Semi-parametric nonlinear regression and transformation using functional networks

Enrique Castillo\textsuperscript{a}, Ali S. Hadi\textsuperscript{b,∗}, Beatriz Lacruz\textsuperscript{c}, Rosa E. Pruneda\textsuperscript{d}

\textsuperscript{a}Department of Applied Mathematics and Computational Sciences, University of Cantabria and University of Castilla La Mancha, Spain
\textsuperscript{b}Department of Mathematics, The American University in Cairo, 113 Kasr El-Aini Street, P.O. Box 2511, Cairo 11511, Egypt
\textsuperscript{c}Department of Statistical Methods, University of Zaragoza, Spain
\textsuperscript{d}Department of Mathematics, University of Castilla-La Mancha, Spain

Received 14 August 2006; received in revised form 6 July 2007; accepted 10 July 2007
Available online 19 July 2007

Abstract

Functional networks are used to solve some nonlinear regression problems. One particular problem is how to find the optimal transformations of the response and/or the explanatory variables and obtain the best possible functional relation between the response and predictor variables. After a brief introduction to functional networks, two specific transformation models based on functional networks are proposed. Unlike in neural networks, where the selection of the network topology is arbitrary, the selection of the initial topology of a functional network is problem driven. This important feature of functional networks is illustrated for each of the two proposed models. An equivalent, but simpler network may be obtained from the initial topology using functional equations. The resultant model is then checked for uniqueness of representation. When the functions specified by the transformations are unknown in form, families of linear independent functions are used as approximations. Two different parametric criteria are used for learning these functions: the constrained least squares and the maximum canonical correlation. Model selection criteria are used to avoid the problem of overfitting. Finally, performance of the proposed method are assessed and compared to other methods using a simulation study and several real-life data.

© 2007 Elsevier B.V. All rights reserved.

Keywords: Adjusted correlation coefficient; Akaike information criterion; Alternating conditional expectation; Canonical correlation; Constrained least squares; Minimum description length measure; Transforming both sides

1. Introduction

The problem of investigating the functional relationship between a response variable $Y$ and one or more predictor variables $X_1, X_2, \ldots, X_k$ is of interest. In many practical situations, the form of such a functional relationship is unknown. Additionally, discovering such a relationship may also require transformation of the response and/or the predictor variables. The problem of transformation in regression modeling has attracted considerable attention in the literature. See, for example, Atkinson (1985), Carroll and Ruppert (1988), and the references therein.

In this paper, we are mainly interested in solving two problems:

∗ Corresponding author. Tel.: +202 2797 5185; fax: +202 2797 7565.
E-mail addresses: castie@unican.es (E. Castillo), ahadi@aucegypt.edu (A.S. Hadi), lacruz@unizar.es (B. Lacruz), rosa.pruneda@uclm.es (R.E. Pruneda).
Problem 1: Transforming prediction variables (modeling). When the goal of the analysis is to model \( Y \) as a function of \( X_1, X_2, \ldots, X_k \), the transformation involves only the predictor variables. Here we assume that the relationship between the response and predictor variables can be expressed as

\[
Y = h(X_1, X_2, \ldots, X_k) + \varepsilon,
\]

where \( \varepsilon \) is a random error whose expected value is assumed to be 0. Then, the problem is to discover the structure of the function \( h \) in (1).

Problem 2: Transforming response variable. In some applications transformation may involve both the response and predictor variables. In this case, we assume that the relationship between the response and predictor variables can be written as

\[
f(Y) = h(X_1, X_2, \ldots, X_k) + \varepsilon,
\]

where \( \varepsilon \) is a random error whose expected value is assumed to be 0. Our task here is to discover the structure of transformations \( f \) and \( h \) in (2).

There are several ways for the formulation and estimation of the models in (1) and (2). On the one hand, one may first consider the classical multiple linear regression

\[
Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + \varepsilon,
\]

where \( \beta_0, \beta_1, \ldots, \beta_k \) are the parameters to be estimated from the data. Then, if needed, a nonlinear transformations of the response and/or the explanatory variables are commonly used in regression problems to achieve certain desirable conditions such as linearity, normality, and homoscedasticity. In this case, (3) can be written as

\[
f(Y) = \beta_0 + \beta_1 h_1(X_1) + \beta_2 h_2(X_2) + \cdots + \beta_k h_k(X_k) + \varepsilon,
\]

and now the problem consists of estimating the parameters \( \beta_0, \beta_1, \ldots, \beta_k \) and to discover the transformations \( h_1, h_2, \ldots, h_k \), and \( f \) (some of them can be the identity function).

The models in (3) and (4) are parametric models and they have considerable success in a variety of applications, despite their rigid linear form for \( h \) in (1) or (2). These forms, however, may not be appropriate for many more complex situations.

On the other hand, nonparametric methods, which make minimal assumptions about the form of \( h \) in (1) or (2), can be used. For example, a nonparametric additive model is obtained by replacing the linear function \( \beta_j X_j \) in (3), \( j = 1, \ldots, k \), by a nonlinear function \( h_j(X_j) \) to get

\[
Y = h_1(X_1) + h_2(X_2) + \cdots + h_k(X_k) + \varepsilon,
\]

or, when nonlinear transformation of the response is necessary,

\[
f(Y) = h_1(X_1) + h_2(X_2) + \cdots + h_k(X_k) + \varepsilon.
\]

This generalization retains some important features of the linear approximation: It is additive in the explanatory variables effect and it allows us to separately examine the roles of the explanatory variables in modeling the response. The nonlinear functions \( h_j, j = 1, \ldots, k \), in (5) are smooth functions and an iterative procedure is needed to fit the model. Also, more complex than simple component-wise additive functions can be used in the additive model (see, e.g., Hastie and Tibshirani, 1990; Friedman and Stuetzle, 1981).

When the transformation of the response variable is also needed, the function \( f \) in (6) can also assumed to be an arbitrary smooth function. In this case, techniques for fitting the model in (6) include alternating conditional expectation (ACE) algorithm (Breiman and Friedman, 1985) and additivity and variance stability (AVAS) transformation procedure (Tibshirani, 1988).

Other nonparametric techniques such as piecewise and local multiple regression, regression trees and multivariate adaptive regression splines (MARS) tackle the approximation of \( h \) in (1) by fitting several simple parametric functions in different pieces of the observed domain (see, for example, Friedman, 1991; Fox, 2000; and the references therein).
Finally, **neural networks** also provide a flexible way to approximate $h$ in (1), but they cannot be used in (2). In its simplest form (one-layer perceptron), this technique proposes a model of the form

$$Y = g \left( \sum_{j=1}^{p} w_{j} h_{j} \left( \sum_{i=1}^{k} w_{ij} X_{i} \right) \right) + \varepsilon,$$

(7)

where $g$ and $h_{j}$ are fixed functions (usually, sigmoidal, logistic, sign or linear functions and $h_{j} = h, \forall j$), called **neural functions**, and the $w$’s are parameters to be estimated from the data. The arguments of the functions are called **neurons**. The number, $p$, of neural functions is arbitrarily chosen. A better understanding of this approximation is obtained when the neural network is represented graphically. **Fig. 1** shows the usual graphical representation of (7) for $k=2$ predictor variables (inputs), $p=3$ neural functions and $h_{j}=h, \forall j$. Arrows indicate the direction of the information flow and $v_{1}, v_{2}$ and $v_{3}$ are **neurons** representing the values given by each $h$. A more detailed representation is shown in **Fig. 2**.

In this paper we propose a new approach using functional networks to solve the transformation problems in (1) and (2). For a general introduction to functional networks, see Castillo et al. (1998, 2000). For other applications of functional networks, see Castillo et al. (2001).

Several advantages of using functional networks in solving these transformation problems are presented and the performances of the proposed procedure are illustrated by a simulation study and by real-life data sets found in the literature on transformations.

The rest of the paper is structured as follows. In Section 2, functional networks are explained by an example. This includes the selection of the initial topology of the functional network, which is derived from the nature of the problem at hand, and the simplification of the initial topology, whenever possible, which is based on the solution of a set of functional equations. Section 3 presents two specific functional network models for handling the transformation problems in (1) and (2). These are the additive and the general models. The uniqueness of representations of each of
these models are discussed in Section 4. Techniques for learning the parameters of these models are given in Section 5, where several examples of linearly independent families are considered and two criteria of estimation (the constrained least squares (CLS) and the maximum canonical correlation (MCC)) are suggested. Section 6 presents a criterion for model selection, which avoids the problem of overfitting. Section 7 investigates the properties of the proposed methods using simulation study and several examples of nonlinear models. Section 8 illustrates the methodology using several real-life data sets taken from the transformation literature and compares the obtained results with those found in the literature using other methods. A summary and some concluding remarks are given in Section 9.

2. Functional networks

Functional networks can be seen as a graphical representation of some functional equations, which provide a better understanding of the properties of the given model. Consider, for example, the functional equation associated with the nonparametric additive model given in (6), where \( f \) is an invertible function. The functional network representing this model is drawn in Fig. 3. The structure of functional networks is similar to that of neural networks (specially if we look at the extended graphical representation in Fig. 2). The main difference is that the functional network topology is constructed based on the problem at hand instead of being arbitrarily selected based on trial and error (e.g., the number of hidden layers and the number of nodes in each hidden layer). Other differences are mentioned below.

In a functional network, data are represented by small filled circles called storing units. In Fig. 3, for example, we have four layers of storing units: the first (left) layer contains \( k \) input units \( (x_1, x_2, \ldots, x_k) \), the second layer consists of \( k \) intermediate units \( (u_1, u_2, \ldots, u_k) \), the third layer contains one intermediate unit \( (v) \), and the fourth (right) consists of one output unit \( (y) \).

