Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 9, No. 3, pp. 535–552, July 2013

A NEW ROBUST OPTIMIZATION TOOL APPLIED ON FINANCIAL DATA

A.Özmen*, G.-W. Weber and A. Karimov

Abstract: Recent financial crises, with an increased volatility and, hence, uncertainty factors, have introduced a high "noise" into the data taken from the financial sectors and overall from any data related to the financial markets, so that the known statistical models do not give trustworthy results. As we know the solutions of the optimization problem can show a remarkable sensitivity to perturbations, coming from the data, in the parameters of the problem. To overcome this kind of difficulties, the model identification problem has been generalized by including the existence of uncertainty with respect to future scenarios through Conic Multivariate Adaptive Regression Splines (CMARS), whose data are assumed to contain certain information with respect to input variables. Then, with the help of robust optimization which can deal with a wider data uncertainty, CMARS has been robustified and named as Robust CMARS (RCMARS). We decrease the estimation variance by using robustification in CMARS. In contrast to early studies, where RCMARS was presented in theory and method and illustrated with a numerical example, in this study, we present RCMARS results for real-world data from financial markets, particularly, from the Istanbul Stock Exchange, Turkish and US economy, showing that RCMARS can generate more accurate models with a smaller variance.

Key words: *regression, uncertainty, robust optimization, CMARS, RCMARS, financial market*

Mathematics Subject Classification: *46N10, 62F35, 65K10, 90C51*

1 Introduction

The first requirement in every problem is to understand the nature of the data used and to be able to correctly represent it. Some models assume for instance that the asset returns follow a multivariate normal distribution. In particular, Markowitz model assumes that the first two moments of the distribution completely describe the distribution of the asset returns and the characteristics of the different portfolios. Real markets on the contrary often exhibit more intricacies, with distributions of returns depending on moments of higher-order with the distribution parameters varying over time, or they exhibit an increased uncertainty which leads to instability in the parameters [1]. Analyzing and modeling such complex financial data is a whole subject in data mining and information technology, which is the area of research of many scientists.

One of the fundamental concepts in finance theory is optimization, and the financial decision making for a rational agent is essentially a question of achieving an optimal trade-off between risk and return. In this way, robustification is starting to attract more attention

*Corresponding author

ISSN 1348-9151

in finance; in particular, some studies report promising results using robust statistical techniques in financial markets.

CMARS has been developed as an alternative method to the well-known regression tool MARS from data mining and estimation theory [24]. This study further improves CMARS to treat uncertainty in data. As we know, real-world data include noise in both input and output variables, meaning that the data of the regression problem are not exactly known or may not be exactly measured, or, because of intrinsic inaccuracy of the devices, the exact solution of the problem may not be carried out [8]. Moreover, the data can experience small changes by variations in the optimal experimental design. All of that can lead to uncertainty in possible constraints and in the objective function. In order to overcome this problem, we modified CMARS algorithm by the important *robust optimization* method developed by Ben-Tal and Nemirovski [3, 4, 6], and El-Ghaoui and Lebret [12], and called it as *Robust CMARS* (*RCMARS*), which gradually reduced the estimation variance. Not anticipating too many details of our elaboration, we mention both (i) that we have parameters at hand to *control* some degree of risk-friendliness or -averseness, and (ii) that we employ a concept of so-called *weak* robustification.

Robust optimization is a modeling methodology to process optimization problems whose data are uncertain and merely belong to some uncertainty set, except for outliers, with the purpose of finding an optimal or a near optimal solution which is feasible for every possible realization of the uncertain scenarios [7]. The robust optimization approach aims at making the optimization models robust regarding constraint violations by solving robust counterparts of these problems within prespecified uncertain sets for the uncertain parameters [14]. Robust counterparts are solved for the worst-case realization of the uncertain parameters based on suitable uncertainty sets, predetermined for the random uncertain parameters.

In our previous studies, we firstly incorporated uncertainty into the CMARS model with complexity terms in the form of integrals of squared first- and second-order derivatives of the model functions, then, into the discretized Tikhonov regularization and, finally, into the *Conic Quadratic Programming* (*CQP*) form of the problem. Afterwards, we introduced a *robustification* of CMARS with robust optimization under polyhedral uncertainty [21, 22]. Because of the computational costs caused by robustification of CMARS, the concept of a weak robustification has been introduced and called as *Weak RCMARS* (*WRCMARS*).

In this study, we used data from Istanbul Stock Exchange like ISE 100 index, ISE transaction number and so on, from Turkish economy like TUFE and TEFE indexes, and also data of the Fed Funds Interest Rate and VIX Index which have been obtained from the US market, because of their strong effect on the economy of Turkey. ISE 100 index has been taken as the dependent variable, and others as the independent variables. We put a correlation threshold in order to limit the unnecessary and meaningless calculations and eliminated several variables which do not satisfy this requirement. Afterwards, we applied RCMARS to the remaining independent variables. This paper is organized as follows. The objectives and outline of the study are represented in Section 1. In Section 2, RCMARS is introduced in theory and method. Our RCMARS application with different uncertainty scenarios is presented in Section 3. A conclusion and an outlook to further studies are stated in the last section.

2 CMARS Model

CMARS is developed to be an alternative to backward elimination part of MARS, which has a great potential for fitting nonlinear multivariate functions. In fact, MARS is a powerful adaptive and flexible nonparametric regression method to estimate general functions of high-

dimensional regression problems. MARS obeys the following general model representation, supposed to exist between the variables [15, 16]:

$$Y = f(\boldsymbol{X}) + \varepsilon, \tag{2.1}$$

where Y is the response variable, $\boldsymbol{X} = (X_1, X_2, \ldots, X_p)^T$ is a vector of predictor variables, and ε is an additive stochastic component which is assumed to have 0 mean and finite variance. MARS aim to obtain reflected pairs for each input variable X_j $(j = 1, 2, \ldots, p)$ with p-dimensional knots $\boldsymbol{\tau}_i = (\tau_{i1}, \tau_{i2}, \ldots, \tau_{ip})^T$ at, or just nearby, each input data vectors $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$ $(i = 1, 2, \ldots, N)$. In MARS, the basis fuctions (BFs) are determined as [16]

$$c^+(x, \tau) = (x - \tau)_+, \quad c^-(x, \tau) = (x - \tau)_-, \tag{2.2}$$

where $(q)_+ := \max\{0, q\}, (q)_- := \max\{0, -q\}$ for $(q \in \mathbb{R})$ and τ is a univariate knot. Here, with a knot value, τ, each function is piecewise linear which is called a *reflected pair*. Consequently, the set of BFs, indicated by T, takes the following form:

$$T := \big\{(X_j - \tau)_+, (\tau - X_j)_+ \mid \tau \in \{x_{1j}, x_{2j}, \ldots, x_{Nj}\},\ j \in \{1, 2, \ldots, p\}\big\},$$

where N is the number of observations and p is the dimension of the input space. The function $f(\boldsymbol{X})$ in equation (2.1) can be closer presented by a successively obtained linear combination of functions constructed from the set T and the intercept, α_0, where Y is written as

$$Y = \alpha_0 + \sum_{m=1}^{M} \alpha_m \psi_m(\boldsymbol{X}) + \varepsilon. \tag{2.3}$$

Here, ψ_m $(m = 1, 2, \ldots, M)$ is a BF from T or products of two or more such functions, and α_m is the unknown coefficient for the mth BF $(m = 1, 2, \ldots, M)$, but m equals to 0 for the constant one. Therefore, the multiplicative form of the mth BF becomes [16]

$$\psi_m(\boldsymbol{x}_i) = \prod_{j=1}^{K_m} (x_{i\kappa_j^m} - \tau_{\kappa_j^m})_{\pm} \quad (i = 1, 2, \ldots, N), \tag{2.4}$$

where the number of truncated linear functions multiplied in the mth BF is denoted by K_m. Moreover, $x_{i\kappa_j^m}$ is the input variable corresponding to the jth truncated linear function in the mth BF, and $\tau_{\kappa_j^m}$ is the knot value corresponding to the variable.

