INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA

UMI®
800-521-0600

REGULARIZED RADIAL BASIS FUNCTION NETWORKS: THEORY AND APPLICATIONS TO PROBABILITY ESTIMATION, CLASSIFICATION, AND TIME SERIES PREDICTION

by

PAUL VAN YEE

B.A.Sc. (Hon.), University of British Columbia, 1989

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfilment of the Requirements
for the Degree
Doctor of Philosophy

McMaster University

DOCTOR OF PHILOSOPHY (1998) McMaster University
(Department of Electrical and Computer Engineering) Hamilton, Ontario

TITLE: Regularized Radial Basis Function Networks: Theory and Applications to Probability Estimation, Classification, and Time Series Prediction

AUTHOR: Paul Van Yee

SUPERVISOR: Dr. Simon Haykin

NUMBER OF PAGES: x, 158

REGULARIZED RADIAL BASIS FUNCTION NETWORKS: THEORY & APPLIC.

Abstract

In this thesis, we study both theoretical and practical aspects of the regularized *strict interpolation radial basis function (SIRBFN)* estimate or neural network. From a theoretical perspective, we show that the regularized SIRBFN can be globally mean-square (m.s.) consistent whenever the Nadaraya-Watson regression estimate is and the regularization parameter sequence for the SIRBFN is chosen to be asymptotically optimal in the mean-squared fitting error. Hence we prove the Bayes risk consistency of the approximate Bayes decision rules formed from (m.s.-consistent) regularized SIRBFN posterior probability estimates. Similarly, we prove the m.s.-consistency of the regularized SIRBFN predictor for the class of Markovian nonlinear autoregressive time series generated by an i.i.d. noise process. In a one-step-ahead prediction experiment with a phonetically-balanced suite of male and female speech waveforms, the proposed predictor offers an average 2.2dB improvement in prediction SNR over corresponding exponentially-weighted RLS predictors. We also show that linearly combining an ensemble of three such proposed predictors via RLS filtering can yield an average 4.2dB improvement over the previous standard RLS predictors, and develop recursive algorithms to update the proposed predictor on-line with reduced computational complexity for certain situations. Two emerging application areas are then considered. The first is the regression-based approach to nonlinear filtering or state estimation, where the proposed network provides comparable performance to a recurrent MLP-based solution. The second is the dynamic reconstruction of chaotic systems from noisy observational data, where the reconstructed system is shown to generate sequences whose estimated long and short-term dynamical invariants agree closely with those of the original, noise-free system. Taken together, these theoretical and practical results point to the regularized SIRBFN as a principled design choice for RBF neural networks.

To my family,

for without their love and encouragement, none of this would have been possible,
and to all dedicated scholars,

for it is upon their shoulders that we see farther.

Acknowledgements

A work such as this cannot help but be the product of the interaction and influence of the many people whom I have had the good fortune to meet during the course of my studies. Among others, I would like to thank my supervisor Dr. Simon Haykin for his continuous support and patient guidance throughout these years; thesis committee members Dr. Zhi-Quan Luo and Dr. Gordon Slade for their thoughtful advice and conscientious assistance; and Dr. Anthony Vaz for sparking my interest in the field of financial engineering (among other topics). I also had the pleasure of collaborating with Dr. Eric Derbez and Dr. Sadasivan Puthusserypady, both of whom were kind enough to share their considerable expertise with me. To Tarun Bhattacharya, Robert Dingman, Graeme Jones, Vytas Kezys, and Andrew Ukrainec, I send sincere thanks for your collegiality and enlightening discussions in all matters, great and small. I am grateful to the Governments of Canada (through the Natural Sciences and Engineering Research Council) and the Province of Ontario (through the Ministry of Education and Training) for their financial support. Finally, I would like to acknowledge all those at the Communications Research Laboratory with whom I have had the pleasure of acquaintance; in more ways than one, the fruition of this thesis is due in no small measure to their kind help and good cheer.

Contents

List of Figures

List of Tables

Chapter 0

Notations and Abbreviations

$\mathbb{F}^+$	positive subfield of a field $\mathbb{F}$
$\mathbb{F}^{m\times n}$	class of $m \times n$ matrices with elements in field $\mathbb{F}$
$X, Y, \ldots$	random variables (r.v.s) denoted by upper case symbols
$x, y, \ldots$	realizations of r.v.s denoted by corresponding lower case symbols
$T_n, Z_n, \ldots$	length n sequences of r.v.s
$t_n, z_n, \ldots$	realizations of length n sequences of r.v.s
$\Pr\{A\}$	probability of an event A
$I_A(\bullet)$	indicator function (r.v.) for event A
$P_{\boldsymbol{X},\boldsymbol{Y}}$	joint probability measure for r.v.s $\boldsymbol{X}$, $\boldsymbol{Y}$
$\int dP_{\boldsymbol{X},\boldsymbol{Y}}, \int dP(\boldsymbol{x},\boldsymbol{y})$	integration with respect to joint probability measure for r.v.s $\boldsymbol{X}$, $\boldsymbol{Y}$
$P_{\boldsymbol{X}\|\boldsymbol{Y}}$	probability measure for r.v. $\boldsymbol{X}$ conditioned on r.v. $\boldsymbol{Y}$
$\int dP_{\boldsymbol{X}\|\boldsymbol{Y}}, \int dP(\boldsymbol{x}\|\boldsymbol{y})$	integration with respect to probability measure for r.v. $\boldsymbol{X}$ conditioned on r.v. $\boldsymbol{Y}$
$X \sim P$	r.v. X is distributed according to (measure) P
$\|\bullet\|_p$	p-norm (Euclidean and $p = 2$ unless otherwise specified)

$[\bullet]_{i=1}^{n}$	vectorization operator for scalar quantity $\bullet$ indexed by i
$[\bullet]_{i,j=1}^{m,n}$	matrification operator for scalar quantity $\bullet$ indexed by i, j
$\operatorname{diag}[\bullet_i]_{i=1}^{n}$	$n \times n$ diagonal matrix with i-th diagonal element equal to $\bullet_i$ and zero elsewhere
$\operatorname{diag}(\bullet_i, i \in I)$	$\lvert I \rvert \times \lvert I \rvert$ diagonal matrix with i-th diagonal element equal to i-th element of sequence $\{\bullet_i, i \in I\}$ and zero elsewhere
$\operatorname{diag}[\bullet]$	$\operatorname{diag}[\bullet_{ii}]_{i=1}^{n}$, where $\bullet$ is a $n \times n$ matrix
$[\bullet]_I$	subelement vector of $\bullet$ specified by index set I (scalar if I is a singleton)
$\{\bullet\}_I$	subsequence of $\bullet$ specified by index set I (single element if I is a singleton)
$i:j,\ i \le j,\ i,j \in \mathbb{Z}$	index set formed by integers i to j, inclusive
$a:\delta:b,\ a \le b,\ 0 \le \delta \le b-a,\ a,\delta,b \in \mathbb{R}$	set of reals from a to b step δ, inclusive
A, A^c	set and its complement
$A \Delta B$	symmetric difference of sets A and B
a.o.	asymptotically optimal (asymptotic optimality)
a.e.$(-\mu)$, a.s.$(-\mu)$	almost everywhere, almost surely (with respect to measure μ)
i.p.$(-\mu)$	in probability (with respect to measure μ)
m.s.(e.)	mean-square(d) (error)
m.s.f.e.	mean-square(d) fitting error
$a_n \nearrow a$	nondecreasing sequence $\{a_n\}$ with limit a
$a_n \searrow a$	nonincreasing sequence $\{a_n\}$ with limit a

$f(n) = \mathcal{O}(g(n))$	$\exists C > 0$ such that $\lvert f(n)\rvert \leq C\lvert g(n)\rvert$ for all n sufficiently large
$f(n) = \Omega(g(n))$	$\exists C > 0$ such that $\lvert f(n)\rvert \geq C\lvert g(n)\rvert$ for all n sufficiently large
supp f	support of the function f
KDE	kernel density estimate or estimator
NWRE	Nadaraya-Watson regression estimate or estimator
(SI)RBF(N)	(strict interpolation) radial basis function (network)
(S)LLN, (U)LLN	(strong), (uniform) law of large numbers
BRC	Bayes risk consistent (consistency)
GSM	geometrically strong (or α) mixing
GPM	geometrically ϕ-mixing
r.f.i.	recursively forward iterated (iterating)
w.s.s.	weak (wide) sense stationary (stationarity)

Chapter 1

Introduction

In the relatively recent field of *artificial neural networks (ANNs)*, two leading classes of feedforward networks have emerged: the *multilayer perceptron (MLP)*, usually with sigmoidal activation function and trained via the *backpropagation* algorithm, and the *radial basis function network (RBFN)*, often with the ubiquitous *Gaussian kernel*. Although both classes have met with considerable success in applications to such difficult engineering problems as nonlinear process estimation and control, there remains a fundamental gap between the theory and practice of *designing* and *applying* these networks. The former relates to the question of "how ought an RBFN be designed to fulfill a particular task" while the latter concerns itself with "for which tasks are RBFNs a justifiably good choice". Indeed, a review of the literature soon reveals the relative dearth of theory explaining the observed successes of ANNs in these areas. It is hoped that this thesis contributes in some small way to improving this situation for the class of RBFNs.

1.1 Problem Description

The type of RBFN that we shall be considering is a single-layer, feedforward network $\tilde{f} : \mathbb{R}^d \to \mathbb{R}$ that consists of a set of *weights* $\{w_i\}_{i=1}^n$ and a set of *basis functions* $\left\{G_i : \mathbb{R}^d \to \mathbb{R}\right\}_{i=1}^n$, where n is the number of basis or kernel functions. An important property of the RBFN is that it is a *linearly weighted* network, in the sense that the output is formed as

$$\tilde{f}(\bullet) = \sum_{i=1}^{n} w_i G_i(\bullet) \tag{1.1}$$

This linear combination of (typically) nonlinear basis functions is key to the RBFN's representational ability while maintaining computational and analytical tractability.

Specifically, the radial nature of the RBFN derives from the choice of basis functions G_i each of whose output is dependent only upon the distance of the input to another predetermined point $\boldsymbol{t}_i \in \mathbb{R}^d$. Such a set of radial basis functions can be derived from a common kernel $G : \mathbb{R}^+ \to \mathbb{R}$ via

$$G_i(\bullet) \triangleq G\left(\|\bullet - \boldsymbol{t}_i\|\right), \qquad i = 1, 2, \ldots, n \tag{1.2}$$

where the $\{\boldsymbol{t}_i\}_{i=1}^n$ are collectively known as the *centres* of the network. The particular norm used can be chosen to reflect prior knowledge regarding the nature of the input space, as we shall see in Section 1.2.

Here and thereafter, we shall focus our attention on the problem of *estimating* an unknown input-output mapping $f : \mathbb{R}^d \to \mathbb{R}$ from a finite set T_N of N examples for that function's behaviour. In ANN parlance, this task is commonly referred to as *learning* or *training the network* and T_N is called the *training set*. Three common *a priori* models for T_N are:

1. $T_N \triangleq \{(\boldsymbol{x}_i, y_i = f(\boldsymbol{x}_i))\}_{i=1}^N$ (the *noise-free* case).
2. $T_N \triangleq \{(\boldsymbol{x}_i, y_i = f(\boldsymbol{x}_i) + \epsilon_i)\}_{i=1}^N$, where $\{\epsilon_i\}$ is a partially known random process (the *noisy* case).
3. $T_N \triangleq \{(\boldsymbol{X}_i, Y_i)\}_{i=1}^N$, where the $(\boldsymbol{X}_i, Y_i)$ are random samples with an unknown common stationary joint probability measure $P_{\boldsymbol{X}Y}$ and bounded variance (the *stochastic* case). In this case, f is implicitly defined by the task at hand, e.g., for minimum mean-square error (m.m.s.e.) estimation, $f(\bullet) \triangleq \mathbb{E}[Y | \boldsymbol{X} = \bullet]$.

When necessary to do so explicitly, we shall denote the space of possible realizations of T_N by τ^N. It is clear then that once the RBFN has been selected by some means as an appropriate structure for f, the problem of "learning" is equivalent to one of estimating the "best" set of network parameters, i.e., weights, basis functions, centres, norms, etc., from T_N. Although the meaning of "best" is problem-dependent, in the ideal case one would like the optimality to extend to regions in the *graph* of f, i.e., the set of ordered pairs $\left\{(\boldsymbol{x}, f(\boldsymbol{x})) : \boldsymbol{x} \in \mathbb{R}^d\right\}$, for which no examples can be found in T_N (otherwise a simple memorization of T_N would suffice). For example, when the input $\boldsymbol{X}$ is a random variable (r.v.) described by a distribution $P_{\boldsymbol{X}}$ and quality is measured with respect to the mean-square (m.s.) estimation error or *risk* R_2^*, we would like to solve the optimization problem

of finding the parameter set $\boldsymbol{\theta}^*$ satisfying

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^m} R_2^*\left(f, \tilde{f}\left(\bullet, \boldsymbol{\theta}\right)\right), \qquad R_2^*\left(f, \tilde{f}(\bullet, \boldsymbol{\theta})\right) \triangleq \int \left(f(\boldsymbol{x}) - \tilde{f}\left(\boldsymbol{x}, \boldsymbol{\theta}\right)\right)^2 dP_{\boldsymbol{X}}(\boldsymbol{x}) \tag{1.3}$$

where $\boldsymbol{\theta} \in \mathbb{R}^m$ represents the m free parameters of the network. The ANN term *generalization* refers precisely to the ability of a network to extrapolate its performance from the training set to the entire graph of f (or a significant portion thereof). Note that the ideal generalization measure R_2^* is dependent only on m, assuming a given parameterized function class, target function f, and input distribution $P_{\boldsymbol{X}}$. In practice, however, we shall be estimating $\tilde{f} = \tilde{f}_N$ on the basis of a realized training set t_N of T_N and possibly some *a priori* information, so that the ideal generalization measure R_2^* should be modified to read

$$R_2\left(f, \tilde{f}_N\right) \triangleq \int \mathbb{E}_{T_N}\left[\left(f(\boldsymbol{x}) - \tilde{f}_N\left(\boldsymbol{x}, \boldsymbol{\theta}\right)\right)^2\right] dP_{\boldsymbol{X}}(\boldsymbol{x}) = \mathbb{E}\left[\left(f(\boldsymbol{X}) - \tilde{f}_N\left(\boldsymbol{X}, \boldsymbol{\theta}\right)\right)^2\right] \tag{1.4}$$

i.e., the m.s. risk averaged over all possible inputs and training sets (this is also known as the *global* m.s. risk). Of course, other variants of the global m.s. risk are possible by averaging over either only the input r.v. $\boldsymbol{X}$ or the training set r.v. T_N in the expectation. Nonetheless, compared to the ideal generalization measure, this training set-dependent generalization measure may now additionally be a function of N and the joint distribution of T_N (in the noisy and stochastic cases). For any reasonable network training procedure, we would expect R_2 to be nonincreasing with increasing m and N. In practice, $m = m(N)$ (as will be explained in the comments regarding network complexity) so that R_2 is primarily indexed by N, the number of available training data.

Notwithstanding the above modifications, the minimization of even R_2 in (1.4) with respect to $\boldsymbol{\theta}$ poses some obvious practical difficulties:

1. the optimal network parameters in $\boldsymbol{\theta}^*$ are dependent upon the behaviour of f, the unknown function of interest, outside of t_N, the particular realization of the training set T_N we happen to have available.

2. the joint distribution P_{T_N} of the training data is often not known exactly and must also be inferred from the sample training data in t_N. An estimate of the input (marginal) distribution $P_{\boldsymbol{X}}$ can then be derived from the estimate of the joint distribution, assuming that stationarity holds.

These two uncertainties imply that in all nontrivial cases θ^* can only be approximately determined. Of course, the availability of T_N partially offsets this lack of knowledge provided that T_N "adequately" captures the behaviour of f and, in the cases of noisy or stochastic T_N, the underlying stochastic processes governing $(\boldsymbol{X}, Y)$. An intuitively obvious solution to this problem is to replace the desired objective or cost function involving f with a *proxy* computable using only a training realization t_N and $\tilde{f}$. Returning to the above example of the noise-free case, for uniformly-weighted (nonprobabilistic) inputs with $dP_{\boldsymbol{X}} = d\lambda$, where λ is the usual Lebesgue measure in $\mathbb{R}^d$, we may replace R_2 with $\tilde{J}_2$, where

$$\tilde{J}_2\left(\tilde{f}, t_N\right) \triangleq \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \tilde{f}\left(\boldsymbol{x}_i\right)\right)^2 \tag{1.5}$$

As $N \to \infty$, the desire is that the parameter vector $\tilde{\theta}^*$ minimizing the proxy yield a network $\tilde{f}^*$ which converges to the (optimum) network f^* that would have been obtained had the exact cost function been used. In other words, we are interested in the convergence of $\tilde{f}^*$ to f^* in some appropriate mode, e.g., pointwise *in probability (i.p.)* or *weakly*, *almost surely (a.s.)* or *strongly*, and in $L_p(D, P_{\boldsymbol{X}})$ norm for $1 \leq p \leq \infty$ and some (compact) set $D \subseteq \mathbb{R}^d$. An estimator which satisfies such a convergence condition in a given mode is said to be *consistent* in that mode. If the consistency holds for all possible distributions $P_{\boldsymbol{X}}$ (for noisy T_N) or $P_{\boldsymbol{X}Y}$ (for stochastic T_N), we say that the estimator is *universally* consistent.

We now note an important equivalence between the functional consistency with which we are interested and the more usual notion of *risk consistency* used in the statistics literature. Using our previous notations, risk consistency occurs when the parameter vector $\tilde{\theta}^*$ minimizing the proxy yields a network $\tilde{f}^*$ with *performance* converging to that of the network f^* which is optimal with respect to the exact cost function. For the case of stochastic T_N and performance measured by the m.s. output estimation error or risk

$$J_2(f) \triangleq \mathbb{E}\left[|Y - f(\boldsymbol{X})|^2\right] \tag{1.6}$$

the optimum estimator is well-known to be $f^*(\bullet) \triangleq \mathbb{E}[Y|\boldsymbol{X} = \bullet]$ (assuming $\mathcal{F}$ contains the conditional mean function). Then J_2-*risk consistency* is equivalent to our functional $L_2\left(\mathbb{R}^d, P_{\boldsymbol{X}Y}\right)$-consistency, since

$$\begin{aligned} J_2\left(\tilde{f}^*\right) - J_2\left(f^*\right) &= \mathbb{E}\left[\tilde{f}^*(\boldsymbol{X})^2 - 2\tilde{f}^*(\boldsymbol{X})Y + Y^2 - f^*(\boldsymbol{X})^2 + 2f^*(\boldsymbol{X})Y - Y^2\right] \\ &= \mathbb{E}\left[\tilde{f}^*(\boldsymbol{X})^2\right] - 2\mathbb{E}\left[\tilde{f}^*(\boldsymbol{X})f^*(\boldsymbol{X})\right] + \mathbb{E}\left[f^*(\boldsymbol{X})^2\right] \\ &= \mathbb{E}\left[\left(f^*(\boldsymbol{X}) - \tilde{f}^*(\boldsymbol{X})\right)^2\right] \end{aligned} \tag{1.7}$$

where (X, Y) are assumed independent of T_N and we have used the facts that

$$\mathbf{E}[(Y - f^*(X))\, f^*(X)] = 0, \qquad \mathbf{E}[g(X) f^*(X)] = \mathbf{E}[g(X) Y] \tag{1.8}$$

for all Borel measurable functions g such that the expectations exist. Thus for this case of *regression* estimation, there is no loss of generality in studying functional consistency versus output estimation consistency.

Although consistency is an asymptotic property, it is considered one of the most desirable fundamental properties of an estimator and is consequently a key object of study. Note that the functional consistency with which we are concerned is a weaker condition than the consistency of the optimal proxy parameter vector $\tilde{\theta}^*$ with respect to the optimal exact parameter vector θ^*; when $\tilde{f}$ is a continuous function, the latter implies the former but the reverse need not be true.

In summary, given a task and an associated training set T_N, the problem of *design* for a RBFN is one of choosing the network parameters that minimize the risk and thereby maximize the desired generalization performance. Since achieving this aim would require knowledge of the unknown function f being estimated, the next logical goal is to produce estimates $\tilde{f}$ of f using T_N (and possibly some prior knowledge about f) with performance converging in N to that of f as rapidly as possible.

1.2 Review of Current Approaches

In the previous section, we have described the problem of RBFN design in rather general terms. Here we shall survey the recent history of RBFN design and examine the some of the major current approaches found in both the ANN and related statistical literature, particularly from the area of *kernel regression estimates (KRE)* (Eubank, 1988; Härdle, 1990). Again the primary objective is to lay bare the design choices made by each method so that their theoretical properties may be properly studied.

1.2.1 Traditional ANN Approaches

The seminal work of Broomhead and Lowe (1988) is representative of early attempts at RBFN design. This school takes the principled view of learning as a problem in *multivariable functional interpolation* from scattered data. In this approach, the kernel function and its parameters are fixed beforehand so that the only the centres and linear

weights need to be determined. The centres may be (randomly) chosen as a subset of the training input data or, more generally, via a *self-organizing* procedure such as *k-means clustering*, in which case the network centres can be different from the training input data. The network weights $\boldsymbol{w}$ are then computed as the solution to the L_2-interpolation problem

$$\boldsymbol{w} = \arg\min_{\omega \in \mathbb{R}^n} \|\boldsymbol{y} - \boldsymbol{G}\omega\|^2 \tag{1.9}$$

where $\boldsymbol{y} \in \mathbb{R}^N$ is the vector of desired output values in T_N and $\boldsymbol{G} \in \mathbb{R}^{N \times n}$ with entries $[\boldsymbol{G}]_{ij} = G(\|\boldsymbol{x}_i - \boldsymbol{t}_j\|/h)$ is the *interpolation matrix* for this problem. The required solution for $\boldsymbol{w}$ is the classical pseudo-inverse formula for an overdetermined linear least-squares problem

$$\boldsymbol{w} = \boldsymbol{G}^+ \boldsymbol{y}, \qquad \boldsymbol{G}^+ \triangleq \left(\boldsymbol{G}^\top \boldsymbol{G}\right)^{-1} \boldsymbol{G}^\top \tag{1.10}$$

The rationale behind this least-square approach is that when the number of training data is much larger than the "number of degrees of freedom" of the underlying function, fitting all the available training data exactly (as would occur if one basis function is assigned to each training datum) results in an estimate $\tilde{f}$ which models irrelevant behaviour as implied by imprecise or noisy training data. In allowing the fit to deviate from the training data, the least-squares solution essentially imposes a *smoothness constraint* on the estimate which, as we shall see shortly, is a basic form of *regularization*. Thus even at this early juncture, we see recognition of the central role that regularization plays obtaining well-behaved estimates from sample data.

Somewhat later, motivated by developments in the related area of *splines* for image surface reconstruction in computer vision, Poggio and Girosi (1990) formulated the seminal approach to RBFN design based on *Tikhonov regularization* (ill-posed problems, 1977). As in splines, the function estimate is derived as the solution to an infinite-dimensional variational problem over a given (Hilbert) space $\mathcal{H}$ of typically "smooth" functions. The problem is to minimize a linear functional H over $\mathcal{H}$ consisting of two terms: the first term, H_1, being the standard sum-of-squared fitting errors over T_N, while the second term, H_2, is a roughness penalty typically measured through the norm induced by a *pseudo-differential* operator $P : \mathcal{H} \to \mathcal{H}$. More precisely, we set

$$Hf \triangleq H_1 f + \lambda H_2 f, \quad H_1 f \triangleq \sum_{i=1}^{N} (y_i - f(\boldsymbol{x}_i))^2, \quad H_2 f \triangleq \|Pf\|^2 \tag{1.11}$$

and seek $\tilde{f}$ satisfying

$$\tilde{f} = \arg\min_{f \in \mathcal{H}} Hf \tag{1.12}$$

where $\lambda \in \mathbb{R}^+$ is the *regularization parameter* which balances the fidelity of $\tilde{f}$ to T_N with the smoothness of $\tilde{f}$. Setting $\lambda = 0$ yields a standard least-squares solution while $\lambda \to \infty$ yields (as we shall show) an estimate closely related to classical KRE methods. This functional optimization approach has the advantage that with the proper choice of P and $\mathcal{H}$, the unique minimizer of (1.11) is precisely the RBFN described by (1.1) and can be shown to have the following characteristics (Yee, 1992; Poggio and Girosi, 1990):

1. each input datum in T_N is a centre for the RBFN, i.e., $n = N$ and $t_i = x_i$, $i = 1, 2, \ldots, n$. This situation is referred to as the *strict interpolation (SI)* case and will be seen to have direct links to the classical KRE methods.

2. for $\mathcal{H} = S$, the Hilbert space of rapidly decreasing, infinitely continuously differentiable functions found in the Schwartz theory of tempered distributions (Friedlander, 1982), and P and its adjoint P^* satisfying

$$P^*P = \sum_{n=0}^{\infty} \frac{(-1)^n}{n!2^n} \nabla_U^{2n}, \qquad \nabla_U^2 \triangleq \sum_{i=1}^{d} \sum_{j=1}^{d} u_{ij} \frac{\partial^2}{\partial x_i \partial x_j} \tag{1.13}$$

   we obtain the important case of the Gaussian kernel with norm weighting matrix $U \triangleq [u_{ij}]_{i,j=1}^n$, viz.

$$G_i(\bullet) = Gr(\bullet, x_i) = \exp\left(-\|\bullet - x_i\|_U^2 / 2\right) \tag{1.14}$$

   where U is assumed to be symmetric positive-definite and $\|\bullet\|_U^2 \triangleq \bullet^\top U \bullet$. More generally, the dyadic function Gr from which the basis functions G_i are derived corresponds to the *Green's function* for the self-adjoint operator P^*P.

3. the network weights $w \triangleq [w_i]_{i=1}^n$ satisfy the *SI* equation over T_N

$$(G + \lambda I)\, w = y \tag{1.15}$$

   In the strict interpolation case, when the G_i are derived from a *positive-definite kernel*[1], G can be shown to be (in principle) positive definite for any distinct set of

[1] recall that a dyadic function G is positive-definite if for all n and $a_1, a_2, \ldots, a_n$, $t_1, t_2, \ldots, t_n$, we have $\sum_{i,j=1}^n a_i a_j G(t_i, t_j) \geq 0$.

centres $\{\boldsymbol{x}_i\}$ (Light, 1992), hence (1.15) always has a solution for $\lambda \geq 0$. In actual implementations such as those we describe in the forthcoming simulation experiments, standard precautions should be taken when n is large to ensure a stable solution to (1.15), e.g., choosing λ sufficiently large and using indirect solution methods based on matrix factorizations such as the LU-decomposition.

We note that the optimum network weights can also be obtained by assuming *a priori* that the estimate $\tilde{f}$ lies in the linear span of $\{G_i\}_{i=1}^n$ and seeking the solution to the constrained problem

$$\boldsymbol{w} = \arg \min_{\boldsymbol{\omega} \in \mathbb{R}^n} \|\boldsymbol{y} - \boldsymbol{G}\boldsymbol{\omega}\| \text{ subject to } \left\|\sqrt{\boldsymbol{G}}\boldsymbol{\omega}\right\| = c \tag{1.16}$$

for some $c > 0$ and where $\sqrt{\boldsymbol{G}}$ is a square-root matrix for $\boldsymbol{G}$ (which, by the positive-definiteness above, always exists). By the Lagrange multiplier method, this condition is equivalent to

$$\boldsymbol{w} = \arg \min_{\boldsymbol{\omega} \in \mathbb{R}^n} \left(\|\boldsymbol{y} - \boldsymbol{G}\boldsymbol{\omega}\|^2 + \lambda \left\|\sqrt{\boldsymbol{G}}\boldsymbol{\omega}\right\|^2 \right) \tag{1.17}$$

The network weights have the further property that they are directly proportional to the interpolation errors over T_N through the regularization parameter λ, i.e.,

$$\lambda w_i = y_i - \tilde{f}(\boldsymbol{x}_i) \tag{1.18}$$

Although it is clear from this relationship that $\lambda = 0$ yields an exact fit to T_N, the situation in the other limiting case $\lambda \to \infty$ is not as obvious.

Further extensions to the method involve setting selecting only $n < N$ centres $\{\boldsymbol{t}_i\}_{i=1}^n$ from the available training centres $\{\boldsymbol{x}_i\}_{i=1}^N$. If we then *assume* that (1.1) is the correct form for $\tilde{f}$, i.e., $f \in \text{span}(\{G_i\}_{i=1}^n)$, the network weights can then be shown to satisfy

$$\left(\boldsymbol{G}^{\mathsf{T}}\boldsymbol{G} + \lambda\boldsymbol{\Gamma}\right)\boldsymbol{w} = \boldsymbol{G}^{\mathsf{T}}\boldsymbol{y} \tag{1.19}$$

where $\boldsymbol{G} = [G_j(\boldsymbol{x}_i)]_{i,j=1}^{N,n}$, $\boldsymbol{\Gamma} = [G_j(\boldsymbol{t}_i)]_{i,j=1}^n$, and $G_i(\bullet) \triangleq G(\|\bullet - \boldsymbol{t}_i\|_U)$, $i = 1, 2, \ldots, n$. Note these definitions are general in that the SI case can be obtained by setting $\boldsymbol{t}_i = \boldsymbol{x}_i$, $i = 1, 2, \ldots, n = N$. We also note that the the pseudo-inverse equation (1.10) for the optimal weights of the Broomhead and Lowe (1988) approach corresponds to $\lambda = 0$ in (1.19). This

formal correspondence implies that when the kernel G can be derived from the Green's function of some self-adjoint operator of the form P^*P, the approach of Broomhead and Lowe (1988) coincides with the optimal solution to the regularized fitting problem (1.11) and (1.12) over $\mathcal{H} = \text{span}\,(\{G_i\}_{i=1}^n)$ in the limit $\lambda \to 0$. Because their method does not require the kernel G to be a Green's function, the method of Broomhead and Lowe (1988) is, strictly speaking, less restrictive than that of Poggio and Girosi (1990). On the other hand, from a theoretical perspective, their method consequently lacks a means of interpreting the assumptions underlying the choice of basis function. Furthermore, from a practical perspective, it is not certain what computational advantages their more general formulation offers, as both methods can lead to networks with "universal approximation" capabilities, as described in the next section. We shall see later in remark 3 of Chapter 2 that for noisy T_n, the choice $\lambda = 0$ is risk-optimal only when the (additive) noise variance is zero, i.e., the training data are exact. In this respect, the explicit Tikhonov regularization afforded by the method of Poggio and Girosi (1990) improves upon the regularization implicit in the straightforward least-squares solution of Broomhead and Lowe (1988).

For even greater network flexibility, Poggio and Girosi (1990) propose the *hyper basis function (HyperBF)* method in which one attempts to optimize (1.11) simultaneously with respect to all free network parameters, i.e., weights, centres, and the entries of the norm weighting matrix. The original work suggests optimization via simple gradient descent , although more sophisticated optimization methods such as conjugate-gradient search (Hutchinson, 1994) have also been used more recently.

1.2.2 Limitations of the ANN Approaches

Since the approach of Poggio and Girosi (1990) subsumes that of Broomhead and Lowe (1988) as a special case, we shall concentrate on the former. In the simplest case of $\lambda = 0$, we see that $H_1 f = N\tilde{J}_2\,(f, T_N)$, the sum-of-squared fitting errors over T_N, is being used as a (scaled) proxy for the desired generalization measure R_2 described previously. Justification for such proxies in the general case can be found in the principle of *empirical risk minimization (ERM)*, which in turn is supported by various theories. In the following discussion, we shall consider the theory of *uniform convergence of empirical averages to their mathematical expectations* as developed by Vapnik (1982); other related approaches include those under the name of *uniform strong law of large numbers (USLLN)* and lead to

similar results (Pollard, 1984; Gallant, 1987). For noise-free T_N consisting of independent, identically distributed (i.i.d.) examples $y_i = f(\boldsymbol{x}_i) \in \{0, 1\}$, this uniform convergence is characterized by an intrinsic coefficient of the combinatorial complexity of the class of functions $\mathcal{F}$, assumed to contain both f and its approximant $\tilde{f}$. This coefficient is called the *Vapnik-Chervonenkis (VC) dimension* of $\mathcal{F}$ (denoted $d_{VC}(\mathcal{F})$) (Vapnik, 1982) and (roughly speaking) indicates the maximum N for which all possible 2^N dichotomies over T_N can be induced by some function in $\mathcal{F}$. A major result of the theory is that a finite $d_{VC}(\mathcal{F})$ is both necessary and sufficient for a distribution-free, worst-case upper bound on the probability of a large deviation between a given desired generalization measure and its empirical proxy to hold. For the usual case $\mathcal{F} = \mathcal{F}_{\boldsymbol{\theta}}$, a function class parameterized by a vector $\boldsymbol{\theta} \in \mathbb{R}^m$, and letting $\tilde{f}_{\boldsymbol{\theta}}$ denote the function in $\mathcal{F}_{\boldsymbol{\theta}}$ determined by $\boldsymbol{\theta}$, we have the result that for any distribution on $\boldsymbol{X}$,

$$\Pr\left\{\sup_{\boldsymbol{\theta}\in\mathbb{R}^m}\left|\tilde{R}\left(\tilde{f}_{\boldsymbol{\theta}}, T_N\right) - R\left(f, \tilde{f}_{\boldsymbol{\theta}}\right)\right| > \epsilon\right\} < C\left(d_{VC}\right)\exp\left(-K\epsilon^2 N\right) \tag{1.20}$$

where $\tilde{R}\left(\tilde{f}_{\boldsymbol{\theta}}, T_N\right)$ and $R\left(f, \tilde{f}_{\boldsymbol{\theta}}\right)$ denote general empirical and actual risk functions, respectively, with $C\left(d_{VC}\right)$ a function growing at most polynomially in N and K a positive constant (independent of N). From this result, we see that if, for each N, we can find a $\tilde{\boldsymbol{\theta}}_N^*$ that minimizes $\tilde{R}\left(\tilde{f}_{\boldsymbol{\theta}}, T_N\right)$, then the sequence of networks $\tilde{f}_{\tilde{\boldsymbol{\theta}}_N^*}$ so obtained has generalization performance converging a.s. to the best possible over all approximating functions, i.e., we have a.s. risk consistency.

To justify choosing $\lambda > 0$, Poggio and Girosi (1990) offer a Bayesian interpretation of the cost functional Hf under the scenario of noisy T_N. The first term, $H_1 f$, corresponds to the negative log-likelihood of T_N assuming that the additive measurement noise ϵ_i is white Gaussian, while the second term, $H_2 f$, corresponds to the negative logarithm of the (improper) prior probability $P(f) \triangleq \exp\left(-\lambda\|Pf\|^2\right)$, i.e., λ determines the "variance" or "spread" of the prior probability over the space of candidate functions. Solving for the *maximum a posteriori (MAP)* estimate under these assumptions is equivalent to the minimization of Hf. A related point of view can be found in the principle of *minimum description length (MDL)* for model selection (Rissanen, 1978). Here Hf is the *hypothesis complexity* of the model function f, composed of $H_1 f$, which is the *data description length/complexity*, and $H_2 f$, which is *model description length/complexity*. The role of λ is to balance these two competing quantities based on some *a priori* knowledge or preferences. Both of these interpretations, however, do not offer any explicit procedure to select $\lambda > 0$.

A somewhat more forthright selection criterion for $\lambda > 0$ can be formulated from the principle of *structural risk minimization (SRM)* (Guyon et al., 1992; Vapnik, 1992). Returning to the VC-dimension theory for binary-valued functions previously introduced, define the *guaranteed risk* as the sum of the empirical risk $\tilde{R}$ and the deviation ϵ in (1.20)) expressed as a function of the bounding probability $\alpha \triangleq C(d_{VC}) \exp(-K\epsilon^2 N)$. We can therefore assert

$$\Pr\left\{R\left(f, \tilde{f}_\theta\right) < \tilde{R}\left(\tilde{f}_\theta, T_N\right) + \epsilon(\alpha)\right\} > 1 - \alpha \tag{1.21}$$

For given N, the empirical risk is a nonincreasing function of d_{VC} while the deviation function or *confidence interval* $\epsilon(\alpha)$ is nondecreasing in d_{VC}. Thus their sum, the guaranteed risk, achieves its minimum with respect to d_{VC}, say at $d_{VC} = d_{VC}^*$. The principle of SRM is to seek the function $\tilde{f}^*$ with this guaranteed-risk-minimizing VC-dimension d_{VC}^* by solving a suitable sequence minimization problems over a corresponding sequence of subsets of the candidate function space $\mathcal{F}_\theta$. The subset sequence is structured to have monotonically increasing VC-dimension, hence nonincreasing empirical risk and nondecreasing confidence interval. Within each subset, we find the empirical-risk-minimizing network and select the subset containing the network with smallest overall empirical risk. Over this best subset, we then seek the network with minimum guaranteed risk. The rationale for SRM is that for a fixed quantity of training data T_N, the *capacity* of the network, measured by d_{VC}, should be matched to the amount of information available in T_N to avoid the problems of *underfitting* (in the case $d_{VC}(\mathcal{F}_\theta) < d_{VC}^*$) and *overfitting* (in the case $d_{VC}(\mathcal{F}_\theta) > d_{VC}^*$). In the former case, the candidate function space is underparameterized and therefore lacks sufficient flexibility to model the relationships in T_N, leading to estimator bias. The opposite is true in the latter case where $\mathcal{F}_\theta$ contains too many free parameters for the given data in T_N to specify accurately, resulting in excessive estimator variance. For more precise details on these issues, the reader may refer to (Geman et al., 1992).

For the case of linearly weighted networks such as the RBFN, an appropriate structure may be introduced in the form of *weight decay*. Following our previous notation, let $\mathcal{F}_k \triangleq \left\{\tilde{f}_\theta : \|\theta\|_E \le c_k\right\}$, $k = 1, 2, \ldots, M$, where $\|\bullet\|_E$ is a suitable weighted Euclidean norm in $\mathbb{R}^m$, and $\{c_k\}_{k=1}^M$ is a strictly increasing sequence of positive reals. Note that by construction, $d_{VC}(\mathcal{F}_j) < d_{VC}(\mathcal{F}_k)$ for $j < k$. To minimize the empirical risk over each

subset $\mathcal{F}_k$, we may use the Lagrange multiplier technique to seek

$$\theta_k^* \triangleq \min_{\theta \in \mathbb{R}^m} \left(\tilde{R}\left(\tilde{f}_\theta, T_N\right) + \lambda_k \|\theta\|_E \right) \tag{1.22}$$

and thereby set $\tilde{R}_k^* \triangleq \tilde{R}\left(\tilde{f}_{\theta_k^*}, T_N\right)$, where $\{\lambda_k\}_{k=1}^M$ is a strictly decreasing sequence of positive reals. Hence for this structure, the principle of SRM chooses as optimal the regularization parameter $\tilde{\lambda}^* = \lambda_j$, where $\mathcal{F}_j$ satisfies $\tilde{R}_j^* \leq \tilde{R}_i^*$ for $j \neq i$, $i = 1, 2, \ldots, M$. For a sufficiently large number of subsets M, it is hoped that one will contain a network with VC-dimension d_{VC}^*.

The basic ERM/SRM framework described above has been significantly extended, particularly in two respects. First, by using the concept of *covering number* (Pollard, 1984) in place of the VC-dimension, the range of the unknown function f and its approximating function space $\mathcal{F}$ can be $\mathbb{R}$ instead of $\{0, 1\}$. Second, similar to the method of SRM, by considering a suitable increasing sequence of candidate function spaces $\mathcal{F}_j \subset \mathcal{F}_k \subset \mathcal{F}$ for $j < k$, we can relax the assumption $f \in \mathcal{F}$ by choosing $\mathcal{F} \triangleq \cup_{k=1}^{\infty} \mathcal{F}_k$ to be dense in a function space large enough to contain f, e.g., $L_p(D, P_X)$ for any distribution P_X. The general notion of allowing $\mathcal{F}$ to grow with N was studied by Grenander (1981) under the name *method of sieves* while the latter property of density was earlier studied for RBFNs by Park and Sandberg (1991, 1993) and commonly called the *universal approximation property (UAP)* in the ANN literature. The seminal work of Krzyżak et al. (1996) combines these two extensions to provide conditions under which SRM-type RBFNs are J_2-risk consistent both i.p. and universally a.s. Also given are distribution-free finite sample upper bounds on the probability of a large deviation between the classification error rates of the empirical and actual classification-error-minimizing RBFNs.

Parallel to the general mathematical developments of the ERM/SRM theory, Niyogi and Girosi (1996) directly analyse nonregularized RBFNs trained with ERM and stochastic T_n to prove their consistency by providing probably-approximately-correct convergence rates on their generalization error $R_2\left(f, \tilde{f}_{n,m}\right)$, where, as before, m is the number of free network parameters. While sharing the same spirit, the results obtained are not universal in the same way that the ERM results are, e.g., the conditional mean f is assumed to belong to a class of functions generated by the convolution of a Gaussian kernel with a signed Radon measure of bounded total variation. A regularized case is addressed in Corradi and White (1995) where the total squared error of two types of "sufficiently kinky polynomial splines" in one dimension (Stinchombe and White, 1990) is studied. There they

show that for noisy T_n with i.i.d. measurement errors and f assumed to lie in a suitable space of smooth functions (specifically a *reproducing kernel Hilbert space (RKHS)* (Aronszajn, 1950; Wahba, 1990)), such regularization networks' error converges at a rate which is optimal with respect to the smoothness of f as measured by the order of its differentiability. The kernel functions considered therein, however, do not include the usual radial cases such as the Gaussian kernel commonly in use.

Despite the theoretical justification for their RBFN design procedures, the ERM-based methods such as the HyperBF method discussed previously suffer from a number of limitations:

1. because they are usually nonlinear in multiple parameters, the empirical risk functions to be minimized are subject to *local minima*. To avoid this difficulty, one may reduce the number of free parameters for the candidate functions space $\mathcal{F}$ from that of the most general one considered in the HyperBF method. Even then, unless density arguments are offered, as Krzyżak et al. (1996) do, one cannot be certain that the restricted class is sufficiently broad to include (eventually) the unknown target function, making the argument for consistency tenuous.

2. for the SRM-based methods such as weight decay, it is unclear exactly how the sequences defining the subsets are determined, given a training set T_N. The previous comment on the difficulty of ERM over just a single candidate function space raises questions about the feasibility of repeating the constrained minimizations over each subset as required by SRM.

3. as currently developed, the ERM framework addresses neither the case of *correlated examples* in the training set T_N, as would occur in applications involving *time series*, nor that of *nonstationary distributions* $P_{\mathbf{X}Y}$, e.g, in the case of noisy T_N with *heteroskedastic* errors ϵ_i.

Given that the ERM-approach to the theory of ANN is relatively recent, one can expect these issues to be resolved for RBFNs after further study. Indeed, we would be remiss if we did not mention the work of White (1990) with MLP networks in this regard. There, in an early application of the method of sieves and the concept of *metric entropy*, White (1990) proves the weak m.s. consistency of single-layer, feedforward MLP networks with complexity determined by the method of *ordinary cross-validation (OCV)* for both

i.i.d. and stationary ϕ or *α-mixing* inputs (both to be discussed further on). Since mixing processes are correlated, the issue raised in point 3 has been advanced and the same is true for the regularization induced by OCV with respect to point 2. Point 1 is answered in part by a result stating the convergence i.p. of network parameters determined by approximate minimization to a neighbourhood of the truly optimal network parameters when $N \to \infty$. What we shall show, however, is that from an ANN point of view, there exists another route to justifying the theoretical and practical RBFN design choices by adapting results from classical KRE theory, which is the subject of the next section.

1.2.3 The Kernel Regression Approach

With its earlier beginning compared to ANNs, the field of KRE is understandably more mature, especially in terms of theory. KRE has its roots in the ideas of nonparametric *kernel density estimation (KDE)* (Parzen, 1962; Cacoullos, 1966), whose origin can be traced back to the well-known frequentist histogram. The generalization from the histogram's effectively rectangular kernel to other kernel shapes as well as the nonlinear *nearest neighbour* type of density estimators soon followed along with commensurate consistency arguments. Particularly appealing is the simplicity of the KDE, which, given an i.i.d. training set $T_n \triangleq \{\boldsymbol{X}_i\}_{i=1}^n$ with product distribution $P_{T_n} \triangleq \prod_{i=1}^n P_{\boldsymbol{X}}$, has the general form

$$\tilde{f}_n(\bullet) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{\bullet - \boldsymbol{X}_i}{h_n}\right) \tag{1.23}$$

with $K : \mathbb{R}^d \to \mathbb{R}$ satisfying

$$0 \leq K(\boldsymbol{x}) < \infty, \quad K(\boldsymbol{x}) = K(-\boldsymbol{x}), \quad \text{for all } \boldsymbol{x} \in \mathbb{R}^d \tag{1.24}$$

$$\lim_{\|\boldsymbol{x}\| \to \infty} \|\boldsymbol{x}\|^d K(\boldsymbol{x}) = 0, \quad \int K = 1, \quad \int \|\boldsymbol{x}\|^2 K(\boldsymbol{x})\, d\lambda(\boldsymbol{x}) < \infty \tag{1.25}$$

where λ indicates Lebesgue measure. By considering the case of the histogram kernel $K(\bullet) \triangleq I_{[-1/2,+1/2]^d}(\bullet)$, it is intuitively clear that $h_n \overset{n\to\infty}{\to} 0$ in order for the approximate density to converge (pointwise) to the actual density. On the other hand, since the expected number of training points in T_n falling inside a hypercube with edge h_n centred about a given point $\boldsymbol{x} \in \mathbb{R}^d$ is roughly $nh_n^d f(\boldsymbol{x})$, we should also require that $nh_n^d \overset{n\to\infty}{\to} \infty$. Remarkably, it turns out that these two conditions, namely

$$h_n \overset{n\to\infty}{\to} 0, \qquad nh_n^d \overset{n\to\infty}{\to} \infty \tag{1.26}$$

are both necessary and sufficient for the L_2 and a.s. pointwise consistency of $\tilde{f}_n$ under some mild assumptions (Györfi et al., 1989; Bosq, 1996).

