MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY
and
CENTER FOR BIOLOGICAL AND COMPUTATIONAL LEARNING
DEPARTMENT OF BRAIN AND COGNITIVE SCIENCES

A.I. Memo No. 1654 March 23, 1999
C.B.C.L Paper No. 171

A unified framework for Regularization Networks and Support Vector Machines

Theodoros Evgeniou, Massimiliano Pontil, Tomaso Poggio

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.
The pathname for this publication is: ai-publications/1500-1999/AIM-1654.ps

Abstract

Regularization Networks and Support Vector Machines are techniques for solving certain problems of learning from examples – in particular the regression problem of approximating a multivariate function from sparse data. We present both formulations in a unified framework, namely in the context of Vapnik's theory of statistical learning which provides a general foundation for the learning problem, combining functional analysis and statistics.

This report describers research done at the Center for Biological & Computational Learning and the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. This research was sponsored by the National Science Foundation under contract No. IIS-9800032, the Office of Naval Research under contract No. N00014-93-1-0385 and contract No. N00014-95-1-0600. Partial support was also provided by Daimler-Benz AG, Eastman Kodak, Siemens Corporate Research, Inc., ATR and AT&T.

Contents

1 Introduction **3**

2 Overview of statistical learning theory **5**
2.1 Uniform Convergence and the Vapnik-Chervonenkis bound 7
2.2 The method of Structural Risk Minimization 10
2.3 ϵ-uniform convergence and the V_γ dimension 10
2.4 Overview of our approach 13

3 Reproducing Kernel Hilbert Spaces: a brief overview **14**

4 Regularization Networks **16**
4.1 Radial Basis Functions 19
4.2 Regularization, generalized splines and kernel smoothers 20
4.3 Dual representation of Regularization Networks 21
4.4 From regression to classification 21

5 Support vector machines **22**
5.1 SVM in RKHS 22
5.2 From regression to classification 24

6 SRM for RNs and SVMs **26**
6.1 SRM for SVM Classification 28
6.1.1 Distribution dependent bounds for SVMC 29

7 A Bayesian Interpretation of Regularization and SRM? **30**
7.1 Maximum A Posteriori Interpretation of Regularization 30
7.2 Bayesian interpretation of the stabilizer in the RN and SVM functionals 32
7.3 Bayesian interpretation of the data term in the Regularization and SVM functional 33
7.4 Why a MAP interpretation may be misleading 33

8 Connections between SVMs and Sparse Approximation techniques **34**
8.1 The problem of sparsity 34
8.2 Equivalence between BPDN and SVMs 36
8.3 Independent Component Analysis 37

9 Remarks **37**
9.1 Regularization Networks can implement SRM 37
9.2 The SVM functional is a special formulation of regularization 38
9.3 SVM, sparsity and compression 38
9.4 Gaussian processes, regularization and SVM 39
9.5 Kernels and how to choose an input representation 39
9.6 Capacity control and the physical world 40

A Regularization Theory for Learning **41**

B An example of RKHS **42**

C Regularized Solutions in RKHS **43**

D Relation between SVMC and SVMR **44**

E Proof of the theorem 6.2 **45**

F The noise model of the data term in SVMR **46**

1 Introduction

The purpose of this paper is to present a theoretical framework for the problem of learning from examples. Learning from examples can be regarded as the regression problem of approximating a multivariate function from sparse data – and we will take this point of view here[1]. The problem of approximating a function from sparse data is ill-posed and a classical way to solve it is regularization theory [92, 10, 11]. Classical regularization theory, as we will consider here[2], formulates the regression problem as a variational problem of finding the function f that minimizes the functional

$$\min_{f \in \mathcal{H}} H[f] = \frac{1}{l} \sum_{i=1}^{l} (y_i - f(\mathbf{x}_i))^2 + \lambda \|f\|_K^2 \tag{1}$$

where $\|f\|_K^2$ is a norm in a Reproducing Kernel Hilbert Space $\mathcal{H}$ defined by the positive definite function K, l is the number of data points or examples (the l pairs $(\mathbf{x}_i, y_i)$) and λ is the regularization parameter (see the seminal work of [102]). Under rather general conditions the solution of equation (1) is

$$f(\mathbf{x}) = \sum_{i=1}^{l} c_i K(\mathbf{x}, \mathbf{x}_i). \tag{2}$$

Until now the functionals of classical regularization have lacked a rigorous justification for a finite set of training data. Their formulation is based on functional analysis arguments which rely on asymptotic results and do not consider finite data sets[3]. Regularization is the approach we have taken in earlier work on learning [69, 39, 77]. The seminal work of Vapnik [94, 95, 96] has now set the foundations for a more general theory that justifies regularization functionals for learning from finite sets and can be used to extend considerably the classical framework of regularization, effectively marrying a functional analysis perspective with modern advances in the theory of probability and statistics. The basic idea of Vapnik's theory is closely related to regularization: for a finite set of training examples the search for the best model or approximating function has to be constrained to an appropriately "small" hypothesis space (which can also be thought of as a space of machines or models or network architectures). If the space is too large, models can be found which will fit exactly the data but will have a poor generalization performance, that is poor predictive capability on new data. Vapnik's theory characterizes and formalizes these concepts in terms of the *capacity* of a set of functions and *capacity control* depending on the training data: for instance, for a small training set the capacity of the function space in which f is sought has to be small whereas it can increase with a larger training set. As we will see later in the case of regularization, a form of capacity control leads to choosing an optimal λ in equation (1) for a given set of data. A key part of the theory is to define and bound the capacity of a set of functions.

Thus the key and somewhat novel theme of this review is a) to describe a unified framework for several learning techniques for finite training sets and b) to justify them in terms of statistical learning theory. We will consider functionals of the form

[1]There is a large literature on the subject: useful reviews are [44, 19, 102, 39], [96] and references therein.

[2]The general regularization scheme for learning is sketched in Appendix A.

[3]The method of quasi-solutions of Ivanov and the equivalent Tikhonov's regularization technique were developed to solve ill-posed problems of the type $Af = F$, where A is a (linear) operator, f is the desired solution in a metric space E_1, and F are the "data" in a metric space E_2.

$$H[f] = \frac{1}{l}\sum_{i=1}^{l} V(y_i, f(\mathbf{x}_i)) + \lambda \|f\|_K^2, \tag{3}$$

where $V(\cdot,\cdot)$ is a *loss function.* We will describe how classical regularization and Support Vector Machines [96] for both regression (SVMR) and classification (SVMC) correspond to the minimization of H in equation (3) for different choices of V:

- Classical (L_2) Regularization Networks (RN)

$$V(y_i, f(\mathbf{x}_i)) = (y_i - f(\mathbf{x}_i))^2 \tag{4}$$

- Support Vector Machines Regression (SVMR)

$$V(y_i, f(\mathbf{x}_i)) = |y_i - f(\mathbf{x}_i)|_\epsilon \tag{5}$$

- Support Vector Machines Classification (SVMC)

$$V(y_i, f(\mathbf{x}_i)) = |1 - y_i f(\mathbf{x}_i)|_+ \tag{6}$$

where $|\cdot|_\epsilon$ is Vapnik's epsilon-insensitive norm (see later), $|x|_+ = x$ if x is positive and zero otherwise, and y_i is a real number in RN and SVMR, whereas it takes values $-1, 1$ in SVMC. Loss function (6) is also called the *soft margin* loss function. For SVMC, we will also discuss two other loss functions:

- The *hard margin* loss function:

$$V(y_i, f(\mathbf{x})) = \theta(1 - y_i f(\mathbf{x}_i)) \tag{7}$$

- The *misclassification* loss function:

$$V(y_i, f(\mathbf{x})) = \theta(-y_i f(\mathbf{x}_i)) \tag{8}$$

Where $\theta(\cdot)$ is the Heaviside function. For classification one should minimize (8) (or (7)), but in practice other loss functions, such as the soft margin one (6) [22, 95], are used. We discuss this issue further in section 6.

The minimizer of (3) using the three loss functions has the same general form (2) (or $f(\mathbf{x}) = \sum_{i=1}^{l} c_i K(\mathbf{x}, \mathbf{x}_i) + b$, see later) but interestingly different properties[4]. In this review we will show how different learning techniques based on the minimization of functionals of the form of H in (3) can be justified for a few choices of $V(\cdot,\cdot)$ using a slight extension of the tools and results of Vapnik's statistical learning theory. In section 2 we outline the main results in the theory of statistical learning and in particular Structural Risk Minimization – the technique suggested by Vapnik to solve the problem of capacity control in learning from "small" training sets. At the end of the section we will outline a technical extension of Vapnik's Structural Risk Minimization framework (SRM). With this extension both RN and Support Vector Machines (SVMs) can be seen within a SRM scheme. In recent years a number of papers claim that SVM cannot be

[4]For general differentiable loss functions V the form of the solution is still the same, as shown in Appendix C.

justified in a data-independent SRM framework (i.e. [86]). One of the goals of this paper is to provide such a data-independent SRM framework that justifies SVM as well as RN. Before describing regularization techniques, section 3 reviews some basic facts on RKHS which are the main function spaces on which this review is focused. After the section on regularization (section 4) we will describe SVMs (section 5). As we saw already, SVMs for regression can be considered as a modification of regularization formulations of the type of equation (1). Radial Basis Functions (RBF) can be shown to be solutions in both cases (for radial K) but with a rather different structure of the coefficients c_i.

Section 6 describes in more detail how and why *both* RN and SVM can be justified in terms of SRM, in the sense of Vapnik's theory: the key to capacity control is how to choose λ for a given set of data. Section 7 describes a naive Bayesian Maximum A Posteriori (MAP) interpretation of RNs and of SVMs. It also shows why a formal MAP interpretation, though interesting and even useful, may be somewhat misleading. Section 8 discusses relations of the regularization and SVM techniques with other representations of functions and signals such as sparse representations from overcomplete dictionaries, Blind Source Separation, and Independent Component Analysis. Finally, section 9 summarizes the main themes of the review and discusses some of the open problems.

2 Overview of statistical learning theory

We consider the case of learning from examples as defined in the statistical learning theory framework [94, 95, 96]. We have two sets of variables $\mathbf{x} \in X \subseteq R^d$ and $y \in Y \subseteq R$ that are related by a probabilistic relationship. We say that the relationship is probabilistic because generally an element of X does not determine uniquely an element of Y, but rather a probability distribution on Y. This can be formalized assuming that a probability distribution $P(\mathbf{x}, y)$ is defined over the set $X \times Y$. The probability distribution $P(\mathbf{x}, y)$ is unknown, and under very general conditions can be written as $P(\mathbf{x}, y) = P(\mathbf{x})P(y|\mathbf{x})$ where $P(y|\mathbf{x})$ is the conditional probability of y given $\mathbf{x}$, and $P(\mathbf{x})$ is the marginal probability of $\mathbf{x}$. We are provided with *examples* of this probabilistic relationship, that is with a data set $D_l \equiv \{(\mathbf{x}_i, y_i) \in X \times Y\}_{i=1}^{l}$ called the *training data*, obtained by sampling l times the set $X \times Y$ according to $P(\mathbf{x}, y)$. The problem of learning consists in, given the data set D_l, providing an *estimator*, that is a function $f : X \to Y$, that can be used, given any value of $\mathbf{x} \in X$, to predict a value y.

In statistical learning theory, the standard way to solve the learning problem consists in defining a *risk functional*, which measures the average amount of error associated with an estimator, and then to look for the estimator, among the allowed ones, with the lowest risk. If $V(y, f(\mathbf{x}))$ is the loss function measuring the error we make when we predict y by $f(\mathbf{x})$[5], then the average error is the so called *expected risk*:

$$I[f] \equiv \int_{X,Y} V(y, f(\mathbf{x}))P(\mathbf{x}, y)\, d\mathbf{x}dy \tag{9}$$

We assume that the expected risk is defined on a "large" class of functions $\mathcal{F}$ and we will denote by f_0 the function which minimizes the expected risk in $\mathcal{F}$:

$$f_0(\mathbf{x}) = \arg\min_{\mathcal{F}} I[f] \tag{10}$$

[5]Typically for regression the loss functions is of the form $V(y - f(\mathbf{x}))$.

The function f_0 is our ideal estimator, and it is often called the *target* function[6].
Unfortunately this function cannot be found in practice, because the probability distribution $P(\mathbf{x}, y)$ that defines the expected risk is unknown, and only a sample of it, the data set D_l, is available. To overcome this shortcoming we need an *induction principle* that we can use to "learn" from the limited number of training data we have. Statistical learning theory as developed by Vapnik builds on the so-called *empirical risk minimization (ERM)* induction principle. The ERM method consists in using the data set D_l to build a stochastic approximation of the expected risk, which is usually called the *empirical risk*, and is defined as[7]:

$$I_{\text{emp}}[f; l] = \frac{1}{l} \sum_{i=1}^{l} V(y_i, f(\mathbf{x}_i)). \tag{11}$$

The central question of the theory is whether the expected risk of the minimizer of the empirical risk in $\mathcal{F}$ is close to the expected risk of f_0. Notice that the question is not necessarily whether we can find f_0 but whether we can *"imitate"* f_0 in the sense that the expected risk of our solution is close to that of f_0. Formally the theory answers the question of finding under which conditions the method of ERM satisfies:

$$\lim_{l \to \infty} I_{\text{emp}}[\hat{f}_l; l] = \lim_{l \to \infty} I[\hat{f}_l] = I[f_0] \tag{12}$$

in probability (all statements are probabilistic since we start with $P(\mathbf{x}, y)$ on the data), where we note with $\hat{f}_l$ the minimizer of the empirical risk (11) in $\mathcal{F}$.
It can been shown (see for example [96]) that in order for the limits in eq. (12) to hold true in probability, or more precisely, for the empirical risk minimization principle to be *non-trivially consistent* (see [96] for a discussion about consistency versus non-trivial consistency), the following *uniform law of large numbers* (which "translates" to *one-sided uniform convergence in probability* of empirical risk to expected risk in $\mathcal{F}$) is a *necessary and sufficient* condition:

$$\lim_{l \to \infty} P \left\{ \sup_{f \in \mathcal{F}} \left(I[f] - I_{\text{emp}}[f; l] \right) > \epsilon \right\} = 0 \quad \forall \epsilon > 0 \tag{13}$$

Intuitively, if $\mathcal{F}$ is very "large" then we can always find $\hat{f}_l \in \mathcal{F}$ with 0 empirical error. This however does not guarantee that the expected risk of $\hat{f}_l$ is also close to 0, or close to $I[f_0]$.
Typically in the literature the *two-sided uniform convergence in probability*:

$$\lim_{l \to \infty} P \left\{ \sup_{f \in \mathcal{F}} \left| I[f] - I_{\text{emp}}[f; l] \right| > \epsilon \right\} = 0 \quad \forall \epsilon > 0 \tag{14}$$

is considered, which clearly implies (13). In this paper we focus on the stronger two-sided case and note that one can get one-sided uniform convergence with some minor technical changes to the theory. We will not discuss the technical issues involved in the relations between consistency, non-trivial consistency, two-sided and one-sided uniform convergence (a discussion can be found in [96]), and from now on we concentrate on the two-sided uniform convergence in probability, which we simply refer to as *uniform convergence*.
The theory of uniform convergence of ERM has been developed in [97, 98, 99, 94, 96]. It has also been studied in the context of *empirical processes* [29, 74, 30]. Here we summarize the main results of the theory.

[6] In the case that V is $(y - f(\mathbf{x}))^2$, the minimizer of eq. (10) is the regression function $f_0(\mathbf{x}) = \int yP(y|\mathbf{x})dy$.
[7] It is important to notice that the data terms (4), (5) and (6) are used for the empirical risks I_{emp}.

2.1 Uniform Convergence and the Vapnik-Chervonenkis bound

Vapnik and Chervonenkis [97, 98] studied under what conditions uniform convergence of the empirical risk to expected risk takes place. The results are formulated in terms of three important quantities that measure the complexity of a set of functions: the *VC entropy*, the *annealed VC entropy*, and the *growth function*. We begin with the definitions of these quantities. First we define the *minimal ϵ-net* of a set, which intuitively measures the "cardinality" of a set at "resolution" ϵ:

Definition 2.1 *Let A be a set in a metric space $\mathcal{A}$ with distance metric d. For a fixed $\epsilon > 0$, the set $B \subseteq \mathcal{A}$ is called an ϵ-net of A in $\mathcal{A}$, if for any point $a \in A$ there is a point $b \in B$ such that $d(a, b) < \epsilon$. We say that the set B is a minimal ϵ-net of A in $\mathcal{A}$, if it is finite and contains the minimal number of elements.*

Given a training set $D_l = \{(\mathbf{x}_i, y_i) \in X \times Y\}_{i=1}^l$, consider the set of l-dimensional vectors:

$$q(f) = (V(y_1, f(\mathbf{x}_1)), ..., V(y_l, f(\mathbf{x}_l))) \tag{15}$$

with $f \in \mathcal{F}$, and define the number of elements of the minimal ϵ-net of this set under the metric:

$$d(q(f), q(f')) = \max_{1 \le i \le l} |V(y_i, f(\mathbf{x}_i)) - V(y_i, f'(\mathbf{x}_i))|$$

to be $\mathcal{N}^{\mathcal{F}}(\epsilon; D_l)$ (which clearly depends both on $\mathcal{F}$ and on the loss function V). Intuitively this quantity measures how many different functions effectively we have at "resolution" ϵ, when we only care about the values of the functions at points in D_l. Using this quantity we now give the following definitions:

Definition 2.2 *Given a set $X \times Y$ and a probability $P(\mathbf{x}, y)$ defined over it, the VC entropy of a set of functions $V(y, f(\mathbf{x}))$, $f \in \mathcal{F}$, on a data set of size l is defined as:*

$$H^{\mathcal{F}}(\epsilon; l) \equiv \int_{X,Y} \ln \mathcal{N}^{\mathcal{F}}(\epsilon; D_l) \prod_{i=1}^{l} P(\mathbf{x}_i, y_i) d\mathbf{x}_i dy_i$$

Definition 2.3 *Given a set $X \times Y$ and a probability $P(\mathbf{x}, y)$ defined over it, the annealed VC entropy of a set of functions $V(y, f(\mathbf{x}))$, $f \in \mathcal{F}$, on a data set of size l is defined as:*

$$H^{\mathcal{F}}_{\text{ann}}(\epsilon; l) \equiv \ln \int_{X,Y} \mathcal{N}^{\mathcal{F}}(\epsilon; D_l) \prod_{i=1}^{l} P(\mathbf{x}_i, y_i) d\mathbf{x}_i dy_i$$

Definition 2.4 *Given a set $X \times Y$, the growth function of a set of functions $V(y, f(\mathbf{x}))$, $f \in \mathcal{F}$, on a data set of size l is defined as:*

$$G^{\mathcal{F}}(\epsilon; l) \equiv \ln \left(\sup_{D_l \in (X \times Y)^l} \mathcal{N}^{\mathcal{F}}(\epsilon; D_l) \right)$$

Notice that all three quantities are functions of the number of data l and of ϵ, and that clearly:

$$H^{\mathcal{F}}(\epsilon; l) \leq H^{\mathcal{F}}_{\text{ann}}(\epsilon; l) \leq G^{\mathcal{F}}(\epsilon; l) \; .$$

These definitions can easily be extended in the case of *indicator functions*, i.e. functions taking binary values[8] such as $\{-1, 1\}$, in which case the three quantities do not depend on ϵ for $\epsilon < 1$, since the vectors (15) are all at the vertices of the hypercube $\{0, 1\}^l$.
Using these definitions we can now state three important results of statistical learning theory [96]:

- For a given probability distribution $P(\mathbf{x}, y)$:
  1. The necessary and sufficient condition for uniform convergence is that
  $$\lim_{l \to \infty} \frac{H^{\mathcal{F}}(\epsilon; l)}{l} = 0 \quad \forall \epsilon > 0$$
  2. A sufficient condition for *fast asymptotic rate of convergence*[9] is that
  $$\lim_{l \to \infty} \frac{H^{\mathcal{F}}_{\text{ann}}(\epsilon; l)}{l} = 0 \quad \forall \epsilon > 0$$
  It is an open question whether this is also a necessary condition.
- A sufficient condition for distribution *independent* (that is, for any $P(\mathbf{x}, y)$) fast rate of convergence is that
  $$\lim_{l \to \infty} \frac{G^{\mathcal{F}}(\epsilon; l)}{l} = 0 \quad \forall \epsilon > 0$$
  For indicator functions this is also a necessary condition.

