Reference Curves Estimation Using Conditional Quantile and Radial Basis Function Network with Mass Constraint

M.-Anas Knefati, Pierre Chauvet, Sylvie Nguyen, Bassam Daya

Reference Curves Estimation Using Conditional Quantile and Radial Basis Function Network with Mass Constraint

M.-Anas Knefati · Pierre E. Chauvet · Sylvie N'Guyen · Bassam Daya

Abstract This paper focuses on the improvement of reference curves building $Y = q(X)$ using a fast algorithm, robust against outliers. Our method consists in plugging a radial basis function neural network in the local linear quantile regression estimation proposed by Yu and Jones (QYJ). This neural network (QRBFc) is designed with a constructive algorithm, introducing a constraint on its integral over the input space. After explaining the different models and algorithms, we compare the QYJ and QRBFc estimators with the quantile regression neural network (QRNN) implemented by A. J. Cannon through simulations with a known underlying model using the R software. We observe that the QRBFc estimator reduces the mean absolute deviation error obtained with other estimators by about 16 %, the introduction of the constraint allowing to lower the number of neurons and therefore the computation time. Finally, using a database of 416 electroencephalograms recorded on preterm infants, we compare the QYJ, QRBFc and QRNN models for the building of brain maturation curves which are based on the dependence of the mean duration of interburst intervals (called IBIs—periods of quiescence between periods of normal electrical activity) with the age. The pathological infants represent 12 % of the total population. Denoting by S_A the set of individuals whose coordinates (age, mean IBI length) are above the 90 %-quantile curve, the QRBFc network

M.-A. Knefati (✉)
Département de Mathématiques, P2MI Université de Poitiers, 86962 Futuroscope-Chasseneuil, France
e-mail: maknefati@hotmail.com

P. E. Chauvet
LUNAM Université, Université Catholique de l'Ouest and LARIS EA 4094, 3 place André-Leroy, BP 10808, 49008 Angers, France
e-mail: pierre.chauvet@uco.fr

S. N'Guyen
LUNAM Université, Child Neurology Unit University Hospital and LARIS EA 4094, 4 rue Larrey, 49000 Angers, France
e-mail: sylvie.nguyenthe@gmail.com

B. Daya
IUT Saida, B.P 813, Saida, Lebanon

improves by 16.5 % the number of pathological infants in S_A compared to QYJ, when QRNN proved to be too unstable.

Keywords Quantile regression · Radial basis function neural network · Reference curve · Brain maturation

1 Motivation

The reference curves are part of the basic tools of the physician practices, in that they can decide on the vulnerability of an individual against a certain disease. Their construction, which is based on measurements made on a given set of individuals, shows a partition between normal and at risk individuals. It was during a project in pediatric neurology that we have developed a general method for construction of reference curves based on conditional quantiles and Radial basis function networks (RBFN). Because cerebral injury in newborns tends to be clinically silent, tools and techniques for neurological evaluation are essential. The electroencephalogram (EEG) is such a bedside, non-invasive and low cost technique. The EEG consists in recording the spontaneous electrical activity of the brain through several electrodes placed on the scalp. In preterm infants, the normal background EEG activity has the unique characteristic of being spontaneously discontinuous with periods of electrical activity alternating with periods of quiescence - called interburst intervals (IBIs). Figure 1 illustrates this phenomenon, which is normal in infancy. The diagnostic and prognostic value of neonatal EEG abnormalities in the preterm infant are well established and the IBI duration have been shown to be related to abnormal brain maturation in preterm infants [14]. In a recent study comparing maturation of cerebral activity and cortical folding, IBI duration was the only parameter significantly linked to morphologic brain maturation [1].

We have developed inside a dedicated web portal a Java application allowing the physician to extract automatically all the IBIs from an EEG. In this area, artificial neural networks are used generally for features extraction and classification, particularly for brain computer

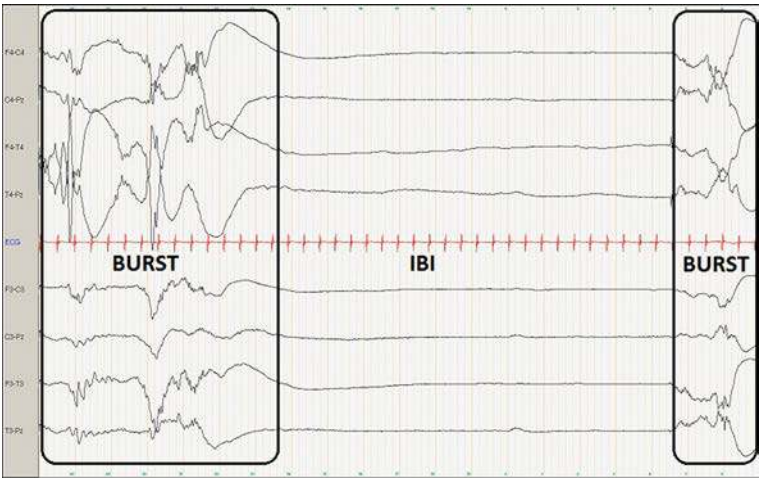

Fig. 1 Normal discontinuous EEG tracing in a preterm infant at 26 weeks of gestationnal age consisting in burst of electrical activities separated by interburst intervals (IBI)

interfaces (see for example [12,13]). In this work, the features (IBIs) are already detected and the objective is to build reference curves for brain maturation using these data, easy to recalculate from new sets of EEG and robust against artefacts and entry mistakes (e.g. age).

