}f\|^2 = \int \left| \frac{\tilde{f}(w)}{P_{\mathrm{ridge}}(w)} \right|^2 \mathrm{d}w \tag{B3}$$

Comparing Eq. (B3) with Eq. (20) leads to the conclusion that the following relation between kernels for support vector machines and ridge regression has to hold:

$$\tilde{P}_{\mathrm{SV}}(w) = |P_{\mathrm{ridge}}(w)|^2 \tag{B4}$$

This also explains the good performance of ridge regression models in a smoothing regularizer context (the squared norm of the Fourier transform of kernel functions describes the regularization properties of the corresponding kernel) and allows us to transform support vector machines to ridge regression models and vice versa. Note, however, that we are loosing the sparsity properties of support vectors.

References

Aizerman M.A., Braverman E.M., & Rozonoér L.I. (1964). Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, *25*, 821–837.

Amari S., Murata N., Müller K.-R., Finke M., & Yang H. (1997). Asymptotic statistical theory of overtraining and cross-validation. *IEEE Trans. Neural Networks*, *8 (5)*, 985–996.

Bishop, C.M. (1995). Neural Networks for Pattern Recognition. Oxford: Clarendon Press.

Bochner, S. (1959). Lectures on Fourier integral. Princeton, NJ: Princeton University Press.

Boser, B.E., Guyon, I.M., & Vapnik, V.N. (1992). A training algorithm for optimal margin classifiers. In: D. Haussler (ed.), Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory, Pittsburgh, PA. ACM Press, pp. 144–152.

Cortes C., & Vapnik V. (1995). Support vector networks. In *Mach. Learning*, *20*, 273–297.

Dyn, N. (1991). Interpolation and approximation by radial and related functions. In: C.K. Chui, L.L. Schumaker, D.J. Ward, (Eds.), Approximation Theory, vol. VI. New York: Academic Press, pp. 211–232.

Ferraro M., & Caelli T.M. (1994). Lie transformation groups, integral transforms, and invariant pattern recognition. *Spatial Vision*, *8*, 33–44.

Girosi, F. (1997). An equivalence between sparse approximation and support vector machines. A.I. Memo No. 1606, Artificial Intelligence Laboratory, Massachusetts Institute of Technology (to appear in Neural Computation).

Girosi, F., Jones, M., & Poggio, T. (1993). Priors, stabilizers and basis functions, From regularization to radial, tensor and additive splines. A.I. Memo No. 1430, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1993.

Kearns M. (1997). A bound on the error of cross validation using the approximation and estimation rates, with consequences for the training-test split. *Neural Comput.*, *9 (5)*, 1143–1161.

Kimeldorf G.S., & Wahba G. (1971). A correspondence between Bayesan estimation on stochastic processes and smoothing by splines. *Ann. Math. Statist.*, *2*, 495–502.

MacKay, D.J.C. (1991). Bayesian modelling and neural networks. Ph.D. thesis, Computation and Neural Systems, California Institute of Technology, Pasadena, CA.

Madych W.R., & Nelson S.A. (1990). Multivariate interpolation and conditionally positive definite functions. II.. *Math. Comput.*, *54 (189)*, 211–230.

Micchelli C.A. (1986). Interpolation of scattered data, distance matrices and conditionally positive definite functions. *Constr. Approx.*, *2*, 11–22.

More J.J., & Toraldo G. (1991). On the solution of large quadratic programming problems with bound coustraints. *SIAM J. Optimization*, *1 (1)*, 93–113.

Poggio T. (1975). On optimal nonlinear associative recall. *Biolog. Cybernetics*, *19*, 201–209.

Riesz, F., & Nagy, B.Sz. (1955). Functional Analysis. Frederick Ungar New York, 1955.

Rissanen J. (1985). Minimum-description-length principle. *Ann. Statist.*, *6*, 461–464.

Schoenberg I.J. (1938a). Metric spaces and completely monotone functions. *Ann. Math.*, *39*, 811–841.

Schoenberg I.J. (1938b). Metric spaces and positive definite functions. *Trans. Amer. Math. Soc.*, *44*, 522–536.

Schölkopf, B. (1997). Support vector learning, Ph.D. thesis, Technische Universität Berlin. Also, GMD-Bericht Nr. 287, Oldenbourg Verlag, Munich.

Schölkopf, B., Burges, C., & Vapnik, V. (1995). Extracting support data for a given task. In: U.M. Fayyad, R. Uthurusamy (Eds.), Proceedings, First International Conference on Knowledge Discovery and Data Mining, Menlo Park. AAAI Press.

Schölkopf, B., Simard, P.Y., Smola, A.J., & Vapnik, V.N. (1998). Prior knowledge in support vector kernels. In: M.I. Jordan, M.J. Kearns, & S.A. Solla (Eds.), Advances in Neural information Processings Systems, vol. 10. Cambridge, MA: MIT Press (in press).

Schölkopf, B., Smola, A.J. and Müller, K.R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, *10(5)*, 1299–1319.

Schölkopf B., Sung K., Burges C., Girosi F., Niyogi P., Poggio T., & Vapnik V. (1997). Comparing support vector machines with gaussian kernels to radial basis function classifiers. *IEEE Trans. Sign. Process.*, *45*, 2758–2765.

Segman J., Rubinstein J., & Zeevi Y.Y. (1992). The canonical coordinates method for pattern deformation, Theoretical and computational considerations. *IEEE Trans. Pattern Anal. Mach. Intell.*, *14*, 1171–1183.

Smola, A.J., & Schölkopf, B. (1997). On a kernel-based method for pattern recognition, regression, approximation and operator inversion. Technical Report 1064, GMD FIRST. Algorithmica, 1998, in press.

Smola, A.J., & Schölkopf, B. (1998). From regularization operators to support vector kernels. In: M.I. Jordan, M.J. Kearns, S.A. Solla (Eds.), Advances in Neural Information Processings Systems, vol. 10. Cambridge, MA: MIT Press (in press).

Smola, A.J., Schölkopf, B., & Müller, K.-R. (1998). General cost functions for support vector regression. In: Proceedings of the ACNN'98, Australian Congress on Neural Networks (in press).

Tikhonov, A.N., & Arsenin, V.Y. (1977). Solution of Ill-Posed Problems. Washington, DC: Winston.

Unser M., Aldroubi A., & Eden M. (1991). Fast B-spline transforms for continuous image representation and interpolation. *IEEE Trans. Pattern Anal. Mach. Intell.*, *13 (3)*, 277–285.

Vanderbei, R.J. (1994). LOQO, An interior point code for quadratic programming. Technical Report SOR 94-15, Princeton University.

Vapnik, V.N. (1982). Estimation of Dependences Based on Empirical Data. Berlin: Springer.

Vapnik, V.N. (1995). The Nature of Statistical Learning Theory. New York: Springer.

Vapnik, V.N., Golowich, S., & Smola, A.J. (1997) Support vector method for function approximation, regression estimation, and signal processing, in: NIPS 9, San Mateo, CA.

Wahba, G. (1990). Splines Models for Observational Data, Series in Applied Mathematics, vol. 59. Philadelphia: SIAM.

Yuille, A., & Grzywacz, N. (1998). The motion coherence theory, in: Proceedings of the International Conference on Computer Vision, pp. 344–354. Washington, D.C., December 1988. IEEE Computer Society Press.