Delete or Merge Regressors algorithm

Agnieszka Prochenka [*1] **and Piotr Pokarowski**[1]

[1] *Faculty of Mathematics, Informatics and Mechanics, University of Warsaw*

This paper addresses a problem of linear and logistic model selection in the presence of both continuous and categorical predictors. In the literature two types of algorithms dealing with this problem can be found. The first one is the well known `group lasso` ([3]) which selects a subset of continuous and a subset of categorical predictors. Hence, it either deletes or not an entire factor. An improvement of the `group lasso` regularization is `group MCP` (using Minimax Concave Penalty) described in [6]. It assumes a concave penalty and therefore uses more difficult optimization algorithms. The second type is `CAS-ANOVA` ([1]) which selects a subset of continuous predictors and partitions of factors. Therefore, it merges levels within factors. Similar method with different optimization method is called `gvcm` and is described in [5].

In the article an algorithm called `DMR` (Delete or Merge Regressors) is described. Like `CAS-ANOVA` it selects a subset of continuous predictors and partitions of factors. However, instead of using regularization, it is based on a stepwise procedure, where in each step either one continuous variable is deleted or two levels of a factor are merged. The order of accepting consecutive hypotheses is based on sorting Wald statistics. Some of the preliminary results for `DMR` are described in [2].

`DMR` algorithm works only for data sets where $p < n$ (number of columns in the model matrix is smaller than the number of observations). In the paper a modification of `DMR` called `DMRnet` is introduced that works also for data sets where $p \gg n$. `DMRnet` uses regularization in the screening step and `DMR after` decreasing the model matrix to $p < n$.

Theoretical results prove that `DMR` for linear and logistic regression are consistent model selection methods even when p tends to infinity with n. Furthermore, upper bounds on the error of selection were calculated. However, in this paper the focus is on description of the algorithm and real data example, for which `DMRnet` chooses smaller models with not higher prediction error than the competitive methods.

Keywords: factorial selection, logistic regression, linear regression, Wald statistics, hierarchical clustering

*Corresponding author: a.prochenka@phd.ipipan.waw.pl

1 Factorial selection

We consider n data points $(y_1, \mathbf{x}_{1\cdot}^T)$, $(y_2, \mathbf{x}_{2\cdot}^T)$, ..., $(y_n, \mathbf{x}_{n\cdot}^T)$ with univariate responses y_i and p-dimensional covariates $\mathbf{x}_{i\cdot}^T$. Denote by $\mathbf{y} = (y_1, y_2, \ldots, y_n)^T$, $\mathbf{x}_j = (x_{1j}, x_{2j}, \ldots, x_{nj})^T$, $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p)$ the n times p model matrix. We assume that $\mathbf{X}$ is a full rank matrix.

Let y_i be independent, such that $y_i \sim f_{\eta_i,\sigma^2}(\cdot)$ and $\eta_i = \mathbf{x}_{i\cdot}^T\boldsymbol{\beta}$ for $\boldsymbol{\beta} \in \mathbb{R}^p$, where f_{η_i,σ^2} is the density function of some distribution in the exponential family. Let us denote $\boldsymbol{\eta} = (\eta_1, \eta_2, \ldots, \eta_n)^T$ and

$$\boldsymbol{\eta}^* = \mathbf{X}\boldsymbol{\beta}^* = \mathbf{1}\beta_{00}^* + \mathbf{X}_0\boldsymbol{\beta}_0^* + \mathbf{X}_1\boldsymbol{\beta}_1^* + \ldots + \mathbf{X}_l\boldsymbol{\beta}_l^*, \tag{1}$$

where

1. $\mathbf{X} = [\mathbf{1}, \mathbf{X}_0, \mathbf{X}_1, \ldots, \mathbf{X}_l]$ is a model matrix organized as follows: $\mathbf{X}_0$ is a matrix corresponding to continuous regressors and $\mathbf{X}_1, \ldots, \mathbf{X}_l$ are zero-one matrices encoding corresponding factors with the first level set as the reference.

2. $\boldsymbol{\beta}^* = [\beta_{00}^*, \boldsymbol{\beta}_0^{*T}, \boldsymbol{\beta}_1^{*T}, \ldots, \boldsymbol{\beta}_l^{*T}]^T \in \mathbb{R}^p$ is a parameter vector organized as follows: β_{00}^* is the intercept, $\boldsymbol{\beta}_0^* = [\beta_{10}^*, \ldots, \beta_{p_0 0}^*]^T$ is a vector of coefficients for continuous variables and $\boldsymbol{\beta}_k^* = [\beta_{2k}^*, \ldots, \beta_{p_k k}^*]^T$ is a vector of parameters corresponding to the k-th factor, $k = 1, \ldots, l$, hence the length of the parameter vector is $p = 1 + p_0 + (p_1 - 1) + \ldots + (p_l - 1)$.