Our paper is structured as follows. First we start by reviewing the state of the art in Sect. 2. Then, after a few reminders about the basics of conditional quantile, we introduce in Sect. 3 the RBFN modeling of the conditional quantile curve; the learning algorithm is based on a constructive approach with a control of the integral of the model over its input domain (its mass), and the coupling with the conditional quantile estimator of Yu and Jones (QYJ), to obtain the so-called QRBFc model. Experiments are carried out in Sect. 4, where we compare on simulated data with known properties the QYJ, the QRBFc, the quantile regression neural network models (QRNN), and the possible improvements of the mass constraint. Finally we present and discuss in Sect. 5 the brain maturation reference curves obtained with the different models, based on the correlation between the mean length of interburst intervals in the newborn EEG and its age. We conclude this work in Sect. 6, providing our future work directions.

2 State of the Art

Conditional quantile regression has gained particular attention during the recent three decades due to their useful applications in various disciplines, such as finance, economics, medicine, and biology. See for exemple Fan and Gijbels [9] and Cai and Wang [3]. Of particular interest is the conditional median which is more explainable and more robust than the mean regression function for asymmetric conditional distributions. For $\tau \in (0, 1)$, the quantile regression function gives the τth quantile $q_\tau(x)$ in the conditional distribution of a response variable Y given $X = x$. It measures the effect of covariates not only in the center of the population, but also in the lower and upper tails. For x varying in a given real interval, $q_\tau(x)$ is a reference curve that predicts vulnerability of an individual with the probability τ or $1-\tau$ as its associated measured pair $(X = x, Y = y)$ is below or above the curve. A classical approach to evaluate the conditional quantile function from a sample (X_i, Y_i) on a discrete finite set is the local linear quantile regression method from Yu and Jones (denoted QYJ) in [16]. As its name suggests, $q_\tau(x)$ is approximated by a piecewise linear curve, resulting is a broken line that emphasizes local fidelity to the detriment of regularity, difficult to interpret as a reference chart.

Cannon [2] developed in R the QRNN package (for Quantile Regression Neural Network) based on the work from Taylor [11] in the field of time series forecasting. J. W. Taylor used a one hidden-layer feedforward neural network (FNN) to fit a non-local quantile model. FNN where successfully used in several domains like robotics and image processing (see for example [6,7]) as non linear parametric models. FNN realizes a global approximation of the data, and by varying the number of neurons it is possible to slide between high/low precision and smoothness of the quantile modelling. However, standard non-linear FNN with sigmoid transfer functions does not allow local settings on some parts of the data, and the learning process can be very slow. RBFN has the advantage to allow a faster constructive approach since each neuron puts emphasis on local data: neurons can be added iteratively, as in Chen et al. [4].

3 The RBFN Modeling of the Conditional Quantile

In this section, we first start with an overview of the conditional quantile. Next, the RBFN model is presented by mentioning its parameters which are the centers and the weights, and

we explain how to calculate them introducing constraint on the weights. Finally, we end this section by explaining our method to improve the estimation of the conditional quantile with a RBFN network, leading to the QRBFc method.

3.1 An Overview of the Conditional Quantile

Let (X, Y) be a bivariate random variable, $F(y|x)$ the conditional distribution function of Y given $X = x$, and $\tau \in (0, 1)$. The τth conditional quantile, noted by $q_\tau(x)$, is given by

$$q_\tau(x) = \inf\{y \in \mathbb{R} : F(y|x) \geq \tau\} \equiv F^{-1}(\tau|x) \tag{1}$$

or equivalently, by

$$q_\tau(x) = \arg\min_\theta E\{\rho_\tau(Y - \theta)|X = x\}, \tag{2}$$

where ρ_τ is the "check function" $\rho_\tau(u) = 0.5(|u| + (2\tau - 1)u)$.

The building of predictive intervals is an important application of quantile regression. Suppose the observations can be modelled as $Y_i = m(X_i) + \gamma\epsilon_i$ where $m(.)$ is an unknown function, γ is unknown reel parameter and the residual ϵ_i are uncorrelated random variables with zero mean and one variance. Then the quantile function can be written as

$$q_\tau(x)) = m(x) + \gamma F_\epsilon^{-1}(\tau), \tag{3}$$

where $F_\epsilon(.)$ denotes the distribution function of ϵ. A predictive interval is an interval that predicts, with certain coverage probability, the future value of the response variable Y for a given covariate $X = x$. The pairs of extreme conditional quantiles $q_{inf}(x)$ and $q_{sup}(x)$ map out a conditional prediction interval within which one expects the majority of individual points to lie. These "reference curves" are popular in medicine (see, e.g, Cole in [5]) and have provided a stimulus for much of the recent statistical work in this area.

Parametric techniques for estimating conditional quantile can be efficient if the underlying functions are correctly specified. But a misspecification may cause serious bias, and model constraint may distort the underlying distribution. Therefore, we will concentrate in this article on the nonparametric quantile regression, with the advantage that little or no restrictive prior information on functionals is needed.

