How to Fit a Ridge-Regression Model with the Practice of a QCQP-Solver- an Example from Operations Research

Andreas Kagoshima, *M. Sc. Student, Mathematisches Institut Albert-Ludwig-Universität*

***Abstract*—The paper shows a simple task-setting from the field of operations research. This setting is the basis for an example of how to fit a ridge-regression model with the application of a QCQP-Solver.**

The numeric data-vectors which are being used were created artificially. The correspondent command lines that were utilized to create the data are presented in the appendix.

Highlights are:

- **high-dimensional numeric data-vectors (dimension 185)**
- **non-linear data-vector transformation**
- **transformation of a QP into a QCQP with arguments of convergence and arguments of SVD**
- **application of a QCQP-solver (yaLMIp).**

The main-part of the source-code can be found on https://github.com/kago1988/NumOpt-Operations-Research.

***Index Terms*—Ridge-Regression, QP, QCQP, SVD, Moore-Penrose, Economy**

I. Introduction

The sales-department of a car-company wants to set up financial plans for the rest of the year and requires a statistical prediction for the number of car-sales in the next quarter.

It decides to make use of a data-vector $X_{\text{Q3, 2016}}$,

$$X_{\text{Q3, 2016}} = \begin{bmatrix} \text{gdp} \\ \text{export.volume} \\ \text{exhaust.gas} \\ \text{popularity} \\ \text{"car.of.the.quarter"} \\ \text{expand.to.new.country} \end{bmatrix}_{\text{Q3, 2016}} \in \mathbb{R}^6,$$

which becomes available at the end the quarter Q3, 2016 and contains quarterly-annual parameters, in order to make the prediction

$$\hat{Y}_{\text{Q4, 2016}} = \begin{bmatrix} \text{prediction.small.cars} \\ \text{prediction.medium.cars} \\ \text{prediction.big.cars} \end{bmatrix}_{\text{Q4, 2016}} \in \mathbb{R}^3.$$

Furthermore there are historical data $\boldsymbol{X}$ and $\boldsymbol{Y}$ available, which contain correspondent data-vectors, listed in transposed form. They range from Q1, 1970 to Q2, 2016.

$$\boldsymbol{X} = \begin{bmatrix} 1 & X^T_{\text{Q1, 1970}} \\ \vdots & \vdots \\ \vdots & \vdots \\ 1 & X^T_{\text{Q2, 2016}} \end{bmatrix} \in \mathbb{R}^{182\times 7},$$

$$\boldsymbol{Y} = \begin{bmatrix} Y^T_{\text{Q1, 1970}} \\ \vdots \\ \vdots \\ Y^T_{\text{Q3, 2016}} \end{bmatrix} \in \mathbb{R}^{182\times 3},$$

$$Y_{\text{Qx, yyyy}} = \begin{bmatrix} \text{small.cars.sold} \\ \text{medium.cars.sold} \\ \text{big.cars.sold} \end{bmatrix}_{\text{Qx, yyyy}}$$

After some weighing up the sales-department decides for a ridge-regression model.

This means they want to utilize the affine-linear relationship

$$\hat{Y}^T_{\text{Q4, 2016}} = \begin{bmatrix} 1 & X^T_{\text{Q3, 2016}} \end{bmatrix} \cdot \hat{\boldsymbol{\beta}}^{\text{ridge}},$$

where $\hat{\boldsymbol{\beta}}^{\text{ridge}} \in \mathbb{R}^{7\times 3}$ is a weight-matrix that was optimized according to a ridge-regression model.

II. Parameter Specification

The estimation

$$\hat{\boldsymbol{f}} : \begin{bmatrix} \text{gdp} \\ \text{export.volume} \\ \text{exhaust.gas} \\ \text{popularity} \\ \text{"car.of.the.quarter"} \\ \text{expand.to.new.country} \end{bmatrix}_{\text{Q2, 2016}} \to \begin{bmatrix} \text{estimation.small.cars} \\ \text{estimation.medium.cars} \\ \text{estimation.big.cars} \end{bmatrix}_{\text{Q3, 2016}}$$

is to be made where

- gdp (= gross domestic product), export.volume and exhaust.gas are factors of national economy,
- popularity is an integer percentage describing the amount of people in the population who favor the car-company over other car-companies and
- "car.of.the.quarter" and expand.to.new.country are constantly set to "+1", if a car-companys car is voted "car of the quarter", respectively if the car-company expands to a new country, and are otherwise set to "+0".

