**Neural Computing and Applications manuscript No.**
(will be inserted by the editor)

# Fast Statistical Regression in Presence of a Dominant Independent Variable

**Simone Fiori**

Received: March 7, 2012. Revised: April 6, 2012

**Abstract** Bivariate statistical regression is a statistical tool that allows performing regression on a multivariate data set under the hypothesis that one of the independent variables is dominant. Statistical regression is profitable when the amount of available data is enough to explain the relevant statistical features of the phenomenon underlying the data. The present paper suggests a fast statistical regression method based on a neural system that is able to match its input-output statistic to the marginal statistic of the available data sets. A key point of the implementation proposed in the present paper is that it is based on purely numerical-algebraic operations, which guarantee a computationally advantageous way of implementing neural systems. A number of numerical experiments, performed on real-world data sets, provide some insights into the behaviour of the devised neural-system-based statistical regression method and its limitations.

**Keywords** Statistical regression · Bivariate regression · Nonparametric regression · Dominant independent variate · Numerical-algebraic neural systems

## 1 Introduction

A number of real-world phenomena cannot be accurately described by a mathematical model to be evaluated analytically. In this case, statistical regression provides a useful tool to build up a model of the phenomenon under observation. Also, statistical regression plays a prominent role in those applications where incomplete data only are available. Data being incomplete means that specific observations are either lost or are not recorded exactly. Missing-data imputation and handling is a rapidly evolving field counting many methods, each applicable in some specific circumstances [15]. When choosing a missing data handling approach, one of the desired outcomes is maintaining the shape of the original data distribution. Applications of statistical regression techniques are found in oceanographic data analysis [1], characterization of the mechanical features of polymer composites [2], analysis of blood pressure data [3], earthquake hazard assessment [4], data handling methods in psychosomatic medicine [6], estimation of social practices diffusion processes [12], analysis of cardiorespiratory data [16], analysis of market data to improve retailer strategy [17], handling data in educational research [18], analysis of climate data [22] and meta-analysis of diagnostic test data in medicine [26].

The processing of bivariate experimental records in natural sciences is a typical task of data analysis. Such a task can include a separate processing of two variates by univariate techniques as well as an application of a bivariate technique which performs a joint analysis. There exist numerous techniques that provide information about an interrelation between two variates. However, often the analysis goes beyond such a task and aims at revealing some information about the system which generates the data. By making such a step, the system cannot be considered as a black box, but requires a certain knowledge or a set of assumptions. A typical assumption is that the system belongs to a certain class of models, for example that it is an input-output system, a delay line or a set of coupled active oscillators. The interpretation of the results then crucially depends on the correctness of the assumptions concerning the model [19].

Dipartimento di Ingegneria dell'Informazione, Università Politecnica delle Marche, Via Brecce Bianche, I-60131 Ancona (Italy) E-mail: s.fiori@univpm.it

Statistical bivariate regression is useful when the phenomenon under observation may be described by 2 variables (the independent variable, or *cause*, and the dependent variable, or *effect*) or when it may be described by $n+2$ variables, with $n>0$ (an independent dominant variable, a set of $n$ secondary independent variables, or *nuisance parameters*, and the dependent variable). Also, statistical regression is useful when the amount of available data is sufficiently large to show relevant statistical features of the phenomenon underlying them. A painting of current state of the art in statistical regression and modeling may be gotten from contributions in the field of statistical modelling in economics and finance [5], applied non-parametric regression [13] and missing data handling [21].

A main assumption on the regression model is that it is *monotonically increasing or decreasing*. The hypothesis of ordering in data modelling occurs frequently in applied fields such as data regression and data mining [20,25] and is related to the notion of *dose-response type relationship* that models a physical situation where a change in intensity of exposure results in a change, either increasing or decreasing, of risk of a specific outcome.

In the present paper, the statistical regression model is implemented by a numerical-algebraic neural system that is able to match its own input-output statistic to the empirical statistic of the available data sets. Namely, statistical regression is interpreted as an input-output statistic matching problem for a or neural system. Therefore, instead of considering the $x \in \mathcal{D}_x$ and $y \in \mathcal{D}_y$ variables as paired and to look for a non-linear model that fits the best the variables values, the statistical information carried on by the data sets $\mathcal{D}_x$ and $\mathcal{D}_y$ in terms of their probability density functions only are made use of. *The neural system's non-linear transference function matches the probability distributions of the variables rather than the variables' values.* The quantities of interest as well as the learnt regression model are represented in terms of vectors of numbers. As a result, the devised regression algorithm does not involve cumbersome computations except for sorting/searching on vectors of numbers and few simple algebraic operations on numbers. In addition, the operations on the data sets $\mathcal{D}_x$ and $\mathcal{D}_y$ might be largely parallelized. Except for monotonicity, the shape of the regression model is unrestricted.

Paper organization. Section 2 discusses the statistical regression problem and the proposed solution in analytic terms in subsection 2.1. The subsection 2.2 introduces the notion of numerical-algebraic neural system and the basic operations on such systems. The subsection 2.3 presents the numerical procedure for statistical regression corresponding to the devised problem setting and analytic solution. Section 3 illustrates the results of numerical experiments obtained real-world data sets in order to provide some insights into the numerical behaviour of the devised neural-system-based statistical regression method and into its merits and limitations. Section 4 concludes the paper.

## 2 Statistical regression by statistic matching

The present section explains the statistical regression problem in an analytic fashion and sets up the framework for the development of the actual numerical-algebraic implementation. It is assumed that the $n+2$ variates of interest in the regression problem are related by:

$$y = \Phi(x, z_1, \ldots, z_n), \tag{1}$$

where $y \in \mathcal{Y}$ represents the dependent variable, $x \in \mathcal{X}$ represents the dominant independent variable and $(z_1, \ldots, z_n) \in \mathcal{Z} \subset \mathbb{R}^n$ represent the nuisance parameters, according to the physical model described by the function $\Phi : \mathcal{X} \times \mathcal{Z} \to \mathcal{Y}$. The joint statistical features of the variates are described by the joint probability density function $p_{y,x,z_1,\ldots,z_n}(y, x, z_1, \ldots, z_n)$, where, by a slight abuse of notation, the variates have been confused with their realizations. The marginal joint probability density function of the dependent variable and of the dominant independent variable is described by:

$$p_{y,x}(y,x) = \int_{\mathcal{Z}} p_{y,x,z_1,\ldots,z_n}(y, x, z_1, \ldots, z_n) \mathrm{d}z_1 \cdots \mathrm{d}z_n. \tag{2}$$

The marginal probability density function of the dependent variable is related to the marginal joint probability density function of the dependent variable and of the dominant independent variable by the relationship:

$$p_y(y) = \int_{\mathcal{X}} p_{y,x}(y,x) \mathrm{d}x, \tag{3}$$

and the marginal joint probability density function of the dependent variable and of the dominant independent variable may be written in terms of the conditional probability density function of the dependent variable conditioned to the dominant independent variable and the marginal probability density function of the dominant independent variable as:

$$p_{y,x}(y,x) = p_{y|x}(y|x) p_x(x). \tag{4}$$

The assumption that the independent variate $x$ is dominant implies that the actual multivariate model (1) may

be approximated by a bivariate model $y = f(x)$ with $f$ to be determined, namely, that:

$$p_{y|x}(y|x) = \delta(y - f(x)), \tag{5}$$

where the symbol $\delta$ denotes the Dirac's *delta* distribution. Replacing equation (5) into equation (4) and then the result into equation (3), gives:

$$p_y(y) = \int_{\mathcal{X}} \delta(y - f(x)) p_x(x) \mathrm{d}x. \tag{6}$$

By the variable-change $u = f(x)$, the latter equality becomes:

$$p_y(y) = \int_{\mathcal{U}} \delta(y - u) p_u(u) \mathrm{d}u, \tag{7}$$

where $p_u$ denotes the probability density function of the variable $u \in \mathcal{U}$. The equation (7) is clearly an identity and tells that the model $f$ is consistent as long as it obeys the mass-transformation law

$$p_x(x)\mathrm{d}x = \pm p_u(u)\mathrm{d}u, \tag{8}$$

where $\mathrm{d}u = f'(x)\mathrm{d}x$.

