

Network Performance Assessment for Neurofuzzy Data Modelling

Steve R. Gunn, Martin Brown and Kev M. Bossley

Image, Speech and Intelligent Systems Group
Department of Electronics and Computer Science
University of Southampton

13 May 1997

Abstract

This paper evaluates the performance of ten significance measures applied to the problem of determining an appropriate network structure, for data modelling with neurofuzzy systems. The advantages of Neurofuzzy systems are demonstrated with application to both real and synthetic data interpretation problems.

1 Introduction

Neurofuzzy systems have recently received an intensified research effort [Brown and Harris \(1994\)](#); [Jang et al. \(1997\)](#), as they combine the learning ability of neural networks with a fuzzy representation, to provide an enhanced linguistic representation. Conventional neural networks are successful at approximating continuous multivariate functions from a supervised training set, but, with the exception of trivial cases, it is difficult for a designer to interpret the knowledge that is stored within such a network. This research focuses on the ability of a learning system to automatically configure its own structure, so that it models the training data and explains the stored knowledge in a transparent fashion. A network structure which encapsulates the principle of transparency is the class of additive B-spline fuzzy networks [Brown and Harris \(1994\)](#), which are members of the group of associative memory networks. They have the benefit that only the output layer weights are adjusted which allows established linear training algorithms to be employed, such as conjugate gradient (CG) or singular value decomposition (SVD).

Conventional multi-layer perceptron (MLP) networks [MacKay \(1995\)](#) use sigmoidal ridge functions to decompose the input space. These networks employ a projection pursuit type learning for hidden layer identification, but they are not transparent. However, they can produce smooth models but are often difficult to train. Radial basis function (RBF) networks [Orr \(1996\)](#) can be constructed using orthogonal least squares. They can have their nodes placed anywhere in the input space and hence they have a semi-transparency. However, the lack of structured placement makes them difficult to interpret and sometimes their learning is badly conditioned. The B-spline networks considered here are based on an additive tree structure which uses a forward selection and backward elimination algorithm to decompose the input space. Their disadvantage is in application to strongly coupled functions in a high dimensional input space, and here MLPs have an advantage. Their strength is a comparatively simple local representation which can be used to explain the knowledge extracted from the training data, with the advantages of a fuzzy rule base interpretation. The fundamental question with all these networks is how to select the size and structure of the network for a particular problem. This paper addresses this issue by evaluating some performance functions for assessing the significance of a network with respect to a set of training data. Additionally the technique of regularisation is discussed as a method for post-processing the network to provide further evidence about the performance of the model and its sub-networks. This is preceded by an introduction to the construction algorithms employed within the neurofuzzy networks.

1.1 B-Spline Neurofuzzy Networks

When neurofuzzy systems are applied to applications they tend to suffer from the curse of dimensionality [Bellman \(1961\)](#). To address this issue an additive decomposition is employed, which can exploit redundancy in the training data [Brown and Harris \(1994\)](#). Here a function is expressed as a sum of simpler sub-functions,

$$f(\mathbf{x}) = f_0 + \sum_{i=0}^{n-1} f_i(x_i) \sum_{i=0}^{n-1} \sum_{j=i+1}^{n-1} f_{i,j}(x_i, x_j) + \dots + f_{0,1,\dots,n-1}(\mathbf{x}), \quad (1)$$

where f_0 represents the bias and the other terms represent the univariate, bivariate etc. additive components. The benefit of this representation is that often many of the sub-functions are redundant providing a more compact description. [Figure 1](#) illustrates an additive network, with its fuzzy B-spline basis functions shown

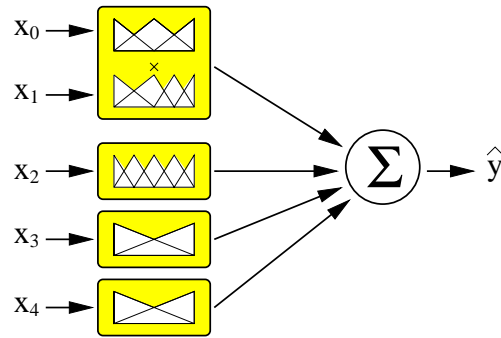


FIGURE 1: A B-spline Neurofuzzy network composed of four sub-networks.

for each input. The B-spline basis functions can be of arbitrary order dependent on the application domain, and are illustrated here for the piece-wise linear case. The introduction of additional knots within the basis functions enables increasingly complex functions to be approximated, whilst an increase in their order enables smoother functions to be obtained.