In CMARS method, firstly, the large model provided by the forward MARS algorithm is built up and addressed. Instead of the backward stepwise algorithm of MARS, as an alternative [24], the *Penalized Residual Sum of Square* (*PRSS*) with $M_{\max}$ BFs is employed as a refinement of the *Least-Squares Estimation* (*LSE*) to control the lack of fit from the viewpoint of the tradeoff between goals of *complexity* and *stability* to estimate and assess the function $f(x)$ in (2.1). Therefore, PRSS has the following form [24]:

$$PRSS := \sum_{i=1}^{N} (y_i - \boldsymbol{\alpha}^T \boldsymbol{\psi}(\tilde{\boldsymbol{b}}_i))^2 + \sum_{m=1}^{M_{\max}} \phi_m \sum_{\substack{|\boldsymbol{\theta}|=1 \\ \boldsymbol{\theta}^T=(\theta_1,\theta_2)}}^{2} \sum_{\substack{r<s \\ r,s \in V(m)}} \int_{Q^m} \alpha_m^2 [D_{r,s}^{\boldsymbol{\theta}} \psi_m(\boldsymbol{t}^m)]^2 d\boldsymbol{t}^m. \tag{2.5}$$

Here, $\boldsymbol{\psi}(\tilde{\boldsymbol{b}}_i) := (1, \psi_1(\tilde{\boldsymbol{x}}_i^1), \psi_2(\tilde{\boldsymbol{x}}_i^2), \ldots, \psi_m(\tilde{\boldsymbol{x}}_i^{M_{\max}}))$; $V(m) := \{\kappa_j^m \mid j = 1, 2, \ldots, K_m\}$ is the variable set associated with the mth BF, ψ_m; $\boldsymbol{t}^m = (t_{m_1}, t_{m_2}, \ldots, t_{m_{K_m}})^T$ represents the vector of variables that contribute to the mth BF, ψ_m (likewise for $\boldsymbol{x}$ and $\boldsymbol{X}$ in

equations (2.3) and (2.4)); $\boldsymbol{\alpha}$ is an $((M_{\max}+1)\times 1)$- parameter vector to be estimated using the data points; $\phi_m \geq 0$ are the *penalty parameters* ($m = 1, 2, \ldots, M_{\max}$). Moreover, Q^m is some suitably large K_m -dimensional parallelpipe where the integration occurs; $D^{\boldsymbol{\theta}}_{r,s}\psi_m(\boldsymbol{t}^m) = ((\partial^{|\boldsymbol{\theta}|}\psi_m)/(\partial^{\theta_1}t_r^m\partial^{\theta_2}t_s^m))\boldsymbol{t}^m$ express the first- or second-order derivatives, where $\boldsymbol{\theta}^T = (\theta_1, \theta_2)$, $|\boldsymbol{\theta}| := \theta_1 + \theta_2$ and $\theta_1, \theta_2 \in \{0, 1\}$.

Since it is not easy to evaluate the multi-dimensional integrals in (2.5), a discretization is applied to approximate the integral $\int \alpha_m^2[D^{\boldsymbol{\theta}}_{r,s}\psi_m(\boldsymbol{t}^m)]^2 d\boldsymbol{t}^m$ (we refer to [22, 24] for more details). Consequently, the approximation of PRSS in (2.5) can be rearranged as

$$PRSS \approx \left\|\boldsymbol{y} - \boldsymbol{\psi}(\tilde{\boldsymbol{b}})\boldsymbol{\alpha}\right\|_2^2 + \phi\left\|\boldsymbol{L}\boldsymbol{\alpha}\right\|_2^2, \tag{2.6}$$

where $\boldsymbol{L}$ is an $((M_{\max}+1)\times(M_{\max}+1))$-diagonal matrix. Afterwards, the *PRSS* problem turns into a classical *Tikhonov Regularization* (*TR*) [2] problem if we employ only one penalty factor $\phi > 0$, $\phi = \lambda^2$ for some $\lambda \in \mathbb{R}$ instead of using different penalty parameters. Therefore, the PRSS form in (2.6) may be formulated as a CQP and, using an appropriate bound $\tilde{M}$, the following optimization problem can be stated [24]:

$$\begin{aligned} \min_{t,\boldsymbol{\alpha}} \; t \quad \text{subject to} \quad & \left\|\boldsymbol{\psi}(\tilde{\boldsymbol{b}})\boldsymbol{\alpha} - \boldsymbol{y}\right\|_2 \leq t, \\ & \left\|\boldsymbol{L}\boldsymbol{\alpha}\right\|_2 \leq \sqrt{\tilde{M}}. \end{aligned} \tag{2.7}$$

We underline that this choice of $\tilde{M}$ have to be the outcome of a careful learning process, with the help of model-free or model-based methods [2].

3 RCMARS Model

3.1 CMARS Model with Uncertainty

We assume that the input and output variables of our model are random variables all. They lead us to *uncertainty sets*; those are assumed to contain *confidence intervals* (*CIs*) (we refer to [20, 22] for more details). For CMARS, the large model that has the maximum number of BFs, $M_{\max}$, is created by Salford MARS® [18]. The following general model represents the relation between both the *random* input variables and the response, itself being affected with noise:

$$Y = f(\underbrace{\breve{X}}_{\text{noisy data}}) + \varepsilon, \tag{3.1}$$

where $\breve{\boldsymbol{X}} = (\breve{X}_1, \breve{X}_2, \ldots, \breve{X}_p)^T$ is a vector of random predictor variables. The random variables $\breve{X}_j$ are assumed to be normally distributed. Here, the following general model is considered for each input $\breve{X}_j$ [20, 22]:

$$\breve{X}_j = \bar{X} + \xi_j. \tag{3.2}$$

When considering that we have p-dimensional input data and incorporate a "*perturbation*" (*uncertainty*) into input data, each input data vector $\breve{\boldsymbol{x}}_i = (\breve{x}_{i1}, \breve{x}_{i2}, \ldots, \breve{x}_{ip})^T$ is represented as $\breve{\breve{\boldsymbol{x}}}_i = (\breve{\breve{x}}_{i1}, \breve{\breve{x}}_{i2}, \ldots, \breve{\breve{x}}_{ip})^T$, including the perturbation $\boldsymbol{\Delta}_i = (\Delta_{i1}, \Delta_{i2}, \ldots, \Delta_{ip})^T$ ($i = 1, 2, \ldots, N$). Since, in each coordinate, value $\breve{x}_{ij}$ can be outlier, but perturbation of outlier is not meaningful, for our problem, we, instead, refer to $\bar{x}$, the average of the input data $\breve{\boldsymbol{x}}_i$, as the reference value wherever we use $\breve{\boldsymbol{x}}$. Here, $\boldsymbol{\Delta}_i$ is a generic element of U_1,

which is the uncertainty set for our input data. Herewith, our new values of piecewise linear BFs are shown in the following:

$$\breve{x}_{ij} \to \breve{\breve{x}}_{ij}; \qquad \breve{\breve{x}}_{ij} = \bar{x}_j + \Delta_{ij}, \ \ |\Delta_i| \le \rho_{ij} \ \ (j = 1, 2, \ldots, p; i = 1, 2, \ldots, N), \tag{3.3}$$

where $\breve{x}_{ij}$ is an noisy input value; $\breve{\breve{x}}_{ij}$ is an input value that has uncertainty; Δ_{ij} is a perturbation of $\breve{x}_{ij}$; ρ_{ij} is the semilength of CI for input data, and the amount of perturbation in each dimension is restricted by ρ_{ij}.

Similarly, when we incorporate a "*perturbation*" (*uncertainty*) into output data, our output data vector $\breve{\boldsymbol{y}} = (\breve{y}_1, \breve{y}_2, \ldots, \breve{y}_N)^T$ is stated as $\breve{\breve{\boldsymbol{y}}} = (\breve{\breve{y}}_1, \breve{\breve{y}}_2, \ldots, \breve{\breve{y}}_N)^T$ including the perturbation $\mathbf{H} = (\mathrm{H}_1, \mathrm{H}_2, \ldots, \mathrm{H}_N)^T$. As again value $\breve{y}_i$ can be outlier and since perturbation of outlier is not meaningful, for our problem, we refer to $\bar{y}$, the average of the output data $\breve{y}_i$, as the reference value wherever we write $\breve{\boldsymbol{y}}$. Here, we restrict the vector $\mathbf{H}$ to be elements of U_2, being the uncertainty set for our output data. So, our new output values can be represented by [20, 22]:

$$\breve{y}_i \to \breve{\breve{y}}_i; \qquad \breve{\breve{y}}_i = \bar{y} + \mathrm{H}_i, \ \ |\mathrm{H}_i| \le \nu_i \ \ (i = 1, 2, \ldots, N). \tag{3.4}$$

Here, the amount of perturbation is limited by ν_i which is the semilength of the CI for the output data.