### 2.1 Statistical regression: Analytic setting and solution

The regression system under consideration was assumed to be described by the transference $y = f(x)$, where $x \in \mathcal{X} \subseteq \mathbb{R}$ denotes the input variate, having probability density function $p_x$, and $y \in \mathcal{Y} \subseteq \mathbb{R}$ denotes the output variate, having probability density function $p_y$.

In the hypothesis that the neural regression system's transference $f : \mathcal{X} \to \mathcal{Y}$ is strictly monotonic, namely, that $f'(x) > 0$, $\forall x \in \mathcal{X}$, or $f'(x) < 0$, $\forall x \in \mathcal{X}$, then the relationship between the input distribution, the output distribution and the system transfer function arises from the probability mass-conservation law (8):

$$p_y(y) = \pm \left. \frac{p_x(x)}{f'(x)} \right|_{x = f^{-1}(y)}, \ y \in \mathcal{Y}, \tag{9}$$

where $f^{-1} : \mathcal{Y} \to \mathcal{X}$ denotes the inverse of function $f : \mathcal{X} \to \mathcal{Y}$ and the sign '+' arises when the non-linear function $f$ is monotonically *increasing* while sign '−' arises when the function $f$ is monotonically *decreasing*.

The equation (9) may be interpreted as a formula that allows for designing the non-linear regression system when the distribution $p_x$ is known and it is desired that the system responds according to a desired distribution $p_y$. In fact, the equation (9) may be rewritten as the differential equation:

$$\frac{\mathrm{d}f(x)}{\mathrm{d}x} = \pm \frac{p_x(x)}{p_y(f(x))}, \ x \in \mathcal{X}, \tag{10}$$

in the unknown $f : \mathcal{X} \to \mathcal{Y}$. In order for the above equation to be consistent, it is required that the function $p_y$ be nonzero in the domain of interest of the variable $y$. In general, solving the equation (10) in the unknown $f$ implies solving a non-linear differential equation, provided that a consistent boundary condition is specified.

Define the unit-step function:

$$1(x) \stackrel{\text{def}}{=} \int_{-\infty}^{x} \delta(u)\mathrm{d}u \tag{11}$$

and the *expectation operator* $\mathbb{E}$ as:

$$\mathbb{E}_x[g(x)] \stackrel{\text{def}}{=} \int_{\mathcal{X}} p_x(x) g(x) \mathrm{d}x, \tag{12}$$

where $g : \mathcal{X} \to \mathbb{R}$. Define the cumulative distribution functions associated to the input-output probability density functions:

$$P_x(\xi) \stackrel{\text{def}}{=} \mathbb{E}_x[1(\xi - x)], \text{ if } \xi \in \mathcal{X}, \tag{13}$$

$$P_y(\xi) \stackrel{\text{def}}{=} \mathbb{E}_y[1(\xi - y)], \text{ if } \xi \in \mathcal{Y}, \tag{14}$$

with their natural constant prolongation for $\xi \notin \mathcal{X}$ or $\xi \notin \mathcal{Y}$. In the case that the cumulative distribution functions associated to the input-output probability density functions are known, the solution of the differential equation (10) may be written in closed form as:

$$f(x) = P_y^{-1}(c \pm P_x(x)), \tag{15}$$

where the constant $c$ depends on the boundary condition, symbol $P_y^{-1}$ denotes the inverse of the cumulative density function $P_y$ and the sign is chosen according to the desired slope type for the regression model-function.

The non-linear function $f(\cdot)$ is sought for in such a way that it maps the data distribution $p_x$ into the distribution $p_y$. Therefore, a sensible choice for the constant $c$ in the model (15) is such that it maps the center of mass of the $x$-distribution into the center of mass of the $y$-distribution. Denote by $\overline{x}$ and $\overline{y}$ the mean-values of the $x$ and $y$ data sets respectively, namely:

$$\overline{x} \stackrel{\text{def}}{=} \mathbb{E}_x[x], \ \overline{y} \stackrel{\text{def}}{=} \mathbb{E}_y[y], \tag{16}$$

then the mass-center-to-mass-center-map condition above may be written as:

$$P_y(\overline{y}) = c \pm P_x(\overline{x}). \tag{17}$$

As it holds that:

$$P_y(\overline{y}) = P_x(\overline{x}) = \frac{1}{2}, \tag{18}$$

the condition (17) gives rise to the following solutions for the increasing/decreasing model type separately:

– **Monotonically increasing model**: The constant $c$ takes on the value 0, therefore the solution (15) to the equation (10) writes $f(x) = P_y^{-1}(P_x(x))$.
– **Monotonically decreasing model**: The constant $c$ takes on the value 1, therefore the solution (15) to the equation (10) writes $f(x) = P_y^{-1}(1 - P_x(x))$.

The problem of sign selection in the equation (15) is solvable by reasoning on the nature of the physical phenomena underlying the involved data sets.

The conclusion drawn from the above problem-setting and analytic solution is that, in order to develop a fully-numerical method for statistical regression, a suitable numerical estimation method for cumulative density functions and a suitable numerical format for function representation/handling are required.

### 2.2 Numerical-algebraic neural systems

Numerical-algebraic neural systems provide a suitable numerical representation for functions and probability-distributions-type quantities and for the related computations. A $N$-point numerical-algebraic neural system is represented by a pair of vectors $(\mathbf{x}, \mathbf{y})$, where $\mathbf{x} \in \mathbb{R}^N$ and $\mathbf{y} \in \mathbb{R}^N$. The entries $x_k$ of the vector $\mathbf{x}$ and $y_k$ of the vector $\mathbf{y}$, with $k \in \{1, \ \ldots, \ N\}$, are paired and provide a point-wise description of an arbitrarily-shaped function.

In order to handle $N$-point numerical-algebraic neural systems for statistical regression purpose, a number of operations are defined, which coincide as much as possible to the implementations of the corresponding operators provided by MATLAB©, in order to make the computer implementation of the learning procedure as straightforward as possible.

Formally, a $N$-point numerical-algebraic neural system is created by the operator $(\mathbf{x}, \mathbf{y}) = \text{nans}(N)$, which initializes the $N$ entries of the vectors $\mathbf{x}$ and $\mathbf{y}$ to zero.