Functions are represented by open circles with their name inside the circles. These are called functional units or neural functions. Functional units evaluate a set of input values and return a set of output values to the next layer of storing units. The functional network in Fig. 3, for example, has three layers of neurons. The first layer consists of \( k \) neurons: \( h_1, h_2, \ldots, h_k \), which evaluate \( x_1, x_2, \ldots, x_k \), respectively, and returns the corresponding values \( u_1, u_2, \ldots, u_k \) to the next layer of storing units. The second layer has only one functional unit, which returns \( v \), (in this case, \( v = u_1 + u_2 + \cdots + u_k \)). The third layer has the functional unit \( f^{-1} \) which computes \( v \) and returns \( y \). Note that neural functions in functional networks are not fixed as they are in neural networks, but learned exactly or approximately from the available data. Moreover, they can be different, multivariate, and/or multargument (not restricted to be a linear combination of the inputs).

The functional units are connected to the storing units by directed arrows. The arrows indicate the direction of the information flow. The functional network in Fig. 3 does not have any converging arrows to an intermediate or output unit, but if a functional network contains converging arrows, this indicates that the neurons (functions) from which they emanate must produce identical outputs. This is an important feature of functional networks that is not available in neural networks. Converging arrows represent constraints which arise from physical and/or theoretical characteristics of the problem at hand. We shall see one example of such constraints later in this paper.

The specification of the above elements of a functional network is referred to as model specification or structural learning. Thus, the initial topology of the functional network is problem driven. Other examples of this important feature are given in Section 3.

Furthermore, as we shall see below, if the initial topology of a functional network contains converging arrows, it is possible to obtain an equivalent but simpler functional network using functional equations. This is illustrated later in

---

*Fig. 3. Functional network associated with the nonparametric additive model in (5), where \( f \) is an invertible function.*
the paper. For a general introduction to functional equations and methods to solve them see, for example, Aczél (1966), Castillo and Ruiz-Cobo (1992) or Castillo et al. (2005).

It can be seen from the above example that a functional network, in general, represents an equation where the unknowns are functions such as those in (3)–(7). This means that all the regression techniques mentioned above (including neural networks) can be represented by a functional network. In particular, our approach can be seen as a generalization and extension of neural networks as we will discuss in Section 3.

Once the initial topology is constructed and simplified, if possible, one needs to address the following questions:

1. Are the neural functions \((h_1, h_2, \ldots, h_k, f)\) in Fig. 3 unique? In other words, is there another set of functions \((e.g., h^*_1, h^*_2, \ldots, h^*_k, f^*)\) that produces the same output \((y)\) for the same input \((x_1, x_2, \ldots, x_k)\)? If the answer to this question is yes, that is, if \(f^{-1}(h_1(x_1) + h_2(x_2) + \cdots + h_k(x_k)) = f^{*-1}(h^*_1(x_1) + h^*_2(x_2) + \cdots + h^*_k(x_k))\), for all values of \(x_1, x_2, \ldots, x_k\), what are the constraints that must be satisfied for the simplified functional network to produce unique estimates of the neural functions \(h_1, h_2, \ldots, h_k\) and \(f\)?
2. How the neural functions of the simplified functional network can be learned from the available set of data?
3. How to select and validate the best model(s)?

The answers to these questions are given in Sections 4, 5, and 6, respectively.

For some examples of application of functional networks, the reader is referred to Castillo (1998), Castillo and Gutiérrez (1998) or Castillo et al. (1999).

3. Proposed transformation models

The first step in functional networks is the specification of the initial topology. As mentioned above, the initial topology of a functional network is problem driven. Our problem here is to find the functions in the transformations models in (1) and (2). In this section we propose two models for the transformation: (a) the additive model and (b) the general model. These models and their corresponding functional networks are presented below.

3.1. The additive model

An interesting justification of the additive model is as follows. Suppose that the information contained in the variable values \(x_1, x_2, \ldots, x_k\) leads to a single value \(h(x_1, x_2, \ldots, x_k)\) for \(y\) and we obtain the variables sequentially. Say, we first get \(x_1\) and \(x_2\), and that we summarize the information these two variables have about \(y\) as a single real number, that can be obtained as a function of the two variable values, \(G_{12}(x_1, x_2)\). If next we know the values of the remaining variables \(x_3, x_4, \ldots, x_k\), then we have

\[
y = h(x_1, x_2, \ldots, x_k) = H_{12}(G_{12}(x_1, x_2), x_3, x_4, \ldots, x_k), \tag{8}
\]

i.e., a function \(H_{12}\) exists such that the value of \(y\) can be calculated from \(G_{12}(x_1, x_2)\) and the values of the remaining variables.

However, we can get first the values of any other two variables, \(x_i\) and \(x_j\), and we can, similarly, summarize the information they have on \(y\), as a function, \(G_{ij}(x_i, x_j)\), of them, and next get the values of the remaining variables. Since the final result must be independent of the order in which we receive the information, we must have

\[
y = h(x_1, x_2, \ldots, x_k) = H_{ij}(G_{ij}(x_i, x_j), x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_{j-1}, x_{j+1}, \ldots, x_k), \tag{9}
\]

\(\forall i, j\). For example, for \(k = 3\), the system of equations (9) becomes

\[
H_{12}(G_{12}(x_1, x_2), x_3) = H_{13}(G_{13}(x_1, x_3), x_2) = H_{23}(G_{23}(x_2, x_3), x_1). \tag{10}
\]

Fig. 4 illustrates the functional network associated with Eq. (10). Note that the functional equations in (10) give rise to the converging arrows in the output \(y\). Then, the structure of this functional network can be simplified using functional equations. The solution of the system of functional equations (9) leads to the additive model, as we see in the following theorem.
Theorem 1 (Additive model). The solution of the system of functional equations in (9) is

\[ H_{ij}(u, x_1, \ldots, x_{i-1}, x_i+1, \ldots, x_{j-1}, x_j+1, \ldots, x_k) = \left[ a_{ij}(u) + \sum_{s \neq i; s \neq j} h_s(x_s) \right]^{-1}, \quad (11) \]

\[ G_{ij}(x_i, x_j) = a_{ij}^{-1}[h_i(x_i) + h_j(x_j)], \quad (12) \]

where the functions \( h_i(x_i) \) are arbitrary, and the functions \( f \) and \( a_{ij}(u) \), \( \forall i, j \), are arbitrary invertible functions.

Replacing (12) into (11) we get the following additive model:

\[ f(y) = h_1(x_1) + h_2(x_2) + \cdots + h_k(x_k), \quad (13) \]

where \( h_1, h_2, \ldots, h_k \) are arbitrary functions, and \( f \) is an arbitrary invertible function.

The model in (13) can be represented by the functional network in Fig. 3. Note that the two functional networks in Figs. 3 and 4 are equivalent because they return the same outputs for the same inputs. But the functional network in Fig. 3 is simpler than the one in Fig. 4.

3.2. The general model

Several other alternatives are possible for selecting the initial topology of functional networks (see for example the introductory part of Chapter 2 in Castillo et al., 1998). Nevertheless, when the user has no information at all about the problem at hand, a general model of the type

\[ y = \sum_{r_1=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} c_{r_1 \ldots r_k} \phi_{r_1}(x_1) \cdots \phi_{r_k}(x_k) \quad (14) \]

can be used, where \( c_{r_1 \ldots r_k} \) are unknown parameters (coefficients) and the sets of functions \( \Phi_s = \{ \phi_{r_s}(x_s), r_s = 1, 2, \ldots, q_s \}, s = 1, 2, \ldots, k \), are linearly independent. An example of this model, for \( k = 2 \) and \( q_1 = q_2 = q \), is shown in Fig. 5.

A generalization of model (14), that includes transformation of the response variable \( y \), is

\[ f(y) = \sum_{r_1=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} c_{r_1 r_2 \ldots r_k} \phi_{r_1}(x_1) \phi_{r_2}(x_2) \cdots \phi_{r_k}(x_k), \quad (15) \]

where \( f(y) \) is a transformation of \( y \). The corresponding functional network, for \( k = 2 \), and \( q_1 = \cdots = q_k = q \), is shown in Fig. 6. Note that since there are no converging arrows in the functional networks associated with the general models (see Figs. 5 and 6), simplification is not possible in these cases.
4. Uniqueness of representation

The uniqueness of representation for each of the models presented in Section 3 should be checked before learning the neural functions from the data using the model. The uniqueness property is very important because some estimation methods (e.g., those that solve systems of equations) rely on uniqueness. For other estimation methods (e.g., direct optimization methods) the uniqueness is not required, but it is convenient because it reduces dimensionality. In this section we examine the models described in Section 3 for uniqueness.