If K is also radially symmetric so that $K\left(\left(\bullet - \boldsymbol{X}_i\right)/h_n\right) = G_i(\bullet)$ in (1.1), we see an immediate similarity between the basic KDE and a restricted form of the SIRBFN with uniform weights $w_i \triangleq 1/\left(nh_n^d\right)$, $i = 1, 2, \ldots, n$, and norm weighting matrix $\boldsymbol{U}_n \triangleq \boldsymbol{I}/h_n$. Of course, an interpretation of the conditions under which such a solution would reasonably occur is more problematic, as there is usually no set of explicit output targets provided with T_n in density estimation. We shall see later, however, that a RBFN trained using the class indicator values as the output targets in T_n corresponds to L_2-estimates of posterior probabilities. We should further mention that the KDE can also be derived as the solution to an appropriately regularized functional minimization problem similar to that for the SIRBFN. The description here follows that of Section 9.9 in (Vapnik, 1982). Restricting to the case of estimating a one-dimensional density $f \in L_2([a,b])$ over some (nonempty) interval $[a,b] \subset \mathbb{R}$ and recognizing that the cumulative distribution function F for f satisfies the functional equation

$$F(x) = (Lf)(x) \triangleq \int_a^b H(x-t)f(t)\,dt, \quad x \in [a,b] \tag{1.27}$$

(where H is the Heaviside function $H(x) = 1$ for $x \geq 0$, $H(x) = 0$ otherwise) the KDE $\tilde{f}_n$ can be shown to satisfy

$$\tilde{f}_n \triangleq \arg \min_{f \in L_2([a,b])} \left(\left\| Lf - \tilde{F}_n \right\|_2^2 + \lambda \Omega(f) \right) \tag{1.28}$$

where $\tilde{F}_n(\bullet) \triangleq \sum_{i=1}^n H\left(\bullet - x_i\right)$ is the *empirical distribution function* based on a random sample $t_n \triangleq \{x_i\}_{i=1}^n$, $\| \bullet \|_2$ denotes the usual $L_2([a,b])$ norm and $\Omega : L_2([a,b]) \mapsto \mathbb{R}^+$ is a suitable regularizing functional. For example, $\Omega(f) = \|f\|_2^2$ yields a KDE with kernel function $K(x) \triangleq \exp\left(-|x|/\sqrt{\lambda}\right) / \left(2\sqrt{\lambda}\right)$. While this theoretical link between the KDE and the regularization theory used in deriving the SIRBFN is interesting, it does not in itself suggest any significant new practical approaches to density estimation over the KDE. For this purpose, Vapnik (1982) develops an estimate of f based on the minimization of $\tilde{F}_n$ in place of F via the SRM method, as discussed previously.

The step from density estimation to regression estimation is a natural one. Suppose that the joint density of $(\boldsymbol{X}, Y)$ can be estimated with a KDE of the form

$$g_n(\boldsymbol{x}, y) \triangleq \frac{1}{n} \sum_{i=1}^n \frac{1}{h_n^d} K\left(\frac{\boldsymbol{x} - \boldsymbol{X}_i}{h_n}\right) \cdot \frac{1}{h_n} K_1\left(\frac{y - Y_i}{h_n}\right) \tag{1.29}$$

where, for simplicity, we assume that $K(\boldsymbol{x}) \triangleq \prod_{j=1}^{d} K_1(x_j)$, $\boldsymbol{x} = [x_j]_{j=1}^{d}$, is the product kernel formed from a valid one-dimensional kernel K_1. Then, using the approximate densities $\tilde{g}_n$ and $\tilde{f}_n$ in place of the true ones, we obtain the induced or *plug-in* regression estimate known as the *Nadaraya-Watson regression estimate (NWRE)* (Nadaraya, 1964, 1965; Watson, 1964)

$$\begin{aligned}
\mathbb{E}[Y \mid \boldsymbol{X} = \boldsymbol{x}] &\approx \int y \frac{\tilde{g}_n(\boldsymbol{x}, y)}{\tilde{f}_n(\boldsymbol{x})} dy \\
&= \left(\tilde{f}_n(\boldsymbol{x})\right)^{-1} \frac{1}{n h_n^d h_n} \sum_{i=1}^{n} \left\{ K\left(\frac{\boldsymbol{x} - \boldsymbol{X}_i}{h_n}\right) \int (h_n u + Y_i) K_1(u) h_n du \right\} \\
&= \frac{\sum_{i=1}^{n} Y_i K\left(\frac{\boldsymbol{x} - \boldsymbol{X}_i}{h_n}\right)}{\sum_{i=1}^{n} K\left(\frac{\boldsymbol{x} - \boldsymbol{X}_i}{h_n}\right)}
\end{aligned} \tag{1.30}$$

where we have used the facts

$$\int u K_1(u)\, du = 0. \qquad \int K_1(u)\, du = 1 \tag{1.31}$$

It is not too hard to conjecture that the consistency of the density estimates should carry over to the NWRE. Indeed, Devroye (1981) showed that along with the basic KDE consistency conditions in (1.26), all that is needed for the $L_2(P_{T_n})$-consistency of $\tilde{f}_n$ with respect to f to hold a.e.–$P_{\boldsymbol{X}}$ is

$$\exists r, c_1, c_2 \in \mathbb{R}^+ : c_1 I_{\{\|\boldsymbol{x}\| \leq r\}} \leq K(\boldsymbol{x}) \leq c_2 I_{\{\|\boldsymbol{x}\| \leq r\}} \tag{1.32}$$

i.e., the kernel K is sufficiently smooth as to be bounded by two cylinders about the origin in $\mathbb{R}^d$.

For both the KDE and NWRE, so long as the kernel satisfies the necessary smoothness and boundedness conditions previously described, the exact shape of the kernel matters less to the resultant predictor performance than the correct choice of bandwidth h_n (Györfi et al., 1989; Bosq, 1996). Here, as with the ERM theory for ANNs, KRE theory offers several methods of selecting the bandwidth parameter according data-based proxies for the desired performance or generalization measure which is typically quadratic. The difference, however, is that the procedures (as originally formulated) justify their choice of proxy by the criterion of *asymptotic optimality (a.o.)*. In what follows, we shall denote the performance measure by V and its proxy by $\tilde{V}$. Given T_n, let $\tilde{h}_n^*$ and $\tilde{f}_n^*$ (h_n^* and f_n^*) be the bandwidth and corresponding function estimate which minimize the proxy $\tilde{V}\left(\tilde{f}_n, T_n\right)$ (the

actual generalization measure $V(f_n, f)$) over all possible estimates $\tilde{f}_n$ (f_n) derived from T_n. Then V-a.o. means that

$$1 \leq \frac{V\left(\tilde{f}_n^*, f\right)}{V\left(f_n^*, f\right)} \overset{n\to\infty}{\to} 1 \tag{1.33}$$

where the convergence is at least i.p.-P_{T_n}, i.e., we have weak V-consistency. Although we shall have more to say regarding such a.o. data-based parameter selection procedures, it suffices here to note that they are typically computationally intensive. For example, for both the NWRE and KDE cases, one is often interested in the (squared-error) *loss* $\tilde{L}_2$ of an estimate $\tilde{f}_n$ with respect to the true function f given t_n, where

$$\tilde{L}_2\left(\tilde{f}_n, f, t_n\right) \triangleq \frac{1}{n}\sum_{i=1}^{n}\left(f(\boldsymbol{x}_i) - \tilde{f}_n(\boldsymbol{x}_i)\right)^2 \tag{1.34}$$

and, in particular, the *expected loss* or *risk* $\tilde{R}_2$, i.e.,

$$\tilde{R}_2\left(\tilde{f}_n, f\right) \triangleq \mathbf{E}_{T_n}\left[\tilde{L}_2\left(\tilde{f}_n, f, T_n\right)\right] \tag{1.35}$$

To avoid confusion with the previously defined risk R_2 (which is independent of n), we shall also denote $\tilde{R}_2$ by $\tilde{R}_{2,n}$ when we wish to emphasize its dependence on n and call it the *m.s. fitting error (m.s.f.e.)*. One of the first proxies studied for the m.s.f.e. is the *ordinary* or *delete one cross-validation (OCV* or simply *CV)* function, computed in the case of a single bandwidth parameter h_n as

$$CV(h_n, t_n) \triangleq \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \tilde{f}_{-i,n}(\boldsymbol{x}_i, h_n)\right)^2 \tag{1.36}$$

where $\tilde{f}_{-i,n}(\bullet, h_n)$ is the function estimate specified by h_n and T_n with its i-th training pair deleted. The CV proxy states that a reasonable estimate for h_n^*, the true m.s.f.e.-minimizing bandwidth parameter, should also yield a series of networks each of which is able to predict the one datum that was removed from T_n during its training. This proxy is known to have a variety of desirable theoretical properties, including a.o. in the case of noisy T_n with heteroskedastic errors (Andrews, 1991) and unbiased convergence to the global or unconditional m.s.e. (Plutowski et al., 1994). It is also clear, however, that the minimization of $CV(h_n, T_n)$ with respect to h_n generally involves the computation of n networks and their respective losses, which can be prohibitively complex. For linearly-weighted networks such as the RBFN, however, the situation is somewhat better: it can be shown (e.g., see

Theorem 4.2.1 in (Wahba, 1990)) that the CV proxy can be equivalently stated as

$$\begin{aligned} CV(h_n, t_n) &= \frac{1}{n} \left\| \mathrm{diag}^{-1} \left[I - A(h_n, t_n) \right] \left(I - A(h_n, t_n) \right) y \right\|^2 \\ &= \frac{1}{n} \sum_{i=1}^{n} \left(\frac{y_i - \tilde{f}_n(x_i)}{1 - a_{ii}(h_n, t_n)} \right)^2 \end{aligned} \tag{1.37}$$

where $\tilde{f}_n$ is the estimate corresponding to h_n and a_{ii} is the i-th diagonal element of the *influence* or *hat matrix* $A(h_n, t_n)$ which relates the estimated output vector $\tilde{y}$ to the training output vector y via $\tilde{y} \triangleq A(h_n, t_n) y$. For example, in the case of the NWRE with bandwidth h_n, it is easily seen that $A(h_n, t_n) = \mathrm{diag}^{-1} \left[k(x_i)^\top 1_n \right]_{i=1}^n K$, where $k(\bullet) \triangleq [K((x - X_i)/h_n)]_{i=1}^n$, 1_n is a vector of n-ones, and $K \triangleq \left[k^\top(x_i) \right]_{i=1}^n$ is the $n \times n$ NWRE interpolation matrix.

It is apparent in (1.37) that the CV criterion is not invariant to rotations of the coordinate axes for noisy T_n, i.e., applying an orthogonal transformation to the problem (1.17). The desire for such rotational invariance motivated the introduction of the so-called *generalized cross-validation (GCV)* proxy (Craven and Wahba, 1979; Wahba, 1990). In the following, let h_n be a generic parameter to be estimated from a realized example set t_n. Then the GCV proxy is defined as

$$\begin{aligned} GCV(h_n, t_n) &= \tilde{J}_2 \left(\tilde{f}_n, t_n \right) / \left(1 - n^{-1} \mathrm{tr}\, A(h_n, t_n) \right)^2 \\ &= \frac{1}{n} \left\| \left(I - A(h_n, t_n) \right) y \right\|^2 / \left(\frac{1}{n} \mathrm{tr} \left(I - A(h_n, t_n) \right) \right)^2 \end{aligned}$$

For the SIRBFN with regularization parameter λ_n, $\tilde{y} = Gw$ so $A(\lambda_n, t_n) = G(G + \lambda_n I)^{-1}$, hence GCV can be seen as an "averaged" version of CV in which the a_{ii} have been replaced by their average value $\sum_{i=1}^n a_{ii}/n = \mathrm{tr}\, A(h_n, t_n)/n$. Like the CV proxy, the GCV proxy is known to have a several favourable theoretical properties such as a.o., which we shall discuss at greater length in Section 2.3.1.

As mentioned earlier, for stochastic T_n and risk measured by the usual (global) m.s.e., the underlying function f is precisely the conditional mean of the output r.v. Y given the input r.v. X. Comparing the NWRE with a corresponding SIRBFN under these conditions, we find that the NWRE appears to be a SIRBFN with norm weighting matrix I/h_n for all basis functions, as in the KDE case, but with the weights w_i now dependent

upon both T_n (through the output training values Y_i) and the evaluation point $\boldsymbol{x}$, viz.,

$$w_i = w_i(\boldsymbol{x}; T_n) \triangleq Y_i / \sum_{i=1}^{n} K\left(\frac{\boldsymbol{x} - \boldsymbol{X}_i}{h_n}\right), \qquad \tilde{f}_n(\boldsymbol{x}) = \sum_{i=1}^{n} w_i(\boldsymbol{x}; T_n)\, K\left(\frac{\boldsymbol{x} - \boldsymbol{X}_i}{h_n}\right) \tag{1.38}$$

This form for the weights suggests that other such general weighting schemes may also be appropriate. By expressing the NWRE in a slightly different form as an *weighted output* mean

$$\tilde{f}_n(\bullet) \triangleq \sum_{i=1}^{n} a_{n,i}(\bullet; T_n)\, Y_i \tag{1.39}$$

Stone (1977) gave a set of sufficient conditions on the output weight vector function $\boldsymbol{a}_n \triangleq [a_{n,i}]_{i=1}^n$ for the $L_2(P_{\boldsymbol{X}Y} \times P_{T_n})$-consistency of $\tilde{f}_n$ with respect f, of which the following two are necessary:

$$\sum_{i=1}^{n} a_{n,i}(\boldsymbol{X}) \overset{n\to\infty}{\to} 1 \text{ i.p.}, \qquad \max_{i=1,2,\ldots,n} |a_{n,i}(\boldsymbol{X})| \overset{n\to\infty}{\to} 0 \text{ i.p.} \tag{1.40}$$

These conditions become both necessary and sufficient when $\{\boldsymbol{a}_n\}$ is a sequence of *probability weights*, i.e., $\sum_{i=1}^n a_{n,i}(\boldsymbol{x}) = 1$ and $a_{n,i}(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x} \in \mathbb{R}^d$. Unfortunately, the SIRBFN *effective* output weights

$$\boldsymbol{a}_n(\bullet) = (\boldsymbol{G} + \lambda \boldsymbol{I})^{-1}\, \boldsymbol{g}(\bullet), \qquad \boldsymbol{g}(\bullet) \triangleq [G_i(\bullet)]_{i=1}^n \tag{1.41}$$

do not satisfy all the sufficient conditions. At the same time, verifying the two necessary conditions is difficult, since the output weight sequence depends on the exact choices of the corresponding sequences for λ_n, the regularization parameter, and $\boldsymbol{U}_n$, the norm weighting matrix. Another route to consistency for the SIRBFN which indicates how these sequences should be chosen is highly desirable.

Thus far, the results for KRE have been stated for independent samples $(\boldsymbol{X}_i, Y_i) \in T_n$ and evaluation point $(\boldsymbol{X}, Y)$. For the NWRE, however, many of the same consistency results have been shown to hold with minor changes when the samples in T_N are drawn from a process whose dependence structure is described by *mixing conditions* (Douhkan, 1994; Bradley, 1986). In this respect, the application of KRE to correlated data as would occur in time series analysis rests upon a sounder foundation than the current ERM theory for ANN. Exploiting this body of theory in the design of RBFN for similar applications is therefore a prime goal in this thesis.

1.2.4 Limitations of the Kernel Regression Approach

Appealing as the solid theoretical basis for the KRE and (in particular) the NWRE design procedures may be, there are some limitations to this approach. As a practical example, the KRE/NWRE framework does not account for the effects of selecting $n < N$ basis functions centres from the N available data in T_N, a situation handled by the *penalized least-squares (PLS)* theory (Golitschek and Schumaker, 1990) supporting regularized RBFNs. More significant, however, is the fact that the SIRBFN design procedure does not yield a set of output weights which fits clearly into the KRE/NWRE framework. One obvious remedy to this difficulty is to use *normalized RBFN* (Xu et al., 1994), i.e.,

$$\tilde{f}_n(\bullet) = \frac{\sum_{i=1}^{n} w_i G_i(\bullet)}{\sum_{i=1}^{n} G_i(\bullet)} \tag{1.42}$$

where the norm weighting matrix $\boldsymbol{U}_n = \boldsymbol{I}/h_n$, with $\{h_n\}$ satisfying (1.26) as for the KDE and NWRE. In the analysis of Xu et al. (1994), the centres are chosen as per the *random centres* method of Broomhead and Lowe (1988) previous described and, correspondingly, the weights $[w_i]_{i=1}^n$ solve the least-squares pseudo-inverse equation given in (1.10) with the modified interpolation matrix $\boldsymbol{G}'$ in place of $\boldsymbol{G}$, where

$$[\boldsymbol{G}']_{ij} = \frac{[\boldsymbol{G}]_{ij}}{\sum_{k=1}^{n} [\boldsymbol{G}]_{kj}} \tag{1.43}$$

From the clear similarity to the NWRE, it is expected that such normalized RBFNs should be consistent in the same modes that the NWRE is. While the rigourous proofs of Xu et al. (1994) supporting this intuitive notion are useful, this approach fails to include the important case of regularized SIRBFNs that we shall be considering. One advantage of explicit regularization for SIRBFNs is that computationally efficient selection procedures for λ exist which are a.o. in the *actual* loss for given a training input sequence rather than just the *average* loss or risk (see the references for the scenarios listed in Section 2.3.1); in this sense, the *pointwise* behaviour of the regularized SIRBFN estimate can be provably correct.

1.3 Dissertation Overview

Given that the current ANN and KRE/NWRE approaches to RBFN design leave something to be desired, the object of this dissertation is to demonstrate that a rigourous basis for regularized SIRBFN can be constructed from the related areas of *regularized* or

penalized least-squares (fitting) (*RLS(F)* or *PLS(F)*) and *spline smoothing*. Building upon this basis, it is possible to:

A1. show that for stochastic T_n and performance measured by the m.s.f.e. $\tilde{R}_2$, the ANN ERM method is nonoptimal because it corresponds to the regularization parameter sequence with $\lambda_n = 0$ for all n, whereas the optimal regularization parameter λ_n^* for each n is nonzero.

A2. give an explicit method for the computation of a.o. estimates $\tilde{\lambda}_n$ of the m.s.f.e.-minimizing regularization parameter $\tilde{\lambda}_n^*$.

A3. show that the m.s.f.e. $\tilde{R}_2$ converges a.s. to the (global) m.s.e. or risk R_2.

A4. prove the m.s.-consistency of the regularized SIRBFN with a.o. regularization parameter sequence under the same conditions as is known for the NWRE.

A5. show that an SIRBFN with positive definite kernel and designed by the ERM method without regularization, i.e. with $\lambda_n = 0$ for all n, cannot be m.s.-consistent.

Items A1 and A2 are discussed in the PLS literature, e.g., see (Golitschek and Schumaker, 1990; Wahba, 1990), while item A5 is a direct consequence of the PLS theory as shown later in Section 2.3.3. Result A4, however, requires more work. First, we present a *constructive* proof of the a.s. uniform and m.s. approximation of the NWRE over compact sets with arbitrary rates of convergence by a class of suitably regularized SIRBFN for the case of stochastic T_n, where the $(\boldsymbol{X}_i, Y_i)$ are drawn from either an i.i.d. or mixing process with a stationary marginal density for $\{\boldsymbol{X}_i\}$. This *asymptotic equivalence* between the NWRE and the SIRBFN justifies the application of the large body of theory surrounding NWRE design and motivates the derivation of result A4 by way of comparison. This key result can be established after result A3 is established to link asymptotically the global risk R_2 to its proxy the m.s.f.e. $\tilde{R}_2$.

Once these theoretical tools are in place, they find natural applications in the areas of probability estimation, classification, and time series prediction. For the first two related areas, we can:

B1. prove the m.s.-consistency of regularized SIRBFN with a.o.-selected regularization parameter sequence for posterior probability estimation.

B2. show the Bayes risk consistency of the approximate Bayes decision rules based on such regularized SIRBFN posterior probability estimates.

B3. prove that m.s.-consistency of posterior probability estimates implies weak convergence of the corresponding approximate Bayes decision rules and their respective classifier error rates (consistent or not). Thus positivity and normalization of m.s.-consistent posterior probability estimates are not required for BRC.

Not surprisingly, the flexibility of RBFNs can be useful for modelling nonlinear time series. As a first step in this direction, we consider the generalization from the usual linear autoregressive (AR) time series to the class of Markovian *nonlinear AR (NLAR)* time series generated by an i.i.d. noise process. For the prediction of this class, we can:

C1. show that, with suitable modifications, result A3 carries over to the NLAR case. This result justifies the use of data-based parameter estimation procedures such as GCV for the a.o. selection of λ in such time series.

C2. derive recursive updating algorithms for regularized SIRBFN similar to the linear RLS updates for *one-step-ahead (OSA)* predictor when new training data are continually available. The following two basic types of network updating are considered:

(a) a new weight/basis function associated with the most recently available training data is added per update (*augmented* or *infinite memory* case)

(b) the weight/basis function associated with the oldest available training data is discarded for a new weight/basis function associated with the most recently available training data (*fixed-size* or *finite memory* case)

C3. characterize performance gain in using nonlinear autoregressive model over usual linear autoregressive model via experiment on OSA prediction of speech signals.

C4. show experimentally improvement possible by linearly combining several predictor outputs using exponentially-weighted RLS algorithm.

Finally, we present some current research on certain recently developing areas, viz.

D1. we discuss the application of regularized SIRBFN in the regression-based approach to nonlinear filtering or state estimation. Simulation results for a nonlinear discrete-time

system governed by difference equations show that the regularized SIRBFN has comparable performance to a recurrent MLP-based regression method while retaining the theoretical support offered by the results of Chapter 4. For a discretized nonlinear stochastic differential system, results suggest that the regularized SIRBFN-based regression method can be a viable alternative to a path-wise convergent stochastic partial differential equation method when its requisite strong prior knowledge is not available.

D2. we show that the regularized SIRBFN can be an effective tool for the dynamic reconstruction of chaotic systems from noisy observations. Specifically, we apply the delay co-ordinate embedding method using the regularized SIRBFN to the reconstruction of the Lorenz system from observations of a single system co-ordinate (variable) at infinite, 30dB, and 20dB SNRs. The resultant networks, when operated in the recursive, forward-iterated predictive mode, generate sequences whose estimated long and short-term dynamical invariants agree closely with those of the original, noise-free system. We also indicate similar results for the reconstruction of a real-life chaotic sea clutter process.

We close the thesis with a few concluding remarks offering some perspective on the contributions and limitations of the thesis, the latter naturally leading to suggestions for further investigations.

Chapter 2

Basic Tools

2.1 Asymptotic Equivalence of NWRE to Regularized SIRBFN via Constructive Approximation

At this point it would be useful to discuss the basic results upon which the forthcoming applications are based. We begin by showing that any NWRE $\widetilde{f}'_n$ with radial kernel K' designed from a given training set t_n can be approximated with vanishing error at a given point $z \in \mathbb{R}^d$ by a suitably designed regularized SIRBFN $\widetilde{f}_n$. In the following, $g_n(\bullet) \triangleq [G_i(\bullet)]_{i=1}^n$ is the basis function vector and primed quantities refer to constructions involving the NWRE with the non-normalized kernel K' as defined in the preamble to the lemma.

Lemma 1. *Let $\widetilde{f}'_n$ be a NWRE using a radial kernel K' with $C \triangleq \sup_{z \in \mathbb{R}^d} K'(z)$, and bandwidth parameter h_n designed from a given training set t_n. Then for any $n > 1$, $\alpha > \log\left(C/\left(h_n^d \widetilde{p}_n(z)\right)\right) / \log n$, and $z \in \mathbb{R}^d$ such that the denominator $nh_n^d \widetilde{p}_n(z) \triangleq 1_n^\top g'_n(z)$ of $\widetilde{f}'_n$ is not zero, a regularized SIRBFN $\widetilde{f}_n$ with kernel $K \triangleq K'/C$ may be constructed such that*

$$\left|\widetilde{f}_n(z) - \widetilde{f}'_n(z)\right| \leq \frac{C^2 M}{n^{\alpha/2} h_n^d \widetilde{p}_n(z) \left[n^{\alpha/2} h_n^d \widetilde{p}_n(z) - C n^{-\alpha/2}\right]} \tag{2.1}$$

where $\|y\|_\infty \leq M$ and where 1_n is a constant vector of n ones.

As an aside, the identification of NWRE denominator $1_n^\top g'_n(z)$ with the *Parzen window (density) estimate (PWE)* (Parzen, 1962) $\widetilde{p}_n$ of the marginal input density p is merely suggestive of a decomposition that is useful in the subsequent proofs.

Proof. For the regularized SIRBFN, set $U_n \triangleq I/h_n$ and $\lambda_n = \lambda_n(z) = n^\alpha g_n^\top(z)1_n = n^{\alpha+1}h_n^d\tilde{p}_n(z)/C$, where $\alpha \geq 0$ is an exponent to be determined later. Let $y_n' \triangleq n^\alpha y_n$ so that

$$\tilde{f}_n'(z) = g_n^\top(z)\left(\lambda_n(z)I\right)^{-1} y_n' \tag{2.2}$$

Comparing the NWRE output to that of the regularized SIRBFN designed from t_n, we find that the difference can be bounded (by Cauchy-Schwarz) as

$$\begin{aligned}
\left|\tilde{f}_n(z) - \tilde{f}_n'(z)\right| &= \left|\left\langle g_n(z),\ (G_n + \lambda_n(z)I)^{-1} y_n' - (\lambda_n(z)I)^{-1} y_n' \right\rangle\right| \\
&\leq \|g_n(z)\| \left\|(G_n + \lambda_n(z)I)^{-1} - (\lambda_n(z)I)^{-1}\right\| \|y_n'\| \\
&\leq \|g_n(z)\| \frac{\|G_n\|\,\|I\|}{\lambda_n(z)\left(\lambda_n(z) - \|G_n\|\right)} \|y_n'\|, \quad \lambda_n(z) > \|G_n\| \qquad (2.3)
\end{aligned}$$

Using the Euclidean norm as an upper bound for all quantities except for G_n, which we bound in Fröbenius norm as $\|G_n\| \leq n$, we obtain

$$\begin{aligned}
\left|\tilde{f}_n(z) - \tilde{f}_n'(z)\right| &\leq \sqrt{n}\frac{n}{\lambda_n(z)\left(\lambda_n(z) - n\right)} n^\alpha \sqrt{n} M \\
&\leq \frac{n^2 n^\alpha M}{\lambda_n(z)\left(\lambda_n(z) - n\right)}
\end{aligned}$$

For our choice of $\lambda_n(z)$, this latter result can be rewritten as

$$\begin{aligned}
\left|\tilde{f}_n(z) - \tilde{f}_n'(z)\right| &\leq \sqrt{n}\frac{n^{\alpha+2}C^2 M}{n^{\alpha+1}h_n^d\tilde{p}_n(z)\left(n^{\alpha+1}h_n^d\tilde{p}_n(z) - nC\right)} \\
&\leq \frac{C^2 M}{n^{\alpha/2}h_n^d\tilde{p}_n(z)\left[n^{\alpha/2}h_n^d\tilde{p}_n(z) - Cn^{-\alpha/2}\right]} \qquad (2.4)
\end{aligned}$$

The condition on $\lambda_n(z)$ in (2.3) can be satisfied by choosing

$$\lambda_n(z) > n \Rightarrow n^{\alpha+1}h_n^d\tilde{p}_n(z)/C > n \Rightarrow \alpha > \log\left(C/\left(h_n^d\tilde{p}_n(z)\right)\right)/\log n \tag{2.5}$$

□

The proof of the lemma shows that by selecting the SIRBFN regularization parameter λ_n to be the denominator of NWRE and scaling it along with training outputs in t_n at rate n^α, an arbitrarily fast rate of approximation at a *fixed point* $z \in \mathbb{R}^d$ is possible. We would, of course, like to extend this pointwise approximation result to corresponding a.s. uniform and m.s. approximation results in the case of stochastic T_n. As a direct route to such results, we shall consider a special class F_z of regularized SIRBFNs in which λ_n (and hence w_n) is

permitted to vary with its input $z \in \mathbb{R}^d$ as in the proof of the lemma. This class is therefore a slight generalization of the usual class of regularized SIRBFNs in which λ_n (and hence $\tilde{w}_n$) is set once via t_n for all inputs z. As will be explained further on, the generalization does not affect the overall tenor of the results on the relative suboptimality of NWREs with respect to the risk or m.s.f.e. over T_n.

In view of future applications to time-series, we shall indicate (discrete) time dependence by parenthesized indices instead of subscripts and denote the input-output processes by $\{Z(i)\}$ and $\{Y(i)\}$, respectively. The upper bound of Lemma 1 can be generalized to hold a.s. uniformly and in m.s. over a compact set $D \subset \mathbb{R}^d$ instead of a single point by assuming conditions which ensure that $\tilde{p}_n$ is lower bounded in the respective modes. For convenience, we shall assume the sufficient condition that the input process have a common marginal density p for which $\tilde{p}_n$ is a convergent KDE (in the same mode). When this situation is not applicable, e.g., when the input process is nonstationary, a more general condition which allows the required lower bounding is condition (A.1) from (Györfi et al., 1989)

$$\begin{aligned} &\exists \Gamma < \infty \text{ such that } \forall i \in \mathbb{N} \text{ and } \forall B \in \mathcal{B}\left(\mathbb{R}^d\right), \quad \Pr\left(Z(i) \in B\right) \leq \Gamma \mu(B) \\ &\exists \gamma, \epsilon > 0 \text{ such that } \forall i \in \mathbb{N} \text{ and } \forall B \in \mathcal{B}\left(D_\epsilon\right), \quad \Pr\left(Z(i) \in B\right) \geq \gamma \mu(B) \end{aligned} \tag{A.1}$$

where $\mathcal{B}\left(\mathbb{R}^d\right)$ (resp. $\mathcal{B}\left(D_\epsilon\right)$) is the σ-algebra of the Borel sets on $\mathbb{R}^d$ (resp. on D_ϵ), μ is the Lebesgue measure on $\mathbb{R}^d$ and $D_\epsilon \ni D$ is the set of all ϵ-neighbourhood compact sets covering D (recall that $A \in D_\epsilon$ if A is compact and for all $x \in A$, there is a $y \in D$ such that $\|x - y\| \leq \epsilon$). As mentioned in Remark 3.3.1 of (Györfi et al., 1989), these conditions are sufficient to ensure that $\tilde{p}_n$ does not vanish on D for each n (which, since $\{\tilde{p}_n\}$ is a sequence of positive, continuous functions over a compact set D, also implies that p must be lower bounded away from zero over D).

Theorem 1. *Assume that $\{Z(i)\}$ has a stationary marginal measure P and density p with respect to Lebesgue measure. Let D be a compact subset of $\mathbb{R}^d$ with $p(z) > 0$ for all $z \in D$. Then*

1. *If $|Y(i)| < M$ almost surely (a.s.) for all i and if K', $\{h_n\}$, and p are such that*

$$\sup_{z \in D} |\tilde{p}_n(z) - p(z)| \overset{n \to \infty}{\to} 0 \tag{2.6}$$

then $\exists N = N(p, D, K', \{h_n\})$ *such that for any* $n > N$ *and* $\alpha > \max\left(2, \log\left(2C/\left(h_n^d m\right)\right) / \log n\right)$, *a regularized SIRBFN* $\tilde{f}_{n,\infty} \in F_z$ *may be constructed such that*

$$\sup_{z \in D} \left| \tilde{f}_{n,\infty}(z) - \tilde{f}_n'(z) \right| = \mathcal{O}\left(C^2 M n^{-\alpha} h_n^{-2d} m^{-2}\right) \ a.s.\text{-}P_{T_n} \tag{2.7}$$

where $m = m(D) \triangleq \inf_{z \in D} p(z)$.

2. *If* $\mathbb{E}\left[Y^2(i)\right] < M^2$ *for all* i *and if* K', $\{h_n\}$, *and* p *are such that*

$$\sup_{z \in D} \mathbb{E}_{T_n}\left[|\tilde{p}_n(z) - p(z)|^2\right] \overset{n \to \infty}{\to} 0 \tag{2.8}$$

and there exist positive constants R_1, R_2, R_3 *and* ν *such that*

$$\lim_{n \to \infty} \sup_{z \in D} \left(\left| \tilde{f}_{n,\infty}(z) \right| + \left| \tilde{f}_n'(z) \right| \right) < R_1 \ a.s.\text{-}P_{T_n} \tag{2.9}$$

$$\lim_{n \to \infty} \sup_{\substack{i,j=1,\ldots,n \\ (x,y) \in \mathbb{R}^d \times \mathbb{R}^d}} \frac{p_{ij}(x, y)}{p(x)p(y)} < R_2 \tag{2.10}$$

$$\lim_{n \to \infty} \inf_{z \in D} n^{\nu} n h_n^d \tilde{p}_n(z) > R_3 > 0 \ a.s.\text{-}P_{T_n} \tag{2.11}$$

where $p_{ij}(\bullet, \bullet) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^+$ *is the joint density for* $Z(i)$ *and* $Z(j)$. *then* $\exists N = N(p, D, K', \{h_n\})$ *such that for* $n > N$ *and* $\alpha > \max\left(1, \nu, \log\left(2C/\left(h_n^d m\right)\right) / \log n\right)$, *a regularized SIRBFN* $\tilde{f}_{n,\infty} \in F_z$ *may be constructed such that*

$$\sup_{z \in D} \mathbb{E}_{T_n}\left[\left| \tilde{f}_{n,\infty}(z) - \tilde{f}_n'(z) \right|^2\right] = \mathcal{O}\left(C^2 M \sqrt{L} \|p\|_2 \|K\|_2^2 n^{-\alpha} h_n^{-d} m^{-2}\right) \tag{2.12}$$

where m *is as before and* $L = L(D) \triangleq \sup_{z \in D} p(z)$.

Proof. We treat each case separately:

1. It is easy to show that (2.6) implies that by choosing N to satisfy

$$n > N \Rightarrow \sup_{z \in D} |\tilde{p}_n(z) - p(z)| < \frac{m}{2} \tag{2.13}$$

we have $n > N \Rightarrow \tilde{p}_n(z) \geq m/2$ for all $z \in D$. Hence for $n > N$, we may replace $\tilde{p}_n(z)$ with $m/2$ in the denominator of the upper bound, and the term $Cn^{-\alpha/2}$ can be dominated by selecting a sufficiently large constant to multiply the numerator of the order bound, i.e., $\exists L > 0$ such that for $n > N$,

$$\frac{L \cdot C^2 M}{n^{\alpha} h_n^{2d} m^2} > \frac{C^2 M}{n^{\alpha/2} h_n^d m/2 \left[n^{\alpha/2} h_n^d m/2 - C n^{-\alpha/2}\right]} \tag{2.14}$$

From the basic KDE consistency condition $nh_n^d \overset{n\to\infty}{\to} \infty$, requiring $\alpha > \max\left(2, \log\left(2C / \left(h_n^d m\right)\right) / \log n\right)$ ensures that the approximation error vanishes with increasing n.

2. While the convergence rates for this case must be at least as rapid as for the a.s. uniform case (by squaring and taking expectations on both sides of (2.3) before computing the sup on the left-hand side), we can obtain slightly better convergence rates with tighter m.s. estimates of the terms in (2.3). We begin by noting that it is sufficient to demonstrate the corresponding result in absolute value, since

$$\begin{aligned}
&\mathbb{E}_{T_n}\left[\left|\tilde{f}_{n,\infty}(z) - \tilde{f}_n'(z)\right|^2\right] \\
&= \mathbb{E}_{T_n}\left[\left(\tilde{f}_{n,\infty}(z) - \tilde{f}_n'(z)\right)\left(\tilde{f}_{n,\infty}(z) + \tilde{f}_n'(z)\right) + 2\tilde{f}_n'(z)\left(\tilde{f}_n'(z) - \tilde{f}_{n,\infty}(z)\right)\right] \\
&\leq \sup_{z\in D}\left(\left|\tilde{f}_{n,\infty}(z) + \tilde{f}_n'(z)\right| + 2\left|\tilde{f}_n'(z)\right|\right) \cdot \mathbb{E}_{T_n}\left[\left|\tilde{f}_{n,\infty}(z) - \tilde{f}_n'(z)\right|\right] \qquad (2.15)
\end{aligned}$$

where the supremum is $\mathcal{O}(R_1)$ for n sufficiently large by assumption (2.9). Returning to the expectation term, taking expectations with respect to P_{T_n} on both sides of (2.3) and applying Cauchy-Schwarz gives

$$\begin{aligned}
\mathbb{E}_{T_n}\left[\left|\tilde{f}_{n,\infty}(z) - \tilde{f}_n'(z)\right|\right] \leq\ & \mathbb{E}_{T_n}^{1/2}\left[\|g_n(z)\|^2\right] \mathbb{E}_{T_n}^{1/2}\left[\|G_n\|^2\right] \\
& \cdot \mathbb{E}_{T_n}^{1/2}\left[\lambda_n^{-2}(z)\left(\lambda_n(z) - \|G_n\|\right)^{-2}\right] \mathbb{E}_{T_n}^{1/2}\left[\|y_n'\|^2\right] \qquad (2.16)
\end{aligned}$$

The first term squared $\mathbb{E}_{T_n}\left[\|g_n(z)\|^2\right]$ can be asymptotically bounded in Euclidean norm as

$$\begin{aligned}
\mathbb{E}_{T_n}\left[\|g_n(z)\|^2\right] &= \sum_{i=1}^{n} \mathbb{E}_{T_n}\left[K^2\left(\frac{z - Z(i)}{h_n}\right)\right] \\
&= \mathcal{O}\left(nh_n^d p(z)\|K\|_2^2\right) \text{ a.e.} \qquad (2.17)
\end{aligned}$$

where $\|\bullet\|_2$ is the standard L_2 norm with respect to Lebesgue measure and we have used the fact that (see (2.10) in (Bosq, 1996)),

$$\left|\int_{\mathbb{R}^d} K^2\left(\frac{\boldsymbol{x} - \boldsymbol{u}}{h_n}\right) p(\boldsymbol{u})\, d\boldsymbol{u} - h_n^d p(\boldsymbol{x})\|K\|_2^2\right| \overset{n\to\infty}{\to} 0 \text{ a.e.} \qquad (2.18)$$

where $\|\bullet\|_2$ is the usual $L_2\left(\mathbb{R}^d\right)$ norm with respect to Lebesgue measure. Similarly, we bound the second term squared, $\mathbb{E}_{T_n}\left[\|G_n\|^2\right]$ in Fröbenius norm and apply (2.18)

with Lebesgue dominated convergence to obtain

$$\begin{aligned}\mathbb{E}_{T_n}\left[\|G_n\|^2\right] &\leq \sum_{i,j=1}^{n} \mathbb{E}_{Z(i),Z(j)}\left[K^2\left(\frac{Z(i)-Z(j)}{h_n}\right)\right] \\ &= \mathcal{O}\left(n^2\int_{\mathbb{R}^d}\int_{\mathbb{R}^d} K^2\left(\frac{z(i)-z(j)}{h_n}\right) p(z(i))p(z(j))\,dz(i)dz(j)\right) \\ &= \mathcal{O}\left(n^2 h_n^d \|p\|_2^2 \|K\|_2^2\right) \qquad (2.19)\end{aligned}$$

where we have applied (2.10). For the square of the middle term, we may again apply the majorization $\|G_n\| < n$ and use the same argument as for (2.14) to obtain the estimate

$$\mathbb{E}_{T_n}\left[\lambda_n^{-2}(z)\left(\lambda_n(z)-\|G_n\|\right)^{-2}\right] \leq L\cdot\mathbb{E}_{T_n}\left[\lambda_n^{-4}(z)\right], \qquad \forall z\in D \text{ when } n > N_1 \qquad (2.20)$$

for some $L > 0$ and $N_1 \in \mathbb{N}$. Next, we may substitute p for $\tilde{p}_n$ in the expectation with error bounded by

$$\begin{aligned}\left|\mathbb{E}_{T_n}\left[\lambda_n^{-4}(z)\right] - \lambda^{-4}(z)\right| &\leq \mathbb{E}_{T_n}\left[\left|\lambda_n^{-4}(z)-\lambda^{-4}(z)\right|\right] \\ &\leq \mathbb{E}_{T_n}\left[\left|\frac{\lambda_n^4(z)-\lambda^4(z)}{\lambda_n^4(z)\cdot\lambda^4(z)}\right|\right] \\ &\leq \mathbb{E}_{T_n}\left[\left|\frac{\left(\lambda_n^2(z)+\lambda^2(z)\right)\left(\lambda_n(z)+\lambda z\right)\right)\left(\lambda_n(z)-\lambda z\right)\right)}{\lambda_n^4(z)\cdot\lambda^4(z)}\right|\right] \qquad (2.21)\end{aligned}$$

where $\lambda(z) \triangleq n^{\alpha+1}h_n^d p(z)/C$, whence, by Cauchy-Schwarz,

$$\begin{aligned}\sup_{z\in D}\left|\mathbb{E}_{T_n}\left[\lambda_n^{-4}(z)\right] - \lambda^{-4}(z)\right| \leq \sup_{z\in D}\mathbb{E}_{T_n}^{1/2}&\left[\left|\frac{\left(\tilde{p}_n^2(z)+p^2(z)\right)\left(\tilde{p}_n(z)+p(z)\right)}{\left(n^{\alpha+1}h_n^d\tilde{p}_n(z)/C\right)^4\cdot p^4(z)}\right|^2\right] \\ &\cdot\sqrt{\sup_{z\in D}\mathbb{E}_{T_n}\left[\left|\tilde{p}_n(z)-p(z)\right|^2\right]} \qquad (2.22)\end{aligned}$$

By conditions (2.9) and (2.11), the first sup term is (at least) bounded for α sufficiently large, while the second term vanishes by (2.8). Therefore we have that, for n sufficiently large,

$$\sup_{z\in D}\mathbb{E}_{T_n}\left[\lambda_n^{-2}(z)\left(\lambda_n(z)-\|G_n\|\right)^{-2}\right] = \mathcal{O}\left(\sup_{z\in D}p^{-4}(z)\right) = \mathcal{O}\left(\left(n^{\alpha+1}h_n^d m/C\right)^{-4}\right) \qquad (2.23)$$

The square of the last term $\mathbb{E}_{T_n}\left[\|y'_n\|^2\right]$ is bounded (by assumption) by $nn^{2\alpha}M^2$, so that combining the square roots of the previous terms leaves us with the conclusion that for sufficiently large n,

$$\begin{aligned}
&\sup_{z\in D}\mathbb{E}_{T_n}\left[\left|\widetilde{f}_{n,\infty}(z)-\widetilde{f}'_n(z)\right|^2\right] \\
&= \mathcal{O}\left(\frac{\left(nh_n^dL\|K\|_2^2\right)^{1/2}\cdot\left(n^2h_n^d\|p\|_2^2\|K\|_2^2\right)^{1/2}\cdot\sqrt{n}n^{\alpha}M}{\left(n^{\alpha+1}h_n^dm/C\right)^2}\right) \\
&= \mathcal{O}\left(C^2M\sqrt{L}\|p\|_2\|K\|_2^2n^{-\alpha}h_n^{-d}m^{-2}\right) \qquad (2.24)
\end{aligned}$$

In order for (2.16) to be valid, we require that (2.3) hold uniformly over D, leading to the previous condition of $\alpha > \log\left(2C/\left(h_n^dm\right)\right)/\log n$, where m is as defined in (2.13). As an aside, if (2.8) is weakened to the corresponding mean-input case, then (2.3) need only hold a.s.–P. Finally, the lower limits of 1 and ν imposed on α by the maximum function ensures that the upper bound on the m.s. approximation error in (2.12) decreases as $n\rightarrow\infty$ by the consistency condition $nh_n^d \overset{n\rightarrow\infty}{\rightarrow}\infty$.

□

One may find in the literature numerous sets of conditions under which (2.6) and (2.8) hold. In particular, we refer the reader to Lemma 2.1, Theorem 2.2, and Corollary 2.2 in the case of (2.6), and Theorem 2.1 and Corollary 2.1 in the case of (2.8), all from (Bosq, 1996). Rather than repeat the theorems and proofs verbatim, we merely make a few pertinent observations concerning their application and consequences:

1. the quoted results from (Bosq, 1996) hold for any kernel satisfying (1.24) and (1.25), including the histogram or rectangular kernel $K(\bullet)\triangleq I_{[-1/2,+1/2]^d}(\bullet)$, the Gaussian kernel $K(\bullet)\triangleq(2\pi)^{-d/2}\exp\left(-\|\bullet\|^2/2\right)$, and the *Epanechnikov* kernel $K(\boldsymbol{x})\triangleq\left(\frac{3}{4}\sqrt{5}\right)^d\prod_{i=1}^d\left(1-x_i^2/5\right)I_{[-\sqrt{5},+\sqrt{5}]}(x_i)$, $\boldsymbol{x}\in\mathbb{R}^d$, which is optimal in the sense of minimizing the asymptotic m.s.e. while having compact support. Although conditions (1.24) and (1.25) are commonly assumed in both kernel density and regression estimation to ensure consistency, one should be aware that kernels which do not satisfy these conditions can have other desirable properties. Specifically, in the one-dimensional i.i.d. case, studies exist which show that KDEs based on strictly positive kernels cannot be globally unbiased for any finite training set and have limited asymptotic rate of bias convergence (Rosenblatt, 1956; Hand, 1982). From these results,

Lowe (1995) argues for the use of *non positive* (definite) kernels for general RBFNs, i.e., those not strictly of the regularization type. The same work also indicates results from polynomial approximation theory which support the use of *unbounded* kernels in the case of uniformly-spaced, noise-free data. That said, the author is not aware of any definitive analysis demonstrating the effect of these negative results on the NWRE or regularized SIRBFN in the regression context with which we are primarily concerned. Indeed, as we later present, the reasonable simulation results obtained using a Gaussian kernel in a range of practical applications suggest that the theoretical penalty imposed by the conditions (1.24) and (1.25) need not be as grave as these studies may imply.

2. both sets of conditions apply to the case of a strictly stationary random process $\{Z(i)\}$ whose dependence structure is described by a *mixing* condition. For (2.8), $\{Z(i)\}$ is assumed to follow a 2-α-*mixing* condition with geometric decay, i.e.,

$$\alpha^{(2)}(k) \triangleq \sup_{i \in \mathbb{Z}} \alpha(\sigma(Z(i)), \sigma(Z(i+k))) = \mathcal{O}\left(k^{\beta}\right), \quad k \geq 1 \tag{2.25}$$

where, for two sub σ-fields $\mathcal{B}$, $\mathcal{C}$ of a common σ-field $\mathcal{A}$ with probability measure P, α is the strong mixing coefficient

$$\alpha = \alpha(\mathcal{B}, \mathcal{C}) \triangleq \sup_{B \in \mathcal{B}, C \in \mathcal{C}} |P(B \cap C) - P(B)P(C)| \tag{2.26}$$

For (2.6), $\{Z(i)\}$ requires (not unexpectedly) the stronger *geometrically strong mixing (GSM)* condition, i.e.,

$$\alpha(k) \triangleq \sup_{j \in \mathbb{Z}} \alpha\left(\sigma\left(\{Z(i)\}_{i=0}^{j}\right), \sigma\left(\{Z(i)\}_{i=j+k}^{\infty}\right)\right) = \mathcal{O}\left(\rho^{k}\right) \text{ for some } 0 \leq \rho < 1 \tag{2.27}$$

Both of these mixing conditions are less restrictive than other types of mixing conditions commonly assumed, e.g., φ and ρ-mixing, and includes (trivially) the i.i.d. case.