On the basis of a $N$-point numerical-algebraic neural system $(\mathbf{x}, \mathbf{y})$, a new numerical-algebraic neural system may be constructed, whose $y$-entries contain the cumulative sum of the $y$-entries of the look-up table $(\mathbf{x}, \mathbf{y})$. Such an operation is represented by the notation $(\mathbf{x}, \mathbf{v}) = \text{cumsum}(\mathbf{x}, \mathbf{y})$, where:

$$v_k = \sum_{h=1}^{k} y_h \ , \ k \in \{1, \ \ldots, \ N\}.$$

The *cumsum* operator constructs a numerical-algebraic neural system $(\mathbf{x}, \mathbf{v})$ that has the same size of the numerical-algebraic neural system $(\mathbf{x}, \mathbf{y})$.

In order to numerically approximate the probability density function of a one-dimensional numerical data set, a histogram operator is defined. Denote by $\mathcal{D}$ such a data set, represented by a numerical vector, and with $B \geq 2$ the number of bins for frequency estimation. The histogram-estimation operation may be described by $(\mathbf{x}, \mathbf{y}) = \text{hist}(\mathcal{D}, B)$. The constructed numerical-algebraic neural system has $B$ points and is built up as follows:

$x_b$ equals the value of the $b^{\text{th}}$ bin center,
$y_b$ equals the number of data-points falling within the $b^{\text{th}}$ bin,
$b \in \{1, \ \ldots, \ B\}$.

The width of each bin may be calculated as:

$$\Delta \stackrel{\text{def}}{=} \frac{\max\{\mathcal{D}\} - \min\{\mathcal{D}\}}{B}. \tag{19}$$

The bin centres are given by the relationship:

$$x_b = \min\{\mathcal{D}\} + (b-1)\Delta + \frac{\Delta}{2} = \min\{\mathcal{D}\} + b\Delta - \frac{\Delta}{2}. \tag{20}$$

The $b^{\text{th}}$ bin is taken as the interval $[\min\{\mathcal{D}\} + (b - 1)\Delta, \ \min\{\mathcal{D}\} + b\Delta)$, which is open on the right side.

If a function is given a point-wise representation by the help of a numerical-algebraic neural system $(\mathbf{x}, \mathbf{y})$, then its inverse function may be easily given a point-wise representation by the numerical-algebraic neural system $(\mathbf{y}, \mathbf{x})$. Namely, the inverse function is obtained by swapping the two vectors which represent the direct function.

A limitation of numerical-algebraic neural-system-based representation of functions is that the $(x, y)$-pairs for the represented relationship are known only on some points. Interpolation may be invoked to make computations with numerical-algebraic neural systems on other points in the domain. In the present context, it is necessary to preserve the monotonicity of a regression function/model, therefore, linear interpolation is made use of in the present paper. Denote by $\mathcal{D}$ the $x$-coordinate point-set represented by a numerical vector of size $D \geq 2$, where the function represented by a numerical-algebraic neural system $(\mathbf{x}, \mathbf{y})$ needs to be interpolated. The interpolation operation may be described by $\mathcal{I} = \text{interp}(\mathbf{x}, \mathbf{y}, \mathcal{D})$ and works as follows:

For every $d \in \{1 \ , \ldots \ , D\}$ :
$\mathcal{I}_d$ equals the linear interpolation of the datum
$\mathcal{D}_d$ within the numerical-algebraic neural system.

The built-up set $\mathcal{I}$ is of size $D$ and is to be considered an ordered set that will be represented by a numerical vector.

Given a data set $\mathcal{D}$ and an integer number $R$, a uniform sampler constructs a set of $R$ values by uniformly

| Operator | Description |
|---|---|
| $(\mathbf{x},\mathbf{y}) = \text{nans}(N)$ | Creates a $N$-point numerical-algebraic neural system |
| $(\mathbf{x},\mathbf{v}) = \text{cumsum}(\mathbf{x},\mathbf{y})$ | Constructs $(\mathbf{x},\mathbf{v})$ as the cumulative sum of $(\mathbf{x},\mathbf{y})$ |
| $(\mathbf{x},\mathbf{y}) = \text{hist}(\mathcal{D},B)$ | Computes the $B$-bins distribution histogram of the data set $\mathcal{D}$ |
| $(\mathbf{y},\mathbf{x})$ | Provides the inverse system of the system $(\mathbf{x},\mathbf{y})$ |
| $\mathcal{I} = \text{interp}(\mathbf{x},\mathbf{y},\mathcal{D})$ | Interpolates the system $(\mathbf{x},\mathbf{y})$ over the values set $\mathcal{D}$ |
| $\mathcal{B} = \text{linspace}(\mathcal{D},R)$ | Returns a set of $R$ values which subdivide uniformly the interval $[\min\{\mathcal{D}\},\ \max\{\mathcal{D}\}]$ |

**Table 1** Summary of the operators that act on $N$-point numerical-algebraic neural systems.

subdividing the interval $[\min\{\mathcal{D}\},\ \max\{\mathcal{D}\}]$. Such an operation is represented by $\text{linspace}(\mathcal{D},R)$, which returns a $R$-size set.

The used operators defined in the present section are summarized in the Table 1.

### 2.3 Numerical-algebraic implementation of the statistical regression method

The devised regression procedure is based on a numerical-algebraic neural system, described by the input-output relationship $y = f(x)$, where $x \in \mathbb{R}$ denotes the input variable, $y \in \mathbb{R}$ denotes the output variable and $f(\cdot)$ denotes a neural transfer function. The numerical-algebraic neural-system-based algorithm utilized in the present paper for numerical statistical regression implements the closed-form solution (15) by the help of the numerical-algebraic neural-system-handling operators defined in subsection 2.2.

First, it is necessary to estimate the probability density functions as well as the cumulative distribution functions of the data within the $\mathcal{D}_x$ and $\mathcal{D}_y$ sets. In the interpretation of statistical regression proposed in the present paper, the size $D_x$ of the data set $\mathcal{D}_x$ and the size $D_y$ of the data set $\mathcal{D}_y$ do not need to be equal. In order to perform a histogram-based estimation of the probability density function underlying the data via the *hist* operator, it is necessary to choose the number $B_x$ and $B_y$ of bins for the two data sets. The following rule was experimentally validated to select the number of bins:

$$B_x \overset{\text{def}}{=} \lceil 20 \log_{10} D_x \rceil,\ B_y \overset{\text{def}}{=} \lceil 20 \log_{10} D_y \rceil, \tag{21}$$

where $\lceil \cdot \rceil$ rounds its argument to the nearest integer towards infinity. Such a choice endows the histogram-based estimation procedure with enough bins to estimate the probability density functions of the $x$ and $y$ data, while the number of bins keeps limited. The probability density functions of the $\mathcal{D}_x$ and $\mathcal{D}_y$ data sets may thus be numerically estimated by:

$$(\mathbf{x},\mathbf{p}_x) = \text{hist}(\mathcal{D}_x, B_x),\ (\mathbf{y},\mathbf{p}_y) = \text{hist}(\mathcal{D}_y, B_y). \tag{22}$$

The vectors $\mathbf{p}_x$ and $\mathbf{p}_y$ actually represent probability distributions up to scale factors, as they should be normalized by the total number of samples in each set and by the bin-widths $\Delta_x$ and $\Delta_y$ computed by the formula (19). Such a simple probability estimation method has been successfully used in the context of machine learning in [7–9].