Nonparametric estimation of conditional Quantile has been tackled by several authors, with direct and indirect methods. Direct methods use the "check" function ρ_τ, taking roots in definition (2). See, for instance, Fan et al. [8] and Yu and Jones [16] for a local linear estimate of conditional quantiles. Indirect methods, inspired from (1), are performed in two steps: the estimation of the conditional distribution is performed first, then the inverse of the obtained estimator is used to estimate the desired conditional quantile. Examples of indirect methods are provided by Cai and Wang [3]. Our work is based on the direct estimator proposed by Yu and Jones [16], because of its robustness in the y direction. The idea is to approximate the unknown τth conditional quantile $q_\tau(x)$ by the linear function

$$q_\tau(z) = q_\tau(x) + q'_\tau(x)(z - x) \equiv a + b(z - x)$$

for z in the neighborhood of x. This motivated us to define an estimator by setting $\hat{q}_\tau(x) = \hat{a}(x)$, with

$$\left(\hat{a}(x), \hat{b}(x)\right) = \arg\min_{(a,b)\in\mathbb{R}^2} \sum_{i=1}^{n} \rho_\tau\left(Y_i - a - b(X_i - x)\right) \times K\left(\frac{x - X_i}{h}\right), \tag{4}$$

where K is a kernel density function and h is the smoothing (also called bandwith) parameter, which is a nonnegative number controlling the size of the local neighborhood. Yu and Jones

have proposed a "rule-of-thumb" for selecting h:

$$h_\tau = h_{\text{mean}} \left(\frac{\tau(1-\tau)}{\phi(\Phi^{-1}(\tau))^2} \right)^{\frac{1}{5}}, \tag{5}$$

where h_{mean} is the optimal choice of h for regression mean estimation, ϕ and Φ are the standard normal density and distribution functions. They recommended to use the technique proposed by Ruppert et al. [10] for selecting h_{mean}.

The solution of the problem (4) can be obtained by using the iteratively reweighted least squares algorithm, see for exemple Yu [15]. We will denote by QYJ this algorithm (as well as the resulting quantile curves) used in conjunction with the rule-of-thumb from Yu and Jones in what follows.

3.2 The RBFN Neural Network Model

A RBFN can be viewed as a FNN with a specific structure that allows an easier constructive approach. It has one hidden layer in which each neuron computes its output using a radial basis function (RBF) receiving the inputs, and an output layer which builds a linear weighted sum of hidden neuron outputs and supplies the network's response.

An RBF function is a function $\phi : [0, \infty) \to \mathbb{R}$ that depends only on the distance from some point c, called a center, so that it has the form $\phi(\|x - c\|)$. In other words, a radial basis function is radially symmetric with respect to a given norm. We generally choose the Gaussian function $\phi(\|x - c\|) = e^{-\frac{\|x-c\|^2}{2}}$.

Because our goal is to produce reference curves, i.e. real functions, we use RBFN with only one output neuron. Its model is given by:

$$\hat{f}(x) = \sum_{j=1}^{N} \omega_j \phi(\|x - c_j\|), \tag{6}$$

where $x \in \mathbb{R}^d$, N is the number of hidden neurons, ω_j $(1 \leq j \leq N)$ are the weights of network output layer, c_j $(1 \leq j \leq N)$ is the center of the jth hidden neuron, $\phi(x)$ is a gaussian RBF function and $\|.\|$ denotes the distance function that is taken, in general, to be the Euclidean norm. In this work we use the Mahalanobis distance, because it takes into account the correlations inside the data set and is scale-invariant; in our case, it contributes to a better adjustment of the neurons width with the data. This distance is defined as

$$\|x - c\| = \sqrt{(x - c)^t \Sigma^{-1} (x - c)}, \tag{7}$$

where Σ is the covariance matrix of $(x_1, \ldots, x_n)$, and $x_i \in \mathbb{R}^d$ $(i = 1, \ldots, n)$ are the training data input.

We use in this work an iterative design of the network: neurons are added one at a time until the mean sum-squared error (MSE) falls beneath an error goal or a maximum number of neurons has been reached. The choice of the number N of neurons on the hidden layer of FNN and RBFN networks is crucial. A low number means a very poor performance, or fidelity, of the network. Instead, a large number of neurons will allow the network to fit exactly all the data (resulting in a very low MSE), including noise and biased observations: its regularity will be low. Because N is an integer defining the structure of the network, it cannot be adjusted like the synaptic weights. It exists for FNN some meta-heuristics (like genetic algorithms) to adjust N, but this is a slow process. Unlike FNN, RBFN authorizes

a faster constructive approach because of the locality of the radial basis transfer function in the hidden layer.

Let $x_i \in \mathbb{R}^d$ be the training data inputs and $y_i \in \mathbb{R}$ be the training data outputs for $i = 1,\ldots, n$. We now explain our RBFN learning algorithm.

3.2.1 Computation of the Centers

The algorithm for computation of the centers can be stated as follows:

- Provide the error threshold (err) and the maximum number of neurons $Nmax$ ($Nmax \leq n$).
- Initialize N to 0: the initial network does not have any radial functions.
- Store the output vector $(y_1, ..., y_n)$ in an other vector $(y_1^*, ..., y_n^*)$.
- While the total error $(\frac{1}{n}\sum_{i=1}^{n}(\hat{f}(x_i) - y_i)^2)$ is greater than err and $N < Nmax$ do:
  - calculate the network output $\hat{f}(x_i)$ using (6) for each input and the error $r_i = |\hat{f}(x_i) - y_i^*|$ for i =1, . . ., n;
  - find the input x_l that causes the greatest error, with $l = \arg\max_{i\in\{1,...,n\}} r_i$;
  - add a radial basis function whose center is this entry ($c_N = x_l,\ N = N + 1$);
  - recalculate the weight vector $W = (\omega_1, ..., \omega_n)$;
  - recalculate $\hat{f}(x_i)$, i =1,...,n;
  - initialize to 0 the output value that causes the greatest error, i.e. $y_l^* = 0$ and $\hat{f}(x_l) = 0$, so it will not be chosen again.

