ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 273-278

Selecting Among Candidate Basis Functions by Crosscorrelation

Andreas Poncet*, Armin Deiss, Schmuel Holles

Institute for Signal and Information Processing
Swiss Federal Institute of Technology ETH, CH-8092 Zurich, Switzerland

Abstract. The problem of determining the "best" functional form for a model is ill-posed because of the always finite amount of data available. Hence the simplified problem of finding "good" basis functions (among candidates) in a stationary environment is addressed. Candidate basis functions can be generated by standard optimization techniques. A suboptimum simple method to select among these candidates is presented and tested with a filtering application.

1. Introduction

1.1. Models and estimation rules

Consider the problem of building a mathematical model of an input-output system $(\boldsymbol{X}(\cdot), Y(\cdot))$ (joint stochastic process) from a realization of the data set

$$\boldsymbol{Z}^n \triangleq \big\{\big(\boldsymbol{X}(1), Y(1)\big), \ldots, \big(\boldsymbol{X}(n), Y(n)\big)\big\}. \tag{1}$$

One of the main issues is how to select in practice an appropriate form for the mapping (in fact the estimation rule) $f : \mathbb{R}^m \to \mathbb{R}$, $\boldsymbol{X} \longmapsto \hat{Y} = f(\boldsymbol{X})$, where $\hat{Y}$ denotes the prediction of Y. From a pragmatic point of view, one would like to obtain a parsimonious approximation to the optimum estimation rule, i.e., the Bayes rule, which, in the case of the *mean squared error* (MSE)

$$\xi = \mathrm{E}\big[(\hat{Y} - Y)^2\big], \tag{2}$$

is given by the conditional mean $f_{\mathrm{Bayes}}(\boldsymbol{x}) = \mathrm{E}(Y|\boldsymbol{X} = \boldsymbol{x})$. In reality f_{Bayes} is unknown; known are only n pairs $(\boldsymbol{x}(k), y(k))$ which can be viewed as points $(\boldsymbol{x}(k), f_{\mathrm{Bayes}}(\boldsymbol{x}(k)))$ perturbed by some noise. Thus in general we must abandon the hope of finding "the best" predictive model and accept the more modest objective of finding a "good" f, i.e., one which is close to f_{Bayes}. A natural parametrization for f is an expansion in b *basis functions*, such that

$$f(\boldsymbol{x};\boldsymbol{\theta}) = c_0 + \sum_{i=1}^{b} c_i \psi_i(\boldsymbol{x}). \tag{3}$$

*On leave to: ABB Research Center, CH-5405 Baden-Dättwil, Switzerland

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 273-278

The vector $\boldsymbol{\theta}$ contains the coefficients c_i and possible adjustable parameters of the (non-linear) basis functions ψ_i. Recently, results have been published about the rate of convergence of adjustable basis functions (such as parametrized sigmoids) compared to fixed basis functions (such as polynomials) [1]. The bounds indicate a clear advantage of *adjustable* basis functions when compared to fixed basis functions. The former are less susceptible to the curse of dimensionality, according to which the required number b of basis functions increases exponentially with the input space dimension m.

1.2. Model flexibility and generalization

In principle, the more basis functions used (i.e., the more flexible the model), the more precisely $f(\cdot;\boldsymbol{\theta})$ can approximate an arbitrary $f_{\text{Bayes}}(\cdot)$. One reason of a theoretical nature speaks against too flexible models though. Since we must *estimate* the parameter values from a finite set of n training data, there are two terms that account for the expected MSE of the model class $\{f(\cdot;\boldsymbol{\theta})\}$. The first term corresponds to the smallest error ξ_0 achievable by an element of the class. It corresponds to the bias and can be *reduced* by increasing the flexibility of the model class (e.g., the number of free parameters). Conversely, the second term of the expected MSE *grows* with flexibility. It corresponds to the model variance. The trade-off between bias and variance depends on the data set size n. For $n \to \infty$, the variance of the parameter estimate vanishes and the best model of the class is obtained with probability one.

