Extraction of the Relevant Delays for Temporal Modelling

Cyril Goutte*

Abstract

When modelling temporal processes just like in pattern recognition, selecting the optimal number of inputs is a central concern. In this contribution, we take advantage of specific features of temporal modelling to propose a novel method for extracting the inputs, that yield the best predictive performance. The method relies on the use of generalisation estimators to assess the performance of the model. This technique is first applied to time series processing, where we perform a number of experiments on synthetic data as well as a real life dataset, and compare the results to a benchmark physical method. Finally, the method is extended to system identification with a linear FIR filter.

This work has been submitted to IEEE Press for possible publication.

*Corresponding address: Department of Mathematical Modelling, Technical University of Denmark, Building 321, DK-2800 Lyngby, Denmark. E-mail: cgimm.dtu.dk

1 Overview

Time series prediction and system identification try to model the future behaviour of a system, based on its past, plus possibly some exogenous control signal. Just like in pattern recognition, performance depends, among other things, on the design of a proper input, or regression vector. With too few inputs, the model does not have sufficient information and is unable to grasp the inner workings of the system, resulting in a large approximation error. On the other hand, if we include irrelevant inputs on top of the necessary ones, the model is over-parameterised, which usually results in poor prediction performance, as suggested by the curse of dimensionality. This trade-off is actually a particular instance of the bias-variance dilemma [8].

In this contribution, we will focus mainly on time series prediction, and address later the problem of system identification. In this context, potential inputs are past values, or in other words delays, of the time series. The aim of our method is to select delays that are necessary to model the system, while discarding unnecessary delays which could prove harmful to the overall performance. The method is related to the iterative feature selection techniques used in traditional statistics [17], and like them it is *not* optimal. Our technique builds on the specificities of temporal processing to provide an original way of selecting potential inputs. The relevance of a candidate delay is assessed directly by its effect on modelling performance, measured using generalisation estimators.

The organisation of this article is as follows: first we give a general presentation of feature selection (section 2), applied to time series modelling, from the statistical and the physical point of view. We suggest the use of estimators of generalisation error to evaluate the quality of a subset of features. Our extraction of the relevant delays (ERD) method is then described (section 3) as a principled alternative. The second part of the paper contains a number of experiments conducted on three different datasets. Time series predictions (section 4) is addressed using the well-known artificial Hénon map and a real time series measuring the mean monthly flow of the Fraser river; ERD is then extended to system identification (section 5) with a functional magnetic resonance imaging dataset. We conclude with a discussion of the method and results.

2 Feature Selection

Let us consider a standard time series modelling problem. A sequence measurements $y(t)$ is collected, $1 \leq t \leq T$. We wish to predict $y(t)$ from a set of past values $y(t-d)$, with delays $d > 0$. Note that in this setting, the sampling time, ie length of the basic time delay (difference between $t+1$ and t) is imposed—we do not address the problem of obtaining the best sampling frequency.

Extracting the relevant delays consists in finding the set of m delays $(d_1, d_2, \ldots d_m)$ such that the input vector

$$\mathbf{x}(t) = [y(t-d_1)\ y(t-d_2)\ \ldots\ y(t-d_m)]$$

allows the best possible prediction of $y(t)$.

2.1 Physical Approach

The physical approach consists in trying to estimate the *embedding dimension* of the time series [31]. This is essentially equivalent to finding the set of *primary* delays, ie delays that have a direct influence on the observed time series values. For example, if a time series is generated by $x(t) = g(x(t-1))$, 1 is the only primary delay. $x(t)$ can also be estimated using eg $x(t-2)$, as $x(t) = g(g(t-2))$, however 2 is not a primary delay.

Several methods have been proposed to estimate the embedding dimension—and the embedding space—of a time series. In the neural networks literature, Pi and Peterson [29] have introduced the "δ-test" to address this issue. In system identification, the use of "Lipschitz quotients" has been proposed in order to identify the order of non-linear input-output systems [15]. A similar method was applied to time series, under the name of "geometrical technique", in the signal processing literature [24]. Though different in practice, these methods rely on a common assumption on the continuity—and relative smoothness— of the underlying mapping g, and use a geometrical approach based on the data alone. The continuity argument means that if there is a mapping between $\mathbf{x}(t)$ and $y(t)$, then close inputs $\mathbf{x}(u)$ and $\mathbf{x}(v)$ should correspond to close outputs $y(u)$ and $y(v)$. Accordingly, as long as the input space is insufficient (i.e. missing delays), close inputs can correspond to arbitrarily distant outputs. This will be illustrated in section 4 on a practical example. Quantifying this closeness is done either by measuring empirical probabilities that two outputs are close given that the corresponding inputs are close (δ-test), or by calculating the ratio between output and input distances (Lipschitz quotients).

It should be noted that the techniques mentioned above are non-parametric. An important side-effect is that they consider all possible input-output pairs and therefore require extensive calculations. Furthermore, these methods rely on the data alone, and need not specify a given model, contrary to the statistical approach presented below. This can turn out to be a disadvantage since for a given data set, they always select the same set of relevant delays, regardless of the ability of our model to actually implement the underlying mapping. For flexible models such as neural networks, the uniform approximation capabilities [18] should balance this limitation. However, it could (and in practice *does*) happen that for the model at hand, the estimation would benefit from the inclusion of *secondary* delays. This will be shown in the experiments, and discussed below.

2.2 Statistical Approach

From a statistical point of view, the extraction of the relevant delays is a special case of feature selection, which in turn can be seen as a part of the more general problem of analysing the structure in the data [17]. This statistical approach relies on specifying a model f with which we try to estimate the input-output mapping $\widehat{y}(t) = f(\mathbf{x}(t))$. f can be either a parametric or a non-parametric model. An important assumption in conventional variable selection is that all necessary variables are available, i.e. a sufficient subset of inputs actually exists. Provided that our data are sampled correctly, this assumption is usually satisfied in the case of time series. It breaks down in the case where a long-term delay is needed, that ranges further than the time period spanned by the data. However, the relevance of such long-term prediction is questionable, as there would be no data to identify the associated parameter(s). Note that in the following, the *delays* are positive integers $d_j > 0$, with $1 \leq j \leq m$, the *variables* or *features* are past values of the time series $y(t - d_j)$ and the *inputs* are the vectors $\mathbf{x}(t)$ containing these past values. Finding the relevant delays is then equivalent to finding the input that give optimal predictive performance.

Feature selection relies on three different components:

1. a selection method searching through possible subsets of variables,

2. an evaluation criterion in order to assess the quality of each subsets,

3. a stop condition.