According to statistical learning theory, these three quantities are what one should consider when designing and analyzing learning machines: the VC-entropy and the annealed VC-entropy for an analysis which depends on the probability distribution $P(\mathbf{x}, y)$ of the data, and the growth function for a distribution *independent* analysis. In this paper we consider only distribution *independent* results, although the reader should keep in mind that distribution dependent results are likely to be important in the future.
Unfortunately the growth function of a set of functions is difficult to compute in practice. So the standard approach in statistical learning theory is to use an upper bound on the growth function which is given using another important quantity, the *VC-dimension*, which is another (*looser*) measure of the complexity, *capacity*, of a set of functions. In this paper we concentrate on this quantity, but it is important that the reader keeps in mind that the VC-dimension is in a sense a "weak" measure of complexity of a set of functions, so it typically leads to loose upper bounds on the growth function: in general one is better off, theoretically, using directly the growth function. We now discuss the VC-dimension and its implications for learning.
The VC-dimension was first defined for the case of indicator functions and then was extended to real valued functions.

[8] In the case of indicator functions, y is binary, and V is 0 for $f(x) = y$, 1 otherwise.

[9] This means that for any $l > l_0$ we have that $P\{\sup_{f \in \mathcal{F}} |I[f] - I_{\text{emp}}[f]| > \epsilon\} < e^{-c\epsilon^2 l}$ for some constant $c > 0$. Intuitively, fast rate is typically needed in practice.

Definition 2.5 *The VC-dimension of a set* $\{\theta(f(\mathbf{x})), f \in \mathcal{F}\}$*, of indicator functions is the maximum number* h *of vectors* $\mathbf{x}_1, \ldots, \mathbf{x}_h$ *that can be separated into two classes in all* 2^h *possible ways using functions of the set.*
If, for any number N*, it is possible to find* N *points* $\mathbf{x}_1, \ldots, \mathbf{x}_N$ *that can be separated in all the* 2^N *possible ways, we will say that the VC-dimension of the set is infinite.*

The remarkable property of this quantity is that, although as we mentioned the VC-dimension only provides an upper bound to the growth function, in the case of indicator functions, *finiteness of the VC-dimension is a* **necessary and sufficient** *condition for uniform convergence (eq. (14))* **independent** *of the underlying distribution* $P(\mathbf{x}, y)$.

Definition 2.6 *Let* $A \leq V(y, f(\mathbf{x})) \leq B$*,* $f \in \mathcal{F}$*, with* A *and* $B < \infty$*. The VC-dimension of the set* $\{V(y, f(\mathbf{x})),\ f \in \mathcal{F}\}$ *is defined as the VC-dimension of the set of indicator functions* $\{\theta\left(V(y, f(\mathbf{x})) - \alpha\right),\ \ \alpha \in (A, B)\}$.

Sometimes we refer to the VC-dimension of $\{V(y, f(\mathbf{x})),\ f \in \mathcal{F}\}$ as the VC dimension of V in $\mathcal{F}$. It can be easily shown that for $y \in \{-1, +1\}$ and for $V(y, f(\mathbf{x})) = \theta(-yf(\mathbf{x}))$ as the loss function, the VC dimension of V in $\mathcal{F}$ computed using definition 2.6 is equal to the VC dimension of the set of indicator functions $\{\theta(f(\mathbf{x})),\ f \in \mathcal{F}\}$ computed using definition 2.5. In the case of real valued functions, finiteness of the VC-dimension is *only sufficient* for uniform convergence. Later in this section we will discuss a measure of capacity that provides also necessary conditions.
An important outcome of the work of Vapnik and Chervonenkis is that the uniform deviation between empirical risk and expected risk in a hypothesis space can be bounded in terms of the VC-dimension, as shown in the following theorem:

Theorem 2.1 *(Vapnik and Chervonenkis 1971) Let* $A \leq V(y, f(\mathbf{x})) \leq B$*,* $f \in \mathcal{F}$*,* $\mathcal{F}$ *be a set of bounded functions and* h *the VC-dimension of* V *in* $\mathcal{F}$*. Then, with probability at least* $1 - \eta$*, the following inequality holds simultaneously for all the elements* f *of* $\mathcal{F}$*:*

$$I_{\mathrm{emp}}[f; l] - (B - A)\sqrt{\frac{h \ln \frac{2el}{h} - \ln(\frac{\eta}{4})}{l}} \leq I[f] \leq I_{\mathrm{emp}}[f; l] + (B - A)\sqrt{\frac{h \ln \frac{2el}{h} - \ln(\frac{\eta}{4})}{l}} \qquad (16)$$

The quantity $|I[f] - I_{\mathrm{emp}}[f; l]|$ is often called *estimation error*, and bounds of the type above are usually called *VC bounds*[10]. From eq. (16) it is easy to see that with probability at least $1 - \eta$:

$$I[\hat{f}_l] - 2(B - A)\sqrt{\frac{h \ln \frac{2el}{h} - \ln(\frac{\eta}{4})}{l}} \leq I[f_0] \leq I[\hat{f}_l] + 2(B - A)\sqrt{\frac{h \ln \frac{2el}{h} - \ln(\frac{\eta}{4})}{l}} \qquad (17)$$

where $\hat{f}_l$ is, as in (12), the minimizer of the empirical risk in $\mathcal{F}$.
A very interesting feature of inequalities (16) and (17) is that they are non-asymptotic, meaning that they hold for any finite number of data points l, and that the error bounds do not necessarily depend on the dimensionality of the variable $\mathbf{x}$.
Observe that theorem (2.1) and inequality (17) are meaningful in practice only if the VC-dimension of the loss function V in $\mathcal{F}$ is finite and less than l. Since the space $\mathcal{F}$ where the

[10] It is important to note that bounds on the expected risk using the annealed VC-entropy also exist. These are tighter than the VC-dimension ones.

loss function V is defined is usually very large (i.e. all functions in L_2), one typically considers smaller hypothesis spaces $\mathcal{H}$. The cost associated with restricting the space is called the *approximation error* (see below). In the literature, space $\mathcal{F}$ where V is defined is called the *target space*, while $\mathcal{H}$ is what is called the *hypothesis space*. Of course, all the definitions and analysis above still hold for $\mathcal{H}$, where we replace f_0 with the minimizer of the expected risk in $\mathcal{H}$, $\hat{f}_l$ is now the minimizer of the empirical risk in $\mathcal{H}$, and h the VC-dimension of the loss function V in $\mathcal{H}$. Inequalities (16) and (17) suggest a method for achieving good generalization: not only minimize the empirical risk, but instead minimize a combination of the empirical risk and the complexity of the hypothesis space. This observation leads us to the method of *Structural Risk Minimization* that we describe next.

2.2 The method of Structural Risk Minimization

The idea of SRM is to define a nested sequence of hypothesis spaces $H_1 \subset H_2 \subset \ldots \subset H_{n(l)}$ with $n(l)$ a non-decreasing integer function of l, where each hypothesis space H_i has VC-dimension finite and larger than that of all previous sets, i.e. if h_i is the VC-dimension of space H_i, then $h_1 \leq h_2 \leq \ldots \leq h_{n(l)}$. For example H_i could be the set of polynomials of degree i, or a set of splines with i nodes, or some more complicated nonlinear parameterization. For each element H_i of the structure the solution of the learning problem is:

$$\hat{f}_{i,l} = \arg \min_{f \in H_i} I_{\mathrm{emp}}[f; l] \tag{18}$$

Because of the way we define our structure it should be clear that the larger i is the smaller the empirical error of $\hat{f}_{i,l}$ is (since we have greater "flexibility" to fit our training data), but the larger the VC-dimension part (second term) of the right hand side of (16) is. Using such a nested sequence of more and more complex hypothesis spaces, the SRM learning technique consists of choosing the space $H_{n^*(l)}$ for which the right hand side of inequality (16) is minimized. It can be shown [94] that for the chosen solution $\hat{f}_{n^*(l),l}$ inequalities (16) and (17) hold with probability at least $(1-\eta)^{n(l)} \approx 1 - n(l)\eta$ [11], where we replace h with $h_{n^*(l)}$, f_0 with the minimizer of the expected risk in $H_{n^*(l)}$, namely $f_{n^*(l)}$, and $\hat{f}_l$ with $\hat{f}_{n^*(l),l}$.
With an appropriate choice of $n(l)$[12] it can be shown that as $l \to \infty$ and $n(l) \to \infty$, the expected risk of the solution of the method approaches in probability the minimum of the expected risk in $\mathcal{H} = \bigcup_{i=1}^{\infty} H_i$, namely $I[f_{\mathcal{H}}]$. Moreover, if the target function f_0 belongs to the closure of $\mathcal{H}$, then eq. (12) holds in probability (see for example [96]).
However, in practice l is finite ("small"), so $n(l)$ is small which means that $\mathcal{H} = \bigcup_{i=1}^{n(l)} H_i$ is a small space. Therefore $I[f_{\mathcal{H}}]$ may be much larger than the expected risk of our target function f_0, since f_0 may not be in $\mathcal{H}$. The distance between $I[f_{\mathcal{H}}]$ and $I[f_0]$ is called the approximation error and can be bounded using results from approximation theory. We do not discuss these results here and refer the reader to [54, 26].

2.3 ϵ-uniform convergence and the V_γ dimension

As mentioned above finiteness of the VC-dimension is *not* a necessary condition for uniform convergence in the case of real valued functions. To get a necessary condition we need a slight

[11] We want (16) to hold simultaneously for all spaces H_i, since we choose the best $\hat{f}_{i,l}$.
[12] Various cases are discussed in [27], i.e. $n(l) = l$.

extension of the VC-dimension that has been developed (among others) in [50, 2], known as the V_γ–dimension[13]. Here we summarize the main results of that theory that we will also use later on to design regression machines for which we will have distribution independent uniform convergence.

Definition 2.7 *Let $A \leq V(y, f(\mathbf{x})) \leq B$, $f \in \mathcal{F}$, with A and $B < \infty$. The V_γ-dimension of V in $\mathcal{F}$ (of the set $\{V(y, f(\mathbf{x})),\ f \in \mathcal{F}\}$) is defined as the the maximum number h of vectors $(\mathbf{x}_1, y_1) \ldots, (\mathbf{x}_h, y_h)$ that can be separated into two classes in all 2^h possible ways using rules:*

$$\textit{class 1 if: } V(y_i, f(\mathbf{x}_i)) \geq s + \gamma$$
$$\textit{class 0 if: } V(y_i, f(\mathbf{x}_i)) \leq s - \gamma$$

for $f \in \mathcal{F}$ and some $s \geq 0$. If, for any number N, it is possible to find N points $(\mathbf{x}_1, y_1) \ldots, (\mathbf{x}_N, y_N)$ that can be separated in all the 2^N possible ways, we will say that the V_γ-dimension of V in $\mathcal{F}$ is infinite.

Notice that for $\gamma = 0$ this definition becomes the same as definition 2.6 for VC-dimension. Intuitively, for $\gamma > 0$ the "rule" for separating points is more restrictive than the rule in the case $\gamma = 0$. It requires that there is a "margin" between the points: points for which $V(y, f(\mathbf{x}))$ is between $s + \gamma$ and $s - \gamma$ are not classified. As a consequence, the V_γ dimension is a decreasing function of γ and in particular is smaller than the VC-dimension.
If V is an indicator function, say $\theta(-yf(\mathbf{x}))$, then for any γ definition 2.7 reduces to that of the VC-dimension of a set of indicator functions.
Generalizing slightly the definition of eq. (14) we will say that for a given $\epsilon > 0$ the ERM method converges ϵ-uniformly in $\mathcal{F}$ in probability, (or that there is ϵ-uniform convergence) if:

$$\lim_{l \to \infty} P\left\{ \sup_{f \in \mathcal{F}} |I_{\mathrm{emp}}[f; l] - I[f]| > \epsilon \right\} = 0. \tag{19}$$

Notice that if eq. (19) holds for every $\epsilon > 0$ we have uniform convergence (eq. (14)). It can be shown (variation of [96]) that ϵ-uniform convergence in probability implies that:

$$I[\hat{f}_l] \leq I[f_0] + 2\epsilon \tag{20}$$

in probability, where, as before, $\hat{f}_l$ is the minimizer of the empirical risk and f_0 is the minimizer of the expected expected risk in $\mathcal{F}$[14].
The basic theorems for the V_γ-dimension are the following:

Theorem 2.2 *(Alon et al. , 1993) Let $A \leq V(y, f(\mathbf{x}))) \leq B$, $f \in \mathcal{F}$, $\mathcal{F}$ be a set of bounded functions. For any $\epsilon > 0$, if the V_γ dimension of V in $\mathcal{F}$ is finite for $\gamma = \alpha\epsilon$ for some constant $\alpha \geq \frac{1}{48}$, then the ERM method ϵ-converges in probability.*

Theorem 2.3 *(Alon et al. , 1993) Let $A \leq V(y, f(\mathbf{x}))) \leq B$, $f \in \mathcal{F}$, $\mathcal{F}$ be a set of bounded functions. The ERM method uniformly converges (in probability) if and only if the V_γ dimension of V in $\mathcal{F}$ is finite for every $\gamma > 0$. So finiteness of the V_γ dimension for every $\gamma > 0$ is a necessary* **and** *sufficient condition for distribution independent uniform convergence of the ERM method for real-valued functions.*

[13] In the literature, other quantities, such as the *fat-shattering* dimension and the P_γ dimension, are also defined. They are closely related to each other, and are essentially equivalent to the V_γ dimension for the purpose of this paper. The reader can refer to [2, 7] for an in-depth discussion on this topic.

[14] This is like ϵ-learnability in the PAC model [93].

Theorem 2.4 *(Alon et al. , 1993) Let $A \leq V(y, f(\mathbf{x})) \leq B$, $f \in \mathcal{F}$, $\mathcal{F}$ be a set of bounded functions. For any $\epsilon \geq 0$, for all $l \geq \frac{2}{\epsilon^2}$ we have that if h_γ is the V_γ dimension of V in $\mathcal{F}$ for $\gamma = \alpha\epsilon$ ($\alpha \geq \frac{1}{48}$), h_γ finite, then:*

$$P\left\{\sup_{f\in\mathcal{F}} |I_{\mathrm{emp}}[f;l] - I[f]| > \epsilon\right\} \leq \mathcal{G}(\epsilon, l, h_\gamma), \tag{21}$$

where $\mathcal{G}$ is an increasing function of h_γ and a decreasing function of ϵ and l, with $\mathcal{G} \to 0$ as $l \to \infty$ [15].

From this theorem we can easily see that for any $\epsilon > 0$, for all $l \geq \frac{2}{\epsilon^2}$:

$$P\left\{I[\hat{f}_l] \leq I[f_0] + 2\epsilon\right\} \geq 1 - 2\mathcal{G}(\epsilon, l, h_\gamma), \tag{22}$$

where $\hat{f}_l$ is, as before, the minimizer of the empirical risk in $\mathcal{F}$. An important observations to keep in mind is that theorem 2.4 requires the V_γ dimension of the loss function V in $\mathcal{F}$. In the case of classification, this implies that if we want to derive bounds on the expected misclassification we have to use the V_γ dimension of the loss function $\theta(-yf(\mathbf{x}))$ (which is the $VC-$*dimension* of the set of indicator functions $\{\mathrm{sgn}\,(f(\mathbf{x})), f \in \mathcal{F}\}$), and *not* the V_γ dimension of the set $\mathcal{F}$.
The theory of the V_γ dimension justifies the "extended" SRM method we describe below. It is important to keep in mind that the method we describe is only of theoretical interest and will only be used later as a theoretical motivation for RN and SVM. It should be clear that all the definitions and analysis above still hold for any hypothesis space $\mathcal{H}$, where we replace f_0 with the minimizer of the expected risk in $\mathcal{H}$, $\hat{f}_l$ is now the minimizer of the empirical risk in $\mathcal{H}$, and h the VC-dimension of the loss function V in $\mathcal{H}$.
Let l be the number of training data. For a fixed $\epsilon > 0$ such that $l \geq \frac{2}{\epsilon^2}$, let $\gamma = \frac{1}{48}\epsilon$, and consider, as before, a nested sequence of hypothesis spaces $H_1 \subset H_2 \subset \ldots \subset H_{n(l,\epsilon)}$, where each hypothesis space H_i has V_γ-dimension finite and larger than that of all previous sets, i.e. if h_i is the V_γ-dimension of space H_i, then $h_1 \leq h_2 \leq \ldots \leq h_{n(l,\epsilon)}$. For each element H_i of the structure consider the solution of the learning problem to be:

$$\hat{f}_{i,l} = \arg\min_{f\in H_i} I_{\mathrm{emp}}[f;l]. \tag{23}$$

Because of the way we define our structure the larger i is the smaller the empirical error of $\hat{f}_{i,l}$ is (since we have more "flexibility" to fit our training data), but the larger the right hand side of inequality (21) is. Using such a nested sequence of more and more complex hypothesis spaces, this *extended SRM* learning technique consists of finding the structure element $H_{n^*(l,\epsilon)}$ for which the trade off between empirical error and the right hand side of (21) is optimal. One practical idea is to find numerically for each H_i the "effective" ϵ_i so that the bound (21) is the same for all H_i, and then choose $\hat{f}_{i,l}$ for which the sum of the empirical risk and ϵ_i is minimized.
We *conjecture* that as $l \to \infty$, for appropriate choice of $n(l,\epsilon)$ with $n(l,\epsilon) \to \infty$ as $l \to \infty$, the expected risk of the solution of the method converges in probability to a value less than 2ϵ away from the minimum expected risk in $\mathcal{H} = \bigcup_{i=1}^{\infty} H_i$. Notice that we described an SRM method for a fixed ϵ. If the V_γ dimension of H_i is finite for every $\gamma > 0$, we can further modify the extended SRM method so that $\epsilon \to 0$ as $l \to \infty$. We *conjecture* that if the target function f_0 belongs to the

[15] Closed forms of $\mathcal{G}$ can be derived (see for example [2]) but we do not present them here for simplicity of notation.

closure of $\mathcal{H}$, then as $l \to \infty$, with appropriate choices of ϵ, $n(l,\epsilon)$ and $n^*(l,\epsilon)$ the solution of this SRM method can be proven (as before) to satisfy eq. (12) in probability. Finding appropriate forms of ϵ, $n(l,\epsilon)$ and $n^*(l,\epsilon)$ is an open theoretical problem (which we believe to be a technical matter). Again, as in the case of "standard" SRM, in practice l is finite so $\mathcal{H} = \bigcup_{i=1}^{n(l,\epsilon)} H_i$ is a small space and the solution of this method may have expected risk much larger that the expected risk of the target function. Approximation theory can be used to bound this difference [61].

The proposed method is difficult to implement in practice since it is difficult to decide the optimal trade off between empirical error and the bound (21). If we had constructive bounds on the deviation between the empirical and the expected risk like that of theorem 2.1 then we could have a practical way of choosing the optimal element of the structure. Unfortunately existing bounds of that type [2, 7] are not tight. So the final choice of the element of the structure may be done in practice using other techniques such as cross-validation [102].