1.2 Construction Algorithm

The purpose of data modelling is to understand the variables and relationships within the data. To achieve this a method of model selection is required which can search through the potentially high dimensional model space for an appropriate data representation. The question of what is an appropriate network will be returned to in [Section 2](#). The process of model selection is achieved here by an evolutionary search algorithm. An initial model is given, which can be empty if no knowledge is available about the network structure, or the designer can introduce information within this network by means of an initial structure, using fuzzy rules. This model is then updated by evaluating the best refinement from a list of potential refinements. Current refinements include: univariate addition, tensor product, tensor split, knot insertion, knot deletion, sub-network deletion, reduce order and regularise. The B-spline basis functions are chosen from a set of candidate basis functions. The refinements are collected into passes to provide a restricted and coherent method of searching the model space. A typical pass structure is,

Pass 1 Univariate addition, Tensor Product, Tensor Split.

Pass 2 Sub-network deletion.

Pass 3 Knot insertion.

Pass 4 Knot deletion.

Pass 5 Reduce order.

Pass 6 Regularise.

This forward selection and backward elimination approach allows the model to increase and decrease in complexity to adapt the network structure to model the data.

1.3 Termination Criterion

To terminate the model selection at each pass a termination criterion, Figure 2, is used which embodies two rules. The first rule allows the refinement to escape from local minima in the refinement process by allowing the model to look ahead in the tree structure. The second rule places an emphasis on the parsimony of the network, by requiring a new refinement which increases the network size to reduce the significance measure by a certain percentage, f_{tol} . Similarly, a refinement which reduces the network size is allowed to increase the significance measure by a certain percentage, b_{tol} . This hysteresis enables superior refinement termination. (A high significance measure corresponds to a poor network). Typically $f_{tol} = 3\%$ and $b_{tol} = -1.5\%$.

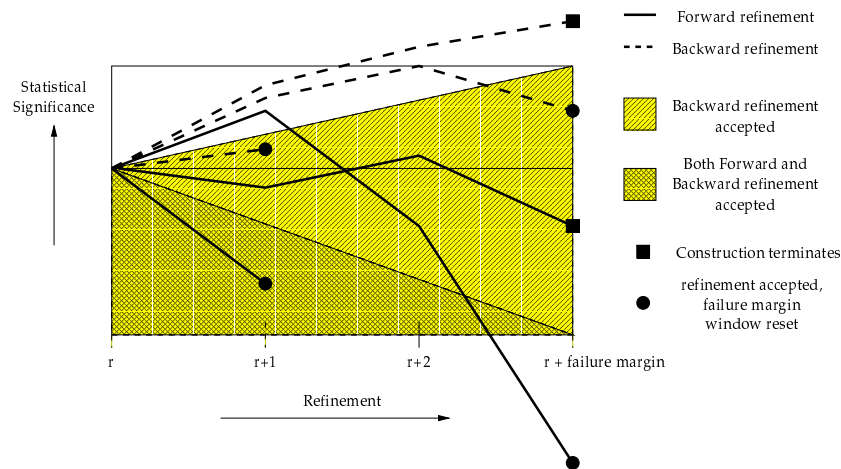


FIGURE 2: Construction termination criterion

2 Significance Measures

What is a significant model of a set of data? Four characteristics that are required of a network are:

Parsimonious The network is as simple as possible to model the data.

Transparent The network is interpretable by the designer allowing the knowledge stored in the network to be understood, and when necessary, modified by the designer.

Generalisation The network should produce good models outside the limits of the training data set.

Accuracy The network should accurately model the data.

Transparency is implicit within the B-spline neurofuzzy networks at four levels; the fuzzy rule base, visualisation of sub-network outputs, linear relationships and the ability to reject redundant inputs. The accuracy, generalisation, and parsimony of the network are closely related. The following two sections considers two ways of measuring these properties; the first primarily measures the accuracy of the network, coupled with an estimate of the networks complexity, the second additionally addresses the question of generalisation by providing an estimate of the noise on the training data. Finally, the method of regularisation is introduced as a post-processing technique to provide additional insight into the validity of the data model.