In order to robustify CMARS, we employ some robust optimization on the BFs provided by the MARS [15] model. MARS method constructs expansions of piecewise linear BFs; by us, it will be based on the new data set that has uncertainty. Aiming at the variable $\breve{\breve{x}}$ we prefer the following notation for the piecewise linear BFs [16]:

$$c^+(\breve{\breve{x}}, \tau) = (\breve{\breve{x}} - \tau)_+, \quad c^-(\breve{\breve{x}}, \tau) = (\breve{\breve{x}} - \tau)_-. \tag{3.5}$$

Incorporating the uncertainty sets $U_1 \subseteq \mathbb{R}^{N \times M_{\max}}$ and $U_2 \subseteq \mathbb{R}^N$, determined below in Subsection 3.3, into the data $(\breve{\boldsymbol{x}}_i, \breve{y}_i)$, the multiplicative form of the mth BF can be stated as

$$\psi_m(\breve{\breve{\boldsymbol{x}}}_i) = \prod_{j=1}^{K_m} (\breve{\breve{x}}_{i\kappa_j^m} - \tau_{\kappa_j^m})_\pm \ \ (i = 1, 2, \ldots, N). \tag{3.6}$$

Then, for our CMARS model with uncertainty, PRSS in (2.6) will have the following approximate representation:

$$PRSS \approx \left\| \breve{\breve{\boldsymbol{y}}} - \boldsymbol{\psi}(\breve{\breve{\boldsymbol{b}}})\boldsymbol{\alpha} \right\|_2^2 + \phi \left\| \boldsymbol{L}\boldsymbol{\alpha} \right\|_2^2. \tag{3.7}$$

Herewith, the *PRSS* minimization problem again looks like a classical TR [2] problem with $\phi > 0$, i.e., $\phi = \lambda^2$ for some $\lambda \in \mathbb{R}$. Then, it can be coped with through CQP [22, 24]. The second (complexity) part of the PRSS approximation remains the same as it is in CMARS after we incorporate a *"perturbation"* into the real input data $\breve{\boldsymbol{x}}_i$, in each dimension, and into the output data $\breve{y}_i$, since we do not make any changes for the function in the multidimensional integrals. When estimating the BFs $(\breve{\breve{x}}_{i\kappa_j^m} - \tau_{\kappa_j^m})_\pm$ in (3.6), we can evaluate them by the following special terms of estimation [22]:

$$(\breve{\breve{x}}_{i\kappa_j^m} - \tau_{\kappa_j^m})_\pm \le (\breve{x}_{i\kappa_j^m} - \tau_{\kappa_j^m})_\pm + (\Delta_{i\kappa_j^m} + (\pm \mathrm{A}_{i\kappa_j^m}))_\pm. \tag{3.8}$$

In our studies on CMARS and RCMARS, we usually write $\psi_m(x^m)$ for the mth BF, where x^m is some subvector of x. For not overloading the exposition by further indices and for the easy of understanding, we denote that value by $\psi_m(x)$ in this paper.

Here, $\mathrm{A}_{i\kappa_j^m}$ is interpreted and employed as *control parameters*. If we consider the *risk friendly* case, we select the value of $\breve{\mathrm{A}}_{i\kappa_j^m}$ between 0 and the absolute value of $\mathrm{A}_{i\kappa_j^m}$ i.e., $\breve{\mathrm{A}}_{i\kappa_j^m} \in [0, |\mathrm{A}_{i\kappa_j^m}|]$. Here, to simplify the notation, we still preserve the notion $\mathrm{A}_{i\kappa_j^m}$ for $\breve{\mathrm{A}}_{i\kappa_j^m}$. To estimate the values $\psi(\breve{\boldsymbol{x}}_i)$ and $\psi(\breve{\breve{\boldsymbol{x}}}_i)$, we can employ (3.8) in the subsequent form, where all the "+" and "-" signs belong to each other, respectively [20]:

$$\underbrace{\prod_{j=1}^{K_m}(\breve{\breve{x}}_{i\kappa_j^m}-\tau_{\kappa_j^m})_{\pm}}_{=:\psi_m(\breve{\breve{\boldsymbol{x}}}_i)} \le \underbrace{\prod_{j=1}^{K_m}(\breve{x}_{i\kappa_j^m}-\tau_{\kappa_j^m})_{\pm}}_{=:\psi_{\boldsymbol{m}}(\breve{x}_i)} + \sum_{A\subsetneq\{1,\dots,K\}}\prod_{a\in A}(\breve{x}_{ia}-\tau_a)_{\pm}\prod_{b\in\{1,\dots,K\}/A}((\pm\mathrm{A}_{ib})+\Delta_{ib})_{\pm}\ \ (i=1,2,\dots,N). \tag{3.9}$$

Then, for each BF, the uncertainty value $|u_{im}|$ can be estimated in the subsequent way [20]:

$$\begin{aligned}
|u_{im}| &\le \sum_{A\subsetneq\{1,\dots,K\}}\prod_{a\in A}\underbrace{|\breve{x}_{ia}-\tau_a|}_{\le \mathrm{B}_{ia}\rho_{ia}}\prod_{b\in\{1,\dots,K\}/A}\underbrace{(|\pm\mathrm{A}_{i\kappa_m^j}+\Delta_{i\kappa_m^j}|)}_{\le\gamma_{ib}+\rho_{ib}}\\
&\le \sum_{A\subsetneq\{1,\dots,K\}}\prod_{a\in A}\mathrm{B}_{ia}\rho_{ia}\prod_{b\in\{1,\dots,K\}/A}(\gamma_{ib}+\rho_{ib})\\
&\le \sum_{A\subsetneq\{1,\dots,K\}}\prod_{a\in A}\underbrace{\mathrm{B}_{ia}}_{\le\mathrm{B}_i}\prod_{a\in A}\rho_{ia}\prod_{b\in\{1,\dots,K\}/A}(\gamma_{ib}+\rho_{ib})\\
&\le \sum_{A\subsetneq\{1,\dots,K\}}\mathrm{B}_i^{|A|-1}\prod_{a\in A}\rho_{ia}\prod_{b\in\{1,\dots,K\}/A}(\gamma_{ib}+\rho_{ib}),
\end{aligned} \tag{3.10}$$

where the amount of the value of $\mathrm{A}_{i\kappa_m^j}$ is restricted by γ, the cardinality of the set A has been denoted through $|A|$, and B_i is also considered to be applied as a *control parameter*. The value of B_i is equal to 2 in cases without outliers, but for outliers, it will be greater than 2. For such a case, we will have to select a different value for B_i.

3.2 Robustification of the CMARS Model

The CMARS model depends on parameters. Small perturbations in data can result in very different model parameters, which may indeed cause unstable solutions. The purpose and basic idea of RCMARS is to decrease the estimation error, while keeping efficiency as high as possible. In order to achieve this goal, one applies some approaches such as usage of more robust estimators, scenario optimization and robust counterpart. We aim to reduce the estimation variance by using a robustification in CMARS [22].

Let us conduct a penalization in the form of TR and study it as a CQP problem for our CMARS model in order to achieve a reduction in the complexity of the regression method MARS. That complexity especially means sensitivity with respect to noise in the data. Regularization in CMARS is already some first kind of robustification, but, in our study, we additionally robustify CMARS through the robust optimization approach, which is some rigorous kind of regularization in the input and output domain. However, as stated in Subsection 3.1, we employ control parameters for a fine tuning. For all these reasons, we

have some generalization effect now in the part of $\left\|\boldsymbol{\psi}(\breve{\breve{\boldsymbol{b}}})\boldsymbol{\alpha}-\breve{\breve{\boldsymbol{y}}}\right\|_2^2$, when we conduct our robustification of CMARS for both input and output variables by including uncertainty, via robust optimization [22]. But, we need not make any change in the additional integration term on the complexity, or energy in Subsection 4.3, equation (4.3). Therefore, the part of remains the same as in CMARS.