4.1. Uniqueness of the additive model

The additive model in (13) is obviously not unique because one can multiply both sides of the equation by an arbitrary nonzero constant and obtain other equivalent models. However, to know the precise uniqueness constraints, assume that we have two sets of functions \( \{h_1, h_2, \ldots, h_k, f\} \) and \( \{h_1^*, h_2^*, \ldots, h_k^*, f^*\} \) such that, \( \forall x_1, \ldots, x_k, \)

\[
f^{-1}[h_1(x_1) + h_2(x_2) + \cdots + h_k(x_k)] = f^{*-1}[h_1^*(x_1) + h_2^*(x_2) + \cdots + h_k^*(x_k)].
\]

This is a functional equation with the following general solution (see Castillo et al., 1998, p. 98):

\[
\begin{align*}
    h_1^*(x) &= ah_1(x) + b_1, \\
    h_2^*(x) &= ah_2(x) + b_2, \\
    \vdots & \vdots \\
    h_k^*(x) &= ah_2(x) + b_k, \\
    f^*(x) &= af(x) + b_1 + b_2 + \cdots + b_k,
\end{align*}
\]

(16)

where \( a \) and \( b_1, b_2, \ldots, b_k \) are arbitrary constants. Thus, to have a unique solution, we must fix the functions \( h_1, h_2, \ldots, h_k \), and \( f \) at a point. Note that in the nonlinear regression context this is equivalent to setting \( E[h_1(x)], \ldots, E[h_k(x)], E[f(y)] \) to a constant.
4.2. Uniqueness of the general model

To analyze the uniqueness of the general model in (14) suppose that we have two sets of parameters \( c_{r_1r_2...r_k} \) and \( c^*_{r_1r_2...r_k} \), \( r_s = 1, 2, \ldots, q_s \) and \( s = 1, 2, \ldots, k \), such that

\[
\sum_{r_1=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} c_{r_1...r_k} \phi_{r_1}(x_1) \ldots \phi_{r_k}(x_k) = \sum_{r_1=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} c^*_{r_1...r_k} \phi_{r_1}(x_1) \ldots \phi_{r_k}(x_k). \tag{17}
\]

Then, we can write (17) as

\[
\sum_{r_1=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} (c_{r_1r_2...r_k} - c^*_{r_1r_2...r_k}) \phi_{r_1}(x_1) \phi_{r_2}(x_2) \ldots \phi_{r_k}(x_k) = 0. \tag{18}
\]

Since the sets of functions \( \Phi_s = \{ \phi_{r_s}(x_s), r_s = 1, \ldots, q_s \}, s = 1, \ldots, k \), are linearly independent, so is the set \( \{ \phi_{r_1}(x_1) \phi_{r_2}(x_2) \ldots \phi_{r_k}(x_k), r_s = 1, \ldots, q_s, s = 1, \ldots, k \} \), then \( c_{r_1,...,r_k} - c^*_{r_1,...,r_k} = 0, \forall r_1, \ldots, r_k \), which implies \( c_{r_1,...,r_k} = c^*_{r_1,...,r_k}, \forall r_1, \ldots, r_k \). This proves that for sets of linearly independent functions, the representation in (14) is unique.

To discuss the uniqueness of the model in (15), we write it as

\[
\sum_{r_0=1}^{q_0} a_{r_0} \phi_{r_0}(y) - \sum_{r_1=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} c_{r_1r_2...r_k} \phi_{r_1}(x_1) \phi_{r_2}(x_2) \ldots \phi_{r_k}(x_k) = 0. \tag{19}
\]

Since the model

\[
\gamma \left( \sum_{r_0=1}^{q_0} a_{r_0} \phi_{r_0}(y) - \sum_{r_1=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} c_{r_1r_2...r_k} \phi_{r_1}(x_1) \phi_{r_2}(x_2) \ldots \phi_{r_k}(x_k) \right) = 0, \tag{20}
\]

is equivalent to model in (19) for any value of \( \gamma \), the general model representation in (15) is not unique. To obtain uniqueness in this case, we add one more constraint such as fixing the right-hand side of (15) to a fixed value.

5. Parametric learning

5.1. Linearly independent families

When dealing with functional networks, there are two types of parametric learning:

- **Exact learning**, which consists of identifying the functions that are the solutions of the functional equation associated with the functional network.
- **Approximate learning**, which consists of estimating the neural functions based on the given data.

Since a data set is always available in regression modeling, approximate learning is always possible.

For the problems we are concerned with in this paper, we propose to approximate the right-hand side of (1) and (2) by a linear combination of functions in a set of linearly independent functions,

\[
\Phi_s = \{ \phi_{r_s}(X_s), r_s = 1, 2, \ldots, q_s \},
\]

where \( q_s \) is the number of elements in \( \Phi_s, s = 1, 2, \ldots, k \). Examples of appropriate families include:

1. **Polynomial family**:
   \[
   \Phi = \{ 1, x, x^2, \ldots, x^q \}. \tag{21}
   \]

2. **Exponential family**:
   \[
   \Phi = \{ e^x, e^{-x}, e^{2x}, e^{-2x}, \ldots, e^{qx}, e^{-qx} \}. \tag{22}
   \]
(3) Fourier series family:

\[ \Phi = \{ 1, \sin x, \cos x, \sin(2x), \cos(2x), \ldots, \sin(qx), \cos(qx) \}. \]  

Using any of the above families, Problem 1 can be represented by

\[ y = \sum_{i=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} c_{r_1r_2\ldots r_k} \phi_{r_1}(x_1) \phi_{r_2}(x_2) \cdots \phi_{r_k}(x_k). \]  

Similarly, Problem 2 can be represented by

\[ f(y) = \sum_{i=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} c_{r_1r_2\ldots r_k} \phi_{r_1}(x_1) \phi_{r_2}(x_2) \cdots \phi_{r_k}(x_k), \]  

where \( f(y) \) can also be approximated by \( f(y) \simeq \sum_{i=0}^{q_0} c_{r_0} \phi_{r_0}(y) \), where \( c_{r_0} \) are unknown parameters and \( \Phi_0 = \{ \phi_{r_0}(y), r_0 = 1, 2, \ldots, q_0 \} \) is a set of linearly independent functions. In this case, since \( \Phi = \{ \phi_{r_1}(x_1) \phi_{r_2}(x_2) \cdots \phi_{r_k}(x_k), r_s = 1, 2, \ldots, q_s; s = 1, 2, \ldots, k \} \cup \{ \phi_{r_0}(y); r_0 = 1, 2, \ldots, q_0 \} \) needs to be a set of linearly independent functions, which is necessary for obtaining a unique solution, then the constant function can be included in only one of the two sets of functions.

We illustrate the methodology of this paper using the polynomial family. Other families can be implemented in a similar way. For example, setting the number of explanatory variables \( k = 2 \) and using the polynomial families with the same number of functions in all of them, that is, \( q_1 = \cdots = q_k = q + 1 \), where \( q \) is the degree of the polynomial, the right-hand side of (24) and (25) can then be written as

\[ h(x_1, x_2) \simeq h_1(x_1) + h_2(x_2) + h_3(x_1, x_2), \]  

where

\[
\begin{align*}
    c_{00} \cdot 1 & 1 \\
    +c_{01} \cdot x_1 & +c_{02} \cdot x_2 \\
    +c_{10} \cdot x_1 & +c_{11} \cdot x_1 \cdot x_2 \\
    +c_{20} \cdot x_1 & +c_{21} \cdot x_1 \cdot x_2 \\
    +c_{11} & +c_{21} \\
    +c_{22} & \\
    +c_{20} & +c_{22} \\
    \vdots & +c_{20} \cdot x_1 \\
    +c_{22} & +c_{20} \cdot x_1 \\
    +c_{20} & +c_{22} \\
\end{align*}
\]  

\[ = h_2(x_2) \\
\]  

\[
\begin{align*}
    c_{00} \cdot 1 & 1 \\
    +c_{01} \cdot x_1 & +c_{02} \cdot x_2 \\
    +c_{10} \cdot x_1 & +c_{11} \cdot x_1 \cdot x_2 \\
    +c_{20} \cdot x_1 & +c_{21} \cdot x_1 \cdot x_2 \\
    +c_{11} & +c_{21} \\
    +c_{22} & +c_{21} \\
\end{align*}
\]  

\[ = h_3(x_1, x_2), \]  

Note that, without lost of generality, the constant term can be included only in \( h_1(x_1) \).

Let \( D = \{ y_i, x_{i1}, x_{i2}, \ldots, x_{ik}; i = 1, \ldots, n \} \) denote the available data set, which consists of \( n \) observations of a response and \( k \) predictor variables. It is assumed that in the range of the observations studied, the nonlinear model in (25) provides an acceptable approximation to the true relation between the response and the predictor variables. Then, for the \( i \)th observation in \( D \), we have for the general case

\[ \sum_{r_0=1}^{q_0} a_{r_0} \phi_{r_0}(y_i) = \sum_{r_1=1}^{q_1} \cdots \sum_{r_k=1}^{q_k} c_{r_1r_2\ldots r_k} \phi_{r_1}(x_{i1}) \phi_{r_2}(x_{i2}) \cdots \phi_{r_k}(x_{ik}) + \epsilon_i, \]  

where \( \epsilon_i \) are random errors and \( f(y) \) is approximated by a linear combination of linearly independent functions, \( \Phi_0 = \{ \phi_{r_0}; r_0 = 1, \ldots, q_0 \}, s = 0, 1, \ldots, k \). If these families are known, we only need to learn the coefficients \( a_{r_0} \) and \( c_{r_1r_2\ldots r_k} \).

The model in (28) can be expressed in a matrix form as

\[ Y \mathbf{z} = W \mathbf{\beta} + \epsilon, \]  

where using the polynomial families, \( Y \) and \( W \) contain \( q_0 \) and \( p = \prod_k (q_k + 1) \) columns, respectively, each of which is an appropriate polynomial function of the response \( Y \) and the explanatory variables \( X_1, X_2, \ldots, X_k \), respectively, which depend on the form of the model. For example, for \( k = 2 \), and \( q_0 = q_1 = q_2 = 2 \), the rows of the matrices \( Y \) and
The problem of finding $W$ obtained from the model in (29) are of the form $\{y, y^2\}$ and $\{1, x_1, x_2, x_1^2, x_2^2, x_1 x_2, x_1^2 x_2, x_2^2 x_1\}$, respectively. Note that, to get uniqueness, we have not included the constant function in the set of functions associated with $Y$.

Note that, (29) is not a multivariate multiple linear regression model, despite the fact that $Y$ contains more than one variable. It would be a multivariate model if $X$ is the identity matrix, which is not the case here.

For parametric learning, i.e., to estimate the two sets of parameters $\alpha$ and $\beta$ in (29), we consider two criteria: the CLS and the MCC methods.

