Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

cmaRs: A powerful predictive data mining package in R

Fatma Yerlikaya-Özkurt [a,*], Ceyda Yazıcı [b], İnci Batmaz [c]

[a] *Department of Industrial Engineering, Atılım University, Ankara, Turkey*
[b] *Department of Mathematics, TED University, Ankara, Turkey*
[c] *Department of Statistics, Middle East Technical University, Ankara, Turkey*

ARTICLE INFO

Dataset link: https://github.com/yaziciceyda/cmaRs, http://archive.ics.uci.edu/ml

Keywords:
Conic multivariate adaptive regression splines
Nonparametric regression
Tikhonov regularization
Conic quadratic programming
Interior point method
Binary classification

ABSTRACT

Conic Multivariate Adaptive Regression Splines (CMARS) is a very successful method for modeling nonlinear structures in high-dimensional data. It is based on MARS algorithm and utilizes Tikhonov regularization and Conic Quadratic Optimization (CQO). In this paper, the open-source R package, `cmaRs`, built to construct CMARS models for prediction and binary classification is presented with illustrative applications. Also, the CMARS algorithm is provided in both pseudo and R code. Note here that `cmaRs` package provides a good example for a challenging implementation of CQO based on MOSEK solver in R environment by linking R to MOSEK through the package `Rmosek`.

Code metadata

Current code version	v0.1.3
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-23-00457
Permanent link to Reproducible Capsule	Not available
Legal Code License	GPL (>= 2)
Code versioning system used	None
Software code languages, tools, and services used	R, Mosek, Rtools
Compilation requirements, operating environments & dependencies	R and R packages: Rmosek, earth, graphics, Rmosek, stats, stringr, utils, Matrix, AUC, Ryacas0, ROCR
If available Link to developer documentation/manual	https://cran.r-project.org/web/packages/cmaRs/cmaRs.pdf https://cran.r-project.org/web/packages/cmaRs/vignettes/Intro_to_cmaRs.pdf
Support email for questions	ceydayazici86@gmail.com fatma.yerlikaya@atilim.edu.tr

1. Motivation and significance

Data mining (DM) is a well-known process of discovering hidden information in huge data sets. Extensively used predictive DM methods include Generalized linear models (GLMs), classification and regression trees (CART), artificial neural networks (ANN), support vector machines (SVM) and multivariate adaptive regression splines (MARS) [1–4]. Convex MARS (CMARS), a relatively new such method, is developed based on MARS [5], which constructs the regression function in two steps by searching through the data with the help of the piecewise linear functions, called basis functions (BFs). In the backward step, penalized residual sum of squares (PRSS) is constructed using the largest set of BFs formed in the forward step. Next, PRSS is turned into a Tikhonov regularization problem, and then, solved by the conic quadratic programming (CQP), using interior point methods (IPMs) [6–8]. This well-conditioned estimation procedure provides a trade-off between the highest accuracy and the smallest complexity leading to stable and robust results.

In the last decade, several studies have been conducted to evaluate performance of the CMARS algorithm against other well-known ones. In a rigorous comparison study, the performances of CMARS and MARS methods are statistically compared by using a cross-validation (CV) approach on several real-life data sets with different features,

[*] Corresponding author.
E-mail addresses: fatma.yerlikaya@atilim.edu.tr (Fatma Yerlikaya-Özkurt), ceydayazici86@gmail.com (Ceyda Yazıcı), ibatmaz@metu.edu.tr (İnci Batmaz).

https://doi.org/10.1016/j.softx.2023.101553
Received 31 July 2023; Received in revised form 2 October 2023; Accepted 6 October 2023
Available online 19 October 2023

Table 1
The CMARS algorithm in pseudo code.

```
Step 1: Forward step
Require: Maximum number of BFs $\left(M_{max}\right)$; degree of interaction
Initialize the first BF to one, $\psi_1(\boldsymbol{x}^1) = 1$ and the set of BFs, $\wp = \left\{\psi_1\right\}$
1 repeat
2 Initialize current minimum lack of fit (lof) value, $lof_{min}$, to infinity
3 for each BF $M = 2$ to $M = M_{max}$
4    for each not covered variable index, $\kappa_j^M$, $j = 1$ to $p$
5        for each possible knot location, $\tau_{\kappa_j^M} \notin \left\{x_{\kappa_j^M} \mid \psi_m\left(x_{\kappa_j^M}\right) > 0\right\}$
6            compute $lof$:
$\min_\theta \left(\sum_{m=1}^{M-1} \theta_m \psi_m(\boldsymbol{x}^m) + \theta_M \psi_m(\boldsymbol{x}^m)\left[x_{\kappa_j^M} - \tau_{\kappa_j^M}\right]_+ + \theta_{M+1}\psi_m(\boldsymbol{x}^m)\left[\tau_{\kappa_j^M} - x_{\kappa_j^M}\right]_+\right)$,
7            if $lof < lof_{min}$ then $lof_{min} = lof$; $m^* = m$; $\kappa_j^{m^*} = \kappa_j^m$; $\tau_{\kappa_j^{m^*}} = \tau_{\kappa_j^m}$
8        end for
9    end for
10      create a new pair of BFs as in Eq. (3)
$\psi_M(\boldsymbol{x}^M) \leftarrow \psi_{m^*}(\boldsymbol{x}^{m^*})\left[x_{\kappa_j^{m^*}} - \tau_{\kappa_j^{m^*}}\right]_+$, $\psi_{M+1}(\boldsymbol{x}^{M+1}) \leftarrow \psi_{m^*}(\boldsymbol{x}^{m^*})\left[\tau_{\kappa_j^{m^*}} - x_{\kappa_j^{m^*}}\right]_+$
11      append them to the set of BFs, $\wp := \wp \cup \left\{\psi_M, \psi_{M+1}\right\}$
12 end for
13 until $M > M_{max}$
Step 2: Backward step
Require: A set of upper bound values, $Z$, for Eq. (6)
14 for each upper bound value, $z$, in $Z$
15     for each existed element in $\wp$ formed in Step 1, line 11
16         for the order of derivative, $|\boldsymbol{\alpha}|$, one and two
17             for each predictor associated with the BF given in $\wp$
18                 take the partial derivatives, $D_{r,s}^{\boldsymbol{\alpha}}\psi_m(\boldsymbol{t}^m) := \frac{\partial^{|\boldsymbol{\alpha}|}\psi_m}{\partial^{\alpha_1} t_r^m \, \partial^{\alpha_2} t_s^m}(\boldsymbol{t}^m)$
19             end for
20         end for
21         for all observations 1 to $N$
22             calculate the diagonal elements of $\boldsymbol{L}$ matrix in Eq. (5)
23         end for
24     end for
25     Prepare: constraints $\left\|\boldsymbol{y} - \boldsymbol{\psi}(\tilde{\boldsymbol{d}})\boldsymbol{\theta}\right\|_2$ and $\|\boldsymbol{L\theta}\|_2$ in Eq. (6)
26     solve the CQP problem in Eq. (6) to obtain an estimate of unknown parameters
27 end for
28 compute $\left\|\boldsymbol{y} - \boldsymbol{\psi}(\tilde{\boldsymbol{d}})\boldsymbol{\theta}\right\|_2$ and $\|\boldsymbol{L\theta}\|_2$ in Eq. (6)
```

