

# A computational approach to nonparametric regression: bootstrapping CMARS method

Ceyda Yazıcı<sup>1</sup> · Fatma Yerlikaya-Özkurt<sup>2</sup> · İnci Batmaz<sup>1</sup>

Received: 31 December 2013 / Accepted: 7 May 2015 / Published online: 20 May 2015  
© The Author(s) 2015

**Abstract** *Bootstrapping* is a computer-intensive statistical method which treats the data set as a population and draws samples from it with replacement. This resampling method has wide application areas especially in mathematically intractable problems. In this study, it is used to obtain the empirical distributions of the parameters to determine whether they are statistically significant or not in a special case of nonparametric regression, conic multivariate adaptive regression splines (CMARS), a statistical machine learning algorithm. CMARS is the modified version of the well-known nonparametric regression model, multivariate adaptive regression splines (MARS), which uses conic quadratic optimization. CMARS is at least as complex as MARS even though it performs better with respect to several criteria. To achieve a better performance of CMARS with a less complex model, three different bootstrapping regression methods, namely, random-X, fixed-X and wild bootstrap are applied on four data sets with different size and scale. Then, the performances of the models are compared using various criteria including accuracy, precision, complexity, stability, robustness and computational efficiency. The results imply that bootstrap methods give more precise parameter estimates although they are computationally inefficient and that among all, random-X resampling produces better models, particularly for medium size and scale data sets.

**Keywords** Bootstrapping regression · Conic multivariate adaptive regression splines · Fixed-X resampling · Random-X resampling · Wild bootstrap · Machine learning

---

Editors: Vadim Strijov, Richard Weber, and Süreyya Ozogur Akyüz.

✉ İnci Batmaz  
ibatmaz@metu.edu.tr

Ceyda Yazıcı  
cyazici@metu.edu.tr

Fatma Yerlikaya-Özkurt  
fatmayerlikaya@gmail.com

<sup>1</sup> Department of Statistics, Middle East Technical University, Ankara, Turkey

<sup>2</sup> Institute of Applied Mathematics, Middle East Technical University, Ankara, Turkey

## 1 Introduction

Models are simple forms of research phenomena that relate ideas and conclusions (Hjorth 1994). In statistics, formulating a model to answer a scientific question is usually the first step taken in an empirical study. Parametric and nonparametric models are two major approaches to statistical modeling in machine learning. Parametric models depend on certain distributional assumptions; if those assumptions hold, they give reliable inferences. Otherwise, nonparametric modeling is recommended.

Multivariate adaptive regression splines (MARS) is a nonparametric regression method (Friedman 1991; Hastie et al. 2001), and widely used in biology, finance and engineering. This method is proved to be useful for handling complex data, which has a nonlinear relationship among numerous variables. MARS builds models by running forward selection and backward elimination algorithms in succession. In the forward algorithm, deliberately, as large model as possible is fitted. Later, in the backward elimination, terms which do not contribute to the model are omitted.

In recent years, a lot of studies have been conducted involving MARS modeling. To exemplify, Denison et al. (1998) provide a Bayesian algorithm for MARS. Moreover, Holmes and Denison (2003) used Bayesian MARS for classification. York et al. (2006) compare the power of the least squares (LS) fitting to that of the MARS with polynomials. Kriner (2007) uses this model for survival analysis. Deconinck et al. (2008) show that MARS is better for fitting nonlinearities, more robust to small changes in data and easy to interpret compared to boosted regression trees. Zakeri et al. (2010) predict the energy expenditure for the first time in this research area by using MARS. Lin et al. (2011) apply MARS to time series data. Lee and Wu (2012) study the MARS applications, where it is used as a metamodel in the global sensitivity analysis of ordinary differential equation models. Ghasemi and Zolfonoun (2013) propose a new approach for MARS using principal component analysis for selection of inputs and apply it to determine the chemical amounts.

Depending on the power of MARS method in modeling high-dimensional and voluminous data, several studies have been conducted to improve its capability. One of them is Conic MARS (CMARS) developed as an alternative to backward elimination algorithm by using conic quadratic programming (CQP) (Yerlikaya 2008), and it is improved to model nonlinearities better (Batmaz et al. 2010). Taylan et al. (2010) compare the performances of MARS and CMARS in classification. Later, its performance is rigorously evaluated and compared with that of MARS using various real-life and simulated data sets with different features (Weber et al. 2012). The results show that CMARS is superior to MARS in terms of accuracy, robustness and stability under different data features. Moreover, it performs better than MARS on noisy data. Nevertheless, CMARS produces models which are at least as complex as MARS.

CMARS has also been compared with several other methods such as classification and regression trees (CART) (Sezgin-Alp et al. 2011), infinite kernel learning (IKL) (Çelik 2010), and generalized additive models (GAMs) with CQP (Sezgin-Alp et al. 2011) for classification, and multiple linear regression (MLR) (Yerlikaya-Özkurt et al. 2014) and dynamic regression model (Özmen et al. 2011) for prediction. These studies reveal that CMARS method performs as good as or even better than the others considered. For detailed findings one can refer to a comprehensive review of CMARS (Yerlikaya-Özkurt et al. 2014).

A quick look into literature demonstrates that almost a decade has been devoted to the development and improvement of the CMARS method. All these studies lead to a powerful alternative to MARS with respect to several criteria including accuracy. Nevertheless, as

stated above, the complexity of CMARS models does not compete with that of MARS. Therefore, in this study, we aim at reducing the CMARS models' complexity. In the usual parametric modeling, the statistical significance of the model parameters can be investigated by testing hypothesis or by constructing confidence intervals (CIs). Because there are no parametric assumptions regarding CMARS models, the methods of computational statistics (CS) may be a plausible approach to take here.

CS is relatively a newer branch of statistics which develops methodologies that intensively use computers (Wegman 1988). Some examples include bootstrap, CART, GAMs, nonparametric regression methods (Efron and Tibshirani 1991) and visualization techniques like parallel coordinates, projection pursuits, and so on (Martinez and Martinez 2002). Advances in the computer science make all these methods feasible and popular especially after 1980s. In this study, the mathematical intractability appears to be the lack of distribution fitting to CMARS parameters. An empirical cumulative distribution function (CDF) is tried to be fitted to each parameter by a CS method, called bootstrap resampling. In this approach, samples are drawn from the original samples with replacement (Hjorth 1994).

There are some applications of this technique to estimate the significance of parameters in a model. Efron (1988) implements bootstrap to least absolute deviation (LAD). Efron and Tibshirani (1993) employ resampling residuals to a model based on least median of squares (LMS). Montgomery et al. (2006) apply bootstrapping residuals to a nonlinear regression method, called Michaelis-Menten. Fox (2002) uses random-X and fixed-X resampling for a robust regression technique which uses *M-estimator* with the Huber weight function. Also, Salibian-Barrera and Zamar (2002) apply bootstrapping to robust regression. Flachaire (2005) compares the pairs bootstrap with wild bootstrap for heteroscedastic models. Austin (2008) uses bootstrap and with backward elimination which results in improvement of estimation. Chernick (2008) uses vector resampling for a kind of nonlinear model used in aerospace engineering. Yetere-Kurşun and Batmaz (2010) compare regression methods employing different bootstrapping techniques.

In this study, to reduce the complexity of CMARS models without degrading its performance with respect to other measures, a new algorithm, called Bootstrapping CMARS (BCMARS), is developed by using three different bootstrapping regression methods, namely fixed-X, random-X and wild bootstrap. Next, these algorithms are run on four data sets chosen with respect to different sample sizes and scales. Then, the performances of the models developed are compared according to the complexity, stability, accuracy, precision, robustness and computational efficiency.