The numerical cumulative distribution functions of the $\mathcal{D}_x$ and $\mathcal{D}_y$ data sets may be estimated by numerical integration of the numerical probability density functions, which may be achieved by the help of the *cumsum* operator applied to numerical-algebraic neural systems $(\mathbf{x},\mathbf{p}_x)$ and $(\mathbf{y},\mathbf{p}_y)$. It should be noted, however, that two adjustments are necessary at this point.

Some entries of the vector $\mathbf{p}_y$ may be zero (therefore some entries of its cumulative-sum vector may be equal to others). This situation violates the hypothesis that the probability density function of the $y$ variable differs from zero in the interval of interest. Such an occurrence should be fixed in a way that does not alter the information content of the vector $\mathbf{p}_y$. As a possible remedy, a small quantity, namely $1/D_y$, is added to every entry of the vector $\mathbf{p}_y$. Therefore, the normalized numerical probability density function to be integrated is:

$$\hat{\mathbf{p}}_y \overset{\text{def}}{=} \left(\mathbf{p}_y + \frac{1}{D_y}\right) \frac{D_y}{B_y + D_y^2} = \frac{D_y \mathbf{p}_y + 1}{B_y + D_y^2}. \tag{23}$$

The entries of the vector $\mathbf{p}_x$ do not need any value-shifting, therefore, the corresponding normalized numerical probability density function to be integrated is:

$$\hat{\mathbf{p}}_x \overset{\text{def}}{=} \frac{\mathbf{p}_x}{D_x}. \tag{24}$$

It is straightforward to verify that the entries of vectors $\hat{\mathbf{p}}_x$ and $\hat{\mathbf{p}}_y$ sum up to one.

The *cumsum* operator provides a cumulative-sum vector whose first entry coincides to the first entry of the summed-up numerical-algebraic neural system, while the numerical cumulative distribution function's first entry should equal zero. Also, the *hist* operator provides the bin-centres coordinate, while for numerical function representation purposes, the boundary-values of

the bins are more profitable. Because of these reasons, first define the numerical-algebraic neural systems:

$$(\mathbf{x}, \mathbf{P}_x) = \mathrm{cumsum}(\mathbf{x}, \hat{\mathbf{p}}_x), \tag{25}$$

$$(\mathbf{y}, \mathbf{P}_y) = \mathrm{cumsum}(\mathbf{y}, \hat{\mathbf{p}}_y). \tag{26}$$

Then, define the actual numerical-algebraic neural systems representing true numerical cumulative distribution functions as follows:

$$(\hat{\mathbf{x}}, \hat{\mathbf{P}}_x) : \hat{\mathbf{x}} \stackrel{\mathrm{def}}{=} \left[ \min\{\mathcal{D}_x\} \ ; \ \mathbf{x} + \frac{\Delta_x}{2} \right], \tag{27}$$

$$\hat{\mathbf{P}}_x \stackrel{\mathrm{def}}{=} [0 \ ; \ \mathbf{P}_x], \tag{28}$$

$$(\hat{\mathbf{y}}, \hat{\mathbf{P}}_y) : \hat{\mathbf{y}} \stackrel{\mathrm{def}}{=} \left[ \min\{\mathcal{D}_y\} \ ; \ \mathbf{y} + \frac{\Delta_y}{2} \right], \tag{29}$$

$$\hat{\mathbf{P}}_y \stackrel{\mathrm{def}}{=} [0 \ ; \ \mathbf{P}_y], \tag{30}$$

where symbol [ ; ] denotes vector concatenation.

The latter operations make the numerical cumulative probability density function numerical-algebraic neural systems $(\hat{\mathbf{x}}, \hat{\mathbf{P}}_x)$, of size $D_x + 1$, and $(\hat{\mathbf{y}}, \hat{\mathbf{P}}_y)$, of size $D_y + 1$, available for the computation of the non-linear regression model according to equation (15).

If the statistical model is monotonically increasing, then the numerical-algebraic neural system $(\hat{\mathbf{x}}, \hat{\mathbf{P}}_x)$ may be used as is. Otherwise, for monotonically decreasing models, it should be replaced by $(\hat{\mathbf{x}}, 1 - \hat{\mathbf{P}}_x)$ in what follows.

The non-linear regression model is to be evaluated on a ordered set of $x$-points $\mathcal{X} \subseteq [\min\{\mathcal{D}_x\}, \ \max\{\mathcal{D}_x\}]$. Here, the ordered set $\mathcal{X}$ that the regression model is to be evaluated over consists of $R$ points equally spaced within the interval $[\min\{\mathcal{D}_x\}, \ \max\{\mathcal{D}_x\}]$, where $R$ depends on the accuracy of the interpolation required for the model $f(\cdot)$. The quantity $R$ appears as a finesse of partition for interpolation purpose. The last step in the procedure is to numerically evaluate the quantity $P_x(\cdot)$ over the set $\mathcal{X}$, then to evaluate the quantity $P_y^{-1}(\cdot)$ over the values of function $P_x(\cdot)$, namely:

$$\mathcal{P}_x \stackrel{\mathrm{def}}{=} \mathrm{interp}(\hat{\mathbf{x}}, \hat{\mathbf{P}}_x, \mathcal{X}), \ \mathcal{Y} \stackrel{\mathrm{def}}{=} \mathrm{interp}(\hat{\mathbf{P}}_y, \hat{\mathbf{y}}, \mathcal{P}_x). \tag{31}$$

In the latter equation, the inverse function $P_y^{-1}$ appears through the swapped numerical-algebraic neural system of function $P_y$. The ordered sets $\mathcal{X}$ and $\mathcal{Y}$, both of size $R$, give rise to the numerical-algebraic neural system representation $(\mathcal{X},\mathcal{Y})$ for the non-linear regression model $f(\cdot)$ to be designed.

The whole statistical regression procedure is summarized in the Algorithm 1 for the convenience of the reader. It is worth remarking that the pseudocode lines from 2 to 10 act independently on the $x$-dataset/variables and on the $y$-dataset/variables, hence they might be rewritten in a parallel fashion.