3. for the convergence of (2.6), when K' is Lipschitz and $p = p(z_1, \ldots, z_d) \in C_{2,d}(b)$ for some $b > 0$, where

$$C_{2,d}(b) \triangleq \left\{ f \in C_2\left(\mathbb{R}^d\right) : \|f\|_{\infty} < \infty \text{ and } \sup_{i,j=1,\ldots d} \left\| \frac{\partial f}{\partial z_i \partial z_j} \right\|_{\infty} < b \right\} \tag{2.28}$$

$h_n = c_n(\log(n)/n)^{1/(d+4)}$, $c_n \overset{n\to\infty}{\to} c > 0$ assures that convergence for $D = D(n) \triangleq \left\{z \in \mathbb{R}^d : \|z\| \leq n^\gamma\right\}$ $\forall \gamma > 0$. This result implies a lower bound of $\mathcal{O}\left(n^{2d/(d+4)-\alpha}\left(c_n^d(\log n)^{d/(d+4)}\right)^{-2}\right)$ on the corresponding rate of convergence in (2.7). If K' is the Gaussian kernel and $\mathbb{E}[\|Z(0)\|] < \infty$ (where $\{Z(i)\} = \{Z(i)\}_{i=0}^{\infty}$), then the convergence holds for $\mathbb{R}^d$ in place of D. If $c_n \searrow c$, the requirement on α for (2.7) to hold can be made independent of n by taking

$$\alpha > 1 + \max\left(1, \frac{d}{d+4}\left(1 - \frac{\log\log 2}{\log 2}\right) + \log\left(C/\left(c^d m\right)\right)/\log 2\right) \tag{2.29}$$

since

$$\begin{aligned}
&\log\left(2C/\left(c^d\left(\frac{n}{\log n}\right)^{-d/(d+4)} m\right)\right)/\log n \\
&= \log\left(2C/\left(c^d m\right)\right)/\log n + \frac{d}{d+4}(\log n - \log\log n)/\log n \\
&= \log\left(2C/\left(c^d m\right)\right)/\log n + \frac{d}{d+4}\left(1 - \frac{\log\log n}{\log n}\right) \\
&\leq \left(\log 2 + \log\left(C/\left(c^d m\right)\right)\right)/\log 2 + \frac{d}{d+4}\left(1 - \frac{\log\log 2}{\log 2}\right) \\
&\leq 1 + \log\left(C/\left(c^d m\right)\right)/\log 2 + \frac{d}{d+4}\left(1 - \frac{\log\log 2}{\log 2}\right)
\end{aligned}$$

4. for the convergence of (2.8), with some conditions on p and $\{Z(i)\}$ (as described in the preamble to Theorem 2.1 of (Bosq, 1996)), $h_n = c_n n^{-1/(d+4)}$ where $c_n \overset{n\to\infty}{\to} c > 0$ is sufficient. Hence $nh_n^d = c_n^d n^{4/(d+4)}$, which implies that the rate of convergence in (2.12) is at least $\mathcal{O}\left(n^{1-\alpha}\right)$ for $d \geq 1$. By a similar argument to that in the uniform case, if $c_n \searrow c$, the requirement on α for (2.12) to hold can be made independent of n by taking

$$\alpha > 1 + \max\left(\nu, \frac{d}{d+4} + \log\left(C/\left(c^d m\right)\right)/\log 2\right) \tag{2.30}$$

Some comments are now in order regarding the approximation theorems themselves:

1. the requirement that D, the region of approximation, be a compact subset of $\mathbb{R}^d$ such that p is strictly positive over D is common in both KDE and NWRE analysis (to ensure that the denominator is bounded away from zero). Given that approximation in regions of zero probability are of theoretical interest only, this requirement is not a practical impediment in applications. Later on, in Chapter 4, we shall extend the results to an increasing sequence of compact sets by imposing some extra conditions on the tail of the input process probability density.

2. the additional conditions required for the m.s. approximation case are quite mild; in particular, (2.9) is satisfied when the NWRE $\tilde{f}_n'$ and its m.s. approximating SIRBFN $\tilde{f}_{n,\infty}$ converge to continuous functions over D. Condition (2.10) is akin to the *local measure of dependence* $\kappa_{ij} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ defined in Section 2.2 of (Bosq, 1996) as

$$\kappa_{ij}(\boldsymbol{x}, \boldsymbol{y}) \triangleq p_{ij}(\boldsymbol{x}, \boldsymbol{y}) - p(\boldsymbol{x})p(\boldsymbol{y}). \qquad \text{for } i \neq j \tag{2.31}$$

Now one possible sufficient condition in the preamble to Theorem 2.1 of (Bosq, 1996) is that the κ_{ij} (uniformly) be Lipschitz when considered as a function over $\mathbb{R}^{2d}$. If this condition holds, then it can be shown (Lemma 1.3 of (Bosq, 1996)) that $\|\kappa_{ij}\|_\infty < M$, where M is a constant (dependent only upon the input dimension and the Lipschitz constant for κ_{ij}) times the first (and largest) 2-α-mixing coefficient $\alpha^{(2)}(1)$ of the input process $\{Z(i)\}$. Because we assume that p is lower bounded over D by $m > 0$,

$$\sup_{\boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^d} |p_{ij}(\boldsymbol{x}, \boldsymbol{y}) - p(\boldsymbol{x})p(\boldsymbol{y})| < M \Rightarrow \sup_{\boldsymbol{x},\boldsymbol{y}\in\mathbb{R}^d} \frac{p_{ij}(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} < 1 + \frac{M}{m^2} \tag{2.32}$$

so (2.10) is fulfilled. The last condition, (2.11), requires that the minimum of the KDE $\tilde{p}_n$ over D go to zero no faster than some power of n. This condition is apparently necessary since, in general, the m.s. convergence (2.8) does not guarantee the corresponding pointwise convergence (so that an argument such as in (2.13) could be invoked), except in cases where $\tilde{p}_n$ and p are extremely regular (smooth), e.g., members of a Sobolev space.

3. in terms of the important constants governing the error bounds, we see that approximation over regions containing low probability events is more difficult than regions containing high probability events, demanding more training data. Ignoring the aiding factor of $(\log n)^{d/(d+4)}$ in the result of observation 3 above, n grows roughly at

rate (no greater than) $m^{2d/(d+4)-\alpha}$ for a fixed level of error in the uniform approximation case. The corresponding rate in the m.s. approximation case, using the result of observation 4 above, is $m^{d/(d+4)-\alpha}$.

4. since uniform approximation is a more stringent condition than m.s. approximation, it is not surprising that the asymptotic rate of convergence for the uniform approximation case is somewhat slower than that for the m.s. approximation case. For a given $\alpha > 2$ and assuming for simplicity $c_n \searrow c$, the rate of convergence in the uniform case is slower by a factor of $(n/\log n)^{2d/(d+4)}$ compared with the m.s. case. In both cases, however, the rates can be made arbitrarily fast by selecting α sufficiently large. This result is somewhat obvious from the construction of the approximating regularized SIRBFN, as the scaling factor n^α being applied to both the training output vector y and the input-dependent regularization parameter $\lambda_n(z)$ is designed to dominate (in norm) the regularization matrix G_n. Eliminating G_n from (1.15) gives weights that are consistent with those for the PWRE. Note that the minimum rate of output scaling is n^2 for uniform approximation vs. n for m.s. approximation.

An immediate consequence of the uniform and m.s. approximation theorems is the trivial consistency (in the same modes) of regularized SIRBFNs in F_z when constructed to approximate known consistent PWREs. The argument could be made, however, that the family F_z with input-dependent regularization is not a natural one, especially in applications where, as will be described, a single regularization parameter determined once from a realized training set t_n is used over the entire domain of the regularized SIRBFN. While the introduction of F_z is motivated by its usefulness in the proofs, the arguments contained therein imply nonetheless that over any given compact set $D \subset \mathbf{R}^d$, the approximating regularized SIRBFNs have λ_n growing asymptotically at rate *at least* $\Omega\left(n^{\alpha+4/(d+4)} \log^d n\right)$ for $\alpha > 2$ in the uniform case and $\Omega\left(n^{\alpha+4/(d+4)}\right)$ for $\alpha > 1$ in the m.s. case, i.e., at least roughly $\Omega(n)$ in both cases. By selecting the regularization parameter sequence to be a.o., we can extend the m.s.-consistency result from F_z to the ordinary noninput-dependent regularized SIRBFNs. To do so, the next section establishes between a relationship between the m.s.f.e. $\tilde{R}_2$ and the usual global (m.s.) risk R_2.

2.2 The Relationship between $\widetilde{R}_2$ and R_2

In this section, we show that under fairly mild conditions on f, $\tilde{f}_n$, and T_n, the m.s.f.e. $\tilde{R}_2$ converges to the true global risk R_2 for stochastic T_n when the input process $\{Z(i)\}$ has a stationary measure $P = P_Z$ and density p with respect to Lebesgue measure. Note that the proof presented is quite general in that we do not assume any particular parametric form for either $\tilde{f}$ or the relationship between the input process $\{Z(i)\}$ and the output process $\{Y(i)\}$. The proof is based on the following lemma for the convergence in probability of one-nearest neighbour distances. For simplicity, the lemma is demonstrated only for the case where Z, the evaluation point, is independent of the training input sequence Z_n, as defined below. The extension to the dependent case is fairly straightforward and will be developed in Chapter 4 for time series applications. Here the notation $X \sim P$ means that X is a r.v. defined with respect to the measure P.

Lemma 2. *Let* $\{Z(1), Z(2), \ldots, Z(n), Z\} \stackrel{\Delta}{=} \{Z_n, Z\} \sim P_{Z,Z_n} \stackrel{\Delta}{=} P_{Z_n} P_Z = P_{Z_n} P$, *i.e., Z is independent of Z_n, and $P_{Z(i)} = P$, $i = 1, 2, \ldots, n$, i.e., Z_n has stationary marginal measure P. Then for each $\epsilon > 0$,*

$$P_{Z,Z_n}\left\{z, z_n : \min_{j=1,2,\ldots,n} \|z - z(j)\| > \epsilon\right\} \leq q^{n/2}(\epsilon) \stackrel{n\to\infty}{\to} 0 \tag{2.33}$$

where $\| \bullet \|$ is the Euclidean norm in $\mathbb{R}^d$ and $0 \leq q(\epsilon) < 1$ is defined as

$$q(\epsilon) \stackrel{\Delta}{=} \int_{\{(x,y)\in\mathbb{R}^d\times\mathbb{R}^d:\|x-y\|>\epsilon\}} dP(x)dP(y) \tag{2.34}$$

Proof. Let $\epsilon > 0$ be given. Set $A_{n,j}(\epsilon) \stackrel{\Delta}{=} \{z, z_n : \|z - z(j)\| > \epsilon\}$. We use the independence bound implied by Cauchy-Schwarz for the intersection of a finite collection of events $\{F_i\}_{i=1}^n$ defined with respect to a common probability measure P

$$P\left(\bigcap_{i=1}^{n} F_i\right) = \mathbb{E}\left[\prod_{i=1}^{n} I(F_i)\right] \leq \prod_{i=1}^{n} P^{1/2}(F_i) \tag{2.35}$$

where $I(\bullet)$ is the indicator function for event $\bullet$, so that

$$P_{Z,Z_n}\left(\bigcap_{j=1}^{n} A_{n,j}(\epsilon)\right) \leq \prod_{j=1}^{n} P^{1/2}_{Z,Z(j)}(A_{n,j}) \tag{2.36}$$

From the stationarity of marginal measure for $\{Z(i)\}$ and the independence of Z from Z_n, we have for $j = 1, 2, \ldots, n$

$$P_{Z,Z(j)}(A_{n,j}) = \int_{\{(x,y)\in\mathbb{R}^d\times\mathbb{R}^d:\|x-y\|>\epsilon\}} dP(x)dP(y) \tag{2.37}$$

$$\triangleq q(\epsilon) \tag{2.38}$$

It suffices therefore to show that for all $\epsilon > 0$, $q(\epsilon) < 1$. If, for a given ϵ, $q(\epsilon) = 1$, then $P_{Z,Z(j)}\left(A^c_{n,j}(\epsilon)\right) = 0$. Since P (and hence $P_{Z,Z(j)}$) is absolutely continuous with respect to Lebesgue measure with density p, the support of p must have nonzero Lebesgue measure. This fact combined with the a.e. continuity of p implies that an arbitrary open ball (of nonzero radius) centred about almost all points in the support of p must have nonzero measure with respect to P, hence $P_{Z,Z(j)}\left(A^c_{n,j}(\epsilon)\right) = q(\epsilon) > 0$ for all $\epsilon > 0$. □

We can now verify the claim A3 in Section 1.3 by showing the convergence of $\tilde{R}_2$ to R_2 (as defined in (1.4)). Since $\tilde{R}_2$ appears to be a random-design quadrature formula for R_2, it may appear at first glance that a simple law of large numbers (LLN) type argument would suffice. If, for instance, $\tilde{R}_2$ were evaluated over *another* set of sample input data $T'_m \triangleq \{Z'_i\}_{i=1}^m$ whose marginal conditional distribution $P_{T'_m|T_n} \triangleq \prod_{i=1}^m P_{Z'_i|T_n}$ satisfies $P_{Z'_i|T_n} = P_{Z|T_n}$ for $i = 1, 2, \ldots, m$, then conditioned on T_n, such LLNs would indeed apply to show that (in the notations of the Introduction) under appropriate conditions

$$\sup_{\tilde{f}\in\mathcal{F}} \left|\tilde{L}_2\left(\tilde{f}, f, T'_m\right) - R_2^*\left(\tilde{f}, f\right)\right| \overset{m\to\infty}{\to} 0 \tag{2.39}$$

Since in our case T_n is used for both design and evaluation of the estimate $\tilde{f}_n$, this situation does not apply: instead we shall give a direct proof under some additional assumptions.

Theorem 2. *Assume that f is bounded as $|f| < L_f$ and Lipschitz with constant K_f over $\mathbb{R}^d$. If $\{Z(i)\}$ and $\{Y(i)\}$ are such that:*

1. *there exists a positive constant L_p satisfying*

$$\sup_{x\in\mathbb{R}^d} |p(x)| < L_p \tag{2.40}$$

where p is the common marginal density for $\{Z(i)\}$.

2. *there exist positive constants L and K for the regularized SIRBFN estimate $\tilde{f}_n(\bullet, T_n)$ constructed from T_n satisfying for $n = 1, 2, \ldots$,*

$$\sup_{z \in \mathbb{R}^d} \left| \tilde{f}_n(z, T_n) \right| \leq L \ a.s.\text{-}P_{T_n} \tag{2.41}$$

$$K_n \leq K \ a.s.\text{-}P_{T_n} \tag{2.42}$$

where K_n is a Lipschitz constant for $\tilde{f}_n$.

then

$$\left| R_2\left(f, \tilde{f}_n\right) - \tilde{R}_2\left(f, \tilde{f}_n\right) \right| \overset{n \to \infty}{\to} 0 \ a.s.\text{-}P_{T_n} \tag{2.43}$$

Proof. For convenience of notation, let $v_{t_n}(\bullet) \triangleq \left(f(\bullet) - \tilde{f}_n(\bullet, t_n)\right)^2$. Consider the ϵ-cover $B_n(\epsilon)$ induced by a realized training sequence t_n i.e., $B_n(\epsilon) \triangleq \bigcup_{i=1}^n B_{n,i}(\epsilon)$, $B_{n,i}(\epsilon) \triangleq \{z, t_n : \|z - z(i)\| < \epsilon\}$. An equivalent disjoint cover may be obtained by replacing $B_{n,i}(\epsilon)$ in the definition of $B_n(\epsilon)$ with $D_{n,i}(\epsilon) \triangleq B_{n,i}(\epsilon) \cap V_{z_n}(z(i))$ where $V_{z_n}(z(i))$ is the Voronoi cell[1] centred at $z(i)$ of the partition induced by the input sequence z_n contained in t_n. Decompose R_2 with respect to $B_n(\epsilon)$, where $\epsilon = \epsilon(n)$ (as will be explained later) so that

$$\begin{aligned} R_2\left(f, \tilde{f}_n\right) &\triangleq \int_{z, t_n} v_{t_n}(z)\, dP(z, t_n) \\ &= \int_{(z,t_n) \in B_n(\epsilon)} v_{t_n}(z)\, dP(z, t_n) + \int_{(z,t_n) \in B_n^c(\epsilon)} v_{t_n}(z)\, dP(z, t_n) \end{aligned} \tag{2.44}$$

By the assumed boundedness of f and condition (2.41) on $\tilde{f}_n$, Lemma 2 implies that the latter integral can be made arbitrarily small for n sufficiently large since for any $\delta > 0$,

$$\int_{(z,t_n) \in B_n^c(\epsilon)} v_{t_n}(z)\, dP(z, t_n) \leq L_v P_{Z, Z_n}\left(B_n^c(\epsilon)\right) \leq \delta \text{ for } n \geq 2 \log\left(\frac{\delta}{L_v}\right) / \log q(\epsilon) \tag{2.45}$$

where $L_v = (L_f + L)^2$ is a global upper bound on v. For the former integral, we may write

$$\begin{aligned} \int_{(z,t_n) \in B_n(\epsilon)} v_{t_n}(z)\, dP(z, t_n) &= \sum_{i=1}^n \int_{(z,t_n) \in D_{n,i}(\epsilon)} v_{t_n}(z)\, dP(z, t_n) \\ &\overset{-}{\underset{+}{\gtrless}} \sum_{i=1}^n \int_{(z,t_n) \in D_{n,i}(\epsilon)} \left(v_{t_n}(z(i)) \mp K_v \epsilon\right)\, dP(z, t_n) \\ &\overset{-}{\underset{+}{\gtrless}} \int_{t_n} \left[\sum_{i=1}^n \left(v_{t_n}(z(i)) \mp K_v \epsilon\right) P_{Z|T_n}\left(D_{n,i}(\epsilon) | T_n = t_n\right) \right] dP(t_n) \end{aligned} \tag{2.46}$$

[1] i.e., $V_{z_n}(z(i)) \triangleq \left\{ z \in \mathbb{R}^d : \|z - z(i)\| \leq \|z - z(j)\| \forall j = 1, 2, \ldots n \right\}$

by the definition of $D_{n,i}(\epsilon)$ and the fact that f and $\tilde{f}_n$ are bounded and Lipschitz implies the same for v with Lipschitz constant not greater than $K_v \triangleq 2(L_f + L)(K_f + K)$. Here the notation $a \underset{+}{\overset{-}{\gtrless}} f(b \mp c)$ is shorthand for the double inequality $f(b-c) \leq a \leq f(b+c)$, where f is an expression containing $b \mp c$. The remainder term containing $K_v\epsilon$ can be bounded uniformly over all possible training realizations t_n (equivalently, over all possible training input realizations z_n) since

$$P_{Z|T_n}\left(D_{n,i}(\epsilon)\,|\,T_n = t_n\right) \leq L_p(2\epsilon)^d. \quad \forall t_n \text{ and } i = 1, 2, \ldots, n \tag{2.47}$$

by (2.40) and where we have used the (Euclidean) volume of a d-dimensional cube in $\mathbb{R}^d$ with edge 2ϵ to upper-bound the volume of corresponding closed ball of radius ϵ. Hence we have the deviation bound

$$\left|\int_{z,t_n \in D_n(\epsilon)} v_{t_n}(z)\, dP(z, t_n) - \int_{t_n} \sum_{i=1}^{n} v_{t_n}(z(i)) P_{Z|T_n}\left(D_{n,i}(\epsilon)\,|\,T_n = t_n\right)\, dP(t_n)\right|$$

$$\leq nK_v L_p \epsilon (2\epsilon)^d \tag{2.48}$$

For the remainder term $r(n) \triangleq nK_v L_p \epsilon (2\epsilon)^d$ to vanish as $n \to \infty$, we require that $\epsilon^{d+1} = \mathcal{O}\left(1/n^{1+\beta}\right)$ for some $\beta > 0$. At the same time, the inequality

$$q(\epsilon) = 1 - P_{Z,Z(i)}\left(B_{n,i}(\epsilon)\right) \geq 1 - L_p(2\epsilon)^d \tag{2.49}$$

implies that we cannot let ϵ decrease too quickly as $n \to \infty$ if (2.45) is to be satisfiable for $\delta = \delta(n) = \mathcal{O}(1/n^\alpha)$ with $\alpha > 0$, since for x small, $\log(1-x) \approx -x$. In other words, for (2.45) to hold with $L_v > \delta(n) \overset{n\to\infty}{\to} 0$, it is necessary that $\epsilon^d = \Omega\left(1/n^{1-\gamma}\right)$ for some $\gamma \in (0, 1)$. Equating the two exponents gives the relationship between β and γ as

$$0 < \beta < 1/d. \qquad \gamma = \frac{1 - \beta d}{1 + d} \tag{2.50}$$

Returning to the integral term in (2.48), we recombine the iterated expectation and note that

$$\begin{aligned}
&\left|\int_{t_n} \sum_{i=1}^{n} v_{t_n}(z(i))\, dP(t_n) - \int_{t_n} \sum_{i=1}^{n} v_{t_n}(z(i)) P_{Z|T_n}\left(D_{n,i}(\epsilon)\,|\,T_n = t_n\right)\, dP(t_n)\right| \\
&\leq \sup_{i=1,2,\ldots,n} |v_{t_n}(z(i))| \left|\sum_{i=1}^{n} \left(\frac{1}{n} - \int_{z,t_n \in D_{n,i}(\epsilon)} dP(z, t_n)\right)\right| \\
&\leq L_v \left|1 - \sum_{i=1}^{n} P_{Z,T_n}\left(D_{n,i}(\epsilon)\right)\right| \qquad (2.51) \\
&\leq L_v \left|1 - P_{Z,T_n}\left(B_n(\epsilon)\right)\right| = L_v q^{n/2}(\epsilon) \\
&\leq \delta \qquad (2.52)
\end{aligned}$$

where we have again invoked Lemma 2 in the last line for δ as defined in (2.45). Combining the inequalities (2.45), (2.48), and (2.52) yields

$$\begin{aligned}\left|R_2\left(f,\tilde{f}_n\right)-\tilde{R}_2\left(f,\tilde{f}_n\right)\right| &\leq r(n)+2\delta(n)\\ &=\mathcal{O}\left(n^{-\beta}\right)+\mathcal{O}\left(n^{-\alpha}\right), \qquad 0<\beta<1/d, \quad \alpha+\beta<1\end{aligned} \tag{2.53}$$

where the condition $\alpha+\beta<1$ is required for (2.45) to hold. This result implies that the asymptotic rate of convergence of $\tilde{R}_2\left(f,\tilde{f}_n\right)$ to $R_2\left(f,\tilde{f}_n\right)$ can be made arbitrarily close to (but strictly less than) $\mathcal{O}\left(n^{-1/d}\right)$, from which the desired conclusion follows.

□

As an aside, the conditions required for the theorem to hold are not quite as restrictive as they may appear at first sight; in particular, conditions (2.41) and (2.42) hold, e.g., when the estimate $\tilde{f}_n$ converges uniformly a.s.-P_{T_n} to a bounded, Lipschitz function. Although, as mentioned in the Introduction, Plutowski et al. (1994) show a similar result in their proof that the (O)CV proxy for the global m.s.e. R_2 converges (unbiasedly) a.s., the proof here relaxes the requirement for i.i.d. samples to allow dependent samples from a bounded and stationary input process density. We also provide an estimate of the rate of convergence as approximately $\mathcal{O}\left(n^{-1/d}\right)$, which illustrates the "curse of dimensionality" prevalent in high-dimensional estimation problems (a similar inverse-dependence of the sample exponent on the input dimensionality can be seen in the KDE/NWRE results of (Bosq, 1996) cited earlier). While at this point we are not aware of any other rigourous demonstrations of the convergence of the m.s.f.e. $\tilde{R}_2$ to the global m.s.e. R_2 under our chosen circumstances, existing SLLN results appear to strongly suggest a possible improvement to an exponential rate of convergence independent of (or only weakly dependent on) the input process dimensionality.

2.3 Implications for Regularized SIRBFN Estimation

2.3.1 Asymptotic Optimality with Respect to Mean-Squared Error

The convergence of $\tilde{R}_2$ to R_2 demonstrated above is desirable because the a.o. of the CV and GCV parameter selection methods with respect to the loss $\tilde{L}_2$ and the risk $\tilde{R}_2$ for linear estimates has been studied under various scenarios. The ones mentioned below

follow the usual practice of assuming that one is observing a sequence of fixed means which have been additively corrupted by an independent sequence of bounded variance, zero-mean errors. At first glance, this model may appear somewhat restrictive but it can include the stochastic T_n case described in the Introduction by setting the error $\epsilon(i) \triangleq Y(i) - f(\boldsymbol{Z}(i))$, where f is the (assumed stationary) regression function $f(\bullet) \triangleq \mathbb{E}[Y(i)|X(i) = \bullet]$, and conditioning on a fixed input sequence $\boldsymbol{z}_n$, i.e., the problem is reduced to the case of noisy T_n. Note that after this change, stochastic T_n with i.i.d. samples $(\boldsymbol{Z}(i), Y(i))$ lead to a conditionally *heteroskedastic* model in which the errors $\epsilon(i)$ are independent zero-mean but not identically distributed, as their variances $\sigma^2(i)$ are conditional on the corresponding input realization $\boldsymbol{z}(i)$ (e.g., see the discussion in Section 4.10 of (Eubank, 1988)). If this dependence on $\boldsymbol{z}(i)$ is known, however, as

$$\sigma^2(i) = \frac{\sigma^2}{\psi^2(\boldsymbol{z}(i))} \tag{2.54}$$

where ψ is a known function, then replacing each pair $(\boldsymbol{z}(i), y(i))$ in t_n with the pseudo-data $(\boldsymbol{z}(i), \psi(\boldsymbol{z}(i))y(i))$ gives an equivalent homoskedastic error with respect to $\psi \cdot f$, i.e. $\epsilon'(i) \triangleq \psi(\boldsymbol{Z}(i))Y(i) - \psi(\boldsymbol{Z}(i))f(\boldsymbol{Z}(i))$ is i.i.d. zero-mean with common variance σ^2 (e.g., see Section 2.1 of (Gallant, 1987)). While this result is theoretically appealing, its practice when ψ is unknown effectively adds the complication of estimating the conditional variance of $Y(i)$ given $\boldsymbol{Z}(i) = \boldsymbol{z}(i)$, which may only be feasible in the case that the heteroskedasticity is mild, i.e., ψ is a relatively smooth function. In any case, the scenarios are:

1. homoskedastic errors and continuous parameter set (Li, 1985).

2. homoskedastic errors and discrete parameter set (with cardinality possibly increasing in the number of data, subject to certain conditions) (Li, 1987).

3. heteroskedastic errors and discrete parameter set (with cardinality possibly increasing in the number of data, subject to certain conditions) (Andrews, 1991).

While CV has been shown to be generally a.o. under all of the above scenarios, the same is known to be true for GCV only in the first two scenarios; more specifically, the third study gave sufficient conditions for a.o. which are not satisfied by GCV for ridge-regression problems such as those involving the selection of the regularization parameter for SIRBFNs. This current drawback of GCV can be theoretically addressed using one of the variance-equalizing methods as described above; in practice, however, the actual loss of performance

due to heteroskedasticity with GCV varies and is not inevitably significant (see the discussion in item 3 of Section 3.3.2).

For our purposes, if we assume that the estimates $\tilde{f}_n$ satisfy the uniform boundedness condition (2.41) of Theorem 2, then the input sequence conditioning of the standard results can be avoided. Let us define the input sequence conditioned version of $\tilde{R}_2$ as

$$\tilde{R}_2(\lambda; z_n) \triangleq \mathbb{E}_{T_n|Z_n}\left[\frac{1}{n}\sum_{i=1}^{n} v_{T_n}(Z(i))\middle| Z_n = z_n\right] \text{ so that } \tilde{R}_2(\lambda) = \mathbb{E}_{Z_n}\left[\tilde{R}_2(\lambda; Z_n)\right] \tag{2.55}$$

where v_{T_n} is as defined in the proof of Theorem 2 with $\tilde{f}_n$ designed using the regularization parameter λ. On occasion, we shall write $\tilde{R}_2$ as $\tilde{R}_{2,n}$ to emphasize its dependence on n when it is not clear from context, e.g., when discussing $\tilde{R}_2$ for fixed λ. Now the $\tilde{R}_2$-a.o. of a regularization parameter sequence $\left\{\tilde{\lambda}_n\right\}$ means that, given any infinite-length input sequence realization z_∞ with subsequence $z_n = \{z_\infty\}_{1:n}$,

$$\frac{\tilde{R}_{2,n}\left(\tilde{\lambda}_n; z_n\right)}{\inf_{\lambda\in\mathbb{R}^+}\tilde{R}_{2,n}(\lambda; z_n)} \overset{n\to\infty}{\to} 1 \tag{2.56}$$

Because the convergence occurs for all possible realizations of z_∞ and hence z_n, it holds i.p.–P_{Z_n}, i.e., given any $\epsilon > 0$,

$$P_{Z_n}\left\{z_n : \left|\frac{\tilde{R}_{2,n}\left(\tilde{\lambda}_n; z_n\right) - \inf_{\lambda\in\mathbb{R}^+}\tilde{R}_{2,n}(\lambda; z_n)}{\inf_{\lambda\in\mathbb{R}^+}\tilde{R}_{2,n}(\lambda; z_n)}\right| > \epsilon\right\} \overset{n\to\infty}{\to} 0 \tag{2.57}$$

By the uniform boundedness of $\tilde{f}_n$, the denominator of the above event must be upper bounded by some $L > 0$ for all n so that (2.57) implies

$$P_{Z_n}\left\{z_n : \left|\tilde{R}_{2,n}\left(\tilde{\lambda}_n; z_n\right) - \inf_{\lambda\in\mathbb{R}^+}\tilde{R}_{2,n}(\lambda; z_n)\right| > L\epsilon\right\} \overset{n\to\infty}{\to} 0 \tag{2.58}$$

i.e., $\tilde{R}_{2,n}\left(\tilde{\lambda}_n; Z_n\right)$ converges to $\inf_{\lambda\in\mathbb{R}^+}\tilde{R}_{2,n}(\lambda; Z_n)$ i.p.–P_{Z_n}. Then, using the fact that for uniformly integrable $\tilde{f}_n$, convergence i.p.–P_{Z_n} also implies convergence in $L_1(P_{Z_n})$ (e.g., see Lemma 4.6.6 of (Gray, 1987)), we may conclude that

$$0 \le \tilde{R}_2\left(\tilde{\lambda}_n\right) - \inf_{\lambda\in\mathbb{R}^+}\tilde{R}_2(\lambda) \overset{n\to\infty}{\to} 0 \tag{2.59}$$

As an aside, note that the convergence specified by a.o. and (2.59), while similar, are not identical. In the former case, the *rate* of convergence is at least that of the denominator $\inf_{\lambda\in\mathbb{R}^+}\tilde{R}_{2,n}(\lambda; z_n)$ to zero, assuming that the class of estimates containing $\tilde{f}_n$ is

$\tilde{R}_2$-consistent (i.e., given any z_∞, $\inf_{\lambda \in \mathbf{R}^+} \tilde{R}_{2,n}(\lambda; z_n) \overset{\text{i.p. } n\to\infty}{\to} 0$); in the latter case, the rate of convergence is not known.

Using the unconditional a.o. of (2.59), it is now possible to show that an a.o. regularization parameter sequence $\{\lambda_n\}$ also asymptotically minimizes the global m.s.e. R_2, i.e., m.m.s.e. estimation over the class of regularized SIRBFNs is asymptotically realized.

Corollary 1. *Let the SIRBFN regularization parameter sequence $\left\{\tilde{\lambda}_n\right\}$ be selected by a procedure under conditions for which the sequence is a.o. in the m.s.f.e. (risk) $\tilde{R}_2 = \tilde{R}_{2,n}$ as per (2.59). In addition, assume that the conditions for Theorem 2 hold. Then*

$$\left| R_2\left(\tilde{\lambda}_n\right) - \inf_{\lambda \in \mathbf{R}^+} R_2(\lambda) \right| \overset{n\to\infty}{\to} 0 \tag{2.60}$$

i.e., $\left\{\tilde{\lambda}_n\right\}$ is a consistent sequence of estimates for the global m.s.e.-minimizing regularization parameter.

Proof. By the triangle inequality, we have

$$\begin{aligned} \left| R_2\left(\tilde{\lambda}_n\right) - \inf_{\lambda \in \mathbf{R}^+} R_2(\lambda) \right| \leq & \left| R_2\left(\tilde{\lambda}_n\right) - \tilde{R}_{2,n}\left(\tilde{\lambda}_n\right) \right| \\ & + \left| \tilde{R}_{2,n}\left(\tilde{\lambda}_n\right) - \inf_{\lambda \in \mathbf{R}^+} \tilde{R}_{2,n}(\lambda) \right| \\ & + \left| \inf_{\lambda \in \mathbf{R}^+} \tilde{R}_{2,n}(\lambda) - \inf_{\lambda \in \mathbf{R}^+} R_2(\lambda) \right| \end{aligned} \tag{2.61}$$

For the last term, we may use the bound

$$\left| \inf_{\lambda \in \mathbf{R}^+} \tilde{R}_{2,n}(\lambda) - \inf_{\lambda \in \mathbf{R}^+} R_2(\lambda) \right| \leq \sup_{\lambda \in \mathbf{R}^+} \left| \tilde{R}_{2,n}(\lambda) - R_2(\lambda) \right| \tag{2.62}$$

(since in the case that the infima occur for different λ, say $\tilde{\lambda}^*$ for $\tilde{R}_{2,n}$ and λ^* for R_2, one of $\left|\tilde{R}_{2,n}\left(\tilde{\lambda}^*\right) - R_2\left(\tilde{\lambda}^*\right)\right|$ or $\left|\tilde{R}_{2,n}\left(\lambda^*\right) - R_2\left(\lambda^*\right)\right|$ must be at least as large as the left-hand side term being bounded). Then the first and last terms converge to zero a.s.-P_{T_n} by virtue of (2.43) (which holds independent of the choice of λ for the last term), while the middle term does the same by (2.59), completing the proof. □

2.3.2 Consistency with Respect to Mean-Squared Error over Compacta

By the previous theorems, a heuristic argument for the m.s.-consistency of regularized SIRBFNs with $\tilde{\lambda}_n$ selected by a suitable a.o. parameter estimation procedure

can be posited as follows: suppose that conditions are such that a NWRE $\tilde{f}_n'$ is m.s.-consistent, i.e., $R_2\left(f, \tilde{f}_n'\right) \overset{n\to\infty}{\to} 0$ in some mode m. Then $\tilde{f}_{n,\infty}$, the a.s. uniform SIRBFN approximation of $\tilde{f}_n'$ constructed according to Theorem 1, must also be m.s.-consistent, i.e., $R_2\left(f, \tilde{f}_{n,\infty}\right) \overset{n\to\infty}{\to} 0$ (in mode m). The a.s. convergence of $\tilde{R}_2$ to R_2 further implies that the m.s.f.e. $\tilde{R}_2\left(f, \tilde{f}_{n,\infty}\right) \overset{n\to\infty}{\to} 0$ (in the same mode m). At the same time, the m.s.f.e. $\tilde{R}_2\left(f, \tilde{f}_n\right)$ of the SIRBFN $\tilde{f}_n$ that is the same SIRBFN as $\tilde{f}_{n,\infty}$ except with $\tilde{\lambda}_n$ chosen a.o. must (by the definition of a.o.) be asymptotically no greater than $\tilde{R}_2\left(f, \tilde{f}_{n,\infty}\right)$. Hence $\tilde{R}_2\left(f, \tilde{f}_n\right) \overset{n\to\infty}{\to} 0$ (in mode m). Once again, the a.s. convergence of $\tilde{R}_2$ to R_2 can be invoked to conclude that $R_2\left(f, \tilde{f}_n\right) \overset{n\to\infty}{\to} 0$ (in mode m), i.e., regularized SIRBFNs with $\tilde{\lambda}_n$ selected a.o. are m.s.-consistent. Before giving a more detailed proof, it would be convenient to extend the results in Theorem 1 to hold globally in some sense rather than locally over a fixed compact set. Although this modification can be effected for both the uniform and m.s. SIRBFN approximation cases, it turns out that the simpler extension in the uniform case is sufficient for our intended applications. We begin by modifying the definition found in Section 3.2.2 of (Bosq, 1996) to say that a sequence $\{S_n\}$ of compact sets in $\mathbb{R}^d$ is *regular* (with respect to a density p) if there exists a sequence $\{\beta_n\}$ and a monotonically increasing sequence $\{\rho_n\}$ of strictly positive real numbers such that for each n,

$$\inf_{z\in S_n} p(z) \geq \beta_n \text{ and } \operatorname{diam}(S_n) \triangleq \sup_{x,y\in\mathbb{R}^d} \|x-y\| \leq \rho_n \tag{2.63}$$

(in (Bosq, 1996), ρ_n is set to n^γ; the reason for our generalization will become clear shortly). A probability density p for a measure P is said to be *regular in probability* if it admits a regular sequence $\{S_n\}$ with $\lim_{n\to\infty} P(S_n) = 1$. We shall also define a tail condition for a density p as

$$\exists R > 0 \text{ such that } p(z) \geq \mu(\|z\|) \text{ when } \|z\| > R \tag{2.64}$$

where $\mu : \mathbb{R}^+ \to \mathbb{R}^+$ is a bicontinuous monotonically decreasing function. We can now give the following

Theorem 3. *Using the same conditions and notations as for (2.7), if p is regular in probability with regular sequence $\{S_n\}$ such that (2.63) and (2.64) jointly satisfy*

$$\lim_{n\to\infty} \frac{\log\left(1/\mu\left(\rho_n\right)\right)}{\log n} < \infty \tag{2.65}$$

then a regularized SIRBFN $\tilde{f}_{n,\infty} \in F_z$ may be constructed such that

$$\sup_{z\in S_n} \left|\tilde{f}_{n,\infty}(z) - \tilde{f}_n'(z)\right| \overset{n\to\infty}{\to} 0 \text{ a.s.-}P_{T_n} \tag{2.66}$$

i.e., the uniform approximation holds globally i.p.-P for almost all training sets T_n.

Proof. Let $\{S_n\}$ be a regular sequence for p with sequences $\{\beta_n\}$ and $\{\rho_n\}$ satisfying (2.63) and μ and R as defined in (2.64). In the proof of (2.7), we replace m, the lower bound on the density p over D, with β_n which, by the previous conditions, satisfies

$$\beta_n \geq \mu(\rho_n), \quad n > N_1 \text{ where } N_1 \text{ is such that } \rho_n > R \quad \text{for all } n > N_1 \tag{2.67}$$

The required condition on α for the approximation error to decrease asymptotically to zero then becomes

$$\alpha > \max\left(2, \log\left(\frac{2C}{h_n^d}\right) / \log n + \log\left(1/\mu(\rho_n)\right) / \log n\right), \quad n > N_1 \tag{2.68}$$

which can be made independent of n as in (2.29) because of the relative stability condition (2.65). Since p is regular in probability, the uniform approximation holds with arbitrarily high $\boldsymbol{Z}$-probability when (2.68) is fulfilled, completing the proof. □

Note that the joint condition (2.65) is equivalent to $\mu(\rho_n) = \Omega(n^{-q})$ for some $q > 0$. Examples of densities regular in probability which satisfy (2.65) include Gaussians and their mixtures (with $\rho_n = r\sqrt{\log n}$ and $\mu(\bullet) = \exp(-k\|\bullet\|^2)$ for some strictly positive reals r and k) and compactly supported, bounded, continuous densities with polynomial tails, e.g., uniform density.

With the above tools in place, we may derive a m.s.-consistency result for regularized SIRBFNs with a.o. regularization sequence $\left\{\tilde{\lambda}_n\right\}$. The notion of "corresponding NWRE" was defined previously in Section 2.1. For greater generality, we do not rely on the assumption that that $\boldsymbol{Z}$ is independent of the training set T_n in this proof, although, as we shall discuss later in Chapter 4, the same can be achieved for the previous proofs with only slightly more effort. The following lemma concerning a triangle-inequality-type bound for R_2 will be useful:

Lemma 3. *For f, g, and $h \in L_2\left(\mathbb{R}^d, P_{\boldsymbol{Z},T_n}\right)$,*

$$|R_2(f,h) - R_2(g,h)| \leq R_2(f,g) + 2\sqrt{R_2(f,g)R_2(g,h)} \tag{2.69}$$

Proof. As a shorthand, let $\|\bullet\|_P \triangleq \mathbf{E}_{\mathbf{Z},T_n}[\bullet]$. Then

$$\begin{aligned}
|R_2(f,h) - R_2(g,h)| &= \left| \left\|(f-h)^2\right\|_P - \left\|(g-h)^2\right\|_P \right| \\
&\leq \| |f-h+g-h| \cdot |f-g| \|_P \\
&\leq \| (|f-g| + 2|g-h|) \cdot |f-g| \|_P \\
&\leq \left\|(f-g)^2\right\|_P + 2\left\|(f-g)^2\right\|_P^{1/2} \left\|(g-h)^2\right\|_P^{1/2} \qquad (2.70)
\end{aligned}$$

which is the desired conclusion. □

Theorem 4. *Assume that Theorem 2 holds and let the stationary marginal density p for* $\{\boldsymbol{Z}(i)\}$ *be regular in probability. Then the regularized SIRBFN with regularization parameter sequence* $\left\{\tilde{\lambda}_n\right\}$ *chosen by a procedure that is* R_2*-a.o. as per (2.59) is globally m.s.-consistent whenever the corresponding NWRE is globally m.s.-consistent.*

Proof. Let $\tilde{f}_n'$ be the globally m.s.-consistent NWRE and $\tilde{f}_{n,\infty}$ the corresponding uniform approximation to $\tilde{f}_n'$ constructed by Theorem 1 to satisfy (2.7) and its extended version (2.66) in Theorem 3. Applying the preceding lemma,

$$\left|R_2\left(\tilde{f}_{n,\infty}, f\right) - R_2\left(\tilde{f}_n', f\right)\right| \leq R_2\left(\tilde{f}_{n,\infty}, \tilde{f}_n'\right) + 2\sqrt{R_2\left(\tilde{f}_n', f\right) R_2\left(\tilde{f}_{n,\infty}, \tilde{f}_n'\right)} \qquad (2.71)$$

Because $\tilde{f}_n'$ is globally m.s.-consistent, the right-hand side above vanishes so long as $R_2\left(\tilde{f}_{n,\infty}, \tilde{f}_n'\right)$ does too. Note that Theorem 3 implies that we have the convergence

$$\left|\tilde{f}_{n,\infty}(\boldsymbol{Z}) - \tilde{f}_n'(\boldsymbol{Z})\right|^2 \overset{n\to\infty}{\to} 0 \quad \text{i.p.-}P_{\boldsymbol{Z},T_n} \qquad (2.72)$$

which, by the assumption that the $\tilde{f}_{n,\infty}$ and $\tilde{f}_n'$ are uniformly integrable, also implies the corresponding m.s. convergence

$$R_2\left(\tilde{f}_{n,\infty}, \tilde{f}_n'\right) \overset{n\to\infty}{\to} 0 \qquad (2.73)$$

Hence,

$$\left|R_2\left(\tilde{f}_{n,\infty}, f\right) - R_2\left(\tilde{f}_n', f\right)\right| \overset{n\to\infty}{\to} 0 \Rightarrow R_2\left(\tilde{f}_{n,\infty}, f\right) \overset{n\to\infty}{\to} 0 \qquad (2.74)$$

so that $\tilde{f}_{n,\infty}$ is also globally m.s.-consistent. From this and the a.s. convergence of $\tilde{R}_2$ to R_2, it follows that

$$\tilde{R}_2\left(\tilde{f}_{n,\infty}, f\right) \overset{n\to\infty}{\to} 0 \qquad (2.75)$$

On the other hand, by definition, for each n,

$$\tilde{R}_2\left(\tilde{f}_{n,\infty}, f\right) \geq \inf_{\lambda \in \mathbb{R}^+} \tilde{R}_{2,n}(\lambda) \Rightarrow \inf_{\lambda \in \mathbb{R}^+} \tilde{R}_{2,n}(\lambda) \overset{n \to \infty}{\to} 0 \tag{2.76}$$

and, reversing the previous reasoning, by the definition of a.o. in (2.59) and the a.s. convergence of $\tilde{R}_2$ to R_2,

$$\tilde{R}_2\left(\tilde{f}_n, f\right) \overset{n \to \infty}{\to} 0 \Rightarrow R_2\left(\tilde{f}_n, f\right) \overset{n \to \infty}{\to} 0 \tag{2.77}$$

where $\tilde{f}_n$ is the SIRBFN corresponding to $\tilde{f}'_n$ with a.o.-selected regularization parameter sequence. □

2.3.3 Discussion

With Theorem 4, we have verified claim A4 made in the Introduction. That is, we have shown that regularized SIRBFNs with a.o. chosen regularization parameter sequence are (globally) m.s.-consistent under conditions for which the NWRE is both m.s.-consistent and uniformly approximable according to Theorem 3. Specific situations for which these conditions hold are analyzed in detail in the following two chapters on probability estimation/pattern classification and nonlinear time series prediction. We should mention that as far as kernel shape is concerned, kernels satisfying (1.24) and (1.25), e.g., the Gaussian, remain admissible in the regression context. Thus, at least a partial answer is provided to the question of a justifiable rather than *ad hoc* design procedure for RBFNs. Following are some other relevant remarks in this regard:

1. the reasoning supporting m.s.-consistency in the regularized SIRBFN case carries over to the regularized *random centres* RBFN case (1.19) if the centre (equivalently, basis function) selection method used is proven to be R_2-consistent. More precisely, denote the centre selection policy by n and the number of centres it selects from the N available training data in T_N as $n(N) \leq N$. Then it would be sufficient that

$$\inf_{\lambda \in \mathbb{R}^+} R_{2,n(N)}(\lambda) \overset{N \to \infty}{\to} 0 \tag{2.78}$$

for a regularized random centres RBFN with centres chosen by policy n and a.o. regularization parameter sequence $\left\{\tilde{\lambda}_n\right\}$ to be m.s.-consistent. The burden of this additional proof is avoided in the SI (full centres) RBFN construction.

2. in addition to being $\tilde{R}_2$-a.o., CV parameter selection procedures are also known to be $\tilde{L}_2$-a.o. i.p.-P_{T_n}, i.e.,

$$\frac{\tilde{L}_2\left(\tilde{\lambda}_n; z_n\right)}{\inf_{\lambda \in \mathbb{R}^+} \tilde{L}_2\left(\lambda; z_n\right)} \overset{n\to\infty}{\to} 1 \text{ i.p.-}P_{T_n} \tag{2.79}$$

so that the true loss for a *particular* training input sequence z_n is correctly estimated. This optimality condition is stronger than $\tilde{R}_2$-a.o. and can be useful when the inputs in z_n represent especially important evaluation points, e.g., the modes of the probability measure P_Z in the stationary input case. In the proof of Theorem 4, however, this $\tilde{L}_2$-a.o. is not necessary.

3. an appropriate level of smoothing through the choice of regularization parameter sequence is central to the consistency of the regularized SIRBFN. In fact, Theorem 8 of (Golitschek and Schumaker, 1990) effectively implies item A5 from the Introduction. That is, for noisy T_n in scenario 1 of Section 2.3.1, RBFNs designed with positive definite kernel function and $\lambda_n = 0$ for all n cannot be m.s.-consistent unless the number of basis functions grows more slowly than the number of data. For any training realization t_n, applying the said theorem yields

$$\tilde{R}_2\left(0; z_n\right) = \mathbb{E}\left[\epsilon^2(i)\right] \overset{\Delta}{=} \sigma^2, \quad \forall z_n \tag{2.80}$$

which is obvious since for any SIRBFN design with $\lambda = 0$, the predicted output vector $\tilde{y}_n$ equals the training output vector y_n. By the convergence of $\tilde{R}_2$ to R_2, this results implies that such unregularized networks must be have m.s.e. bounded away from zero for n sufficiently large, hence claim A5 of the Introduction is verified. This result merely confirms the ANN intuition that exact fitting of noisy t_n with n basis functions leads to an overfit and hence poor generalization. On the other hand, the situation for $\lambda_n = \mathcal{O}\left(n^\alpha\right)$ for some $\alpha > 2$ as in Theorem 3 is not quite as clear-cut, since the NWREs being approximated are known to be (m.s.) consistent. What we can conclude, however, is that such an effective choice of asymptotically increasing regularization parameter sequence (as for the NWRE) is not optimal in terms of minimizing either the m.s.f.e. $\tilde{R}_2$ or the actual loss $\tilde{L}_2$. More definitive statements require a specific analysis of the eigenvalues of the interpolation matrix G_n (Golitschek and Schumaker, 1990). Section 4.4 of (Wahba, 1990) performs this for the smoothing spline case and argues that $\tilde{R}_2$ cannot vanish unless, among other conditions, $\lambda_n \overset{n\to\infty}{\to} 0$.