To our knowledge, the only general optimal method for selecting the best features is to perform an exhaustive search through all possible subsets of variables. This is usually impossible for combinatorial reasons, because the number of possible subsets among P variables is $2^P - 1$. For time series, an aggravating factor is the fact that the maximum delay, and hence the total number of variables (P), is not known in advance. Typically, we would accept to probe quite far into the past. When the evaluation criterion is monotonous, the branch and bound algorithm [27] provides an efficient alternative. However, the computational cost is still overwhelming in most cases, and the predictive ability is not a monotonous criterion.

Most sub-optimal methods perform an iterative search by regularly increasing or decreasing the number of features selected [17]:

- *Forward selection*: start from an empty set (no variables), and add variables according to the evaluation criteria, until the stop condition is reached.

- *Backward elimination*: start with a full set (all possible variables), and proceed by deleting one variable at a time according to the evaluation criteria, until the *stop condition* is reached.

- *Stepwise regression*: start from a given set (usually all variables), and alternate between both methods. One possibility is to perform a backward elimination after each inclusion, another method chooses between adding or deleting variables according to the evolution of the evaluation criterion.

Note that the focus in neural networks research, for example, is almost entirely on backward elimination, through various pruning schemes [22]. All these sub-optimal methods rely heavily on the evaluation criterion. Typical choices include for example the F-statistic [17], extensions to non-linear models, or mutual information [5]. We will now present an alternative to these choices.

2.3 Generalisation Approach

Our goal in time series modelling is to obtain the model with the best prediction. Thus it is natural to use a measure of the predictive abilities as a base for our evaluation criterion. Consider a model f providing a mapping from an input vector containing m delays $\mathbf{x}(t) = [y(t-d_1)\dots y(t-d_m)]$ to output $y(t)$. The risk $\ell(y, f(\mathbf{x}))$ measures the cost associated with estimating the output y observed in $\mathbf{x}$ with the model $f(\mathbf{x})$. Assuming Gaussian noise on the output, this lead to the choice of the quadratic risk $\ell(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$. The *generalisation error* (or expected risk) associated with model f is defined as the expectation of the risk over the unknown, but fixed, joint input-output distribution:

$$G(f) = \int (y - f(\mathbf{x}))^2 \, p(\mathbf{x}, y) \, \mathrm{d}y \mathrm{d}\mathbf{x} \tag{1}$$

Ideally, the evaluation criterion for a given subset should be the generalisation error of the model using this subset as input [10]. However, equation 1 cannot be used directly as the joint input-output probability is unknown. It shall be estimated from the available data $(\mathbf{x}(t), y(t))$, with $t \in \mathcal{D} = \{T_{min}, \dots T_{max}\}$. The *split-sample* estimator is obtained by replacing the joint distribution $p(\mathbf{x}, y)$ by the empirical estimator on a particular validation set $\mathcal{V} \subset \mathcal{D}$:

$$\widehat{G}_{SS} = \frac{1}{|\mathcal{V}|} \sum_{t \in \mathcal{V}} (y(t) - f(\mathbf{x}(t)))^2 \tag{2}$$

where $|\mathcal{V}|$ is the size (or cardinal) of set $\mathcal{V}$. Note that the validation set $\mathcal{V}$ and the set on which f is estimated (the training set $\mathcal{T} \subset \mathcal{D}$) should be distinct ($\mathcal{V} \cap \mathcal{T} = \emptyset$) in order for (2) to be an unbiased estimator of (1).

The cross-validation[1]estimator [33, 34] resamples the validation and training sets from the available data in order to increase the reliability of the resulting estimator. In L-fold cross-validation, $\mathcal{D}$ is split in

[1]The split sample method is sometimes referred to as "cross-validation" in the neural networks literature. This is inconsistent with the definition of [33] and we will here reserve the term exclusively to the averaging method (3).

L disjoint subsets $(S_j)_{j=1,\ldots L}$ of roughly equivalent size, $\bigcup_{j=1}^{L} S_j = \mathcal{D}$. The split-sample estimator (2) calculated on each S_j, using the remaining data for model estimation, is averaged over the subsets:

$$\widehat{G}_{CV} = \frac{1}{N} \sum_{t \in \mathcal{D}} \left(y(t) - f^{-t}(\mathbf{x}(t)) \right)^2 \tag{3}$$

where f^{-t} is the model estimated excluding the subset S_j containing t. Strictly speaking, equation (2) does not estimate the generalisation error (1), but its average over all possible datasets of size $\mathcal{D}$ sampled from $p(\mathbf{x}, y)$.

Finally, a number of analytical estimators of the average generalisation error have been proposed in the literature, e.g. Final Prediction Error (FPE) [1], Generalised Prediction Error (GPE) [25], Final Prediction Error for Regularised problems (FPER) [20] or Network Information Criterion (NIC) [26]. We will here settle for an expression similar to GPE, i.e. a Final Prediction Error where the number of parameters is replaced by the *number of efficient parameters* $\widehat{P}$:

$$\widehat{G}_{FPE} = \left(\frac{N + \widehat{P}}{N - \widehat{P}} \right) R(f) \tag{4}$$

where $R(f)$ is the training error, i.e. the average risk, on dataset $\mathcal{D}$, of the model estimated from $\mathcal{D}$:

$$R(f) = \sum_{t \in \mathcal{D}} \left(y(t) - f(\mathbf{x}(t)) \right)^2$$

The calculation of $\widehat{P}$ depends on the estimator and the regularisation method used during training (see e.g. [20, 9]).

In the following, we will use the estimators of generalisation error as our evaluation criterion in order to derive an original delay extraction scheme.

3 Extraction of the Relevant Delays

Statistical feature selection, as presented above, applies to general regression problems. The particular application of time series prediction (and to some extent, system identification) displays a number of characteristics that are worth mentioning.

The features of interest are past values, or delays, of the time series. As mentioned above, this means that all possible variables will be available. On the other hand, the maximum delay is unknown *a priori*. As a consequence, even though we potentially have all necessary features, we have no upper bound on their number, or their order. This makes the extraction of relevant delays a rather bad candidate for backward elimination schemes.

The chronological order gives us a natural way of ordering and selecting the variables. The following scheme takes advantage of this order to select candidate delays in the natural, chronological order. The rationale for this choice is that we will always test primary delays before secondary delays. Due to the temporal structure in the data, secondary delays are necessarily placed after the primary delay they are related to (eg $t-2$ and $t-1$ above). Therefore, when a secondary delay is tested for inclusion, it is likely to bring nothing to the prediction, unless 1) the model is unable to represent the underlying mapping using primary delays alone or 2) the secondary delay is also a primary delay[2]. This heuristic can also be justified by the fact that the traditional caveat against sequential selection is that variables that are combinations of other variables might confuse the feature selection algorithm. In the context of time series prediction, combination of variables arise naturally from past values of the series. However, the chronological selection ensures that each variable is tested for inclusion before the variables it depends on. This guarantees that the selection stays parsimonious. Let us stress the fact that we do not claim that this heuristic choice is optimal. Indeed the experiments presented below suggest that it is not, though the results appear to be close to optimum.