### 5.2. CLS method

As indicated in Section 4.2, to have uniqueness we need to add one constraint, for example

$$w_0^T \beta = c_1,$$  \hspace{1cm} (30)

where for $k = 2$ and $q_1 = q_2 = 2$, $w_0^T = \{1, x_1, x_2, x_1^2, x_2^2, x_1 x_2, x_1^2 x_2, x_2^2 x_1\}$ and $(x_1, x_2)$ is a given point different from $(0, 0)$ and $c_1$ is an arbitrary constant different from 0. Therefore, using the Lagrange multiplier method, we minimize:

$$L = (Y \alpha - W \beta)^T (Y \alpha - W \beta) - 2\lambda (w_0^T \beta - c_1).$$  \hspace{1cm} (31)

Calculating the derivatives, we have

$$\frac{\partial L}{\partial \alpha} = 2Y^T Y \alpha - 2Y^T W \beta = 0,$$

$$\frac{\partial L}{\partial \beta} = 2W^T W \beta - 2W^T Y \alpha - 2\lambda w_0 = 0,$$

$$\frac{\partial L}{\partial \lambda} = -2w_0^T \beta + 2c_1 = 0.$$

The solution of this system of linear equations leads to the parameter estimators:

$$\hat{\beta} = \frac{c_1 [W^T (I - P_Y) W]^{-1} w_0}{w_0^T [W^T (I - P_Y) W]^{-1} w_0},$$

$$\hat{\alpha} = (Y^T Y)^{-1} Y^T W \hat{\beta},$$

$$\hat{\lambda} = \frac{c_1}{w_0^T [W^T (I - P_Y) W]^{-1} w_0},$$

where $P_Y = Y (Y^T Y)^{-1} Y^T$.

### 5.3. MCC method

Another estimator of $\alpha$ and $\beta$ is obtained by maximizing $\rho(Y \alpha, W \beta)$, the correlation coefficient between $Y \alpha$ and $W \beta$. The correlation is not affected by changes of location and scale, that is, $\rho(Y \alpha, W \beta) = \rho(\kappa_1 Y \alpha + \delta_1, \kappa_2 W \beta + \delta_2)$, $\kappa_1, \kappa_2, \delta_1, \delta_2 \in \mathbb{R}$. Therefore, the correlation coefficient for model (29) is the same as the correlation coefficient without the constant term. For simplicity, let us just write (29) without the constant term as

$$Y \alpha = W \beta + \epsilon.$$  \hspace{1cm} (34)

The problem of finding $\alpha$ and $\beta$ to maximize the correlation between $Y \alpha$ and $W \beta$ can be recognized as the canonical correlation problem. The solution of which can be obtained as follows:

1. Compute the covariance matrix of $Z = (Y : W)$, which is given by

$$S = \begin{pmatrix} S_{yy} & S_{yw} \\ S_{wy} & S_{ww} \end{pmatrix},$$  \hspace{1cm} (35)

where $S_{yy} = \text{Var}(Y)$, $S_{yw} = \text{Cov}(Y, W)$, $S_{wy} = \text{Cov}(W, Y)$ and $S_{ww} = \text{Var}(W)$. 


(2) Compute the matrices \( S_{yy}^{-1/2} \) and \( S_{ww}^{-1/2} \), where \( S_{yy}^{-1/2} \) is the inverse of the symmetric square root of \( S_{yy} \). Consequently, \( S_{yy} = S_{yy}^{1/2} S_{yy}^{1/2} \) and \((S_{yy}^{1/2})^T = S_{yy}^{1/2} \).

(3) Compute the eigenvalues \( \lambda_1 \geq \cdots \geq \lambda_p \) of \( M_y = S_{yy}^{-1/2} S_{wy} S_{wy}^{-1} S_{yw} S_{ww}^{-1/2} \), and the corresponding eigenvectors \( v_1, \ldots, v_p \).

(4) The estimate of \( \alpha \) is obtained by
\[
\hat{\alpha} = S_{yy}^{-1/2} v_1. \tag{36}
\]

(5) Compute \( d = S_{ww}^{-1/2} S_{wy} S_{yy}^{-1/2} v_1 \), normalize \( d \), and obtain the estimate of \( \beta \) as
\[
\hat{\beta} = S_{ww}^{-1/2} d. \tag{37}
\]

(6) The MCC is \( \sqrt{\lambda_1} \).

Note that \( \beta \) can also be computed as follows:

(1) Compute the eigenvalues \( \mu_1 \geq \cdots \geq \mu_p \) of \( M_w = S_{ww}^{-1/2} S_{wy} S_{yy}^{-1} S_{yw} S_{ww}^{-1/2} \), and the corresponding eigenvectors \( u_1, \ldots, u_p \).

(2) The estimate of \( \beta \) is obtained by \( \hat{\beta} = S_{ww}^{-1/2} u_1 \).

Note that the matrices \( M_y \) and \( M_w \) have the same eigenvalues but different eigenvectors.

One advantage of the solutions in (36) and (37) is that they are given in closed-forms. On the other hand, the solutions in (36) and (37) are not unique. But unique solutions, can be obtained by imposing certain constraints as described in the following theorem.

**Theorem 2** (Unconstrained and constrained problems). Let \( u = Y \alpha \) and \( v = W \beta + \gamma \). The slope and intercept of the least squares line when \( u \) is regressed on \( v \) are given by
\[
b(\alpha, \beta, \gamma) = \frac{\sum_{i=1}^{n} (u_i - \bar{u})(v_i - \bar{v})}{\sum_{i=1}^{n} (v_i - \bar{v})^2}, \tag{38}
\]
\[
a(\alpha, \beta, \gamma) = \bar{u} - \bar{v}. \tag{39}
\]

Let
\[
\rho(\alpha, \beta, \gamma) = \frac{\sum_{i=1}^{n} (u_i - \bar{u})(v_i - \bar{v})}{\sqrt{\sum_{i=1}^{n} (u_i - \bar{u})^2 \sum_{i=1}^{n} (v_i - \bar{v})^2}}, \tag{40}
\]
be the correlation coefficient between \( u \) and \( v \). Also, let
\[
\hat{k}_1 = \frac{1}{||\hat{z}||}, \quad \hat{k}_2 = \frac{b(\hat{\alpha}, \hat{\beta}, \hat{\gamma})}{||\hat{z}||}, \quad \hat{\gamma}_1 = \hat{Y} \hat{k}_1 \hat{z} - \hat{W} \hat{k}_2 \hat{\beta}. \tag{41}
\]
If \( \{\hat{z}, \hat{\beta}, \hat{\gamma}\} \) maximizes \( \rho(\alpha, \beta, \gamma) \) without constraints, then \( \{\hat{k}_1 \hat{z}_1, \hat{k}_2 \hat{\beta}_1, \hat{\gamma}_1\} \), maximizes \( \rho(\alpha, \beta, \gamma) \) subject to
\[
b(\alpha, \beta, \gamma) = 1, \tag{42}
\]
\[
a(\alpha, \beta, \gamma) = 0, \tag{43}
\]
\[
||\alpha|| = 1, \tag{44}
\]
and is unique among all solutions of the form \( \{k_1 \hat{z}, k_2 \hat{\beta}, \gamma_1\} \).

**Proof.** It is obvious that if \( \{\hat{z}, \hat{\beta}, \hat{\gamma}\} \) maximizes (40), then \( \{k_1 \hat{z}, k_2 \hat{\beta}, \gamma_1\} \) also maximizes (40) since \( \rho(\alpha, \beta, \gamma) \) is independent of location and scale transformations of \( \alpha \) and \( \beta \), and does not depend on \( \gamma \). From (38), (39) and (41)
Thus, the set \( \{ \hat{k}_1 \hat{x}, \hat{k}_2 \hat{\beta}, \hat{\gamma}_1 \} \) gives the same value of \( \rho(\mathbf{x}, \mathbf{\beta}, \gamma) \) as the unconstrained maximum and satisfies the constraints. Consequently, it is the constrained maximum. Since system (45)–(47) gives a unique solution for \( \hat{k}_1, \hat{k}_2, \) and \( \hat{\gamma}_1, \) we get the uniqueness part of the theorem. \( \square \)

The practical conclusion of this theorem is that we can find a solution for \( f(y) \) and \( h(x_1, x_2, \ldots, x_k) \) such that its correlation is maximized and its regression line is the bisectrix.

6. Model selection

As we have seen in Section 5, to approximate the functions specified in the two models of Section 3, we use a set of linearly independent functions, \( \Phi_s = \{ \phi_{rs}(X_s) \}, r_s = 1, 2, \ldots, q_s \}, \) such as those in (21)–(23). When \( q_s \) is large, the model depends on a large set of unknown parameters (coefficients) to be estimated from the data. This raises the issue of model selection, that is, which of these parameters should be retained in or removed from the model?

There are several criteria and methods for dealing with the model selection problem and the reader can refer to any of the standard regression analysis textbooks. We have implemented several criteria such as \( \text{MSE} \), \( R^2 \), adjusted \( R^2 \) (\( R_a^2 \)), Mallows \( C_p \), minimum description length (\( \text{MDL} \)), and several versions of Akaike information criterion (\( \text{AIC} \)). (Note that the \( \text{MDL} \) and \( \text{AIC} \) assume normally distributed errors). We have checked their behavior with all the simulated examples presented in Section 7. We have observed that all criteria, except \( R_a^2 \), tend to select more complex models than the true model in the functional networks context. Moreover, \( R_a^2 \) has the additional advantage that it is distributional free. For our problems

\[
R_a^2 = 1 - \frac{\sum_{i=1}^{n} e_i^2/(n - p)}{\sum_{i=1}^{n} (f(y_i) - \bar{f}(y_i))^2/(n - 1)}
\]

where \( p \) is the number of parameters in the model and \( \bar{f}(y) = (1/n)\sum_{i=1}^{n} f(y_i) \).