This paper is organized as follows. In Sect. 2, MARS, CMARS, bootstrap regression and validation methods are described. The proposed approach, BCMARS, is explained in Sect. 3. In Sect. 4, applications and findings are presented. Results are discussed in Sect. 5. In the last section, conclusions and further studies are stated.

## 2 Methods

### 2.1 MARS

MARS, developed by Friedman (1991), is a nonparametric regression model where there is no specific assumption regarding the relationship between the dependent and independent variables; it constructs one of the best models which approximates the nonlinearity and handles the high dimensionality in data. MARS models are built in two steps: forward and

backward. In the forward step, the largest possible model is obtained. However, this large model leads to over fitting. Thus, a backward step is required to remove terms that do not contribute significantly to the model.

In general, a nonparametric regression model is defined as

$$y = f(\boldsymbol{\theta}, \mathbf{x}) + \varepsilon, \tag{1}$$

where  $\boldsymbol{\theta}$  represents the unknown parameter vector;  $\mathbf{x}$  shows the independent variable vector;  $\varepsilon$  is the error term. In the model,  $f(\boldsymbol{\theta}, \mathbf{x})$  is the unknown form of the relation function. In MARS model, instead of original predictor variables, a special form of them is used to construct models. These are called basis functions (BFs) and represented with the following equations

$$(x - t)_+ = \begin{cases} x - t, & \text{if } x > t, \\ 0, & \text{otherwise.} \end{cases} \quad (t - x)_+ = \begin{cases} t - x, & \text{if } x < t, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Here,  $t \in \{x_{1,j}, x_{2,j}, \dots, x_{N,j}\}$  ( $j = 1, 2, \dots, p$ ) is called as a knot value and these two BFs are the reflected pairs of each other. Note that  $(\cdot)_+$  denotes the positive part of the component in (2). The multivariate spline BFs take the following form to employ the BF that is tensor products of univariate spline functions

$$B_m(\mathbf{x}^m) = \prod_{k=1}^{K_m} [s_{km}(x_{km} - t_{km})]_+, \tag{3}$$

where  $K_m$  represents the number of truncated functions in the  $m^{th}$  BF;  $x_{km}$  shows the input variable corresponding to the  $k^{th}$  truncated linear function in the  $m^{th}$  BF;  $t_{km}$  is the corresponding knot value. Note that  $s_{km}$  takes the value of 1 or -1. As a result, the MARS model is defined as

$$y = f(\boldsymbol{\theta}, \mathbf{x}) + \varepsilon = \theta_0 + \sum_{m=1}^M \theta_m B_m(\mathbf{x}^m) + \varepsilon, \tag{4}$$

where each  $B_m$  is the  $m^{th}$  BF, and  $M$  represents the number of BFs in the final model. Given a choice for the  $B_m$ , the coefficients for the parameters ( $\theta_m$ ) are estimated by minimizing the residual sum of squares (RSS) with the same method used in the MLR, called LS. The important point here is to determine the  $B_m(\mathbf{x}^m)$ . For this purpose,  $B_0(\mathbf{x}^0) = 1$  is taken as the starting function, and then, by considering all elements in the set of BFs as candidate functions, the one which causes the most amount of reduction in the RSS is included in the model. When the maximum number of terms (determined by the user) is reached, the forward step ends. After obtaining the largest model, backward step starts to prevent overfitting. In this step, a term in the model whose deletion causes the least amount of reduction in RSS is deleted first. This procedure leads to the best estimated model function,  $\hat{f}_M$ , for each size (number of terms)  $M$ . Cross validation (CV) is a possible technique for finding the optimal value for  $M$ . However, *generalized cross validation (GCV)* is preferred by [Friedman \(1991\)](#) in his original work since it reduces the computational burden; it is defined as

$$GCV = \frac{1}{N} \frac{\sum_{i=1}^N (y_i - \hat{f}_M(\boldsymbol{\theta}, \mathbf{x}_i))^2}{(1 - C(M)/N)^2}, \tag{5}$$

here, the number of observations (i.e. number of data points) is represented by  $N$ ; the numerator of (5) is the usual RSS;  $C(M)$  in denominator represents the cost penalty measure of a

model with  $M$  BFs. The MARS model is constructed, when the minimum value of the GCV is reached.

### 2.2 CMARS

CMARS is an alternative to the backward step of MARS developed by [Weber et al. \(2012\)](#), [Yerlikaya \(2008\)](#). It uses BFs generated by the forward step of MARS, and applies CQP to prevent over fitting. For this purpose, penalized RSS (PRSS) is constructed as the sum of two components: RSS and the complexity measure, as follows

$$PRSS := \sum_{i=1}^N (y_i - f(\boldsymbol{\theta}, \tilde{\mathbf{x}}_i))^2 + \sum_{m=1}^{M_{\max}} \lambda_m \sum_{|\boldsymbol{\alpha}|=1}^2 \sum_{r < s} \int_{\boldsymbol{\alpha}=(\alpha_1, \alpha_2)^T, r, s \in V(m)} \theta_m^2 [D_{r,s}^\alpha B_m(\mathbf{z}^m)]^2 d\mathbf{z}^m, \tag{6}$$

where  $(\tilde{\mathbf{x}}_i, y_i)$  ( $i = 1, 2, \dots, N$ ) represents our data points with  $p$ -dimensional predictor variable vector  $\tilde{\mathbf{x}}_i = (\tilde{x}_{i1}, \tilde{x}_{i2}, \dots, \tilde{x}_{ip})^T$  ( $i = 1, 2, \dots, N$ ) and  $N$  response values  $(y_1, y_2, \dots, y_N)$ . Furthermore,  $M_{\max}$  is the number of BFs reached at the end of the forward step of MARS,  $V(m) = \{\kappa_j^m | j = 1, 2, \dots, K_m\}$  is the variable set associated with  $m^{th}$  BF.  $\mathbf{z}^m = (z_{m1}, z_{m2}, \dots, z_{m\kappa_m})^T$  represent variables that contribute to the  $m^{th}$  BF. The  $\lambda_m$  ( $m = 1, 2, \dots, M_{\max}$ ) values are always nonnegative and used as *penalty parameters*. Moreover, in Eq. (6),  $D_{r,s}^\alpha B_m(\mathbf{z}^m) = \frac{\partial^{|\alpha|} B_m}{\partial \alpha_1^{z_{m1}} \partial \alpha_2^{z_{m2}}}(\mathbf{z}^m)$  is the partial derivative for the  $m^{th}$  BF where  $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$ ,  $|\boldsymbol{\alpha}| = \alpha_1 + \alpha_2$ , and  $\alpha_1, \alpha_2 \in \{0, 1\}$ .

Here, the optimization approach adopted takes both the accuracy and complexity into account. While accuracy is guaranteed by the RSS, complexity is measured by the second component of PRSS in (6). The tradeoff between these two criteria are represented by the penalty parameters  $\lambda_m$  ( $m = 1, 2, \dots, M_{\max}$ ).