The proposed Extraction of Relevant Delays method is a forward selection method. It takes all delays in their natural order, and adds a candidate variable if and only if it yields a *significant* decrease in generalisation error. The algorithm can be described as follows:

[2]If $y(t) = g(y(t-1), y(t-2))$, 2 is a primary delay but also a secondary delay, through the primary delay 1.

1. Initialise: $d = 0$;
   $G_{min} = \sigma_y^2$ (time series variance);
   No input selected.
2. Model: $d = d + 1$;
   Add delay $t - d$ to the selected inputs;
   Estimate the generalisation error $\widehat{G}$ for the resulting model.
3. Test: If $\widehat{G}$ is *significantly* smaller than G_{min} then keep delay $t - d$ and set $G_{min} = \widehat{G}$.
   Otherwise, discard delay $t - d$.
4. Iterate: Go to step 2 until the stop condition is reached.

3.1 Significant Decrease in Error.

When a candidate delay yields a decrease in (estimated) generalisation error, step 3 requires that we assess the significance of this decrease. This requirement avoids that we include a delay that leads to negligible decrease in generalisation error, which could happen by chance alone. We take advantage of the fact that the generalisation estimators described above are all based on averaging a statistics, and test whether the statistics associated with the generalisation estimators obtained for two different models have statistically significantly distinct means.

Inspection of equations (2), (3) and (4) shows that the can be written more generally:

$$\widehat{G} = \sum_t \epsilon(t) \tag{5}$$

where the sum runs over $|\mathcal{V}|$ (split-sample) or N (cross-validation and FPE) elements. $\epsilon(t)$ is a residual, the expression of which depends on the estimator:

SS: $\epsilon(t) = (y(t) - f(\mathbf{x}(t)))^2$

CV: $\epsilon(t) = (y(t) - f^{-t}(\mathbf{x}(t)))^2$

FPE: $\epsilon(t) = \frac{N+\widehat{P}}{N-\widehat{P}}(y(t) - f(\mathbf{x}(t)))^2$

We wish to test whether G_m, the generalisation error for the model with m delays (current best in the algorithm) is different from G_{m+1}, the generalisation error for the model with $m+1$ delays (augmented with the candidate delay). This is done on the basis of the residuals $\epsilon_m(t)$ and $\epsilon_{m+1}(t)$ and their average $\widehat{G}_m$ and $\widehat{G}_{m+1}$, which are unbiased estimators of G_m and G_{m+1} (respectively). Under the Gaussian noise assumption, the generalisation error is roughly χ^2 distributed [21]. Assuming that the difference between the residuals $\epsilon_m(t)$ and $\epsilon_{m+1}(t)$ is approximately normally distributed, and using the fact that all residuals are calculated on the same data, we can assess the significance of an observed decrease in generalisation error by using a one-tailed paired t-test [30] between the residuals.

For operational reasons, the amount of data available for both models will be different. Typically, the candidate model has an additional long-term delay, meaning that less input-output pairs $(\mathbf{x}(t), y(t))$ can be formed. In order to apply the paired t-test, it is thus necessary to discard the additional data examples used by the smaller model. The waste of a couple of data points is counter-balanced by the superior power of the paired t-test (compared to the non-paired version).

Finally, the algorithm requires a stop condition. This can be set in a heuristic way by specifying the maximum delay that we are willing to probe, or the maximum period allowed without delay selection. With roughly periodic data such as the Fraser river and the fMRI datasets below, for example, it is reasonable to decide that we want to go back only a limited number of periods.

4 Time Series Prediction

4.1 Hénon Map

Let us first consider a time series generated by the well-known Hénon map [16]:

$$y(t) = 1 - 1.4\,(y(t-1))^2 + 0.3\,y(t-2) \tag{6}$$

We use equation (6) to generate a dataset containing 500 data. Independently, a dataset with 10000 elements is generated in order to empirically test the generalisation abilities. Both the non-noisy and the noisy maps are investigated, with an additive Gaussian noise of variance $\sigma^2 = 0.1$. In the following experiments, we use two different models:

- a linear model using the FPE [1] as a generalisation estimator. This is obviously a bad choice when modelling the non-linear Hénon map (6).
- a kernel smoother [13, 35] using *leave-one-out* cross-validation (ie N-fold) as a generalisation estimator.

The ERD is compared to the results obtained by several competitors. The embedding dimension is estimated using the δ-test [29]. The traditional statistical approach is tested using an F-inclusion scheme, ie a forward method based on the F-statistic. In that scheme, a candidate delay is included if and only if:

$$F = \left(\frac{R_m(f) - R_{m+1}(f)}{N\, R_{m+1}(f)/(N-m-1)} \right) > F_\alpha \tag{7}$$

where $R_m(f)$ and $R_{m+1}(f)$ are the average risk for the models with m delays and $m+1$ delays (respectively), calculated on the same data, and F_α is the F threshold for an α confidence level, 1 and $(N-m-1)$ degrees of freedom. The interpretation of (7) is that a candidate is included when it leads to a significance improvement in observed performance.

Finally, in order to test the performance of our generalisation estimators, we generate a third, independent set in order to provide an additional unbiased estimate of the generalisation error similar to (2). This estimate is used for both model to decide which candidate delays should be included, using the paired t-test as above. Note that this "large validation set" or "divine guidance" method is of course impractical for real data. It is used here only in order to check the behaviour of the generalisation estimators that are used with the ERD.