$\hat{\boldsymbol{f}}$ can be seen as an affine linear function, $\mathbb{R}^6 \to \mathbb{R}^3$ which is fully determined by the optimized weight-matrix $\hat{\boldsymbol{\beta}}^{\text{ridge}} \in \mathbb{R}^{(6+1)\times 3}$.

Optimizing the weight-matrix is the main-task and can be accomplished by calculating each column $\hat{\boldsymbol{\beta}}_i^{\text{ridge}} \in \mathbb{R}^7$, $i = 1, 2, 3$ independently with one QCQP.

III. Numeric Data-Vectors

Because $\boldsymbol{X} \in \mathbb{R}^{182\times 7}$ is defined as

$$\boldsymbol{X} = \begin{bmatrix} 1 & X^T_{\text{Q1, 1970}} \\ \vdots & \vdots \\ \vdots & \vdots \\ 1 & X^T_{\text{Q2, 2016}} \end{bmatrix}$$

We can write the columns $\boldsymbol{X}_i$, $i = 0, \ldots, 6$ as the numeric vectors:

$$\boldsymbol{X}_0 = \begin{bmatrix} 1 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}, \boldsymbol{X}_1 = \begin{bmatrix} \text{gdp}_{\text{Q1, 1970}} \\ \vdots \\ \vdots \\ \text{gdp}_{\text{Q2, 2016}} \end{bmatrix},$$

$$\boldsymbol{X}_2 = \begin{bmatrix} \text{export.volume}_{\text{Q1, 1970}} \\ \vdots \\ \vdots \\ \text{export.volume}_{\text{Q2, 2016}} \end{bmatrix}, \boldsymbol{X}_3 = \begin{bmatrix} \text{exhaust.gas}_{\text{Q1, 1970}} \\ \vdots \\ \vdots \\ \text{exhaust.gas}_{\text{Q2, 2016}} \end{bmatrix},$$

$$\boldsymbol{X}_4 = \begin{bmatrix} \text{popularity}_{\text{Q1, 1970}} \\ \vdots \\ \vdots \\ \text{popularity}_{\text{Q2, 2016}} \end{bmatrix}, \boldsymbol{X}_5 = \begin{bmatrix} \text{"car.of.the.quarter"}_{\text{Q1, 1970}} \\ \vdots \\ \vdots \\ \text{"car.of.the.quarter"}_{\text{Q2, 2016}} \end{bmatrix},$$

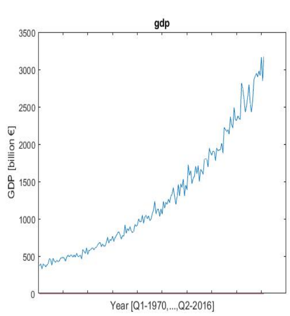
$$\boldsymbol{X}_6 = \begin{bmatrix} \text{expand.to.new.country}_{\text{Q1, 1970}} \\ \vdots \\ \vdots \\ \text{expand.to.new.country}_{\text{Q2, 2016}} \end{bmatrix}.$$

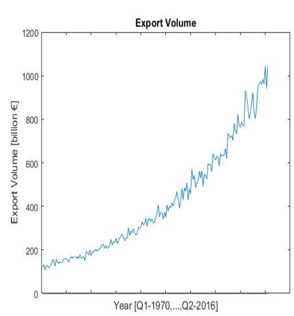
And the columns of $\boldsymbol{Y}_i$, $i = 1, 2, 3$ as:

$$\boldsymbol{Y}_1 = \begin{bmatrix} \text{small.cars.sold}_{\text{Q2, 1970}} \\ \vdots \\ \vdots \\ \text{small.cars.sold}_{\text{Q3, 2016}} \end{bmatrix}, \boldsymbol{Y}_2 = \begin{bmatrix} \text{medium.cars.sold}_{\text{Q2, 1970}} \\ \vdots \\ \vdots \\ \text{medium.cars.sold}_{\text{Q3, 2016}} \end{bmatrix},$$

$$\boldsymbol{Y}_3 = \begin{bmatrix} \text{big.cars.sold}_{\text{Q2, 1970}} \\ \vdots \\ \vdots \\ \text{big.cars.sold}_{\text{Q3, 2016}} \end{bmatrix},$$