3.3 Polyhedral Uncertainty and Robust Counterpart for the CMARS Model

As we know, robustification is more successful when ellipsoidal uncertainty sets are employed, rather than polyhedral uncertainty sets. Nevertheless, using ellipsoidal uncertainty sets can increase the complexity of our optimization models [23, 25]. We study *robust CQP* (*robust second order optimization problem*, *robust SCOP*) under polyhedral uncertainty and we shall find out that it equivalently means a *standard CQP*.

To analyze the robustness problem, we assume that the given model uncertainty is represented by a family of matrices $\boldsymbol{\psi}(\breve{\breve{\boldsymbol{x}}}) = \boldsymbol{\psi}(\breve{\boldsymbol{x}}) + \boldsymbol{U}$ and vectors $\breve{\breve{\boldsymbol{y}}} = \breve{\boldsymbol{y}} + \boldsymbol{v}$, where U_1, containing $\boldsymbol{U}$, and U_2, containing $\boldsymbol{v}$, are bounded sets which need to be specified. Here, the uncertainty matrix $\boldsymbol{U} \in U_1$ and uncertainty vector $\boldsymbol{v} \in U_2$ are of the formats [21, 22]

$$\boldsymbol{U} = \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1M_{\max}} \\ u_{21} & u_{22} & \dots & u_{2M_{\max}} \\ \vdots & \vdots & \ddots & \vdots \\ u_{N1} & u_{N2} & \dots & u_{NM_{\max}} \end{bmatrix} \text{ and } \boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{bmatrix}. \tag{3.11}$$

As we do not want to increase the overall complexity of our optimization problems, we select the uncertainty sets U_1 and U_2 of type *polyhedral* for both input and output data in our model, to study our robustness problem. Based on these sets, the robust counterpart is defined as

$$\min_{\boldsymbol{\alpha}} \max_{\substack{\boldsymbol{W}\in U_1 \\ \boldsymbol{z}\in U_2}} \left\|\boldsymbol{W}\boldsymbol{\alpha}-\boldsymbol{z}\right\|_2^2 + \phi\left\|\boldsymbol{L}\boldsymbol{\alpha}\right\|_2^2. \tag{3.12}$$

with some $\phi \geq 0$. Here, U_1 is a polytope with $2^{N\cdot M_{\max}}$ vertices $\boldsymbol{W}^1, \boldsymbol{W}^2, \dots, \boldsymbol{W}^{2^{N\cdot M_{\max}}}$. It is not a singleton, but permits a representation [21, 22]

$$U_1 = \left\{ \sum_{j=1}^{2^{N\cdot M_{\max}}} \delta_j \boldsymbol{W}^j | \delta_j \geq 0 \ (j \in \{1, 2, \dots, 2^{N\cdot M_{\max}}\}), \sum_{j=1}^{2^{N\cdot M_{\max}}} \delta_j = 1 \right\}, \tag{3.13}$$

i.e., $U_1 = \text{conv}\{\boldsymbol{W}^1, \boldsymbol{W}^2, \dots, \boldsymbol{W}^{2^{N\cdot M_{\max}}}\}$ is the convex hull. Furthermore, U_2 is a polytope with 2^N vertices $\boldsymbol{z}^1, \boldsymbol{z}^2, \dots, \boldsymbol{z}^{2^N}$. It permits the form

$$U_2 = \left\{ \sum_{i=1}^{2^N} \varphi_i \boldsymbol{z}^i | \varphi_i \geq 0 \ (i \in \{1, 2, \dots, 2^N\}), \sum_{j=1}^{2^N} \varphi_i = 1 \right\}, \tag{3.14}$$

where $U_2 = \text{conv}\{\boldsymbol{z}^1, \boldsymbol{z}^2, \dots, \boldsymbol{z}^{2^N}\}$ is the convex hull. Here, any uncertainty sets U_1 and U_2 can be represented as a convex combination of vertices $\boldsymbol{W}^j$ $(j \in \{1, 2, \dots, 2^{N\times M_{\max}}\})$ and $\boldsymbol{z}^i$ $(i \in \{1, 2, \dots, 2^N\})$ of the polytope, respectively. The entries are found to have become intervals. Therefore, our matrix $\boldsymbol{W}$ and vector $\boldsymbol{z}$ with uncertainty are lying in the Cartesian product of intervals that are parallelpipes (see [21, 22] for more details). Here, we represented the matrix $\boldsymbol{W}$ as a vector with uncertainty which generates a parallelpipe.

We have a $(N \times M_{\max})$-matrix $\boldsymbol{W} = (w_{ij})_{\substack{i=1,2,\ldots,N \\ j=1,2,\ldots,M_{\max}}}$ and we can write it as a vector $\boldsymbol{t} = (t_k)_{k=1,2,\ldots,N\times M_{\max}}$, where $t_k := u_{ij}$ with $k = i + (j-1)N$. So, our matrix $\boldsymbol{W}$ can be canonically represented as a vector $\boldsymbol{t}_k = (t_1, t_2, \ldots, t_{N\times M_{\max}})^T$ by putting the columns of $\boldsymbol{W}$ behind each other [22].

3.4 Robust CQP with Polyhedral Uncertainty

For our CMARS model, the optimization problem is written as follows [21, 22]:

$$\begin{aligned} \min_{t,\boldsymbol{\alpha}} \; t \quad \text{subject to} \quad & \left\|\boldsymbol{\psi}(\breve{\boldsymbol{b}})\boldsymbol{\alpha} - \breve{\boldsymbol{y}}\right\|_2 \leq t, \\ & \left\|\boldsymbol{L}\boldsymbol{\alpha}\right\|_2 \leq \sqrt{\breve{M}}, \end{aligned} \tag{3.15}$$

with some parameter $\breve{M} \geq 0$. When *polyhedral* uncertainty implied into the CMARS model based on the uncertainty sets U_1 and U_2, the robust counterpart is defined by

$$\min_{\boldsymbol{\alpha}} \max_{\substack{\boldsymbol{W}\in U_1 \\ \boldsymbol{z}\in U_2}} \left\|\boldsymbol{W}\boldsymbol{\alpha} - \boldsymbol{z}\right\|_2^2 + \phi\left\|\boldsymbol{L}\boldsymbol{\alpha}\right\|_2^2, \tag{3.16}$$

with some $\phi \geq 0$. So, via height variable t (by an epigraph argument), the robust CQP for our optimization problem is equivalently represented in the following form [22]:

$$\begin{aligned} \min_{t,\boldsymbol{\alpha}} \; t \quad \text{subject to} \quad & \left\|\boldsymbol{W}\boldsymbol{\alpha} - \boldsymbol{z}\right\|_2 \leq t \quad \forall \underbrace{\boldsymbol{W}}_{=\sum_{j=1}^{2^{N\cdot M_{\max}}} \delta_j \boldsymbol{W}^j} \in U_1, \; \underbrace{\boldsymbol{z}}_{=\sum_{i=1}^{2^N} \varphi_i \boldsymbol{z}^i} \in U_2, \\ & \left\|\boldsymbol{L}\boldsymbol{\alpha}\right\|_2 \leq \sqrt{\breve{M}}. \end{aligned} \tag{3.17}$$

Here, U_1 and U_2 are polytopes which are described by their vertices as

$$U_1 = \operatorname{conv}\{\boldsymbol{W}^1, \boldsymbol{W}^2, \ldots, \boldsymbol{W}^{2^{N\cdot M_{\max}}}\}, \;\; U_2 = \operatorname{conv}\{\boldsymbol{z}^1, \boldsymbol{z}^2, \ldots, \boldsymbol{z}^{2^N}\}. \tag{3.18}$$

Therefore, our robust CQP can be equivalently stated by a standard CQP [5, 13] as follows:

$$\begin{aligned} \min_{t,\boldsymbol{\alpha}} \; t \quad \text{subject to} \quad & \left\|\boldsymbol{W}^j\boldsymbol{\alpha} - \boldsymbol{z}^i\right\|_2 \leq t \;\; (i = 1, 2, \ldots, 2^N; j = 1, 2, \ldots, 2^{N\cdot M_{\max}}), \\ & \left\|\boldsymbol{L}\boldsymbol{\alpha}\right\|_2 \leq \sqrt{\breve{M}}. \end{aligned} \tag{3.19}$$

Afterwards, we can solve our robust CQP by using MOSEK$^{\text{TM}}$ [19] software program. Here, we note that the values $\sqrt{\breve{M}}$ are determined by a model-free method. When we employ the $\sqrt{\breve{M}}$ values in our RCMARS code and solve by using MOSEK, we apply the $\sqrt{\breve{M}}$ value that has the minimum value of PRSS in the equation (3.7).