e.g. size and scale, obtained from a well-known machine learning data repository University of California at Irvine (UCI) [7,9]. Here, the method-free measures are used to evaluate the performances of the models developed according to the following criteria: accuracy, complexity, stability, robustness and efficiency. Moreover, performances of the two methods on noisy data are also evaluated by a Monte Carlo simulation study. Results of the study reveal that, in general, CMARS produces more accurate, robust and stable models than MARS method, highlighting its superiority in various aspects, including use of higher order interaction and modeling different data structure. Besides, CMARS overperforms mostly on large-size and -scale data sets as well as on a simulated noisy data set. Furthermore, CMARS' performance is also compared to those of CART, generalize additive models (GAMs), infinite kernel learning (IKL), random forest (RF), ANN, SVM, and GLMs by using several real-life and simulation data sets having different characteristics [10–17]. All of these comparative studies indicate that CMARS is a powerful predictive DM method and a strong alternative to the others [18]. These results encourage us to develop the cmaRs package in R to make it open to the interested DM researchers freely to let them benefit from its superior prediction characteristics.

2. Software description

2.1. The CMARS algorithm

In nonparametric regression, the form of prediction model is given as

$$y = f(\boldsymbol{x}) + \epsilon, \tag{1}$$

where y is the response; $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)^T$ is a vector of predictors; ϵ is the error term with mean zero and finite variance. When the MARS model is of concern, $f(\boldsymbol{x})$ in Eq. (1) is represented as a linear combination of the intercept and the BFs as follows [6,7]

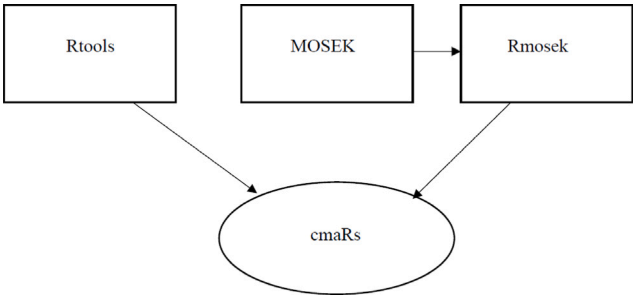

Fig. 1. A flowchart for the installation of cmaRs package for Windows.

$$y = \theta_0 + \sum_{m=1}^{M_{max}} \theta_m \psi_m(\boldsymbol{x}^m) + \epsilon. \tag{2}$$

Here, θ_m $(m = 1, 2, \ldots, M_{max})$ is an unknown coefficient of the mth BF; $\boldsymbol{x}^m = (x_1, x_2, \ldots, x_p)^T$ is the predictor vector of the mth BF; ψ_m is the mth BF defined as

$$\psi_m(\boldsymbol{x}^m) := \prod_{j=1}^{K_m} [s_{\kappa_j^m} \cdot (x_{\kappa_j^m} - \tau_{\kappa_j^m})]_+, \tag{3}$$

where $[q]_+ := \max\{0, q\}$; K_m is the number of truncated linear functions; $x_{\kappa_j^m}$ is the input variable corresponding to the jth truncated linear function; $\tau_{\kappa_j^m}$ is the knot value corresponding to the variable $x_{\kappa_j^m}$; $s_{\kappa_j^m}$ is +1 or −1 [5]. The largest set of BFs that may lead to overfitting is collected as a result of the forward step of MARS algorithm. To find optimal MARS model, in a backward step, the BFs with the

Table 2
Input to `cmaRs` function.

Input	Default	Definition
`formula`	–	Define dependent and predictor variables
`data`	–	Define data frame
`classification`	FALSE	Construct a model for prediction or binary classification when it is TRUE
`threshold.class`	0.5	Classify predicted values for binary classification
`degree`	1	Specify highest degree of interactions
`nk`	20	Specify maximum number of BFs (i.e. M_{max})
`Auto.linpreds`	FALSE	Define predictor in a truncated linear function form given in Eq. (3)

Table 3
Functions of `cmaRs` package.

Function	Called by	Definition
`cmaRs`	Main function	Creates a `cmaRs` object with the input in Table 2
`cmaRs.fit`	`cmaRs`	Prepare data for CMARS modeling, create BFs, construct CMARS model and calculates performance measures
`derivative_one`	`cmaRs.fit`	Take derivative of main effect BFs with respect to the predictor and assign it to symbolic derivative matrix (DMS)
`derivative_more_than_one`	`cmaRs.fit`	Take partial derivatives of interaction effects BFs with respect to predictors and assign it to DMS matrix
`identical_function`	`cmaRs.fit`	Prepare DMS matrix for evaluation
`mosek_optimization`	`cmaRs.fit`	Based on `Rmosek` [19,20] estimate CMARS model parameters for different z values

lowest contribution to the model are removed by using generalized cross-validation (GCV) criteria given in Eq. (4).

$$GCV = \frac{\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f(\boldsymbol{x}_i)\right)^2}{\left(1 - \frac{M_{max}+\vartheta\times(M_{max}-1)/2}{N}\right)^2}, \tag{4}$$

in which ϑ is the penalizing parameter, N is the number of observations and $(M_{max} - 1)/2$ is the number of BF knots [21,22].