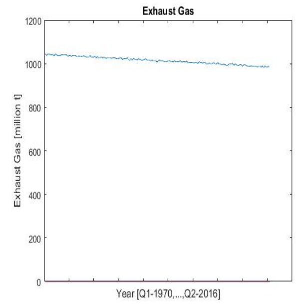
Some plots are shown below:

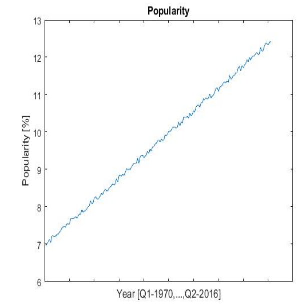


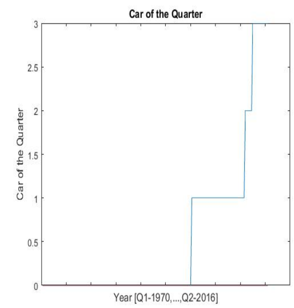


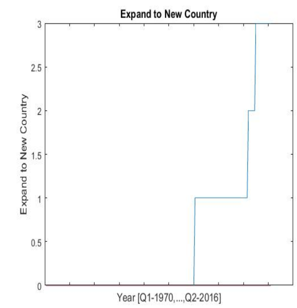
First Problem: Here we can see, that gdp and export.volume have exponential growth.

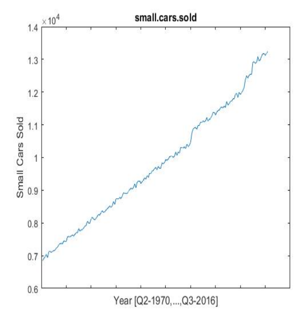
Because these are exponential growing input-variables (predictor variables), but the goal-variables (target variables) are growing linearly, gdp and export.volume will be logarithmized.

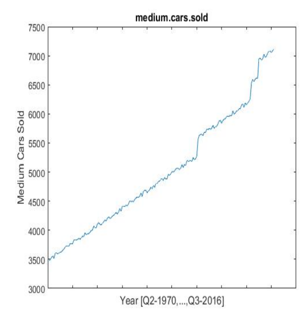


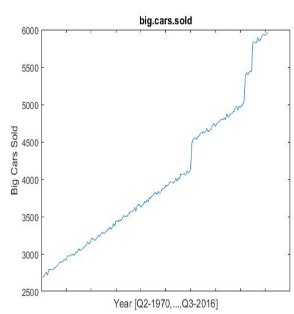












Second Problem: Numeric accurateness is an issue and therefore calculating with the gdp should happen in the parameter-intervall of 360 to 3500 (and not the parameter-intervall of $360 * 10^6$ to $3500 * 10^6$).

Third Problem: We already see, that the columns "car.of.the.quarter" and expand.to.new.country" are not linearly independent, which leads to the conclusion that $(\boldsymbol{X}^T\boldsymbol{X})$ is not invertible. This means we cannot compute the solution of an ordinary least squares regression (OLS) algebraically. Trying to do so would lead to a set of solutions lying on a straight line (because $(\boldsymbol{X}^T\boldsymbol{X})$ is under-determined in one rank). This would have been hard to justify from a statistical point of view.

IV. Calculation of $\hat{\boldsymbol{\beta}}^{\text{ridge}}$

The fit of a ridge-regression model $\mathbb{R}^n \to \mathbb{R}^l$ starts exactly like the fit of an ordinary least squares regression (OLS) with the affine linear relationship