Denote sets of indexes: $N = \{0, 1, \ldots, l\}$, $N_0 = \{0, 1, \ldots, p_0\}$ and $N_k = \{2, 3, \ldots, p_k\}$ for $k \in N \setminus \{0\}$. Let us define an elementary constraint for model (1) as a linear constraint of one of two types:

$$\mathcal{H}_{jk} : \ \beta_{jk}^* = 0 \text{ where } j \in N_k \setminus \{0\}, \ k \in N, \tag{2}$$

$$\mathcal{H}_{ijk} : \ \beta_{ik}^* = \beta_{jk}^* \text{ where } i, j \in N_k, \ i \neq j, \ k \in N \setminus \{0\}. \tag{3}$$

1.1 Feasible models

A feasible model can be defined as a sequence $M = (P_0, P_1, ..., P_l)$, where P_0 denotes a subset of indexes of continuous variables and P_k is a particular partition of levels of the k-th factor. Such a model can be encoded by a set of elementary constraints. A set of all feasible models is denoted by $\mathcal{M}$. Let us denote a model $F \in \mathcal{M}$ without constraints of types (2) or (3) as the full model.

Example. For illustration, let us consider a linear predictor with one factor and one continuous variable:

$$\mathbf{X}\boldsymbol{\beta}^* = \mathbf{1}\cdot 1 + \mathbf{X}_0 \cdot 2 + \mathbf{X}_1 \cdot \begin{bmatrix} -2 \\ -2 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \cdot 1 + \begin{bmatrix} -0.96 \\ -0.29 \\ 0.26 \\ -1.15 \\ 0.2 \\ 0.03 \\ 0.09 \\ 1.12 \end{bmatrix} \cdot 2 + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ -2 \\ 0 \end{bmatrix}$$

Then $\beta^* = [1, 2, -2, -2, 0]^T$. The full model $F = (P_0 = \{1\}, P_1 = \{\{1\}, \{2\}, \{3\}, \{4\}\})$ with $p_0 = 1, p_1 = 4, p = 5$. The true model is $(P_0 = \{1\}, P_1 = \{\{1, 4\}, \{2, 3\}\})$ and is the same as the full model with two elementary constraints: $\beta^*_{41} = 0$ and $\beta^*_{21} = \beta^*_{31}$.

Our goal is to find the best feasible model according to Generalized Information Criterion (GIC) or estimated prediction error using cross-validation, taking into account that the number of feasible models grows faster than exponentially with p. In order to significantly reduce the amount of computations, we propose a greedy backward search.

2 `DMR` and `DMRnet` algorithms

`DMR` for generalized linear models is described in details in Algorithm 1. `DMRnet` is a generalization of `DMR` to high-dimensional data where $p \gg n$ by adding screening step using `group lasso`. After reduction of the dimension of the model to $p < n$, `DMR` algorithm is used. In order to make the screening step more accurate and to better balance the impact of screening and the `DMR` selection steps, the screening is done multiple times.

Example. Example 1.1 continued, `DMR` algorithm with GIC:

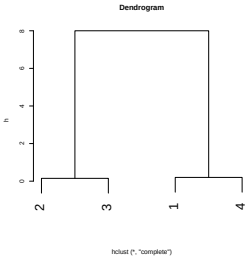
$$w^2_{110} = 9.35, \mathbf{D_1} = \begin{bmatrix} 0 & w^2_{121} & w^2_{131} & w^2_{141} \\ w^2_{121} & 0 & w^2_{231} & w^2_{241} \\ w^2_{131} & w^2_{231} & 0 & w^2_{341} \\ w^2_{141} & w^2_{241} & w^2_{341} & 0 \end{bmatrix} = \begin{bmatrix} 0 & 8.01 & 4.52 & 0.20 \\ 8.01 & 0 & 0.15 & 3.09 \\ 4.52 & 0.15 & 0 & 2.91 \\ 0.20 & 3.09 & 2.91 & 0 \end{bmatrix},$$

cutting heights for agglomerative clustering illustrated in Figure 1:

$$\mathbf{h} = [0, 0.15, 0.20, 8.01, 9.35]^T,\ \mathbf{A_0} = \begin{array}{c} \\ \\ \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccc} \beta_{00} & \beta_{10} & \beta_{21} & \beta_{31} & \beta_{41} \end{array} \\ \begin{bmatrix} 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \end{array},$$

$\mathbf{GIC} = [28.33, 26.65, 25.36, 34.68, 39.59]^T$. The selected model according to GIC is the third one (GIC = 25.36) with two elementary constraints: $\beta^*_{41} = 0$ and $\beta^*_{21} = \beta^*_{31}$, which is the true model.

Figure 1: Dendrogram for hierarchical clustering used in Example 1.1.



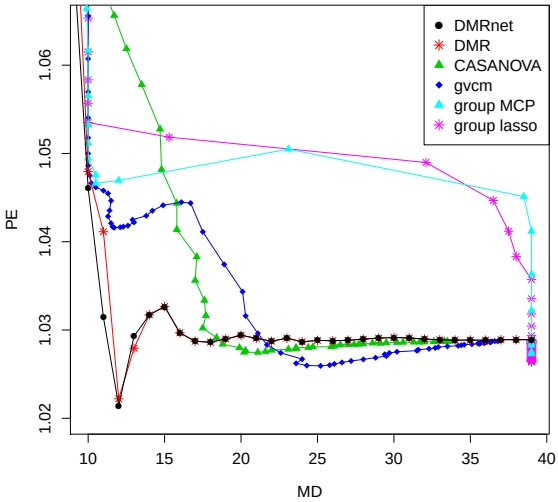
3 Real data example: Miete

The data set Miete comes from `http://www.statistik.lmu.de/service/datenarchiv`. The data consists of $n = 2053$ households interviewed for the Munich rent standard 2003. The response is monthly rent per square meter in Euros, data is described in detail in [4]. 8 categorical and 2 continuous variables give 36 and 3 (including the intercept) parameters. This gives $p = 39$.