Riemann sums are used to approximate the discretized form of the integrals in (6) as follows ([Weber et al. 2012](#); [Yerlikaya 2008](#))

$$\begin{aligned} & \int_{\mathcal{Q}^m} \theta_m^2 [D_{r,s}^\alpha B_m(\mathbf{z}^m)]^2 d\mathbf{z}^m \\ & \approx \sum_{(\sigma^j)_{j \in \{1, 2, \dots, p\}} \in \{0, 1, 2, \dots, N+1\}^{K_m}} \theta_m^2 \left[ D_{r,s}^\alpha B_m \left( \tilde{x}_{\sigma^1_{K_m}, \kappa^m_{K_m}}, \dots, \tilde{x}_{\sigma^p_{K_m}, \kappa^m_{K_m}} \right) \right]^2 \\ & \times \prod_{j=1}^{K_m} \left( \tilde{x}_{\sigma^j_{K_m+1}, \kappa^m_{K_m}} - \tilde{x}_{\sigma^j_{K_m}, \kappa^m_{K_m}} \right). \end{aligned} \tag{7}$$

As a result, PRSS is rearranged in the following form

$$\begin{aligned} PRSS \approx & \sum_{i=1}^N (y_i - \mathbf{B}(\tilde{\mathbf{d}}_i)\boldsymbol{\theta})^2 \\ & + \sum_{m=1}^{M_{\max}} \lambda_m \theta_m^2 \sum_{i=1}^{(N+1)^{K_m}} \left( \sum_{|\boldsymbol{\alpha}|=1}^2 \sum_{r < s} \left[ D_{r,s}^\alpha B_m \left( \tilde{x}_{\sigma^1_{K_m}, \kappa^m_{K_m}}, \dots, \tilde{x}_{\sigma^p_{K_m}, \kappa^m_{K_m}} \right) \right]^2 \right) \end{aligned}$$

$$\times \prod_{j=1}^{K_m} \left( \tilde{x}_{l_{\sigma^{k_j^{m+1}}, k_j^m}}^{k_j^m} - \tilde{x}_{l_{\sigma^{k_j^m}, k_j^m}}^{k_j^m} \right). \tag{8}$$

A short representation of PRSS is as follows

$$PRSS \approx \left\| \mathbf{y} - \mathbf{B}(\tilde{\mathbf{d}})\boldsymbol{\theta} \right\|_2^2 + \sum_{m=1}^{M_{\max}} \lambda_m \sum_{i=1}^{(N+1)^{K_m}} L_{im}^2 \theta_m^2, \tag{9}$$

where  $\mathbf{B}(\tilde{\mathbf{d}}) = \left( 1, B_1(\tilde{\mathbf{x}}^1), \dots, B_M(\tilde{\mathbf{x}}^M), B_{M+1}(\tilde{\mathbf{x}}^{M+1}), \dots, B_{M_{\max}}(\tilde{\mathbf{x}}^{M_{\max}}) \right)^T$  is an  $(N \times (M_{\max} + 1))$  matrix with the point  $\tilde{\mathbf{d}} := (\tilde{\mathbf{x}}^1, \dots, \tilde{\mathbf{x}}^M, \tilde{\mathbf{x}}^{M+1}, \dots, \tilde{\mathbf{x}}^{M_{\max}})^T$ , and  $\boldsymbol{\theta} := (\theta_0, \theta_1, \dots, \theta_{M_{\max}})^T$ ;  $\|\cdot\|_2$  denotes the Euclidean norm in the argument. Here, the elements of  $\tilde{\mathbf{d}}$ , which are  $\tilde{\mathbf{x}}^1, \tilde{\mathbf{x}}^2, \dots, \tilde{\mathbf{x}}^{M_{\max}}$ , represent predictor data vector used in the  $m^{th}$  BF ( $m = 1, 2, \dots, M_{\max}$ ). On the other side,  $L_{im}$  are defined as

$$L_{im} = \left[ \left( \sum_{|\alpha|=1}^2 \sum_{\substack{r < s \\ \alpha = (\alpha_1, \alpha_2)^T, r, s \in V(m)}} [D_{r,s}^\alpha B_m(\hat{\mathbf{x}}_i^m)]^2 \right) \Delta \hat{\mathbf{x}}_i^m \right]^{1/2}. \tag{10}$$

Here,  $\hat{\mathbf{x}}_i^m$  ( $i = 1, 2, \dots, N$ ) are the canonical projections of the data points into the input dimensions of  $m^{th}$  BF with the same increasing order;  $\Delta \hat{\mathbf{x}}_i^m$  represent the differences raised on the  $i^{th}$  data vector,  $\hat{\mathbf{x}}_i^m$  (Weber et al. 2012; Yerlikaya 2008) is as given in (11)

$$\hat{\mathbf{x}}_i^m = \left( \tilde{x}_{l_{\sigma^{k_1^m}, k_1^m}}^{k_1^m}, \dots, \tilde{x}_{l_{\sigma^{k_{K_m}^m}, k_{K_m}^m}}^{k_{K_m}^m} \right), \quad \Delta \hat{\mathbf{x}}_i^m = \prod_{j=1}^{K_m} \left( \tilde{x}_{l_{\sigma^{k_j^{m+1}}, k_j^m}}^{k_j^m} - \tilde{x}_{l_{\sigma^{k_j^m}, k_j^m}}^{k_j^m} \right). \tag{11}$$

Through a uniform penalization, in other words, by taking the same  $\lambda$  value for each derivative term, PRSS can be turned into the Tikhonov regularization problem form given as follows (Aster et al. 2012)

$$PRSS \approx \left\| \mathbf{y} - \mathbf{B}(\tilde{\mathbf{d}})\boldsymbol{\theta} \right\|_2^2 + \lambda \|\mathbf{L}\boldsymbol{\theta}\|_2^2. \tag{12}$$

This problem can be evaluated from the view point of CQP, a technique used for continuous optimization, and handled by placing an appropriate bound,  $\tilde{M}$ , as follows to obtain the optimal solution

$$\begin{aligned} \min_{t, \boldsymbol{\theta}} t, \text{ such that } & \boldsymbol{\chi} = \begin{pmatrix} \mathbf{0}_N & \mathbf{B}(\tilde{\mathbf{d}}) \\ 1 & \mathbf{0}_{M_{\max}+1}^T \end{pmatrix} \begin{pmatrix} t \\ \boldsymbol{\theta} \end{pmatrix} + \begin{pmatrix} -\mathbf{y} \\ 0 \end{pmatrix}, \\ & \boldsymbol{\eta} = \begin{pmatrix} \mathbf{0}_{M_{\max}+1} & \mathbf{L} \\ 0 & \mathbf{0}_{M_{\max}+1}^T \end{pmatrix} \begin{pmatrix} t \\ \boldsymbol{\theta} \end{pmatrix} + \begin{pmatrix} \mathbf{0}_{M_{\max}+1} \\ \sqrt{\tilde{M}} \end{pmatrix}, \\ & \boldsymbol{\chi} \in L^{N+1}, \boldsymbol{\eta} \in L^{M_{\max}+2}, \end{aligned} \tag{13}$$

where  $L^{N+1}$ ,  $L^{M_{\max}+2}$  are the  $(N + 1)$ - and  $(M_{\max} + 2)$ - dimensional *second-order cones*, defined by

$$L^{N+1} = \left\{ \mathbf{x} = (x_1, x_2, \dots, x_{N+1})^T \in \mathbb{R}^{N+1} \mid x_{N+1} \geq \sqrt{x_1^2 + x_2^2 + \dots + x_N^2} \right\} \quad (N \geq 1),$$

and

$$L^{M_{\max}+2} = \left\{ \mathbf{x} = (x_1, x_2, \dots, x_{M_{\max}+2})^T \in \mathbb{R}^{M_{\max}+2} \mid x_{M_{\max}+2} \geq \sqrt{x_1^2 + x_2^2 + \dots + x_{M_{\max}+1}^2} \right\} \quad (M_{\max} > 0). \quad (14)$$

In applications, we observe that the log–log scale plot of the two criteria,  $\|\mathbf{L}\boldsymbol{\theta}\|_2$  versus  $\|\mathbf{y} - \mathbf{B}(\tilde{\mathbf{a}})\boldsymbol{\theta}\|_2$ , has a particular “L” shape whose corner point provides the optimum value of  $\sqrt{\tilde{M}}$ , and the proposed method provides a reliable solution to our problem. However, more efficient and robust algorithm(s) for locating the corner of an L-curve can be developed.