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{X} \cdot \boldsymbol{\beta} + \boldsymbol{\varepsilon}, \\ \iff \boldsymbol{Y}_i &= \boldsymbol{X} \cdot \boldsymbol{\beta}_i + \boldsymbol{\varepsilon}_i, \quad i = 1, \dots, l \\ \iff \langle \boldsymbol{\varepsilon}_i, \boldsymbol{\varepsilon}_i \rangle &= (\boldsymbol{Y}_i - \boldsymbol{X} \cdot \boldsymbol{\beta}_i)^T (\boldsymbol{Y}_i - \boldsymbol{X} \cdot \boldsymbol{\beta}_i) \end{aligned}$$

whereas the optimized weight-matrix $\hat{\boldsymbol{\beta}}^{\text{ridge}}$ now comes from calculating each column $\hat{\boldsymbol{\beta}}_i^{\text{ridge}}$ independently with

$$\boxed{\min_{\boldsymbol{\beta}_i} \quad \frac{1}{2}(\boldsymbol{Y}_i - \boldsymbol{X} \cdot \boldsymbol{\beta}_i)^T (\boldsymbol{Y}_i - \boldsymbol{X} \cdot \boldsymbol{\beta}_i) + \frac{\lambda}{2} \boldsymbol{\beta}_i^T \boldsymbol{\beta}_i \quad \lambda > 0 \qquad (1)}$$

Which is a fully ranked QP, because it can be written in the form

$$\boxed{\min_{\boldsymbol{\beta}_i} \quad \frac{1}{2} \boldsymbol{\beta}_i^T (\boldsymbol{X}^T \boldsymbol{X} + \lambda I) \boldsymbol{\beta}_i - (\boldsymbol{Y}_i^T \boldsymbol{X}) \boldsymbol{\beta}_i \quad \lambda > 0 \quad (2)}$$

This QP immediately yields the solution

$$\hat{\boldsymbol{\beta}}_i^{\text{ridge}}(\lambda) = (\boldsymbol{X}^T \boldsymbol{X} + \lambda I)^{-1} \boldsymbol{X}^T \boldsymbol{Y}_i \quad \text{(cFOSC)}. \qquad (3)$$

This already is the optimal solution for $\hat{\boldsymbol{\beta}}_i^{\text{ridge}}$. But since the goal of this paper was to solve the QP (2) numerically with a QCQP-solver, we will use (3) only to transform (2) into the QCQP (4).

For justification it is important to take a closer look at the singular value decomposition (SVD) of $\boldsymbol{X}$. Let $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T \in \mathbb{R}^{m \times n}$, $m \ll n$, be the SVD of $\boldsymbol{X}$, where $\boldsymbol{U} \in \mathbb{R}^{m \times m}$, $\boldsymbol{V} \in \mathbb{R}^{n \times n}$ are orthogonal matrices and $\boldsymbol{S} \in \mathbb{R}^{m \times n}$ is an diagonal matrix of rank r with positive-definite but shrinking entries $s_1 \geq s_2 \geq \cdots \geq s_r$.

By using the SVD it is already possible to receive simple convergence properties of $\hat{\boldsymbol{\beta}}_i^{\text{ridge}}(\lambda)$, when $\lambda \to 0$, respectively when $\lambda \to \infty$:

$$\begin{aligned} &(\boldsymbol{X}^T \boldsymbol{X} + \lambda I_n)^{-1} \boldsymbol{X}^T \\ &= \left(\left(\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T \right)^T \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T + \lambda I_n \right)^{-1} \left(\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T \right)^T \\ &= \left(\boldsymbol{V}\boldsymbol{S}^T\boldsymbol{S}\boldsymbol{V}^T + \lambda I_n \right)^{-1} \left(\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T \right)^T \\ &= \boldsymbol{V} \left(\boldsymbol{S}^T\boldsymbol{S} + \lambda I_n \right)^{-1} \boldsymbol{S}^T \boldsymbol{U}^T \\ &= \boldsymbol{V} \begin{bmatrix} \frac{s_1}{s_1^2+\lambda} & & & & 0 & \dots & \dots & \dots & \dots & 0 \\ & \frac{s_2}{s_2^2+\lambda} & & & \vdots & & & & & \vdots \\ & & \ddots & & \vdots & & & & & \vdots \\ & & & \frac{s_r}{s_r^2+\lambda} & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ \vdots & & & & & & & & & \vdots \\ 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 \end{bmatrix} \boldsymbol{U}^T \\ &\to \begin{cases} 0, & \text{for } \lambda \to \infty, \\ \boldsymbol{V}\boldsymbol{S}^+\boldsymbol{U}^T = \boldsymbol{X}^+, & \text{for } \lambda > 0, \lambda \to 0. \end{cases} \end{aligned}$$