Chapter 3

Probability Estimation and Pattern Classification

In this chapter, we prove certain results pertaining to the SIRBFN estimation of posterior probabilities using least-squares regression of indicator functions and the subsequent implications for pattern classification. Since no great effort is required to establish these results using the theoretical tools developed in Chapter 2, this material is included primarily for completeness and continuity in demonstrating the generality of the tools. As a result, the treatment is deliberately brief and theoretically-oriented compared to those which follow it, i.e., Chapter 4. Furthermore, we should add the caveat that the results obtained do not justify *a priori* such least-squares-based techniques as the most suitable or natural ones for classification problems; indeed, if a maximum likelihood estimate of the posterior probabilities is desired, then *logistic*, not least-squares, regression is appropriate (M^{c}Lachan, 1992). It is also intuitively clear that a posterior probability estimate which is optimal with respect to a squared-error criterion such as R_2 need not be optimal with respect to the average probability of classification error when used in an approximate Bayes decision rule, as described below. That said, since least-squares procedures are widely used in practice due to their analytical as well as computational tractability, a clearer understanding of their theoretical properties in such applications is arguably useful. Some pertinent discussion on these countervailing issues may be found in Sections 25.9 (for the former) and remarks at the ends of Sections 29.3 and 29.6 (for the latter), all from (Devroye et al., 1996)).

3.1 Problem Description

In this section, we set the notations and the basic framework within which we shall consider the related problems of probability estimation and pattern classification. The type of probability estimates that we shall be considering are the so-called *posterior* or conditional probabilities $P_A(\boldsymbol{x}) \triangleq \Pr(A|\boldsymbol{X} = \boldsymbol{x})$, where $\boldsymbol{X} = \boldsymbol{X}(\omega)$ is an $\mathbb{R}^d$-valued r.v. generically known as the *feature* and A is an event, both defined with respect to a common underlying sample space Ω. That such posterior probabilities arise naturally in pattern classification stems from the well-known *Bayes rule* for *optimal*, i.e., minimum average error, decisions when Ω is a discrete sample space representing the occurrence of each class. Without loss of generality, if there are M classes, we can equivalently enumerate the classes via a discrete r.v. $Y \in C = \{1, 2, \ldots, M\}$. Then given a realized feature $\boldsymbol{x}$ corresponding to an unknown class y, the Bayes rule $d^* : \mathbb{R}^d \mapsto C$ chooses the class $d^*(\boldsymbol{x})$, where

$$d^*(\boldsymbol{x}) \triangleq \arg\max_{c \in C} P_c(\boldsymbol{x}), \quad P_c(\boldsymbol{x}) \triangleq \Pr\{Y = c|\boldsymbol{X} = \boldsymbol{x}\} \tag{3.1}$$

i.e., Bayes rule chooses the class with *maximum a posteriori probability (MAP)*. On occasion we shall also equivalently work with the associated Bayes *discriminant functions* $\delta^*_{i,j} \triangleq P_i - P_j$, $i, j \in C$, so that given a realized feature $\boldsymbol{x}$, by Bayes rule, we would select class $i \in C$ iff $\delta^*_{i,j}(\boldsymbol{x}) > 0$ for all $j \neq i$, $j \in C$. The decision rule and the equivalent discriminant functions induce a corresponding set of M *decision regions* $\{D_c\}_{c \in C}$ where $D_c \triangleq \left\{\boldsymbol{x} \in \mathbb{R}^d : d^*(\boldsymbol{x}) = c\right\}$, i.e., a partition of $\mathbb{R}^d$ according to the class chosen when the classifier is presented with an input $\boldsymbol{x} \in \mathbb{R}^d$. In relation to these concepts, we are interested in conditions under which:

1. posterior probability estimates $\tilde{P}_{n,c}$, $c \in C$, constructed from an n-i.i.d. random sample $T_n = \left\{(\boldsymbol{X}_i, Y_i) \in \mathbb{R}^d \times C\right\}_{i=1}^n \sim P_{T_n} = \prod_{i=1}^n P_{\boldsymbol{X},Y}$ are (m.s.) consistent
2. the consistency of the $\tilde{P}_{n,c}$ implies a corresponding consistency for the *plug-in* or *approximate* Bayes decision rule $\tilde{d}_n$ constructed by substituting the estimated for the actual posterior probabilities, i.e.,

$$\tilde{d}_n(\boldsymbol{x}) \triangleq \arg\max_{c \in C} \tilde{P}_c(\boldsymbol{x}) \tag{3.2}$$

The approximate Bayes discriminant functions $\tilde{\delta}^{i,j}_n : \mathbb{R}^d \mapsto \mathbb{R}$, $i, j \in C$, are defined analogously with the estimated posterior probabilities in place of the actual ones.

The consistency of a decision rule $d : \mathbb{R}^d \to C$ is defined with respect to its *local risk* $r(d, \boldsymbol{x}) \triangleq \Pr\{d(\boldsymbol{X}) \neq Y | \boldsymbol{X} = \boldsymbol{x}\}$ and its *global risk* $R(d) \triangleq \mathbb{E}[r(d, \boldsymbol{X})]$, which is also the average probability of classification error. Decision rules with risk converging (in an appropriate mode) to the minimum risk $R^* \triangleq R(d^*)$ achieved by the Bayes rule are known as *Bayes risk consistent (BRC)* (in that mode). Following the point of view in Chapter 2, we shall say BRC when we mean the $L_1(P_{T_n})$ convergence of the *training set conditional* Bayes risk $\mathbb{E}_{\boldsymbol{X}|T_n}[r(d, \boldsymbol{X})|T_n]$ to R^*: other definitions commonly found in the literature are *weak* BRC (when the convergence is i.p.-P_{T_n}) and *strong* BRC (when the convergence is a.s.-P_{T_n}).

3.2 Review of Current Approaches

While the general comments in Section 1.2 concerning current approaches to RBFN design remain valid, we shall expand the discussion here to include both generic ANN and kernel-based design strategies for probability estimation and classification, largely in recognition of the importance attached to those fields in their own right.

3.2.1 Traditional ANN Approaches

As ANN training procedures were (and, for the most part, remain) based on minimizing data-based proxies for the global m.s.e. J_2 (as defined in (1.6)), it is not surprising that one approach to an M-class classifier design codes the training targets $Y_i = c \in C$ as the Euclidean unit basis vectors $\boldsymbol{e}_c$ in $\mathbb{R}^M$ so that Y_i is equivalently represented by a vector $\boldsymbol{Y}_i \in \{0,1\}^M$. In addition to being particularly well-suited for the $(0,1)$ output range of the sigmoidal activation function commonly used in MLP networks, this approach, which we shall call the method of *least-squares fitting of indicator functions (LSFI)*, is motivated by the observations that:

1. $\boldsymbol{e}_c$ is the vector of the M class indicator function outputs for the event $Y = c$, i.e., $\boldsymbol{e}_c = \left[I_{\{Y=i\}}(Y = c)\right]_{i=1}^{M}$
2. regression of a class indicator function yields an L_2 estimate of the corresponding posterior class probability, since

$$P_c(\boldsymbol{x}) = \mathbb{E}\left[I_{\{Y=c\}}(\boldsymbol{X}) \middle| \boldsymbol{X} = \boldsymbol{x}\right] \tag{3.3}$$

Note that for each class $c \in C$, we may therefore equivalently write the regression model for P_c as

$$Z_c = P_c(\boldsymbol{X}) + \epsilon_c, \quad c \in C \tag{3.4}$$

where $Z_c = I_{\{Y=c\}}(\boldsymbol{X}) \in \{0,1\}$ is the indicator function for class c given input $\boldsymbol{X}$ and $|\epsilon_c| = |Z_c - P_c(\boldsymbol{x})| \leq 1$ for all $\boldsymbol{x} \in \mathbb{R}^d$. By the standard properties of conditional expectation, ϵ_c is uncorrelated with X, $\mathbb{E}[\epsilon_c] = 0$, and $\mathrm{Var}[\epsilon_c] \triangleq \sigma_c^2 \leq 1$.

The initial practical motivations behind the LSFI method have been accompanied by a widely-held belief in the ANN community that the method results in m.s.-consistent estimates of posterior probabilities. The line of reasoning involved is as follows:

1. versions of the (strong) law of large numbers (SLLN) imply the (a.s.) convergence of the proxy $\tilde{R}_2$ to the actual desired m.s.e. cost function J_2 as the size of the training set $n \to \infty$.

2. ANN training procedures which, given a training realization t_n, select the $\tilde{R}_2$-minimizing network parameters $\tilde{\boldsymbol{\theta}}_n$ (and hence network $\tilde{f}\left(\bullet, \tilde{\boldsymbol{\theta}}_n\right)$) must equivalently select the J_2-minimizing network parameters as $n \to \infty$.

3. the absolute J_2-minimizing function when Y is the indicator function for a given class is the posterior probability function for that class. Hence ANN training procedures asymptotically (in the number of training data n) yield the J_2-minimizing approximating function $\tilde{f}$ within the approximating function class $\mathcal{F} = \mathcal{F}_{\boldsymbol{\theta}}$.

4. if in addition $\mathcal{F}$ has "sufficient functional capacity" to contain the unknown posterior probabilities, then the ANN output functions recovered asymptotically should be identically equal to the unknown posterior probabilities so that Bayes classification performance is possible.

Several early proofs, e.g., those by (Hampshire and Perlmutter, 1990) and (Kanaya and Miyake, 1991), essentially follow this model. The work of Ruck et al. (1990) is slightly more guarded in their conclusions as it ends at point 3 and acknowledges that the quality of the approximate Bayes rule depends on the "functional capacity" of the ANN architecture (although the relationship between R_2 and the expected classification error rate is not monotonic, as we shall see). This argument, while suggestive, suffers from a number of deficiencies, many of which were noted in the Introduction in more general terms:

1. ignoring for the moment the lack of verification of the conditions under which SLLNs hold, step 1 of the argument is incomplete without the key requirement that the a.s. convergence of $\tilde{R}_2\left(\tilde{f}, T_n\right)$ to $J_2\left(\tilde{f}\right)$ is uniform over $\mathcal{F} \ni \tilde{f}$ (see, e.g., Section 4.1.2 of (White, 1989)). Otherwise, the null set over which the convergence fails can depend on the unknown function f being estimated, an undesirable feature in applications. A proper statement of the desired consistency result with a detailed discussion of the corresponding conditions can be found as Theorem 1 of (White, 1989).

2. in most ANN applications, one does not usually know *a priori* a *minimally* parameterized function class which is guaranteed to contain the unknown posterior probabilities (otherwise, standard finite parameter estimation methods such as maximum likelihood could be used). On the other hand, for general unknown functions f, e.g., continuous f over a known compact set, an infinitely-parameterized function class $\mathcal{F}$ is necessary to contain f. Therefore for the antecedent condition of point 4 to hold from the outset, one would have to select a network with an unjustifiably large number of free parameters compared to the amount of training data at hand and risk overfitting, leading to poor generalization as discussed in Section 1.2.2. As Barnard (1992) noted in his comment on (Kanaya and Miyake, 1991), the notion of "sufficient functional capacity" is largely an impractical one.

A somewhat different approach is taken by (Richard and Lippmann, 1991) when they assume from the start that the ANN training procedure minimizes the ensemble average

$$\Delta \triangleq \mathbf{E}\left[\sum_{c=1}^{M}\left(Y_c - f_c\left(\boldsymbol{X}, \boldsymbol{\theta}\right)\right)^2\right] \tag{3.5}$$

where $Y_c \triangleq I_{\{Y=c\}}(\boldsymbol{X})$, f_c is the ANN output function for class $c \in C$, and $\boldsymbol{X}$ is the feature input r.v. With some basic probabilistic manipulations, they arrive at

$$\Delta = \mathbf{E}\left[\sum_{c=1}^{M}\left(P_c(\boldsymbol{X}) - f_c\left(\boldsymbol{X}, \boldsymbol{\theta}\right)\right)^2\right] + \mathbf{E}\left[\sum_{c=1}^{M}\mathrm{Var}\left(Y_c|\,\boldsymbol{X}\right)\right] \tag{3.6}$$

Note that this expression could be derived simply by applying the derivation of (1.7) to each term of the summation. In any case, since the latter expectation term does not depend on the ANN output functions f_c, they (correctly) conclude that minimizing Δ with respect to $\boldsymbol{\theta}$ is equivalent to minimizing the first term with respect to $\boldsymbol{\theta}$. Clearly, however, this does not imply that *each term* in the former expectation is simultaneously minimized, unless

one assumes that the f_c each have effectively independent parameters, which is not usually the case for the reasons given previously with regards to overfitting. Even then, without a proper invocation of an appropriate SLLN, there remains the questionable assumption that ANN training actually selects the network parameters to minimize Δ instead of $\tilde{R}_2$.

In contrast to these incomplete attempts at proving that the ANN-LSFI method yield "estimates" of the Bayes posterior probabilities, the results of Chapter 2 can be used to demonstrate rigourously the consistency of the LSFI method using regularized SIRBFNs with an a.o. parameter selection technique. We can then demonstrate the BRC of the method and show that, from an asymptotic discrimination point of view, the positivity and normalized output constraints often imposed on posterior probability estimates are not necessary (although, as we shall discuss, they may be desirable under certain circumstances). Furthermore, these theoretical properties of the regularized SIRBFN design method are proven under precise, (approximately) realizable conditions unlike those often required in the other proofs for the other methods.

3.2.2 Kernel-Based Approaches

Given that probability density estimation was a primary motivation behind kernel-based design methods, their advanced development, particularly in theoretical aspects, should come as no surprise. Assuming that a training set T_n is available, posterior probability estimates can be derived in two ways:

1. the *plug-in KDE method*: partition the samples in T_n by class, i.e., $T_n = \cup_{c \in C} T_{c,n_c}$, $\sum_{c \in C} n_c = n$, where T_{c,n_c} contains only the n_c samples pairs in T_n with $y_i = c$. For each class $c \in C$, form the KDE $\tilde{p}_c$ of the actual *class-conditional* density p_c from the sub-training set T_{c,n_c} in the usual way, i.e., each KDE is of the form (1.23), albeit not necessarily with the same kernel K or bandwidth sequence $\{h_n\}$. Estimate the corresponding *prior class probability* or *mixing proportion* π_c via, e.g., the maximum-likelihood estimator $\tilde{\pi}_c \stackrel{\Delta}{=} n_c/n$ (this estimate is also minimum-variance unbiased). Then applying Bayes rule with the $\tilde{p}_c$ and $\tilde{\pi}_c$ in place of the p_c and π_c yields the estimate $\tilde{P}_c$ of the actual posterior probability P_c of class c as

$$\tilde{P}_c(\boldsymbol{x}) \stackrel{\Delta}{=} \tilde{\pi}_c \tilde{p}_c(\boldsymbol{x}) / \sum_{i \in C} \tilde{\pi}_i \tilde{p}_i(\boldsymbol{x}) \tag{3.7}$$

which may then be used in an approximate Bayes rule.

2. the *KRE method*: generate a KRE (typically a NWRE) of the posterior class probability for each class $c \in C$ from the M sub-training sets

$$T_{n,c} \triangleq \left\{ \left(I_{\{Y=c\}}(\boldsymbol{X}_i), \boldsymbol{X}_i \right) \in \{0,1\} \times \mathbb{R}^d \right\}_{i=1}^{n} \tag{3.8}$$

for use in an approximate Bayes rule. As this method is regression-based, it may be considered a LSFI approach.

Among others, Van Ryzin (1966) demonstrates the weak BRC of the plug-in KDE method and derives the order of convergence to consistency under certain conditions on the kernel K and bandwidth sequence $\{h_n\}$. Similarly, Glick (1972) shows that the a.s. pointwise convergence of the density estimates $\tilde{p}_c$, $c \in C$, is sufficient to ensure the a.s. convergence of the sample-based classification error rate to the true classification error rate uniformly over the domain of all decision rules. For the KRE method, many of the works discussed in the Introduction also include BRC results: (Stone, 1977) proves the BRC i.p. of his weighted output mean estimators for multiple classification, which is extended by (Devroye, 1981) in the case of the NWRE to universal BRC a.s. when the bandwidth sequence $\{h_n\}$ satisfies $nh_n^d / \log n \overset{n\to\infty}{\to} \infty$ in addition to the basic NWRE m.s.-consistency conditions (1.26) and (1.32). It is clear that proofs of the BRC of the KRE method carry over to the plug-in KDE method when the latter is consistent as a posterior probability estimate in the same mode that the KRE is, and in this sense there is no essential difference between the two methods. As pointed out in Section 16.2 of (Devroye et al., 1996), however, the corresponding rates to BRC can be arbitrarily loose, in the sense that the convergence (in some appropriate mode) of the parameter and density estimates comprising the plug-in KDE estimate of the posterior probability may be either faster or slower than the convergence of the Bayes risk. This shortcoming is shared with the KRE method, where the m.s. convergence of the posterior probability estimates implies the BRC of the associated approximate Bayes rule (as will be proven). In a practical sense, however, the KRE LSFI method has an advantage over the plug-in KDE method in that the former uses *all* the available training data to estimate of each posterior class probability, while the latter estimates the posterior probability for a given class with only the training data labelled with that class. In other words, even "negative" examples in the form "$\boldsymbol{x}_i$ is not in class c", contribute to the estimate of each posterior class probability using the KRE LSFI method. This choice provides more sample data for each problem and can improve the quality of the resultant estimates, particularly for the a priori less probable classes.

Perhaps the closest in spirit to the regularized RBFN framework for estimating posterior probabilities is the method of *inequality-constrained smoothing splines* proposed by Villalobos and Wahba (1987). In their case, by selecting the solution space to be $H(m, d)$, the vector space of d-dimensional functions whose partial derivatives up to total order m (in the distributional sense) are square-integrable, and choosing an appropriate penalty functional $H_2 f$ in (1.12), the solution function becomes a linear expansion in multivariate *thin plate* spline basis functions of the form $K_m(r) = \theta_m r^{2m-d} \ln(r)$ (and the standard multivariate monomials) instead of the typical Gaussian basis function. They then solve a regularized matrix interpolation equation analogous to (1.15) except that the solution is subject to a pair of linear equality and inequality constraints, where the latter is an attempt to enforce a $[0, 1]$ range for the posterior probability estimate over a pre-specified grid of points. For this purpose, a computational procedure is proposed for a suitably constrained version of GCV (called *GCVC*) for the regularization parameter. Although experimental results with simulated data for a two-dimensional, two-class problem with $m = d = 2$ and a mixture of Gaussian densities indicate visually reasonable modelling of the posterior probabilities, the study does not offer any theoretical results on the consistency for the proposed method. While the (approximate) positive unit range constraint on the solution function is arguably natural when the objective lies solely in the probability estimates themselves, we shall show later that such a range constraint is unnecessary when the objective is discrimination using the approximate Bayes rules formed from those estimates. Indeed, it is clear that what is *necessary* for classification is the approximation of the *relative* magnitudes of the posterior probabilities, not their *absolute* magnitudes individually. Thus the extra computational complexity introduced by the positive unit range constraint and subsequent GCVC do not, at least in principle, offer any distinct advantage in classification performance for sufficiently large samples. If direct modelling of posterior probabilities is the goal, penalized likelihood estimates of the *logit* function $\log(P_c/(1 - P_c))$ which treat the output class variable as a multinomially-distributed r.v. may be considered more appropriate (O'Sullivan et al., 1987).

3.3 Theoretical Results for Regularized Probability Estimates

3.3.1 Consistency of Probability Estimates in Mean-square and Bayes Risk

By Theorem 4 of the previous chapter, regularized SIRBFN posterior probability estimates designed according to the LSFI method with a.o. regularization parameter sequence are m.s.-consistent whenever the corresponding NWRE is. Because the training data are i.i.d., the conditional error r.v. $\epsilon(i)$ is no longer homoskedastic, hence of the available scenarios listed in Section 2.3.1, only scenario 3 is potentially relevant. Specifically for that scenario, Andrews (1991) shows that OCV but not GCV satisfies a set of sufficient conditions for $\tilde{L}_2$ and $\tilde{R}_2$-a.o. parameter selection (when conditioned on the training input sequence) in the class of linear estimates. Strictly speaking, since this result is only proven for a discrete parameter space (whose cardinality is permitted to increase with the number of training data), it cannot be automatically assumed to hold for the continuous regularization parameter case. On the other hand, for practical reasons the OCV/GCV cost functions are commonly approximately minimized by discrete sampling of the regularization parameter in any case (e.g., see Section 4.6). With these provisos in mind, we may state the following specific instance of Theorem 4:

Theorem 5. *Assume that Theorem 2 holds and let the stationary marginal density p for $\{Z(i)\}$ be regular in probability. Then the regularized SIRBFN posterior probability estimates designed according to the LSFI method with regularization parameter sequence $\left\{\tilde{\lambda}_n\right\}$ R_2-a.o. as per (2.59) is globally m.s.-consistent whenever the corresponding NWRE is globally m.s.-consistent.*

Proof. By the LSFI method, the output class indicator r.v. Y is clearly bounded as $|Y| \leq 1$ and the i.i.d. input r.v. $\boldsymbol{Z}$ is clearly stationary, so result (1) of Theorem 1 and its extension Theorem 3 hold under our assumed conditions. The remainder of the proof parallels that of Theorem 4. □

From the m.s.-consistency of the regularized SIRBFN posterior probability estimates, the corresponding BRC is simple to infer from the basic chain of inequalities beginning with the following lemma. The lemma gives a bound relating the expected deviation of posterior probability estimates with the corresponding expected deviation in local risk of the approximate Bayes decision rule formed using those estimates.

Lemma 4 (Györfi (1978)). *Let* $\left\{\tilde{P}_{n,c}\right\}_{c\in C}$ *be a collection of estimates for a collection of posterior probabilities* $\{P_c\}_{c\in C}$ *and* $\tilde{d}_n$ *be the approximate Bayes rule formed from* $\left\{\tilde{P}_{n,c}\right\}_{c\in C}$. *Then*

$$\mathbb{E}\left[\left|r\left(\tilde{d}_n, \boldsymbol{X}\right) - r\left(d^*, \boldsymbol{X}\right)\right|\right] \leq \sum_{c\in C} \mathbb{E}\left[\left|\tilde{P}_{n,c}(\boldsymbol{X}) - P_c(\boldsymbol{X})\right|\right] \tag{3.9}$$

Theorem 6. *In addition to the notation of Lemma 4, let* d^* *represent the (exact) Bayes rule based on the collection of (true) posterior probabilities* $\{P_c\}_{c\in C}$. *Then*

$$\left|R\left(\tilde{d}_n\right) - R^*\right| \leq \sum_{c\in C} \sqrt{\mathbb{E}\left[\left|\tilde{P}_{n,c}(\boldsymbol{X}) - P_c(\boldsymbol{X})\right|^2\right]} \tag{3.10}$$

where $R^* \triangleq R(d^*)$ *is the Bayes (optimal) risk for the classification problem defined by* $\{P_c\}_{c\in C}$.

Proof. Applying Jensen's inequality to the left-hand side of (3.9) gives

$$\mathbb{E}\left[\left|r\left(\tilde{d}_n, \boldsymbol{X}\right) - r\left(d^*, \boldsymbol{X}\right)\right|\right] \geq \left|\mathbb{E}\left[r\left(\tilde{d}_n, \boldsymbol{X}\right) - r\left(d^*, \boldsymbol{X}\right)\right]\right| = \left|R\left(\tilde{d}_n\right) - R^*\right| \tag{3.11}$$

while Cauchy-Schwarz implies that the right-hand side satisfies

$$\mathbb{E}\left[\left|\tilde{P}_{n,c}(\boldsymbol{X}) - P_c(\boldsymbol{X})\right|\right] \leq \sqrt{\mathbb{E}\left[\left|\tilde{P}_{n,c}(\boldsymbol{X}) - P_c(\boldsymbol{X})\right|^2\right]} \tag{3.12}$$

Combining the two inequalities gives the desired result. □

Theorem 6 bounds the rate to BRC as being no slower than the square-root of the minimum rate to the R_2-consistency amongst the corresponding posterior class probability estimates. Theorem 6.5 of (Devroye et al., 1996) state that the *ratio* of left-hand side of (3.10) to its bound on the right-hand side vanishes (as $n \to \infty$) in the two-class case for the risk of any approximate Bayes rule formed from (weakly) m.s.-convergent posterior probability estimates. In fact, if the Bayes risk R^* is zero, i.e., in cases where perfect classification is (in principle) possible, $\left|R\left(\tilde{d}_n\right) - R^*\right|$ vanishes as fast as the *square* of the corresponding rates to m.s.-consistency of the posterior probability estimates. Devroye et al. (1996) also give an example of a case where convergence of an approximate Bayes rules' risk to the Bayes risk occurs *without* the m.s.-consistency of the class posterior probability estimates. Taken together, these results reinforce the point that while m.s.-consistency of posterior

probability estimates is certainly *sufficient* to guarantee a worst-case rate to BRC for the corresponding approximate Bayes rule, such consistency is not by any means necessary and, as is typical of such worst-case rates, may be of more theoretical than practical interest.

We should also point out another way of proving the BRC of regularized SIRBFN posterior probability estimates with a.o. regularization parameter sequence. This approach shows that the convergence i.p.-$P_{\boldsymbol{X},T_n}$ of the approximate Bayes discriminant functions to the actual Bayes discriminant functions is sufficient to ensure the convergence i.p.-$P_{\boldsymbol{X},T_n}$ of their classification decisions and hence risk or error rates. As discussed previously, this result implies that the positive unit range constraint imposed by the method of (Villalobos and Wahba, 1987) is not necessary for asymptotically consistent classification, only the replication of the signs of the discriminant functions are. In this restricted sense, our result supports the intuition "classification is easier than regression" enunciated in Section 6.7 of (Devroye et al., 1996). We begin by proving the two-class case as a lemma and then extend it to the multiclass case in a subsequent theorem.

Lemma 5. *Let $\tilde{P}_{n,1}$ and $\tilde{P}_{n,2}$ be globally m.s.-consistent estimates of posterior probabilities P_1 and P_2, respectively, according to the conditions of Theorem 4. Assume further that $\tilde{P}_{n,1} \neq \tilde{P}_{n,2}$ a.s.-$P_{\boldsymbol{X},T_n}$ and $P_1 \neq P_2$ a.s.-$P_{\boldsymbol{X}}$. Let $\tilde{d}$ (d^*) be the approximate (exact) Bayes decision rule formed from $\tilde{P}_{n,1}$ and $\tilde{P}_{n,2}$ (P_1 and P_2). Then*

$$\lim_{n\to\infty} P_{\boldsymbol{X},T_n}\left(\left\{\boldsymbol{x} \in \mathbb{R}^d, t_n \in \tau^n : \tilde{d}(\boldsymbol{x}) \neq d^*(\boldsymbol{x})\right\}\right) = 0 \tag{3.13}$$

Proof. For $c \in \{1, 2\}$, since we have assumed that $R_2\left(P_c, \tilde{P}_{n,c}\right) \overset{n\to\infty}{\to} 0$, we also have the convergence in probability

$$\left|\tilde{P}_{n,c}(\boldsymbol{X}) - P_c(\boldsymbol{X})\right| \overset{n\to\infty}{\to} 0 \text{ i.p.-}P_{\boldsymbol{X},T_n} \tag{3.14}$$

which implies the same convergence of corresponding approximate Bayes discriminant function $\tilde{\delta}_n \triangleq \tilde{P}_{n,1} - \tilde{P}_{n,2}$ to the actual Bayes discriminant function $\delta^* \triangleq P_1 - P_2$. Hence $D^* \triangleq \left\{\boldsymbol{x} \in \mathbb{R}^d : \delta^*(\boldsymbol{x}) > 0\right\}$ and $\tilde{D}_n \triangleq \left\{\boldsymbol{x} \in \mathbb{R}^d, t_n \in \tau^n : \tilde{\delta}_n(\boldsymbol{x}) > 0\right\}$ are the actual and approximate Bayes decision regions, respectively, for selecting class 1 over class 2. The desired result 3.13 is equivalently stated

$$\lim_{n\to\infty}\left(P_{\boldsymbol{X},T_n}\left(D^* \Delta \tilde{D}_n\right) + P_{\boldsymbol{X},T_n}\left(D^{*c} \Delta \tilde{D}_n^c\right)\right) = 0 \tag{3.15}$$

Let us consider the first probability term, as symmetry will allow us to apply the subsequent argument to the second probability term with only minor changes. By additivity for disjoint

events,

$$P_{\boldsymbol{X},T_n}\left(D^*\Delta\tilde{D}_n\right) = P_{\boldsymbol{X},T_n}\left(D^*\cap\tilde{D}_n^c\right) + P_{\boldsymbol{X},T_n}\left(D^{*c}\cap\tilde{D}_n\right) \tag{3.16}$$

For $(\boldsymbol{x},t_n)\in\left(D^*\cap\tilde{D}_n^c\right)\cup\left(D^{*c}\cap\tilde{D}_n\right)$, we have, by the definitions of δ^* and $\tilde{\delta}_n$, the implication

$$\left|\delta^*(\boldsymbol{x})-\tilde{\delta}_n(\boldsymbol{x})\right| < \epsilon \Rightarrow |\delta^*(\boldsymbol{x})| + \left|\tilde{\delta}_n(\boldsymbol{x})\right| < \epsilon \tag{3.17}$$

Letting $s_n \triangleq |\delta^*| + \left|\tilde{\delta}_n\right|$ and $S_{n,\epsilon} \triangleq \left\{\boldsymbol{x}\in\mathbb{R}^d, t_n\in\tau^n : s_n(\boldsymbol{x}) < \epsilon\right\}$, this implication is equivalently stated

$$\left(D^*\cap\tilde{D}_n^c\right)\cup\left(D^{*c}\cap\tilde{D}_n\right)\subseteq S_{n,\epsilon} \tag{3.18}$$

so that

$$P_{\boldsymbol{X},T_n}\left(D^*\cap\tilde{D}_n^c\right) + P_{\boldsymbol{X},T_n}\left(D^{*c}\cap\tilde{D}_n\right) \le P_{\boldsymbol{X},T_n}\left(S_{n,\epsilon}\right) \tag{3.19}$$

For any given n, we claim that $P_{\boldsymbol{X},T_n}(S_{n,\epsilon})$ can be made arbitrarily small by choosing ϵ sufficiently small. To see this, take $\epsilon = 1/m$ and let $A_m \triangleq S_{n,1/m}$ (for fixed n). Because $A_1 \supset A_2 \supset \cdots \supset A_m \supset \ldots$ forms a monotone decreasing sequence of events, by continuity in probability we have that

$$\lim_{m\to\infty} P_{\boldsymbol{X},T_n}(A_m) = P_{\boldsymbol{X},T_n}\left(\bigcap_{i=1}^{\infty} A_i\right) \tag{3.20}$$

The right-hand side $\cap_{i=1}^{\infty}A_i = A_0 \triangleq \left\{\boldsymbol{x}\in\mathbb{R}^d, t_n\in\tau^n : s_n(\boldsymbol{x}) = 0\right\}$ is precisely the set of points where neither the discriminant function δ nor $\tilde{\delta}_n$ yield an unambiguous decision, a set which by assumption has null $P_{\boldsymbol{X},T_n}$. Thus the claim is verified so that the first probability term on the right-hand side of (3.16) can be made arbitrarily small if n is sufficiently large. This same argument applies to the second probability term on the right-hand side of (3.16) by exchanging D and $\tilde{D}_n$ with D^c and $\tilde{D}_n^c$, respectively, and replacing δ and $\tilde{\delta}_n$ with $-\delta$ and $-\tilde{\delta}_n$, respectively. With these results in place, given any $\gamma > 0$, we can choose n sufficiently large so that

$$\max\left(P_{\boldsymbol{X},T_n}\left(D^*\Delta\tilde{D}_n\right), P_{\boldsymbol{X},T_n}\left(D^{*c}\Delta\tilde{D}_n^c\right)\right) < \gamma/2 \tag{3.21}$$

whence (3.13) is proven. □

Theorem 7. *Let* $\left\{\tilde{P}_{n,c}\right\}_{c\in C}$ *be a collection of globally m.s.-consistent estimates for a corresponding collection of* M *posterior probabilities* $\{P_c\}_{c\in C}$*, according to the conditions of Theorem 4. Assume further that for any* $i,j \in C$ *with* $i \neq j$*,* $\tilde{P}_{n,i} \neq \tilde{P}_{n,j}$ *a.s.-*$P_{\boldsymbol{X},T_n}$ *and* $P_i \neq P_j$ *a.s.-*$P_{\boldsymbol{X}}$*. Let* $\tilde{d}$ *(*d^**) be the approximate (exact) Bayes decision rule formed from* $\left\{\tilde{P}_{n,c}\right\}_{c\in C}$ *(*$\{P_c\}_{c\in C}$*). Then*

$$\lim_{n\to\infty} P_{\boldsymbol{X},T_n}\left(\left\{\boldsymbol{x}\in\mathbb{R}^d, t_n\in\tau^n : \tilde{d}_n(\boldsymbol{x}) \neq d^*(\boldsymbol{x})\right\}\right) = 0 \tag{3.22}$$

and, consequently,

$$\lim_{n\to\infty} R\left(\tilde{d}_n(\boldsymbol{x})\right) = R^* \tag{3.23}$$

i.e., the approximate Bayes decision rules are BRC.

Proof. For the multiclass case, we have the basic bound

$$P_{\boldsymbol{X},T_n}\left(\left\{\boldsymbol{x}\in\mathbb{R}^d : \tilde{d}(\boldsymbol{x}) \neq d^*(\boldsymbol{x})\right\}\right) \leq P_{\boldsymbol{X},T_n}\left(\bigcup_{i\in C}\bigcup_{\substack{j\in C,\\ j\neq i}}\left\{\boldsymbol{x}\in\mathcal{R}^d, t_n\in\tau^n : d^*_{i,j}(\boldsymbol{x}) \neq \tilde{d}^{i,j}_n(\boldsymbol{x})\right\}\right) \tag{3.24}$$

where $d^*_{i,j}$ and $\tilde{d}^{i,j}_n$ are the actual and approximate Bayes rules, respectively, for the two-class subproblem defined by P_i and P_j. The inequality arises from the fact that if the actual and approximate Bayes decision rules for the full M-class problem disagree for a given $\boldsymbol{x}\in\mathbb{R}^d$, then $d^*_{i,j}(\boldsymbol{x}) \neq \tilde{d}^{i,j}_n(\boldsymbol{x})$ for at least one pair of classes i and j with $i \neq j$, from the $N_2 \triangleq \binom{M}{2} = M(M-1)/2$ possible two-class subproblems over C. Given $\epsilon > 0$ and applying the union bound to the right-hand side of (3.24), we see that by choosing n sufficiently large so that

$$\max_{\substack{i,j\in C\\ i\neq j}} P_{\boldsymbol{X},T_n}\left(\left\{\boldsymbol{x}\in\mathcal{R}^d, t_n\in\tau^n : d^*_{i,j}(\boldsymbol{x}) \neq \tilde{d}^{i,j}_n(\boldsymbol{x})\right\}\right) \leq \frac{\epsilon}{N_2} \tag{3.25}$$

we satisfy (3.22). To show Theorem 7 using this result, let $\tilde{E}_n \triangleq \left\{(\boldsymbol{x},y)\in\mathbb{R}^d\times C, t_n\in\tau^n : \tilde{d}_n(\boldsymbol{x}) \neq y\right\}$ and $E^* \triangleq \left\{(\boldsymbol{x},y)\in\mathbb{R}^d\times C : d^*(\boldsymbol{x}) \neq y\right\}$ be the classifier error events for the approximate and actual Bayes decision rules, respectively. Since

$$\left|P_{\boldsymbol{X},Y,T_n}\left(\tilde{E}_n\right) - P_{\boldsymbol{X},Y}\left(E^*\right)\right| \leq P_{\boldsymbol{X},Y,T_n}\left(\tilde{E}_n \Delta E^*\right) \tag{3.26}$$

and because the event $\tilde{E}_n \Delta E^*$ occurs if and only if exactly one of the two decision rules commits an error, it is a subset of the event that the two decision rules disagree. But we have just proven in (3.22) that this disagreement probability can be made arbitrarily small, thus completing the proof of Theorem 7. □

3.3.2 Discussion

Some further remarks are warranted based on the previous developments:

1. in the alternative proof given in Theorem 7 for the BRC of approximate Bayes decision rules based on m.s. approximations of posterior probabilities, we have demonstrated that the approximate and actual Bayes decision rules agree with probability approaching one without any requirement that the m.s. approximations satisfy the positive unit range constraint of the actual posterior probabilities. Instead, m.s.-consistent posterior probability estimates are shown to result in corresponding i.p.-consistent approximations of the Bayes decision regions for each class, which is the minimum requirement for BRC classification. On the other hand, a plausible argument could be made that imposing such positivity and normalized output constraints on the posterior probability estimates should have some practical benefits, e.g., a more rapid m.s.-convergence to the actual posterior probabilities. While the heuristics behind this notion are clear, we are not aware of any theoretical evidence supporting it. Furthermore, given the results of (Devroye et al., 1996) regarding the rather weak connection between the rate to m.s.-consistency for posterior probability estimates and rate to BRC for the corresponding approximate Bayes decision rule, it is uncertain what value such theoretical evidence would have, if obtained.

2. to establish global BRC, we require that the density for the feature r.v. $\boldsymbol{X}$ satisfy the regularity conditions discussed in Section 2.3.2 so that Theorem 4 may hold. Often in the pattern recognition literature one deals with the class-conditional densities $\{p_c\}_{c \in C}$, where p_c is the density of $\boldsymbol{X}|Y = c$, e.g., see the description of the plug-in KDE method. Since the (unconditional) feature density p is related to the $\{p_c\}_{c \in C}$ by $p = \sum_{c \in C} \pi_c p_c$, where the $\{\pi_c\}_{c \in C}$ are the prior class probabilities, p is regular in probability if the $\{p_c\}_{c \in C}$ are.

3. while in principle GCV(C) is not provably a.o. for continuous parameter selection with heteroskedastic errors as in the LSFI approach, Villalobos and Wahba (1987) report little loss in modelling performance for their simulated data using the "plain", i.e., non-heteroskedasticity corrected, GCV procedure versus a weighted GCV procedure which does correct for the heteroskedasticity of the error (as elaborated in Section 2.3.1). This result accords with our general experience in the experiments of this thesis and has been corroborated by others, e.g., Section 4.9 of (Wahba, 1990). A more theoretically sound approach is to use the "ordinary" or leave-out-one CV procedure for parameter selection, although even this ameliorative measure has its limitations as indicated earlier.

We have now formally established the BRC of the approximate Bayes decision rules for m.s.-consistent posterior probability estimates via the regularized SIRBFN method with a.o. parameter selection procedure. The verification and accompanying discussion substantiate the long-standing belief within the ANN community that the LSFI approach can lead to BRC-approximate Bayes decision rules while pointing out some of the pitfalls and limitations of the LSFI methods which are not as well-known as they ought to be. Even though such theoretical results do not imply that the LSFI method is the most appropriate choice in general situations, their existence does at least support their application to problems for which the method is sufficient to achieve the desired results.

Chapter 4

Nonlinear Time Series Prediction

The following chapter is an extended version of (Yee and Haykin, 1998), albeit with slightly modified notations to agree with those established in this thesis.

By a "time series", we usually mean a pair of discrete-time stochastic process $\left\{(\boldsymbol{Z}(i), Y(i)) \in \mathbb{R}^d \times \mathbb{R}\right\}_{i=0}^{\infty}$. Time series which are in principle deterministic, e.g., *chaotic* time series, can be handled as stochastic processes with singular distributions or, more realistically, as processes with a deterministic component contaminated by additive noise (see Section 5.4 for more details). In applications, the vector input process $\{\boldsymbol{Z}(i)\}$ is often composed from one or more (observable) scalar processes $\{X_j(i)\}$, $j = 1, 2, \ldots, d$, via $\boldsymbol{Z}(i) \triangleq [X_j(i)]_{j=1}^{d}$. A particularly important construction of this kind is the *autoregressive (AR)* case in which $Y(i) \triangleq X(i)$, $X_j(i) \triangleq X(i-j)$ are the *delay inputs*, and $\boldsymbol{Z}(i)$ is the *delayed input vector*. Here d is called the *autoregressive* or *delayed input order* and represents the length of dependence between the present and the immediate past of the process $\{X(i)\}$. An AR time series with unbounded input order is in principle possible but, as a practical measure, we shall normally assume that d is effectively finite.

4.1 Problem Description

The *filtering* or *estimation* problem is to estimate $Y(i)$ given $\boldsymbol{Z}(i)$ with minimum J_2-risk, i.e., m.m.s.e. The *prediction* problem can viewed as a case of the filtering problem in which $\boldsymbol{Z}(i)$ is a delayed input vector, i.e., contains information only up to (and including) time step $i-1$. It is well-known that the optimum estimator in this situation is the condition mean $f_i(\bullet) \triangleq \mathbb{E}[Y(i)|\boldsymbol{Z}(i) = \bullet]$. Such optimal m.m.s.e. estimators are well-developed for the case where $\{Y(i)\}$ and $\{\boldsymbol{Z}(i)\}$ are jointly weakly (or wide sense) stationary (w.s.s.)

Gaussian processes, in which case f_i is a linear function, and for cases where $\{Y(i)\}$ and $\{\boldsymbol{Z}(i)\}$ are known to be related by a given parametric form. When neither of these conditions necessarily obtains, the nonparametric ANN and KRE approaches once again generate an estimate $\tilde{f}$ of f under the stochastic or noisy data models, as appropriate. The estimate $\tilde{Y}(i)$ is then formed by the plug-in estimate $\tilde{Y}(i) \triangleq \tilde{f}(\boldsymbol{Z}(i))$. Aside from the specific nomenclature of time series, the distinguishing characteristics of the time series estimation problem are:

1. the possibility that the various quantities of interest, e.g., the statistics of the input and output processes, may fluctuate with time, i.e. *nonstationarity*

2. the possibility that the training data may be dependent from sample to sample, i.e., *dependence*

as opposed to the i.i.d. assumptions under which functional estimation problems such as probability estimation and classification are overwhelmingly posed. Note that these two conditions need not occur simultaneously, which is somewhat fortuitous since dealing with both of these added complications together would indeed be a challenging task. For the most part, we shall content ourselves with specific cases of each condition individually to allow tractable analysis.

4.2 Review of Current Approaches

The comments of Section 1.2 concerning current approaches to RBFN design generally apply in the area of time series estimation, as they did in the probability estimation problem, except that in this case we shall focus our attention on only those approaches which explicitly relate to RBFs.

4.2.1 Traditional ANN Approaches

A survey of the literature indicates that time series prediction may be considered one of the original applications for RBFN. Early design methods are characterized by their assumption of stationarity and reliance on heuristics, often to improve prediction performance or reduce computational complexity. For example, when Casdagli (1989) applies SIRBFNs (without regularization) to the autoregressive prediction of chaotic time series,

the computational complexity of solving the SI equation (1.15) exactly for large n is reduced by considering only the fifty inputs in the training set t_n closest (in Euclidean norm) to a given input training point $z(i)$ when approximating its target value $y(i)$, resulting in a predictor with a piecewise rather than global smoothness normally associated with RBFNs. Broomhead and Lowe (1988) consider similar chaotic modelling problems for the doubling and quadratic maps using their pseudo-inverse method as outlined in Section 1.2.1; network parameters such as the number of centres and the kernel bandwidth are chosen *ad hoc*, except for the network centers which are chosen to be uniformly spaced within the known unit range of the maps.

Keeping with the autoregressive prediction of chaotic and cyclical time series, He and Lapedes (1993) propose a *successive approximation* approach in which the available training data are partitioned into disjoint subsets, each of which is used to provide the centres of a corresponding RBF subnetwork trained with the pseudo-inverse method to approximate the overall (common) training set. The outputs of these RBF subnetworks are then assigned linear weights via the standard least-squares pseudo-inverse approximation to the overall training set. Compared to the usual approach of designing a single SIRBFN using all the available data as centres and training targets, it is difficult to see what the provable advantages of this two-stage least-squares approximation design technique are, other than the lower computational complexity and lessened likelihood of singularity in the interpolation matrices for cyclical time series as claimed by the authors.

Although these pseudo-inverse-based approaches are motivated by the desire for lower computational complexity in design and better generalization by avoiding overfitting in the presence of noise, they offer neither any specific theory with which to select the network centres from the data nor do they indicate what sort of noise-immunity can be expected as the number of centres varies. More generally, these methods also do not explicitly address the issues of noisy/stochastic data models or possible correlation and nonstationarity in the time series data.

Another route to the design of RBFNs for statistical time series prediction lies in the point of view that the output time series is linear in a number of unknown *state variables* which are assumed *a priori* to be related to the delayed input vector (of a known order) in the observable inputs via a known radial kernel. In such a case, if the output process is assumed to contain additive white Gaussian noise so that the optimal linear weights are posteriorly Gaussian distributed, one may, as Terano et al. (1992) do, apply

the standard linear Kalman filter (which, in this case, reduces to the *recursive least-squares (RLS) algorithm*) to recursively estimate the required weights. Note that in this approach, although the centres of the RBFN naturally follow changes in the input time series, there is still no guarantee of suitability for nonstationary systems as the weight update algorithm assumes weak stationarity for all processes involved.

More recent efforts have been directed towards the development of principled time-adaptive RBF network so that both nonstationary and stationary processes may be *tracked* on an ongoing basis. One example is (Kadirkamanathan and Kadirkamanathan, 1996), which extends the work of (Terano et al., 1992) by using an *extended* version of the RLS algorithm that allows the optimal state-space weights $\boldsymbol{w}^*(i)$ to drift according to a random walk $\boldsymbol{w}^*(i+1) = \boldsymbol{w}^*(i) + \boldsymbol{e}(i)$, where $\boldsymbol{e}(i)$ is a Gaussian white noise process. For modally nonstationary time series, i.e., time series generated by piece-wise constant switching amongst a fixed number of state-space mappings, and first-order Markovian transition between modes, they further use a *multiple model algorithm* to select (via Bayes inference) the "best" predictor from a number of candidate models running in parallel. Other applications of Bayesian inference in the nonstationary case can be found in (Kadirkamanathan and Niranjan, 1993) and (Lowe and McLachlan, 1995). In these works, however, *arbitrary* nonlinear state-space mappings, i.e., those not necessarily in the linear span of the chosen radial basis functions, are accommodated by extended (in the case of (Kadirkamanathan and Niranjan, 1993)) and iterated (in the case of (Lowe and McLachlan, 1995)) extended Kalman filters of second and higher order which produce recursive Bayes estimates of the RBFN weights that best approximate (in mean-square) the nonlinear mapping. As with all methods, the success of these methods hinges on the validity of their accompanying assumptions.

Compared with the earlier pseudo-inverse-based design techniques, these more principled approaches can account for both nonstationarity and dependence in the time series data; the tradeoff is, of course, the need for stronger assumptions regarding the time series model, e.g., that the output process is a nonlinear function of the input process with additive Gaussian noise of a known covariance. Physical intuition can justify such assumptions in some situations but it would certainly be desirable if optimal estimates could be obtained without them.