Figures 1 and 2 show a plot proposed by Aleksic [3] to investigate the embedding dimension. Each point corresponds to a pair of data (t_1, t_2). All the points that are displayed have close inputs, ie $\|\mathbf{x}(t_1) - \mathbf{x}(t_2)\|$ is small. Clearly, with only one delay in the input (figure 1), close inputs do not guarantee close outputs. Therefore there can't be a continuous mapping between the time series values and their first delays. On the other hand, with the first two delays (figure 2), all points displayed show that data pairs with close inputs also have close outputs, which

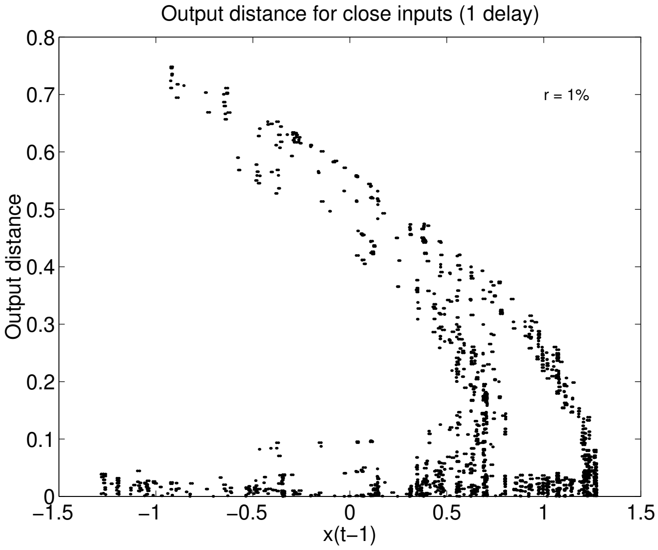

Figure 1: Plot of the output distance vs. input, when the input contains only the first delay. Each point represents a pair of data and only the points with the 1% smallest input distances have been included. Notice that many close inputs lead to distant outputs, indicating an insufficient input space.

Hénon map:	No noise		Noisy	
	Linear	Kernel	Linear	Kernel
LVS	1–7	1–2	1–7	1–3
ERD	1,3–6	1–2	1,3,4	1–3
F-inclusion	1–6	1–2	1–6,10	1–8,11–13, 16,17,19,20
δ-test	1–2		1–2	

Table 1: Delays selected on the Hénon map dataset by four methods: the "divine guidance" method using a large independent validation set (LVS); the proposed Extraction of the Relevant Delays (ERD) method (section 3); the F-inclusion method (equation 7) and the δ-test [29].

suggests that a continuous mapping between the first two delays and the time series value can be implemented.

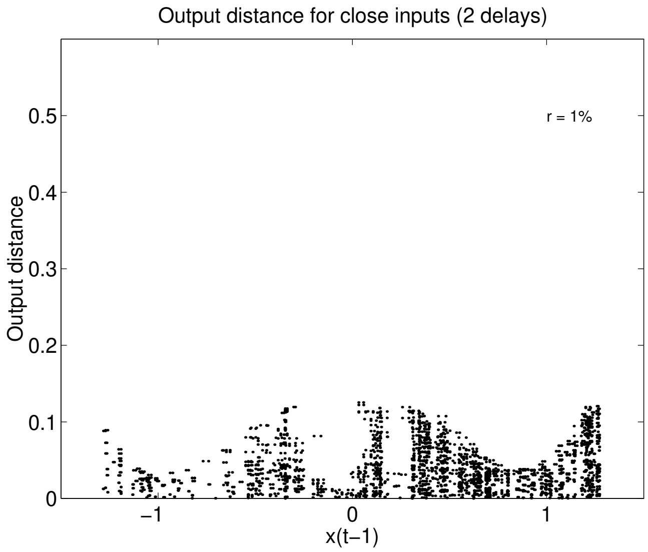

Figure 2: Plot of the output distance vs. input, when the input contains the first two delay. Each point represents a pair of data and only the points with the 1% smallest input distances have been included. For all points, the output distance is small, indicating a sufficient input space.

Accordingly, the δ-test selects the first 2 delays from our Hénon map dataset [29]. The other three methods select at least one additional delay, as shown in table 1. Finally, all resulting models are tested on the independent 10000 noise-free data in order to check their generalisation abilities (figure 3). Not surprisingly, the flexible kernel smoother provides much better modelling performance than the linear model. Note than in the noise-free case, all selection methods extract the first 2 delays and the kernel smoother is able to implement the Hénon map perfectly from the 500 available data. A second salient feature is that the δ-test manages to outperform the statistical methods only once (F-inclusion, for the kernel smoother on noisy data). This does by no means imply that the method is inefficient. On the contrary, it manages to efficiently extract the "true" delays from the data. However, as we will later argue, it does not address the relevant goal—prediction of future time series value—and suffers from this limitation. The close

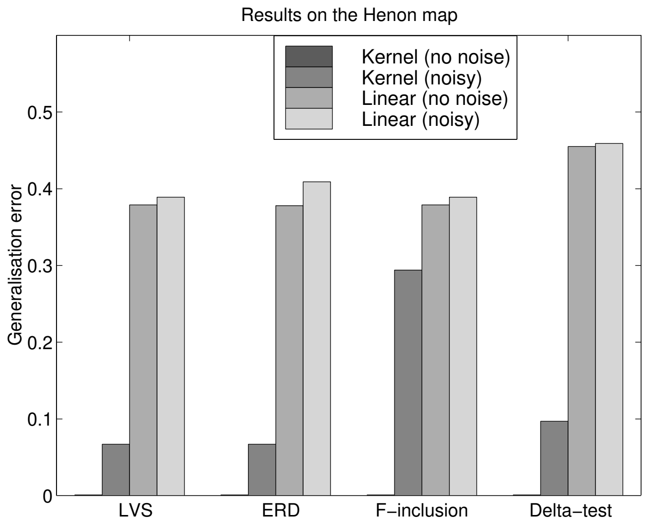

Figure 3: Results on the Hénon map, for both models and both noise conditions, of the 4 selections schemes: the "divine guidance" method using a large validation set (LVS); the proposed Extraction of the Relevant Delays (ERD) method (section 3); the F-inclusion method (equation 7) and the δ-test [29].

agreement between the large validation set (LVS) and the ERD results in figure 3 shows that the generalisation estimators that are used in the ERD algorithm perform their duty efficiently. The behaviour of the F-inclusion scheme with the kernel smoother on the noisy data suggests that this method is not well-adapted to non-linear models, and especially flexible models. This is due to the presence of the noise estimate in (7), which favours over-fitting. When the flexible model over-fits the data, $R(f)$ becomes overly small so that the denominator of (7) becomes small. As a consequence, the F-statistic is biased upwards, which leads to an overly optimistic selection (15 delays in that case, instead of 2 or 3 for the other three methods). Notice also on table 1 that ERD seems to favour parsimonious models (5 and 3 delays for the linear model, instead of 7 and 7 for LVS), while keeping good generalisation abilities.

Finally, it is interesting to notice that on the noisy data both the LVS and ERD methods select one more delay than the δ-test. This is due to the fact that we try to learn the underlying relationship on a limited number of data. In these conditions, the additional delay, though not theoretically necessary, improves the predictive performance by a significant 30%.