Here $\boldsymbol{S}^+ \in \mathbb{R}^{n \times m}$ is the Moore-Penrose pseudo-inverse. It is clearly defined as

$$\boldsymbol{S}^+ := \begin{bmatrix} \frac{1}{s_1} & & & & 0 & \dots & \dots & \dots & \dots & 0 \\ & \frac{1}{s_2} & & & \vdots & & & & & \vdots \\ & & \ddots & & \vdots & & & & & \vdots \\ & & & \frac{1}{s_r} & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ \vdots & & & & & & & & & \vdots \\ 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 \end{bmatrix} \in \mathbb{R}^{n \times m}.$$

The Moore-Penrose pseudo-inverse of $\boldsymbol{X}$ is then defined as $\boldsymbol{X}^+ := \boldsymbol{V}\boldsymbol{S}^+\boldsymbol{U}^T$.

The following theorems (which we will not be proven in detail) are now easier to understand:

Theorem 1 It holds:

$$\lim_{\substack{\lambda \to 0 \\ \lambda > 0}} \hat{\boldsymbol{\beta}}_i^{\text{ridge}}(\lambda) = \lim_{\substack{\lambda \to 0 \\ \lambda > 0}} (\boldsymbol{X}^T \boldsymbol{X} + \lambda I_n)^{-1} \boldsymbol{X}^T \boldsymbol{Y}_i = \boldsymbol{X}^+ \boldsymbol{Y}_i$$

Theorem 2 It holds:

$$\lim_{\lambda \to \infty} \hat{\boldsymbol{\beta}}^{\text{ridge}}(\lambda) = \lim_{\lambda \to \infty} (\boldsymbol{X}^T \boldsymbol{X} + \lambda I_n)^{-1} \boldsymbol{X}^T \boldsymbol{Y}_i = 0.$$

Theorem 3 For $\lambda \in (0, \infty)$ $\hat{\boldsymbol{\beta}}_i^{\text{ridge}}(\lambda)$, travels on a straight line between $\boldsymbol{X}^+ \boldsymbol{Y}_i$ and 0. (Which could be proven by $\frac{\partial \hat{\boldsymbol{\beta}}_i^{\text{ridge}}(\lambda)}{\partial \lambda} / \| \frac{\partial \hat{\boldsymbol{\beta}}_i^{\text{ridge}}(\lambda)}{\partial \lambda} \|_2 = \text{constant}$).

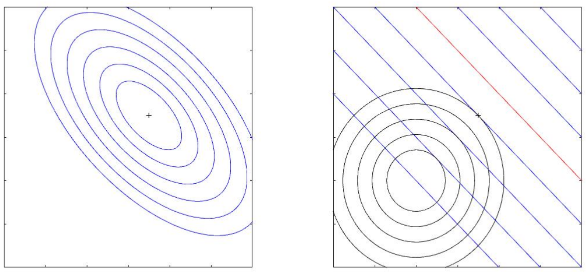
Theorem 4 It holds: With growing $\lambda \in (0, \infty)$, $\|\hat{\boldsymbol{\beta}}^{\text{ridge}}(\lambda)\|_2^2$ decreases.

It is now possible to conclude from **Theorem 3** in combination with **Theorem 4** that the QP (2) can be translated one-to-one into the QCQP (4), where the choice for t is $t := \|\hat{\boldsymbol{\beta}}_i^{\text{ridge}}\|_2^2$.

$$\boxed{\begin{aligned} \min_{\boldsymbol{\beta}_i} \quad & \frac{1}{2}(\boldsymbol{Y}_i - \boldsymbol{X} \cdot \boldsymbol{\beta}_i)^T (\boldsymbol{Y}_i - \boldsymbol{X} \cdot \boldsymbol{\beta}_i) && \text{(4a)} \\ \text{s.t.} \quad & \boldsymbol{\beta}_i^T \boldsymbol{\beta}_i \leq t \qquad t > 0 && \text{(4b)} \end{aligned}}$$

The reason for this choice is simple: Solving (4) will result in the closest approximation of $(\boldsymbol{X}^+\boldsymbol{Y}_i)$ under the constraint $\|\boldsymbol{\beta}_i\|_2^2 \leqslant \|\hat{\boldsymbol{\beta}}_i^{\text{ridge}}\|_2^2$ which is exactly equivalent to solving (2).