In the simulated and real-life data (with few explanatory variables), discussed in Sections 7 and 8, we use the exhaustive variable selection method, i.e., all possible models are investigated, in order to gain understanding of the performance of our method. For real-life data with moderate to high number of explanatory variables, where exhaustive search is practically computationally infeasible, we have implemented a genetic algorithm (see Michalewicz, 1999 for details) for this purpose.

7. Assessing performance using simulation

To assess the performance of the proposed method first we use several sets of simulated data to facilitate the analysis, since the true model is known. In all the examples, we compare the two proposed techniques for the parametric learning in functional networks: the CLS and the MCC methods. For both methods, a polynomial family \( \Phi = \{1, t, t^2, \ldots, t^q\} \) (with \( q = 2, 3 \}) is used for approximating the neural functions, since we expect that monotone functions approximate well both the transformation of the response variable (if needed) and the possible nonlinear contribution of the explanatory variables to the response variable.

We point out here that we have examined the distribution of the estimated coefficients \( \mathbf{\hat{x}} \) and \( \mathbf{\hat{\beta}} \) in (29) over a large set of simulated data. Since some of the distributions estimates are slightly skewed, normality cannot be assumed. Therefore, we use Bootstrap confidence intervals. Point estimation is obtained by averaging the coefficients of the models selected as best in the exhaustive search process based on 500 Monte Carlo samples of size 200 from the theoretical model. The point estimators are ordered to compute the 2.5-th and 97.5-th quantiles to obtain the 95\% Bootstrap confidence
interval, where (0, 0) confidence intervals indicate that the corresponding coefficient has almost never been included in the best model. More specifically, the confidence intervals are obtained as follows:

1. We generate a data set at random from the true model.
2. We search the best model for this data set.
3. We repeat 1 and 2, \( m \) times.
4. We average the \( m \) estimators and we order them to compute the 2.5-th and 97.5-th quantiles to obtain the 95% Bootstrap confidence interval.

With the aim of illustrating the proposed methods and to show that they can learn even simple functions, we have simulated sets of data using the following models:

1. A linear model of the form
   \[ Y = X_1 + X_2 + \varepsilon, \]  
   where \( X_1 \) and \( X_2 \) are independent \( U[0, 10] \), and \( \varepsilon \) is \( N[0, 1] \) and independent of \( X_1 \) and \( X_2 \). Furthermore, in order to study if the learning process tends to add more variables than necessary, we have consider the model
   \[ Y = X_1 + X_2 + 0 \times X_3 + 0 \times X_4 + 0 \times X_5 + \varepsilon, \]  
   where \( X_i, i = 1, \ldots, 5 \), are independent \( U[0, 10] \), and \( \varepsilon \) is \( N[0, 1] \), independent of \( X_i, i = 1, \ldots, 5 \).

2. A polynomial model of the form
   \[ Y^3 = X_1 + X_1^2 + X_2^2 + X_2^3 + \varepsilon, \]  
   where \( X_1 \) and \( X_2 \) are independent \( U[0, 5] \), and \( \varepsilon \) is \( N[0, 0.1] \) and independent of \( X_1 \) and \( X_2 \).

3. An exponential model of the form
   \[ \ln Y = X_1 + X_2 + \varepsilon, \]  
   where \( X_1 \) and \( X_2 \) are independent \( U[0, 1] \), and \( \varepsilon \) is \( N[0, 0.1] \) and independent of \( X_1 \) and \( X_2 \).

4. A model with interactions of the form
   \[ Y = X_1 + X_2 + X_1X_2 + \varepsilon, \]  
   where \( X_1 \) and \( X_2 \) are independent \( U[0, 10] \), and \( \varepsilon \) is \( N[0, 1] \) and independent of \( X_1 \) and \( X_2 \).

We have used the additive model \( f(Y) = h_1(X_1) + h_2(X_2) \), for the first three cases, and the general model \( f(Y) = h_1(X_1) + h_2(X_2) + h_{12}(X_1, X_2) \), for the fourth case, where all the functions have been approximated by second degree polynomials for the linear and interactions model and third degree polynomials for the rest of the models. We have also used the general model in the linear case just to see if interactions are included when they are not in the true model. We have obtained good results, but not all of them are included here to save some space.

Tables 1–5 show the point and confidence interval estimations of the coefficients \( \alpha \) and \( \beta \) in (29). Note that in all the cases we obtain a very good approximation of the true model. We also observe that the MCC criterion provides unbiased estimators in all of the cases. However, when CLS criterion is used, \( \beta \) is always estimated with some bias in the simplest models (49), (50) and (53) and unbiased estimators are obtained for the most complex models (51) and (52). When the simulation error is reduced, bias is also reduced and unbiased estimators are obtained with small errors. (We have tried models (49), (50) with smaller error \( \varepsilon \sim N(0, 0.1) \) and then the estimators are unbiased.) Furthermore, note that narrower confidence intervals are obtained for the simplest models (49), (50) and (53) when using MCC, but larger for the most complex models (51) and (52).

Figs. 7 and 8 show the distributions of the estimated coefficients when using the CLS criterion. Figs. 9 and 10 show the distribution of the estimated coefficients when using the MCC criterion. Note that both methods provide for this model symmetric histograms.

For space saving purpose, we do not include the histograms of the coefficient estimators for the rest of the simulated models. The results are similar to those for the linear case. We wish to note here, however, that when the coefficient
Table 1
Point and confidence interval estimation of the coefficients of the linear model in (49) when using CLS criterion (left) and MCC criterion (right)

<table>
<thead>
<tr>
<th>Term</th>
<th>CLS</th>
<th>MCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{f}(y)$</td>
<td>$y$</td>
<td>1</td>
</tr>
<tr>
<td>$y^2$</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>Constant</td>
<td>1</td>
<td>1.8970</td>
</tr>
<tr>
<td>$\hat{h}_1(x_1)$</td>
<td>$x_1$</td>
<td>0.8321</td>
</tr>
<tr>
<td>$x_1^2$</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>$\hat{h}_2(x_2)$</td>
<td>$x_2$</td>
<td>0.8547</td>
</tr>
<tr>
<td>$x_2^2$</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
</tbody>
</table>

Table 2
Point and confidence interval estimation of the coefficients of the linear model in (50) when using CLS criterion (left) and MCC criterion (right)

<table>
<thead>
<tr>
<th>Term</th>
<th>CLS</th>
<th>MCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{f}(y)$</td>
<td>$y$</td>
<td>1</td>
</tr>
<tr>
<td>$y^2$</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>Constant</td>
<td>1</td>
<td>0.0070</td>
</tr>
<tr>
<td>$\hat{h}_1(x_1)$</td>
<td>$x_1$</td>
<td>0.9982</td>
</tr>
<tr>
<td>$x_1^2$</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>$\hat{h}_2(x_2)$</td>
<td>$x_2$</td>
<td>1.0010</td>
</tr>
<tr>
<td>$x_2^2$</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
</tbody>
</table>

estimator of a term not included in the model is different from zero, the distribution is highly asymmetric with the mode placed in this value and small probability is given for the rest of the values.
Table 3
Point and confidence interval estimation of the coefficients of the polynomial model in (51) when using CLS criterion (left) and MCC criterion (right)

<table>
<thead>
<tr>
<th>Term</th>
<th>CLS</th>
<th>Bootstrap C.I.</th>
<th>MCC</th>
<th>Bootstrap C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(\hat{f}(y))</td>
<td>(y)</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(y^2)</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(y^3)</td>
<td>1</td>
<td>(0.9984, 1.0017)</td>
</tr>
<tr>
<td></td>
<td>(\hat{h}_1(x_1))</td>
<td>(x_1)</td>
<td>0.9993</td>
<td>(0.9576, 1.0404)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(x_2)</td>
<td>1</td>
<td>(0.9922, 1.0077)</td>
</tr>
<tr>
<td></td>
<td>(\hat{h}_2(x_2))</td>
<td>(x_2)</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(x_2^2)</td>
<td>1</td>
<td>(0.9966, 1.0033)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(x_2^3)</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
</tbody>
</table>

Table 4
Point and confidence interval estimation of the coefficients of the exponential model in (52) when using CLS criterion (left) and MCC criterion (right)

<table>
<thead>
<tr>
<th>Term</th>
<th>CLS</th>
<th>Bootstrap C.I.</th>
<th>MCC</th>
<th>Bootstrap C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(\hat{f}(y))</td>
<td>(y)</td>
<td>0.6634</td>
<td>(0.6272, 0.6982)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(y^2)</td>
<td>-0.0493</td>
<td>(-0.0550, -0.0437)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(y^3)</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td></td>
<td>(\hat{h}_1(x_1))</td>
<td>(x_1)</td>
<td>0.4416</td>
<td>(0.3770, 0.5071)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(x_2)</td>
<td>1</td>
<td>(0.9473, 1.0547)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(x_2^2)</td>
<td>-0.0028</td>
<td>(0, 0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(x_2^3)</td>
<td>0.0026</td>
<td>(0, 0)</td>
</tr>
<tr>
<td></td>
<td>(\hat{h}_2(x_2))</td>
<td>(x_2)</td>
<td>1.0290</td>
<td>(0.9999, 1.0606)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(x_2^2)</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(x_2^3)</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
</tbody>
</table>
Table 5
Point and confidence interval estimation of the coefficients of the interactions model in (53) when using CLS criterion (left) and MCC criterion (right)