### 2.3 Bootstrap regression

#### 2.3.1 Bootstrap resampling

The bootstrap is a resampling technique that takes samples from the original data set with replacement (Chernick 2008). It is a data-based simulation method useful for making inferences such as estimating standard errors and biases, constructing CIs, testing hypothesis, and so on. Implementation of this method is not difficult, but depends heavily on computers. The bootstrap procedure can be defined as in Table 1.

Efron and Tibshirani (1993) indicate that bootstrap is applicable to any models such as nonlinear ones and the models which use estimation techniques other than LS. According to them, bootstrapping regression is applicable to nonparametric models as well as the parametric ones with no analytical solutions.

Let  $\mathbf{y} = \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\varepsilon}$  be a usual MLR model, where  $\mathbf{X}$  and  $\boldsymbol{\theta}$  represent the vector of independent variables as its columns and model parameters, respectively. The error term,  $\boldsymbol{\varepsilon}$ , is normally distributed with zero mean and constant variance. If assumptions regarding the model are satisfied, reliable inferences can be made. In cases such as nonnormal error distribution or nonlinear model fitting, alternative approaches using bootstrap are recommended (Freedman 1981; Hjorth 1994). Three bootstrap regression methods used in the study are described below.

#### 2.3.2 Fixed-X resampling (residual resampling)

In this method, the response values are considered to be random due to the error component. It is more advantageous when it is used with fixed (known) independent variables, with small data sets, and adequate models (Fox 2002). The step-by-step algorithm of the method is given in Table 2.

**Table 1** The bootstrap procedure

Step 1	Generate $a$ -th bootstrap sample ( $x^{*a}$ ) of size $N$ from the original sample with replacement
Step 2	Compute the statistic of interest for this sample
Step 3	Repeat steps 1–2 $a = 1, \dots, A$ times and obtain the empirical CDF of the statistic of interest

**Table 2** The fixed-X resampling procedure

Step 1	Fit the model $y = X\theta + \varepsilon$ to the data and obtain the fitted values, $\hat{y}$ , and the residuals, $e$
Step 2	Select a bootstrap sample $e^{*a}$ ( $a = 1, 2, \dots, A$ ) of residuals from $e$ using the procedure in Table 1, and add them to the fitted values to obtain new response variables, $y_{new} = \hat{y} + e^{*a}$
Step 3	Fit the model $y_{new} = X\theta + \varepsilon$ to the original independent variables, $X$ , and the new response variables, $y_{new}$ , and collect new parameter estimates, $\hat{\theta}$
Step 4	Repeat steps 2–3 $A$ times

2.3.3 Random-X resampling (pairs bootstrap)

This technique can be used in case of heteroscedasticity, lack of significant independent variables, and the need for semiparametric or nonparametric model approach (Chernick 2008). The step-by-step algorithm of the method is given in Table 3.

2.3.4 Wild bootstrap

The wild bootstrap is a relatively new approach, when compared to random-X resampling, proposed for handling heteroscedastic models (Flachaire 2005). Its algorithm is the same as to that of fixed-X resampling given in Table 2 with the only change in Step 2 that the bootstrap of residuals, that is errors,  $e^{*a}$  ( $a = 1, \dots, A$ ), are attached to the fitted values after they are randomly assigned to be 1 or -1 with equal probability.

2.4 Validation technique and performance criteria

In the comparison of models, 3-fold CV technique is used (Martinez and Martinez 2002; Gentle 2009). In this technique, data sets are randomly divided into three parts (folds). At each of the three attempts, two different folds (66.6% of observations) are combined to develop models while the other fold (33.3% of observations) is kept to test them. The combined part and the other fold are referred to as training and test data sets, respectively.

The performances of the models developed are evaluated with respect to different criteria including accuracy, precision, complexity, stability, robustness and efficiency. The accuracy criterion is used to measure the predictive ability of the models while precision criterion is used to determine the amount of variation in the parameter estimates; the less variable ones indicate more precision. The mean absolute error (MAE), determination of coefficient ( $R^2$ ) and percentage of residuals within three standard deviations (PWI). On the other hand, the precision of parameter estimates are determined by their empirical CIs. Other criterion

**Table 3** The random-X resampling procedure

Step 1	Select bootstrap samples of size $N$ , using the procedure in Table 1, among the rows of the augmented matrix, $Z = (y X)$ , $Z^{*a}$
Step 2	Fit the model $y = X\theta + \varepsilon$ to $Z^{*a}$ , and collect new parameter estimates, $\hat{\theta}$
Step 3	Repeat steps 1–2 $A$ times



used in comparisons is the complexity; it is measured by the mean squared error (MSE). It is expected that, in general, the performance measures for test data may not be as good as to that of the training data. Besides, the stabilities of the accuracy and complexity measures obtained from the training and test data sets are also evaluated. The definitions as well as bounds on these measures, where applies, are presented in the Appendix. Furthermore, robustness of the measures with respect to different data sets is evaluated by considering the standard deviations of the measures. Moreover, to assess the computational efficiency of the models build, computational run times are utilized.

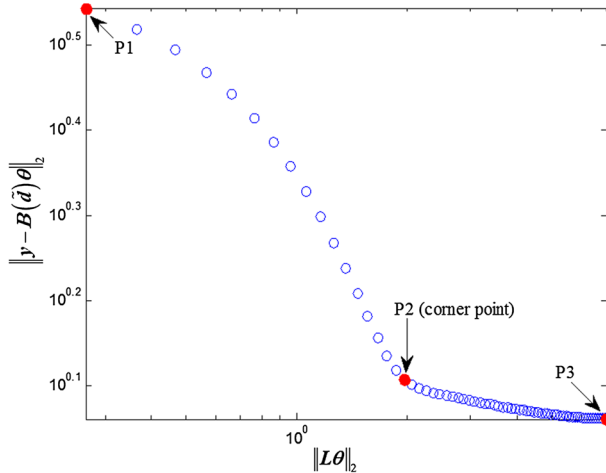
### 3 BCMARS: bootstrapping CMARS

As stated above, studies indicate that CMARS is a good alternative to the backward part of MARS method. However, CMARS produces models at least as complex as MARS models. To overcome this problem, we propose to use a CS method, called bootstrap, due to the lack of distributional assumptions of CMARS, and developed the BCMARS algorithm. The steps of the algorithm given in Table 4 are followed for obtaining three different BCMARS models, labeled as BCMARS-F (uses Fixed-X Resampling), BCMARS-R (uses Random-X Resampling) and BCMARS-W (uses Wild Bootstrap) (Yazıcı 2011; Yazıcı et al. 2011).