2.4 Overview of our approach

In order to set the stage for the next two sections on regularization and Support Vector Machines, we outline here how we can justify the proper use of the RN and the SVM functionals (see (3)) in the framework of the SRM principles just described.

The basic idea is to define a structure in terms of a nested sequence of hypothesis spaces $H_1 \subset H_2 \subset \ldots \subset H_{n(l)}$ with H_m being the set of functions f in the RKHS with:

$$\|f\|_K \le A_m, \tag{24}$$

where A_m is a monotonically increasing sequence of positive constants. Following the SRM method outlined above, for each m we will minimize the empirical risk

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

subject to the constraint (24). This in turn leads to using the Lagrange multiplier λ_m and to minimizing

$$\frac{1}{l}\sum_{i=1}^{l} V(y_i, f(\mathbf{x}_i)) + \lambda_m(\|f\|_K^2 - A_m^2),$$

with respect to f and maximizing with respect to $\lambda_m \ge 0$ for each element of the structure. We can then choose the optimal $n^*(l)$ and the associated $\lambda^*(l)$, and get the optimal solution $\hat{f}_{n^*(l)}$.

The solution we get using this method is clearly the same as the solution of:

$$\frac{1}{l}\sum_{i=1}^{l} V(y_i, f(\mathbf{x}_i)) + \lambda^*(l)\|f\|_K^2 \tag{25}$$

where $\lambda^*(l)$ is the optimal Lagrange multiplier corresponding to the optimal element of the structure $A_{n^*(l)}$. Notice that this approach is quite general. In particular it can be applied to classical L_2 regularization, to SVM regression, and, as we will see, to SVM classification with the appropriate $V(\cdot,\cdot)$.

In section 6 we will describe in detail this approach for the case that the elements of the structure are infinite dimensional RKHS. We have outlined this theoretical method here so that the reader understands our motivation for reviewing in the next two sections the approximation schemes resulting from the minimization of functionals of the form of equation (25) for three specific choices of the loss function V:

- $V(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$ for regularization.
- $V(y, f(\mathbf{x})) = |y - f(\mathbf{x})|_\epsilon$ for SVM regression.
- $V(y, f(\mathbf{x})) = |1 - yf(\mathbf{x})|_+$ for SVM classification.

For SVM classification the loss functions:

- $V(y, f(\mathbf{x})) = \theta(1 - yf(\mathbf{x}))$ (hard margin loss function), and
- $V(y, f(\mathbf{x})) = \theta(-yf(\mathbf{x}))$ (misclassification loss function)

will also be discussed. First we present an overview of RKHS which are the hypothesis spaces we consider in the paper.

3 Reproducing Kernel Hilbert Spaces: a brief overview

A Reproducing Kernel Hilbert Space (RKHS) [5] is a Hilbert space $\mathcal{H}$ of functions defined over some bounded domain $X \subset R^d$ with the property that, for each $\mathbf{x} \in X$, the evaluation functionals $\mathcal{F}_\mathbf{x}$ defined as

$$\mathcal{F}_\mathbf{x}[f] = f(\mathbf{x}) \quad \forall f \in \mathcal{H}$$

are linear, bounded functionals. The boundedness means that there exists a $U = U_x \in R^+$ such that:

$$|\mathcal{F}_\mathbf{x}[f]| = |f(\mathbf{x})| \leq U||f||$$

for all f in the RKHS.
It can be proved [102] that to every RKHS $\mathcal{H}$ there corresponds a *unique positive definite* function $K(\mathbf{x}, \mathbf{y})$ of two variables in X, called the *reproducing kernel* of $\mathcal{H}$ (hence the terminology RKHS), that has the following *reproducing property*:

$$f(\mathbf{x}) =< f(\mathbf{y}), K(\mathbf{y}, \mathbf{x}) >_\mathcal{H} \quad \forall f \in \mathcal{H}, \tag{26}$$

where $< \cdot, \cdot >_\mathcal{H}$ denotes the scalar product in $\mathcal{H}$. The function K behaves in $\mathcal{H}$ as the delta function does in L_2, although L_2 is not a RKHS (the functionals $\mathcal{F}_\mathbf{x}$ are clearly not bounded).
To make things clearer we sketch a way to construct a RKHS, which is relevant to our paper. The mathematical details (such as the convergence or not of certain series) can be found in the theory of integral equations [45, 20, 23].
Let us assume that we have a sequence of positive numbers λ_n and linearly independent functions $\phi_n(\mathbf{x})$ such that they define a function $K(\mathbf{x}, \mathbf{y})$ in the following way [16]:

$$K(\mathbf{x}, \mathbf{y}) \equiv \sum_{n=0}^{\infty} \lambda_n \phi_n(\mathbf{x}) \phi_n(\mathbf{y}), \tag{27}$$

[16] When working with complex functions $\phi_n(\mathbf{x})$ this formula should be replaced with $K(\mathbf{x}, \mathbf{y}) \equiv \sum_{n=0}^{\infty} \lambda_n \phi_n(\mathbf{x}) \phi_n^*(\mathbf{y})$

where the series is well defined (for example it converges uniformly). A simple calculation shows that the function K defined in eq. (27) is positive definite. Let us now take as our Hilbert space to be the set of functions of the form:

$$f(\mathbf{x}) = \sum_{n=0}^{\infty} a_n \phi_n(\mathbf{x}) \tag{28}$$

for any $a_n \in R$, and define the scalar product in our space to be:

$$< \sum_{n=0}^{\infty} a_n \phi_n(\mathbf{x}), \sum_{n=0}^{\infty} d_n \phi_n(\mathbf{x}) >_{\mathcal{H}} \equiv \sum_{n=0}^{\infty} \frac{a_n d_n}{\lambda_n}. \tag{29}$$

Assuming that all the evaluation functionals are bounded, it is now easy to check that such an Hilbert space is a RKHS with reproducing kernel given by $K(\mathbf{x}, \mathbf{y})$. In fact we have:

$$< f(\mathbf{y}), K(\mathbf{y}, \mathbf{x}) >_{\mathcal{H}} = \sum_{n=0}^{\infty} \frac{a_n \lambda_n \phi_n(\mathbf{x})}{\lambda_n} = \sum_{n=0}^{\infty} a_n \phi_n(\mathbf{x}) = f(\mathbf{x}), \tag{30}$$

hence equation (26) is satisfied.
Notice that when we have a finite number of ϕ_n, the λ_n can be arbitrary (finite) numbers, since convergence is ensured. In particular they can all be equal to one.
Generally, it is easy to show [102] that whenever a function K of the form (27) is available, it is possible to construct a RKHS as shown above. Vice versa, for any RKHS there is a unique kernel K and corresponding λ_n, ϕ_n, that satisfy equation (27) and for which equations (28), (29) and (30) hold for all functions in the RKHS. Moreover, equation (29) shows that the norm of the RKHS has the form:

$$\|f\|_K^2 = \sum_{n=0}^{\infty} \frac{a_n^2}{\lambda_n}. \tag{31}$$

The ϕ_n consist a basis for the RKHS (not necessarily orthonormal), and the kernel K is the "correlation" matrix associated with these basis functions. It is in fact well know that there is a close relation between Gaussian processes and RKHS [58, 40, 72]. Wahba [102] discusses in depth the relation between regularization, RKHS and correlation functions of Gaussian processes. The choice of the ϕ_n defines a space of functions – the functions that are spanned by the ϕ_n.
We also call the space $\{(\phi_n(\mathbf{x}))_{n=1}^{\infty},\ \mathbf{x} \in X\}$ the *feature space* induced by the kernel K. The choice of the ϕ_n defines the feature space where the data $\mathbf{x}$ are "mapped". In this paper we refer to the dimensionality of the feature space as the *dimensionality of the RKHS*. This is clearly equal to the number of basis elements ϕ_n, which does not necessarily have to be infinite. For example, with K a Gaussian, the dimensionality of the RKHS is infinite ($\phi_n(\mathbf{x})$ are the Fourier components $e^{i\mathbf{n}\cdot\mathbf{x}}$), while when K is a polynomial of degree k ($K(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x} \cdot \mathbf{y})^k$ - see section 4), the dimensionality of the RKHS is finite, and all the infinite sums above are replaced with finite sums.
It is well known that expressions of the form (27) actually abound. In fact, it follows from Mercer's theorem [45] that any function $K(\mathbf{x}, \mathbf{y})$ which is the kernel of a positive operator [17] in $L_2(\Omega)$ has an expansion of the form (27), in which the ϕ_i and the λ_i are respectively the orthogonal eigenfunctions and the positive eigenvalues of the operator corresponding to K. In

[17] We remind the reader that positive definite operators in L_2 are self-adjoint operators such that $< Kf, f > \geq 0$ for all $f \in L_2$.

[91] it is reported that the positivity of the operator associated to K is equivalent to the statement that the kernel K is positive definite, that is the matrix $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$ is positive definite for all choices of distinct points $\mathbf{x}_i \in X$. Notice that a kernel K could have an expansion of the form (27) in which the ϕ_n are not necessarily its eigenfunctions. The only requirement is that the ϕ_n are linearly independent but not necessarily orthogonal.
In the case that the space X has finite cardinality, the "functions" f are evaluated only at a finite number of points $\mathbf{x}$. If M is the cardinality of X, then the RKHS becomes an M-dimensional space where the functions f are basically M-dimensional vectors, the kernel K becomes an $M \times M$ matrix, and the condition that makes it a valid kernel is that it is a symmetric positive definite matrix (semi-definite if M is larger than the dimensionality of the RKHS). Positive definite matrices are known to be the ones which define dot products, i.e. $fKf^T \geq 0$ for every f in the RKHS. The space consists of all M-dimensional vectors f with finite norm fKf^T.
Summarizing, RKHS are Hilbert spaces where the dot product is defined using a function $K(\mathbf{x}, \mathbf{y})$ which needs to be positive definite just like in the case that X has finite cardinality. The elements of the RKHS are all functions f that have a finite norm given by equation (31). Notice the equivalence of a) choosing a specific RKHS $\mathcal{H}$ b) choosing a set of ϕ_n and λ_n c) choosing a reproducing kernel K. The last one is the most natural for most applications. A simple example of a RKHS is presented in Appendix B.
Finally, it is useful to notice that the solutions of the methods we discuss in this paper can be written both in the form (2), and in the form (28). Often in the literature formulation (2) is called the *dual* form of f, while (28) is called the *primal* form of f.

4 Regularization Networks

In this section we consider the approximation scheme that arises from the minimization of the quadratic functional

$$\min_{f \in \mathcal{H}} H[f] = \frac{1}{l} \sum_{i=1}^{l} (y_i - f(\mathbf{x}_i))^2 + \lambda \|f\|_K^2 \tag{32}$$

for a fixed λ. Formulations like equation (32) are a special form of regularization theory developed by Tikhonov, Ivanov [92, 46] and others to solve ill-posed problems and in particular to solve the problem of approximating the functional relation between $\mathbf{x}$ and y given a finite number of examples $D = \{\mathbf{x}_i, y_i\}_{i=1}^{l}$. As we mentioned in the previous sections our motivation in this paper is to use this formulation as an approximate implementation of Vapnik's SRM principle.
In classical regularization the data term is an L_2 loss function for the empirical risk, whereas the second term – called *stabilizer* – is usually written as a functional $\Omega(f)$ with certain properties [92, 69, 39]. Here we consider a special class of stabilizers, that is the norm $\|f\|_K^2$ in a RKHS induced by a symmetric, positive definite function $K(\mathbf{x}, \mathbf{y})$. This choice allows us to develop a framework of regularization which includes most of the usual regularization schemes. The only significant omission in this treatment – that we make here for simplicity – is the restriction on K to be symmetric positive definite so that the stabilizer is a norm. However, the theory can be extended without problems to the case in which K is positive semidefinite, in which case the stabilizer is a semi-norm [102, 56, 31, 33]. This approach was also sketched in [90].
The stabilizer in equation (32) effectively constrains f to be in the RKHS defined by K. It is possible to show (see for example [69, 39]) that the function that minimizes the functional (32)

has the form:

$$f(\mathbf{x}) = \sum_{i=1}^{l} c_i K(\mathbf{x}, \mathbf{x}_i), \tag{33}$$

where the coefficients c_i depend on the data and satisfy the following linear system of equations:

$$(K + \lambda I)\mathbf{c} = \mathbf{y} \tag{34}$$

where I is the identity matrix, and we have defined

$$(\mathbf{y})_i = y_i \ , \quad (\mathbf{c})_i = c_i \ , \quad (K)_{ij} = K(\mathbf{x}_i, \mathbf{x}_j).$$

It is remarkable that the solution of the more general case of

$$\min_{f \in \mathcal{H}} H[f] = \frac{1}{l} \sum_{i=1}^{l} V(y_i - f(\mathbf{x}_i)) + \lambda \|f\|_K^2, \tag{35}$$

where the function V is any differentiable function, is quite similar: the solution has exactly the same general form of (33), though the coefficients cannot be found anymore by solving a linear system of equations as in equation (34) [37, 40, 90]. For a proof see Appendix C.
The approximation scheme of equation (33) has a simple interpretation in terms of a network with one layer of hidden units [71, 39]. Using different kernels we get various RN's. A short list of examples is given in Table 1.

Kernel Function	Regularization Network
$K(\mathbf{x}-\mathbf{y}) = \exp(-\|\mathbf{x}-\mathbf{y}\|^2)$	Gaussian RBF
$K(\mathbf{x}-\mathbf{y}) = (\|\mathbf{x}-\mathbf{y}\|^2 + c^2)^{-\frac{1}{2}}$	Inverse Multiquadric
$K(\mathbf{x}-\mathbf{y}) = (\|\mathbf{x}-\mathbf{y}\|^2 + c^2)^{\frac{1}{2}}$	Multiquadric
$K(\mathbf{x}-\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|^{2n+1}$ $K(\mathbf{x}-\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|^{2n} \ln(\|\mathbf{x}-\mathbf{y}\|)$	Thin plate splines
$K(\mathbf{x}, \mathbf{y}) = \tanh(\mathbf{x} \cdot \mathbf{y} - \theta)$	(only for some values of θ) Multi Layer Perceptron
$K(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x} \cdot \mathbf{y})^d$	Polynomial of degree d
$K(x, y) = B_{2n+1}(x - y)$	B-splines
$K(x, y) = \frac{\sin(d+1/2)(x-y)}{\sin\frac{(x-y)}{2}}$	Trigonometric polynomial of degree d

Table 1: Some possible kernel functions. The first four are radial kernels. The multiquadric and thin plate splines are positive semidefinite and thus require an extension of the simple RKHS theory of this paper. The last three kernels were proposed by Vapnik [96], originally for SVM. The last two kernels are one-dimensional: multidimensional kernels can be built by tensor products of one-dimensional ones. The functions B_n are piecewise polynomials of degree n, whose exact definition can be found in [85].

When the kernel K is positive semidefinite, there is a subspace of functions f which have norm $\|f\|_K^2$ equal to zero. They form the null space of the functional $\|f\|_K^2$ and in this case the minimizer of (32) has the form [102]:

$$f(\mathbf{x}) = \sum_{i=1}^{l} c_i K(\mathbf{x}, \mathbf{x}_i) + \sum_{\alpha=1}^{k} b_\alpha \psi_\alpha(\mathbf{x}), \tag{36}$$

where $\{\psi_\alpha\}_{\alpha=1}^{k}$ is a basis in the null space of the stabilizer, which in most cases is a set of polynomials, and therefore will be referred to as the "polynomial term" in equation (36). The coefficients b_α and c_i depend on the data. For the classical regularization case of equation (32), the coefficients of equation (36) satisfy the following linear system:

$$(K + \lambda I)\mathbf{c} + \Psi^T \mathbf{b} = \mathbf{y}, \tag{37}$$

$$\Psi \mathbf{c} = 0, \tag{38}$$

where I is the identity matrix, and we have defined

$$(\mathbf{y})_i = y_i \ , \quad (\mathbf{c})_i = c_i \ , \quad (\mathbf{b})_i = b_i \ ,$$
$$(K)_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) \ , \quad (\Psi)_{\alpha i} = \psi_\alpha(\mathbf{x}_i).$$

When the kernel is positive definite, as in the case of the Gaussian, the null space of the stabilizer is empty. However, it is often convenient to redefine the kernel and the norm induced by it so that the induced RKHS contains only zero-mean functions, that is functions $f_1(\mathbf{x})$ s.t. $\int_X f_1(\mathbf{x})dx = 0$. In the case of a radial kernel K, for instance, this amounts to considering a new kernel

$$K'(\mathbf{x}, \mathbf{y}) = K(\mathbf{x}, \mathbf{y}) - \lambda_0$$

without the zeroth order Fourier component, and a norm

$$\|f\|_{K'}^2 = \sum_{n=1}^{\infty} \frac{a_n^2}{\lambda_n}. \tag{39}$$

The null space induced by the new K' is the space of constant functions. Then the minimizer of the corresponding functional (32) has the form:

$$f(\mathbf{x}) = \sum_{i=1}^{l} c_i K'(\mathbf{x}, \mathbf{x}_i) + b, \tag{40}$$

with the coefficients satisfying equations (37) and (38), that respectively become:

$$(K' + \lambda I)\mathbf{c} + \mathbf{1}b = (K - \lambda_0 I + \lambda I)\mathbf{c} + \mathbf{1}b = (K + (\lambda - \lambda_0)I)\mathbf{c} + \mathbf{1}b = \mathbf{y}, \tag{41}$$

$$\sum_{i=1}^{l} c_i = 0. \tag{42}$$

Equations (40) and (42) imply that the the minimizer of (32) is of the form:

$$f(\mathbf{x}) = \sum_{i=1}^{l} c_i K'(\mathbf{x}, \mathbf{x}_i) + b = \sum_{i=1}^{l} c_i (K(\mathbf{x}, \mathbf{x}_i) - \lambda_0) + b = \sum_{i=1}^{l} c_i K(\mathbf{x}, \mathbf{x}_i) + b. \tag{43}$$

Thus we can effectively use a positive definite K *and* the constant b, since the only change in equation (41) just amounts to the use of a different λ. Choosing to use a non-zero b effectively means choosing a different feature space and a different stabilizer from the usual case of equation (32): the constant feature is not considered in the RKHS norm and therefore is not "penalized". This choice is often quite reasonable, since in many regression and, especially, classification problems, shifts by a constant in f should not be penalized.

In summary, the argument of this section shows that using a RN of the form (43) (for a certain class of kernels K) is equivalent to minimizing functionals such as (32) or (35). The choice of K is equivalent to the choice of a corresponding RKHS and leads to various classical learning techniques such as RBF networks. We discuss connections between regularization and other techniques in sections 4.2 and 4.3.

Notice that in the framework we use here the kernels K are not required to be radial or even shift-invariant. Regularization techniques used to solve supervised learning problems [69, 39] were typically used with shift invariant stabilizers (tensor product and additive stabilizers are exceptions, see [39]). We now turn to such kernels.