<table>
<thead>
<tr>
<th>Term</th>
<th>Mean</th>
<th>Bootstrap C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CLS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \hat{f}(y) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( y )</td>
<td>1</td>
<td>(0.9487, 1.0565)</td>
</tr>
<tr>
<td>( y^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>Constant</td>
<td>1.5120</td>
<td>(1.0636, 1.9120)</td>
</tr>
<tr>
<td>( \hat{h}_1(x_1) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_1 )</td>
<td>0.7884</td>
<td>(0.6799, 0.9222)</td>
</tr>
<tr>
<td>( x_1^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( \hat{h}_2(x_2) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0.8049</td>
<td>(0.6901, 0.9407)</td>
</tr>
<tr>
<td>( x_2^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( \hat{h}_{12}(x_1, x_2) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_1 \cdot x_2 )</td>
<td>1.0270</td>
<td>(0.97996, 1.0752)</td>
</tr>
<tr>
<td>( x_1 \cdot x_2^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( x_1^2 \cdot x_2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( x_1^2 \cdot x_2^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td><strong>MCC</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \hat{f}(y) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( y )</td>
<td>1</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>( y^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>Constant</td>
<td>-0.0003</td>
<td>(−0.6202, 0.6494)</td>
</tr>
<tr>
<td>( \hat{h}_1(x_1) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_1 )</td>
<td>1</td>
<td>(0.8875, 1.1006)</td>
</tr>
<tr>
<td>( x_1^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( \hat{h}_2(x_2) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0.9994</td>
<td>(0.9032, 1.1019)</td>
</tr>
<tr>
<td>( x_2^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( \hat{h}_{12}(x_1, x_2) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_1 \cdot x_2 )</td>
<td>1</td>
<td>(0.9823, 1.0180)</td>
</tr>
<tr>
<td>( x_1 \cdot x_2^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( x_1^2 \cdot x_2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( x_1^2 \cdot x_2^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
</tbody>
</table>

Fig. 7. Histograms for the estimated coefficients of the linear model with two variables: \( y \)-coefficient \( \hat{z}_1 \) in \( \hat{f}(y) \) and constant term \( \hat{f}_0 \) when using CLS criterion.

From the above results, one can conclude the following:

1. For the linear case, both the CLS and MCC methods select the true model even when including more variables or interactions than those in the theoretical model. Then, the proposed method tends to include neither more variables nor interactions than necessary. Furthermore, MCC provides unbiased estimators even when these models are simulated with large errors.

2. For the polynomial case, both the CLS and MCC methods select the true model and provide unbiased estimators for \( \beta \) with smaller variances than those given by the CLS criterion. Moreover, they produce very good fit of the estimated polynomial for the explanatory and response variables. This is shown in Fig. 11.
Fig. 8. Histograms for the estimated coefficients of the linear model with two variables: $x_1$-coefficient $\beta_1$ in $\hat{h}_1(x_1)$ and $x_2$-coefficient $\beta_3$ in $\hat{h}_2(x_2)$ when using CLS criterion.

Fig. 9. Histograms for the estimated coefficients of the linear model with two variables: $y$-coefficient $\alpha_1$ in $\hat{f}(y)$ and constant term $\beta_0$ when using MCC criterion.

Fig. 10. Histograms for the estimated coefficients of the linear model with two variables: $x_1$-coefficient $\beta_1$ in $\hat{h}_1(x_1)$ and $x_2$-coefficient $\beta_3$ in $\hat{h}_2(x_2)$ when using MCC criterion.
Fig. 11. Scatter plot for the estimated polynomial function in the explanatory variables and the true relationship versus \( Y \), when using CLS criterion (left) and MCC criterion (right).

Fig. 12. Exponential model: Scatter plot for the estimated function \( \hat{f}(y) \) for the response variable and ln \( Y \) versus \( Y \), when using (left) CLS criterion and (right) MCC criterion.

(3) For the exponential case, the true model for the right-hand side of the equation and a very good approximation for the left-hand side is obtained with both CLS and MCC criteria. The left-hand side approximation of the true transformation ln \( Y \) is shown in Fig. 12. Furthermore, both criteria provide unbiased estimators for the \( x \)-coefficients.

(4) For the interactions case, the true model is selected with both CLS and MCC criteria. Conclusions about the bias and variance of \( \hat{\beta} \) obtained with CLS criterion are also valid for this model.

8. Assessing performance using real data

In this section we use the proposed methods to analyze four sets of real data, where the relationship between the predictor and the explanatory variables is assumed to be nonlinear in the variables: (1) the temperature data set, found in Peixoto (1990), who considers the use of polynomial regression models, (2) the tree data set, found in Ryan and Joiner (1994) and extensively analyzed by Atkinson (1985), who uses classical parametric techniques for determining transformations of the response and the explanatory variables, (3) the diabetes data set, which is a result of a study by Sockett et al. (1987) and used by Hastie and Tibshirani (1990) for illustrating the application of additive models, and (4) the Boston Housing data set of Harrison and Rubinfeld (1978), which has been used in Breiman and Friedman (1985), Breiman et al. (1993) and Pace (1993) (among others) to illustrate the ACE algorithm, regression trees and kernel estimators, respectively.
We have chosen these data sets to facilitate the comparison of our method with other parametric and nonparametric regression methods. For illustrative purposes, we use the following models:

1. The additive model \( f(Y) = h_1(X_1) + \cdots + h_k(X_k) \) with \( k = 2 \) for the temperature data and the diabetes data sets, and \( k = 13 \) for the Boston Housing data set, and

2. The additive model with \( k = 2 \) and the general model \( Y = h_1(X_1) + h_2(X_2) + h_{12}(X_1, X_2) \) for the tree data set.

We use the learning methods proposed in this paper, using a third degree polynomial for approximating the functions. We use the exhaustive search method to select the best model in all the cases, except for the Boston Housing data set for which we implement a genetic algorithm as described in Michalewicz (1999) because of its relatively high dimension. We rescale some of the data values to avoid overflows.

For the three first examples, first we show the best model selected using both CLS and MCC criteria. We also present scatter plots for the relationship between \( \hat{f}(y) \) versus the adjusted function in the explanatory variables and residuals versus predicted values. Point estimators are the coefficients of the best model with the complete data set. Confidence interval estimations of the coefficients are based on 500-times 10-fold bootstrap cross-validation.

Finally, we note the differences between our technique and that of the corresponding references mentioned above. For the last two examples, we present the results obtained with CLS criterion and we show the results that allow us to compare our method with those considered in the corresponding references mentioned above.

8.1. Temperature data

Data description: The temperature data set gives the average January minimum temperature (in Fahrenheit) as a function of the latitude and longitude of 56 US cities. For each year from 1931 to 1960, the daily minimum temperatures in January were added together and the sum is then divided by 31 (the number of days in January). Then, the averages for each year were averaged over the 30 years. The variables are:

- \( y \) average January minimum temperature (degrees Fahrenheit) from 1931 to 1960,
- \( x_1 \) latitude in degrees north of the equator,
- \( x_2 \) longitude in degrees west of the prime meridian.

The author reports a study in which a linear relationship is assumed between temperature and latitude; then, after adjusting for latitude, a cubic polynomial in longitude accurately predicts the temperature.

Results using functional networks: The selected models using the additive model are:

1. The CLS criterion:
   \[
   y = -7.2151 - 2.5613x_1 + 29.4335x_2 - 33.4876x_2^2 + 12.4537x_2^3. 
   \]  \hspace*{1cm} (54)

2. The MCC criterion:
   \[
   y = -7.9070 - 2.3577x_1 + 31.2138x_2 - 35.1538x_2^2 + 12.9559x_2^3. 
   \]  \hspace*{1cm} (55)

Scatter plots of \( \hat{f}(y) \) versus \( \hat{h}_1(x_1) + \hat{h}_2(x_2) \) and residuals versus predicted values are shown in Figs. 13 and 14. One can see that the two sets of scatter plots are similar. Table 6 shows the point and confidence interval estimation of the coefficients. Note that all the terms included in the selected model are significatively different from 0 and the rest of the terms have never been included in the model. Thus, we have obtained the same results as Peixoto (1990), who needs to examine several residual plots to select the model as compared to our method which is an automatic search method that considers the transformation of all variables simultaneously.

8.2. Tree data

Data description: The tree data set consists of a set of measurements on the volume, diameter, and height of 31 black cherries trees. The purpose of the measurements on the felled trees was to provide a model for predicting the
volume of timber in unfelled trees and hence to predict the value of timber in an area of forest. The variables are:

\[ y \] volume of the tree in cubic feet,

\[ x_1 \] diameter in inches at 4' 6" above the ground,

\[ x_2 \] height of the tree in feet.

In a preliminary analysis by Atkinson (1985), several models are proposed:

1. A linear model including higher-order polynomial terms in both explanatory variables.
2. Transformation of the response following the hint provided by the increase of variance with volume.
3. The volume of a cone which is a good first approximation to the shape of a tree trunk.

Index plots of the deletion residuals and Cook’s distances show departures from the first-order linear model for the smallest (observations 1–3) and largest trees (observation 31). Atkinson (1985, p. 125), examines the following models for this data set:

1. Model 1: \( \sqrt[3]{y} = x_1 + x_2 \), suggested by the obtained value of \( \hat{\lambda} = 0.3066 \) in the Box and Cox power transformation.
2. Model 2: \( \log y = x_1 + x_2^2 + x_2 \).
3. Model 3: \( y = x_1^2 x_2 \), the volume of a cone.
4. Model 4: \( \log y = \log x_1 + \log x_2 \).