4.2 Fraser River

Let us now turn our attention to a real time series dataset. The data records mean monthly measurements of the flow of the Fraser river, in Hope (British Columbia, Canada) from march 1913 to December 1990. This dataset was analysed in [23] and is publicly available. It contains 946 values with a rough periodicity and maximum every 11 to 13 months. Figure 4 shows the measurements recorded between January 1925 and January 1955. The data set is split in half, so that the first 473 data are used for model estimation and delay extraction, and the last 473 are used for testing the generalisation abilities of the resulting model. In the following experiments, the data have been log-transformed and centred.

The selection method using the large validation set is of course not sensible in this context, as no extra data is available. Accordingly, we will consider only two selection schemes: the proposed Extraction of the Relevant Delays, and the δ-test. These will be applied using three models:

- A linear model,
- A kernel smoother,
- A non-linear neural network model, ie a multi-layer perceptron with one hidden layer.

The parameters of the linear and neural network models are estimated by minimising the mean squared error on the transformed data. The generalisation estimators are the FPE [1] and GPE [25], respectively. For the kernel smoother, the generalisation estimator is the leave-one-out estimator as above. The non-parametric δ-test selects 6 delays, while the ERD extracts 6 to 12 delays, depending on the model

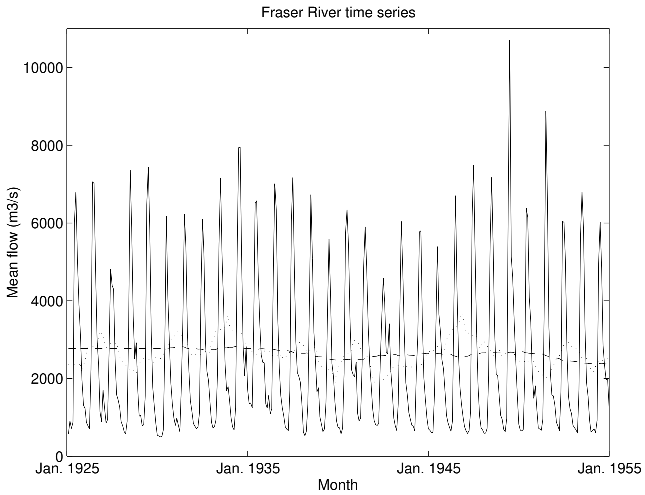

Figure 4: Mean monthly flow of the Fraser river recorded in Hope (British Columbia) between January 1925 and January 1955 (solid line). The twelve month and one year moving averages are shown as dotted and dashed lines, respectively.

Fraser river	Linear	Kernel	Neural net
ERD	1,2,4–7,10,11, 23,26,35,48	1,2,4,7,11,13	1,2,4,7,11,23
δ-test	1,2,4,7,8,11		

Table 2: Delays extracted, for each model, by both methods: the proposed Extraction of the relevant delays (ERD) and the δ-test.

(table 2). Not surprisingly, the more flexible models, ie kernel smoother and neural network, use less delay, while the linear model uses twice as many.

It is interesting to notice that, though the δ-test and ERD extract the same number of delays for two of our three models, these delays do not coincide. Typically, ERD probes further into the future: across 2 periods for the neural network, and even 4 for the linear model. This is an interesting artefact of the δ-test method: it tries to consider all pairs of data that have close inputs. As the number of inputs grows,

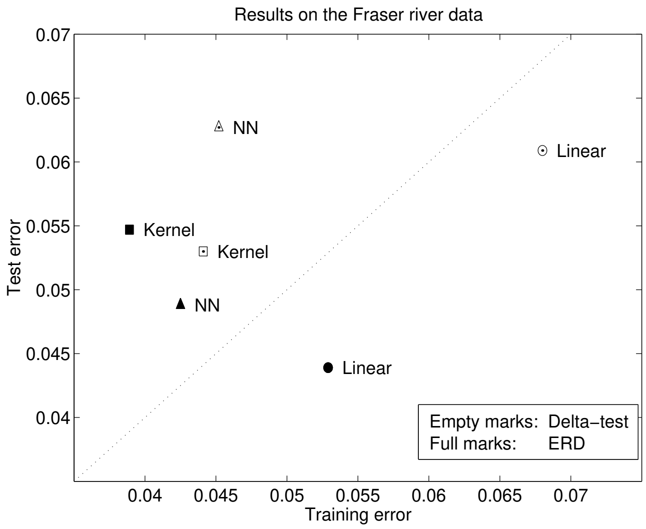

Figure 5: Result for 3 different models of both extraction schemes: the proposed Extraction of the Relevant Delays (ERD, full marks) and the δ-test (empty marks). The three models are: linear, kernel smoother and non-linear neural network (NN).

there are fewer and fewer of these pair that meet this requirement. This leads to a high variance of the estimated probabilities, and the test is unable to select additional delays.

The results of both schemes on all three models are measured by calculating the prediction error on the test set. Figure 5 shows the predictive performance plotted versus the training performance for both methods. Points far above the dotted line indicate probable over-fitting. Not surprisingly, the kernel smoother seems to over-fit the data quite severely. These models are known to suffer acutely from the curse of dimensionality, and have problems handling high dimensional inputs, especially when only one smoothing parameter is used. A surprising result is that the best performance overall is achieved by the combination of ERD and linear model. The flexible neural network model, even though it was trained with a combination of weight decay and Optimal Brain Surgeon (OBS) pruning [14], does better than the linear model on the training set, but worse on the test set. It should be noted that the performance reported in figure 5 depends mainly on two factors: the training algorithm used, and the extracted delays. With a good training algorithm, the neural network would in theory be able to outperform the linear model *using the same set of delays*. However, the ERD scheme has predictably not extracted the same delays for both models. In particular, the linear model uses 3 long-term delays (26, 35 and 48) that the neural network does not have access to.

Finally, these results illustrate again the limitations of the physical approach based on the estimation of the embedding dimension, which do not take into account the model used for predicting. The statistical approach based on the model performance allows to gather more information in the past, and yields an improvement in performance which is here most impressive for the linear model.

5 System Identification

We will here extend the ERD scheme to the extraction of the relevant delays in a simple system identification problem. The dataset is acquired during a functional Magnetic Resonance Imaging (fMRI) experiment. While the subject lies in the scanner, he is asked to either lie motionless (rest) or perform a simple motor task (activation). In this experiment, the task is left-handed finger-to-thumb opposition. Two 64x64 slices of whole brain echo-planar fMRI images are acquired every 2.5 seconds. The dimension of each voxel is $3.1 \times 3.1 \times 8$ mm. 9 runs of 48 images are acquired, in the following sequence: 12 images (30 seconds) of rest, 24 images (60 seconds) during activation, and 12 images of rest again. The complete time series consists of 432 measurements (18 minutes), in each voxel. In the following experiments, we analyse one slice and focus on the 39×49 area containing the brain.