In the Step 2 of Table 4, the optimal value of  $\sqrt{\tilde{M}}$  is the closest solution to the corner of L-curve which is the point with maximum curvature. To determine the corner point,  $\|y - B(\tilde{d})\theta\|_2$  versus  $\|L\theta\|_2$  is plotted in the log–log scale and its corner is located. This point tries to minimize both criteria,  $\|y - B(\tilde{d})\theta\|_2$  and  $\|L\theta\|_2$ , in a balanced manner. We should note here that the corner point is data dependent. There can be many solutions to the CQP problem for different  $\sqrt{\tilde{M}}$  values, which may lead to different estimates. To illustrate,

**Table 4** The BCMARS algorithm

Step 1	The forward part of MARS algorithm is run and the set of BFs is constructed using the original data, $y$ and $X$ . Note that these BFs are considered as fixed
Step 2	CMARS model is constructed and the optimal value of $\sqrt{\tilde{M}}$ is decided as the corner point of the plot of $\ y - B(\tilde{d})\theta\ _2$ versus $\ L\theta\ _2$ in the log–log scale (see Fig. 1). The selected value gives the best solution for both accuracy and complexity in terms of PRSS in (12)
Step 3	Select one of the following BCMARS methods <ul style="list-style-type: none"> <li>• BCMARS-F: follow the procedure given in Table 2 by using the model in (13) in place of the MLR model</li> <li>• BCMARS-R: follow the procedure given in Table 3 by using the model in (13) in place of the MLR model</li> <li>• BCMARS-W: follow the procedure given in Sect. 2.3.4</li> </ul>
Step 4	Decide on the level of significance, $\alpha$ , and construct bootstrap percentile interval using Eq. (17), given in the Appendix, to determine the significance of the parameters. If the percentile interval includes zero, then the parameter is found to be insignificant
Step 5	Repeat Steps 2–5 until there is not any insignificant parameters in the model



**Fig. 1** The curve of  $\|L\theta\|_2$  versus  $\|y - B(\tilde{d})\theta\|_2$  in log–log scale

let us consider three representative points for the L-curve, P1, P2 and P3, as given in Fig. 1. While P1 and P3 minimize  $\|L\theta\|_2$  and  $\|y - B(\tilde{d})\theta\|_2$ , respectively, P2, the corner of L-curve, tries to minimize both simultaneously. Here, P1 represents the least complex and least accurate solution whereas P3 represents the most complex and most accurate solution. On the other hand, P2 provides better prediction performance than the other points with respect to both complexity and accuracy criteria (Weber et al. 2012).

### 4 Application and findings

In order to evaluate and compare the performances of models developed by using MARS, CMARS and BCMARS methods, they are run on four different data sets to observe the effects of certain data characteristics such as size (i.e. the number of observations,  $N$ ) and scale (i.e. the number of independent variables,  $p$ ) on the methods’ performances. Note that the data sets are classified as small and medium subjectively. The data sets used in comparisons are presented in Table 5.

**Table 5** Data sets used in comparisons

Scale ( $p$ )		
( $N, p$ )	Small ( $p < 10$ )	Medium ( $10 < p < 20$ )
<i>Sample size (<math>N</math>)</i>		
Small ( $N \sim 150$ )	Concrete slump (CS), (Yeh 2007), (103,7)	Uniform sampling (US), (Kartal 2007), (160,10)
Medium ( $N \sim 500$ )	PM10 (Aldrin 2006), (500,7)	Forest fires (FF) (Cortez and Morais 2007), (517,11)

While validating the models, 3-fold CV is used as described in Sect. 2.4. As a result, three models are developed and tested for each of the method applied on a data set. In applications, the R package “*Earth*” Milborrow (2009), MATLAB (2009) and the MOSEK optimization software (2011) run in MATLAB are utilized.

To construct BCMARS models, the algorithms given in Sect. 3 are applied step-by-step by taking  $A$ , in Table 1, 2 and 3, as 1000. Then, the performance measures for each model are calculated. Moreover, the computational run times of the methods are recorded to be compared.

## 5 Results and discussion

In this section, it is aimed to compare the performances of the methods studied, namely MARS, CMARS, BCMARS-F, BCMARS-R and BCMARS-W, in general, according to different features of data sets such as size and scale. In these comparisons, various criteria including accuracy, precision, stability, efficiency and robustness are considered.

### 5.1 Comparison with respect to overall performances

The mean and standard deviations of measures obtained from four data sets are given in Table 6. These values are calculated for both training and testing data sets in addition to the stability of measures. Definitions of the measures and their bounds are given in the Appendix. In this table, for training and test data, lower means for MAE and MSE; higher means for  $R^2$  and PWI measures indicate better performances. Besides, stability values for all means close to one indicate better performances. On the other hand, smaller standard deviations imply robustness for the corresponding measure. The following conclusions can be drawn from this table:

- BCMARS-F and BCMARS-R are the most accurate, robust and least complex for training and testing data sets, respectively.
- BCMARS-R and BCMARS-W methods are the most stable, and BCMARS-R has the most robust stability.

### 5.2 Comparison with respect to sample size

Table 7 presents the performance measures of the methods studied with respect to two sample size categories: small and medium. Depending on the results given in the table, following conclusions can be reached.

- All methods perform the best in small data sets when compared to the medium size for training and testing data.
- BCMARS-F and MARS perform the best for small training and testing data sets, respectively. Moreover, BCMARS-W competes with MARS in small testing data sets.
- Among all, BCMARS-W method is the most stable one in small data sets.
- BCMARS-F and BCMARS-W are the most stable methods in small size data when compared to medium size.

Note that “the best” here indicates better performance with respect to at least two measures out of four.

**Table 6** Overall performances (Mean±SD) of the methods

Performance measures	MARS	CMARS	BCMARS-F	BCMARS-R	BCMARS-W
<i>Training</i>					
MAE	0.3453 ±0.2336	0.4040 ±0.3980	0.3204*±0.2260**	0.3356 ±0.2263	0.4251 ±0.2797
MSE	0.4015 ±0.3064	0.6070 ±0.9080	0.3117*±0.2700**	0.4230 ±0.3990	0.5770 ±0.4950
R <sup>2</sup>	0.6005 ±0.2797	0.5911 ±0.3407	0.6827*±0.2492**	0.6120 ±0.3350	0.5127 ±0.3398
PWI	0.9944*±0.0082**	0.9942 ±0.0082**	0.9909 ±0.0153	0.9932 ±0.0140	0.9855 ±0.0158
<i>Testing</i>					
MAE	0.4576*±0.2956**	0.5800 ±0.4580	0.4838±0.3076	0.6460±0.6110	0.4977±0.2998
MSE	3.0700±7.0900	1.5780 ±2.1350	1.2670±1.9970	0.5480*±0.3660**	1.0720±1.2710
R <sup>2</sup>	0.4480±0.3820	0.3630±0.4030	0.4500±0.3800	0.4530*±0.3770**	0.3840±0.4010
PWI	0.9930*±0.0108	0.9930*±0.0106**	0.9884±0.0177	0.9890±0.0169	0.9878±0.0120
<i>Stability</i>					
MAE	0.7657±0.1848	0.7440±0.2383	0.7252±0.1939	0.7375±0.2870	0.8690*±0.1783**
MSE	0.5500±0.3710	0.5690±0.3400	0.5550±0.3450	0.6374±0.2174**	0.7616*±0.2852
R <sup>2</sup>	0.6070±0.3680	0.4690±0.3940	0.5750±0.3640	0.6577*±0.3063**	0.6300±0.3650
PWI	0.9950*±0.0070	0.9940±0.0070	0.9940±0.0070	0.9950*±0.0060**	0.9940±0.0080