4.3 The Kernel Regression Approach

With their localized smoothing akin to filtering, KRE finds natural application to statistical time series. Early motivation for this approach to time series analysis arose from the desire to apply the KDE to (strictly) stationary processes with dependent samples rather than i.i.d. samples, as is usually assumed (Földes, 1974; Bosq, 1973, 1975; Pham and Tran, 1991). From this point, study developed naturally for the NWRE as both a *function* estimate as well as a *plug-in* estimate described previously. Surveys may be found in (Györfi et al., 1989) and, more recently, in (Bosq, 1996). For both, the primary focus is on the problem of estimating the stationary regression function $f(\bullet) = \mathbb{E}[Y(i)|\boldsymbol{Z}(i) = \bullet], \forall i$, in the case of a stationary input density under a mixing condition. Györfi et al. (1989) show and provide rates for the a.s. uniform consistency of the NWRE as a function estimate over compact sets in $\mathbb{R}^d$ under ϕ, ρ, and α-mixing conditions. For stationary Markov processes of order $q < \infty$ under conditions leading to GPM, they also demonstrate the a.s. consistency in absolute value of the plug-in NWRE autoregressive predictor by comparison to an existing a.s. uniform consistency result over compact sets. Finally, an a.s. uniform consistency result for the NWRE as an autoregressive function estimate for stationary ergodic processes with continuous conditional densities is stated, although no rates are given. Considering the NWRE as a function estimate when the input process is GSM, Bosq (1996) proves its pointwise m.s.-consistency over $\mathbb{R}^d$ and its a.s. uniform consistency over regular sequences of sets (see (2.63)). Estimates of the rate of convergence for the plug-in NWRE predictor quadratic and absolute error in the case of stationary fixed-order Markov processes are proven, along with a brief note on the extension to general stationary processes. In all cases, conditions akin to those in the corresponding KDE cases are (not surprisingly) required to hold.

Actually, as developed in (Györfi et al., 1989), the assumption of stationary input density is necessary only if rates of convergence are required; otherwise, a condition such as (A.1) discussed preceding Theorem 1 is sufficient to ensure convergence. Thus so long as the regression function f is fixed, situations in which the nonstationarity in the output process $\{Y(i)\}$ is caused by a nonstationary input process $\{\boldsymbol{Z}(i)\}$ are permitted. Beyond this general case, the first survey goes on to examine specific cases of nonstationarity which remain within the regression framework developed. A particularly interesting one which we shall consider further on is the class of autoregressive processes of order $d \geq 1$ generated by

i.i.d. innovations, i.e.,

$$X(i) = f(\boldsymbol{X}_d(i-1)) + \epsilon(i), \qquad i = 1, 2, \ldots \tag{4.1}$$

where $\boldsymbol{X}_d(i) \triangleq [X(i-j)]_{j=0}^{d-1}$ and $\{\epsilon(i)\}$ is an i.i.d. noise process with zero mean, bounded variance σ^2, and independent of the initial state vector $\boldsymbol{X}_d(0)$. The process $\{X(i)\}$ is clearly dependent and Markov with nonstationary marginal input density. It can be shown that if f is bounded and the probability law of $\{\epsilon(i)\}$ is absolutely continuous with respect to Lebesgue measure, then $\{X(i)\}$ is *geometrically ϕ-mixing (GPM)* and the NWRE of f is a.s.-P_{T_n} convergent in absolute value (Theorem 3.4.11 of (Györfi et al., 1989)). The second survey, however, considers only a simple form of nonstationarity in which

$$Z(i) = X(i) + s(i), \quad i \in \mathbb{Z} \tag{4.2}$$

where $\{X(i)\}$ is an unobserved, scalar real-valued strictly stationary process and $\{s(i)\}$ is a deterministic sequence. Under some conditions on the perturbation induced by $\{s(i)\}$ on the marginal density p of $\{X(i)\}$ and the joint density $p_{0,1}$ of $(X(0), X(1))$, it is demonstrated that the NWRE estimate of the autoregression $E[Z(i)|\boldsymbol{Z}_i(i)]$ is a.s. consistent in absolute value with rate n^{δ}, where $\delta \geq 0$ is determined by the perturbation conditions. In effect, this result states that the NWRE exhibits some robustness to mild forms of nonstationarity.

A related area involving nonstationary processes to which the NWRE has been applied is the *identification* of nonlinear systems. Rutkowski (1985a) considers identifying in the limit a measurable function $m : \mathbb{R}^d \to \mathbb{R}$ from a noisy training set T_n satisfying

$$Y(i) = f_i(\boldsymbol{Z}(i)) + \epsilon(i), \quad i \in \mathbb{N} \tag{4.3}$$

where $\{\boldsymbol{Z}(i)\}$ is an i.i.d. process with density p and independent of the i.i.d. zero-mean, finite variance noise process $\{\epsilon(i)\}$. Under these conditions, we see that f_i is the regression function of $Y(i)$ with respect to $\boldsymbol{Z}(i)$, and the nonstationarity of the output process $\{Y(i)\}$ is due solely to the time-variation of the regression function. Then, if one imposes the *quasi-stationarity* condition

$$\lim_{i \to \infty} \sup_{z \in \mathbb{R}^d} |f_i(z) - f(z)| = 0 \tag{4.4}$$

such that $f_i \cdot p$ converges (as $i \to \infty$) to $f \cdot p$ either uniformly over $\mathbb{R}^d$ or in $L_p\left(\mathbb{R}^d, P_{T_n}\right)$-norm sufficiently fast for $p = 1$ or 2, then the NWRE $\tilde{f}_n$ of f_n based on T_n is pointwise

consistent i.p. or a.s.-P_{T_n} according to the conditions placed upon the bandwidth and kernel. Note, however, that the number of kernel functions in the NWRE must grow in step with the size of the training set to achieve such consistency, which makes this result principally of theoretical interest. This shortcoming is alleviated, however, when (Rutkowski, 1985b) presents a *recursive* version of the aforementioned algorithm to *track* f_i, which need not be convergent in the sense of (4.4). What is salient about this recursion is that the contribution $a(n+1) > 0$ of a new training input datum $z(n+1)$ to $\tilde{f}_n$ via its kernel function $K\left((\bullet - z(n+1))/h_n\right)$ asymptotically vanishes (but not too quickly, as $\sum_{i=1}^{\infty} a(i) = \infty$), i.e., the function estimate is, for practical purposes, of fixed size when n is sufficiently large. Among one of several conditions, if the bandwidth and kernel for the NWRE are such that the KDE $\tilde{p}_n$ is pointwise consistent i.p. or a.s.-P_{T_n} with respect to p, the stationary input density, then this algorithm is also pointwise consistent in the same mode.

Once again, compared to the ANN RBF approaches, the KRE-based approaches offer stronger theoretical support for their designs decisions. The issue of dependent training data arising in correlated time series is addressed by adapting existing KDE results for the same, while the more general question of nonstationarity is analyzed for certain important cases. With some modifications, it is expected that the regularized SIRBFN approach can exploit this body of knowledge to demonstrate its consistency and asymptotic R_2-optimality (which in this case translates into m.s. prediction error), and that is precisely the topic for the forthcoming sections.

4.4 Consistency of Prediction

For time series with stationary input density and dependence described by a GSM condition, the relevant m.s.-consistency results from (Bosq, 1996) for the NWRE as a function estimate and a plug-in predictor carry over directly (by application of Theorem 4) to the regularized SIRBFN with a.o.-selected regularization parameter. If the input process does not possess a stationary density, there are two possible solutions:

1. we may construct modified versions of the constructive approximation theorems in Section 2.1 which use condition (A.1) in place of the assumption of a stationary marginal input density. With this change, nonstationary input processes would be

admissible in the function estimation context, so long as the underlying regression function is fixed.

2. for the specific case of the prediction in the Markovian autoregressive model (4.1), it is known (e.g., see Theorem 3.4.10 of (Györfi et al., 1989)) that if f is bounded and $\{\epsilon(i)\}$ is an i.i.d. zero-mean, finite variance process with measure absolutely continuous with respect to Lebesgue measure, then both the scalar process $\{X(i)\}$ defined by (4.1) and the *equivalent vector process* $\{\boldsymbol{X}_d(i)\}$ defined by

$$\begin{aligned}\boldsymbol{X}_d(i) &= [f(\boldsymbol{X}_d(i-1)),\ X(i-1),\ X(i-2),\ldots,X(i-d+1)]^{\mathsf{T}} + \epsilon(i)\boldsymbol{e}_1 \\ &\triangleq T(\boldsymbol{X}_d(i-1)) + \epsilon(i)\boldsymbol{e}_1 \end{aligned} \tag{4.5}$$

where $\boldsymbol{e}_1 \triangleq [1,\ 0,\ldots,0]^{\mathsf{T}}$ is the first unit vector in $\mathbb{R}^d$, are GPM and therefore also GSM, since ϕ-mixing implies α-mixing. In other words, the dependence induced by the Markov autoregressive construction is compatible with the conditions of the approximation theorems of Section 2.1. To deal with the remaining issue of input process stationarity, we may impose conditions so that the autoregressive process is *geometrically ergodic*, i.e.,

$$|P_n(P_0) - \pi|_V = \mathcal{O}(\rho^n) \text{ for some } \rho \in (0,1) \tag{4.6}$$

where P_n is the (marginal) probability measure for $\boldsymbol{X}_d(n)$, $P_i(P_j)$ for $i \geq j$ denotes the probability measure for $\boldsymbol{X}_d(i)$ given that $\boldsymbol{X}_d(j)$ has probability measure P_j, and $\|\bullet\|_V$ denotes the total variation norm for the space $\mathcal{L}$ of probability measures over $\mathcal{B}\left(\mathbb{R}^d\right)$, i.e., for two probability measures $P, Q \in \mathcal{L}$,

$$\|P - Q\|_V \triangleq \sup_{B \in \mathcal{B}(\mathbb{R}^d)} |P(B) - Q(B)| \tag{4.7}$$

Here π denotes the *stationary* or *invariant* measure of the continuous $\mathbb{R}^d$-valued Markov chain (4.5). The idea of geometric ergodicity is that regardless of the distribution P_0 of the initial state $\boldsymbol{X}_d(0)$, the state of the Markov chain approaches stationarity exponentially fast. Hence, applying (4.6) with n sufficiently large, the previously developed approximation theorems for the stationary marginal input density case hold after replacing the common input measure P, density p, and cross-densities p_{ij} with the invariant measure π, density p_π, and cross-densities $p_{\pi,ij}$, respectively

(here $p_{\pi,ij}$ is defined as the limiting cross-density corresponding to the invariant joint measure between $Z(i)$ and $Z(j)$). An exposition on the conditions for the general continuous vector-valued version of (4.5) to be geometrically ergodic may be found in Chapter 4 and Appendix 1 of (Tong, 1990) and are summarized in Section 2.4.0.2 of (Douhkan, 1994). Adapting the results for the general case to the equivalent autoregressive models (4.1) and (4.5), we find that sufficient conditions are:

(a) $\{\epsilon(i)\}$ satisfies $\mathbb{E}[|\epsilon(i)|] < \infty$ and has an everywhere continuous and positive density with respect to Lebesgue measure

(b) f is Lipschitz (and hence bounded over bounded sets) in $\mathbb{R}^d$, has $f(\mathbf{0}) = 0$ (so that $T(\mathbf{0}) = \mathbf{0}$), and is *exponentially asymptotically stable in the large*, i.e., $\exists A, c > 0$ such that $\forall n \in \mathbb{Z}^+$ and $\boldsymbol{x}(0) \in \mathbb{R}^d$, $\|\boldsymbol{x}(n)\| \leq A \exp(-cn)\|\boldsymbol{x}(0)\|$, where $\boldsymbol{x}(n) \triangleq T^n(\boldsymbol{x}(0))$ is the n-fold composition of T applied to $\boldsymbol{x}(0)$.

Of the two conditions, the second is obviously the more restrictive one because it requires that the underlying mapping f satisfy a strong contractivity condition (although it does allow the stable point of the map T to be other than $\mathbf{0}$ by applying suitable translation). Exponential decay in transiently driven physical systems is quite plausible, however, which implies that the exponentially asymptotic stability condition may hold at least locally within a given time series. As an aside, we note that if the initial state r.v. $Z(0)$ satisfies $\|Z(0)\| \leq R$ for some constant $R > 0$, then the extension of Theorem 1 to non-compact sets (as given in Theorem 3) is not necessary, as the trajectory of $\{Z(i)\}$ is a.s. contained in closed ball about the origin of radius $A \cdot R$.

Since the Markovian autoregressive model (4.1) plays a central role in our speech prediction application, we shall follow the second approach in dealing with input process nonstationarity. In the following, we let $\boldsymbol{Z}(i) \triangleq \boldsymbol{X}_d(i)$ for all i.

The asymptotic stationarity implied by the geometric ergodicity of the autoregressive process also simplifies the extensions of Lemma 2 and hence Theorem 2 in Chapter 2 to the deal with the dependence between the prediction (or evaluation) point $\boldsymbol{Z} = \boldsymbol{X}_d(n)$ and the training set $T_n = \boldsymbol{Z}_n$ that occurs in the prediction of the Markovian autoregressive model (4.1). This dependence arises because after training, the last training datum $\boldsymbol{Z}(n) = \boldsymbol{X}_d(n)$ becomes the prediction input for the next process value $X(n+1)$, i.e., the

prediction is $\tilde{X}(n+1) \stackrel{\Delta}{=} \tilde{f}_n(\boldsymbol{X}_d(n))$. Here we indicate the necessary changes to Lemma 2 and Theorem 2 in this situation.

To maintain the notation of Lemma 2, let p and p_j be the marginal densities (with respect to Lebesgue measure) of $\boldsymbol{Z}$ and $\boldsymbol{Z}(j)$ with supports S and S_j, respectively. Then (2.38) may be equivalently written as

$$\begin{aligned} q_j(\epsilon) &\stackrel{\Delta}{=} P_{\boldsymbol{Z},\boldsymbol{Z}(j)}(A_{n,j}) \\ &= \int_{\boldsymbol{x}\in S_j} P_{\boldsymbol{Z}}\{\boldsymbol{z}\in S : \|\boldsymbol{z}-\boldsymbol{x}\| > \epsilon\}\ p_j(\boldsymbol{x})d\boldsymbol{x} \\ &= \int_{\boldsymbol{x}\in S_j\Delta S} P_{\boldsymbol{Z}}\{\boldsymbol{z}\in S : \|\boldsymbol{z}-\boldsymbol{x}\| > \epsilon\}\ p_j(\boldsymbol{x})d\boldsymbol{x} \\ &\quad + \int_{\boldsymbol{x}\in S_j\cap S} P_{\boldsymbol{Z}}\{\boldsymbol{z}\in S : \|\boldsymbol{z}-\boldsymbol{x}\| > \epsilon\}\ p_j(\boldsymbol{x})d\boldsymbol{x} \end{aligned} \tag{4.8}$$

By the geometric ergodicity condition (4.6), the first integral vanishes for j sufficiently large (since $P_{\boldsymbol{Z}(j)}(\boldsymbol{z}(j)\in S_j\Delta S) \stackrel{j\to\infty}{\to} 0$ by the triangle-inequality $\left\|P_{\boldsymbol{Z}} - P_{\boldsymbol{Z}(j)}\right\|_V \le \|P_{\boldsymbol{Z}}-\pi\|_V + \left\|\pi - P_{\boldsymbol{Z}(j)}\right\|_V$, where π is the stationary measure for $\{\boldsymbol{Z}(i)\}$). The second integral can be expressed as

$$\begin{aligned} &\int_{\boldsymbol{x}\in S_j\cap S} P_{\boldsymbol{Z}}\{\boldsymbol{z}\in S : \|\boldsymbol{z}-\boldsymbol{x}\| > \epsilon\}\ p_j(\boldsymbol{x})d\boldsymbol{x} \\ &= \int_{\boldsymbol{x}\in S_j\cap S} (1 - P_{\boldsymbol{Z}}\{\boldsymbol{z}\in S : \|\boldsymbol{z}-\boldsymbol{x}\| < \epsilon\})\ p_j(\boldsymbol{x})d\boldsymbol{x} \\ &= P_{\boldsymbol{Z}(j)}(\boldsymbol{z}(j)\in S_j\cap S) - \int_{\boldsymbol{x}\in S_j\cap S} P_{\boldsymbol{Z}}\{\boldsymbol{z}\in S : \|\boldsymbol{z}-\boldsymbol{x}\| < \epsilon\}\ p_j(\boldsymbol{x})d\boldsymbol{x} \end{aligned} \tag{4.9}$$

in which the first term is no greater than unity while the strict positivity of the second term given $\epsilon > 0$ follows from that of the integrand, as argued in the conclusion of the proof for Lemma 2. The desired convergence in probability then follows.

Except in the application of Lemma 2 and the associated definition of $q(\epsilon)$, Theorem 2 does not assume the independence of $\boldsymbol{Z}$ from T_n and, accordingly, the changes required to have it hold for the prediction of the Markovian autoregressive model (4.1) are relatively minor:

- condition (2.40) on the common marginal density p for $\{\boldsymbol{Z}(i)\}$ is replaced with

$$\sup_{j\in\mathbf{N}}\sup_{\boldsymbol{z}\in\mathbf{R}^d} |p_j(\boldsymbol{z})| < L_p \tag{4.10}$$

  (where p_j is the marginal density for $\boldsymbol{Z}(j)$). This modified condition is equivalent to the original condition when the density p_π for the stationary measure π (assumed

absolutely continuous with respect to Lebesgue measure) is bounded since (4.6) implies the a.e. pointwise-convergence of p_j to p_π as $j \to \infty$ (by choosing B to be a point set). Thus p_j can be bounded either by the bound on p_π, when $j > N$ for some $N \in \mathbb{N}$ sufficiently large, or by $\sup_{j=1,2,\ldots,N} \sup_{z \in \mathbb{R}^d} |p_j(z)|$ for $1 \leq j \leq N$.

- condition (2.45) on n is replaced with

$$\text{for } n \text{ such that } \log\left(\frac{\delta}{L_v}\right) / \sum_{j=1}^{n} \log q_j(\epsilon) < 1/2 \tag{4.11}$$

- condition (2.49) on $q(\epsilon)$ is replaced with

$$q_i(\epsilon) = 1 - P_{\boldsymbol{Z}, \boldsymbol{Z}(i)}\left(B_{n,i}(\epsilon)\right) \geq 1 - L_p(2\epsilon)^d, \quad i = 1, 2, \ldots, n \tag{4.12}$$

- in (2.52), replace $q^{n/2}(\epsilon)$ with $\prod_{j=1}^{n} q_j^{1/2}(\epsilon)$.

With these amendments, Theorem 2 follows as before.

4.5 Recursive Updating for Prediction using Regularized SIRBF Networks

As there is no substantial difficulty in doing so, we shall, where possible, develop the subsequent algorithms for a general pair of input/output processes $\{\boldsymbol{Z}(i), Y(i)\}$ rather than specifically for the autoregressive case $Y(i) \triangleq X(i)$ and $\boldsymbol{Z}(i) \triangleq \boldsymbol{X}_d(i-1)$. Thus far, both the NWRE and regularized SIRBF network assume that the process to be predicted admits a time-invariant regression function; in practice, as our speech prediction experiment will show, this condition does not always hold. If the regression function f drifts slowly with time index i as f_i, i.e., exhibits a form of *local stationarity*, the idea of updating the regression function parameters periodically, say every l time steps, as new data arrive is intuitively appealing, particularly when it can be performed *efficiently* in a *recursive* fashion. The basis of comparison will be the standard adaptive linear estimation procedures such as the recursive least-squares (RLS) algorithm. Let us consider the limiting case $l = 1$ and assume for now that n, the size of the training set and hence the number of basis functions in the estimate for f_i, is fixed. Before continuing, let us set the notations for the following discussion:

subscripts: for *vector* and (square) *matrix quantities*, the *first subscript* refers to its *dimension*, while for a *scalar quantity*, it refers to the dimension of the associated vector or matrix quantity being indexed. The *second subscript*, if present, refers to *either* the *time index* of the training set from which the quantity is constructed (in the case of a *scalar* or *vector function*) or a *particular element* of that quantity (in the case of an *ordinary vector*).

parenthesized arguments: for *nonfunctional quantities*, a parenthesized argument indicates *time dependence*, i.e., $\bullet(i)$ mean quantity $\bullet$ uses data up to and including time step i. For functions, it indicates the usual argument.

As an example, $t_n(i) \triangleq \{(x(j), y(j))\}_{j=i-n+1}^{i}$ denotes the realized training set for the network at time step i, where in the NLAR case, this training set is formed from the time series segment $\{x(j)\}_{j=i-n-d+1}^{i}$. Then $g_{n,i}(\bullet)$ corresponds to $g(\bullet)$ in (1.41) and $w_{n,j}(i)$ corresponds to the j-th element of $\boldsymbol{w}$ in (1.15) when $t_n(i)$ is used in place of t_n.

Given $t_n(i)$, a realized set of input/output examples for f_i, and $\widetilde{f}_{n,i}$ the corresponding regression function estimate, the problem is to recursively compute $\widetilde{f}_{n,i+1}$, the estimate associated with $t_n(i+1)$, from $\widetilde{f}_{n,i}$. For the NWRE, this network updating and subsequent prediction are simple, as shown in Table 4.1.

If we are using some data-based method of selecting the bandwidth, it may also be advantageous to adjust the bandwidth from $h_n = h_n(i)$ to $h_n(i+1)$ at the same time. The basic order of the updating, excluding the cost of computing an updated bandwidth parameter, for the NWRE is $\mathcal{O}(1)$ and that of computing the prediction $\widetilde{y}(i+1)$ is $\mathcal{O}(n)$.

For the regularized SIRBFN, we shall analyse the effect of the one-step updating in two stages and thereby find interesting parallels to the standard RLS estimation algorithm. In the first stage, we allow the size of the SIRBFN to grow with incoming data so that one weight is added per update, leading to an *augmented* network with *infinite memory* (cf. for linear adaptive filters, this growth is usually called *order recursion*, e.g., see Chapter 15 of (Haykin, 1996)). The second stage is to simultaneously add one (new) weight and truncate the oldest weight per update, leading to a network of *fixed size* with *finite memory*.

Initialization: assume the NWRE has been generated from $t_n(i)$ in the usual way, i.e.,

$$\widetilde{f}_{n,i}(\bullet) = \frac{\sum_{j=0}^{n-1} y(i-j) K\left(\frac{\bullet - z(i-j)}{h_n}\right)}{\sum_{j=0}^{n-1} K\left(\frac{\bullet - z(i-j)}{h_n}\right)} \tag{4.13}$$

Updating: when the new datum $(z(i+1), y(i+1))$ becomes available,

1. replace the basis function $K\left(\|\bullet - z(i-n+1)\|/h_n\right)$ with $K\left(\|\bullet - z(i+1)\|/h_n\right)$ in (4.13).
2. replace the corresponding prediction target $y(i-n+1)$ with $y(i+1)$ in (4.13).

Prediction: for the NLAR case $y(i) \triangleq x(i)$ and $z(i) \triangleq \boldsymbol{x}_d(i-1)$, set $\tilde{x}(i+2) = \widetilde{f}_{n,i+1}\left(\boldsymbol{x}_d(i+1)\right)$.

Iteration: $i \rightarrow i+1$ and repeat from Updating step.

Table 4.1: NWRE basic fixed-size prediction update algorithm

4.5.1 Augmented (Infinite Memory) Case

We begin by decomposing the $(n+1) \times (n+1)$ regularized SI equation for the *combined* realized training set $t_{n+1}(i+1) = t_n(i) \cup t_n(i+1)$ as

$$\left(\begin{bmatrix} \boldsymbol{G}_n(i) & \boldsymbol{\gamma}_n(i+1) \\ \boldsymbol{\gamma}_n^{\mathsf{T}}(i+1) & K(0) \end{bmatrix} + \begin{bmatrix} \boldsymbol{\Lambda}_n(i) & \boldsymbol{0} \\ \boldsymbol{0}^{\mathsf{T}} & \lambda_{n+1}(i+1) \end{bmatrix}\right) \cdot \left(\begin{bmatrix} \boldsymbol{w}_n(i) \\ 0 \end{bmatrix} + \begin{bmatrix} \Delta \boldsymbol{w}_n(i) \\ w_{n+1,n+1}(i+1) \end{bmatrix}\right) = \begin{bmatrix} \boldsymbol{y}_n(i) \\ y(i+1) \end{bmatrix} \tag{4.14}$$

which we may write more compactly as

$$\begin{bmatrix} \boldsymbol{F}_n(i) & \boldsymbol{\gamma}_n(i+1) \\ \boldsymbol{\gamma}_n^{\mathsf{T}}(i+1) & K(0) + \lambda_{n+1}(i+1) \end{bmatrix} \left(\begin{bmatrix} \boldsymbol{w}_n(i) \\ 0 \end{bmatrix} + \begin{bmatrix} \Delta \boldsymbol{w}_n(i) \\ w_{n+1,n+1}(i+1) \end{bmatrix}\right) = \begin{bmatrix} \boldsymbol{y}_n(i) \\ y(i+1) \end{bmatrix}$$

$$\boldsymbol{F}_{n+1}(i+1) \cdot \boldsymbol{w}_{n+1}(i+1) = \boldsymbol{y}_{n+1}(i+1) \tag{4.15}$$

where $\boldsymbol{F}_n(i) \triangleq \boldsymbol{G}_n(i) + \boldsymbol{\Lambda}_n(i)$ and $\boldsymbol{\gamma}_n(i)$ is the vector formed from the first n elements of the last column of $\boldsymbol{G}_{n+1}(i)$, i.e., $\boldsymbol{\gamma}_n(i) \triangleq \left[\boldsymbol{g}_{n,i}(\boldsymbol{z}(i))\right]_{1:n}$ (the notation $i{:}j$ means indices i to j

inclusive). Here, as a slight generalization, $\mathbf{\Lambda}_n(i) \triangleq \text{diag}\,(\lambda_n(i-j), j = n-1, n-2, \ldots, 0)$ is the diagonal weighting matrix formed from the most recent n regularization parameters up to and including time step i. Let $\boldsymbol{w}_n(i)$ be the previously computed solution to the regularized SI equation $(\boldsymbol{G}_n(i) + \mathbf{\Lambda}_n(i))\,\boldsymbol{w}_n(i) = \boldsymbol{y}_n(i)$ over $t_n(i)$. We assume that the new regularization parameter $\lambda_{n+1}(i+1)$ has been chosen on the basis of $t_{n+1}(i+1)$. The objective is to find the new weight $w_{n+1,n+1}(i+1)$ and the weight change vector $\Delta\boldsymbol{w}_n(i)$ to be applied to $\boldsymbol{w}_n(i)$, such that the *augmented* regularized SI equation (4.14) is satisfied. The solution is

$$w_{n+1,n+1}(i+1) = \frac{y(i+1) - (\boldsymbol{w}_n(i) + \Delta\boldsymbol{w}_n(i))^{\mathsf{T}}\,\boldsymbol{\gamma}_n(i+1)}{\lambda_{n+1}(i+1) + K(0)} \tag{4.16}$$

$$\Delta\boldsymbol{w}_n(i) = -\left(\boldsymbol{F}_n(i) - \frac{\boldsymbol{\gamma}_n(i+1)\boldsymbol{\gamma}_n^{\mathsf{T}}(i+1)}{\lambda_{n+1}(i+1) + K(0)}\right)^{-1} \cdot \frac{\boldsymbol{\gamma}_n(i+1)}{\lambda_{n+1}(i+1) + K(0)}\left(y(i+1) - \boldsymbol{w}_n^{\mathsf{T}}(i)\boldsymbol{\gamma}_n(i+1)\right) \tag{4.17}$$

The resultant prediction update algorithm is listed in Table 4.2. Because $\boldsymbol{\gamma}_n(i+1)$ is also the vector of basis function outputs of the previous network from time step i in response to the newly available input $\boldsymbol{z}(i+1)$, we see that the new weight $w_{n+1,n+1}(i+1)$ is merely a scaled version of the *a posteriori* estimation error, i.e., the estimation error that would have been obtained had the previous weight vector $\boldsymbol{w}_n(i)$ been updated to $\boldsymbol{w}_n(i) + \Delta\boldsymbol{w}_n(i)$. In contrast, the weight change vector $\Delta\boldsymbol{w}_n(i)$ is proportional to the *a priori* estimation error, i.e., the actual estimation error using the previous weight vector $\boldsymbol{w}_n(i)$ prior to any updating, similar to what occurs in the RLS algorithm. This partitioning of roles between $w_{n+1,n+1}(i+1)$ and $\Delta\boldsymbol{w}_n(i)$ is intuitively satisfying: the change $\Delta\boldsymbol{w}_n(i)$ applied to the existing weight vector attempts to account for estimation error incurred by the existing (non-updated) network, while the new weight element $w_{n+1,n+1}(i+1)$ attempts to account for the estimation error remaining after the existing network has been updated. Analogous to the RLS algorithm, we may also expect the ratio of the m.s. *a priori* and the m.s. *a posteriori* estimation errors to converge to unity as $n \to \infty$ if the regression function being estimated is not significantly time-varying. If the ratio is nonconvergent, it may be an indication that old training samples are no longer representative of the regression function behaviour currently being estimated. For this situation, the effective *memory* of the SIRBF network can be limited by fixing its size to n weights/basis functions computed from the

Initialization: assume the regularized SIRBFN has been generated from $t_n(i)$ in the usual way, i.e., via equations (1.1) and (1.15) with $t_n(i)$ in place of T_N, and assume that $F_n^{-1}(i)$ is known.

Updating: when the new datum $(\boldsymbol{z}(i+1), y(i+1))$ becomes available,

1. select the new regularization parameter $\lambda_{n+1}(i+1)$ and the norm weighting matrix $\boldsymbol{U}_{n+1}(i+1)$, typically from $t_{n+1}(i+1)$.
2. compute the new basis function vector $\boldsymbol{\gamma}_n(i+1)$.
3. compute $\left(\boldsymbol{F}_n(i) - \frac{\boldsymbol{\gamma}_n(i+1)\boldsymbol{\gamma}_n^T(i+1)}{\lambda_{n+1}(i+1)+K(0)}\right)^{-1}$. Note the complexity of this calculation may be reduced to $\mathcal{O}(n^2)$ if $F_n^{-1}(i)$ is optionally propagated from time step to time step as indicated below, since the Sherman-Woodbury-Morrison formula (Hager, 1989) (or *matrix inversion lemma* in the statistical signal processing field (Haykin, 1996)) for the inverse of the sum of a given matrix and a low rank perturbation may be applied.
4. compute the weight change vector $\Delta \boldsymbol{w}_n(i)$ according to (4.17).
5. add the weight change vector $\Delta \boldsymbol{w}_n(i)$ to the existing network weight vector.
6. compute the new weight $w_{n+1,n+1}(i+1)$ via (4.17).
7. add the new basis function $K\left(\|\bullet - \boldsymbol{z}(i+1)\|_{\boldsymbol{U}_{n+1}(i+1)}\right)$ with weight $w_{n+1,n+1}(i+1)$ to the network
8. (optional) compute $\boldsymbol{F}_{n+1}^{-1}(i+1)$ from $\boldsymbol{F}_n^{-1}(i)$ with complexity $\mathcal{O}(n^2)$ via a partitioned matrix inverse formula applied to the decomposition (4.15).

Prediction: for the NLAR case $y(i) \triangleq x(i)$ and $\boldsymbol{z}(i) \triangleq \boldsymbol{x}_d(i-1)$, set $\tilde{x}(i+2) = \tilde{f}_{n+1,i+1}(\boldsymbol{x}_d(i+1))$.

Iteration: $i \to i+1$, $n \to n+1$ and repeat from Updating step.

Table 4.2: Regularized SIRBFN augmented prediction update algorithm

most recent n training data available, which leads us to the second stage of updating described next.

4.5.2 Fixed-Size (Finite Memory) Case

Let us return to the original task and assume that the size of the SIRBF network is fixed at n weights/basis functions. The desire is to relate $\boldsymbol{w}_n(i+1)$, the weights satisfying the regularized SI equation over $t_n(i+1)$, to the previously computed weights $\boldsymbol{w}_n(i)$ which do the same for $t_n(i)$. Before we do so, let us establish the notations. Decompose the $n \times n$ regularized SI equation for the previous training set $t_n(i)$ as

$$\begin{bmatrix} \lambda_n(i-n+1)+K(0) & \boldsymbol{\beta}_{n-1}^{\mathsf{T}}(i) \\ \boldsymbol{\beta}_{n-1}(i) & \boldsymbol{F}_{n-1}(i) \end{bmatrix} \begin{bmatrix} w_{n,1}(i) \\ \boldsymbol{w}_{n,2:n}(i) \end{bmatrix} = \begin{bmatrix} y(i-n+1) \\ \boldsymbol{y}_{n,2:n}(i) \end{bmatrix}$$

$$\boldsymbol{F}_n(i) \cdot \boldsymbol{w}_n(i) = \boldsymbol{y}_n(i) \tag{4.18}$$

where $\boldsymbol{\beta}_{n-1}(i)$ is the vector of the last $n-1$ elements of the first column of the previous interpolation matrix $\boldsymbol{G}_n(i)$, i.e., $\boldsymbol{\beta}_{n-1}(i) \triangleq \left[g_{n,1}(z(i-n+1))\right]_{2:n}$. This time, the objective is to find $\Delta\boldsymbol{w}_{n,2:n}(i)$ and $w_{n,n}(i+1)$ satisfying

$$\begin{bmatrix} \boldsymbol{F}_{n-1}(i) & \boldsymbol{\gamma}_{n-1}(i+1) \\ \boldsymbol{\gamma}_{n-1}^{\mathsf{T}}(i+1) & \lambda_n(i+1)+K(0) \end{bmatrix} \left(\begin{bmatrix} \boldsymbol{w}_{n,2:n}(i) \\ 0 \end{bmatrix} + \begin{bmatrix} \Delta\boldsymbol{w}_{n,2:n}(i) \\ w_{n,n}(i+1) \end{bmatrix} \right) = \begin{bmatrix} \boldsymbol{y}_{n,2:n}(i) \\ y(i+1) \end{bmatrix}$$

$$\boldsymbol{F}_n(i+1) \cdot \boldsymbol{w}_n(i+1) = \boldsymbol{y}_n(i+1) \tag{4.19}$$

In other words, the new weight vector for the updated network can be considered the result of

1. shifting the last $n-1$ weights in the old weight vector $\boldsymbol{w}_n(i)$ which are associated with the most recent $n-1$ data in $t_n(i)$ upwards into positions 1 to $n-1$ and setting the n-th element to zero.

2. adding a perturbation $\Delta\boldsymbol{w}_{n,2:n}(i)$ to the shifted vector.

3. adding a new weight $w_{n,n}(i+1)$ in the n-th position.

It is not difficult to show that the resultant update equations become

$$w_{n,n}(i+1) = \frac{y(i+1) - (\boldsymbol{w}_{n,2:n}(i) + \Delta\boldsymbol{w}_{n,2:n}(i))^{\top}\boldsymbol{\gamma}_{n-1}(i+1)}{\lambda_n(i+1) + K(0)} \tag{4.20}$$

$$\Delta\boldsymbol{w}_{n,2:n}(i) = \left(\boldsymbol{F}_{n-1}(i) - \frac{\boldsymbol{\gamma}_{n-1}(i+1)\boldsymbol{\gamma}_{n-1}^{\top}(i+1)}{\lambda_n(i+1) + K(0)}\right)^{-1} \cdot \left[w_{n,1}(i)\boldsymbol{\beta}_{n-1}(i) - \frac{\boldsymbol{\gamma}_{n-1}(i+1)}{\lambda_n(i+1) + K(0)}\left(y(i+1) - \boldsymbol{w}_{n,2:n}^{\top}(i)\boldsymbol{\gamma}_{n-1}(i+1)\right)\right] \tag{4.21}$$

Except for the additional term $w_{n,1}(i)\boldsymbol{\beta}_{n-1}(i)$ in (4.21), the forms of the update equations for this fixed-size case are identical to those for the augmented case. The additional term can be regarded as embodying the effect of weight vector augmentation from size n to $n+1$, followed by truncation to the weights computed from the most recent n training data. We summarize the prediction update algorithm for the fixed-size case in Table 4.3. Note that the formula for $\boldsymbol{F}_{n-1}^{-1}(i+1)$ in Updating step 9 follows from the identity

$$\begin{bmatrix} 1 & \mathbf{0}^{\top} \\ \mathbf{0} & \boldsymbol{F}_{n-1}^{-1}(i+1) \end{bmatrix} = \boldsymbol{F}_n^{-1}(i+1) \begin{bmatrix} \lambda_n(i-n+2) + K(0) & \boldsymbol{\beta}_n^{\top}(i+1)\boldsymbol{F}_{n-1}^{-1}(i+1) \\ \boldsymbol{\beta}_n(i+1) & \boldsymbol{I}_n \end{bmatrix} \tag{4.22}$$

Although the parallels between the recursive update algorithms described here and those in the RLS algorithm are interesting in their own right, one must be careful not to conclude that the algorithms presented are merely expressions of the RLS algorithm after a nonlinear mapping $\boldsymbol{z}(i) \in \mathbb{R}^d \mapsto \boldsymbol{g}_{n,i-1}(\boldsymbol{z}(i)) \in \mathbb{R}^n$. We can see this difference clearly in the fact that infinite memory regularized SIRBF networks require an infinite number of weights/basis functions; fixed-size regularized SIRBF networks can only have a finite memory of the same size. This condition stands in contrast to the situation with the RLS filter where a fixed number of weights are updated to reflect all the past history of the input data. Of course, the exponentially-weighted variant of the RLS algorithm is commonly used in practice and one can argue that its memory is, for all practical purposes, limited. Indeed, the introduction of the exponentially-weighted variant of the RLS algorithm was motivated by the heuristic that decaying memory would improve estimation when the input/output processes are nonstationary, although it has now been established that this notion is incorrect (Haykin et al., 1997). In this respect, the fixed size regularized SIRBF network is somewhat more explicit in the way it deals with nonstationarity.

With both the augmented and fixed-size update algorithms, their computational efficiency is derived from the low rank of the perturbation applied to the existing interpo-

Initialization: assume the regularized SIRBFN has been generated from $t_n(i)$ in the usual way, i.e., via equations (1.1) and (1.15) with $t_n(i)$ in place of T_N, and assume that $\boldsymbol{F}_{n-1}^{-1}(i)$ is known.

Updating: when the new datum $(\boldsymbol{z}(i+1), y(i+1))$ becomes available,

1. select the new regularization parameter $\lambda_n(i+1)$ and the norm weighting matrix $\boldsymbol{U}_n(i+1)$, typically from $t_n(i+1)$.
2. compute the new basis function vector $\boldsymbol{\gamma}_{n-1}(i+1)$.
3. compute $\left(\boldsymbol{F}_{n-1}(i) - \frac{\boldsymbol{\gamma}_{n-1}(i+1)\boldsymbol{\gamma}_{n-1}^{\mathsf{T}}(i+1)}{\lambda_{n+1}(i+1)+K(0)}\right)^{-1}$. Complexity can be reduced to $\mathcal{O}(n^2)$ if $\boldsymbol{F}_{n-1}^{-1}(i)$ is optionally propagated from time step to time step as in step 3 of the Updating procedure in Table 4.2.
4. compute the shifted weight change vector $\Delta \boldsymbol{w}_{n,2:n}(i)$ and the new weight $w_{n,n}(i+1)$ according to (4.21).
5. delete the basis function $K\left(\|\bullet - \boldsymbol{z}(i-n+1)\|)_{\boldsymbol{U}_n(i)}\right)$ and its weight $w_{n,1}$ associated with the oldest data in $t_n(i)$ from the network.
6. add the shifted weight change vector $\Delta \boldsymbol{w}_n(i)$ to the remaining $n-1$ network weights.
7. add the new basis function $K\left(\|\bullet - \boldsymbol{z}(i+1)\|_{\boldsymbol{U}_n(i+1)}\right)$ with weight $w_{n,n}(i+1)$ to the network
8. (optional) compute $\boldsymbol{F}_n^{-1}(i+1)$ from $\boldsymbol{F}_{n-1}^{-1}(i)$ with complexity $\mathcal{O}(n^2)$ via a partitioned matrix inverse formula applied to the decomposition (4.18). Hence compute $\boldsymbol{F}_{n-1}^{-1}(i+1)$ with complexity $\mathcal{O}(n^2)$ via $\boldsymbol{F}_{n-1}^{-1}(i+1) = \left(\boldsymbol{I} - \boldsymbol{h}_{n-1}(i+1)\boldsymbol{\beta}_{n-1}^{\mathsf{T}}(i+1)\right)^{-1} \boldsymbol{H}_{n-1}(i+1)$ where $\boldsymbol{h}_{n-1}(i+1)$ is the vector formed from the last $n-1$ elements of the first column of $\boldsymbol{F}_n^{-1}(i+1)$ and $\boldsymbol{H}_{n-1}(i+1)$ is the lower right $(n-1)\times(n-1)$ submatrix of $\boldsymbol{F}_n^{-1}(i+1)$.

Prediction: for the NLAR case $y(i) \triangleq \boldsymbol{x}(i)$ and $\boldsymbol{z}(i) \triangleq \boldsymbol{x}_d(i-1)$, set $\tilde{\boldsymbol{x}}(i+2) = \tilde{f}_{n,i+1}\left(\boldsymbol{x}_d(i+1)\right)$.

Iteration: $i \to i+1$ and repeat from Updating step.

Table 4.3: Regularized SIRBFN fixed-size prediction update algorithm

lation matrix at a given time step through augmentation and addition, respectively. As we shall see in the experimental results for speech prediction, in some cases the nonstationarity of the input process is sufficient to require bandwidth and/or regularization parameter updates to *all* entries of the regularized interpolation matrix $\boldsymbol{F}_n(i)$, not just those in the new basis function vectors $\boldsymbol{\gamma}_n(i+1)$ (in the case of the augmented updates) and $\boldsymbol{\gamma}_{n-1}(i+1)$ (in the case of the fixed-size updates). Nevertheless, the recursive update algorithms for both cases provide useful insight into the essential character and operation of the dynamic regularized SIRBF network as a time series estimator.

4.6 Application to Speech Prediction

For a benchmark problem with real-world data, we turn to speech prediction. That the human speech signal is generally nonlinear and nonstationary is well-known; even so, the linear prediction of speech with *analytic* methods such as the LMS/RLS/Kalman algorithms (Haykin, 1996) and *synthetic* methods such as CELP (CELP, 1985) has been met with surprising success. Of course, these results are achieved after significant prior knowledge regarding the characteristics of human speech have been carefully embedded into the corresponding methods to realize maximum performance. In contrast, we should emphasize that our interest in speech as the test signal for the proposed algorithms is limited to the characterization of the gains possible from nonlinear and nonstationary processing, and should not be taken to imply that the proposed predictors (in their current form) are either practical or optimally tuned for actual speech prediction applications such as speech coding. Further speech-specific research and evaluation would clearly be necessary to reach that state. That said, our objective in this set of experiments is to demonstrate that even without significant tuning, the dynamic regularized SIRBF network can provide a nontrivial improvement in prediction SNR over the standard LMS/RLS algorithm-based predictors. We also indicate the further improvement possible in exploiting the residual correlations between the predictions of several dynamic regularized SIRBF networks and the predicted speech signal by way of an additional stage of RLS estimation. Taken jointly, these results offer evidence of the performance gains possible when the nonlinearity and nonstationarity of speech signal are addressed.

4.6.1 Description of Speech Data

The speech data to be predicted consist of samples from ten different male and ten different female speakers, each reading a distinct phonetically-balanced sentence. In their original format, the continuous speech signals were 16-bit linear PCM sampled at 16kHz rate with 8kHz bandwidth. These samples were subsequently filtered by a third-order Butterworth filter with a cutoff frequency of 3.2kHz, decimated to 8kHz rate, and re-centred to zero-mean. Both the original and final speech signals are of high quality with little discernible background noise. The sentence samples and some of their key characteristics as discrete time series are summarized in Tables 4.4 and 4.5. As can be seen from these tables, the total length of a speech signal being tested varies from approximately 2.5 to 4 seconds. Before beginning, it is useful to quantify the degree of nonlinearity in the speech

ID	No. of samples	Sentence
m130	20215	Type out three lists of orders.
m131	23525	The harder he tried, the less he got done.
m132	25128	The boss ran the show with a watchful eye.
m133	25129	The cup cracked and spilled its contents.
m134	23260	Paste can cleanse the most dirty brass.
m135	29073	The slang word for raw whiskey is booze.
m136	27746	It caught its hind paw in a rusty trap.
m137	26724	The wharf could be seen at the farthest shore.
m138	22435	Feel the heat of the weak dying flame?
m139	20877	The tiny girl took off her hat.

Table 4.4: Male speech sample parameters

samples, as this factor will ultimately determine the gains possible in our approach. Using some software for chaotic time series analysis developed by the chemical reactor engineering group at Delft University of Technology in the Netherlands (Schouten and Bleek, 1994), the method of surrogate data analysis (Theiler et al., 1992) with a Mann-Whitney rank-sum test rejects the null hypothesis that each speech sample is that of a stationary linear process with a maximum sample Z-statistic of less than -13 in each case (a Z-statistic of less than

ID	No. of samples	Sentence
f150	21530	The young kid jumped the rusty gate.
f151	25471	Guess the results from the first scores.
f152	25064	A salt pickle taste fine with ham.
f153	24674	The just claim got the right verdict.
f154	24361	These thistles bend in a high wind.
f155	22098	Pure bred poodles have curls.
f156	26954	The tree top waved in a graceful way.
f157	32522	The spot on the blotter was made by green ink.
f158	27220	Mud was spattered on the front of his white shirt.
f159	23306	The cigar burned a hole in the desk top.

Table 4.5: Female speech sample parameters

-3 is considered grounds for strong rejection). This result indicates that significant benefit from nonlinear processing should be possible.

4.6.2 Approach using Regularized SIRBFN

The particular approach taken is to treat each speech sample as a realization of a discrete-time Markov process obeying (4.1) with order $d = p$. For one-step-ahead (OSA) prediction, we consider the limiting case of per time step updating, i.e., $l = 1$. Key design issues to consider are:

input order: preliminary experiments showed that, for a given speech sample, the prediction performance of the dynamic regularized SIRBF network varied with the order p depending on the local characteristics of the speech over which the network was operating. For example, in the transition periods between voiced, unvoiced, and silent segments, networks with small p, e.g., $p = 10$, were generally found to perform better than those with large p, e.g., $p = 50$. Conversely, within a given type of speech segment, the networks with larger p tended to be the better predictors. While techniques for estimating the order of NLAR processes have been recently proposed (Auestad and Tjøstheim, 1990), for computational simplicity three fixed-sized networks with $p = 10$,

30, and 50 are run in parallel for each speech sample and, as we shall see later, linearly combined.