CMARS, a relatively new nonparametric regression method, is based on the first part of the MARS algorithm [5]. In backward step, a Tikhonov regularization problem is formed as follows [7,23]

$$\left\|\boldsymbol{y} - \boldsymbol{\psi}(\tilde{\boldsymbol{d}})\boldsymbol{\theta}\right\|_2^2 + \lambda \left\|\boldsymbol{L\theta}\right\|_2^2. \tag{5}$$

Here, $\boldsymbol{L}$ is a diagonal $(M_{max} + 1) \times (M_{max} + 1)$ matrix that based on the values of first and second partial derivatives of model functions. Noted that this problem is based on the tradeoff between both *accuracy*, that is, a small sum of error squares, and not too high *complexity*. This tradeoff is established through the penalty parameter λ. This is a *continuous* and *convex* optimization problem, and solved by the CQP as follows [7,24]

$$\begin{aligned} &\min_{t,\theta} && t, \\ &\text{subject to} && \left\|\boldsymbol{y} - \boldsymbol{\psi}(\tilde{\boldsymbol{d}})\boldsymbol{\theta}\right\|_2 \le t, \\ & && \left\|\boldsymbol{L\theta}\right\|_2 \le z. \end{aligned} \tag{6}$$

Multiple solutions are obtained for Z consisting of different z values determined based on the data set, by using IPMs [8]. Note also that since the second constraint in Eq. (6) is for smoothness and does not give completely different results for different z values, the first constraint dominates in construction of the final model. Indeed, accuracy of the predictive models is important and critical as it determines the quality of the predictions. Therefore, predictive accuracy is of the main interest for decision-making. Note here that MARS provides the same tradeoff in using GCV measure. Tikhonov regularization problem solved by the CQP, however, also considers the stability, in other words, robustness or well-conditionedness, leading CMARS to give more stable and robust results. The CMARS algorithm, consisting of two steps, is presented in pseudo code in Table 1.

2.2. Installation of cmaRs

In order to construct a CMARS model by using the `cmaRs` package, the packages listed in the dependencies part of the metadata table should be made readily available in the R environment. The `cmaRs` package developed here is a good example for a challenging implementation of CQP based on MOSEK solver in R environment by linking R to MOSEK [19] through the package `Rmosek`. Because `Rmosek` is used as an interface to MOSEK, here, it needs special attention to start the installation without errors [25]. A flowchart for the installation of `cmaRs` package for Windows is given in Fig. 1. Please visit https://docs.mosek.com/latest/rmosek/install-interface.html for other systems and detail of the installation.

2.3. Software functionalities

`cmaRs` is a package written in R language to construct CMARS models for prediction and binary classification. It needs several inputs to be provided by the user (Table 2). After defining the inputs, the `cmaRs` object of type S3 in R, is created with object-oriented programming approach to construct a CMARS model through the main function, `cmaRs`, which has the same name with the package, and then, other functions are defined on top of it (Table 3). In addition, this package includes several methods and accessors operating on `cmaRs` object to provide end users specific output on demand such as BFs, parameter estimates, fitted values, $\mathbf{L}$ matrix, Symbolic Derivative Matrix (DMS), Symbolic Variable Name Matrix (VARMS) and z value of the final model (see Table 4 for complete list). Here, DMS contains derivatives of a given expression with respect to a specified variable, obtained by using a symbolic differentiation program (`D`) which operates on expression trees or forests to produce new formulas [26]. On the other side, VARMS contains variable names whose derivatives exist in the DMS. Note that $\mathbf{L}$, DMS, VARMS matrices mentioned are low-level functions created in the implementation of the algorithm, which may be useful for the interested researches to improve the `cmaRs` package further (see Section 4). In the next section, some code snippets are presented to illustrate the use of them. For more detailed examples one can refer to the vignette of the package [27]. Ultimately, the final model is constructed and related performance measures shown in Table 4 are calculated and presented.

Table 4
Methods and accessors of `cmaRs` package with their output for the end users.

Model	Method	Output	Accessors
Prediction	Summary(obj)	Residual Sum of Squares (RSS)	obj$RSS
		Correlation between actual and predicted response values (r)	obj$r
		Multiple coefficient of determination (R^2)	obj$R2
	Plot(obj)	Scatterplot of actual versus predicted response values	–
		scatterplot of actual response values versus order of the observations	–
		Scatterplot of actual response values versus standardized residuals	–
	Predict(obj, data)	Fitted values of the prediction model for given data	obj$fitted.values
Binary classification	Summary(obj)	Misclassification Rate (MCR)	obj$MCR
		Percentage of Correct Classification (PCC)	obj$PCC
		Precision	obj$precision
		Recall	obj$recall
		Specificity	obj$specificity
		Area Under ROC Curve (AUC)	obj$AUC
	Plot(obj)	ROC curve: a plot of Recall (true positive rate) versus (1-Specificity: false positive rate) at different classification thresholds	–
	Predict(obj, data)	Probabilities of the fitted binary classification models	obj$fitted.values
Prediction/	No method	BFs of fitted model	obj$bf.cmars
Binary classification	Available	Parameter estimates of fitted model	obj$coefficients
		L matrix	obj$L
		DMS matrix	obj$DMS
		Symbolic variable name matrix (VARMS) whose derivative exists in DMS matrix	obj$VARMS
		TRUE for a binary classification model; FALSE otherwise	obj$classification
		Number of BFs in model	obj$number.of.BF
		$\sqrt{z}$ values	obj$final.sqrtz
		Name of response	obj$response.name
		Ordinary residuals	obj$residuals
		Response values	obj$y

3. Illustrative examples

3.1. An application for prediction

To exemplify the use of `cmaRs` package for prediction, a real-life data set, Concrete Slump Test, is selected from the machine learning repository UCI [9]. One of the output variables (28-day Compressive Strength) and seven input variables are used to build the predictive CMARS model. For this purpose, data is first standardized to prevent different scale effects on the results (`preddata.std` in `cmaRs` package). To give some information regarding the data set, correlation matrix of all attributes (Table 5) and distribution of the output variable (Fig. 2) are obtained. Response variable is slightly right skewed and input variables are not severely correlated at all (correlations ≤ 0.6). Next, to evaluate performance of the model, it is split into train (80%) and test (20%) data sets randomly (Listing 1, lines 1–5). In this section, refer to Listing 1 when code lines mentioned. Then, the CMARS model is fit to the train data (Line 6). Note that the best model obtained as a result of selected configurations and setting the upper bound value of the CQP problem z to 24.3 is printed (Line 7). In the package, z values are defined in the range [0.1, 100], which satisfies the needs of many problems. Both configuration parameters, `degree` and `nk`, and the upper bound value z, are determined by trial and error approach after several plausible attempts.