We determined here the number of hidden neurons N and the centers c_j. We explain in the next section the computation of the weights.

3.2.2 Computation of the Weights

Due to the fact that the mapping from hidden layer to output layer is linear, the weights computation becomes a linear problem. Minimization of the MSE yields to the well-known least square solution:

$$W = (\Phi^t\Phi)^{-1}\Phi^t Y, \tag{8}$$

where $Y = (y_1, ..., y_n)^t$ and

$$\Phi = \begin{pmatrix} \phi(x_1 - c_1) & \cdots & \phi(x_1 - c_N) \\ \vdots & \vdots & \cdots & \vdots \\ \phi(x_n - c_1) & \cdots & \phi(x_n - c_N) \end{pmatrix}$$

A first possibility for a robust calculation of the weights is to use Tikhonov regularization (also called ridge regression). Another suggestion to avoid possible numerical difficulties due to a singular or near-singular matrix, is to use the pseudo-inverse of a matrix which generalizes the inverse of a squared matrix. Any matrix can be factored using its singular value decomposition (SVD) from which its Moore-Penrose or generalized inverse can be obtained.

After testing these two methods we finally prefer the pseudo-inverse method, that provides the same results with far less computation time.

3.2.3 Computation of the Weights using a Mass Constraint

The objective in this section is to modify the previous algorithm to control the mass c of the RBFN model (its integral over the input domain). This is easier than controlling the energy of the RBFN (integral of the squared function), since we will be able to bring a linear constraint on the weights. In the case where the model is positive for all x, the mass of the model is equivalent to its norm L_1, and c appears as a global smoothing parameter. We calculate the weights of the model (6) such that it satisfies $\int_{\mathbb{R}^d} \hat{f}(x)dx = c$. We have

$$\int_{\mathbb{R}^d} \hat{f}(x)dx = \sum_{j=1}^{N} \alpha_j \omega_j,$$

where $\alpha_j = \int_{\mathbb{R}^d} \phi(\frac{\|x-c_{x_j}\|}{\sigma_j})dx$ for $j = 1, ..., N$. In our case, $\phi(\|x-c_{x_j}\|) = \exp(-\frac{\|x-c_{x_j}\|^2}{2})$ is a Gaussian function, then $\alpha_j = \alpha = (\sqrt{\pi})^d$ for all $j = 1, ..., N$. Therefore, for calculating the weights under the assumption that $\int_{\mathbb{R}^d} \hat{f}(x)dx = c$ we obtain the linear constraint: $\alpha \sum_{j=1}^{N} \omega_j = c$, with $\alpha = (\sqrt{\pi})^d$. To solve this least squares constrained problem, let introduce the Lagrange function given by

$$\begin{aligned} L(W, \lambda) &= \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \hat{f}(x_i) \right)^2 + \lambda \left(\alpha \sum_{j=1}^{N} \omega_j - c \right) \\ &= \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \sum_{j=0}^{N} \omega_j \phi_{ij} \right)^2 + \lambda \left(\alpha \sum_{j=1}^{N} \omega_j - c \right), \end{aligned}$$

where λ is the Lagrange multiplier, $\phi_{ij} = \phi(\frac{\|x_i - c_{x_j}\|}{2})$ and W is the vector of weights.

The necessary conditions for optimality are

$$\frac{\partial L(W, \lambda)}{\partial W} = 0 \Rightarrow -\sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{N} \phi_{ij} \omega_j \right) \phi_{ik} + d_0 \lambda = 0$$

and

$$\frac{\partial L(W, \lambda)}{\partial \lambda} = 0 \Rightarrow \alpha \sum_{j=1}^{N} \omega_j - c = 0.$$

Thus, we have to solve the $N + 1$ following equations with $N + 1$ variables:

$$\begin{cases} \sum_{j=1}^{N} \left(\sum_{i=1}^{n} \phi_{ij} \phi_{ik} \right) \omega_j + \alpha \lambda = \sum_{i=1}^{n} \phi_{ik} y_i, \quad k = 1, ..., N \\ \alpha \sum_{j=1}^{N} \omega_j = c \end{cases}$$

Like in paragraph III-B2, we use the SVD method to solve these equations.

3.3 Improvement of Conditional Quantile Estimation with a RBFN Network

We explain below our algorithm to improve the QYJ estimator using the RBFN model. Let $(X_1, Y_1), ..., (X_n, Y_n)$ be a random sample from (X, Y), and assume the homoscedastic model:

$$Y_i = m(X_i) + \gamma \epsilon_i, \quad i = 1, ..., n, \tag{9}$$

where $m(.)$ is an unknown function, γ unknown reel parameter and the residuals ϵ_i are uncorrelated random variables of known distribution with zero mean and one variance. Therefore the conditional quantile function is defined by:

$$q_\tau(x) = m(x) + \gamma F_\epsilon^{-1}(\tau). \tag{10}$$

The proposed algorithm is explained as follow :

- Estimate first the conditional quantile function $q_\tau(x)$ using the QYJ estimator with the Yu and Jone method to select its smoothing parameter; we get $\hat{q}_\tau^{QYJ}(x)$.
- From (10), we have

$$\hat{m}^{QYJ}(x_k) = \hat{q}_\tau^{QYJ}(x_k) - \hat{\gamma} F_\epsilon^{-1}(\tau),$$

  where $\hat{\gamma}^2$ is the variance of the data outputs, and x_k, $k = 1, ..., n_0$ are either regularly spaced values along x-axis or the values $(X_1, ..., X_n)$ taken in the random sample itself.
- Use the RBFN model with entries x_k and $y_k = \hat{m}^{QYJ}(x_k)$ to approximate the unknown function $m(x)$ to obtain $\hat{m}^{QRBF}(x)$.
- Plug-in $\hat{m}^{QRBF}(x)$ and $\hat{\gamma}$ in (10) to get the improved RBFN conditional quantile estimator:

$$\hat{q}_\tau^{QRBF}(x) = \hat{m}^{QRBF}(x) + \hat{\gamma} F_\epsilon^{-1}(\tau) \tag{11}$$

  for all x in the input space.