The fMRI signal measures the haemodynamic response to focal neuronal activation. It is widely believed that the response in the brain can be characterised as a convolution of the reference stimulus signal representing the activation with a linear filter. In our experiment, the reference stimulus $h(t)$ is a square wave with positive values for the scans acquired while the subject was performing the motor task and negative values during rest (figure 6). Previous approaches have tried

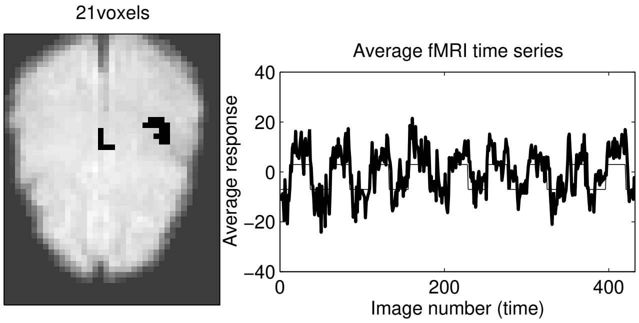

Figure 6: Right: brain map indicating the 21 activated voxels (black) analysed with ERD. The background is an anatomical reference obtained from the average activation. Left: average time series for the activated voxels (thick line) and reference stimulus (thin black).

to characterise the haemodynamic response using several convolution filters: Poisson [7], Gamma [19] or FIR filter with a fixed length [28]. We will try to extend the latter to general FIR filters by extracting the delays that yield the best modelling performance. In other words, we are looking for the set of delays $d_1, \ldots d_m$ such that:

$$\widehat{y}(t) = \sum_{i=1}^{m} \beta_i \, h(t - d_i) \tag{8}$$

models the observed fMRI signal $y(t)$ as well as possible. The extension of the ERD algorithm presented in section 3 is straightforward: In step 2 of the algorithm, the delays added to the inputs should be the delays of the stimulus reference $h(t)$ rather than of the time series $y(t)$. The modelling and generalisation estimations are independent of the particular choice of inputs and the rest of the algorithm is not affected by this extension.

Forming the cross-correlation between the time series $y(t)$ and the reference signal $h(t)$, we isolate 21 voxels which display a significant activation (figure 6). This pre-processing step isolates two groups of voxels, covering the primary motor area and the SMA. As expected, the left-handed opposition produces a contra-lateral activation in the right hemisphere.

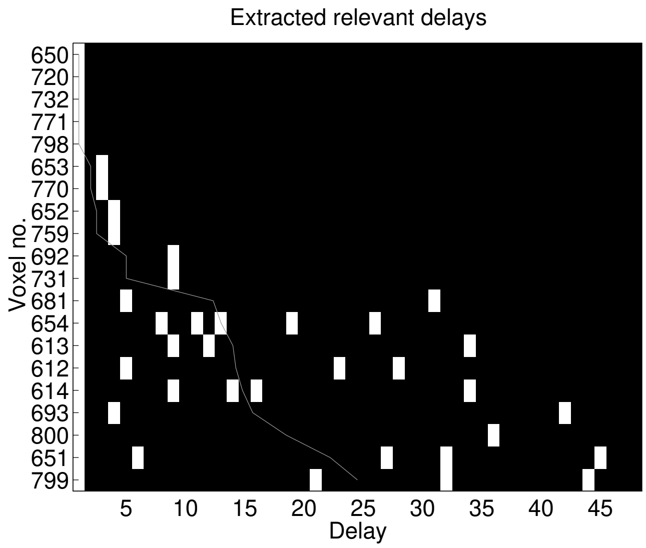

Figure 7: Delays extracted for each voxel (white squares). The gray line indicates the average of the delay indexes.

The ERD algorithm is run on each voxel in order to extract the delays that allow the best modelling of the fMRI signal from the reference signal. 5 out of the 9 runs, corresponding to 240 data points, are used for delay extraction and modelling, while the remaining 4 runs (192 points) are used to test the generalisation abilities of the resulting model. The stop condition is set by limiting the extraction to 48 delays, so that the FIR filter can probe at most one run in the past. In order to loosen the selection criterion, and in agreement with standard statistical practice in subset selection [6], the significance level of the paired t-test is set to 80%. The performance of the filter using the extracted delays will be compared to a FIR filter of fixed size. As the average haemodynamic delay has been observed between 5 and 10 seconds [4], ie 2 to 4 images here, we will use a FIR filter with 7 delays as reference.

Figure 7 shows the delays extracted for each voxel. They range from 1 (selected for all voxels) to 45. The average delay is estimated by computing the weighted mean of the delays indices d_i, with the absolute filter coefficients β_i as weights. According to this average value, the

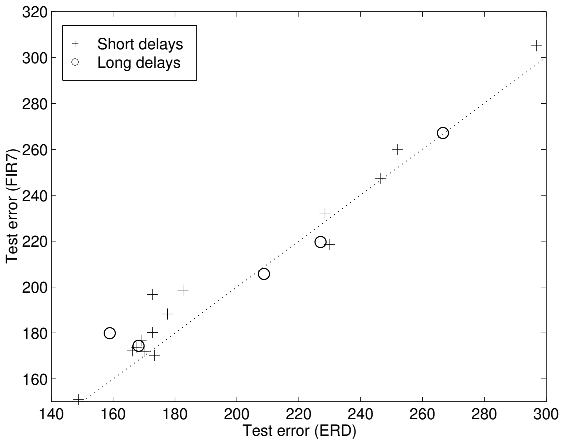

Figure 8: Test error for the 7 parameter FIR filter versus test error for the FIR filter extracted with ERD. The crosses indicate voxels for which the average delay is less than 10 seconds. The circles indicate voxels with longer-term delays (more than 15 seconds). One voxel with very large test error in both cases has been discarded to improve clarity.

activated voxels can be roughly divided in 2 categories: 15 out of the 21 time series have an average delay between 3 and 10 seconds, which is roughly within the accepted range of 5–10 seconds. The remaining voxels have average delays above 15 seconds, which suggests that the extracted delays do not make sense from a biological point of view, even though they manage to efficiently improve the modelling of the relationship between the reference signal $h(t)$ and the fMRI $y(t)$.