Of the four models, Models 1 and 3 are the most satisfactory.
Table 6

Temperature data: point and confidence interval estimation of the coefficients when using CLS criterion (left) and MCC criterion (right)

<table>
<thead>
<tr>
<th>Term</th>
<th>Mean</th>
<th>Bootstrap C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLS</td>
<td>f(y)</td>
<td></td>
</tr>
<tr>
<td>y</td>
<td>0.9998</td>
<td>(0.9779, 1.0260)</td>
</tr>
<tr>
<td>y^2</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>y^3</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>Constant</td>
<td>-7.1630</td>
<td>(-8.7440, -5.7480)</td>
</tr>
<tr>
<td>h_1(x_1)</td>
<td>x_1</td>
<td>(-2.5750, -2.5480)</td>
</tr>
<tr>
<td>x_1^2</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>x_1^3</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>h_2(x_2)</td>
<td>x_2</td>
<td>(26.520, 34.400)</td>
</tr>
<tr>
<td>x_2^2</td>
<td>-33.320</td>
<td>(-38.980, -28.310)</td>
</tr>
<tr>
<td>x_2^3</td>
<td>12.400</td>
<td>(10.550, 14.420)</td>
</tr>
<tr>
<td>MCC</td>
<td>f(y)</td>
<td></td>
</tr>
<tr>
<td>y</td>
<td>1</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>y^2</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>y^3</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>Constant</td>
<td>-7.842</td>
<td>(-9.2710, -6.5500)</td>
</tr>
<tr>
<td>h_1(x_1)</td>
<td>x_1</td>
<td>(-2.4390, -2.3020)</td>
</tr>
<tr>
<td>x_1^2</td>
<td>0.0297</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>x_1^3</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>h_2(x_2)</td>
<td>x_2</td>
<td>(27.150, 35.420)</td>
</tr>
<tr>
<td>x_2^2</td>
<td>-34.980</td>
<td>(-39.800, -30.780)</td>
</tr>
<tr>
<td>x_2^3</td>
<td>11.290</td>
<td>(11.380, 14.600)</td>
</tr>
</tbody>
</table>

Results using functional networks: Selected models using the additive model are:

1. **The CLS criterion:**

   \[ 0.5440 \ y - 0.7343 \ y^2 + 0.4447 \ y^3 + 0.0971 = x_1 + 0.0889 \ x_2. \]  

2. **The MCC criterion:**

   \[ 0.5556 \ y - 0.7443 \ y^2 + 0.4490 \ y^3 + 0.1012 = x_1 + 0.0976 \ x_2. \]
Table 7
Tree data: point and confidence interval estimation of the coefficients when using (left) CLS criterion and (right) MCC criterion for the additive model

<table>
<thead>
<tr>
<th>Term</th>
<th>Mean</th>
<th>Bootstrap C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CLS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \hat{f}(y) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( y )</td>
<td>0.5432</td>
<td>(0.5086, 0.5752)</td>
</tr>
<tr>
<td>( y^2 )</td>
<td>-0.7330</td>
<td>(-0.8319, -0.6420)</td>
</tr>
<tr>
<td>( y^3 )</td>
<td>0.4440</td>
<td>(0.3739, 0.5186)</td>
</tr>
<tr>
<td>Constant</td>
<td>-0.0972</td>
<td>(-0.1073, -0.0882)</td>
</tr>
<tr>
<td>( h_1(x_1) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_1 )</td>
<td>1</td>
<td>(0.9941, 1.0060)</td>
</tr>
<tr>
<td>( x_1^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( x_1^3 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( h_2(x_2) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0.0887</td>
<td>(0.0740, 0.1029)</td>
</tr>
<tr>
<td>( x_2^2 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( x_2^3 )</td>
<td>0</td>
<td>(0, 0)</td>
</tr>
<tr>
<td><strong>MCC</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \hat{f}(y) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( y )</td>
<td>0.5577</td>
<td>(0.5246, 0.6007)</td>
</tr>
<tr>
<td>( y^2 )</td>
<td>-0.7438</td>
<td>(-0.7531, -0.7299)</td>
</tr>
<tr>
<td>( y^3 )</td>
<td>0.4472</td>
<td>(0.4101, 0.4746)</td>
</tr>
<tr>
<td>Constant</td>
<td>-0.0840</td>
<td>(-0.1242, -0.0552)</td>
</tr>
<tr>
<td>( h_1(x_1) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_1 )</td>
<td>1.0030</td>
<td>(0.8704, 1.1830)</td>
</tr>
<tr>
<td>( x_1^2 )</td>
<td>0.0745</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>( x_1^3 )</td>
<td>-0.3185</td>
<td>(-2.4960, 0)</td>
</tr>
<tr>
<td>( h_2(x_2) )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0.0244</td>
<td>(0, 0.1106)</td>
</tr>
<tr>
<td>( x_2^2 )</td>
<td>0.0984</td>
<td>(0, 0.3333)</td>
</tr>
<tr>
<td>( x_2^3 )</td>
<td>-0.0430</td>
<td>(-0.2649, 0.0718)</td>
</tr>
</tbody>
</table>

Fitting the model \( \hat{f}(y) = \beta_0 \ y^{\beta_1} \) reveals that left-hand sides of Eqs. (56) and (57) are an approximation of the Box and Cox power transformation proposed by Atkinson (1985). The obtained power fits are

- **CLS criterion:** \( \hat{f}(y) = 0.3030 \ y^{0.3226} \), \hspace{1cm} (58)
- **MCC criterion:** \( \hat{f}(y) = 0.3129 \ y^{0.3215} \). \hspace{1cm} (59)

Fig. 15 shows the relationship of \( \hat{f}(y) \) and the power fit versus \( y \).

Scatter plots for the relationship between \( \hat{f}(y) \) versus \( \hat{h}_1(x_1) + \hat{h}_2(x_2) \) and residuals versus predicted values obtained using the CLS are shown in Fig. 16. The graphs obtained using the MCC (not shown here) exhibit similar patterns.
Table 8
Tree data: point and confidence interval estimation of the coefficients when using CLS criterion (left) and MCC criterion (right) for the general model

<table>
<thead>
<tr>
<th>Term</th>
<th>Mean</th>
<th>Bootstrap C.I.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\hat{f}(y)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$y$</td>
<td>1.7840</td>
<td>(0.9840, 7.446)</td>
</tr>
<tr>
<td>$y^2$</td>
<td>-0.2928</td>
<td>(-3.7110, 0)</td>
</tr>
<tr>
<td>Constant</td>
<td>-0.3588</td>
<td>(-2.4020, -0.0044)</td>
</tr>
<tr>
<td>$h_1(x_1)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_1$</td>
<td>3.9840</td>
<td>(0.23, 25.2)</td>
</tr>
<tr>
<td>$x_2$</td>
<td>-0.7129</td>
<td>(0.0)</td>
</tr>
<tr>
<td>$h_2(x_2)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_1 \cdot x_2$</td>
<td>-0.2457</td>
<td>(0.0)</td>
</tr>
<tr>
<td>$x_2$</td>
<td>0.1356</td>
<td>(0.1, 5.660)</td>
</tr>
<tr>
<td>$h_{12}(x_1, x_2)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_1 \cdot x_2$</td>
<td>18.91</td>
<td>(0.21720)</td>
</tr>
<tr>
<td>$x_2$</td>
<td>0</td>
<td>(0.0)</td>
</tr>
<tr>
<td>$x_1 \cdot x_2$</td>
<td>0</td>
<td>(0.0)</td>
</tr>
<tr>
<td>MCC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\hat{f}(y)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$y$</td>
<td>0.9573</td>
<td>(0.8872, 1)</td>
</tr>
<tr>
<td>$y^2$</td>
<td>-0.0459</td>
<td>(-0.4531, 0)</td>
</tr>
<tr>
<td>Constant</td>
<td>-0.0761</td>
<td>(-0.3845, -0.0028)</td>
</tr>
<tr>
<td>$h_1(x_1)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_1$</td>
<td>0.8640</td>
<td>(0.51910)</td>
</tr>
<tr>
<td>$x_2$</td>
<td>0</td>
<td>(0.0)</td>
</tr>
<tr>
<td>$h_2(x_2)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_1 \cdot x_2$</td>
<td>0.0189</td>
<td>(0.0, 0.2131)</td>
</tr>
<tr>
<td>$x_2$</td>
<td>0</td>
<td>(0.0)</td>
</tr>
<tr>
<td>$h_{12}(x_1, x_2)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_1 \cdot x_2$</td>
<td>0.0860</td>
<td>(0.0)</td>
</tr>
<tr>
<td>$x_2$</td>
<td>0</td>
<td>(0.0)</td>
</tr>
<tr>
<td>$x_1 \cdot x_2$</td>
<td>15.73</td>
<td>(0.2155)</td>
</tr>
<tr>
<td>$x_2$</td>
<td>0</td>
<td>(0.0)</td>
</tr>
</tbody>
</table>

Table 7 shows the point and confidence interval estimation of the coefficients. Note that all terms included in the selected model are significantly different from 0, except the $x_2$-coefficient when the model is selected with the MCC criteria. However, its distribution is highly asymmetric with a median equal to 0.0968, which is much closer to the estimator obtained with the full data set (0.0976) than the mean (0.0244) obtained from the 500-times 10-fold samples.

Selected models using the general model are:

1) the CLS criterion: $y = -0.0097 + 21.7200 \cdot x_1^2 \cdot x_2$, and
2) the MCC criterion: $y = -0.0030 + 21.2437 \cdot x_1^2 \cdot x_2$.

The two models are very close to the volume of the cone proposed by Atkinson (1985). Table 8 shows the point and confidence interval estimation of the coefficients. Note that, even though, the confidence interval contains 0 for the term $x_1^2 \cdot x_2$, the histograms of the coefficients (Figs. 17 and 18) show that most of the probability is concentrated in positive values. We also observe that for the constant term the mode is placed in the interval that contains 0. The same happens for the terms which are not in the selected model but their means are different from zero.

As can be seen, we have obtained the same results as those obtained by Atkinson (1985) who examines several residual plots and uses the Box–Cox transformation technique. Instead, we use an automatic search method and we consider the transformation of all the variables and the inclusion of interactions.