We denote by QRBF (when we do not use the mass constraint) and QRBFc (when using the mass constraint) this algorithm and the resulting quantile curves in what follows.

4 Simulations with a Known Underlying Model

The following applications have been made using the R software, with the package "quantreg" from Koenker [17] for our implementation of the QYJ algorithm and the QRNN R-package implemented by Cannon [18].

Consider the model

$$Y_i = \exp(-X_i) + \exp\left(-4(X_i - 1)^2\right) + \epsilon_i$$

for $i = 1, ..., n$, where $\{\epsilon_i\}$ are independent and identically $\mathcal{N}(0, 1)$ random variables, and $\{X_i\}$ are exponential random variables with mean 1 independent from $\{\epsilon_i\}$. Then the true conditional quantile function is given by

$$q_\tau(x) = \exp(-x) + \exp\left(-4(x - 1)^2\right) + F_\epsilon^{-1}(\tau),$$

where F_ϵ is the cumulative $\mathcal{N}(0, 1)$ distribution function.

First, we compare for different sample sizes n and different quantile values τ the performance of the QRBF estimator without constraint with the QYJ estimator and the QRNN estimator with 5 neurons on the hidden layer (after several tries, 5 neurons appear to be a good compromise between the error goal and the calculation speed). For this purpose, we ran 100 replications per experiment for several combination of the parameters values n and τ. We also demonstrate the performance of the estimators in terms of the mean absolute deviation error (MADE):

$$MADE = \frac{1}{n_0} \sum_{j=1}^{n_0} |\hat{q}_\tau(x_j) - q_\tau(x_j)|,$$

where x_j, $j = 1, ..., n_0$ are regularly spaced values between 0 and 3.5.

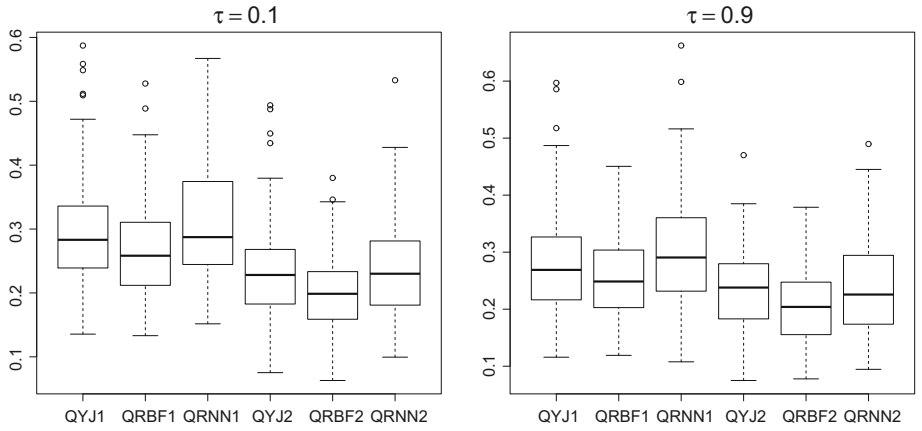

Fig. 2 Mean absolute deviation error boxplots obtained for sample sizes $n = 300$ and $n = 500$, with the two quantile values $\tau = 0.1$ and $\tau = 0.9$. QYJ_i denotes the Yu and Jones QYJ estimator, $QRBF_i$ its QRBF improvement, and $QRNN_i$ denotes the QRNN estimator, where i equals 1 for size 300 and 2 for size 500

Table 1 Median for 100 values of N, N_c and $MADE$ with mass constraint c

Sample size	$\tau = 0.9$	$\bar{N}$	$\bar{N}_1$	$\bar{N}_5$	$\bar{N}_{10}$	$\bar{N}_{20}$	$\bar{N}_{50}$	$\bar{N}_{100}$	$\bar{N}_{500}$	$\bar{N}_{1000}$	$\bar{N}_{5000}$
$n = 300$	Neurons	82	80	78	58	56	59	76	100	100	100
	MADE	0.242	0.238	0.24	0.24	0.24	0.24	0.24	0.284	0.329	2.962
$n = 500$	Neurons	102	87	92	62	51	59	88	166	166	166
	MADE	0.197	0.196	0.199	0.198	0.198	0.198	0.199	0.218	0.254	0.757

$\bar{N}$ denotes the median number of neurons on the RBFN hidden layer with no mass constraint and $\bar{N}_c$ this same median number using a mass constraint equals to c

We can see on Fig. 2 the boxplots of the 100 MADE values for the QYJ, QRBF and QRNN estimators, with $\tau = 0.1$ and $\tau = 0.9$. The QRBF estimator is better than its Yu and Jones counterpart and the QRNN estimator : (i) the MADE median of QRBF is lower than QYJ and QRNN; (ii) the MADE spread of QYJ and QRBF is smaller than QRNN. The computation time for the QRNN model is an average of 3 times that of QRBF and QYJ.