2.1 Mean Square Error based Measures

The *MSE* is an estimator of the accuracy of the model, and as such is often employed within significance measures. However, as a biased estimator [Geman et al. \(1992\)](#) it is impractical because a network can always reduce the *MSE* by introducing additional degrees of freedom. The *MSE* is often utilised as an unbiased estimator by considering the degrees of freedom of the model. There are many such functions in the literature of the form,

$$ss = MSE \cdot f(n_w, n_p) \quad (2)$$

where n_w is the number of weights and n_p is the number of training patterns. To illustrate the behaviour of some of these significant measures, Figure 3 shows the equi-potential functions, $ss = \text{constant}$, plotted against the number of weights, for a fixed training pattern size, $n_p = 100$; the measures are: bayesian information criterion (*BIC*), generalised cross validation (*GCV*), unbiased estimate of variance (*UEV*), final prediction error (*FPE*), Akaike information criterion (*AIC*), unbiased Akaike information criterion (*UAC*), full generalised cross validation (*FGV*), structural risk management (*SRM*), and minimum descriptor length (*MDL*) [Orr \(1996\)](#); [Kavli and Weyer \(1995\)](#); [Bossley \(1997\)](#); [Droge \(1994\)](#). From inspection it is evident that the measures can be grouped into three classes: Class I: $\{AIC, FGV, MDL\}$, Class II: $\{BIC, GCV, UEV, FPE, UAC\}$ and Class III: $\{SRM\}$. All the functions are monotonically increasing, with respect to n_w . Class I functions place no upper limit on the number of weights and potentially allow over fitting of the data to occur. Class II functions have an asymptote at $n_w = n_p$ and hence limit the number of weights in the network to be less than the number of training patterns. They can be ordered such that $UEV < FPE < (GCV, BIC, UAC)$. The Class III function places an upper limit on n_w which is dependent upon its parameter K_1 ; for $K_1 = 1.0$ the maximum number of weights is $\approx 0.37n_p$ ($n_p > 40$). Additionally, it is a rapidly increasing function and it will have a tendency to under-fit the data with respect to the functions of class I and II. The *MDL*, *BIC* and *SRM* measures are dependent upon the number of training pairs whereas the other measures are solely dependent upon the ratio n_w/n_p . To limit the testing of these functions, a representative of each class is used for comparison on the data sets. The measures that were chosen limit the tendency to overfitting from their respective class. They are,

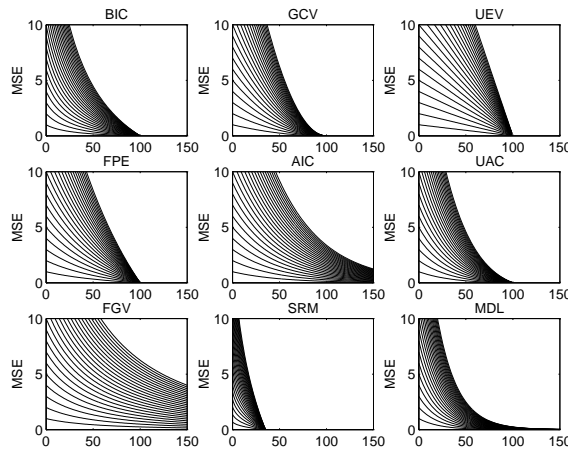


FIGURE 3: Equi-potential curves for *MSE* based measures ($n_p = 100$).

$$\begin{aligned}
 MDL(MSE, n_w, n_p) &= MSE \cdot \exp\left(n_w \frac{\ln(n_p)}{n_p}\right), \\
 BIC(MSE, n_w, n_p) &= MSE \cdot \left[\frac{n_p + (\ln(n_p) - 1)n_w}{n_p - n_w} \right]_{\infty}, \\
 SRM(MSE, n_w, n_p) &= MSE \cdot \left[\frac{1}{1 - K_1 \sqrt{\frac{(1+n_w) \ln(2n_p) - \ln((1+n_w)!) + K_2}{n_p}}} \right]_{\infty}
 \end{aligned}$$

where

$$[x]_{\infty} = \begin{cases} x & x \geq 0 \\ \infty & x < 0 \end{cases} . \quad (3)$$

The disadvantage of these measures is that they are inherently dependent upon the *MSE* as a measure of model suitability. The training data is often sparse and in order to avoid relying on one particular instance it can be advantageous to divide the training set in different ways to provide several different training and test sets. There are two common frameworks for achieving this, bootstrapping [Efron and Tibshirani \(1993\)](#) and cross validation [Orr \(1996\)](#). Here we consider the method of cross validation.