SIRBFN parameters: based on some previous work (Yee and Haykin, 1995), each of the networks is chosen to have:

network size: A fixed-size of $n = 100$ basis functions is used. This fixed-size corresponds to the assumption that a useful memory for the networks is 12.5ms, the average length of the 5-20ms window of stationarity usually associated with speech.

basis function: the "smooth" (in the sense of satisfying (1.12) with the operator defined in (1.13)) Gaussian basis function $K(r) = \exp(-r^2/2)$ is used.

norm weighting matrix: common to all basis functions is a diagonal norm weighting matrix $U_n(i)$ whose inverse, at time step i, is set to p times the diagonal of the empirical covariance matrix for the input samples in $t_n(i)$. This particular form of the norm weighting matrix allows the multidimensional network basis function to be decomposed into a p-fold product of one dimensional (Gaussian) kernels, each with bandwidth parameter equal to the variance of a particular input variable over $t_n(i)$. In the one-dimensional i.i.d. density estimation setting, such a form of bandwidth has been shown to be consistent in the L_1 sense (Devroye and Györfi, 1985).

regularization parameter: for each of the three networks ($p = 10$, 30, and 50), the regularization parameter for each time step is selected as the value which minimizes the GCV criterion function evaluated over 1000 logarithmically spaced points from λ_{min} to λ_{max} for that network as given in Table 4.6. Since the speech signals are largely noise-free, the upper bound on $\lambda_n(i+1)$ prevents undue over-regularization while the lower bound is necessary to ensure the numerical nonsingularity of the regularized SI matrix at each time step. The slight differences in the evaluation limits account for the varying degrees of sensitivity of each network to these two conditions.

update algorithm: because the norm weighting matrix $U_n(i)$ is updated for *all* network basis functions when new data arrive, the update from $F_n(i)$ to $F_n(i+1)$ is full-rank and hence (4.19) must be solved directly without using the recursion aids (4.20) and (4.21). It was found in previous experiments (Yee and Haykin, 1995) that the

speech samples were sufficiently nonstationary so that without careful choice of the update parameters indicated in the first Updating step, the recursively updated fixed-size network outputs would frequently loose track of the speech samples within an order of n time steps from the last full-rank update. More recent work indicates that this problem can be effectively alleviated by *resetting* the network, i.e., performing a full update, when a large-deviation in the prediction error is detected (Yee and Haykin, 1998). Nonetheless, the issue of how best to select the update parameters in the recursive fixed-size update algorithm so as to minimize performance loss from partial updating remains an open question.

Network no.	p	λ_{min}	λ_{max}
1	50	0.0001	0.001
2	30	0.00001	0.01
3	10	0.0001	0.01

Table 4.6: GCV criterion function evaluation limits

4.6.3 Comparison to Linear RLS Algorithm and Previous Work

The performance measure we shall use is the *predicted signal-to-noise ratio (PSNR)* defined for an actual or input signal sequence $\{y(i)\}_{i=1}^{N}$ by

$$\text{PSNR (dB)} \triangleq 10 \log_{10} \left(\widetilde{\sigma_y^2} / \widetilde{\sigma_\epsilon^2} \right) \tag{4.23}$$

where $\widetilde{\sigma_y^2}$ and $\widetilde{\sigma_\epsilon^2}$ are the actual and error signal *powers* estimated by

$$\widetilde{\sigma_y^2} \triangleq \frac{1}{N} \sum_{i=1}^{N} y^2(i), \qquad \widetilde{\sigma_\epsilon^2} \triangleq \frac{1}{N} \sum_{i=1}^{N} \epsilon^2(i) \text{ where } \epsilon(i) \triangleq y(i) - \tilde{y}(i) \tag{4.24}$$

and $\tilde{y}(i)$ is the network prediction for actual signal $y(i)$. In our case of zero-mean input signals, because we use estimated signal powers rather than estimated signal variances as is sometimes the case in defining the PSNR, the following performance figures are somewhat conservative (for example, a non-zero mean level of error will degrade performance by the former definition but not by the latter definition). In ANN terms, the PSNR can be considered a measure of the *generalization* performance of the dynamic network, since in

our NLAR case, each prediction $\tilde{y}(i) = \tilde{x}(i+1)$ at time step i is for the first time series point *outside* the window $\{x(j)\}_{j=i-(n+p-1)}^{i}$ of data effectively seen during training ($n+p$ sequential data are needed to form $t_n(i)$). This effective training window, along with the predicted point, shift forward in time as the dynamic network advances through the entire input signal sequence. Although, strictly speaking, the test set per time step is a single (out-of-sample) point, by iterating the training/prediction cycle over the available input time series (the number of samples in each speech signal listed in Tables 4.4 and 4.5 less $n+p$ samples for initialization), this pointwise prediction performance can be averaged to gauge the generality of our method. For example, the PSNR figure in Table 4.7 for network 1 operating on signal m130 is computed according to (4.24) and (4.25) with $N = 20215 - 100 - 50 = 20065$ effective test data. That said, the PSNR results of the three SIRBFN predictors individually and jointly (as will be explained) over the complete speech samples can be found in Tables 4.7 to 4.11. Summary tables of minimum, average, and maximum performance gains are listed in Tables 4.9, 4.12, and 4.13 for the male only, female only, and joint male/female samples, respectively. The first four lines of

Network no.	m130	m131	m132	m133	m134
1	14.02	13.76	15.30	9.93	12.79
2	14.91	13.83	15.53	9.91	12.51
3	14.04	13.49	15.24	11.27	13.40
NL avg.	14.32	13.69	15.36	10.37	12.90
RLS (auto)	11.76 (14, 0.99)	11.77 (50, 0.999)	14.58 (50, 0.99)	10.85 (50, 0.999)	11.83 (50, 0.999)
RLS (NL)	16.07 (1, 0.98)	15.09 (1, 0.99)	17.08 (1, 0.98)	12.36 (1, 0.99)	14.75 (1, 0.999)
RLS (NL+auto)	16.43 (1+6, 0.99)	15.58 (4+14, 0.999)	17.73 (1+10, 0.99)	13.14 (3+6, 0.999)	15.31 (3+14, 0.999)

Table 4.7: Overall experimental results for speech prediction, samples m130 to m134 (all PSNR in dB)

each table list the individual predictor performances along with their arithmetic average. We see an average gain of 1.65dB of the basic regularized SIRBFN predictors over the

Network no.	m135	m136	m137	m138	m139
1	17.37	12.22	15.41	15.09	15.41
2	17.30	12.81	15.43	15.13	17.30
3	16.69	13.22	15.74	15.52	16.69
NL avg.	17.12	12.75	15.53	15.25	15.53
RLS (auto)	15.48 (46, 0.99)	10.92 (10, 0.99)	13.24 (50, 0.99)	12.48 (50, 0.999)	13.43 (50, 0.999)
RLS (NL)	18.63 (1, 0.99)	14.42 (1, 0.99)	17.05 (1, 0.999)	16.92 (1, 0.99)	16.20 (1, 0.999)
RLS (NL+auto)	19.35 (1+6, 0.99)	14.75 (1+4, 0.99)	17.42 (2+6, 0.999)	17.11 (5+10, 0.999)	16.86 (1+14, 0.999)

Table 4.8: Overall experimental results for speech prediction example, samples m135 to m139 (all PSNR in dB)

Gain	min.	avg.	max.
NL avg. over RLS(auto)	-0.48	1.65	2.77
RLS(NL) over NL avg.	0.67	1.58	1.99
RLS(NL+auto) over RLS(NL)	0.19	0.51	0.78
Total over RLS(auto)	2.29	3.73	4.67

Table 4.9: Summary of gains in experimental results for speech prediction, male speech samples (all figures in dB)

Network no.	f150	f151	f152	f153	f154
1	15.29	15.35	14.72	13.76	15.49
2	15.58	15.67	14.91	12.90	15.07
3	14.46	15.55	14.55	13.78	15.15
NL avg.	15.11	15.52	15.36	13.48	15.24
RLS (auto)	11.05 (50, 0.999)	13.10 (44, 0.99)	12.29 (44, 0.99)	9.869 (50, 0.999)	13.68 (46, 0.99)
RLS (NL)	16.74 (3, 0.999)	17.37 (1, 0.99)	16.42 (1, 0.98)	15.37 (3, 0.999)	16.93 (1, 0.99)
RLS (NL+auto)	16.93 (4+6, 0.999)	17.43 (1+4, 0.99)	16.47 (2+6, 0.999)	15.50 (4+8, 0.999)	17.16 (2+48, 0.999)

Table 4.10: Overall experimental results for speech prediction, samples f150 to f154 (all PSNR in dB)

Network no.	f155	f156	f157	f158	f159
1	18.02	15.60	15.42	11.74	15.38
2	18.14	15.97	15.95	12.40	14.39
3	17.78	15.89	16.19	13.06	16.15
NL avg.	17.98	15.82	15.85	12.40	15.31
RLS (auto)	16.08 (42, 0.99)	11.85 (50, 0.999)	13.15 (50, 0.999)	10.09 (50, 0.999)	14.26 (50, 0.999)
RLS (NL)	19.86 (1, 0.99)	17.45 (3, 0.999)	17.51 (1, 0.99)	13.96 (1, 0.999)	17.48 (1, 0.999)
RLS (NL+auto)	20.60 (1+4, 0.99)	17.70 (4+8, 0.999)	17.84 (4+32, 0.999)	14.29 (3+8, 0.999)	17.83 (3+8, 0.999)

Table 4.11: Overall experimental results for speech prediction example, samples f155 to f159 (all PSNR in dB)

Gain	min.	avg.	max.
NL avg. over RLS(auto)	1.05	2.67	4.06
RLS(NL) over NL avg.	1.06	1.70	2.17
RLS(NL+auto) over RLS(NL)	0.05	0.27	0.74
Total over RLS(auto)	3.48	4.63	5.88

Table 4.12: Summary of gains in experimental results for speech prediction. female speech samples (all figures in dB)

Gain	min.	avg.	max.
NL avg. over RLS(auto)	-0.48	2.16	4.06
RLS(NL) over NL avg.	0.67	1.64	2.17
RLS(NL+auto) over RLS(NL)	0.05	0.39	0.78
Total over RLS(auto)	2.29	4.18	5.88

Table 4.13: Summary of gains in experimental results for speech prediction, male and female speech samples (all figures in dB)

RLS predictor for the male speech samples while the average gain for the female speech samples is somewhat better at 2.67dB. Over both the male and female speech samples, the average gain is 2.2dB. The RLS predictor performance reported in the fifth line (with the corresponding autoregressive input order and exponential weight in parentheses) is the best one observed in a series of experiments for which the parameters vary as in Table 4.14. To allow a fair assessment of the gains possible from nonlinear versus linear prediction, the maximum order p of the linear predictor is set to 50, the same as for the SIRBFN. With regards to nonlinear speech predictors, these figures are in general agreement with

RLS parameter	Trial range/setting
$\boldsymbol{P}(0)$	$100\boldsymbol{I}$
$1-\rho$	$0, 10^{-6}, 10^{-4}, 10^{-3}$, 0.01:0.01:0.20
p	2:2:50

Table 4.14: Trial parameters for reference adaptive linear predictor (a:h:b denotes sequence from a to b inclusive sampled at h, $\boldsymbol{P}(0)$ is initial inverse of input correlation matrix, ρ is exponential weight)

those in previously published work (Townshend, 1991; Maria and Figueiras-Vidal, 1995). In particular, Townshend (1991) reported an increase in prediction gain of 2.8dB when a nonlinear predictor is trained on the residuals of a time-varying LPC predictor, which may be considered a *linear-nonlinear* processing scheme. We take another point of view to improving our nonlinear predictor performance by linearly combining the three predictor outputs, resulting in the *nonlinear-linear* processing scheme described below.

4.6.4 Linearly Combining Predictor Outputs for Improved Performance

During the course of the experiment, we noted that the error sequences produced by an ensemble of nonlinear predictor outputs trained on a given speech sample with different parameters exhibit some residual correlation with the desired prediction. This observation suggested that, by standard properties of least-squares estimators, some further improvement in prediction performance should be possible when the predictor outputs are used as inputs in an additional level of regression on the desired (actual) speech signal. In selecting a compatible structure for this subsequent processing, it was desirable to retain as much as possible the recursive on-line nature of the algorithm without significantly increasing the

computational burden. Thus the sixth line of the overall result tables shows the best observed performance for each speech sample when the three SIRBFN predictor outputs $\tilde{Y}_1(i)$, $\tilde{Y}_2(i)$, and $\tilde{Y}_3(i)$ at each time step i are formed into 3-tuples $\widetilde{\boldsymbol{Y}}(i) = \left[\tilde{Y}_1(i),\ \tilde{Y}_2(i),\ \tilde{Y}_3(i)\right]^T$ and taken as regressive vector inputs into another exponentially-weighted RLS predictor or *linear combiner* (to avoid confusion with the reference adaptive linear predictor). As before, the regressive orders and weights of the best such RLS linear combiners are given in parentheses following their performance figures and are chosen from trials conducted over the parameter ranges specified in Table 4.15. In most cases, only the most recent SIRBFN predictor outputs are necessary to provide a further nontrivial performance gain averaging 1.64dB over both the male and female speech samples. Augmenting the SIRBFN predictor

RLS parameter	Trial range/setting
$\boldsymbol{P}(0)$	$100\boldsymbol{I}$
$1-\rho$	$0,\ 10^{-6},\ 10^{-4},\ 10^{-3},\ 0.01,\ 0.02$
p	1:1:6

Table 4.15: Trial parameters for RLS linear combiner on SIRBFN outputs only (a:h:b denotes sequence from a to b inclusive sampled at h, $\boldsymbol{P}(0)$ is initial inverse of input correlation matrix, ρ is exponential weight)

output 3-tuples with autoregressive inputs drawn directly from the speech samples gives an additional small improvement of 0.51dB for the male speech samples and 0.27dB for the female speech samples, on average, for the best observed linear combiners. The exact performance figures for this nonlinear-linear input configuration are given in the seventh line of the tables, where the notation in parentheses is (*nonlinear 3-tuple order+linear autoregressive order, RLS weight*). Table 4.16 lists the trial parameter ranges in this final configuration, for which the average performance gain over the RLS predictor for both male and female speech samples is 4.18dB. This gain naturally comes at the price of increased computational complexity, namely $\mathcal{O}(n^3)$ per time step, where n is the number of basis function, versus $\mathcal{O}(p^2)$ for the linear RLS predictor, where p is the linear autoregressive order. Whether the increased computational complexity of the regularized SIRBFN predictor over a linear one such as the RLS predictor is acceptable depends upon the intended application but we should note that further gains in the nonlinear predictor's performance over the linear one should (at least in principle) still be possible since not all network parameters were fully

optimized, e.g., the bandwidth parameters.

4.7 Conclusion

Experiments performed on a suite of phonetically-balanced male and female speech samples demonstrate the nontrivial gains over linear techniques possible when the nonlinear processing of the regularized SIRBFN is applied to the one-step-ahead prediction of NLAR processes. We also describe how a simple linear combination of an ensemble of nonlinear predictor outputs via the RLS algorithm can yield further improvements in prediction performance with little added computational complexity while alleviating the difficulty of optimal model parameter estimation.

RLS parameter	Trial range/setting
$\boldsymbol{P}(0)$	$100\boldsymbol{I}$
$1-\rho$	$0, 10^{-6}, 10^{-4}, 10^{-3}, 0.01, 0.02$
p_{NL}	1:1:6
p_{auto}	2:2:50

Table 4.16: Trial parameters for RLS linear combiner on both SIRBFN outputs and autoregressive inputs (a:h:b denotes sequence from a to b inclusive sampled at h, $\boldsymbol{P}(0)$ is initial inverse of input correlation matrix), ρ is exponential weight)

Chapter 5

Other Applications

In this chapter, we introduce selected results from several other works in progress which are suggestive of the capabilities of the regularized SIRBFN method discussed in the main thesis. The first topic of the regression approach to the optimum nonlinear filtering problem is excerpted from (Haykin et al., 1997). The second topic of dynamic reconstruction of chaotic processes is based on work performed jointly with Sadasivan Puthusserypady at the CRL (Haykin et al., 1997). Because of this format, there may be some minor repetition of some concepts previously explained in a more general context but the same notations and conventions are followed as much as possible.

5.1 Nonlinear State Estimation

5.1.1 Problem Description

The classical *nonlinear state estimation* or *filtering* problem can be posited as follows: suppose that we are given a dynamical system described by the continuous-time *stochastic differential (s.d.e.)* equations

$$d\boldsymbol{X}(t) = f(\boldsymbol{X}(t))dt + d\boldsymbol{V}(t) \qquad \text{(state)} \qquad (5.1)$$

$$d\boldsymbol{Y}(t) = g(\boldsymbol{X}(t))dt + d\boldsymbol{W}(t) \qquad \text{(observation)} \qquad (5.2)$$

where $\{\boldsymbol{X}(t) \in \mathbb{R}^d : t \in \mathcal{T}\}$ and $\{\boldsymbol{Y}(t) \in \mathbb{R}^p : t \in \mathcal{T}\}$ are the *state* and *observation/measurement* processes defined over some index set $\mathcal{T} \subseteq \mathbb{R}^+$ and driven by (possibly correlated) noise processes $\{\boldsymbol{V}(t) \in \mathbb{R}^d : t \in \mathcal{T}\}$ and $\{\boldsymbol{W}(t) \in \mathbb{R}^d : t \in \mathcal{T}\}$, respectively. For example, one common case is $\mathcal{T} = (0, \infty)$ and the initial state $\boldsymbol{X}(0)$ has a given distribution. We would like estimate with m.m.s.e. the current state $\boldsymbol{X}(t)$ given the current

and past observations $\{\boldsymbol{Y}(s) : s \leq t,\ s,t \in \mathcal{T}\}$, i.e., determine the conditional expectation $\widetilde{\boldsymbol{X}}(t) \triangleq \mathbb{E}\left[\boldsymbol{X}(t) \middle| \mathcal{F}^{\boldsymbol{Y}}(t)\right]$, where $\mathcal{F}^{\boldsymbol{Y}}(t)$ is the *filtration* of the process $\{\boldsymbol{Y}(t)\}$. When $f : \mathbb{R}^d \mapsto \mathbb{R}^d$ and $g : \mathbb{R}^d \mapsto \mathbb{R}^p$ are known linear maps and $\{\boldsymbol{V}(t)\}$ and $\{\boldsymbol{W}(t)\}$ are *Wiener* processes so that their formal derivatives are w.s.s. *white noises* with known correlation matrices, the celebrated Kalman filter provides a recursive solution to the problem. When f and g are not linear (but still known), however, more general mathematical methods are required, which in practical terms calls for the *numerical solution* of the continuous-time s.d.e. system (5.1) and (5.2). To set the stage for comparison with the ANN approach to the same problem, we mention the following salient points and defer the full details to (Haykin et al., 1997):

1. the motivation behind the s.d.e. approaches is to deduce conditions under which a recursive, finite-dimensional scheme can be used to compute the conditional probability distribution of the state $\boldsymbol{X}(t)$ given the filtration $\mathcal{F}^{\boldsymbol{Y}}(t)$ produced by the observation process $\{\boldsymbol{Y}(s) : s \leq t,\ s,t \in \mathcal{T}\}$, where the expectation is with respect to the $(d+p)$-dimensional probability space P corresponding to the Wiener processes $(\boldsymbol{V}(t), \boldsymbol{W}(t))$. A (m.m.s.e.) optimal point estimate of the state can then be obtained as the mean of this conditional distribution.

2. in general, the solution of (5.1) and (5.2) results in conditions on *all* the higher-order conditional moments (of an appropriately transformed version of the state $\boldsymbol{X}(t)$) and thus leads to an *infinite-dimensional* system, which is clearly undesirable in applications. Of course, studies also show that, under suitable restrictions, these infinite-dimensional systems can be reduced to finite-dimensional systems.

3. a major practical difficulty with even the finite-dimensional schemes is they are often *unimplementable*, in the sense that the algorithms cannot be used to generate numerical results with only minor modifications such as the truncation of an infinite domain and the introduction of an iterative method (see the discussion on p.220 of (Sun and Glowinski, 1993)).

4. *pathwise* (i.e., a.s.) convergence is possible as well as a.s. and i.p. L_2 (i.e., m.s.) convergence are possible with the s.d.e. solution methods. For example, Sun and Glowinski (1993) apply the method of *operator splitting* to solve the *Zakai filtering equation* (related to the original system via a linear p.d.e.) for the unnormalized state

density and provide an estimate (under certain conditions) of the uniform rate of convergence for almost all sample paths.

The key point is that while the theoretical aspects of the nonlinear state estimation problem by the direct solution of the corresponding continuous-time s.d.e.s have generated many algorithms, their application in practical problems have been limited by both their requirement for strong *a priori* knowledge of the underlying maps and noise characteristics, and their difficulties in computationally feasible implementation (a complementary overview of factors inhibiting the widespread use of the s.d.e. approach can be found in the Introduction of (Lo, 1995)).

5.1.2 The ANN Approach

As with the continuous-time s.d.e. approach, the ANN approach is concerned with estimating the conditional mean of the state given present and past observations. On the other hand, the different nature of ANNs results in some modifications to the s.d.e. framework approach previously described:

1. Since ANNs have finite-dimensional domains, we must *time-discretise* the s.d.e.s in (5.1) and (5.2) governing the state transition and observations. We assume that after appropriately discretising time, we have the state equation

$$\boldsymbol{X}(i+1) = f(\boldsymbol{X}(i)) + \boldsymbol{V}(i), \quad i \in \mathbb{Z}^+ \text{ (state)} \tag{5.3}$$

where $\boldsymbol{X}(i) \in \mathbb{R}^d$ is the state vector at time step i, $f(\bullet)$ is a vector-valued nonlinear function of its argument, and $\boldsymbol{V}(i)$ is the process noise vector. Similarly, for the observation equation, we have

$$\boldsymbol{Y}(i) = h(\boldsymbol{X}(i)) + \boldsymbol{W}(i), \quad i \in \mathbb{Z}^+ \text{ (observation)} \tag{5.4}$$

where $\boldsymbol{Y}(i) \in \mathbb{R}^p$, $h : \mathbb{R}^d \to \mathbb{R}^p$, and $\boldsymbol{V}(i)$ is the observation noise vector.

2. As a result, the conditional expectation being estimated is that of the state with respect to a *finite* number m of present and past observations. This approximation is exact only if the conditional density is *Markov* to the truncation order m, otherwise the present and all past observations are, in theory, required. There is often, however, a practical limit on the performance improvements possible with increasing m, so this finite memory assumption is not unreasonably restrictive.

3. Given the above, after suitable least-squares training, the ANN output function then estimates the *regression function* of the state with respect to the chosen m observations and, by plugging in the m most current observations $\boldsymbol{y}_m(i) \triangleq [y(i), y(i-1), \ldots, y(i-m+1)]^{\mathsf{T}}$, yields a *point estimate* of the actual state $\boldsymbol{x}(i)$. All this stands in contrast to the continuous-time s.d.e. case, where the output is an estimate of the conditional state density with respect to the available observations, i.e., a *function*. Note that the ANN approach implicitly assumes that the regression function being estimated is *stationary* (or at least slowly varying) in the time index i, whereas the s.d.e. approach usually assumes the same for the state transition and observation function.

4. By their nature as flexibly parameterized classes of functions, ANNs typically require a random sample or *training set* of the process paths, denoted by

$$T_{N,S} = \left\{ (\boldsymbol{X}(i,\omega), Y(i,\omega)) \in \mathbb{R}^d \times \mathbb{R} : \omega \in S, i = 1, 2, \ldots, N \right\} \tag{5.5}$$

where S is a random sample of length $\#S$ from the sample space Ω, in place of the s.d.e. approach's *a priori* knowledge, e.g., of the state transition and observation functions, their associated noise statistics, and the initial state distribution. Although the assumed availability of state sample paths may appear as problematic as the s.d.e. approach's assumed knowledge of the underlying system functions, one can conceive of a scenario in which the system of interest is under control during the training phase and one is interested in estimating its state when such control is not possible. Not surprisingly, there is a price to pay for the generality of the ANN approach; for example, it is clear that, in general, because a given state transition or observation process sample path explores only a portion of its respective function domain, any finite training set, no matter how large, is not nearly as informative as knowing the actual underlying function. For ANN approaches to be both practical and successful, they must address this and the related design issues of appropriate network size and training algorithm complexity needed to construct a reasonable estimate.

5.1.3 Review of Current Approaches

In this section we review several current ANN approaches to the state estimation problem. Among the early applications of ANNs to optimum nonlinear filtering is the work

of (Lo, 1994, 1995). Here it is assumed that the state and observation equations are unknown, but that one could obtain enough sample data (in his notation) $\{(\boldsymbol{x}(t,\omega), \boldsymbol{y}(t,\omega)) . t = 1,\ldots,T, \omega \in S\}$ to adequately capture the underlying statistics of the state and observation variables. Assuming stationarity, the following result is proven for two distinct *recurrent multilayer perceptron (RMLP)* (Perlmutter, 1989) architectures which are described below.

The first RMLP architecture of (Lo, 1995) is called the *neural filter with fully-interconnected neurons (NFFN)* and, as its name suggests, has a fully recurrent input-output structure. Let t be the discrete time variable, and let the weight from the i-th input to the j-th neuron in the first layer be ω_{ji}^1; let ω_{ji}^2 be the weight of the output of the i-th neuron into the j-th hidden neuron, and ω_{ji}^r be the weight of the lagged feedback (by one time unit) from the i-th neuron to the j-th neuron (in the first input layer). Thus, the activation level $\beta_j(t)$ and the weighted sum $\eta_j(t)$ of the j-th neuron satisfy

$$\beta_j(t) = g(\eta_j(t)$$
$$\eta_j(t) = \omega_{j0} + \sum_{i=1}^{m} \omega_{ji}^1 y_i(t) + \sum_{i=1}^{q} \omega_{ji}^r \beta_i(t-1)$$

where g is a monotone increasing function such as tanh. The i-th output $\alpha_i(t)$ is then given by,

$$\alpha_i(t) = \sum_{j=1}^{q} \omega_{ji}^2 \beta_j(t) \tag{5.6}$$

for $i = 1, 2, \ldots, k$. The second RMLP architecture, called the *neural filter with ring-interconnected neurons (NFRN)*, has a partially recurrent input-output structure as sketched in Figure 5.1. Of the $i + j$ output nodes, $\alpha_1(t+1), \ldots, \alpha_k(t+1)$ are teacher-forced, and $\beta_1(t+1), \ldots, \beta_j(t+1)$ are free outputs trained to have the teacher forced outputs. All the free outputs and the i-th teacher forced outputs $\alpha_1(t+1), \ldots, \alpha_i(t+1)$ are delayed by one time unit before being fed into the input nodes $\beta_1(t), \ldots, \beta_j(t)$, $\alpha_1(t), \ldots, \alpha_i(t)$. In addition, the network has m input nodes on which the external inputs $\gamma_1(t+1), \ldots, \gamma_m(t+1)$ are clamped. Both of these networks are separately analyzed and the following result is deduced for both networks:

Theorem. *Consider the (discrete) random d-dimensional state process and p-dimensional observation processes* $\boldsymbol{x}(t)$ *and* $\boldsymbol{y}(t)$ *for* $t = 0, 1, \ldots, T$ *defined on a probability space* $(\Omega, \mathcal{F}, P)$.

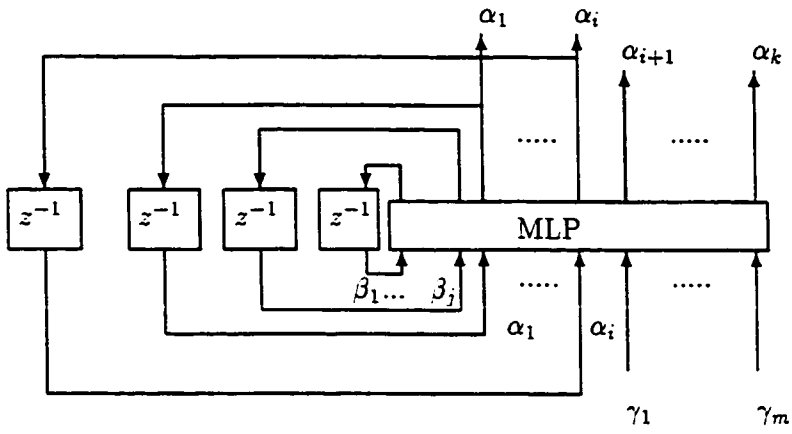

Figure 5.1: Input-output structure of NFRN (after (Lo, 1995))

Suppose that the range of $\{\boldsymbol{Y}(t,\omega)|\omega \in \Omega\}_{t=1}^{T} \subset \mathbb{R}^p$ *is a.s. compact, and that* $\mathbb{E}[\boldsymbol{X}(t)^2] \leq \infty$ *for* $t = 0, .., T$. *Then, if* $\boldsymbol{\alpha}(t)$ *is the network's output at time* t, *and* $\epsilon > 0$ *is given, there exists a sufficiently large RMLP such that*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\boldsymbol{\alpha}(t) - \mathbb{E}\left[\boldsymbol{X}(t)|\boldsymbol{Y}^{T}\right]\|^2\right] < \epsilon$$

where $\boldsymbol{Y}^{T} \triangleq [Y(1), \ldots, Y(T)]$.

The theorem states that the RMLP architectures in question are sufficiently flexible to approximate the behavior of the desired conditional mean function in mean-square to an arbitrary degree of accuracy over any given finite set of time points. While this theoretical result is necessary if the RMLP approach to optimum nonlinear filtering is to be a fruitful one, there are at least two distinct practical difficulties with implementing the conclusion of such an *existence* theorem, as we have elaborated in the Introduction:

1. For any nontrivial training set sizes and functions to be learned, the cost functions which must be minimized over all possible network parameters during learning are usually nonconvex and admit many *local minima*. Even if this optimization problem is alleviated, there remains the issue of the error induced in the "optimal" network parameters $\boldsymbol{w}^*$ when, as is common in many ANN learning schemes, these parameters are obtained by minimizing the *sample-based estimate*

$$\boldsymbol{w}^* \triangleq \inf_{\boldsymbol{w}} C(\boldsymbol{w}), \quad C(\boldsymbol{w}) \triangleq \frac{1}{T(\#S)}\sum_{\omega\in S}\sum_{t=1}^{T}\left\|\boldsymbol{\alpha}(t,\omega) - \mathbb{E}\left[\boldsymbol{X}(t)|\boldsymbol{Y}^{T}(\omega)\right]\right\|^2 \tag{5.7}$$

(where $\#S$ is the number of samples per time point and $\boldsymbol{\alpha}(t)$ is the network output at time t) in place of the desired target cost function itself. It should again be mentioned, however, that both of these issues have been addressed in other work on ANN learning, e.g., (White, 1989, 1990).

2. Given a finite training set, no guidelines are provided for selecting an appropriately-sized network, e.g., of n neurons, to yield the best *out-of-sample* or *generalization* performance. It is well-known that without such guidelines, the problems of *over-fitting* (too large an n) and *under-fitting* (too small an n) can lead to poor generalization performance, e.g., see (Geman et al., 1992).

Nonetheless, a clear advantage of such an ANN approach is that no *a priori* knowledge of the statistics of the state and observation processes is required, other than having sufficient sample data to properly train the network (via temporal backpropagation). On the other hand, the larger T (the period of operation of the filter) is, the larger the network needed and correspondingly the greater the training time. Furthermore, the iterative optimization methods used are not particularly well-suited to the incremental learning desired in a nonstationary environment.

In view of some of the shortcomings of the previous approach, we should mention the earlier work of (Parsini and Zoppoli, 1994; Parisini and Zoppoli, 1996). The first paper presents nonrecursive as well as recursive techniques, which reduce the filtering problem described in (5.3) and (5.4) to one of nonlinear optimization. Specifically, their recursive scheme involves sequentially minimizing n (where n is proportional to the observation period) functionals of the form:

$$J_i = \sum_{p=1}^{i} \phi_p \left(\| y_p - h_p \left(\hat{\boldsymbol{x}}_p \right) \| \right) + \sum_{p=1}^{i} \psi_{p-1} \left(\| \hat{\boldsymbol{x}}_p - f \left(\hat{\boldsymbol{x}}_{p-1} \right) \| \right) \tag{5.8}$$

for $i = 1, 2, \ldots, n-1$ (the case $i = 0$ is treated differently) where $\hat{\boldsymbol{x}}_i^p$ is an estimate of $\boldsymbol{x}_i$, and ϕ_p and ψ_p are arbitrary smooth increasing functions with $\phi_p(0) = \psi_p(0) = 0$. As compared with (5.7), these functionals are formulated according to what the authors call the *linear-structure preserving principle* which is designed to emulate the linear structure of the Kalman filter. The actual solution to each minimization problem relies on the following procedure performed upon a suitable MLP (the $\boldsymbol{x}_i$ are the state variables as in (5.3) and the $\boldsymbol{y}_i$ are the observation variables as in (5.4); a variable with a p-superscript are the *predicted*

versions of that variable):

$$\hat{x}_i = \hat{x}_i^p + \tilde{\gamma}\left(y_i - y_i^p, \omega_i\right), \quad i = 0, 1, \ldots, n-1$$
$$x_0^p = \alpha$$

where α is an *a priori* estimate of x_0, and $\tilde{\gamma}(e_i, \omega_i)$ is a multilayer feedforward network with weights ω_i and inputs $e_i = y_i - y_i^p$. The flow-chart for the scheme of (Parsini and Zoppoli, 1994) is reproduced in Figure 5.2 (recall that v and w are the state and observation noises, respectively): Thus, at step $m+1$, a nonlinear optimisation is performed on the

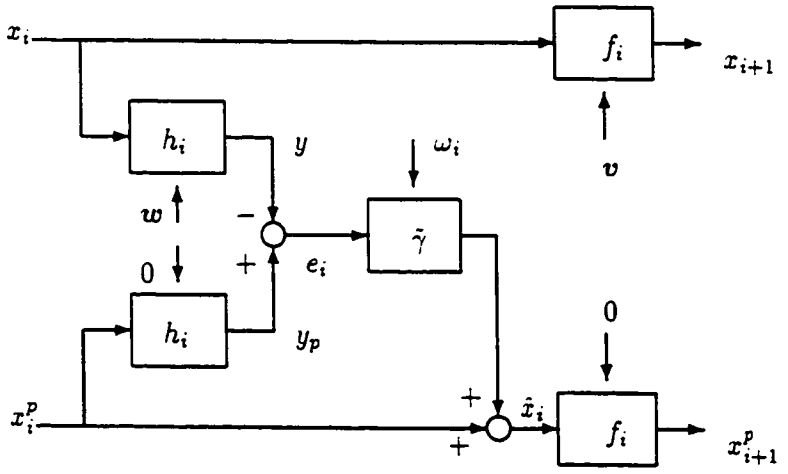

Figure 5.2: Flow-chart for scheme of (Parsini and Zoppoli, 1994)

set of weights ω_{m+1} of the $(m+1)$-st network while freezing the m previously computed weight vectors $\{\omega_i\}_{i=1}^m$. This recursive implementation has the price of being structurally sub-optimal (although not very much so in practice, according to the authors) compared to the alternative nonrecursively implementable scheme laid out by the authors. Fortunately, in contrast to the approach of (Lo, 1995), the weights are adjusted by a gradient descent algorithm, since the assumed knowledge of the probability distribution functions of the state and observation noises allows the generation of "realisations" of a gradient function. Then by using standard back-propagation rules, these gradients may be computed for use in an appropriate weight update function until convergence is achieved. Other assumptions in their method are that the state and observation processes are zero mean, i.i.d., and mutually independent. As with (Lo, 1994, 1995), numerical simulations are performed on

the problem of *bearings only measurement*, which is drawn from the more general class of *target motion analysis* problems. The test problem consists of an observer performing a series of tight manœuvres while acquiring noisy observations of the line of sight angle it makes with a target moving at constant speed. The results presented therein show significant performance gains over the extended Kalman filter where it is known that the filter can diverge due to ill-conditioning of the covariance matrix.

Even with the *a priori* knowledge of the underlying statistics of the system, the recursive nature of the above scheme addresses neither the problem of excessive network complexity (when the observation period is large or has no *a priori* bound) nor the problem of actual network design (i.e., how to determine the optimal network size and structure for a given observation period).

5.1.4 Proposed Approach

We now describe the application of the regularized SIRBFN in the ANN, i.e., regression, approach to the optimal nonlinear state estimation problem. As with the other ANN approaches previously considered, we assume that the nonlinear functions $f(\bullet)$ and $h(\bullet)$ (without loss of generality, we assume h to be scalar) are both unknown, as are the statistics of the corresponding noise processes $\{\boldsymbol{V}(i)\}$ and $\{W(i)\}$. This lack of knowledge is again partially compensated by the ability to perform discrete-time measurements on the system before it is put into normal operation. Unlike (Lo, 1994, 1995), however, under the stationary regression function assumption discussed in the previous section, we require only a single (sufficiently long) joint realization of the state and observation processes for network training, i.e., S in (5.5) is a singleton, hence we simply write $T_N \stackrel{\Delta}{=} \left\{(\boldsymbol{X}(i), Y(i)) \in \mathbf{R}^d \times \mathbf{R}\right\}_{i=1}^{N}$ where N is sequence length and d is the dimension of state vector $\boldsymbol{X}(i)$. Aside from this difference, the following issues from Chapter 4 bear repeating in the context of optimum nonlinear filtering:

1. for the regularized SIRBFN, Corollary 1 directly addresses item 1 of the discussion on the RMLP neural filter. The result also addressed item 2 by implying that the level of smoothing chosen by an appropriate CV method asymptotically achieves a proper balance between the extremes of underfitting (excessive estimator bias) and overfitting (excessive estimator variance), since otherwise the m.s.e. would not be minimized.

2. the training data are, in general, correlated from sample to sample, i.e., *dependence* is present. For example, the discrete state process $\{\boldsymbol{X}(i)\}$ constructed according to (5.3) is clearly dependent by the action of f (even when $\{\boldsymbol{V}(i)\}$ is an i.i.d. process). From this it follows that the observation process $\{Y(i)\}$ in (5.4) is also dependent, e.g., $Y(i)$ and $Y(i-1)$ are dependent. Because the m.s.- consistency of neural estimators has been usually claimed only in the case of i.i.d. training data, the optimality of such estimators in the stochastic filtering context cannot be asserted without further study.

3. the processes $\{\boldsymbol{X}(i)\}$ and $\{Y(i)\}$ may be *nonstationary*. For general f in (5.3), the state process $\{\boldsymbol{X}(i)\}$ is clearly nonstationary (see, e.g., p.47 of (Györfi et al., 1989)), hence the corresponding observation process $\{Y(i)\}$ is also generally nonstationary. Once again, the i.i.d. data assumption underlying the majority of neural network studies in not applicable.

Thus, in theoretical terms, the regularized SIRBFN appears to be well-suited to the regression approach to m.m.s.e. filtering or state estimation, in contrast with, e.g., the RMLP. In practical terms, the regression estimate is implemented in the obvious way:

1. from T_N, derive the d component-wise training subsets of length $n = N - m + 1$ as $T_{n,k} \triangleq \{(\boldsymbol{Y}_m(i), X_k(i)) \in \mathbb{R}^m \times \mathbb{R}\}_{i=m}^{N}$ for $k = 1, 2, \ldots, d$, where $X_k(i)$ is the k-th component of the state vector $\boldsymbol{X}(i)$ and $\boldsymbol{Y}_m(i) \triangleq [Y(i-j)]_{j=0}^{m-1}$.

2. for $k = 1, 2, \ldots, d$, obtain the k-th regularized SIRBFN estimate $\tilde{f}_{n,k}$ as the solution to the interpolation problem specified by $T_{n,k}$ in the usual way. Note that with this construction, each network may have its own regularization parameter $\tilde{\lambda}_{n,k}$ (among others) chosen by CV on its particular training subset. Alternatively, one could simply aggregate the d interpolation problems of the form (1.15) together with a *single* regularization parameter λ_n as

$$(\boldsymbol{G}_n + \lambda_n \boldsymbol{I})\, \boldsymbol{\theta}_n = \boldsymbol{X}_n \tag{5.9}$$

where the d-th column of $\boldsymbol{\theta}_n \in \mathbb{R}^{n \times d}$ is the weight vector for $\tilde{f}_{n,k}$ and the d-th column of $\boldsymbol{X}_n \in \mathbb{R}^{n \times d}$ is $[X_k(i)]_{i=m}^{N}$. For the small d occurring in the subsequent experiments, we use the former method since in that case the additional computational burden is not overwhelming.

3. form the overall network estimate $\widetilde{\boldsymbol{X}}(i)$ for $\boldsymbol{X}(i)$, $i > N$, as

$$\widetilde{\boldsymbol{X}}(i) = \left[\tilde{f}_{n,k}\left(\boldsymbol{Y}_m(i)\right)\right]_{k=1}^{d} \tag{5.10}$$

5.2 Experimental Results

In this section, we present two sets of experimental results. The results presented in Section 5.2.1 pertain to a comparison of the regularized SIRBFN approach to the s.d.e. approach described in (Sun and Glowinski, 1993). The results presented in Section 5.2.2 pertain to a comparison of the regularized SIRBFN approach to the RMLP approach described in (Lo, 1994, 1995).

5.2.1 Comparison to the SDE Approach

Here we repeat example 3 of (Sun and Glowinski, 1993) under modified conditions. The original s.d.e. defining the system is

$$\begin{aligned} d\boldsymbol{X}(t) &= \begin{bmatrix} 2\left(\cos 2t - X_1^2(t) - X_2^2(t)\right)\cos t \\ 2X_1(t) + \left(4 - X_1^2(t) - X_2^2(t)\right)\sin t \end{bmatrix} dt + \epsilon_X d\boldsymbol{V}(t) \\ d\boldsymbol{Y}(t) &= \begin{bmatrix} 2\sin\left(X_1(t)\right) \\ 2X_2^2(t) \end{bmatrix} dt + \epsilon_Y d\boldsymbol{W}(t) \end{aligned} \tag{5.11}$$

where $\boldsymbol{X}(t) \triangleq [X_1(t), X_2(t)]^\top$ and with parameters $\epsilon_X = \epsilon_Y = 0.1$, $\mathbb{E}[\boldsymbol{X}(0)] = [0.0]^\top$, $\boldsymbol{Y}(0) = [0,0]^\top$, and $t \in [0, \pi/2]$. The initial state $\boldsymbol{X}(0)$ is distributed as the standard normal about $\mathbb{E}[\boldsymbol{X}(0)]$. We discretise the s.d.e. in time using a simple forward Euler scheme with $\Delta t = 2\pi/(n-1)$ and $n = 10N$, where N is number of training pairs to be obtained by subsampling the full state and observation sample paths at rate 10. The discretized vector noise differentials $d\boldsymbol{V}(t)$ and $d\boldsymbol{W}(t)$ are sequentially simulated by using component-wise independent samples from a normal pseudo-random number generator with zero mean and variance Δt. To avoid numerical instability with the crude discretisation scheme used, we set $\epsilon_X = \epsilon_Y = 0.05$ and fixed the initial state at $\mathbb{E}[\boldsymbol{X}(0)]$; as will be seen, even these choices result in discretised state sample paths with somewhat greater spatial variability than that shown in Figure 3 of (Sun and Glowinski, 1993).

We follow the SIRBFN design procedure detailed in the previous section. The only change is that because we have two-dimensional *vector* observations, the *effective* input vector is formed by concatenating m observation vectors into a single $2m$-long vector

as $\boldsymbol{y}_{2m}(i) \triangleq \left[\boldsymbol{y}^{\mathsf{T}}(i), \boldsymbol{y}^{\mathsf{T}}(i-1), \ldots, \boldsymbol{y}^{\mathsf{T}}(i-m+1)\right]^{\mathsf{T}}$. The result is two scalar output networks $\widetilde{f}_{n,1}$ and $\widetilde{f}_{n,2}$ which estimate the regression of the discretised state components $X_1(i)$ and $X_2(i)$ with respect to a common input vector $\boldsymbol{Y}_{2m}(i)$. The vector regression order m is determined for $\widetilde{f}_{n,k}$, $k = 1, 2$, by setting their RBFN input norm weighting matrices to $\boldsymbol{U}_{n,k}^{-1} \triangleq \text{diag}\left[\alpha_{k,j}\sigma_{k,j}^2\right]_{j=1}^{m}$, where $\sigma_{k,j}^2$ is the sample variance of the j-th input variable over $T_{n,k}$. These *input scaling parameters* $\alpha_{k,j}$ are estimated from the training data $T_{n,k}$ along with the regularization parameter $\lambda_{n,k}$ by the GCV procedure. The approximate minimization of the GCV criterion function is performed by the MATLAB™version 4.2c Optimization Toolbox routine `constr`, which implements a simplex-type method of nonlinear multivariate minimization over a quadrantal region specified by upper and lower bounds on the input variables. The relevant settings used in this problem are listed in Table 5.1 (optimizer settings not listed are kept at their default values). Because of the inverse weighting,

Optimizer settings	
`OPTIONS(2)` (input termination tolerance)	10^{-6}
`OPTIONS(3)` (function termination tolerance)	10^{-6}
`OPTIONS(14)` (max. no. of iterations)	1000

Variable search settings			
Input variable	Min.	Init.	Max.
λ	10^{-8}	0.001	1
α	10^{-4}	0.5	10^{8}

Table 5.1: Optimization settings for `constr` routine in approximate GCV minimization (settings for α apply to each input scaling parameter)

scalar observations (input components) corresponding to the larger estimated input scaling parameters have less influence on the network output (for a given input) than those associated with smaller input scaling parameters, so it appears reasonable to set m just large enough to include these values. What is most interesting is that even as m is increased, the smallest input scaling parameters for both networks are associated with only the most recent observation vector $\boldsymbol{y}(i)$, with all other input scaling parameters at least an order of

magnitude larger, i.e., less significant. A typical example of this phenomenon for $m = 2$ (albeit for $\epsilon_x = \epsilon_y = 0.1$) can be found in Table 5.2. Given that $m = 1$ appears sufficient,

Estimate	$\alpha_{k,1}$	$\alpha_{k,2}$	$\alpha_{k,3}$	$\alpha_{k,4}$
$\tilde{x}_1(i)$ $(k = 1)$	0.2568	1.990	2.889	1.906
$\tilde{x}_2(i)$ $(k = 2)$	2.008	0.1205	1.240	1.039

Table 5.2: Example of GCV-selected input scaling parameters for $m = 2$ and $\epsilon_x = \epsilon_y = 0.1$

the final network input scaling and regularization parameters as determined by the GCV procedure are listed in Table 5.3. The generalization performance of the two networks is

Estimate	$\alpha_{k,1}$	$\alpha_{k,2}$	λ
$\tilde{x}_1(i)$ $(k = 1)$	0.1551	0.9091	0.004085
$\tilde{x}_2(i)$ $(k = 2)$	0.3480	2.578	0.001449

Table 5.3: GCV-selected parameters for final network, s.d.e. comparison

tested against 20 other state sample paths generated independently in the same fashion as the training data; some representative results are presented in Figures 5.3 to 5.6, in approximate order of increasing performance as measured by the root m.s.e. (r.m.s.e.) for each state component. Although the reader is invited to compare these figures with the corresponding Figure 3 of (Sun and Glowinski, 1993), the substantially different nature of our approach, i.e., regression, compared to that of (Sun and Glowinski, 1993), i.e., pathwise-convergent numerical solution of s.d.e.s, due caution should be exercised in drawing any definitive conclusions, especially in view of the limited scope of the experiment. We should also mention that a comparison to an extended Kalman filter was attempted but was not successful due to difficulties in obtaining an appropriate equivalent discrete-time system for the continuous-time nonlinear system (5.11); such a comparison, however, can be found in the next experiment. That a m.s.e.- minimizing procedure can yield reasonable pathwise approximations can, nonetheless, be taken as a positive sign for the regularized SIRBFN approach to other similar problems. Not surprisingly, this same facet of regression-based

approaches can and does result in the loss of pathwise convergence, an example of which is shown in Figure 5.3. In applications, other factors such as computational complexity may well determine the choice between the two modes of convergence.