We then studied the impact of the mass constraint on the MADE median and on the number N of neurons in the hidden layer. We denote N_c the number of hidden neurons obtained by the constructive learning method under the constraint $\int_{\mathbb{R}} \hat{f}(x)dx = c$. We ran 100 replications for each simulation with different values of c. We found that the median of N_c values is smaller than the median of N values for all values of τ and some values of c.

In the Table 1, we see the median of $MADE$, N and N_c obtained for several values of c, in the case of $\tau = 0.9$ and sample sizes equal to 300 and 500, after 100 simulations. It appears clearly that controlling the RBFN mass (i.e calculating weights under constraint) uses, for some values of c, less neurons and the MADE obtained is nearly the same one obtained from RBFN without constraint. In our example, we can see from the Table 1 that $c = 10$ or $c = 20$ could be a suitable choice, but we think, according to this table, that an optimal choice should exist. This is the center of our actuel research.

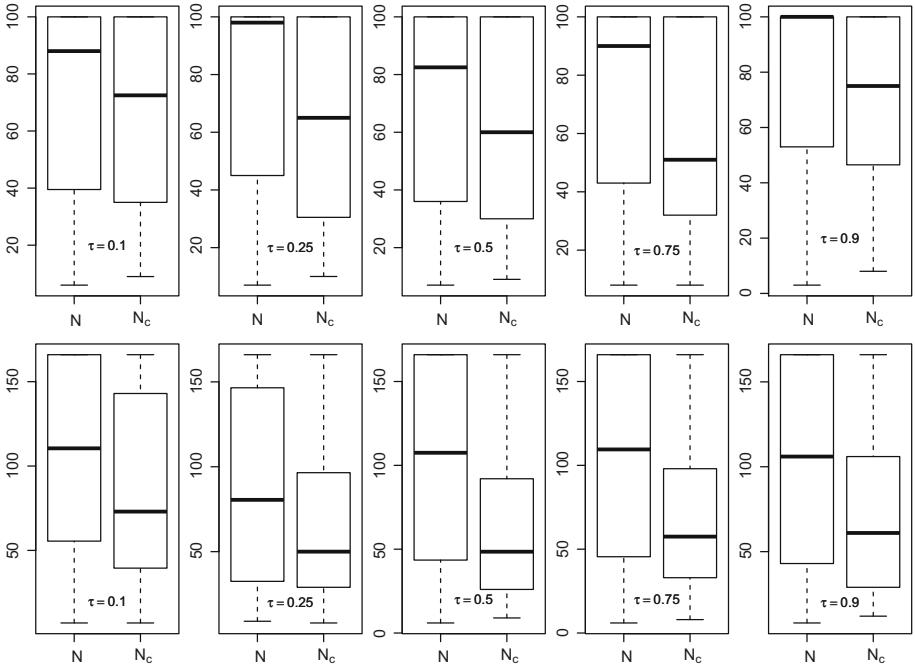

Fig. 3 Boxplot of the number of hidden neurons in the QRBF estimator for both cases: computing the weights without constraint (N) and with constraint (N_c with $c = 10$). First row: sample size equals to 300; second row: sample size equals to 500

Table 2 cfor the different estimators; *QYJ* denotes the classical Yu and Jones linear quantile estimator, *QRBF* its improvement using the RBFN without mass constraint, *QRBFc* the RBFN with mass constraint, and *QRNN* denotes the quantile regression feedforward neural network from A. J. Cannon ($\tau = 0.9$ and sample size: $n = 500$)

	QRNN	QYJ	QRBF	QRBF_{10}
Error	0.237	0.237	0.198	0.198
Time	2.69	0.91	1.385	1.01

The Fig. 3 illustrates, also, this fact for five quantile values, two sample sizes and three values of c. We think that the mass constraint act like a smoothing or regularization parameter and it would reduce the variance of the RBFN estimator by keeping, nearly, the same MADE error obtained by RBFN without constraint.

In Table 2, we put the median of MADE and computation time for every estimator, with the sample size equals to 500 and the quantile order equals to 0.9. According to this table we can see that the QRBF estimator (with or without mass constraint) reduces in this example the relative error by about 16 % for QRNN and for QYJ. Moreover, introducing the mass constraint in the QRBF estimator improves its computation time by reducing significantly its number of neurons. For this reason, we recommend to calculate the RBFN weights using the mass constraint as explained in paragraph (3.2.3), as we reduce the error against the QYJ estimator while we improve the speed against other neural network estimators.

5 Application to the Building of Brain Maturation Reference Curves

The next results were obtained from the automatic analysis of 416 EEG recorded between years 2003 and 2004 on newborn infants, with relative informations such that the infant outcome, age (from birthday and gestational), other neurological evaluations, etc. Moreover, each EEG in this database was visually analyzed and categorized as "artefact", "normal", "doubtful" or "pathological". A specific algorithm was built during this research to detect IBIs using an approach mimicking visual analysis. The algorithm is designed to study on each channel the variation of the signal estimated variance between contiguous short time-windows. An IBI on one channel is detected if this variation is lower than a given threshold and EEG (global) IBIs are finally computed as the intersection of channel's IBIs. More precisely, our algorithm is described by the following sequence of operations:

- each channel is filtered at 50Hz (that corresponds to frequency of electricity supply within European Union) with a second order Butterworth designed IIR filter;
- each channel is smoothed with a moving window using the simple average;
- each channel is processed to produce an estimate of the standard deviation on overlapped windows, a standard deviation series.
- this is a two steps operation; (i) for each resulting standard deviation series, if the difference between two successive values is lower than a given V_T threshold (in μV) the corresponding time intervals are aggregated and an IBI is defined by an aggregated time interval if it lasts at least m_1 seconds; (iii) finally, IBIs separated by less than m_2 seconds are regrouped.
- the intersection of IBIs between all the EEG channels is computed, and only the IBIs of a length greater than m_3 seconds are retained.