4.1 Radial Basis Functions

Let us consider a special case of the kernel K of the RKHS, which is the standard case in several papers and books on regularization [102, 70, 39]: the case in which K is shift invariant, that is $K(\mathbf{x},\mathbf{y}) = K(\mathbf{x}-\mathbf{y})$ and the even more special case of a *radial* kernel $K(\mathbf{x},\mathbf{y}) = K(||\mathbf{x}-\mathbf{y}||)$. Section 3 implies that a radial positive definite K defines a RKHS in which the "features" ϕ_n are Fourier components that is

$$K(\mathbf{x},\mathbf{y}) \equiv \sum_{n=0}^{\infty} \lambda_n \phi_n(\mathbf{x})\phi_n(\mathbf{y}) \equiv \sum_{n=0}^{\infty} \lambda_n e^{i2\pi\mathbf{n}\cdot\mathbf{x}} e^{-i2\pi\mathbf{n}\cdot\mathbf{y}}. \tag{44}$$

Thus any positive definite radial kernel defines a RKHS over $[0,1]$ with a scalar product of the form:

$$< f, g >_{\mathcal{H}} \equiv \sum_{n=0}^{\infty} \frac{\tilde{f}(\mathbf{n})\tilde{g}^*(\mathbf{n})}{\lambda_n}, \tag{45}$$

where $\tilde{f}$ is the Fourier transform of f. The RKHS becomes simply the subspace of $L_2([0,1]^d)$ of the functions such that

$$\|f\|_K^2 = \sum_{n=1}^{\infty} \frac{|\tilde{f}(\mathbf{n})|^2}{\lambda_n} < +\infty. \tag{46}$$

Functionals of the form (46) are known to be *smoothness* functionals. In fact, the rate of decrease to zero of the Fourier transform of the kernel will control the smoothness property of the function in the RKHS. For radial kernels the minimizer of equation (32) becomes:

$$f(\mathbf{x}) = \sum_{i=1}^{l} c_i K(||\mathbf{x}-\mathbf{x}_i||) + b \tag{47}$$

and the corresponding RN is a *Radial Basis Function Network*. Thus Radial Basis Function networks are a special case of RN [69, 39].

In fact *all* translation-invariant stabilizers $K(\mathbf{x},\mathbf{x}_i) = K(\mathbf{x}-\mathbf{x}_i)$ correspond to RKHS's where the basis functions ϕ_n are Fourier eigenfunctions and only differ in the spectrum of the eigenvalues (for a Gaussian stabilizer the spectrum is Gaussian, that is $\lambda_n = Ae^{(-n^2/2)}$ (for $\sigma = 1$)). For

example, if $\lambda_n = 0$ for all $n > n_0$, the corresponding RKHS consists of all bandlimited functions, that is functions with zero Fourier components at frequencies higher than n_0[18]. Generally λ_n are such that they decrease as n increases, therefore restricting the class of functions to be functions with decreasing high frequency Fourier components.
In classical regularization with translation invariant stabilizers and associated kernels, the common experience, often reported in the literature, is that the form of the kernel does not matter much. We conjecture that this may be because all translation invariant K induce the same type of ϕ_n features - the Fourier basis functions.

4.2 Regularization, generalized splines and kernel smoothers

A number of approximation and learning techniques can be studied in the framework of regularization theory and RKHS. For instance, starting from a reproducing kernel it is easy [5] to construct kernels that correspond to tensor products of the original RKHS; it is also easy to construct the additive sum of several RKHS in terms of a reproducing kernel.

- **Tensor Product Splines:** In the particular case that the kernel is of the form:

  $$K(\mathbf{x}, \mathbf{y}) = \Pi_{j=1}^{d} k(x^j, y^j)$$

  where x^j is the jth coordinate of vector $\mathbf{x}$ and k is a positive definite function with one-dimensional input vectors, the solution of the regularization problem becomes:

  $$f(\mathbf{x}) = \sum_i c_i \Pi_{j=1}^{d} k(x_i^j, x^j)$$

  Therefore we can get tensor product splines by choosing kernels of the form above [5].

- **Additive Splines:** In the particular case that the kernel is of the form:

  $$K(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^{d} k(x^j, y^j)$$

  where x^j is the jth coordinate of vector $\mathbf{x}$ and k is a positive definite function with one-dimensional input vectors, the solution of the regularization problem becomes:

  $$f(\mathbf{x}) = \sum_i c_i (\sum_{j=1}^{d} k(x_i^j, x^j)) = \sum_{j=1}^{d} (\sum_i c_i k(x_i^j, x^j)) = \sum_{j=1}^{d} f_j(x^j)$$

  So in this particular case we get the class of *additive approximation* schemes of the form:

  $$f(\mathbf{x}) = \sum_{j=1}^{d} f_j(x^j)$$

A more extensive discussion on relations between known approximation methods and regularization can be found in [39].

[18]The simplest K is then $K(x, y) = sinc(x - y)$, or kernels that are convolution with it.

4.3 Dual representation of Regularization Networks

Every RN can be written as

$$f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{K}(\mathbf{x}) \tag{48}$$

where $\mathbf{K}(\mathbf{x})$ is the vector of functions such that $(\mathbf{K}(\mathbf{x}))_i = K(\mathbf{x}, \mathbf{x}_i)$. Since the coefficients $\mathbf{c}$ satisfy the equation (34), equation (48) becomes

$$f(\mathbf{x}) = (K + \lambda I)^{-1}\mathbf{y} \cdot \mathbf{K}(\mathbf{x}) \ .$$

We can rewrite this expression as

$$f(\mathbf{x}) = \sum_{i=1}^{l} y_i b_i(\mathbf{x}) = \mathbf{y} \cdot \mathbf{b}(\mathbf{x}) \tag{49}$$

in which the vector $\mathbf{b}(\mathbf{x})$ of basis functions is defined as:

$$\mathbf{b}(\mathbf{x}) = (K + \lambda I)^{-1}\mathbf{K}(\mathbf{x}) \tag{50}$$

and now depends on all the data points and on the regularization parameter λ. The representation (49) of the solution of the approximation problem is known as the *dual*[19] of equation (48), and the basis functions $b_i(\mathbf{x})$ are called the *equivalent kernels*, because of the similarity with the kernel smoothing technique [88, 41, 43]. Notice that, while in equation (48) the difficult part is the computation of coefficients c_i, the kernel function $K(\mathbf{x}, \mathbf{x}_i)$ being predefined, in the dual representation (49) the difficult part is the computation of the basis function $b_i(\mathbf{x})$, the coefficients of the expansion being explicitly given by the y_i.
As observed in [39], the dual representation of a RN shows clearly how careful one should be in distinguishing between local vs. global approximation techniques. In fact, we expect (see [88] for the 1-D case) that in most cases the kernels $b_i(\mathbf{x})$ decrease with the distance of the data points $\mathbf{x}_i$ from the evaluation point, so that only the neighboring data affect the estimate of the function at $\mathbf{x}$, providing therefore a "local" approximation scheme. Even if the original kernel K is not "local", like the absolute value $|x|$ in the one-dimensional case or the multiquadric $K(\mathbf{x}) = \sqrt{1 + \|\mathbf{x}\|^2}$, the basis functions $b_i(\mathbf{x})$ are bell shaped, local functions, whose locality will depend on the choice of the kernel K, on the density of data points, and on the regularization parameter λ. This shows that apparently "global" approximation schemes can be regarded as local, *memory-based* techniques (see equation 49) [59].

4.4 From regression to classification

So far we only considered the case that the unknown function can take any real values, specifically the case of regression. In the particular case that the unknown function takes only two values, i.e. -1 and 1, we have the problem of binary pattern classification, i.e. the case where we are given data that belong to one of two classes (classes -1 and 1) and we want to find a function that separates these classes. It can be shown [28] that, if V in equation (35) is $(y - f(\mathbf{x}))^2$, and if K defines a finite dimensional RKHS, then the minimizer of the equation

[19] Notice that this "duality" is different from the one mentioned at the end of section 3.

$$H[f] = \frac{1}{l}\sum_{i=1}^{l}(f(\mathbf{x}_i) - y_i)^2 + \lambda\|f\|_K^2, \tag{51}$$

for $\lambda \to 0$ approaches *asymptotically* the function in the RKHS that is closest in the L_2 norm to the regression function:

$$f_0(\mathbf{x}) = P(y = 1|\mathbf{x}) - P(y = -1|\mathbf{x}) \tag{52}$$

The *optimal Bayes rule classifier* is given by thresholding the regression function, i.e. by $\text{sign}(f_0(\mathbf{x}))$. Notice that in the case of infinite dimensional RKHS asymptotic results ensuring consistency are available (see [27], theorem 29.8) but depend on several conditions that are not automatically satisfied in the case we are considering. The Bayes classifier is the best classifier, given the correct probability distribution P. However, approximating function (52) in the RKHS in L_2 does not necessarily imply that we find the best approximation to the Bayes classifier. For classification, only the sign of the regression function matters and not the exact value of it. Notice that an approximation of the regression function using a mean square error criterion places more emphasis on the most probable data points and not on the most "important" ones which are the ones near the separating boundary.
In the next section we will study Vapnik's more natural approach to the problem of classification that is based on choosing a loss function V different from the square error. This approach leads to solutions that emphasize data points near the separating surface.

5 Support vector machines

In this section we discuss the technique of Support Vector Machines (SVM) for Regression (SVMR) [95, 96] in terms of the SVM functional. We will characterize the form of the solution and then show that SVM for binary pattern classification can be derived as a special case of the regression formulation.

5.1 SVM in RKHS

Once again the problem is to learn a functional relation between $\mathbf{x}$ and y given a finite number of examples $D = \{\mathbf{x}_i, y_i\}_{i=1}^l$.
The method of SVMR [96] corresponds to the following functional

$$H[f] = \frac{1}{l}\sum_{i=1}^{l}|y_i - f(\mathbf{x}_i)|_\epsilon + \lambda\|f\|_K^2 \tag{53}$$

which is a special case of equation (35) and where

$$V(x) = |x|_\epsilon \equiv \begin{cases} 0 & \text{if } |x| < \epsilon \\ |x| - \epsilon & \text{otherwise,} \end{cases} \tag{54}$$

is the $\epsilon-$Insensitive Loss Function (ILF) (also noted with L_ϵ). Note that the ILF assigns zero cost to errors smaller then ϵ. In other words, for the cost function $|\cdot|_\epsilon$ any function closer than ϵ to the data points is a perfect interpolant. We can think of the parameter ϵ as the resolution at

which we want to look the data. For this reason we expect that the larger ϵ is, the simpler the representation will be. We will come back to this point in section 8.
The minimizer of H in the RKHS $\mathcal{H}$ defined by the kernel K has the general form given by equation (43), that is

$$f(x) = \sum_{i=1}^{l} c_i K(\mathbf{x}_i, \mathbf{x}) + b, \tag{55}$$

where we can include the constant b for the same reasons discussed in section 4.
In order to find the solution of SVM we have to minimize functional (53) (with V given by equation (54)) with respect to f. Since it is difficult to deal with the function $V(x) = |x|_\epsilon$, the above problem is replaced by the following equivalent problem (by *equivalent* we mean that the same function minimizes both functionals), in which an additional set of variables is introduced:

Problem 5.1

$$\min_{f,\xi,xi^*} \Phi(f, \xi, \xi^*) = \frac{C}{l}\sum_{i=1}^{l}(\xi_i + \xi_i^*) + \frac{1}{2}\|f\|_K^2 \tag{56}$$

subject to the constraints:

$$\begin{array}{lcll} f(\mathbf{x}_i) - y_i & \leq & \epsilon + \xi_i & i = 1, \ldots, l \\ y_i - f(\mathbf{x}_i) & \leq & \epsilon + \xi_i^* & i = 1, \ldots, l \\ \xi_i, \xi_i^* & \geq & 0 & i = 1, \ldots, l. \end{array} \tag{57}$$

The parameter C in (56) has been introduced in order to be consistent with the standard SVM notations [96]. Note that λ in eq. (53) corresponds to $\frac{1}{2C}$. The equivalence is established just noticing that in problem (5.1) a (linear) penalty is paid only when the absolute value of the error exceeds ϵ, (which correspond to the Vapnik's ILF). Notice that if either of the two top constraints is satisfied with some non-zero ξ_i (or ξ_i^*), the other is automatically satisfied with a zero value for ξ_i^* (or ξ_i).
Problem (5.1) can be solved through the technique of Lagrange multipliers. For details see [96]. The result is that the function which solves problem (5.1) can be written as:

$$f(\mathbf{x}) = \sum_{i=1}^{l}(\alpha_i^* - \alpha_i)K(\mathbf{x}_i, \mathbf{x}) + b,$$

where α_i^* and α_i are the solution of the following QP-problem:

Problem 5.2

$$\min_{\boldsymbol{\alpha},\boldsymbol{\alpha}^*} \mathcal{W}(\boldsymbol{\alpha}, \boldsymbol{\alpha}^*) = \epsilon\sum_{i=1}^{l}(\alpha_i^* + \alpha_i) - \sum_{i=1}^{l} y_i(\alpha_i^* - \alpha_i) + \frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(\mathbf{x}_i, \mathbf{x}_j),$$

subject to the constraints:

$$\sum_{i=1}^{l}(\alpha_i^* - \alpha_i) = 0,$$

$$0 \leq \alpha_i^*, \alpha_i \leq \frac{C}{l}, \qquad i = 1, \ldots, l.$$

The solutions of problems (5.1) and (5.2) are related by the Kuhn-Tucker conditions:

$$\alpha_i(f(\mathbf{x}_i) - y_i - \epsilon - \xi_i) = 0 \qquad i = 1, \ldots, l \tag{58}$$
$$\alpha_i^*(y_i - f(\mathbf{x}_i) - \epsilon - \xi_i^*) = 0 \qquad i = 1, \ldots, l \tag{59}$$
$$(\frac{C}{l} - \alpha_i)\xi_i = 0 \qquad i = 1, \ldots, l \tag{60}$$
$$(\frac{C}{l} - \alpha_i^*)\xi_i^* = 0 \qquad i = 1, \ldots, l. \tag{61}$$

The input data points $\mathbf{x}_i$ for which α_i or α_i^* are different from zero are called *support vectors* (SVs). Observe that α_i and α_i^* cannot be simultaneously different from zero, so that the constraint $\alpha_i\alpha_i^* = 0$ holds true. Any of the SVs for which $0 < \alpha_j < \frac{C}{l}$ (and therefore $\xi_j = 0$) can be used to compute the parameter b. In fact, in this case it follows from the Kuhn-Tucker conditions that:

$$f(\mathbf{x}_j) = \sum_{i=1}^{l}(\alpha_i^* - \alpha_i)K(\mathbf{x}_i, \mathbf{x}_j) + b = y_j + \epsilon.$$

from which b can be computed. The SVs are those data points $\mathbf{x}_i$ at which the error is either greater or equal to ϵ[20]. Points at which the error is smaller than ϵ are never support vectors, and do not enter in the determination of the solution. A consequence of this fact is that if the SVM were run again on the new data set consisting of only the SVs the same solution would be found. Finally observe that if we call $c_i = \alpha_i^* - \alpha_i$, we recover equation (55). With respect to the new variable c_i problem (5.2) becomes:

Problem 5.3

$$\min_{\mathbf{c}} E[\mathbf{c}] = \frac{1}{2}\sum_{i,j=1}^{l} c_i c_j K(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^{l} c_i y_i + \epsilon \sum_{i=1}^{l} |c_i|$$

subject to the constraints

$$\sum_{i=1}^{l} c_i = 0,$$
$$-\frac{C}{l} \leq c_i \leq \frac{C}{l}, \qquad i = 1, \ldots, l.$$

This different formulation of SVM will be useful in section 8 when we will describe the relation between SVM and sparse approximation techniques.

5.2 From regression to classification

In the previous section we discussed the connection between regression and classification in the framework of regularization. In this section, after stating the formulation of SVM for binary pattern classification (SVMC) as developed by Cortes and Vapnik [22], we discuss a connection

[20] In degenerate cases however, it can happen that points whose error is equal to ϵ are not SVs.

between SVMC and SVMR. We will not discuss the theory of SVMC here; we refer the reader to [96]. We point out that the SVM technique has first been proposed for binary pattern classification problems and then extended to the general regression problem [95]. Here our primary focus is regression and we consider classification as a special case of regression.
SVMC can be formulated as the problem of minimizing:

$$H(f) = \frac{1}{l}\sum_{i}^{l} |1 - y_i f(\mathbf{x}_i)|_+ + \frac{1}{2C}\|f\|_K^2, \tag{62}$$

which is again of the form (3). Using the fact that $y_i \in \{-1, +1\}$ it is easy to see that our formulation (equation (62)) is equivalent to the following quadratic programming problem, originally proposed by Cortes and Vapnik [22]:

Problem 5.4

$$\min_{f\in\mathcal{H},\boldsymbol{\xi}} \Phi(f,\boldsymbol{\xi}) = \frac{C}{l}\sum_{i=1}^{l}\xi_i + \frac{1}{2}\|f\|_K^2$$

subject to the constraints:

$$\begin{array}{lll} y_i f(\mathbf{x}_i) & \geq 1 - \xi_i, & i = 1, \ldots, l \\ \xi_i & \geq 0, & i = 1, \ldots, l. \end{array} \tag{63}$$

The solution of this problem is again of the form:

$$f(\mathbf{x}) = \sum_{i=1}^{l} \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b, \tag{64}$$

where it turns out that $0 \leq \alpha_i \leq \frac{C}{l}$. The input data points $\mathbf{x}_i$ for which α_i is different from zero are called, as in the case of regression, *support vectors* (SVs). It is often possible to write the solution $f(\mathbf{x})$ as a linear combination of SVs in a number of different ways (for example in case that the feature space induced by the kernel K has dimensionality lower than the number of SVs). The SVs that appear in *all* these linear combinations are called *essential support vectors.* Roughly speaking the motivation for problem (5.4) is to minimize the empirical error measured by $\sum_{i=1}^{l}\xi_i$[21] while controlling capacity measured in terms of the norm of f in the RKHS. In fact, the norm of f is related to the notion of *margin*, an important idea for SVMC for which we refer the reader to [96, 15].
We now address the following question: what happens if we apply the SVMR formulation given by problem (5.1) to the binary pattern classification case, i.e. the case where y_i take values $\{-1, 1\}$, treating classification as a regression on binary data?
Notice that in problem (5.1) each example has to satisfy two inequalities (which come out of using the ILF), while in problem (5.4) each example has to satisfy one inequality. It is possible to show that for a given constant C in problem (5.4), there exist C and ϵ in problem (5.1) such that the solutions of the two problems are the same, up to a constant factor. This is summarized in the following theorem:

[21] As we mentioned in section 2, for binary pattern classification the empirical error is defined as a sum of binary numbers which in problem (5.4) would correspond to $\sum_{i=1}^{l}\theta(\xi_i)$. However in such a case the minimization problem becomes computationally intractable. This is why in practice in the cost functional $\Phi(f,\boldsymbol{\xi})$ we approximate $\theta(\xi_i)$ with ξ_i. We discuss this further in section 6.

Theorem 5.1 *Suppose the classification problem (5.4) is solved with parameter C, and the optimal solution is found to be f. Then, there exists a value $a \in (0,1)$ such that for $\forall \epsilon \in [a, 1)$, if the regression problem (5.1) is solved with parameter $(1-\epsilon)C$, the optimal solution will be $(1-\epsilon)f$.*

We refer to [76] for the proof. A sketch of the proof is given in Appendix D. A direct implication of this result is that one can solve any SVMC problem through the SVMR formulation. A formal proof of this result can also be given in the framework of SRM as discussed in Appendix D. It is an open question what theoretical implications theorem 5.1 may have about SVMC and SVMR. In particular in section 6 we will discuss some recent theoretical results on SVMC that have not yet been extended to SVMR. It is possible that theorem 5.1 may help to extend them to SVMR.