4 A Real-World Application of RCMARS in the Financial Market

4.1 Data Description

We selected the time-series data for the empirical part from the website of Central Bank of the Republic of Turkey [9]. The data contain the economic indicators which are the most commonly used ones for the interpretation of an economic situation. Monthly data

have been preferred in order to have more definite and stationary results, relative to daily or weekly data. If we could not find the monthly data, we used daily data and converted them to monthly data by taking averages, or for some of them the last data of the month were taken as the data of the month, like Net Foreign Exchange Reserves and International Gold Reserves. *ISE 100 stock index* is the dependent variable in our data set. We used this index, because it is a statistical measure of change in an economy or a securities market. For financial markets, an index is an imaginary portfolio of securities representing a particular market or a portion of it. It has its own calculation methodology and is usually expressed in terms of a change from a base value. Thus, the percentage change is more important than the actual numerical value.

The independent variables are *ISE Transaction Number* (the number of transaction during a defined time period, in our case during the month), *ISE Trading Volume* (the number of shares or contracts of a security traded during a defined time period, again for a month), *Capacity Usage Ratio* (the ratio of the production capacity of the economy to the maximum capacity of economy), *Euro and Dollar Exchange Rate*, *Net Foreign Exchange Reserves and International Gold Reserves*, *Gold Price*, *Credit Volume*, *Price Indexes* like *WPI* and *CPI* (*TEFE* and *TUFE*, respectively). WPI or *Wholesale Price Index* (TEFE) is the price of a representative basket of wholesale goods, while a CPI or *Consumer Price Index* (TUFE) measures changes in the price level of consumer goods and services purchased by households.

Two indicators from the USA are taken to our analyses: *Fed Funds Interest Rate* and *VIX Index* (a measure of the market's expectation of stock market volatility over the next 30 day period), because of the strong effect of the USA on the economy of Turkey and the world.

As it is said above, in this study, we use ISE 100 Stock Market index as a dependent variable. This is the successor of the *Composite Index*, which was introduced in 1986 including the stocks of 40 companies and was in time limited to the stocks of 100 companies. It consists of 100 stocks, which have been selected among the stocks of companies listed on the National Market, and the stocks of real estate investment trusts and venture capital investment trusts, listed on the Corporate Products Market, and it covers ISE 30 and ISE 50 stocks.

The data cover the time horizon between January 1999 and December 2009. Some of the series do not contain the data of December 2009; therefore, the absent values are calculated in Excel using interpolation. We also checked the correlation among these series, in order to prevent from unnecessary and meaningless calculations. We assumed a correlation threshold of 0.90 to decide about the strength of correlation. The most correlated factors are ISE Trading Volume, International Gold Reserves, Net Foreign Exchange Reserves and WPI (TEFE). For example, there is a correlation of 0.94 between ISE Transaction Number and ISE Trading Volume. So, ISE Transaction Number is taken out from the list. Eventually, our data set consists of ISE Trading Volume, Capacity Usage Ratio, Euro and Dollar Exchange Rates, Credit Volume, Gold Price, WPI (TEFE), Fed Funds Interest Rate and VIX Index.

4.2 Obtaining Large Model from MARS Program

For the implementation of our RCMARS algorithm developed, we used a data set from the financial market and, eliminating some of the predictor variables which have the correlation.

At the end we have 8 predictor input variables:

X_1 : ISE Trading Volume,	X_2 : Capacity Usage Ratio,
X_3 : Euro Exchange Rate,	X_4 : Credit Volume,
X_5 : Dollar Exchange Rate,	X_6 : Price Index (TEFE),
X_7 : Federal Funds Interest Rate,	X_8 : VIX Index,

with 76 observations. However, we do not have enough computer capacity to solve our problem (3.12) that is given as a *tradeoff* between tractability and robustification. Therefore we divide our data set into two subsets, each of which has 38 observations. Firstly, we validate our assumption that the input variables and the output variable are distributed normally, using *bootstrapping method* [11] from statistics. In order to implement RCMARS algorithm, first, the MARS models are constructed for each subset by using the Salford MARS version 3 [8] and, then, the maximum number of BFs ($M_{\max}$) and the highest degree of interactions are determined by trial and error. In first part of our data set, $M_{\max}$ is assigned to be 12, and the highest degree of interaction is assigned to be 3. Then, the largest models for the first part and the second part of the data set are constructed in the forward MARS algorithm by its software.

To prevent from nondifferentiability in our optimization problem, we choose the knot values different from data points. However, these values are very much nearby to the corresponding input data. Then, the BFs for the first part of the data set can be introduced into the largest model subsequent way:

$$
\begin{aligned}
Y &= \alpha_0 + \sum_{m=1}^{M} \alpha_m \psi_m(\boldsymbol{x}) + \varepsilon \\
&= \alpha_0 + \alpha_1\psi_1(\boldsymbol{x}) + \alpha_2\psi_2(\boldsymbol{x}) + \alpha_3\psi_3(\boldsymbol{x}) + \alpha_4\psi_4(\boldsymbol{x}) + \alpha_5\psi_5(\boldsymbol{x}) + \alpha_6\psi_6(\boldsymbol{x}) \\
&\quad + \alpha_7\psi_7(\boldsymbol{x}) + \alpha_8\psi_8(\boldsymbol{x}) + \alpha_9\psi_9(\boldsymbol{x}) + \alpha_{10}\psi_{10}(\boldsymbol{x}) + \alpha_{11}\psi_{11}(\boldsymbol{x}) + \alpha_{12}\psi_{12}(\boldsymbol{x}) + \varepsilon \\
&= \alpha_0 + \alpha_1 \max\{0, x_8 - 0.365\} + \alpha_2 \max\{0, 0.365 - x_8\} \\
&\quad + \alpha_3 \max\{0, x_1 + 0.567\} + \alpha_4 \max\{0, -0.567 - x_1\} \\
&\quad + \alpha_5 \max\{0, x_2 + 0.542\} + \alpha_6 \max\{0, -0.542 - x_2\} \\
&\quad + \alpha_7 \max\{0, x_4 + 2.187\} \cdot \max\{0, -0.542 - x_2\} \\
&\quad + \alpha_8 \max\{0, x_4 + 0.098\} \cdot \max\{0, 0.365 - x_8\} \\
&\quad + \alpha_9 \max\{0, -0.098 - x_4\} \cdot \max\{0, 0.365 - x_8\} \\
&\quad + \alpha_{10} \max\{0, x_7 + 2.216\} \cdot \max\{0, x_1 + 0.567\} \\
&\quad + \alpha_{11} \max\{0, x_6 - 0.542\} \cdot \max\{0, x_7 + 2.216\} \cdot \max\{0, x_1 + 0.567\} \\
&\quad + \alpha_{12} \max\{0, 0.542 - x_8\} \cdot \max\{0, x_7 + 2.216\} \cdot \max\{0, x_1 + 0.567\} + \varepsilon.
\end{aligned}
$$