By comparisons between the IBIs marked by the specialist and the IBIs detected by this algorithm, we set the parameter values to $V_T = 15\ \mu V$, $m_1 = 1$ s, $m_2 = 0.5$ s and $m_3 = 1$ s.

For each EEG the mean IBI length is computed and plotted versus the gestational age of the infant. From the couple (age , mean IBI length) we have traced the reference curves using the QRNN, the QYJ and the QRBFc estimators. Figures 4 and 5 shows the results obtained with two levels of the quantile (0.1 and 0.9). It clearly appears that the improved

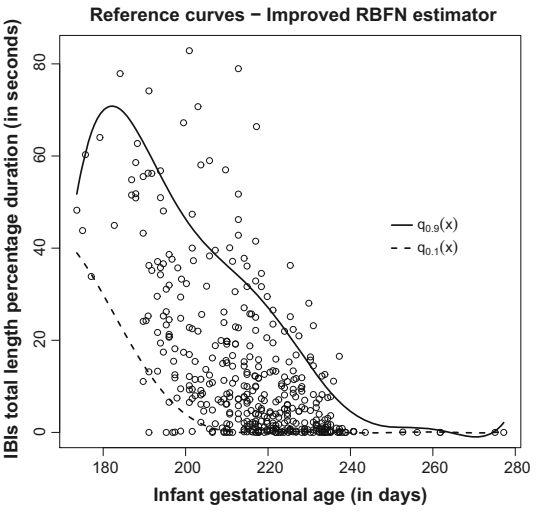

Fig. 4 Brain maturation reference curve for age in days (quantile levels $\tau = 0.1$ and $\tau = 0.9$) obtained with the improved QRBF_{10} estimator using a regular grid with 100 points and the automatic IBI analysis of 416 EEG recorded in 2003–2004

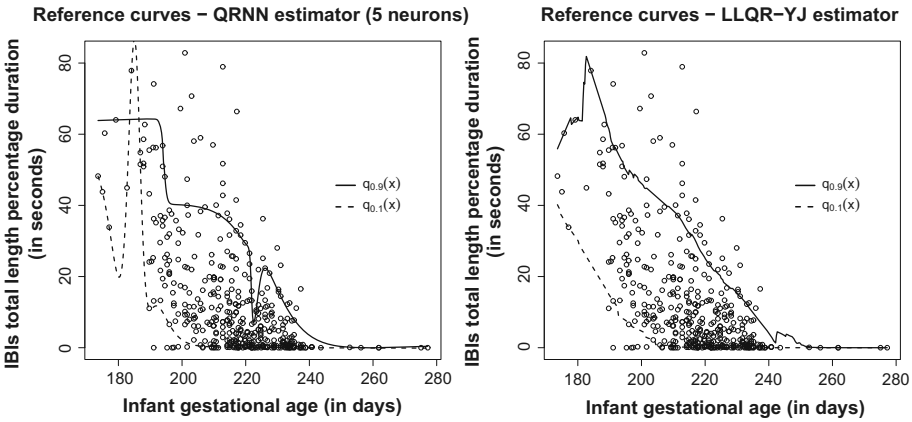

Fig. 5 Brain maturation reference curves obtained with the QRNN (*left*) and the QYJ (*right*) estimators

Table 3 Mean IBI's length and age in each EEG set; the $Total$ row is the initial set of 416 EEG

EEG set	Mean IBI length (s)	Mean age (days)
Total	13.3369	216.0832
QYJ		
S_A	39.4740	202.9307
S_{IN}	9.1188	219.2450
S_U	2.5125	218.0835
QRBF		
S_A	40.4922	216.1256
S_{IN}	11.9003	215.1669
S_U	1.1202	222.3292

QRBFc estimator (Fig. 4) produces smoother curves, easier to use, than the QYJ estimator. The QRNN estimator produces very different cuves from one try to another as the learning algorithm is stochastic. The QRNN curves are generally less smooth, with sometimes a totally flat curve for $\tau = 0.1$, and other times an inversion between the lower and the higher curves.

The two curves define three areas allowing one to build three disjoint EEG sets denoted S_A, S_{IN} and S_U, by taking EEG whose coordinates (age, mean IBI length) are respectively above the higher curve, between the two curves inclusively and under the lower curve. The union of these three sets is equal the initial total set, and the population at risk is defined by the set S_A. To compare these sets we have first calculated for each one the mean of the EEG mean IBI lengths and the mean of the ages (from birthday): the results are provided in Table 3. One can see that the sets S_A defined by QYJ and QRBFc have a longer mean IBI length by about three times that of the total set. This is perfectly consistent with the following clinical outcome: the higher the mean IBI length, the greater the risk of abnormal maturation is important. The mean age of the population in S_A is about 203 days for QYJ and 216 days for QRBFc. Clinical studies have shown that the older the age, the less the IBIs must be long: in an individual with normal cerebral development, the IBIs should disappear. Therefore the QRBFc best defines the population at risk: its mean age is slightly higher than the mean age of the total population, when QYJ provides a younger population for which it is less unusual to have longer IBIs. We have computed too the percentages of EEG for each set in each