6 SRM for RNs and SVMs

At the end of section 2 we outlined how one should implement both RN and SVM according to SRM. To use the standard SRM method we first need to know the VC-dimension of the hypothesis spaces we use. In sections 4 and 5 we saw that both RN and SVM use as hypothesis spaces sets of bounded functions f in a RKHS with $||f||_K$ bounded (i.e. $||f||_K \leq A$), where k is the kernel of the RKHS. Thus, in order to use the standard SRM method outlined in section 2 we need to know the VC dimension of such spaces under the loss functions of RN and SVM.
Unfortunately it can be shown that when the loss function V is $(y - f(\mathbf{x}))^2$ (L_2) and also when it is $|y_i - f(\mathbf{x}_i)|_\epsilon$ (L_ϵ), the VC-dimension of $V(y, f(\mathbf{x}))$ with f in $H_A = \{f : ||f||_K \leq A\}$ does not depend on A, and is infinite if the RKHS is infinite dimensional. More precisely we have the following theorem (for a proof see for example [103, 36], or for an outline of the proof see Appendix E):

Theorem 6.1 *Let N be the dimensionality of a RKHS $\mathcal{R}$. For both the L_2 and the ϵ-insensitive loss function V, the VC-dimension of V in the space $H_A = \{f \in \mathcal{R} : ||f||_K \leq A\}$ is $O(N)$, independently of A. Moreover, if N is infinite, the VC-dimension is infinite for any positive A.*

It is thus impossible to use SRM with this kind of hypothesis spaces: in the case of finite dimensional RKHS, the RKHS norm of f cannot be used to define a structure of spaces with different VC-dimensions, and in the (typical) case that the dimensionality of the RKHS is infinite, it is not even possible to use bound (16). So *the VC-dimension cannot be used directly neither for RN nor for SVMR.*
On the other hand, we can still use the V_γ dimension and the extended SRM method outlined in section 2. Again we need to know the V_γ dimension of our loss function V in the space H_A defined above. In the typical case that the input space X is bounded, the V_γ dimension depends on A and is not infinite in the case of infinite dimensional RKHS. More precisely the following theorem holds (for a proof see [36]):

Theorem 6.2 *Let N be the dimensionality of a RKHS $\mathcal{R}$ with kernel K. Assume our input space X is bounded and let R be the radius of the smallest ball B containing the data $\mathbf{x}$ in the feature space induced by kernel K. The V_γ dimension h for regression using L_2 or L_ϵ loss functions for hypothesis spaces $H_A = \{f \in \mathcal{R} \mid \|f\|_K \leq A\}$ and y bounded, is finite for $\forall\ \gamma > 0$, with $h \leq O(\min\ (N, \frac{(R^2+1)(A^2+1)}{\gamma^2}))$.*

Notice that for fixed γ and fixed radius of the data the only variable that controls the V_γ dimension is the upper bound on the RKHS norm of the functions, namely A. Moreover, the V_γ dimension is finite for $\forall\ \gamma > 0$; therefore, according to theorem (2.3), ERM uniformly converges in H_A for any $A < \infty$, both for RN and for SVMR. Thus both RNs and SVMR are *consistent* in H_A for any $A < \infty$. Theoretically, we can use the extended SRM method with a sequence of hypothesis spaces H_A each defined for different As. To repeat, for a fixed $\gamma > 0$ (we can let γ go to 0 as $l \to \infty$) we first define a structure $H_1 \subset H_2 \subset \ldots \subset H_{n(l)}$ where H_m is the set of bounded functions f in a RKHS with $||f||_K \leq A_m$, $A_m < \infty$, and the numbers A_m form an increasing sequence. Then we minimize the empirical risk in each H_m by solving the problem:

$$\begin{aligned} &\text{minimize } \frac{1}{l}\sum_{i=1}^{l} V(y_i, f(\mathbf{x}_i)) \\ &\text{subject to : } ||f||_K \leq A_m \end{aligned} \tag{65}$$

To solve this minimization problem we minimize

$$\frac{1}{l}\sum_{i=1}^{l} V(y_i, f(\mathbf{x}_i)) + \lambda_m(||f||_K^2 - A_m^2) \tag{66}$$

with respect to f and maximize with respect to the Lagrange multiplier λ_m. If f_m is the solution of this problem, at the end we choose the optimal $f_{n^*(l)}$ in $F_{n^*(l)}$ with the associated $\lambda_{n^*(l)}$, where optimality is decided based on a trade off between empirical error and the bound (21) for the fixed γ (which, as we mentioned, can approach zero). In the case of RN, V is the L_2 loss function, whereas in the case of SVMR it is the ϵ-insensitive loss function.

In practice it is difficult to implement the extended SRM for two main reasons. First, as we discussed in section 2, SRM using the V_γ dimension is practically difficult because we do not have tight bounds to use in order to pick the optimal $F_{n^*(l)}$ (combining theorems 6.2 and 2.4, bounds on the expected risk of RN and SVMR machines of the form (65) can be derived, but these bounds are not practically useful). Second, even if we could make a choice of $F_{n^*(l)}$, it is computationally difficult to implement SRM since (65) is a constrained minimization problem one with non-linear constraints, and solving such a problem for a number of spaces H_m can be computationally difficult. So implementing SRM using the V_γ dimension of nested subspaces of a RKHS is practically a very difficult problem.

On the other hand, if we had the optimal Lagrange multiplier $\lambda_{n^*(l)}$, we could simply solve the unconstrained minimization problem:

$$\frac{1}{l}\sum_{i=1}^{l} V(y_i, f(\mathbf{x}_i)) + \lambda_{n^*(l)}||f||_K \tag{67}$$

both for RN and for SVMR. This is exactly the problem we solve in practice, as we described in sections 4 and 5. Since the value $\lambda_{n^*(l)}$ is not known in practice, we can only "implement" the extended SRM approximately by minimizing (67) with various values of λ and then picking the best λ using techniques such as cross-validation [1, 100, 101, 49], Generalized Cross Validation, Finite Prediction Error and the MDL criteria (see [96] for a review and comparison).

Summarizing, both the RN and the SVMR methods discussed in sections 4 and 5 can be seen as approximations of the extended SRM method using the V_γ dimension, with nested hypothesis spaces being of the form $H_A = \{f \in \mathcal{R} : ||f||_K \leq A\}$, $\mathcal{R}$ being a RKHS defined by kernel K.

For both RN and SVMR the V_γ dimension of the loss function V in H_A is finite for $\forall\ \gamma > 0$, so the ERM method uniformly converges in H_A for any $A < \infty$, and we can use the extended SRM method outlined in section 2.

6.1 SRM for SVM Classification

It is interesting to notice that the same analysis can be used for the problem of classification. In this case the following theorem holds [35]:

Theorem 6.3 *Let N be the dimensionality of a RKHS $\mathcal{R}$ with kernel K. Assume our input space X is bounded and let R be the radius of the sphere where our data $\mathbf{x}$ belong to, in the feature space induced by kernel K. The V_γ dimension of the soft margin loss function $|1 - yf(\mathbf{x})|_+$ in $H_A = \{f \in \mathcal{R} : ||f||_K \leq A\}$ is $\leq O(min(N, \frac{R^2A^2}{\gamma^2}))$. In the case that N is infinite the V_γ dimension becomes $\leq O(\frac{R^2A^2}{\gamma^2})$, which is finite for $\forall\ \gamma > 0$.*

This theorem, combined with the theorems on V_γ dimension summarized in section 2, can be used for a distribution *independent* analysis of SVMC (of the form (65)) like that of SVMR and RN. However, a direct application of theorems 6.3 and 2.4 leads to a bound on the expected soft margin error of the SVMC solution, instead of a more interesting bound on the expected *misclassification* error. We can bound the expected *misclassification* error as follows.
Using theorem 2.4 with the soft margin loss function we can get a bound on the expected soft margin loss in terms of the empirical one (the $\sum_{i=1}^{l} \xi_i$ of problem 5.4) and the V_γ dimension given by theorem 6.3. In particular theorem 2.4 implies:

$$Pr\left\{\sup_{f \in H_A} |I_{\text{emp}}[f;l] - I[f]|\ \epsilon >\right\} \leq \mathcal{G}(\epsilon, m, h_\gamma), \tag{68}$$

where both the expected and the empirical errors are measured using the soft margin loss function, and h_γ is the V_γ dimension of theorem 6.3 for $\gamma = \alpha\epsilon$ and α as in theorem 2.4. On the other hand, $\theta(-yf(\mathbf{x})) \leq |1 - yf(\mathbf{x})|_+$ for $\forall\ (\mathbf{x}, y)$, which implies that the expected misclassification error is less than the expected soft margin error. Inequality (68) implies that (uniformly) for all $f \in H_A$:

$$Pr\left\{I[f] > \epsilon + I_{\text{emp}}[f;l]\right\} \leq \mathcal{G}(\epsilon, m, h_\gamma), \tag{69}$$

Notice that (69) is different from existing bounds that use the empirical hard margin ($\theta(1 - yf(\mathbf{x}))$) error [8]. It is similar in spirit to bounds in [87] where the $\sum_{i=1}^{l} \xi_i^2$ is used[22]. On the other hand, it can be shown [35] that the V_γ dimension for loss functions of the form $|1 - yf(\mathbf{x})|_+^\sigma$ is of the form $O(\frac{R^2A^2}{\gamma^{\frac{2}{\sigma}}})$ for $\forall\ 0 < \sigma \leq 1$. Thus, using the same approach outlined above for the soft margin, we can get bounds on the misclassification error of SVMC in terms of $\sum_{i=1}^{l}(\xi_i)^\sigma$, which, for σ near 0, is close to the margin error used in [8] (for more information we refer the reader to [35]). It is important to point out that bounds like (69) hold only for the machines of the form (65), and not for the machines of the form (3) typically used in practice [35]. This is unlike the bound in [8] which holds for machines of the form (65) and is derived using the theoretical

[22] The $\sum_{i=1}^{l} \xi_i$ can be very different from the hard margin (or the misclassification) error. This may lead to various pathological situations (see for example [80]).

results of [6] where a type of "continuous" SRM (for example for a structure of hypothesis spaces defined through the continuous parameter A of (65)) is studied[23].
In the case of classification the difficulty is the minimization of the empirical misclassification error. Notice that SVMC does *not* minimize the misclassification error, and instead minimizes the empirical error using the soft margin loss function. One can use the SRM method with the soft margin loss function (6), in which case minimizing the empirical risk is possible. The SRM method with the soft margin loss function would be consistent, but the misclassification error of the solution may not be minimal. It is unclear whether SVMC is consistent in terms of misclassification error. In fact the V_γ dimension of the misclassification loss function (which is the same as the VC-dimension - see section 2) is known to be equal to the dimensionality of the RKHS plus one [96]. This implies that, as discussed at the beginning of this section, it cannot be used to study the expected misclassification error of SVMC in terms of the empirical one.

6.1.1 Distribution dependent bounds for SVMC

We close this section with a brief reference to a recent distribution *dependent* result on the generalization error of SVMC. This result does not use the V_γ or VC dimensions, which, as we mentioned in section 2, are used only for distribution *independent* analysis. It also leads to bounds on the performance of SVMC that (unlike the distribution independent ones) can be useful in practice[24].
For a given training set of size l, let us define SV_l to be the number of essential support vectors of SVMC, (as we defined them in section 5 - see eq. (64)). Let R_l be the radius of the smallest hypersphere in the feature space induced by kernel K containing all essential SVs, $\|f\|_K^2(l)$ the norm of the solution of SVMC, and $\rho(l) = \frac{1}{\|f\|_K^2(l)}$ the margin. Then for a fixed kernel and for a fixed value of the SVMC parameter C the following theorem holds:

Theorem 6.4 *(Vapnik, 1998) The expected misclassification risk of the SVM trained on m data points sampled from $X \times Y$ according to a probability distribution $P(\mathbf{x}, y)$ is bounded by:*

$$E\left\{\frac{\min\left(SV_{l+1}, \quad \frac{R_{l+1}^2}{\rho(l+1)}\right)}{l+1}\right\}$$

where the expectation E is taken over $P(\mathbf{x}, y)$.

This theorem can also be used to justify the current formulation of SVMC, since minimizing $||f||_K^2(l)$ (which is what we do in SVMR) affects the bound of theorem (6.4). It is an open question whether the bound of (6.4) can be used to construct learning machines that are better than current SVM. The theorem suggests that a learning machine should, instead of only minimizing $||f||_K^2$, minimize $\min\left(SV_l, \ \frac{R_{l+1}^2}{\rho(l+1)}\right)$. Finally, it is an open question whether similar results exist for the case of SVMR. As we mentioned in section 5, the connection between SVMC and SVMR outlined in that section may suggest how to extend such results to SVMR. The problem of finding better distribution dependent results on the generalization capabilities of SVM is a topic of current research which may lead to better learning machines.

[23] All these bounds are not tight enough in practice.

[24] Further distribution dependent results have been derived recently - see [47, 16, 34].

7 A Bayesian Interpretation of Regularization and SRM?

7.1 Maximum A Posteriori Interpretation of Regularization

It is well known that a variational principle of the type of equation (1) can be derived not only in the context of functional analysis [92], but also in a probabilistic framework [51, 102, 100, 73, 58, 11]. In this section we illustrate this connection for both RN and SVM, in the setting of RKHS. Consider the classical regularization case

$$\min_{f\in\mathcal{H}} H[f] = \frac{1}{l}\sum_{i=1}^{l}(y_i - f(\mathbf{x}_i))^2 + \lambda\|f\|_K^2 \tag{70}$$

Following Girosi et al. [39] let us define:

1. $D_l = \{(\mathbf{x}_i, y_i)\}$ for $i = 1, \cdots, l$ to be the set of training examples, as in the previous sections.
2. $\mathcal{P}[f|D_l]$ as the conditional probability of the function f given the examples D_l.
3. $\mathcal{P}[D_l|f]$ as the conditional probability of D_l given f. If the function underlying the data is f, this is the probability that by random sampling the function f at the sites $\{\mathbf{x}_i\}_{i=1}^l$ the set of measurement $\{y_i\}_{i=1}^l$ is obtained. This is therefore a model of the noise.
4. $\mathcal{P}[f]$: is the *a priori* probability of the random field f. This embodies our *a priori* knowledge of the function, and can be used to impose constraints on the model, assigning significant probability only to those functions that satisfy those constraints.

Assuming that the probability distributions $\mathcal{P}[D_l|f]$ and $\mathcal{P}[f]$ are known, the posterior distribution $\mathcal{P}[f|D_l]$ can now be computed by applying the Bayes rule:

$$\mathcal{P}[f|D_l] \propto \mathcal{P}[D_l|f]\ \mathcal{P}[f]. \tag{71}$$

If the noise is normally distributed with variance σ, then the probability $\mathcal{P}[D_l|f]$ can be written as:

$$\mathcal{P}[D_l|f] \propto e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{l}(y_i - f(\mathbf{x}_i))^2} .$$

For now let us write informally the prior probability $\mathcal{P}[f]$ as

$$\mathcal{P}[f] \propto e^{-\|f\|_K^2} . \tag{72}$$

Following the Bayes rule (71) the *a posteriori* probability of f is written as

$$\mathcal{P}[f|D_l] \propto e^{-[\frac{1}{2\sigma^2}\sum_{i=1}^{l}(y_i - f(\mathbf{x}_i))^2 + \|f\|_K^2]} . \tag{73}$$

One of the several possible estimates [58] of the function f from the probability distribution (73) is the so called MAP (*Maximum A Posteriori*) estimate, that considers the function that maximizes the *a posteriori* probability $\mathcal{P}[f|D_l]$, and therefore minimizes the exponent in equation (73). The MAP estimate of f is therefore the minimizer of the functional:

$$\frac{1}{l}\sum_{i=1}^{l}(y_i - f(\mathbf{x}_i))^2 + \frac{1}{l}\alpha\|f\|_K^2$$

where α is the *a priori* defined constant $2\sigma^2$, that is

$$\frac{1}{l}\sum_{i=1}^{l}(y_i - f(\mathbf{x}_i))^2 + \tilde{\lambda}\|f\|_K^2 .$$

where $\tilde{\lambda} = \frac{\alpha}{l}$. This functional is the same as that of equation (70), but here it is important to notice that $\lambda(l) = \frac{\alpha}{l}$. As noticed by Girosi et al. [39], functionals of the type (72) are common in statistical physics [67], where the stabilizer (here $\|f\|_K^2$) plays the role of an energy functional. As we will see later, the RKHS setting we use in this paper makes clear that the correlation function of the physical system described by $\|f\|_K^2$ is the kernel $K(\mathbf{x},\mathbf{y})$[25].

Thus in the standard MAP interpretation of RN the data term is a model of the noise and the stabilizer is a prior on the regression function f. The informal argument outlined above can be made formally precise in the setting of this paper in which the stabilizer is a norm in a RKHS (see also [102]). To see the argument in more detail, let us write the prior (72) as:

$$P[f] \propto e^{-\|f\|_K^2} = e^{-\sum_{n=1}^{M}\frac{a_n^2}{\lambda_n}}$$

where M is the dimensionality of the RKHS, with possibly $M=\infty$. Of course functions f can be represented as vectors $\mathbf{a}$ in the reference system of the eigenfunctions ϕ_n of the kernel K since

$$f(\mathbf{x}) = \sum_{n=1}^{M} a_n\phi_n(\mathbf{x}) . \tag{74}$$

The stabilizer

$$\|f\|_K^2 = \sum_{n=1}^{M}\frac{a_n^2}{\lambda_n} = \mathbf{a}^T\Lambda^{-1}\mathbf{a}$$

can of course be also expressed in any other reference system ($\phi' = A\phi$) as

$$\|f\|_K^2 = \mathbf{b}^T\Sigma^{-1}\mathbf{b}$$

which suggests that Σ can be interpreted as the covariance matrix in the reference system of the ϕ'. It is clear in this setting that the stabilizer can be regarded as the Malahanobis distance of f from the mean of the functions. $P[f]$ is therefore a multivariate Gaussian with zero mean in the Hilbert space of functions defined by K and spanned by the ϕ_n:

$$P[f] \propto e^{-\|f\|_K^2} = e^{-(\mathbf{b}^T\Sigma^{-1}\mathbf{b})} .$$

[25] As observed in [39, 69] prior probabilities can also be seen as a measure of complexity, assigning high complexity to the functions with small probability. This is consistent with the Minimum Description Length (MDL) principle proposed by Rissanen [81] to measure the complexity of a hypothesis in terms of the bit length needed to encode it. The MAP estimate mentioned above is closely related to the Minimum Description Length Principle: the hypothesis f which for given D_l can be described in the most compact way is chosen as the "best" hypothesis. Similar ideas have been explored by others (see [95, 96] for a summary).

Thus the stabilizer can be related to a Gaussian prior on the function space.
The interpretation is attractive since it seems to capture the idea that the stabilizer effectively constrains the desired function to be in the RKHS defined by the kernel K. It also seems to apply not only to classical regularization but to any functional of the form

$$H[f] = \frac{1}{l}\sum_{i=1}^{l} V(y_i - f(\mathbf{x}_i)) + \lambda \|f\|_K^2 \tag{75}$$

where $V(\mathbf{x})$ is any monotonically increasing loss function (see [40]). In particular it can be applied to the SVM (regression) case in which the relevant functional is

$$\frac{1}{l}\sum_{i=1}^{l} |y_i - f(\mathbf{x}_i)|_\epsilon + \lambda \|f\|_K^2 \; . \tag{76}$$

In both cases, one can write appropriate $P[D_l|f]$ and $P[f]$ for which the MAP estimate of

$$P[f|D_l] \propto P[D_l|f]P[f]$$

gives either equation (75) or equation (76). Of course, the MAP estimate is only one of several possible. In many cases, the average of $f = \int f dP[f|D_l]$ may make more sense[26] (see [58]). This argument provides a formal proof of the well-known equivalence between Gaussian processes defined by the previous equation with $P[f|D_l]$ Gaussian and the RN defined by equation (70)[27]. In the following we comment separately on the stabilizer – common to RN and SVM – and on the data term – which is different in the two cases.