The above considerations suggest the following strategy. Begin the search with basis functions that are signal-adaptable. This allows a small bias. In a second step, reduce the variance by selecting a limited subset of the adapted basis functions. Finally, one may compensate for the small increase of error by slightly readapting the retained functions.

2. Selection of basis functions among candidates

2.1. The basic idea

We propose the following method, which has been found to be sound in practice. In a first step, B candidate basis functions $\psi_1, \ldots, \psi_B$ (possibly of different type) are generated in several optimization runs. In a second step, the candidates which contribute at least to reducing the error are eliminated iteratively. This is done until a subset of b ($\ll B$) basis functions is found such that the error is only slightly larger than for the complete set.

At first sight, the computational task appears to be prohibitive, because there are initially $2^B - 1$ possible subsets. Even by avoiding an exhaustive search using the above method, we still have in principle to recalculate the best linear combination for each candidate subset. In the case of the MSE, however, this can be done very efficiently as follows.

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 273-278

2.2. A suboptimal simple algorithm

Consider a subset of b fixed basis functions and let $\boldsymbol{\psi}(\boldsymbol{x}) \triangleq [1, \psi_1(\boldsymbol{x}), \ldots, \psi_b(\boldsymbol{x})]^{\mathrm{T}}$. Define the correlation matrix $\mathbf{R}_{\boldsymbol{\Psi}}$ and the crosscorrelation vector $\boldsymbol{s}_{\boldsymbol{\Psi}Y}$ by

$$\mathbf{R}_{\boldsymbol{\Psi}} \triangleq \mathrm{E}(\boldsymbol{\psi}(\boldsymbol{X})\boldsymbol{\psi}(\boldsymbol{X})^{\mathrm{T}}) \tag{4}$$

$$\boldsymbol{s}_{\boldsymbol{\Psi}Y} \triangleq \mathrm{E}(\boldsymbol{\psi}(\boldsymbol{X})Y). \tag{5}$$

The minimum MSE achievable by the subset $\boldsymbol{\psi}$ is equal to

$$\xi_{\boldsymbol{\Psi}} = \mathrm{E}(Y^2) - \boldsymbol{s}_{\boldsymbol{\Psi}Y}^{\mathrm{T}} \mathbf{R}_{\boldsymbol{\Psi}}^{-1} \boldsymbol{s}_{\boldsymbol{\Psi}\mathbf{Y}} \tag{6}$$

(c.f. [3]). In principle the matrix $\mathbf{R}_{\boldsymbol{\Psi}}$ and the vector $\boldsymbol{s}_{\boldsymbol{\Psi}Y}$ should be recomputed for each of the $2^B - 1$ possible subsets $\boldsymbol{\psi}$. Note, though, that element (i, j) of $\mathbf{R}_{\boldsymbol{\Psi}}$ is the correlation $\mathrm{E}[\psi_i(\boldsymbol{X})\psi_j(\boldsymbol{X})]$. Thus if later we consider another candidate subset which contains the functions ψ_i and ψ_j, we need not recompute the corresponding term. In fact it is sufficient to compute the $(B+1)(B+2)/2$ different correlations once, thereby obtaining the symmetric correlation matrix $\mathbf{R}$. Then for any subset $\boldsymbol{\psi}$, the matrix $\mathbf{R}_{\boldsymbol{\Psi}}$ is immediately given by picking the corresponding submatrix from $\mathbf{R}$. The same holds for $\boldsymbol{s}_{\boldsymbol{\Psi}Y}$ and the $(B+1) \times 1$ crosscorrelation vector $\boldsymbol{s}$ of elements $\mathrm{E}[\psi_i(\boldsymbol{X})Y]$ [3].