**Algorithm 1** Statistical bivariate regression procedure based on a numerical-algebraic neural system at a glance.

1: Input data sets $\mathcal{D}_x$ and $\mathcal{D}_x$ and interpolation finesse $R$
2: Extract sizes $D_x$ and $D_y$
3: Compute $B_x \stackrel{\mathrm{def}}{=} \lceil 20 \log_{10} D_x \rceil$ and $B_y \stackrel{\mathrm{def}}{=} \lceil 20 \log_{10} D_y \rceil$
4: Compute estimates $(\mathbf{x}, \mathbf{p}_x) = \mathrm{hist}(\mathcal{D}_x, B_x)$ and $(\mathbf{y}, \mathbf{p}_y) = \mathrm{hist}(\mathcal{D}_y, B_y)$
5: Compute normalizations $\hat{\mathbf{p}}_y = \frac{D_y \mathbf{p}_y + 1}{B_y + D_y^2}$ and $\hat{\mathbf{p}}_x \stackrel{\mathrm{def}}{=} \frac{\mathbf{p}_x}{D_x}$
6: Compute estimates $(\mathbf{x}, \mathbf{P}_x) = \mathrm{cumsum}(\mathbf{x}, \hat{\mathbf{p}}_x)$ and $(\mathbf{y}, \mathbf{P}_y) = \mathrm{cumsum}(\mathbf{y}, \hat{\mathbf{p}}_y)$
7: Create $(\hat{\mathbf{x}}, \hat{\mathbf{P}}_x) = \mathrm{nans}(B_x + 1)$
8: Set $\hat{\mathbf{x}} = \left[\min\{\mathcal{D}_x\} \ ; \ \mathbf{x} + \frac{\Delta_x}{2}\right]$ and $\hat{\mathbf{P}}_x = [0 \ ; \ \mathbf{P}_x]$
9: Create $(\hat{\mathbf{y}}, \hat{\mathbf{P}}_y) = \mathrm{nans}(B_y + 1)$
10: Set $\hat{\mathbf{y}} = \left[\min\{\mathcal{D}_y\} \ ; \ \mathbf{y} + \frac{\Delta_y}{2}\right]$ and $\hat{\mathbf{P}}_y = [0 \ ; \ \mathbf{P}_y]$
11: Compute $\mathcal{X} = \mathrm{linspace}(\mathcal{D}_x, R)$
12: **if** Monotonically increasing model invoked **then**
13: Compute $\mathcal{P}_x = \mathrm{interp}(\hat{\mathbf{x}}, \hat{\mathbf{P}}_x, \mathcal{X})$
14: **else**
15: Compute $\mathcal{P}_x = \mathrm{interp}(\hat{\mathbf{x}}, 1 - \hat{\mathbf{P}}_x, \mathcal{X})$
16: **end if**
17: Compute $\mathcal{Y} = \mathrm{interp}(\hat{\mathbf{P}}_y, \hat{\mathbf{y}}, \mathcal{P}_x)$
18: Output regression model $(\mathcal{X}, \mathcal{Y})$

## 3 Numerical experiments

In the present section, computer-based numerical experiments on real-world data sets are illustrated and commented on. In particular, a successful, a dubious and an unsuccessful regression case are presented in order to discuss the *dominant independent variate* assumption and to illustrate the merits and the limitations of the devised statistical regression method.

### 3.1 Successful regression case

The first regression test is conducted on a *body fat* data set obtained by the SOCR Database [23]. Such a data set contains a list of estimates of the percentage of body fat determined by underwater weighing and various body circumference measurements for 252 men. The purpose of conducting regressions on the *body fat* data set is that accurate measurement of body fat is inconvenient and costly and it is therefore desirable to devise easy methods for estimating body fat that are neither inconvenient nor costly. Patients assess their health, at least in part, by estimating their percentage of body fat. Some texts give predictive equations for body fat using body circumference measurements (e.g. abdominal circumference) and/or skin-fold measurements [14].

The data set contains the following 15 variates: Density determined from underwater weighing, Percentage of body fat, Age, Weight, Height, Neck circumference, Chest circumference, Abdomen circumference, Hip cir-

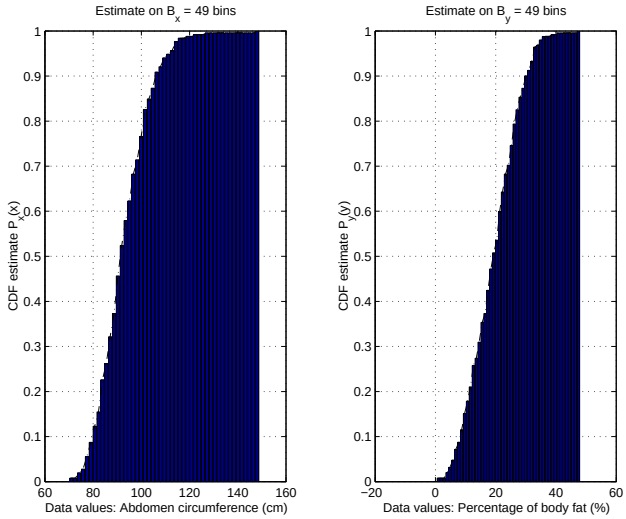


**Fig. 1** Experiment on *body fat* data: Cumulative distribution functions of the data sets $\mathcal{D}_x$ (*abdomen circumference* data) and $\mathcal{D}_y$ (*percentage of body fat* data) required by the modeling procedure. The data set sizes are $D_x = D_y = 252$.

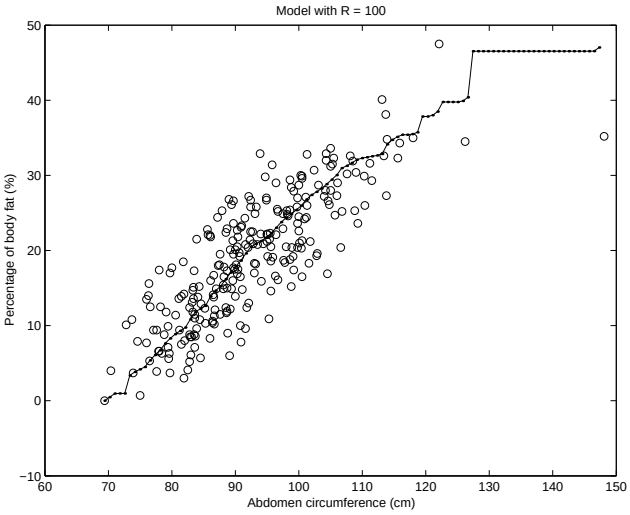


**Fig. 2** Experiment on *body fat* data: Data set and computed regression model represented by a 100-point numerical-algebraic neural system.

cumference, Thigh circumference, Knee circumference, Ankle circumference, Biceps circumference, Forearm circumference and Wrist circumference. In the present analysis, it is assumed that the dominant independent variable is the *abdomen circumference* and that the dependent variable is the *percentage of body fat*. The result of univariate analysis of the two variates separately, namely, of the estimation of their cumulative probability functions, is illustrated in the Figure 1. The result of regression by Algorithm 1 is illustrated in the Figure 2.

As it can be readily appreciated, the numerical model obtained with the proposed algorithm of subsection 2.3, on the basis of the estimated marginal distributions of

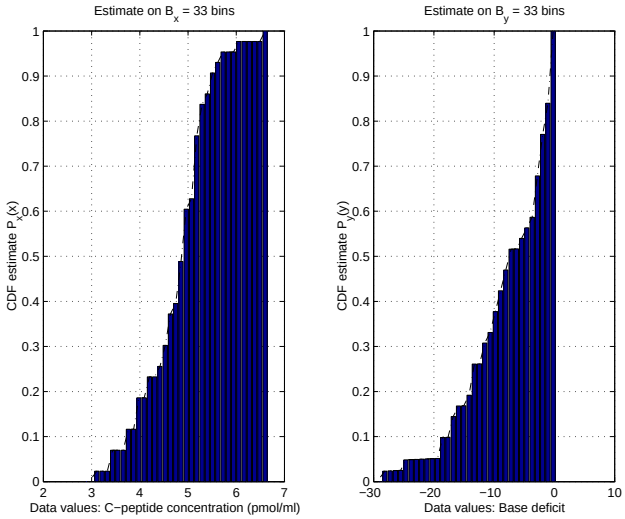


**Fig. 3** Experiment on *diabetes* data: Cumulative distribution functions of the data sets $\mathcal{D}_x$ (*C-peptide concentration* data) and $\mathcal{D}_y$ (*base deficit* data) required by the modeling procedure. The data set sizes are $D_x = D_y = 43$.

the dominant independent variate and of the dependent variate is in good agreement with the data and the dominant-independent-variate assumption is consistent with the data.