7.2 Bayesian interpretation of the stabilizer in the RN and SVM functionals

Assume that the problem is to estimate f from sparse data y_i at location $\mathbf{x}_i$. From the previous description it is clear that choosing a kernel K is equivalent to assuming a Gaussian prior on f with covariance equal to K. Thus choosing a prior through K is equivalent a) to assume a Gaussian prior and b) to assume a correlation function associated with the family of functions f. The relation between positive definite kernels and correlation functions K of Gaussian random processes is characterized in details in [102], Theorem 5.2. In applications it is natural to use an empirical estimate of the correlation function, whenever available. Notice that in the MAP interpretation a Gaussian prior is *assumed* in RN as well as in SVM. For both RN and SVM when empirical data are available on the statistics of the family of functions of the form (74) one should check that $P[f]$ is Gaussian and make it zero-mean. Then an empirical estimate of the correlation function $E[f(\mathbf{x})f(\mathbf{y})]$ (with the expectation relative to the distribution $P[f]$) can be used as the kernel[28].
Notice also that the basis functions ϕ_n associated with the positive definite function $K(\mathbf{x}, \mathbf{y})$ correspond to the Principal Components associated with K.

[26] In the Gaussian case - Regularization Networks - the MAP and the average estimates coincide.
[27] Ironically, it is only recently that the neural network community seems to have realized the equivalence of many so-called neural networks and Gaussian processes and the fact that they work quite well (see [55] and references therein).
[28] We neglect here the question about how accurate the empirical estimation is.

7.3 Bayesian interpretation of the data term in the Regularization and SVM functional

As we already observed the model of the noise that has to be associated with the data term of the SVM functional is *not* Gaussian additive as in RN. The same is true for the specific form of Basis Pursuit Denoising considered in section 8, given the equivalence with SVM. Data terms of the type $V(y_i - f(\mathbf{x}_i))$ can be interpreted [40] in probabilistic terms as non-Gaussian noise models. Recently, Pontil, Mukherjee and Girosi [75] have derived a noise model corresponding to Vapnik's ϵ-insensitive loss function. It turns out that the underlying noise model consists of the superposition of Gaussian processes with different variances and means, that is[29]:

$$\exp\left(-|x|_\epsilon\right) = \int_{-\infty}^{+\infty} dt \int_0^\infty d\beta \lambda(t)\mu(\beta)\sqrt{\beta}\exp\left(-\beta(x-t)^2\right), \tag{77}$$

with:

$$\lambda_\epsilon(t) = \frac{1}{2(\epsilon+1)}\left(\chi_{[-\epsilon,\epsilon]}(t) + \delta(t-\epsilon) + \delta(t+\epsilon)\right), \tag{78}$$

$$\mu(\beta) \propto \beta^2 \exp\left(-\frac{1}{4\beta}\right). \tag{79}$$

where $\chi_{[-\epsilon,\epsilon]}(t)$ is 1 for $t \in [-\epsilon, \epsilon]$, 0 otherwise,
For the derivation see Appendix F or [75]. Notice that the variance has a unimodal distribution that does not depend on ϵ, and the mean has a distribution which is uniform in the interval $[-\epsilon, \epsilon]$, (except for two delta functions at $\pm\epsilon$, which ensures that the mean has not zero probability to be equal to $\pm\epsilon$). The distribution of the mean is consistent with the current understanding of Vapnik's ILF: errors smaller than ϵ do not count because they may be due entirely to the bias of the Gaussian noise.

7.4 Why a MAP interpretation may be misleading

We have just seen that minimization of both the RN and the SVMR functionals can be interpreted as corresponding to the MAP estimate of the posterior probability of f given the data, for certain models of the noise and for a specific Gaussian prior on the space of functions f. However, a MAP interpretation of this type may in general be *inconsistent* with Structural Risk Minimization and more generally with Vapnik's analysis of the learning problem. The following argument due to Vapnik shows the general point.
Consider functionals (32) and (53). From a Bayesian point of view, instead of the parameter λ – which in RN and SVM is a function of the data (through the SRM principle) – we have $\tilde{\lambda}$ which depends on the data as $\frac{\alpha}{l}$: the constant α has to be independent of the training data (i.e. their size l). On the other hand, as we discussed in section 2, SRM dictates a choice of λ depending on the training set. It seems unlikely that λ could simply depend on $\frac{\alpha}{l}$ as the MAP interpretation requires for consistency. Figure (7.4) gives a preliminar empirical demonstration that in the case of SVMR the "MAP" dependence of λ as $\frac{\alpha}{l}$ may not be correct.
Fundamentally, the core of Vapnik's analysis is that the key to learning from finite training sets is *capacity control*, that is the control of the complexity of the hypothesis space as a function of the training set. From this point of view the ability to choose λ as a function of the training

[29]In the following we introduce the variable $\beta = (2\sigma^2)^{-1}$.

Figure 1: An experiment (suggested by V. Vapnik) where the optimal λ does not simply depend on the training set as $\lambda = \frac{\alpha}{l}$ with α a constant and l the number of data points in the training set. In the right figure we plot λl as a function of the number of data. The data were generated from a 1-d sinusoid along 3 periods, with small uniform noise added. A SVMR with Gaussian kernel was used. We scaled the ordinate by 50 to compare with the $\log(\log(l))$ plot shown on the left. The number of training data ranged from 10 to 500. For each l we plot λl with λ being the optimal one (i.e. $\frac{2}{C}$ for the SVMR) estimated by using the true function for validation. The right figure shows that λl is not a constant as the MAP interpretation would require.

data is essential to our interpretation of Regularization and SVM in terms of the VC theory (compare the procedure described in our SRM section 2). Full capacity control and appropriate dependency of λ on the training set, which we expect in the general case not to be simply of the form $\frac{\alpha}{l}$, is lost in the direct MAP interpretation that we described in this chapter. Of course, an empirical Bayesian interpretation relying on hyper-parameters in the prior is possible and often useful but it amounts to little more than a parametric form for the posterior distribution, usually used in conjunction with maximum likelihood estimation of the parameters from the data.

8 Connections between SVMs and Sparse Approximation techniques

In recent years there has been a growing interest in approximating functions and representing signals using linear superposition of a small number of basis functions selected from a large, redundant set of basis functions, called a *dictionary*. These techniques go under the name of Sparse Approximations (SAs) [18, 17, 65, 42, 24, 57, 21, 26]. We will start with a short overview of SAs. Then we will discuss a result due to Girosi [38] that shows an equivalence between SVMs and a particular SA technique. Finally we will discuss the problem of Independent Component Analysis (ICA), another method for finding signal representations.

8.1 The problem of sparsity

Given a dictionary of basis functions (for example a frame, or just a redundant set of basis functions) $\{\varphi_1(\mathbf{x}), \ldots, \varphi_n(\mathbf{x})\}$ with n very large (possibly infinite), SA techniques seek an approximation of a function $f(\mathbf{x})$ as a linear combination of the smallest number of elements of the dictionary, that is, an approximation of the form:

$$f_{\mathbf{c}}(\mathbf{x}) = \sum_{i=1}^{n} c_i \varphi_i(\mathbf{x}), \tag{80}$$

with the smallest number of non-zero coefficients c_i. Formally, the problem is formulated as minimizing the following cost function:

$$E[\mathbf{c}] = D(f(\mathbf{x}), \sum_{i=1}^{n} c_i \varphi_i(\mathbf{x})) + \epsilon \|\mathbf{c}\|_{L_0}, \tag{81}$$

where D is a cost measuring the distance (in some predefined norm) between the true function $f(\mathbf{x})$ and our approximation, the L_0 norm of a vector counts the number of elements of that vector which are different from zero, and ϵ is a parameter that controls the trade off between sparsity and approximation. Observe that the larger ϵ is in (81), the more sparse the solution will be.
In the more general case of learning function f is not given, and instead we have a data set $D_l = \{(\mathbf{x}_1, y_1), \ldots (\mathbf{x}_l, y_l)\}$ of the values y_i of f at locations $\mathbf{x}_i$[30]. Note that in order to minimize $E[\mathbf{c}]$ we need to know f at all points $\mathbf{x}$. In the learning paradigm, in the particular case that $D(f(\mathbf{x}), \sum_{i=1}^{n} c_i \varphi_i(\mathbf{x})) = \|f(\mathbf{x}) - \sum_{i=1}^{n} c_i \varphi_i(\mathbf{x})\|_{L_2}^2$, the first term in equation (81) is replaced by an empirical one, and (81) becomes:

$$\frac{1}{l} \sum_{i=1}^{l} (y_i - \sum_{j=1}^{n} c_j \varphi_j(\mathbf{x_i}))^2 + \epsilon \|\mathbf{c}\|_{L_0} \tag{82}$$

Minimizing (81) can be used as well to find sparse approximations in the case that the function f is generated by a function f_0 corrupted by additive noise. In this case the problem can be formulated as finding a solution $\mathbf{c}$ to:

$$f = \Phi \mathbf{c} + \eta \tag{83}$$

with the smallest number of non-zero elements, where Φ is the matrix with columns the elements of the dictionary, and η is the noise. If we take a probabilistic approach and the noise is Gaussian, the problem can again be formulated as minimizing:

$$E[\mathbf{c}] = \|f(\mathbf{x}) - \sum_{i=1}^{n} c_i \varphi_i(\mathbf{x})\|_{L_2}^2 + \epsilon \|\mathbf{c}\|_{L_0}, \tag{84}$$

Unfortunately it can be shown that minimizing (81) is NP-hard because of the L_0 norm. In order to circumvent this shortcoming, approximated versions of the cost function above have been proposed. For example, in [18, 17] the authors use the L_1 norm as an approximation of the L_0 norm, obtaining an approximation scheme that they call *Basis Pursuit De-Noising* (BPDN) which consists of minimizing:

$$E[\mathbf{c}] = \|f(\mathbf{x}) - \sum_{i=1}^{n} c_i \varphi_i(\mathbf{x})\|_{L_2}^2 + \epsilon \sum_{i=1}^{n} |c_i|, \tag{85}$$

[30]For simplicity we consider the case where $P(\mathbf{x}$ is the uniform distribution.

8.2 Equivalence between BPDN and SVMs

In this section we consider the particular case in which we are given a data set $D_l = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l)\}$, and the dictionary consists of basis functions of the form:

$$\varphi_i(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}_i) \ \ \forall i = 1, \ldots, l \tag{86}$$

where K is the reproducing kernel of a RKHS $\mathcal{H}$, and the size l of D_l is equal to the size n of the dictionary. Moreover, following [38], we assume that $f(\mathbf{x})$ in eq. (81) is in the RKHS, and we use as the cost D in (81) the norm in the RKHS $\mathcal{H}$ induced by the kernel K, and approximate the L_0 norm with L_1. Under these assumptions, we get the SA technique that minimizes:

$$E[\mathbf{c}] = \|f(\mathbf{x}) - \sum_{i=1}^{n} c_i \varphi_i(\mathbf{x})\|_K^2 + \epsilon \|\mathbf{c}\|_{L_1}. \tag{87}$$

subject to $f(\mathbf{x}_i) = y_i$.
It can be shown [38] that this technique is equivalent to SVMR in the following sense: the two techniques give the same solution, which is obtained by solving the same quadratic programming problem. Girosi [38] proves the equivalence between SVMR and BPDN under the assumption that the data set $\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ has been obtained by sampling, *in absence of noise*, the target function f. Functional (87) differs from (85) only in the cost D. While Chen et al., in their BPDN method, measure the reconstruction error with an L_2 criterion, Girosi measures it by the true distance, in the $\mathcal{H}$ norm, between the target function f and the approximating function f^*. This measure of distance, which is common in approximation theory, is better motivated than the L_2 norm because it not only enforces closeness between the target and the model, but also between their derivatives, since $\|\cdot\|_K$ is a measure of smoothness.
Notice that from eq. (87) the cost function E cannot be computed because it requires the knowledge of f (in the first term). If we had $\|\cdot\|_{L_2}$ instead of $\|\cdot\|_K$ in eq. (87),this would force us to consider the approximation:

$$\|f(\mathbf{x}) - f^*(\mathbf{x})\|_{L_2}^2 \approx \frac{1}{l} \sum_{i=1}^{l} (y_i - f^*(\mathbf{x}_i))^2 \tag{88}$$

However if we used the norm $\|\cdot\|_K$ we can use the reproducing property (26) obtaining (see [38]):

$$E[\mathbf{c}] = \frac{1}{2}(\|f\|_K^2 + \sum_{i,j=1}^{l} c_i c_j K(\mathbf{x}_i; \mathbf{x}_j) - 2 \sum_{i=1}^{l} c_i y_i) + \epsilon \|\mathbf{c}\|_{L_1} \tag{89}$$

Observe that functional (89) is the same as the objective function of SVM of problem 5.3 up to the constant $\frac{1}{2}\|f\|_K^2$. However, in the SVM formulation the coefficients c_i satisfy two constraints, which in the case of sparsity are trivially satisfied under further assumptions. For details see [38]. It also follows from eq. (80) and (86) that the approximating function is of the form:

$$f^*(\mathbf{x}) \equiv f_{\mathbf{c}}(\mathbf{x}) = \sum_{i=1}^{l} c_i K(\mathbf{x}; \mathbf{x}_i). \tag{90}$$

This model is similar to the one of SVM (eq. (55)), except for the constant b.
This relation between SVMR and SA suggests directly that SVM yield a sparse representation.

8.3 Independent Component Analysis

Independent Component Analysis (ICA) is the problem of finding unknown sources whose linear superposition gives a number of observed signals, under the only assumption that the sources are statistically independent. A particular application is *Blind Source Separation* (BSS) where one is given a signal and seeks to decompose it as a linear combination of a number of unknown statistically independent sources. Following the notation in [4], the problem can be formulated as finding at any time t *both* the n (n predefined) sources $\mathbf{x}(t) = (x_1(t), \dots, x_n(t))$ **and** the mixing matrix A (which is assumed to be the same for every t) of the system of linear equations:

$$\mathbf{s}(t) = A\mathbf{x}(t) + \eta \tag{91}$$

where $\mathbf{s}(t)$ is our observed signal at time t, the elements of $\mathbf{x}(t)$, namely $x_i(t)$, are generated by statistically independent sources, and η is additive noise.
Observe that for any t the formulations of ICA and SA (see eq. (83)) are similar (Φ is A, f is $\mathbf{s}(t)$ and $\mathbf{c}$ is $\mathbf{x}(t)$). The difference is that in the case of SA we know the mixing matrix ("basis") A (Φ) and we only solve for the sources $\mathbf{x}$ ($\mathbf{c}$) with the smallest number of non-zero elements, while for ICA and BSS both the matrix A and the sources $\mathbf{x}$ are unknown, and we assume that $x_i(t)$ are statistically independent, while we don't have any explicit restriction on $\mathbf{A}$.
Various methods for ICA have been developed in recent years [3, 9, 63, 53, 65]. A review of the methods can be found in [52]. Typically the problem is solved by assuming a probability distribution model for the sources $x_i(t)$. A typical prior distribution is the Laplacian, namely $P(\mathbf{x}(t)) \propto \cdot e^{|x_1(t)|+\cdots+|x_n(t)|}$. Moreover, if the noise η is Gaussian with zero mean and variance σ^2, then, for a given A, the probability of $\mathbf{s}(t)$ given A can be written as:

$$P(\mathbf{s}(t)|A) = P(\mathbf{s}(t)|A, \mathbf{x}(t)) \cdot P(\mathbf{x}(t)) \propto \cdot e^{\frac{||\mathbf{s}(t)-A\mathbf{x}(t)||^2}{2\sigma^2}} \cdot e^{|\mathbf{x}_1(t)|+\cdots+|\mathbf{x}_n(t)|} \tag{92}$$

The MAP estimate of (92) gives $\mathbf{x}(t)$ as the minimizer of:

$$||\mathbf{s}(t) - A\mathbf{x}(t)||^2 + \epsilon \cdot \sum_{i=1}^{n} |x_i(t)| \tag{93}$$

Observe that this is the same as that of BPDN (eq. (85)). Therefore, for a fixed A the sources can be found by solving a (BPDN) problem. In fact iterative methods where at every iteration A is fixed and the sources are found, and then for fixed sources, A is updated using a learning rule have been developed in [65].
To summarize, using a Laplacian prior on the sources and following an iterative method for solving both for the sources and for their linear combination, ICA and BSS can be seen as iterative methods where at each iteration one solves a SA problem. This connection between ICA and sparsity has also been studied in [64]. Notice that if the prior on the sources is different, in particular if it is super-Gaussian, then the solution at every iteration need not be sparse.

9 Remarks

9.1 Regularization Networks can implement SRM

One of the main focuses of this review is to describe and motivate the classical technique of regularization – minimization of functionals such as in equation (1) – within the framework of

VC theory. In particular we have shown that classical regularization functionals can be motivated within the statistical framework of capacity control.

9.2 The SVM functional is a special formulation of regularization

Classical Regularization	$H[f] = \frac{1}{l}\sum_{i=1}^{l}(y_i - f(\mathbf{x}_i))^2 + \lambda\|f\|_K^2$
SVM Regression (SVMR)	$H[f] = \frac{1}{l}\sum_{i=1}^{l}\|y_i - f(\mathbf{x}_i)\|_\epsilon + \lambda\|f\|_K^2$
SVM Classification (SVMC)	$H[f] = \frac{1}{l}\sum_{i=1}^{l}\|1 - y_i f(\mathbf{x}_i)\|_+ + \lambda\|f\|_K^2$

Table 2: A unified framework: the minimizer of each of these three functionals has always the same form: $f(\mathbf{x}) = \sum_{i=1}^{l} c_i K(\mathbf{x}, \mathbf{x}_i)$ or $f(\mathbf{x}) = \sum_{i=1}^{l} c_i K(\mathbf{x}, \mathbf{x}_i) + b$. Of course in classification the decision function is $sign(f(\mathbf{x}))$.

Throughout our review it is clear that classical Regularization Networks, as well as Support Vector Machines for regression and Support Vector Machines for classification (see Table (2)), can be justified within the same framework, based on Vapnik's SRM principle and the notion of V_γ dimension. The three functionals of the table have different loss functions $V(\cdot,\cdot)$ but the same stabilizer. Thus the minimizer has the same general form and, as a consequence, the associated network has the same architecture. In particular, RKHS, associated kernels, and the mapping they induce from the input space into a higher dimensional space of features ϕ_n, are exactly the same in SVM as in RN. The different loss functions of SVM determine however quite different properties of the solution (see Table (2)) which is, unlike regularization, sparse in the c_n. Notice that loss functions different from quadratic loss have been used before in the context of regularization. In particular, the physical analogy of representing the data term using nonlinear spring (classical L_2 regularization corresponds to linear springs) was used and studied before (for instance see [40]). It is, however, the specific choice of the loss functions in SVMC and SVMR that provides several of their characteristic features, such as sparsity of the solution. Notice also that the geometric interpretation of $\|f\|_K^2$ in terms of the *margin* [96] is true only for the classification case and depends on the specific loss function $V(\cdot,\cdot)$ used in SVMC.

9.3 SVM, sparsity and compression

From the Kuhn-Tucker conditions of the QP problem associated with SVM one expects the Support Vectors to be usually sparser than the data. Notice that this is not obvious from a direct inspection of the functional $H[f]$ itself, where the regularizer is a L_2 norm on the function space. Especially in the case of regression it is not immediately obvious that the $H[f]$ in SVMR should yield a sparser solution than the $H[f]$ of classical regularization (see Table (2)). The equivalence of SVMR with a special form of Basis Pursuit Denoising shows that the ϵ-insensitive loss function with a L_2 regularizer is equivalent to a L_2 loss function *and* a L_1 regularizer. The latter is known to yield sparsity, though it is only an approximation of a "true" sparsity regularizer with the L_0 norm. Notice that SVM – like regularization – uses typically many *features* ϕ_n, but only – unlike regularization – a *sparse* subset of the examples. Thus SVM is not sparse in the primal representation (see section 3) of the classifier (or regressor) but it is sparse in the dual representation since it tends to use a subset of the dictionary consisting of the set of $K(\mathbf{x}, \mathbf{x}_i)$.