Once the prediction model is built, some measures for its performance are automatically listed as the output of the `cmaRs` package (Table 4). Besides, it can also produce three different scatter plots at once for train (Line 10, Fig. 3). Moreover, the same for the test data is given in Fig. 4. Results show that the model developed performs quite well for the given data set (Table 6). Last but not least, the fitted values (Line 8) and the test data predictions (Line 9) can also be calculated.

Furthermore, all of the BFs in the CMARS model, the corresponding DMS, VARMS and **L** matrices, fitted values and the parameter estimates can simply be obtained on demand by calling relevant accessors on the object created (see Lines 11–14 for illustrative examples). Note however that DMS and VARMS matrices are automatically created only for the last BF of the model. To obtain these matrices for the other BFs, some modifications are needed in the code (see last Section in the vignette) [27].

Depending on the successful results of the model developed, later, one can combine train and test data sets and remodel it to obtain the final predictive model.

```
1 > library(caret) # install package caret to
    be able to use createDataPartition
    function
2 > set.seed(222) # set a seed number to be
    able to reproduce the same data set
3 > training.samples <- createDataPartition(
    preddata.std$Compressive_Strength, p =
    0.8, list = FALSE) # generate random
    indices for creating train data set
4 > train.dat <- preddata.std[training.samples,
     ] # create train data set
5 > test.dat <- preddata.std[-training.samples,
     ] # create test data set
```

Table 5
The correlation matrix of all attributes for Concrete Slump Test data set.

	Cement	Slag	Fly_ash	Water	SP	Coarse_Aggr	Fine_Aggr	Compressive_Strength
Cement	1	−0.2436	−0.4865	0.2211	−0.1064	−0.3099	0.0570	0.4457
Slag	−0.2436	1	−0.3226	−0.0268	0.3065	−0.2238	−0.1835	−0.3316
Fly_ash	−0.4865	−0.3226	1	−0.2413	−0.1435	0.1726	−0.2829	0.4444
Water	0.2211	−0.0268	−0.2413	1	−0.1555	−0.6022	0.1146	−0.2542
SP	−0.1064	0.3065	−0.1435	−0.1555	1	−0.1042	0.0583	−0.0379
Coarse_Aggr	−0.3099	−0.2238	0.1726	−0.6022	−0.1042	1	−0.4885	−0.1607
Fine_Aggr	0.0570	−0.1835	−0.2829	0.1146	0.0583	−0.4885	1	−0.1545
Compressive_Strength	0.4457	−0.3316	0.4444	−0.2542	−0.0379	−0.1607	−0.1545	1

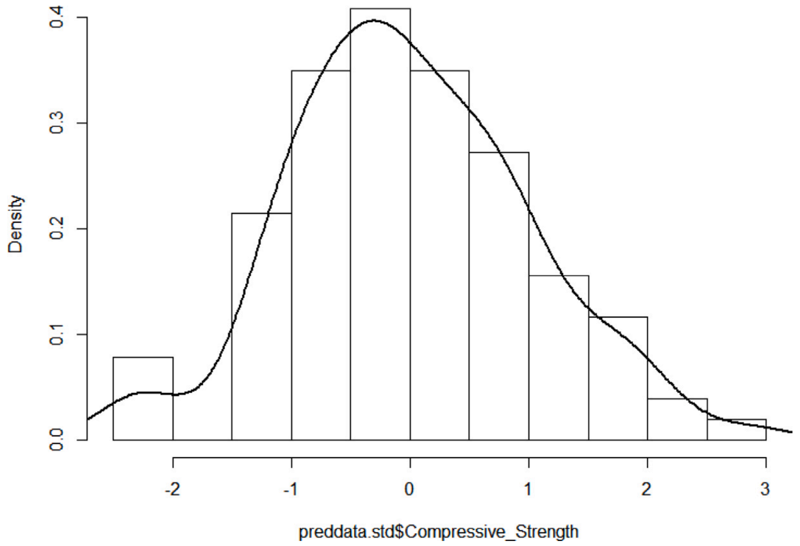

Fig. 2. Distribution of the output variable of Concrete Slump Test data.