### 3.2 Dubious regression case

The second regression test is conducted on a *diabetes* data set obtained from the LIACC Repository [24]. Such a data set concerns the study of the factors affecting patterns of insulin-dependent *diabetes mellitus* in children. The objective is to investigate the dependence of the level of serum C-peptide on the various other factors in order to understand the patterns of residual insulin secretion. The response measurement is the logarithm of C-peptide concentration at the diagnosis and the predictor measurements age and base deficit (a measure of acidity). The available measures come in the form of 43 records in total.

The data set contains the following 3 variates: Age, Base deficit, C-peptide concentration. The available data are reported in the Table 2. In the present analysis, it is assumed that the dominant independent variable is the *C-peptide concentration* and that the dependent variable is the *base deficit*. The estimates of their cumulative probability functions are illustrated in the Figure 3, while the result of regression by the Algorithm 1 is illustrated in the Figure 4. The numerical model obtained with the proposed algorithm and the distribution of the data reveal that the regression model represented by a numerical-algebraic neural system is in good agreement with the data, which look however quite spread

| Age (Year) | Deficit | C-peptide (pmol/mol) |
|---|---|---|
| 5.2 | −8.1 | 4.8 |
| 8.8 | −16.1 | 4.1 |
| 10.5 | −0.9 | 5.2 |
| 10.6 | −7.8 | 5.5 |
| 10.4 | −29.0 | 5.0 |
| 1.8 | −19.2 | 3.4 |
| 12.7 | −18.9 | 3.4 |
| 15.6 | −10.6 | 4.9 |
| 5.8 | −2.8 | 5.6 |
| 1.9 | −25.0 | 3.7 |
| 2.2 | −3.1 | 3.9 |
| 4.8 | −7.8 | 4.5 |
| 7.9 | −13.9 | 4.8 |
| 5.2 | −4.5 | 4.9 |
| 0.9 | −11.6 | 3.0 |
| 11.8 | −2.1 | 4.6 |
| 7.9 | −2.0 | 4.8 |
| 11.5 | −9.0 | 5.5 |
| 10.6 | −11.2 | 4.5 |
| 8.5 | −0.2 | 5.3 |
| 11.1 | −6.1 | 4.7 |
| 12.8 | −1.0 | 6.6 |
| 11.3 | −3.6 | 5.1 |
| 1.0 | −8.2 | 3.9 |
| 14.5 | −0.5 | 5.7 |
| 11.9 | −2.0 | 5.1 |
| 8.1 | −1.6 | 5.2 |
| 13.8 | −11.9 | 3.7 |
| 15.5 | −0.7 | 4.9 |
| 9.8 | −1.2 | 4.8 |
| 11.0 | −14.3 | 4.4 |
| 12.4 | −0.8 | 5.2 |
| 11.1 | −16.8 | 5.1 |
| 5.1 | −5.1 | 4.6 |
| 4.8 | −9.5 | 3.9 |
| 4.2 | −17.0 | 5.1 |
| 6.9 | −3.3 | 5.1 |
| 13.2 | −0.7 | 6.0 |
| 9.9 | −3.3 | 4.9 |
| 12.5 | −13.6 | 4.1 |
| 13.2 | −1.9 | 4.6 |
| 8.9 | −10.0 | 4.9 |
| 10.8 | −13.5 | 5.1 |

**Table 2** *Diabetes* data set obtained from the LIACC Repository [24].

over the data space and around the model. Such a distribution leaves a dubious interpretation of the regression results and, in particular, on the dominance of the *C-peptide concentration* variate. Also, note that the number of available instances is quite limited.

### 3.3 Unsuccessful regression case

The third regression test was conducted on a *concrete* data set. It is anticipated that the present case leads to unsuccessful modelling results. Such a case *is presented on purpose to show the limitations of the devised procedure arising from the violation of the assumption of the*

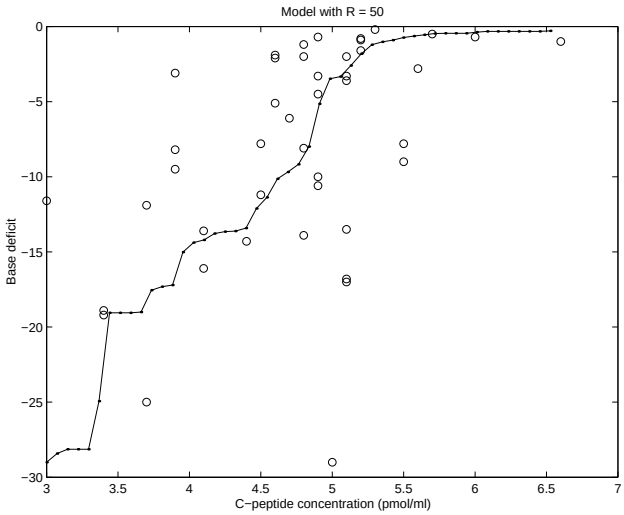


**Fig. 4** Experiment on *diabetes* data: Data set and computed regression model represented by a 50-point numerical-algebraic neural system.

*existence of a dominant variate* in the set of attributes. Concrete is an important material in civil engineering. The concrete compressive strength is a highly nonlinear function of age and concrete ingredients. Such ingredients include cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate as well as fine aggregate. In the available data set for regression purpose, the actual concrete compressive strength for a set of given mixtures under a specific age was determined from laboratory. Such a dataset was obtained by the UCI Repository [11]. The data set contains 1030 records.

The data set contains the following 9 variates: Cement, Blast furnace slag, Fly ash, Water, Superplasticizer, Coarse aggregate, Fine aggregate, Age and Concrete compressive strength. A multivariate regression method based on artificial neural networks was suggested in [27]. In the present test, it is assumed that the dominant independent variate is the *cement* and that the dependent variate is the *concrete compressive strength*. The estimates of their cumulative probability functions are illustrated in the Figure 5, while the result of regression by the Algorithm 1 is illustrated in the Figure 6. The numerical model obtained with the proposed algorithm of subsection 2.3 and the distribution of the data reveal that the choice of the dominant independent variate is unsuitable.

Further tests conducted by choosing a different independent variate as the dominant one reveal that the relationship between the concrete compressive strength and the concrete ingredients proportion and age is highly nonlinear and that there seem to be no dominant independent variate. The best result was obtained by

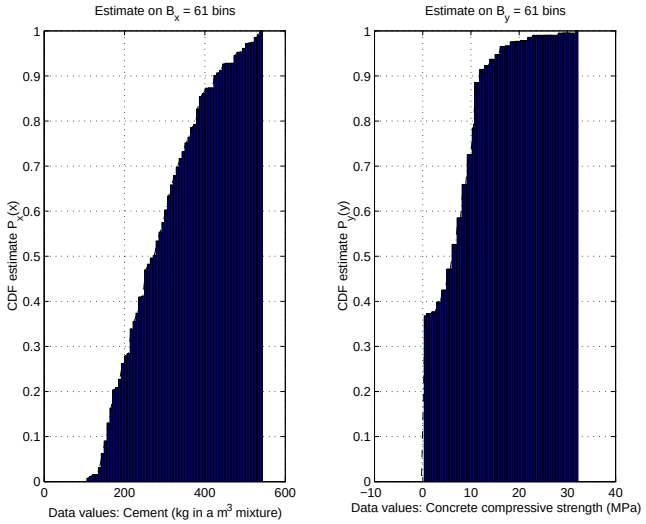


**Fig. 5** Experiment on *concrete* data: Cumulative distribution functions of the data sets $\mathcal{D}_x$ (*cement* data) and $\mathcal{D}_y$ (*concrete compressive strength* data) required by the modeling procedure. The data set sizes are $D_x = D_y = 1030$.