\* indicates better performance with respect to means; \*\* indicates better performance with respect to spread

**Table 7** Averages of performance measures with respect to different sample sizes

Sample size	Performance measures	MARS	CMARS	BCMARS-F	BCMARS-R	BCMARS-W
<i>Training</i>						
Small	MAE	0.2340	0.3570	0.1899*	0.2092	0.3410
	MSE	0.1773	0.6020	0.1158*	0.1387	0.3000
	$R^2$	0.8208	0.7840	0.8824*	0.8596	0.7350
	PWI	1.0000*	0.9970	0.9910	0.9910	0.9870
Medium	MAE	0.4563	0.4498*	0.4769	0.4874	0.5090
	MSE	0.6257	0.6125	0.5469*	0.7630	0.8540
	$R^2$	0.3802	0.3978	0.4431*	0.3140	0.2908
	PWI	0.9888	0.9900	0.9890	0.9940*	0.9830
<i>Testing</i>						
Small	MAE	0.3300*	0.5560	0.3440	0.7280	0.3790
	MSE	0.3520*	1.0010	0.3980	0.3670	0.3570
	$R^2$	0.7110*	0.5760	0.6770	0.6800	0.6500
	PWI	1.0000*	1.0000*	0.9910	0.9910	0.9920
Medium	MAE	0.5849	0.6052	0.6518	0.5468*	0.6160
	MSE	5.7800	2.1500	2.3100	0.7658*	1.7880
	$R^2$	0.1853*	0.1497	0.1765	0.1817	0.1178
	PWI	0.9860*	0.9860*	0.9850	0.9860*	0.9830
<i>Stability</i>						
Small	MAE	0.2250	0.7300	0.7265	0.6110	0.9359*
	MSE	0.4980	0.5770	0.5750	0.5530	0.8835*
	$R^2$	0.7700	0.5960	0.7350	0.7510	0.7710*
	PWI	1.0000*	0.9970	0.9990	0.9990	0.9940
Medium	MAE	0.4431	0.7578	0.7236	0.8888*	0.8022
	MSE	0.5760	0.5620	0.4410	0.7049*	0.6150
	$R^2$	0.4450	0.3410	0.3830	0.5460*	0.4890
	PWI	0.9915	0.9900	0.9898	0.9920	0.9948*

\* indicates better performance with respect to the corresponding measure and sample

### 5.3 Comparisons with respect to scale

In Table 8, the performance measures of the studied methods with respect to two scale types: small and medium are presented. Depending on the results given in the table, following conclusions can be drawn:

- For training data sets, medium scale produces better models for all of the methods. Moreover, BCMARS-F is the best performing one regardless of the scale.
- For testing data sets, MARS, BCMARS-F and BCMARS-W perform equally well on both scales; while medium scale gives the best results for the other methods studied.
- MARS and BCMARS-W are the most stable methods for small scale data compared to medium scale; CMARS and BCMARS-R are the most stable methods for medium scale compared to small scale. BCMARS-F performs equally well on both scales.

**Table 8** Averages of performance measures with respect to different scale

Scale	Performance measures	MARS	CMARS	BCMARS-F	BCMARS-R	BCMARS-W
<i>Training</i>						
Small	MAE	0.5229	0.4140*	0.4720	0.4910	0.6561
	MSE	0.4572	0.8992	0.3830*	0.4040	0.7728
	$R^2$	0.5483	0.4985	0.6139*	0.5928	0.4078
	PWI	0.9970	0.9924	0.9980*	0.9980*	0.9934
Medium	MAE	0.1677	0.1773	0.1384*	0.1492	0.1940
	MSE	0.3417	0.3500	0.2260*	0.4450	0.3810
	$R^2$	0.6591	0.6630	0.7650*	0.6340	0.6170
	PWI	0.9913	0.9920*	0.9820	0.9870	0.9770
<i>Testing</i>						
Small	MAE	0.6696	0.5445*	0.6747	0.6776	0.7130
	MSE	0.7327*	2.3959	0.7717	0.7443	0.8469
	$R^2$	0.3297	0.3377*	0.3240	0.3293	0.2721
	PWI	0.9964*	0.9901	0.9960	0.9960	0.9932
Medium	MAE	0.2703	0.2790	0.2550*	0.6070	0.2820
	MSE	5.4630	1.7800	1.8600	0.3130*	1.2980
	$R^2$	0.5107	0.5040	0.6000	0.6020*	0.4960
	PWI	0.9892	0.9900*	0.9790	0.9810	0.9820
<i>Stability</i>						
Small	MAE	0.5008	0.7801	0.7041	0.7300	0.9200*
	MSE	0.6515	0.5771	0.5183	0.5539	0.8378*
	$R^2$	0.6521*	0.3714	0.5540	0.5539	0.6277
	PWI	0.9984*	0.9930	0.9977	0.9979	0.9980
Medium	MAE	0.7666	0.7900	0.7505	0.7470	0.8100*
	MSE	0.3474	0.5920	0.3480	0.8040*	0.6850
	$R^2$	0.5628	0.5310	0.6000	0.7600*	0.6330
	PWI	0.9931*	0.9930*	0.9910	0.9930	0.9920

\* indicates better performance with respect to the corresponding measure and scale

- MARS and BCMARS-W are more stable for small scale among all methods. BCMARS-R is more stable for medium scale data sets.

## 5.4 Evaluation of the computational efficiencies

The elapsed time for each method applied on each data set are recorded on Pentium (R) Dual-Core CPU 2.80 GHz processor and 32-bit operating system Windows ®computer during the runs (Table 9). Depending on the results, following conclusions can be stated:

- Run times increases as sample size and scale increases, except MARS.
- Bootstrap methods run considerably longer times than MARS and CMARS. Three bootstrap regression methods have almost the same computational efficiencies in small size

**Table 9** Run times (in seconds) of methods with respect to size and scale of data sets

	Scale	
	Small	Medium
<i>Sample size</i>		
Small	MARS: <0.08 s*	MARS: <0.08 s*
	CMARS: <4.47 s	CMARS: <19.52 s
	BCMARS-F: < $1.6 \times 10^3$ s	BCMARS-F: < $1.3 \times 10^4$ s
	BCMARS-R: < $1.6 \times 10^3$ s	BCMARS-R: < $1.8 \times 10^4$ s
	BCMARS-W: < $1.6 \times 10^3$ s	BCMARS-W: < $1.5 \times 10^4$ s
Medium	MARS: <0.08 s*	MARS: <0.09 s*
	CMARS: <18.20 s	CMARS: < 21.67 s
	BCMARS-F: < $1.5 \times 10^4$ s	BCMARS-F: < $1.8 \times 10^4$ s
	BCMARS-R: < $0.7 \times 10^4$ s	BCMARS-R: < $3.1 \times 10^4$ s
	BCMARS-W: < $0.8 \times 10^4$ s	BCMARS-W: < $1.6 \times 10^4$ s

\* indicates better performance with respect to run times

and small scale data sets. Run times of these methods increase almost ten times as much as the scale increases from small to medium.

- BCMARS-R and BCMARS-W have similar better efficiencies in medium size small scale data sets. Their run times increase almost five times as much as the sample size increases in small scale data sets.
- BCMARS-F and BCMARS-W have similar better efficiencies for medium size medium scale data sets.