The performance of the ERD scheme is contrasted with that of the fixed-length FIR filter using seven delays, ($d_i = i, \forall 1 \leq i \leq 7$). Figure 8 plots the test error for the fixed-length FIR filter against the test error for the FIR filter using the delays extracted by ERD. Points above the dotted diagonal show that the test error of the fixed-length filter is higher, ie the modelling is worse. Clearly, most voxels benefit from using extracted delays, even though a few display a small decrease in performance. Overall, it should be noted that the ERD filters yield better performance, while using fewer parameter.

Finally, figure 9 displays the location and average fMRI activation

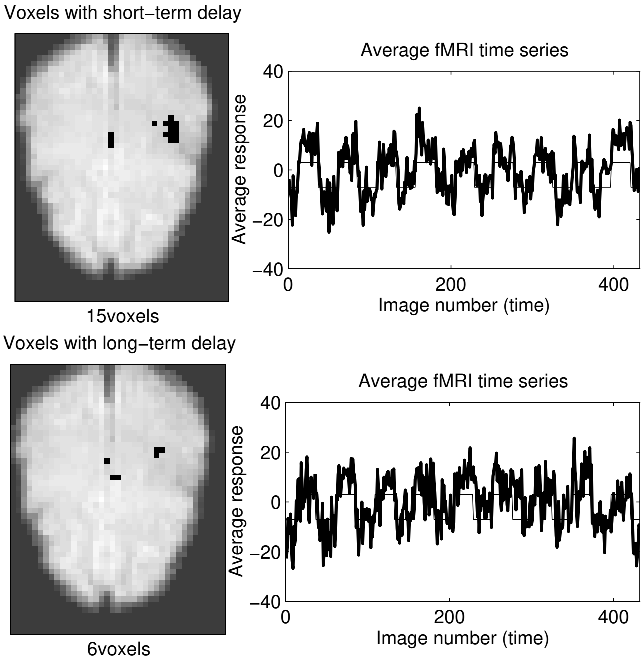

Figure 9: Voxels with short-term delays (top), between 3 and 10 seconds, and voxels with long-term delays (bottom), more than 15 seconds, with no biological explanation. The brain maps on the left show the location of the analysed voxels (black dots); the right plot shows the average fMRI signal in these voxels (thick) and the reference signal (thin).

signal of both groups of voxels. The voxels with short-term delays (top) and long-term delays (bottom) seem to have slightly different patterns of activation, especially in the last four runs. However, the small sample sizes (15 and 6 respectively) makes a strict comparison difficult.

6 Discussion and Conclusions

The main difference between the δ-test and the Extraction of the Relevant Delays is a matter of approach. The physical approach investigates the "true" delays, ie those with a probable physical meaning. This is what we called the *primary* delays above. The main advantage of this approach is that it is probably the closest we can get in terms of interpretation of the selected delays. On the other hand, the statistical approach embodied in the ERD method focuses on performance. It extracts the delays that yield a model with the best generalisation, by directly optimising the relevant performance criterion. Though the resulting delays are not as easily interpreted, the resulting predictive performance is improved. this is exemplified by the fMRI example where a quarter of the filters have no apparent biological justification. In problems like time series prediction and system identification, good modelling performance is usually what we are after, especially when the model is knowingly not a parametric model of the system.

This raises a related issue. The physical approach makes no assumption on the underlying model with which we intend to predict the time series or the system. This means that delays need be extracted only once when we wish to compare several models on the same dataset as above. On the other hand, the output of the ERD method is totally dependent on the model. This means that the extraction of the relevant delays must be repeated for each model we wish to use. Note however that the ERD scheme is more straightforward to implement and usually less demanding computationally than its physical counterparts (δ-test, Lipschitz quotients, ...). In particular, extraction of the relevant delays for the simple linear model is extremely fast.

An interesting side-effect is that the ERD method takes naturally into account the limited dataset size. Indeed, the estimation of the model parameters are done on a limited number of data, leading to a possible variability of the model estimate. A similar effect is known from the use of AIC [2] and BIC [32] information criteria: BIC selects the asymptotically optimal model size, which AIC tends to over-estimate. However, AIC effectively minimises the average generalisation error for finite datasets. Similarly, the ERD method selects additional delays, which, though there are not "true" delays, can help predict the time series. For the Hénon map, the methods based on generalisation estimators include an additional delay $d = 3$, which leads to a sizeable improvement in predictive performance. This example also shows that the generalisation estimators (used in ERD) can pick up the right model size, or rather the model size that generalises best, according to both the "Large Validation Set" and the test set. Generalisation estimation is crucial part of the ERD method, and it should be emphasised that we are actually interested in the *location* of the minimum generalisation, rather than the precise *value* of the error. The ERD scheme could very well accommodate an estimator with a consistent bias.

The Extraction of Relevant Delays method presented in this contribution is essentially a statistical forward selection method. It would be straightforward to apply the same scheme to backward elimination, using the loss in generalisation as an evaluation criterion. However, as we argued in section 3, this is probably badly suited to the temporal modelling problems addressed here. On the other hand, a straightforward extension would be the design of a method similar to stepwise regression, with alternate forward and backward cycles. In many models, this can be addressed during the model identification or estimation procedure. For Kernel smoothers, the use of an adaptive metric [12] will handle irrelevant inputs. For linear models or neural networks, an interesting option is the use of a regularisation term with a pruning effect [9, 11]. These methods allow to discard variables that have been included in the input vector but turn out to be irrelevant at a later point.

An interesting prospect for future research is to extend the ERD scheme to more general system identification architecture. The Auto-Regressive filter with eXogenous input (ARX) would be a natural candidate. Note however that in that case the filter input consists of two time series values. The precise ordering of the candidate inputs and the way in which one should alternate between delays from the system output $y(t)$ and the control signal $u(t)$ have to be determined.

As a conclusion we have presented a generalisation-based method for the extraction of the relevant delays in temporal modelling, with a focus on time series prediction and system identification. This method is related to traditional forward selection methods, and is therefore well-founded. It is easy to implement and requires little time in addition to model estimation. Furthermore, it accommodates efficiently the trade-off between number of selected delays (model size) and model flexibility (approximation error). The method provides interesting results on a number of experiments, performed in different contexts.

Acknowledgements

The Fraser river dataset (section 4.2) is publicly available on statlib: `http://lib.stat.cmu.edu/datasets/`. The fMRI dataset (section 5) was acquired by Robert Savoy at Massachusets General Hospital (Boston). The author thanks Patrick Gallinari, Jan Larsen and the learning group at IMM, DTU, for valuable discussions on previous versions of this work. This work was supported by a research fellowship from the Technical University of Denmark.

References

[1] H. Akaike. Fitting autoregressive models for prediction. *Annals of the Institute of Statistical Mathematics*, 21:243–247, 1969.

[2] Hirotugu Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, december 1974.