In Figure 2 a plot of prediction error (PE) vs model dimension (MD) calculated by 10-fold C-V for 100 λ values for `CAS-ANOVA`, `gvcm`, `group MCP` and `group lasso` and from 1 to p for `DMR` and from 1 to $\min\{p, \frac{n}{2}\}$ for `DMRnet` is shown. For every algorithm we can find a global minimum: for `DMRnet` and `DMR` these are when MD = 12, for `CAS-ANOVA` when MD = 21.1, for `gvcm` when MD = 25.2 and for `group MCP` and `group lasso` for the full model, MD=39. If we chose models with the lowest prediction error, `DMRnet` would have both the smallest error and the smallest number of parameters.

Acknowledgements: The research is supported by the Polish National Science Center grant 2015/17/B/ST6/01878.

Figure 2: PE vs MD calculated by 10-fold C-V for Miete data set.



References

[1] Bondell, Howard D., and Brian J. Reich. "Simultaneous factor selection and collapsing levels in ANOVA." Biometrics 65.1 (2009): 169-177.

[2] Maj-Kańska, Aleksandra, Piotr Pokarowski, and Agnieszka Prochenka. "Delete or merge regressors for linear model selection." Electronic Journal of Statistics 9.2 (2015): 1749-1778.

[3] Yuan, Ming, and Yi Lin. "Model selection and estimation in regression with grouped variables." Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68.1 (2006): 49-67.

[4] Tutz, Gerhard. Regression for categorical data. Vol. 34. Cambridge University Press, 2011.

[5] Oelker, Margret-Ruth, Jan Gertheiss, and Gerhard Tutz. "Regularization and model selection with categorical predictors and effect modifiers in generalized linear models." Statistical Modelling 14.2 (2014): 157-177.

[6] Breheny, Patrick, and Jian Huang. "Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors." Statistics and computing 25.2 (2015): 173-187.

Algorithm 1: DMR (Delete or Merge Regressors for generalized linear models)

Input: $\mathbf{y}$, $\mathbf{X}$

1. Computation of Wald statistics

Calculate Wald statistics for all elementary constraints defined in (2):

for $j \in N_k \setminus \{0\},\ k \in N$ **do**

$$w_{1jk}^2 = \frac{\widehat{\beta}_{jk}^2}{\widehat{Var}(\widehat{\beta}_{jk})}$$

end for

Calculate Wald statistics for all elementary constraints defined in (3):

for $i, j \in N_k,\ i \neq j,\ k \in N \setminus \{0\}$ **do**

$$w_{ijk}^2 = \frac{(\widehat{\beta}_{ik} - \widehat{\beta}_{jk})^2}{\widehat{Var}(\widehat{\beta}_{ik} - \widehat{\beta}_{jk})}$$

end for

2. Agglomerative clustering for factors (using complete linkage clustering)

For each factor perform agglomerative clustering using $\mathbf{D}_k = [d_{ijk}]_{ij}$ as dissimilarity matrix.

for $k \in N \setminus \{0\}$ **do**

$d_{1jk} = d_{j1k} = w_{1jk}$ for $j \in N_k$,

$d_{ijk} = w_{ijk}$ for $i, j \in N_k,\ i \neq j$,

$d_{iik} = 0$ for $i \in N_k$.

end for

Denote cutting heights obtained from the clusterings of l factors as $\mathbf{h}_1^T, \mathbf{h}_2^T, \ldots, \mathbf{h}_l^T$.

3. Sorting constraints (hypotheses) according to the likelihood ratio test statistics

Combine vectors of cutting heights: $\mathbf{h} = [0, \mathbf{h}_0^T, \mathbf{h}_1^T, \ldots, \mathbf{h}_l^T]^T$, where $\mathbf{h}_0$ is a vector of likelihood ratio test statistics for constraints concerning continuous variables and 0 corresponds to the full model. Sort elements of $\mathbf{h}$ in increasing order and construct a corresponding $(p-1) \times p$ matrix $\mathbf{A}_0$ of consecutive constraints.

4. Computation of log-likelihood for models on the nested path

for $m = 0, \ldots, p-1$ **do**

$L_{M_m} = \ell(\widehat{\boldsymbol{\beta}}_{M_m})$, where M_m is the model with m first constraints from A_0 accepted.

end for

Output: $\mathscr{M}^{\texttt{DMR}} = \{M_0, \ldots, M_{p-1}\}$, $\mathbf{L}^{\texttt{DMR}} = (L_{M_0}, \ldots, L_{M_{p-1}})^T$.