The algorithm can be described as follows:

- *Step 0.* Collect the B candidate basis functions $\psi_1, \ldots, \psi_B$. Set $b := B$.
- *Step 1.* Using (1), compute

$$\hat{Q}_Y := \frac{1}{n} \sum_{k=1}^{n} Y^2(k) \tag{7}$$

$$\hat{\mathbf{R}}_{\boldsymbol{\Psi}} := \frac{1}{n} \sum_{k=1}^{n} \boldsymbol{\psi}(\boldsymbol{X}(k))\boldsymbol{\psi}(\boldsymbol{X}(k))^{\mathrm{T}} \tag{8}$$

$$\hat{\boldsymbol{S}}_{\boldsymbol{\Psi}Y} := \frac{1}{n} \sum_{k=1}^{n} \boldsymbol{\psi}(\boldsymbol{X}(k))Y(k). \tag{9}$$

- *Step 2.* For each of the b remaining basis functions ψ_i, do

$$\hat{\mathbf{R}}_{\boldsymbol{\Psi}}^{(i)} := \left(\hat{\mathbf{R}}_{\boldsymbol{\Psi}} \text{ without row and column of label } i\right) \tag{10}$$

$$\hat{\boldsymbol{S}}_{\boldsymbol{\Psi}Y}^{(i)} := \left(\hat{\boldsymbol{S}}_{\boldsymbol{\Psi}Y} \text{ without element of label } i\right) \tag{11}$$

$$\hat{\xi}_{\boldsymbol{\Psi}}^{(i)} := \frac{1}{n-b}\left(\hat{Q}_Y - \hat{\boldsymbol{S}}_{\boldsymbol{\Psi}Y}^{(i)\mathrm{T}} \left(\hat{\mathbf{R}}_{\boldsymbol{\Psi}}^{(i)}\right)^{-1} \hat{\boldsymbol{S}}_{\boldsymbol{\Psi}Y}^{(i)}\right). \tag{12}$$

- *Step 3.* Set $j := \arg\min_i \hat{\xi}_{\boldsymbol{\Psi}}^{(i)}$ and update $b := b - 1$,

$$\hat{\mathbf{R}}_{\boldsymbol{\Psi}} := \hat{\mathbf{R}}_{\boldsymbol{\Psi}}^{(j)}, \tag{13}$$

$$\hat{\boldsymbol{S}}_{\boldsymbol{\Psi}Y} := \hat{\boldsymbol{S}}_{\boldsymbol{\Psi}Y}^{(j)}. \tag{14}$$

If $\hat{\xi}_{\boldsymbol{\Psi}}^{(j)} < \xi_{\max}$, go to step 2, otherwise end.

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 273-278

In practice, if two candidate basis functions are (almost) identical, then a numerical problem due to the ill-conditioning of $\hat{\mathbf{R}}_{\Psi}$ may appear for the computation of the quadratic form in (12). Using the singular value decomposition (SVD) algorithm typically solves this problem or at least allows one to determine which basis functions are causing the problem.

3. Example

The illustration is a problem of filtering (noise reduction, Fig. 1) from [2]. A beamformer with two microphones, M_1 and M_2, records the acoustic environment consisting of a stationary source $S(\cdot)$ (equidistant from M_1 and M_2) and an independent jammer signal $J(\cdot)$ reaching M_2 with a certain delay after M_1. The goal of the system is to reduce the jammer noise that is superposed to the original source signal by exploiting the assymetry of the jammer with respect to the microphones.

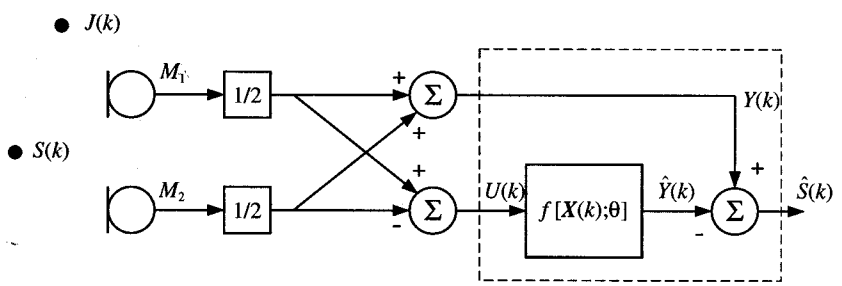

Figure 1: Beamformer for noise reduction

In this case, the target is the mean $Y(\cdot)$ of the two microphone signals. The difference $U(\cdot)$ of both signals is correlated with the noise component and can thus be used to estimate it. Depending on the probability distribution of the jammer, the Bayes rule is linear or nonlinear.