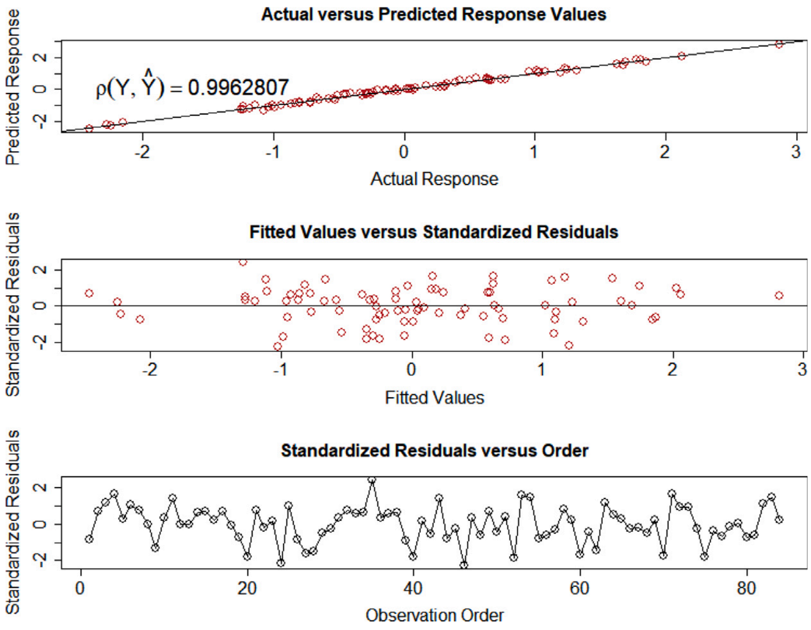

Fig. 3. Scatter plots of predictive CMARS model developed using train data.

```
> cmars.mod.pred <- cmaRs(Compressive_
   Strength(*@\asciitilde@*)., degree = 3,
   nk = 28, Auto.linpreds = FALSE, data =
   train.dat)
```

```
> summary(cmars.mod.pred)
> predict(cmars.mod.pred, train.dat)
> predict(cmars.mod.pred, test.dat)
> plot(cmars.mod.pred)
```

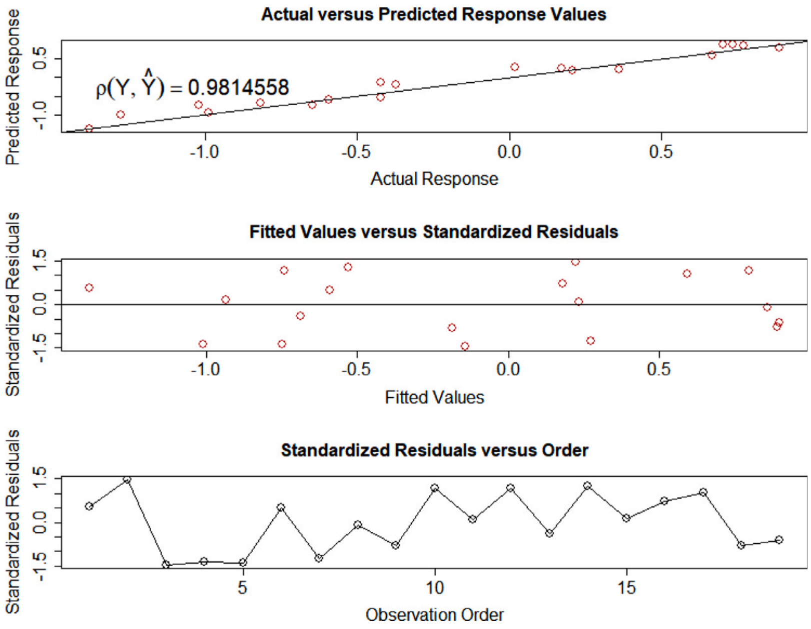

Fig. 4. Scatter plots of predictive CMARS model developed using test data.

Table 6
Performance of CMARS models developed in illustrative examples.

Performance measures	Prediction model			Binary classification model					
	R^2	r	RSS	AUC	MCR	PCC	Precision	Recall	Specificity
Train	0.9926	0.9963	0.6778	0.9742	0.0987	0.9013	0.9547	0.8846	0.9294
Test	0.9547	0.9814	0.4706	0.9959	0.0354	0.9646	1.0000	0.9437	1.0000

```
11 > cmars.mod.pred$bf.cmars
12 > cmars.mod.pred$DMS
13 > cmars.mod.pred$VARMS
14 > cmars.mod.pred$L
```

Listing 1: R snippets for CMARS prediction

3.2. An application for binary classification

Another example is carried out for binary classification DM function, and CMARS is applied to the Breast Cancer Wisconsin (Diagnostic) data set [9] to diagnose if breast cancer is malignant or benign. Again, all predictors are standardized first (`classdata.std` in the `cmaRs` package). Next, only one of the highly correlated predictors, namely `x6, x8, x11, x14, x15, x17, x21, x22, x24, x27, x28, x29, x30`, are retained in the data set to develop valid models. The correlation matrix of selected input variables (Table 7) and class distribution of the binary response variable are obtained as below (Listing 2). Thus, the model for the standardized data with the selected predictors (Listing 3, Lines 1–5) is constructed. In this section, refer to Listing 3, when the code snippet lines are mentioned.

```
> Frequency = table(datanew$y)
> Percentage = round(percent.table(datanew$y)
  , 2)
> Table = rbind(Frequency, Percentage)
> Table
         0     1
Frequency  357.00  212.00
Percentage  62.74  37.26
```

Listing 2: R snippets for running class distribution of the binary response variable

After this stage, the data set is split as above for model validation (Lines 6–11), and the CMARS model is obtained on the train data set (Line 12). For the best model, `degree` and `nk` are defined as 4 and 10, respectively. Here, the assignment `classification` = TRUE indicates that we attempt to construct a binary classification model. Besides, the upper bound value of the CQP for the corresponding CMARS model is determined to be $z = 1$, which gives the maximum value for the AUC measure. Also, the model displayed contains main and up to three degrees of interaction effects. Eventually, the performance of the CMARS model built can be evaluated for the default threshold value (i.e. 0.5) by the following well-known classification performance measures automatically provided by the `cmaRs` package (Line 13) (Table 4). It is however possible to convert the fitted values to classes by redefining the threshold value, e.g. 0.25 (Line 15). The best CMARS model obtained for binary classification application is given in Listing 4. Moreover, to illustrate the performance of the model, the ROC curve can also be drawn (Line 14, Fig. 5). In addition to this, the same plot for test data is also given in Fig. 6.