[3] Zoran Aleksić. Estimating the embedding dimension. *Physica D*, 52:362–368, 1991.

[4] Peter A. Bandettini, A. Jesmanowicz, Eric C. Wong, and James S. Hyde. Processing strategies for time-course data sets in functional MRI of the human brain. *Magnetic Resonance in Medicine*, 30(2):161–173, august 1993.

[5] R. Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks*, 5(4):537–550, 1994.

[6] Robert B. Bendel and A. A. Afifi. Comparison of stopping rules in forward "stepwise" regression. *Journal of the American Statistical Association*, 72(357):46–53, March 1977.

[7] K.J. Friston, P. Jezzard, and R. Turner. Analysis of functional MRI time series. *Human Brain Mapping*, 1:153–171, 1994.

[8] Stuart Geman, Elie Bienenstock, and René Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4(1):1–58, January 1992.

[9] Cyril Goutte. On the use of a pruning prior for neural networks. In S. Usui, Y. Tohkura, S. Katagiri, and E. Wilson, editors, *Neural Networks for Signal Processing VI – Proceedings of the 1996 IEEE Workshop*, number VI in NNSP, pages 52–61, Piscataway, New Jersey, 1996. IEEE.
`ftp://eivind.imm.dtu.dk/dist/1996/goutte.nnsp96.abs`.

[10] Cyril Goutte. Lag space estimation in time series modelling. In *Proceedings of ICASSP'97*, pages 3313–6. IEEE, 1997.
`ftp://eivind.imm.dtu.dk/dist/1997/goutte.icassp97.ps.gz`.

[11] Cyril Goutte and Lars Kai Hansen. Regularization with a pruning prior. *Neural Networks*, 10(6):1053–1059, 1997.
`ftp://eivind.imm.dtu.dk/dist/1996/goutte.regularization.ps.Z`.

[12] Cyril Goutte and Jan Larsen. Adaptive metric kernel regression. In *Neural Networks for Signal Processing VIII – Proceedings of the 1998 IEEE Workshop*, number VIII in NNSP, page in press, Piscataway, New Jersey, 1998. IEEE.
`ftp://eivind.imm.dtu.dk/dist/1998/goutte.nnsp98.html`.

[13] W. Härdle. *Applied nonparametric regression*. Number 19 in Econometric Society Monographs. Cambridge University Press, 1990.

[14] B. Hassibi and D. G. Stork. Second order derivatives for network pruning: Optimal brain surgeon. In S.J. Hanson, J.D. Cowan, and C.L. Giles, editors, *Advances in Neural Information Processing Systems*, volume 5 of *NIPS*, pages 164–171. Morgan Kaufmann, 1993.

[15] X. He and H. Asada. A new method for identifying orders of input-output models for nonlinear dynamic systems. In *American Conference on Control*, San Francisco, California, 1993.

[16] M. Hénon. *Comm. Math. Phys. Rev.*, 50(69), 1976.

[17] R. R. Hocking. The analysis and selection of variables in linear regression. *Biometrics*, 32:1–49, March 1976.

[18] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–368, 1989.

[19] Nicholas Lange and Scott L. Zeger. Non-linear Fourier time series analysis for human brain mapping by functional magnetic resonance imaging. *Journal of the Royal Statistical Society, Series C, Applied Statistics*, 46(1), 1997.

[20] J. Larsen and L. K. Hansen. Generalized performance of regularized neural networks models. In J. Vlontzos, J. N. Hwang, and E. Wilson, editors, *Neural Networks for Signal Processing IV – Proceedings of the 1994 IEEE Workshop*, number IV in NNSP, pages 42–51, Piscataway, New Jersey, 1994. IEEE.

[21] J. Larsen and L. K. Hansen. Empirical generalization assessment of neural network models. In F. Girosi, J. Makhoul, E. Manolakos, and E. Wilson, editors, *Neural Networks for Signal Processing V – Proceedings of the 1995 IEEE Workshop*, number V in NNSP, pages 42–51, Piscataway, New Jersey, 1995. IEEE.

[22] Philippe Leray and Patrick Gallinari. Feature selection with neural networks. Technical report LIP6 1998/012, LIP6, 1998. to appear in Behaviormetrika.

[23] A. I. McLeod. Diagnostic checking of periodic autoregression models with application. *Journal of Time Series Analysis*, 15(2):221–233, 1994.

[24] Christophe Molina, Nick Sampson, William J. Fitzgerald, and Mahesan Niranjan. Geometrical techniques for finding the embedding dimension of time series. In *Neural Networks for Signal Processing VI – Proceedings of the 1996 IEEE Workshop*, pages 161–169, 1996.

[25] J. Moody. Note on generalization, regularization and architecture selection in nonlinear learning systems. In B. H. Juang, S. Y. Kung, and C. A. Kamm, editors, *Proceedings of the first IEEE Workshop on Neural Networks for Signal Processing*, number I in NNSP, pages 1–10, Piscataway, New Jersey, 1991. IEEE.

[26] N. Murata, S. Yoshizawa, and S. Amari. Network Information Criterion—determining the number of hidden units for an artificial neural network model. *IEEE Transactions on Neural Networks*, 5(6):865–872, 1994.

[27] P.M. Narendra and K. Fukunaga. A branch and bound algorithm for feature subset selection. *IEEE Transactions on Computers*, 26(9):917–922, 1977.

[28] Finn Årup Nielsen, Lars Kai Hansen, Peter Toft, Cyril Goutte, Niels Mørch, Claus Svarer, Robert Savoy, Bruce Rosen, Egill Rostrup, and Peter Born. Comparison of two convolution models for fMRI time series. In L. Friberg, A. Gjedde, S. Holm, N. A. Lassen, and M. Nowak, editors, *Third International Conference on Functional Mapping of the Human Brain*, *Neuroimage* 3(3), part 2, page S473. Academic Press, May 1997. `http://eivind.imm.dtu.dk`.

[29] Hong Pi and Carsten Peterson. Finding the embedding dimension and variable dependences in time series. *Neural Computation*, 6(3):509–520, May 1994.

[30] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 2nd edition, 1992.

[31] R. Savit and M. Green. Time series and dependent variables. *Physica D*, 50(1):95–116, 1991.

[32] Gideon Schwartz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.

[33] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society B*, 36:111–147, 1974. with discussion.

[34] G. Toussaint. Bibliography on estimation of misclassification. *IEEE Transactions on Information Theory*, 20(4):472–479, July 1974.

[35] M.P. Wand and M.C. Jones. *Kernel Smoothing*. Number 60 in Monographs on Statistics and Applied Probability. Chapman & Hall, London, 1995.