3.1. Experiment with a linear Bayes rule

First we tested our algorithm in the case of a Gaussian distributed jammer, where it can be proved that the optimum basis functions are in fact linear [2]. To see whether the algorithm could identify these optimum basis functions, the system was simulated and a data set of size $n = 5 \cdot 10^3$ was generated. From the $m = 5$-dimensional vector $\boldsymbol{X}$ (consisting of past values of U), monomials (i.e., polynomial terms) up to third degree were considered, giving a total of $B = 55$ candidate basis functions.

The results of the backward elimination algorithm are plotted in Figure 2, which represents the (normalized) mean squared error vs. the number of selected basis functions. (Since the algorithm backward-eliminated the basis functions, the curve was actually obtained from right to left, with $\xi_{\max} = \mathrm{E}(Y^2) = 1$.) The first ten selected basis functions are listed in Table 1 together with the value of their coefficient and the error achievable by combining the first-rank monomials. The error clearly saturates from the

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 273-278

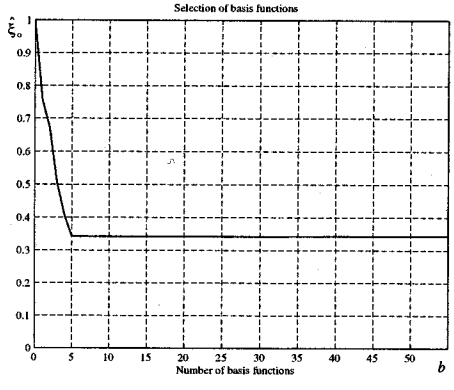

Figure 2: Beamforming with Gaussian jammer: selection among 55 monomials

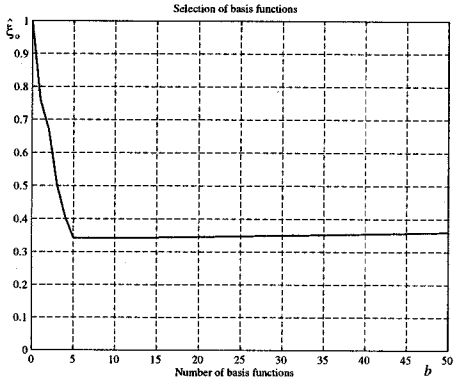

Figure 3: Selection among 10 monomial, 20 sigmoidal, and 20 radial functions

five first monomials (see Fig. 2). Note that the algorithm ranked indeed the linear terms first (c.f. Table 1) in accordance with theory [2]. Moreover, the weights c_i of the next terms are much smaller than for the linear terms.

In another situation, the algorithm was presented with the 10 monomials of Table 1 together with 20 sigmoidal and 20 radial basis functions obtained by nonlinear regression. The results are reproduced in Figure 3. The linear terms were again ranked first, thereby indicating the irrelevance of sigmoidal and radial basis functions in this case.