8.3. Diabetes data

Data description: This data set is a result of a study by Sockett et al. (1987) for determining the factors affecting patterns of insulin-dependent diabetes mellitus in children. The objective is to investigate the dependence of the level of
Fig. 17. Tree data: histograms of the constant term ($\beta_0$) (left) and term $x_1^2 \cdot x_2$ in $\hat{h}_{12}(x_1, x_2)$ (right) when using CLS criterion.

Fig. 18. Tree data: histograms of the constant term ($\beta_0$) (left) and term $x_1^2 \cdot x_2$ in $\hat{h}_{12}(x_1, x_2)$ (right) when using MCC criterion.

Fig. 19. Diabetes data: scatter plots of $\hat{h}_1(x_1)$ versus $x_1$ (left) and $\hat{h}_2(x_2)$ versus $x_2$ (right) for selected model with CLS criterion. The points plotted are partial residuals: the fitted values for each function plus the overall residuals from the additive model.
Fig. 20. Diabetes data: additive surface constructed from the functional network additive model.

Table 9
Transformations in the models obtained by Harrison and Rubinfeld (1978) (HR), Breiman and Friedman (1985) (ACE) and the functional network additive model (FN), for Boston Housing data set

<table>
<thead>
<tr>
<th>Variable</th>
<th>HR model</th>
<th>ACE model</th>
<th>FN model</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y$</td>
<td>log $y$</td>
<td>Milder than log</td>
<td>$7.2632 y - 0.2215 y^2 + 0.0022 y^3$</td>
</tr>
<tr>
<td>$x_1$</td>
<td>$x_1$</td>
<td>$-$</td>
<td>$-0.7619 x_1 + 0.0059 x_1^2$</td>
</tr>
<tr>
<td>$x_2$</td>
<td>$x_2$</td>
<td>$-$</td>
<td>$-$</td>
</tr>
<tr>
<td>$x_3$</td>
<td>$x_3$</td>
<td>$-$</td>
<td>$0.2330 x_3 - 0.0001 x_3^3$</td>
</tr>
<tr>
<td>$x_4$</td>
<td>$x_4$</td>
<td>$-$</td>
<td>$x_4$</td>
</tr>
<tr>
<td>$x_5$</td>
<td>$x_5^2$</td>
<td>Nonmonotonic</td>
<td>$x_5^2$</td>
</tr>
<tr>
<td>$x_6$</td>
<td>$x_6^2$</td>
<td>Different than $x_6^2$</td>
<td>$-$</td>
</tr>
<tr>
<td>$x_7$</td>
<td>$x_7$</td>
<td>$-$</td>
<td>$x_7$</td>
</tr>
<tr>
<td>$x_8$</td>
<td>log $x_8$</td>
<td>$-$</td>
<td>$x_8^3$</td>
</tr>
<tr>
<td>$x_9$</td>
<td>log $x_9$</td>
<td>$-$</td>
<td>$x_9^3$</td>
</tr>
<tr>
<td>$x_{10}$</td>
<td>$x_{10}$</td>
<td>Some transformation</td>
<td>$-$</td>
</tr>
<tr>
<td>$x_{11}$</td>
<td>$x_{11}$</td>
<td>$x_{11}$</td>
<td>$x_{11}^2$</td>
</tr>
<tr>
<td>$x_{12}$</td>
<td>$x_{12}$</td>
<td>$-$</td>
<td>$-$</td>
</tr>
<tr>
<td>$x_{13}$</td>
<td>log $x_{13}$</td>
<td>log $x_{13}$</td>
<td>$-0.0445 x_{13}^2 + 0.0008 x_{13}^3$</td>
</tr>
</tbody>
</table>

serum C-peptide on various other factors in order to understand the patterns of residual insulin secretion. The variables in this data set are:

$y$ logarithm of C-peptide concentration (pmol/ml) at diagnosis,

$x_1$ children age,

$x_2$ base deficit (a measure of acidity).

Hastie and Tibshirani (1990) use this data set to illustrate an example of an additive model when there are two predictors.

Results using functional networks: We obtain the same results as those found in Hastie and Tibshirani (1990, p. 87), Fig. 4.3 and p. 88, Fig. 4.4, when using scatterplot smoothers which need an iterative algorithm instead of just the solution of a linear system of equations as in our method. The results are shown in Figs. 19 and 20.
Fig. 21. Boston Housing data: transformations of the response $y$ and some explanatory variables suggested by the selected model.
Fig. 22. Boston Housing data: scatter plots of \( \hat{f}(y) \) versus the adjusted function in the explanatory variables (left) and residuals versus predicted values (right) for selected model.

8.4. Boston Housing data

Boston Housing data set contains 506 observations on 12 continuous variables, one discrete variable and one binary variable related to housing values in suburbs of Boston. The variables are:

- \( y \)  median value of owner-occupied homes in dollar 1000’s,
- \( x_1 \)  per capita crime rate by town,
- \( x_2 \)  percentage of residential land zoned for lots over 25,000 sq.ft.,
- \( x_3 \)  percentage of nonretail business acres per town,
- \( x_4 \)  Charles River dummy variable ( = 1 if tract bounds river; 0 otherwise),
- \( x_5 \)  nitric oxides concentration (parts per 100 millions),
- \( x_6 \)  average number of rooms per dwelling,
- \( x_7 \)  percentage of owner-occupied units built prior to 1940,
- \( x_8 \)  weighted distances to five Boston employment centers,
- \( x_9 \)  index of accessibility to radial highways,
- \( x_{10} \)  full-value property–tax rate per dollar 10,000,
- \( x_{11} \)  pupil–teacher ratio by town,
- \( x_{12} \)  \((Bk - 0.63)^2\) where \( Bk \) is the proportion of blacks by town,
- \( x_{13} \)  proportion of lower status of the population.

The purpose is to find the best fitting functional form and, in particular, to determine the pattern of the influence of air pollution on housing values as measured by \( x_5 \).

The analysis by Harrison and Rubinfeld (1978) leads to a linear regression model where \( y, x_8, x_9 \) and \( x_{13} \) are transformed by logarithms and \( x_5 \) and \( x_6 \) are squared. The exponent of \( x_5 \) is determined using Box–Cox transformations and a grid search. All the explanatory variables except \( x_2, x_3 \) and \( x_7 \) are highly significant. Pace (1993) obtains similar results by applying kernel nonparametric regression.

Breiman and Friedman (1985) apply the ACE algorithm to the transformed variables suggested by Harrison and Rubinfeld in order to determine if those transformations are close to the optimal ones. Linear functions are expected to be obtained with ACE in that case. They use a forward stepwise procedure, which includes in the model \( x_5^2, x_{10}, x_{11} \) and \( \log x_{13} \), as best predictors. The obtained plots suggest a milder transformation than logarithmic for \( y \), a transformation for \( x_6 \) different from squared and some transformation for \( x_{10} \). The authors added \( x_5^2 \) to the four variable model in order to estimate its marginal effect on median home value. A nonlinear and nonmonotonic structure is suggested by ACE.

Finally, Breiman et al. (1993) model these data using regression trees. They obtain a model for \( y \) and only four explanatory variables: \( x_1, x_6, x_8 \) and \( x_{13} \). The rest of the variables do not appear in splits. Any variable needs to be transformed.

Results using functional networks: The selected model using the additive model with the unknown functions approximated by third degree polynomials are shown in the third column of Table 9 and in Fig. 21. The transformation
suggested for \( y \) is different from logarithmic but with a structural change in the middle values of \( y \), as suggested by ACE. The variable \( x_5 \) is squared as suggested by Harrison and Rubinfeld (1978). The transformations of \( x_1, x_3 \) and \( x_{11} \) are almost linear, except for high values of the two first explanatory variables. Finally, the transformations of \( x_8, x_9 \) and \( x_{13} \) are not logarithmic but cubic for the two first variables and almost linear, except for high values, for the last.

The scatter plots of \( \hat{f}(y) \) versus the adjusted function in the explanatory variables and residuals versus predicted values are depicted in Fig. 22. Both show no evidence of additional structure not captured in the model.

9. Conclusions

A method for obtaining the optimal transformations of the explanatory and/or the response variables is presented. The method is based on functional networks for model construction (structural learning). Two such models are proposed in Section 3. As discussed in Section 5, the method also uses two criteria for parametric learning: the CLS and the MCC. When the forms of the functions are unknown, we use parametric approximations. Uniqueness and model selection problems are discussed in Sections 4 and 6.

Since the learning methods are based on parametric techniques, our technique have the following advantages with respect to the nonparametric techniques:

1. The learning procedure for the transformations is closed (fixed time) and is much faster that the iterative technique used for the nonparametric approach.
2. Once the transformations are learnt, they can be applied directly to data, instead of the interpolating formulas used by nonparametric techniques.

Based on the simulation study of Section 7, the method provides good approximations even in simple often occurring functions, which is not always possible with nonparametric techniques. The application to simulated and real data confirms that the method is useful for solving practical cases.

We have also compared our method with other methods for discovering transformations of the response and the explanatory variables such as examining residual plots, Box–Cox procedure and ACE. Furthermore, we have compared our results with those obtained when using nonparametric scatterplot smoothers and regression trees. Our method has shown to be a powerful automatic technique to model nonlinear relationships among variables. It provides the functional form of the nonlinear contribution of each variable to the model, which allows us to understand the properties of the true underlying function. Furthermore, it is computationally fast and it provides smooth approximations.

The proposed method can be seen as a procedure for global modeling such as multiple linear regression and neural networks. Moreover, our proposed technique selects automatically predictor variables and interaction terms (as in MARS), and, simultaneously, provides the response transformation of the response variables if it is necessary (as in ACE or AVAS).

References