Table 4 Percentages of EEG in categories "artefact", "normal", "doubtful" or "pathological" for each set; the $Total$ row is the initial set of 416 EEG

EEG set	Artefact	Normal	Doubtful	Pathological
Total	1.9231	65.8654	20.1923	12.0192
QYJ				
S_A	0	33.3333	29.3333	37.3333
S_{IN}	6.3291	84.8101	5.0633	3.7975
S_U	1.1450	69.4656	22.1374	7.2519
QRBF				
S_A	0	25.6410	20.5128	53.8462
S_{IN}	10.4167	75.0000	6.2500	8.3333
S_U	0.9119	69.3009	22.1884	7.5988

category normal, doubtful and pathological (results in Table 4). The pathological individuals represent 12 % of the total population, 37 % in the S_A set defined by QYJ and 54 % in the S_A set defined by QRBFc. This confirms the improved ability of the QRBFc model to build the population at risk.

6 Conclusion and Perspectives

Our approach to build reference curves is based on the nonparametric linear quantile regression from Yu and Jones. Its advantages are its robustness to outliers and measurement errors. Moreover, the fact that it is non-parametric allows us to construct the reference curve without a priori assumption. We improved ease of use and performance by reshaping the QYJ estimator with a RBFN network. Indeed, the network is defined for all values of x in the input space: whatever the input x presented by the user, the network produces a prediction without going through all the steps of QYJ algorithm. Somehow, we constructed a parametric model based on a non-parametric approach.

We have shown with simulations that the median (and to a lesser extent the spread) of the mean absolute deviation error obtained for 100 replications with the QRBF estimator is less than that obtained with the QYJ and QRNN estimators for different values of the quantile level τ. We have also find in several simulations that the QRNN estimator is not stable specially when n is small. We introduced in paragraph II-B3 a constraint on the "estimator's mass", i.e. its integral value over the input space, that can act like a smoothing or regularization parameter if the estimator is a positive function. This mass constraint becomes a linear constraint on the weights that is easily integrated in the algorithm of the RBFN construction, with very low additional computational cost. This constraint significantly reduces the number of hidden layer's neurons for a given range of the values of the constraint c: the computation time for plug-in a RBFN neural network in the classical quantile regression approach becomes marginal when the efficiency in term of error gains ground. Further work will be carried out to improve the qualities of the QRBFc estimator by automating the settings of c.

Concerning the IBI curves reflecting the brain maturation, further studies are in progress in order to establish the relationships between these values and the neurological outcome of the infants. We ought to propose reference curves that could help the medical physician in the assessment of the EEG and the neurological status of the preterm infants.

Acknowledgments The authors wish to express their appreciation to the three referees for their constructive and helpful comments and suggestions. The writing of this manuscript was supported by the Agence Nationale de la Recherche grant "BB EEG Platform".

References

1. Biagioni E, Frisone MF, Laroche S, Kapetanakis BA, Ricci D, Adeyi-Obe M et al (2007) Maturation of cerebral electrical activity and development of cortical folding in young very preterm infants Clin Neurophysiol. J PubMed 118(1):53–59 M
2. Cannon AJ (2011) Quantile regression neural networks: implementation in R and application to precipitation downscaling. J Comput Geosci 37(4):1277–1284
3. Cai Z, Wang X (2008) Nonparametric methods for estimating conditional VaR and expected shortfall. J Econom 174:120–130
4. Chen S, Cowan CFN, Grant PM (1991) Orthogonal least squares learning algorithm for radial basis function networks. IEEE Trans Neural Netw 2(2):302–309
5. Cole TJ (1988) Fitting smoothed centile curves to reference data. J R Stat Soc Ser A 151(3):385–418
6. Daya B, Ismail A (2006) A neural control system of a two joints robot for visual conferencing. Neural Process Lett 23:289–303
7. Daya B (1999) A multilayer perceptrons model for the stability of a bipedal robot. Neural Process Lett 9:221–227
8. Fan J, Hu T-C, Troung YK (1994) Robust Non-parametric function estimation. Scand J Stat 21:433–446
9. Fan J, Yao Q, Tong H (1996) Estimation of conditional densities and sensitivity mesures in nonlinear dynamic systems. j biomet 83:189–206
10. Ruppert D, Seather ST, Wand MP (1995) An effective bandwidth selector for local least squares regression. J Am Stat Assoc 90:1257–1270
11. Taylor JW (2000) A quantile regression neural network approach to estimating the conditional density of multiperiod returns. J Forecast 19(4):299–311
12. Mateo J, Torres AM, Garca MA (2014) Dynamic fuzzy neural network based learning algorithms for ocular artefact reduction in EEG recordings. Neural Process Lett 39:45–67
13. Wang H (2012) Harmonic mean of Kullback–Leibler divergences for optimizing multi-class EEG spatio-temporal filters. Neural Process Lett 36:161–171
14. Watanabe K, Hayakawa F, Okumura A (1999) eonatal EEG: a powerful tool in the assessment of brain damage in preterm infants. J Brain Dev 21(6):361–372
15. Yu K (1997) Smooth regression quantile estimation. unpublished Ph.D. thesis, The open university
16. Yu K, Jones MC (1998) Local Linear quantile rregression. J Am Stat Assoc 93:228–237
17. Koenker R (2013) Quantreg R package, University of Illinois. http://cran.r-project.org/web/packages/quantreg/index.html
18. Cannon AJ (2011) QRNN R package, University of British Columbia. http://cran.r-project.org/web/packages/qrnn/