2.2 Cross Validation

To address the problems of model significance our research has focused on the method of cross validation as an alternative to the above *MSE* based measures. Cross validation allows all the training patterns to be used for both training and testing of the network, by partitioning the data into packets of size m , and training the network on all but one of these, which is then used to measure the error. This is repeated for all the packets in turn to provide a mean estimate. When $m = 1$ the technique is referred to as leave one out cross validation (*LOOCV*), which is the method considered here due to its applicability to networks which are linear in their parameters. The increased computation of *LOOCV* is minimised in such cases and it may be calculated from the projection matrix, **P** [Orr \(1996\)](#),

$$LOOCV = \frac{\hat{\mathbf{y}}^T \mathbf{P} (\text{diag}(\mathbf{P}))^{-2} \mathbf{P} \hat{\mathbf{y}}}{n_p} \quad (4)$$

The projection matrix is calculated via a SVD of the auto-correlation matrix, which can be used to directly solve for the network weights. This typically is about three times slower than the *MSE* based measures which can exploit the increased speed of the CG method for weight training.

2.3 Regularisation

Regularisation of a network smoothes the output surface and enhances the models interpolation and its extrapolation capabilities, particularly where training data is sparse. It achieves this by reducing the models sensitivity to individual data sets. It has been found [Bossley \(1997\)](#) that second order regularisation, which introduces a soft prior smoothness constraint, is relatively simple to implement and gives good results, effectively giving more reliable fuzzy rules. The complexity of the resulting model is controlled by the regularisation coefficients which can be determined using a re-estimation formula derived from bayesian inferencing.

2.4 Summary

To evaluate the significance measures considered in this section, a representative set containing, *MDL*, *BIC*, *SRM*, and *LOOCV* is applied to real and simulated data sets in the next section. The method of second order regularisation is demonstrated as an aid to model interpretation.

3 Performance Measure Evaluation

To evaluate the performance of the four measures two problems were chosen. The first data set is taken from [Friedman \(1991\)](#) and concerns the modelling of an additive function. This example was used to investigate the behaviour of the performance measures as the training data size varies. The second example was taken from [Blake and Merz \(1998\)](#) and concerns the modelling of automobile MPG data. In both experiments the models were initialised with an empty network structure, and the refinements were chosen by minimising the current performance measure; piecewise linear B-splines were used. The model refinements were stopped according to the termination criterion of Section 1.3, with a failure margin of three and $f_{tol} = 3\%$, $b_{tol} = -1.5\%$. The coefficients in the *SRM* measure were taken from [Kavli and Weyer \(1995\)](#), $K_1 = 1.0$, $K_2 = 4.8$.

3.1 Example 1: Additive Data Modelling

The model considered is a ten input function, five of which are redundant, given by,

$$f(x_0, x_1, \dots, x_9) = 10 \sin(\pi x_0 x_1) + 20 \left(x_2 - \frac{1}{2}\right)^2 + 10x_3 + 5x_4 + \mathcal{N}(0, 1) \quad (5)$$

where $\mathcal{N}(0, 1)$ is zero mean additive Gaussian noise, corresponding to approximately 20% noise, and the inputs were generated independently and randomly from a uniform distribution in the interval $[0, 1]$. The trials were performed for five different training data sizes, $\{50, 100, 200, 500, 1000\}$, with ten independent data sets for each size. The results are presented in Figure 4, showing the sample mean and sample standard deviation of the network size for each of the ten data sets. The function can be modelled well with four sub-networks