A simple illustration of countour lines can give an intuitive idea what has been achieved in this section. This section justified the transformation of the left side (QP) into the right side (QCQP).



The red line illustrates, that trying to solve (4a) alone leads to a set of solutions lying on an affine linear subspace (because $(\boldsymbol{X}^T\boldsymbol{X})$ is in our example under-determined).

V. Fitting the Ridge Regression-Model with matlab using yaLMIp

Each column $\hat{\boldsymbol{\beta}}_i^{\text{ridge}}$ can be calculated independently with one QP or one QCQP-solver. We will refer to the first variant as $\hat{\boldsymbol{\beta}}_i^{\text{ridge, QP}}$ and to the second variant as $\hat{\boldsymbol{\beta}}_1^{\text{ridge, QCQP}}$.

The direct approach for the application of a QCQP-solver is here briefly explained: First of all we our sampling-technique for the target-variables is:

$$
\begin{aligned}
Y_1 =& small.cars.sold \\
=& 0.1 * log(gdp) + 0.3 * log(export.volume) \\
&- 0.1 * exhaust.gas + 1000 * popularity \\
&+ 200 * car.of.the.quarter \\
=& + 100 * expand.to.new.country \\
Y_2 =& medium.cars.sold \\
=& 0.15 * log(gdp) + 0.5 * log(export.volume) \\
&- 0.01 * exhaust.gas + 500 * popularity \\
&+ 100 * car.of.the.quarter \\
&+ 200 * expand.to.new.country \\
Y_3 =& big.cars.sold \\
=& 0.7 * log(gdp) + 0.35 * log(export.volume) \\
&- 0.1 * exhaust.gas + 400 * popularity \\
&+ 60 * car.of.the.quarter \\
&+ 300 * expand.to.new.country
\end{aligned}
$$

One could have added error-terms again but this practice was omitted in order to keep the paper more tractable. From these data-vectors one is able to calculate:

$$\hat{\boldsymbol{\beta}}_i^{\text{ridge, QP}}(\lambda) = (\boldsymbol{X}^T\boldsymbol{X} + \lambda I)^{-1}\boldsymbol{X}^T\boldsymbol{Y}_i$$

where, for simplicity, the choice of λ is $\lambda = 1$.

Now one is able to use the QCQP-solver yaLMIp in order to solve:

$$
\boxed{
\begin{aligned}
\min_{\boldsymbol{\beta}_i} \quad & \frac{1}{2}(\boldsymbol{Y}_i - \boldsymbol{X}\cdot\boldsymbol{\beta}_i)^T(\boldsymbol{Y}_i - \boldsymbol{X}\cdot\boldsymbol{\beta}_i) && \text{(4a)} \\
\text{s. t.} \quad & \boldsymbol{\beta}_i^T\boldsymbol{\beta}_i \leq t \qquad t > 0 && \text{(4b)}
\end{aligned}
}
$$

with $t = \|\hat{\boldsymbol{\beta}}^{\text{ridge}}\|_2^2$. (For more detail see the source-codes "calculateYi.m", "calculateBi.m" and "calculateBiWithQCQPSolver.m").

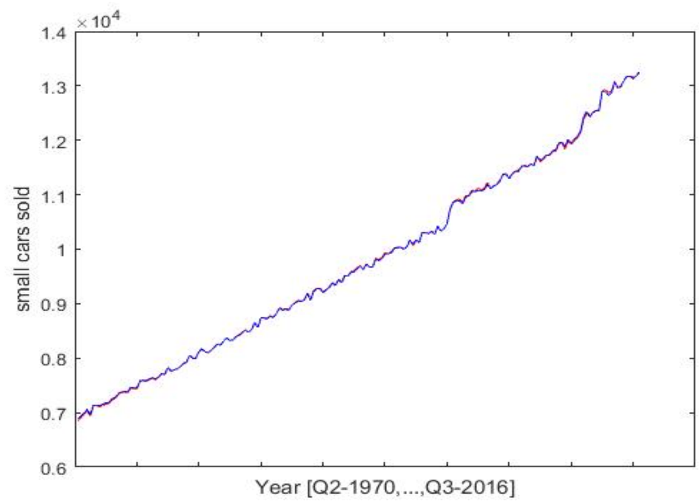
Comparing the optimal solution $\hat{\boldsymbol{\beta}}_i^{\text{ridge, QP}}$ with the numerical solution $\hat{\boldsymbol{\beta}}_i^{\text{ridge, QCQP}}$ we can see a good approximation:

$$
\hat{\boldsymbol{\beta}}_1^{\text{ridge, QP}} = \begin{bmatrix} 9.3269 \\ 0.6421 \\ -1.5815 \\ 0.1451 \\ 959.8200 \\ 132.7400 \\ 132.7400 \end{bmatrix}, \quad
\hat{\boldsymbol{\beta}}_1^{\text{ridge, QCQP}} = \begin{bmatrix} 9.3268 \\ 0.6421 \\ -1.5815 \\ 0.1451 \\ 959.8199 \\ 132.7404 \\ 132.7404 \end{bmatrix},
$$

$$
\hat{\boldsymbol{\beta}}_2^{\text{ridge, QP}} = \begin{bmatrix} 4.7022 \\ 0.3526 \\ -0.8611 \\ 0.1245 \\ 478.0700 \\ 139.3500 \\ 139.3500 \end{bmatrix}, \quad
\hat{\boldsymbol{\beta}}_2^{\text{ridge, QCQP}} = \begin{bmatrix} 4.7022 \\ 0.3526 \\ -0.8611 \\ 0.1245 \\ 478.0682 \\ 139.3532 \\ 139.3532 \end{bmatrix},
$$

$$
\hat{\boldsymbol{\beta}}_3^{\text{ridge, QP}} = \begin{bmatrix} 3.7935 \\ 0.3060 \\ -0.7409 \\ 0.0180 \\ 381.0400 \\ 169.8500 \\ 169.8500 \end{bmatrix}, \quad
\hat{\boldsymbol{\beta}}_3^{\text{ridge, QCQP}} = \begin{bmatrix} 3.7935 \\ 0.3060 \\ -0.7409 \\ 0.0180 \\ 381.0374 \\ 169.8529 \\ 169.8529 \end{bmatrix}.
$$

Plotting the target-variables "small.cars.sold" (red line) over "year" and comparing it with yaLMIps prediction (blue line)



one can see, that the 7-dimensional fit fits almost perfect to the simple example.

VI. Critical Discussion

- The task-setting is very simple. "Year" itself would have been a very good predictor, because the car-sales grow linearly.

- Year alone would have yielded a straight line as a prediction.

VII. Appendix: Command Lines that were used for the Numeric Vectors

create predictor variables: On the website www.statista.de there are many economic data available. For example the GDP of germany in 1970 added up to EUR 360.6 billion and the GDP in 2015 to EUR 3025.9 billion. Further GDP and export.volume are often described as growing exponentially. So we can apply the command line:

```
for(i in 1:182)
gdp[i]
= 360*exp((i-1)
*(log((3025.9*0.25)/(360*0.25))/179))
+ error
```

the terms 0.25 come from the fact that we have quarterly-annual data-vectors. For the export volume we take:

```
export.volume = (1/3)*gdp + error
```

For the exhaust gases we have the numbers: 1005 million t in year 2000 and 935 million t in year 2014. We apply linear, descending growth with slope $-(935*0.25-1005*0.25)/55)$ (Because in between there are 55 quarters):

```
for(i in 1:182)
exhaustGas[i]
= 1005+120*(935*0.25-1005*0.25)/55)
- (i-1)*(935*0.25-1005*0.25)/55) + error
```

The other data-vectors were created very similar. (For more detail see the source-codes in the folder "../source codes/numeric vectors and plots").

In order to make the numeric data-vectors more realistic, gdp and export.volume have exponential growing errors, while exhaust.gas for example has identical distributed errors for each quarter.

The main-part of the source-code can be found on https://github.com/kago1988/NumOpt-Operations-Research.

VIII. List of References

- www.statista.de
- Prof. M. Diehl- Lecture: Numerical Optimization (SS 2016, Uni Freiburg)
- Hastie, Tibshirani, Friedman- The Elements of Statistical Learning