Gas sensor array temperature modulation

Mame Diarra Toure-Imane Alla: Binome 20

11/4/2019

Introduction

We've seen in our previous corrplot that our target variable (CO) is strongly correlated (negatively) to the variables (R8,. . . .,R14). Our aim here is te determine which sensors are the best to explaint(predict) the CO concentration (target variable). So we decided after computing our baseline model to proceed to a variable selection in order to select the best sensors.

I. Baseline model

Our baseline model (CO accroding to all the explanatory variables) gives us an Residual standard error(RSE) of 4.246 and an adjusted R-squared of 0.5635 We plot our estimated target with respect to our real target and then superimpose the first bissextrix (y=x) If the values are perfectly predicted, we expect to see points along the y = x line.The plot(see section plot below) shows us that our estimation is quite different from the real values. So we are going to try a small transformation to see if we can obtain a better fit

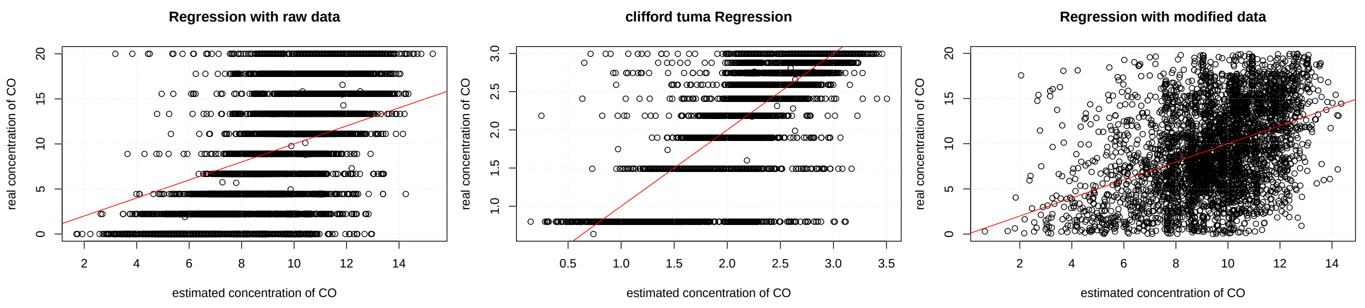
Cliffor Tuma Model

According to the paper "Multivariate estimation of the limit of detection by orthogonal partial least squares in temperature-modulated MOX sensors"; For CO detection, the most widely used method is the **Clifford-Tuma model**. This empirical model uses a linear relationship between the logarithm of the sensor conductance g_s(kΩ^-1)(the conductance is the inverse of the resistance) and the logarithm of the analyte concentration of CO (ppm) log(c). $log(g_s) = \alpha + \beta log(c)$ We transformed our data taking off all O values of CO (because we compute log(c)) then replace the CO colum with Log(CO) and the resitance columns with $log(\frac{1}{R_i})$(which represent the conductances)Our trasformation did not give us a better fit of our estimations to the real values (see section plots below) neither did feature selection or penalized regression.

Breakthrough idea

What we have notice in the dataset is that a there a 10 values of concentrations 0 2.22 4.44 6.67 8.89 11.11 13.33 15.56 17.78 20 that have a frequency greater than 300000 . **We think that those values influence negatively our regression so we got rid of them and used the resulting data to do our regression.** The new baseline model gives us a better superimposition (see section plots below) that we hope will improve with variable selection.

Plots of $(Y, \hat{Y})$



Feature Selection

Stepwise regression is a method of selecting independent variables in order to choose a set of predictors that have the best relationship with the dependent variable. In the R package the step function uses the AIC criterion for weighing the choices, which takes proper account of the number of parameters fit; at each step an add or drop will be performed that minimizes the AIC score. We've performed Stepwise, Forward and backward Selection The backward selection gives us the following model $y = CO(concetration) = Time + Humidity + Flow.rate + Heater.voltage + R1 + R7 + R10 + R11 + R12 + R13$

The forward selection gives us the following model $y = CO(concetration) = R10 + Heater.voltage + R4 + Time + Flow.rate + Humidity + R1 + R11 + R13 + R12 + R7 + R8$

The stepwise selection gives us the following model $y = CO(concetration) = R10 + Heater.voltage + Time + Flow.rate + Humidity + R1 + R11 + R13 + R12 + R7 + R8$

the six sensors R1, R7, R10, R11, R12, R13 have been selected by all 3 methods

Penalized regression methods

Ridge regression

Ridge regression is a parsimonious model which performs L2 regularization. The L2 regularization adds a penality equivalent to the square of the maginitude of regression coefficients and tries to minimize them.Ridge regression does a proportional shrinkage and handles collinear variables but it does not perform a selection.We use the cv.glmnet() function available in the glmnet package to find the best λ. This function does k-fold cross-validation for glmnet, produces a plot, and returns a value for the best lambda value which is equal in our case to $\lambda_{ridge} = 0.1300923$. **The variables times, R2, R3, R5, R6, R8 have strongly penelized coefficients**. It could mean that those variables are not important to explain our target.