In this context, an interesting perspective on SVM is to consider its information compression properties. The support vectors represent in this sense the most informative data points and compress the information contained in the training set: for the purpose of, say, classification of future vectors, only the support vectors need to be stored, while all other training examples can be discarded. There is in fact a relation between the compression factor expressed as the ratio of data points to support vectors and the probability of test error. Vapnik [96], in comparing the empirical risk minimization principle with the Minimum Description Length principle [81], derives a bound on the generalization error as a function of the compression coefficient.

9.4 Gaussian processes, regularization and SVM

The very close relation between Gaussian processes and RN is well known [58, 102]. The connection is also valid for SVM in regression as well as in classification, since it depends on the form of the stabilizer, which is the same. The functional H of classical regularization is the exponent of the Gaussian conditional probability distribution characterizing the Gaussian process. The MAP estimate applied to the probability distribution corresponds to minimization of H yielding Regularization Networks – of which Radial Basis Function networks are a special case. Thus RN are connected to Gaussian processes via the MAP estimate, which in this case coincides with another estimate – the posterior mean.

9.5 Kernels and how to choose an input representation

A key issue in every learning problem concerns the input (and output) representation. This issue is outside the scope of the theory outlined in this review. There are however a few interesting observations that can be made. As pointed out by Vapnik, the choice of the kernel K is equivalent to choosing features related to the original inputs $\mathbf{x}$ by well-behaved functions $\phi_n(\mathbf{x})$, where the ϕ_n are defined by $K(\mathbf{x},\mathbf{y}) \equiv \sum_{n=1}^{N} \lambda_n \phi_n(\mathbf{x})\phi_n(\mathbf{y})$. Assume that K is given and that the input representation is now changed through a vector function $\mathbf{h}(\mathbf{x})$ mapping the original input $\mathbf{x}$ into the new feature vector $\mathbf{h}$. This is equivalent to using a new kernel K' defined in terms of the composite features $\phi_n(\mathbf{h}(\mathbf{x}))$ as $K'(\mathbf{x},\mathbf{y}) \equiv \sum_{n=1}^{N} \lambda_n \phi_n(\mathbf{h}(\mathbf{x}))\phi_n(\mathbf{h}(\mathbf{y}))$. For example, in the case of a polynomial kernel $K = (1 + \mathbf{x}\cdot\mathbf{y})^d$, a linear transformation of the input data $\mathbf{x}' = P^T\mathbf{x}$ is equivalent to using a new kernel $K'(\mathbf{x},\mathbf{y}) = (1+\mathbf{x}PP^T\mathbf{y})^d$. Clearly in the case that the projection is onto an orthonormal basis so that matrix P is orthonormal, the transformation does not affect the learning machine. On the other hand, if P is a matrix whose columns form an overcomplete or undercomplete set of basis functions, the transformation can change the learning machine. In many cases – especially when K is an expansion in an infinite series of ϕ_n – the most natural description is in terms of the kernel itself. In other cases, the best strategy is to define a finite set of features ϕ_n and then construct the kernel by computing $K(\mathbf{x},\mathbf{y}) = \sum_{n=1}^{N} \lambda_n \phi_n(\mathbf{x})\phi_n(\mathbf{y})$.

Synthesis of kernels from kernels

There are several symmetric positive definite kernels and a number of ways to construct new ones from existing kernels by operating on them with a few operations such as addition and convolution. For instance, if K_1 and K_2 are kernels then $K_1 + K_2$ is a kernel and K_1K_2 is a kernel; $(K_1)^n$ is a kernel. Thus the kernel $\sum_{i=0}^{d}(\mathbf{x}\cdot\mathbf{y})^i$ corresponds to the features of a polynomial of degree d in the spirit of [68]; Vapnik's kernel $K(\mathbf{x},\mathbf{y}) = (1 + \mathbf{x}\cdot\mathbf{y})^d$ is in fact equivalent and more compact. Aronszajn [5] describes several ways to construct positive definite kernels and thereby the associated RKHS. A completely equivalent analysis exists for correlation functions.

Exploiting prior information
In practical problems the choice of the regressors is often much more important than the choice of the learning machine. The choice of an appropriate input representation depends of course on prior information about the specific regression or classification problem. A general theory on how to use prior information to determine the appropriate representation is likely to be very far away. There are however a few approaches which yield some promise.

- Kernels and estimates of correlation functions.

  Assume that the problem is to estimate f from sparse data y_i at location $\mathbf{x}_i$. As we described in section 7, let us assume that there is prior information available in terms of the correlation function $R(\mathbf{x}, \mathbf{y}) = E[f(\mathbf{x})f(\mathbf{y})]$ of the family of functions to which f belongs. In applications, for instance, it may be possible to obtain an empirical estimate of the correlation function. From a Bayesian point of view this prior information together with the assumption of a Gaussian prior, determines the choice of the kernel $K = R$ and this automatically determines the feature representation – the ϕ_n – to be used in the regression problem. Preliminary experiments indicate that this strategy may give better results than other regression approaches [66].

- Invariances and Virtual Examples.

  In many pattern recognition problem specific invariances are known to hold a priori. Niyogy et al. [62] showed how several invariances can be embedded in the stabilizer or, equivalently, in virtual examples (see for a related work on tangent distance [89] and [84]).

- Generative probabilistic models.

  Jaakkola and Haussler [47] consider the case in which prior information is available in terms of a parametric probabilistic model $P(\mathbf{x}, y)$ of the process generating the data. They argue that good features for classification are the derivatives of $\log P$ with respect to the natural parameters of the distributions at the data points.

9.6 Capacity control and the physical world

An interesting question, outside the realm of mathematics, which has been asked recently is why large margin classifiers seem to work well in the physical world. As we saw throughout this review, the question is closely related to the question of why to assume smoothness in regression, that is why to use stabilizers such as $\|f\|_K^2$, which are usually smoothness functionals. Smoothness can be justified by observing that in many cases smoothness of input-output relations are implied directly by the existence of physical laws with continuity and differentiability properties. In classification, minimization of $\|f\|_K$ corresponds to maximization of the margin in the space of the ϕ_n; it is also equivalent to choosing the decision boundary resulting from thresholding the smoothest f in the original space, according to the smoothness criterion induced by K (notice that the decision boundary is the level crossing of f and not necessarily smooth everywhere). Conversely, we would not be able to generalize for input-output relations that are not smooth, that is for which "similar" inputs do not correspond to "similar" outputs (in an appropriate metric!). Such cases exist: for instance the mapping provided by a telephone directory between names and telephone numbers is usually not "smooth" and it is a safe bet that it would be difficult to learn it from examples. In cases in which physical systems are involved, however, input-output relations have some degree of smoothness and can be learned. From this point of view large margin (in feature space) and smoothness are properties of the physical world that are key to allow generalization, learning and the development of theories and models.

Acknowledgments
We would like to thank for suggestions Chris Burges, Peter Bartlett, Nello Cristianini, Grace Wahba and Berhard Schölkopf. We are grateful for many discussions with Alessandro Verri, Sayan Mukherjee and Ryan Rifkin. Very special thanks go to Federico Girosi and Vladimir Vapnik.

A Regularization Theory for Learning

Classical regularization methods proposed by Tikhonov and Arsenin [92] solve the learning problem by restricting the space of functions to be the domain of a functional $\Omega(f)$, called the *stabilizer*, which possesses the following three properties:

- The unknown function f is assumed to belong to the domain $D(\Omega)$ of functional $\Omega(f)$.
- On the domain $D(\Omega)$ the functional $\Omega(f)$ admits real nonnegative values.
- The sets:

$$M_c = \{f : \Omega(f) \leq c\}$$

  are compact for every real nonnegative c.

For example, a functional of this sort typically used is the sum of the L_2 norms of the first k derivatives of f. In this case, the sets M_c defined above are Sobolev spaces. Using such a functional means that we restrict our space of functions to be the space of *smooth* functions, the functions whose derivatives are in L_2.
Given such a functional $\Omega(f)$, the idea of regularization is to find f as the minimizer of a certain loss functional which we take to be:

$$H[f] = \sum_{i=1}^{N} V(f(\mathbf{x}_i) - y_i) + \lambda\Omega[f] \ . \tag{94}$$

where V is the loss function and λ is a positive number that is usually called the *regularization parameter*. The first term is enforcing closeness to the data, and the second enforces the solution to be in a set M_c with a small c, while the regularization parameter controls the tradeoff between these two terms. For example, in the particular case that M_c is a Sobolev space, the second term of the minimized functional enforces the smoothness of f. The first term in equation (94) is the *empirical error*, while the second term is usually called the *smoothness functional* since it enforces some sort of smoothness. Various methods for choosing λ are proposed in the literature [1, 100, 101, 49, 96]. Under some conditions on the regularization parameter λ, it can be shown

[92] that as the number of training examples increases the minimizer of equation (94) converges to the exact solution f in the space $D(\Omega)$.
To summarize, to solve the ill-posed problem of learning from examples using classical regularization methods, we need to restrict the space where we search for the solution, and minimize a functional that depends on the empirical error and a cost related with a functional defined on the space of functions we search.

B An example of RKHS

Here we present a simple way to construct meaningful RKHS of functions of one variable over $[0, 2\pi]$. In the following all the normalization factors will be set to 1 for simplicity.
Let us consider any function $K(x)$ which is continuous, symmetric, periodic, and whose Fourier coefficients λ_n are positive. Such a function can be expanded in a uniformly convergent Fourier series:

$$K(x) = \sum_{n=0}^{\infty} \lambda_n \cos(nx) \ . \tag{95}$$

An example of such a function is

$$K(x) = 1 + \sum_{n=1}^{\infty} h^n \cos(nx) == \frac{1}{2\pi} \frac{1 - h^2}{1 - 2h\cos(x) + h^2}$$

where $h \in (0, 1)$.
It is easy to check that, if (95) holds, then we have:

$$K(x-y) = 1 + \sum_{n=1}^{\infty} \lambda_n \sin(nx)\sin(ny) + \sum_{n=1}^{\infty} \lambda_n \cos(nx)\cos(ny) \tag{96}$$

which is of the form (27) in which the set of *orthogonal* functions ϕ_n has the form:

$$\{\phi_i(x)\}_{i=0}^{\infty} \equiv (1, \sin(x), \cos(x), \sin(2x), \cos(2x), \ldots, \sin(nx), \cos(nx), \ldots) \ .$$

Therefore, given any function K which is continuous, periodic and symmetric we can then define a RKHS $\mathcal{H}$ over $[0, 2\pi]$ by defining a scalar product of the form:

$$< f, g >_{\mathcal{H}} \equiv \sum_{n=0}^{\infty} \frac{f_n^c g_n^c + f_n^s g_n^s}{\lambda_n}$$

where we use the following symbols for the Fourier coefficients of a function f:

$$f_n^c \equiv < f, \cos(nx) > \ , \quad f_n^s \equiv < f, \sin(nx) >$$

The functions in $\mathcal{H}$ are therefore functions in $L_2([0, 2\pi])$ whose Fourier coefficients satisfy the following constraint:

$$\|f\|_{\mathcal{H}}^2 = \sum_{n=0}^{\infty} \frac{(f_n^c)^2 + (f_n^s)^2}{\lambda_n} < +\infty \tag{97}$$

Since the sequence λ_n is decreasing, the constraint that the norm (97) has to be finite can be seen as a constraint on the rate of decrease to zero of the Fourier coefficients of the function f, which is known to be related to the smoothness properties of f. Therefore, choosing different kernels K is equivalent to choose RKHS of functions with different smoothness properties, and the norm (97) can be used as the smoothness functional.

C Regularized Solutions in RKHS

Let us look more closely at the solution of the minimization of functional (3). This is equivalent to assume that the functions in $\mathcal{H}$ have a unique expansion of the form:

$$f(\mathbf{x}) = \sum_{n=1}^{\infty} c_n \phi_n(\mathbf{x})$$

and that their norm is:

$$\|f\|_{\mathcal{H}}^2 = \sum_{n=1}^{\infty} \frac{c_n^2}{\lambda_n}.$$

We can think of the functional $H[f]$ as a function of the coefficients c_n. In order to minimize $H[f]$ we take its derivative with respect to c_n and set it equal to zero, obtaining the following:

$$-C \sum_{i=1}^{l} V'(y_i, f(\mathbf{x}_i)) \phi_n(\mathbf{x}_i) + \frac{c_n}{\lambda_n} = 0. \tag{98}$$

where we note by V' the partial derivative of V w.r.t. f. Let us now define the following set of unknowns:

$$a_i \equiv CV'(y_i, f(\mathbf{x}_i)).$$

Using eq. (98) we can express the coefficients c_n as a function of the a_i:

$$c_n = \lambda_n \sum_{i=1}^{l} a_i \phi_n(\mathbf{x}_i).$$

The solution of the variational problem has therefore the form:

$$f(\mathbf{x}) = \sum_{n=1}^{\infty} c_n \phi_n(\mathbf{x}) = \sum_{n=1}^{\infty} \sum_{i=1}^{l} a_i \lambda_n \phi_n(\mathbf{x}_i) \phi_n(\mathbf{x}) = \sum_{i=1}^{l} a_i K(\mathbf{x}, \mathbf{x}_i), \tag{99}$$

where we have used the expansion (27). This shows that, independently of the form of V, as long as it is differentiable, the solution of the regularization functional $H[f]$ is always a linear superposition of kernel functions, one for each data point. The loss function V affects the computation of the coefficients a_i. In fact, plugging eq. (99) back in the definition of the a_i we obtain the following set of equations for the coefficients a_i:

$$a_i = CV'\left(y_i, \sum_{j=1}^{l} K_{ij} a_j\right), \quad i = 1, \ldots, l$$

where we have defined $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$. In the case in which $V(\cdot,\cdot) = (\cdot - \cdot)^2$ we obtain the classical regularization theory solution (see Girosi, Jones and Poggio, 1995 for an alternative derivation):

$$(K + \gamma I)\,\mathbf{a} = \mathbf{y},$$

where we have defined $\gamma \equiv \frac{1}{C}$.

D Relation between SVMC and SVMR

We want to study problem 5.1 in the classification case ($y_i \in \{-1, 1\}\ \forall i$). Note that when performing SVMR on $\{-1, 1\}$-valued data, if $\epsilon \geq 1$, the optimal solution to problem 5.1 is $f = 0, \boldsymbol{\xi} = \boldsymbol{\xi}^* = \mathbf{0}$. Therefore, we restrict our attention to cases were $\epsilon < 1$.
We start by making the following variable substitution:

$$\eta_i = \begin{cases} \xi_i & \text{if } y_i = 1 \\ \xi_i^* & \text{if } y_i = -1. \end{cases} \quad , \quad \eta_i^* = \begin{cases} \xi_i^* & \text{if } y_i = 1 \\ \xi_i & \text{if } y_i = -1. \end{cases} \tag{100}$$

Combining this substitution with our knowledge that $y_i \in \{-1, 1\}$ yields the following modification of problem 5.1:

Problem D.1

$$\min_{f, \boldsymbol{\eta}, \boldsymbol{\eta}^*} \Phi_C(f, \boldsymbol{\eta}, \boldsymbol{\eta}^*) = \frac{C}{l} \sum_{i=1}^{l} (\eta_i + \eta_i^*) + \frac{1}{2} \|f\|_K^2$$

subject to the constraints:

$$\begin{array}{lll} y_i f(\mathbf{x}_i) & \geq 1 - \epsilon + \eta_i & i = 1, \ldots, l \\ y_i f(\mathbf{x}_i) & \leq 1 + \epsilon + \eta_i^* & i = 1, \ldots, l \\ \eta_i, \eta_i^* & \geq 0, & i = 1, \ldots, l. \end{array} \tag{101}$$

Continuing, we divide both sides of each constraint in problem D.1 by $1 - \epsilon$, and make the variable substitutions $f' = \frac{w}{1-\epsilon}$, $\eta' = \frac{\eta}{1-\epsilon}$, $\eta'^* = \frac{\eta^*}{1-\epsilon}$:

Problem D.2

$$\min_{f, \boldsymbol{\eta}', \boldsymbol{\eta}'^*} \Phi_{\frac{C}{1-\epsilon}}(f', \boldsymbol{\eta}', \boldsymbol{\eta}'^*) = \frac{1}{l} \frac{C}{1+\epsilon} \sum_{i=1}^{l} (\xi_i' + \xi_i'^*) + \frac{1}{2} \|f'\|_K^2 \tag{102}$$

subject to the constraints:

$$\begin{array}{llll} y_i f(\mathbf{x}_i) & \geq & 1 - \eta_i' & i = 1, \ldots, l \\ y_i f(\mathbf{x}_i) & \leq & \frac{1+\epsilon}{1-\epsilon} + \eta_i'^* & i = 1, \ldots, l \\ \eta_i', \xi_i'^* & \geq & 0 & i = 1, \ldots, l. \end{array} \tag{103}$$

Notice that Problem D.2 looks very similar to the SVMC problem 5.4, the only difference being given by the additional constraint in problem D.2 associate to the variable η'^*. Through an analysis of the KKT conditions of problem D.2, it is easy to see that if f, ξ solves problem 5.4 with parameter C, under the additional condition that $\epsilon \in [a, 1)$, $f' = f, \eta_i' = \xi, \eta_i'^* = 0$ solves problem D.2 with parameter $C(1+\epsilon)$. Then $(1-\epsilon)f$ is the solution of problem 5.1 with parameter $C(1 + \epsilon)$. This result can be applied as well to formulation (65): if f, ξ solves of problem 5.4 with parameter A and $\epsilon \in [a, 1)$, then $(1 - \epsilon)f$ solves problem D.2 with parameter $\frac{A}{1-\epsilon}$. The constant a under which the relation is true can be related to the radius R of the smallest sphere containing all the data points and to the norm of the solution to the SVMC problem 5.4. See [76] for further details and a complete proof of the results reported here.

E Proof of the theorem 6.2

Below we always assume that data X are within a sphere of radius R in the feature space defined by the kernel K of the RKHS. Without loss of generality, we also assume that y is bounded between -1 and 1. Let's consider first the case of the L_1 loss function. Let B be the upper bound on the loss function (which always exists under our assumptions). From definition 2.7 we can decompose the rules for separating points as follows:

$$\begin{array}{rl} \text{class 1 if}: & y_i - f(\mathbf{x}_i) \geq s + \gamma \\ \text{or} & f(\mathbf{x}_i) - y_i \geq s + \gamma \\ \text{class } -1 \text{ if}: & y_i - f(\mathbf{x}_i)) \leq s - \gamma \\ \text{or} & f(\mathbf{x}_i)) - y_i \leq s - \gamma \end{array} \tag{104}$$

for some $B - \gamma \geq s \geq \gamma$. Using this observation it is clear that for any N points, the number of separations we can get using rules (105) is not more than the number of separations we can get using the product of two "indicator functions with margin":

$$\begin{array}{rrl} \text{function (a)}: & \text{class } -1 \text{ if}: & y_i - f_1(\mathbf{x}_i) \geq s_1 + \gamma \\ & \text{class 1 if}: & y_i - f_1(\mathbf{x}_i)) \leq s_1 - \gamma \\ \text{function (b)}: & \text{class 1 if}: & f_2(\mathbf{x}_i) - y_i \geq s_2 + \gamma \\ & \text{class } -1 \text{ if}: & f_2(\mathbf{x}_i)) - y_i \leq s_2 - \gamma \end{array} \tag{105}$$

where f_1 and f_2 are in $\mathcal{H}$, $B - \gamma \geq s_1, s_2 \geq \gamma$. For $s_1 = s_2 = s$ and for $f_1 = f_2 = f$ we recover (105): for example, if $y - f(\mathbf{x}) \geq s + \gamma$ then indicator function (a) will give -1, indicator function (b) will give also -1, so their product will give $+1$ which is what we get if we follow (105). So since we give more freedom to f_1, f_2, s_1, s_2 clearly we can get more separations for any set of points than we get using (105).
As discussed in section 2, for any N points the number of separations is bounded by the growth function. Moreover, for products of indicator functions it is known [96] that the growth function is bounded by the product of the growth functions of the indicator functions. Furthermore, the indicator functions in (106) are hyperplanes with margin in the $N + 1$ dimensional space of vectors $\{\phi_n(\mathbf{x}), y\}$ where the radius of the data is $R^2 + 1$, the norm of the hyperplane is bounded by $A^2 + 1$, (where in both cases we add 1 because of y), and the margin is bounded by $\frac{\gamma^2}{A^2+1}$. The V_γ dimension h_γ of these hyperplanes with margin is known [96, 8] to be bounded by $h_\gamma \leq \min((N + 1) + 1, \frac{(R^2+1)(A^2+1)}{\gamma^2})$. So the growth function of the separating rules (105) is bounded by $\mathcal{G}(l) \leq (\frac{el}{h_\gamma})^{h_\gamma}(\frac{el}{h_\gamma})^{h_\gamma}$ whenever $l \geq h_\gamma$. If h_γ^{reg} is the V_γ dimension of the L_1 loss function, then clearly h_γ^{reg} cannot be larger than the larger number l for which the inequality:

$$2^l \leq (\frac{el}{h_\gamma})^{h_\gamma}(\frac{el}{h_\gamma})^{h_\gamma} \tag{106}$$

holds. From this we get that $l \leq 5h_\gamma$, therefore $h_\gamma^{reg} \leq 5 \min\ (N + 2, \frac{(R^2+1)(A^2+1)}{\gamma^2})$ which proves the theorem for the case of L_1 loss functions.
The sketched proof can be extended to the general L_p loss function and to the Vapnik's ILF [36].