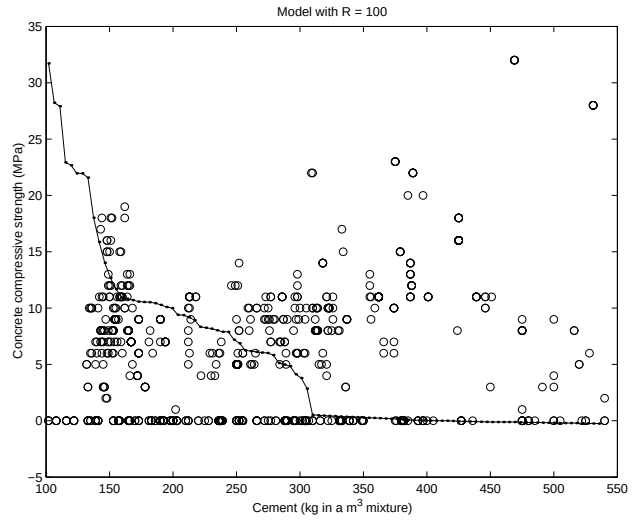


**Fig. 6** Experiment on *concrete* data: Data set and computed regression model represented by a 100-point numerical-algebraic neural system based on the assumption that the dominant independent variate is the *cement* variate.

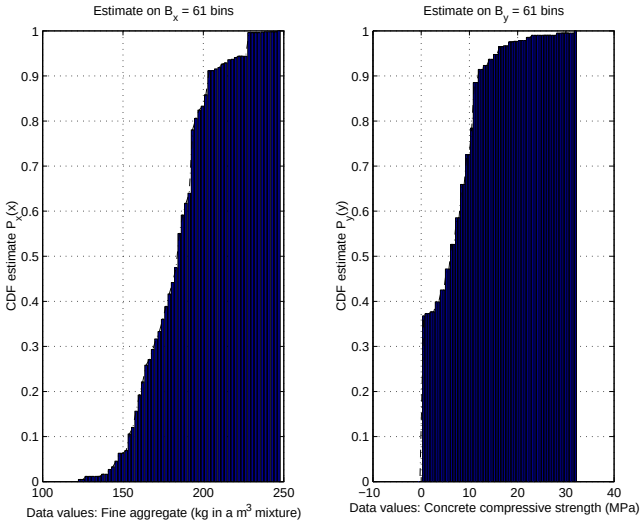


**Fig. 7** Experiment on *concrete* data: Cumulative distribution functions of the data sets $\mathcal{D}_x$ (*fine aggregate* data) and $\mathcal{D}_y$ (*concrete compressive strength* data) required by the modeling procedure. The data set sizes are $D_x = D_y = 1030$.

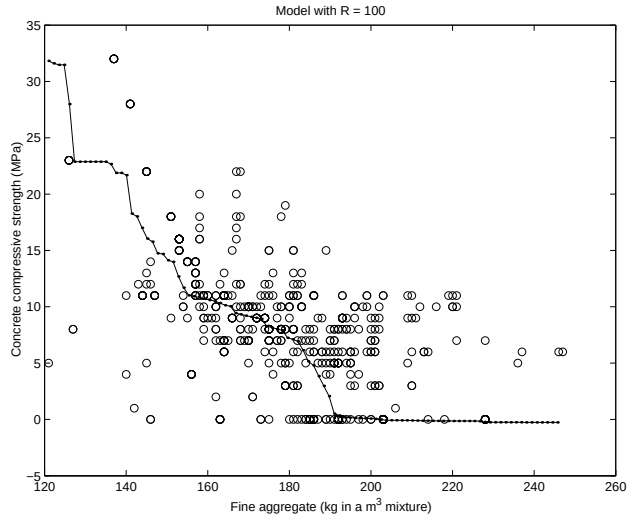


**Fig. 8** Experiment on *concrete* data: Data set and computed regression model represented by a 100-point numerical-algebraic neural system based on the assumption that the dominant independent variate is the *fine aggregate* variate.

assuming that the dominant independent variate is the *fine aggregate*. The estimates of their cumulative probability functions are illustrated in the Figure 7 and the result of regression by the Algorithm 1 is illustrated in the Figure 8. Even such a result appears quite unsatisfactory.

## 4 Conclusion

The aim of the present paper is to discuss the problem of bivariate statistical regression via numerical-algebraic neural systems that are able to match their input-output statistic to the empirical statistic of data sets, a relationship among which is sought for.

Statistical regression was given here an interpretation as a pooled-statistic matching problem for a non-linear system. Namely, instead of considering the $x \in \mathcal{D}_x$ and $y \in \mathcal{D}_y$ variables as paired and to look for a non-linear regression model that fits the variables values the best, only cumulative information that arise by pooling the values within the data sets $\mathcal{D}_x$ and $\mathcal{D}_y$ are made use of. The neural system non-linear transference thus matches the probability distributions of the variables, rather than the variables' values themselves. As a con-

sequence, the proposed statistical regression technique allows to cope with the regression problem even when the size of the two data sets do not match and/or when the pairing relationship of the values within $x \in \mathcal{D}_x$ and $y \in \mathcal{D}_y$ is unknown. Also, as the individual data in the sets $\mathcal{D}_x$ and $\mathcal{D}_y$ are not accessed directly by the regression procedure, if the data sets of the phenomena under regression are not available but only their pooled cumulative statistical distributions are available, the regression process may nevertheless take place. A key assumption is that the data are either bivariate or that there exists a dominant independent variable that the dependent variable is a function of.

A key feature of the devised regression method is that the quantities of interest are represented by numerical vectors and the regression model is represented in terms of a numerical-algebraic neural system, which provide an efficient way of representing and handling the variables appearing within the devised statistical regression algorithm. A prominent advantage of the procedure is the lack of hard computational requirement. Although, as an inherent restriction of the method, the developed theory requires the model to be monotonic, the regression-model shape is otherwise unrestricted, being thus free of any other shape-constraint and resulting free of biasing effects inherently tied to other regression methods.

In order to assess the numerical statistical regression method proposed in the present work, numerical experiments performed on real-world data sets were conducted. The results of numerical experiments showed that the whenever the dominant-independent-variate assumption is sensible, the computed regression model fits the data in a satisfactory way.

An extension of the discussed bivariate statistical regression approach to a 2-*independent-variates*-to-1-*dependent-variate* regression procedure is currently under investigation. The main challenge here is how to formulate the multivariate statistical regression by statistic matching. A possible solution under investigation seems to involve partial differential equations on the regression function based on joint probability density functions of the independent variates and of the dependent variate.