Lasso regression

Lasso regression find a parsimonious model which performs L1 regularization. The L1 regularization adds a penality equivalent to the absolute of the maginitude of regression coefficients and tries to minimize them Lasso translates each coefficient by a constant factor λ, truncating at zero. This is called "soft thresholding".We use the cv.glmnet.This function does k-fold cross-validation for glmnet, produces a plot, and returns a value for the best lambda value which is in our case $\lambda_{lasso} = 0.001314899$ When we performed Lasso regression **we see that the variables R3, R6 and R9 have been dropped**. The Lasso regression penalty term, using the absolute value (rather than the square, as in the regression Ridge), forces some coefficients to be exactly equal to zero, if λ is large enough. In practice, Lasso automatically performs a real selection of variables.

Elastic Net

ElasticNet is a hybrid of both Lasso and Ridge regression. It is trained with both L1-norm and L2-norm prior as regularizer. Like LASSO regularization it results in sparse solutions, however it also has the advantage of performing well with highly correlated variables like ridge regularization. Elastic net is used by solving the following optimization problem: $min_x \|y - Ax\| + \lambda_1 \|x\|_1 + \lambda_2 \|x\|_2$ Ultimately, we can say that ElasticNet balance the trade-off bias-variance with the choice of λ. It assumes that part of the coefficients are zero, or at least not significant. When computed, the Elastic Net regularization dropped the variables R3 and R6 and other variables like R4 and R9 have really penelized coefficients. As for the other regularized methods the function cv.glmnet compute a cross validation to determine the best lambda which is equal to $\lambda_{ElasticNet} = 0.002663069$

Group Lasso

In this experiment the sensors come from two specific fabricants: R1-R7 correspond to FIGARO TGS 3870 A-04 sensors, whereas R8-R14 correspond to FIS SB-500-12 units. The group Lasso method should be appropriate to decide which group give a best prediction of CO concentration.In 2006, Yuan and Lin introduced the group lasso in order to allow pre-defined groups of covariates to be selected **into or out of a model together**, so that all the members of a particular group are either included or excluded The objective function for the group lasso is a natural generalization of the standard lasso objective $\min_{\beta\in\mathbb{R}^p}\left\{\left\|y-\sum_{j=1}^{J}X_j\beta_j\right\|_2^2+\lambda\sum_{j=1}^{J}\|\beta_j\|_{K_j}\right\}$where the design matrix $X$and covariate vector β have been replaced by a collection of design matrices X_j and covariate vectors β_j, one for each of the J groups. Additionally, the penalty term is now a sum over $ł^{2}
$norms defined by the positive definite matrices K_j. If each covariate is in its own group and $K_j = I$, then this reduces to the standard lasso, while if there is only a single group and $K_1 = I$, it reduces to ridge regression.We use the function gglasso of the R package gglasso in order to proceed to group lasso regualrizatioon. We considered 7 groups. The first 5 groups corespond to the first five expalnatory variables (except of CO) the 6th group is composed of R1,...R7 and the 7th group of R8,...,R14.

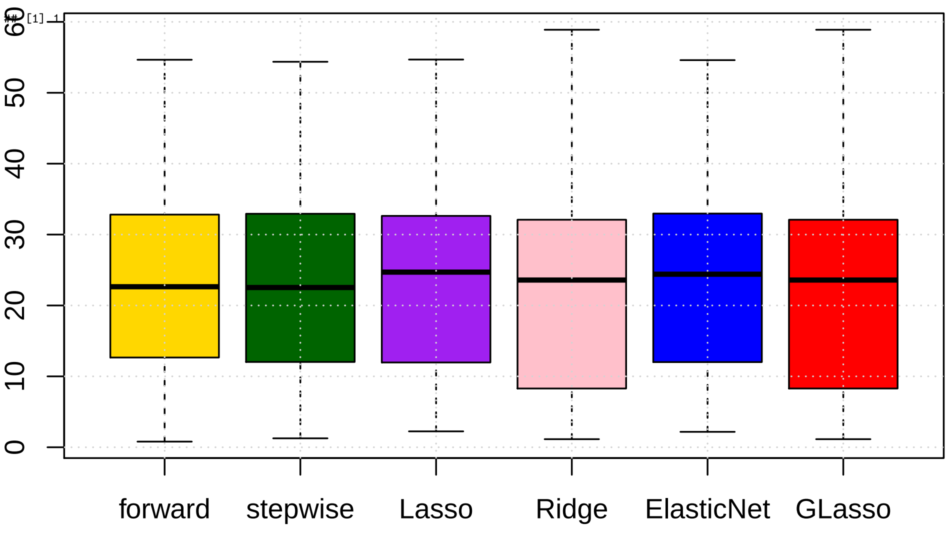
The group lasso regularization removed the group 2 (humidity),3 (temperature), 4(flow rate),5 (heater_voltage),and 6 (R1....R7).This could mean that the best sensors are in the group 7 (from the second fabricant)

Cross Validation in order to choose the best model

For each model,we split randomly the initial dataset in two dataframes containing 75% of the observations(The 'Training' data base) and 25% of the remaining observations(The 'Test' data set).We use the training data set to estimate the parameters of the model.Given the previous model,we use the test data set to compute the RMSE to evaluate the performances of the model.By repeating the two first steps 20 times,we compare the results obtained with the help of 6 boxplots.

```
1
```

```
## [1] 1
```



According to these boxplot it is hard to tell which model is the best. Indeed the boxplot are all quite the same and adding to that the computed RMSE foreach model s really high with respoct to the vales in our dataset.