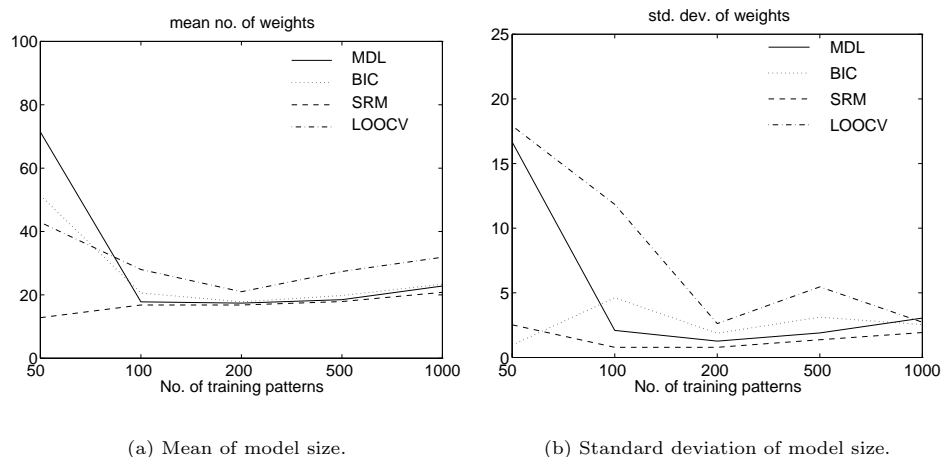


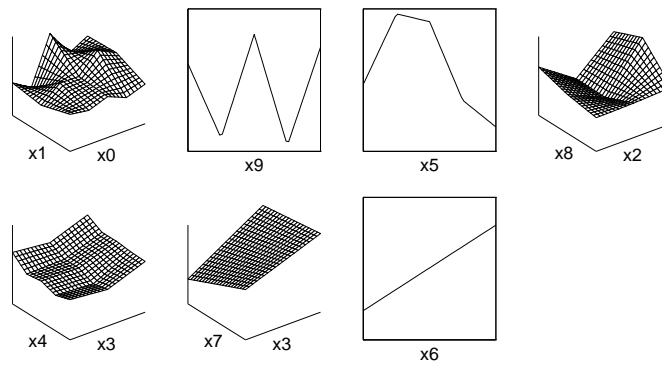
FIGURE 4: Model size vs. training pattern size.

corresponding to a network in Figure 1, with 16 weights. It is evident that the *BIC*, *MDL* and *LOOCV* measures generate over parameterised models for the 50 sample case, whereas the *SRM* measure generates a slightly undersized model. The *MDL* measure produces a mean model with $n_w > n_p$ and a high variation in model size, the *BIC* measure produces a mean model with $n_w \approx n_p$ and a low variation in model size, and the *LOOCV* measure produces a mean model with $n_w < n_p$ and a high variation in model size. The *SRM* measure is relatively successful, with a mean model size of 13 weights and a small variation in model size, because the cut-off for the *SRM* measure corresponds to 18 weights, just allowing the model to be approximated. When the sample size is increased above 100 all measures provide a similar mean model size that increases slightly with an increase in n_p . The standard deviations of the model size for the *BIC* and *SRM* measures are approximately constant across the different training sizes, whereas *MDL* and *LOOCV* initially have a large variation that reduces as n_p increases. Table 1. illustrates the number of correct network structures. It can be seen that the *SRM* measure provides the best models for small n_p . Figure 5 illustrates the method of regularisation

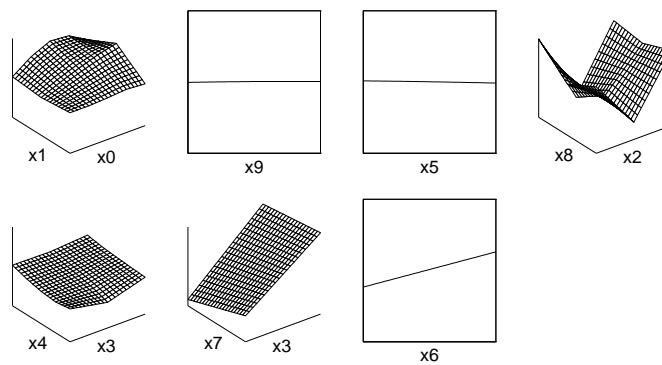
n_p	50	100	200	500	1000
<i>MDL</i>	0	6	10	10	10
<i>BIC</i>	0	4	10	10	10
<i>SRM</i>	6	8	10	10	10
<i>LOOCV</i>	0	4	9	10	10

TABLE 1: Number of correct network structures for the 10 trials.

applied to a model obtained using *LOOCV* for $n_p = 50$. Figure 5(a) shows the sub-network outputs before regularisation and Figure 5(b) show the sub-network outputs after regularisation. These serve to demonstrate that the redundant inputs can be identified by regularisation even when the model is structure is over complex.



(a) Model without regularisation.



(b) Model with regularisation.

FIGURE 5: Regularisation example for an *LOOCV* measured model (x_{5-9} are redundant).

3.2 Example 2: Automobile MPG Data Modelling

The automobile MPG training set contains the following data: no. of cylinders, displacement, horsepower, weight, acceleration, year and the mpg for 392 cars. The performance measures were used to produce four respective models. Figure 6 illustrates the resulting models, with the model structure on the left and the output of the sub-networks on the right. All four networks choose a structure consisting of the three inputs, weight, horsepower and year. However, the *LOOCV* measure produced a tensor product between the year and horsepower properties, and for the purposes of comparison the remaining measures are displayed in the same manner. The difference of the *LOOCV* model is the plateau around the region, 75 hp./1972, that is consistent with the data in this region, which is dominated by VW cars! In contrast, the other measures predict an increase in MPG with an increase in horsepower, which is due to the less flexible models that were chosen. However, the important feature here is that overall the models have identified similar structures from the data.