Likewise, the BFs for the second part of the data set become inserted in the largest model in the following manner:

$$
\begin{aligned}
Y &= \alpha_0 + \sum_{m=1}^{M} \alpha_m \psi_m(\boldsymbol{x}) + \varepsilon \\
&= \alpha_0 + \alpha_1\psi_1(\boldsymbol{x}) + \alpha_2\psi_2(\boldsymbol{x}) + \alpha_3\psi_3(\boldsymbol{x}) + \alpha_4\psi_4(\boldsymbol{x}) + \alpha_5\psi_5(\boldsymbol{x}) + \alpha_6\psi_6(\boldsymbol{x})
\end{aligned}
$$

$$
\begin{aligned}
&+ \alpha_7\psi_7(\boldsymbol{x}) + \alpha_8\psi_8(\boldsymbol{x}) + \alpha_9\psi_9(\boldsymbol{x}) + \alpha_{10}\psi_{10}(\boldsymbol{x}) + \alpha_{11}\psi_{11}(\boldsymbol{x}) + \alpha_{12}\psi_{12}(\boldsymbol{x}) + \varepsilon \\
&= \alpha_0 + \alpha_1 \max\{0, x_4 - 0.575\} + \alpha_2 \max\{0, 0.575 - x_3\} \\
&\quad + \alpha_5 \max\{0, x_1 - 0.019\} \cdot \max\{0, 0.275 - x_3\} \\
&\quad + \alpha_6 \max\{0, 0.019 - x_1\} \cdot \max\{0, 0.275 - x_3\} \\
&\quad + \alpha_7 \max\{0, x_1 + 2.172\} \cdot \max\{0, x_4 - 0.575\} \\
&\quad + \alpha_8 \max\{0, x_7 + 0.583\} \cdot \max\{0, 0.575 - x_4\} \\
&\quad + \alpha_9 \max\{0, x_5 + 0.309\} \cdot \max\{0, x_7 + 2.583\} \cdot \max\{0, 0.575 - x_4\} \\
&\quad + \alpha_{10} \max\{0, -0.309 - x_5\} \cdot \max\{0, x_7 + 2.583\} \cdot \max\{0, 0.575 - x_4\} \\
&\quad + \alpha_{11} \max\{0, x_2 + 0.499\} \cdot \max\{0, 0.575 - x_4\} \\
&\quad + \alpha_{12} \max\{0, -0.499 - x_2\} \cdot \max\{0, 0.575 - x_4\} + \varepsilon.
\end{aligned}
$$

4.3 Evaluating Accuracy and Complexity of PRSS Form

For this numeric example, we approximate the PRSS formula as follows:

$$
PRSS \approx \overbrace{\left\|\boldsymbol{y} - \boldsymbol{\psi}(\tilde{\boldsymbol{b}})\boldsymbol{\alpha}\right\|_2^2}^{=Accuracy} + \overbrace{\phi\left\|\boldsymbol{L}\boldsymbol{\alpha}\right\|_2^2}^{=Complexity}. \tag{4.1}
$$

Herein, the first part of the TR term, which is the right-hand side, and that of the PRSS function, are equal to each other, whereas, their second parts are equal approximately. Subsequently, all those parts are stated:

Accuracy:

$$
\left\|\boldsymbol{y} - \boldsymbol{\psi}(\tilde{\boldsymbol{b}})\boldsymbol{\alpha}\right\|_2^2 = (\boldsymbol{y} - \boldsymbol{\alpha}^T\boldsymbol{\psi}(\tilde{\boldsymbol{b}}))^T(\boldsymbol{y} - \boldsymbol{\alpha}^T\boldsymbol{\psi}(\tilde{\boldsymbol{b}})) = \sum_{i=1}^{N}(y_i - \boldsymbol{\alpha}^T\boldsymbol{\psi}(\tilde{\boldsymbol{b}_i}))^2 =: (*), \tag{4.2}
$$

Complexity:

$$
\phi\left\|\boldsymbol{L}\boldsymbol{\alpha}\right\|_2^2 \approx \sum_{m=1}^{12}\phi_m \sum_{\substack{|\boldsymbol{\theta}|=1 \\ \boldsymbol{\theta}^T=(\theta_1,\theta_2)}}^{2} \sum_{\substack{r<s \\ r,s\in V(m)}} \int_{Q^m} \alpha_m^2 [D_{rs}^{\boldsymbol{\theta}}\psi_m(\boldsymbol{t}^m)]^2 d\boldsymbol{t}^m =: (**), \tag{4.3}
$$

where, indeed, $PRSS := (*) + (**)$ and $\phi = \phi_m$ $(m = 1, 2, \ldots, 12)$. Having discretized all the multi-dimensional integrals in the ***complexity*** part, they jointly turn into the form of equation (3.7) and, finally, the discretized form is indicated by $\boldsymbol{L}$. As a result, the matrix $\boldsymbol{L}$ becomes a diagonal matrix and the first column elements of $\boldsymbol{L}$ are all zero. The diagonal elements of this matrix, L_m $(m = 1, 2, \ldots, 12)$ are given below for the first part of our data set:

$$
\boldsymbol{L} = \begin{bmatrix} 0 & 0 & \ldots & 0 \\ 0 & 1.296 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0.294 \end{bmatrix}. \tag{4.4}
$$

For the second part of our data set, the diagonal elements of $\boldsymbol{L}$, L_m $(m = 1, 2, \ldots, 12)$ are comprised as follows:

$$\boldsymbol{L} = \begin{bmatrix} 0 & 0 & \ldots & 0 \\ 0 & 1.177 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 2.345 \end{bmatrix}. \tag{4.5}$$

4.4 Calculating Uncertainty Values for Input and Output Data under Polyhedral Uncertainty

We incorporate a perturbation (uncertainty) into the real input data in each dimension and into the output data, after we obtain ***accuracy*** and ***complexity*** terms, to employ our robust optimization technique on the CMARS model. For this purpose, the right-hand side on an uncertainty bound from (3.10) is evaluated for all input and output values which are represented by *CIs*, and the uncertainty matrices and vectors based on *polyhedral uncertainty* sets are obtained by using (3.13) and (3.14).

Furthermore, to perform the given calculations, we need normally distributed data and, since in our data set some variables are not normally distributed, we use the *bootstrapping* method of statistics [11], which is the general approach to statistical inference based on building a sampling distribution for a statistic by resampling from the data at hand. After the normalization of the variables, we transform them into the standard normal distribution; the CI is constructed to be [-3, 3].

With our worst case approach, for the each observation, we use the equation (3.10) to receive the uncertainty vectors with their entries u_{im} $(i = 1, 2, \ldots, 38; m = 1, 2, \ldots, 12)$:

$$|u_{im}| = |\psi_m(\breve{\breve{\boldsymbol{x}}}_i) - \psi_m(\breve{\boldsymbol{x}}_i)| = \sum_{A \subseteq \atop \neq} \sum_{\{1,\ldots,K\}} B_i^{|A|-1} \prod_{a \in A} \rho_{ia} \prod_{b \in \{1,\ldots,K\}/A} (\gamma_{ib} + \rho_{ib}). \tag{4.6}$$

Now, we can write our uncertainty matrix for the input data as follows:

$$\boldsymbol{U} = \begin{bmatrix} u_{11} & u_{12} & \ldots & u_{112} \\ u_{21} & u_{22} & \ldots & u_{212} \\ \vdots & \vdots & \ddots & \vdots \\ u_{381} & u_{382} & \ldots & u_{3812} \end{bmatrix} \in \begin{bmatrix} [3.525, -3.525] & 0 & \ldots & 0 \\ [3.802, -3.802] & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & [3.201, -3.201] & \ldots & [46.419, -46.419] \end{bmatrix}.$$

After we have incorporated uncertainty for each input value, matrices of our BFs can be expressed in the following forms, just by concentrating on the lower and upper interval boundaries, respectively:

$$\boldsymbol{W}_{upper} = \boldsymbol{\psi}(\breve{\boldsymbol{b}}) + \boldsymbol{U}_{upper} = \begin{bmatrix} 1 & 3.817 & \ldots & 0 \\ 1 & 3.817 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \ldots & 47.364 \end{bmatrix}, \tag{4.7}$$

$$\boldsymbol{W}_{lower} = \boldsymbol{\psi}(\breve{\boldsymbol{b}}) + \boldsymbol{U}_{lower} = \begin{bmatrix} 1 & -3.232 & \ldots & 0 \\ 1 & -3.787 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \ldots & -45.474 \end{bmatrix}. \tag{4.8}$$

The output data, the uncertainty vector and the vectors with uncertainty are represented below, respectively:

$$\boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{38} \end{bmatrix} \in \begin{bmatrix} [3,-3] \\ [3,-3] \\ \vdots \\ [3,-3] \end{bmatrix}, \boldsymbol{z}_{upper} = \breve{y} + \boldsymbol{v}_{upper} = \begin{bmatrix} -4.489 \\ -3.561 \\ \vdots \\ -1.874 \end{bmatrix}, \boldsymbol{z}_{lower} = \breve{\boldsymbol{y}} + \boldsymbol{v}_{lower} = \begin{bmatrix} 1.511 \\ 2.439 \\ \vdots \\ 4.127 \end{bmatrix}. \tag{4.9}$$

The calculation done above is applicable for both parts of our training data set.