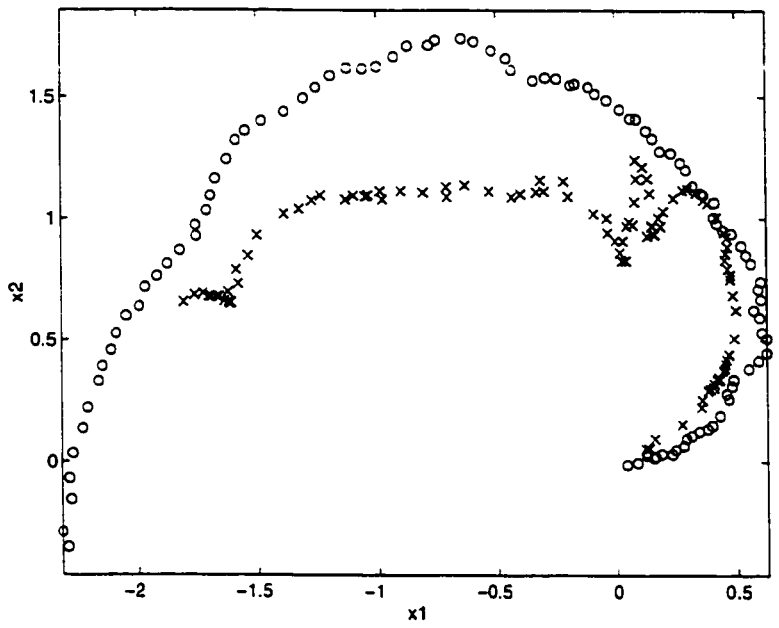

Figure 5.3: Example of estimated ('x') vs. actual ('o') state sample path, rmse $x_1 = 0.3423$, rmse $x_2 = 0.3940$

5.2.2 Comparison to the RMLP Approach

To compare the regularized SIRBFN approach to the RMLP approach, we repeat the experiment for Example 1 in (Lo, 1995). For reference, the one-dimensional signal/sensor system defined for $i \in \mathbf{Z}^+$ as

$$X(i+1) = 1.1\exp(-2X^2(i)) - 1 + 0.5V(i) \tag{5.12}$$

$$Y(i) = X^3(i) + 0.1W(i) \tag{5.13}$$

where $X(0)$ is Gaussian with mean -0.5 and variance 0.1^2. The signal/sensor noises $\{V(i)\}$ and $\{W(i)\}$ are statistically independent, standard white Gaussian sequences with zero mean and unit variance.

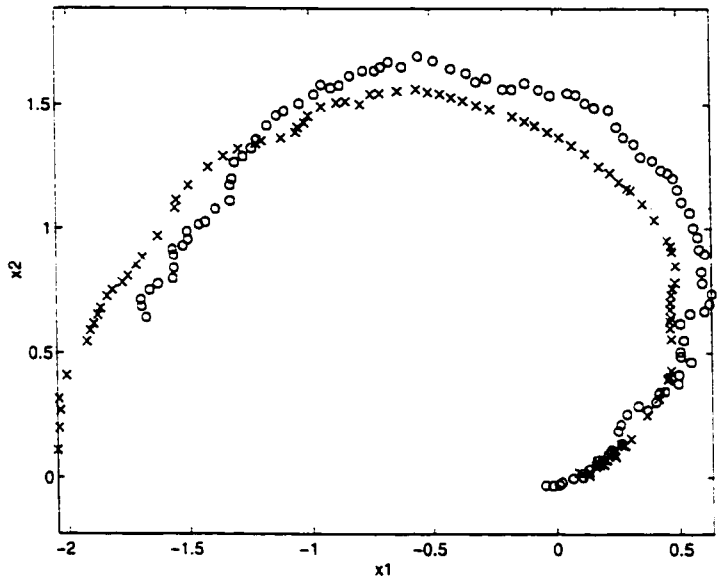

Figure 5.4: Example of estimated ('x') vs. actual ('o') state sample path, rmse $x_1 = 0.1762$, rmse $x_2 = 0.1950$

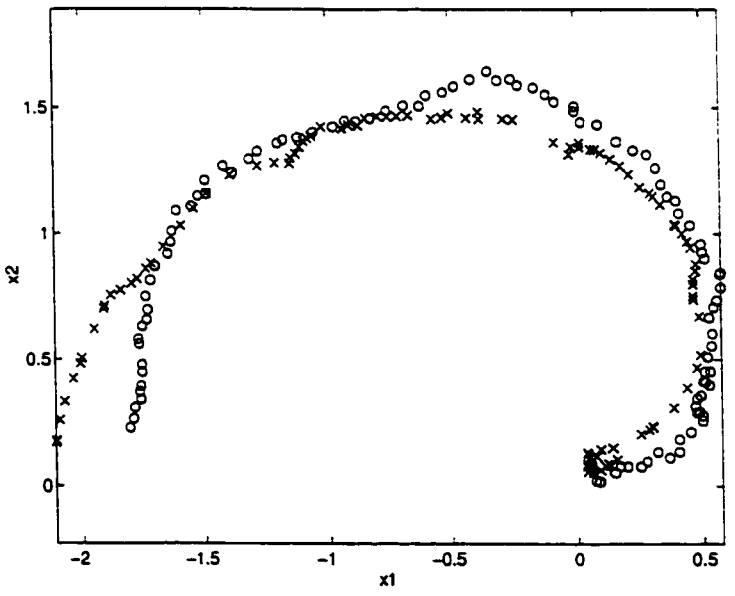

Figure 5.5: Estimated ('x') vs. actual ('o') state sample path, rmse $x_1 = 0.1757$, rmse $x_2 = 0.0967$

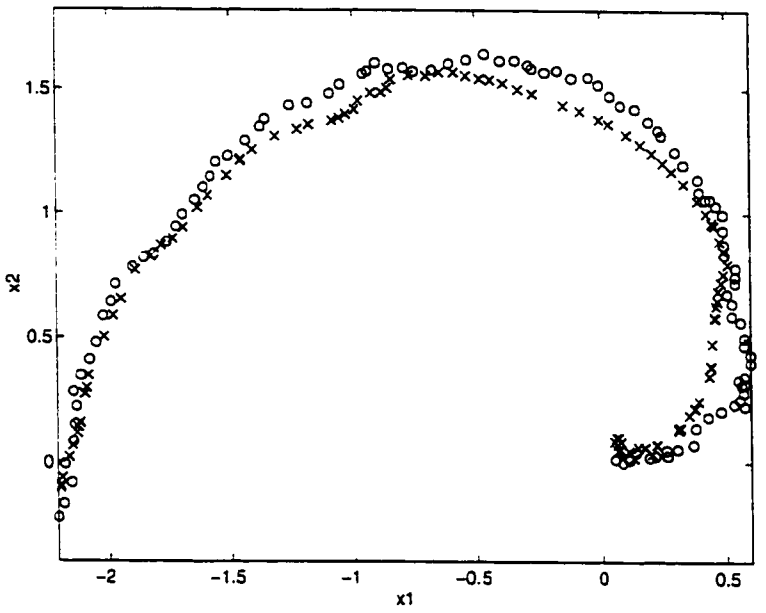

Figure 5.6: Example of estimated ('x') vs. actual ('o') state sample path, rmse $x_1 = 0.1252$, rmse $x_2 = 0.1191$

Figures 5.7 and 5.8 show that even with as few as $N = 100$ training data, f and h can be estimated from T_N with fair accuracy. As before, each regularized RBFN network has a Gaussian kernel $K(r) = \exp(-r^2/2)$, $r \in \mathbb{R}^+$, with diagonal norm weighting matrix of the form $\boldsymbol{U}_n^{-1} \triangleq \text{diag}\left[\alpha_j \sigma_j^2\right]_{j=1}^d$, where $d = 1$ is the state dimensionality and the σ_j^2 is the sample variance of the j-th input variable over T_N. These relative input weights and the regularization parameter λ_n are again determined from T_N by approximately minimizing the GCV criterion using the same optimizer settings as in Table 5.1.

Naturally, the quality of the estimation improves in the region containing a higher density of training points than at the outliers in the training set; this aspect is most clear in Figure 5.7, where the outlier near $x(i-1) = -2.5$ has an exaggerated effect on the shape of the estimated curve.

In addition to direct regression of the state at time step i with respect to the m most recently available observations at time step i, two other heuristic approaches were tried to see whether the estimates $\widetilde{f}$ and $\widetilde{h}$ could be used to improve the regression performance. Specifically, additional regressor variables were introduced as follows:

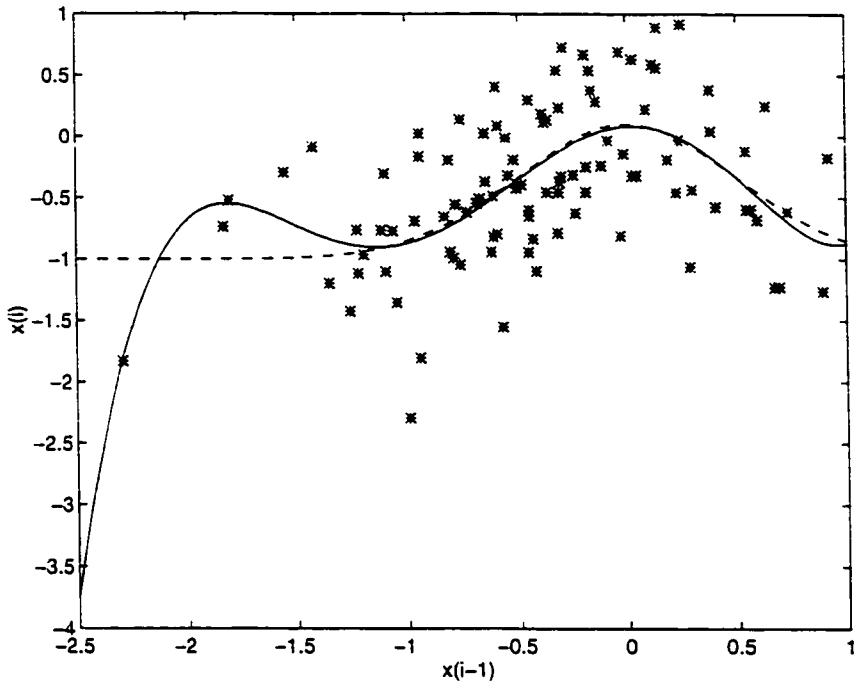

Figure 5.7: Estimated transition function $\tilde{f}$ (solid) vs. actual transition function f (dashed)

Estimate	α_1	α_2	λ
$\tilde{x}(i)$	0.5746	274.5	0.05228

Table 5.4: GCV-selected parameters for final network, RMLP comparison

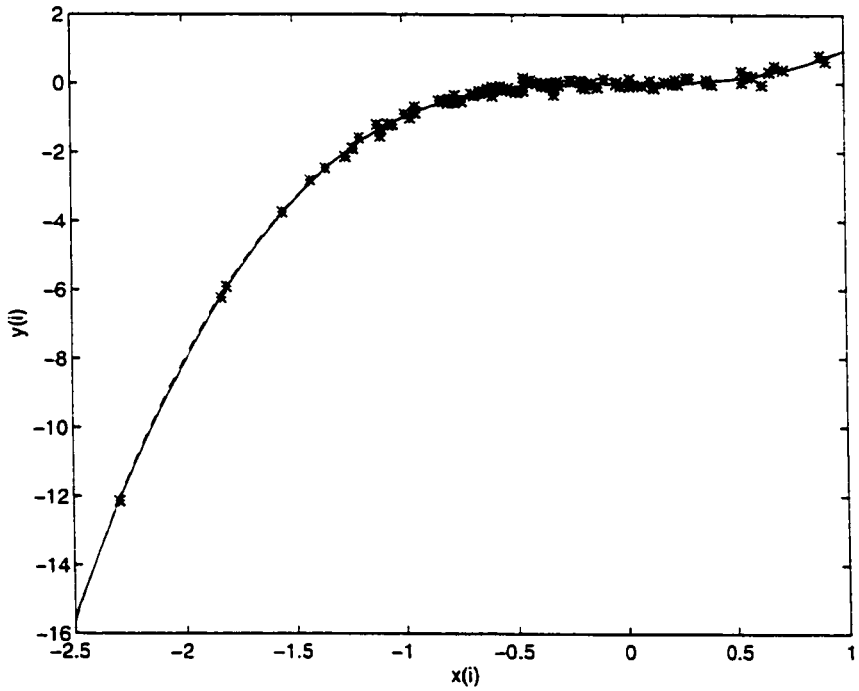

Figure 5.8: Estimated observation function $\widetilde{h}$ (solid) vs. actual observation function h (dashed)

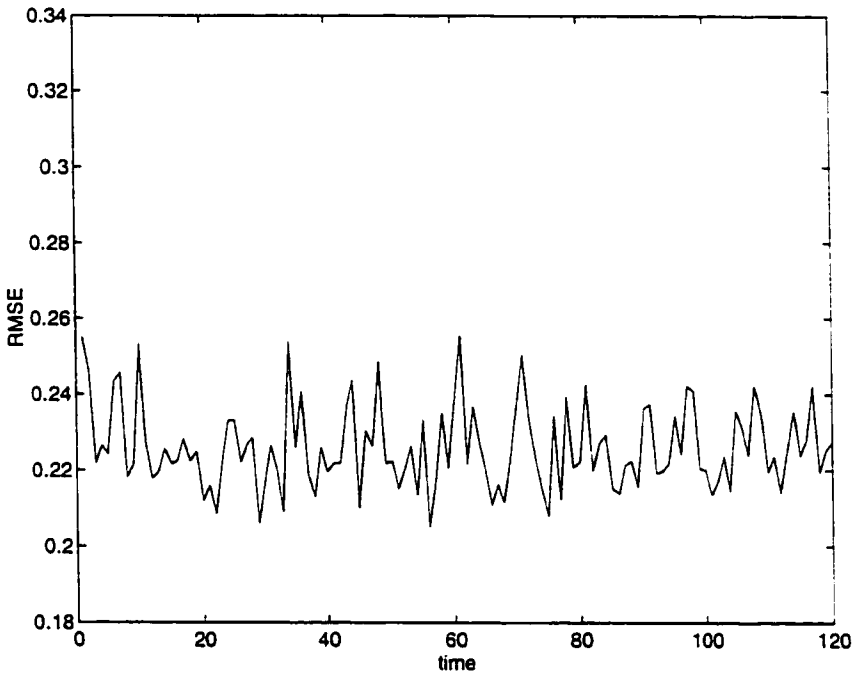

Figure 5.9: State estimate r.m.s.e over 1000 test sequences of 120 time points.

1. the p most recently available *estimated observations* at time step i computed using the *observation function estimate* $\tilde{h}$ as $\tilde{y}(i-j) = \tilde{h}(\mathbf{x}(i-j))$, $j = 1, 2, \ldots, p$, are added as inputs. Note that $\tilde{y}(i)$ is not included as an input because it would require $\boldsymbol{x}(i)$ first.

2. the p most recently available *estimated states* at time step i computed using the *state function estimate* $\tilde{f}$ as $\tilde{\boldsymbol{x}}(i-j) = \tilde{f}(\boldsymbol{x}(i-1-j))$, $j = 0, 1, \ldots, p-1$, are added as inputs.

Our simulations indicated no statistically significant improvement in estimation performance with these possible additional inputs, even with increased m and p. In fact, the scaling factor α_1 selected by the GCV criterion for the input variable $y(i)$ is much smaller, i.e., indicates greater significance, than the other α_j for both the other approaches with the additional regressor variables. For example, using $m = 2$ for $\tilde{\mu}$ gives $\alpha_1 = 0.5746$ for $y(i)$, while $y(i-1)$ is assigned $\alpha_2 = 274.5$. That this observation holds across the different m and p for the other approaches strongly suggests that most of the information about $x(i)$ is contained in $y(i)$, i.e., $m = 1$ would be sufficient. As a representative result, Figure 5.9 shows the average r.m.s.e. over 1000 test sequences of 120 time points with $N = 800$ training data and a regressive order of $m = 2$ for the observations. Visually, this figure appears quite similar to that in Figure 2.4 of (Lo, 1994, 1995) for the RMLP neural filters. A summary of the numerical results in comparison to the original results of (Lo, 1994, 1995) for the same problem can be found in Table 5.5. Although the mean r.m.s.e. of 0.2260 (with standard

Network	r.m.s.e. of state estimate
RBFN	0.2260
IEKF (Lo)	0.2806
NFFN (Lo)	0.2120
NFRN (Lo)	0.2122

Table 5.5: Comparison of state estimate r.m.s.e. over 120 time points for system defined by (5.12) and (5.13).

deviation of 0.0111) over the 120 time points is somewhat larger than the 0.2120 and 0.2122 reported for the two neural filters in Figure 2.4 of (Lo, 1995), one should keep in mind that

the number of training data $N = 800$ we used is much smaller than the 200,000 training samples effectively used in the iterative algorithm of (Lo, 1995) (100 sweeps or *epochs* over $\#S = 200$ and $N = 100$). Of course, we use many more basis functions than the seven neurons that (Lo, 1995) uses but this tradeoff between the number of training data and final network complexity is expected. Furthermore, the regularized RBFN figure is still lower than 0.2806 reported in that report for the iterated extended Kalman filtering algorithm.

5.3 Discussion

The stochastic partial differential equation (s.d.e.) approach to the nonlinear state estimation problem is well-established theoretically and has led to extensive studies of their numerical implementation. The necessary tradeoff for this precision to obtain, in the form of the rather strong *a priori* knowledge required, is impractical in many cases, leading to the development of nonparametric approaches, such as ANNs, which relax these assumptions. On the other hand, the ANN approach requires the availability of (potentially many) sample state and observation paths for training, i.e., estimation of the regression of the state with respect to the observations, which conceivably may not be realistic in some situations. Barring the computational complexity of the s.d.e. methods, it is clear that when strong prior knowledge is available, one should exploit it fully by using the model-based s.d.e. approach to generate parsimonious solutions with known properties. If such prior knowledge is not available but sufficiently many representative sample paths are, then the first experiment, the comparison to the s.d.e. solution method of (Sun and Glowinski, 1993), suggests that ANN regression-based methods can offer a useful alternative, keeping in mind the key differences of function vs. point estimate and pathwise vs. m.s.-convergence between the two approaches.

On the basis of the limited simulation results obtained, it would be clearly premature to claim that either the regularized SIRBFN (with a.o.-selected regularization parameter sequence) or the RMLP-based method provides superior performance among the ANN-based regression approaches to optimum, i.e., m.m.s.e., nonlinear state estimation. Nonetheless, the relatively stronger theoretical support for the practical techniques used in regularized SIRBFN design is certainly a significant advantage when compared to the uncertainties surrounding the theoretical effectiveness of the design methods used in the RMLP-based approach. We should also note that, from a computational standpoint, the

comparatively simplified training of regularized SIRBFN also admits the possibility of applying one of the reduced complexity recursive update algorithms detailed in Section 4.5, while the RMLP-based method does not easily do so. Such updating can be important when the state transition function f and the observation function h are time-variant.

Despite the initial success of the ANN-based regression approaches to the nonlinear state estimation problem, many open questions remain to be answered as listed below:

1. what are the particular features of the regression induced by the specific structure in the coupled dynamics of equations (5.3) and (5.4)? It seems intuitive that the induced regression should admit more structure than for an arbitrary regression problem. If such features do indeed exist, how can they then be exploited to obtain improved performance or otherwise aid in estimator design, e.g., selecting an appropriate input order m?

2. can the preliminary state transition estimate $\tilde{f}_n$ and observation function estimate $\tilde{h}_n$ be exploited in the regression approach to improve performance? In the comparison to the RMLP-based method, we did not observe any significant change in estimation performance when $\tilde{f}_n$ and $\tilde{h}_n$ were used in the "obvious" way to generate additional regressor variables in the form of estimated states or observations. It would be useful to develop some theory to account for this behaviour in more general situations.

3. is it possible to implement the optimal state estimate in the structure that parallels the linear Kalman filter? For example, under what conditions would a m.m.s.e. optimal state estimate $\widetilde{\boldsymbol{X}}^*(i) \triangleq \mathbb{E}[\boldsymbol{X}(i)\,|\boldsymbol{Y}_m(i)]$ be expressible in the the *predictor-corrector* form

$$\widetilde{\boldsymbol{X}}^*(i) = \tilde{f}_n\left(\widetilde{\boldsymbol{X}}^*(i-1)\right) + G\left(\boldsymbol{Y}_m(i)\right) \tag{5.14}$$

and how could such a G (if it exists) be estimated from the training data in T_N? Although equivalent in performance (at least in principle) to the direct regression of the state with respect to the observations, such a construction would have the advantage of efficient recursive computability. Variations on (5.14) include allowing general F in place of $\tilde{f}_n$ and ascertaining conditions under which the argument $\boldsymbol{Y}_m(i)$ could be replaced by the *innovations* or *a priori errors* $Y(j)-\tilde{Y}(j), j = i, i-1, \ldots, i-m+1$, as in the linear Kalman filter.

4. when f or h is nonstationary, i.e., time-varying, which procedures should be used to ensure that the state estimates continue to adapt to or *track* the actual state (with m.m.s.e.)? In view of the discussion above, it would be particularly advantageous these updates could be made recursive while maintaining the predictor-corrector structure (5.14).

In applications where our knowledge of the system dynamics is practically limited, nonparametric techniques such as those based on ANNs hold much promise. Of these ANN-based regression approaches to the nonlinear state estimation problem, the regularized SIRBFN stands out as being computationally feasible as well as theoretically supported in its application. As the issues raised above are clarified, it is expected that RBFN-based ANN approaches will have a significant role to play in providing efficacious solutions to the nonlinear state estimation problem.

5.4 Dynamic Reconstruction of Chaotic Processes

5.4.1 Problem Description

Chaotic dynamical systems have emerged recently as an important theory for understanding the possible mechanisms behind observed time signals. Indeed, one may say that the theory attempts to describe seemingly *random*, complex behaviours as the product of a set of "simple" (in a well-defined sense), *deterministic* coupled differential equations. Due to the breadth of the field, we shall necessarily confine our attention to those aspects of chaos which are absolutely relevant in understanding the application of regularized SIRBFNs to the *dynamic reconstruction* of chaotic systems. For further background, the reader may consult one of several possible general references, e.g., (Abarbanel, 1996; Baker and Gollub, 1996; Ott, 1993; Ott et al., 1994). Certain recent works, among them (Scargle, 1992) and (Tong, 1992), argue that tools from statistical time series are still relevant in the chaotic context, a viewpoint which we shall also reflect in our approach.

In a generic sense, the problem of the *dynamic reconstruction* of chaotic systems is fairly straightforward to state: given a finite length sample t_N of a real-valued time series $\{x(i)\}_{i=1}^{N}$, estimate a function $\widetilde{f} : \mathbf{R}^p \mapsto \mathbf{R}$ for some given $p > 0$ such that the *recursively forward iterated (r.f.i.)* predicted sequence $\{\widetilde{x}(i)\}_{i=N+1}^{\infty}$ formed by feeding successive r.f.i.

estimates back into the function input as

$$\tilde{x}(i+1) \triangleq \tilde{f}(\tilde{\boldsymbol{x}}_p(i)) \tag{5.15}$$

where

$$\begin{aligned} \tilde{\boldsymbol{x}}_p(i) &\triangleq [\tilde{x}(i), \tilde{x}(i-1), \ldots, \tilde{x}(i-p+1)]^\top, \qquad i = N+1, N+2, \ldots \\ \tilde{x}(i) &\triangleq x(i), \qquad i = N-p+1, N-p+2, \ldots, N \end{aligned} \tag{5.16}$$

(approximately) reproduces given chaotic characteristics of the system responsible for the generation of the sequence $\{x(i)\}_{i=1}^{N}$. Where matters become more involved are, of course, in the chaos theory behind the interrelated issues of the choice of p, the existence and properties of the function $\tilde{f}$, and the definition of the characteristics to be reproduced. These issues will be addressed as needed to clarify their role in the design of regularized SIRBFNs for dynamic reconstruction. Even without delving into these technical details, it is already clear that the objective here is different in character from that of the previous chapters:

1. in the previous chapters, the main concern was minimizing the (global) m.s. *error* of the estimate $\tilde{f}$ in approximating a function f assumed to relate the training input and output data (perhaps implicitly, as in the case of stochastic T_n and m.m.s.e. estimation). For dynamic reconstruction, the emphasis is on the *reproduction* of particular *features* of the chaotic dynamical system underlying the training data, e.g., local and global *attractors* in the *phase space*. While the former is necessary for the latter, it is clearly not sufficient, in the sense that two estimates with the same level of m.s.e. can have radically different phase portraits compared to the original chaotic system.

2. in the regression context, the unknown function f being estimated was well-defined in the sense that given stationary processes $\{\boldsymbol{Z}(i)\}$ and $\{Y(i)\}$, the conditional expectation function $f(\bullet) \triangleq \mathbf{E}[Y(i)|\boldsymbol{Z}(i) = \bullet]$ is a.s. unique, i.e., if g is another version of the conditional expectation, then $f = g$ a.s.–$P_{\boldsymbol{Z}}$. On the other hand, it is intuitive that the function f used in dynamic reconstruction is not unique, i.e., there may exist more than one function whose r.f.i. predicted time series captures the key chaotic characteristics of the original chaotic system. In fact, the key theorems supporting the dynamic reconstruction approach, e.g., *Takens' embedding theorem* (Takens, 1981),

merely assert (under suitable conditions) the *existence* of a smooth map f whose r.f.i. predicted time series has the required chaotic properties.

3. the r.f.i. operational mode of the estimate $\tilde{f}$ is unusual for the regularized SIRBFN, given its design by simultaneous interpolation for the OSA prediction problem, which is a *nonsequential* optimization procedure. The obvious error measure to use in place of $\tilde{J}_2$ (as defined in (1.5)) is the average squared *r.f.i.* prediction error over t_N, viz.

$$\rho\left(\tilde{f}, t_N\right) \triangleq \frac{1}{n} \sum_{i=p+1}^{N} \left(x(i) - \tilde{f}\left(\tilde{\boldsymbol{x}}_p(i)\right)\right)^2 \tag{5.17}$$

whose minimization (with respect to the network parameters in $\tilde{f}_n$) requires a *sequential* procedure due to the recursive construction of the input vector $\tilde{\boldsymbol{x}}_p(i)$ at each time step i. The intuition that such r.f.i. prediction-based cost functions are more appropriate for dynamic modelling than simultaneous OSA prediction-based cost functions is borne out in several simulation studies, e.g., (Principe et al., 1992) and (Principe and Kuo, 1995), among others. While there is no theoretical barrier to the application of such procedures to RBFNs, several issues would have to be addressed:

   (a) the development of analogous CV theory and techniques for the choice of λ (and possibly other parameters). It should be noted that, as a heuristic, one can select λ on the basis of the r.f.i. prediction error (in a suitable norm) over a holdout set that is distinct from the training data used to define the network. This heuristic parameter selection method, which we shall call *cross-validation via iterated prediction (CVIP)*, did not improve performance significantly for the Lorenz and sea clutter reconstruction experiments which are described further on.

   (b) the computational complexity, which can be geometrically greater than that of solving the SI linear algebraic equations (e.g., at least $\mathcal{O}(Mn)$ function evaluations for M iterations over the n training pairs in T_N for the former case vs. $\mathcal{O}(n^3)$ arithmetic operations for the latter case. Because of the lack of an *a priori* bound on M, it can be much larger than n^2.).

   (c) the resultant (global) *stability* of the closed-loop system generated by operating in the r.f.i. predictive mode. Such stability is not, in general, a natural byproduct of either standard simultaneous (regularized) least-squares or the proposed design

procedure based on sequential optimization in the r.f.i. predictive mode. Note that while sufficient, accurate reproduction of the attractors (e.g., fixed points and orbits in the corresponding system phase space) by the estimate $\tilde{f}_n$ is clearly not necessary to ensure such stability.

Given the elementary state of our knowledge regarding the precise effects of these factors, they should be kept in mind when evaluating the performance of the regularized SIRBFN in the following experiments. That said, the experiments do show that the regularized SIRBFN can form the basis for an effective dynamic reconstruction technique, particularly when the training data are *noisy*. As we will elaborate further on, this critical aspect of any realistic dynamic reconstruction problem has not to date been satisfactorily addressed by existing methods, which typically rely on *ad hoc* pre-filtering of the training data.

5.4.2 Review of Current Approaches

Roughly speaking, methods used in the dynamic reconstruction of chaotic processes can be divided into two classes (Casdagli, 1989):

1. *global methods*, which construct a single approximating function $\tilde{f}$ for all training data in T_N. Classical examples include polynomials and their rational compositions.

2. *local methods*, which construct the approximation $\tilde{f}$ by concatenating functions with *localized* support. More precisely, the prediction at some $\boldsymbol{x} \in \mathbb{R}^p$ in the domain of $\tilde{f}$ is computed by a function constructed only with data in the neighbourhood of $\boldsymbol{x}$. Some well-known examples of such methods include nearest-neighbour methods and their generalized kernel extensions along with piecewise polynomials (e.g., cubic splines).

Each class of methods has its own particular advantages and disadvantages; for example, global methods offer the possibility of parsimonious (in the number of function parameters) representations but in practice, without significant prior knowledge, one must often resort to flexible nonparametric methods with a large number of free parameters whose solution is computationally intensive. On the other hand, local methods involve functions which may be simpler to estimate from the training data over each local region but again, without some prior knowledge, the number of localized functions required can be large, leading to a large run-time network. Strictly speaking, the regularized SIRBFN with Gaussian kernel which we shall be using is, a global method, since the support of each network

basis function is all of $\mathbf{R}^p$, but it is often effectively considered a local method, since each basis function models primarily the region about its centre (which is an input datum in T_N). Indeed, this hybrid nature is one of the factors which lead Casdagli (1989) to an early application of nonregularized RBFNs to the dynamic reconstruction problem. That work shows that, compared with polynomial and MLP-based predictors, nonregularized RBFNs can predict well in short-term one-step-ahead (OSA) mode, i.e., the prediction of $x(i+j)$ with $\tilde{x}(i+j) \triangleq \tilde{f}(\boldsymbol{x}(i+j-1))$, $j = 1, 2, \ldots, M$ for small M, as well as the more difficult r.f.i. mode. The work notes, however, degraded prediction performance and, more importantly, poor recovery of key dynamical invariants for the nonregularized RBFN when white Gaussian noise is added to the time series considered. While stating that this shortcoming can be addressed by a variety of techniques, the work does not specifically examine nor test any one of those techniques. Similarly, in the global modelling method of (Principe et al., 1992) and later (Principe and Kuo, 1995) using MLPs whose parameters are estimated by least-square minimization of the r.f.i. prediction error, the potential deleterious influence of noise on the quality and accuracy of reconstruction is not considered. We shall later give an example which clearly demonstrates the importance of regularization to SIRBFN predictor performance in dynamic reconstruction from noisy observations of chaotic processes.

5.5 Experiment: Reconstruction of the Lorenz System from Noisy Data

5.5.1 Results for Noise-free Case

The *Lorenz system* (Lorenz, 1963), one of the most well-known chaotic systems, involves three coupled ordinary differential equations as follows:

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= -xz + rx - y \\ \dot{z} &= xy - bz \end{aligned} \tag{5.18}$$

These equations have their basis in a Galerkin approximation to the p.d.e.s describing thermal convection in the lower atmosphere. In physical terms, the variable x is related to the intensity of the convective motions, y is the temperature difference between the upward

and downward convective currents, and z measures the deviation of the vertical temperature profile from a linear one, while σ, r, and b are system parameters. For our experiment, we use the standard parameter settings $\sigma = 16$, $b = 4$ and $r = 45.92$, which are known to give rise to chaos.

In the sequel, the success of the dynamic reconstruction of the Lorenz system from experimental data will be primarily judged by the degree to which its reproduces the following characteristics, i.e., the *chaotic* or *dynamic invariants*, of the original system. Our treatment of the material follows (Malinetskii, 1993) and (Tong, 1992).

correlation dimension D_2: this quantity is a specific version of the *generalized dimension* defined as

$$D_q \triangleq \frac{1}{q-1} \lim_{\epsilon \to 0} \frac{\log \sum_i p_i^q}{\log \epsilon}, \qquad q \in \mathbb{Z}^+ \tag{5.19}$$

Roughly speaking, one covers the set of orbits in the phase portrait of a given chaotic system by cubes of edge ϵ and considers the probability p_i that points of the set lie within the i-th such cube. Actually, these generalized dimensions may be applied to arbitrary compact sets so that, for example, when $q = 0$, we may compute the *Kolmogorov capacity* D_C of the familiar one-dimensional Cantor set to be $\log 2 / \log 3$ (since 2^n segments of length $1/3^n$ are needed to cover the Cantor set). The importance of the correlation dimension is that it is used in reconstruction theory to determine an appropriate input (autoregressive) order or *embedding dimension* for the reconstructed map.

Lyapunov spectrum: one of the distinguishing characteristics of a chaotic system is its *sensitive dependence on initial conditions*, i.e., divergent trajectories resulting from arbitrarily close initial states $\boldsymbol{x}(0)$ and $\boldsymbol{x}'(0)$. In the continuous-time case, set $\boldsymbol{d}(t) \triangleq \boldsymbol{x}'(t) - \boldsymbol{x}(t)$ and define the (first) *Lyapunov exponent* λ

$$\lambda(\boldsymbol{x}(0), \boldsymbol{d}(0)) \triangleq \lim_{t \to \infty} \lim_{\|\boldsymbol{d}(0)\| \to 0} \frac{1}{t} \log \frac{\|\boldsymbol{d}(t)\|}{\|\boldsymbol{d}(0)\|} \tag{5.20}$$

More generally, one may define the d Lyapunov exponents or spectrum $\{\lambda_i\}_{i=1}^d$ of a d-dimensional discrete-time dynamical system $\boldsymbol{x}(i+1) = f(\boldsymbol{x}(i))$, $i \in \mathbf{N}$, started from an initial state $\boldsymbol{x}(0)$ by

$$\lambda_i(\boldsymbol{x}(0)) \triangleq \frac{1}{n} \lim_{n \to \infty} \log a_i(n, \boldsymbol{x}(0)) \tag{5.21}$$

where $a_i(n, \boldsymbol{x}(0))$ is the magnitude of the i-th largest eigenvalue of the Jacobian matrix $Jf^{(n)}$ of $f^{(n)}$ (the n-fold composition of f) evaluated at $\boldsymbol{x}(0)$. Under suitable conditions (Oseledets, 1968), it can be shown that these exponents are independent of the initial states $\boldsymbol{x}(0)$ and $\boldsymbol{x}'(0)$ when both lie within a neighbourhood of a *strange attractor* in the phase space of the chaotic system, where an attractor is considered "strange" if it has at least one positive λ_i. Roughly speaking, for points inside a strange attractor, the *distance* between the trajectories from two infinitesimally close initial states grows with time step i as $\exp(\lambda i) = \exp(\lambda_1 i)$, while the *area* of the triangle defined by the trajectories from three infinitesimally close initial states grows as $\exp((\lambda_1 + \lambda_2) i)$, etc. Hence negative exponents are associated with dissipative systems while positive exponents indicate sensitive dependence on initial conditions.

Kaplan-Yorke dimension D_{KY}: the *Kaplan-Yorke dimension* (Kaplan and Yorke, 1978) is defined as

$$D_{KY} \triangleq M + \frac{\sum_{i=1}^{M} \lambda_i}{|\lambda_{M+1}|} \tag{5.22}$$

where $M \in \mathbb{N}$ is the largest value such that $\sum_{i=1}^{M} \lambda_i > 0$. The significance of this dimension lies in the hypothesis proposed by Kaplan and Yorke (1978) that the Kolmogorov capacity of a strange attractor is equal to its Kaplan-Yorke dimension, i.e., $D_C = D_{KY}$. As noted in (Malinetskii, 1993), this hypothesis has been verified for a number of continuous-time dynamical systems (up to $d = 3$) and discrete-time dynamical systems (up to $d = 2$). The hypothesis, if proven to hold generally, would provide a fundamental link between the *dynamical* properties (as embodied by its Lyapunov spectrum) of a strange attractor and its *topological* properties (as embodied by its Kolmogorov capacity).

Because these invariants are intrinsic characteristics of the chaotic system and are therefore present in any sample of its realization, they may be considered *global* or *long-term* measures of a given reconstruction. On a *local* or *short-term* basis, an important measure of the quality of a given reconstruction is its (local or short-term) *predictability*. The presence of one or positive Lyapunov exponents implies that as long as the estimation of an appropriate time-delay embedding from $\boldsymbol{x}_m(i)$ to $\boldsymbol{x}(i+1)$ is only approximate, i.e., the estimated map $\tilde{f}$ does not *exactly* map $\boldsymbol{x}_m(i)$ to $\boldsymbol{x}(i+1)$ over *all* of the attractors in phase space, then there will be an exponential divergence in the evolutions of the original and the reconstructed signals

when initialized with the same value. In their studies of the local prediction problem for chaotic systems, Farmer and Sidorowich (1987) provide the approximate recursive relation

$$T \approx \frac{1}{\lambda_{max}} \log\left(\frac{\sigma_T}{\sigma_0}\right) \tag{5.23}$$

to define the *horizon of predictability* T in terms of the largest Lyapunov exponent of the chaotic process. In the formula, the parameter σ_T is the normalized standard deviation of the prediction error at time T of the r.f.i. prediction process and is defined by

$$\sigma_T^2 = \frac{\frac{1}{T}\sum_{j=0}^{T-1}\left(\tilde{x}(i+j) - x(i+j)\right)^2}{\frac{1}{N}\sum_{j=1}^{N}\left(x(j) - \mu_x\right)^2} \tag{5.24}$$

where N is the total number of the time series samples and μ_x is the mean value of the samples over the observation period. The parameter σ_0 is the normalized standard deviation of the prediction error at the start of the r.f.i. prediction, i.e., at time step i. For noise-free data, σ_T^2 should (in principle) be invariant to the starting point i for the r.f.i. prediction (since the Lyapunov exponents are global characteristics of a chaotic system), hence T should also be independent of the starting point i. In practice, the observed horizon of predictability after reconstruction from simulated data varies somewhat depending on the initial point but manifestation of the phenomenon remains clear in results presented below.

For our choice of system parameters, the corresponding theoretical values of the invariants for the system are listed in Table 5.6. For the purposes of simulation, we nu-

Parameter	D_2	D_{KY}	Lyapunov exponents (nat/s)		
Theoretical value	2.07	2.07	1.50	0.00	−22.50

Table 5.6: Dynamical invariants for Lorenz system, $\sigma = 16$, $b = 4$, $r = 45.92$

merically solve the Lorenz system equations by 4-th order Runge-Kutta integration with a time step of 0.025s and the initial conditions $x(0) = y(0) = z(0) = 1.0$ to obtain a 40000 long signal sequence, of which the first 5000 samples are removed to avoid signal transients (for a sample of the x-component of the generated data, see Figure 5.10). Because the Takens' embedding theorem implies that one ought to be able to reconstruct the attractor structure of a chaotic system by observing any one system variable, we concentrate on the x-component of the Lorenz system in our studies. In particular, it follows from Takens'

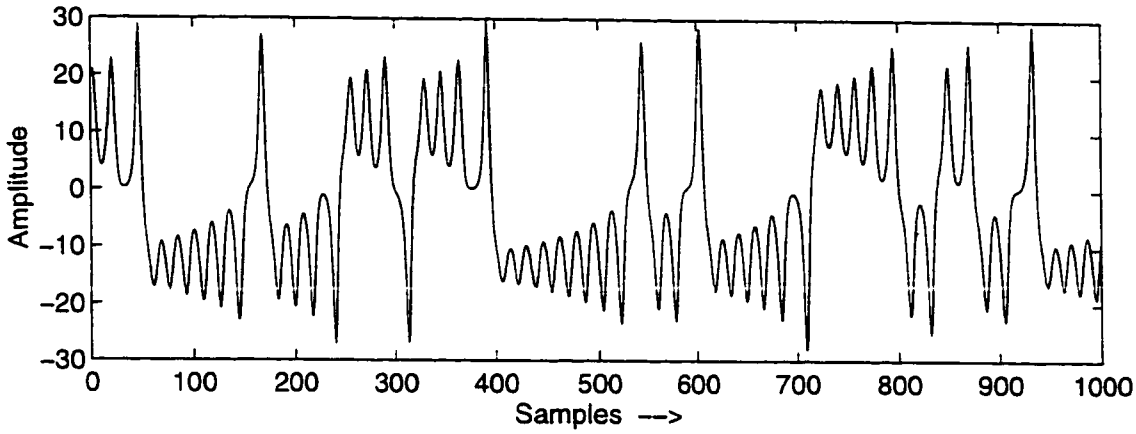

Figure 5.10: Sample of simulated Lorenz x-component ($\Delta t = 0.025$s)

embedding theorem that, under suitable conditions, a "smooth" function exists to map $x(i+1)$ from its associated *delayed input vector* $\boldsymbol{x}_p(i) \triangleq [x(i), x(i-1), \ldots, x(i-p+1)]^\top$, i.e., the observed system variable is a (deterministic) autoregressive process. Dynamic reconstruction by the estimation of such a map is known as the *delay co-ordinate embedding* method (Sauer et al., 1991). In this method, the input order p required is determined by the relation

$$p \geq d_E \cdot \tau \tag{5.25}$$

where d_E is the *embedding dimension* and τ is the *characteristic time delay* of the (continuous-time) chaotic system. These two parameters are estimated from simulated time series described above using the *global false nearest neighbours (GFNN)* method (Kennel et al., 1992) and the *mutual information (MI)* method (Fraser and Swinney, 1986; Fraser, 1989; Pineda and Sommerer, 1994), respectively. For the noise-free Lorenz time series, we find that $d_E = 3$ and $\tau = 4$s, hence for the SIRBFN, an input dimension of $p = 12$ is sufficient; for the noisy data considered later, different d_E and τ will be necessary, so p will vary. Allowing for this variation in p, the SIRBFN design parameters common to this and the subsequent experiments are given in Table 5.7. Since the networks in the following studies are implemented with the same software as for the nonlinear speech prediction experiment, details of the parameters can be found in Section 4.6. With these design parameters, the resultant regularized SIRBFN is repeatedly trained and operated in the r.f.i. predictive mode

Parameter	Value
No. of centres n	400
Input dimensionality p	12 (noise-free and 30dB) or 20 (20 and 25dB),
Basis function K	$K(r) = \exp(-r^2/2)$
Norm weighting matrix U_n	$U_n^{-1} = \text{diag}\left[m\sigma_j^2\right]_{j=1}^{m}$, σ_j^2 is sample variance of the j-th input variable over t_n.
Regularization parameter λ_n	$\min_{j=1,2,\ldots,N_\lambda} GCV(\lambda_j, t_n)$ (see (1.38)), λ_j is j-th of $N_\lambda = 100$ logarithmically spaced values from $\lambda_{min} = 10^{-14}$ to $\lambda_{max} = 0.01$

Table 5.7: SIRBFN design parameters for dynamic reconstruction of the Lorenz system

beginning at various time steps in the data. Each repetition consists of selecting n sequential sample data of the time series from, say, time step $i - n + 1$ to i as the training set and then generating a r.f.i. predicted sequence estimating the actual time series at time steps $i + 1, i + 2, \ldots, i + M$ for some $M > 0$. Figure 5.11 shows such a r.f.i. predicted sequence beginning at time step $i + 1 = 419$. The predicted sequence follows closely the original sequence up to approximately 300 samples before diverging from it. To visually compare more global structures, we may plot $x(i)$ versus $x(i - 4)$ to produce two-dimensional phase portrait (the choice of four for the delay is not significant except for yielding a relatively simple phase portrait). The agreement between the resultant attractor phase portraits of the original and reconstructed systems in Figure 5.12 are further evidence of a successful reconstruction. The dynamic invariants for the original and reconstructed systems are also compared in Table 5.5.1. For this purpose, the reconstructed system is used to generate a 35000-long sample sequence in the r.f.i. predictive mode to be analyzed as follows:

1. the correlation dimension D_2 is estimated by the maximum likelihood-based algorithm described in (Schouten et al., 1994). In the table, this estimate is denoted D_{ML}.

2. the Lyapunov spectrum is estimated by the method of (Brown et al., 1991). Compu-

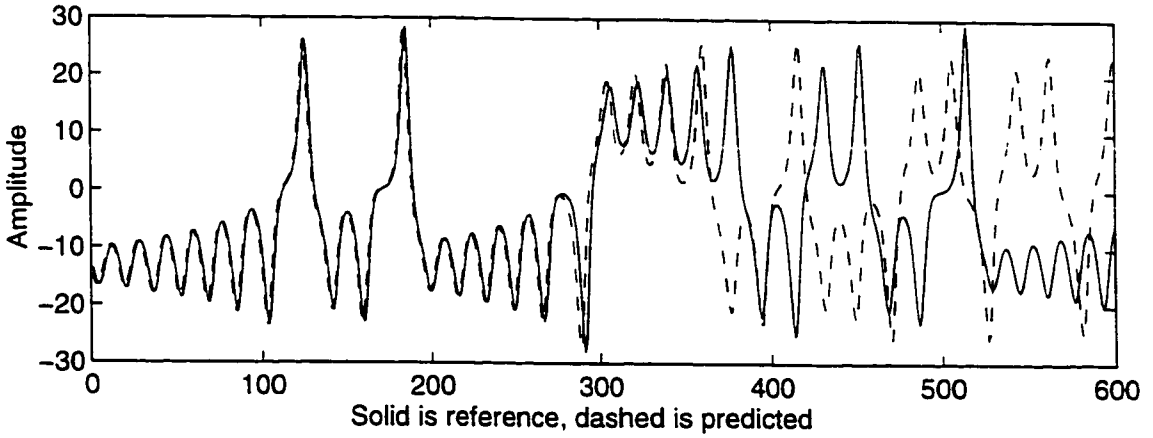

Figure 5.11: Example of RFI predicted sequence for simulated Lorenz x-component, noise-free case

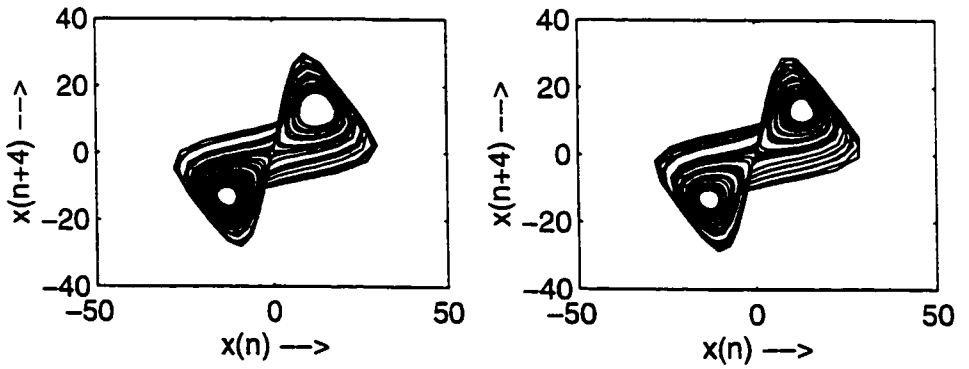

Figure 5.12: Actual (left) and reconstructed (right) attractors projected onto x-y plane, noise-free case

tationally, this algorithm exploits a recursive QR-decomposition of the Jacobian of a function which maps the state variable $\boldsymbol{x}(i)$ to $\boldsymbol{x}(i+n)$, $n > 0$, for different delays n, which, as we have discussed, is central in the definition of Lyapunov exponents.