## 5.5 Evaluation of the precision of model parameters

In addition to the accuracy, complexity and stability measures of the models, the CIs using Eq. (17) given in the Appendix and two different standard deviations of the parameters as described in Eq. (18) in the Appendix are calculated after bootstrapping. These values are compared with those obtained from bootstrapping CMARS. For the detailed results, one can refer to Yazıcı (2011). The shorter the lengths of the CIs and the smaller the standard deviations are, the more precise the parameter estimates are. According to the results, following conclusions can be drawn.

- In US (small size medium scale) data, CMARS, BCMARS-F and BCMARS-R build the same models. Hence, the precision of their parameters are the same.
- For all data sets except US, the lengths of CIs become narrower and standard deviations of the parameters become smaller after bootstrapping CMARS, thus, resulting in more precise parameter estimates.
- In general, two different types of standard deviations obtained for all BCMARS methods are smaller than the ones obtained from CMARS.

## 6 Conclusion and further research

In this study, three different bootstrap methods are applied to a machine learning method, called CMARS, which is an improved version of the backward step of the well-known method

MARS. Although CMARS overperforms MARS with respect to several criteria, it constructs models which are at least as complex as MARS (Weber et al. 2012). In this study, it is aimed to reduce the complexity of CMARS models without degrading its performance. To achieve this aim, bootstrapping regression methods, namely fixed-X and random-X resampling, and wild bootstrap, are utilized by adopting an iterative approach to determine whether the parameters statistically contribute to the developed CMARS model or not. The reason of using a computational method here is the lack of prior knowledge regarding the distributions of the model parameters.

The performances of the methods are empirically evaluated and compared with respect to several criteria (e.g. accuracy, complexity, stability, robustness, precision, computational efficiency) by using four data sets which are selected subjectively to represent the small and medium sample size and scale categories. All performance criteria are explained in the Appendix. In addition, to validate all models developed, three-fold CV approach is used.

Depending on the comparisons, particularly for testing data and stability results presented in Sect. 5, one may conclude the followings:

- In the overall, BCMARS-R is the best performing method.
- Small size (training and testing) data sets produce the best results for all methods; for small and medium size data, BCMARS-W and BCMARS-R overperform the others, respectively.
- Medium scale produces the best results for CMARS and BCMARS-R when compared to the others, and BCMARS-R is the better performing one.
- Bootstrapping methods give the most precise parameter estimates; however, they are computationally the least efficient.

In short, depending on the above conclusions, it may be suggested that BCMARS-R method leads to more accurate, precise and less complex models, particularly for medium size and medium scale data. Nevertheless, it is the least efficient method among the others for this type of data set in terms of run time.

In the future, BCMARS methods are going to be applied on more data sets with small to large size and scale to be able to examine the interactions that may exist between data size and scale more clearly.

**Acknowledgments** Authors would like to thank to the editor and the anonyms referees for their valuable comments and criticisms. Their contributions lead to the improved version of this paper.

## Appendix: Definitions of comparison measures

### Nomenclature

$y_i$	is the response value for the $i^{th}$ observation,
$\hat{y}_i$	is the estimated response value for the $i^{th}$ observation,
$\bar{y}$	is the value of the mean response,
$N$	is the number of observations (sample size),
$p$	is the number of terms (i.e. BF <sub>s</sub> ) in the model,
$e_i = y_i - \hat{y}_i$	is the residual for the $i^{th}$ observation,



## Accuracy measures

### Mean absolute error (MAE)

$$MAE = \frac{1}{N} \sum_{i=1}^N |e_i| = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|, \text{ where } MAE \geq 0. \quad (15)$$

Small values are the better.

### The coefficient of determination ( $R^2$ )

$$R^2 = \frac{\sum_{i=1}^N (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^N (y_i - \bar{y})^2}, \text{ where } 0 \leq R^2 \leq 1. \quad (16)$$

Higher values indicate better fit.

### Proportion of residuals within some user-specified range (PWI)

PWI is the proportion of residuals within some user-specified range such as three standard deviations. The greater the percentage is, the better the performance is.

## Precision measure

### Percentile interval and bootstrap estimate of bias

$$\left( \hat{\theta}_A^{*(\alpha/2)}, \hat{\theta}_A^{*(1-\alpha/2)} \right). \quad (17)$$

Percentile interval uses the empirical CDF of the bootstrap sample to find the upper  $\left( \hat{\theta}_A^{*(1-\alpha/2)} \right)$  and lower  $\left( \hat{\theta}_A^{*(\alpha/2)} \right)$  endpoints.

### Bootstrap estimate of standard deviation

Standard deviations of the distributions of parameters are calculated in two-ways

1.

$$\widehat{se}_A(\hat{\theta}) = \left\{ \frac{1}{A-1} \sum_{a=1}^A \left( \hat{\theta}^{*a} - \bar{\hat{\theta}}^* \right)^2 \right\}^{1/2}, \quad (18)$$

where  $\bar{\hat{\theta}}^* = \frac{1}{A} \sum_{a=1}^A \hat{\theta}^{*a}$  and  $\hat{\theta}^{*a}$  is the bootstrap replication of  $\hat{\theta}$  (Martinez and Martinez 2002).

2. Using the associated empirical CDF, the standard deviation of each parameter is calculated many (i.e.  $A$ ) times, and the standard deviations of these values are obtained (Yazici 2011). Thus, it measures the spread of the standard deviations around the mean of standard deviations.

## Complexity measure

Mean square error (MSE)

$$MSE = \frac{1}{N-p} \sum_{i=1}^N (y_i - \hat{y}_i)^2, \text{ where } MSE \geq 0. \quad (19)$$

Larger values indicate more complex models.

## Stability measure

The stability of a model measured by

$$STAB_{CR} = \min \left\{ \frac{CR_{TR}}{CR_{TE}}, \frac{CR_{TE}}{CR_{TR}} \right\}, 0 < STAB_{CR} \leq 1, \quad (20)$$

where  $CR_{TR}$  and  $CR_{TE}$  represents the performance measures obtained from training and testing samples, respectively. The stability measure close to one indicates higher stability. A stable model performs equally well both on training and testing data (Osei-Bryson 2004).