Results indicate that classification capability of the CMARS model built is very good, and the model captures the main structure of data well. Furthermore, all of the BFs in the CMARS model, the corresponding DMS, VARMS and **L** matrices, fitted values and the model parameter estimates can simply be obtained on demand by calling on the object created (see Lines 16–17 for illustrative examples). In the end, train and test data sets can be merged and refit to obtain ultimate CMARS model for classification of malignant or benign breast cancer.

```
1 > library(caret) # install package caret to
    be able to use createDataPartition
    function
2 > colnr <- c(6, 8, 11, 14, 15, 17, 21, 22,
    24, 27, 28, 29, 30) # create a vector of
    predictor numbers to be used
3 > whole_colnr <- c(1, colnr + 1) # create a
    vector of column numbers to be used
```

Table 7
The correlation matrix of selected input variables for Breast Cancer Wisconsin data set.

	x6	x8	x11	x14	x15	x17	x21	x22	x24	x27	x28	x29	x30
x6	1	0.8311	0.4975	0.4557	0.1353	0.5705	0.5353	0.2481	0.5096	0.8163	0.8156	0.5102	0.6874
x8	0.8311	1	0.698	0.6903	0.0277	0.4392	0.8303	0.2928	0.8096	0.7524	0.9102	0.3757	0.3687
x11	0.4975	0.698	1	0.9518	0.1645	0.3324	0.7151	0.1948	0.7515	0.3806	0.5311	0.0945	0.0496
x14	0.4557	0.6903	0.9518	1	0.0752	0.2709	0.7574	0.1965	0.8114	0.3851	0.5382	0.0741	0.0175
x15	0.1353	0.0277	0.1645	0.0752	1	0.2687	−0.2307	−0.0747	−0.1822	−0.0583	−0.1020	−0.1073	0.1015
x17	0.5705	0.4392	0.3324	0.2709	0.2687	1	0.1869	0.1002	0.1884	0.6626	0.4405	0.1978	0.4393
x21	0.5353	0.8303	0.7151	0.7574	−0.2307	0.1869	1	0.3599	0.9840	0.5740	0.7874	0.2435	0.0935
x22	0.2481	0.2928	0.1948	0.1965	−0.0747	0.1002	0.3599	1	0.3458	0.3684	0.3598	0.2330	0.2191
x24	0.5096	0.8096	0.7515	0.8114	−0.1822	0.1884	0.9840	0.3458	1	0.5433	0.7474	0.2091	0.0796
x27	0.8163	0.7524	0.3806	0.3851	−0.0583	0.6626	0.5740	0.3684	0.5433	1	0.8554	0.5325	0.6865
x28	0.8156	0.9102	0.5311	0.5382	−0.1020	0.4405	0.7874	0.3598	0.7474	0.8554	1	0.5025	0.5111
x29	0.5102	0.3757	0.0945	0.0741	−0.1073	0.1978	0.2435	0.2330	0.2091	0.5325	0.5025	1	0.5378
x30	0.6874	0.3687	0.0496	0.0175	0.1015	0.4393	0.0935	0.2191	0.0796	0.6865	0.5111	0.5378	1

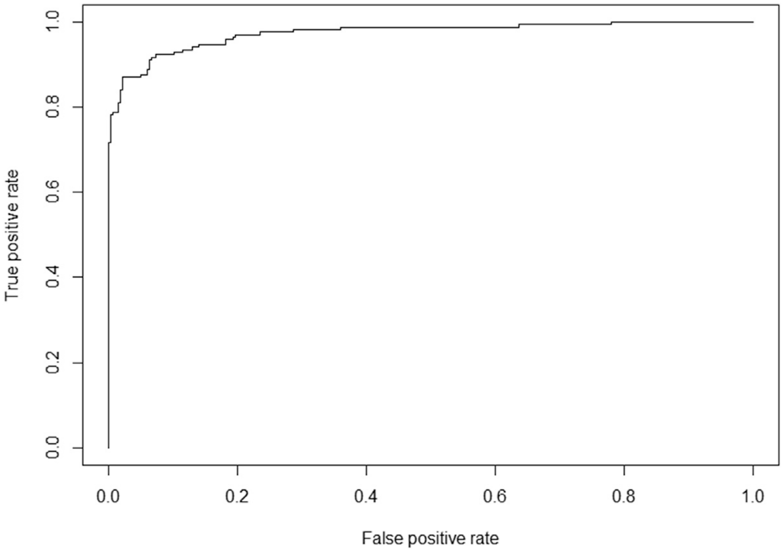

Fig. 5. ROC curve for binary classification CMARS model developed using train data.

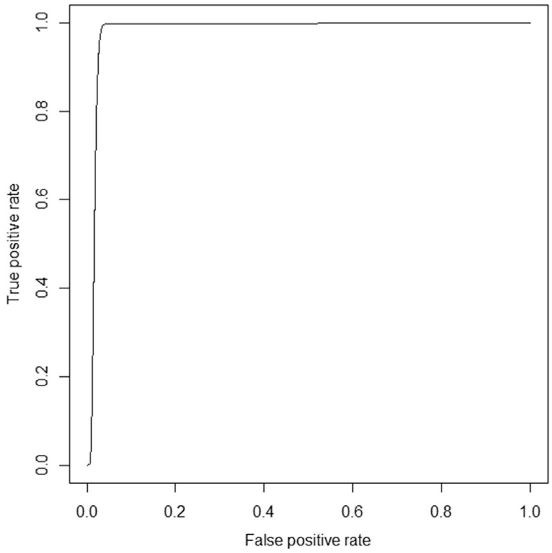

Fig. 6. ROC curve for binary classification CMARS model developed using test data.

```
> datanew <- classdata.std[, whole_colnr] #
    create a new data set containing
    independent predictors only
```

```
> datanew$y <- as.factor(datanew$y) # convert
    response column in the new data set from
    numeric to factor
```

```
> set.seed(20) # set a seed number to be able
   to reproduce the same data set
> training.samples <- createDataPartition(
  datanew$y, p = 0.8, list = FALSE) #
  generate random indices for creating
  train data set
> train.dat  <- datanew[training.samples, ] #
   create train data set
> test.dat <- datanew[-training.samples, ] #
  create test data set
> train.dat$y <- as.integer(as.character(
  train.dat$y)) # return train data
  response value in character object as
  integer object
> test.dat$y <- as.integer(as.character(test.
  dat$y)) # return test data response value
   in character object as integer object
> cmars.mod.clss <- cmaRs(formula = y (*@\
  asciitilde@*) ., data = train.dat,
  classification = TRUE, degree = 4, nk =
  10, Auto.linpreds = FALSE)
> summary(cmars.mod.clss)
> plot(cmars.mod.clss)
> cmars.mod.clss <- cmaRs(formula = y (*@\
  asciitilde@*) ., data = train.dat,
  classification = TRUE, degree = 4, nk =
  10, threshold.class = 0.25, Auto.linpreds
   = FALSE)
> cmars.mod.clss$fitted.values
> cmars.mod.clss$coefficients
```