3. the estimated Kaplan-Yorke dimension is computed from the estimated Lyapunov spectrum according to (5.22). Assuming that the corresponding conjecture holds, the estimated Kaplan-Yorke dimension should approximately equal the estimated correlation dimension.

The table also lists the estimated *number of local dimensions* or degrees of freedom, as determined by the *local false nearest-neighbours (LFNN)* method (Abarbanel and Kennel, 1993). Because this figure should agree with the actual number of Lyapunov exponents, it is useful as an additional verification of the simulated data quality and subsequent reconstruction. The close agreement between the estimated chaotic invariants, i.e., long-term

Time Series	d_E	d_L	τ	D_{ML}	D_{KY}	Lyapunov exponents (nats/s)			Avg. hor.
						λ_1	λ_2	λ_3	of pred.
Original	3	3	4	2.07	2.07	+1.5697	-0.0314	-22.3054	99.69
Reconstructed	3	3	4	2.11	2.07	+1.6314	-0.0407	-21.5878	95.92

Table 5.8: Long and short-term measures for actual and predicted Lorenz x-component, noise-free case

measures, indicated in Table 5.5.1 demonstrates that the reconstruction has truly captured the dynamics of the system underlying the original simulated data. The last column of the table also gives the average horizon of predictability as determined from the estimated Lyapunov spectrum by the BBA algorithm of (Grassberger, 1990). As we noted earlier, the observed horizon of predictability can fluctuate with the initial predicted point; in this regard, Figure 5.11 shows one of the longest horizons observed when repeating the training and r.f.i. prediction process at different points in the sample and should be considered exceptional in this respect.

5.5.2 Results for 30dB SNR Case

For a more challenging and realistic problem, we repeat the previous experiments for the simulated Lorenz x-component after the addition of white Gaussian noise. This white Gaussian noise is physically generated using a commercially available analog noise generator (NC 1107A-1, Noise Com Inc.) coupled with an amplifier and A/D converter. This device contains a hermetically-packaged noise diode that has been burned-in for 168 hours and operates in a temperature range of −35 to +100°C to produce white Gaussian noise at power level of +13dBm over a frequency band between 100Hz and 100MHz. Hence the noise sequence added to the original Lorenz data avoids the issues of undesirable correlation and submaximal period that mark noise sequences derived from typical pseudo-random number generators, e.g., that in the computational language system MATLAB™.

For the most part, the SIRBFN design parameters in the present noisy case can be the same as for the previous noise-free case, except for the crucial input autoregressive order p. Analysis of the noisy time series by the same methods used in the noise-free case return estimates of $d_E = 5$ and $\tau = 4$, indicating that p should be at least 20 by (5.25). On the other hand, that relation is derived for noise-free chaotic time series and it is known that using more inputs than necessary can allow the noise to adversely affect the quality of the reconstruction (as farther lagged inputs contribute more "noise" than "information" to the reconstruction). We therefore choose $p = 20$, the minimum permissible input order, for the noisy reconstruction as indicated in Table 5.7. As we shall see, the results obtained suggest that this choice is reasonable.

Using the same training method in the noise-free case, we obtain Figure 5.13 showing the the r.f.i. predicted sequence beginning at time step $i + 1 = 612$. In this case, the r.f.i. predicted sequence successfully tracks the original, i.e., *noise-free*, sequence for approximately 230 samples before deviating. We should note, as before, that this r.f.i. sequence is one of the longer ones encountered in the course of repeating the training/prediction cycle with different starting time steps i over the available training data. The corresponding reconstructed attractor can be found in Figure 5.14 which again, visually, agrees better with the original attractor in Figure 5.12 than the same attractor for the noisy time series. These results confirm the ability of the regularized SIRBFN to recover the original dynamics of the system from a noisy observed sequence. We can give an example of how important regularization is to reconstruction from noisy time series by comparing two r.f.i.

predicted sequences corresponding to two SIRBFNs trained according to the parameters of Table 5.7 with the same 25dB SNR sample sequence. Specifically, Figure 5.15 displays the r.f.i. predicted sequence again beginning from time step $i + 1 = 612$ produced by a nonregularized, i.e., $\lambda = 0$, SIRBFN while Figure 5.16 shows the same sequence when the SIRBFN is regularized through the usual GCV-based method listed in Table 5.7. The inability of the nonregularized SIRBFN to discern the true underlying dynamics from the noisy data is particularly graphic in this example. The dynamic invariants and the hori-

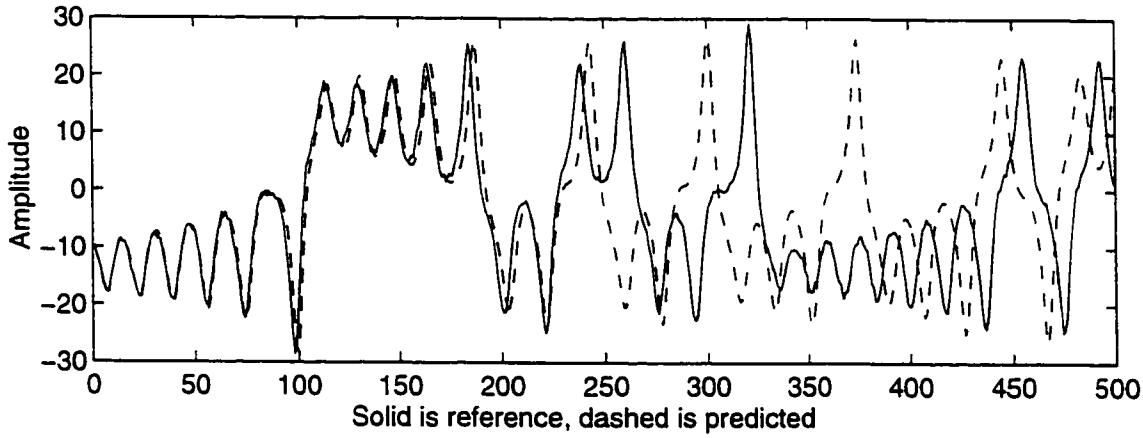

Figure 5.13: Example of RFI predicted sequence for simulated Lorenz x-component, 30dB SNR case

zon of predictability for this 30dB SNR case are given in Table 5.5.2, where the figures are computed in the same fashion as in the noise-free case. Note that even at the relative high 30dB SNR level, the added noise is sufficient to mislead greatly the estimation algorithms for the chaotic invariants previously described, e.g., the first Lyapunov exponent for the noise-corrupted signal is estimated to be 10.43 compared to 1.57 for the original (noise-free) signal. The added noise also allows the Lyapunov spectrum estimation algorithm to detect a spurious fourth Lyapunov exponent of -45.00 which is not theoretically present in the original Lorenz system. In comparison, the regularized SIRBFN reconstruction "denoises" sufficiently the noisy signal both to yield a r.f.i. predicted sequence whose estimated first Lyapunov exponent of 2.17 is much closer to that of the original signal and to avoid the introduction of an extraneous fourth Lyapunov exponent. As for the average horizon of pre-

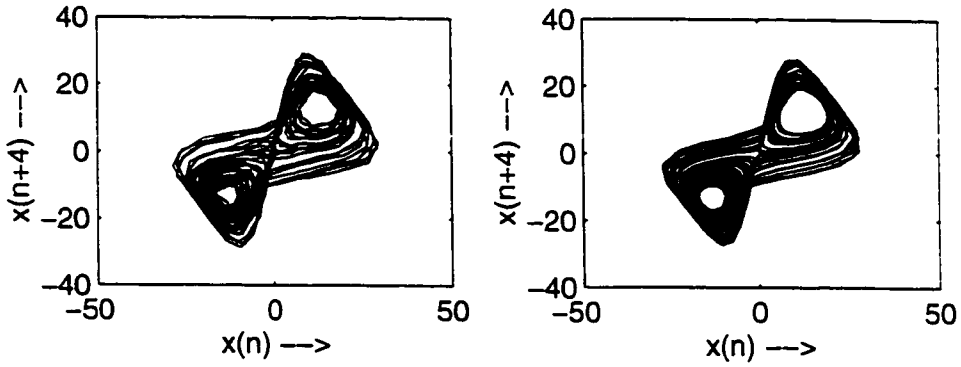

Figure 5.14: Noisy (left) and reconstructed (right) attractors projected onto x-y plane, 30dB SNR case

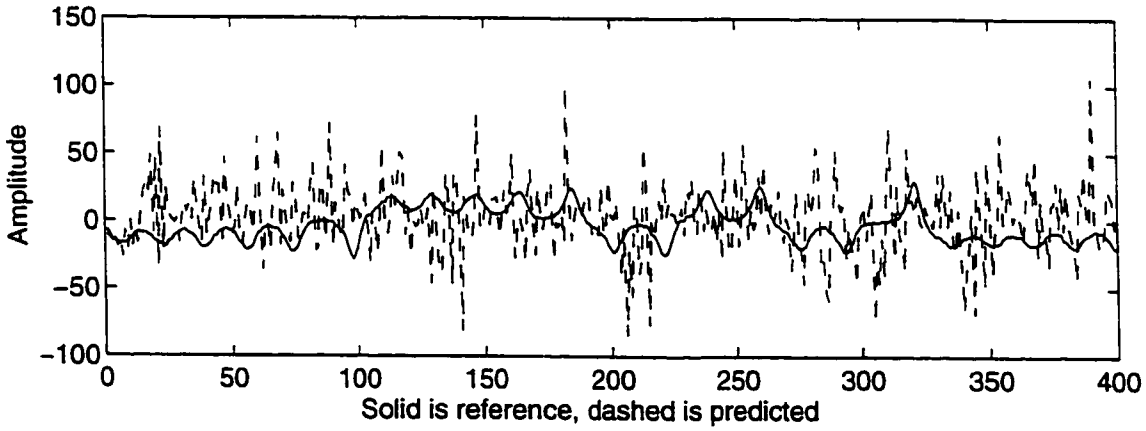

Figure 5.15: Example of RFI predicted sequence for simulated Lorenz x-component, 25dB case with no regularization

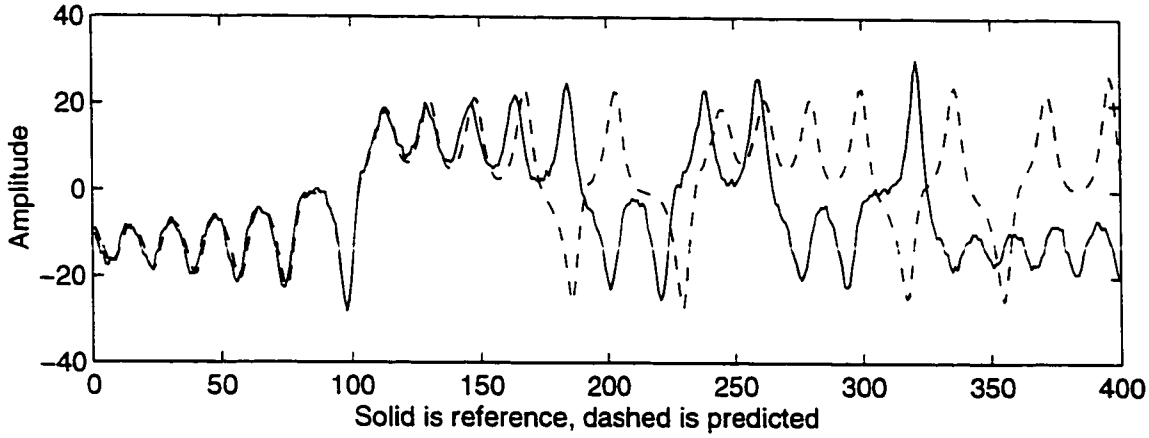

Figure 5.16: Example of RFI predicted sequence for simulated Lorenz x-component, 25dB case with regularization

dictability, here again the reconstructed signal gives an estimate of 72 samples which show less deterioration from the original horizon of 99 samples than the 15 samples estimated from the noisy signal. These results too may be taken as further signs of the efficacy of the regularized SIRBFN for noisy reconstruction problems.

Time Series	d_E	τ	D_{ML}	D_{KY}	Lyapunov exponents (nats/sec.)				Avg. hor.
					λ_1	λ_2	λ_3	λ_4	of pred.
30dB Signal	4	4	2.25	2.83	+10.4287	+0.9483	-13.5706	-45.0007	15.01
Reconstructed	3	4	2.02	2.09	+2.1667	-0.4793	-18.1266	-	72.22

Table 5.9: Long and short-term measures for noisy and predicted Lorenz x-component, 30dB SNR case

5.5.3 Results for 20dB SNR Case

Here we repeat the previous noisy reconstruction experiment except that the observed noisy sample data has a reduced SNR of 20dB and the SIRBFN design parameters are set as indicated in Table 5.7. One of the better tracking r.f.i. predicted sequences from the resultant regularized SIRBFN estimate beginning at time step $i+1=422$ can be seen

in Figure 5.17. Despite the increased noise level, the reconstructed system can still yield a r.f.i. predicted sequence which corroborates with the corresponding sequence from the original time series up to approximately 150 samples. The smoothing effect of the regularized SIRBFN reconstruction is clear in Figure 5.18, where the associated reconstructed attractor is shown beside the same attractor derived from the noisy time series. Qualitatively, the reconstructed attractor still exhibits a definite similarity to the original, noise-free attractor in Figure 5.12. The same can be said for the long-term dynamical invariants estimated for

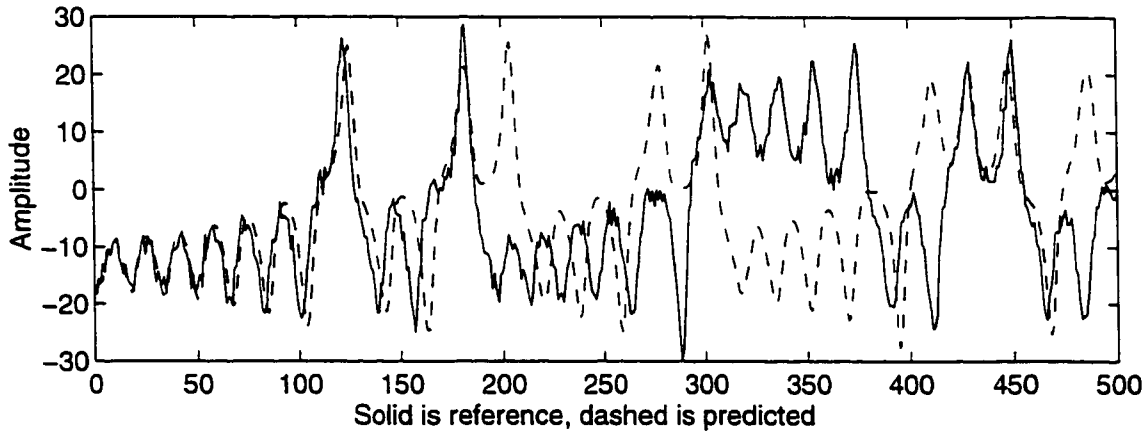

Figure 5.17: Example of RFI predicted sequence for simulated Lorenz x-component, 20dB SNR case

the reconstructed system as listed in Table 5.5.3. As in the 30dB SNR case, we observe that when analyzed with the standard chaotic parameter estimation algorithms, the noisy time series returns values which are far from the true values of the underlying noise-free series. In particular, the presence of noise adds to the divergence of the local trajectories, resulting in a grossly overestimated first Lyapunov exponent of 16.60 for the noisy time series vs. 1.57 for the original (noise-free) time series. Again, we also see that the added noise results in an estimated Lyapunov spectrum with extraneous exponents, except that in this case there is an additional minimum exponent $\lambda_5 = -46.7312$ (not listed in Table 5.5.3 because of space constraints). The reconstructed system, however, suffers from neither of these problems, although the accuracy of the estimated invariants is understandably decreased at this relative low SNR. We may therefore conclude that dynamic reconstruction using the

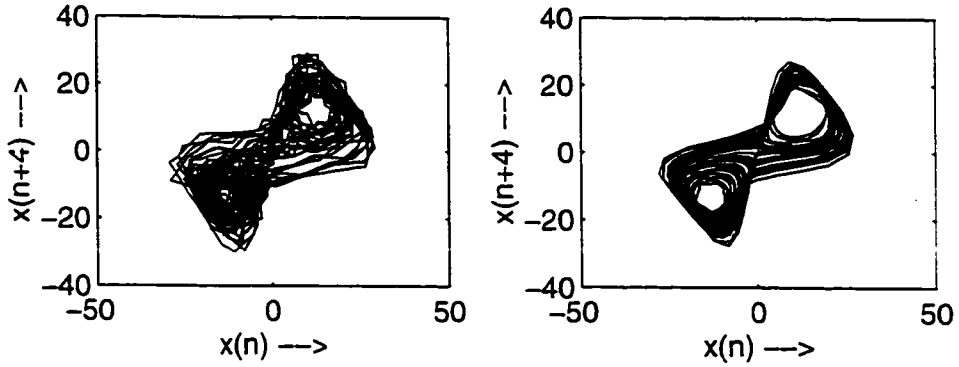

Figure 5.18: Noisy (left) and reconstructed (right) attractors projected onto x-y plane, 20dB SNR case

regularized SIRBFN remains a viable method even in the presence of a moderate amount of noise.

Time Series	d_E	τ	D_{ML}	D_{KY}	Lyapunov exponents (nats/sec.)				Avg. hor. of pred.
					λ_1	λ_2	λ_3	λ_4	
20dB signal	5	4	3.15	4.15	+16.5982	+8.4169	-2.1425	-15.6714	9.42
Reconstructed	3	4	2.09	2.12	+2.6379	-0.2902	-19.8489	-	59.32

Table 5.10: Long and short-term measures for noisy and predicted Lorenz x-component, 20dB SNR case

5.5.4 Discussion

The results of the simulations demonstrate that the regularized SIRBFN (with regularization parameter chosen via an a.o. procedure) is an effective tool for the reconstruction of chaotic dynamical systems from its observed time series, even in the presence of moderate amounts of noise. Here, the quality or success of the reconstruction is judged according to both qualitative factors, viz., the agreement (up to the horizon of predictability) between short-term r.f.i. predicted sequences compared to the original sequence starting at the same point and their corresponding long-term attractors, as well as quantitative factors, viz., the

agreement between the dynamical invariants of the reconstructed and original system as estimated from the r.f.i. sequences they generate. Although not discussed here, we ought to mention that these studies have been extended to dynamic reconstruction for the chaotic *sea clutter* process, i.e., the process responsible for the signals associated with radar backscatter from the ocean surface, where again high quality reconstruction is obtained (Haykin et al., 1997). In that case, we have an example of dynamic reconstruction from noisy observations of a real-life process for which there is not (as yet) a definitive mathematical model. It is in such contexts that the flexible nonparametrics behind the regularized SIRBFN method can arguably find greatest application.

Despite the success of the regularized SIRBFN method as a tool for dynamic reconstruction in the experiments conducted, many open questions remain as raised in the discussion of Section 5.4.1. Foremost amongst these questions may be those concerning the stability of the reconstructed system in the r.f.i. predictive mode. While this stability was not an issue for the simulated Lorenz data, a stable reconstruction of the considerably more complex (as measured by the correlation coefficient D_2) sea clutter process did require careful selection of the SIRBFN design parameters through some trial-and-error procedure. A straightforward approach to the stability issue is to constrain the SIRBFN parameters, specifically the weights and the regularization parameter, so the that resultant map, when used recurrently in the r.f.i. predictive mode, is a (globally) contractive map (e.g., see (Steck, 1992)). Unfortunately, this approach cannot allow reproduction of more complex attractor structures such as the quasi-periodic orbits seen in the Lorenz examples. Clearly more refined techniques are necessary to achieve stable, high quality reconstruction from observational data, noisy or otherwise.

Chapter 6

Concluding Remarks

In this concluding chapter, we review the results presented in the thesis from a broader perspective to place them in the context of current and past developments. Along the way, we offer a few remarks on the positive and negative aspects of our approach in the thesis along with discussions on its possible extension.

Since the resurgence of interest in ANNs, the RBFN has stood alongside the MLP as one of the paradigms of choice in a large variety of applications. Yet the theoretical properties of the actual choices made in popular RBFN design procedures remain largely unknown, save for some general results concerning the density of certain restricted classes of RBFNs. We therefore posed the question: what practical RBFN design procedures are theoretically justified? In this thesis, for the term "theoretically justified", we confined our attention primarily to (global) mean-square consistency, as is common in engineering applications - other measures of network performance are of course possible, as we shall indicate below.

We began by examining the theoretical justification for current ANN and KRE approaches to RBFN design, i.e., selection of network parameters. In this respect, the breadth and depth of the ERM theory can rightly be considered a breakthrough compared with previous *ad hoc* attempts. While the basic ERM-approach can be provably consistent, in practice some sort of architectural constraint on the network is necessary to avoid the undesirable extremes of underfitting (excessive estimator bias) and overfitting (excessive estimator variance) for finite training sets. The SRM method addresses this problem in a principled manner by generalizing the quadratic regularizers used in regularized RBFN to impose an *a priori* structure on the candidate solution space but this leaves open the question of how such a suitable structure should be determined. On an operational level,

when nonquadratic cost functionals are specified, the SRM method typically results in a difficult optimization problem, except in special circumstances. Also the minimization of the guaranteed risk central to the SRM method is not necessarily the type of optimality desired in applications.

On the other hand, it appears as if the ANN community is not fully aware of how the large, relatively mature body of theory surrounding the KRE can be relevant to the RBFN design problem. Perhaps this lack of interest stems, in part, from the notion within some ANN circles that ANNs constitute a "special" and separate class from traditional statistical estimators. Nonetheless, the similarity between the KRE, particularly in its most prevalent form the NWRE, and the SIRBFN is obvious, leading to some of the studies mentioned in the Introduction. These studies, however theoretically intriguing, do not relate to regularized RBFNs as they are typically constructed and hence are of limited practical value. The question thus remains open: does there exist another path to consistent RBFN design other than through the ERM/SRM theory?

The affirmative answer we propose in this thesis is based on a combination of elements from KRE and penalized least-squares (PLS)/spline smoothing theory and practice. It turns out, not surprisingly, that the key is the presence of the additive quadratic regularizer with parameter λ. In conjunction with the original least-squares cost function, the resultant simple algebraic equations for the optimum weights in the strict interpolation case give a direct link to the NWRE through λ, albeit under some restrictions. It is interesting to note that the link takes the form of constructive asymptotic approximation theorems in sup-norm as well as the usual L_2-norm, suggesting that both modes of consistency could carry over from the NWRE to the regularized SIRBFN. Although this intuition can be proven correct, there is little to be gained if a regularized SIRBFN is designed to approximate asymptotically a given NWRE in either mode; after all, if it is NWRE behaviour one desires, one could just use the NWRE rather than its SIRBFN approximation.

Instead, what makes the approximation theorems nontrivial is the knowledge from PLS and spline smoothing theory concerning the optimal selection of the regularization parameter λ with respect to the risk or m.s.f.e. Using one of the asymptotically optimal (a.o.) parameter selection methods for λ, we can ensure that the m.s.f.e. of a regularized SIRBFN is (asymptotically) minimum over all other choices for λ, including the m.s.f.e. for the corresponding NWRE-equivalent SIRBFN. This result can be useful in itself, e.g., in the case where the training inputs represent important operating points, but often we are

more interested in the global m.s.e. For this purpose, we introduce a theorem relating the m.s.f.e. to the global m.s.e. under conditions compatible with those assumed for the NWRE approximation theorems. Together with the approximation results, we can conclude that the regularized SIRBFN is globally m.s.-consistent whenever the corresponding NWRE is and where the regularization parameter has been chosen a.o. The implied prescription for provably (m.s.) consistent RBFN design is therefore:

- from KRE theory, start with a m.s.-consistent NWRE design. For the RBFN, choose as the basis function the kernel/bandwidth combination of the NWRE. For simplicity, we can set one basis function for each training input datum as in the NWRE, i.e., use a strict interpolation RBFN. Note, however, that this choice is not necessary if NWRE approximation theorems can be obtained for nonstrict interpolation RBFNs. Even limiting ourselves to nonstrict interpolation methods based on selecting a subset of the available centres for basis functions, a more sophisticated analysis than in the strict case would be necessary to establish corresponding constructive approximation results. The practical advantages, of course, of nonstrict interpolation RBFNs are their lower training time and run-time complexity.

- from PLS/spline smoothing theory, choose λ via a suitable a.o. parameter selection procedure. By "suitable", we mean one which is known to be a.o. under the given network operating conditions, e.g., homoskedastic or heteroskedastic noise. As we saw in the nonlinear state estimation experiments in the Other Topics chapter, an a.o. parameter selection can be profitably used to select not only the regularization parameter λ but also the input scaling parameters that weight the relative contributions of each (covariance) normalized co-ordinate in the network input vector. The resultant input scaling factors then indicate which input variables are of lesser importance to the network output, leading to a form of input variable selection or dimensionality reduction. Given the importance of this topic in data analysis, its continued study is warranted.

It can certainly be argued that the route to provably consistent RBFN design proposed in the above is neither the most direct nor the most elegant from a theoretical standpoint; in particular, it should well be possible to produce a proof for the consistency of the regularized SIRBFN without recourse to approximating the NWRE (although the choice of regulariza-

tion parameter would still require justification). That said, there are some definite merits to our approach:

1. we can exploit the substantial knowledge that exists concerning the NWRE in the statistical regression context. Because of its simplicity, the analysis of the NWRE has yielded a comprehensive understanding of its properties under various important scenarios such as mixing processes and some forms of nonstationarity. In fact, as we have previously discussed in the thesis, the formal analyses of the properties of the (regularized strict interpolation) RBFN that are available have usually made the (rather strong) assumption of i.i.d. training data. The relaxation of this assumption can be considered one of the desirable consequences to our approach.

2. we have considerable theoretical and practical experience with a.o. parameter selection methods, the latter of which is embodied by the availability of efficient computational methods in some special cases and the numerous case studies under different conditions. This significant corpus is not something to be ignored in applications.

There are naturally some limitations that arise from our synthetic approach to a justifiable RBFN design procedure; we point some of them out here to suggest avenues for further research:

1. the strict interpolation condition, i.e., one basis function per training input datum, can result in impractically large networks for significantly-sized problems, as well as a loss of numerical stability during training which can yield low quality estimates. In principle, choosing a high degree of smoothing, i.e., λ sufficiently large, should be able to alleviate the stability problem but this choice could run counter to the m.s.f.e. optimality condition that λ decrease to zero as the number of training data (but not necessarily basis functions) goes to infinity. As we mentioned before, it should be possible to include the nonstrict interpolation case in our synthetic approach once the appropriate approximation theorems are in place.

2. because most CV theory centres around a.o. with respect to the squared-error loss and its expected value, the risk or m.s.f.e., our approach can currently only demonstrate m.s.-consistency. A direct analysis of the regularized SIRBFN would not be subject to this restriction. We should point out that some research on the properties of the GCV

estimate for the optimal λ under other loss functions has been performed. For example, Wahba (1990) reports of situations in spline smoothing where the GCV estimate for the optimal λ also (approximately) minimizes the integrated squared-error, i.e., R_2^* in (1.3) with uniform input distribution. Even these preliminary results, however, indicate that the properties (more precisely, the rate of decay) of the minimizing λ for the different loss functions vary with the exact circumstances of the problem at hand. It ought to be possible to apply other cross-validatory methods which select the regularization parameter to minimize (asymptotically) other loss functions such as the mean absolute-value, resulting in a "hybrid" least-squares estimator. We could then prove that the regularized SIRBFN with regularization parameter selected according to such a CV method is consistent in the mode specified by its corresponding loss function.

3. the (asymptotic) rate to m.s.-consistency of a regularized SIRBFN (with a.o. selection of λ) in our approach is lower-bounded by, i.e., is at least as fast as, the rate for the corresponding NWRE and upper-bounded by the rate at which the m.s.f.e. converges to the global m.s.e., which we crudely estimated in the proof of Theorem 2 to be $\mathcal{O}\left(n^{-1/d}\right)$ under only uniform boundedness and Lipschitzian assumptions on the functions concerned. We expect that significantly improved rates can be obtained by assuming additional regularity, i.e., smoothness, for the functions in the theorem. These assumptions are reasonable for RBFNs since smooth basis functions such as the Gaussian are commonly used and since it is known that the underlying function must be assumed smooth if estimation is to be possible in high-dimensional spaces.

Obviously there are many other possible developments that may be pursued, e.g., nonasymptotic estimation error bounds, but we hope that the preceding remarks indicate clearly the main boundaries of the thesis work. In the end, one could say that this thesis bears two messages: to the ANN community, that the KRE/NWRE theory has much to offer as a guide for justifiable RBFN design under a wide variety of conditions; and to the kernel regression community, that the extension provided by the presence of the regularization parameter is nontrivial, computationally feasible, and well-supported in theory and practice from the PLS/spline smoothing field. We have tried in this thesis to bring exactly these messages together and if it succeeds in influencing the thinking of either community, then the effort was well-spent.

References

Abarbanel, H. D. I. (1996). *Analysis of observed chaotic data.* Springer.

Abarbanel, H. D. I., and Kennel, M. B. (1993). Local false nearest neighbors and dynamical dimensions from observed chaotic data. *Phys. Rev. E, 47*, 3057-68.

Andrews, D. W. K. (1991). Asymptotic optimality of generalized C_L, cross-validation, and generalized cross-validation in regression with heteroskedastic errors. *Journal of Econometrics, 47*, 359-77.

Aronszajn, N. (1950). Theory of reproducing kernel Hilbert spaces. *Trans. Amer. Math. Soc., 68*. 337-404.

Auestad, B., and Tjøstheim, D. (1990). Identification of nonlinear time series: first order characterization and order determination. *Biometrika*, 77(4), 669-87.

Baker, G. L., and Gollub, J. P. (1996). *Chaotic dynamics: an introduction.* Cambridge University Press.

Barnard, E. (1992). Comments on "Bayes statistical behavior and valid generalization of pattern classifying neural networks". *IEEE Transactions on Neural Networks, 3*(6), 1026-7.

Bosq, D. (1973). Sur l'estimation de la densité d'un processus stationnaire et mélangeant. *Comptes Rendus d'Academie Sciences, Series A, Paris, 277*, 535-538.

Bosq, D. (1975). Inégalité de Bernstein pour les processus stationnaires et mélangeants. applications. *Comptes Rendus d'Academie Sciences, Series A, Paris, 281*, 1095-1098.

Bosq, D. (1996). *Nonparametric statistics for stochastic processes* (Vol. 110). New York, NY: Springer-Verlag.

Bradley, R. C. (1986). Basic properties of strong mixing conditions. In E. Eberlein and M. S. Taqqu (Eds.), *Dependence in probability and statistics* (Vol. 11, p. 165-92). Birkhauš.

Broomhead, D. S., and Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, *2*, 321-335.

Brown, R., Bryant, P., and Abarbanel, H. D. I. (1991). Computing the Lyapunov exponents of a dynamical systems from observed time series. *Phys. Rev. A*, *43*, 2787-806.

Cacoullos, T. (1966). Estimation of a multivariate density. *Ann. Inst. Stat. Math. (Tokyo)*, *18*(2), 179-189.

Casdagli, M. (1989). Nonlinear prediction of chaotic time series. *Physica D*, *35*, 335-356.

Corradi, V., and White, H. (1995). Regularized neural networks: some convergence rate results. *Neural Computation*, *7*, 1225-1244.

Craven, P., and Wahba, G. (1979). Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation. *Numer. Math.*, *31*, 377-403.

Devroye, L. (1981). On the almost everywhere convergence of nonparametric regression function estimates. *Ann. Statist.*, *9*(6), 1310-9.

Devroye, L., and Györfi, L. (1985). *Nonparametric density estimation, the L_1 view*. New York, NY: Wiley.

Devroye, L., Györfi, L., and Lugosi, G. (1996). *A probabilistic theory of pattern recognition*. Springer.

Douhkan, P. (1994). *Mixing: properties and examples* (Vol. 85). New York, NY: Springer-Verlag.

Eubank, R. L. (1988). *Spline smoothing and nonparametric regression* (Vol. 90). New York, NY: Marcel Dekker.

Farmer, J. D., and Sidorowich, J. J. (1987). Predicting chaotic time series. *Physics Review Letters*, *59*(8), 845-8.

Földes, A. (1974). Density estimation for dependent sample. *Studia Scientarium Mathematicarum Hungarica, 9*, 443-52.

Fraser, A. M. (1989). Information and entropy in strange attractors. *IEEE Trans. on Info. Theory, 35*, 245-62.

Fraser, A. M., and Swinney, H. L. (1986). Independent co-ordinates for strange attractors from mutual information. *Phys. Rev. A, 33*(2), 1134-40.

Friedlander, F. G. (1982). *Introduction to the theory of distributions.* Cambridge Univ. Press.

Gallant, A. (1987). *Nonlinear statistical models.* Wiley.

Geman, S., Bienenstock, E., and Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation, 4*, 1-58.

Glick, N. (1972). Sample-based classification procedures derived from density estimators. *J. Amer. Statist. Assoc., 67*, 116-22.

Golitschek, M. von, and Schumaker, L. L. (1990). Data fitting by penalized least squares. In J. C. Mason and M. G. Cox (Eds.), *Algorithms for approximation II* (p. 210-227). London, Great Britain: Chapman and Hall.

Grassberger, P. (1990). An optimized box-assisted algorithm for fractal dimension. *Phys. Lett. A, 148*, 63-8.

Gray, R. M. (1987). *Probability, random processes, and ergodic properties.* Springer-Verlag.

Grenander, U. (1981). *Abstract inference.* Wiley.

Guyon, I., Vapnik, V., Boser, B., Bottou, L., and Solla, S. A. (1992). Structural risk minimization for character recognition. In D. Touretzky (Ed.), *Advances in neural information processing systems 4* (p. 105-19). Morgan Kaufmann.

Györfi, L. (1978). On the rate of convergence of nearest neighbour rules. *IEEE Trans. Info. Theory*(24), 509-12.

Györfi, L., Härdle, W., Sarda, P., and Vieu, P. (1989). *Nonparametric curve estimation from time series* (Vol. 60). Heidelberg, Germany: Springer-Verlag.

Hager, W. W. (1989). Updating the inverse of a matrix. *SIAM Review, 31*(2), 221-39.

Hampshire, J. B., II, and Perlmutter, B. (1990). Equivalence proofs for multi-layer perceptron classifiers and the Bayes discriminant function. In *Proc. of the connectionist models summer school* (p. 159-72). San Mateo, CA: Morgan Kaufman.

Hand, D. (1982). *Kernel discriminant analysis* (Vol. 2). Chichester: Research Studies Press.

Härdle, W. (1990). *Applied nonparametric regression* (Vol. 19). Cambridge, UK: Cambridge University Press.

Haykin, S. (1996). *Adaptive filter theory* (3rd ed.). Englewood Cliffs, NJ: Prentice-Hall.

Haykin, S., Puthusserypady, S., and Yee, P. (1997). *Reconstruction of underlying dynamics of an observed chaotic process* (CRL Report No. 353). McMaster University, Hamilton, ON: Communications Research Laboratory.

Haykin, S., Sayed, A. H., Zeidler, J., Yee, P., and Wei, P. (1997). Adaptive tracking of linear time-variant systems by extended RLS algorithms. *IEEE Trans. Signal Processing, 45*(5), 1118-28.

Haykin, S., Yee, P., and Derbez, E. (1997). Optimum nonlinear filtering. *IEEE Trans. Signal Processing, 45*(11), 2774-2786.

He, X., and Lapedes, A. (1993). Successive approximation radial basis function networks for nonlinear modeling and prediction. In *Proc. IJCNN* (Vol. 2, p. 1997-2000). Nagoya, Japan.

Hutchinson, J. M. (1994). *A radial basis function approach to financial time series analysis.* Ph.D. thesis, Dept. of EECS, MIT.

Kadirkamanathan, V., and Kadirkamanathan, M. (1996). Recursive estimation of dynamic modular RBF networks. In *Advances in neural information processing systems* (Vol. 8, p. 239-45). San Mateo, CA: Morgan Kaufman.

Kadirkamanathan, V., and Niranjan, M. (1993). A function estimation approach to sequential learning with neural networks. *Neural Computation, 5*, 954-975.

Kanaya, F., and Miyake, S. (1991). Bayes statistical behavior and valid generalization of pattern classifying neural networks. *IEEE Transactions on Neural Networks*, *2*(4), 471-5.

Kaplan, K., and Yorke, J. (1978). Functional differential equations and the approximation of fixed points. In *Lecture notes in mathematics* (Vol. 730, p. 228-237). Springer-Verlag.

Kennel, M. B., Brown, R., and Abarbanel, H. D. I. (1992). Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Phys. Rev. E*, *45*, 3403-11.

Krzyżak, A., Linder, T., and Lugosi, G. (1996). Nonparametric estimation and classification using radial basis functions. *IEEE Trans. Neural Networks*, *7*(2), 475-87.

Li, K. C. (1985). From Stein's unbiased risk estimates to the method of generalized cross validation. *Ann. Statist.*, *13*(4), 1352-77.

Li, K. C. (1987). Asymptotic optimality for C_p, C_L, cross-validation and generalized cross-validation: discrete index set. *Ann. Statist.*, *15*(3), 958-75.

Light, W. A. (1992). Some aspects of radial basis function approximation. In S. P. Singh (Ed.), *Approximation theory, spline functions and applications* (Vol. 356, p. 163-190). Dordrecht, Netherlands: Kluwer.

Lo, J. T. (1994). Synthetic approach to optimal filtering. *IEEE Transactions on Neural Networks*, *5*(5), 803-11.

Lo, J. T. (1995). *Neural network approach to optimal filtering* (Tech. Rep. No. RL-TR-94-197). Griffiss Air Force Base, NY: Rome Laboratory, Air Force Materiel Command.

Lorenz, E. N. (1963). Deterministic non-periodic flows. *J. Atmos. Sciences*, *20*, 130-41.

Lowe, D. (1995). On the use of nonlocal and non positive definite basis functions in radial basis function networks. In *Proc. IEE ANN* (p. 206-211). Cambridge, UK.

Lowe, D., and M^{c}Lachlan, A. (1995). Modelling of nonstationary processes using radial basis function networks. In *Proc. IEE ANN* (p. 300-305). Cambridge, UK.

Malinetskii, G. G. (1993). Synergetics, predictability, and deterministic chaos. In Y. A. Kratsov (Ed.), *Limits of predictability* (Vol. 60, p. 75-141). Springer-Verlag.

Maria, F. D. de, and Figueiras-Vidal, A. R. (1995). Nonlinear prediction for speech coding using radial basis functions. In *Proc. ICASSP* (Vol. 1, p. 788-791). Detroit, MI.

McLachan, G. (1992). *Discriminant analysis and statistical pattern recognition.* New York, NY: Wiley.

Nadaraya, E. A. (1964). On estimating regression. *Theor. Probab. Appl., 9*, 141-2.

Nadaraya, E. A. (1965). On nonparametric estimation of density functions and regression curves. *Theor. Probab. Appl., 10*, 186-90.

Niyogi, P., and Girosi, F. (1996). On the relationship between generalization error, hypothesis complexity, and sample complexity for radial basis functions. *Neural Computation, 8*(4), 819-42.

Oseledets, V. I. (1968). The multiplicative ergodic theory. Characteristic Lyapunov exponents for dynamical systems. *Trans. Moscow Math. Soc., 19*, 179-210.

O'Sullivan, F., Yandell, B., and Raynor, W. (1987). Automatic smoothing of regression functions in generalized linear models. *Journal of the American Statistical Association, 81*, 96-103.

Ott, E. (1993). *Chaos in dynamical systems.* Cambridge University Press.

Ott, E., Sauer, T., and Yorke, J. A. (Eds.). (1994). *Coping with chaos: analysis of chaotic data and the exploitation of chaotic systems.* Wiley.

Parisini, T., and Zoppoli, R. (1996). Neural approximation for multistage optimal control of nonlinear stochastic systems. *IEEE Transactions on Automatic Control, 41*(6), 889-95.

Park, J., and Sandberg, I. W. (1991). Universal approximation using radial basis function networks. *Neural Computation, 3*(2), 246-57.

Park, J., and Sandberg, I. W. (1993). Approximation and radial basis function networks. *Neural Computation, 5*(2), 305-16.

Parsini, T., and Zoppoli, R. (1994). Neural networks for nonlinear state estimation. *International Journal of Robust and Nonlinear Control, 4*, 231-48.

Parzen, E. (1962). On estimation of a probability density function and mode. *Ann. Math. Statist.*, *33*, 1065-76.

Perlmutter, B. A. (1989). Learning state trajectories in recurrent neural networks. *Neural Computation*, *1*, 263-9.

Pham, T. D., and Tran, L. T. (1991). Kernel density estimation under a locally mixing condition. In G. Roussas (Ed.), *Nonparametric functional estimation and related topics* (Vol. 335, p. 419-30). Kluwer.

Pineda, F. J., and Sommerer, J. C. (1994). A fast algorithm for estimating the generalized dimension and choosing time delays. In A. S. Weigend and N. A. Gershenfeld (Eds.), *Time series prediction: Forecasting the future and understanding the past* (p. 367-85).

Plutowski, M., Sakata, S., and White, H. (1994). Cross-validation estimates IMSE. In *Advances in neural information processing systems* (Vol. 6, p. 391-393). San Mateo, CA: Morgan Kaufman.

Poggio, T., and Girosi, F. (1990). Networks for approximation and learning. *Proc. IEEE*, *78*(9), 1484-1487.

Pollard, D. (1984). *Convergence of stochastic processes.* Springer-Verlag.

Principe, J. C., and Kuo, J.-M. (1995). Dynamic modelling of chaotic time series with neural networks. In G. Tesauro, D. S. Touretzky, and T. Leen (Eds.), *Advances in neural information processing systems* (Vol. 7, p. 311-318). Cambridge, MA: MIT Press.

Principe, J. C., Rathie, A., and Kuo, J.-M. (1992). Prediction of chaotic time series with neural networks and the issue of dynamic modelling. *International Journal of Bifurcation and Chaos*, *2*(4), 989-996.

Richard, M. D., and Lippmann, R. P. (1991). Neural network classifiers estimate Bayesian a posteriori probabilities. *Neural Computation*, *3*, 461-83.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*, 465-471.

Rosenblatt, M. (1956). Remarks on some nonparametric estimates of a density function. *Ann. Math. Statist.*, *27*, 832-837.

Ruck, D., Rogers, S., Kabrisky, M., Oxley, M., and Suter, B. (1990). The multilayer perceptron as an approximation to a Bayes optimal discriminant function. *IEEE Transactions on Neural Networks*, *1*(4), 296-8.

Rutkowski, L. (1985a). Nonparametric identification of quasi-stationary systems. *Systems and Controls Letters*, *6*, 33-5.

Rutkowski, L. (1985b). Real-time identification of time-varying systems by non-parametric algorithms based on Parzen kernels. *Internation Journal of Systems Science*, *16*(9), 1123-30.

Sauer, T., Yorke, J. A., and Casdagli, M. (1991). Embedology. *Journal of Statistical Physics*, *65*(3/4), 579-616.

Scargle, J. D. (1992). Predictive deconvolution of chaotic and random processes. In D. Brillinger, P. Caines, J. Geweke, E. Parzen, M. Rosenblatt, and M. S. Taqu (Eds.), *New directions in time series analysis, part I* (Vol. 45, p. 335-56). Springer-Verlag.

Schouten, J. C., and Bleek, C. M. van den. (1994). *RRCHAOS Time Series Analysis Software*. Delft University of Technology, Delft, Netherlands.

Schouten, J. C., Takens, F., and Bleek, C. M. van den. (1994). Estimation of the dimension of a noisy attractor. *Phys. Rev. E*, *50*, 1851-61.

Schroeder, M. R., and Atal, B. (1985). Code-excited linear prediction (CELP): high quality speech at very low bit rates. In *Proc. ICASSP* (p. 937). Tampa, FL.

Steck, J. E. (1992). Convergence of recurrent networks as contraction mappings. In *Proc. IJCNN* (Vol. 3, p. 462-467). Baltimore, MD.

Stinchombe, M., and White, H. (1990). Approximating and learning unknown mappings using multilayer feedforward neural networks with bounded weights. In *Proc. IJCNN* (Vol. 3). Detroit, MI.

Stone, C. J. (1977). Consistent nonparametric regression. *Ann. Statist.*, *5*(4), 595-645.

Sun, M., and Glowinski, R. (1993). Pathwise approximation and simulation for the zakai filtering equation through operator splitting. *Calcolo*, *30*, 219-39.

Takens, F. (1981). Detecting strange attractors in turbulence. In *Dynamical systems and turbulence* (Vol. 898, p. 366-81). Springer-Verlag.

Terano, T., Asai, K., and Sugeno, M. (1992). *Fuzzy systems theory and its applications.* Boston: Academic Press.

Theiler, J., Eubank, S., Longtin, A., Galdrikian, B., and Farmer, J. D. (1992). Testing for nonlinearity in time series: the method of surrogate data. *Physica D, 58*, 77-94.

Tikhonov, A. N., and Arsenin, V. (1977). *Solutions of ill-posed problems.* Washington: Winston.

Tong, H. (1990). *Non-linear time series: a dynamical systems approach.* Oxford Science.

Tong, H. (1992). Contrasting aspects of non-linear time analysis. In D. Brillinger, P. Caines, J. Geweke, E. Parzen, M. Rosenblatt, and M. S. Taqu (Eds.), *New directions in time series analysis, part I* (Vol. 45, p. 357-70). Springer-Verlag.

Townshend, B. (1991). Nonlinear prediction of speech. In *Proc. ICASSP* (Vol. 1, p. 425-428). Toronto, Canada.

Van Ryzin, J. (1966). Bayes risk consistency of classification procedures using density estimates. *Sankhyā A, 28*, 161-70.

Vapnik, V. (1982). *Estimation of dependences based on empirical data.* Springer-Verlag.

Vapnik, V. (1992). Principles of risk minimization for learning theory. In D. Touretzky (Ed.), *Advances in neural information processing systems 4* (p. 831-838). Morgan Kaufmann.

Villalobos, M., and Wahba, G. (1987). Inequality-constrained multivariate smoothing splines with application to the estimation of posterior probabilities. *Journal of the American Statistical Association, 82*(397), 239-48.

Wahba, G. (1990). *Spline models for observational data* (Vol. 59). SIAM.

Watson, G. S. (1964). Smooth regression analysis. *Sankhyā A, 26*, 359-72.

White, H. (1989). Learning in artificial neural networks: a statistical perspective. *Neural Computation, 1*, 425-64.

White, H. (1990). Connectionist nonparametric regression: multilayer feedforward networks can learn arbitrary mappings. *Neural Networks*, *3*, 535-50.

Xu, L., Krzyżak, A., and Yuille, A. (1994). On radial basis function nets and kernel regression: statistical consistency, convergence rates, and receptive field size. *Neural Networks*, *7*(4), 609-628.

Yee, P. (1992). *Classification experiments involving backpropagation and radial basis function networks* (CRL Report No. 249). McMaster University, Hamilton, ON: Communications Research Laboratory.

Yee, P., and Haykin, S. (1995). A dynamic, regularized Gaussian radial basis function network for nonlinear, nonstationary time series prediction. In *Proc. ICASSP* (p. 3419-3422). Detroit, MI.

Yee, P., and Haykin, S. (1998). A dynamic, regularized radial basis function network for nonlinear, nonstationary time series prediction. *IEEE Trans. Signal Processing*, accepted for publication.