4.5 Receiving Weak RCMARS Models Using Combinatorial Approach

As we mentioned in the previous section, PRSS is approximated by a TR problem, and we can easily formulate it as a CQP problem. Moreover, we incorporate a perturbation (uncertainty) into the real input data, $\breve{\boldsymbol{x}}_i$ $(i = 1, 2, \ldots, 38)$, in each dimension and into the output data, $\boldsymbol{y}$, by using our robust optimization approach for a robustification of CMARS. For this aim, by applying (3.6) and (3.7) we obtain the uncertainty matrices and vectors based on polyhedral uncertainty. Then, using relation (4.4) we evaluate uncertainty for all input and output values which are represented by CIs. The boundaries of CIs are assumed to be [-3, 3], after the variables are transformed into the standard normal distribution.

For our example, the uncertainty matrix for input data has a huge size, and we do not have enough computer capacity to solve our problem for this uncertainty matrix. Indeed, we have a *tradeoff* between tractability and robustification. To overcome that obstacle, in this example, we robustify our CQP problem for each sample value (observation) using the combinatorial approach, which we call *weak robustification*. As a result, we obtain 38 different *weak RCMARS* (*WRCMARS*) models, for each part of our data set, and resolve them with MOSEK [19].

Based on polyhedral uncertainty sets, to solve our problem, we use their vertices. In order to find them, we need especially to apply the Cartesian product of all the intervals of input data in the observations. Hence, our WRCMARS models have different structures depending on the number of entries (BFs), which are used to explain the observations. For instance, we can represent the last observations WRCMARS model, which has 3 entries, in the following form:

$$\begin{aligned}
&\underset{t,\boldsymbol{\alpha}}{\text{minimize}} \;\; t, \\
&\text{subject to} \quad 1.51069 - \alpha_0 - 0.29234\alpha_1 - 0.35539\alpha_4 = \beta_1, \\
&\qquad\qquad 2.43887 - \alpha_0 - 0.01516\alpha_1 - 0.10152\alpha_3 = \beta_2, \\
&\qquad\qquad \vdots \\
&\qquad\qquad -1.87353 - \alpha_0 + 2.677\alpha_2 + 3.090\alpha_3 + 45.474\alpha_5 = \beta_{320}, \\
&\qquad\qquad (\beta_1^2 + \beta_2^2 + \ldots + \beta_{20}^2)^{1/2} \leq t, \\
&\qquad\qquad (\beta_{21}^2 + \beta_{22}^2 + \ldots + \beta_{40}^2)^{1/2} \leq t, \\
&\qquad\qquad \vdots \\
&\qquad\qquad (\beta_{301}^2 + \beta_{302}^2 + \ldots + \beta_{320}^2)^{1/2} \leq t, \\
&\qquad\qquad (\beta_{321}^2 + \beta_{322}^2 + \beta_{323}^2 + \beta_{324}^2 + \beta_{325}^2)^{1/2} \leq \breve{M}^{1/2}.
\end{aligned}$$

In order to solve this problem, we transform it into the MOSEK format above. For this transformation, we attribute new unknown variables in the linear terms which are lying in these 17 cones. By this, in fact, we simplify the notations in the cones and write them as equality and inequality constraints. Therefore, for our last sample, our problem includes 325 linear constraints and 17 quadratic cones.

We write this formulation for each value of our sample ($N = 38$) and solve them separately by using MOSEK program [19]. MOSEK apply an interior-point optimizer, which is an implementation of a homogeneous and self-dual algorithm. We obtain MOSEK results and find the t values for all auxiliary problems; then, using the worst-case approach, we select the solution which has the *maximum* t value. Then we continue with our calculations using the parameter values α_j ($j = 1, 2, \ldots, 12$) that we find from the auxiliary problem which has the highest t value.

4.6 Sensitivity to the Changes in the Confidence Interval Limits of RCMARS

Here, the boundaries of CIs are supposed to be [-3, 3], after the variables are transformed into the standard normal distribution. In order to represent sensitivity to the changes in the CI limits of the input data and output data and to find suitable interval limits for us, we obtain different uncertainty matrices, $\boldsymbol{U}$, for the input data and different uncertainty vectors, $\boldsymbol{v}$, for the output data as the form of (3.11) by using 7 different intervals. These ones are given by the pairs $\pm 3, \pm 3/2, \pm 3/4, \pm 3/6, \pm 3/8, \pm 3/10$ and, as a special case, the mid-point value of our interval (i.e., zero lengths interval). In the *latter case*, it reduces to the CMARS model. This shows that CMARS is a *special case* of RCMARS. Therefore, we calculate our parameters with 7 different uncertainty scenarios using these values under polyhedral uncertainty sets for our training data set.

In Subsection 4.7, all of the parameter estimates as well as model accuracies for different uncertainty scenarios are shown. We note here that we defined the values $\sqrt{\widetilde{M}}$ by a model-free method. When we apply the $\sqrt{\widetilde{M}}$ values in our RCMARS code and solve it by MOSEK, we use that $\sqrt{\widetilde{M}}$ value which has the minimum value of PRSS approximately in equation (2.7). In order to compare the results concerning accuracy for RCMARS and CMARS, we employ *Average Absolute Error* (*AAE*) and *Root Mean Squared Error* (*RMSE*). Also, we represent variances (σ^2) of CMARS and RCMARS in Subsection 4.7.

4.7 Results

In this study, we construct uncertainty matrices, $\boldsymbol{U}$, for the input data and uncertainty vectors, $\boldsymbol{v}$, for the output data and, we recieve 7 different uncertainty scenarios by using the interval values, $\pm 3, \pm 3/2, \pm 3/4, \pm 3/6, \pm 3/8, \pm 3/10$ and zero.

From Tables 1 and 2 below it seems that the solutions obtained are sensitive to the limits of CIs. When the lengths of the CIs are narrow, we evaluate better performance results. Moreover, as in our previous study [21], when we use the *mid-point* (zero value) of our interval values for both input and output data, which is the certain data case; we receive the same parameter estimates as we obtained for CMARS. This is our particular special case.

The values $\sqrt{\widetilde{M}}$ in our application are defined by a model-free (train and error) method. When we assess the $\psi_m(\boldsymbol{x})$ values in our RCMARS code and employ MOSEK, RCMARS provides us several solutions, each of them based on 12 BFs.

For the training data, models for RCMARS have a smaller variance, but a lower accuracy than CMARS, which is consistent with our expectation. However, we have unexpected

Table 1: Parameter estimates and the model performances for the training data.

U, v	± 3	$\pm 3/2$	$\pm 3/4$	$\pm 3/6$	$\pm 3/8$	$\pm 3/10$	$zero$
	RCMARS	RCMARS	RCMARS	RCMARS	RCMARS	RCMARS	CMARS
α_0	-0.053	0.013	0.135	0.139	0.151	0.139	0.110
α_1	0.078	0.050	-0.040	-0.051	-0.065	-0.063	-0.061
α_2	0.008	0.016	0.009	0.010	0.006	-0.006	-0.024
α_3	-0.045	-0.059	-0.091	-0.103	-0.119	-0.138	-0.139
α_4	-0,021	-0.101	-0.175	-0.166	-0.164	-0.163	-0.155
α_5	0.000	-0.058	-0.113	-0.117	-0.122	-0.124	-0.118
α_6	0.031	0.052	0.066	0.063	0.063	0.072	0.085
α_7	0.054	0.016	-0.018	-0.011	-0.013	-0.007	0.008
α_8	0.216	0.451	0.497	0.470	0.473	0.474	0.453
α_9	-0.003	-0.008	-0.013	-0.007	-0.021	-0.001	0.082
α_{10}	0.001	0.001	0.002	0.002	0.002	0.004	-0.024
α_{11}	-0.002	-0.018	-0.031	-0.022	-0.013	-0.007	-0.066
α_{12}	-0.005	-0.005	-0.004	-0.004	0.006	0.012	0.038
$\boldsymbol{\sigma^2}$	**0.028**	**0.057**	**0.085**	**0.085**	**0.092**	**0.101**	**0.165**
AAE	0.735	0.707	0.678	0.673	0.662	0.656	0.627
RMSE	1.175	1.121	1.078	1.070	1.052	1.037	0.999

results for the testing data.