Listing 3: R snippets for CMARS binary classification

```
y = +0.9265
+0.038 * pmax(0,x24-0.771777)
-0.6864 * pmax(0,0.771777-x24)
+0.1048 * pmax(0,0.771777-x24)*
          pmax(0,x28-0.121611)
-0.0671 * pmax(0,0.771777-x24)*
          pmax(0,0.121611-x28)
+0.1678 * pmax(0,x21+0.0101774)*
          pmax(0,0.771777-x24)
+0.1454 * pmax(0,-0.0101774-x21)*
          pmax(0,0.771777-x24)
+1.0821 * pmax(0,x21+0.0101774)*
          pmax(0,x22+0.640589)*
          pmax(0,0.771777-x24)
-12.5033 * pmax(0,x21+0.0101774)*
           pmax(0,-0.640589-x22)*
           pmax(0,0.771777-x24)
```

Listing 4: The best CMARS model for binary classification application

4. Impact

CMARS has successfully been applied in diverse fields from health, finance, energy, geology to meteorology [18]. Some examples include predicting credit default [11], modeling precipitation [28] and atmospheric effects on satellite images [29], diagnosing Alzheimer's disease [30], inpainting [16], predicting earthquake [31], forecasting natural gas consumption [32–34]. In spite of its dazzling prediction properties and remarkable success in applications and comparative studies, it could not be able to make a splash it deserves mainly due to difficulties in coding the CMARS algorithm. To the best of our knowledge, the cmaRs package is the first publicly available software enabling implementation of the CMARS method. CRAN statistics show that cmaRs was downloaded 10012 times between 2-10-2023 and 23-02-2021 when the first version (0.1.0) is uploaded. With easy accessibility of this package, we are sure that it will be utilized much more by the DM researchers [2–4,35–38].

On the other hand, cmaRs package introduced here includes only basic features of the CMARS method. However, it has already been improved further to cover several other advanced features as well. To illustrate, BCMARS algorithm is developed to reduce the complexity of the CMARS models by using bootstrapping technique [39]. RCMARS, on the other hand, enables modeling random variables [35]. Yet, another one, SCMARS, is developed to reduce the runtime of CMARS algorithm by using self-organizing maps [40]. In addition, it is also adapted into a wide frame of advanced methods of statistics and applied mathematics [41–43]. The cmaRs package can also be further upgraded such that the new versions can functionate these advanced abilities on demand. Note that one can refer to the vignette of cmaRs package to activate functions useful for code developers.

5. Conclusions

DM is the process of capturing patterns which may exist in data collected from various sources. CMARS method developed recently is based on the MARS algorithm by using Tikhonov regularization and CQP. It is a very successful statistical learning method, especially, for extracting nonlinear structures in high-dimensional data. Several studies have been conducted to evaluate and compare its performance. Results indicate that it is a powerful alternative, especially, when the prediction accuracy and/or large size and scale data sets are of the main interest. Depending on its apparent achievement, we aim to make the cmaRs package freely available to the interested DM researchers to let them take the advantage of its remarkable features. In this paper, we introduce the package cmaRs, where the CMARS algorithm is implemented, in both pseudo and R code with illustrative examples.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

https://github.com/yaziciceyda/cmaRs, http://archive.ics.uci.edu/ml.

References

[1] Batmaz I, Köksal G. Overview of knowledge discovery in databases process and data mining for surveillance technologies and ews. In: Surveillance Technologies and Early Warning Systems: Data Mining Applications for Risk Detection. PA: IGI Global Publisher (Idea Group Publisher); 2011, p. 1–30.

[2] Liu S, Wang L, Zhang W, Sun W, Fu J, Xiao T, et al. A physics-informed data-driven model for landslide susceptibility assessment in the three gorges reservoir area. Geosci Front 2023;14(5):101621.

[3] Phoon K-K, Zhang W. Future of machine learning in geotechnics. Georisk: Assess Manag Risk Eng Syst Geohazards 2023;17(1):7–22.

[4] Zhang W, Gu X, Tang L, Yin Y, Liu D, Zhang Y. Application of machine learning, deep learning and optimization algorithms in geoengineering and geoscience: Comprehensive review and future challenge. Gondwana Res 2022;109:1–17.

[5] Friedman JH. Multivariate adaptive regression splines. Ann Statist 1991;19(1):1–141.

[6] Yerlikaya F. A new contribution to nonlinear robust regression and classification with MARS and its application to data mining for quality control in manufacturing. (Thesis (MSc)), Turkey: Middle East Technical University; 2008.

[7] Weber GW, Batmaz İ, Köksal G, Taylan P, Yerlikaya-Özkurt F. CMARS: A new contribution to nonparametric regression with multivariate adaptive regression splines supported by continuous optimization. Inverse Probl Sci Eng 2012;20(3):371–400.

[8] Nesterov YE, Nemirovski AS. Interior-point polynomial algorithms in convex programming. Philadelphia: Society for Industrial and Applied Mathematics; 1994.

[9] Dua D, Graff C. UCI machine learning repository. 2017, URL http://archive.ics.uci.edu/ml.

[10] Batmaz İ, Yerlikaya-Özkurt F, Kartal-Koç E, Köksal G, Weber GW. Evaluating the CMARS performance for modeling nonlinearities. In: Proceedings of the 3rd global conference on power control and optimization. Vol. 1239, 2010, p. 351–7.

[11] Sezgin-Alp Ö, Büyükbebeci E, Işcanoğlu-Çekiç A, Yerlikaya-Özkurt F, Taylan P, Weber GW. -CMARS and GAM and CQP- modern optimization methods applied to international credit default prediction. J Comput Appl Math 2011;235:4639–51.

[12] Yerlikaya-Özkurt F, Askan A, Weber GW. An alternative approach to the ground motion prediction problem by a non-parametric adaptive regression method. Eng Optim 2014;46(12):1651–68.