F The noise model of the data term in SVMR

We compute here the probability distributions $\lambda_\epsilon(t)$ and $\nu(\sigma)$ (see equations (78) and (79)) solving equation (77).
From equation (77) computing the integral with respect to β we obtain:

$$e^{-|x|_\epsilon} = \int_{-\infty}^{+\infty} dt \lambda(t) G(x-t) \tag{107}$$

where we have defined:

$$G(t) = \int_0^\infty d\beta \mu(\beta) \sqrt{\beta} e^{-\beta t^2} \tag{108}$$

Observe that the function G is a density distribution, because both the functions in the r.h.s. of equation (108) are densities. In order to compute G we observe that for $\epsilon = 0$ the function $e^{-|x|_\epsilon}$ becomes the Laplace distribution. In this case we can simply set $\lambda_{\epsilon=0}(t) = \delta(t)$ and obtain $G(t)$ from equation (107):

$$G(t) = e^{-|t|}. \tag{109}$$

We can then compute the probability distribution μ by inverting equation (108). This requires to computing the inverse Laplace transform of $e^{-|t|}$. We obtain:

$$\mu(\beta) = \beta^{-2} e^{-\frac{1}{4\beta}}. \tag{110}$$

It remains to obtain the expression of $\lambda(t)$ for $\epsilon > 0$. To this purpose we write equation (107) in Fourier space:

$$\tilde{F}[e^{-|x|_\epsilon}] = \tilde{G}(\omega) \tilde{\lambda}_\epsilon(\omega) \tag{111}$$

with:

$$\tilde{F}[e^{-|x|_\epsilon}] = \frac{\sin(\epsilon\omega) + w\cos(\epsilon\omega)}{\omega(1+\omega^2)}. \tag{112}$$

and:

$$\tilde{G}(\omega) = \frac{1}{1+\omega^2}. \tag{113}$$

Plugging equations (112) and (113) in equation (111) we obtain:

$$\tilde{\lambda}_\epsilon(\omega) = \frac{sin\epsilon\omega}{\omega} + cos\epsilon\omega. \tag{114}$$

Finally taking the inverse Fourier Transform and normalizing we obtain equation (78). For more details see [75].

References

[1] D. Allen. The relationship between variable selection and data augmentation and a method for prediction. *Technometrics*, 16:125–127, 1974.

[2] N. Alon, S. Ben-David, N. Cesa-Bianchi, and D. Haussler. Scale-sensitive dimensions, uniform convergence, and learnability. *Symposium on Foundations of Computer Science*, 1993.

[3] S. Amari, A. Cichocki, and H. Yang. A new learning algorithm for bling signal separation. In *Advances in Neural Information Processing System*, pages 757–763, Cambridge, MA, 1995. MIT Press.

[4] S.I. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10:251–276, 1998.

[5] N. Aronszajn. Theory of reproducing kernels. *Trans. Amer. Math. Soc.*, 686:337–404, 1950.

[6] P. Bartlett. The sample complexity of pattern classification with neural networks: the size of the weights is more important that the size of the network. *IEEE Transactions on Information Theory*, 1998.

[7] P. Bartlett, P.M. Long, and R.C. Williamson. Fat-shattering and the learnability of real-valued functions. *Journal of Computer and Systems Sciences*, 52(3):434–452, 1996.

[8] P. Bartlett and J. Shawe-Taylor. Generalization performance of support vector machine and other patern classifiers. In C. Burges B. Scholkopf, editor, *Advances in Kernel Methods–Support Vector Learning*. MIT press, 1998.

[9] A.J. Bell and T. Sejnowski. An informationb maximization approach to blind separation and blind deconvolutio. *Neural Computation*, 7:1129–1159, 1995.

[10] M. Bertero. Regularization methods for linear inverse problems. In C. G. Talenti, editor, *Inverse Problems*. Springer-Verlag, Berlin, 1986.

[11] M. Bertero, T. Poggio, and V. Torre. Ill-posed problems in early vision. *Proceedings of the IEEE*, 76:869–889, 1988.

[12] L. Bottou and V. Vapnik. Local learning algorithms. *Neural Computation*, 4(6):888–900, November 1992.

[13] M.D. Buhmann. Multivariate cardinal interpolation with radial basis functions. *Constructive Approximation*, 6:225–255, 1990.

[14] M.D. Buhmann. On quasi-interpolation with Radial Basis Functions. Numerical Analysis Reports DAMPT 1991/NA3, Department of Applied Mathematics and Theoretical Physics, Cambridge, England, March 1991.

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

[16] O. Chapelle and V. Vapnik. Model selection for support vector machines. In *Advances in Neural Information Processing Systems*, 1999.

[17] S. Chen, , D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. Technical Report 479, Department of Statistics, Stanford University, May 1995.

[18] S. Chen. *Basis Pursuit*. PhD thesis, Department of Statistics, Stanford University, November 1995.

[19] V. Cherkassky and F. Mulier. *Learning from Data: Concepts, Theory, and Methods*. Wiley, New York, 1998.

[20] J.A. Cochran. *The analysis of linear integral equations*. McGraw-Hill, New York, 1972.

[21] R.R. Coifman and M.V. Wickerhauser. Entropy-based algorithms for best-basis selection. *IEEE Transactions on Information Theory*, 38:713–718, 1992.

[22] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:1–25, 1995.

[23] R. Courant and D. Hilbert. *Methods of mathematical physics. Vol. 2*. Interscience, London, England, 1962.

[24] I. Daubechies. *Ten lectures on wavelets*. CBMS-NSF Regional Conferences Series in Applied Mathematics. SIAM, Philadelphia, PA, 1992.

[25] C. de Boor. Quasi-interpolants and approximation power of multivariate splines. In M. Gasca and C.A. Micchelli, editors, *Computation of Curves and Surfaces*, pages 313–345. Kluwer Academic Publishers, Dordrecht, Netherlands, 1990.

[26] R.A. DeVore. Nonlinear approximation. *Acta Numerica*, pages 51–150, 1998.

[27] L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Number 31 in Applications of mathematics. Springer, New York, 1996.

[28] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.

[29] R.M. Dudley. A course on empirical processes. *Lecture Notes in Mathematics*, 1097:2–142, 1984.

[30] R.M. Dudley, E. Gine, and J. Zinn. Uniform and universal glivenko-cantelli classes. *Journal of Theoretical Probability*, 4:485–510, 1991.

[31] N. Dyn. Interpolation and approximation by radial and related functions. In C.K. Chui, L.L. Schumaker, and D.J. Ward, editors, *Approximation Theory, VI*, pages 211–234. Academic Press, New York, 1991.

[32] N. Dyn, I.R.H. Jackson, D. Levin, and A. Ron. On multivariate approximation by integer translates of a basis function. Computer Sciences Technical Report 886, University of Wisconsin–Madison, November 1989.

[33] N. Dyn, D. Levin, and S. Rippa. Numerical procedures for surface fitting of scattered data by radial functions. *SIAM J. Sci. Stat. Comput.*, 7(2):639–659, April 1986.

[34] T. Evgeniou, L. Perez-Breva, M. Pontil, and T. Poggio. Bounds on the generalization performance of kernel machines ensembles. A.i. memo, MIT Artificial Intelligence Lab., 1999.

[35] T. Evgeniou and M. Pontil. A note on the generalization performance of kernel classifiers with margin. A.i. memo, MIT Artificial Intelligence Lab., 1999.

[36] T. Evgeniou and M. Pontil. On the v-gamma dimension for regression in reproducing kernel hilbert spaces. A.i. memo, MIT Artificial Intelligence Lab., 1999.

[37] F. Girosi. Models of noise and robust estimates. A.I. Memo 1287, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1991.

[38] F. Girosi. An equivalence between sparse approximation and Support Vector Machines. *Neural Computation*, 10(6):1455–1480, 1998.

[39] F. Girosi, M. Jones, and T. Poggio. Regularization theory and neural networks architectures. *Neural Computation*, 7:219–269, 1995.

[40] F. Girosi, T. Poggio, and B. Caprile. Extensions of a theory of networks for approximation and learning: outliers and negative examples. In R. Lippmann, J. Moody, and D. Touretzky, editors, *Advances in Neural information processings systems 3*, San Mateo, CA, 1991. Morgan Kaufmann Publishers.

[41] W. Härdle. *Applied nonparametric regression*, volume 19 of *Econometric Society Monographs*. Cambridge University Press, 1990.

[42] G.F. Harpur and R.W. Prager. Development of low entropy coding in a recurrent network. *Network*, 7:277–284, 1996.

[43] T. Hastie and R. Tibshirani. *Generalized Additive Models*, volume 43 of *Monographs on Statistics and Applied Probability*. Chapman and Hall, London, 1990.

[44] S. Haykin. *Neural networks : a comprehensive foundation*. Macmillan, New York, 1994.

[45] H. Hochstadt. *Integral Equations*. Wiley Classics Library. John Wiley & Sons, 1973.

[46] V.V. Ivanov. *The Theory of Approximate Methods and Their Application to the Numerical Solution of Singular Integral Equations*. Nordhoff International, Leyden, 1976.

[47] T. Jaakkola and D. Haussler. Probabilistic kernel regression models. In *Proc. of Neural Information Processing Conference*, 1998.

[48] I.R.H. Jackson. *Radial Basis Functions methods for multivariate approximation*. Ph.d. thesis, University of Cambridge, U.K., 1988.

[49] M. Kearns, Y. Mansour, A. Ng, and D. Ron. An experimental and theoretical comparison of model selection methods. *In Proceedings of the Eighth Annual ACM Conference on Computational Learning Theory*, 1995.

[50] M. Kearns and R.E. Shapire. Efficient distribution-free learning of probabilistic concepts. *Journal of Computer and Systems Sciences*, 48(3):464–497, 1994.

[51] G.S. Kimeldorf and G. Wahba. A correspondence between Bayesan estimation on stochastic processes and smoothing by splines. *Ann. Math. Statist.*, 41(2):495–502, 1971.

[52] T-W. Lee, M. Girolami, A.J. Bell, and T. Sejnowski. A unifying information-theoretical framework for independent component analysis. *Int. J. on Math. and Comp. Mod.*, 1998. In Press.

[53] M. Lewicki and T. Sejnowski. Learning nonlinear overcomplete representation for efficient coding. In *Advances in Neural Information Processing System*. 1997. In Press.

[54] G. G. Lorentz. *Approximation of Functions*. Chelsea Publishing Co., New York, 1986.

[55] D.J.C. MacKay. Introduction to gaussian processes. 1997. (available at the URL: http://wol.ra.phy.cam.ac.uk/mackay).

[56] W.R. Madych and S.A. Nelson. Polyharmonic cardinal splines: a minimization property. *Journal of Approximation Theory*, 63:303–320, 1990a.

[57] S. Mallat and Z. Zhang. Matching Pursuit in a time-frequency dictionary. *IEEE Transactions on Signal Processing*, 41:3397–3415, 1993.

[58] J. L. Marroquin, S. Mitter, and T. Poggio. Probabilistic solution of ill-posed problems in computational vision. *J. Amer. Stat. Assoc.*, 82:76–89, 1987.

[59] H.N. Mhaskar. Neural networks for localized approximation of real functions. In C.A. Kamm et al., editor, *Neural networks for signal processing III, Proceedings of the 1993 IEEE-SP Workshop*, pages 190–196, New York, 1993a. IEEE Signal Processing Society.

[60] C. A. Micchelli. Interpolation of scattered data: distance matrices and conditionally positive definite functions. *Constructive Approximation*, 2:11–22, 1986.

[61] P. Niyogi and F. Girosi. On the relationship between generalization error, hypothesis complexity, and sample complexity for radial basis functions. *Neural Computation*, 8:819–842, 1996.

[62] P. Niyogi, F. Girosi, and T. Poggio. Incorporating prior information in machine learning by creating virt ual examples. *Proceedings of the IEEE.*, 86(11):2196–2209, 1998.

[63] E. Oja. The nonlinear pca learning rule in independent component analysis. *Neurocomputing*, 17:25–45, 1997.

[64] B. Olshausen. Learning linear, sparse, factorial codes. A.I. Memo 1580, MIT Artificial Intelligence Lab., 1996.

[65] B.A. Olshausen and D.J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[66] C. Papageorgiou, F. Girosi, and T.Poggio. Sparse correlation kernel based signal reconstruction. Technical Report 1635, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1998. (CBCL Memo 162).

[67] G. Parisi. *Statistical Field Theory.* Addison-Wesley, Reading, Massachusets, 1988.

[68] T. Poggio. On optimal nonlinear associative recall. *Biological Cybernetics*, 19:201–209, 1975.

[69] T. Poggio and F. Girosi. A theory of networks for approximation and learning. A.I. Memo No. 1140, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1989.

[70] T. Poggio and F. Girosi. Networks for approximation and learning. *Proceedings of the IEEE*, 78(9), September 1990.

[71] T. Poggio and F. Girosi. Networks for Approximation and Learning. In C. Lau, editor, *Foundations of Neural Networks*, pages 91–106. IEEE Press, Piscataway, NJ, 1992.

[72] T. Poggio and F. Girosi. A Sparse Representation for Function Approximation. *Neural Computation*, 10(6), 1998.

[73] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317:314–319, 1985.

[74] D. Pollard. *Convergence of stochastic processes.* Springer-Verlag, Berlin, 1984.

[75] M. Pontil, S. Mukherjee, and F. Girosi. On the noise model of support vector machine regression. A.I. Memo, MIT Artificial Intelligence Laboratory, 1998. (in preparation).

[76] M. Pontil, R. Rifkin, and T. Evgeniou. From regression to classification in support vector machines. A.I. Memo 1649, MIT Artificial Intelligence Lab., 1998.

[77] M.J.D. Powell. The theory of radial basis functions approximation in 1990. In W.A. Light, editor, *Advances in Numerical Analysis Volume II: Wavelets, Subdivision Algorithms and Radial Basis Functions*, pages 105–210. Oxford University Press, 1992.

[78] C. Rabut. How to build quasi-interpolants. applications to polyharmonic B-splines. In P.-J. Laurent, A. Le Mehautè, and L.L. Schumaker, editors, *Curves and Surfaces*, pages 391–402. Academic Press, New York, 1991.

[79] C. Rabut. An introduction to Schoenberg's approximation. *Computers Math. Applic.*, 24(12):149–175, 1992.

[80] R. Rifkin, M. Pontil, and A. Verri. A note on support vector machine degeneracy. A.i. memo, MIT Artificial Intelligence Lab., 1999.

[81] J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465–471, 1978.

[82] I.J. Schoenberg. Contributions to the problem of approximation of equidistant data by analytic functions, part a: On the problem of smoothing of graduation, a first class of analytic approximation formulae. *Quart. Appl. Math.*, 4:45–99, 1946a.

[83] I.J. Schoenberg. Cardinal interpolation and spline functions. *Journal of Approximation theory*, 2:167–206, 1969.

[84] B. Scholkpof, P. Simard, A. Smola, and V. Vapnik. Prior knowledge in suport vector kernels. In *Advances in Neural Information Processing Systems 9*, 1997.

[85] L.L. Schumaker. *Spline functions: basic theory.* John Wiley and Sons, New York, 1981.

[86] J. Shawe-Taylor, P. L. Bartlett, R. C. Williamson, and M. Anthony. Structural risk minimization over data-dependent hierarchies. *IEEE Transactions on Information Theory*, 1998. To appear. Also: NeuroCOLT Technical Report NC-TR-96-053, 1996, ftp://ftp.dcs.rhbnc.ac.uk/pub/neurocolt/tech_reports.

[87] J. Shawe-Taylor and N. Cristianini. Robust bounds on generalization from the margin distribution. Technical Report NeuroCOLT2 Technical Report NC2-TR-1998-029, NeuroCOLT2, 1998.

[88] B.W. Silverman. Spline smoothing: the equivalent variable kernel method. *The Annals of Statistics*, 12:898–916, 1984.

[89] P. Simard, Y. LeCun, and J. Denker. Efficient pattern recognition using a new transformation distance. In *Advances in Neural Information Processing Systems 5*, pages 50–58, 1993.

[90] A. Smola and B. Schölkopf. On a kernel-based method for pattern recognition, regression, approximation and operator inversion. *Algorithmica*, 22:211 – 231, 1998.

[91] J. Stewart. Positive definite functions and generalizations, an historical survey. *Rocky Mountain J. Math.*, 6:409–434, 1976.

[92] A. N. Tikhonov and V. Y. Arsenin. *Solutions of Ill-posed Problems.* W. H. Winston, Washington, D.C., 1977.

[93] L.G. Valiant. A theory of learnable. *Proc. of the 1984 STOC*, pages 436–445, 1984.

[94] V. N. Vapnik. *Estimation of Dependences Based on Empirical Data.* Springer-Verlag, Berlin, 1982.

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

[96] V. N. Vapnik. *Statistical Learning Theory.* Wiley, New York, 1998.

[97] V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequences of events to their probabilities. *Th. Prob. and its Applications*, 17(2):264–280, 1971.

[98] V.N. Vapnik and A. Ya. Chervonenkis. The necessary and sufficient conditions for the uniform convergence of averages to their expected values. *Teoriya Veroyatnostei i Ee Primeneniya*, 26(3):543–564, 1981.

[99] V.N. Vapnik and A. Ya. Chervonenkis. The necessary and sufficient conditions for consistency in the empirical risk minimization method. *Pattern Recognition and Image Analysis*, 1(3):283–305, 1991.

[100] G. Wahba. Spline bases, regularization, and generalized cross-validation for solving approximation problems with large quantities of noisy data. In J. Ward and E. Cheney, editors, *Proceedings of the International Conference on Approximation theory in honour of George Lorenz*, Austin, TX, January 8–10 1980. Academic Press.

[101] G. Wahba. A comparison of GCV and GML for choosing the smoothing parameter in the generalized splines smoothing problem. *The Annals of Statistics*, 13:1378–1402, 1985.

[102] G. Wahba. *Splines Models for Observational Data*. Series in Applied Mathematics, Vol. 59, SIAM, Philadelphia, 1990.

[103] R. Williamson, A. Smola, and B. Scholkopf. Generalization performance of regularization networks and support vector machines via entropy numbers. Technical Report NC-TR-98-019, Royal Holloway College University of London, 1998.