4 Conclusions

The transparency of additive B-spline fuzzy networks has been demonstrated by their ability to describe the data in a form that the designer can inspect; the outputs of the sub-networks can be visualised to provide good understanding of the data structure and reasoning can be done using fuzzy rules. In order to form these models ten performance measures were considered. These were reduced to four representative measures which were evaluated on a synthetic modelling problem and an example to model automobile MPG data. The *SRM* measure showed the best performance on the synthetic modelling problem, providing good results over all data

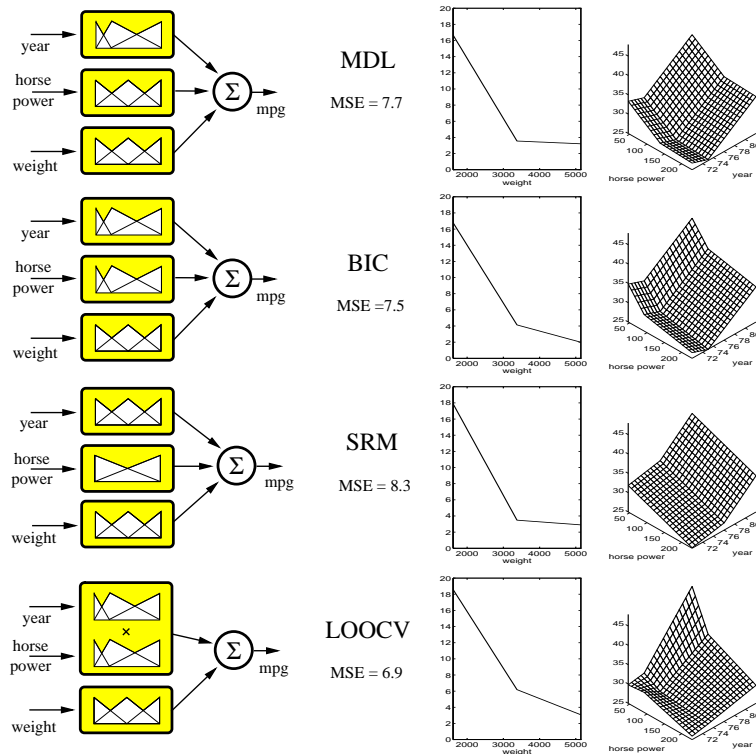


FIGURE 6: Automobile MPG network models and their sub-networks.

sizes considered, and out performing the other measures significantly for small data sizes. The automobile data demonstrated that for larger data sizes *LOOCV* can provide a better interpretation of the data. However, it may advantageous for a designer to form models using different performance measures to determine the sensitivity of the models to the performance measure used.

References

- R.E. Bellman. *Adaptive Control Processes*. Princeton University Press, Princeton, NJ, 1961.
- C.L. Blake and C.J. Merz. [UCI repository of machine learning databases](#), 1998.
- K. M. Bossley. *Neurofuzzy modelling approaches in systems identification*. Ph.d. thesis, Faculty of Engineering, University of Southampton, Southampton, U.K., 1997.
- M. Brown and C. J. Harris. *Neurofuzzy Adaptive Modelling and Control*. Prentice Hall, Hemel Hempstead, 1994.
- B. Droge. Some comments on cross-validation. Technical report, Humboldt-Universitat, Berlin, 1994.
- B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall, London, UK, 1993.
- J.H. Friedman. Multivariate adaptive regression splines. *The Annals of Statistics*, 19:1–141, 1991.
- S. Geman, E. Bienenstock, and R. Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4:1–58, 1992.
- J. Jang, C. Sun, and E. Mizutani. *Neurofuzzy and Softcomputing*. Prentice Hall, NJ, 1997.
- T. Kavli and E. Weyer. On ASMOD - an algorithm for building multivariable spline models. In G.R. Irwin K.J. Hunt and K. Warwick, editors, *Advances in Neural Networks for Control Systems*, Springer series on Advances in Industrial Control, pages 83–104. Springer Verlag, 1995.

-
- D. J. C. MacKay. Bayesian methods for supervised neural networks. In M. A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*, pages 144–149. MIT Press, 1995.
- M. Orr. [Introduction to radial basis function networks](#). Technical report, Center for Cognitive Science, Univ. of Edinburgh, 1996.