Rank	$\psi_i(\boldsymbol{x})$	c_i	Cumul. $\hat{\xi}_o$
1	x_2	1.360	0.758
2	x_3	1.014	0.672
3	x_1	0.678	0.505
4	x_4	0.676	0.407
5	x_5	0.345	0.342
6	$x_1x_5^2$	0.019	0.342
7	$x_1^2x_5$	0.011	0.342
8	$x_2^2x_4$	−0.048	0.342
9	$x_1x_2x_4$	−0.043	0.342
10	$x_2^2x_5$	−0.041	0.342

Table 1: Selected basis functions for the beamformer with Gaussian jammer (correspond to $b = 0 \ldots 10$ in Fig. 2)

Rank	$\psi_i(\boldsymbol{x})$	c_i	Cumul. $\hat{\xi}_o$
1	x_2	1.335	0.750
2	x_2^2	0.337	0.639
3	x_1	0.656	0.446
4	x_3	1.002	0.364
5	x_4	0.675	0.308
6	x_4^2	0.343	0.251
7	x_5	0.346	0.221
8	x_5^2	0.201	0.205
9	x_1^2	0.204	0.194
10	$x_1^2x_2$	−0.073	0.185

Table 2: Selected basis functions for the beamformer with exponential jammer (correspond to $b = 0 \ldots 10$ in Fig. 4)

3.2. Experiment with a nonlinear Bayes rule

The experiment was repeated for an exponential distributed jammer. In this case, the Bayes rule is nonlinear [2]. The selection among the same $B = 55$ monomials as in 3.1. resulted in Fig. 4 (compare with Fig. 2). The first ten basis functions selected by the

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 273-278

algorithm are listed in Table 2. Note that there are now nonlinear terms that contribute significantly to reducing the error.

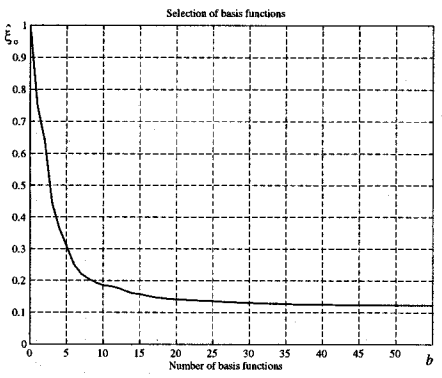

Figure 4: Beamforming with exponential jammer: selection among 55 monomials

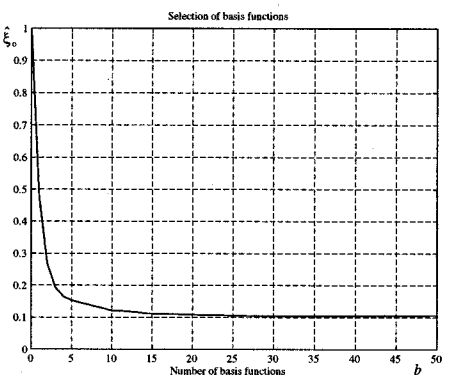

Figure 5: Selection among 10 monomial, 20 sigmoidal, and 20 radial functions

In another case, candidates of different type were mixed and submitted to the algorithm (10 monomial, 20 sigmoidal, and 20 radial basis functions), which gave Fig. 5. Comparison with Fig. 4 indicates that a smaller error could be reached. For instance, the subset of $b = 10$ basis functions selected by the algorithm consisted of 5 sigmoids, 3 radial functions, 1 polynomial and 1 linear term.

4. Conclusion

Selecting a functional structure (infinite information) from a finite number of samples (finite information) is in essence an ill-posed problem. If a nonlinear model is needed, care must be taken to avoid over-parametrization, which leads to overfitting and bad generalization. A simple method was suggested to select basis functions from a set of candidates which may be of different type. The method was illustrated by simulation of a noise reduction system.

References

[1] A.R. Barron. "Universal approximation bounds for superpositions of a sigmoidal function". *IEEE Transactions on Information Theory*, 39(3):930–45, 1993.

[2] W. Knecht, R. Steiner, M. Joho, and G.S. Moschytz. "Cancelling spatial interference with nonlinear filters". In H. Dedieu, editor, *Proc. European Conference on Circuit Theory and Design*, volume 1, pages 537–42. Elsevier, 1993.

[3] A. Poncet. *Design of Adaptive Models for System Identification, Signal Prediction, and Pattern Classification*, volume 9 of *ETH Series in Information Processing*. Hartung-Gorre Verlag, Konstanz, 1997.