**Acknowledgements** The present paper is an extended version of the conference paper [10]. The author wishes to thank Andrew Leung for the invitation to submit the present extended version to the special issue of Neural Computation and Applications dedicated to the ICONIP'2011 conference.

## Biography

Simone Fiori received the Italian Laurea (Dr. Eng.) *cum laude* in electronics engineering in July 1996 from the University of Ancona (Italy), and the Ph.D. degree in electrical engineering (circuit theory) in March 2000 from the University of Bologna (Italy). In November 2005, he joined the Università Politecnica delle Marche, where he is currently an Adjunct Professor. His research interests include unsupervised learning theory for artificial neural networks, linear and non-linear adaptive discrete-time filter theory, geometrical methods for machine learning and signal processing. He is author of more than 150 refereed journal and conference papers. Dr. Fiori was the recipient of the 2001 "E.R. Caianiello Award" for the best Ph.D. dissertation in the artificial neural network field and the 2010 "Rector Award" as a proficient researcher of the Faculty of Engineering of the Università Politecnica delle Marche. He is currently serving as Associate Editor of Neurocomputing (Elsevier), Computational Intelligence and Neuroscience (Hindawi) and Cognitive Computation (Springer). Dr. Fiori was awarded several scholarships to visit RIKEN research institute (Japan), the Tokyo University of Agriculture and Technology (Japan), the Trondheim Technical University (Norway), the University of Bari (Italy) and the A$^\star$STAR Bioinformatics Institute (Singapore).

## References

1. J.M. Beckers and M. Rixen, *EOF calculations and data filling from incomplete oceanographic data sets*, Journal of Atmospheric and Ocean Technology, Vol. 20, No. 12, pp. 1839 – 1856, December 2003
2. J. Biagiotti, S. Fiori, L. Torre, M.A. López-Manchado and J.M. Kenny, *Mechanical properties of polypropylene matrix composites reinforced with natural fibers: A statistical approach*, Polymer Composites, Vol. 25, No. 1, pp. 26 – 36, 2004
3. N.R. Cook, *Imputation strategies for blood pressure data nonignorably missing due to medication use*, Clinical Trials, Vol. 3, No. 5, pp. 411 – 420, October 2006
4. G.R. Dargahi-Noubary and M. Razzaghi, *Earthquake hazard assessment based on bivariate exponential distribution*, Reliability Engineering and System Safety, Vol. 44, pp. 135 – 166, 1994
5. J. Dupacová, J. Hurt and J. Štepán, *Stochastic Modeling in Economics and Finance*, Kluwer Academic Publisher, Applied Optimization Series, Sept. 2002
6. C.K. Enders, *A primer on the use of modern missing-data methods in psychosomatic medicine research*, Psychosomatic Medicine, Vol. 68, No. 3, pp. 427 – 436, May 2006
7. S. Fiori, *Non-symmetric PDF estimation by artificial neurons: Application to statistical characterization of reinforced composites*, IEEE Transactions on Neural Networks, Vol. 14, No. 4, pp. 959 – 962, July 2003

8. S. Fiori and R. Rossi, *Statistical characterization of some electrical and mechanical phenomena by a neural probability density function estimation technique*, Neural Network World, Vol. 2, pp. 153 – 176, 2004
9. S. Fiori, *Neural systems with numerically-matched input-output statistic: Variate generation*, Neural Processing Letters, Vol. 23, No. 2, pp. 143 – 170, April 2006
10. S. Fiori, *Statistical nonparametric bivariate isotonic regression by look-up-table-based neural networks*, Proceedings of the *2011 International Conference on Neural Information Processing* (ICONIP 2011, Shanghai (China), November 14-17, 2011), B.-L. Lu, L. Zhang and J. Kwok (Eds.), Part III, LNCS 7064, pp. 365 – 372, Springer, Heidelberg, 2011
11. A. Frank and A. Asuncion, *UCI Machine Learning Repository* [`http://archive.ics.uci.edu/ml`], University of California at Irvine, School of Information and Computer Science, 2010
12. H.R. Greve, N. Brandon Tuma and D. Strang, *Estimation of diffusion processes from incomplete data (A simulation study)*, Sociological Methods & Research, Vol. 29, No. 4, pp. 435 – 467, May 2001
13. W. Härdle, *Applied Nonparametric Regression*, Cambridge University Press, 1992
14. F. Katch and W. McArdle, *Nutrition, Weight Control, and Exercise*, Houghton Mifflin Co., Boston, 1977
15. R.J.A. Little and D.A. Rubin, *Statistical Analysis with Missing Data*, New York: John Wiley and Sons, 1987
16. D.G. Luchinsky, M.M. Millonas, V.N. Smelyanskiy, A. Pershakova, A. Stefanovska and P.V.E. McClintock, *Nonlinear statistical modeling and model discovery for cardiorespiratory data*, Physical Review E, Vol. 72, 021905, 2005
17. A.K. Nikoloulopoulos and D. Karlis, *Regression in a copula model for bivariate count data*, Journal of Applied Statistics, Vol. 37, No. 9, pp. 1555 – 1568, 2010
18. J.L. Peugh and C.K. Enders, *Missing data in educational research: A review of reporting practices and suggestions for improvement*, Review of Educational Research, Vol. 74, No. 4, pp. 525 – 55, 2004
19. M. Rosenblum, L. Cimponeriu and A. Pikovsky, *Coupled oscillators approach in analysis of bivariate data*, in B. Schelter, M. Winterhalder and J. Timmer (Eds.), "Handbook of Time Series Analysis", Wiley-VCH, pp. 159 – 180, 2006
20. G. Salanti, *The isotonic regression framework: Estimating and testing under order restrictions*, Ph.D. Dissertation, Fakultät für Matematik, Ludwig-Maximilians-Universität München, 2003
21. J.L. Schafer and J.W. Graham, *Missing data: Our view of the state of the art*, Psychological Methods, Vol. 7, No. 2, pp. 147 – 177, 2002
22. T. Schneider, *Analysis of incomplete climate data: Estimation of mean values and covariance matrices and imputation of missing Values*, Journal of Climate, Vol. 14, pp. 853 – 871, March 2001
23. *SOCR Data BMI Regression* [`http://wiki.stat.ucla.edu/socr/index.php/SOCR_Data_BMI_Regression`], University of California at Los Angeles, 2012
24. L. Torgo, *Regression DataSets* [`http://www.liaad.up.pt/~ltorgo/Regression/DataSets.html`], Artificial Intelligence and Computer Science Laboratory, University of Porto (Portugal), 2007
25. M.V. Velikova, *Monotone models for prediction in data mining*, Ph.D. Dissertation, Dutch Graduate School for Information and Knowledge Systems and Graduate School of the Faculty of Economics and Business Administration of Tilburg University, 2006
26. P.E. Verde, *Meta-analysis of diagnostic test data: A bivariate Bayesian modeling approach*, Statistical Medicine, Vol. 29, pp. 3088 – 3102, 2010
27. I.-C. Yeh, *Modeling of strength of high performance concrete using artificial neural networks*, Cement and Concrete Research, Vol. 28, No. 12, pp. 1797 – 1808, 1998