[13] Özmen A, Kropat E, Weber GW. Spline regression models for complex multi-modal regulatory networks. Optim Methods Softw 2014;29(3):515–34.

[14] Kuter S, Weber GW, Akyürek Z, Özmen A. Inversion of top of atmospheric reflectance values by conic multivariate adaptive regression splines. Inverse Probl Sci Eng 2015;23(4):651–69.

[15] Özmen A, Yılmaz Y, Weber GW. Natural gas consumption forecast with MARS and CMARS models for residential users. Energy Econ 2018;70:357–81.

[16] Kurt C. Comparison of computational inpainting methods. (Thesis (MSc)), Turkey: Middle East Technical University; 2018.

[17] Altinok G, Karagoz P, Batmaz I. Learning to rank by using multivariate adaptive regression splines and conic multivariate adaptive regression splines. Comput Intell 2021;37(1):371–408.

[18] Yerlikaya-Özkurt F, Batmaz İ, Weber GW. Springer volume modeling, optimization, dynamics and bioeconomy, series springer proceedings in mathematics. Springer; 2013, Ch. A Review and New Contribution on Conic Multivariate Adaptive Regression Splines (CMARS): A Powerful Tool for Predictive Data Mining.

[19] MOSEK-ApS. The MOSEK optimization tools manual. 2011, http://www.mosek.com/.

[20] MOSEK-ApS. Rmosek: The R to MOSEK optimization interface. 2017, http://rmosek.r-forge.r-project.org/, http://www.mosek.com/.

[21] Zhang W, Goh A. Multivariate adaptive regression splines for analysis of geotechnical engineering systems. Comput Geotech 2013;48:82–95.

[22] Zhang W, Li H, Wu C, Li Y, Liu Z, Liu H. Soft computing approach for prediction of surface settlement induced by earth pressure balance shield tunneling. Undergr Space 2021;6(4):353–63.

[23] Aster RC, Borchers B, Thurber C. Parameter estimation and inverse problems. Burlington: Academic Press; 2012.

[24] Ben-Tal A, Nemirovski A. Lectures on modern convex optimization: analysis, algorithms and engineering applications. Philadelphia: SIAM; 2001.

[25] Installation of MOSEK Rmosek package, URL https://docs.mosek.com/latest/rmosek/install-interface.html.

[26] Tolsma JE, Barton PI. On computational differentiation. Comput Chem Eng 1998;22(4):475–90.

[27] cmaRs package, URL https://github.com/cran/cmaRs.

[28] Özmen A, Batmaz İ, Weber G-W. Precipitation modeling by polyhedral RCMARS and comparison with MARS and CMARS. Environ Model Assess 2014;19:425–35.

[29] Kuter S, Weber GW, Özmen A, Akyürek Z. Modern applied mathematics for alternative modeling of the atmospheric effects on satellite images. In: Modeling, dynamics, optimization and bioeconomics I: Contributions from ICMOD 2010 and the 5th bioeconomy conference 2012. Springer; 2014, p. 469–85.

[30] Çevik A. Computer-aided diagnosis of alzheimer's disease and mild cognitive impairment with MARS/CMARS classification using structural MR images. (Thesis (PhD)), Turkey: Middle East Technical University; 2017.

[31] Priyanto D, Zarlis M, Mawengkang H, Efendi S. Approach of analysis of data mining prediction in earthquake case using non parametric adaptive regression method. 2020.

[32] Özmen A, Yılmaz Y, Weber G-W. Natural gas consumption forecast with MARS and CMARS models for residential users. Energy Econ 2018;70:357–81.

[33] Özmen A, Zinchenko Y, Weber G-W. Robust multivariate adaptive regression splines under cross-polytope uncertainty: an application in a natural gas market. Ann Oper Res 2022;1–31.

[34] Ozmen A. Multi-objective regression modeling for natural gas prediction with ridge regression and CMARS. Int J Optim Control: Theories Appl (IJOCTA) 2022;12(1):56–65.

[35] Özmen A, Weber GW, Batmaz İ, Kropat E. RCMARS: Robustification of CMARS with different scenarios under polyhedral uncertainty set. Commun Nonlinear Sci Numer Simul (CNSNS): Nonlinear Fract Complex 2011;16:4780–7.

[36] Nalcaci G, Özmen A, Weber GW. Long-term load forecasting: models based on MARS, ANN and LR methods. CEJOR Cent Eur J Oper Res 2019;27:1033–49.

[37] Graczyk-Kucharska M, Szafrański M, Gütmen S, Goliński M, Spychała M, Weber G-W, et al. Modeling for human resources management by data mining, analytics and artificial intelligence in the logistics departments. In: Smart and sustainable supply chain and logistics–trends, challenges, methods and best practices: Volume 1. Springer; 2020, p. 291–303.

[38] Ewertowski T, Güldoğuş BÇ, Kuter S, Akyüz S, Weber G-W, Sadłowska-Wrzesińska J, et al. The use of machine learning techniques for assessing the potential of organizational resilience. CEJOR Cent Eur J Oper Res 2023;1–26.

[39] Yazıcı C, Yerlikaya-Özkurt F, Batmaz İ. A computational approach to nonparametric regression: bootstrapping CMARS method. Mach Learn 2015;101:211–30.

[40] Kartal-Koc E, Iyigün C, Batmaz İ, Weber G-W. Efficient adaptive regression spline algorithms based on mapping approach with a case study on finance. J Global Optim 2014;60:103–20.

[41] Yerlikaya-Özkurt F. Refinements, extensions and modern applications of conic multivariate adaptive regression splines (Ph.D. thesis), Turkey: Middle East Technical University; 2013.

[42] Yerlikaya-Özkurt F, Vardar-Acar C, Yolcu-Okur Y, Weber GW. Estimation of the hurst parameter for fractional Brownian motion using the CMARS method. J Comput Appl Math 2014;259(PART B):843–50.

[43] Yerlikaya-Özkurt F, Askan A, Weber GW. A hybrid computational method based on convex optimization for outlier problems: Application to earthquake ground motion prediction. Informatica (Netherlands) 2016;27(4):893–910.