Table 2: Parameter estimates and the model performances for the testing data.

U, v	± 3	$\pm 3/2$	$\pm 3/4$	$\pm 3/6$	$\pm 3/8$	$\pm 3/10$	$zero$
	RCMARS	RCMARS	RCMARS	RCMARS	RCMARS	RCMARS	CMARS
$\boldsymbol{\sigma^2}$	**0.005**	**0.006**	**0.005**	**0.005**	**0.005**	**0.006**	**0.012**
AAE	0.830	0.831	0.818	0.818	0.812	0.814	0.825
RMSE	1.156	1.163	1.146	1.145	1.138	1.145	0.168

For the testing data and for some suitable uncertainty values, RCMARS produced more accurate model with a smaller variance than CMARS, which can be seen in Table 2. This is mainly due to the randomness involved in the input-output variables. According to the above results we can say that RCMARS can be a more *accurate model* with a **smaller variance** than CMARS.

5 Conclusion and Further Studies

Some models assume that the returns follow a multivariate normal distribution. For example, Markowitz model assumes that the first two moments of the distribution suffice to completely describe the distribution of the asset returns and the characteristics of the different portfolios. But in the real life, these models are too simplistic, leading to parameter instability [17], because of the increased uncertainty after the recent crises. In order to get reliable results

of volatility or risk of the investment, we have to incorporate this uncertainty into the model [10].

It is well known that the variance and standard deviation are especially useful measures of risk for the variables that are normally distributed or for those that can be represented by normal distribution. This distribution is very useful in finance because the returns for many assets and, hence, the indexes tend to be normally distributed, which makes variance and standard deviation practical measures of the uncertainty associated with investment returns, credit defaults, etc..

In our study, we employ a regression algorithm called RCMARS with training data and check the results using testing data. For the training data, RCMARS models show a smaller variance, but a lower accuracy than CMARS, which is consistent with our expectation, because we expect to see that the variation of the parameter estimates and, hence, the variation of accuracy measures, will be much less than that of CMARS. However, for the testing data, there were unexpected results because for some uncertainty values, RCMARS produced a more accurate model with a smaller variance than CMARS. As a result, we can say that RCMARS has been more accurate model with, what is very important, a smaller variance than CMARS. From the financial point of view we may deduce that RCMARS models can give more accurate strategies to implement with a relatively low risk.

Above we indicated that in this study, we used normally distributed data. As a future project, we will develop a model that will successfully work with other types of distributions. We also had to divide the training data set into two parts, because of the capacity problems of the computer. In future, we will use different computational methods, such as parallel computing, to overcome this difficulty. Moreover, we will test the RCMARS performance on different data, taken from inside and outside of financial market sector.

In all of our future studies, we go on facing the complexity of our model and trying to turn all model-free, e.g., trial-and-error, sides of our treatment, into a model-based form. In particular, we plan to reinterpret the parametric bound $\breve{M}$ as another state variable (unknown), including it into the objective function also. Herewith, we would still remain in our "conic" setting of CQP. This would lead to another support and strengthening of the model-basedness of our approach and would make it even more rigorous mathematically. Modern continuous and global optimization will certainly be a key-technology for this. We can also diversify our optimization by differentiating between different values of the penalty parameters. This would lead to further *control* variables.

References

[1] E. Andreou, E. Ghysels and A. Kourtellos, Should Macroeconomic Forecasters Use Daily Financial Data and How?, Available at SSRN: http://ssrn.com/abstract=1711899, 2010.

[2] R.C. Aster, B. Borchers and C. Thurber, *Parameter Estimation and Inverse Problems*, Academic Press, Boston, 2004.

[3] A. Ben-Tal and A. Nemirovski, Robust convex optimization, *Math. Oper. Res.* 23 (1998) 769–805.

[4] A. Ben-Tal and A. Nemirovski, Robust solutions to uncertain linear programs, *Oper. Res. Lett.* 25 (1999) 1–13.

[5] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*, MPR-SIAM Series on Optimization, SIAM, Philadelphia, 2001.

[6] A. Ben-Tal and A. Nemirovski, Robust Optimization methodology and applications, *Math. Program.* 92 (2002) 453–480.

[7] D. Bertsimas, D.B. Brown and C. Caramanis, Theory and applications of robust optimization, Tech. Rep., University of Texas, Austin, TX, USA, 2007.

[8] O. Boni, Robust Solutions of conic quadratic problems, Ph.D. diss., Technion, Israeli Institute of Technology, 2007.

[9] Central Bank of the Republic of Turkey: http://www.tcmb.gov.tr

[10] G. Corsetti, M. Pericoli and M. Sbracia, Correlation Analysis of Financial Contagion, Book chapter, Financial Contagion: The Viral Threat to the Wealth of Nations, Robert W. Kolb, ISBN 978-0-470-92238-5

[11] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*, Boca Raton, FL: Chapman and Hall/CRC, 1993.

[12] L. El-Ghaoui and H. Lebret, Robust solutions to least-square problems to uncertain data matrices, *SIAM J. Matrix Anal. Appl.* 18 (1997) 1035–1064.

[13] L. El-Ghaoui, *Robust Optimization and Applications*, IMA Tutorial, 2003.

[14] F.J. Fabozzi, P.N. Kolm, D.A. Pachamanova and S.M. Focardi, *Robust Portfolio Optimization and Management*, Wiley Finance, New Jersey, 2007.

[15] J.H. Friedman, Multivariate adaptive regression splines, *The Ann. Statist.* 19 (1991) 1–91.

[16] T. Hastie, R. Tibshirani and J.H. Friedman, *The Element of Statistical Learning*, Springer Verlag, New York, 2001.

[17] W.P. Louis, Financial Crisis Management in Europe and Beyond, *Contribution to Political Economy* 27 (2008) 73–89.

[18] MARS®Salford Systems; software available at http://www.salfordsystems.com.

[19] MOSEK$^{\mathrm{TM}}$; software available at http://www.mosek.com.

[20] A. Özmen, G.-W. Weber and I. Batmaz, The new robust CMARS (RCMARS) method, in *ISI Proceedings of 24th MEC-EurOPT 2010 Continuous Optimization and Information-Based Technologies in the Financial Sector*, Izmir, Turkey 2010, pp. 362-368.

[21] A. Özmen, G.-W. Weber, I. Batmaz and E. Kropat, RCMARS: Robustification of CMARS with different scenarios under polyhedral uncertainty set, *Commun. Nonlin. Sci. Num. Simul.* 16 (2011) 4780–4787.

[22] A. Özmen, Robust conic quadratic programming applied to quality improvement- A robustification of CMARS, M.Sc. Thesis, Institute of Applied Mathematics, METU, 2010.

[23] K. Schöttle and R. Werner, Consistency of robust portfolio estimators, Optimization-Online, 2006.

[24] G.-W. Weber, I. Batmaz, G. Köksal, P. Taylan and F. Yerlikaya, CMARS: A new contribution to nonparametric regression with multivariate adaptive regression splines supported by continuous optimization, *Inverse Probl. Sci. and Engineering (IPSE)* 20 (2012) 371–400.

[25] R. Werner, Cascading: an adjusted exchange method for robust conic programming, *CEJOR, Cent. Eur. J. Oper. Res.* 16 (2008) 179–189.

Manuscript received 16 February 2012
revised 13 August 2012, 22 November 2012
accepted for publication 23 November 2012

A.Özmen
Middle East Technical University, Institute of Applied Mathematics, 06800, Ankara, Turkey
E-mail address: ayseozmen19@gmail.com

G.-W. Weber
Middle East Technical University, Institute of Applied Mathematics, 06800, Ankara, Turkey
E-mail address: gweber@metu.edu.tr

A. Karimov
Middle East Technical University, Institute of Applied Mathematics, 06800, Ankara, Turkey
E-mail address: azar.karimov@metu.edu.tr