## References

- Aldrin, M. (2006). Improved predictions penalizing both slope and curvature in additive models. *Computational Statistics and Data Analysis*, 50(2), 267–284.
- Aster, R. C., Borchers, B., & Thurber, C. (2012). *Parameter estimation and inverse problems*. Burlington: Academic Press.
- Austin, P. (2008). Using the bootstrap to improve estimation and confidence intervals for regression coefficients selected using backwards variable elimination. *Statistics in Medicine*, 27(17), 3286–3300.
- Batmaz, İ., Yerlikaya-Özkurt, F., Kartal-Koç, E., Köksal, G., Weber, G. W. (2010). Evaluating the CMARS performance for modeling nonlinearities. In *Proceedings of the 3rd global conference on power control and optimization, gold coast (Australia)*, vol. 1239, pp. 351–357.
- Çelik, G. (2010). *Parameter estimation in generalized partial linear models with conic quadratic programming*. Master Thesis, Graduate School of Applied Mathematics, Department of Scientific Computing, METU, Ankara, Turkey.
- Chernick, M. (2008). *Bootstrap methods: A guide for practitioners and researchers*. New York: Wiley.
- Cortez, P., & Morais, A. (2007). Data mining approach to predict forest fires using meteorological data. In J. Neves, M. F. Santos, J. Machado (ed.), *New trends in artificial intelligence, proceedings of the 13th EPIA 2007 - Portuguese conference on artificial intelligence, December, Guimarães (Portugal)*, pp. 512–523.
- Deconinck, E., Zhang, M. H., Petitot, F., Dubus, E., Ijjaali, I., Coomans, D., et al. (2008). Boosted regression trees, multivariate adaptive regression splines and their two-step combinations with multiple linear regression or partial least squares to predict blood-brain barrier passage: A case study. *Analytica Chimica Acta*, 609(1), 13–23.
- Denison, D. G. T., Mallick, B. K., & Smith, F. M. (1998). Bayesian MARS. *Statistics and Computing*, 8(4), 337–346.
- Efron, B. (1988). Computer-intensive methods in statistical regression. *Society for Industrial and Applied Mathematics*, 30(3), 421–449.
- Efron, B., & Tibshirani, R. J. (1991). Statistical data analysis in the computer age. *Science*, 253, 390–395.
- Efron, B., & Tibshirani, R. J. (1993). *An introduction to the bootstrap*. New York: Chapman & Hall.
- Flachaire, E. (2005). Bootstrapping heteroskedastic regression models: Wild bootstrap vs. pairs bootstrap. *Computational Statistics and Data Analysis*, 49(2), 361–376.
- Fox, J. (2002). *Bootstrapping regression models. An R and S-plus companion to applied regression: Web appendix to the book*. Sage, CA: Thousand Oaks.
- Freedman, D. A. (1981). Bootstrapping regression models. *The Annals of Statistics*, 9(6), 1218–1228.
- Friedman, J. (1991). Multivariate adaptive regression splines. *Annals of Statistics*, 19(1), 1–67.

- Gentle, J. E. (2009). *Computational statistics*. New York: Springer.
- Ghasemi, J. B., & Zolfonoun, E. (2013). Application of principal component analysis–multivariate adaptive regression splines for the simultaneous spectrofluorimetric determination of dialkyltins in micellar media. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 115, 357–363.
- Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning, data mining, inference and prediction*. New York: Springer.
- Hjorth, J. S. U. (1994). *Computer intensive statistical methods: Validation model selection and bootstrap*. New York: Chapman & Hall.
- Holmes, C. C., & Denison, D. G. T. (2003). Classification with bayesian MARS. *Machine Learning*, 50, 159–173.
- Kartal, E. (2007). *Metamodeling complex systems using linear and nonlinear regression methods*. Master Thesis, Graduate School of Natural and Applied Sciences, Department of Statistics, METU, Ankara, Turkey.
- Kriner, M. (2007). *Survival analysis with multivariate adaptive regression splines*. Dissertation, LMU Munchen: Faculty of Mathematics, Computer Science and Statistics, Munchen.
- Lee, Y., & Wu, H. (2012). MARS approach for global sensitivity analysis of differential equation models with applications to dynamics of influenza infection. *Bulletin of Mathematical Biology*, 74, 73–90.
- Lin, C. J., Chen, H. F., & Lee, T. S. (2011). Forecasting tourism demand using time series, artificial neural networks and multivariate adaptive regression splines: evidence from Taiwan. *International Journal of Business Administration*, 2(2), 14–24.
- Martinez, W. L., & Martinez, A. R. (2002). *Computational statistics handbook with Matlab*. New York: Chapman & Hall.
- MATLAB Version 7.8.0 (2009). The math works, USA.
- Milborrow, S. (2009). Earth: Multivariate adaptive regression spline models.
- Montgomery, D. C., Peck, E. A., & Vining, G. G. (2006). *Introduction to linear regression analysis*. New York: Wiley.
- MOSEK, Version 6. A very powerful commercial software for CQP, ApS, Denmark. <http://www.mosek.com>. Accessed Jan 7, 2011.
- Osei-Bryson, K. M. (2004). Evaluation of decision trees: A multi-criteria approach. *Computers & Operational Research*, 31, 1933–1945.
- Özmen, A., Weber, G. W., Batmaz, İ., & Kropat, E. (2011). RCMARS: Robustification of CMARS with different scenarios under polyhedral uncertainty set. *Communications in Nonlinear Science and Numerical Simulation (CNSNS)*, 16(12), 4780–4787.
- Salibian-Barrera, M., & Zamar, R. Z. (2002). Bootstrapping robust estimates of regression. *The Annals of Statistics*, 30(2), 556–582.
- Sezgin-Alp, O. S., Büyükbekci, E., Iscanoglu Cekic, A., Yerlikaya-Özkurt, F., Taylan, P., & Weber, G.-W. (2011). CMARS and GAM & CQP—modern optimization methods applied to international credit default prediction. *Journal of Computational and Applied Mathematics (JCAM)*, 235, 4639–4651.
- Taylan, P., Weber, G.-W., & Yerlikaya-Özkurt, F. (2010). A new approach to multivariate adaptive regression spline by using Tikhonov regularization and continuous optimization. *TOP (the Operational Research Journal of SEIO (Spanish Statistics and Operations Research Society))*, 18(2), 377–395.
- Weber, G. W., Batmaz, İ., Köksal, G., Taylan, P., & Yerlikaya-Özkurt, F. (2012). CMARS: A new contribution to nonparametric regression with multivariate adaptive regression splines supported by continuous optimization. *Inverse Problems in Science and Engineering*, 20(3), 371–400.
- Wegman, E. (1988). Computational statistics: A new agenda for statistical theory and practice. *Journal of the Washington Academy of Sciences*, 78, 310–322.
- Yazıcı, C. (2011). *A computational approach to nonparametric regression: Bootstrapping CMARS method*. Master Thesis, Graduate School of Natural and Applied Sciences, Department of Statistics, METU, Ankara, Turkey.
- Yazıcı, C., Yerlikaya-Özkurt, F., & Batmaz, İ. (2011). A computational approach to nonparametric regression: Bootstrapping CMARS method. In *ERCIM'11:4th international conference of the ERCIM W&G on computing and statistics*. London, UK. December 17–19. Book of Abstracts, 129.
- Yeh, I.-C. (2007). Modeling slump flow of concrete using second-order regressions and artificial neural networks. *Cement and Concrete Composites*, 29(6), 474–480.
- Yerlikaya, F. (2008). *A new contribution to nonlinear robust regression and classification with mars and its applications to data mining for quality control in manufacturing*. Master Thesis, Graduate School of Applied Mathematics, Department of Scientific Computing, METU, Ankara, Turkey.
- Yerlikaya-Özkurt, F., Batmaz, İ., & Weber, G.-W. (2014). A review of conic multivariate adaptive regression splines (CMARS): A powerful tool for predictive data mining, to appear as chapter in book. In D.

- Zilberman, A. Pinto, (eds.) *Springer volume modeling, optimization, dynamics and bioeconomy, series springer proceedings in mathematics*.
- Yetere-Kurşun, & A., Batmaz, İ. (2010). Comparison of regression methods by employing bootstrapping methods. *COMPSTAT2010: 19th international conference on computational statistics*. Paris, France. August 22–27. Book of Abstracts, 92.
- York, T. P., Eaves, L. J., Van Den Oord, E., & JC, G. (2006). Multivariate adaptive regression splines: A powerful method for detecting disease-risk relationship differences among subgroups. *Statistics in Medicine*, 25(8), 1355–1367.
- Zakeri, I. F., Adolph, A. L., Puyau, M. R., Vohra, F. A., & Butte, N. F. (2010). Multivariate adaptive regression splines models for the prediction of energy expenditure in children and adolescents. *Journal of Applied Psychology*, 108, 128–136.